

UNIVERZITA KARLOVA
PŘÍRODOVĚDECKÁ FAKULTA



ALGORITMY POČÍTAČOVÉ KARTOGRAFIE
Digitální modely terénu

Jáchym Černík, Monika Novotná

1 Zadání

Vstup: množina $P = \{p_1, \dots, p_n\}$, $p_i = \{x_i, y_i, z_i\}$.

Výstup: polyedrický DMT nad množinou P představovaný vrstevnicemi doplněný vizualizací sklonu trojúhelníků a jejich expozicí.

Metodou inkrementální konstrukce vytvořte nad množinou P vstupních bodů 2D Delaunay triangulaci. Jako vstupní data použijte existující geodetická data (alespoň 300 bodů) popř. navrhněte algoritmus pro generování syntetických vstupních dat představujících významné terénní tvary (kupa, údolí, spočinek, hřbet, ...).

Vstupní množiny bodů včetně níže uvedených výstupů vhodně vizualizujte. Grafické rozhraní realizujte s využitím frameworku QT. Dynamické datové struktury implementujte s využitím STL. Nad takto vzniklou triangulací vygenerujte polyedrický digitální model terénu. Dále proveďte tyto analýzy:

- S využitím lineární interpolace vygenerujte vrstevnice se zadaným krokem a v zadaném intervalu, proveďte jejich vizualizaci s rozlišením zvýrazněných vrstevnic.
- Analyzujte sklon digitálního modelu terénu, jednotlivé trojúhelníky vizualizujte v závislosti na jejich sklonu.
- Analyzujte expozici digitálního modelu terénu, jednotlivé trojúhelníky vizualizujte v závislosti na jejich expozici ke světové straně.

Zhodnoťte výsledný digitální model terénu z kartografického hlediska, zamyslete se nad slabinami algoritmu založeného na 2D Delaunay triangulaci. Ve kterých situacích (různé terénní tvary) nebude dávat vhodné výsledky? Tyto situace graficky znázorněte.

Zhodnocení činnosti algoritmu včetně ukázek proveďte alespoň na **3 strany** formátu A4. V

rámci úlohy nebyly vypracovány žádné bonusové úkoly

Hodnocení:

Krok	Hodnocení
Delaunay triangulace, polyedrický model terénu.	10b
Konstrukce vrstevnic, analýza sklonu a expozice.	10b
Triangulace nekonvexní oblasti zadané polygonem.	+5b
Výběr barevných stupnic při vizualizaci sklonu a expozice.	+3b
Automatický popis vrstevnic.	+3b
Automatický popis vrstevnic respektující kartografické zásady (expozice, vhodné rozložení).	+10b
Algoritmus pro automatické generování terénních tvarů (kupa, údolí, spočinek, hřbet, ...).	+10b
3D vizualizace terénu s využitím promítání.	+10b
Barevná hypsometrie.	+5b
Max celkem:	65b

2 Popis a rozbor problému

Digitální model terénu (DMT) představuje matematickou interpretaci zemského povrchu, která zahrnuje i konkrétní terénní útvary, jako jsou pahorky, údolí a další topografické elementy. Zobrazení DMT je náročným procesem, který zahrnuje snahu o co nejlepší digitální reprezentaci terénu. Heterogenost zemského povrchu tomuto procesu přitěžuje svojí komplexitou, a autor DMT tak musí vyvažovat mezi komplexitou reprezentace, množstvím detailu a dostupnou výpočetní kapacitou. K vytvoření DMT se využívají podkladová data sbíraná prostřednictvím technologie LiDAR či fotogrammetrie. Značný objem těchto datových bodů poté slouží jako základ pro konstrukci vlastního modelu, u kterého lze pro každý bod určit jeho nadmořskou výšku. Spojením s dalšími zdroji, kupříkladu satelitními záběry nebo leteckými snímky, lze docílit vyšší přesnosti výsledných modelů a map.

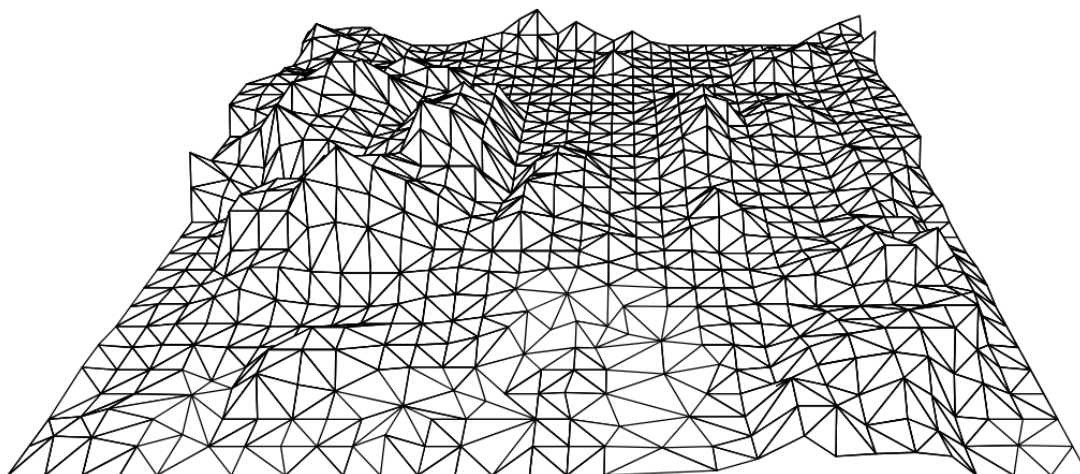
Z pravidla jsou typy DMT děleny na jejich způsob reprezentace (základní ploška):

- Trojúhelníkový typ terénu; Polyedrický model
 - *Plošky jsou představovány nepravidelnými trojúhelníky, které mají společně nejvýše společnou hranu*
- Rastrový model terénu
 - *Neboli grid, je tvořen pravidelnými čtyřúhelníky (buněkami) s pravidelným rozestupem*
- Plátový model terénu
 - *Složitější, povrch je sítí vzájemně propojených nepravidelných trojúhelníků, které sdílejí společné hrany a zachovávají původní výškové body.*

Cílem této úlohy je vytvořit Trojúhelníkový Polyedrický model s využitím 2D Delaunay triangulace a po sléze provést analýzu DMT

2.1 Triangulace; Polyedrický model

Triangulace, v rámci DMT, fundamentálním procesem tvorby trojúhelníkového typu terénu, který transformuje diskrétní množinu bodů $P = \{p_1, \dots, p_n\}$, kde každý bod p_i je korespondujícím souřadnicemi (x_i, y_i, z_i) , na spojitou reprezentaci povrchu. Tento proces spočívá v rozdělení rovinného průmětu dané oblasti na soustavu nepřekrývajících se trojúhelníků, jejichž vrcholy jsou právě body z množiny P . Výsledná struktura se označuje jako triangulovaná nepravidelná síť (TIN)(Obrázek 1). TIN efektivně zachycuje topologii povrchu, přičemž hustota trojúhelníků se může adaptivně měnit v závislosti na variabilitě terénu a dostupnosti dat.



Obrázek 1: Triangulovaná nepravidelná síť

Polyedrický model terénu je specifickým typem DMT, který využívá TIN jako svou základní geometrickou strukturu. Povrch terénu je aproximován soustavou trojúhelníkových ploch, kde každý trojúhelník reprezentuje malou, přibližně rovinnou část terénu. Pro triangulaci T množiny n bodů, kde k bodů leží na konvexní obálce $H(P)$, platí Eulerovy vztahy pro planární grafy:

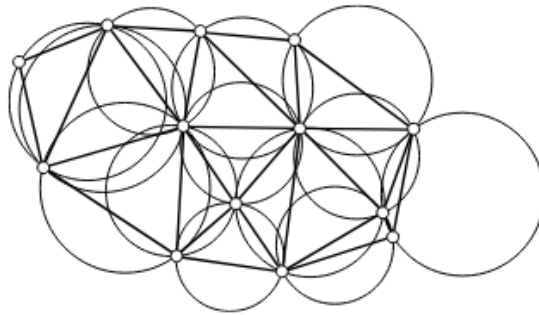
- Počet trojúhelníků: $n_t = 2n - 2 - k$
- Počet hran: $n_h = 3n - 3 - k$

Model je výpočetně relativně nenáročný a geometricky jednoduchý, což usnadňuje implementaci a umožňuje efektivní provádění různých prostorových analýz.

2.2 Delaunayho triangulace

Delaunayho triangulace (DT) je kanonický typ triangulace bodové množiny P v rovině, který je standardem v oblasti GIS a má několik klíčových optimalizačních vlastností. Algoritmus funguje na principu snahy tvorby trojúhelníků které co nejvíc blíží rovnostraným. Základní definující vlastnosti jsou:

- **Podmínka prázdné kružnice opsané (Obrázek 2):** Uvnitř kružnice opsané libovolnému trojúhelníku $t \in DT$ neleží žádný jiný bod z množiny P .
- **Maximalizace minimálního úhlu (maximin angle property):** DT maximalizuje minimální vnitřní úhel všech trojúhelníků v síti.
- **Vytvoření je jednoznačné pod podmínkou že žádné čtyři body neleží na kružnici.**
- **DT je lokálně i globálně optimální vůči kritériu minimálního úhlu**



Obrázek 2: Podmínka DT

Pro vytvoření DT je v rámci této úlohy konstrukce založena na seznamu aktivních hran (Active Edge List - AEL). Algoritmus začíná hranou spojující dva nejbližší body, přičemž p_1 je iniciován jako bod s nejmenší x souřadnicí, p_2 je nejbližší bod. Využívá seznam aktivních hran (AEL). Pro každou následující hranu $e = (p_1, p_2)$ v AEL se následně hledá bod p takový, že maximalizuje úhel $\angle p_1 p p_2$ a současně leží v levé polorovině od hrany e . Tímto vznikne trojúhelník $p_1 p p_2$. Nové hrany (p_2, p) a (p, p_1) se přidají do AEL; pokud však v AEL již existuje hrana s opačnou orientací (hrana (p, p_2) nebo (p_1, p)), tato existující hrana se z AEL odstraní, což signalizuje dokončení polygonu na druhé straně této hrany. Všechny tři hrany jsou přidány do DT. Původní hrana (p_1, p_2) se z AEL odstraní, jelikož se stala vnitřní hranou nově vytvořeného trojúhelníku a byla tak zpracována. Celý proces se opakuje, dokud AEL není prázdný.

Algorithm 1 Delaunayho Triangulace

- 1: Inicializuj DT a AEL jako prázdné seznamy
 - 2: Najdi bod p_1 s nejmenší x -ovou souřadnicí
 - 3: Najdi bod p_2 , který bude Euklidovskou vzdáleností k p_1 nejbližší
 - 4: Z nalezených bodů p_1 a p_2 vytvoř hranu e
 - 5: **Dokud** není AEL prázdná
 - 6: Vezmi první hranu e_1 a otoč její orientaci
 - 7: Najdi Delaunayovský bod $\underline{p} = \arg \max_{p_i \in \sigma_L} \angle (p_1, p_i, p_2)$
 - 8: **Pokud** \underline{p} existuje
 - 9: Vytvoř zbývající strany trojúhelníku $e_2 = (p_2, \underline{p})$ a $e_3 = (\underline{p}, p_1)$
 - 10: Vzniklé hrany přidej do DT
 - 11: Aktualizuj AEL : $updateAEL(e_2, ael)$, $updateAEL(e_3, ael)$
-

2.3 Konstrukce vrstevnic

Vrstevnice byly zkonstruovány lineární interpolací. Tato metoda interpolace aproximuje křivky lineárními segmenty. Jsou-li pro interpolaci dány dva referenční body (např. $P_1 = (x_1, y_1, z_1)$ a $P_3 = (x_3, y_3, z_3)$), lineární interpolace určí souřadnice (x_a, y_a) hledaného bodu na spojnici těchto bodů pro danou výšku z . Z vlastností podobných trojúhelníků lze odvodit následující rovnice:

$$\begin{aligned} x_a &= \frac{x_3 - x_1}{z_3 - z_1}(z - z_1) + x_1, & x_b &= \frac{x_2 - x_1}{z_2 - z_1}(z - z_1) + x_1, \\ y_a &= \frac{y_3 - y_1}{z_3 - z_1}(z - z_1) + y_1, & y_b &= \frac{y_2 - y_1}{z_2 - z_1}(z - z_1) + y_1 \end{aligned}$$

Tímto způsobem se tedy hledají průsečnice roviny, která je definována trojúhelníkem z DT, s horizontální rovinou ρ na kótě z . Podmínku, že rovina ρ protíná hranu určenou body s výškami z_i a z_{i+1} , lze ověřit pomocí nerovnice:

$$(z - z_i)(z - z_{i+1}) < 0$$

Algorithm 2 Generování Vrstevnic

- 1: Inicializuj CL jako prázdný seznam
 - 2: Pro každou úroveň výšky z od z_{min} do z_{max} s krokem dz :
 - 3: Pro každý trojúhelník T v DT :
 - 4: Získej vrcholy p_1, p_2, p_3
 - 5: Vypočítej rozdíly výšek dz_1, dz_2, dz_3
 - 6: **Pokud** trojúhelník je koplanární s rovinou z , **pokračuj**
 - 7: **Pokud** hrana je kolineární s rovinou z , přidej ji do CL
 - 8: **Pokud** trojúhelník protíná rovinu z :
 - 9: Vypočítej průsečíky a, b
 - 10: **Pokud** a a b existují, přidej hranu (a, b) do CL
 - 11: **return** CL
-

2.4 Analýza sklonu terénu

Sklon ϕ je definován jako úhel mezi rovinou daného trojúhelníku T_i z DMT a horizontální (vodorovnou) rovinou. Pro výpočet tohoto úhlu postupujeme následovně:

Nechť má trojúhelník T_i vrcholy $P_1 = [x_1, y_1, z_1]$, $P_2 = [x_2, y_2, z_2]$ a $P_3 = [x_3, y_3, z_3]$. Rovina trojúhelníku je určena dvěma vektory ležícími v této rovině, například vektory \vec{u} a \vec{v} definovanými jako:

$$\begin{aligned} \vec{u} &= P_2 - P_1 = (x_2 - x_1, y_2 - y_1, z_2 - z_1) = (u_1, u_2, u_3) \\ \vec{v} &= P_3 - P_1 = (x_3 - x_1, y_3 - y_1, z_3 - z_1) = (v_1, v_2, v_3) \end{aligned}$$

Normálový vektor \vec{n}_T k rovině trojúhelníku (kolmý k rovině trojúhelníku) se vypočítá jako vektorový součin těchto dvou vektorů:

$$\vec{n}_T = \vec{u} \times \vec{v} = (a, b, c)$$

kde složky a, b, c normálového vektoru jsou:

$$\begin{aligned} a &= u_2v_3 - u_3v_2 \\ b &= u_3v_1 - u_1v_3 \\ c &= u_1v_2 - u_2v_1 \end{aligned}$$

Tento normálový vektor (a, b, c) odpovídá koeficientům v obecné rovnici roviny $ax + by + cz + d = 0$ a také gradientu $\nabla\rho$ funkce definující rovinu.

Normálový vektor horizontální roviny je $\vec{n}_{xy} = (0, 0, 1)$. Úhel ϕ mezi rovinou trojúhelníku a horizontální rovinou se určí z kosinu úhlu mezi jejich normálovými vektory \vec{n}_T a \vec{n}_{xy} :

$$\cos \phi = \frac{|\vec{n}_T \cdot \vec{n}_{xy}|}{\|\vec{n}_T\| \cdot \|\vec{n}_{xy}\|}$$

Jelikož $\vec{n}_{xy} = (0, 0, 1)$ a $||\vec{n}_{xy}|| = 1$, zjednoduší se vzorec na:

$$\cos \phi = \frac{|(a, b, c) \cdot (0, 0, 1)|}{||\vec{n}_T|| \cdot 1} = \frac{|c|}{||\vec{n}_T||}$$

kde $||\vec{n}_T|| = \sqrt{a^2 + b^2 + c^2}$ je velikost (norma) normálového vektoru \vec{n}_T . Samotný sklon (odchylka od roviny) ϕ se pak vypočítá jako:

$$\phi = \arccos \left(\frac{|c|}{||\vec{n}_T||} \right)$$

Tento úhel ϕ představuje spád v daném trojúhelníku.

Algorithm 3 Výpočet Sklonu Trojúhelníku

- 1: Získej vrcholy trojúhelníku p_1, p_2, p_3
 - 2: Vypočítej vektory U a V ze stran trojúhelníku
 - 3: Vypočítej normálový vektor $N \leftarrow U \times V$
 - 4: Vypočítej úhel α mezi N a osou Z
 - 5: Sklon $\theta \leftarrow 90 - \deg(\alpha)$
 - 6: **return** θ
-

2.5 Analýza expozice terénu

Expozice terénu (Orientace) popisuje směr, kterým je svah natočen. Je definována jako azimut průmětu normálového vektoru $\vec{n}_T = (a, b, c)$ roviny trojúhelníku do horizontální roviny xy . Pro výpočet se využívají složky a a b tohoto normálového vektoru, jejichž odvození je popsáno v kapitole o analýze sklonu.

Průmět normálového vektoru do roviny xy je vektor $\vec{n}'_{xy} = (a, b)$. Azimut A tohoto vektoru, a tím i expozice trojúhelníku, se určuje pomocí funkce `atan2`:

$$A = \text{atan2}(b, a)$$

Algorithm 4 Analýza expozice Svahů

- 1: Inicializuj *North, South, West, East* jako prázdné seznamy
 - 2: Pro každý trojúhelník T v DT :
 - 3: Získej transformované vrcholy A, B, C
 - 4: Vypočítej vektory AB a AC
 - 5: Vypočítej normálový vektor $N \leftarrow AB \times AC$
 - 6: **Pokud** $N_z \approx 0$, **pokračuj** (vertikální stěna)
 - 7: Vypočítej úhel expozice svahu *aspect*
 - 8: **Pokud** $45 \leq \text{aspect} < 135$, přidej T do *East*
 - 9: **Jinak pokud** $135 \leq \text{aspect} < 225$, přidej T do *South*
 - 10: **Jinak pokud** $225 \leq \text{aspect} < 315$, přidej T do *West*
 - 11: **Jinak** přidej T do *North*
 - 12: **return** *North, South, West, East*
-

3 Implementace

3.1 Vstupní data

Vstupní data Digitálního modelu reliéfu České republiky 5. generace byla stažena z Geoportálu Zeměměřického úřadu ve formátu `.laz`. Následně byla data otevřena v softwaru CloudCompare, kde byla bodová mračna oříznuta a poté na nich byla použita metoda `Subsample`, která dle zadaných parametrů mění hustotu bodů v datovém souboru. Parametry byly nastaveny tak, aby v bodovém mračně bylo méně než 5000 bodů. Jako testovací oblasti byly vybrány lokality v okolí Českého středohoří, které představují geomorfologicky rozmanitou oblast. Jedna z lokalit (viz Obrázek 3) zahrnuje horu Trojhora, která se vyznačuje hřbetem vzniklým vypreparováním hlubinného magmatického tělesa. Jedná se tedy o poměrně dobře definovatelný geomorfologický útvar, který by neměl DT způsobovat problémy.

Druhou, rozsáhlejší lokalitou, která měla být obtížnější, byla celá plocha stažená v okolí Brné u břehu Brná(Obrázek 5). V této lokalitě se nacházejí různé útvary, které by mohly být pro DT náročné, například část toku Brnámaskovaná dodavatelem, strmé svahy či nepravidelnosti v antropogenních površích, taktéž se jedná o údolí.

V poslední řadě bylo otestován dataset z Digitálního model upovrchu 1. generace obsahující budovy. Zde kvůli velkému měřítku nebyla využita metoda subsampling a tudíž bylo bodové mračno ponechané v surové formě. To na rozdíl od předešlých ukázek se projevilo v prostorově nevyvážené hustotě bodového mračna.

4 Moduly a Třídy

4.1 algorithms.py

Tento modul obsahuje třídu `Algorithms`, která implementuje geometrické a analytické algoritmy výpočtu triangulace, vrstevnic, sklou a expozice.

Třída `Algorithms`

- `__init__()`: Konstruktor třídy.
- `get2VectorsAngle(self, p1, p2, p3, p4)`: Vypočítá úhel mezi dvěma vektory.
- `analyzePointAndLinePosition(self, p, p1, p2)`: Analyzuje pozici bodu vzhledem k úsečce.
- `distance2D(self, p1, p2)`: Vypočítá 2D vzdálenost mezi dvěma 3D body.
- `getNearestPoint(self, q, points)`: Najde nejbližší bod k danému bodu ze seznamu.
- `findDelaunayPoint(self, p1, p2, points)`: Najde bod pro Delaunayho triangulaci.
- `delaunayTriangulation(self, points)`: Provádí Delaunayho triangulaci sady bodů.
- `update_ael(self, e, ael)`: Aktualizuje seznam aktivních hran (AEL) pro algoritmus.
- `contourPoint(self, p1, p2, z)`: Vypočítá bod na vrstevnici.
- `createContourLines(self, dt, zmin, zmax, dz)`: Generuje vrstevnice z Delaunayho triangulace.
- `computeSlope(self, p1, p2, p3)`: Vypočítá sklon trojúhelníku.
- `convertDTToTriangles(self, dt, triangles)`: Převádí Delaunayho triangulaci na seznam trojúhelníků.
- `analyzeDTMSlope(self, dt, triangles)`: Analyzuje sklon digitálního modelu terénu (DTM).
- `analyzeAspect(self, dt, scale, xmin, ymin)`: Analyzuje orientaci svahů (aspect) DTM.

4.2 draw.py

Tento modul obsahuje třídu `Draw`, která je zodpovědná za vykreslování prostorových dat a vizualizací.

Třída `Draw`

- `__init__(*args, **kwargs)`: Konstruktor widgetu pro kreslení.
- `calculateTransform()`: Vypočítá transformační matici pro vykreslování.
- `transformPoint(self, p)`: Transformuje 3D bod na souřadnice obrazovky.
- `paintEvent(self, e)`: Obsluhuje událost kreslení.
- `setPoints(self, points_)`: Nastaví body k vykreslení.
- `setViewPoints(self, view_points_)`: Nastaví viditelnost bodů.
- `setViewDT(self, view_dt_)`: Nastaví viditelnost Delaunayho triangulace.

- `setViewContourLines(self, view_contour_lines_)`: Nastaví viditelnost vrstevnic.
- `setViewSlope(self, view_slope_)`: Nastaví viditelnost sklonů.
- `setViewAspect(self, view_aspect_)`: Nastaví viditelnost expozice svahů.
- `getPoints()`: Získá aktuální body.
- `getDT()`: Získá aktuální Delaunayho triangulaci.
- `getTriangles()`: Získá aktuální trojúhelníky.
- `setDT(self, dt_)`: Nastaví Delaunayho triangulaci.
- `setTriangles(self, triangles_)`: Nastaví trojúhelníky.
- `setContourLines(self, contour_lines_)`: Nastaví vrstevnice.
- `setAspectData(self, north, south, west, east)`: Nastaví data pro orientaci svahů.

4.3 edge.py

Tento modul definuje třídu `Edge`, která reprezentuje hranu v geometrických strukturách trojúhelníků.

Třída `Edge`

- `__init__(self, p1, p2)`: Konstruktor hrany se dvěma koncovými body.
- `getStart()`: Získá počáteční bod hrany.
- `getEnd()`: Získá koncový bod hrany.
- `switchOrientation()`: Přepne orientaci hrany.
- `__eq__(self, other)`: Porovnává dvě hrany na rovnost.

4.4 MainForm.py

Tento modul obsahuje třídu `Ui_MainForm`, která definuje strukturu a chování hlavního okna aplikace, včetně interakce s uživatelským rozhraním a spouštění algoritmů.

Třída `Ui_MainForm`

- `__init__()`: Konstruktor hlavního formuláře.
- `setupUi(self, MainForm)`: Nastaví uživatelské rozhraní hlavního okna.
- `dtClick()`: Obsluhuje kliknutí na tlačítko pro Delaunayho triangulaci.
- `contourLinesClick()`: Obsluhuje kliknutí na tlačítko pro vrstevnice.
- `settingsClick()`: Obsluhuje kliknutí na tlačítko pro nastavení.
- `analyzeSlopeClick()`: Obsluhuje kliknutí na tlačítko pro analýzu sklonu.
- `analyzeAspectClick()`: Obsluhuje kliknutí na tlačítko pro analýzu expozice svahů.
- `pointsChanged()`: Obsluhuje změnu bodů.
- `DTChanged()`: Obsluhuje změnu Delaunayho triangulace.
- `contourLinesChanged()`: Obsluhuje změnu vrstevnic.
- `slopeChanged()`: Obsluhuje změnu sklonu.
- `aspectChanged()`: Obsluhuje změnu expozice svahů.
- `openFile()`: Obsluhuje otevření souboru.
- `clearData()`: Vymaže aktuální data.
- `clearAll()`: Vymaže všechna data a nastavení.
- `retranslateUi(self, MainForm)`: Přeloží texty uživatelského rozhraní.

4.5 qpoint3df.py

Tento modul definuje třídu `QPoint3DF`, která rozšiřuje `QPointF` o třetí (Z) souřadnici, čímž umožňuje reprezentaci 3D bodů s desetinnou čárkou.

Třída `QPoint3DF`

- `__init__(self, x, y, z)`: Konstruktor 3D bodu.
- `getZ()`: Získá Z souřadnici bodu.

4.6 Settings.py

Tento modul obsahuje třídu `Ui.Settings`, která definuje uživatelské rozhraní pro nastavení parametrů vrstevnic, jako jsou rozsahy Z souřadnic a měřítko.

Třída `Ui.Settings`

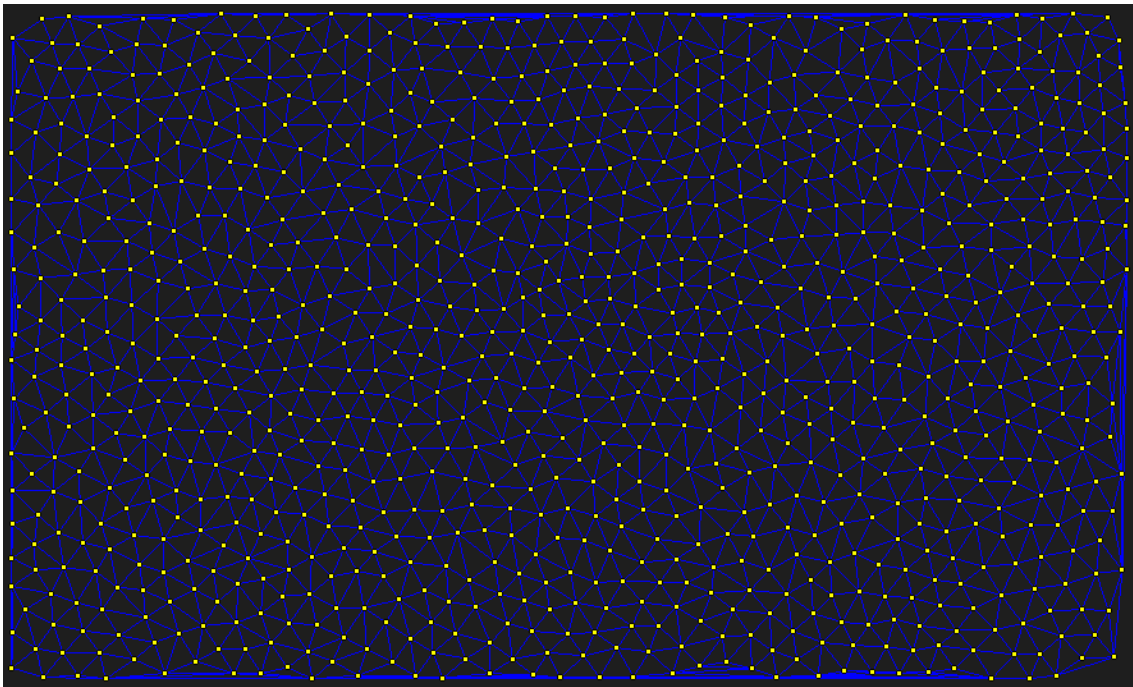
- `__init__()`: Konstruktor formuláře nastavení.
- `setupUi(self, Settings)`: Nastaví uživatelské rozhraní okna nastavení.
- `retranslateUi(self, Settings)`: Přeloží texty uživatelského rozhraní.
- `getZmin()`: Získá minimální Z hodnotu.
- `getZmax()`: Získá maximální Z hodnotu.
- `getdZ()`: Získá krok pro vrstevnice.
- `getScale()`: Získá měřítko.
- `getXmin()`: Získá minimální X hodnotu.
- `getYmin()`: Získá minimální Y hodnotu.

4.7 triangle.py

Tento modul definuje třídu `Triangle`, reprezentuje trojúhelník.

Třída `Triangle`

- `__init__(self, p1_, p2_, p3_, slope_, aspect_)`: Konstruktor trojúhelníku.
- `getVertices()`: Získá vrcholy trojúhelníku.
- `getSlope()`: Získá sklon trojúhelníku.
- `getAspect()`: Získá orientaci svahu trojúhelníku.
- `setSlope(self, slope_)`: Nastaví sklon trojúhelníku.
- `setAspect(self, aspect_)`: Nastaví orientaci svahu trojúhelníku.



Obrázek 3: Bodové mračno a Tin: Trojhora

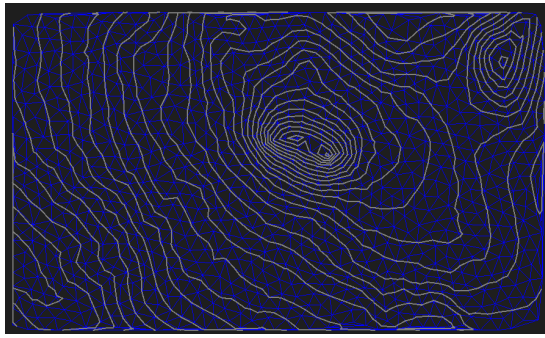
5 Výsledky

Po spuštění třídy MainForm, která zobrazila uživatelské prostředí vytvořené v Qt, byly pomocí tlačítka Open otevřeny upravené soubory LAZ/LAS. Na plátně se následně zobrazily body bodového mračna. Poté byly spuštěny jednotlivé analýzy ze třídy Algorithms, přičemž jako první musela být provedena analýza DT, jakožto základní kámen pro navazující analýzy.

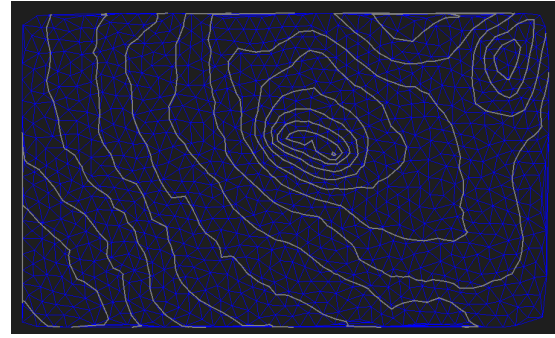
5.1 Trojhora

Jak lze vidět na obrázku 3, bodové mračno je v oblasti Trojhory velice pravidelné. Funkce subsample v CloudCompare zde byla nastavena na vzorkování s rozestupy 30 metrů, a tudíž se původní počet bodů snížil z 5000 na 986. Následně je zde také vidět již zpracovaná triangulace v modré barvě. TIN je velice pravidelný.

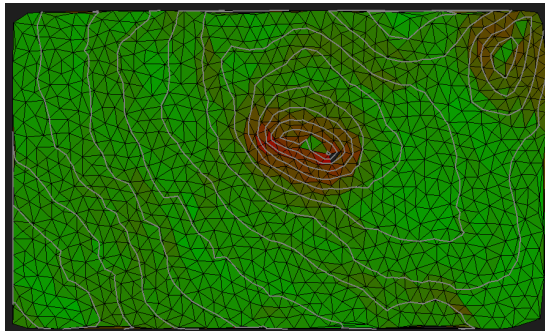
Dále na obrázku 4 můžeme vidět jednotlivé analýzy, které proběhly nad modelem TIN z obrázku 3. Část (a) ukazuje vygenerované vrstevnice se základním intervalem (ZIV) 5 metrů. Vrstevnice v šedé barvě vizuálně vyhovují terénu a naznačují téměř hladké hrany, které zobrazují hlavní a vedlejší vrchol Trojhory. Na druhou stranu v části (b) lze vidět, že ZIV 10 metrů je již příliš velký interval, který neodpovídá dané hustotě bodového mračna; hrany jsou roztržité a hrubé, nicméně základní cíl vrstevnic byl splněn. Co se týče analýzy sklonu (c), aplikace správně vyznačila srázy Trojhory, kterých je zde, vzhledem vulkanickému původu, mnoho. Srázy a strmější svahy jsou znázorněny červenou barvou a mírnější terén zelenou. V poslední řadě analýza expozice svahů (d), využívající žlutou, modrou, červenou a zelenou barvu, ukazuje orientaci jednotlivých trojúhelníků modelu ke světovým stranám.



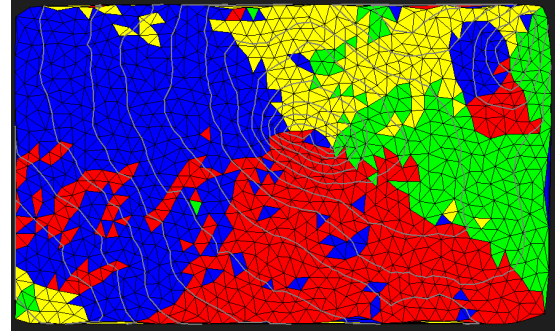
(a) Oblast Trojhora: Vrstevnice (ZIV 5 metrů)



(b) Oblast Trojhora: Vrstevnice (ZIV 10 metrů)



(c) Oblast Trojhora: Sklon

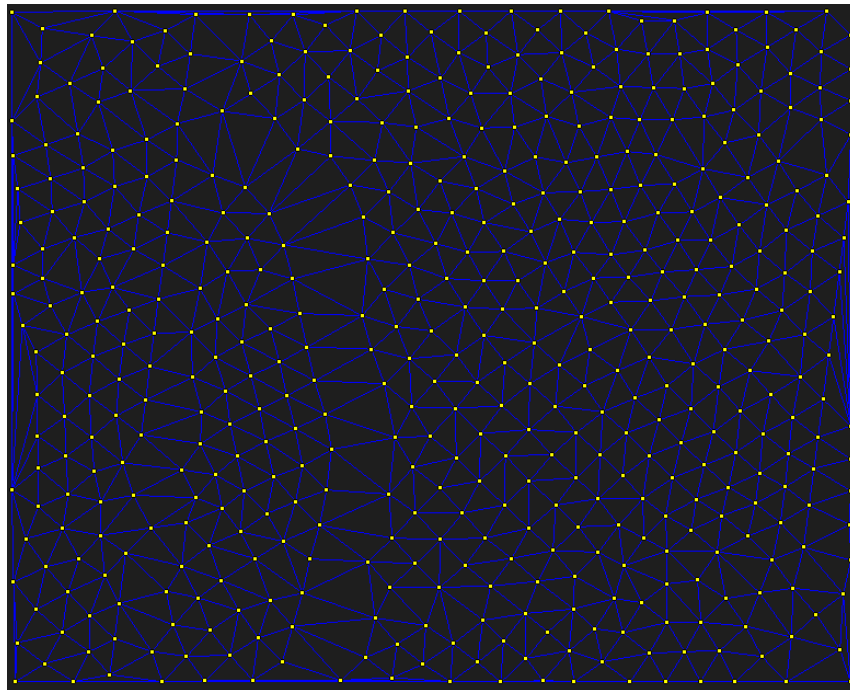


(d) Oblast Trojhora: expozice svahů

Obrázek 4: DMT analýza oblasti Trojhora

5.2 Brná

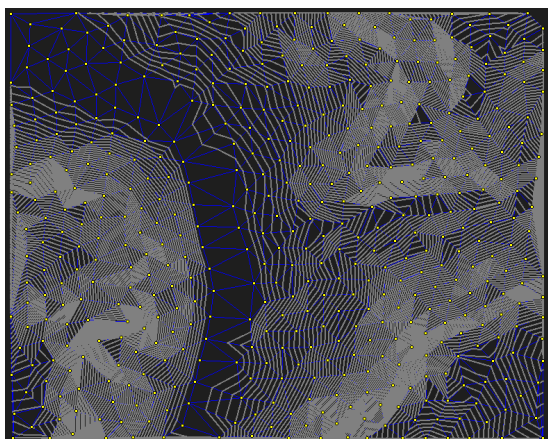
Další lokalitou v okolí toku Brnáu obce Brná byla vybrána větší testovací oblast, která byla pomocí funkce subsample snížena na 490 bodů s rozestupem 100 metrů. Výsledné bodové mračno je znázorněno na obrázku 5, kde je patrná díra v datech v místě vodního toku Labe, která zde existovala již před úpravou v programu CloudCompare. Na obrázku jsou proto viditelné větší trojúhelníky v TIN, které tuto díru vyplnily, body nacházející se poblíž břehových linií.



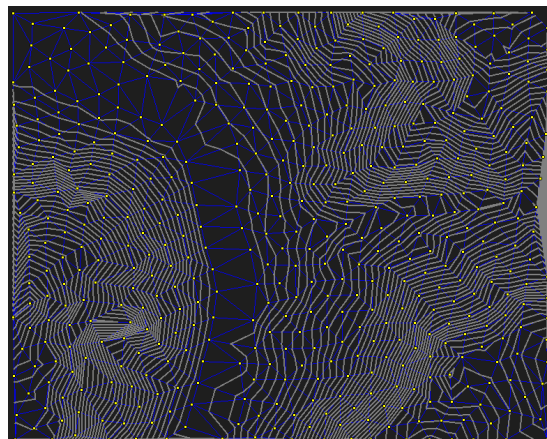
Obrázek 5: Oblast Labe: Bodové mračno a TIN

Na obrázku 6 v části (a) jsou vygenerovány vrstevnice se ZIV 5 metrů, což ukazuje, že pro

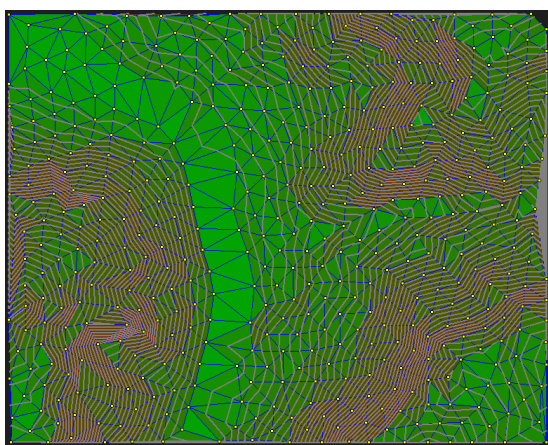
danou lokalitu a danou hustotu bodového mračka je tento interval příliš malý. Část (b) znázorňuje vhodnější interval vrstevnic, konkrétně ZIV 10 metrů. V části (c) je vidět slabina DT a triangulace obecně v tomto kontextu, a to zejména při absenci dat, která byla v tomto datasetu patrná v oblasti vodního toku. Výsledný sklon triangulace pak nereálně reprezentuje sklon vodního toku, jelikož trojúhelníky, které by měly pokrývat řeku, mají vrcholy umístěné na pobřežních liniích. Výška těchto bodů tedy nemusí odpovídat skutečné výšce hladiny toku, čímž dochází ke ztrátě topologické správnosti řeky, která představuje jeden z nejdůležitějších kartografických prvků. Toto zkreslení je rovněž evidentní v části (d), kde je tok řeky orientován podle hran jednotlivých trojúhelníků odpovídajících pobřežním liniím.



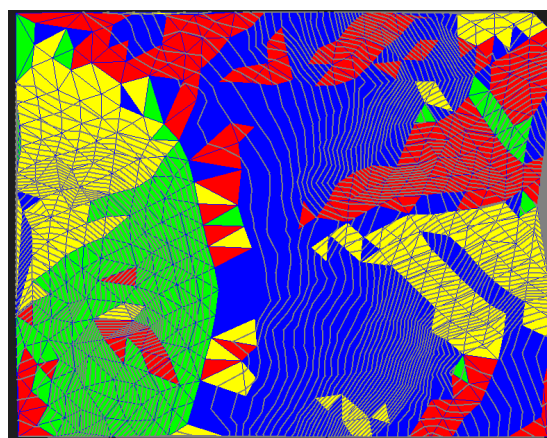
(a) Oblast Labe: Vrstevnice (ZIV 5 metrů)



(b) Oblast Labe: Vrstevnice (ZIV 10 metrů)



(c) Oblast Labe: Sklon

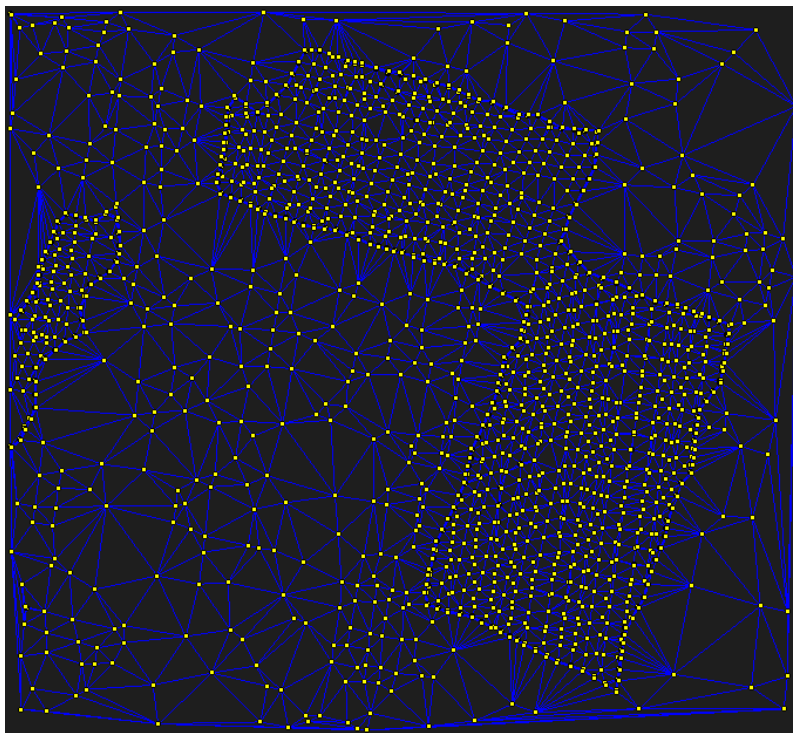


(d) Oblast Labe: expozice svahů

Obrázek 6: DMT analýza oblasti Brná

5.3 Budova

Dataset bodového mračka budovy byl jediný, který nebyl upraven v softwaru CloudCompare. To je patrné na obrázku 7, kde lze pozorovat neúměrnou a náhodnou hustotu bodů, která se projevuje vyšší koncentrací bodů na střechách a nižší koncentrací na povrchu země. Tomu odpovídá i vygenerovaný TIN, který je ve srovnání s lokalitami Brná a Trojhory značně nepravidelný.

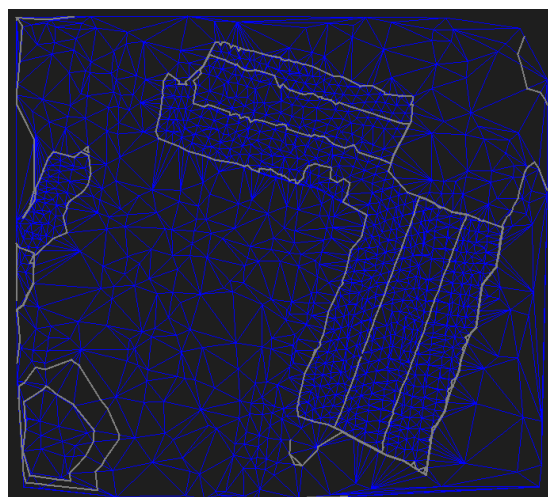


Obrázek 7: Budova: Bodové mračno a TIN

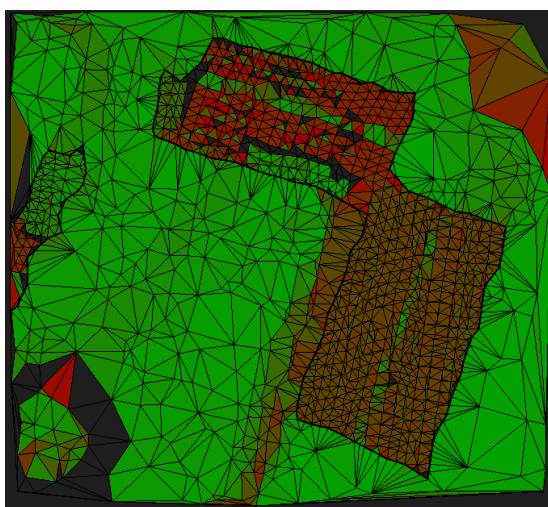
Nepravidelnost bodového mračna se projevuje také ve výsledných analýzách. V částech (a) a (b) obrázku 8 jsou zobrazeny vygenerované vrstevnice, které správně interpolují nejvyšší objekty, tedy střechy. ZIV 1 metr je ovšem přesnější a dokonce umožňuje identifikovat jednotlivé komíny. V levém dolním rohu lze také pozorovat artefakt vzniklý nedostatečným počtem bodů v této lokalitě, který je reprezentován většími trojúhelníky a může naznačovat existenci malého vrcholku či hromady. Tento artefakt je také obtížně identifikovatelný v části (c). Sklon střech je však zobrazen adekvátně a naznačuje, že se jedná o sedlové, nikoliv ploché střechy. expozice trojúhelníků v části (d) jednoznačně určuje orientaci střech, zatímco povrch země vykazuje kvůli šumu spíše náhodnou orientaci. Přesto by tento výstup byl vhodný například k určení optimálních míst pro instalaci solárních panelů.



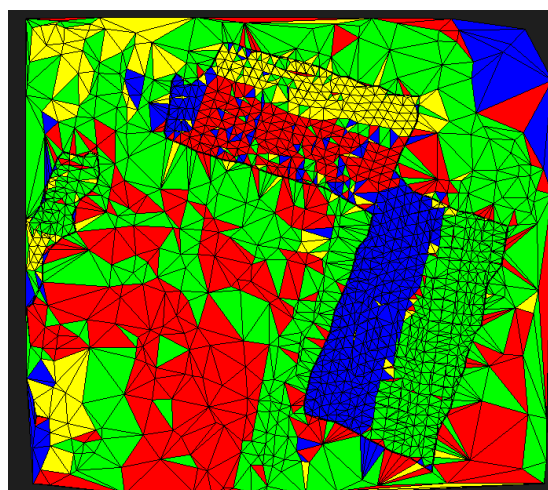
(a) Budova: ZIV 1 metr



(b) Budova: ZIV 5 metrů



(c) Budova: Sklon



(d) Budova: expozice

Obrázek 8: DMT analýza objektu Budova

6 Závěr

2D Delaunayova triangulace a analýzy DMT byly úspěšně implementovány a dosáhly dobrých výsledků, především u upravených pravidelných bodových mračen. Irregularnější bodové mračno ukázalo slibné výsledky co se týče střech, nicméně terén mimo střechy nebyl ani po provedených analýzách dostatečně interpretovatelný. V budoucnu bychom v rámci tohoto úkolu celý program vylepšili následovně:

- Zabudováním předzpracování dat přímo do aplikace. Uživatel by měl možnost nahrát libovolně velký dataset, nad kterým by byla implementována funkce subsample obdobná té v programu CloudCompare, a to s parametry volenými uživatelem.
- Úpravou modulu `settings.py`, aby bylo možné nastavit ZIV menší než 1 metr, což by umožnilo hlubší analýzu digitálních modelů povrchu.
- Zavedením možnosti výběru barev při analýzu sklonu

Reference

- [1] BAYER, T. (2008). *Algoritmy v digitální kartografii*. Nakladatelství Karolinum, Praha.
- [2] BAYER, T. (2025). *2D triangulace, DMT*. Přednáška pro předmět Algoritmy počítačové kartografie, Katedra aplikované geoinformatiky a kartografie, Přírodovědecká fakulta UK. Dostupné z: https://web.natur.cuni.cz/~bayertom/images/courses/Adk/adk5_new.pdf [cit. 10. 5. 2025].
- [3] ČUZK(2025). *Geoportál Zeměměřického úřadu*. dostupné z <https://ags.cuzk.gov.cz/geoprohlizec/>
- [4] GOOGLE(2025) Google Ai Studio, Gemini 2.5 Pro Experimental. Dostupné z https://aistudio.google.com/prompts/new_chat