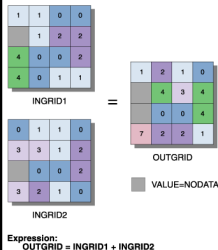


Mapová algebra

Soubor metod analýzy prostorových dat uchovávaných v rastrovém datovém modelu. Používá map jako proměnných a prostorových operací jako operátorů v algebraických výrazech.

Nová_Mapa = $f(\text{Vstupní_Mapa1}, \text{Vstupní_Mapa2}, \dots)$



Rastr je analogií matice a MA je analogií maticového počtu.

Rastr využívá modelu vrstev. Jednotlivé vrstvy (vstupní rastry, gridy) jsou pomocí operátorů, funkcí, výrazů či iterací kombinovány do výstupního rastru (většinou).

Mapová algebra – obecný princip

- MA tvoří podstatnou část prostorových analýz v GIS a je také základem řady metod zpracování obrazu v DPZ
- Jsou definovány obecné principy MA, neexistuje norma, nejednotná terminologie a nejednotná syntaxe v závislosti na implementaci v konkrétním systému.
- Řada funkčně shodných prostorových operací vystupuje pod jiným označením v různých systémech.
- Základní myšlenky formuloval Tomlin (1990).

Mapová Algebra v prostředí ArcView, ArcMap

Extenze Spatial Analyst, GRID object

GRID – podtřída třídy Object. Je to objekt, který uchovává data v rastrovém formátu. Vhodný pro prezentaci spojitě prostorové informace. Skládá se z buněk (cell).

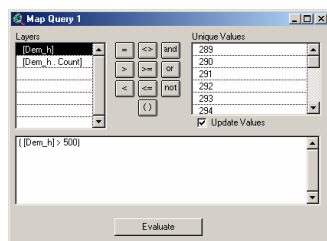
Nově vytvořený grid je soubor dočasný. Je na uživateli, zda ho uloží trvale na disk

Fyzicky je na disku Grid uložen jako složka se jménem, které zadal uživatel. K přenosu jinam je třeba mít i složku s názvem INFO.

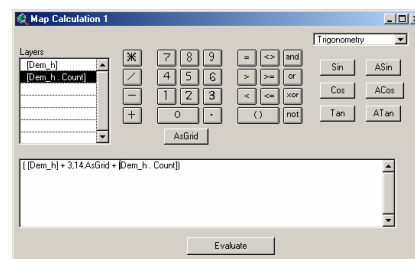
Možné formy použití MA (uživatelská rozhraní)

1. **Prostorové dotazování (Map query)** – výsledkem je tzv. bitová mapa nominálních hodnot (0 – nesplňuje, 1 – splňuje podmínku prostorového dotazu)
2. **Mapový kalkulátor** - Sestavování výrazů mapové algebry pomocí funkcí a operátorů
3. **Skriptovací jazyk AVENUE** - Sekvence výrazů, algoritmus, program
4. **Grafické modelování** - (Model Builder, PCI Modeller), nástroje využívající makrojazyka a grafických prvků vývojových diagramů

Prostorové dotazování (Map query)



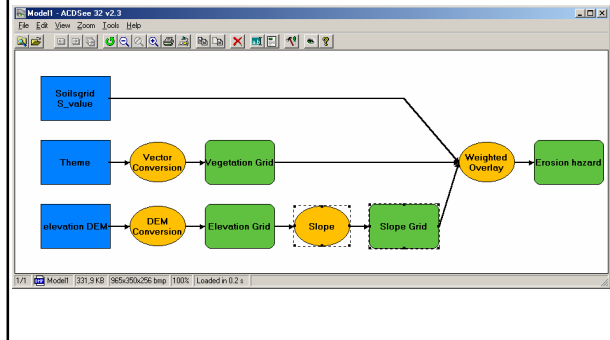
Mapový kalkulátor – základní nástroj mapové algebry



AVENUE Script

```
theView = av.GetProject.FindDoc("View1")
theSrcName = Grid.MakeSrcName( "c:\uran\grid\dem_h" )
if (theSrcName = NIL) then
  MsgBox.Error("Chybí jméno", "Chyba")
  return NIL
end
theGrid = Grid.Make(theSrcName)
theGTheme = GTheme.Make( theGrid )
theView.AddTheme( theGTheme )
theGTheme.SetVisible(true)
theView.Invalidate
```

Grafické modelování - Model Builder



Požadavky na vstupní datové soubory

I nejjednodušší model často využívá více datových vrstev majících různý původ, různou projekci, velikost buňky atd.

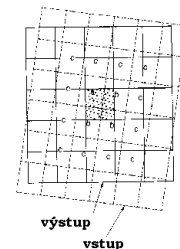
Pokud používaný modelovací nástroj neřeší sám problém unifikace vstupních dat (např. sčítat dva různé rastry s různým rozsahem či s různou velikostí buňky), je nutné ji provést předem. Tento úkol může zahrnovat dvě činnosti:

1. změnu projekce
2. převzorkování (unifikaci velikosti buňky a výpočet nových hodnot každé buňky)

Ad 1 – musí být známy převodní vztahy (transformační rovnice – např. TRANSMAP)

Ad 2 – nepoužívanější způsoby převzorkování:

- interpolace metodou nejbližšího souseda
- bilineární interpolace
- kubická konvoluce



Základní způsoby převzorkování

- a. Nejbližší sused – jediný možný způsob převzorkování použitelný na nominální (kategorická, kvalitativní) data, jeho použití na data kvantitativní nedává nejlepší výsledky. Je založen na prostém posouvání hodnot do pozice nejbližší buňky, ne na interpolaci
- b. Bilineární interpolace – hodnota nové buňky je vypočtena jako vážený průměr čtyř nejbližších buněk původních
- c. Kubická konvoluce – hodnota nové buňky je vypočtena jako vážený průměr šestnácti nejbližších buněk původních

Praktické poznámky k předzpracování dat

- Pokud systém dokáže spojit dva rozdílné rastry – má někde nastaven defaultní způsob převzorkování.
- Jak je řešen problém okrajů rastru při převzorkování?
- Je zachován typ dat? (integer vs. Floating point)
- Převzorkováním se mění statistické charakteristiky, nezachovává se minimální a maximální hodnota v gridu

Základní nástroje (objekty) pro manipulaci s prostorovými daty v mapové algebře

Operátory: aritmetické, relační, booleovské, kombinatorní, logické, akumulací, přiřazovací

Funkce: aritmetické, logické, trigonometrické, logaritmické

Třídy funkcí v mapové algebře: lokální, fokální, zonální, globální

Hlavní skupiny procedur: mapování vzdálenosti, hustoty, analýzy povrchů, hydrologické modelování, ... (aplikační procedury a funkce)

Základní nástroje (objekty) pro manipulaci s prostorovými daty v mapové algebře

Operátory: aritmetické, relační, booleovské, kombinatorní, logické, akumulací, přiřazovací

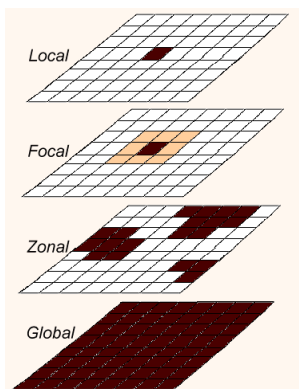
Funkce: aritmetické, logické, trigonometrické, logaritmické

Třídy funkcí v mapové algebře: lokální, fokální, zonální, globální

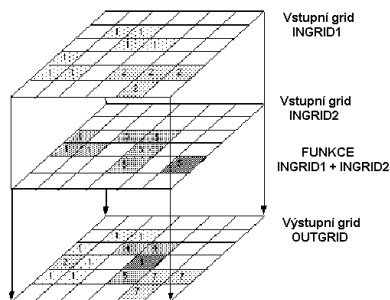
Hlavní skupiny procedur: mapování vzdálenosti, hustoty, analýzy povrchů, hydrologické modelování, ... (aplikační procedury a funkce)

Základní třídy funkcí v mapové algebře

- lokální (point)
- fokální (neighborhood)
- zonální (area)
- globální (map)



LOKÁLNÍ OPERACE



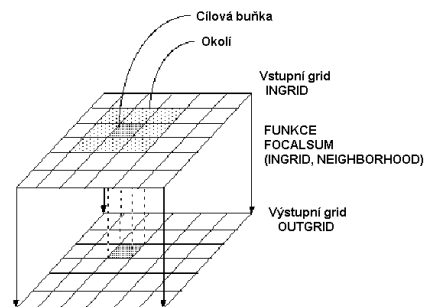
LOKÁLNÍ (BODOVÉ) OPERACE

Nová hodnota buňky je počítána pouze z jedné buňky určitou matematickou operací či jako kombinace několika odpovídajících buněk z jiných vrstev.

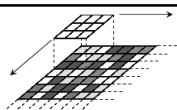
Dělení podle počtu vstupních gridů

- Unární operace (vytvoření, přidání konstanty)
- Binární operace (porovnávání)
- Operace s více vstupy (hledání min, max, ...)

FOKÁLNÍ OPEARACE



FOKÁLNÍ OPEARACE (neighborhood analysis, filtrace)

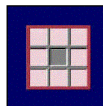


- Výstupní hodnota buňky je počítána jako jistá operace s touto buňkou a s buňkami jejího okolí.
- Velikost a tvar okolí lze definovat. Velikost je udávána v jednotkách použitého zobrazení nebo v počtu buněk (je udávána lichým číslem).
- Problém okrajů zpracovávaného gridu (zmenšení o polovinu okénka -1, replikace vstupních hodnot či replikace výsledků)
- Fokální funkce mají velký význam pro zpracování obrazu (filtrace obrazu). Použití v kartografickém modelování: úprava spojitých polí (shlazování DEM)

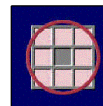
Fokální funkce

- „fokální“ statistika
- nízkofrekvenční filtrace
- vysokofrekvenční filtrace
- způsoby definování filtrovacího okna a jeho tvary
- filtrace pod maskou (podle zadaného atributu)

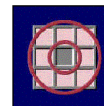
- Typy okolí:
- pravoúhelník
 - kružnice
 - mezikruží
 - výseč
 - nepravidelné okolí



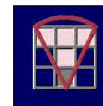
SQUARE



CIRCLE



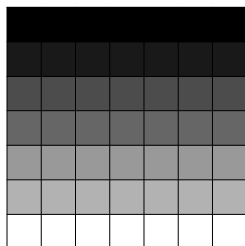
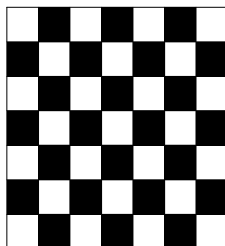
DONUT



WEDGE

Fokální funkce

- nízkofrekvenční informace
- vysokofrekvenční informace



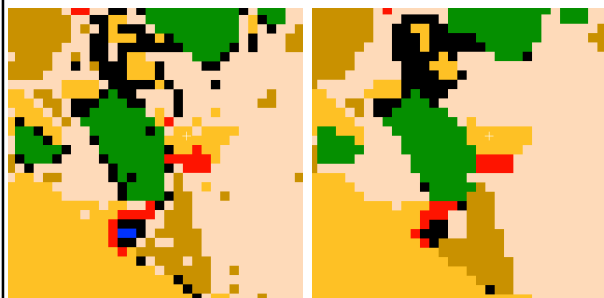
Fokální statistika - příklad

0	5	10
5	20	10
20	10	30

SUM	110
AVERAGE	12,22
MAXIMUM	30
MINIMUM	0
MEDIAN	10
MAJORITY	10
MINORITY	30
DIVERSITY	5
RANGE	30

Příklady využití fokálních funkcí:

- simulování šíření požáru na napě landuse
- odstranění „šumu“
- úprava výsledků klasifikace - potlačení malých plošek



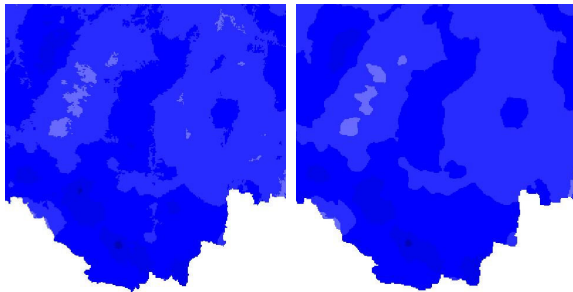
Příklady využití fokálních funkcí:

Sieve filter - „síto“

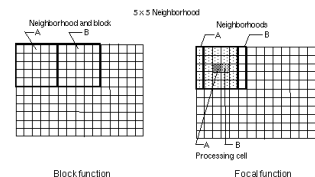


Příklady využití fokálních funkcí:

Pole srážek vytvořené interpolací a shlazené průměrovým filtrem



BLOKOVÉ OPERACE



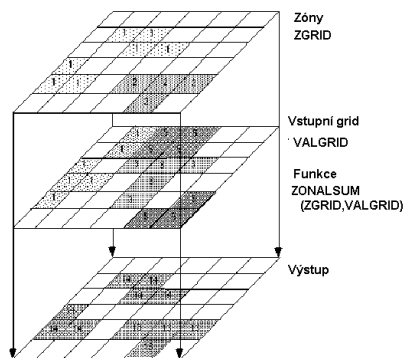
Jistou variantou fokálních funkcí jsou tzv. **blokové funkce**. Mají tyto odlišnosti:

- Na rozdíl od předchozího případu je **bloková statistika** a **blokové funkce** počítána ne pro každou buňku, ale pro blok.
- U **blokových funkcí** se okolí **nepřekrývá** ale **pokládá se na grid jako „dlaždice“**
- **Výsledná hodnota** je **přiřazena k bloku** definovanému jako **minimální pravoúhelník**, do kterého se vejde ve funkci použité okolí

ZONÁLNÍ OPERACE

- Vedle jednoho či několika vstupních gridů je v **zonálních operacích** použit také **grid**, který definuje tzv. **zóny** – plochy se stejnou hodnotou gridu.
- **Zóna** je použita na místo okolí ve **fokálních funkcích**, jinak je princip stejný.
- **Výstupní grid** je **funkcí hodnot buněk** všech vstupních gridů a také jejich **asociace s jinými buňkami** dané zóny.
- **Zónou** může být **kategorie landuse, parcela, ...**

ZONÁLNÍ OPERACE



Základní skupiny zonálních operací

- **ZONÁLNÍ STATISTIKA** - základní **průměrová statistika** (průměr, sd, max, min, diverzita, modus, medián, ...)
- **ZONÁLNÍ GEOMETRIE** - počítají se **geometrické atributy zóny** v rámci vstupního gridu - **plocha, obvod (perimetr), centroid**
- **KONTINGENČNÍ TABULKY (CROSSTABULATION)** - **maticy ukazující možné kombinace buněk** dvou vstupních gridů.

(Zóny prvního gridu tvoří řádky matice, zóny druhého vstupního gridu potom sloupce matice.
Např. – dvě mapy landuse z různého období - matice ukazuje, která kategorie se změnila na jinou kategorii, nebo - landuse a nadmořská výška - zastoupení tříd landuse v určitých výškových stupních).

Podle použité metody může být **výstupem zonálních funkcí**:

1. **tabulka**
2. **Histogram (pro každou zónu)**

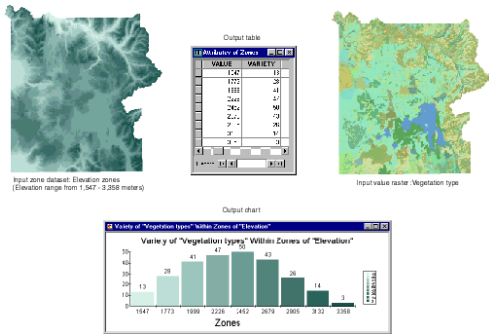
Ekvivalenty v menu Spatial Analyst:

- Tabulate Areas
- Histogram by Zones

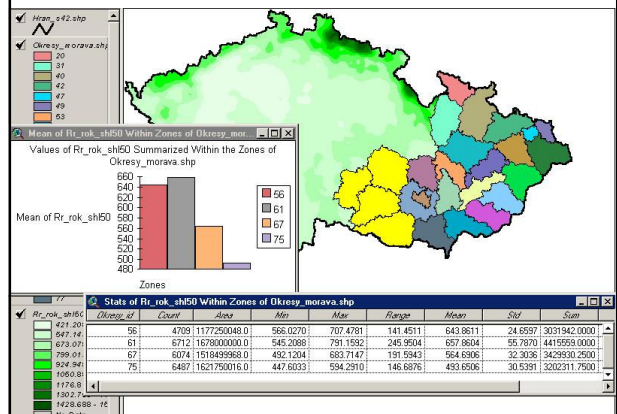
Příklady použití zonálních funkcí:

- **Výpočet ploch jednotlivých kategorií landuse v rámci každého povodí (= zóny)**
- **Identifikování ohrožených (atd.) parcel (= zón) v rámci zpracovaného území**
- **Vyplnění lokálních depresí („bezodtokých oblastí“) ve spojitěm povrchu (např. DEM)**

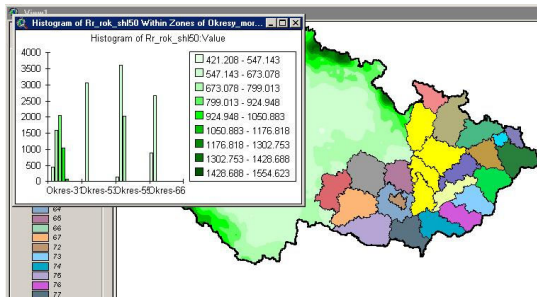
Příklad využití zonálních funkcí:



Zonální statistika



Zonální statistika



Zastoupení tříd úhrnů srážek ve vybraných okresech

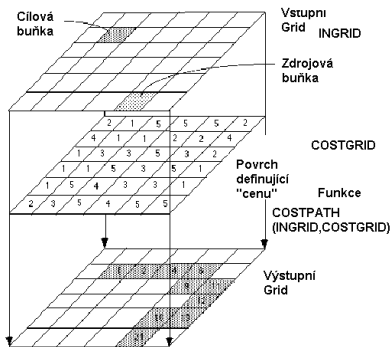
GLOBALNÍ OPERACE

Nová mapa vznikne transformací, jejíž rovnice byla sestavena na základě hodnot všech buněk rastru (např. analýza vzdálenosti, polynomičká transformace, definování obalových zón, Thiessenovy polygony, ...)

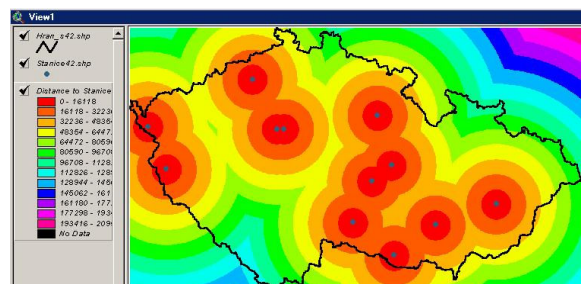
Oblasti využití

- Hledání optimální trasy
- Morfometrická analýza
- Hydrologické modelování

GLOBALNÍ OPERACE



Vzdálenost od vybraných meteorologických stanic



Mapová algebra

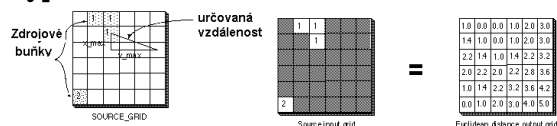
Hlavní skupiny procedur (aplikačních)

- Mapování vzdálenosti
- Mapování hustoty
- Optimalizační úlohy
- Morfometrické analýzy povrchů
- Hydrologické modelování
- Metody interpolace
- Překryvné operace
- (Re)klasifikační úlohy

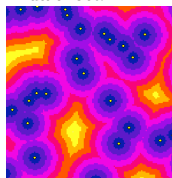
Mapování vzdálenosti

- Určení euklidovské vzdálenosti
- Určení vážené vzdálenosti (cost distance, frikční povrchy, bariéry)
- Nalezení optimální trasy

Výpočet euklidovské vzdálenosti:



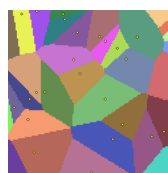
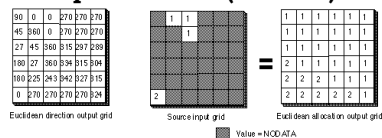
Na vstupu je grid, který obsahuje jednu nebo několik buněk nenulových – tzv. zdrojové buňky, od kterých se počítá např. vzdálenost.



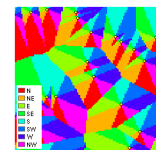
Vedle vzdálenosti lze získat i rastr, který představuje směr (direction) a příslušnost k určité zdrojové buňce (allocation).

Předstupeň pro definování obalových zón (buffer)

Určování příslušnosti (alokace)



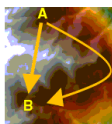
Thiessenovy polygony



Určování směru od zdrojové buňky

Slouží ke konstrukci oblastí, ze kterých je k dané zdrojové buňce či buňkám nejbližší

Funkce pro výpočet vážených vzdáleností (weighted distance, cost distance)

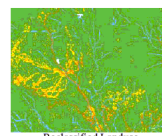


Najde nejmenší akumulovanou „cenu“ z každé buňky k buňce (buňkám) zdrojovým a to na povrchu (costGrid), který představuje „cenu“ nutnou pro překonání dané buňky,

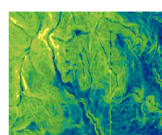
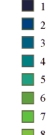
Vedle vážené vzdálenosti lze získat i Grid, který představuje směr (direction) a příslušnost k určité zdrojové buňce (allocation).

Vytvoření COST povrchu

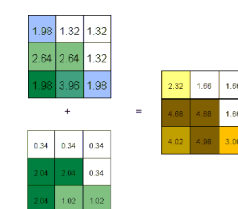
1)

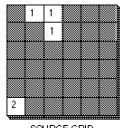


2)



3)

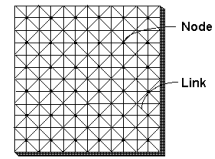




White Cells Value = 0 Value = NODATA

Na vstupu je zdrojový Grid (vlevo), který nese zdrojové buňky (nemají NoData) a costGrid (vpravo) – povrch, který v hodnotách buněk nese hodnoty, které představují „cenu“, kterou je zapotřebí vynaložit při cestě z jedné buňky do druhé.

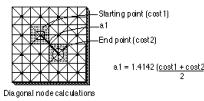
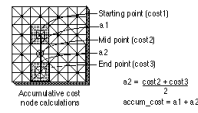
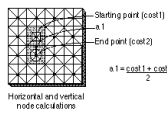
Tyto „náklady“ se počítají a ve výstupním gridu je v buňkách akumulovaná cena k nejbližší zdrojové buňce.



Node – střed buňky

Link – spojnice středů dvou sousedních buněk (nódů).

Každé spojnici přísluší určitá impedance (tření).



Výpočet impedance:

$$a1 = cost1 + cost2 / 2$$

pro vertikální či horizontální pohyb, a nebo

$$a1 = 1.414216 * (cost1 + cost2) / 2$$

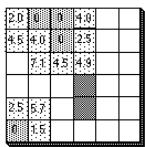
pro diagonální pohyb.

Algoritmus výpočtu vážené vzdálenosti (COST DISTANCE)

Iterační výpočet:

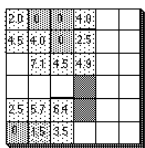
1. Nejprve se ve výstupním gridu přiřadí zdrojovým buňkám hodnota nula.
2. Potom se podle výše uvedených vzorců vypočte cena za pohyb ze zdrojové buňky do všech buněk sousedních.
3. Najde se buňka s nejnižší „cenou“ a pro tuto se opět vyhodnotí sousední buňky z cost gridu.

Vytváření povrchu nejmenší akumulované „ceny“



Active accumulative cost cell list				
1.5	2.0	2.5	2.5	4.0
4.0	4.0	4.5	4.5	4.5
4.9	5.7	7.1		

Value = NODATA Cells on active cost list Source cell Allocated cells to cost distance



Active accumulative cost cell list				
1.5	2.0	2.5	2.5	4.0
4.0	4.0	4.5	4.5	4.5
4.9	5.7	7.1		

Value = NODATA Cells on active cost list Source cell Allocated cells to cost distance New neighborhood cells to be added to active list

Vyhledání optimální trasy – výsledný soubor akumulované „nejmenší ceny“

2.0	0	0	4.0	6.7	9.2
4.5	4.0	0	2.5	7.5	13.5
8.0	7.1	4.5	4.9	8.9	14.5
5.0	7.5	10.5		10.6	9.2
2.5	5.7	6.4		7.1	11.1
0	1.5	3.5	5.0	7.0	10.5

Cost distance output raster

Vyhledání optimální trasy – pomocné soubory

6	7	8
5	0	1
4	3	2

Back-link positions

1	0	0	5	4	5
7	1	0	5	5	6
3	8	7	6	6	3
3	5	7	3	4	
3	4	4	4	5	
0	5	5	5	5	

Back-link output grid

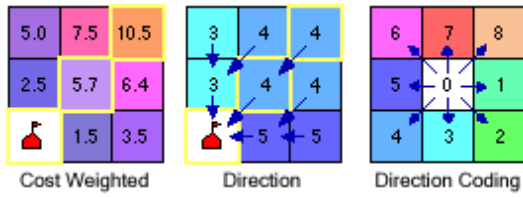
1	1	1	1	1	1
1	1	1	1	1	1
2	1	1	1	1	2
2	2	1			2
2	2	2			2
2	2	2			2

Cost allocated on output grid

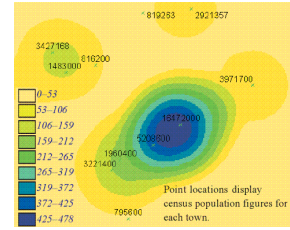
Rekonstrukce cesty ke zdrojové buňce (buňkám)

Příslušnost ke zdrojové buňce (buňkám)

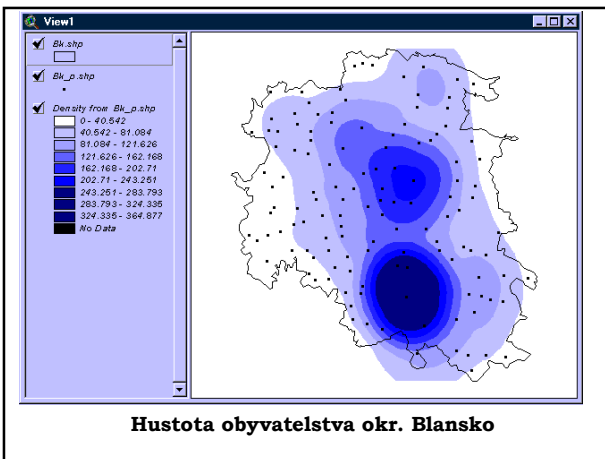
Vyhledání optimální trasy.



Mapování hustoty



- Z bodových měření je vytvářen povrch mapující hustotu jevu
- Pro každou buňku výsledného rastru je definováno kruhové okolí
- Hustota je počítána jako podíl sumy hodnot všech bodů uvnitř kruhu a plochy kruhu
- Bodům blízkým je možné dát větší váhu jak bodům vzdáleným – výsledkem je potom více shladený povrch mapující hustotou jevu



Optimalizační úlohy – základní rozdělení

Nalezení nejvhodnější trasy

- libovolné cesty,
- nejkratší cesty (tj. cesty s nejmenším počtem buněk)
- nejlevnější cesty (tj. cesty s nejmenším součtem ohodnocení buněk)
- všech cest splňujících zadaná kritéria
 - mezi dvojicí zadaných buněk
 - ze zadané buňky do všech ostatních
 - ze všech ostatních do zadané koncové buňky
 - mezi všemi (uspořádanými) dvojicemi buněk
 - optimální spojení zadané množiny buněk

Optimální lokalizace objektu