



ZÁPADOČESKÁ  
UNIVERZITA  
V PLZNI

Západočeská univerzita v Plzni  
Fakulta aplikovaných věd  
Katedra matematiky

## **Diplomová práce**

# **Matematické nástroje analýzy v geomatice**

Plzeň, 2006

Jaroslava Nováčková

## **Prohlášení**

Předkládám k obhajobě tuto diplomovou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem diplomovou práci vypracovala zcela samostatně a za odborného vedení vedoucího diplomové práce. Všechny informační zdroje, ze kterých jsem čerpala, jsou uvedeny v úplném seznamu použité literatury, který je součástí diplomové práce.

V Plzni dne 24.5.2006

---

Ing. Jaroslava Nováčková

## **Poděkování**

Na tomto místě bych ráda poděkovala vedoucímu diplomové práce panu Doc. RNDr. Františku Ježkovi, CSc., zejména za odborné konzultace a čas, po který se mi věnoval, a za pomoc a odborné vedení při zpracování této diplomové práce.

Dále pak Doc. Dr. Ing. Ivaně Kolingerové za poskytnutí cenných rad, podkladů, zkušeností a znalostí.

## **Anotace**

Diplomová práce je zaměřena na matematické nástroje pro analýzy v geomatice. Popisují datové modely a datové struktury, které se používají v GIS a uvádím přehled analytických funkcí GIS. Velká pozornost věnuji teorii Voronoiových diagramů a jejich zobecnění. V praktické části předvádím využití vybrané analytické metody.

## **Abstract**

Focus of diploma thesis is concentrated on mathematical analysis tools in geomatics. I describe data models and data structures used in GIS and I present a review of analytical GIS functions. My attention is paid to the Voronoi diagram theory and generalized Voronoi diagrams. In practical part I show the chosen method from analytical GIS functions.

## **Klíčová slova**

Datové modely v GIS, datové struktury v GIS, analytické funkce v GIS, Voronoiův diagram, zobecněný Voronoiův diagram

## **Key words**

Data models in GIS, data structures in GIS, analytical functions in GIS, Voronoi diagram, generalized Voronoi diagram

# Obsah

1	Úvod.....	8
2	Přehled datových modelů a struktur používaných v GIS.....	10
2.1	Geografické datové modely a geografická datová primitiva .....	11
2.1.1	Vektorové datové modely entit .....	11
2.1.1.1	Jednoduché body, linie a polygony .....	12
2.1.1.2	Komplexní (složitě) body, linie, polygony a objekty s funkcionalitou.....	12
2.1.2	Tesselace souvislých polí .....	13
2.1.3	Pixely a voxely jako entity .....	14
2.1.4	Axiomy a procedury pro manipulaci s daty v informačních systémech .....	15
2.1.5	Datové modelování a prostorová analýza .....	15
2.2	Databázové struktury a DBMS .....	16
2.2.1	Hierarchická databázová struktura .....	16
2.2.2	Síťová databázová struktura .....	18
2.2.3	Relační databázová struktura .....	18
2.2.4	Objektově-orientovaná databázová struktura.....	20
2.2.5	Deduktivní databázový přístup.....	22
2.3	Datové struktury pro reprezentaci geometrie prostorových jevů.....	23
2.3.1	Rastrové datové struktury.....	23
2.3.1.1	Komprimovací metody pro uložení rastrových dat.....	25
2.3.1.2	Dvojdímenzionální řazení .....	30
2.3.2	Vektorové datové struktury .....	31
2.3.2.1	Vektorové datové struktury pro uložení vektorových dat.....	32
2.3.2.2	Vektorové datové struktury ke speciálnímu účelu – Triangular Irregular Network (TIN) .....	39
2.3.2.3	Vývoj vektorových datových struktur za účelem zlepšení časového přístupu k datům a úspornějšího uložení.....	40
2.3.3	Databázové uložení vektorových a rastrových datových struktur .....	45
2.3.3.1	ARC-NODE – RDBMS .....	45
2.3.3.2	Kompaktní rastr – RDBMS.....	46
2.3.3.3	Quadtree – RDBMS .....	46
2.3.3.4	Objekty – RDBMS .....	47
2.3.4	Objektově orientované databázové struktury – sjednocení uložení atributů a geometrií .....	47
3	Další matematické nástroje pro analýzy v GIS .....	49
3.1	Rozdělení analytických funkcí GIS.....	49
3.1.1	Statistické analýzy.....	51
3.1.1.1	Klasické statistické metody .....	51
3.1.1.2	Grafy.....	51
3.1.1.3	Regresní analýzy .....	51
3.1.1.4	Přímé napojení na statistické programy .....	52
3.1.2	Analýzy modelů terénu (povrchů).....	52
3.1.2.1	Povrch sklonitosti svahu ( <i>slope</i> ) .....	52
3.1.2.2	Základní morfologické analýzy.....	53
3.1.2.3	Analýza osvětlení reliéfu.....	54
3.1.2.4	Tvorba vrstevnic (izočar) .....	54
3.1.2.5	Generování profilů .....	56
3.1.2.6	Počítání objemů DMR.....	56
3.1.2.7	Analýza viditelnosti (z bodu, z linie) .....	56

3.1.2.8	Identifikace významných bodů a linií (hřebenů, údolnic), rozpoznání geomorfologických tvarů .....	57
3.1.2.9	Modelování eroze .....	57
3.1.2.10	Modelování směrů proudění .....	57
3.1.3	Analýzy obrazů .....	58
3.1.3.1	Filtrace, roztažení histogramu .....	58
3.1.3.2	Vyrovnaní jasu/kontrastu mezi snímky, mozaikování apod. ....	58
3.1.3.3	Metody klasifikace obrazů .....	59
3.1.4	Měřicí funkce .....	60
3.1.5	Nástroje na dotazování databáze .....	60
3.1.5.1	Atributové dotazy .....	60
3.1.5.2	Prostorové dotazy .....	61
3.1.5.3	Kombinované dotazy .....	61
3.1.6	Topologické překrytí .....	61
3.1.7	Mapová algebra .....	62
3.1.8	Vzdálenostní analýzy .....	62
3.1.8.1	Buffer .....	62
3.1.8.2	Analýzy sousedství (proximity analysis) .....	63
3.1.8.3	Vážená vzdálenost (weighted distance) .....	63
3.1.8.4	Nalezení nejlevnější cesty .....	64
3.1.9	Analýzy sítí .....	64
3.1.9.1	Hledání konektivity .....	64
3.1.9.2	Modelování zatížení sítě .....	64
3.1.9.3	Hledání optimální trasy .....	64
3.1.9.4	Hledání cesty do nejbližšího zařízení .....	65
3.1.9.5	Alokace zdrojů .....	65
4	Voronoiovy diagramy .....	66
4.1	Základní vlastnosti Voronoiových diagramů .....	66
4.2	Zobecněné Voronoiovy algoritmy .....	67
4.2.1	Vyšší dimenze .....	67
4.2.2	Jiná metrika .....	67
4.2.3	Použití vah .....	68
4.2.3.1	Aditivní (vážené) váhy .....	68
4.2.3.2	Multiplikativní váhy .....	68
4.2.3.3	Power diagram .....	69
4.2.4	Jiné tvary míst .....	69
4.2.4.1	Hrany polygonu .....	69
4.2.4.2	Úsečky .....	70
4.2.5	Změna prostoru .....	71
4.2.6	Dynamizace .....	71
4.2.6.1	Pohyb míst .....	71
4.2.6.2	Vkládání a vybírání bodu/linie .....	72
4.2.7	$k$ nejbližších míst .....	72
4.3	Algoritmy pro tvorbu Voronoiových diagramů .....	73
4.3.1	Protínání polorovin .....	73
4.3.2	Inkrementální (přírůstková) konstrukce .....	73
4.3.3	Algoritmus „Rozděl a panuj“ .....	74
4.3.4	Fortuneho algoritmus (technika plane-sweep) .....	75
4.3.5	Konvexní obálka .....	75
4.3.6	Aproximace pomocí bodů .....	76

4.3.7	Aproximace digitálním obrazem.....	77
4.4	Aplikace Voronoiových diagramů v GIS.....	77
4.4.1	Nalezení nejbližšího souseda (problém poštovního úřadu, lokace bodu).....	77
4.4.1.1	Hledání všech nejbližších sousedů.....	77
4.4.1.2	Hledání nejbližší dvojice.....	77
4.4.2	Největší prázdný kruh (možnost umístění).....	77
4.4.3	Nejmenší kružnice obsahující všechny body.....	78
4.4.4	Minimal spanning tree (MST).....	78
4.4.5	Případ obchodního cestujícího.....	79
4.4.6	Střední osa (middle axis).....	80
4.4.7	Konvexní obálka.....	80
4.4.8	Skeleton (kostra).....	81
4.4.9	Centroid.....	81
4.4.10	Hranice (frontier).....	82
4.4.11	Roztažení (dilation).....	82
4.4.12	Stažení (erosion).....	83
4.4.13	Uzavření (closure).....	84
4.4.14	Otevření (opening).....	84
4.4.15	Hausdorffova vzdálenost mezi liniemi.....	85
4.4.16	Hausdorffova vzdálenost mezi oblastmi.....	85
4.4.17	Geometrické vodní izočáry.....	86
4.5	Přínos Voronoiových diagramů.....	87
4.5.1	Situace v minulosti.....	87
4.5.2	Vznikající požadavky na datové struktury prostorových dat.....	87
4.5.3	Výhody Voronoiových diagramů.....	88
5	Použití vybrané analytické funkce s konkrétními daty ve zvoleném prostředí GIS.....	90
5.1	Výběr vhodných testovacích dat.....	90
5.1.1	Reálná data.....	90
5.2	Výběr programového vybavení.....	91
5.2.1	Zpracování dat.....	91
5.2.2	Vizualizace vypočítaných dat.....	91
5.3	Příprava dat k triangulaci.....	92
5.4	Možné kombinace pro vytvoření TIN.....	93
5.4.1	Přímo z vrstevnic.....	93
5.4.2	Z bodového vektorového souboru.....	94
5.4.3	Z liniového vektorového souboru.....	94
5.5	Porovnání s referenčním souborem.....	95
5.6	Zhodnocení získaných výsledků.....	95
6	Závěr.....	97
	Seznam použité literatury.....	98
	Seznam obrázků.....	101
	Seznam příloh.....	103
	Přílohy.....	104

# 1 Úvod

Díky rozvoji informačních technologií vznikl nový vědní obor geomatika, kterou lze chápat jako vědeckou disciplínu, která zkoumá přírodní a socioekonomické geosystémy (jejich strukturu, interakci, dynamiku, fungování v časoprostoru) pomocí počítačového modelování na základě databází a geografických poznatků. Předmětem geoinformatiky jsou prostorové údaje a tato vědecká a technická disciplína se soustřeďuje na jejich sběr, uložení, manipulaci s nimi a jejich zprostředkovávání.

Pro označení geograficky orientované počítačové technologie, integrovaných systémů pro různé aplikace, jakož i nové disciplíny, která se poslední dobou velmi rychle vyvíjí a rozšiřuje se používá pojem geografický informační systém (GIS). Lze jej chápat jako technologii, jako aplikační nástroj a jako vědecký obor. GIS mají klíčovou úlohu v geoinformatice, geo- a environmentálních vědách a prakticky ve všech oblastech monitorování, plánování a řízení životního a přírodního prostředí. Integrace všech metod, modelů a systémů s GIS a interdisciplinární přístup přináší s sebou celou řadu problémů a je nyní hlavní oblastí výzkumu a vývoje v nejbližším období.

Na rozdíl od jiných druhů dat používaných v moderních informačních systémech geografická data jsou komplikovanější, protože musí zahrnovat informaci o pozici, možných topologických spojeních a attributech zaznamenaných objektů. Topologické a prostorové aspekty zpracování geografických dat odlišují systémy navržené pro grafiku a mapování od ostatních systémů používaných v bankovníctví, knihovnách, při objednávání na letišti nebo lékařských záznamech. Ačkoli jsou geografická data popsána vědci jako explicitní objekty, jejich přesná forma a rozsah je diskutabilní nebo se s časem mění. Aby byl použitý algoritmus efektivní jak z hlediska časové náročnosti, tak z hlediska potřebného paměťového prostoru, je nutné sladění struktury údajů a způsobu přístupu k nim podle potřeb algoritmu tzn. vhodná datová reprezentace.

Protože se GIS zabývají vlastnostmi bodů a oblastí na zemském povrchu, objevují se hojně geometrické problémy. Výpočetní geometrie studuje vlastnosti algoritmů pro řešení geometrických problémů z více hledisek – z hlediska existence jakéhokoli algoritmu pro řešení problému, z hlediska efektivnosti algoritmů, z hlediska možnosti strukturování údajů pro potřeby těchto algoritmů. Počítačová geometrie vznikla z oboru návrh a analýza algoritmů v pozdějších 70. letech. Úspěch oboru jako výzkumné disciplíny na jedné straně může být vysvětlen z hlediska studovaných problémů a získaných řešení a na straně druhé množstvím aplikačních oblastí –



počítačová grafika, GIS, robotika, CAD/CAM a jiné – ve kterých geometrické algoritmy hrají podstatnou roli. Pro mnoho geometrických problémů dřívější algoritmická řešení byla buď pomalá nebo obtížná na porozumění a implementaci. V nedávných letech se vyvinulo mnoho nových algoritmických technik, které zdokonalují a zjednodušují mnoho předchozích přístupů.

GIS se stále více stávají běžnou součástí informačních systémů ve všech těchto oblastech - obchod (analýzy nalezení nejvhodnější lokality pro nový obchod, síťové analýzy rozvozu zboží), ochrana proti pohromám (zjištění aktuální situace při pohromě, modely povodní, směřování záchranných prostředků), distribuční společnosti (databáze a analýza sítí, směřování v sítích), životní prostředí (studium ekosystémů, modely znečišťování ovzduší), ochrana přírody, ekologické studie a syntézy, krajinné inženýrství, geomorfologické analýzy, zemědělství a lesnictví, geologie, hornictví, těžba surovin a využívání přírodních zdrojů, hydrologie, meliorace, regulace toků, státní správa, správa katastru, městské úřady, doprava, zdravotnictví a v neposlední řadě i školy.

## 2 Přehled datových modelů a struktur používaných v GIS

Přehled bude podán podle nejnovějších informací uvedených ve zdroji [9]. Datové modely jsou formální abstrakcí reality. Pozorovatel prostředím zjednodušuje jeho nedělitelnou složitost abstrakcí klíčových rysů, pomocí nichž vytváří model území. Tento postup je ovlivněn kulturními normami pozorovatele a účelem pozorování. Dále budou uvedena stádia vývoje různých modelů.

Dva extrémní v přístupu chápou prostor buď jako řadu entit, které jsou popsány jejich vlastnostmi a mapovány použitím souřadnicového systému, nebo jako souvislé pole měnících se hodnot bez odlišení hranic. Aby k těmto konceptuálním modelům byly přiřazeny jednotky, které mohou být zaznamenány a mapovány, jsou použity formalizované geografické datové modely. Pak se k popsání různých rysů krajiny používají buď řady bodů, linií a polygonů, nebo jednotky tesselačí. Použití modelu pak ovlivňuje typ dat, která mohou být použita k popsání jevu, a prostorovou analýzu, kterou lze použít.

### Prostorové datové modely a datové struktury

Vytvoření datových sad s analogovými a digitálními prostorovými daty zahrnuje sedm úrovní vývoje modelů a abstrakcí:

- (a) Pohled na realitu (*konceptuální model*)
- (b) Vytváření pojmů, které vede k analogové abstrakci (*analogový model*)
- (c) Formalizace analogových abstrakcí bez jakýchkoli omezení a konvencí v implementaci (*prostorový datový model*)
- (d) Reprezentace datového modelu, která odráží, jak jsou data zaznamenána v počítači (*databázový model*)
- (e) Struktura souboru, která je zvláštní reprezentací datových struktur v paměti počítače (*fyzický počítačový model*)
- (f) Přijaté axiomy a pravidla pro zacházení s daty (*model manipulace s daty*)
- (g) Přijatá pravidla a procedury pro zobrazování a prezentaci prostorových dat lidem (*grafický model*)

Úspěch nebo neúspěch GIS nezávisí především na technologii, ale daleko více na konceptuálních modelech prostoru a prostorových vzájemných vlivech.

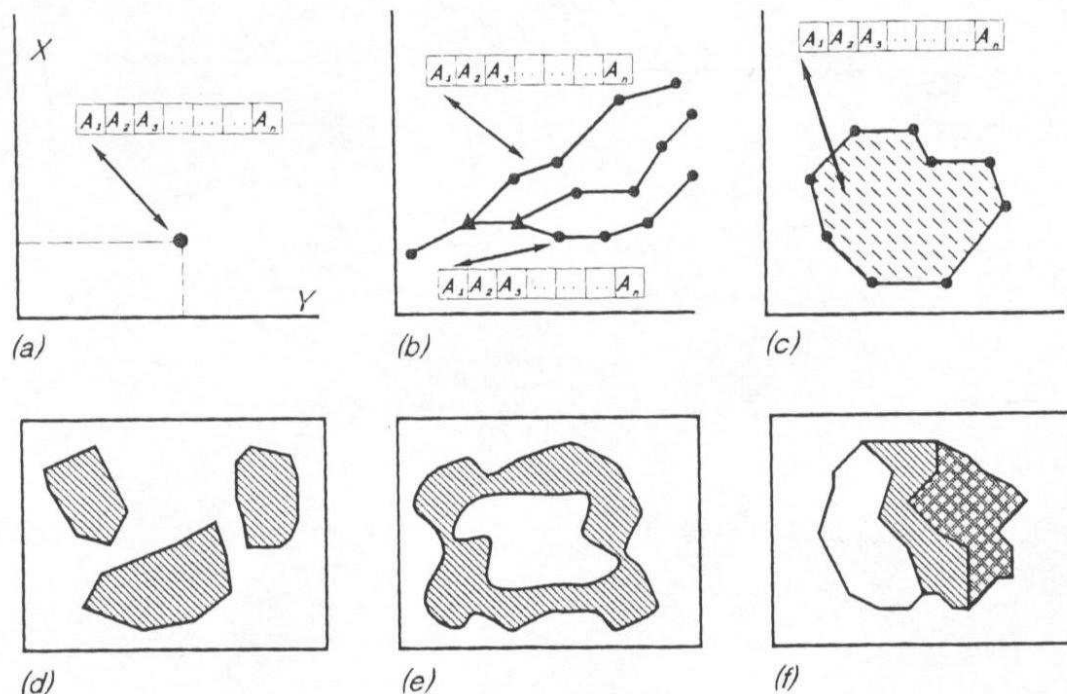
## 2.1 Geografické datové modely a geografická datová primitiva

Geografické datové modely jsou formalizované ekvivalenty konceptuálních modelů a lidé je používají, aby pochopili geografické fenomény. Formalizují rozdělení prostoru na části kvůli analýze a komunikaci a předpokládají, že fenomény jsou unikátně identifikovány, že atributy se dají měřit nebo specifikovat a že zeměpisné souřadnice se registrují. Důležitou součástí datového modelu je informace o metodě nebo stupni rozlišení pozorování nebo měření. Nejjednodušší a nejčastěji používaný datový model je základní prostorová entita, která je dále specifikována atributy a zeměpisným umístěním. Ta může být dále rozdělena na tři základní geografická datová primitiva – „bod“, „linie“, „plocha“ nebo také „polygon“ viz. obr. 1. Alternativní prostředky pro reprezentaci entit pomocí mozaiky z pravidelných polygonů jsou sady pixlů.

### 2.1.1 Vektorové datové modely entit

Vektorové datové modely reprezentují prostor jako řadu diskrétních entit – definovaných bodů, linií nebo polygonů, které mají odkaz na kartézské souřadnice.

Data Models and Axioms



**Obrázek 1** Základní geografická primitiva (body, linie, polygony) (převzato z [9])

### **2.1.1.1 Jednoduché body, linie a polygony**

Jsou to základní statické reprezentace jevů v souřadnicích XY. Předpokládá se, že se nemění a neobsahují informaci o časové nebo prostorové variabilitě.

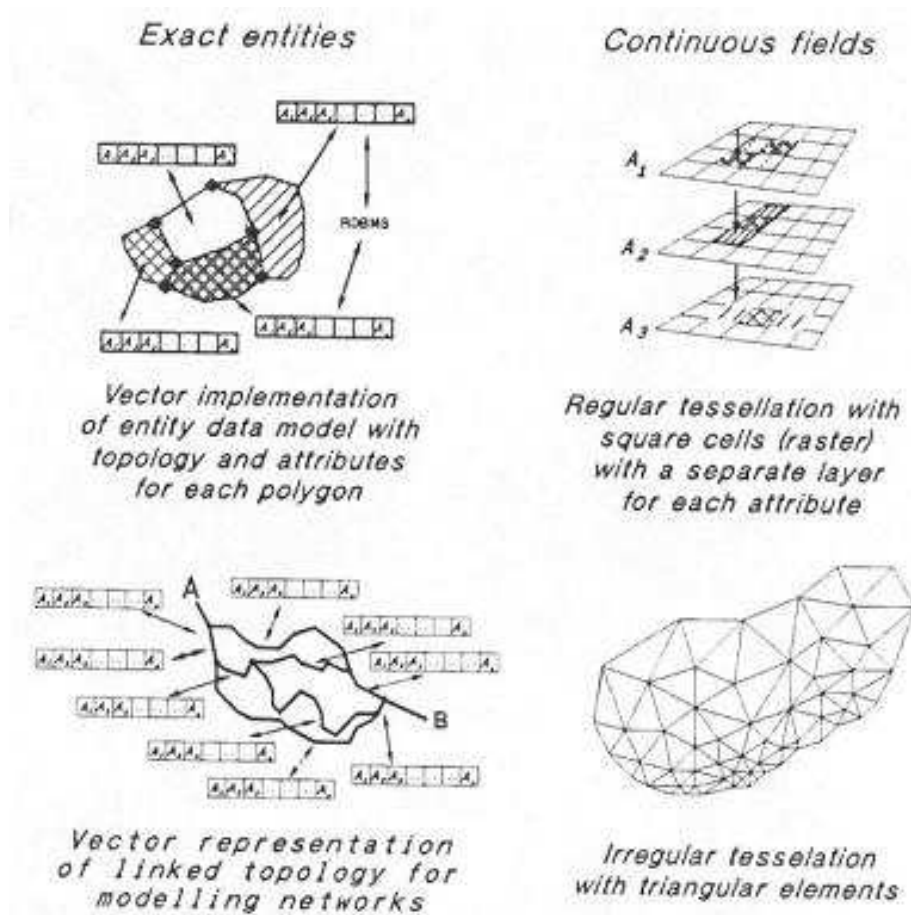
Bodem rozumíme polohu danou jedinou sadou XY souřadnic (město může být reprezentováno bodem na kontinentální úrovni, ale jako polygon na úrovni regionální).

Linií rozumíme zeměpisnou rozlohu objektu je reprezentována páry XY souřadnic, které definují propojenou cestu skrze prostor, která ale nemá žádnou tloušťku (cesta na národní úrovni je reprezentována linií, ale na úrovni ulic se již uvažuje jako plocha i s chodníky).

Polygonem rozumíme homogenní reprezentaci 2D prostoru, která závisí na stupni rozlišení. Může být reprezentován XY souřadnicemi jeho hranice nebo sadou XY souřadnic, které jsou uzavřeny takovou hranicí. Mohou obsahovat díry. Velice často je stupeň rozlišení dán počtem desetinných míst, na které jsou specifikovány souřadnice. Proto je úspornější používání hranic než bodů uvnitř obálky dané hranicí.

### **2.1.1.2 Komplexní (složitě) body, linie, polygony a objekty s funkcionalitou**

V tomto případě jde o zachycení vnitřní struktury entity. Tyto definice mohou být funkční nebo popisné (město zahrnuje ulice, domy a parky s odlišnými funkcemi). Topologické linky ukazují, jak jsou linie spojeny do polygonů nebo sítí viz obr. 2 a), c). Toto je možné pomocí objektově orientované databáze a programování.



**Obrázek 2** Kódování objektů (entit) a souvislých polí v různých datových modelech. a) vektorová reprezentace polygonu; b) rastrový model souvislého pole; c) vektorová reprezentace spojených linií; d) Delaunayova triangulace souvislého pole (převzato z [9])

### 2.1.2 Tesselace souvislých polí

Souvislé povrchy mohou být rozděleny do sady jednoduchých základních jednotek jako jsou čtvercové, trojúhelníkové nebo šestiúhelníkové buňky nebo do nepravidelných trojúhelníků nebo polygonů (Thiessenovy, Dirichletovy, Voronoiovy), jejichž mozaika tvoří geografickou reprezentaci. Použití nepravidelných trojúhelníků (používaných v geodézii) je založeno na principu triangulace, kdy souvislý povrch Země je nahrazen sítí trojúhelníků, jejichž vrcholy nebo uzly jsou naměřené výšky v trigonometrických bodech viz obr. 2 d). Velkou výhodou je, že hustota sítě je přizpůsobena variabilitě povrchu – plochy s malou variabilitou mohou být reprezentovány menším počtem trojúhelníků, zatímco plochy s velkou variabilitou potřebují větší počet trojúhelníků. Tyto datové modely jsou v podstatě statické reprezentace hypsometrických povrchů (povrchů nadmořských výšek), u nichž se předpokládá, že se s časem nebudou měnit.

Nejpoužívanější alternativou k triangulaci je pravidelná mozaika nebo pravidelná mříž. 2D geometrický povrch je rozdělen na čtvercové buňky (*pixely*), jejichž velikost je určena rozlišením, které je nutné, aby se zachytila změna atributu. Mřížková reprezentace je známa jako *rastrový model* viz obr. 2 b).

Ačkoli pravidelná mřížka nejčastěji reprezentuje statické jevy, může být použita i pro dynamickou změnu. Změny v čase mohou být zaznamenány v jednotlivých vrstvách, každá vrstva (mřížka) pro jeden časový okamžik. Základní struktura se pouze opakuje pro každý časový okamžik. Pravidelné mozaiky mohou být hnížděny za použití lineárních nebo regionálních čtyřstronů nebo jiných hnížděných struktur.

Třídímenzionální ekvivalenty pixelů se nazývají *voxely* – jsou to základní jednotky prostorových změn v 3D prostoru.

### 2.1.3 Pixely a voxely jako entity

Základní jednotky dělení v pravidelných mozaikách souvislého prostoru mohou poskytnout geometrické odkazy na jednoduché datové jednotky – body, linie a plochy. Bod může být reprezentován jednoduchou buňkou, linie řadou sousedních buněk o šířce jedné buňky, které mají stejnou hodnotu atributu a polygon sadou sousedních buněk, které mají stejnou hodnotu atributu. Často se ale dává přednost vektorové reprezentaci, protože pravidelné buňky mřížky mohou ztratit prostorové detaily, i když se to stává menším problémem, jak se zvyšuje výkon počítačů a paměť.

Překrývání se těchto dvou přístupů se může vyskytnout např. když pracujeme s půdními jednotkami, využitím půdy apod. Klasický přístup je definovat třídy (půdního využití) a pak identifikovat plochy (*entity*), které odpovídají těmto třídám. Tyto plochy mohou být reprezentovány vektorovou hranicí uzavřenou polygonem nebo řadou sousedních buněk se stejnou hodnotou atributu. Takové reprezentaci se říká choroplethová mapa, protože obsahuje zóny se stejnou hodnotou, nebo chorochromatická mapa, protože každá zóna je zobrazena jednou barvou nebo stínem.

Alternativním přístupem je takové entity prohlásit za souvislé proměnné, nikoli entity, a zeměpisný povrch je tvořen zónami (*polygony*), kde atributy mají stejnou hodnotu a zónami, kde se atributy náhle mění.

#### 2.1.4 Axiomy a procedury pro manipulaci s daty v informačních systémech

Po objasnění datových modelů jako reprezentací reálného světa a způsobů, jak je konstruovat z geografických primitiv jako jednoduché nebo složité entity nebo diskrétní hodnoty souvislého povrchu, je třeba specifikovat základní pravidla a axiomy, kterými se musí datové modely řídit. Musí se identifikovat entity – *primitivní* jako body, linie, polygony, *složité* jako sady bodů, linií a polygonů

- všechny základní entity jsou definovány na základě zeměpisné polohy (prostorové souřadnice nebo geometrie), jejich atributů (vlastností) a vztahů (topologie)
- entity jsou rozlišitelné jedna od druhé svými atributy, polohou, vnitřními a vnějšími vztahy
- jak entity tak atributy mohou být klasifikovány do užitečných kategorií
- Booleovská algebra se může provádět nad entitami, jejich atributy, jejich vztahy a skupinami, ke kterým přísluší

#### 2.1.5 Datové modelování a prostorová analýza

Datové modely a datové typy umožňují provádět různé druhy analýz.

- Jestliže poloha a forma entity je neměnná a je třeba znát přesné hodnoty atributů, které pak odráží změnu stavu, pak je vhodná vektorová reprezentace
- Jestliže atributy jsou fixní, ale entita může měnit formu nebo tvar, ale ne polohu (vysychání jezera), pak rastrový model zachytí změny vodního povrchu
- Jestliže atributy se mohou měnit a entita může měnit pozici, ale ne formu, takové chování dobře popíše objektově-orientovaný model
- Jestliže nejsou nalezeny žádné entity, pak se celý jev bere jako souvislé pole

## 2.2 Databázové struktury a DBMS

Je třeba organizovat způsob uložení všech souborů v prostorové databázi a propojit je tak, aby modelovaly jev reálného života a zajistit vhodné uložení a přístup k datům. Počítačový program, který je určen pro uložení a řízení velkého objemu dat se nazývá *system řízení báze dat (SŘBD)*.

Moderní SŘBD používají mnoho metod, ale všechny jsou založeny na třech základních způsobech organizace informací, které jsou známy a to *hierarchický, síťový a relační*. Nedávno se začala používat čtvrtá struktura zvaná *objektově-orientovaná (O-O)*. Je to další vývoj síťového modelu, který umožňuje účelně modelovat interkonektivitu a relace mezi prostorovými entitami.

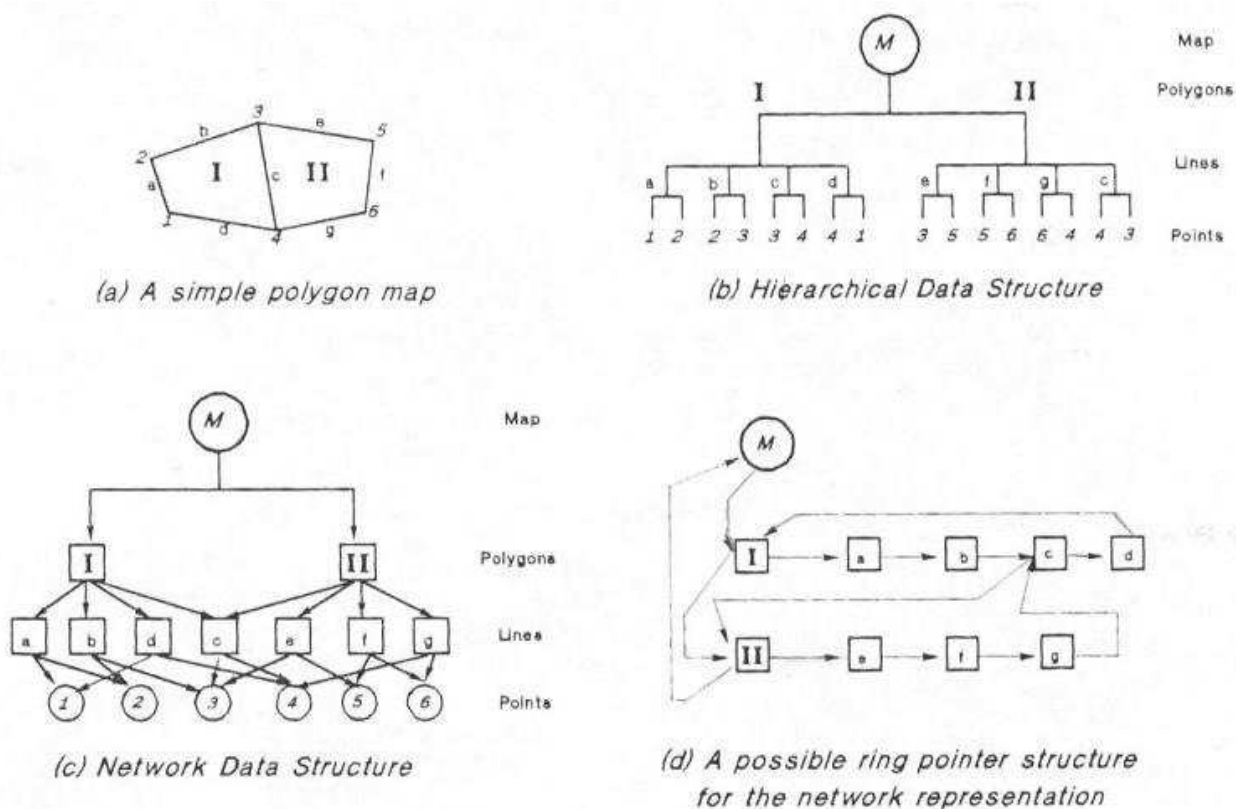
*Modelování databáze* je úloha navrhnout databázi, která pracuje účelně, obsahuje korektní informace, má logickou strukturu a co nejnadhěji se udržuje a dá se rozšířit. Aby se toho dosáhlo, je nutné nejprve si vysvětlit základní principy těchto různých organizačních struktur a zjistit, jak mohou být použity pro zaznamenané prostorové informace.

Tyto techniky jsou v prostorových systémech často používány společně tak, aby se vzájemně doplňovaly, než aby se veškerá práce prováděla jednou technikou. Hierarchický přístup je často užitečný pro rozdělení dat na snadno ovladatelná témata nebo oblasti, takže je již možné bezešvé mapování. Síťový přístup je ideální pro topologicky spojené vektorové linie a polygony. Relační přístup je dobrý pro zjištění objektů na základě jejich atributů nebo pro vytvoření nových atributů a atributových hodnot z existujících dat. Objektově orientovaný přístup je užitečný, když entity sdílejí atributy nebo se vzájemně ovlivňují speciálním způsobem.

### 2.2.1 Hierarchická databázová struktura

Když mají data strukturu rodič – potomek nebo relaci 1:m, jako jsou oblasti s různými stupni administrace, půdní druhy ve skupině půd nebo pixely uvnitř polygonu, pak hierarchické metody poskytují rychlý a pohodlný prostředek přístupu k datům. Používají se ve vědách o životním prostředí pro taxonomii rostlin a zvířat, půdní klasifikaci apod. Předpokladem je, že každá část hierarchie je dosažitelná přes klíč (sada určujících kritérií). Další předpoklad je vytvoření dobré korelace mezi klíčovými atributy a dalšími sdruženými atributy, které položky mohou vlastnit.





**Obrázek 3** Hierarchická a síťová organizace vektorových dat (převzato z [9])

Obrázek 3 a) ukazuje jednoduchý příklad mapy ze dvou polygonů a b) ukazuje její kódování v hierarchickém modelu. Hlavní výhody jsou jednoduchost a snadnost přístupu přes klíče. Jsou srozumitelné, snadno se aktualizují a rozšiřují a jsou užitečné pro organizaci dat v obrovských skladech. Přístup k datům přes klíč je snadný a rychlý, ale naneštěstí je velmi obtížný pro sdružené atributy. Následně, hierarchické systémy jsou dobré pro nalezení dat, pokud je struktura všech možných dotazů známa předem. Všeobecně to platí například v knihovnách, bankách nebo letištních systémech. Data životního prostředí, jenž mají spíše výzkumnou povahu, nemohou být spojena s pevnou hierarchií.

Další nevýhodou je udržování velkých indexových souborů a to, že jisté atributy se musí opakovat vícekrát, což vede k redundanci dat, a to zvyšuje náklady na uložení a přístup k nim. Hierarchická struktura také plýtvá prostorem např. při operaci, kdy polygony I a II dostanou společné jméno (spojí se), není cesty jak potlačit linii c, která se již nemusí zobrazovat.

### 2.2.2 Síťová databázová struktura

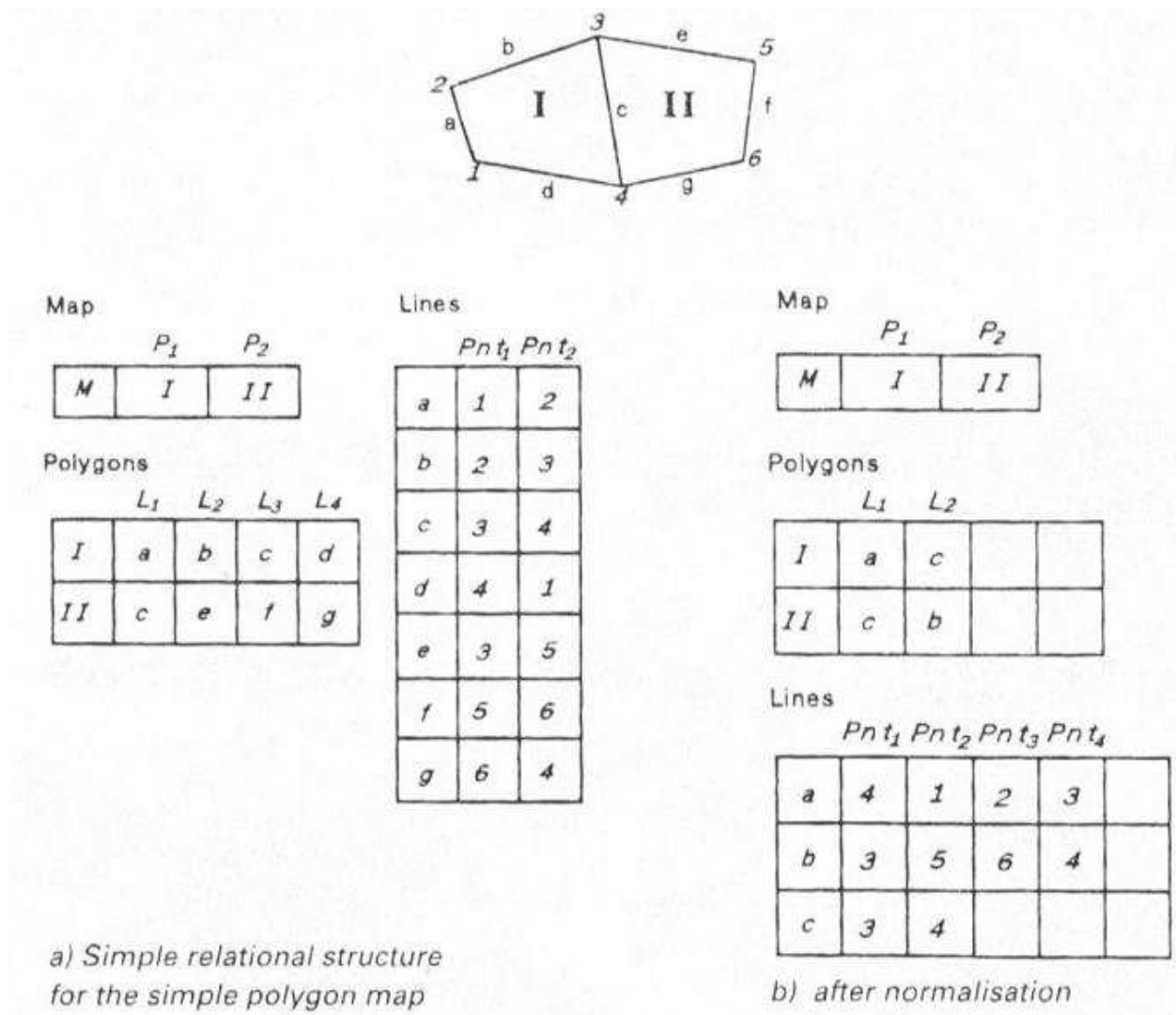
U datových struktur pro grafické objekty sousední položky v mapě potřebují být spojeny spolu, i když aktuální data o jejich souřadnicích mohou být zapsána ve zcela odlišných částech databáze.

V kompaktní síťové struktuře na obr. 3 c) , kde je každá linie a každá souřadnice pouze jednou, se nevyskytuje redundance a problémy se spojením. Jde pak jednoduše potlačit tisk linie c v případě, že oba polygony budou spojeny do polygonu s jedním jménem (snadnější tvorba generalizace map). Velmi často se v grafice v síťových strukturách používají kruhové struktury pointerů. Jsou velmi užitečné pro navigaci ve složitých topologických strukturách jako jsou polygonové sítě.

U síťových databázových struktur je velmi užitečné, když jsou relace a spojení specifikovány předem. Zabrání redundanci a velmi dobře jsou dostupná data. Nevýhodou je, že databáze se mimořádně zvětší o pointery, které ve složitém systému mohou tvořit docela podstatnou část databáze. Tyto pointery se musí udržovat aktuální v každém okamžiku, kdy je provedena změna v databázi a vytvoření a udržování struktur pointerů může značně zvětšit databázový systém.

### 2.2.3 Relační databázová struktura

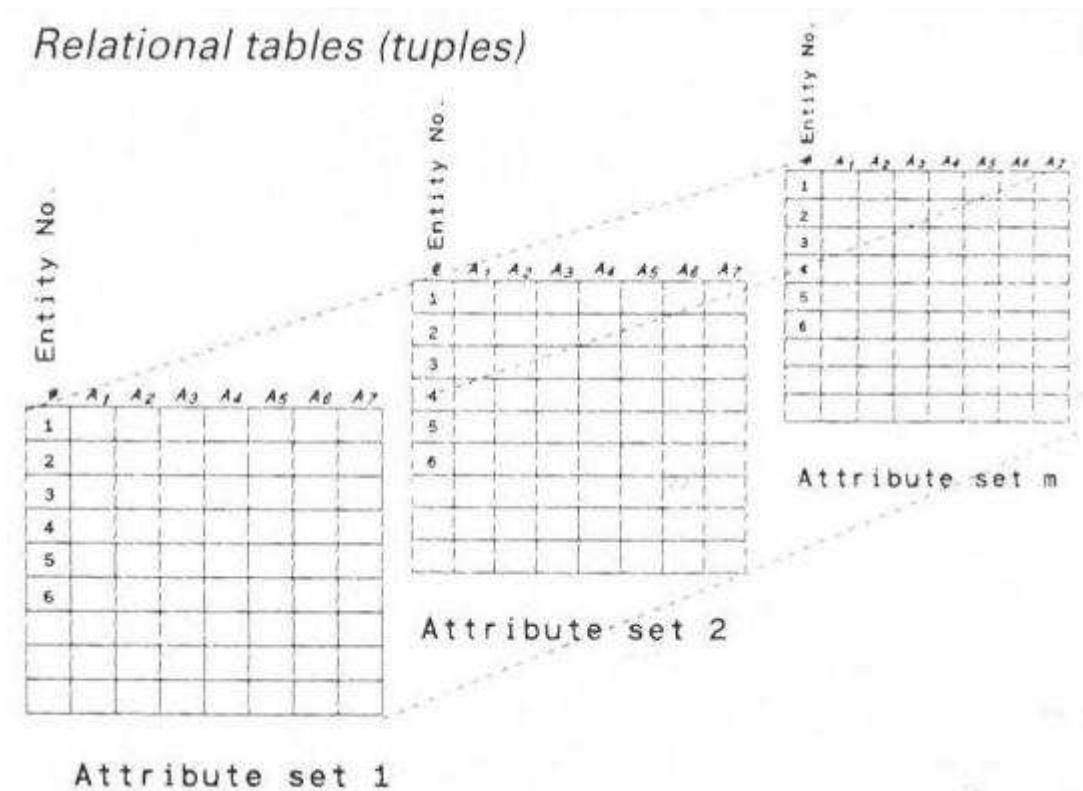
V nejjednodušší formě relační databázová struktura neobsahuje žádné pointery a žádnou hierarchii. Místo toho jsou data uložena v jednoduchých záznamech (*tuples*), což jsou sady polí, kdy každé pole obsahuje atribut. „Tuply“ jsou seskupeny dohromady ve dvojdimenzionálních tabulkách (*relacích*). Každá tabulka je samostatný soubor. Struktury pointerů nebo klíče jsou nahrazeny redundancí dat ve formě identifikačních kódů, které jsou použity jako unikátní klíče k identifikaci záznamů v souboru. – obr. 4 a)



**Obrázek 4** Struktury identifikačních kódů. a) jednoduchá relační struktura pro jednoduchou polygonovou mapu; b) po normalizaci (převzato z [9])

Data se získávají z relační databáze pomocí uživatelem definovaného přesného dotazu. Tato relace nemusí být v existujících souborech, pak se používají metody relační algebry ke konstrukci nových tabulek. Dají se použít i metody normalizace. Například adresování linií jako sady přímých liniových segmentů (*oblouků*) zjednodušuje relační strukturu bez ztráty informace (Obr. 4 b).

Relační databáze mají velkou výhodu, že jejich struktury jsou velice pružné a mohou splnit požadavky všech dotazů formulovaných s pomocí Booleovské logiky a matematických operací. Přidání nebo odstranění dat je rovněž jednoduché - jde o odstranění hodnoty v relační tabulce nebo celé tabulky (Obr. 5).



**Obrázek 5** Přidání nových dat do relačního systému je jenom záležitost definování nových tabulek (převzato z [9])

Avšak pokud jsou relace mezi tabulkami složité a je potřeba řada spojení, bude tato operace zabírat značné množství počítačového času dokonce i na velmi rychlých počítačích. Relační databázové systémy jsou velmi sofistikovaně navrženy, aby dokázaly vyhledávat data rozumnou rychlostí (proto jsou tak drahé). Nyní jsou hlavním nástrojem pro operace s atributy prostorových entit.

#### 2.2.4 Objektově-orientovaná databázová struktura

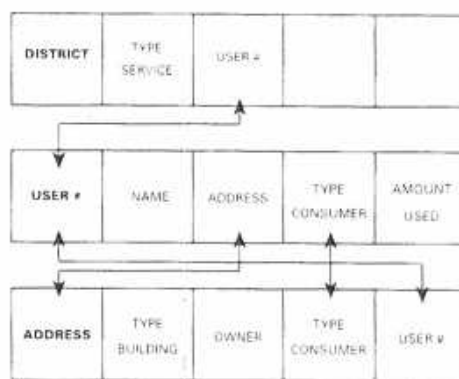
Vývoj těchto struktur byl způsoben problémy s redundancí a sekvenčním vyhledáváním v relačních strukturách. V GIS byl vývoj vyvolán potřebou pracovat se složitými prostorovými entitami inteligentněji než jen jako s bodem, linií, polygonem a také problémy s modifikací databáze, když se provádí analytické operace jako překrytí polygonů.

Objektově-orientované databázové struktury kombinují rychlost hierarchických a síťových přístupů s pružností relačních databází tím, že organizují data kolem aktuálních entit oproti způsobu zpracovávání funkcí. V objektově-orientovaných databázích jsou data definována

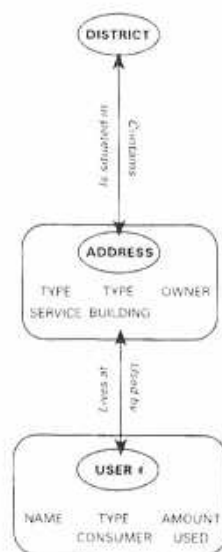
v sériích unikátních objektů, které jsou organizovány do skupin podobných jevů (objektové třídy). Relace mezi odlišnými objekty a odlišnými třídami jsou zavedeny pomocí explicitních linků.

Charakteristika objektu může být popsána v databázi pomocí jeho atributů tzv. stav stejně jako sadou procedur, které popisují jeho chování tzv. operace a metody. Tato data jsou zapouzdřena uvnitř objektu, který je definován unikátním identifikátorem uvnitř databáze, který zůstává neměnný.

Strukturování objektů uvnitř databáze je založeno na použití ukazatelů, které se odkazují přímo na unikátní identifikátory. Třídy a instance uvnitř jsou spojeny ukazateli, aby se ukázaly různé relace a hierarchie. Když jsou vytvořeny všeobecné třídy, podtřídy a supertřídy, stavy a metody se předávají dál pomocí systému dědičnosti.



(a) Example of a relational database for utility consumer management.



(b) Object orientation hierarchy for the same data

**Obrázek 6** a) Relační tabulky pro databázi utilit včetně redundantní informace. b) Objektově-orientovaný přístup redukuje uložení dat a zdokonaluje spojení mezi záznamy (převzato z [9])

Obrázek 6 porovnává relační databázi s O-O přístupem. Všechna data jsou uložena pouze jednou a ukazatele slouží nejen kvůli rychlému přístupu k datům, ale také k předávání příkazů nahoru a dolů v databázové hierarchii.

Jakmile jsou data jednou uvnitř objektu v databázi, jediný způsob, jak je změnit nebo se na ně dotazovat, je poslat požadavek tzv. zprávu a provést jednu z operací. Typy možných dotazů závisí na operacích, které objekt používá. Odpověď objektu na zprávu bude záviset na jeho stavu a stejná zpráva může vrátit jinou reakci, pokud je od jiného objektu nebo v jiném kontextu.

Vytvoření O-O databáze je časově hodně náročné, protože objekty musí být definovány explicitně a musí se vytvořit různé linky. Když už je databáze jednou vytvořena, pak poskytuje velice výkonnou strukturu pro dotazy zvláště na specifické objekty.

### **2.2.5 Deduktivní databázový přístup**

Deduktivní databáze ukládají jak data, tak logiku, která definuje fakt nebo která vyjadřuje vztah. Datové záznamy jsou tvořeny vnější částí (podobné formě relační databáze) a vnitřní částí (virtuální relace vyjádřená logicky za použití jiných relací). Výhodou deduktivního databázového přístupu je, že dovoluje složitější objekty a modelování oproti použití relačních, síťových nebo hierarchických struktur a tak nabízí velký potenciál pro manipulaci s prostorovými daty.

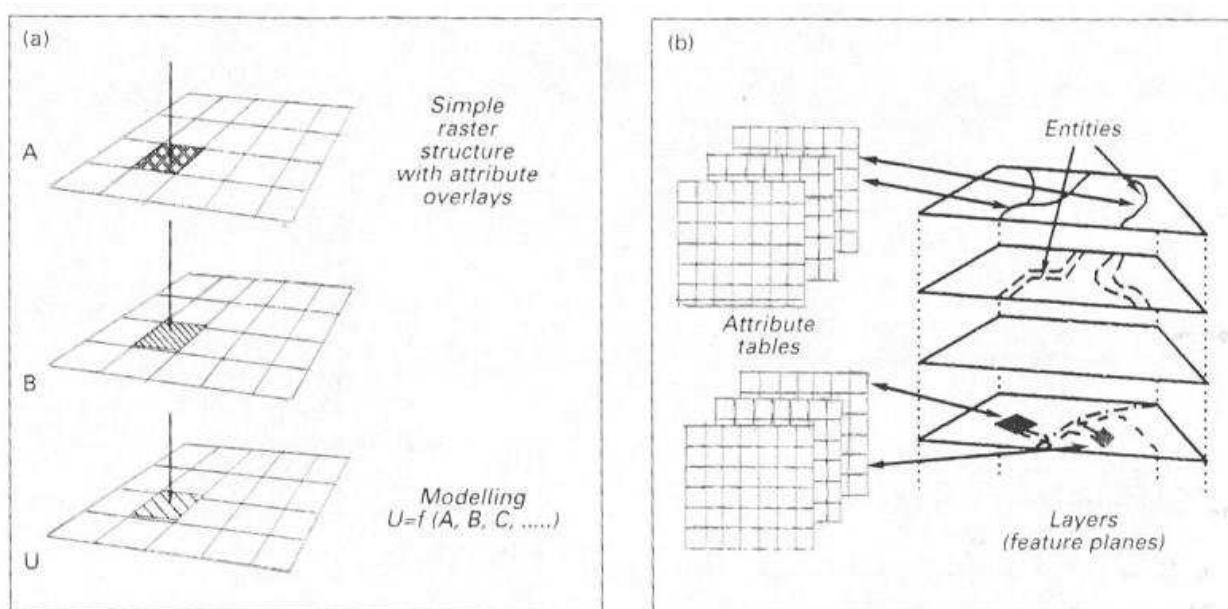
Vývoj GIS založených na těchto databázových strukturách je v současnosti stále v teoretické fázi.

## 2.3 Datové struktury pro reprezentaci geometrie prostorových jevů

Tyto hlavní struktury ovlivňují, jak jsou geografická data rozdělena a uložena v GIS. Mnoho systémů, ať už rastrových nebo vektorových, používá metody strukturování dat do intuitivně užitečných skupin.

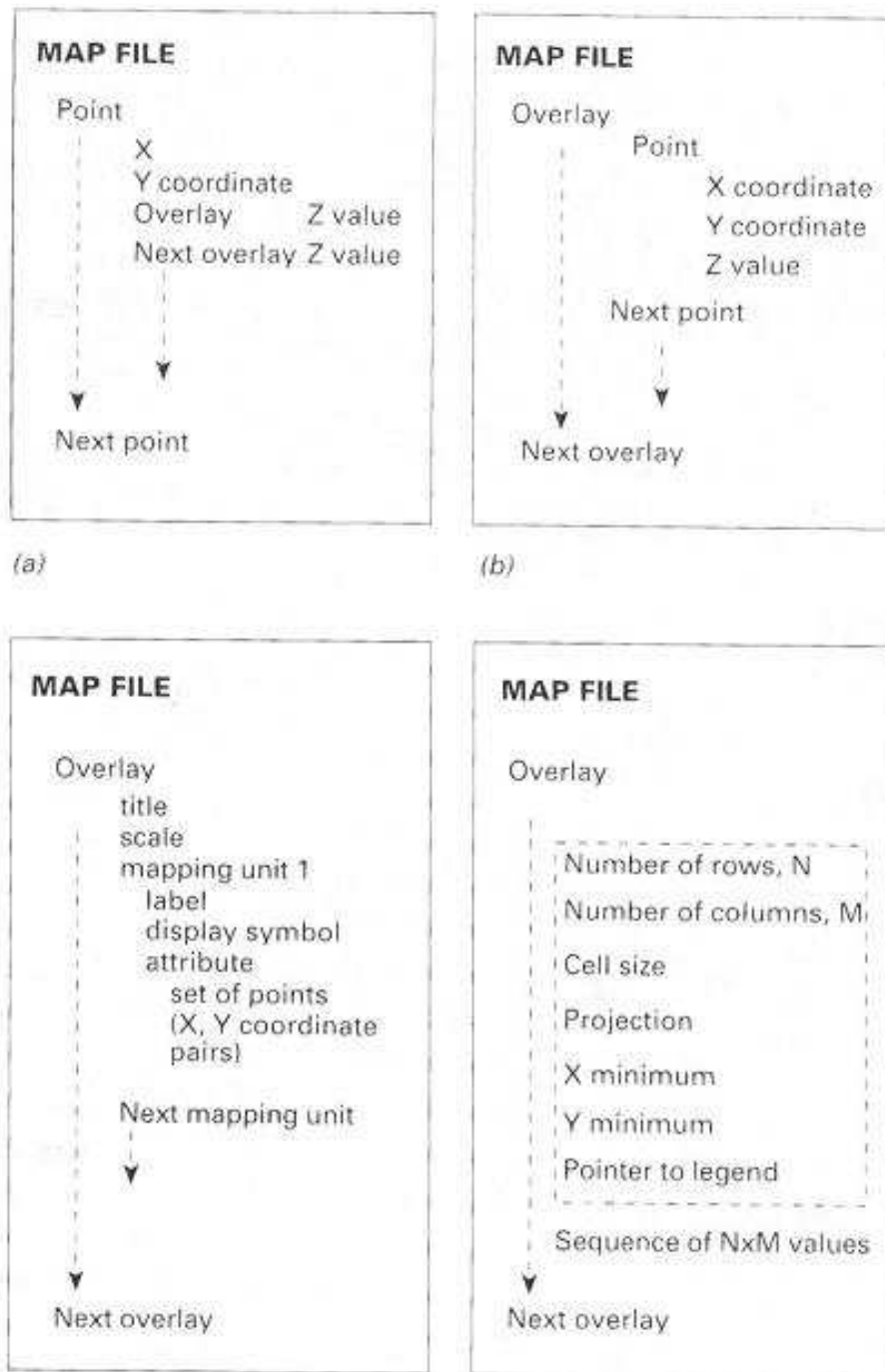
### 2.3.1 Rastrové datové struktury

Rastrová databáze je vystavěna jako řada vrstev, které se překrývají. Poskytnutí velkého počtu souřadnic pro organizaci schémat pomáhá optimalizovat přístup k datům a minimalizovat požadavky na uložení a zpracování viz obr. 7 a).



**Obrázek 7** Datové roviny a feature vrstvy v a) rastrové a b) vektorové datové struktuře (převzato z [9])

V jednoduchých rastrových strukturách, kde každá buňka v každé vrstvě je nezávislá jednotka v databázi (relace 1:1 mezi datovou hodnotou, pixlem a polohou), každá buňka je identifikována párem souřadnic a řadou atributových hodnot pro každou vrstvu viz obr. 8 a). Tato datová struktura má hodně chybějících dat (je „dřevá“), když nemá data o tom, jak je velká buňka, nebo chybí zobrazovací symboly a hlavně – žádné kompresní techniky na redukci uložených dat se nemohou použít.



**Obrázek 8** Čtyři rozdílné způsoby vytvoření rastrové datové struktury a) Každá buňka je odkazována přímo; b) každé překrytí je odkazováno přímo; c) každá mapová jednotka je odkazována přímo; d) každé překrytí je odděleným souborem s obecnou hlavičkou (převzato z [9])

U alternativní metody podle obr. 8 b) může být každá vrstva reprezentována v databázi jako dvojdimenzionální matice bodů nesoucích hodnotu jednoduchého atributu. To ale ještě vyžaduje dost prostoru, protože struktura obsahuje seznamy redundantních souřadnic, které se opakují pro každou vrstvu a opět chybí data o tom, jak je velká buňka, nebo chybí zobrazovací symboly.



Hierarchická struktura na obr. 8 c) vytváří M:1 relace mezi atributovými hodnotami a řadou bodů v dané jednotce, takže polygon stejných hodnot lze snadno adresovat. Změna hodnot nebo zobrazovacích symbolů se provede snadno přepsáním pouze jedné hodnoty v jednotce jedné vrstvy. Tato struktura také dovoluje run-length kódování nebo kompresní techniku quadtree.

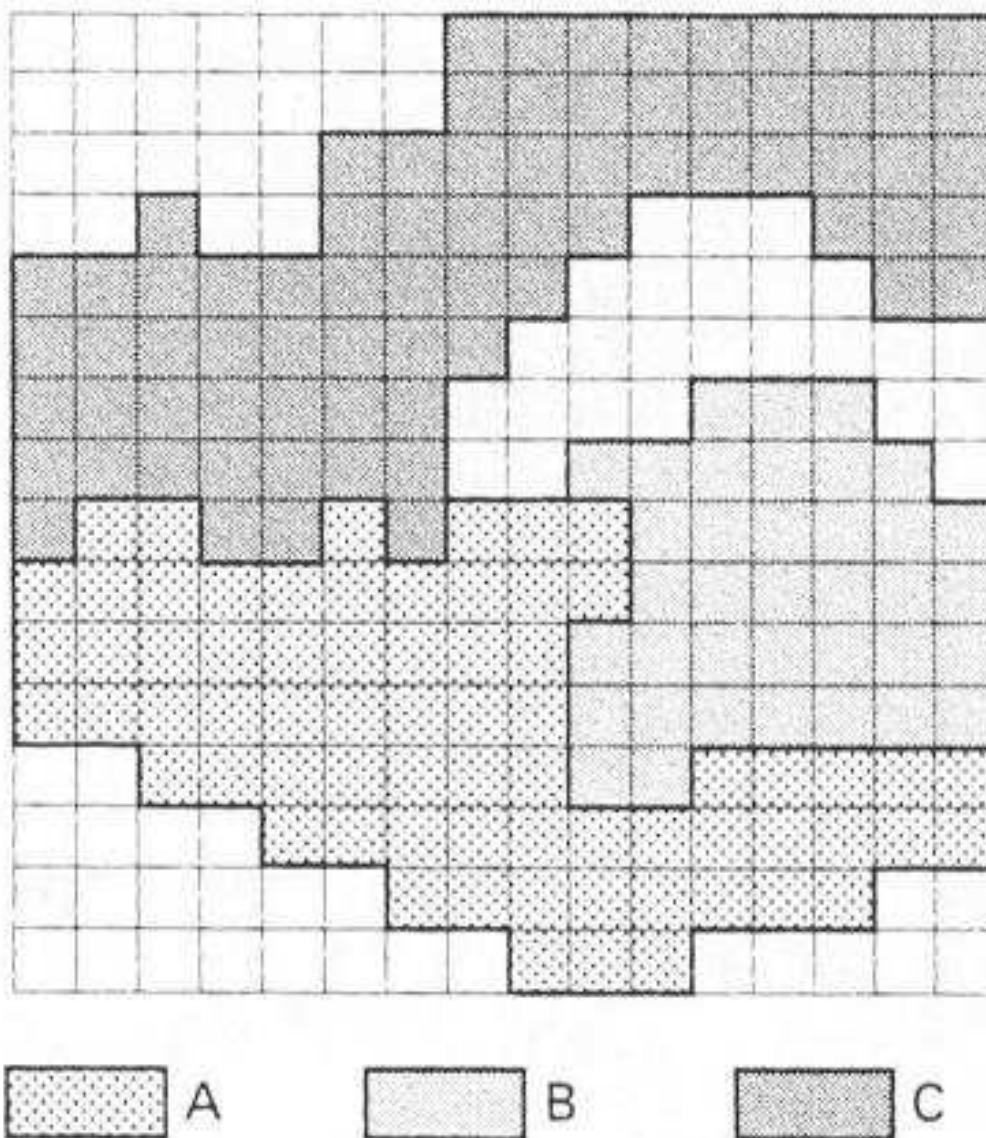
U čtvrté struktury je každá vrstva uložena jako oddělený soubor s všeobecnou hlavičkou obsahující informace jako zobrazení, velikost buňky, počet řad a sloupců a datový typ. Za tím je pak jednoduchý seznam hodnot řazených po celých řádkách viz obr. 8 d). Je evidentní, že souřadnice nejsou uloženy pro každou buňku a všeobecné geometrické vlastnosti a zobrazovací hodnoty jsou zapsány v hlavičce vrstvy.

### **2.3.1.1 Komprimovací metody pro uložení rastrových dat**

Když jsou rastrové datové struktury použity pro reprezentaci souvislého povrchu, kdy každá buňka má unikátní hodnotu, pak se musí kódovat všech  $n \times m$  buněk pro každou takovou vrstvu. Nejvíce zabírají vrstvy, kde buňky obsahují reálná čísla (např. zeměpisná šířka pro digitální model terénu). Jiné datové typy potřebují méně prostoru, protože mohou být kódovány na menší počet bitů.

Rastrové struktury pro reprezentaci linií a ploch, kdy pixely všude mají stejnou hodnotu, umožňují značné ušetření prostoru. Až třetí struktura na obr. 8 c) dovoluje 4 následující způsoby ekonomického uložení dat a to jsou tyto: řetězení kódů, run-length kódování, blokové kódování a quadtree.

## Řetězení kódů



**Obrázek 9** Jednoduchá rastrová mapa (převzato z [9])

V obr. 9 je hranice plochy A dána jejím počátkem a jednotkovými vektory v základních směrech (východ = 0, sever = 1, západ = 2, jih = 3). Když např. začneme v buňce řada 10, sloupec 1, hranice bude po směru hodinových ručiček zakódována takto:

$0, 1, 0^2, 3, 0^2, 1, 0, 3, 0, 1, 0^3, 3^2, 2, 3^3, 0^2, 1, 0^5, 3^2, 2^2, 3, 2^3, 1, 2^2, 1, 2^2, 1, 2^2, 1^3$

kde počet pixelů v každém směru je mocnina daného směru.

Řetězové kódy mohou být uloženy jako celočíselné datové typy a proto umožňují velkou kompresi dat pro uložení ploch.

## Run length kódování

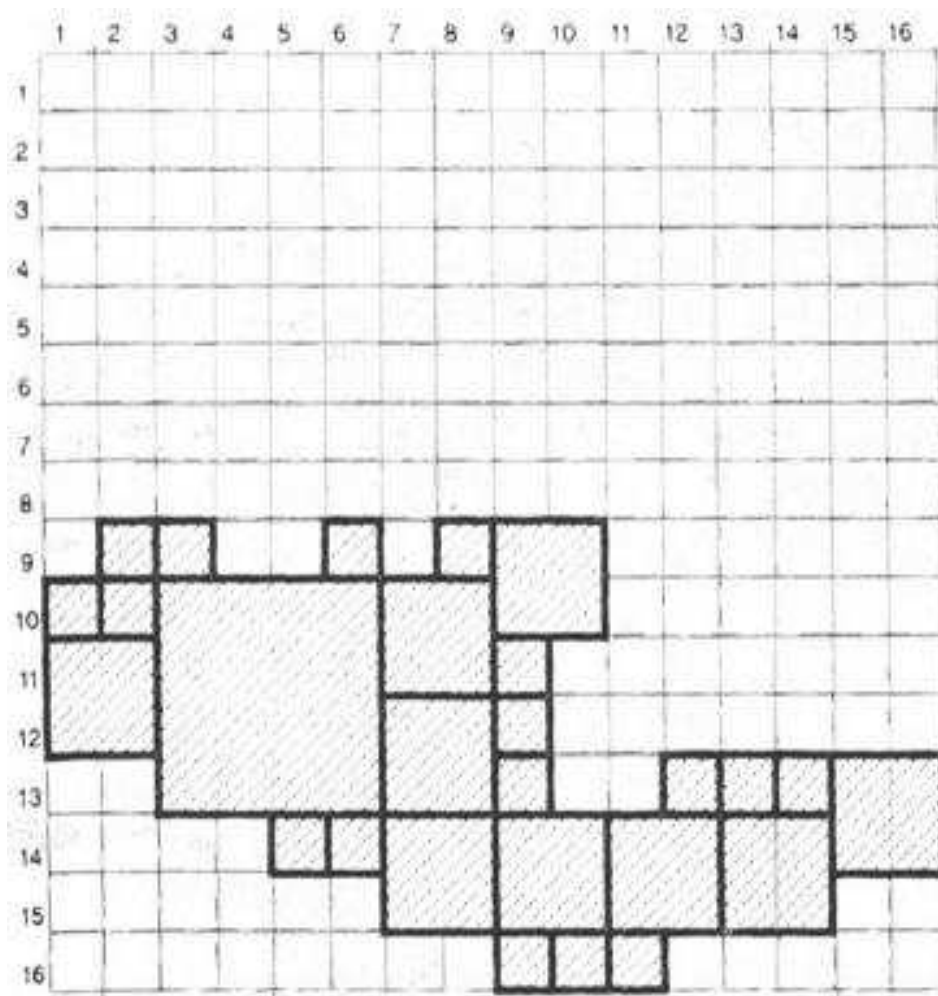
Toto kódování umožňuje, aby body zobrazované jednotky byly uloženy po řádkách zleva doprava a v každém řádku pro jednu třídu všechny dvojice (sloupec počáteční buňky, sloupec konečné buňky) dané třídy v daném řádku. Stačí pak celočíselný datový typ, dokonce celé číslo na 8 bitů. Pro plochu A z obr. 9 bude kód vypadat takto:

Řádek 9	2,3 6,6 8,10
Řádek 10	1,10
Řádek 11	1,9
Řádek 12	1,9
Řádek 13	3,9 12,16
Řádek 14	5,16
Řádek 15	7,14
Řádek 16	9,11

Toto kódování poskytuje značnou redukci prostoru na uložení dat. Na druhou stranu přílišná komprese může vést ke zvýšení požadavků na zpracování při kartografickém zpracování a manipulaci. Také není vhodné pro kódování souvislých změn, protože každá buňka má unikátní hodnotu a prostor na uložení by pak narůstal.

## Blokové kódování

Myšlenka run length kódování může být rozšířena na dvě dimenze, kdy se pak používají různě velké čtverce pro „vydláždění“ mapované oblasti.



**Obrázek 10** MAT polygonu A v Obrázku 9 (převzato z [9])

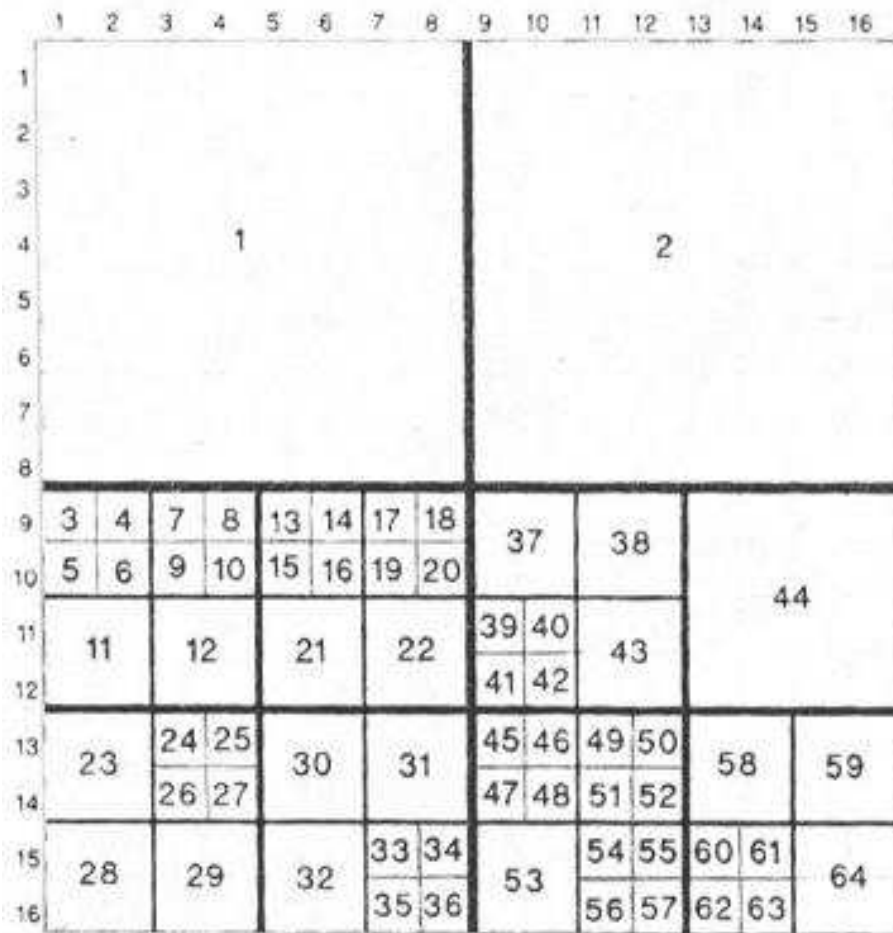
Obr. 10 ukazuje, jak je tento způsob použit pro plochu A z obr. 9. Datová struktura se skládá pouze ze tří čísel – souřadnice počátku (střed nebo levý dolní roh čtverce) a velikost strany čtverce v pixelech. Tato metoda je známa pod zkratkou MAT (Medial Axis Transformation). Plocha A tak může být uložena pomocí 17 jednotkových čtverců, 9 čtverců o straně 2 pixely a 1 čtverce o straně 4 pixely. Je uložena pomocí 57 čísel (54 pro souřadnice a 3 pro velikost použitých čtverců).

Jak run length kódování tak blokové kódování jsou neúčinnější pro kompresi dat pro velké jednoduché plochy a nejméně vhodné pro malé složitější oblasti o něco málo větší než je základní buňka.

### **Quadtree a binární strom**

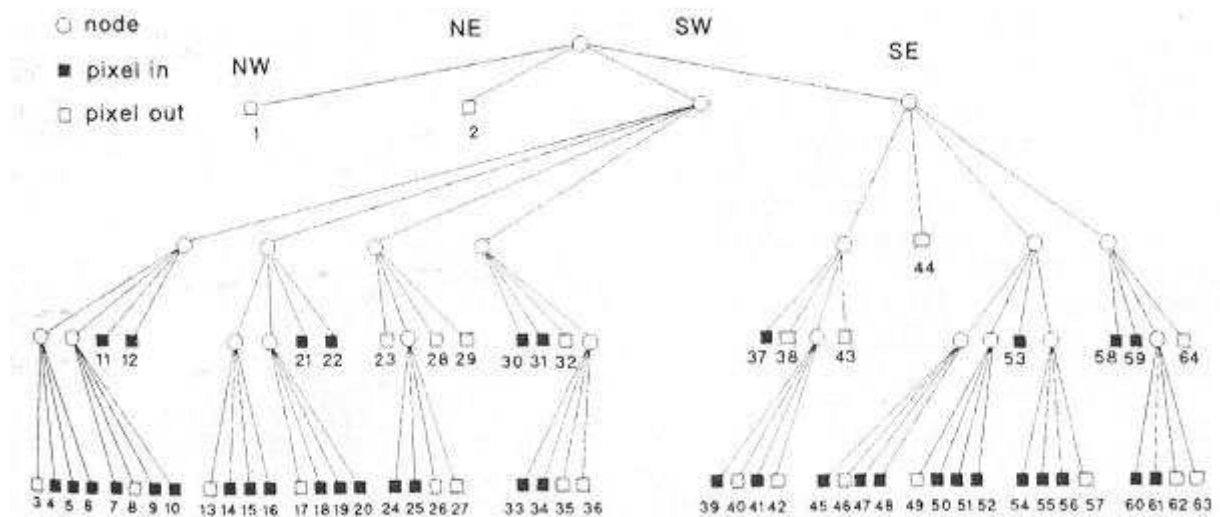
Problémem u pravidelných mřížek je rozlišení, které je omezeno velikostí základní buňky mřížky. Binární strom a quadtree poskytují přístup v principu až do nekonečné úrovně.

Nejvýkonnější metody komprese jsou založeny na postupném hierarchickém dělení pole  $2^n \times 2^n$ . Pokud se dělí oblast vždy na polovinu, metoda se nazývá *binární strom* a pokud se oblast skládá z dlaždic do kvadrantů a kvadrant je úplně obsažen v oblasti, pak se hovoří o *quadtree*.



**Obrázek 11** Kódování quadtree polygonu A na Obrázku 9

Obrázek 11 ukazuje postupné dělení oblasti A na bloky kvadrantů. Tato bloková struktura může být popsána stromem stupně 4 zvaném quadtree, který je na obr. 12. Každý uzel má čtyři syny. Listy odpovídají kvadrantům, které se již dále nedělí.



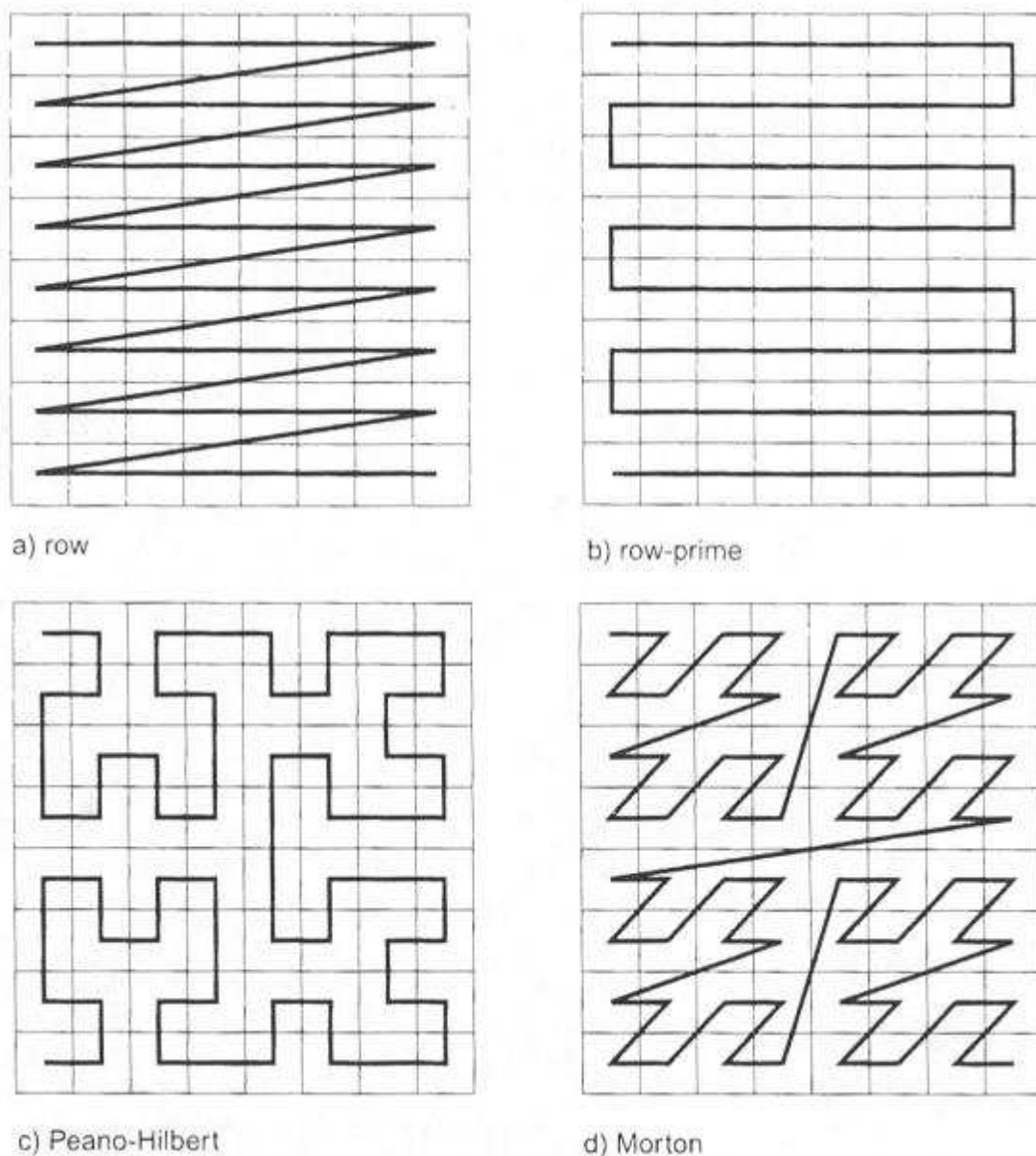
**Obrázek 12** Quadtree hierarchie polygonu A na Obrázku 9 (převzato z [9])

Pro 3D systémy existuje *octree*, což je analogie 2D quadtree.

Největší problém je, že tato struktura nezachovává invariančnost převedení dat do struktury a tak dvě oblasti stejného tvaru mohou mít dva nebo více zcela odlišných quadtree.

### 2.3.1.2 Dvojdímenzionální řazení

Používají se různé metody řazení, které umožňují v důsledku odkazování na buňky nebo definované oblasti až přes jedno-dímenzionální souřadnici. Tak se opět ušetří místo pro uložení na disku a paměť. Seřazení definuje cestu přes dvou- a tří-dímenzionální mřížkový prostor viz obr. 13.



**Obrázek 13** Alternativní způsoby kódování dat na různých stupních rozlišení (převzato z [9])

Různé cesty mají různé cíle (celková délka, jednotková délka nebo spojení sousedních buněk sekvenčně apod.). Nejpoužívanější jsou Mortonovo (obr. 13 c) ) a Peano-Hilbertovo (obr. 13 d) ) řazení, kdy pixely jsou indexovány podle speciálních rekurzivních sekvencí založených na subkvadrantech. Tato řazení umožňují adresování jednotek mřížky, které značně šetří čas při dotazování do databáze.

### 2.3.2 Vektorové datové struktury

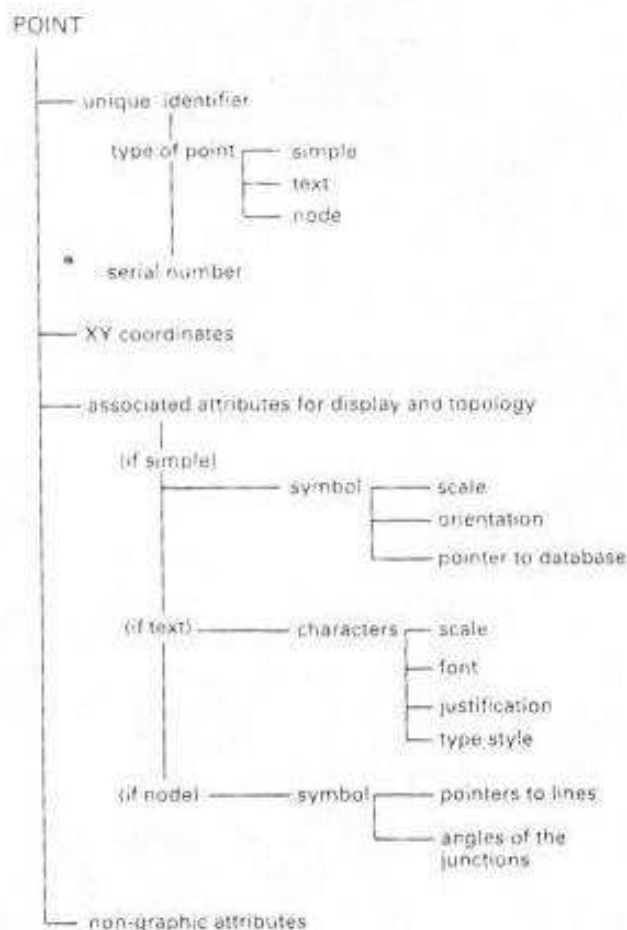
Vektorová databáze je vystavěna z vrstev, ve kterých jsou jednotlivé featury, které jsou použity v různých třídách daného jevu. Jednotky jsou reprezentovány objekty v souřadnicovém prostoru, který je souvislý a dovoluje definovat přesně všechny pozice, délky a rozměry. Ve skutečnosti to

není tak přesné, protože existují omezení na délku slova v počítači a tudíž i na přesnou reprezentaci souřadnice a protože všechna zobrazovací zařízení mají základní velikost kroku nebo rozlišení. Kromě tohoto předpokladu, vektorové metody uložení dat dovolují, aby složitá data byla uložena na minimálním prostoru.

### 2.3.2.1 Vektorové datové struktury pro uložení vektorových dat

#### Body

Bodové entity odrážejí zeměpisné a grafické entity, které jsou alokovány jedním párem souřadnic. Přesto musí být k těmto souřadnicím přiřazena další data, aby bylo možno indikovat, o jaký druh bodu se jedná a s ním spojená jeho další data. Bod může být prostý symbol (přesto se k němu pojí informace o zobrazovací velikosti a orientaci symbolu), text (pak je nutná informace o fontu, zarovnání, měřítku a orientaci) nebo uzel viz obr. 14.



**Obrázek 14** Vektorová datová struktura jednoduché bodové entity (převzato z [9])



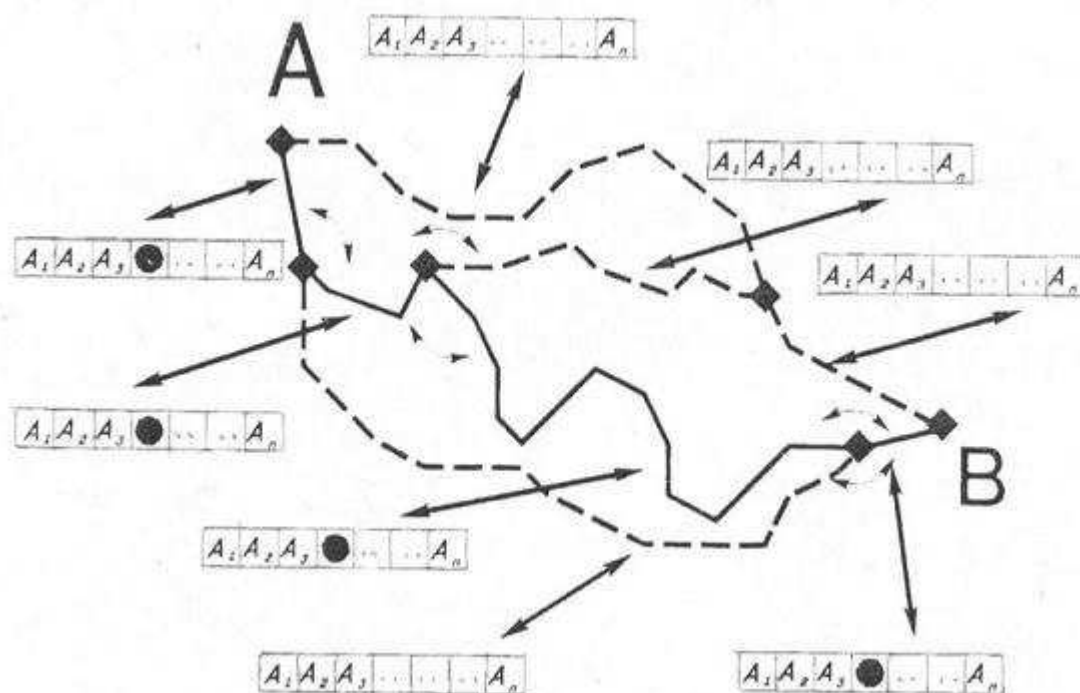
## Linie

Liniové entity mohou být definovány jako lineární featury vystavěné z přímých liniových segmentů tvořených dvěma nebo více souřadnicemi. Nejjednodušší linie požaduje uložení počátečního a koncového bodu plus záznam zobrazovaném symbolu.

Oblouk je řadou párů souřadnic XY popisujících souvislou složitou čáru. Čím je kratší linie a čím větší počet takových párů, tím více se řetěz blíží složité křivce. Prostor pro uložení dat se může ušetřit použitím B-splinů na úkor delšího času zpracování.

## Sítě

Jednoduché linie a řetězy nenesou žádnou dědičnou informaci o konektivitě, která je potřeba u síťových analýz u cest, dopravy nebo drenáží. Aby se dala linie trasovat linie po linii, je nezbytné přidat topologický pointer na datovou strukturu, který se odkazuje na uzly viz obr. 15.



**Obrázek 15** Hybridní datová struktura pro síťovou analýzu (převzato z [9])

Kromě pointerů na oblouky, uzly ponosou datové záznamy, které indikují úhel, pod kterým se každý řetěz napojuje na uzel, tím pádem je definována topologie sítě. Tato

jednoduchá struktura s sebou nese jistou redundanci. Atributy mohou být použity pro určité vybrané cesty.

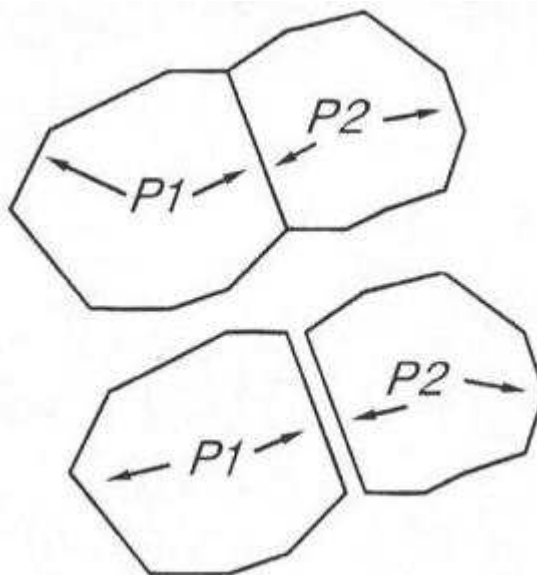
## Polygony

Polygony mohou být reprezentovány ve vektorové databázi různými způsoby. Cílem polygonové datové struktury je popsat topologické vlastnosti ploch (tj. jejich tvary, sousedy a hierarchii) takovým způsobem, že připojené atributy se mohou zobrazovat a lze s nimi manipulovat jako s daty tématické mapy.

Je užitečné vyjmenovat požadavky na polygony. První - všechny polygony na mapě mají každý unikátní tvar, obvod a obsah. Neexistuje žádná jednoduchá standardní jednotka jako u rastrových systémů. Druhý - geografické analýzy vyžadují, aby datové struktury byly schopné zaznamenat sousedy každého polygonu. Třetí – polygony na tématické mapě nejsou vždy stejné úrovně (ostrovy leží na jezeře, které je samo na větším ostrově atd.)

## Jednoduché polygony

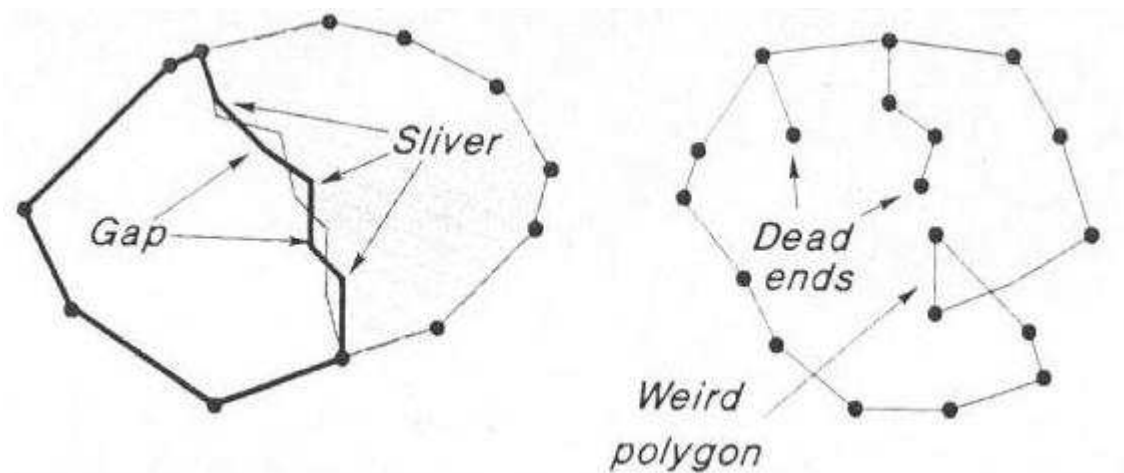
Nejjednodušší způsob reprezentace polygonu je rozšířit jednoduchý řetěz tzn. reprezentovat každý polygon jako řadu XY souřadnic hranic viz obr. 16.



**Obrázek 16** Bez topologie nemůže databáze rozlišit mezi polygony, které sdílejí hraniční linie (nahore) nebo které skutečně oddělují entity (dole) (převzato z [9])

Jména a symboly jsou pak uložena jako řada textových entit. Tato metoda je jednoduchá, ale má mnoho nevýhod:

- linie mezi sousedními polygony musí být digitalizována a uložena dvakrát. To vede k vážným chybám jako nesprávné přebytky a úbytky podél společné hranice
- není žádná informace o sousednosti
- ostrovy nejsou možné, výjimečně jako grafické konstrukce
- nelze jednoduše zkontrolovat, jestli topologie hranice je správná nebo zda je neúplná (mrtvé konce, nepřípustné smyčky) viz obr. 17

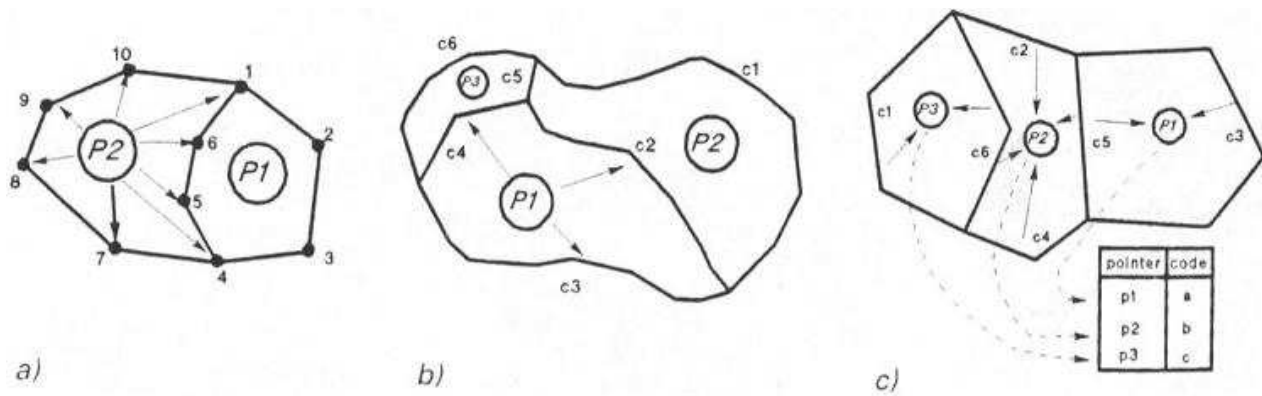


**Obrázek 17** Topologické chyby v polygonové síti (převzato z [9])

Struktura jednoduchého polygonu může být rozšířena tak, že každý polygon je reprezentován počtem řetězců, ale to neodstraní základní problémy.

### **Polygony s bodovým slovníkem**

V této reprezentaci jsou všechny souřadnicové páry číslovány sekvenčně a odkazují na slovník, který zaznamenává, které body tvoří který polygon viz obr. 18 a).



**Obrázek 18** Tři různé způsoby včleňování jednoduchých topologií v polygonové síti (převzato z [9])

Databáze bodového slovníku má výhodu, že hranice mezi sousedními polygony jsou unikátní, ale stále existuje problém funkce sousedství. Struktura rovněž nedovoluje, aby se hranice snadno potlačily nebo zrušily, když přečíslování nebo reklasifikace způsobí, že obě hranice jsou alokovány do téže třídy. Zůstává problém ostrovních polygonů, stejně jako problém mrtvých konců a nepřipustných smyček.

U jednoduchých polygonů se může použít řetězový slovník viz obr. 18 b). Má tu výhodu, že řetězy, které jsou výsledkem souvislé digitalizace, mohou být redukovány ve velikosti použitím vyřazovacího algoritmu, aniž by se modifikoval slovník.

Atributy polygonu jsou spojeny s pointery na datové tabulky viz obr. 18 c).

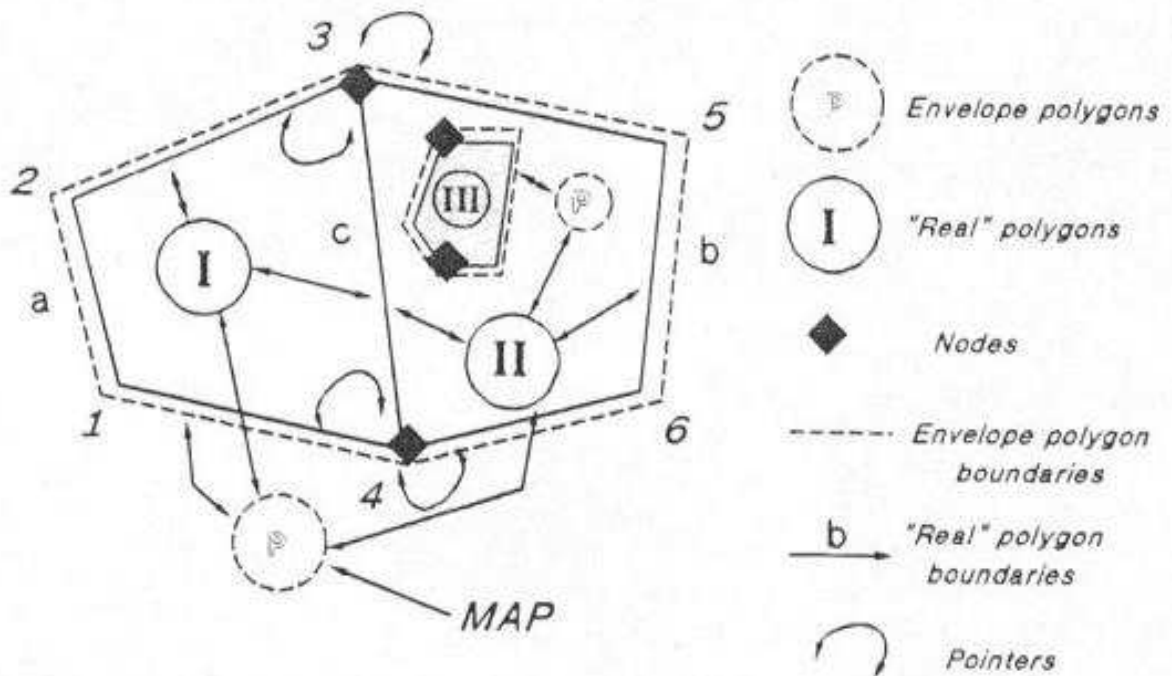
### Polygonový systém s explicitní topologickou strukturou

S ostrovy a sousedy se dá přesně pracovat pouze při zavedení explicitních topologických vztahů. Jdou zavést dvěma způsoby – vytvořením topologických linků během vstupu dat nebo použitím softwaru na vytvoření topologie ze sady řetězů nebo řetězců. V prvním případě je největší břímě na operátorovi a v druhém případě leží na výkonu počítače. Obě metody mají za výsledek nárůst dat k popisu plné topologické struktury, která musí být také uložena.

První pokus o explicitní topologické vazby používal jako základní element jednoduchý liniový segment definovaný dvěma koncovými body, složité linie jsou definovány jako série segmentů. Segment má dva pointery na uzly a na kódy pro polygon na každé straně segmentu. Protože uzly neukazují zpět na segmenty ani na sousední segmenty, je pracné vyhledat obrysy

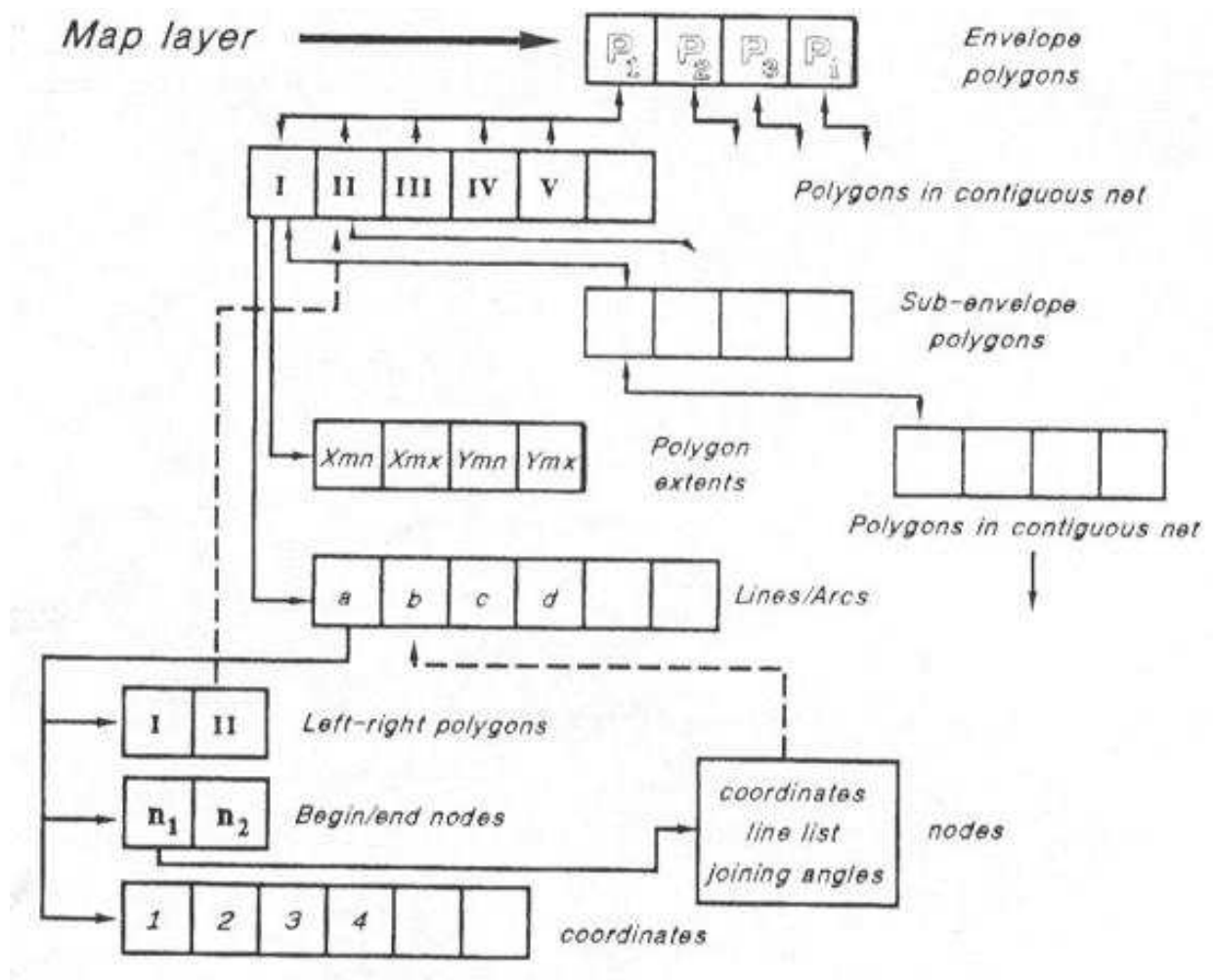
polygonů. Navíc struktura jednoduchých segmentů činí práci se složitými liniemi těžkopádnou kvůli velké redundanci dat.

### Plně topologická polygonová síťová struktura



**Obrázek 19** Plná topologická struktura v polygonové síti (převzato z [9])

Struktura na obr. 19 může být vystavěna z řady hraničních řetězů nebo řetězců, které byly digitalizovány v jakémkoli pořadí a jakémkoli směru. Systém dovoluje ostrovy a jezera do jakékoli podúrovně, dovoluje automatické kontroly na mrtvé konce a nepřípustné smyčky a dovoluje automatické nebo poloautomatické spojení s neprostorovými atributy u výsledných polygonů. Vyhledání sousedů je plně podporováno.



**Obrázek 20** Datová struktura topologicky spojené polygonové sítě (převzato z [9])

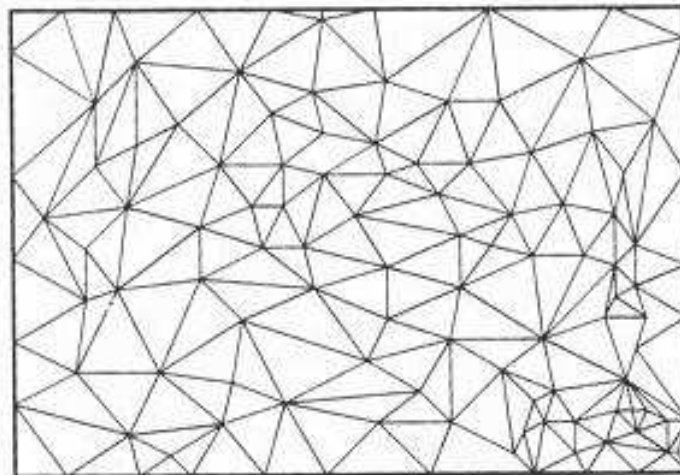
Je potřeba složitý software ke konstrukci této struktury, ale zato má výsledná datová struktura viz obr. 20 následující výhody:

- polygonová síť je plně integrována a bez přebytků a úbytků hranic a nadměrného množství redundantních souřadnic
- všechny polygony, oblouky a připojené atributy jsou části propojených jednotek, aby byly možné všechny druhy sousedských analýz.
- počet vnoření kontinentálních ostrovů je neomezený
- polohová přesnost databáze je omezena pouze přesností digitizéru a délkou počítačového slova

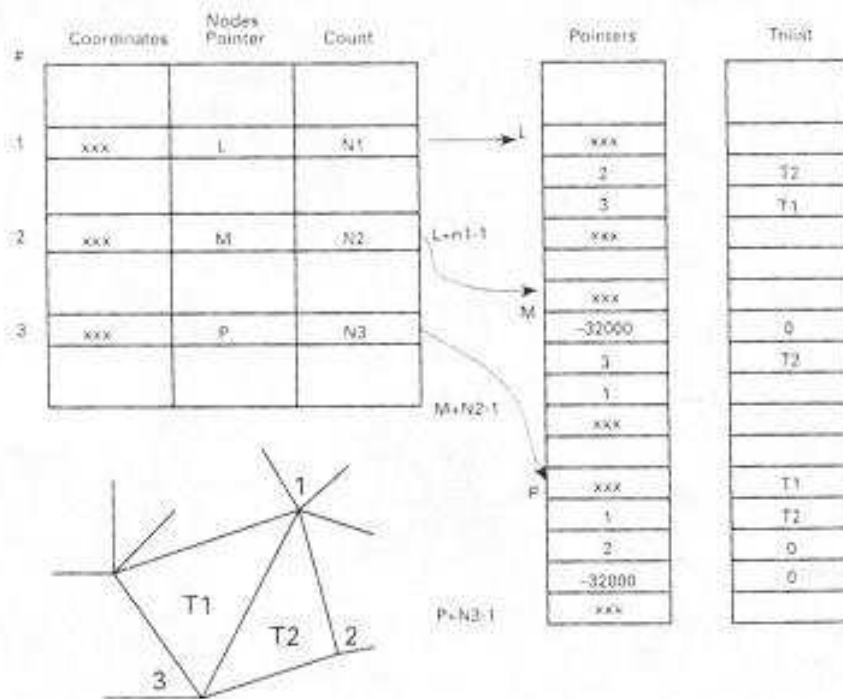
Tento druh datových struktur ale není vhodný pro prostorové vzory, které se konstantně mění.

### 2.3.2.2 Vektorové datové struktury ke speciálnímu účelu – Triangular Irregular Network (TIN)

Důležitá a nejvíce používaná vektorová struktura je TIN. Je tvořena ze spojených známých bodů do řady trojúhelníků založených na Delaunayově triangulaci. Rozdělení a hustota bodů odráží změny hodnot atributu uvnitř plochy viz. obr. 21 a).



(a) Triangular Irregular Network based on a Delaunay Triangulation



(b) Data structure of a TIN (detail)

**Obrázek 21** Nepravidelné triangulační síť založené na Delaunayově triangulaci poskytují vektorový datový model souvislého pole (převzato z [9])

Topologické vztahy jsou uloženy v databázi vytvořením pointerů z každého uzlu do každého z jeho sousedů. Seznam sousedů je seřazen podle směru hodinových ručiček a začíná se od severu.

Obr. 21 b) ukazuje část síťové datové struktury (tři uzly a dva trojúhelníky), které jsou použity v definici TIN. Databáze se skládá ze tří sad záznamů nazvaných seznam uzlů, seznam pointerů a trilst (seznam trojúhelníků).

Seznam uzlů a seznam pointerů obsahuje všechny podstatné polohové informace a spojení, takže jsou vhodné pro mnoho aplikací. Pro další aplikace jako mapování sklonitosti, stínování nebo spojení jiných atributů s trojúhelníky je nezbytné se odkazovat na trojúhelníky přímo. To se děje použitím trilstu, kdy se každá hrana spojuje s trojúhelníkem po pravé straně.

Uzly umístěné v oblastech s největší změnou pomáhají redukovat chyby v odvozených formách. Při použití těchto nepravidelně umístěných bodů TIN se tak vyhneme redundanci pravidelných mřížek a poskytneme vhodné prostředky pro výpočet odvozených dat jako je sklon.

### **2.3.2.3 Vývoj vektorových datových struktur za účelem zlepšení časového přístupu k datům a úspornějšího uložení**

Vektorový datový model je relativně vhodný prostředek pro uložení geometrické informace geografických dat se zaznamenáním příslušných souřadnicových hodnot. Hlavní problémy jsou spojeny s přístupem k datům, zvláště topologická a atributová informace. Např. mezivztahy mezi základními vektorovými jednotkami jako jsou body, linie nebo polygony jsou jednotně identifikovány pointerem nebo návěštími. Topologické struktury mohou být přeměněny prostřednictvím těchto unikátních návěští. V dřívějších GIS tato návěští, někdy nazývaná master index pointer, byla často udržována v sekvenčním seznamu, který byl klíčem k přístupu do zbytku databáze. Tento seznam měl dva větší problémy:

- *první* - byl zřídka po sobě jdoucí, během editování se objevovaly „mezery“ nebo se zvětšovala délka, což znamenalo, že vyhledávání bylo většinou sekvenční
- *druhý* – vyhledávací čas se ostře zvýšil s délkou tabulky, což znamenalo, že čas zpracování mapy se zvýšil nelineárně s velikostí databáze



Technický vývoj v GIS databázích se v 80. letech koncentroval na problém zpracování prostorových dat méně závislých na velikosti databáze. Rychlost počítačů způsobila, že se již tolik nehledí na uložení a přístup k datům. Změny v organizaci databází a vnitřní odkazování také pomohlo zlepšit výkonnost.

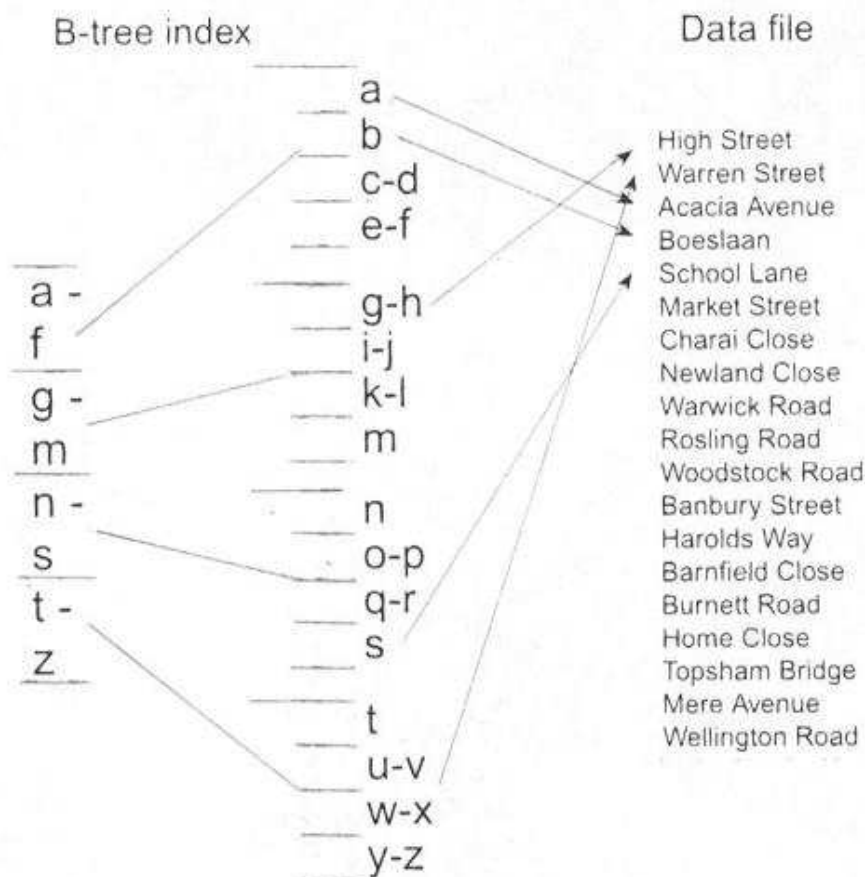
### **Clusterování dat na paměťovém médiu**

První pokusy o zdokonalení přístupových časů do databáze používaly počítačové metody brutální síly k rychlému skenování pointerových polí nebo koncentrovaly pole master indexu do malé sousední plochy disku nebo vnitřní paměti. Zatímco toto nepochybně přineslo některá vylepšení, udělalo se velmi málo pro řešení podřadných problémů. Jiný přístup zahrnoval clusterování master index pointerů nejenom podle typu entity, ale podle prostorového umístění. Data se clusterovala na disk podle geografického umístění.

### **Indexování databáze s použitím B-stromů a R-stromů**

Důležitost toho, aby indexování databáze zrychlilo dotazování, byla zdůrazněna výše. Avšak indexy pro velké nebo složité databáze, jako u GIS aplikací, se mohou samy stát dlouhými a zpomalit dotazy. Proto se vyvinuly struktury zvané hierarchické indexy indexů. Jsou známy jako multi-level (vícestupňové) indexy a jsou obzvláště užitečné u vektorových struktur, kde se mnoho topologických a atributových dat udržuje v indexových souborech.

B-strom poskytuje multi-level index, který je strukturován za použití vnitřních uzlů a listových uzlů, což je analogické větším a listům stromu viz obr. 22.

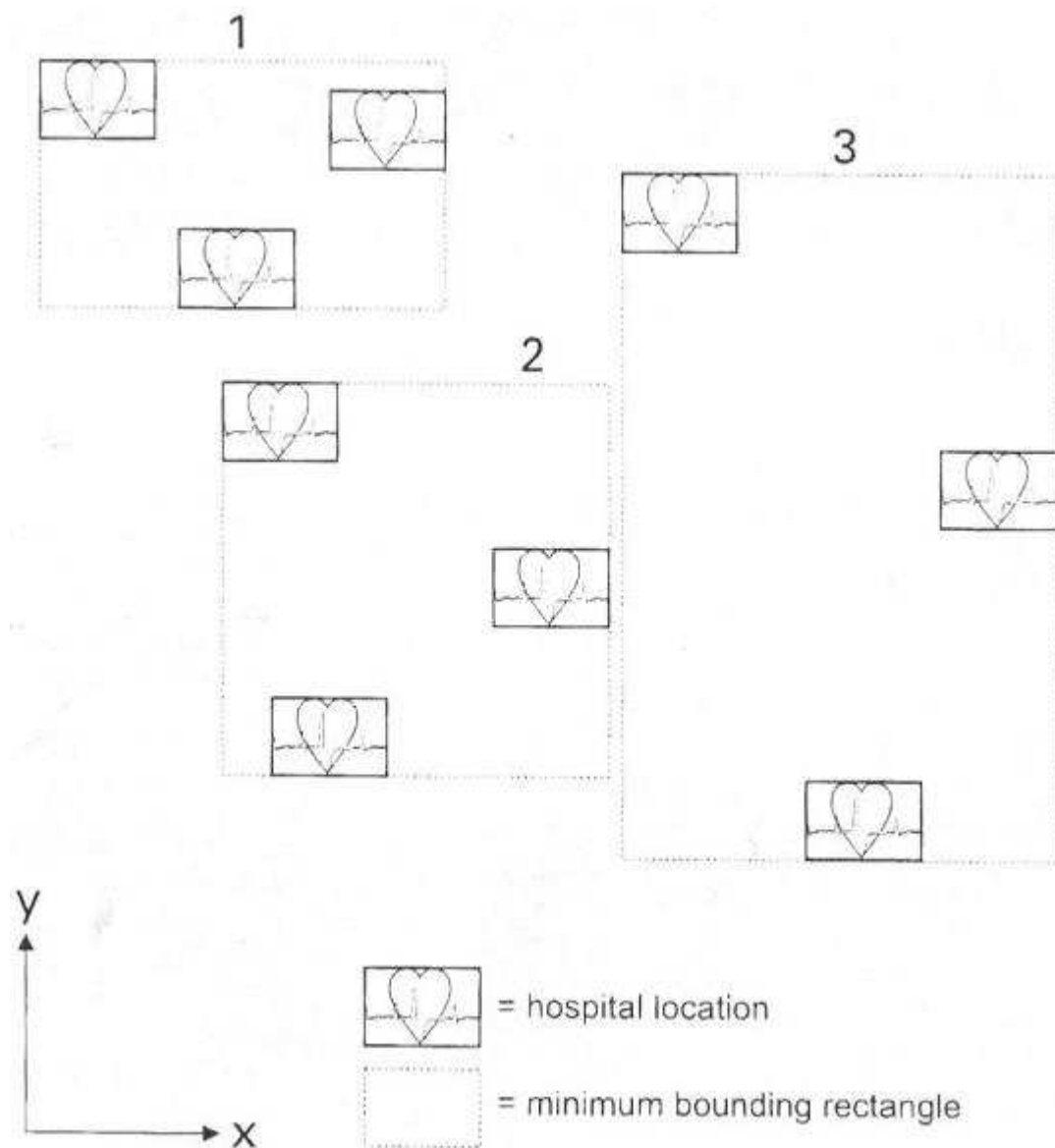


**Obrázek 22** Indexování dat za použití B-stromu (převzato z [9])

Vyhledávání tak v každé úrovni šetří čas. B-strom zajišťuje dynamické změny v databázi.

Jakýkoli datový typ, který má lineární řazení může být použit jako indexové pole v B-stromu. V GIS indexových souborech jsou hodnoty čísel nebo textových řetězců užitečné pro hledání na atributových záznamech. Ale tato struktura neadresuje prostorově, jak je potřeba v mnoha dotazech v GIS.

Jsou různé alternativy modelu B-stromu, které dovolují, aby geometrické vlastnosti v databázi byly zahrnuty ve struktuře indexu. Například R-strom rozděluje prostor na řadu boxů, známých jako minimální ohraničující obdélník (MOO).



**Obrázek 23** Indexování dat za použití MOO Indexování dat za použití MOO (převzato z [9])

Obr. 23 ukazuje sérii umístění nemocnic a jsou použity 3 MOO k rozdělení prostoru. Uzly v indexu reprezentují tyto obdélníky a hledání umístění nemocnice bude řízeno algoritmem na jeden z nich. Hierarchická struktura obdélníků rozdílných velikostí MOO může být nastavena použitím stromové struktury na index. Vyhledávací algoritmus kontroluje, která entita je obsažena v největším obdélníku a potom se prochází větve stromu dolů do různých úrovní, až se najde, kde jsou splněny podmínky dotazu.

### Quadtrees

Základní koncepty quadtree byly vysvětleny dříve. Tato datová technika může být použita pro vektorová data, kde přítomnost nebo absence atributu určuje kód definovaného kvadrantu.

V některých složitých quadtree systémech se lze na buňky odkazovat na základě jednotek geometrie jako jsou hrany nebo vrcholy nebo počet bodů.

Quadtree struktury, které se používají ke kódování prostorových dat se geometricky mohou lišit, jak bylo popsáno dříve. Přesto v některých strukturách data samotná byla použita k určení pozice a tvaru podrozdělení. Rozdělení nebo pozice bodů, linií a vektorů může být použita k rozdělení prostoru na nepravidelné tvary v každém podrozdělení a tak pomáhat reprezentovat různost v hustotě a rozdělení datových hodnot v různých měřítkách. Strukturální tvar nepravidelného quadtree je vysoce závislý na době, kdy jsou různé body do něj vkládány. Hlavním problémem při použití quadtree u vektorových dat je ztráta explicitních topologických odkazů a že přesnost bodu a liniových feature je omezená.

### **Souvislé vektorové povrchy – dlaždicování**

Reálný svět je souvislý, nezastavuje se na hranicích mapových listů počítačových souborů. Bezešvá databáze simuluje kontinuitu spojením souborů pro sousední plochy systémem známým jako dlaždicování (tiling). Teoreticky je svět reprezentován nekonečnou řadou dlaždic jdoucích do obou směrů. Každá dlaždice nebo stránka, jak je někdy nazývána, se může odkazovat na jisté množství informací. Extra detaily mohou být umístěny dělením každé hlavní dlaždice do poddlaždic způsobem podobným jako u struktury quadtree. Dlaždicování přináší extra kompilace, protože všechny oblouky musí automaticky končit a začínat na hranici dlaždice a topologické pointery nesmí odkazovat pouze na jiné entity, ale i jiné dlaždice. Cenou je větší databáze, ale ve které čas vyhledávání je nízký na základě poziční hierarchie.

V principu dlaždicování neomezuje tvoření a uložení map, protože pouze data z několika málo dlaždic jsou používána v jeden časový okamžik a ostatní jsou uloženy na disku. Prakticky pouhý objem dat se zvětší, až budou finančně dostupné diskové prostory, dovolující dokonce dramatický pokles ceny hard disků. Proto je třeba klást důraz na síťové databáze nebo skladovací zařízení jako jsou CD-ROM, optické disky nebo magnetické pásky pro ukládání většiny celkové databáze, dokud je požadována.

### **Nový vývoj ve vektorových datových strukturách**

Jeden z problémů u konvenčních GIS struktur je, že úroveň detailu nebo rozlišení dat, která se mají uložit a zobrazit, je pevně daná a určená během vstupního stádia. Při práci s rozdílným mapovým měřítkem zůstává detail stejný, ale uživatelův pohled je omezen rozlišením

grafického displeje a stupněm zoomování. Byl vyvinut způsob datové struktury, která reaguje na úroveň detailu, a geometrická reprezentace je určena měřítkem displeje. Zoomování do velkého měřítka ukáže větší detail než reprezentace malého měřítka. Liniová entita může být zobrazena jako polygon v prvním případě, jako linie v případě druhém.

### **2.3.3 Databázové uložení vektorových a rastrových datových struktur**

Komerční databáze jako INFO, ORACLE, INGRES, INFORMIX a podobné produkty dovolily designérům rozdělit problémy správy prostorových dat na dvě části. První část je, jak reprezentovat geometrii a topologii prostorových objektů – půjde to udělat použitím vektorových nebo rastrových struktur? Druhá část je, jak manipulovat s atributy prostorových objektů, což se může dělat za použití komerčního systému řízení báze dat (RDBMS). Výsledné hybridní struktury (georelační modely) mají řadu odlišných výhod:

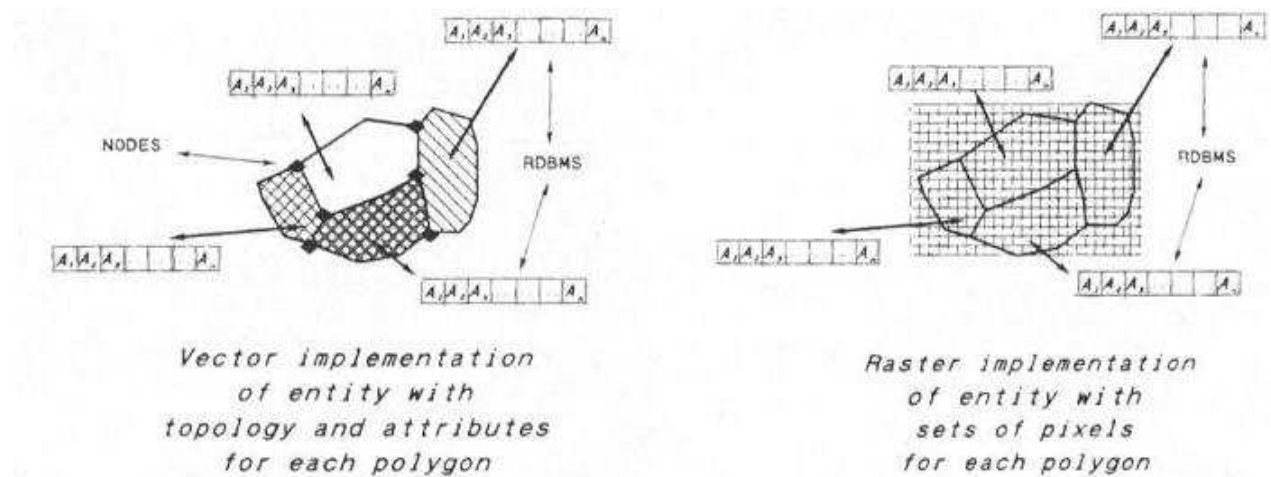
- a) atributová data nemusí být uložena spolu s prostorovou databází, ale mohou být kdekoli v systému nebo dokonce on-line na síti
- b) atributová data se mohou rozšiřovat, lze k nim přistupovat, mazat je, aktualizovat bez modifikace prostorové databáze.
- c) komerční RDBMS zajišťuje, že nový vývoj je zahrnut jako standard
- d) datové struktury mohou být definovány standardním způsobem za použití slovníků: data mohou být získána všeobecnými metodami jako je SQL (Standard Query Language), které jsou nezávislé na RDBMS
- e) držení atributových dat v RDBMS nezasahuje do základních principů vrstev v GIS
- f) atributy v RDBMS mohou být spojeny s prostorovými jednotkami, které mohou být reprezentovány širokou paletou způsobů

GIS designéři vytvořili varianty hybridních struktur za použití RDBMS.

#### **2.3.3.1 ARC-NODE – RDBMS**

Pravděpodobně to je nejpoužívanější databázový systém, ve kterém je použita plná vektorová topologie oblouk-uzel k popisu sítí linií a hranic polygonů. Každá prostorová jednotka

je identifikovatelná unikátním číslem nebo kódem a může být umístěna ve vybrané vrstvě nebo překryvu. Atributy jsou uloženy jako záznamy v relačních tabulkách řízených pomocí RDBMS. Některé z topologických informací a atributů jako souřadnice, sousedi, plochy, souřadnice minimální obdélníkové obálky a podobná data mohou také být uložena tabulkách v RDBMS. Obr. 24 a) ukazuje typický příklad takové datové struktury.



**Obrázek 24** Ekvivalentní hybridní vektorová (a - nalevo) a rastrová (b - napravo) datová struktura pro modelování "crisp" polygonů (převzato z [9])

Jednou komplikací u takových systémů je vytvoření nových prostorových entit a s nimi spojených atributů, kdy vrstvy se kříží, protože to znamená vytvořit nové sady záznamů a spojení v RDBMS.

### 2.3.3.2 Kompaktní rastr – RDBMS

Když jsou prostorové objekty reprezentovány sadou pixelů místo topologicky spojených linií, potom se může vytvořit rastrový ekvivalent k výše uvedenému. Když většina prostorových dat se odkazuje na tématické jednotky, které jsou interně homogenní, jako jsou jednotky choroplethové mapy, pak kvůli úspoře prostoru se použijí rastrové kompresní metody run-length kódování. Obr. 24 b) ukazuje příklad jednoduchého hybridního přístupu pro homogenní polygony.

### 2.3.3.3 Quadtree – RDBMS

Quadtree se může také použít pro kompresi dat, ale protože dovolují, aby data byla reprezentována v různých úrovních prostorového seskupení, dovolují také použít různou úroveň prostorového rozlišení v různých vrstvách. To může být užitečné, když se kříží data úzkého říčního údolí s širokou jednotkou územního plánování, což lze udělat ve vektorovém módu, ale

bylo by to velice obtížné v rastrovém systému, který dovoluje pouze velikost nebo stupeň rozlišení jedné buňky.

#### **2.3.3.4 Objekty – RDBMS**

V současné době byl již přijat objektově-orientovaný přístup pro organizaci jak rastrových tak vektorových datových struktur v jednom GIS. V těchto systémech jsou různá geometrická a atributová data uložena v relačních databázích a objektově-orientované programovací jazyky poskytují analytickou funkčnost stejně jako graficky objektově založený interface k datům. Systémy dovolují, aby výhody objektově-orientované organizace geografických dat byly využity uvnitř dobře známého prostředí relační databáze.

#### **2.3.4 Objektově orientované databázové struktury – sjednocení uložení atributů a geometrií**

Geografická data jsou charakterizována sérií atributů a hodnotami jejich chování, které definují jejich prostorové, grafické, dočasné a textové/numerické dimenze. Část definic atributů bude popisovat geometrickou povahu (bod, linii, polygon, buňku), více než jeden geometrický typ může být použit, aby odrážel rozdíly ve tvaru nalezeném v různých prostorových měřítkách.

Definují se atributy a proměnné jejich chování jako objektové třídy, v nichž jsou uloženy jejich vlastnosti a metody. Hierarchické vztahy mohou být nastaveny s různými třídami (např. ‚obloukový‘ objekt může být podtřídou ‚polygonového‘ objektu. Topologická spojení mezi různými objektovými instancemi a třídami jsou zavedeny explicitně přes objektové pointery a operátory jako je ‚směr‘, ‚protínání‘, ‚soused k‘, ‚překrytí‘, ‚nalevo od‘ nebo ‚napravo od‘.

Organizace prostorových dat v objektově-orientované databázi se ukázala být atraktivní pro jisté GIS uživatele, protože nabízí modelování a zpracování reálného světa integrovanějším a intuitivnějším způsobem než je možné v relačních systémech. Lidé zjistili, že tyto systémy poskytují přístup, který vyhovuje datovým typům, které používají a možnostem dotazování, které potřebují. Hierarchické strukturování a reprezentace relativně složitých vztahů mezi objektovými třídami se může přímo kontrolovat, což dává pružnost aktualizaci a změnám v databázi.

Strukturování databáze do samostatných, základních jednotek přináší s sebou i problémy i možnosti. Je velmi obtížné rozbít souvislé prostorové pole do oddělených jednotek. Jak rozbít vrch na řadu odlišných objektů, které mohou být použity k analýze a modelování? Výběr hranic je

často subjektivní. Přesto, kde toto data podporují a objektivní diskretizace je možná, tam se zvýší možnost výměny dat. Objekty mohou být použity v nových aplikacích dokonce, i když je požadováno odlišné strukturování. Zatímco znovu použitelné objektové knihovny vyžadují značnou investici do času a peněz, přesto ukázaly značnou návratnost. K prostorovým datovým objektovým knihovnám se může přistupovat přes internet.



### 3 Další matematické nástroje pro analýzy v GIS

Základem geografických informačních systémů je geografická databáze (prostorová databáze), která obsahuje geografická data s jejich prostorovou a atributovou složkou. Jak se popisuje ve zdroji [1] obecně jsou GIS tvořeny řadou programových modulů:

- *system řízení báze dat* – nejdůležitější vlastnost GIS je ukládat a obhospodařovat prostorové údaje
- *moduly pro sběr, resp. vstup údajů* – konverze a verifikace informací již v digitální formě
- *moduly pro transformaci (restrukturalizaci) údajů* – změna formy údajů, jako je generalizace, deklasifikace, změny měřítka, změny projekce, ...
- *moduly pro prostorové analýzy* – dotazování a prohledávání databáze, modelování procesů jako jsou statistické analýzy, analýzy sítí, analýzy obrazů apod.
- *zobrazovací moduly a moduly pro tvorbu výstupů* – úprava výsledků analýz do podoby map, diagramů, tabulek, grafů apod.
- *uživatelské prostředí* – překládání příkazů uživatele formy pochopitelné systémem

#### 3.1 Rozdělení analytických funkcí GIS

Ve zdroji [1] je podrobně rozebíráno několik způsobů rozdělení těchto funkcí. Názory nejsou jednotné, což dokládají např. následující dva způsoby dělení.

V [1] jsou funkce GIS *děleny podle Aronoffa, 1989*:

- Prohledávání
- Reklasifikace
- Měření
- Překrytí (topologické)
- Sousedské operace:
  - Prohledávání oblastí
  - Thiessenovy polygony (metoda interpolace používající hranice a myšlenky, že v daném dvourozměrném poli bodů se nejlepší informace o nenavštíveném bodě může odhadnout z k němu nejbližšího datového bodu viz [6])

- Topologické funkce
- Interpolace
- Generování izobar
- Konektivitní funkce:
  - Měření kontiguity
  - Vzdálenostní analýzy
  - Analýza sítí
  - Funkce šíření
  - Analýza viditelnosti
  - Osvětlování povrchů
  - Prostorové zobrazování

Ve zdroji [4] je uvedeno *dělení funkcí GIS podle Albrechta*:

- Prohledávání
  - Interpolace
  - Prostorové hledání
  - Tématické hledání
  - Deklasifikace
- Lokační analýzy
  - Buffer
  - Koridor
  - Překrytí
  - Thiessen/Voronoi
- Terénní analýzy
  - Sklon, aspekt
  - Povodí
  - Drenáž/síť
  - Viditelnost
- Dělení/sousednost
  - Cena/difuze/šíření
  - Blízkost (proximity)
  - Nejbližší soused
- Prostorová analýza
  - Multivariantní analýza
  - Rozpoznávání/disperse

- Vystředění/konektivita
- Tvar
- Měření

Podle *autorů zdrojů* [1] a [2] se jeví účelné následné dělení:

- Statistické analýzy
- Analýzy modelu terénu
- Analýzy obrazů
- Měřicí funkce
- Geografické analýzy:
  - Nástroje na dotazování databáze
  - Topologické překrytí
  - Mapová algebra
  - Vzdálenostní analýzy
  - Analýzy sítí

Toto dělení je v současné době nejaktuálnější a jak se vyvíjí matematické nástroje, tak nejlépe vystihuje funkce, které se používají v GIS, a proto bude podrobně popsáno a použito.

### **3.1.1 Statistické analýzy**

Slouží k potřebě prezentovat výsledky pomocí nejrůznějších ukazatelů a čísel (průměrná velikost parcely, četnost zarovnaných povrchů v závislosti na nadmořské výšce, ...). Je jich samozřejmě celá řada, z nichž nejpoužívanější jsou:

#### **3.1.1.1 Klasické statistické metody**

Pod tímto pojmem rozumíme statistické metody jako sumy, mediány, minima, maxima, standardní odchylky apod.

#### **3.1.1.2 Grafy**

Často se využije možnost data prezentovat pomocí grafů, kartodiagramů, kartogramů, histogramů apod.

#### **3.1.1.3 Regresní analýzy**

Jde o speciální případ statistických analýz, kdy se snažíme zpětně nalézt souvislost mezi jednotlivými prostorovými jevy.

*Příklad:*

Je známo, kde a kolik stromů napadl kůrovec. Chceme zjistit závislost mezi těmito místy a jejich charakteristikami (např. sklon, směr sklonu, typ lesa, stáří lesa, ...). Podle velikosti závislosti lze usuzovat na náchylnost místa ke kůrovcové kalamitě.

#### **3.1.1.4 Přímé napojení na statistické programy**

Je možné zpracovávat data v jiném statistickém programu jako je např. MS Excel, Statistica, SPSS, ...

### **3.1.2 Analýzy modelů terénu (povrchů)**

Provádí se nad digitálním modelem reliéfu (DMR) a je jedno, v jaké je reprezentaci, zda TIN nebo GRID. Odvozené parametry jsou mimořádně důležité při analýzách terénu z hlediska sjízdnosti různými dopravními prostředky, potenciálu eroze, příkonu slunečního záření, geomorfologické studie apod.

#### **3.1.2.1 Povrch sklonitosti svahu (*slope*)**

Ve zdrojích [1], [2], [3] a [7] je definován sklon takto:

Je dána topografická plocha a bod  $A = (x_0, y_0, z_0)$ , kterým plynule prochází jedna vrstevnice a jedna spádová křivka. Jednotkový normálový vektor v tomto bodě, orientovaný na vnější stranu plochy, označme  $N = (N_x, N_y, N_z)$ . Tečná rovina k topografické ploše v bodě A určená normálovým vektorem  $N$  má rovnici

$$N_x \cdot (X - x_0) + N_y \cdot (Y - y_0) + N_z \cdot (Z - z_0) = 0,$$

kde X, Y, Z jsou body tečné roviny.

Úhlem sklonu topografické plochy ve směru spádové křivky v jejím libovolném bodě nazveme úhel  $\lambda_N$  sevřený tečnou rovinou k topografické ploše a rovinou paralelní se skalární bází  $(x, y)$  v tomto bodě.

Povrch sklonitosti má dvě složky:

- *gradient* G (vertikální, maximální poměr změny výšky, někdy se používá pojem sklon i v tomto významu, měří se v % nebo ve stupních)

- *aspekt* A (horizontální orientace změny výšky, směr sklonu svahu, někdy se používá pojem expozice v tomto významu, měří se ve stupních).

$$G = ((\partial Z / \partial X)^2 + (\partial Z / \partial Y)^2)^{1/2} = \operatorname{tg} \lambda_N$$

$$\operatorname{tg} A = (-\partial Z / \partial Y) / (\partial Z / \partial X) \quad (-\pi < A \leq \pi),$$

kde  $Z = Z(X, Y)$  je výškové skalární pole,  $X, Y$  jsou souřadnice  $[X, Y] \in \Omega$  daného bodu, kterému je přiřazena výška  $Z$ .

### 3.1.2.2 Základní morfologické analýzy

Podle zdrojů [1] a [2] jde o nalezení *lokálních minim a maxim*, rovinné (horizontální) a profilové (vertikální) konvexnosti a konkávnosti (u nás vžitá označení *horizontální a vertikální křivost georeliéfu*, což jsou druhé derivace gradientu a aspektu).

Důležitý je prostorový průběh ortogonální sítě křivek (tvořené soustavou vrstevnic a soustavou spádových křivek) jak ve skalární bázi, tak i na topografické ploše v závislosti na prostorovém rozložení singulárních izolovaných pozitivních a negativních bodů a singulárních dvojných bodů, kterým na topografické ploše odpovídají vrcholové, depresní a sedlové body.

Normálovou křivostí  $\omega_s = K_N^n$  topografické plochy georeliéfu ve směru spádových křivek v každém jejím bodě  $A$  rozumíme křivost normálového řezu v bodě  $A$ , který je určen jako průsečnice topografické plochy s rovinou řezu obsahující normálu  $N$  topografické plochy a tečnu  $n$  k její spádové křivce v bodě  $A$ .

Normálovou křivostí  $\omega_v = K_N^t$  topografické plochy georeliéfu ve směru tečny k vrstevnici v každém jejím bodě  $A$  rozumíme křivost normálového řezu v bodě  $A$ , který je určen jako průsečnice topografické plochy s rovinou řezu obsahující normálu  $N$  topografické plochy a tečnu  $t$  k vrstevnici v bodě  $A$ .

Horizontální křivostí  $K_r$  topografické plochy georeliéfu v každém jejím bodě  $A$  rozumíme křivost horizontálního řezu v bodě  $A$ , který je určen jako průsečnice topografické plochy s rovinou řezu, která je rovnoběžná se skalární bází  $xy$  a obsahuje tečnu  $t$  k vrstevnici v bodě  $A$ .

Na základě dvojice křivostí  $\omega_s, \omega_v$ , resp.  $\omega_s, K_r$  můžeme z hlediska potřeb morfometrické analýzy georeliéfu vyjádřit geometrické tvary topografické plochy v jejím libovolném bodě  $A$ .

### 3.1.2.3 Analýza osvětlení reliéfu

Zdroj [2] a [7] popisuje osvětlení jako množství dopadajícího světla na jednotku plochy zemského povrchu za jednotku času a prostorové rozložení tohoto záření. Prostorovou a časovou bilanci dopadajícího slunečního záření na zemském povrchu určují dva základní faktory:

- poloha studované oblasti na Zemi určená zeměpisnými souřadnicemi  $(\varphi, \lambda)$ , zejména zeměpisnou šířkou  $\varphi$ , na které závisí výška Slunce nad rovinou horizontu
- reliéf Země, prostřednictvím svých morfometrických parametrů, z nichž nejvýznamnější je vliv sklonu reliéfu  $\lambda_N$  ve směru spádových křivek, orientace reliéfu  $A_N$  ke světovým stranám, nadmořské výšky  $z$  a dalších

Je vhodné použít tuto analýzu např. pro vyhledání nejlepší lokality pro pěstování vína, pro ověření hypotézy, že kůrovec má rád slunce či pro tvorbu estetických map se stínovaným terénem.

### 3.1.2.4 Tvorba vrstevnic (izočar)

Podle zdrojů [1] a [2] se jedná o převod mezi reprezentacemi DMR – vektorová linie. Vrstevnicemi jsou průsečnice TIN modelu georeliéfu s vodorovnými rovinami, které mají pravidelný rozstup  $\Delta Z$ . V případě TIN je v každém vrcholu prostorového trojúhelníku známá hodnota výšky.

Podle [7] metody triangulace dávají výsledný DMR podle toho, zda se jedná o:

- **datově nezávislé triangulace** – vytvářejí trojúhelníkové sítě v rovině, přičemž je ignorována výška bodů a body jsou dodatečně „zvednuty“ zpět do své správné výšky. Kvalita trojúhelníkových sítí, které vznikají touto cestou, nevyhovuje v těch místech terénu, kde se vyskytují příkré svahy.
- **datově závislé triangulace (DDT)** – při triangulaci je uvažována také informace o výšce. DDT vytvoří ve směru zlomu více „hubených“ trojúhelníků, natočených svojí nejdelší stranou téměř rovnoběžně s hranou zlomu. Větší počet trojúhelníků zajistí věrnější vystihnutí tvaru terénu. Takto vytvořený DTM dobře popisuje terén a taktéž dobře vypadají i výsledné vrstevnice.

Podle [8] během testování Delaunayovy triangulace (DT) byly nalezeny tyto chyby na vrstevnicích:

- vrstevnice leží na jedné hraně trojúhelníka a pak se vrací zpět po téže hraně
- vrstevnice mezi dvěma vrcholy těsně pod vrcholem se spojuje do jedné linie
- problém „příliš blízkých linií“
- chyba vznikající díky horizontálním trojúhelníkům (trojúhelníky mají vrcholy v téměř stejné výšce)

Model vytvořený pomocí Delaunayovy triangulace zlomy spíše vyhlazuje a k jejich vystihnutí je třeba podle [8] použít povinných hran – jsou to tzv. *constrained (omezené) triangulace*. Vstupem těchto triangulací je vedle reprezentativního bodového pole také množina takových hran, které musí být povinně použity v triangulaci, a to i za cenu, že v okolí povinné hrany nebude splněno kritérium kladené na výslednou trojúhelníkovou síť (prázdné opsané kružnice). Pak jsou výsledky lépe odpovídající skutečnosti.

V [8] je doporučena pro tvorbu vrstevnic *Delaunayova triangulace s použitím povinných hran*.

V [7] najdeme popis těchto metod tvorby vrstevnic:

- **Přímý výpočet vrstevnic** - nalezení vrstevnice s výškou  $Z_i$ : Procházíme celý TIN a pro každý trojúhelník zjišťujeme, zda obsahuje segment vrstevnice. Pokud ano, uložíme jej do seznamu. Segmenty vrstevnice jsou v seznamu zaznamenány nahodile, nestrukturovaně.
- **Předzpracování vrstevnic (preprocessing)** - u rozsáhlých DTM, ve kterých celkový počet trojúhelníků TIN je mnohem větší než počet trojúhelníků, přes které vrstevnice prochází, je přímá metoda neefektivní. Musí se vybudovat pomocná datová struktura (intervalový strom), pomocí které jsou vrstevnice zkonstruovány efektivněji.

### 3.1.2.5 Generování profilů

Zdroj [2] a [7] uvádí, že pomocí DMR je možné počítat profily liniových prvků. V mnoha aplikacích, např. při projektování komunikací nebo vodních toků, se tak používají pro generování pohledů na terén vertikální řezy (zpracování podélných profilů podél tras zadaných polygony bodů na terénu, dále výpočet příčných profilů v bodech zvolené množiny bodů podélného profilu).

### 3.1.2.6 Počítání objemů DMR

Zdroj [2] uvádí „*Cut and Fill*“ analýzu, což jsou změny objemu mezi dvěma DMR (např. kubatura obecného prostorového tělesa se počítá ze znalosti průběhu hraničních ploch), *analýzy reálné plochy a reálné délky na povrchu* (výsledkem je reálná plocha/délka na DMR a ne planimetrická ze 2D).

### 3.1.2.7 Analýza viditelnosti (z bodu, z linie)

Ve zdrojích [1], [2] a [3] se uvádí, že základem je dvojdimenzionální diagram – vzdálenost a čára pohledu sledující zdvih. Blízké věci se ale stávají překážkou pro věci vzdálené a tak každý viditelný objekt nastavuje novou hranici vertikálního úhlu podél čáry pohledu. Každý další vzdálenější objekt pak musí být ve vyšším vertikálním úhlu, pokud má být viditelný. Takto iterativně pracuje procedura směrem dál od výchozího bodu, která zároveň nese s sebou výsledek z předcházejících viditelných míst (lokací). Toto poslouží ke zjištění, které oblasti je možné a nemožné vidět z daného bodu, např. pro komunikační buňky mobilních telefonních služeb, pro přenos radiových vln nebo komunikační sítě, pak hovoříme o *analýze šíření signálu*.

Ale nové lokace podél poněkud rozdílných linií pohledu vyžadují interpolaci podél horizontu. Tato horizontální linie se aktualizuje, jak proces zpracování postupuje dál. Zároveň se zaznamenává nejvyšší úhel pohledu. Tak se simuluje např. pohled z dálnice nebo řeky. Lze tak navrhnout např. minimální počet radiových vysílačů, aby bylo pokryto celé území.

Ze sady viditelností pak mohou získat např. údaj, jak často je vidět dané místo z linie (třeba dálnice – vyhledání nejvhodnější lokality pro reklamu).



### **3.1.2.8 Identifikace významných bodů a linií (hřebenů, údolnic), rozpoznání geomorfologických tvarů**

Ve zdroji [1] a [3] je také tato úloha uvedena mezi analýzami povrchů. V rastrovém modelu se hledají významné body povrchu pomocí masky. Maskou 3x3 určíme vrcholy (bod je vrcholem, když má všechny sousedy nižší) a deprese (všichni sousedé jsou vyšší), sedla a přechody. Maskou 2x2 určíme hřebeny (bod je na hřebenu, pokud jeho hodnota není nižší při žádné její poloze masky) a údolnice (hodnota bodu není vyšší při žádné poloze masky). Nakonec se propojují vrcholy přes hřebeny a deprese přes údolnice. Tím získáme kostru terénu.

### **3.1.2.9 Modelování eroze**

Podle zdrojů [1] a [3] sehrává klíčovou úlohu v tomto modelování výpočet délek svahů a geomorfologických forem terénu.

Ze všech křivek na topografické ploše georeliéfu procházejících bodem  $A$  jsou důležité křivky ortogonální sítě, tvořené spádnicemi a vrstevnicemi. Ve směru spádových křivek probíhají procesy svahové modelace georeliéfu. Na konvexních formách reliéfu ve směru spádnic probíhají erozní procesy a na konkávních formách akumulární procesy svahové modelace. Samozřejmě záleží také na hodnotě sklonu  $\lambda_N$  ve směru spádnic a orientaci reliéfu  $A_N$  ke světovým stranám.

### **3.1.2.10 Modelování směrů proudění**

Směr proudění je maximální gradient z hodnot dané buňky do okolních. Směr proudění, rychlost proudění a další operace z analýz proudění jsou základem pro analýzy povodí, hydrologické analýzy a modelování eroze.

Podle zdrojů [3] a [7] je pro každý bod uvnitř trojúhelníka TIN tento směr stejný. Ve vrcholech a v bodech na hranách TIN je třeba uvažovat incidující prvky (trojúhelníky, hrany) a vybrat správný směr. Na základě znalosti směru vodního toku v libovolném bodě TIN, můžeme definovat cesty vodních toků i místa jejich spojení.

Odvodňovací síť je formována všemi hranami údolnic a všemi cestami proudění, které vycházejí z těch níže položených vrcholů údolnicových hran, ve kterých nenavazuje žádná další hrana údolnice a které nejsou depresními body. To vyplývá z faktu, že se voda na reálném terénu začíná akumulovat na údolnicích. Uzly odvodňovací sítě reprezentují

soutoky, kde se dva nebo více vodních toků spojuje a pokračuje dále. Protože předpokládáme, že v každém bodě TIN je směr proudění vody jednoznačný, nemůže dojít k rozdělení vodního toku.

### **3.1.3 Analýzy obrazů**

Analýzy obrazů souvisí s pořizováním dat pro GIS a jsou podrobně probírány ve zdrojích [1], [2], [3] a [5].

#### **3.1.3.1 Filtrace, roztažení histogramu**

Podle zdrojů [3] a [5] se používají ke zvýraznění nejrůznějších charakteristik v obrazu tyto techniky. Základem je histogram, který ukazuje poměrné zastoupení hodnot šedi (0 – 255).

Při běžném snímkování, obsahuje histogram hodnoty využívající asi třetinu hodnot celého rozsahu. Tato informace se dá jednoduchými způsoby zviditelnit pomocí transformace stupňů šedi přes *lineární roztažení histogramu, equalizaci histogramu nebo jiné roztažení histogramu*.

Další technikou je filtrace, která zajistí např. potlačení šumu, zmírnění rozostření obrazu, korekce z digitalizace, zvýraznění kontrastu mezi objekty nebo postklasifikační filtrace. Není vhodné řešit úlohy najednou v celém obraze, ale posouvá se okénko s okolím pixelu a pro daný pixel se vypočte nová hodnota v závislosti na hodnotách sousedních pixelů v okénku. Posouváním okénka se modifikují hodnoty obrazové funkce v celém obraze. Filtry se dělí na *prosté průměrování, průměrování pro liniové prvky, metodu nejbližšího souseda, průměrování s inverzním gradientem* a řadu dalších filtrací (uvedeny ve zdroji [5]).

#### **3.1.3.2 Vyrovnání jasu/kontrastu mezi snímky, mozaikování apod.**

Ve zdrojích [3] a [5] se uvádí, že jde o nástroje pro vyrovnání přechodů mezi jednotlivými snímky. Tento postklasifikační filtr musí být založen na logickém operátoru a ne na aritmetických operacích (průměr ze tříd 3 a 5 je 4, což je v tomto případě nesmysl). Příkladem je majoritní filtr. Pohyblivé okno se posouvá přes data a je určena majoritní třída. Jestliže v okolí pixelu je hodnota majoritní třídy a ve středním pixelu není, změní se hodnota tohoto pixelu na majoritní. Jestliže v okně není hodnota majoritní třídy vůbec, hodnota středního pixelu se nezmění.

Podle zdroje [5] existují i algoritmy, které mohou zachovávat hranice mezi oblastmi typů povrchu, příp. minimální plochy pro daný typ povrchu, což je prostor pro prostorově orientované klasifikátory.

### 3.1.3.3 Metody klasifikace obrazů

Podle zdrojů [1], [2], [3] a [5] jde o statistické metody a metody umělé inteligence, které umožňují na zpracovávaném obrazu identifikovat homogenní oblasti.

Ve zdroji [1] je stručně popsána *metoda analýzy hlavních komponentů*, kdy pro dvě pásma údajů definujeme nový souřadnicový systém - nové osy – jednu ve směru delšího rozměru rozpětí hodnot a druhou na ni kolmou. Tím se omezí redundance údajů multispektrálního obrazu (korelace mezi pásmy) a zároveň jde o kompresi dat.

Ve zdrojích [1], [2], [3] a [5] je dále popsána *metoda neřízené klasifikace*, kdy se nepoužívá trénovací množina. Klasifikace vychází z principu, že pixely ze stejných nebo příbuzných ploch mají blízkou spektrální odezvu, a proto se shlukují do shluků. Výsledkem jsou spektrální třídy. Po klasifikaci je nutné výsledky konfrontovat s výsledky pozemního šetření tedy vykonat interpretaci výsledků.

Jako další je ve zdrojích [1], [2], [3] a [5] popsána *metoda řízené klasifikace*, kdy interpretátor si vybírá tzv. trénovací plochy jako vzory pro jednotlivé třídy. Pak následuje zjištění příznakového prostoru, protože obrazy jsou charakterizovány vektorem, jehož souřadnice tvoří hodnoty jednotlivých příznaků. Při vhodném výběru příznaků je podobnost předmětů v každé třídě vyjádřena geometrickou blízkostí jejich obrazů. Následuje samotná klasifikace. Nejznámější jsou tyto tři druhy klasifikace:

- klasifikátor založený na minimální vzdálenosti – vypočítá se průměr ze spektrálních hodnot pro každou třídu pro každý kanál (pásmo spektra). Hodnota v daném pixlu se porovnává s těmito průměry a ke kterému má nejbližší, do jeho třídy se zařadí
- rovnoběžníkový klasifikátor – zjišťuje se, zda hodnota pixlu padne do prostoru vymezeného minimální a maximální hodnotou v každém kanálu (pásmu spektra). Pracuje se vlastně s vícerozměrnými čtyřúhelníky
- Gaussovský klasifikátor – je založen na předpokladu, že rozdělení bodů tvořících jednu třídu tréninkových dat má Gaussovo, tedy normální rozdělení. Můžeme pak spočítat statistickou pravděpodobnost dané hodnoty pixelu jako člena vybrané třídy. Funkce hustot pravděpodobnosti se užívají, aby klasifikovaly pixel pomocí výpočtu pravděpodobnosti, že patří do dané třídy.

Zjistí se pravděpodobnost, že se hodnota pixelu vyskytuje ve všech pásmech a pixel se zařadí do té třídy, kde má největší pravděpodobnost (nebo nikam, pokud je menší než hodnota prahu)

### 3.1.4 Měřicí funkce

Tyto funkce jsou stručně popsány ve zdroji [2]. Systémy poskytují funkce na měření vzdáleností a ploch

Při měření je třeba dát pozor na zobrazení:

- Konformní – zachovává úhly
- Ekvivalentní – zachovává plochy
- Ekvidistantní – zachovává délku určité soustavy křivek (poledníky, rovnoběžky, ...)
- Kompenzační – dochází k deformaci všech geometrických prvků (úhlů, délek i ploch), ale hodnoty deformací nejsou extrémní

Systémy umí efektivně používat různé délkové jednotky (stopy, cm ...) a automatické převody mezi nimi. GIS při projektech v malém měřítku (většího plošného obsahu) počítají i s vlivem zakřivení zemského povrchu.

### 3.1.5 Nástroje na dotazování databáze

Ve zdrojích [1] a [2] je popsána tato možnost analýz v GIS. Prohledávání databáze je základem všech analytických nástrojů a postupů v GIS. Dotazováním se vybírají údaje splňující kritérium. V dotazu musí být specifikovány údaje, kterých se týká, dále formulovány podmínky, které údaj musí splňovat a nakonec následuje instrukce, co se má s údaji nebo nad údaji vykonat.

Dotazy dále dělíme na atributové, prostorové a kombinované.

#### 3.1.5.1 Atributové dotazy

Atributový dotaz (např. které objekty mají definovanou vlastnost?) lze uskutečnit:

- Identifikací objektu na základě jeho jména, označení nebo jiného atributu
- Vyhledáním všech objektů splňujících intervalové a logické kritérium jednoho nebo více atributů

U *vektorové reprezentace* se zpracovávají údaje atributových tabulek připojených k jednotlivým vektorovým objektům.

U *rastrové reprezentace* se zpracovávají údaje v buňkách jednotlivých vrstev.

### 3.1.5.2 Prostorové dotazy

Prostorový dotaz (např. co se nachází v této oblasti? ) lze uskutečnit:

- Identifikací geografického objektu na základě zadání jeho souřadnic (ručně nebo interaktivně ukázáním na objekt myši)
- Prohledáváním v prostoru geometrických tvarů jako obdélníky, kružnice, polygony, linie, ...

U *vektorové reprezentace* je vybrán vždy celý objekt.

U *rastrové reprezentace* je vybrána vždy konkrétní buňka nebo skupina buněk.

### 3.1.5.3 Kombinované dotazy

Kombinovaný dotaz (např. které objekty mají tuto vlastnost a zároveň se nachází v této oblasti? ) na rozdíl od atributového nebo prostorového dotazu pracuje najednou i s více vrstvami než jednou. Používá pro ně operátory Booleovské logiky a také zčásti topologické překrývání vrstev, které bude popsáno dále.

## 3.1.6 Topologické překrytí

Zdroje [1] a [2] uvádí, že dotazování více vrstev je vlastně topologickým překrytím, které se v GIS řeší pomocí algoritmů počítačové grafiky (test bodu v polygonu, hledání průsečíku dvou objektů, ořezávání). Výsledkem jsou nové objekty s žádanou kombinací vlastností.

Systémy nabízí tyto operace:

- INTERSECT (=AND) – průnik
- UNION (=OR) – sjednocení
- IDENTITY (=AND NOT) – přiřazení na základě prostorového umístění
- CLIP – oříznutí podle druhé vrstvy
- ERASE – odstranění dat pod druhou vrstvou, opak CLIP
- SPLIT – rozdělení vstupní vrstvy podle hranic ve druhé vrstvě
- UPDATE – nahradí ve vstupní vrstvě část, pro kterou jsou aktualizovaná data ve druhé vrstvě
- DISSOLVE – rozpuštění hranic na základě stejného atributu

- MERGE – spojení dvou vrstev do jedné a odstranění hranic mezi objekty se stejnými atributy

### 3.1.7 Mapová algebra

Tento nástroj je popsán ve zdrojích [1] a [2] a umožňuje kombinovat rastrové vrstvy pomocí různých matematických operací nad jednou nebo více vrstvami. Jde o počítačový jazyk, který používá operátory, funkce, objekty, činnosti.

*Operátory* – vykonávají matematický výpočet uvnitř nebo mezi rastrovými reprezentacemi, konstantami a jejich kombinacemi.

*Funkce* se dělí na:

- Lokální – vykonávají se v dané buňce, vypočítává se nová hodnota z hodnot v 1 nebo více vrstvách
- Fokální – vykonávají se v okolí buňky vypočítáním jedné nové hodnoty z hodnot v okolí
- Zonální – vykonávají se na specifické oblasti vrstvy, nové hodnoty se vypočítávají v zónách daných jednou vrstvou z hodnot v druhé vrstvě
- Globální – týkají se všech buněk vrstvy

*Objekty* – slouží k uložení informací nebo jsou to hodnoty (rastry, tabulky, konstanty, ..)

*Činnosti* – vykonávají na vstupních objektech příkazy (operátory a funkce jazyka)

*Kvalifikátory a parametry* – řídí, jak a kde se vykoná činnost (pomocné konstrukce jazyka, podmínky, cykly, ...)

### 3.1.8 Vzdálenostní analýzy

Tato problematika je popsána ve zdrojích [1], [2] a [3]. Na první pohled jednoduchá problematika se ve spolupráci s mapovou algebrou rozvine ve velice složitý a sofistikovaný nástroj.

#### 3.1.8.1 Buffer

Jde o nejpoužívanější nástroj vzdálenostních analýz.

Ve *vektorové reprezentaci* spočívá ve vytvoření polygonů v určené vzdálenosti kolem bodů, linií a polygonů, které jsou jako standardní vrstva.

V *rastrové reprezentaci* se označí všechny buňky, které jsou od daného objektu v menší vzdálenosti než definovaná. Vzdálenost je jen otázkou metriky. Podobně se počítá vzdálenost každé buňky od požadovaného objektu, což se opět uloží do nové vrstvy a je možno ji dále reklasifikovat. Tato analýza se nazývá nalezení vzdálenosti.

### 3.1.8.2 Analýzy sousedství (proximity analysis)

Tyto analýzy spočívají v tom, že se vytvoří plochy kolem každého ze vstupních bodů tak, že definují prostor, který je nejbližší ke každému ze vstupních bodů. Pro tento výpočet se používá metody *Thiessenových polygonů* a *Voronoiových diagramů*. Provádí se v rastrové i ve vektorové podobě.

### 3.1.8.3 Vážená vzdálenost (weighted distance)

Tato analýza hledá nejméně nákladnou cestu z každé buňky do nejlevnějšího z objektů. Napodobuje reálný svět pomocí různých faktorů:

- Faktor terénního reliéfu  $L$  – převádí rovinnou vzdálenost mezi buňkami na vzdálenost po reliéfu
- Vertikální (stupňovitý) faktor  $V_{ab}$  – bere v úvahu vliv gradientu mezi dvěma sousedními buňkami na jejich nákladovou impedanci, je anizotropní (z A do B je jiný než z B do A)
- Horizontální faktor  $H_a, H_b$  – udává účinek převládajícího horizontálního směru působení faktoru na nákladovou impedanci buňky
- Frikční povrch buňky  $F_a, F_b$  – náklady v buňce, které jsou potřeba k jejímu překonání
- Další faktory podle konkrétního použitého matematického modelu

Rovnice vypadá takto:

$$D = 0,5 \cdot L \cdot V_{ab} \cdot H_a \cdot F_a \cdot H_b \cdot F_b$$

Z rovnice pak vznikne povrch nákladů (cost surface) a z něho se pak počítá vážená vzdálenost jako nejlevnější cesta z každé buňky do pro ni nejlevnějšího zdroje (objektu).

Používá se např. pro modelování šíření ohně, nalezení nejlepší lokality pro výstavbu v oblasti, kde náklady rostou se vzdáleností stavby od cest, inženýrských sítí, ...

#### **3.1.8.4 Nalezení nejlevnější cesty**

Jde o nalezení nejlevnější cesty mezi dvěma buňkami. Vstupem je povrch nákladů a obraz obsahující cílový objekt. Pak se pomocí metody vážené vzdálenosti vypočítá vlastní vážená vzdálenost od cílového objektu. Nakonec pomocí analýzy nalezení nejlevnější cesty se zjistí nejlevnější koridor z počátku do cíle.

#### **3.1.9 Analýzy sítí**

Tato problematika je zmiňována ve zdrojích [1], [2] a [3]. V podstatě se jedná o hledání nejkratší vzdálenosti, ale s tím rozdílem, že sítě jsou reprezentovány vektorově. Síť tvoří orientovaný ohodnocený graf z uzlů a hran. Před využíváním síťových analýz je nutné vytvořit všechny datové struktury – liniovou vrstvu, nad kterou se budou provádět analýzy a uzlová a hranová pravidla pro pohyb mezi uzly, uložená v atributových tabulkách a také je možné přiřadit další atributy pro výstupy z analýz (jméno ulice, název křižovatky, ...)

Pravidla pak umožní simulovat následující vlastnosti:

- Cena cesty – základní atribut hrany (maximální rychlost, čas cesty, vzdálenost, ...)
- Směrování – přikázané směry jízdy, zákazy, přiřazení cen za provedení změny směru, ...
- Definice neuzlových bodů – modelování podjezdů a nadjezdů

##### **3.1.9.1 Hledání konektivity**

Jde o hledání všech propojených prvků k a od daného uzlu např. nalezení všech vedení a odběratelů postižených vyhořelým transformátorem.

##### **3.1.9.2 Modelování zatížení sítě**

Jde např. o analýzu transportu vody/splavenin ve vodních tocích, pohyb plynu v potrubích na základě objemu, průřezu, sklonu, tlaku.

##### **3.1.9.3 Hledání optimální trasy**

Jde o nalezení optimální trasy mezi dvěma a více body (ve stanoveném pořadí nebo bez) na základě ceny cesty (vzdálenost, čas, ...)



#### **3.1.9.4 Hledání cesty do nejbližšího zařízení**

Jde o nalezení optimální trasy do nejbližšího (optimálního) zařízení např. nalezení optimální trasy k nehodě pro nejbližší sanitku.

#### **3.1.9.5 Alokace zdrojů**

Jde o vyhledání všech lokalit, které jsou od vybraného objektu vzdáleny nějakou cenou cesty např. vzdálenost do 30 minut od vyhlášené vzdálenosti. Výsledkem jsou izochrony, což jsou čáry spojující body se stejným časem k dosažení výchozího bodu.

## 4 Voronoiovy diagramy

### 4.1 Základní vlastnosti Voronoiových diagramů

Ve zdroji [10] jsou popsány základní vlastnosti Voronoiových diagramů, které budou uvedeny dále.

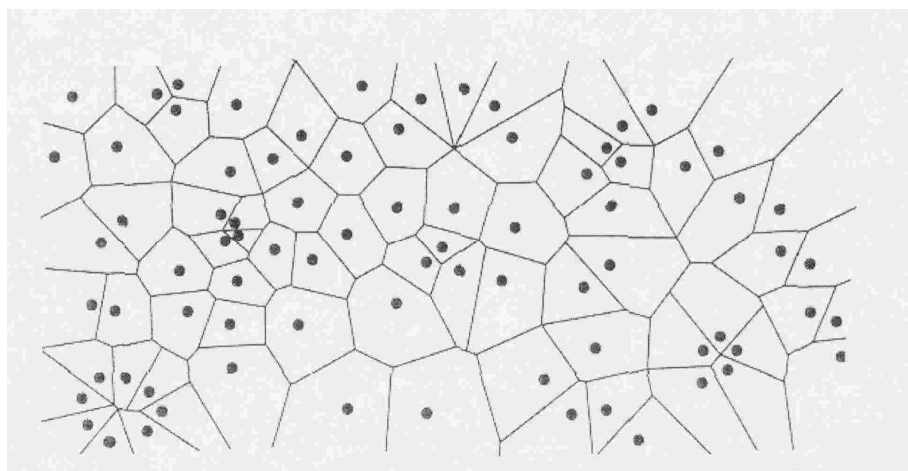
*Definice:* Necht'  $S$  je množina  $n$  bodů (zvaných míst „sites“) v rovině. Pro dvě různá místa  $p, q \in S$ , je *dominance  $p$  nad  $q$*  definována jako podmnožina roviny, která má stejně blízko k  $p$  jako k  $q$ . Formálně

$$\text{dom}(p, q) = \{x \in R^2 \mid \delta(x, p) \leq \delta(x, q)\}, \text{ kde } \delta \text{ je funkce Eukleidovské vzdálenosti}$$

$\text{dom}(p, q)$  je uzavřená polorovina ohraničená kolmým bisektorem mezi  $p$  a  $q$ . Tento bisektor odděluje všechny body roviny bližší k  $p$  od těch bodů bližších k  $q$  a nazývá se *separátor  $p$  a  $q$* . Oblast místa  $p \in S$  je část roviny ležící ve všech dominancích  $p$  nad ostatními místy v  $S$ . Formálně

$$\text{reg}(p) = \bigcap_{q \in S, q \neq p} \text{dom}(p, q)$$

Protože oblasti vznikají protínáním  $n-1$  polorovin, jsou to konvexní polygony. Takže hranice tvoří maximálně  $n-1$  hran (maximálně otevřených přímých segmentů) a vrcholů (jejich koncové body). Každý bod na hranici je stejně vzdálený přesně od dvou míst a každý vrchol je stejně vzdálený přinejmenším od tří. Oblasti tvoří polygonální část roviny, která se nazývá *Voronoiov diagram,  $V(S)$* , z konečné množiny bodů  $S$  viz obr. 25



**Obrázek 25** Voronoiov diagram (převzato z [12])

Oblast zvaná  $\text{reg}(p)$  nemůže být prázdná, protože obsahuje alespoň všechny body roviny stejně daleko k  $p$  jako k jakémukoli jinému místu v  $S$ , zvláště  $p \in \text{reg}(p)$ . To znamená, že  $V(S)$  obsahuje přesně  $n$  oblastí. Některé z nich jsou samozřejmě neuzavřené. Ty jsou

definovány oblastmi ležícími na hranici konvexní obálky  $S$ , protože právě pro tyto oblasti existují bod libovolně vzdálené, ale ještě nejbližší. Konvexní obálka  $S$  je nejmenší konvexní polygon, který obsahuje  $S$ . Jinak tři nebo více hranic se potkají ve společném vrcholu. Je třeba si uvědomit, že každý vrchol je středem kružnice, která prochází přinejmenším 3 místy a neobsahuje žádné místo uvnitř.

Každá z  $e$  hran má dva vrcholy a každý z  $v$  vrcholů patří přinejmenším do tří hran, tedy  $2e \geq 3v$ . Z Eulerovské relace  $n + v - e \geq 2$  vyplývá, že  $e \leq 3n - 6$  a  $v \geq 2n - 4$ .

Lineární chování velikosti Voronoiova diagramu v rovině znamená, že tato struktura není o nic víc složitější než základní konfigurace míst. To je první důvod pro jejich časté použití. Druhý důvod je, že  $V(S)$  porovnávají celou informaci o sousedství o  $S$  explicitním a počítačově použitelným způsobem. Například jeho aplikace na problém poštovního úřadu je založena na triviálním zjištění, že bod  $x$  spadá do oblasti bodu  $p$  tehdy a jen tehdy, jestliže  $p$  je nejbližší  $x$  mezi všemi místy v  $S$ . Navíc, jestliže místo  $p$  je nejbližší místu  $q$ , pak  $\text{reg}(p)$  a  $\text{reg}(q)$  sdílejí společnou hranu. Z toho vyplývá, nejbližší pár míst v  $S$  dává vzniknout některé hraně  $V(S)$ .

## 4.2 Zobecněné Voronoiovy algoritmy

Voronoiovy diagramy mohou být zobecněny různými způsoby.

### 4.2.1 Vyšší dimenze

Toto zobecnění je uvedeno ve zdrojích [10], [11] a [12]. Je použita sada bodů ve vícedimenzionálním prostoru. Pro sadu  $n$  bodů a  $d$  počet dimenzí je v prostoru  $E^d$  maximální kombinatorická složitost Voronoiova diagramu  $\Theta(n^{d/2})$  a může být spočítána v  $O(n \log n + n^{d/2})$  optimálním čase. Duálem je stále triangulace.

### 4.2.2 Jiná metrika

Toto zobecnění je uvedeno ve zdrojích [10], [11] a [12]. Jsou použity jiné metriky než Eukleidovská..

#### **$L_1$ nebo Manhattanská**

$$\text{dist}_1(p, q) = |p_x - q_x| + |p_y - q_y|$$

Ve Voronoiově diagramu v  $L_1$  metrice jsou všechny hrany – horizontální, vertikální nebo diagonální v úhlu  $45^\circ$  se souřadnými osami.

### $L_p$ nebo obecná metrika

$$\text{dist}_p(p, q) = \sqrt[p]{(|p_x - q_x|^p + |p_y - q_y|^p)}$$

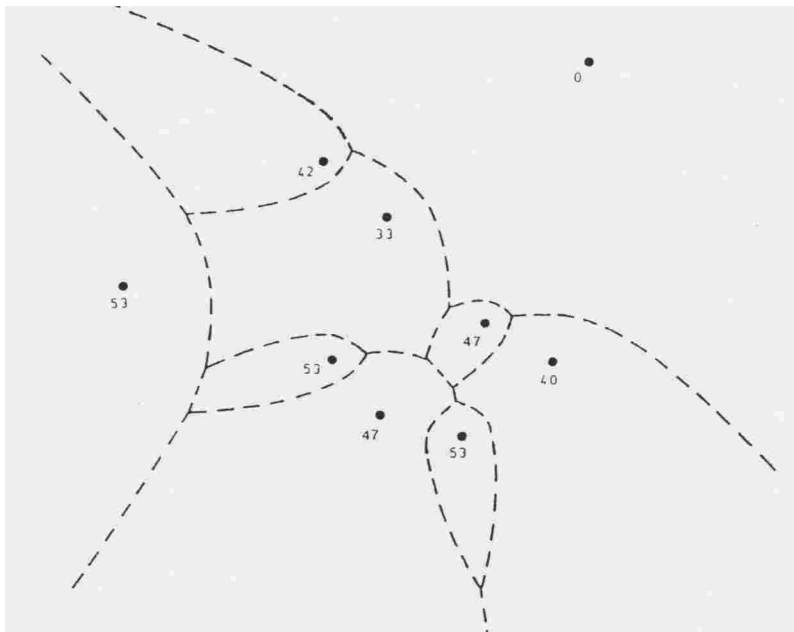
$L_2$ -metrika je jednoduše Eukleidovská metrika.

### 4.2.3 Použití vah

Toto zobecnění je uvedeno ve zdrojích [10], [11] a [12]. Jiná vzdálenost vznikne, když se bodům přiřadí váhy (aditivní neboli vážené, multiplikační anebo jiné).

#### 4.2.3.1 Aditivní (vážené) váhy

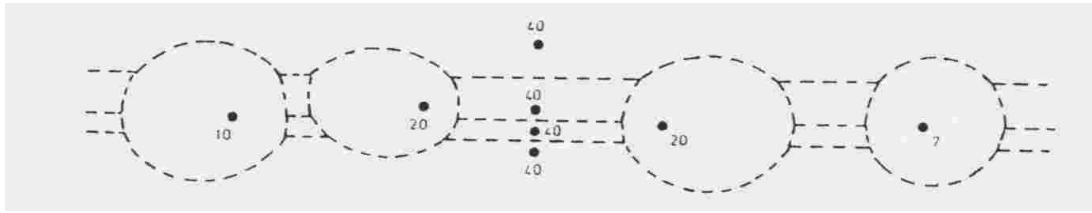
Výsledkem je Johnson-Mehlův model viz obr. 26.



Obrázek 26 Johnson-Mehlův model (převzato z [10])

#### 4.2.3.2 Multiplikační váhy

Výsledkem je Apolloniův model viz obr. 27. Separátor dvou bodů v rovině popisuje jejich Apolloniův kruh.

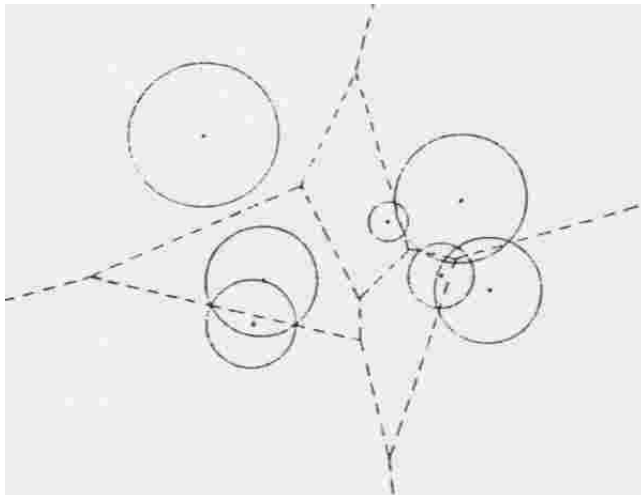


**Obrázek 27** Apolloniův model (převzato z [10])

#### 4.2.3.3 Power diagram

$$\text{dist}^2(p, q) - w$$

$w$  vyjadřuje možnost vlivu bodu  $p$  na své okolí. Zvláště dvojice  $(p, w)$  může být interpretována jako koule („sféra“) se středem  $p$  a poloměrem  $\sqrt{w}$ , jestliže  $w > 0$  viz obr. 28.



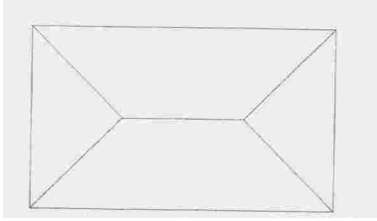
**Obrázek 28** Power diagram pro sedm kružnic (převzato z [10])

#### 4.2.4 Jiné tvary míst

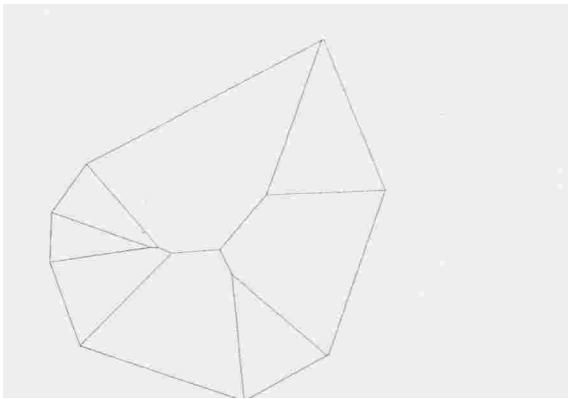
Toto zobecnění je uvedeno ve zdrojích [10], [11], [12] a [13].

##### 4.2.4.1 Hrany polygonu

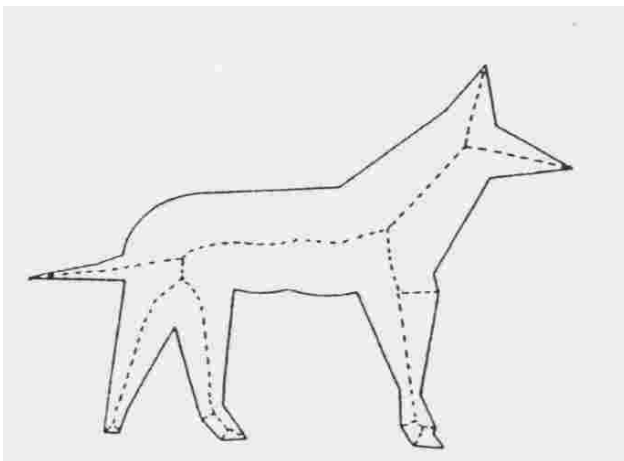
V tom případě se používá střední osa polygonu  $P$ , což je množina bodů uvnitř  $P$ , které mají na hranici  $P$  více než 1 nejbližší bod viz obr. 29, 30 a 31.



**Obrázek 29** Střední osa obdélníka (převzato z [13])



**Obrázek 30** Střední osa konvexního polygonu o osmi vrcholech (převzato z [13])

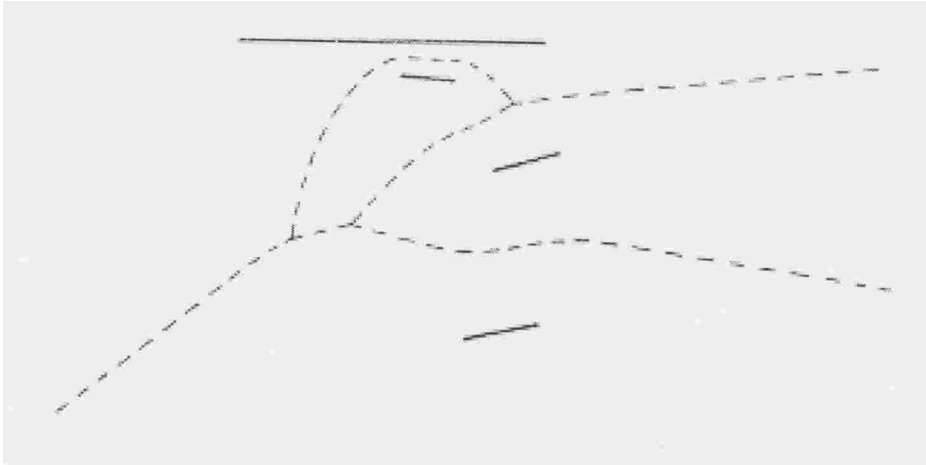


**Obrázek 31** Obrys a jeho střední osa (převzato z [13])

Střední osa polygonu byla použita nejprve k biologickým účelům viz. obr. 31, později pro rozpoznávání obrazů a počítačové vidění. Střední osa obecného polygonu o  $n$  vrcholech může být vytvořena v čase  $O(n \log n)$ , pro konvexní polygony stačí čas  $O(n)$ .

#### 4.2.4.2 Úsečky

Místem jsou úsečky viz. obr. 32.



**Obrázek 32** Voronoiův diagram, když místem jsou úsečky (převzato z [12])

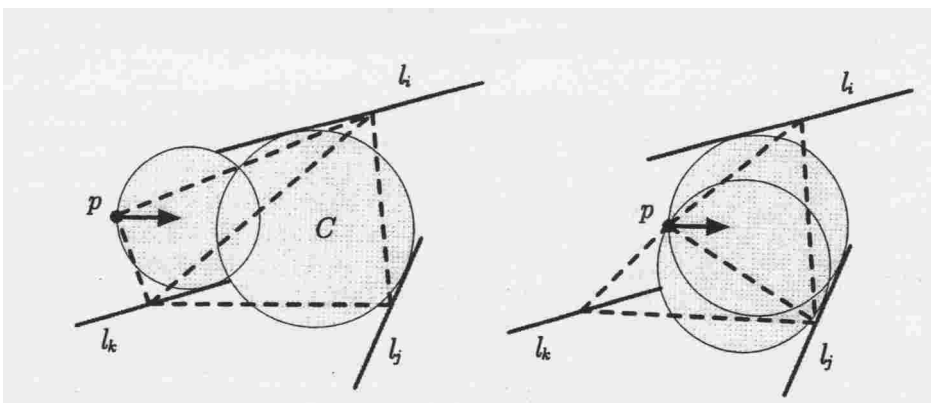
#### 4.2.5 Změna prostoru

Toto zobecnění je uvedeno ve zdrojích [10] a [12]. Voronoiovy diagramy se tvoří na kouli, toroidu, mnohostěnu, kuželi apod.

#### 4.2.6 Dynamizace

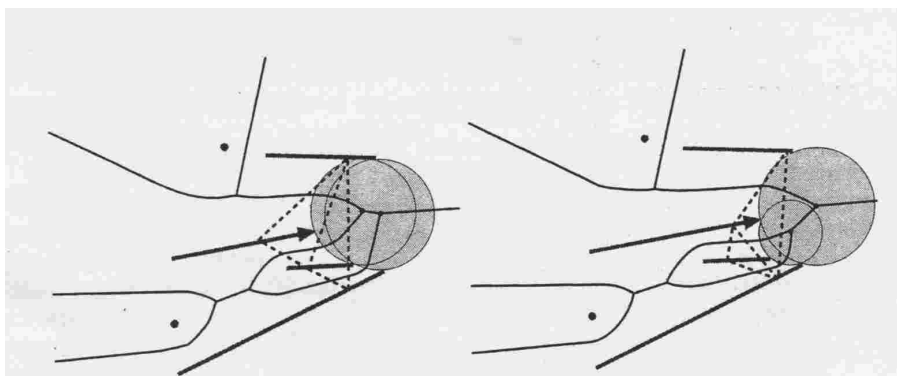
Toto zobecnění je uvedeno ve zdrojích [10] a [12]. V tomto případě stačí, že se pohybuje jeden bod podél přímé linie uvnitř statické scény  $S$  liniových segmentů. Jak se  $p$  pohybuje, Voronoiův diagram se průběžně mění, ale pouze v jistých kritických časech nastanou topologické události, které způsobí změnu topologie. Topologická struktura se může změnit, když se objeví/zmizí Voronoiův bod nebo když se objeví změna na hranici konvexní obálky.

##### 4.2.6.1 Pohyb míst



**Obrázek 33** Čtyřúhelník před a po swapování (převzato z [14])

#### 4.2.6.2 Vkládání a vybírání bodu/linie

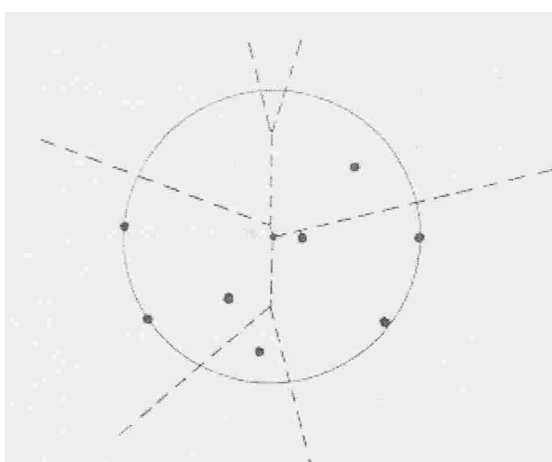


**Obrázek 34** Voronoiův diagram během dynamické operace (převzato z [14])

#### 4.2.7 $k$ nejbližších míst

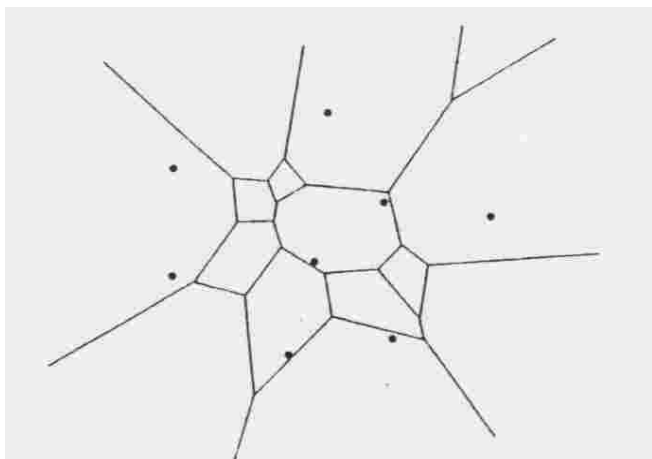
Toto zobecnění je uvedeno ve zdrojích [10], [11] a [12]. Lze dělit prostor podle  $k$  nejbližších míst, pro nějaké  $k$ , pro které platí  $1 \leq k \leq n - 1$ , kde  $n$  je počet bodů. Jde pak o Voronoiův diagram vyššího řádu (řádu  $k$ ). Voronoiův diagram 1. řádu je standardní Voronoiův diagram. Voronoiův diagram  $(n-1)$ . řádu je Voronoiův diagram nejvzdálenějšího bodu, protože Voronoiova buňka bodu  $p_i$  je nyní oblast bodů, pro které je  $p_i$  nejvzdálenějším bodem.

Maximální složitost Voronoiova diagramu vyššího řádu pro  $n$  bodů v rovině je  $\Theta(k(n - k))$ . Výpočet Voronoiova diagramu vyššího řádu pomocí nejznámějšího algoritmu se provede v čase  $O(n \log^3 n + k(n - k))$ .



**Obrázek 35** Voronoiův diagram nejvzdálenějšího bodu ( $k=7$ ) spolu s nejmenší opsanou kružnicí (převzato z [12])





**Obrázek 36** Voronoiův diagram řádu 2 (převzato z [10])

### 4.3 Algoritmy pro tvorbu Voronoiových diagramů

#### 4.3.1 Protínání polorovin

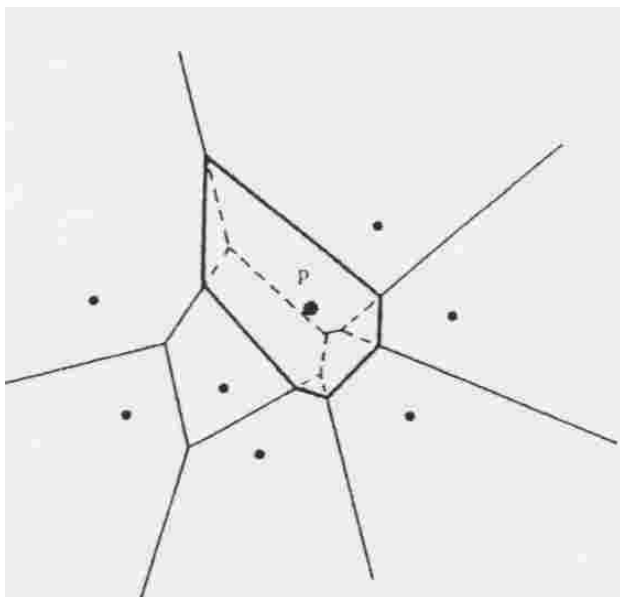
Tento algoritmus je uveden ve zdroji [13]. Voronoiův diagram lze zkonstruovat (odděleně) protínáním  $n - 1$  polorovin podle následující rovnice:

$$V(p_i) = \bigcap_{i \neq j} H(p_i, p_j)$$

Konstrukce protínáním  $n$  polorovin je duální k úloze konstruování konvexní obálky  $n$  bodů ve dvourozměrném prostoru a tak lze použít podobné algoritmy s časem  $O(n \log n)$ . Při použití pro každé místo by byl čas  $O(n^2 \log n)$ .

#### 4.3.2 Inkrementální (přírůstková) konstrukce

Tento algoritmus je uveden ve zdrojích [10], [12], [13] a [19]. Voronoiův diagram pro  $k$  bodů je již zkonstruován a přidává se další místo  $p$  a konstruuje nový Voronoiův diagram. Jde o dva kroky. První krok znamená nalézt oblast, kde  $p$  leží. Nechť  $q$  je místo definující tuto oblast. Separátor  $p$  a  $q$  tak vytvoří hranu  $e$ , která patří do oblasti místa  $p$ . Ve druhém kroku se tvoří hranice oblasti místa  $p$  a tvoří se hrana po hraně s tím, že se začíná od hrany  $e$ . Během tohoto procesu části starého diagramu nejbližší k  $p$  (čárkovaně) jsou přerušeny a odmazány. Jde o části, které jsou na  $p$  straně separátorů  $p$  a jeho nových sousedů viz obr. 37.

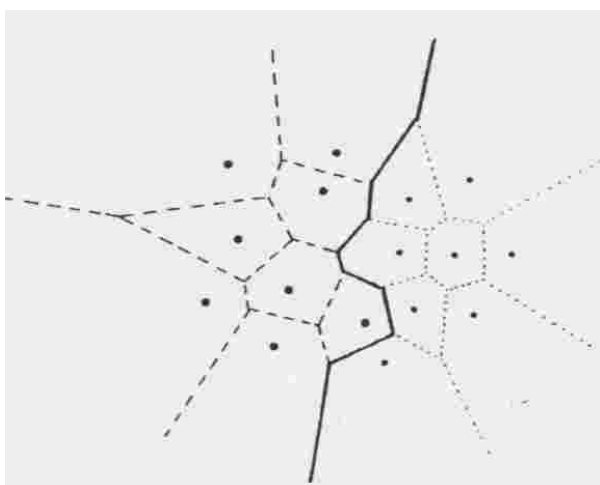


**Obrázek 37** Vkládání Voronoivy oblasti (převzato z [10])

V nejhorším případě se výpočet provede v čase  $O(n^2)$ . Jde zlepšit na očekávané  $O(n)$ .

### 4.3.3 Algoritmus „Rozděl a panuj“

Tento algoritmus je uveden ve zdrojích [10], [11], [13] a [19]. Daná množina  $n$  bodů je rozdělena vertikálními liniemi na dvě podmnožiny. Diagramy pro tyto dvě podmnožiny se vypočítají rekurzivně a pak jsou spojeny, aby vytvořily celkový diagram. Obr. 38 ukazuje řetěz hran (tučně), které jsou konstruovány během procesu spojování diagramu levých míst (čárkovaně) s diagramem pravých míst (tečkovaně).

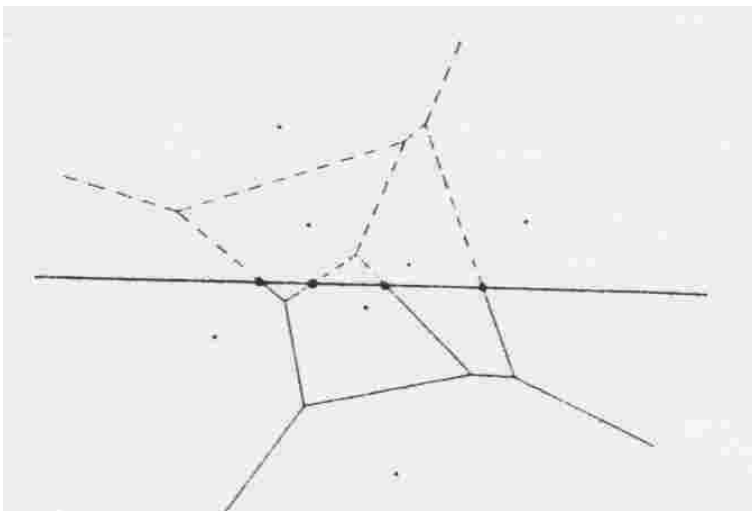


**Obrázek 38** Spojení dvou Voronoiových diagramů (převzato z [10])

Aby byl krok dělení proveden vyváženým způsobem, bude potřeba čas  $O(n \log n)$ . Dva diagramy mohou pak být spojeny v čase  $O(n)$ .

#### 4.3.4 Fortuneho algoritmus (technika plane-sweep)

Tento algoritmus je uveden ve zdrojích [10], [11] a [13]. Plane-sweep technika zmenšuje dimenzi problému. Statický problém výpočtu Voronoiova diagramu v rovině je redukován na dynamický problém získávání průsečíků diagramu s přímkou (sweeping line). Algoritmus simuluje posouvání linie po rovině odzdoła nahoru. V jakémkoli časovém bodě je část diagramu pod touto linií kompletní viz obr. 39.



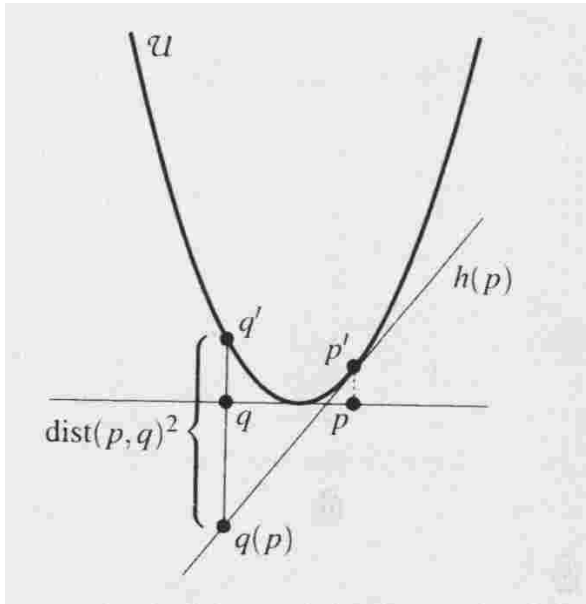
**Obrázek 39** Zpracování protínání s linií (převzato z [10])

Výpočet diagramu se provede v  $O(n \log n)$ .

#### 4.3.5 Konvexní obálka

Tento algoritmus je uveden ve zdrojích [10], [11] a [13]. Voronoiov diagram je sada bodů v rovině. Existuje úzký vztah mezi rovinným Voronoiovým diagramem a průsečíky horních polorovin v třídímenzionálním prostoru.

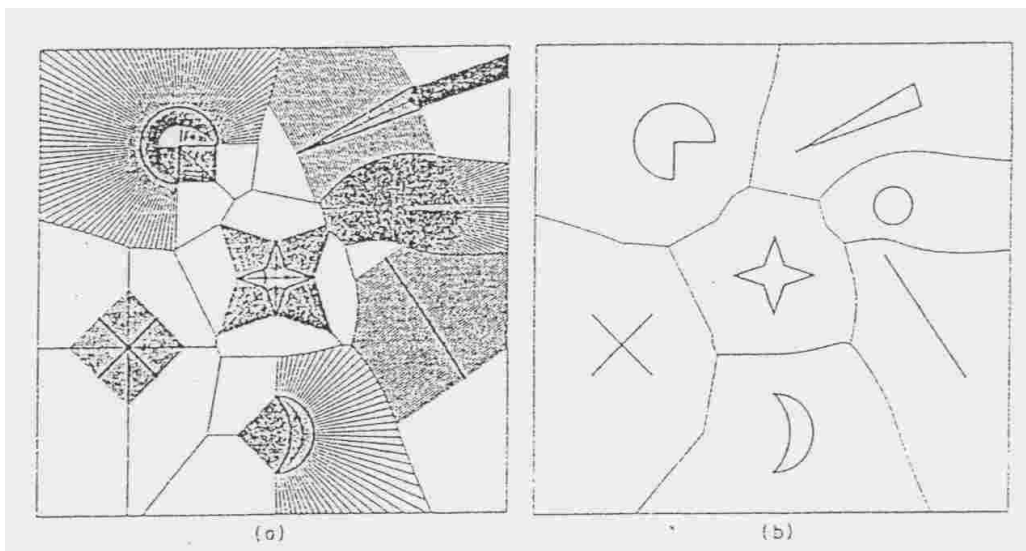
Nechť  $P$  je sada bodů v třídímenzionálním prostoru a všechny leží v rovině  $z = 0$ . Nechť  $H$  je sada rovin  $h(p)$  pro  $p \in P$ . Pak projekce horní obálky rovin  $UE(H)$  v sadě rovin  $H$  na rovinu  $z = 0$  je Voronoiov diagram pro  $P$  viz obr. 40.



**Obrázek 40** Voronoiův diagram jako projekce horní obálky rovin na rovinu  $z=0$  (převzato z [11])

#### 4.3.6 Aproximace pomocí bodů

Uvedeno podle zdroje [15]. Je sada  $n$  samostatných obrázků  $A_1, A_2, \dots, A_n$  v rovině. Pro každý z obrázků vytvoříme konečnou sadu  $P_i$  bodů, které aproximují hranici obrázku  $A_i$ . Pak se tvoří Voronoiův diagram se všemi sadami bodů. Následuje vymazání těch hran, které vznikly přidáním bodů na hranici a vymažou se izolované vrcholy. Zůstane Voronoiův diagram pro původní obrázky viz obr 41.



**Obrázek 41** Aproximace generátorů sadou bodů: a) Voronoiův diagram pro 882 bodů, které aproximují sedm generátorů; b) aproximace zobecněného Voronoiova diagramu získaného smazáním nadbytečných hran a vrcholů z a) (převzato z [15])

### 4.3.7 Aproximace digitálním obrazem

Uvedeno podle zdroje [15]. Je sada  $n$  samostatných obrázků  $A_1, A_2, \dots, A_n$  v rovině. Každý obrázek je nahrazen sadou pixelů. Z těchto pixelů - generátorů se současně a rovnoměrně vytvářejí jejich teritoria, dokud nedojde ke kolizi s teritoriem jiného generátoru. Toto se dělá opakovaním lokální paralelní operace nad sousedními body pixelu. Výsledkem je opět Voronoiův diagram.

## 4.4 Aplikace Voronoiových diagramů v GIS

Voronoiovy diagramy lze použít v aplikacích územního plánování, jako tržní oblasti, ekologické kontaktní modely, prostorové interpolace, v kartografii, jako možnost umístění, pro nalezení optimálního dělení na školní nebo volební obvody, v geografické variační analýze, grafech konektivity, pro řešení problému obchodního cestujícího, apod.

### 4.4.1 Nalezení nejbližšího souseda (problém poštovního úřadu, lokace bodu)

Tato aplikace je uvedena ve zdrojích [10], [12] a [13]. Z definice Voronoiova diagramu platí, že Voronoiova oblast daného místa obsahuje všechny body bližší tomuto místu než k jakémukoli jinému místu. Tím se redukuje problém na nalezení oblasti, která obsahuje dotazovaný bod. Dnes se již používají vhodné datové struktury.

#### 4.4.1.1 Hledání všech nejbližších sousedů

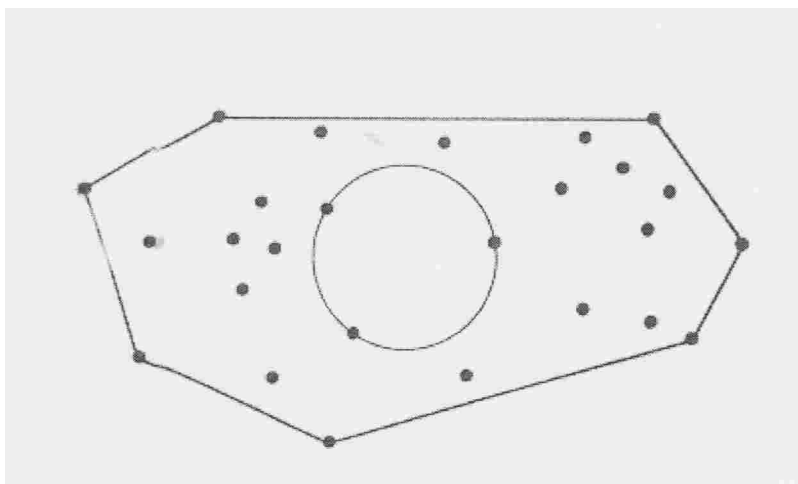
Je dáno  $n$  bodů v rovině, hledá se ke každému bodu  $p_i$  jeho nejbližší soused. Nejbližší sousedé sdílejí hranu ve Voronoiově diagramu, pro každý bod  $p_i$  se musí projít hrany jeho oblasti.

#### 4.4.1.2 Hledání nejbližší dvojice

Je dáno  $n$  bodů v rovině, hledají se z  $n$  bodů ty dva body, jejichž vzdálenost je minimální.

### 4.4.2 Největší prázdný kruh (možnost umístění)

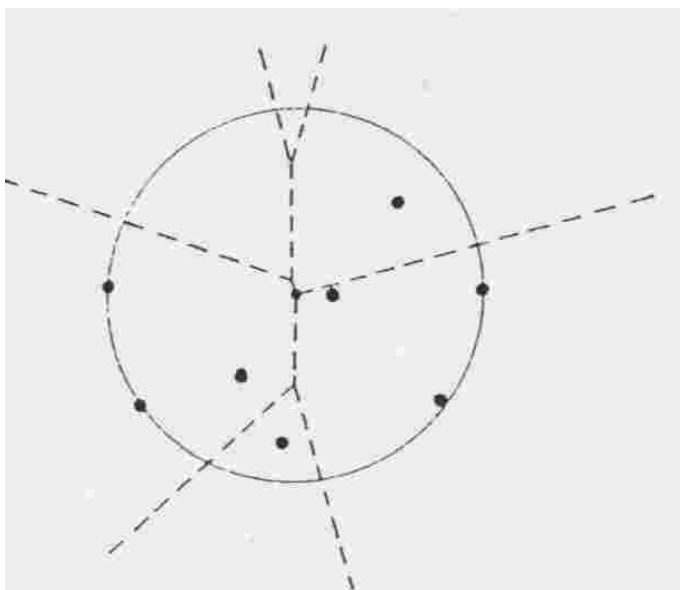
Tato aplikace je uvedena ve zdrojích [10], [12] a [13]. Jde např. o hledání místa pro nový obchod, tedy hledá se umístění, jehož vzdálenost k nejbližšímu obchodu je největší tzn. nový obchod bude umístěn ve středu největší prázdné kružnice viz. obr. 42.



**Obrázek 42** Největší prázdný kruh (převzato z [12])

#### 4.4.3 Nejmenší kružnice obsahující všechny body

Tato aplikace je uvedena ve zdrojích [10]. Tato kružnice vždy bude mít střed ve vrcholu nejdelšího místa Voronoiova diagramu viz obr. 43.

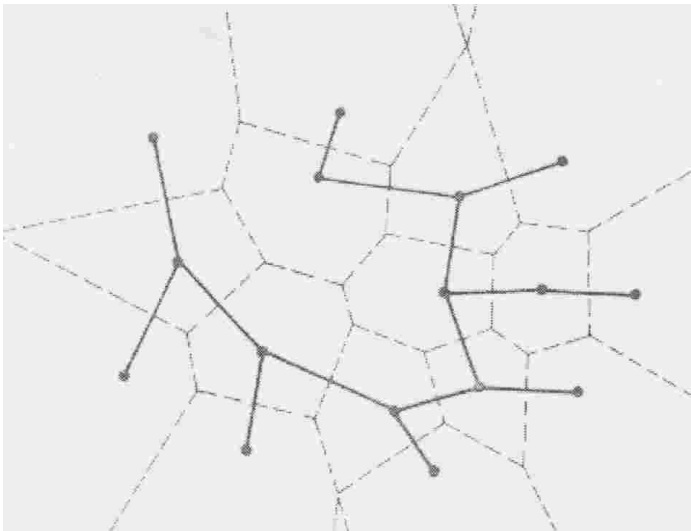


**Obrázek 43** Nejvzdálenější místo a nejmenší kružnice obsahující všechny body (převzato z [10])

#### 4.4.4 Minimal spanning tree (MST)

Tato aplikace je uvedena ve zdrojích [10], [12], [13] a [16]. MST znamená strom s minimální délkou, procházející všemi body. Pokud se délka hrany měří eukleidovskou délkou segmentu, pak hovoříme o Eukledovském MST (EMST). MST má mnoho použití např. je to síťová

topologie, která minimalizuje např. celkovou délku drátů a obvykle minimalizuje jak cenu, tak časové zpoždění viz. obr. 44.

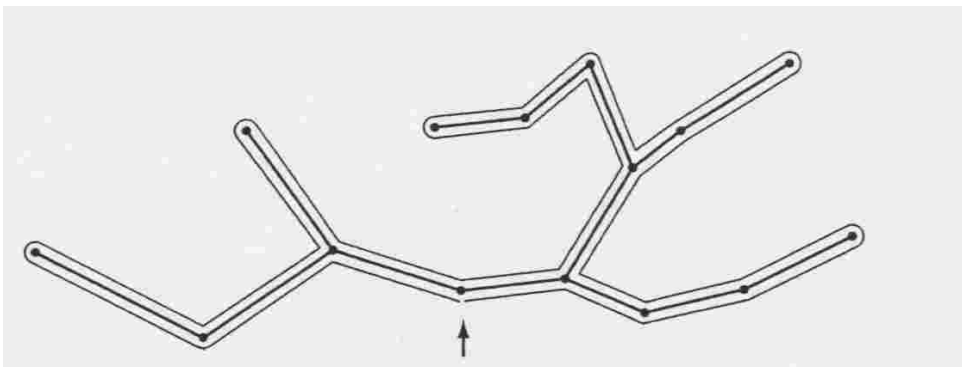


**Obrázek 44** Strom s minimální délkou (převzato z [12])

Je znám také Kruskalův algoritmus, který vychází z předpokladu, že nejkratší strom se skládá z nejkratších hran, proto se strom konstruuje inkrementálně. Hovoří se o tzv. greedy (žravém) algoritmu.

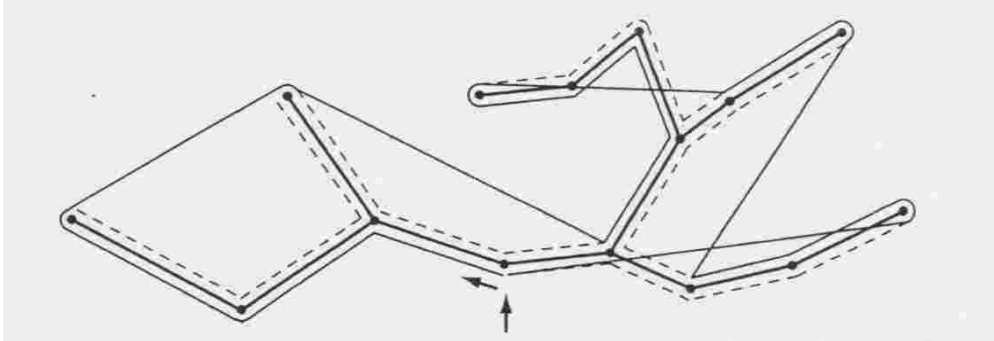
#### 4.4.5 Příklad obchodního cestujícího

Tato aplikace je uvedena ve zdrojích [10], [12] a [13]. Jde o úlohu najít nejkratší uzavřenou cestu  $n$  danými body v rovině. Problém je ale  $NP$ -obtížnosti, proto se používají heuristiky nebo přibližná řešení. Jeden z nejjednodušších přibližných algoritmů je projít EMST dvakrát viz obr. 45.



**Obrázek 45** Cesta vytvořená stromem s minimální délkou (převzato z [13])

Při použití heuristiky pokud bod, kam se má jít, byl již navštíven, přeskočí se a hledá se další ještě navštívený bod. Dojde tak ke zkrácení dvojnásobné cesty. Zkrácená cesta se ale může křížit viz obr. 46.



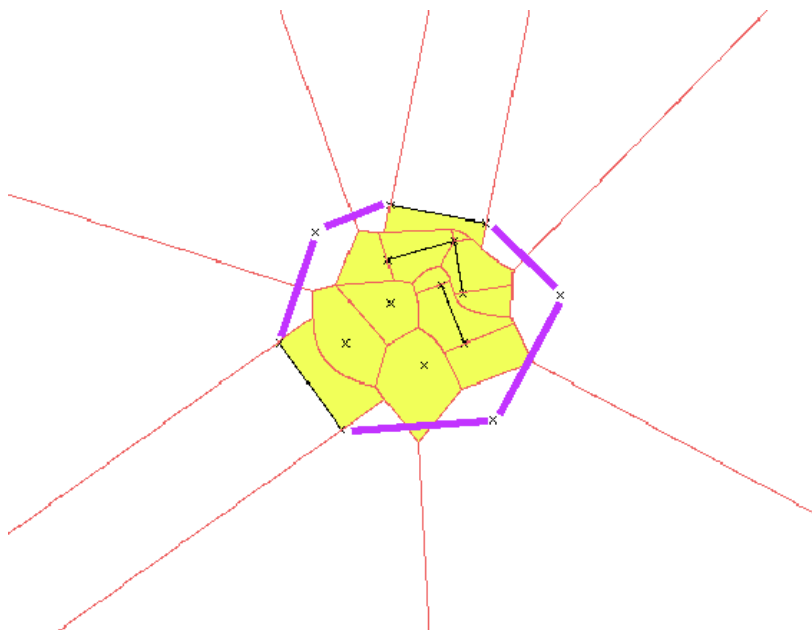
**Obrázek 46** Zkrácení dvojnásobné cesty (převzato z [13])

#### 4.4.6 Střední osa (middle axis)

Tato aplikace je uvedena ve zdrojích [10], [12] a [13]. Střední osa polygonu  $P$  je množina bodů uvnitř  $P$ , které mají na hranici  $P$  více než 1 nejbližší bod viz obr. 29, 30 a 31.

#### 4.4.7 Konvexní obálka

Tato aplikace je uvedena ve zdroji [16]. Konvexní obálka obrazce je nejmenší konvexní obrys, který jej obsahuje viz obr. 47.

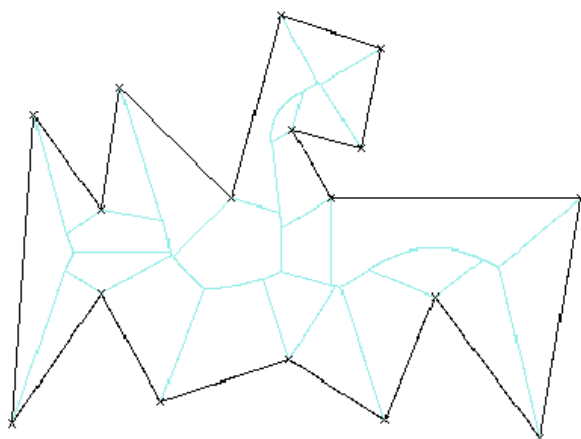


**Obrázek 47** Konvexní obálka (převzato z [16])



#### 4.4.8 Skeleton (kostra)

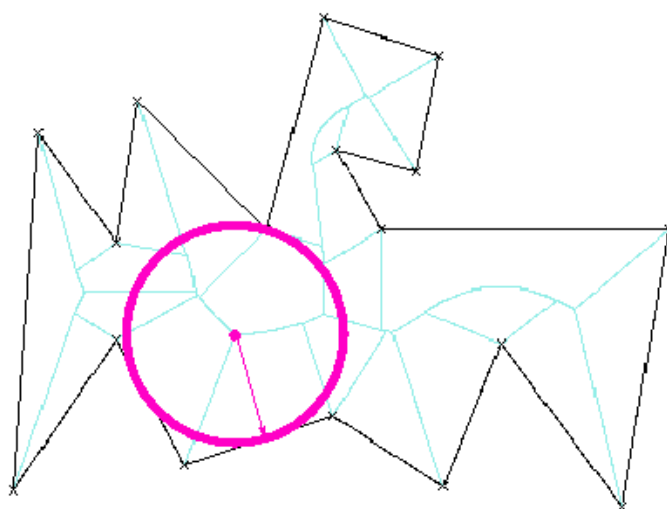
Tato aplikace je uvedena ve zdroji [16] viz obr. 48.



**Obrázek 48** Kostra (převzato z [16])

#### 4.4.9 Centroid

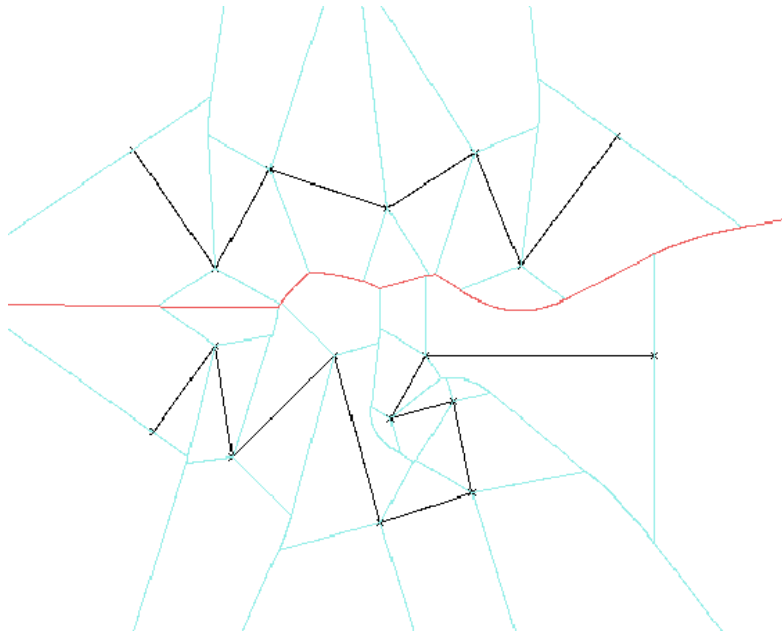
Tato aplikace je uvedena ve zdroji [16]. Centroid je bod uvnitř kontury, který poskytuje jistý druh služby vzhledem k celému povrchu. Pro konvexní tvary je těžiště často vybráno jako reprezentativní centroid. Pro složité tvary však může těžiště ležet vně kontury. U Voronoiových diagramů je jakýkoli bod skeletonu jistým centroidem. Na obr. 49 je zvýrazněn jeden, což je uzel s největší  $dMax$ .



**Obrázek 49** Centroid (převzato z [16])

#### 4.4.10 Hranice (frontier)

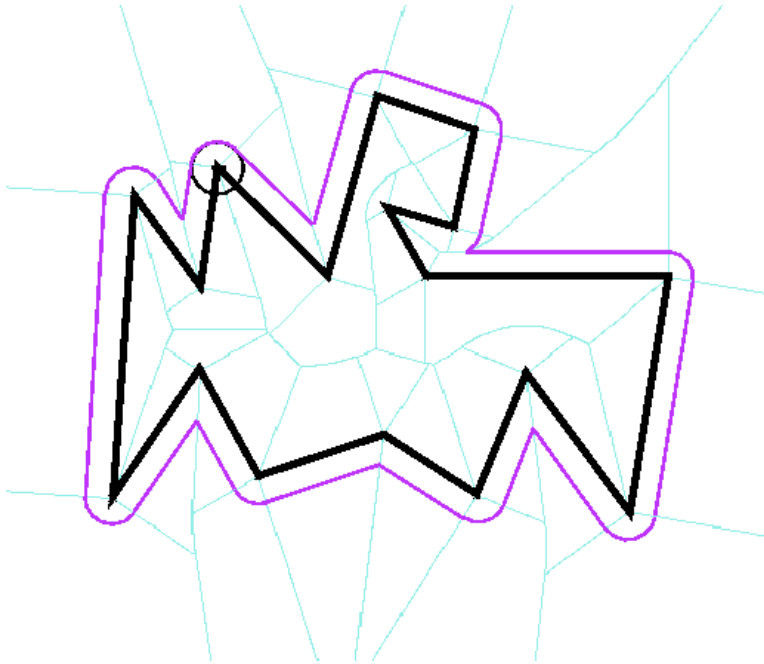
Tato aplikace je uvedena ve zdroji [16]. Hranice mezi dvěma objekty (dvě pobřeží, dvě banky, atd.) může být definována jako místo bodů stejně vzdálených od objektů. Jednoduše se vybere *interface* Voronoiových hran viz obr. 50.



**Obrázek 50** Hranice (převzato z [16])

#### 4.4.11 Roztažení (dilation)

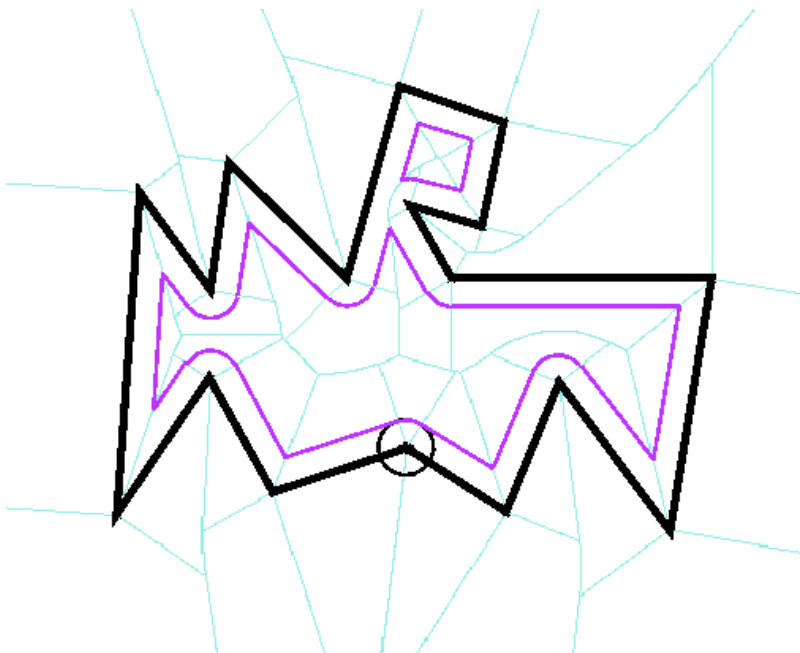
Tato aplikace je uvedena ve zdroji [16]. Roztažení spočívá v provádění vnějšího odvalování kružnice s poloměrem  $r$  a se středem na kontuře a zaznamenání trasy vnějšího obalu kružnice viz obr. 51.



**Obrázek 51** Roztažení (převzato z [16])

#### 4.4.12 Stažení (erosion)

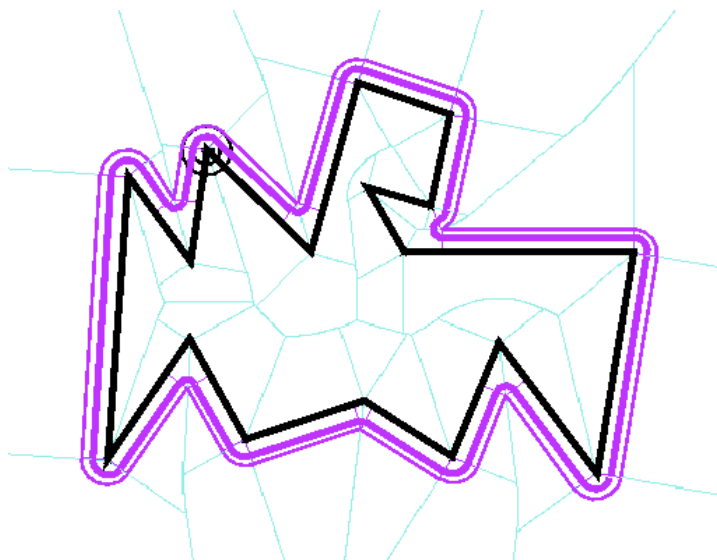
Tato aplikace je uvedena ve zdroji [16]. Stažení spočívá v provádění vnitřního odvalování kružnice s poloměrem  $r$  a se středem na kontuře a zaznamenání trasy vnitřního obalu kružnice viz obr. 52.



**Obrázek 52** Stažení (převzato z [16])

#### 4.4.13 Uzavření (closure)

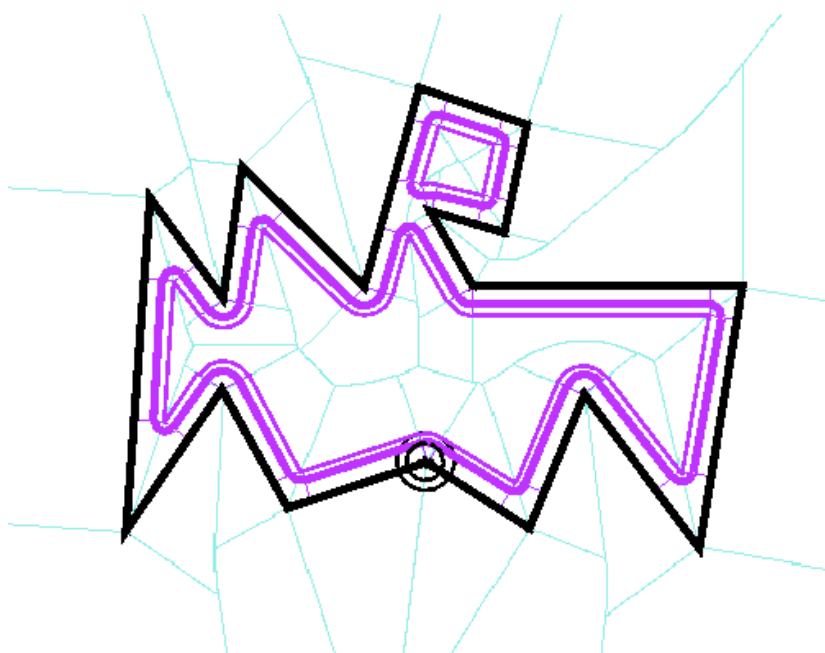
Tato aplikace je uvedena ve zdroji [16]. Jde o roztažení s poloměrem  $r_1$  a pak stažení s poloměrem  $r_2$ ,  $r_2 < r_1$  viz obr. 53.



Obrázek 53 Uzavření (převzato z [16])

#### 4.4.14 Otevření (opening)

Tato aplikace je uvedena ve zdroji [16]. Jde o stažení s poloměrem  $r_1$  a pak roztažení s poloměrem  $r_2$ ,  $r_2 < r_1$  viz obr. 54.



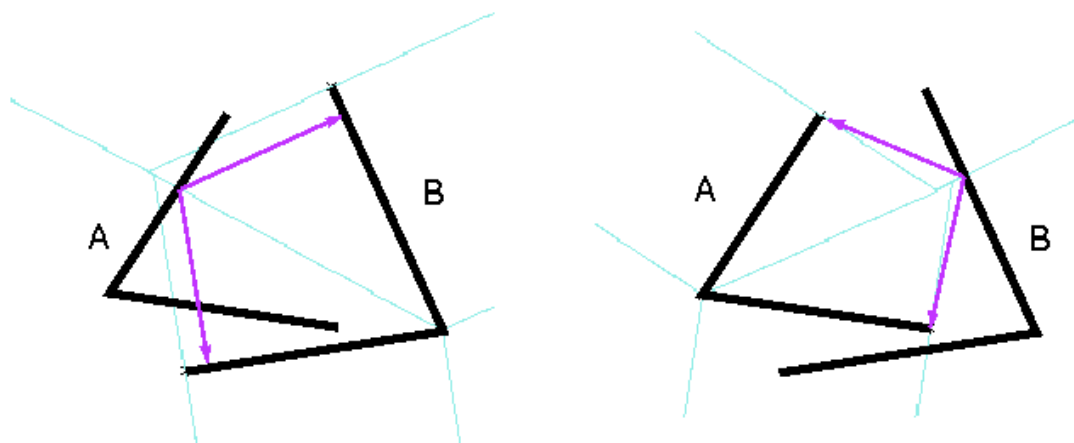
Obrázek 54 Otevření (převzato z [16])

#### 4.4.15 Hausdorffova vzdálenost mezi liniemi

Tato aplikace je uvedena ve zdroji [16]. Hausdorffova vzdálenost mezi dvěma liniemi A a B je větší ze dvou měření:

- $dH_{a>b}$ : největší ze všech nejmenších vzdáleností z bodů od A do B
- $dH_{b>a}$ : největší ze všech nejmenších vzdáleností z bodů od B do A

U Voronoiova diagramu pouze konečná sada vzdáleností potřebuje vyšetřit. Pro změření  $dH_{a>b}$ , se spočítají nejmenší vzdálenosti k B z vrcholů A a průsečíků A s Voronoiovým diagramem B a vybere se největší z nich. Analogicky pro změření  $dH_{b>a}$  viz obr. 55.



**Obrázek 55** Dvě měření pro Hausdorffovu vzdálenost mezi dvěma liniemi, z A do B, z B do A (převzato z [16])

#### 4.4.16 Hausdorffova vzdálenost mezi oblastmi

Tato aplikace je uvedena ve zdroji [16]. Hausdorffova vzdálenost mezi dvěma oblastmi A a B je větší ze dvou měření:

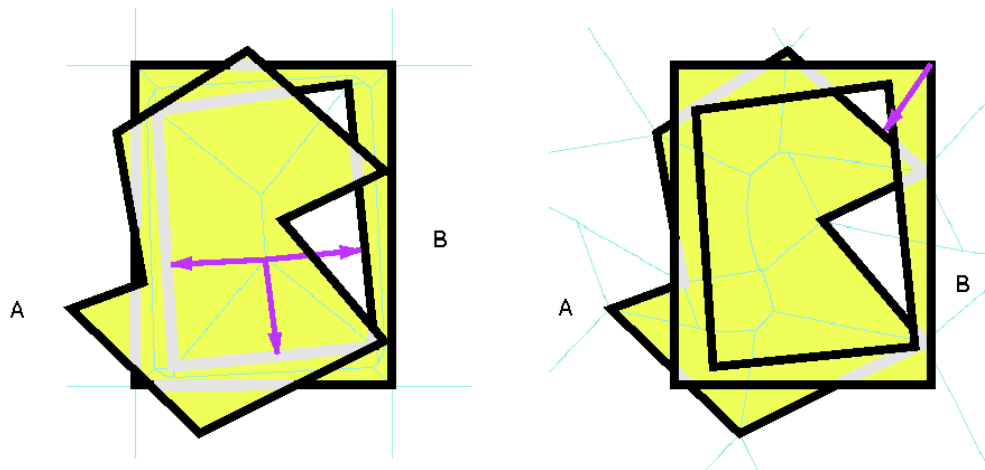
- $dH_{a>b}$ : největší ze všech nejmenších vzdáleností z bodů od A do B
- $dH_{b>a}$ : největší ze všech nejmenších vzdáleností z bodů od B do A

U Voronoiova diagramu pouze konečná sada vzdáleností potřebuje vyšetřit. Pro změření  $dH_{a>b}$ , se spočítají nejmenší vzdálenosti k B:

1. z vrcholů A

2. z průsečíků A s Voronoiovým diagramem B
3. z uzlů vnějšího Voronoiova diagramu B uvnitř A

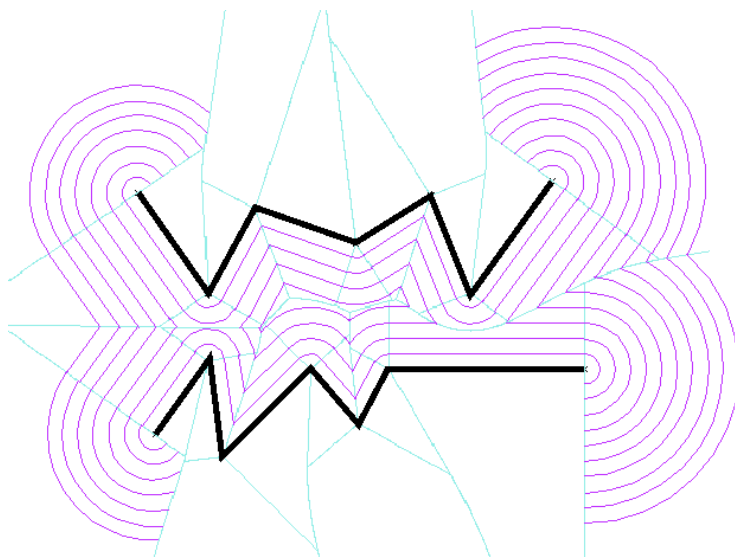
a vybere se největší z nich. Analogicky pro změření  $dH_{b>a}$  viz obr. 56.



**Obrázek 56** Dvě měření pro Hausdorffovu vzdálenost mezi dvěma oblastmi, z A do B, z B do A (převzato z [16])

#### 4.4.17 Geometrické vodní izočáry

Tato aplikace je uvedena ve zdroji [16]. V kartografii jsou vodní izočáry definovány jako ornamentální reprezentace moře nebo velkých řek na terénních mapách. Nakreslit vodní izočáry ve Voronoiově diagramu znamená pouze nakreslit roztažení půdy do vody se zvětšujícím se poloměrem viz obr. 57.



**Obrázek 57** Geometrické vodní izočáry mezi dvěma břehy (převzato z [16])

## 4.5 Přínos Voronoiových diagramů

O situaci dříve, vznikajících požadavcích v GIS a jejich řešení pomocí Voronoiových diagramů se píše ve zdrojích [14], [17] a [18].

### 4.5.1 Situace v minulosti

Před lety se řešily problémy s manipulací s prostorovými daty. Výsledkem byl vývoj efektivnějších algoritmů pro lokaci a řazení prostorových segmentů a také došlo k vývoji významně rychlejších a hospodárnějších počítačů. Metody však měly obvykle dvě slabosti:

- vyvinuly se vždy až ad hoc a řešily jeden úzký problém
- nedokázaly rozpoznat jakoukoli strukturu dat před načtením celé datové struktury pro celý datový set

Nebylo tedy možné přímé editování procesu při stavbě mapy, jednoduchý lokální update a mít možnost detekovat polygony ostrovů. Neexistoval model prostorového sousedství, který by byl vhodný i pro izolované mapové objekty.

Dále nebylo možné detekovat průsečíky, protože to byla drahá operace náchylná k chybám díky omezením digitalizace. Důsledkem bylo, že pokud nějaký průsečík není nalezen, není tedy kompletní topologie. A protože blízkost byla definována přes průsečíky, špatně se pracovalo s izolovanými objekty např. ostrovy.

Nebyla tedy možnost lokálně modifikovat topologii a neexistoval žádný dynamický systém dovolující přidání nebo vymazání individuálního mapového objektu. Vzrostla nespokojenost kvůli těmto omezením, protože i menší modifikace v mapě vyžadovala globální přestavbu mapy. I když bylo možné použít různé techniky na modifikaci malých záplat a pak je „přišít“ zpět k hlavní mapě, nastaly ale problémy se synchronizací a lokalizací.

### 4.5.2 Vznikající požadavky na datové struktury prostorových dat

Je požadováno, aby provedení akcí bylo okamžité jak v prostorových dotazech, tak prostorových modifikacích mapy.

Jde o to, aby se mohly detekovat sousední objekty kdykoli při konstrukci mapy. To umožní testování průsečíků liniových segmentů a snapování bodů na liniové segmenty.

Dále je třeba detekovat polygony nebo jiné struktury, které se neuzavřely správně.

Nakonec je třeba lokalizovat polygony, izolované datové body nebo ostrovy bez nějakého zvláštního zpracování.

Je tedy požadavek na prostorový model umožňující dlaždicování a vlastnost sousednosti rastrového modelu, ale také s přímým vztahem k objektům reálného světa ve vektorovém modelu.

### **4.5.3 Výhody Voronoiových diagramů**

Voronoiovy diagramy jsou polygony, které lze vizualizovat jako „bubliny“ rostoucí rovnoměrně kolem každého mapového objektu, dokud není zaplněn prostor celé mapy. Tak každý bod spadá do Voronoiovy oblasti nejbližšího mapového objektu.

Objekty nemusí být jen body, mohou to být jakékoli objekty, např. body a liniové segmenty.

Voronoiova oblast pro jakýkoli objekt je definována stejným způsobem jako pro body a může být ihned spočítána. Voronoiova hranice mezi kterýmikoli dvěma mapovými objekty je stejně vzdálená od obou objektů. Mezi bodem a linií hranici tvoří parabola, mezi páry bodů a páry linií jsou hranicemi linie.

Použití Voronoiových diagramů dovoluje všeobecnou konverzi geometrické informace do graficky strukturované formy.

Protože Voronoiovy prostorové dotazy nezávisí na liniových průsečících, aby se zjistila prostorová sousednost, prostorové operace se mohou provádět na částečně kompletních datových setech, stejně jako bodových datech. To, kombinované s dynamickou povahou modifikací prostorových datových struktur, umožňuje kompilaci relativně jednoduchých dat a lokální prostorové dotazy.

Objekty jsou použity k vytvoření bezešvé datové struktury, pružná datová struktura je dynamická a dovoluje ihned vložit nebo smazat individuální mapový objekt bez porušení prostorových spojení a co je nejdůležitější, jsou jiné způsoby myšlení o prostorových datových strukturách než u rastrového nebo bodového/liniového/polygonového modelu jak tomu bylo v uplynulých 15 letech.



Voronoiův diagram je univerzální datová struktura pro reprezentaci blízkosti, která ztělesňuje formu automatické topologie. Podporuje dotazy na nejbližšího souseda, poskytuje základní vlastnosti prostorové blízkosti mezi mapovými objekty. Tím se vyřešily základní potíže s modelem liniových průsečíků a zároveň dovoluje udržovat topologii GIS.

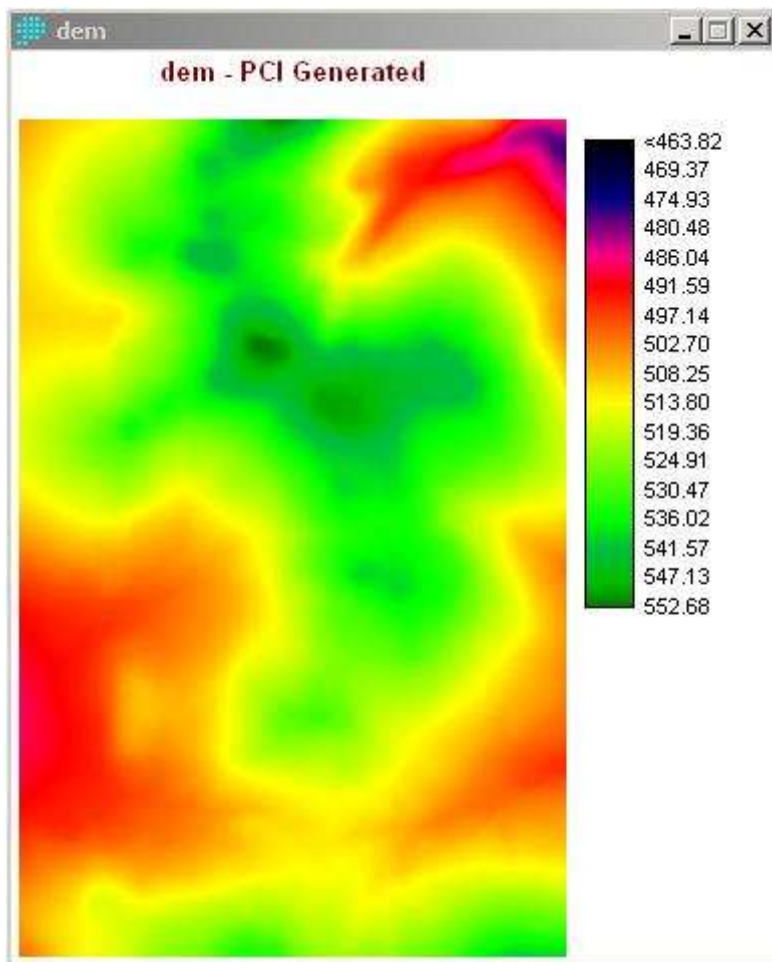
Dříve byl uveden přehled statických, kinematických a dynamických Voronoiových diagramů jako základního nástroje pro GIS. Byly prezentovány metody, které dovolují vložení, smazání a translaci bodů a liniových segmentů ve Voronoiově diagramu pro  $n$  bodů. Všechny elementární operace jsou dostupné v  $O(n \log n)$  čase. Také se velice zjednodušily některé základní GIS dotazy a jsou možné nové typy dotazů vyšší úrovně. Koncept stále lokálně modifikovatelné datové struktury, která je vždy kompletní, poskytuje důležitý nový přístup k práci s prostorovými daty, který není dostupný v existujících systémech.

## 5 Použití vybrané analytické funkce s konkrétními daty ve zvoleném prostředí GIS

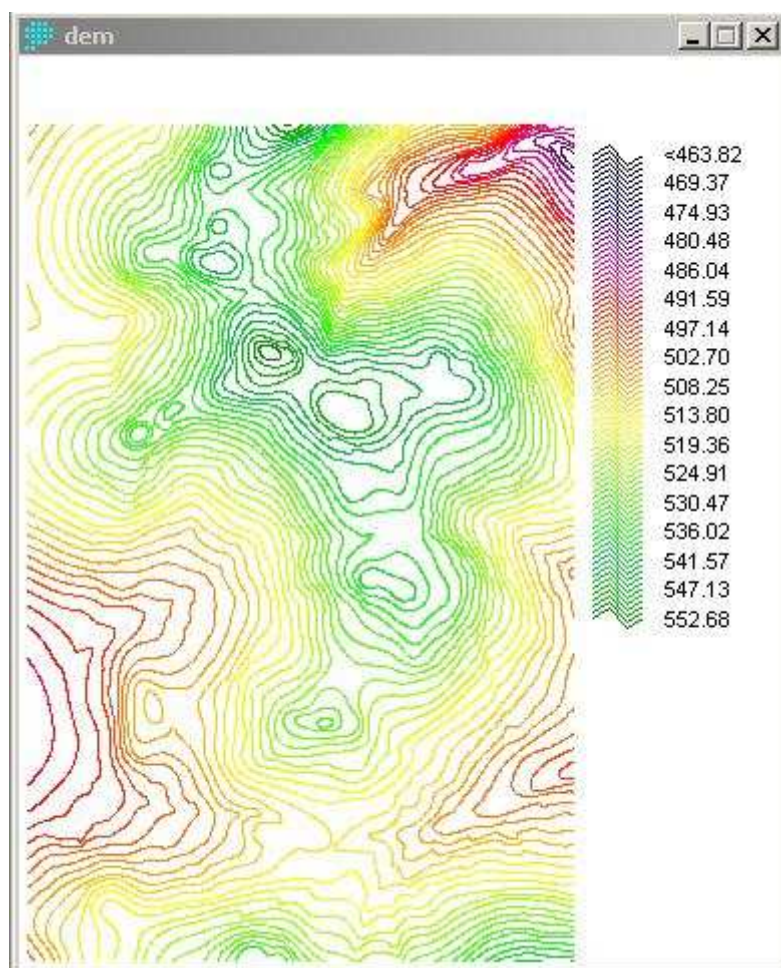
### 5.1 Výběr vhodných testovacích dat

#### 5.1.1 Reálná data

Pro předvedení některých z výše uvedených algoritmů mi byla na Akademii věd České republiky, oddělení GIS a DPZ, poskytnuta data z oblasti Dehtáře, což je oblast o velikosti 1500 x 2300 m<sup>2</sup>. Jde o soubor rastrového digitálního modelu terénu z DPZ ze systému EASI/PACE od firmy PCI, který byl převeden do vektorové podoby digitálního modelu terénu tzn. vrstevnic, použitelného v systému GIS zvaném IDRISI32 v.132.21.



Obrázek 58 Rastrový DEM ze systému EASI/PACE pro DPZ



**Obrázek 59** Vektorový DEM pro systém IDRISI

## 5.2 Výběr programového vybavení

### 5.2.1 Zpracování dat

Pro zvolené algoritmy tj. tvorba TIN z vrstevnic a zpracování topologických úloh bylo vybráno prostředí GIS zvaného IDRISI32 v.I32.21, který mi byl ke studijním účelům rovněž zapůjčen na Akademii věd České republiky, oddělení GIS a DPZ. Tento systém má v sobě řadu funkcionalit a volitelných možností i v rámci tvorby TIN, které budou následně předvedeny.

### 5.2.2 Vizualizace vypočítaných dat

K vizualizaci dat je použit volně šiřitelný program ArcExplorer 2.0 a dále GIS IDRISI32 v.I32.21.

### 5.3 Příprava dat k triangulaci

Příprava dat před triangulací je popsána ve zdrojích [20], [21] a [22]. Při přípravě dat k triangulaci je třeba rozlišit, zda se zpracovávají body nebo linie, rastry nebo vektory. Podle toho se pak zvolí některá z dalších úprav:

- PNTGEN – ztenčuje nebo generalizuje bodová vektorová data kvůli zrychlení zpracování dat
- INTERPOL – interpoluje rastrový povrch z bodových dat, pokud požadujeme konečný výstup jako rastrový povrch
- LINTOPNT – extrahuje body z vektorových linií do bodového vektorového souboru před triangulací, aby se zmenšil počet bodů pro vizualizaci
- TINPREP – přidává nebo ubírá body podél vektorové linie, což je někdy třeba kvůli rozlišení TIN
- LINEGEN – generalizuje liniová vektorová data
- INTERCON – interpoluje rastrový povrch z rastrových iso-liniových (vrstevnicových) dat
- TIN – vytváří TIN z bodových nebo liniových vektorových dat
- TINSURF – vytváří rastrový povrch z TIN

**Non-constrained TIN** se používá pro bodová data. Algoritmus byl navržen kvůli rychlosti zpracování – vrcholy iso-linií jsou rozděleny do sekcí a v každé se provede triangulace, „mini-TINs“ jsou pak spojeny dohromady, při spojování je použita lokální procedura, která maximalizuje minimální úhly, aby data vyhověla kritériím Delaunayovy triangulace.

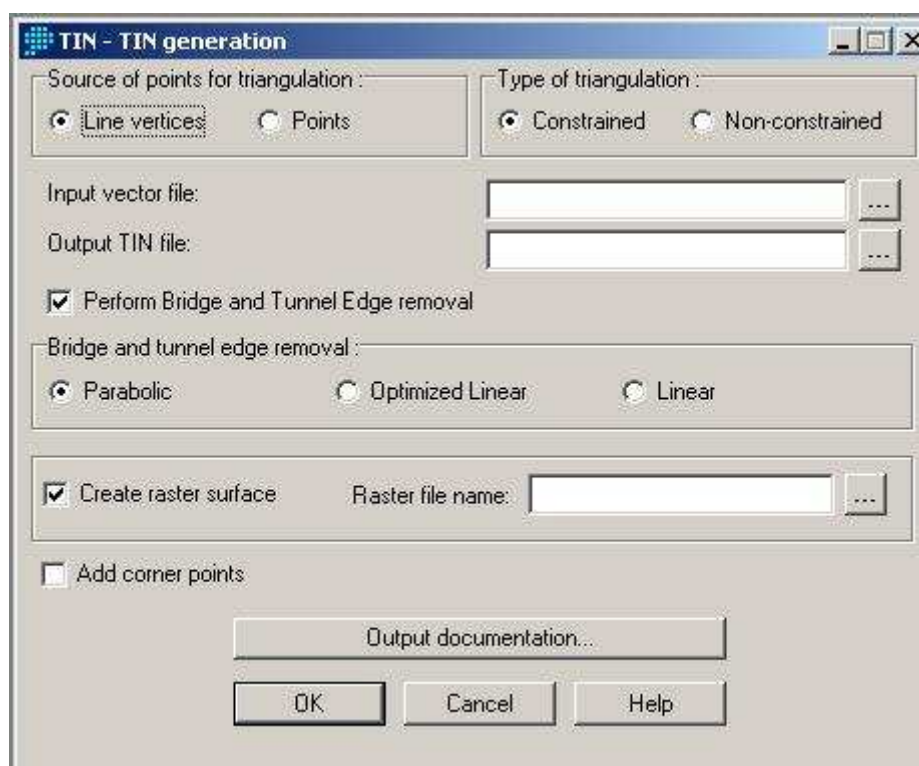
**Constrained TIN** je rozšíření non-constrained triangulace. Uplatňují se ještě další dodatečné podmínky při výběru vrcholů trojúhelníka (trojúhelník nesmí křížit isolinie). Pak ale ne všechny trojúhelníky vyhovují Delaunayovým kritériím (kružnice procházející třemi body nesmí obsahovat další bod, trojúhelníky se nepřekrývají, nejsou žádné mezery na triangulovaném povrchu). Metoda je dvoukroková – nejprve se provádí non-constrained

triangulace a pak kontrola hran, zda nekříží iso-linie, pokud ano, spustí se lokální optimalizační procedura.

**Odstraňování hran „bridge“ a „tunnel“ (B/T hrany)** – B/T hrany jsou identifikovány, když trojúhelník, jehož koncové body mají stejnou hodnotu a nejsou sousedními body na iso-linii. Tzv. kritické body se vytvoří ve středu B/T hrany a tyto oblasti jsou znovu triangulovány.

## 5.4 Možné kombinace pro vytvoření TIN

Funkce TIN nabízí kombinace podle obrázku:



**Obrázek 60** Generování TIN

### 5.4.1 Přímou z vrstevnic

Na liniový vektorový soubor byla použita funkce TIN, která nabízí tyto kombinace:

- Linie, constrained TIN, parabolické odstranění B/T – viz Příloha 1
- Linie, constrained TIN, optimalizované lineární odstranění B/T – viz Příloha 2
- Linie, constrained TIN, lineární odstranění B/T – viz Příloha 3

- Linie, constrained TIN, bez odstranění B/T – ve zdroji [22] toto doporučují nepoužívat
- Linie, non-constrained TIN – viz Příloha 4

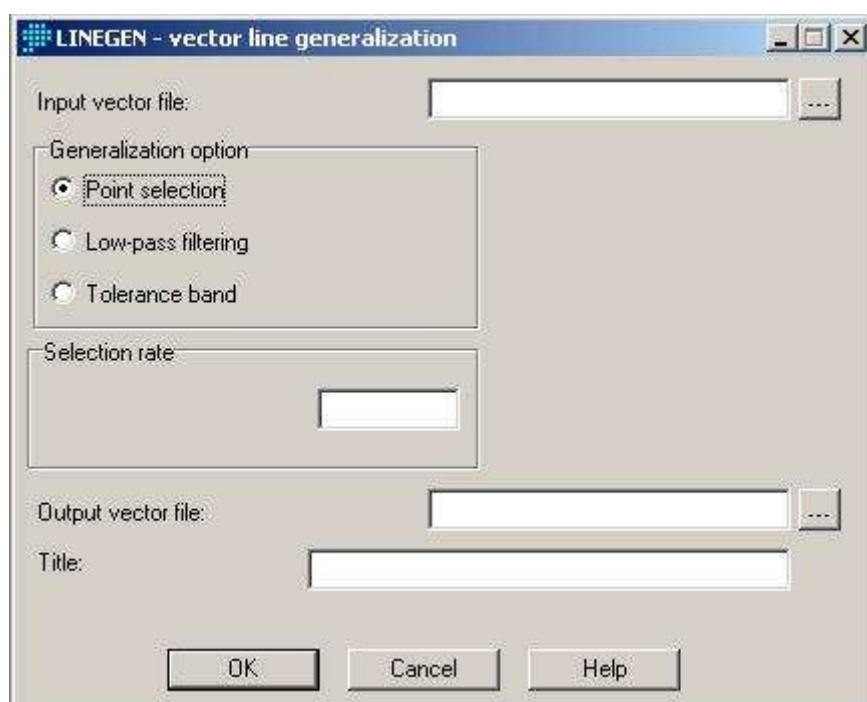
Vždy byly použity hodnoty v rohových bodech a zároveň se generoval rastrový digitální model terénu.

#### 5.4.2 Z bodového vektorového souboru

Z digitálního modelu vrstevnic se přes funkci LINTOPNT vygeneruje vektorový bodový soubor. Následuje generalizace bodů přes funkci PNTGEN a pak se provede non-constrained triangulace. Byly použity hodnoty v rohových bodech a zároveň se generoval rastrový digitální model terénu – viz Příloha 5

#### 5.4.3 Z liniového vektorového souboru

Z digitálního modelu vrstevnic se nejprve generalizují linie přes funkci LINEGEN, která umožňuje čtyři kombinace podle obrázku:



**Obrázek 61** Generalizace linií

Na každou z metod generalizace se pak použije generování TIN přes funkci TIN se svými pěti možnostmi.

- Generalizace s tolerančním pásmem, která poskytuje vynikající snížení kontextově citlivých dat, i když výsledek není tak hladký – viz Příloha 6, Příloha 7, Příloha 8, Příloha 9, Příloha 10
- Generalizace s filtrem „low-pass“ s průměrem přes tři body, která vynikajícím způsobem vyhladí hrany, ale nezredukuje data – viz Příloha 11, Příloha 12, Příloha 13, Příloha 14, Příloha 15
- Generalizace s filtrem „low-pass“ s průměrem přes pět bodů, která také vynikajícím způsobem vyhladí hrany, ale nezredukuje data – viz Příloha 16, Příloha 17, Příloha 18, Příloha 19, Příloha 20
- Generalizace s výběrem bodu, která je spíše doplněna z pedagogických důvodů, zřídka je výsledkem dobrá generalizace – viz Příloha 21, Příloha 22, Příloha 23, Příloha 24, Příloha 25

## 5.5 Porovnání s referenčním souborem

Ve zdrojích [20], [21] a [22] je jako nejlepší uvedena metoda TIN, která pracuje s liniemi, tvoří constrained TIN a používá parabolické odstranění B/T hran. Proto byl jako referenční soubor vybrán rastrový digitální model vytvořený touto metodou viz Příloha 1.

Od tohoto referenčního souboru se pak odečetly přes funkci OVERLAY hodnoty jednotlivých rastrových modelů, vznikly soubory s odchylkami proti referenčnímu souboru a výsledkem z odchylek jsou histogramy. Tyto soubory s odchylkami a histogramy jsou v jednotlivých přílohách Příloha 2, Příloha 3, ..., Příloha 25.

## 5.6 Zhodnocení získaných výsledků

Při porovnání všech histogramů je podle střední hodnoty odchylek vidět, že u všech generalizovaných linií, na které se potom aplikuje metoda TIN, nejlépe vyhovují triangulace z linií, metodou constrained TIN a s parabolickým odstraňováním B/T hran.

Při generování přímo z vrstevnic měla nejlepší výsledky odchylek triangulace z linií, metodou constrained TIN a optimalizovaným lineárním odstraňováním B/T hran.

Při generování z bodového vektorového souboru nejsou výsledky příliš dobré, na což ale bylo ve zdrojích [20], [21] a [22] upozorňováno.

GIS IDRISI32 v.132.21 nabízí širokou škálu analytických funkcí, které jsou v něm již implementovány. Toto byla jen jedna vybraná funkce, která umožňovala 25 různých kombinací nastavení pro porovnání. Podobně lze pak porovnat i ostatní funkce. Systém je neustále rozvíjen a obohacován o další funkcionality. Práce s ním je velice intuitivní a uživatelsky příjemná. Rozhodně tento systém vyhovuje dnešním nárokům na GIS a je rovněž velkou výhodou, že je velice názorný a proto vhodný pro pedagogické, výzkumné a další účely.



## 6 Závěr

Největší přínos této práce lze spatřovat:

1. V podrobném přehledu datových modelů a typů pro GIS, který byl zpracován na základě řady hlavně cizojazyčných pramenů a to z hlediska posledního vývoje v tomto oboru GIS
2. Ve zpracování podrobného a co nejúplnějšího přehledu analytických funkcí jako matematických nástrojů z hlediska GIS a to rovněž na základě prostudování mnoha cizojazyčných materiálů s cílem poskytnout co nejaktuálnější informace o názorech na dělení analytických funkcí použitých v GIS
3. Ve vypracování detailního popisu teorie o Voronoiových diagramech, podrobného přehledu algoritmů pro uplatnění teorie Voronoiových diagramů a v neposlední řadě co nejširšího přehledu aplikací Voronoiových diagramů v GIS. To vše bylo sestaveno pouze ze zahraničních pramenů, protože se v současné době jedná o velice dynamicky se vyvíjející oblast, o níž je třeba mít přehled
4. V široké analýze jedné z početných funkcí v GIS IDRISI32 v.I32.21a to triangulace a z ní vytvoření rastrového digitálního modelu terénu a předvedení vhodnosti jednotlivých nastavení, což odpovídá doporučením uvedeným v popisu tohoto GIS. I když tato část není tak důkladně rozebrána, domnívám se, že i jen popis jediné funkce a to takto podrobně přesto naznačuje široké možnosti využití i ostatních funkcí GIS a GIS obecně

Celkově se domnívám, že práce může být vhodná jako studijní materiál pro obory geomatika, geografické informační systémy, informatika, geografie, počítačová grafika, robotika, ... Snažila jsem se zodpovědně přistoupit k plnění zadání diplomové práce, což se mi podařilo nejvíce v prvních třech bodech zadání. Ale i poslednímu bodu jsem se i přes časovou tíseň snažila věnovat důkladně. Překážkou snad bylo hledání vhodného SW a konkrétních dat k použití pro tuto diplomovou práci, což jsem mnohokrát konzultovala jak s vedoucím diplomové práce Doc. RNDr. Františkem Ježkem, CSc. , Doc. Dr. Ing. Ivanou Kolingerovou a vedoucím laboratoře GIS při Akademii věd České republiky Ing. Mgr. Františkem Zemkem.

## Seznam použité literatury

- (1) Tuček Ján, Geografické informační systémy, Principy a praxe, Computer Press, 1998, 424 s.
- (2) Břehovský Martin, Jedlička Karel, Úvod do geografických informačních systémů, přednáškové texty, ZČU, 2002, 116s.
- (3) Chrisman Nicholas, Exploring Geographic Information Systems, John Wiley & Sons, Inc., 1996, 298s.
- (4) Albrecht Jochen, Geographic Information Research: transatlantic perspectives, Universal analytical GIS operation – a task-oriented systematization of data structure-independent GIS functionality, pp. 577-591
- (5) Kolář Jan, Halounová Lena, Pavelka Karel, Dálkový průzkum Země 10, Vydavatelství ČVUT, 1997, 164s.
- (6) Borrough P. A., Principles of geographical information systems for land resources assessment, Butler and Tanner Ltd, 1986, 190s.
- (7) Záborský Jaromír, Diplomová práce Triangulace povrchů a úlohy na nich, ZČU, 2005, 102s.
- (8) Kolingerová Ivana, Štrych Václav, Čada Václav, Using Constraints in Delaunay and Greedy Triangulation for Contour Lines Improvement. University of West Bohemia, Czech Republic, Plzeň, 2004, 8s.
- (9) Borrough P. A., McDonnell R. A., Principles of Geographical Information Systems, Oxford University Press, 1998, ISBN 0-1-823365-5
- (10) Aurenhammer Franz, Voronoi Diagrams, A Survey of a Fundamentals Geometric Data Structure, ACM Computing Surveys, Vol. 23, No 3, September 1991

- (11) de Berg Mark, van Kreveld Marc, Overmars Mark, Schwarzkopf Otfried, Computational Geometry, Algorithms and Applications, Springer-Verlag Berlin Heidelberg, 1997, ISBN 3-540-61270-X.
- (12) Kolingerová Ivana, přednášková prezentace k předmětu Aplikovaná výpočetní geometrie, 2005
- (13) O'Rourke Joseph, Computational Geometry in C, Cambridge University Press, 1998, ISBN 0 521 64010 5
- (14) van Kreveld Marc, Nievergelt Jürg, Roos Thomas, Widmayer Peter, Algorithmic Foundations of Geographic Informations Systems, Springer-Verlag Berlin Heidelberg, 1997, ISBN 3-540-63818-0
- (15) Kolingerová Ivana, materiál poskytnutý Doc. Dr. Ing. Kolingerovou Ivanou při ústní konzultaci
- (16) <http://www.geocomputation.org/2000/GC005/Gc005.htm>
- (17) Gold M. Christopher, Neighbours, Adjacency and Theft – the Voronoi Process for Spatial analysis, European Conference on Geographical Information Systems, materiál poskytnutý Doc. Dr. Ing. Kolingerovou Ivanou při ústní konzultaci
- (18) Gold M. Christopher, Spatial Adjacency as the Basis for Spatial Data Structures, Workshop on Voronoi Methods for Handling Geographic Data, materiál poskytnutý Doc. Dr. Ing. Kolingerovou Ivanou při ústní konzultaci
- (19) Healey Richard, Dowers Steve, Gittings Bruce, Minete Mike, Parallel Processing Algorithms for GIS, 1998, ISBN 0-7484-0508-9
- (20) Eastman J. Ronald, IDRISI32, Tutorial, Clark Labs, 1999

- (21) Eastman J. Ronald, IDRISI32, Guide to GIS and Image Processing, Volume 1, Clark Labs, 1999
- (22) Eastman J. Ronald, IDRISI32, Guide to GIS and Image Processing, Volume 2, Clark Labs, 1999

## Seznam obrázků

<b>Obrázek 1</b> Základní geografická primitiva (body, linie, polygony) (převzato z [9]) .....	11
<b>Obrázek 2</b> Kódování objektů (entit) a souvislých polí v různých datových modelech. a) vektorová reprezentace polygonu; b) rastrový model souvislého pole; c) vektorová reprezentace spojených linií; d) Delaunayova triangulace souvislého pole (převzato z [9])...	13
<b>Obrázek 3</b> Hierarchická a síťová organizace vektorových dat (převzato z [9]) .....	17
<b>Obrázek 4</b> Struktury identifikačních kódů. a) jednoduchá relační struktura pro jednoduchou polygonovou mapu; b) po normalizaci (převzato z [9]).....	19
<b>Obrázek 5</b> Přidání nových dat do relačního systému je jenom záležitost definování nových tabulek (převzato z [9]) .....	20
<b>Obrázek 6</b> a) Relační tabulky pro databázi utilit včetně redundantní informace. b) Objektově-orientovaný přístup redukuje uložení dat a zdokonaluje spojení mezi záznamy (převzato z [9]) .....	21
<b>Obrázek 7</b> Datové roviny a feature vrstvy v a) rastrové a b) vektorové datové struktuře (převzato z [9]) .....	23
<b>Obrázek 8</b> Čtyři rozdílné způsoby vytvoření rastrové datové struktury a) Každá buňka je odkazována přímo; b) každé překrytí je odkazováno přímo; c) každá mapová jednotka je odkazována přímo; d) každé překrytí je odděleným souborem s obecnou hlavičkou (převzato z [9]).....	24
<b>Obrázek 9</b> Jednoduchá rastrová mapa (převzato z [9]).....	26
<b>Obrázek 10</b> MAT polygonu A v Obrázku 9 (převzato z [9]) .....	28
<b>Obrázek 11</b> Kódování quadtree polygonu A na Obrázku 9 .....	29
<b>Obrázek 12</b> Quadtree hierarchie polygonu A na Obrázku 9 (převzato z [9]).....	30
<b>Obrázek 13</b> Alternativní způsoby kódování dat na různých stupních rozlišení (převzato z [9]) .....	31
<b>Obrázek 14</b> Vektorová datová struktura jednoduché bodové entity (převzato z [9]) .....	32
<b>Obrázek 15</b> Hybridní datová struktura pro síťovou analýzu (převzato z [9]).....	33
<b>Obrázek 16</b> Bez topologie nemůže databáze rozlišit mezi polygony, které sdílejí hraniční linie (nahore) nebo které skutečně oddělují entity (dole) (převzato z [9]).....	34
<b>Obrázek 17</b> Topologické chyby v polygonové síti (převzato z [9]) .....	35
<b>Obrázek 18</b> Tři různé způsoby včleňování jednoduchých topologií v polygonové síti (převzato z [9]) .....	36
<b>Obrázek 19</b> Plná topologická struktura v polygonové síti (převzato z [9]) .....	37
<b>Obrázek 20</b> Datová struktura topologicky spojené polygonové sítě (převzato z [9]).....	38
<b>Obrázek 21</b> Nepravidelné triangulační síť založené na Delaunayově triangulaci poskytují vektorový datový model souvislého pole (převzato z [9]).....	39
<b>Obrázek 22</b> Indexování dat za použití B-stromu (převzato z [9]) .....	42
<b>Obrázek 23</b> Indexování dat za použití MOO Indexování dat za použití MOO (převzato z [9]) .....	43
<b>Obrázek 24</b> Ekvivalentní hybridní vektorová (a - nalevo) a rastrová (b - napravo) datová struktura pro modelování "crisp" polygonů (převzato z [9]) .....	46
<b>Obrázek 25</b> Voroniův diagram (převzato z [12]).....	66
<b>Obrázek 26</b> Johnson-Mehlův model (převzato z [10]) .....	68
<b>Obrázek 27</b> Apolloniův model (převzato z [10]) .....	69
<b>Obrázek 28</b> Power diagram pro sedm kružnic (převzato z [10]) .....	69
<b>Obrázek 29</b> Střední osa obdélníka (převzato z [13]) .....	70
<b>Obrázek 30</b> Střední osa konvexního polygonu o osmi vrcholech (převzato z [13]).....	70
<b>Obrázek 31</b> Obrys a jeho střední osa (převzato z [13]) .....	70

<b>Obrázek 32</b> Voronoiův diagram, když místem jsou úsečky (převzato z [12]) .....	71
<b>Obrázek 33</b> Čtyřúhelník před a po swapování (převzato z [14]) .....	71
<b>Obrázek 34</b> Voronoiův diagram během dynamické operace (převzato z [14]) .....	72
<b>Obrázek 35</b> Voronoiův diagram nejbližšího bodu ( $k=7$ ) spolu s nejmenší opsanou kružnicí (převzato z [12]) .....	72
<b>Obrázek 36</b> Voronoiův diagram řádu 2 (převzato z [10]).....	73
<b>Obrázek 37</b> Vkládání Voronoiovy oblasti (převzato z [10]) .....	74
<b>Obrázek 38</b> Spojení dvou Voronoiových diagramů (převzato z [10]).....	74
<b>Obrázek 39</b> Zpracování protínání s linií (převzato z [10]) .....	75
<b>Obrázek 40</b> Voronoiův diagram jako projekce horní obálky rovin na rovinu $z=0$ (převzato z [11]).....	76
<b>Obrázek 41</b> Aproximace generátorů sadou bodů: a) Voronoiův diagram pro 882 bodů, které aproximují sedm generátorů; b) aproximace zobecněného Voronoiova diagramu získaného smazáním nadbytečných hran a vrcholů z a) (převzato z [15]).....	76
<b>Obrázek 42</b> Největší prázdný kruh (převzato z [12]).....	78
<b>Obrázek 43</b> Nejbližší místo a nejmenší kružnice obsahující všechny body (převzato z [10]).....	78
<b>Obrázek 44</b> Strom s minimální délkou (převzato z [12]) .....	79
<b>Obrázek 45</b> Cesta vytvořená stromem s minimální délkou (převzato z [13]) .....	79
<b>Obrázek 46</b> Zkrácení dvojnásobné cesty (převzato z [13]) .....	80
<b>Obrázek 47</b> Konvexní obálka (převzato z [16]).....	80
<b>Obrázek 48</b> Kostra (převzato z [16]) .....	81
<b>Obrázek 49</b> Centroid (převzato z [16]) .....	81
<b>Obrázek 50</b> Hranice (převzato z [16]) .....	82
<b>Obrázek 51</b> Roztažení (převzato z [16]) .....	83
<b>Obrázek 52</b> Stažení (převzato z [16]) .....	83
<b>Obrázek 53</b> Uzavření (převzato z [16]) .....	84
<b>Obrázek 54</b> Otevření (převzato z [16]) .....	84
<b>Obrázek 55</b> Dvě měření pro Hausdorffovu vzdálenost mezi dvěma liniemi, z A do B, z B do A (převzato z [16]) .....	85
<b>Obrázek 56</b> Dvě měření pro Hausdorffovu vzdálenost mezi dvěma oblastmi, z A do B, z B do A (převzato z [16]) .....	86
<b>Obrázek 57</b> Geometrické vodní izočáry mezi dvěma břehy (převzato z [16]).....	86
<b>Obrázek 58</b> Rastrový DEM ze systému EASI/PACE pro DPZ.....	90
<b>Obrázek 59</b> Vektorový DEM pro systém IDRISI.....	91
<b>Obrázek 60</b> Generování TIN.....	93
<b>Obrázek 61</b> Generalizace linií.....	94

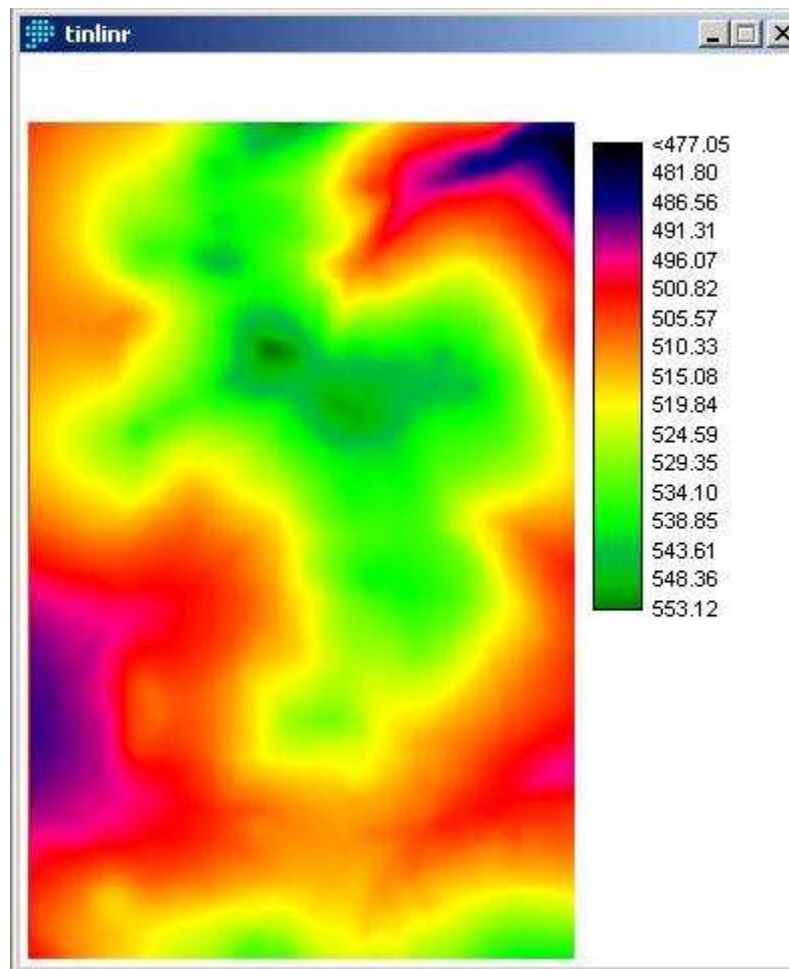
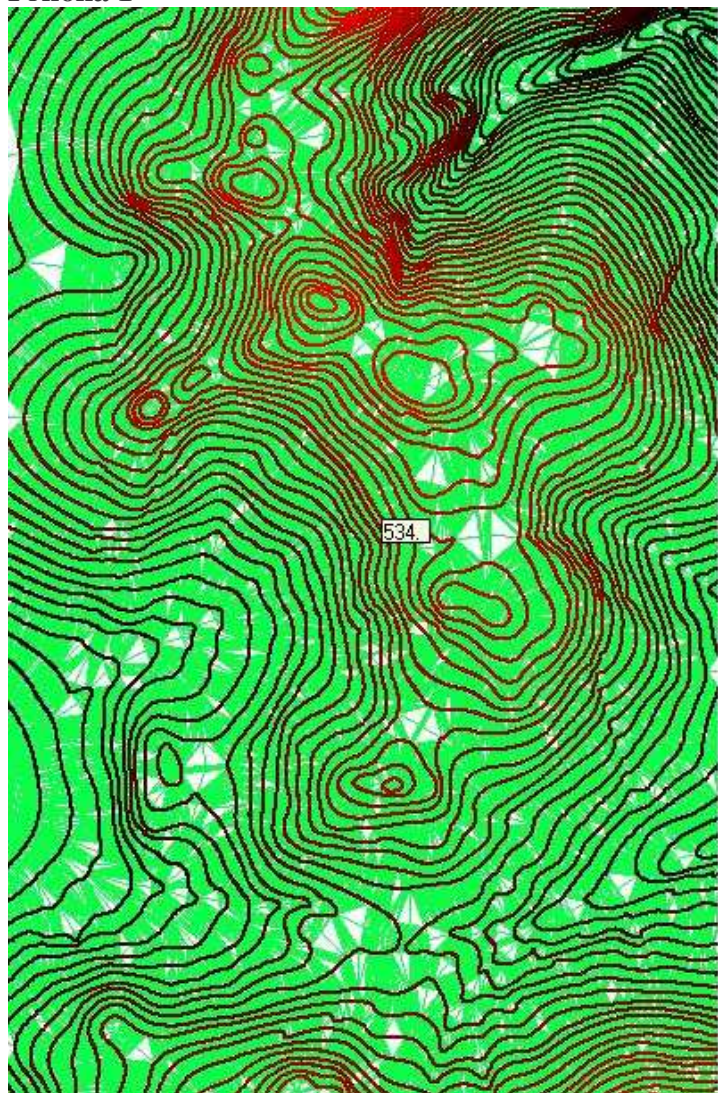
## Seznam příloh

Příloha 1 .....	105
Příloha 2 .....	106
Příloha 3 .....	108
Příloha 4 .....	110
Příloha 5 .....	112
Příloha 6 .....	114
Příloha 7 .....	116
Příloha 8 .....	118
Příloha 9 .....	120
Příloha 10 .....	122
Příloha 11 .....	124
Příloha 12 .....	126
Příloha 13 .....	128
Příloha 14 .....	130
Příloha 15 .....	132
Příloha 16 .....	135
Příloha 17 .....	137
Příloha 18 .....	139
Příloha 19 .....	141
Příloha 20 .....	143
Příloha 21 .....	146
Příloha 22 .....	148
Příloha 23 .....	150
Příloha 24 .....	152
Příloha 25 .....	154
Příloha 26 .....	156

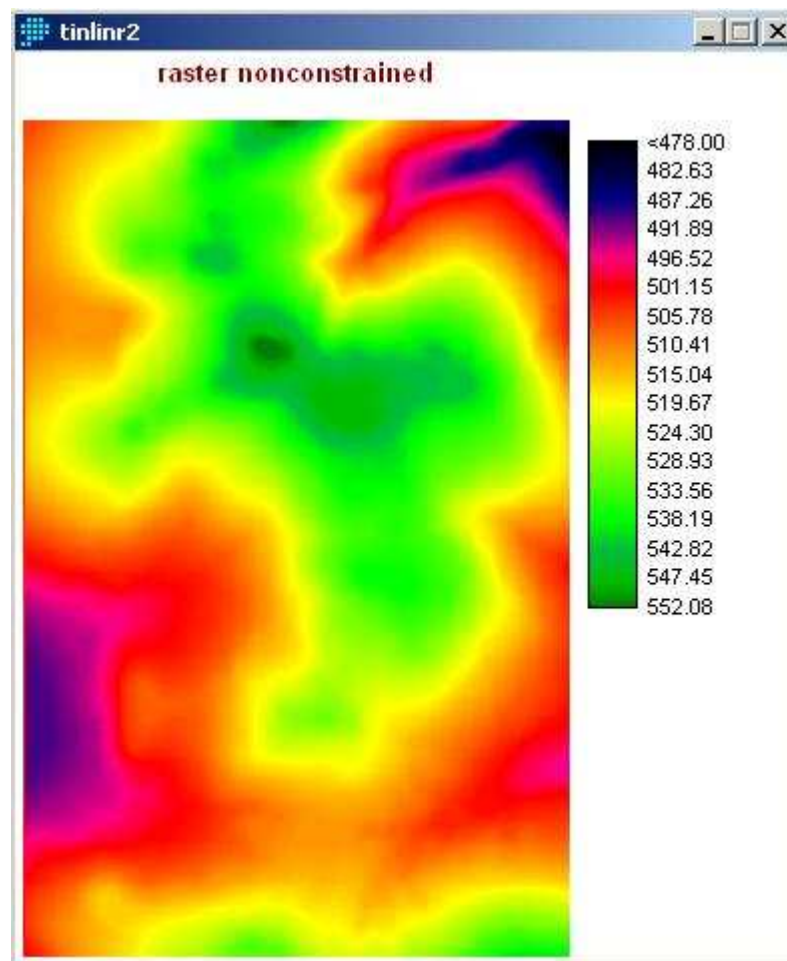
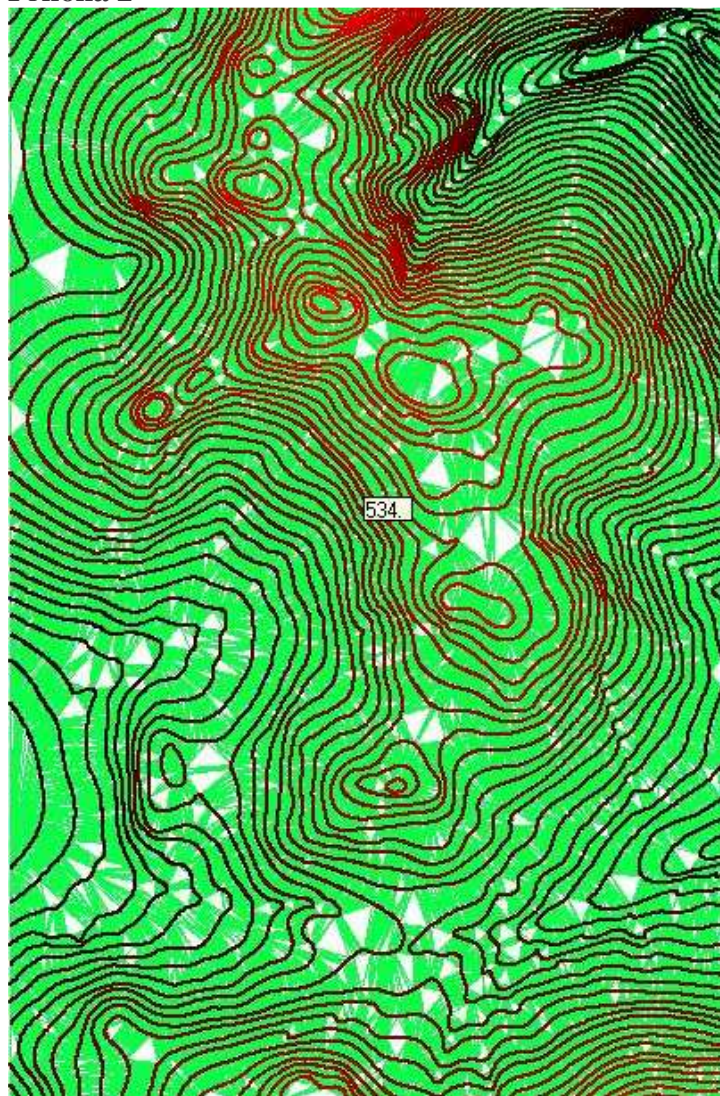
## **Přílohy**

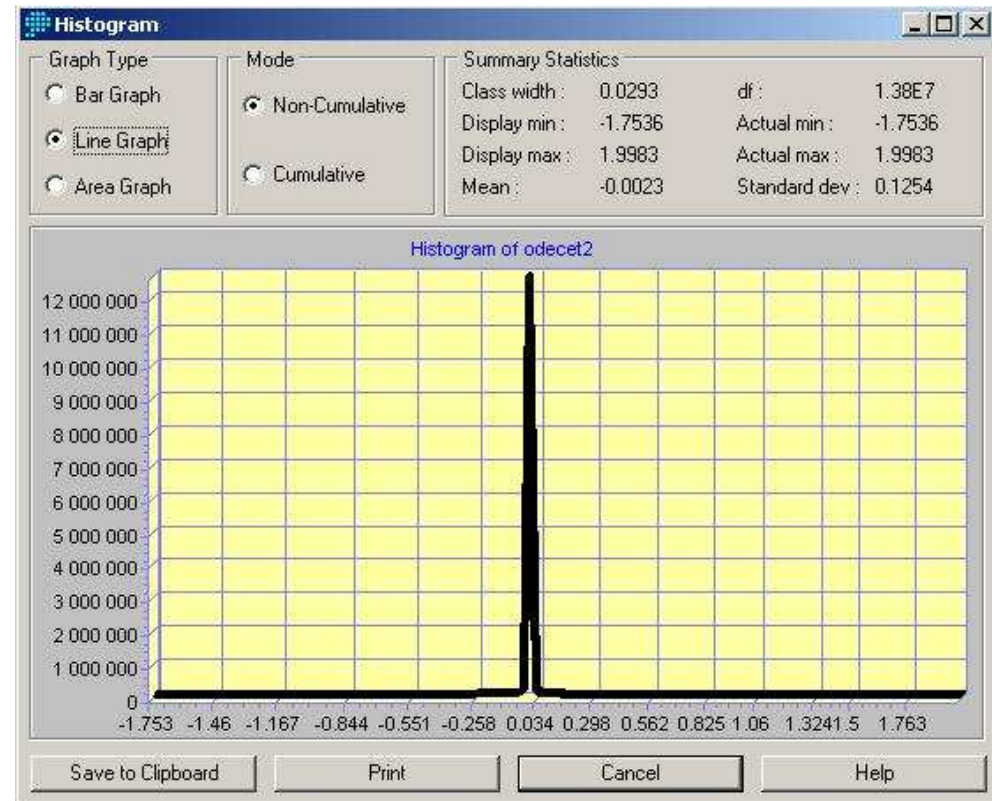
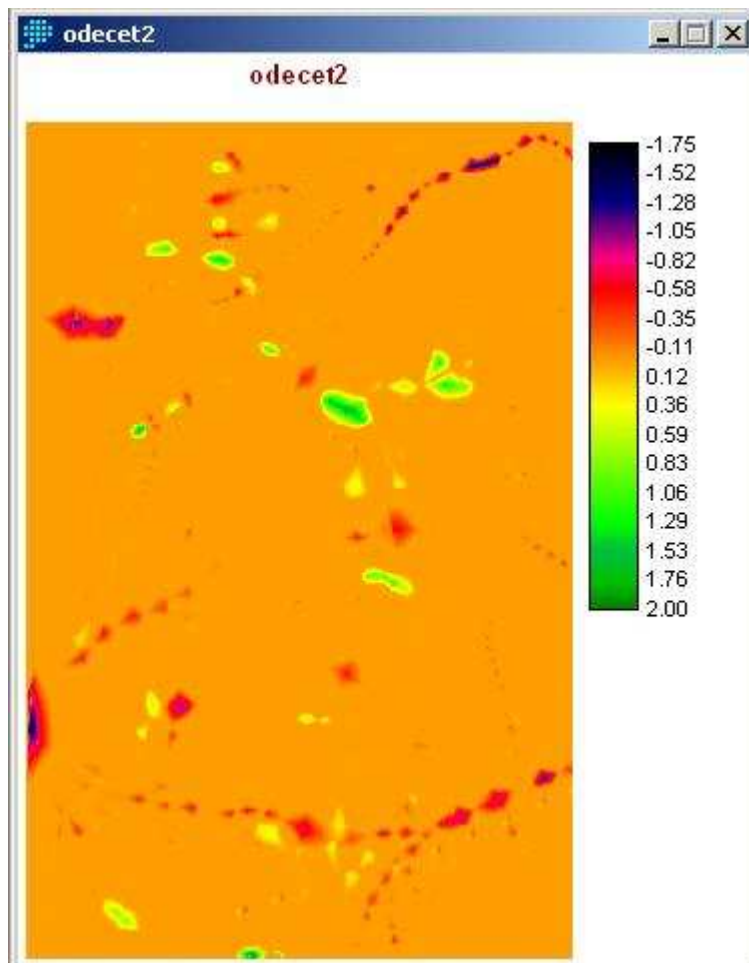


# Příloha 1

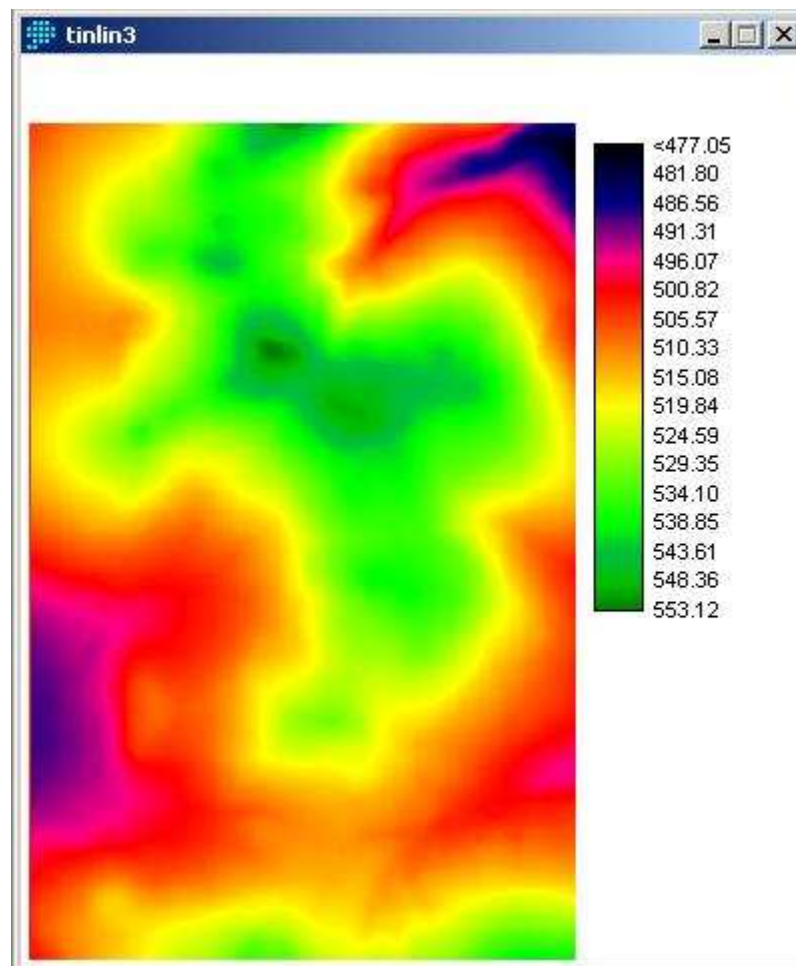
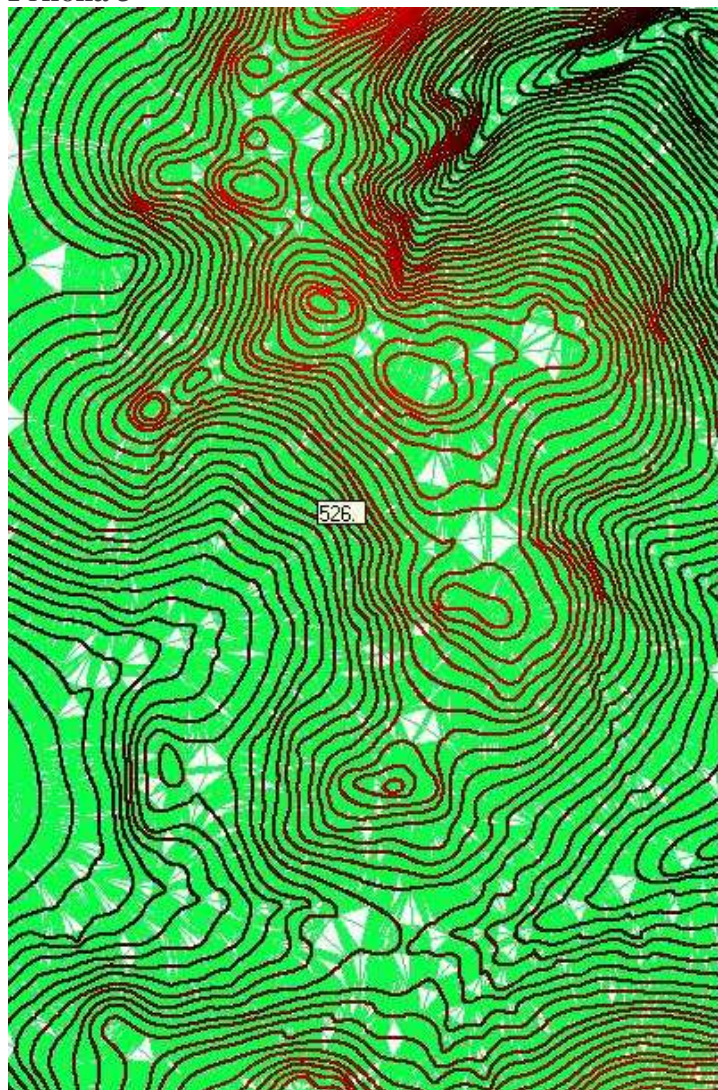


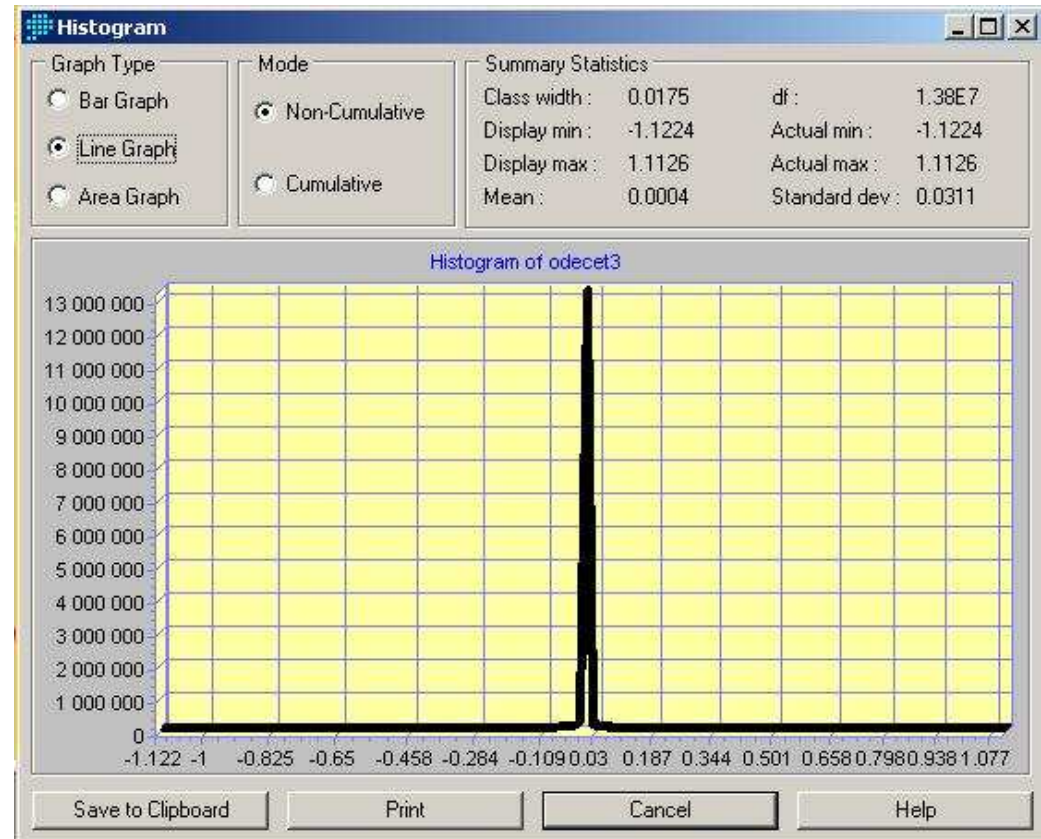
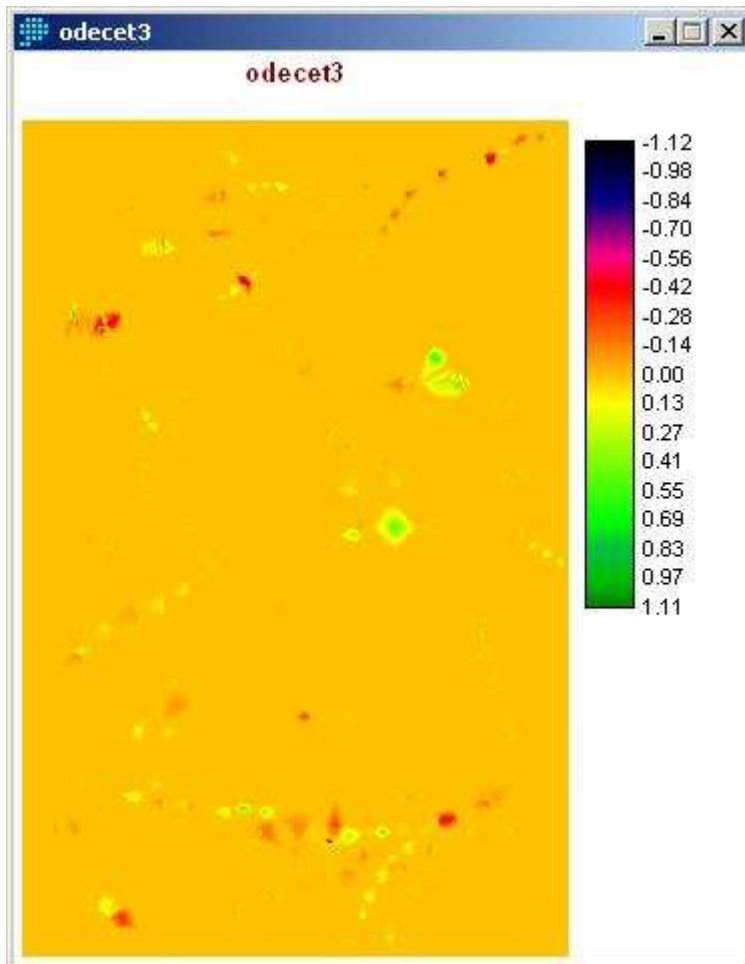
Příloha 2



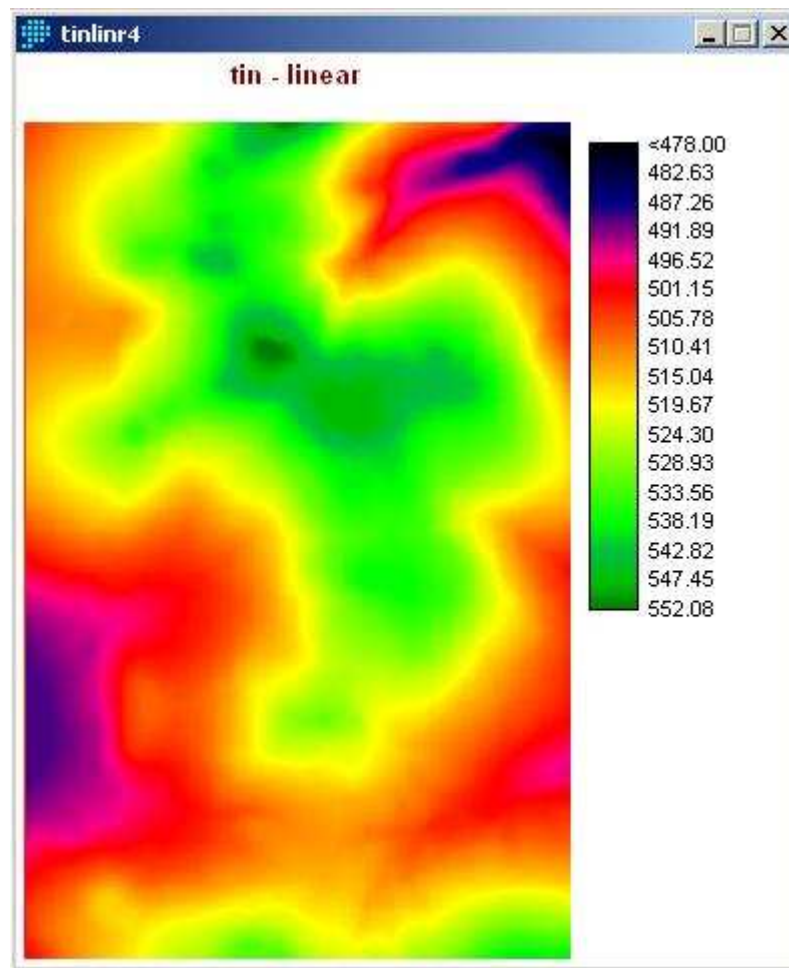
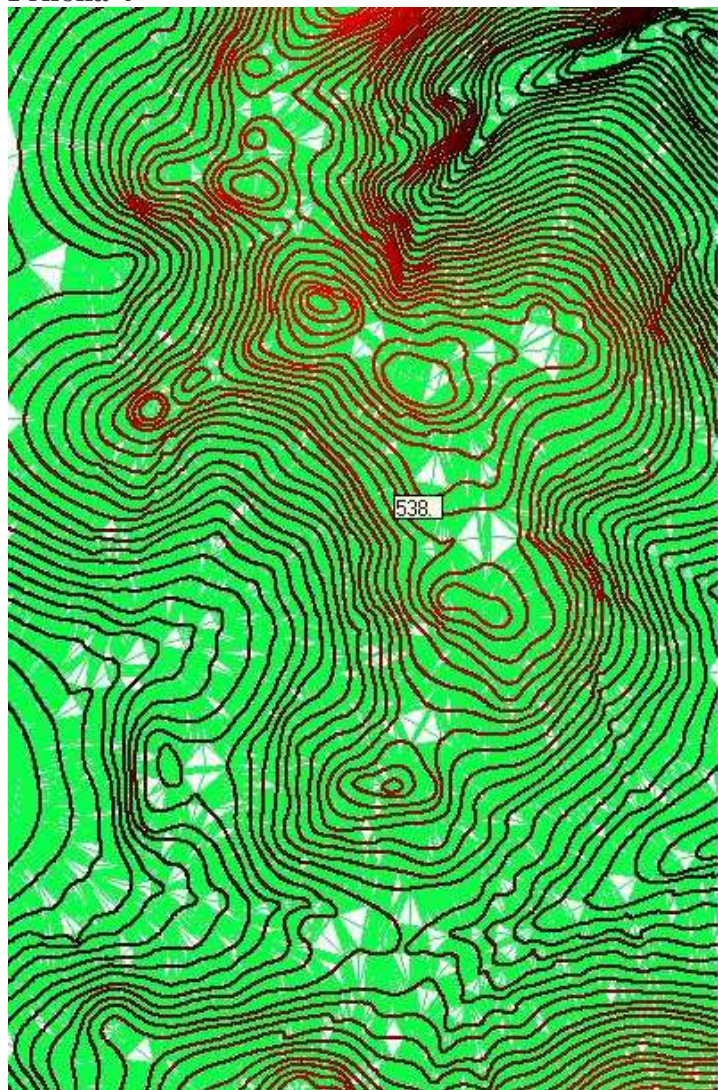


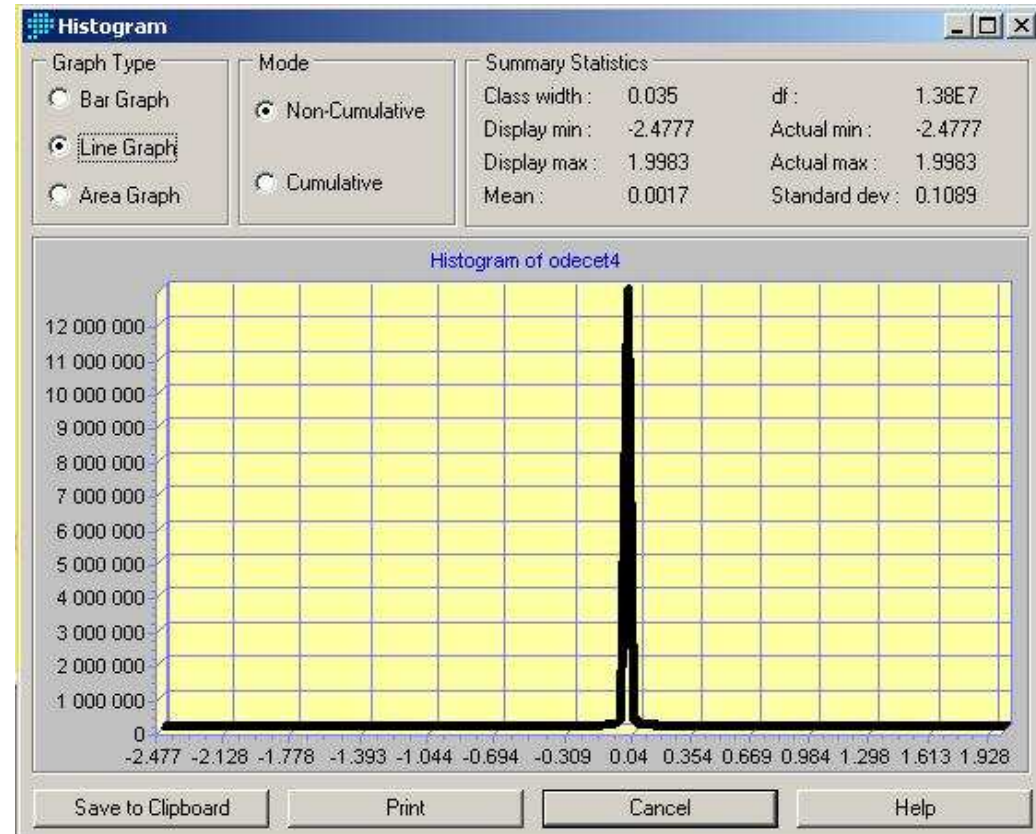
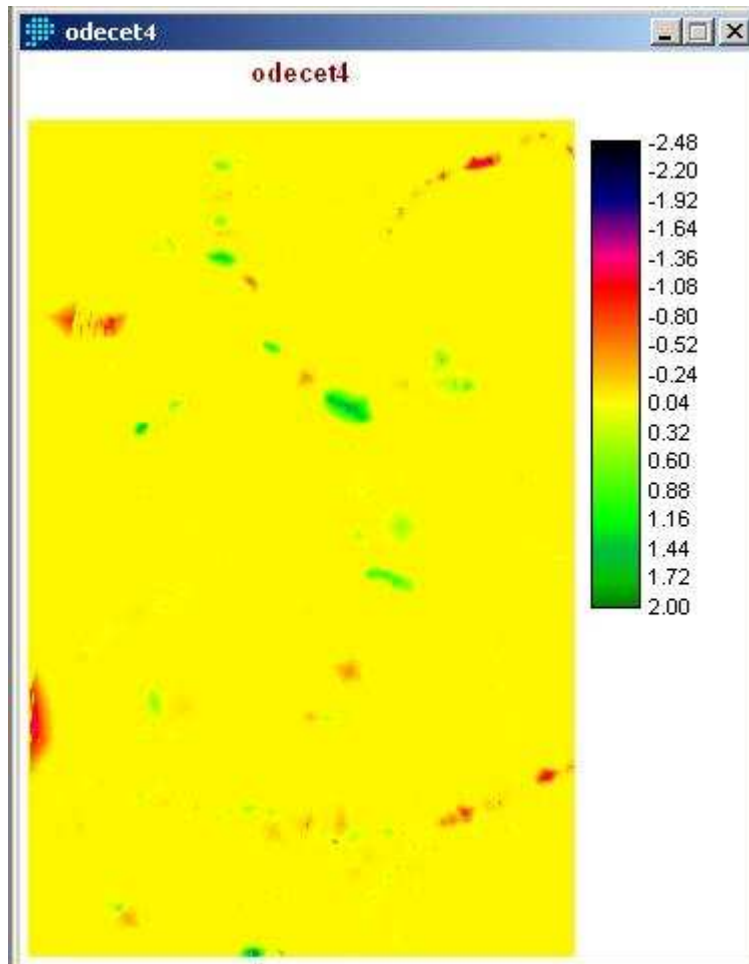
### Příloha 3



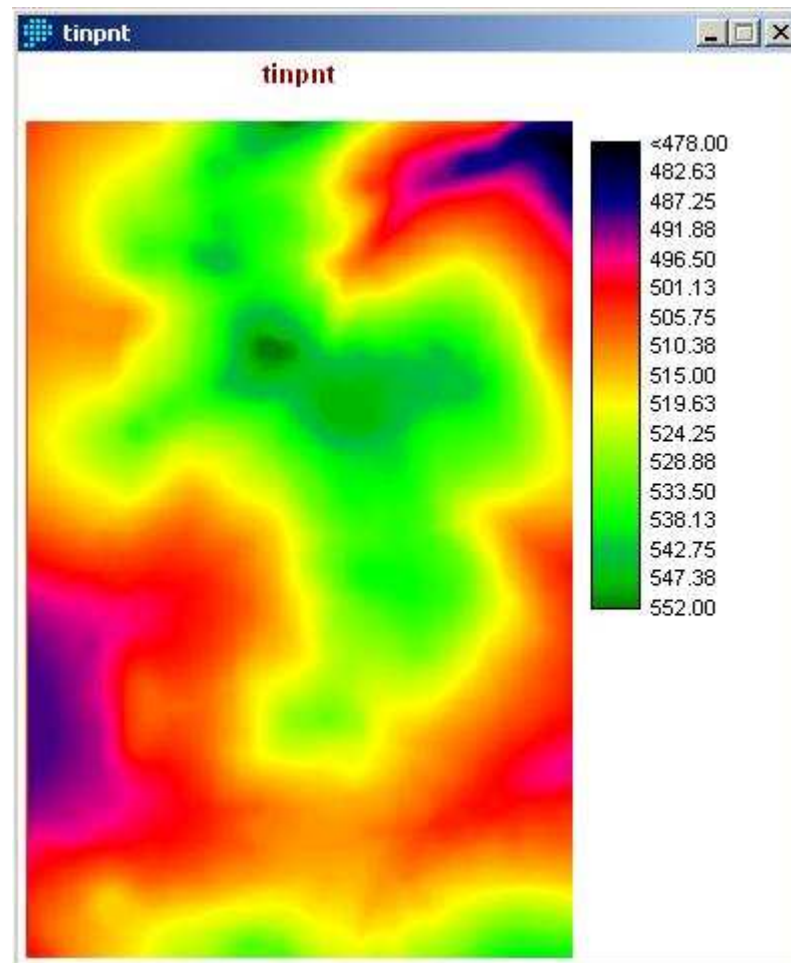


Příloha 4

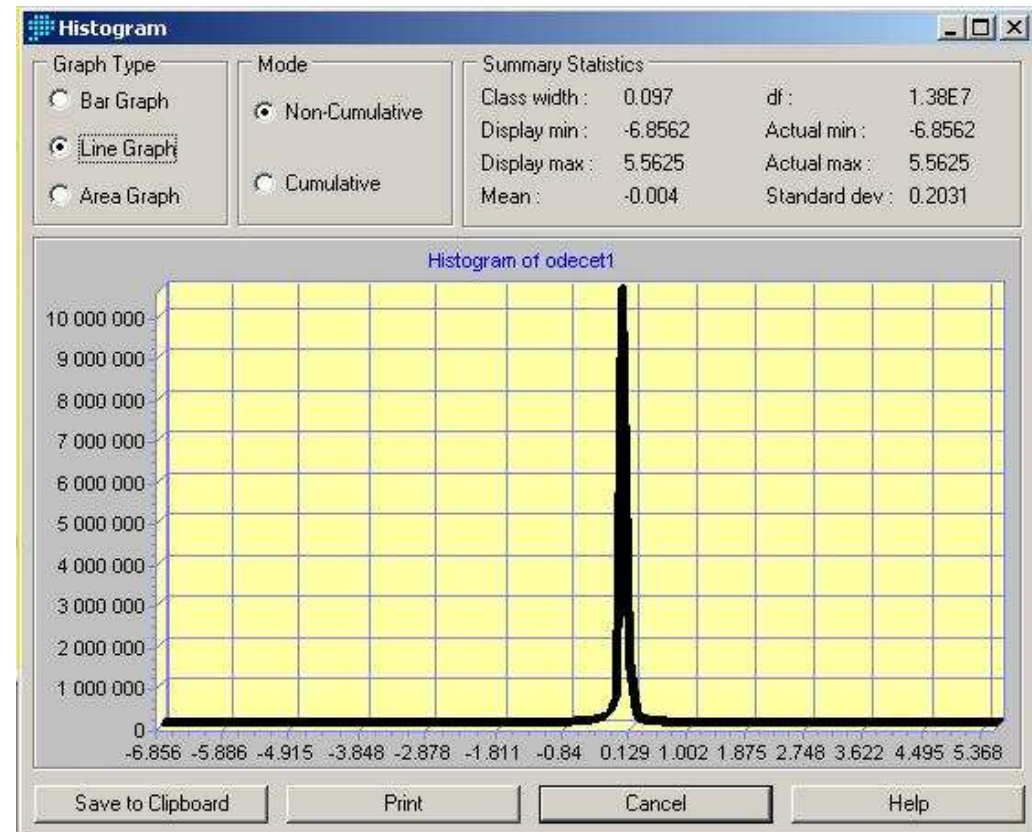
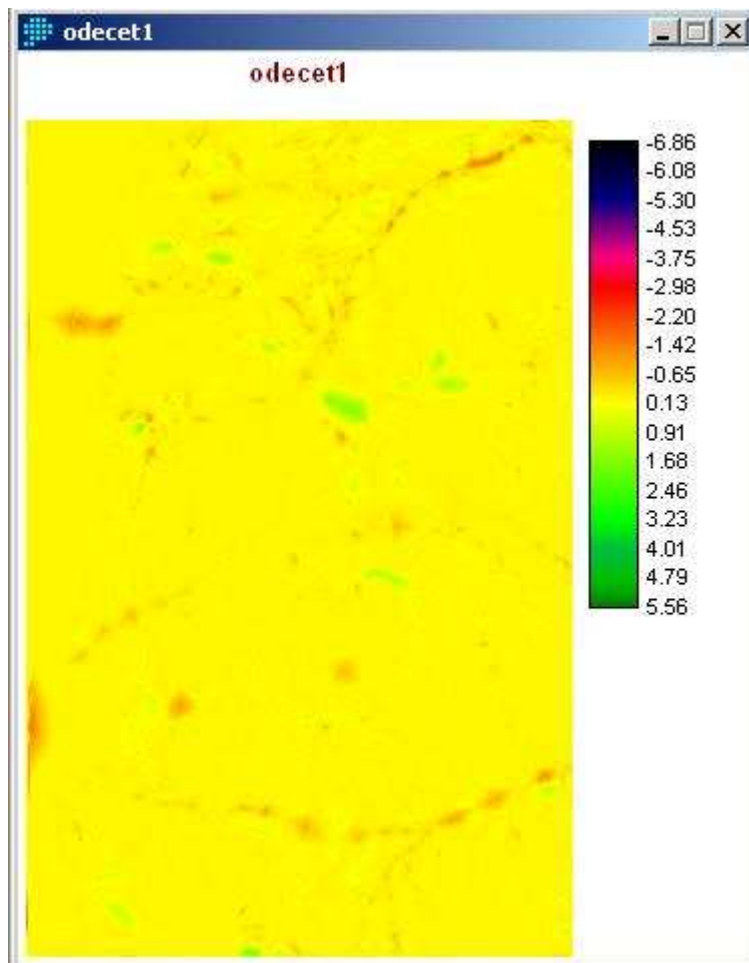




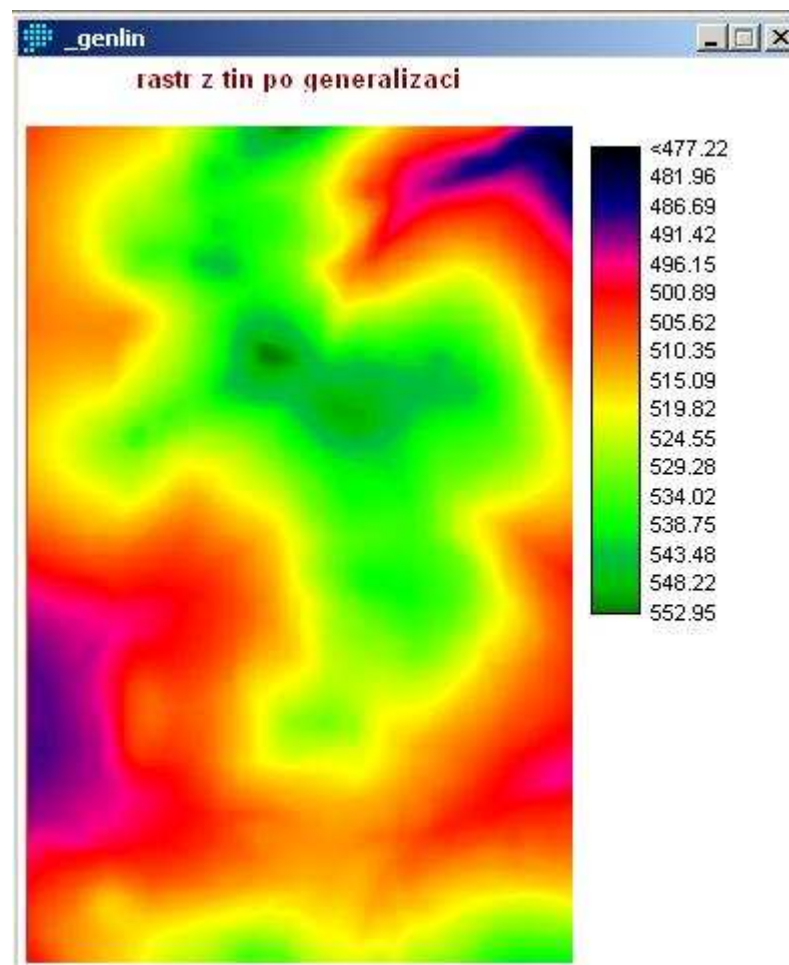
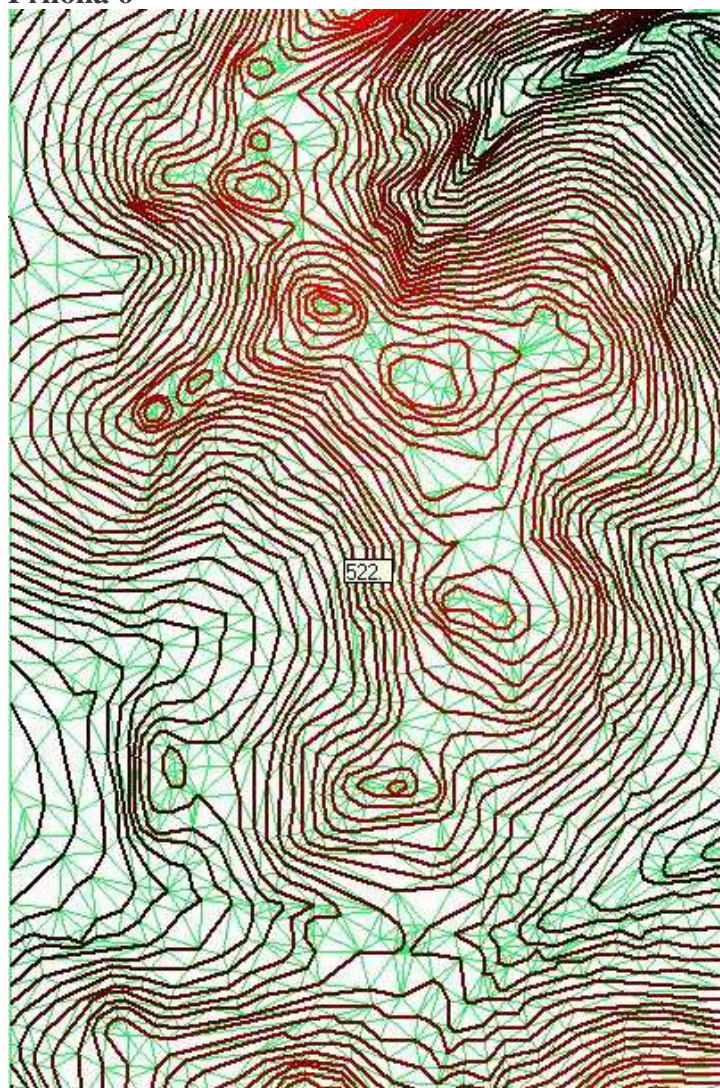
Příloha 5

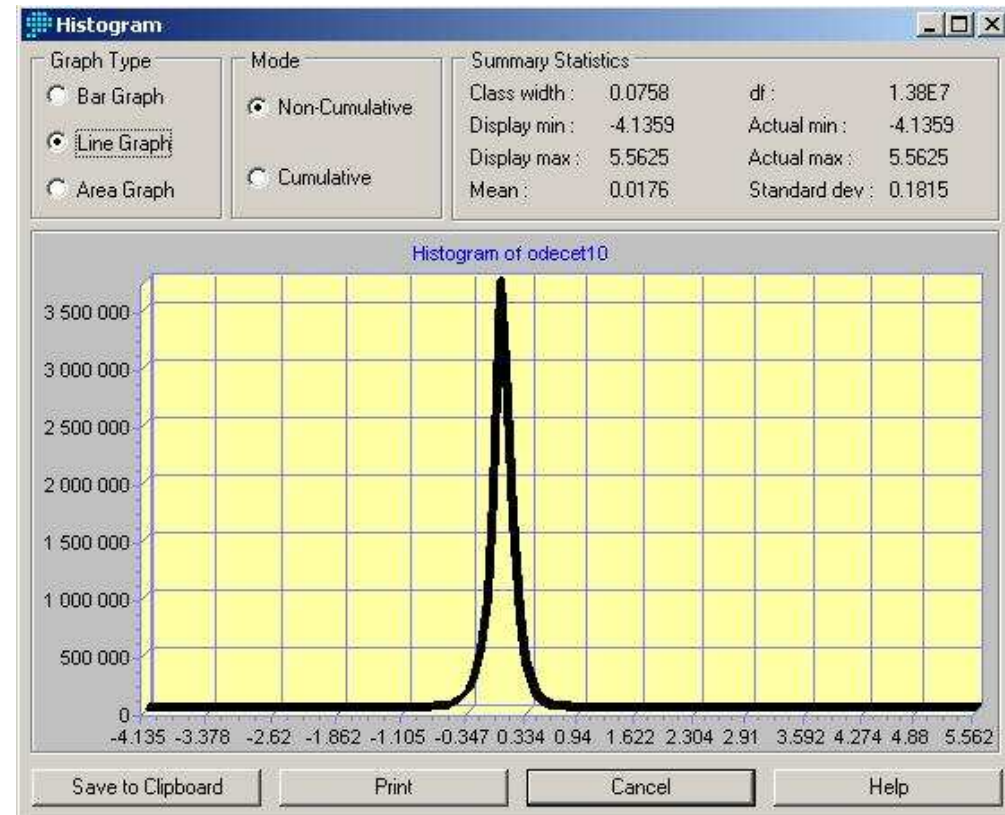
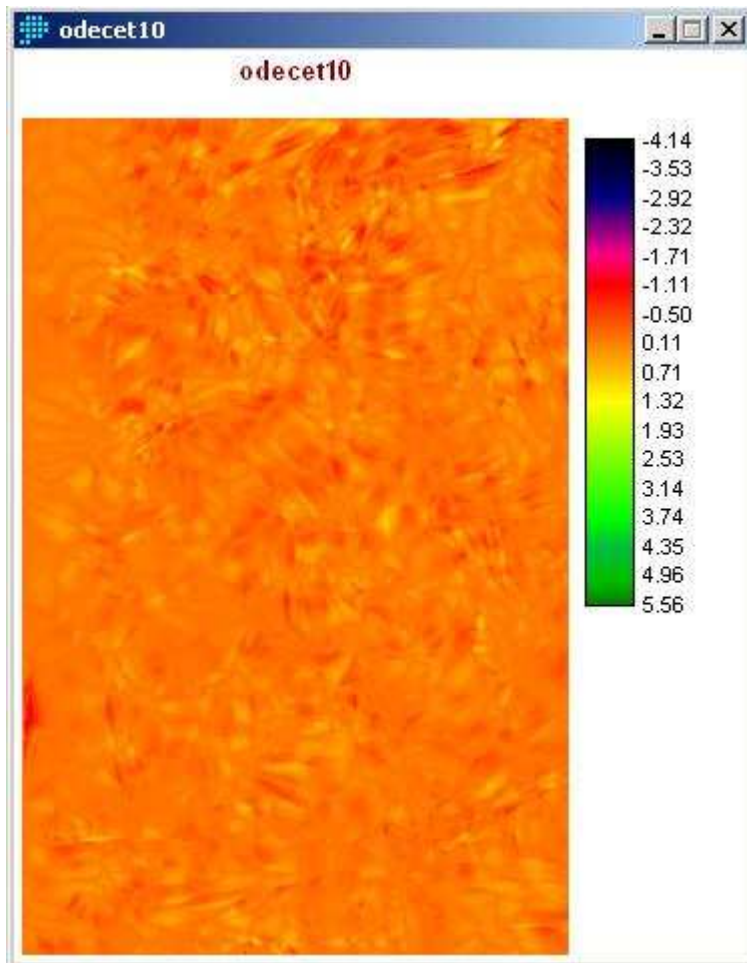




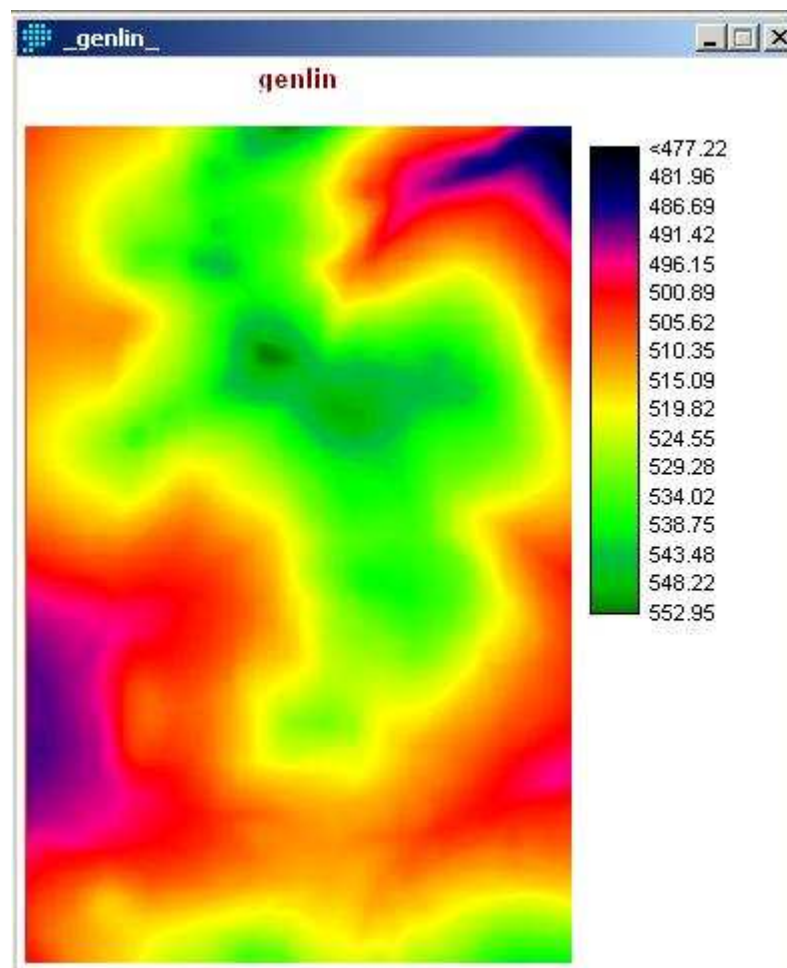


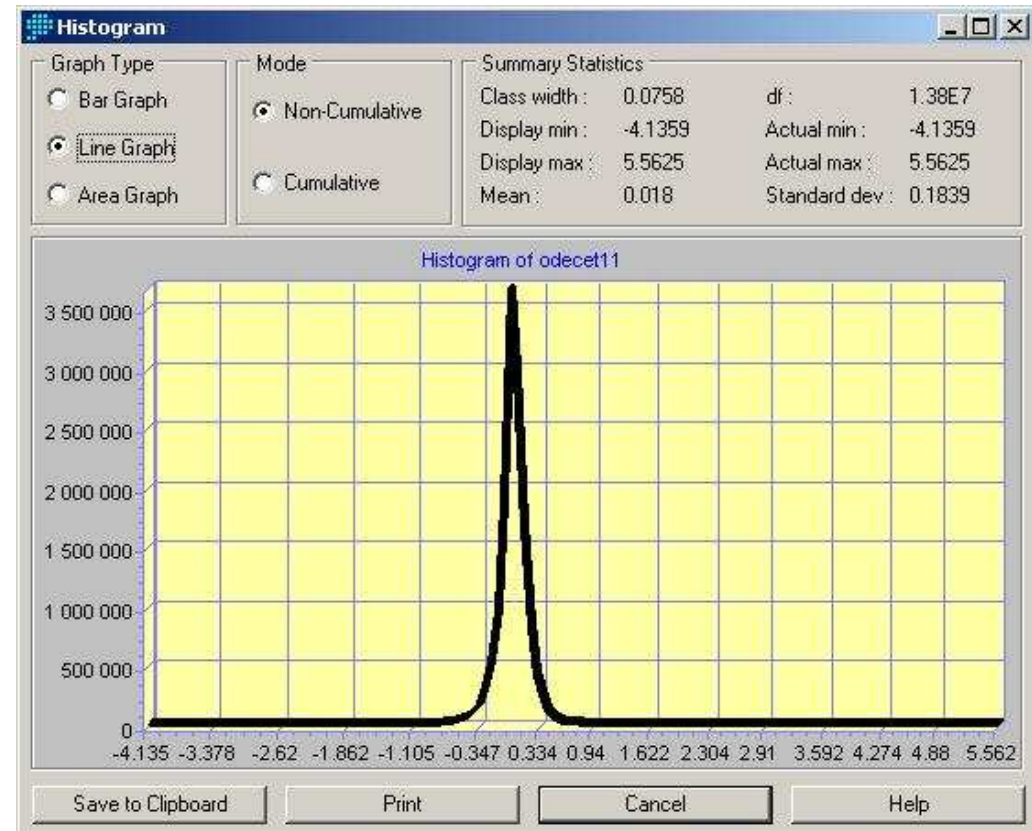
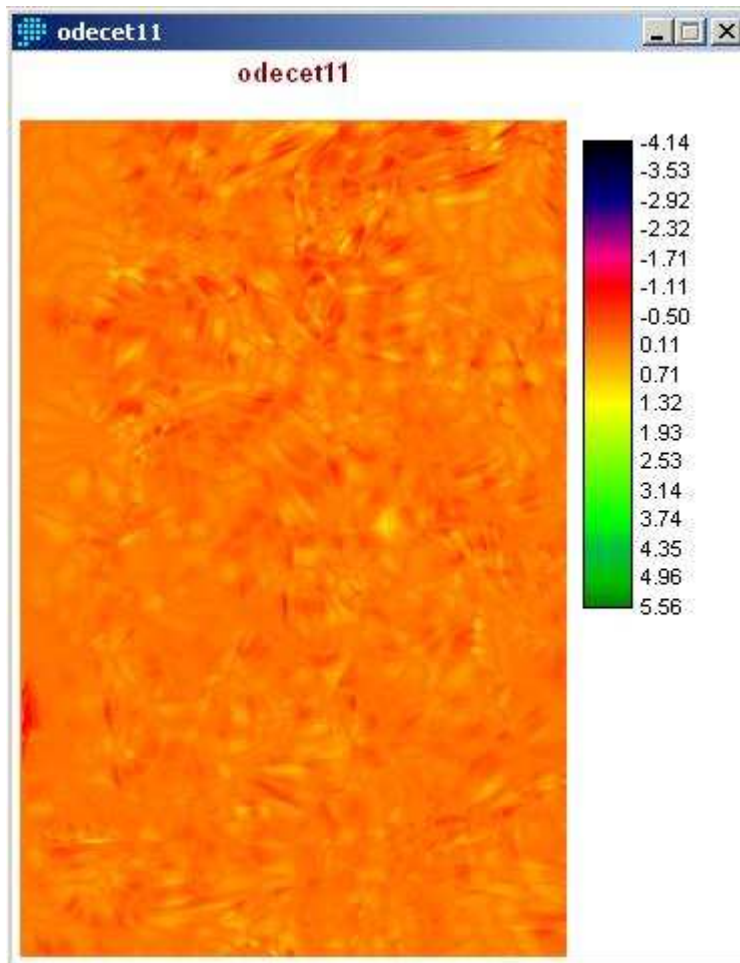
Příloha 6



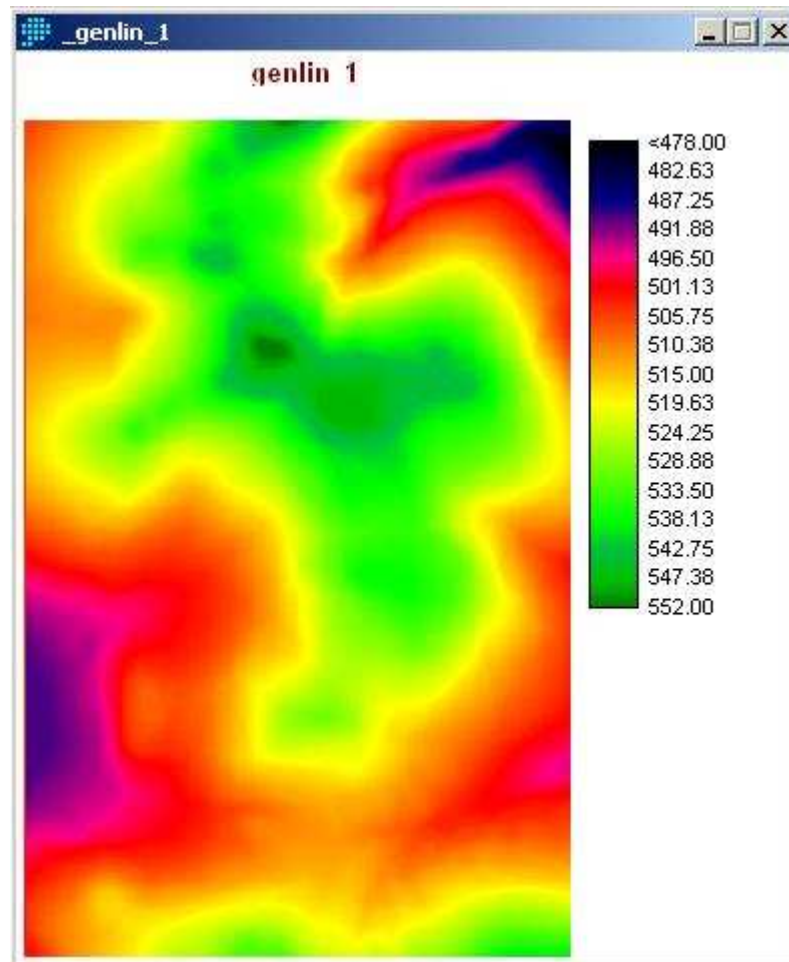


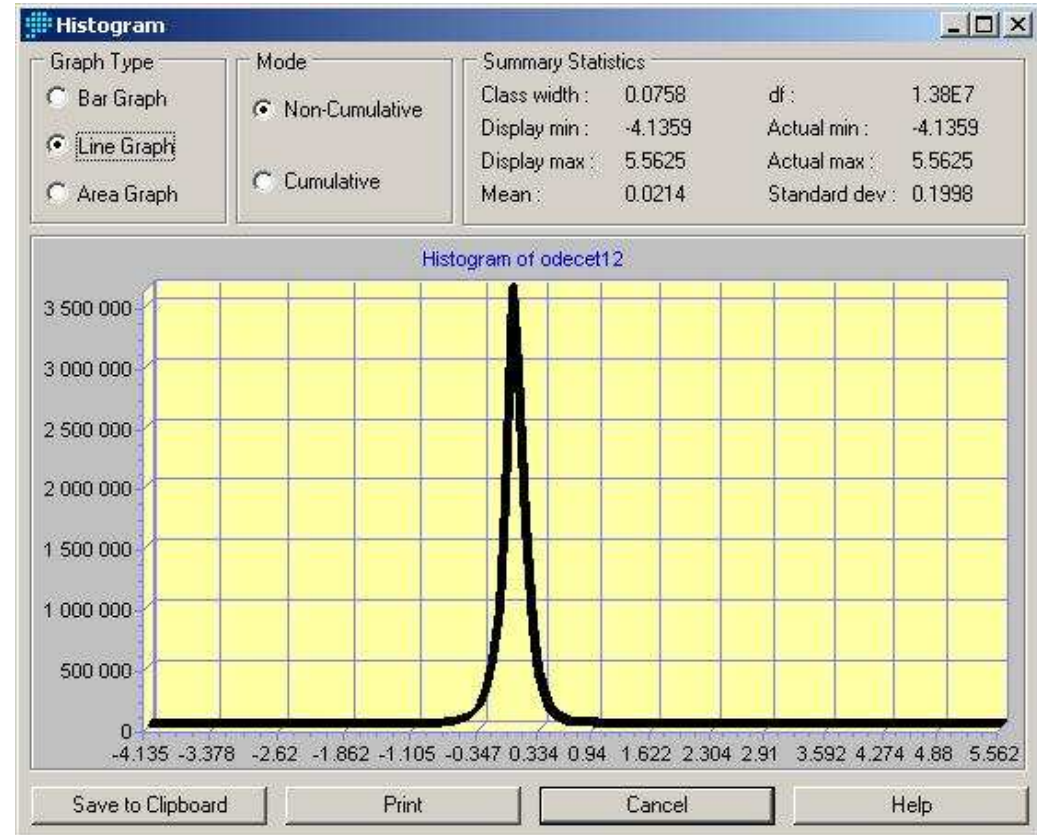
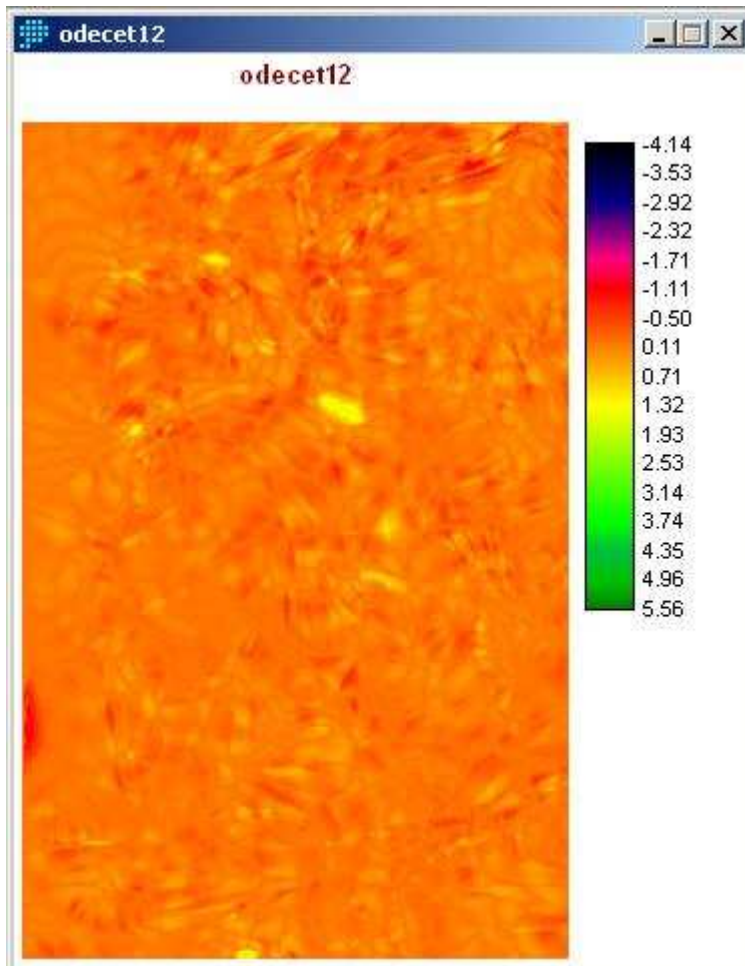
Příloha 7



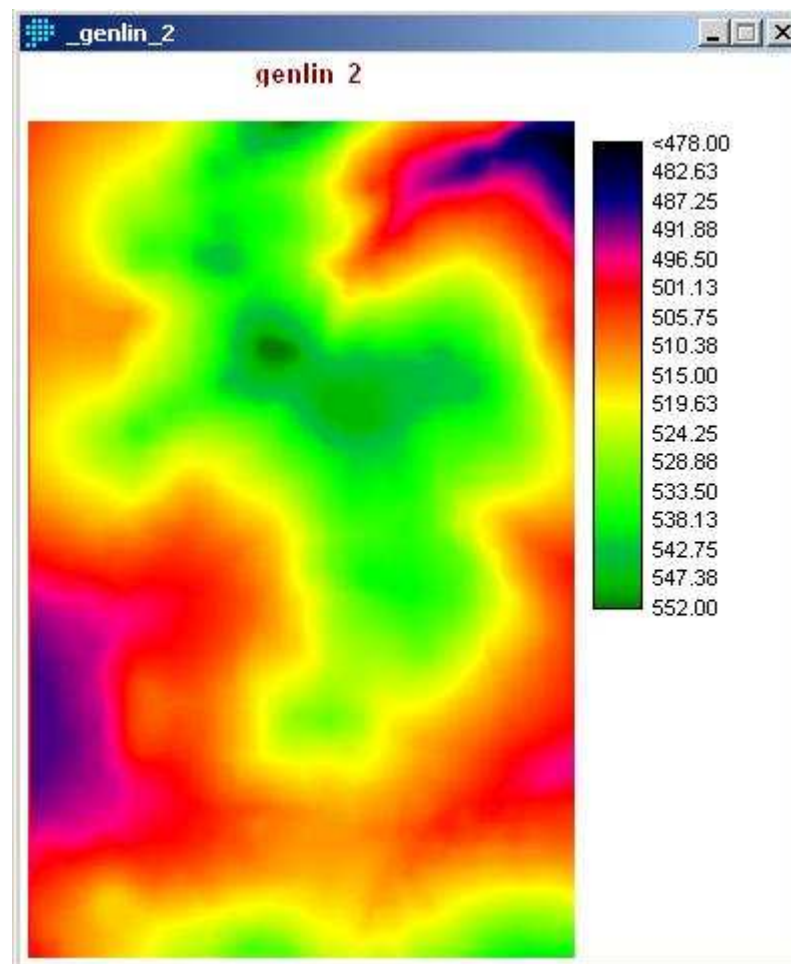
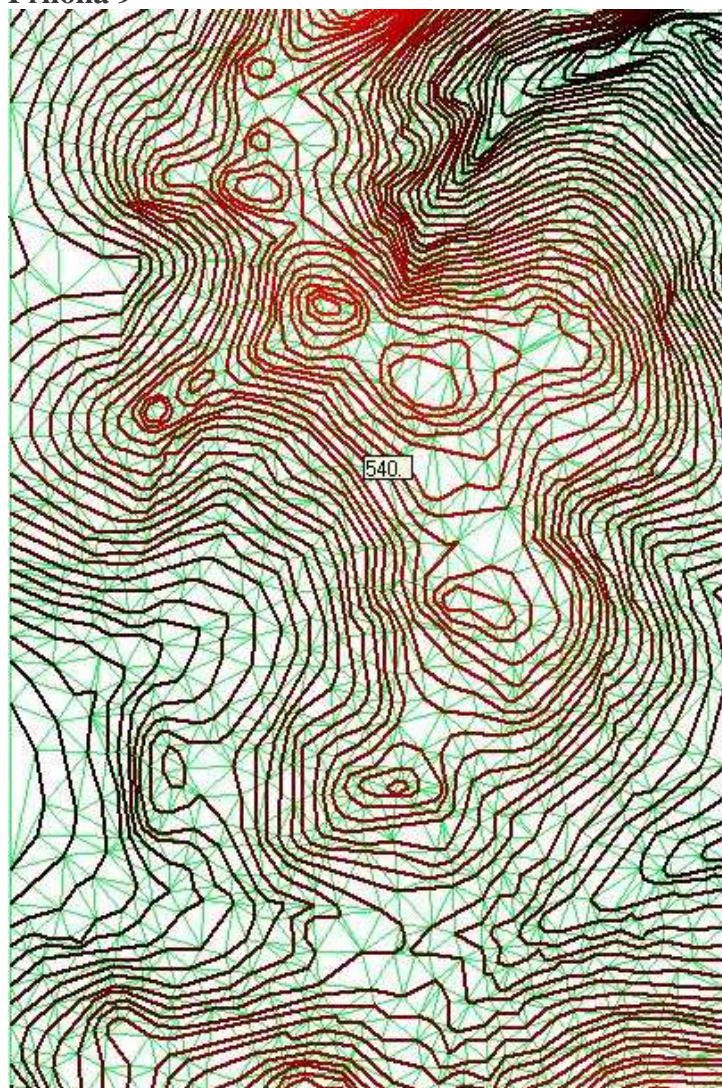


Příloha 8

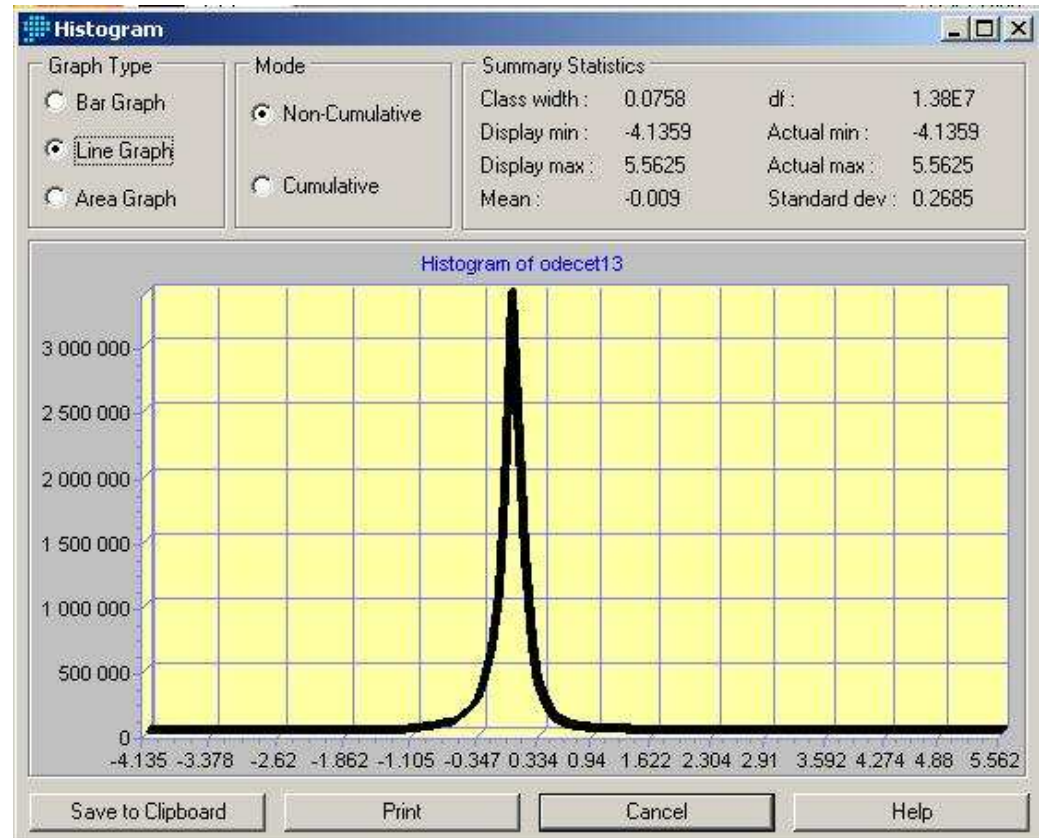
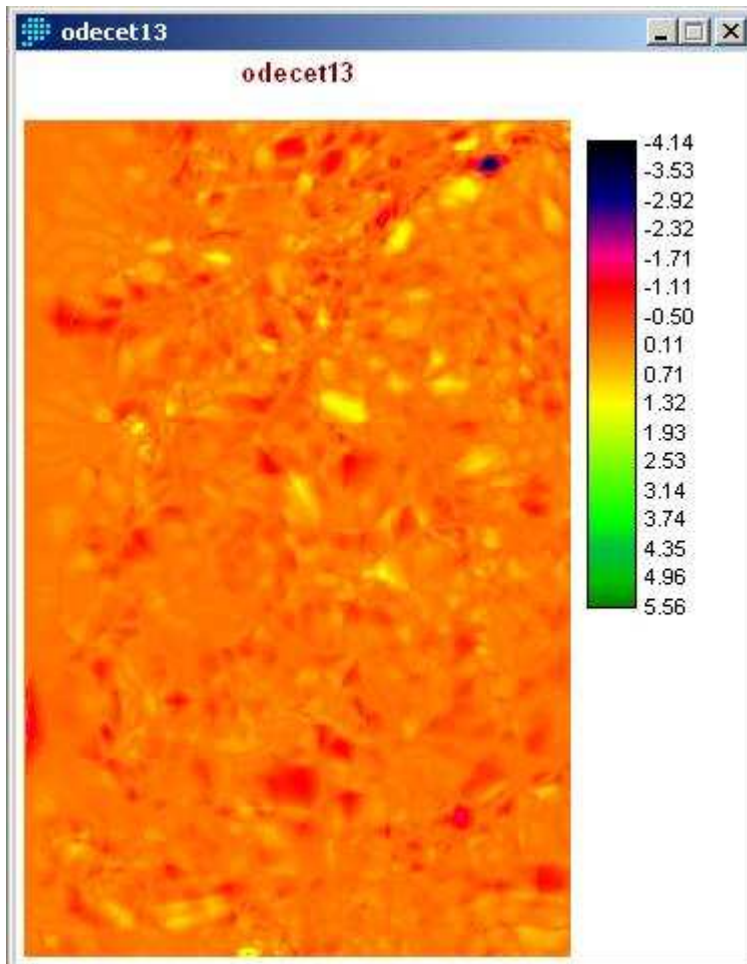




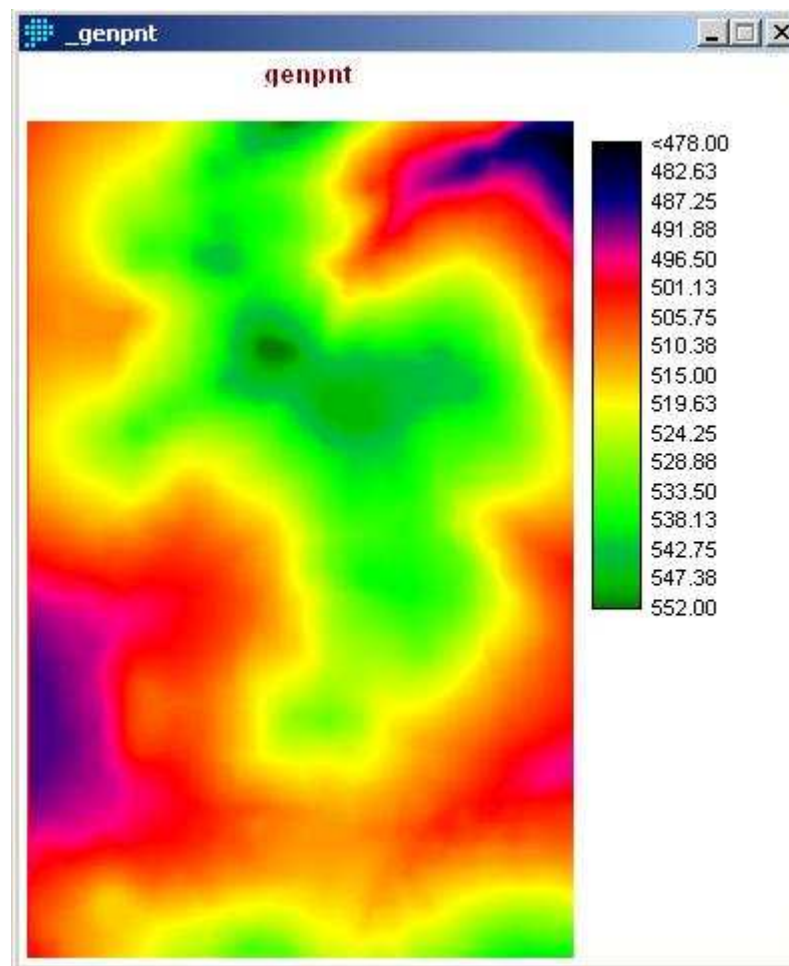
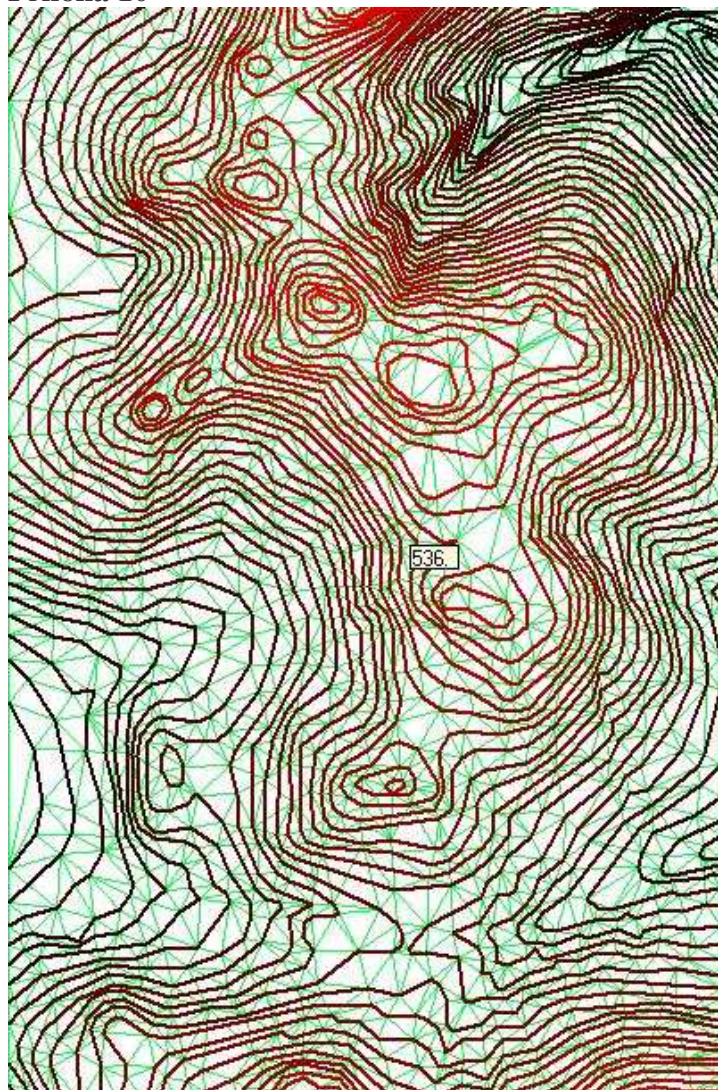
Příloha 9

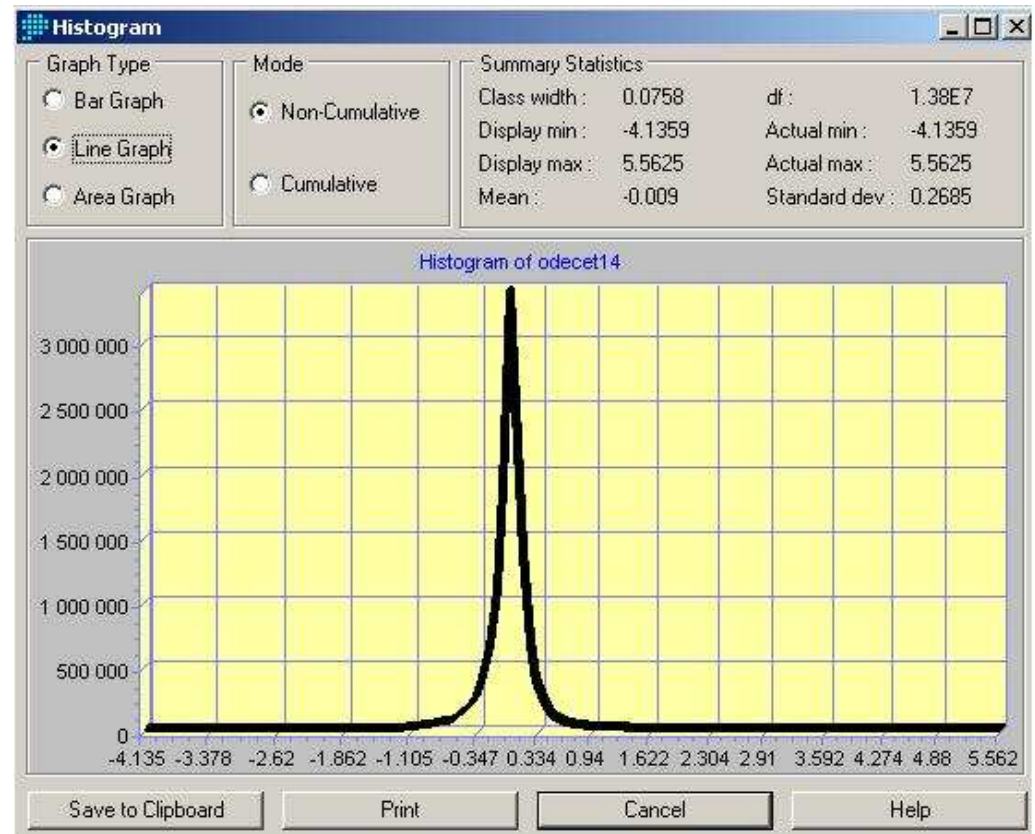
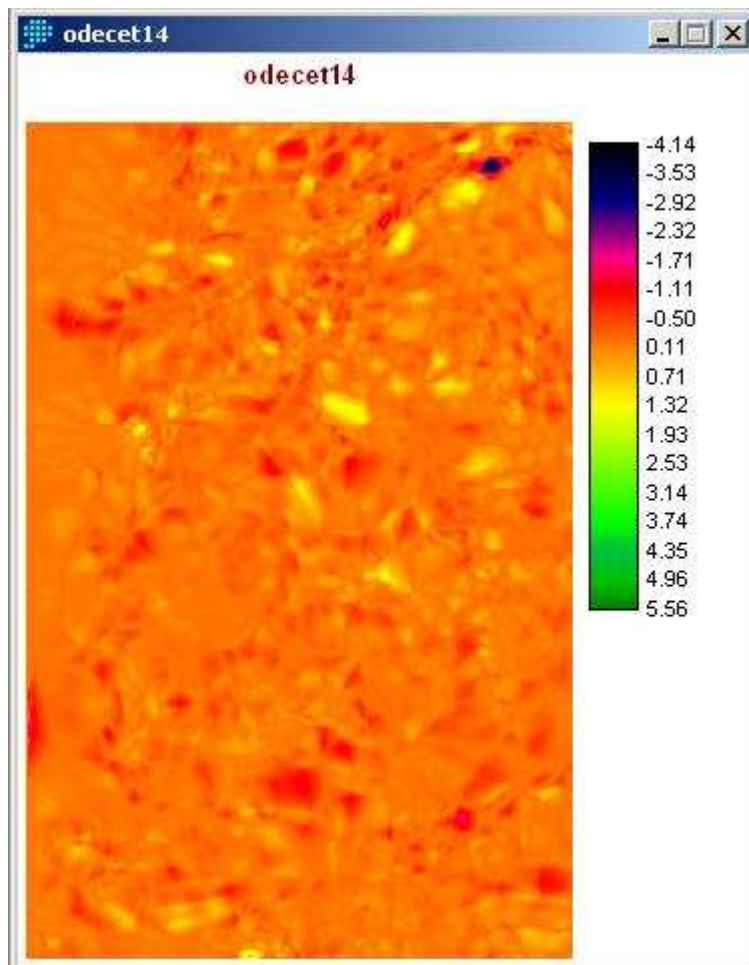




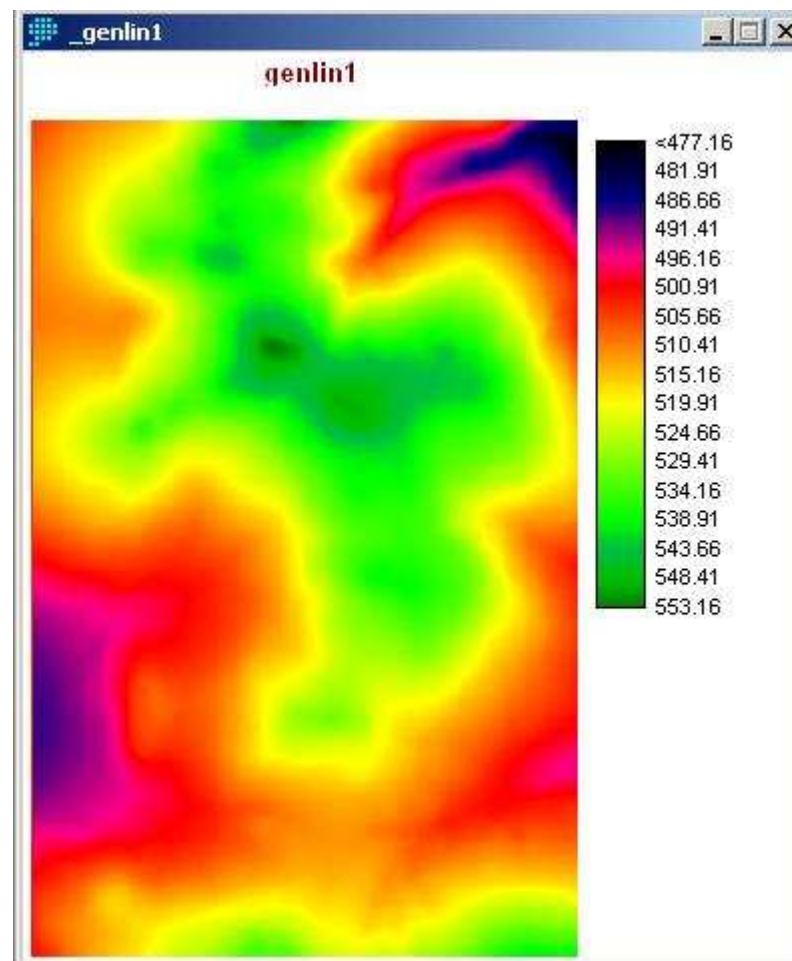
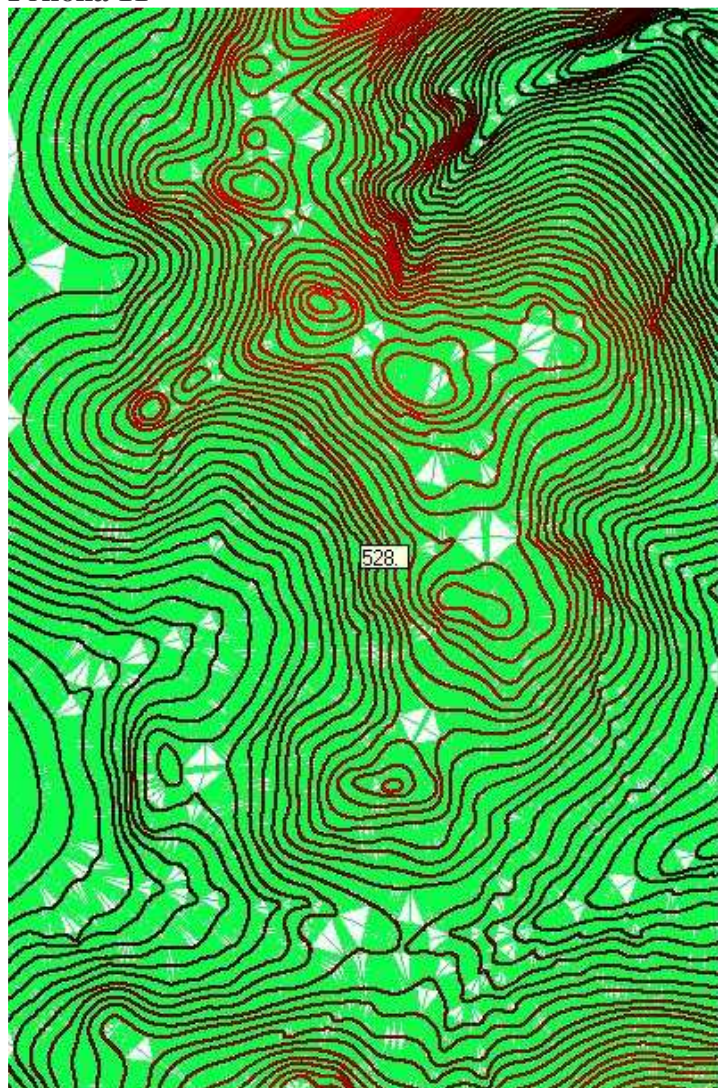


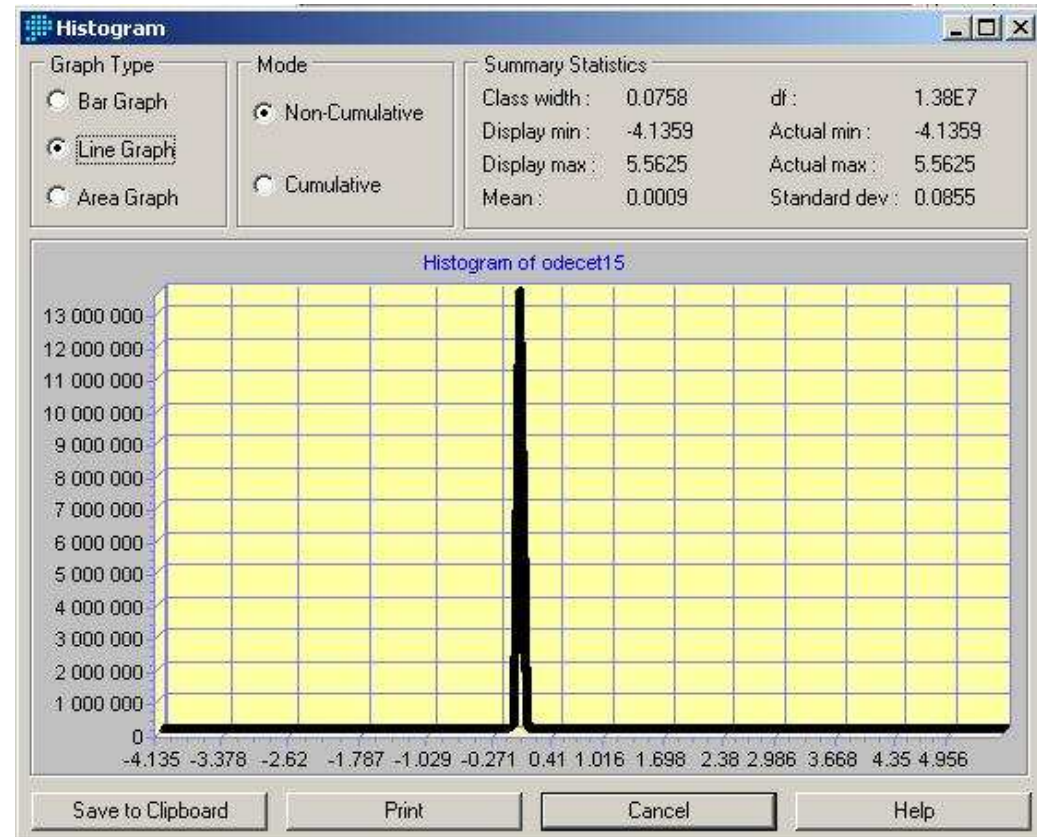
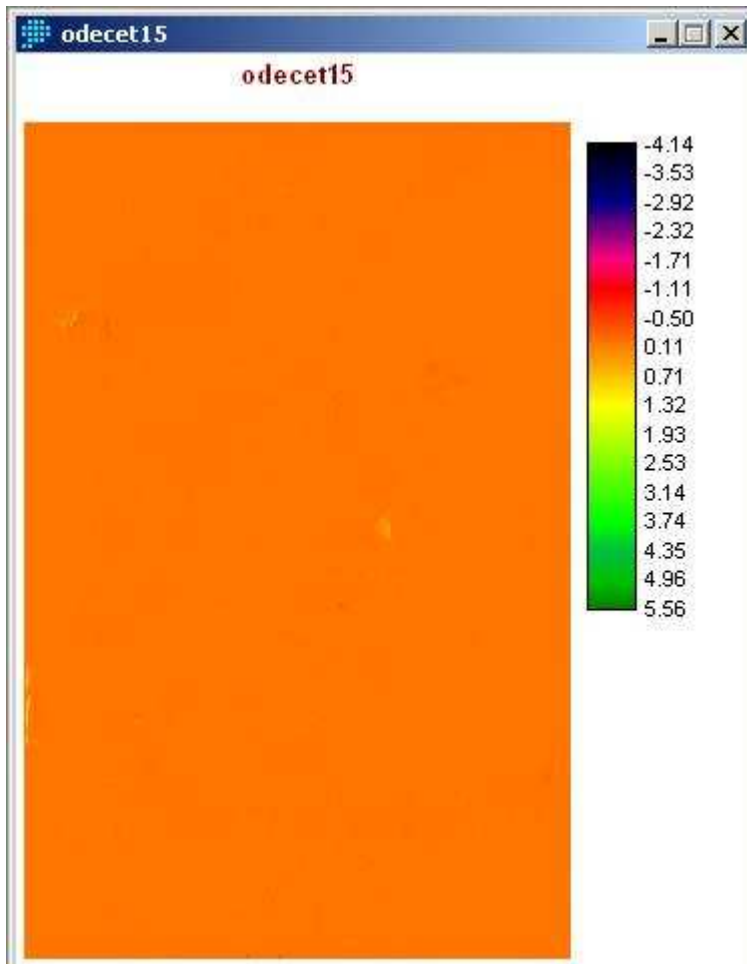
Příloha 10



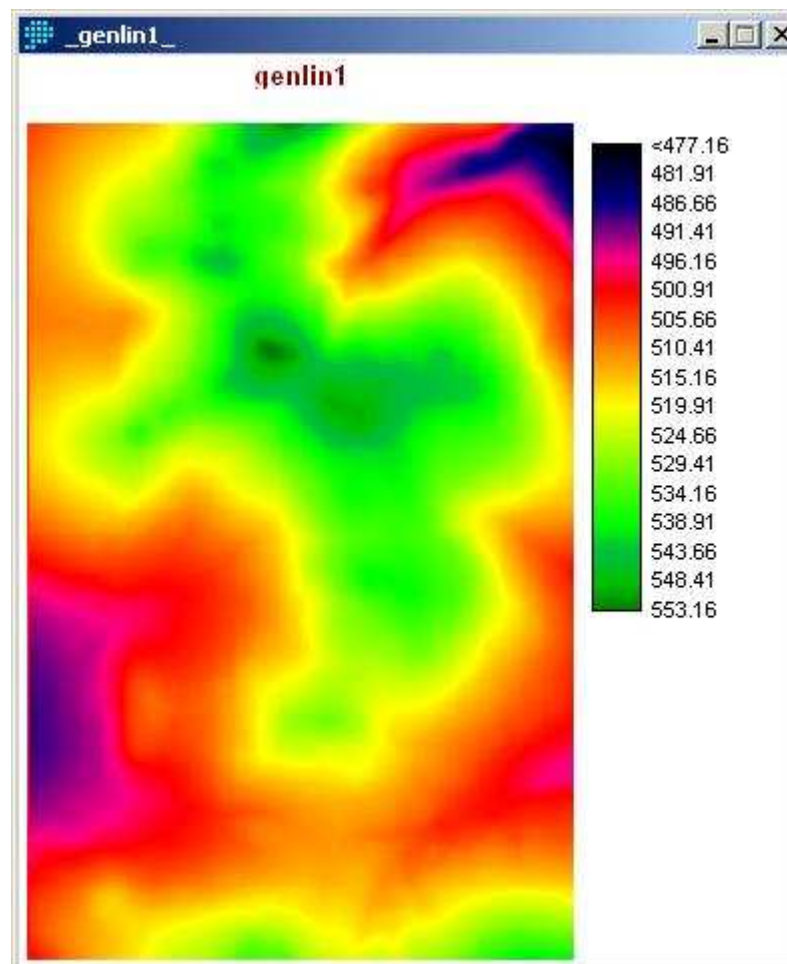
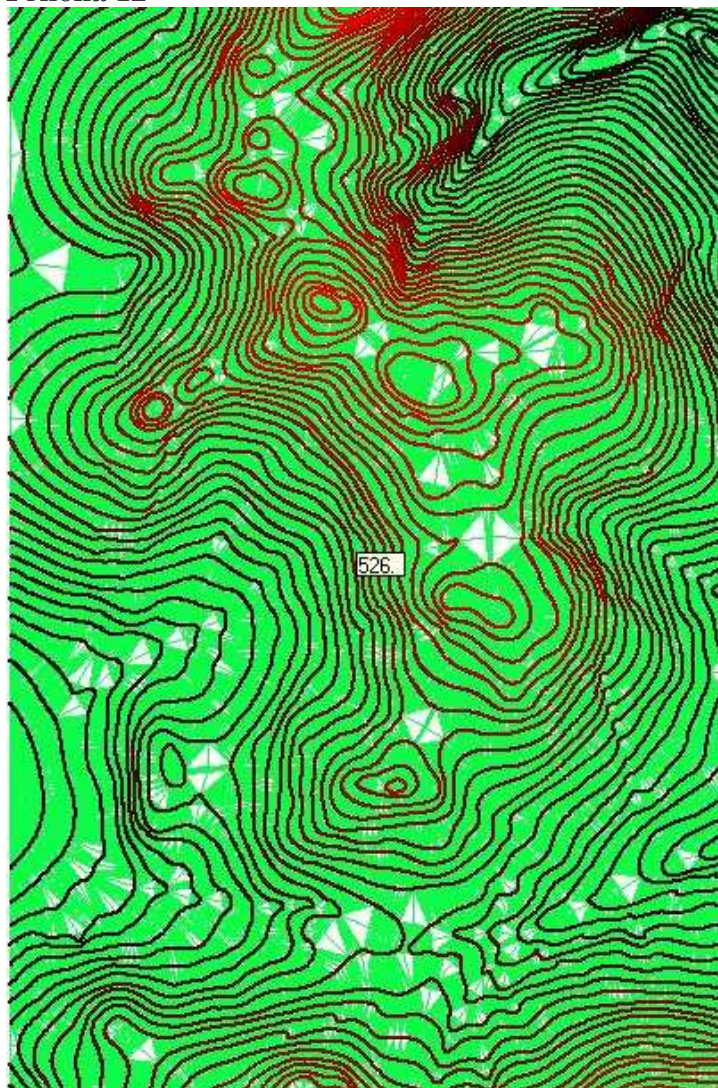


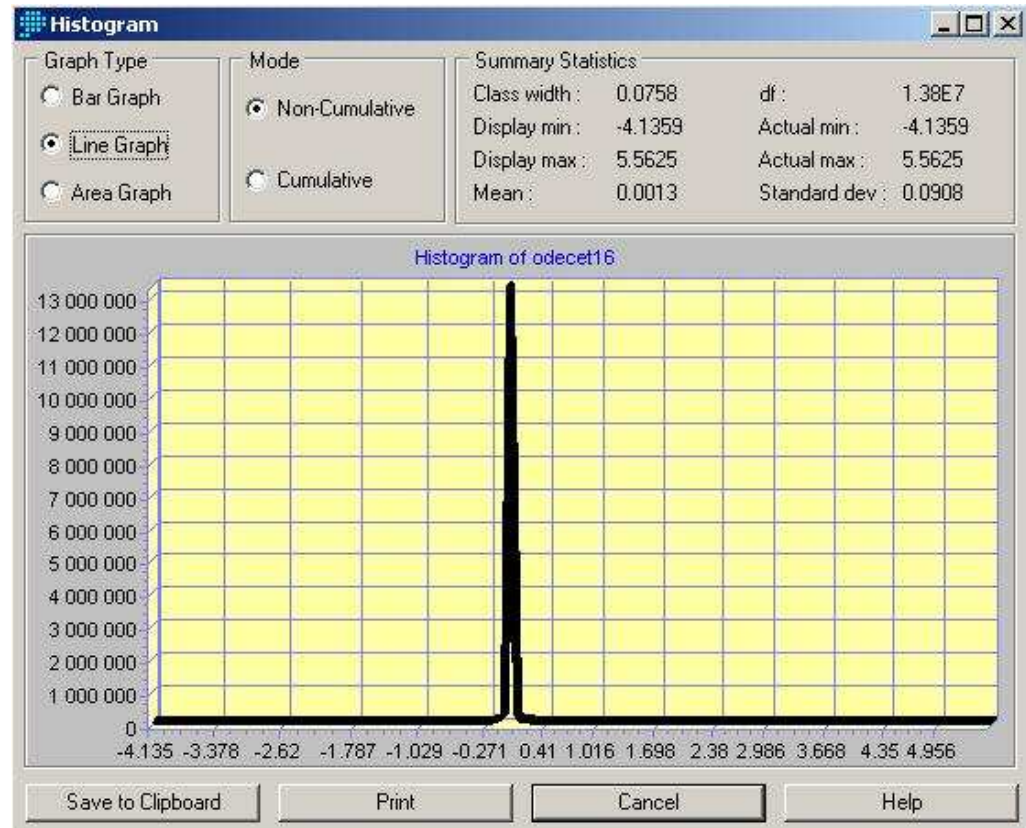
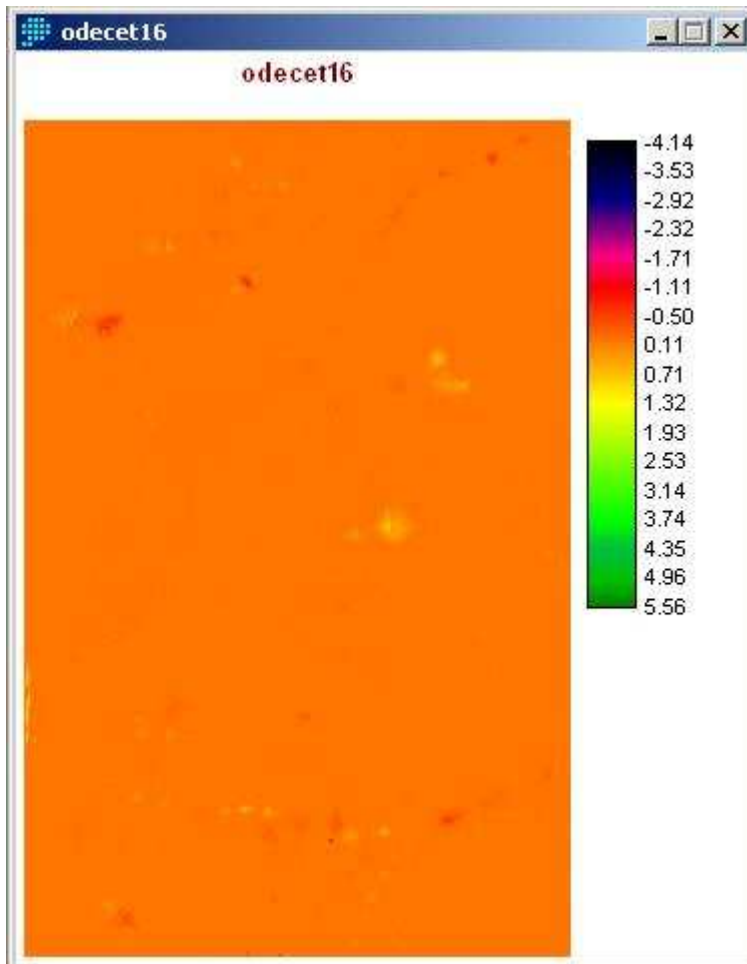
## Příloha 11



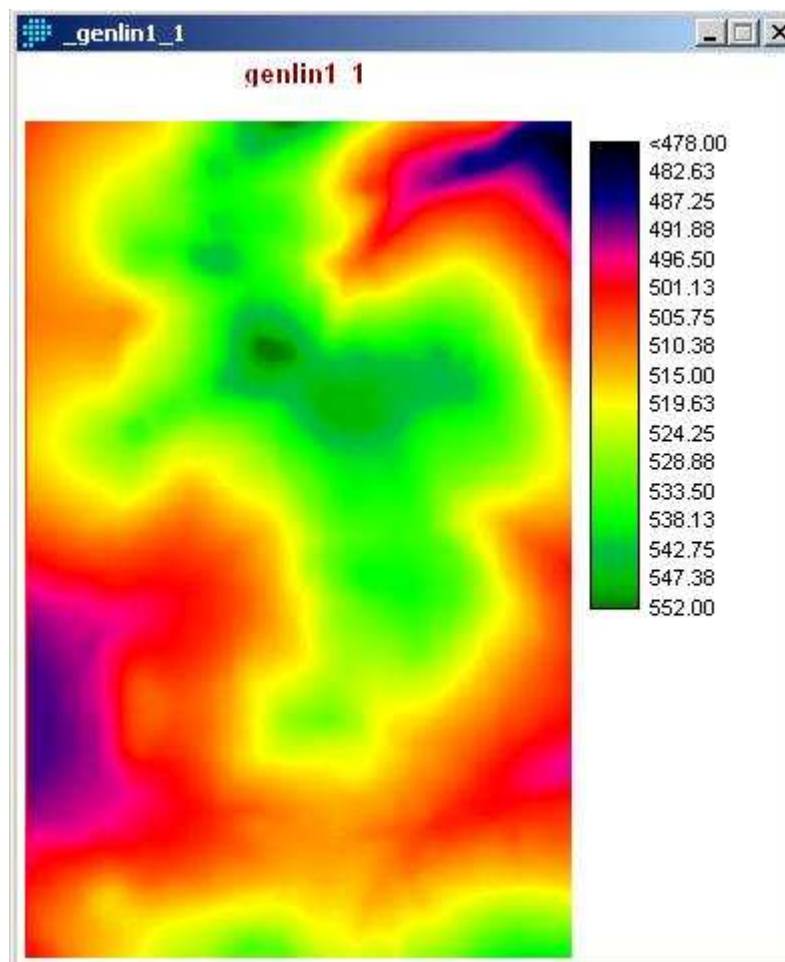
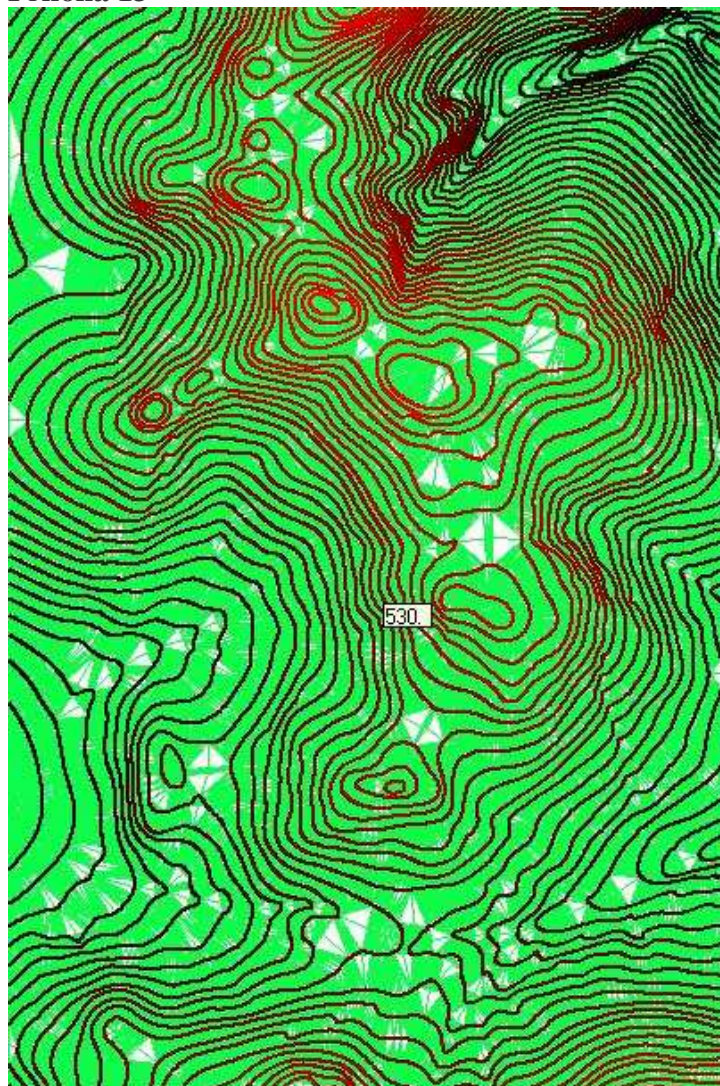


Příloha 12

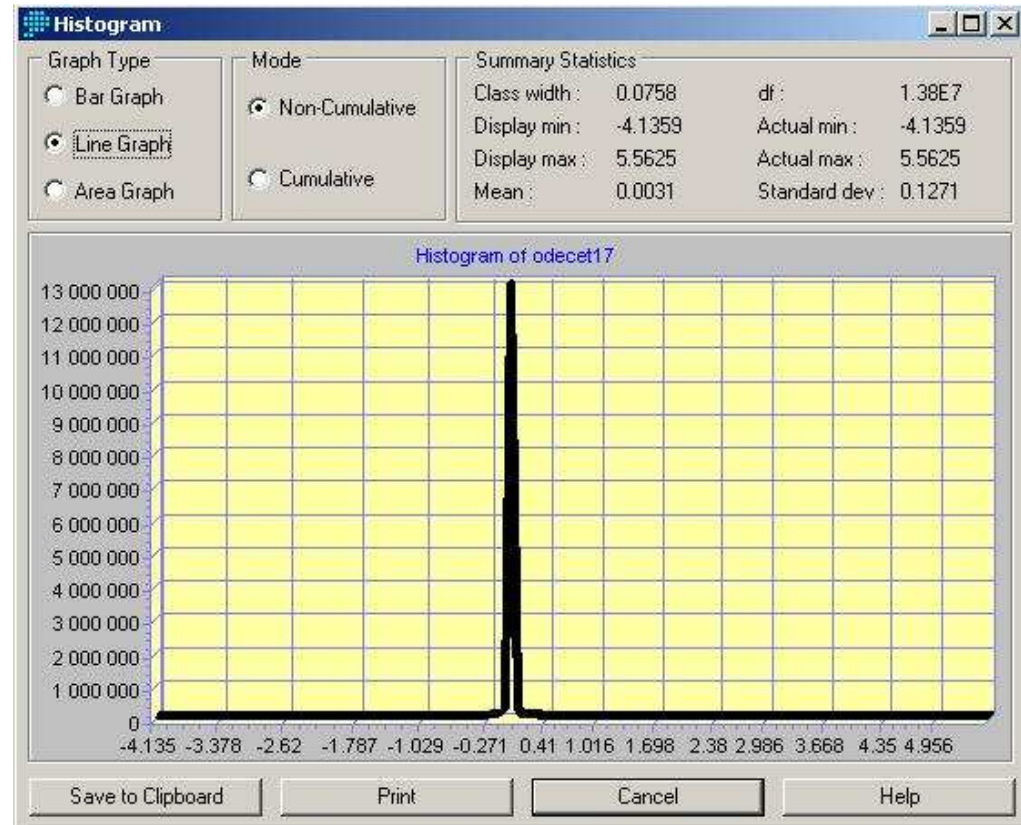
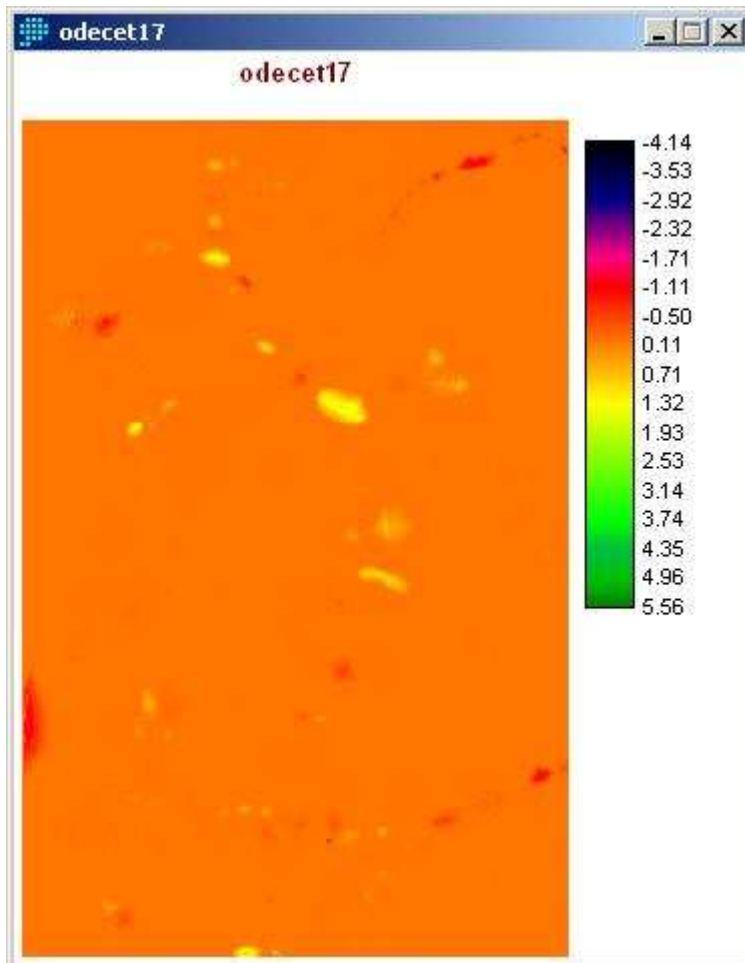




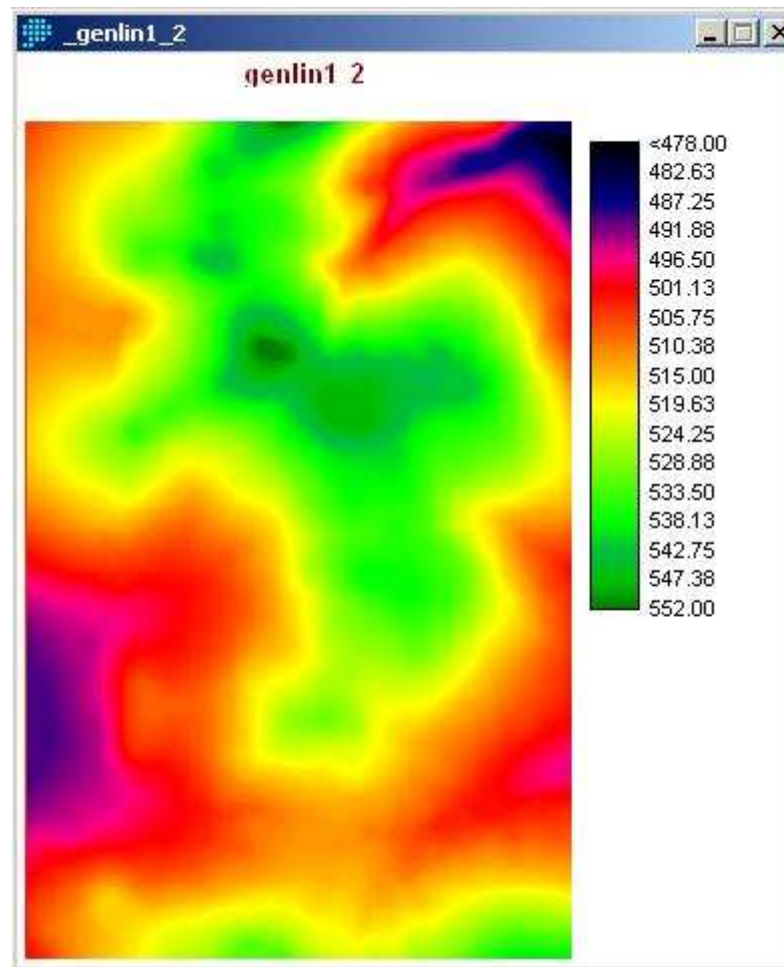
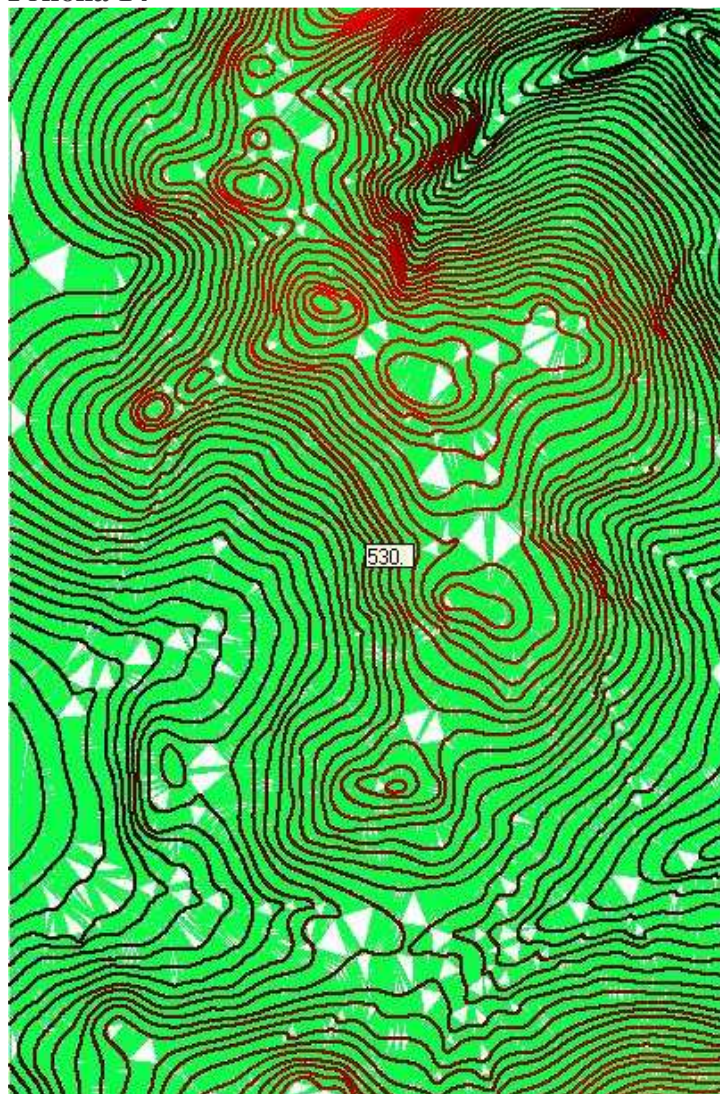
Příloha 13

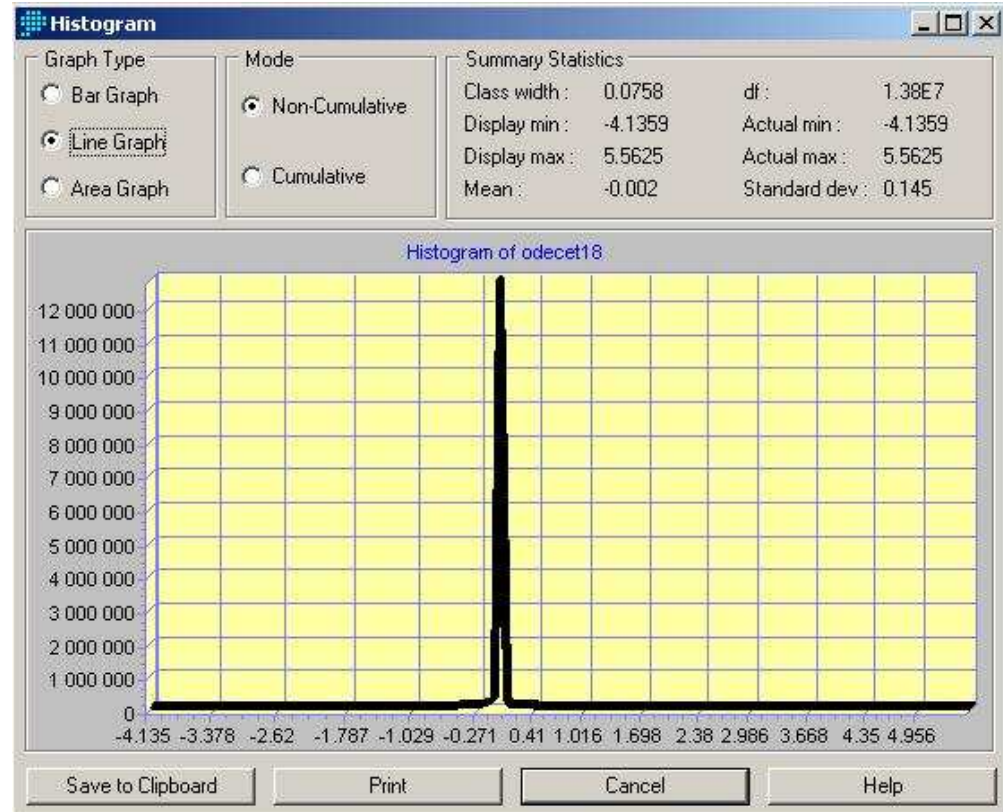
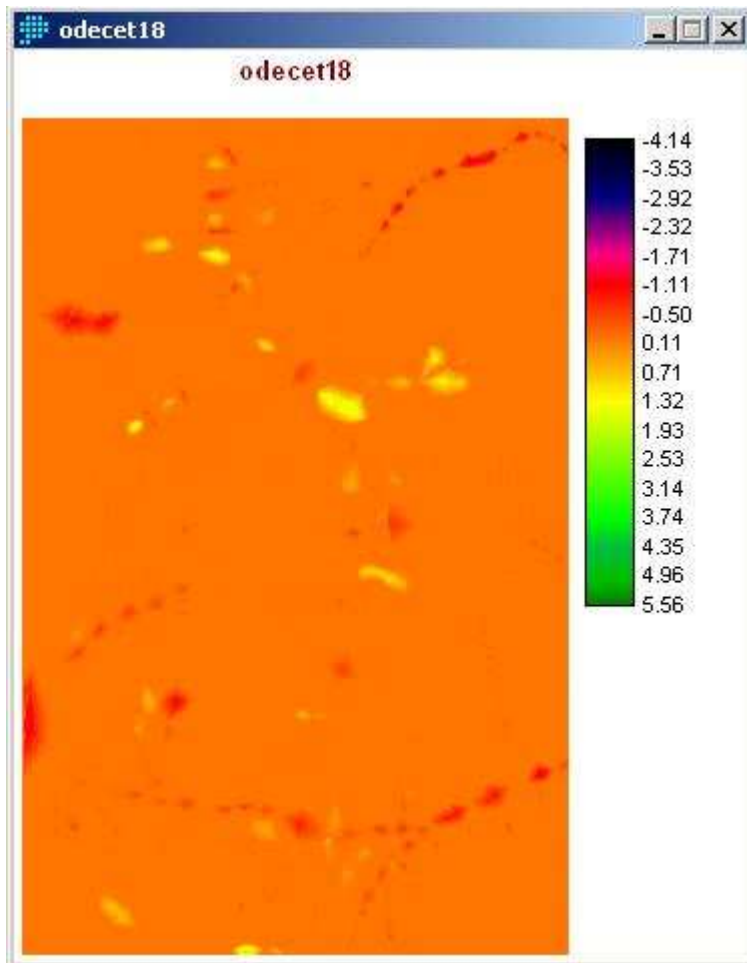






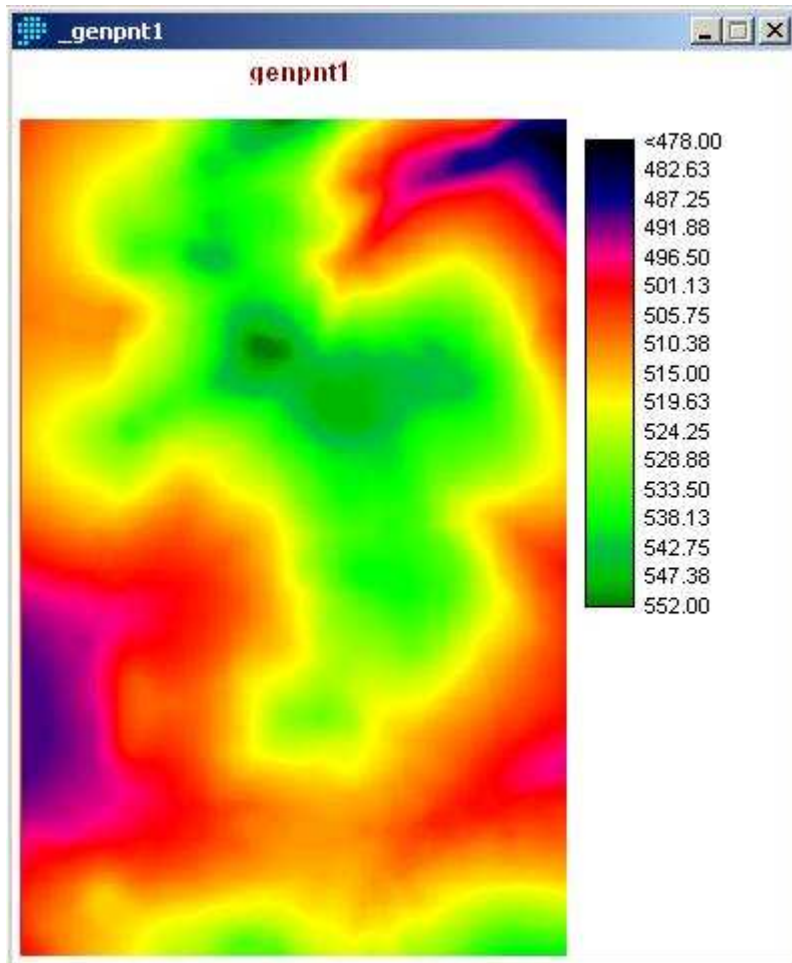
Příloha 14

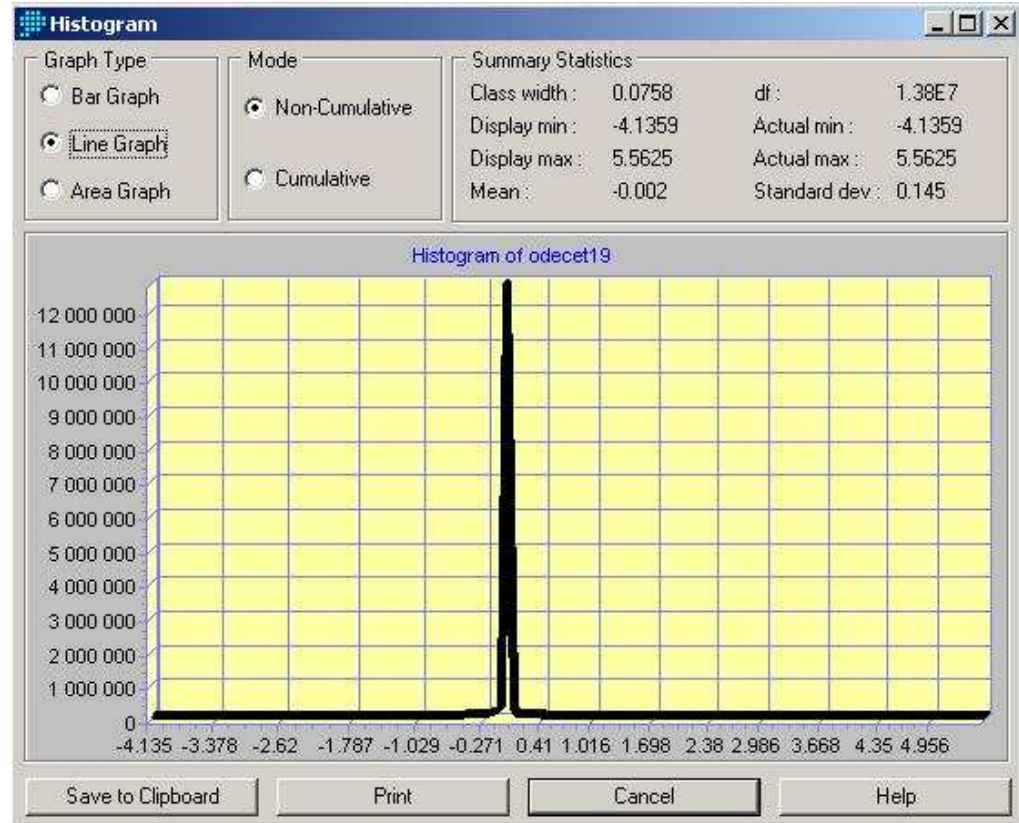
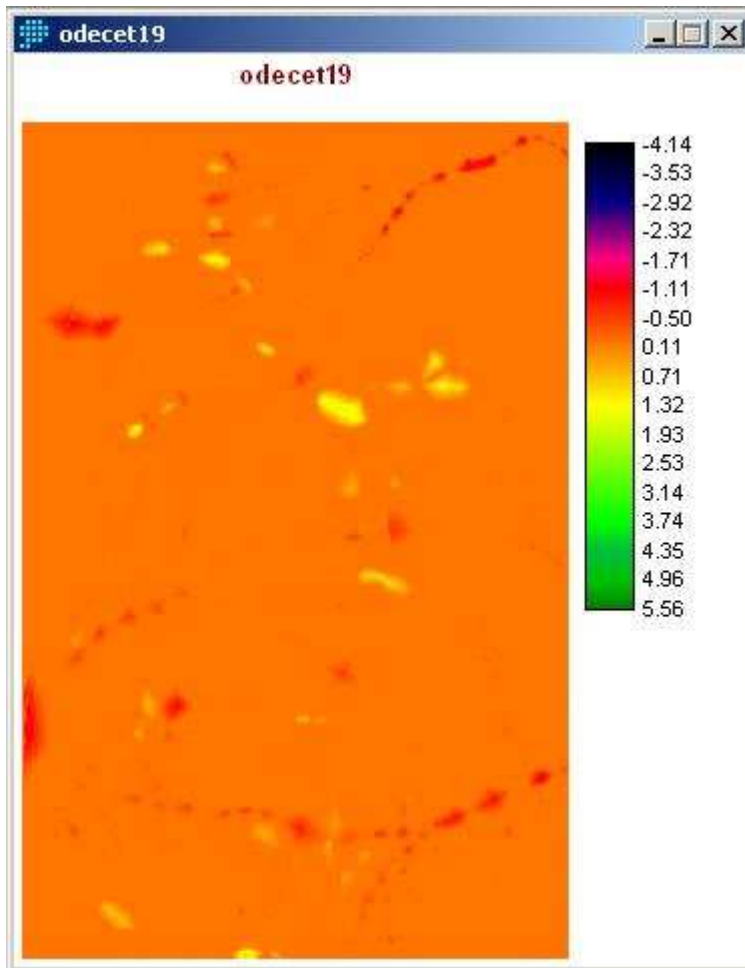




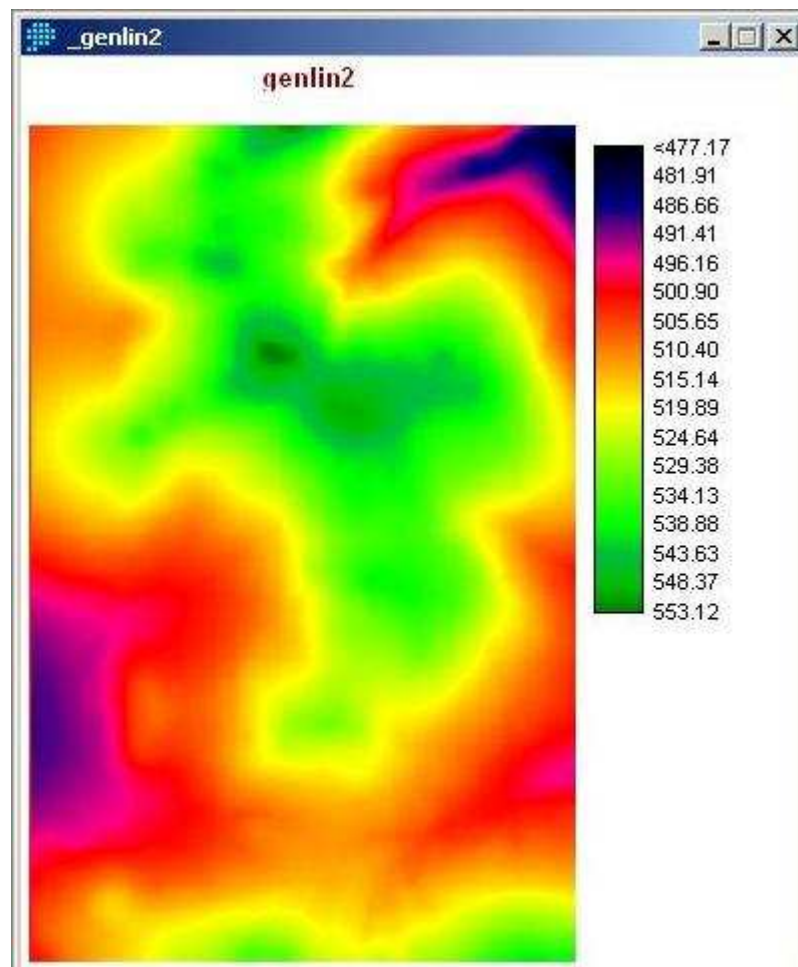
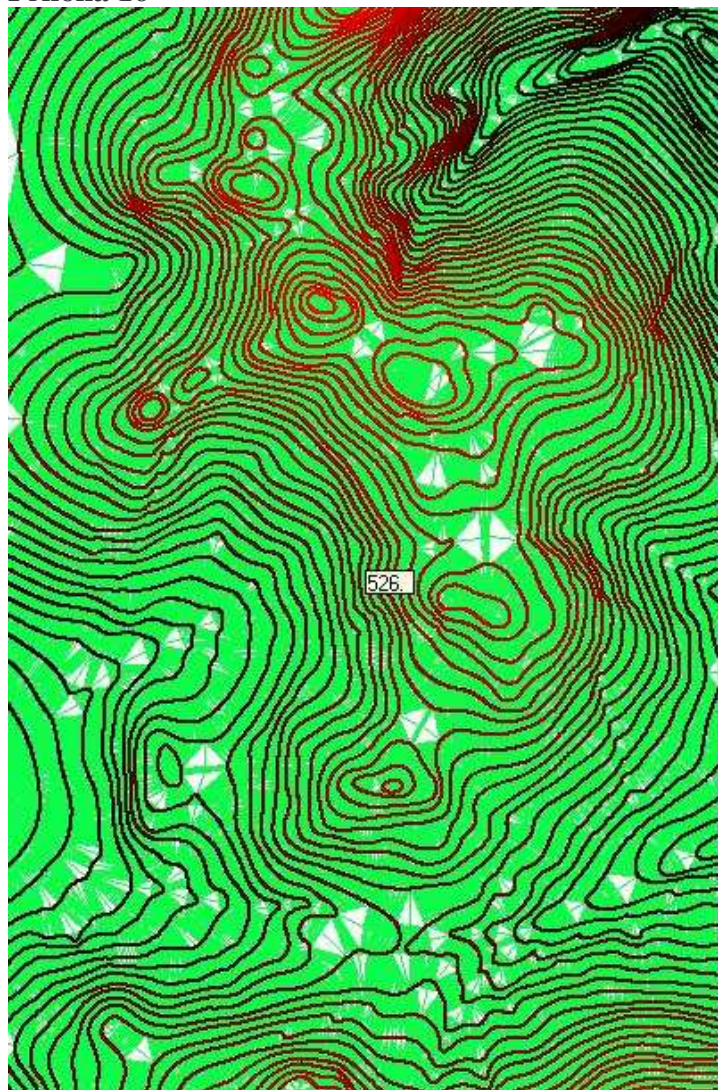
Příloha 15

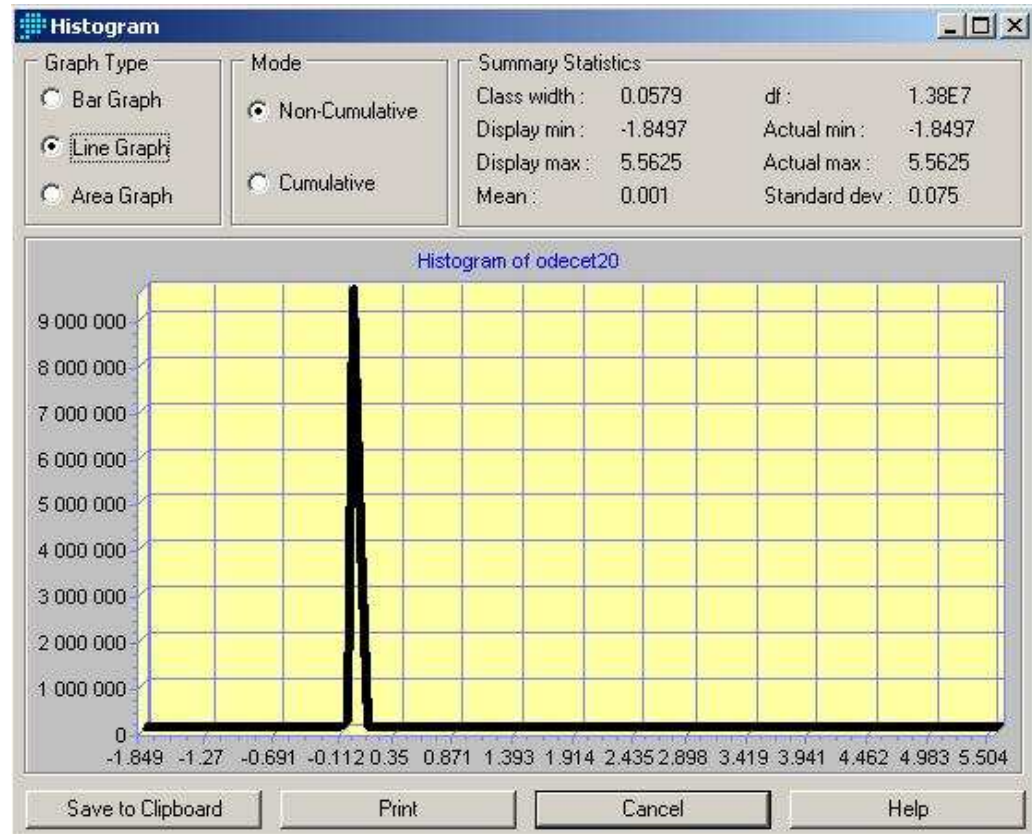
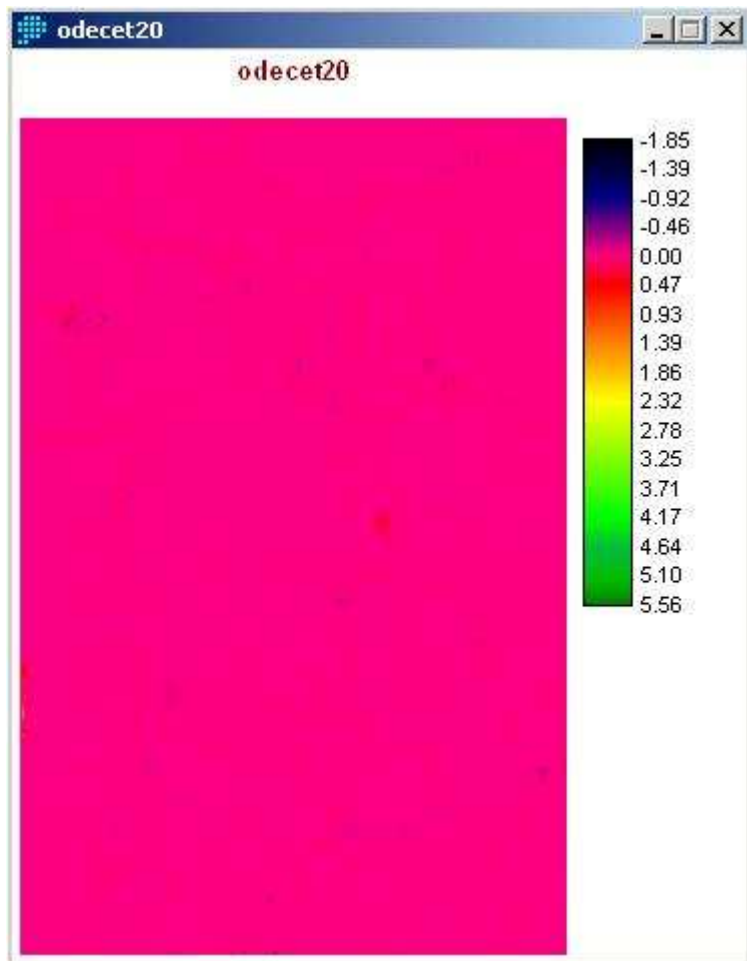






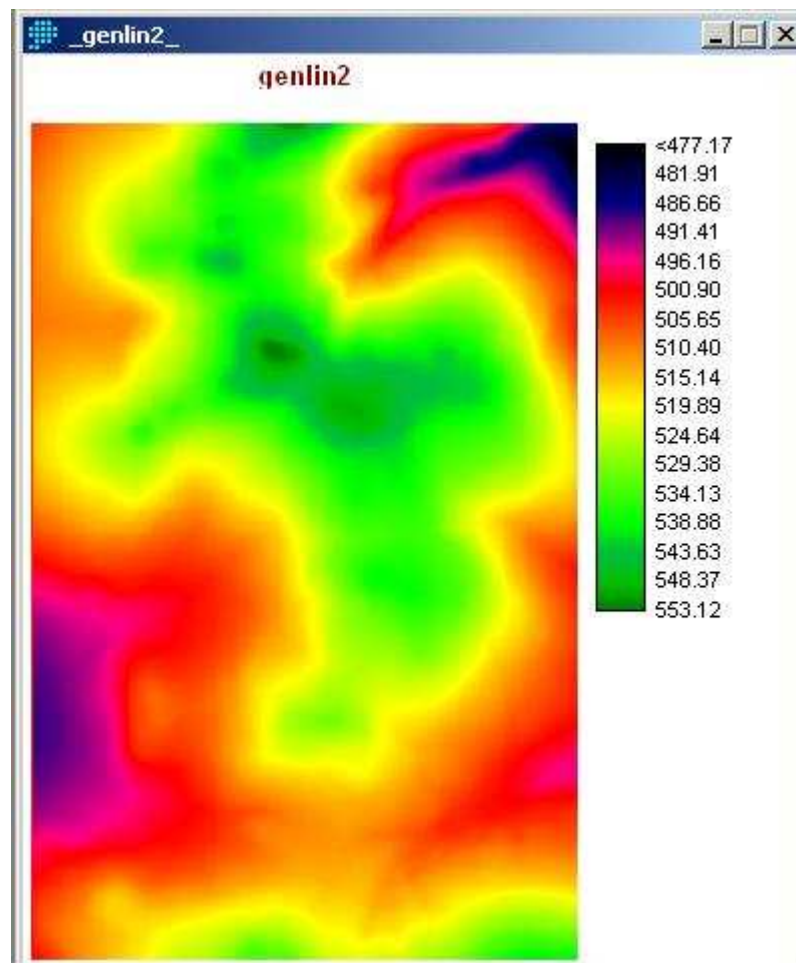
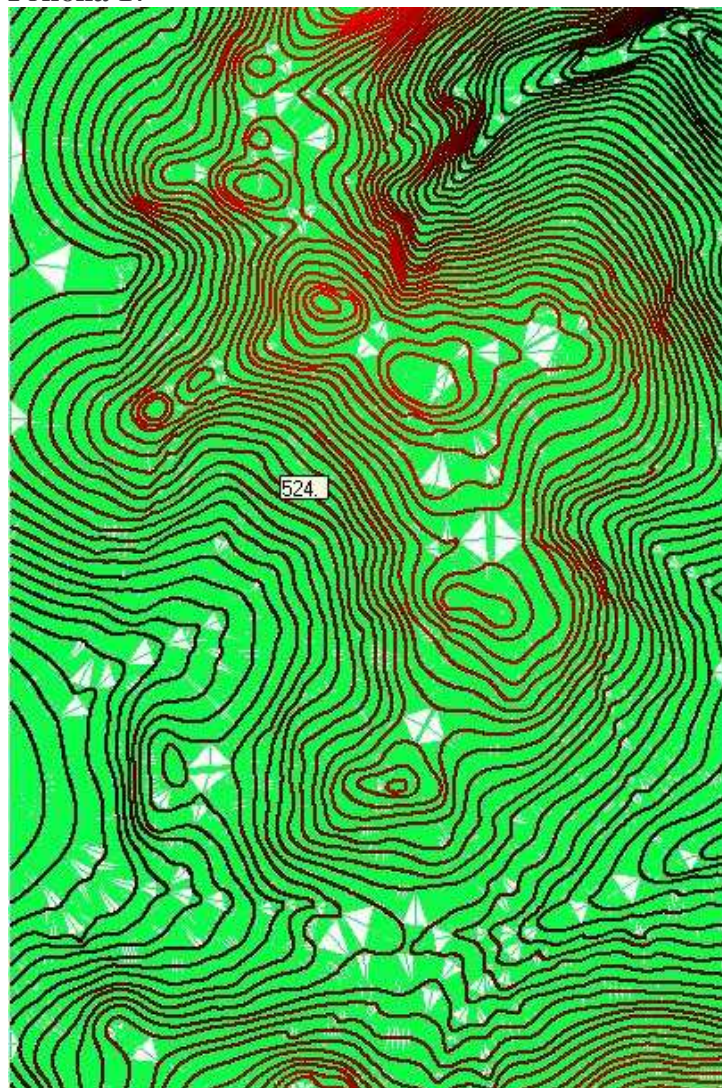
Příloha 16

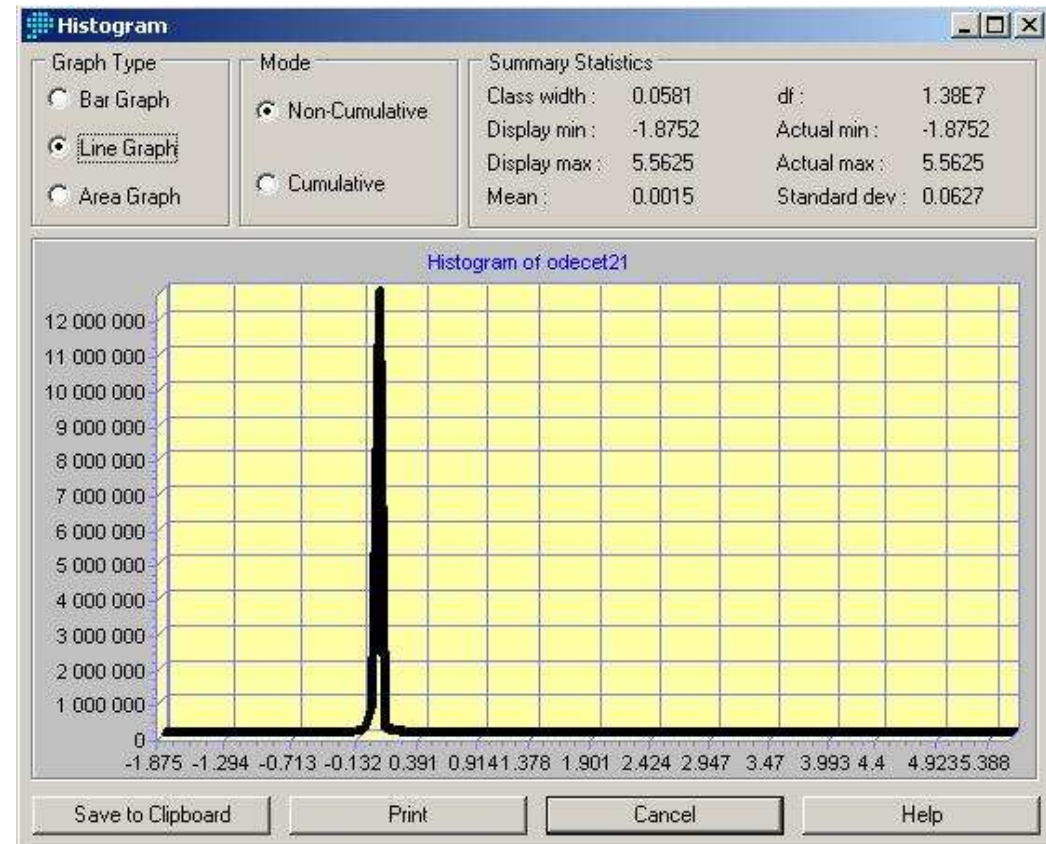
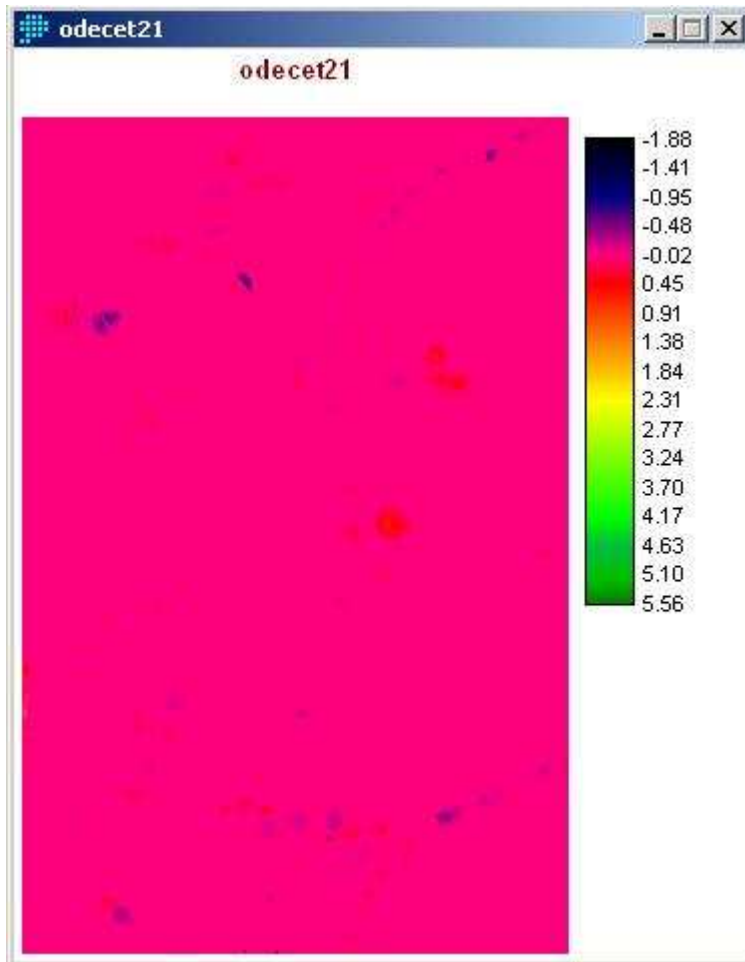




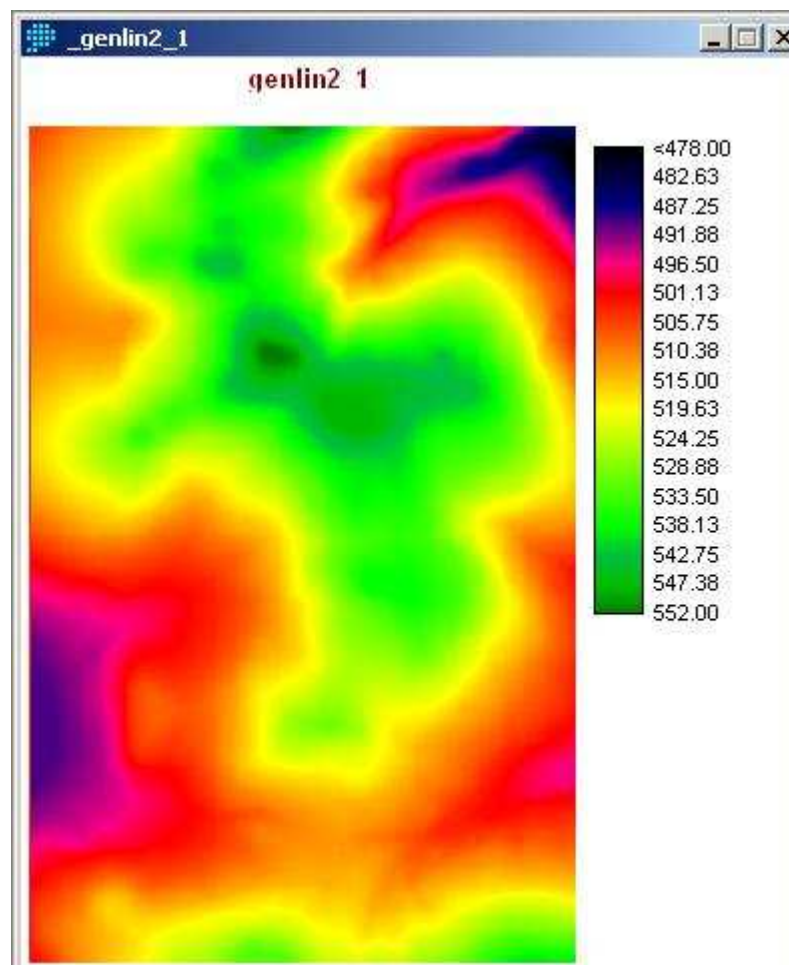
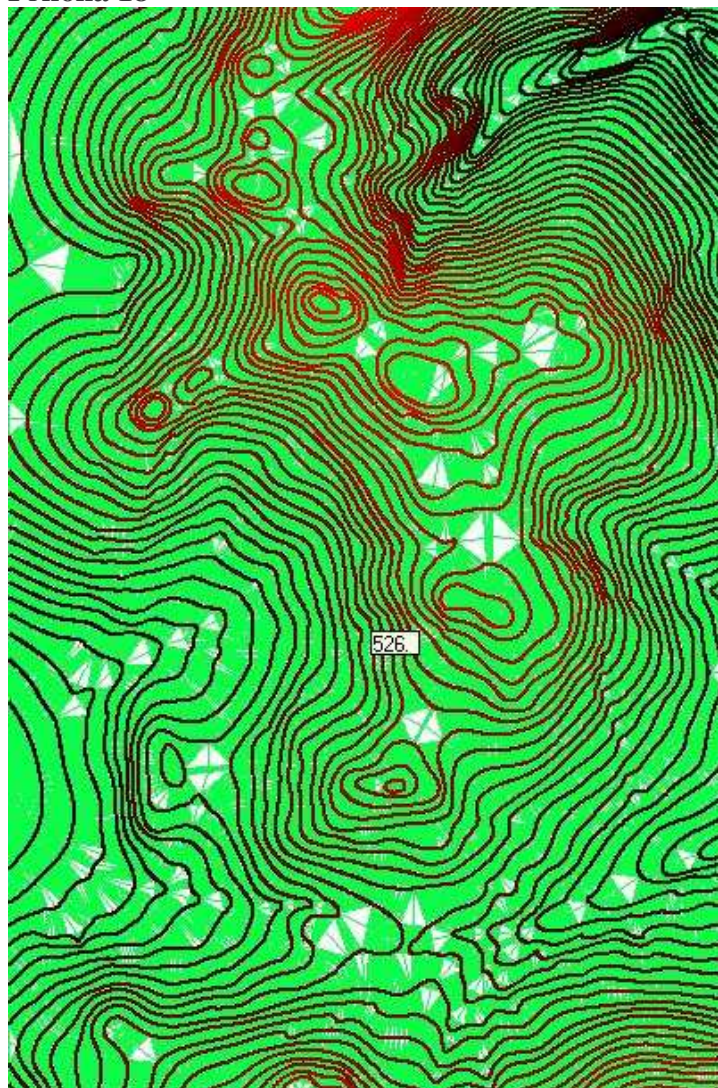


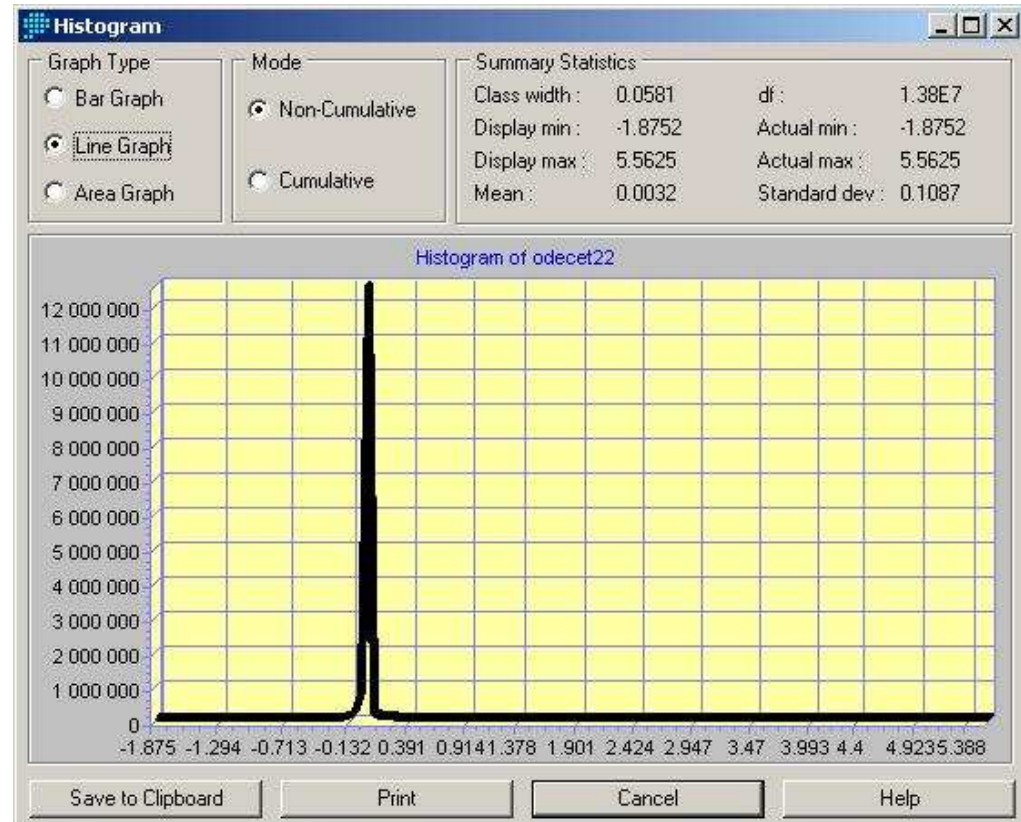
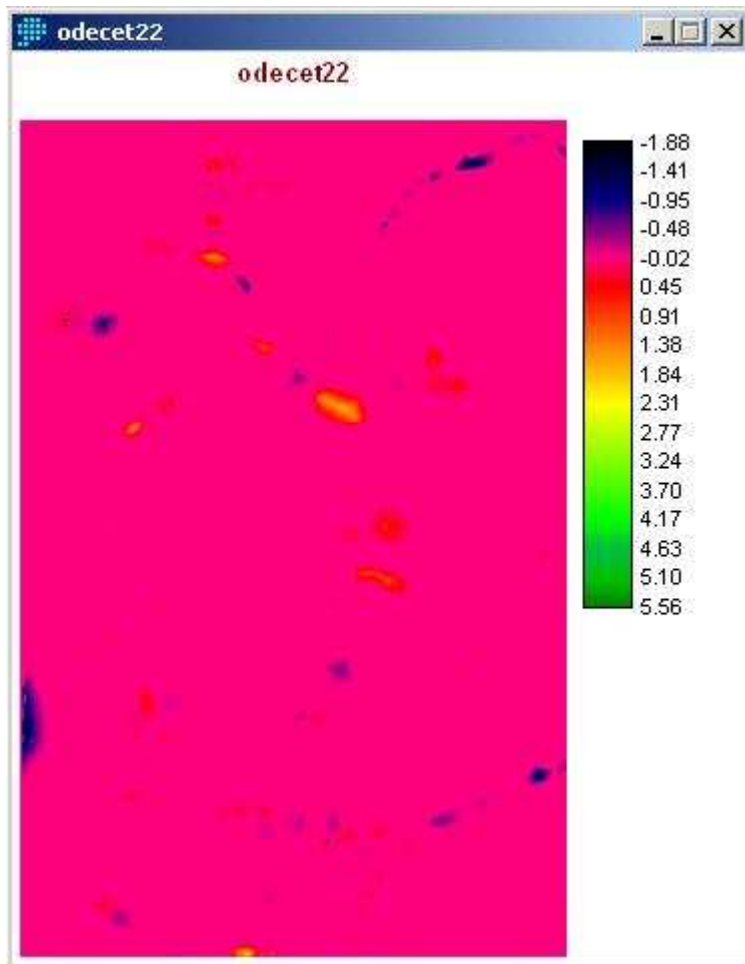
Příloha 17



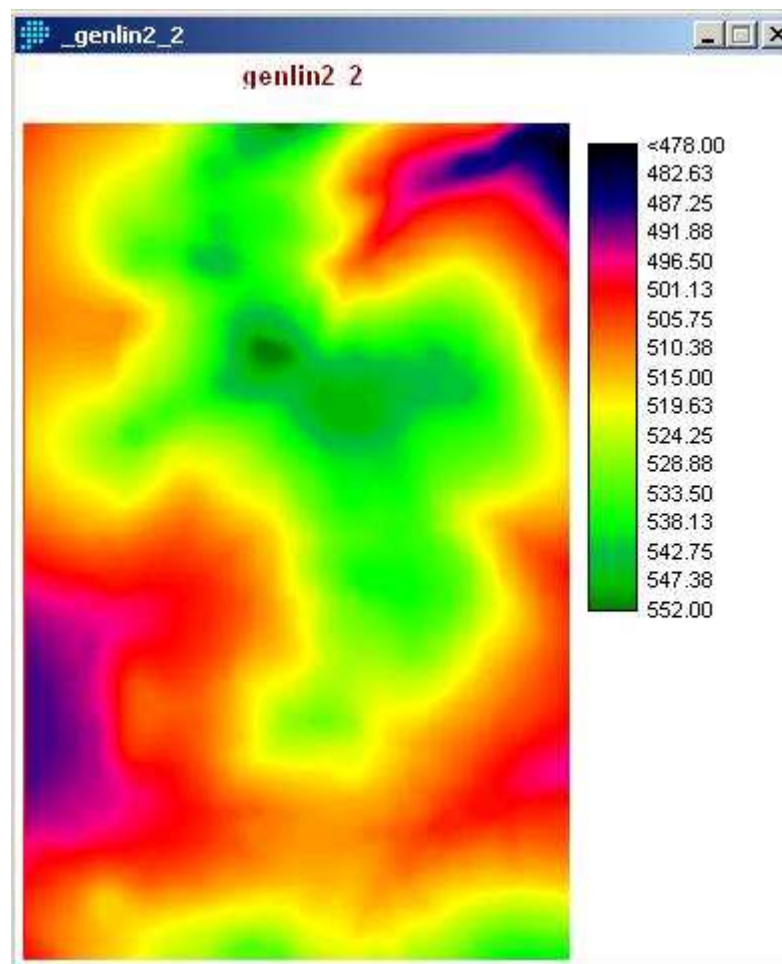
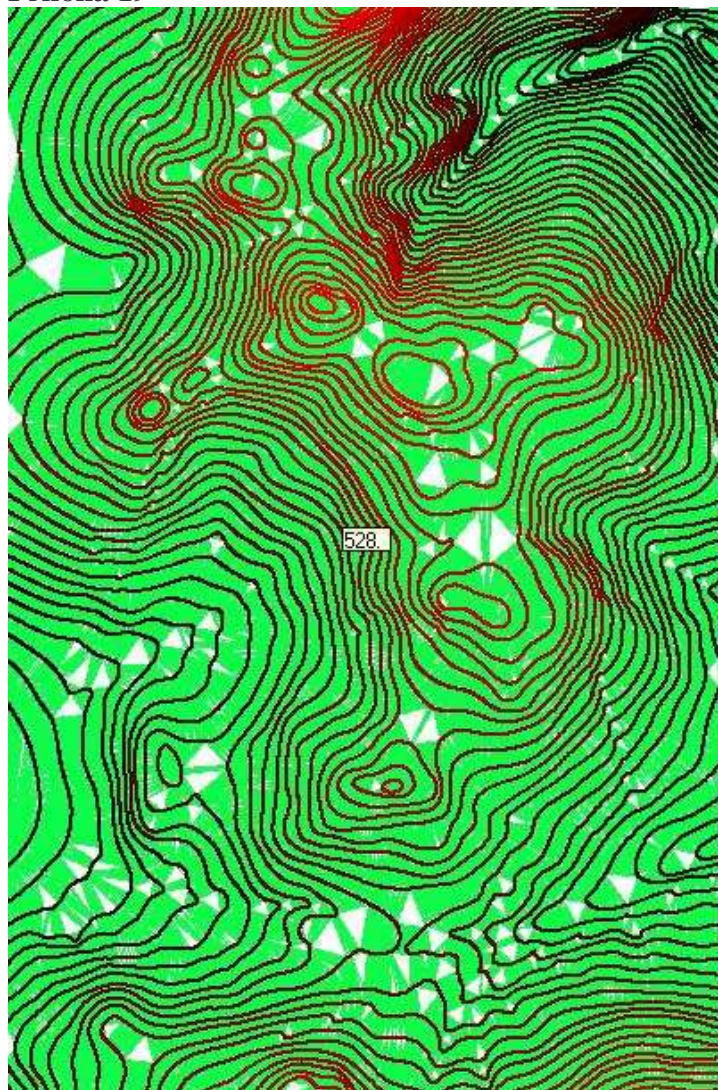


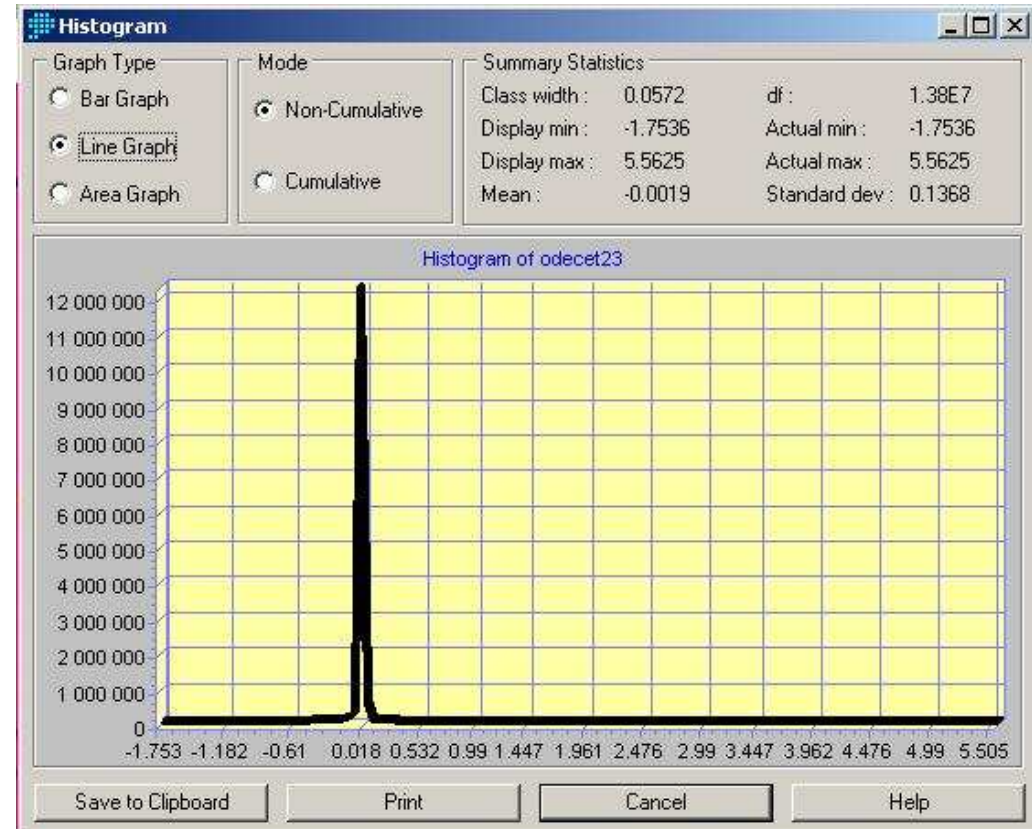
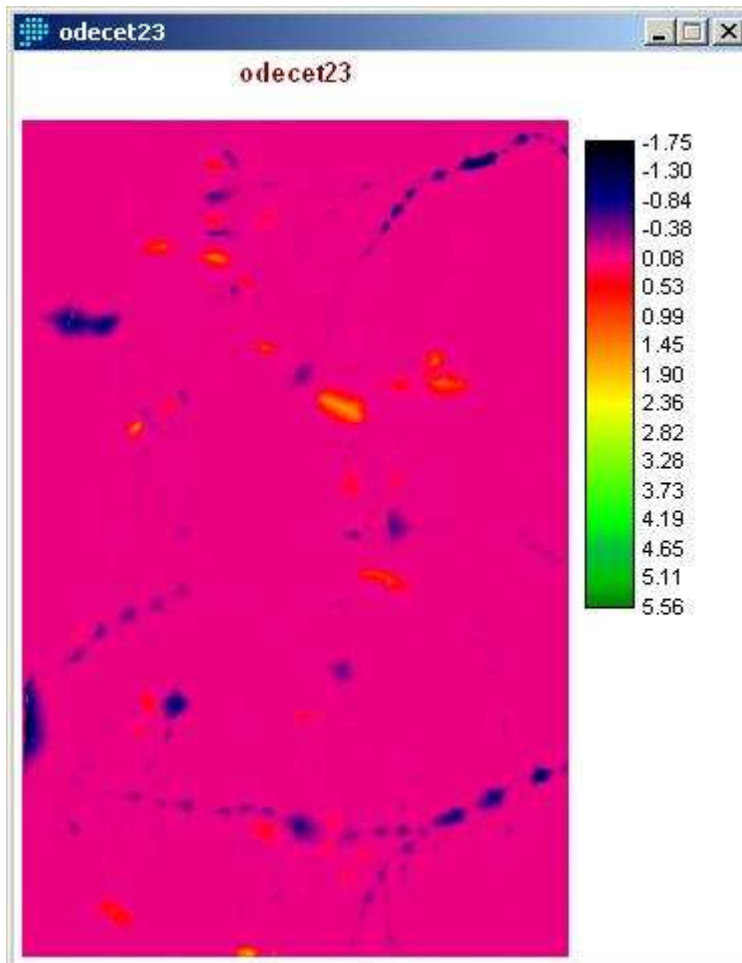
Příloha 18





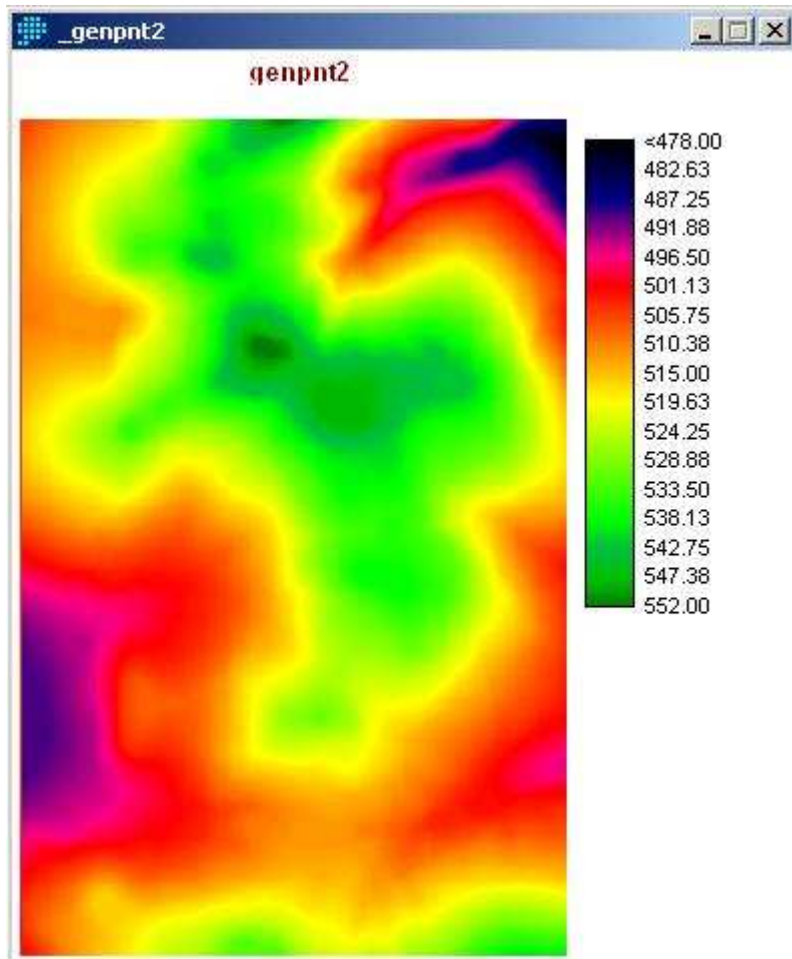
Příloha 19



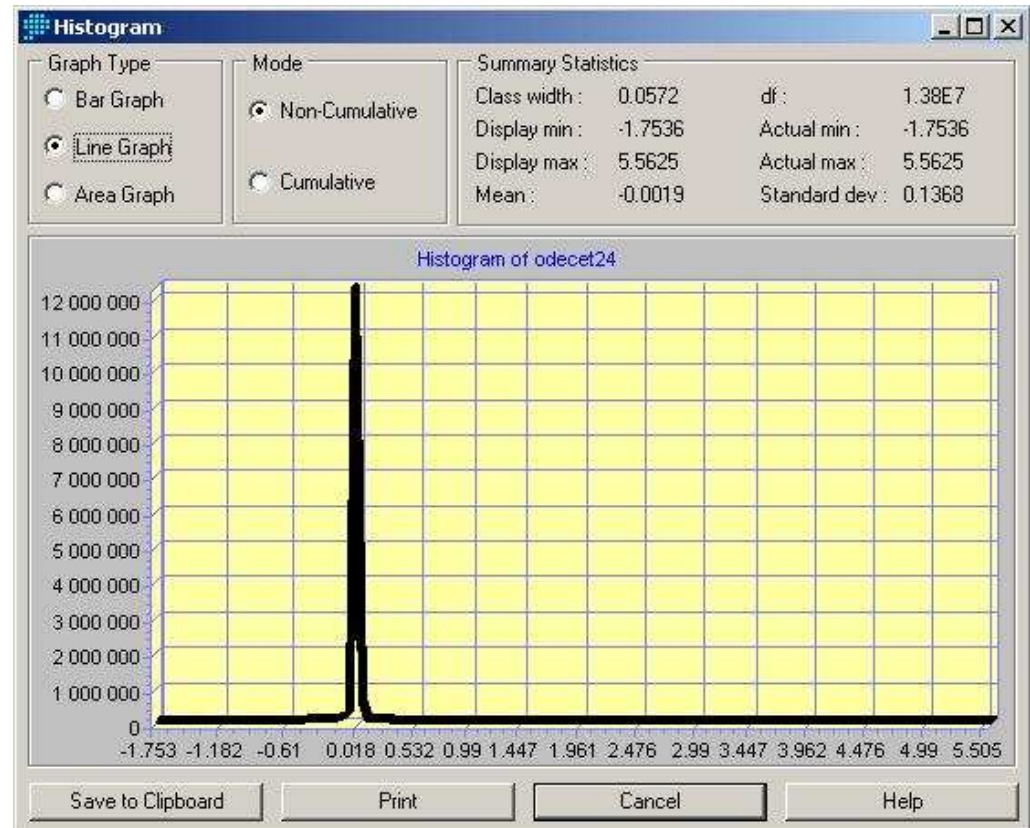
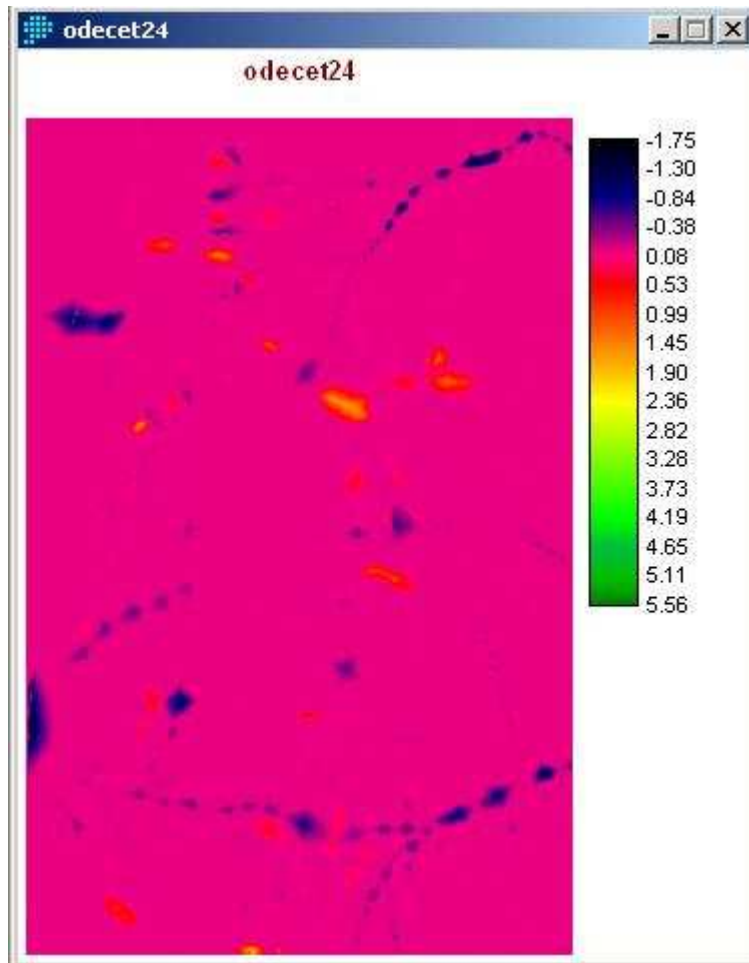


Příloha 20

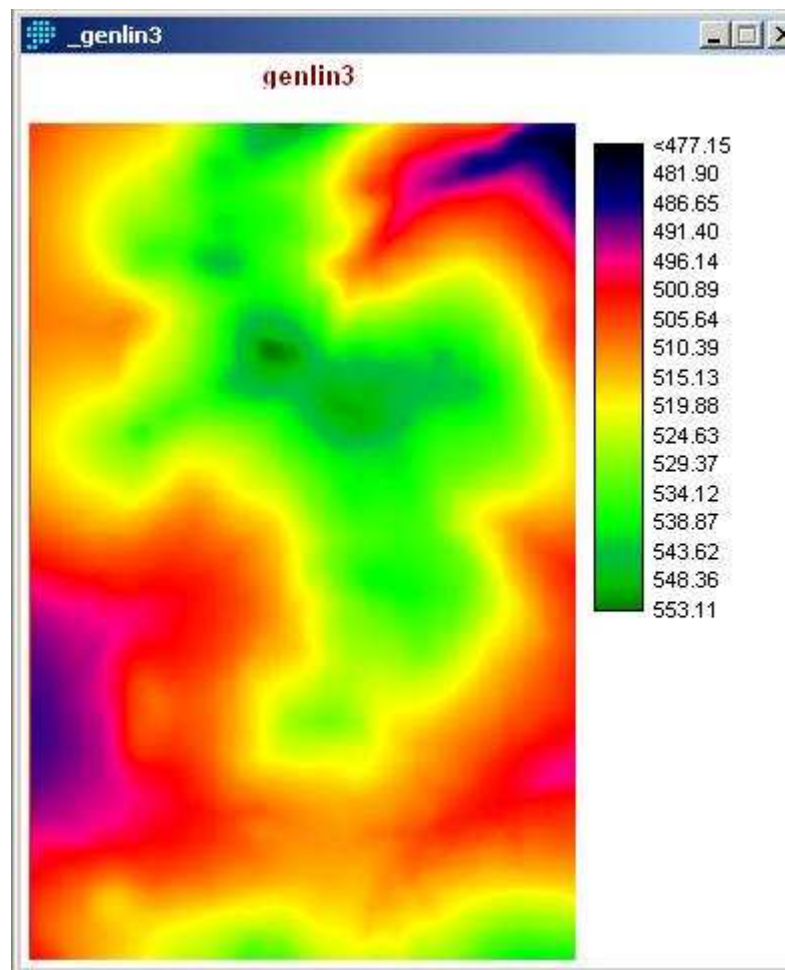


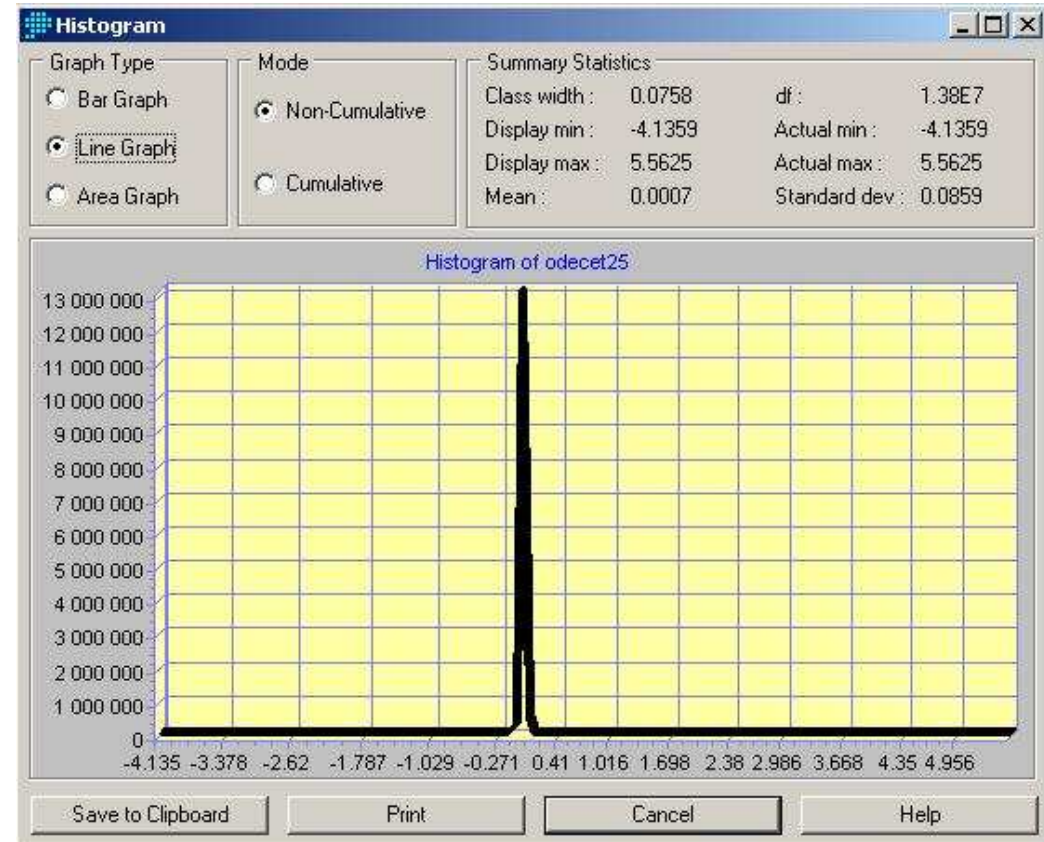
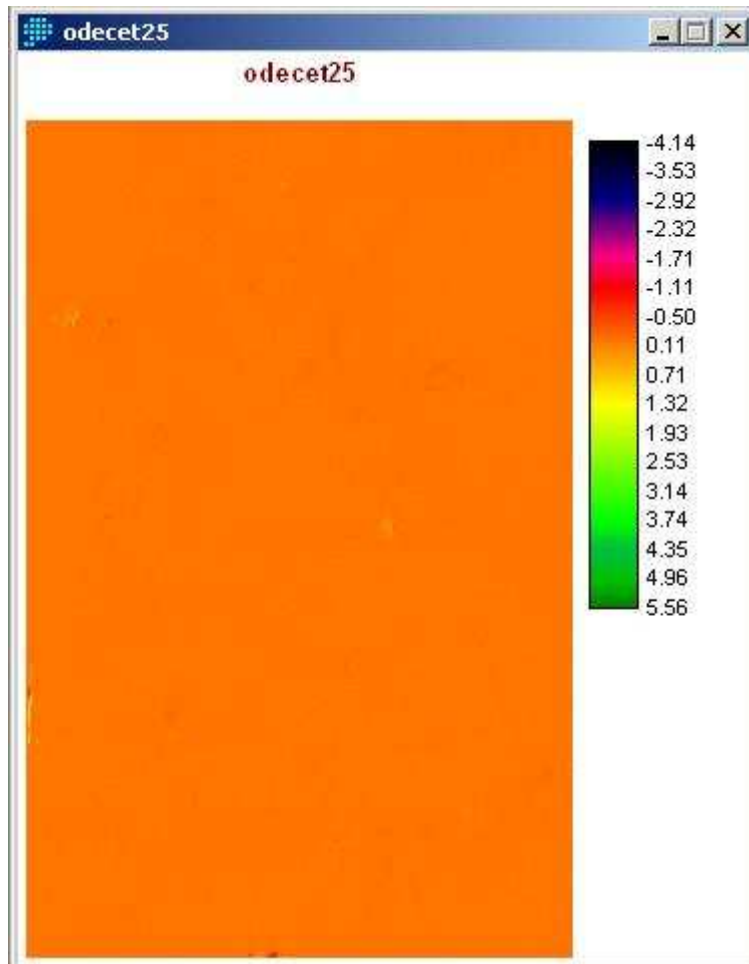




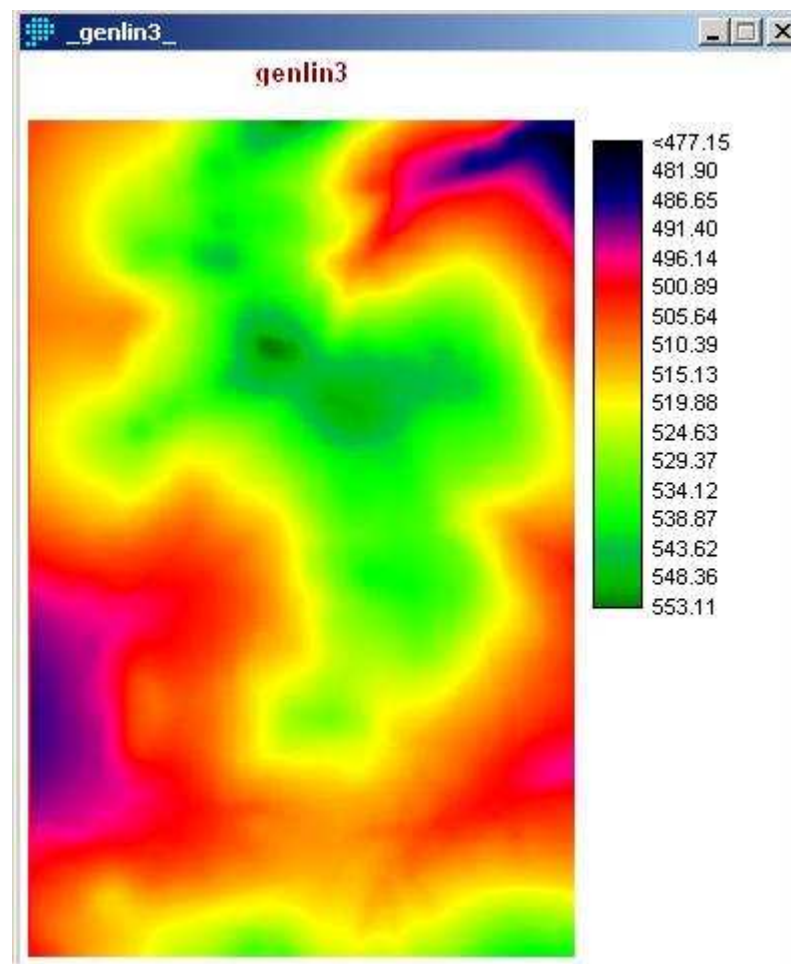


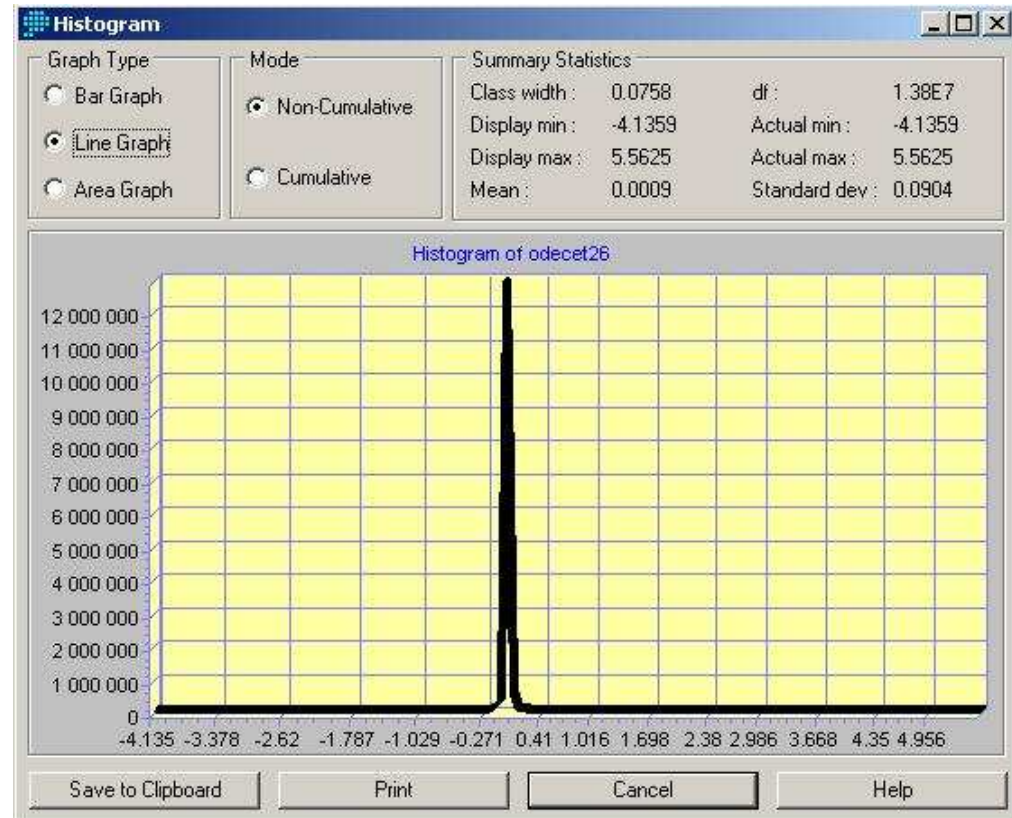
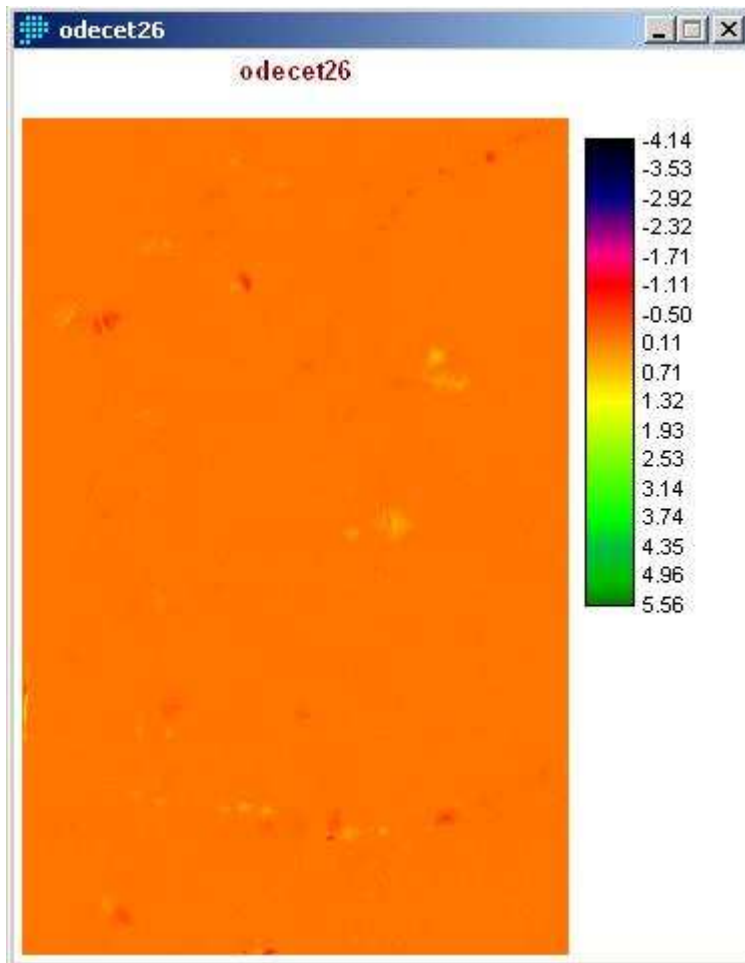
Příloha 21



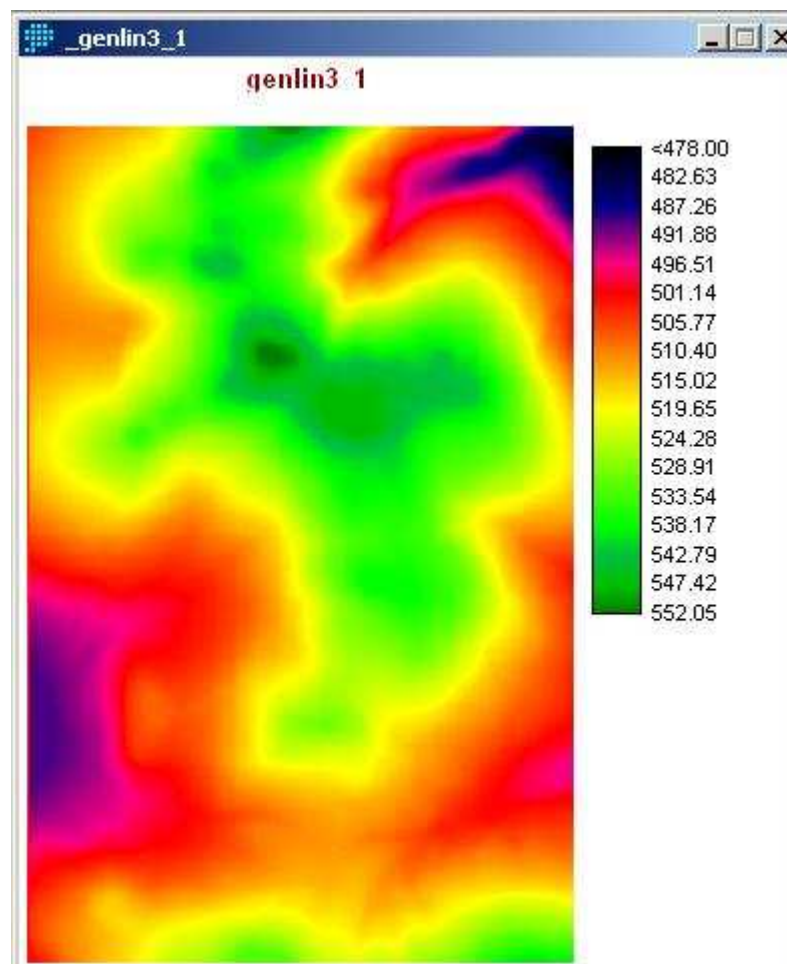


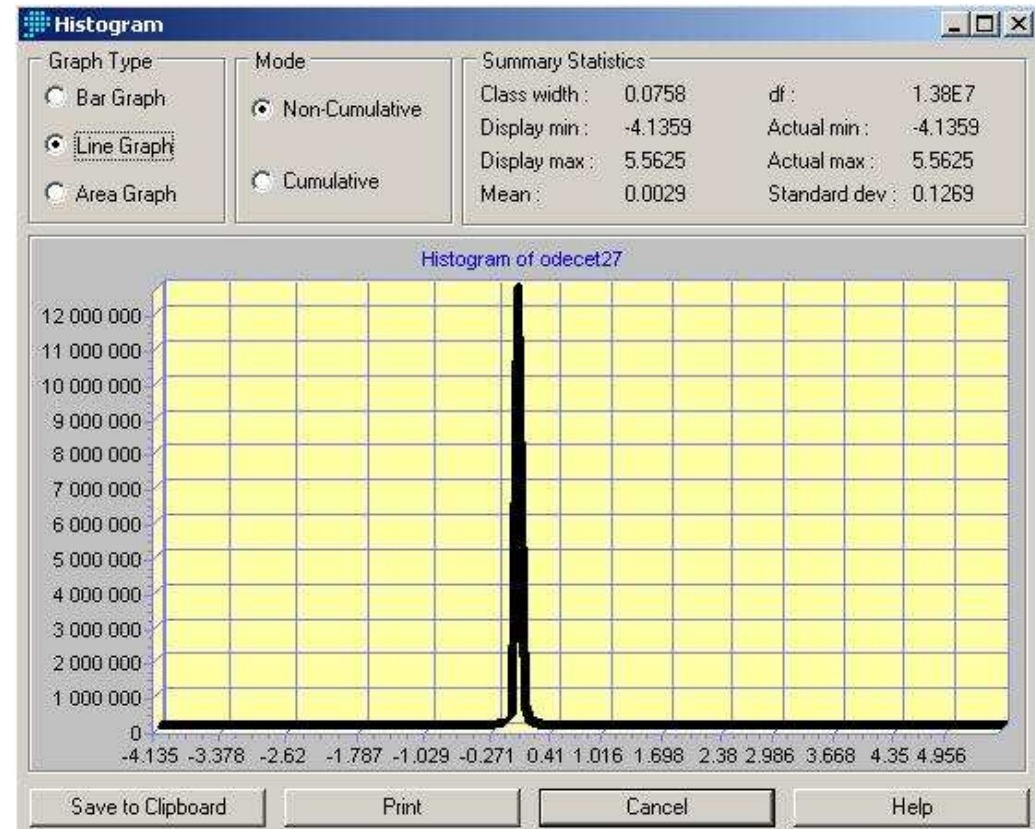
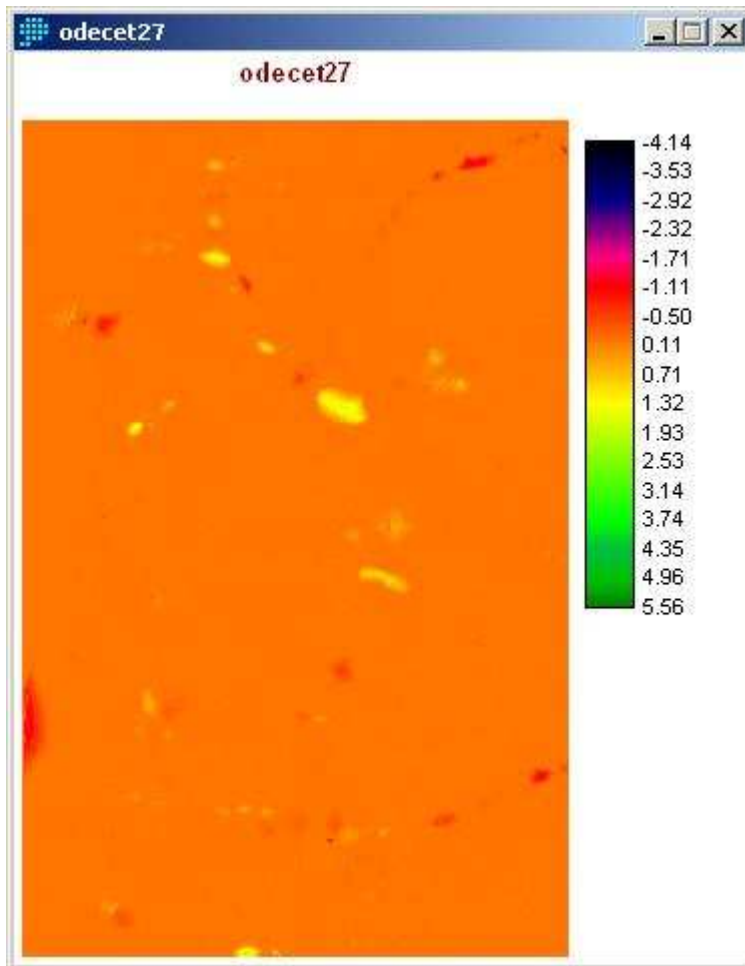
Příloha 22



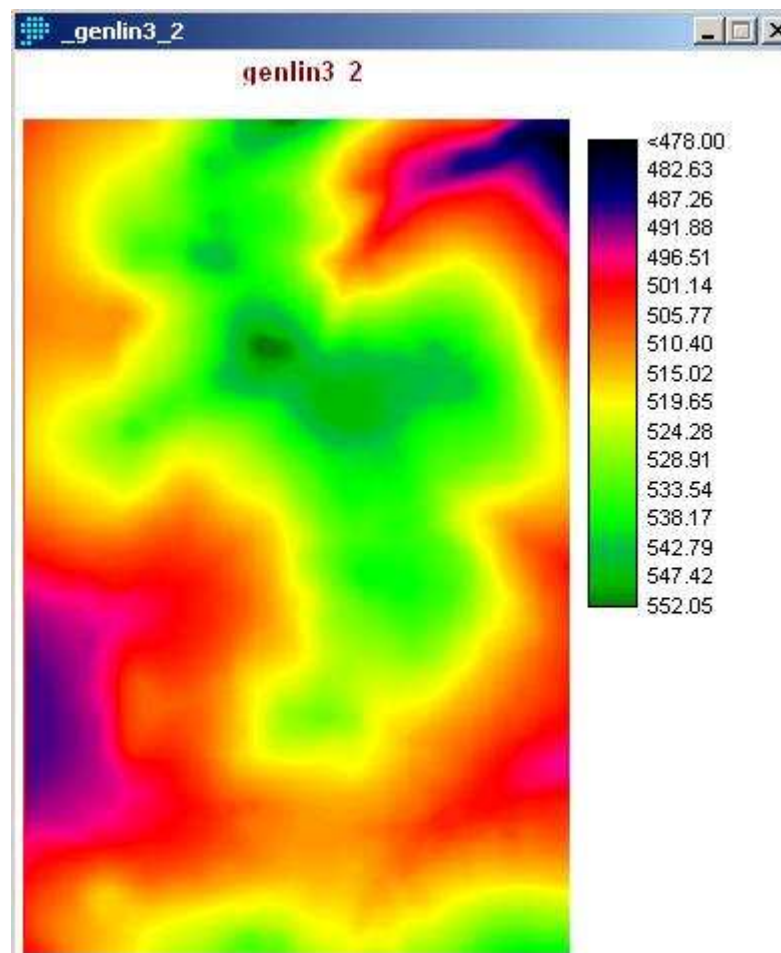


Příloha 23

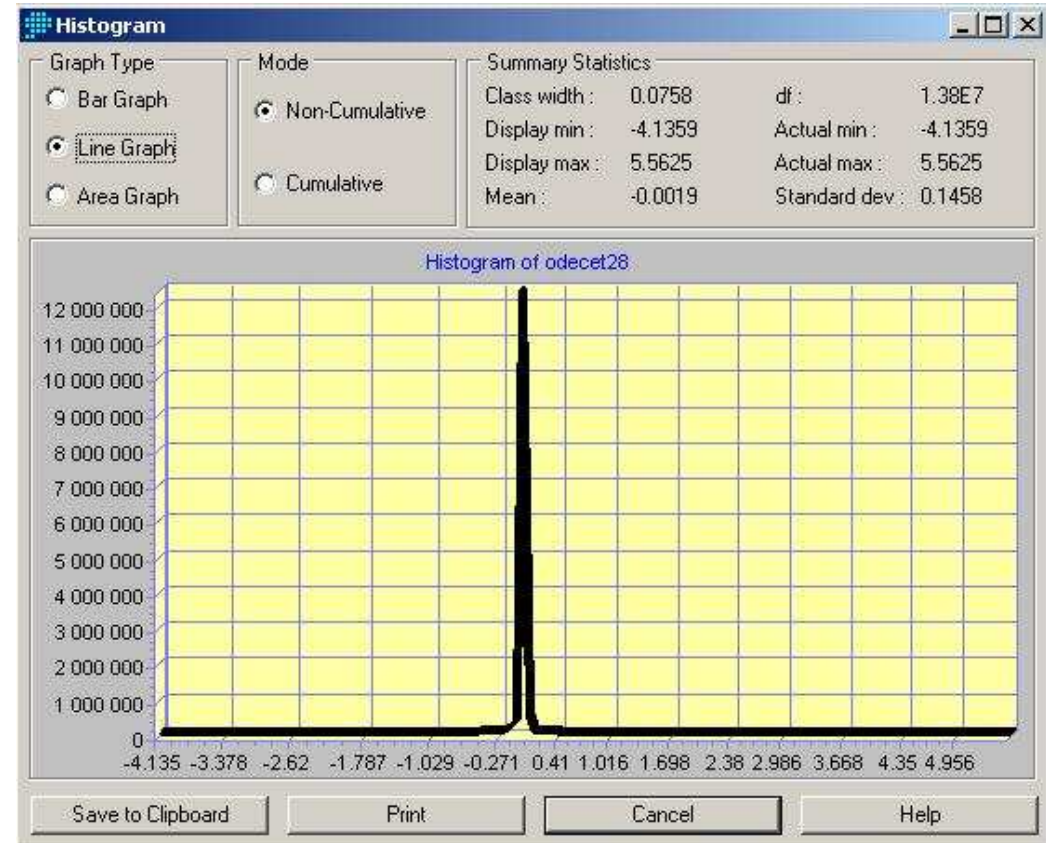
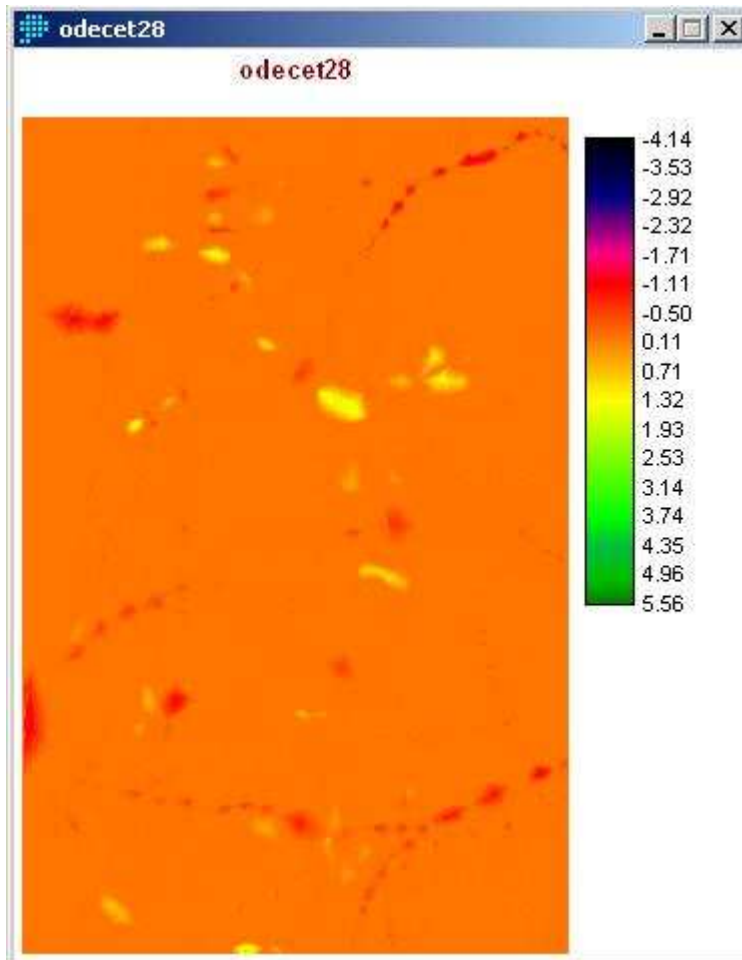




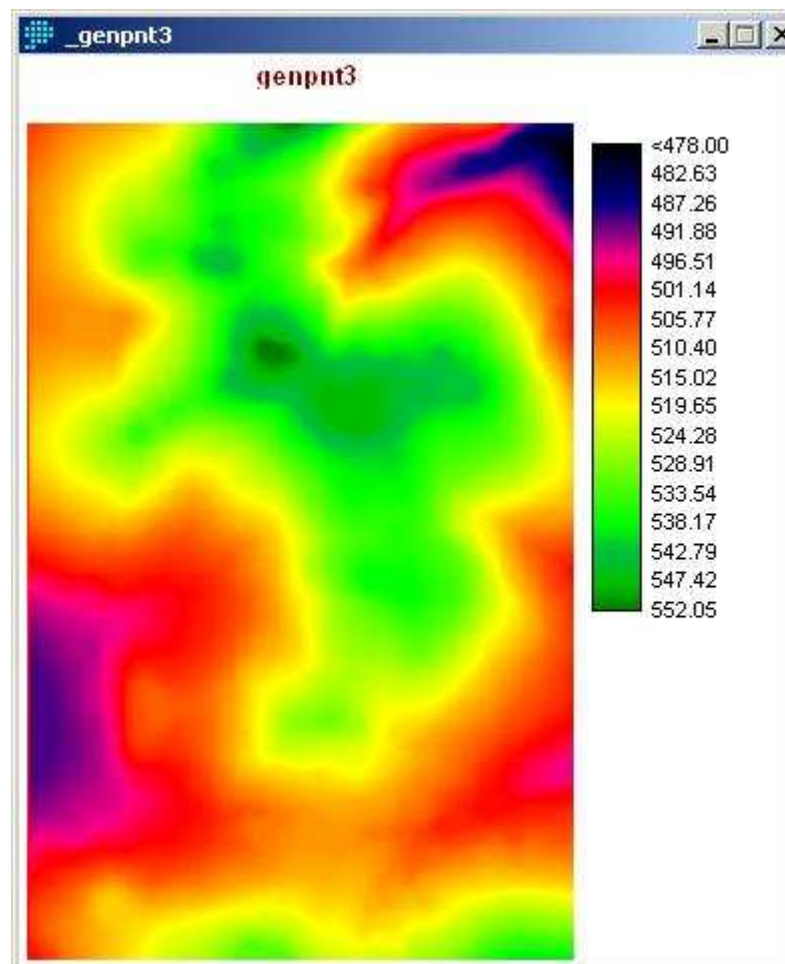
Příloha 24

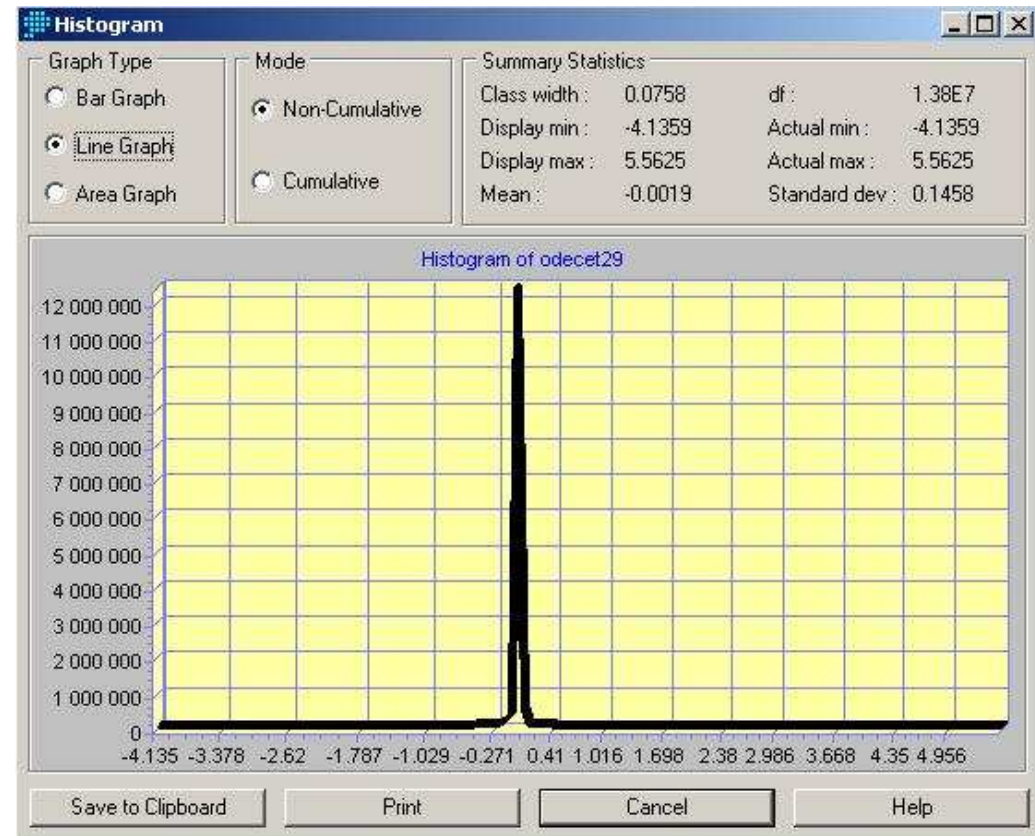
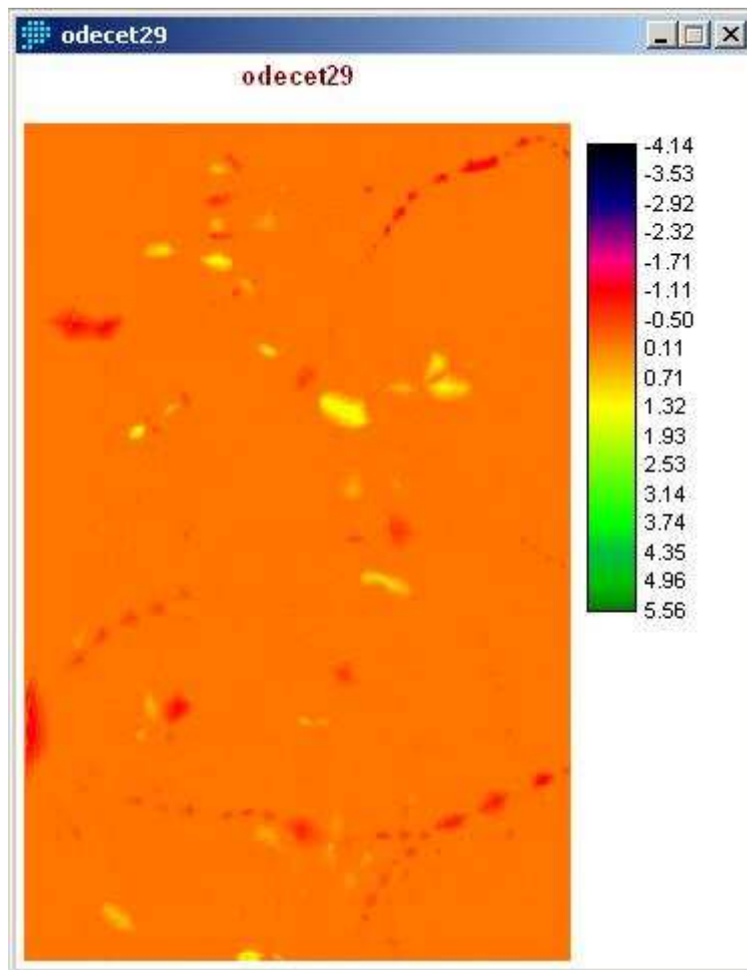






Příloha 25





## Příloha 26

Popis obsahu přiloženého CD média:

Adresář	Popis obsahu
histogramy	Obsahuje soubory JPG s histogramy odchylek jednotlivých variant rastrového DEM od referenčního souboru
porovnani	Obsahuje soubory JPG s odchylkami jednotlivých variant rastrového DEM od referenčního souboru
rastrovy_dem	Obsahuje soubory JPG s jednotlivými variantami rastrového DEM
text_dp	Obsahuje text diplomové práce ve formátu PDF
tin_a_vrstevnice	Obsahuje soubory JPG s jednotlivými variantami TIN a liniiovými DEM
vstupni_data	Obsahuje vstupní data pro otestování algoritmů

