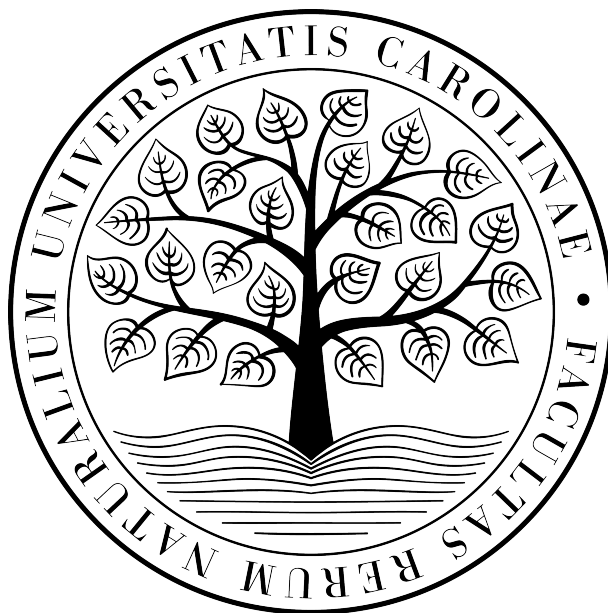


Přírodovědecká fakulta  
Univerzita Karlova



Algoritmy počítačové kartografie  
Úkol č. 3: Digitální model terénu

Nováková Lucie, Martínek David a Anna Zikešová

1.N-GKDPZ

Praha 2024

## Zadání

### Úloha č. 3: Digitální model terénu

*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 slabunami 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.

**Hodnocení:**

<b>Krok</b>	<b>hodnocení</b>
Delaunay triangulace, polyederický model terénu.	10 b.
Konstrukce vrstevnic, analýza sklonu a expozice	10 b.

# 1 Popis a rozbor problému

Model terénu byl používán již ve starých mapách nejprve šlo o kopečkovou metodu, dále byl terén šrafován. V současnosti se využívají vrstevnice spolu se stínováním. První DMT byly poprvé použity v 50. letech 20. století. DMT může být definován jako statistická reprezentace spojitého povrchu Země pomocí velkého množství bodů se známými souřadnicemi X, Y a Z v definovaném souřadnicovém systému. Vstupní data jsou tedy získávána pomocí LiDARu nebo fotogrammetrie. Body z datového souboru jsou použity k tvorbě DMT. Tato data lze kombinovat se snímky satelitními či leteckými, čímž je možné výsledné modely a mapy upřesnit. DMT je model, který zobrazuje pouze povrch Země bez objektů lidské činnosti a bez objektů přírodní sféry. Modely se využívají pro vizualizaci a analýzu Země ve 3D, slouží tedy pro plánování výstavby a infrastruktury, pro výpočet sklonu a orientace svahu. Tyto výpočty pak slouží k vyhodnocení hrozby přírodních rizik např. eroze, sesuvy půdy. Taktéž se využívá ve vojenství, lesnictví, počítačové grafice atd.

Mezi metody zpracování DMT patří TIN - *Triangular Irregular Network* a reprezentace povrchu pomocí rastru. Triangulace do podoby TIN je všestranným způsobem reprezentace reálného terénu. Výsledné trojúhelníkové plochy jsou považovány za rovinné a z toho důvodu poskytují spojitý model terénu. Oblasti s velkou variabilitou terénu jsou pokryté hustou sítí bodů, čímž vzniká velké množství malých trojúhelníků. Rovinné oblasti pak obsahují méně bodů čímž vznikají velké trojúhelníky. Mimo informaci o výšce zachycují údaje o sklonu terénu. Oproti jiným modelům využívající pravidelné sítě nebo vrstevnice počítá TIN výšku pro každý bod pomocí interpolace výšek okolních bodů, čímž je přesněji odhadnuta jeho výška.

Reprezentace povrchu pomocí pravidelného rasteru jsou narození od TIN náchylné k nadhodnocení nebo podhodnocení při pokrývání oblastí s výraznou strukturou a členitostí. Pravidelný raster má pevně danou velikost buňky a při změně členitosti terénu je pak rovinná oblast pokryta rasterem s malou velikostí buňky, pak je členitý terén pokryt rasterem s velkou velikostí buňky.

## 1.1 Delaunay triangulace

*Delaunayho triangulace* (DT) je metodou pro vytváření trojúhelníku, které patří k nejčastěji používaným triangulacím. Lze ji tvořit jak ve 2D tak ve 3D. DT tvoří trojúhelníky tak, aby se co nejvíce blížily právě rovnostranným trojúhelníkům. Tímto se pak minimalizuje případná deformace trojúhelníků a to tak, že se snaží maximalizovat minimální vnitřní úhel. Tímto vzniká pravidelnější síť. Princip algoritmu je poměrně jednoduchý, právě třemi body je proložena kružnice a dochází ke zkoumání zda uvnitř této kružnice leží další bod, pokud tomu tak je jsou zvoleny jiné tři body. Pokud v kružnici neleží další bod je vytvořen trojúhelník. Při triangulaci může dojít ke vzniku rovinných trojúhelníků. Ty jsou vytvořeny ve chvíli, kde algoritmus vybere pro tvorbu trojúhelníku body se stejnou nadmořskou výškou, k čemuž zpravidla dochází u vrcholů kopců a v okolí hřbetnic a údolnic. Tyto trojúhelníky mají sklon  $0^\circ$  a nemají definovanou orientaci vůči světovým stranám. Tato skutečnost může způsobovat problémy při následných analýzách a modelování. Tato chyba se dá odstranit dodáním výškových bodů nebo linií terénní kostry.

Při tvorbě triangulace dochází k postupnému vkládání jednotlivých bodů do již vytvořené DT. Nejprve je vybrán náhodný bod  $P_1$  ze vstupní množiny. K němu je poté na základě Euklidovské vzdálenosti nalezen nejbližší bod  $P_2$ . Tímto vzniká hrana  $e = (P_1, P_2)$ . Dalším hledaným bodem je bod  $\underline{P}$  ležící v levé polo-rovině vůči  $e$ , a který zároveň minimalizuje poloměr kružnice opsané hraně  $e$  a tomuto bodu. Po nalezení nejvhodnějšího bodu  $\underline{P}$  splňující dané podmínky, vznikají nové hrany  $e_2 = (\underline{P}, P_1)$  a  $e_3 = (P_2, \underline{P})$ , které tvoří první trojúhelník. Pokud by bod  $P_3$  algoritmus nenalezl, otočí orientaci hrany  $e$  a vyhledávání bodu se opakuje.

Vytvořené hrany jsou přidány do *Active Edge List (AEL)*. U první hrany dojde k otočení její orientace a je nalezen bod  $\underline{P}$ . Čímž vzniknou další dvě nové hrany, které jsou, v případě, že se tam již s opačnou orientací nenachází, přidány do AEL. Pokud ano, z AEL je odstraněna a přidána do výsledné triangulace. Pokud by pro aktuální hranu nebyl nalezen bod, je hrana přidána do výsledné triangulace, jelikož je součástí konvexního obalu. Tento postup se opakuje do té doby, dokud není *Active Edge List* prázdný.

## Pseudokód Delaunay Triangulace

---

### Algorithm 1 Delaunay triangulation

---

- 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 bodu  $p_1$  nejbližší
  - 4: Z nalezených bodů  $p_1$  a  $p_2$  vytvoř hranu  $e$  a hranu  $e_{op}$  s opačnou orientací
  - 5: Přidej hrany do  $ael$
  - 6: Dokud není  $ael$  prázdná:
    - 7: Vezmi první hranu  $e_1$  a změň její orientaci
    - 8: Najdi Delaunayovský bod  $p_{dt} = \arg \max_{p_i \in \sigma_L} < (P_1, P_i, P_2)$
    - 9: Pokud  $p_{dt}$  existuje:
      - 10: Vytvoř zbývající strany trojúhelníku  $e_2 = (e_{1op}, p_{dt})$  a  $e_3 = (p_{dt}, e_{1op})$
      - 11: Vzniklé hrany přidej do  $dt$
      - 12: Aktualizuj  $ael$
- 

## 1.2 Konstrukce vrstevnic

Vrstevnice jsou konstruovány lineární interpolací. Tato metoda je založena na analytické geometrii, prokládá křivky lineárními funkcemi. V případě, že jsou zadány dva body  $x_a$  a  $y_a$  pomocí souřadnic, lineární interpolace je přímka mezi nimi. Z podobnosti trojúhelníků lze odvodit rovnici vzájemných vztahů.

$$x_a = \frac{x_3 - x_1}{z_3 - z_1}(z - z_1) + x_1,$$

$$y_a = \frac{y_3 - y_1}{z_3 - z_1}(z - z_1) + y_1$$

$$x_b = \frac{x_2 - x_1}{z_2 - z_1}(z - z_1) + x_1,$$

$$y_b = \frac{y_2 - y_1}{z_2 - z_1}(z - z_1) + y_1$$

Tato metoda tedy hledá průsečnice roviny, které jsou určeny trojúhelníkem DT a vodorovné roviny  $\rho$  o výšce  $h$ . Zda rovina  $\rho$  prochází hranou tvořenou zadanými body lze ověřit pomocí rovnice

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

### Pseudokód konstrukce vrstevnic

---

**Algorithm 2** *Create contour lines*


---

```

1: Inicializuj prázdný seznam vrstevnic contours
2: Zpracuj každý trojúhelník iterací nad dt v krocích po 3:
3:   Pro každý trojúhelník získej vrcholy a jejich nové z souřadnice
4:   Pro z v intervalu (zmin, zmax) s krokem dz:
5:     Počítej výškové rozdíly na hranách trojúhelníků
6:     Vynechej trojúhelníky, které jsou koplanární
7:     Hrany, které jsou kolineární přidej do contours
8:     Pro každou dvojici hran v trojúhelníku, které protíná vrstevnice:
9:       Spočítej průsečíky a a b
10:      Vytvoř z průsečíků hranu e1
11:      Přidej hranu e1 do seznamu contours

```

---

## 1.3 Analýza sklonu terénu

DMT jsou využívány z důvodu jejich snadné dostupnosti zdrojových dat. Dlouží k analýzám hydrologických poměrů, sesuvů půdy či návrhů infrastruktura atd. Mezi potřebné charakteristiky morfologie povrchu se řadí orientace, normální a horizontální křivost či sklon.

Analýza sklonu terénu je prováděna nad každým trojúhelníkem z DMT. Pro zadanou rovinu  $\rho$  je vypočítán gradient  $\nabla\rho$ , tedy maximální vektor spádu, který má v daném bodě směr normály k vrstevnici a je orientován ve směru dané funkce  $p$ .

$$\nabla\rho(x_0, y_0, z_0) = \left( \frac{\partial p}{\partial x}(x_0), \frac{\partial p}{\partial y}(y_0), \frac{\partial p}{\partial z}(z_0) \right) = (a, b, c)$$

Rovina  $\rho$  je definována maticí

$$\begin{vmatrix} x - x_1 & y - x_1 & z - x_1 \\ x_2 - x_1 & y_2 - x_1 & z_2 - x_1 \\ x_3 - x_1 & y_3 - x_1 & z_3 - x_1 \end{vmatrix} = 0$$

Následně pak z této matice může být vypočítána odchylka  $\varphi$  od roviny  $\pi$

$$\varphi = \arccos \left| \frac{n_1 n_2}{\|n_1\| \|n_2\|} \right|$$

**Algorithm 3** *Compute slope*

- 
- 1: Vypočítej normálový vektor pro vstupní body  $p_1, p_2$  a  $p_3$
  - 2: Vypočítej jejich normu  $n = \sqrt{n_x^2 + n_y^2 + n_z^2}$
  - 3: Vypočítej sklon  $\rho = \frac{n_z}{n}$
- 

**Pseudokód výpočtu sklonu terénu****1.4 Analýza orientace terénu**

Další charakteristikou je orientace terénu. Je definována jako azimut průmětu gradientu  $\nabla\rho$  roviny trojúhelníku do roviny  $x, y$ . Pro vektor gradientu  $\vec{v}$  platí, že

$$\vec{v} = \left( \frac{\partial p}{\partial x}(x_0), \frac{\partial p}{\partial y}(y_0) \right) = (a, b, 0)$$

Azimut vektoru  $\vec{V}$  pak lze spočítat pomocí vzorce:

$$A = \arctan\left(\frac{b}{a}\right)$$

**Pseudokód výpočtu orientace terénu****Algorithm 4** *Compute aspect*

- 
- 1: Vypočítej normálové vektory pro  $p_1, p_2$  a  $p_3$
  - 2: aspect = arctan  $\frac{n_y}{n_x}$
- 

**2 Aplikace**

Vytvořená aplikace pro tvorbu digitálního modelu terénu byla vytvořena v prostředí *QT Creator*. Vstupní data tvoří množinu bodů a výstupem je polyedrický DMT, který je vytvořen právě nad vstupní množinou a je představován vrstevnicemi, které jsou doplněny vizualizací sklonu trojúhelníků a jejich expozicí.

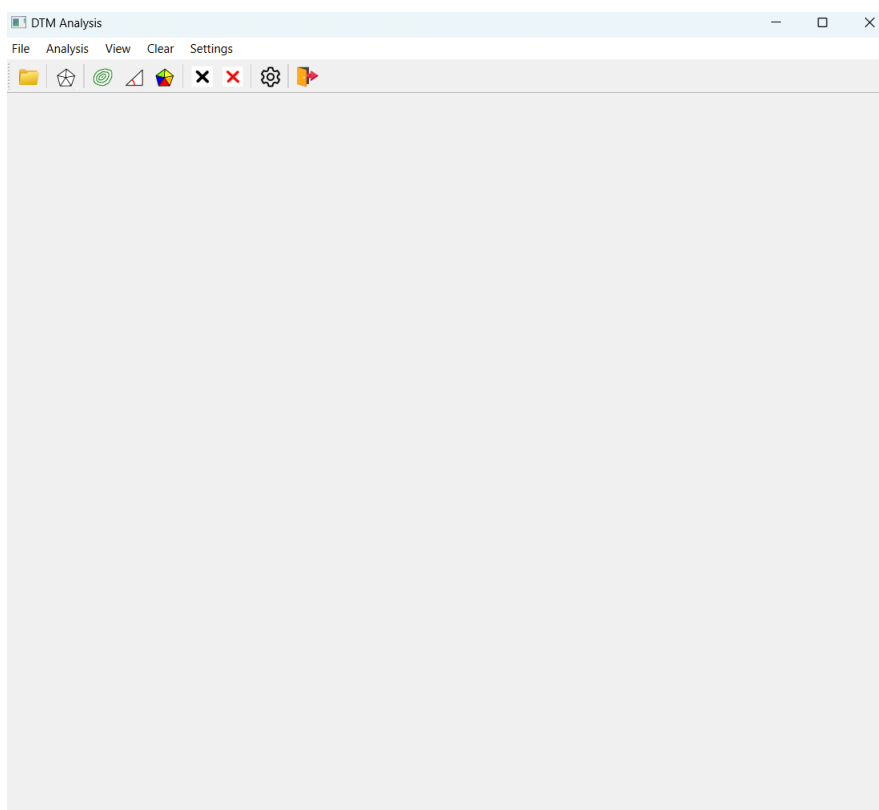
Po spuštění aplikace přes soubor *mainform.py* je otevřeno okno, jehož většinou část tvoří prázdná plocha pro vykreslení bodů (viz obr. 1). V horní části je pak lišta, kde se nachází menu s jednotlivými funkcemi. V záložce *File* je možné pomocí funkce *Open* vyvolat dialogové okno a vybrat vstupní data ve formátu *.txt*, která jsou následně vykreslena (viz obr. 2).

Nahrané body je možné analyzovat s využitím funkcí v záložce *Analysis*. Funkce *Create DT* vytvoří z bodů trojúhelníky Delaunayho triangulace. Jejich vytvoření je nutné pro spuštění dalších analýz. Funkce *Create contour lines* vytváří vrstevnice, funkce *Analyse slope* vede k obarvení trojúhelníků na základě jejich sklonu a funkce *Analyse aspect* vede k přebarvení trojúhelníků dle jejich orientace vůči světovým stranám.

Funkce *Clear results* vymaže výsledky, ale ponechá vstupní body, funkce *Clear all* pak slouží k vymazání výsledků včetně vykreslených bodů.

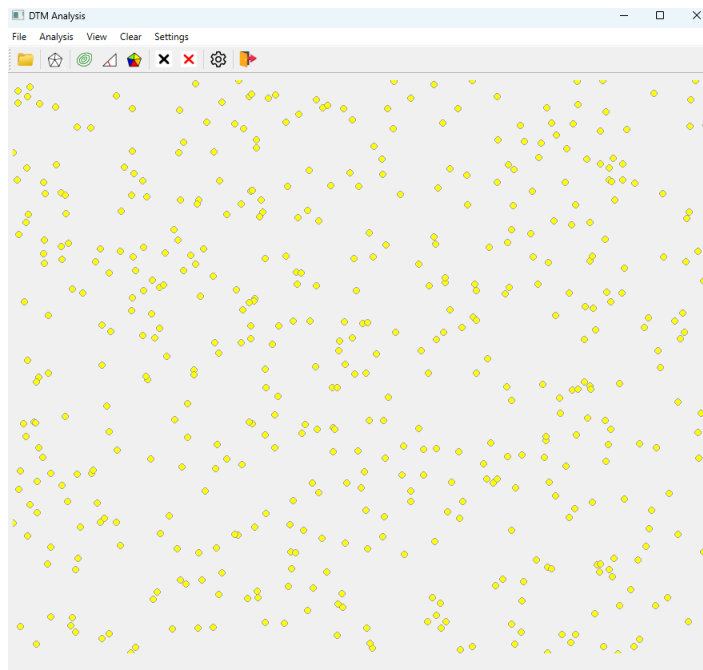
V záložce *Settings* je možné definovat parametry pro tvorbu vrstevnic, konkrétně minimální a maximální nadmořskou výšku a krok, se kterým mají být vytvořeny. Po kliknutí se objeví vyskakovací okno (obr. 3), ve kterém může uživatel upravit požadované hodnoty a tlačítkem *OK* je potvrdí.

Popsané funkce je možné vyvolat i pomocí ikon umístěných pod záložkami menu, v pořadí zleva se jedná o funkce *Open file*, *Generate DT*, *Create contour lines*, *Analyse slope*, *Analyse aspect*, *Clear results*, *Clear all* a *Exit*. Ikony odpovídají jednotlivým funkcím v záložkách menu a po umístění kurzoru myši na ikonu je vypsán i název odpovídající funkce.

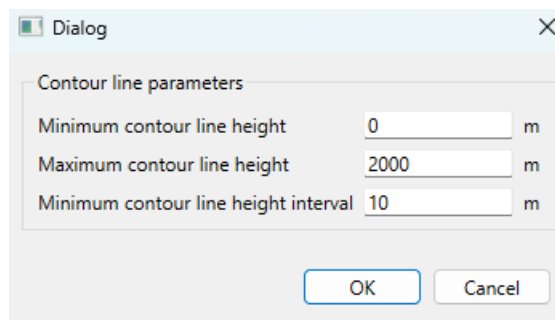


Obrázek 1: Ukázka aplikace s nahranou bodovou vrstvou





Obrázek 2: Ukázka aplikace s nahranou bodovou vrstvou



Obrázek 3: Nastavení parametrů pro tvorbu vrstevnic

### 3 Dokumentace

Program byl vytvořen v prostředí PyCharm v programovacím jazyce Python s využitím *QT Creator* a modulu *pyshp*, který umožňuje práci se shapefiley. Program se skládá ze souborů *mainform.py*, *draw.py*, *algorithms.py*, *Settings.py*, *qpoint3df.py*, *edge.py* a *triangle.py*, které názvem odpovídají příslušným třídám. Ve složce *icons* se nachází obrázkové soubory využívané pro ikony jednotlivých funkcí.

#### 3.1 Třída MainForm

Třída *mainform* zabezpečuje inicializaci okna samotné aplikace, horní lišty, panelu nástrojů, ikon a tlačítek. Propojuje jednotlivé interaktivní položky s metodami vykonávající specifické akce.

Metody *setupUi* a *retranslateUi* byly vytvořeny automaticky na základě vytvořeného prostředí v *Qt Creator*. V první z nich je mimo jiné provedeno napojení jednotlivých položek menu a tlačítek na připravené metody.

Metoda *openClick* ukládá do proměnných *w* a *h* aktuální šířku a výšku okna pro vykreslování bodů. Následně volá metodu *getData* třídy *Draw*, čímž dochází k načtení vstupních dat a jejich vykreslení.

Metoda *createDTClick* slouží k vytvoření Delaunayho triangulace a k jejímu vykreslení. Nejprve jsou pomocí metody *getPoints* třídy *Draw* načteny vykreslené body. Metodou *createDT* třídy *Algorithms* je z nich vytvořena Delaunayho triangulace, která je metodou *setDT* předána třídě *Draw*. Následně je vyvolána metoda pro překreslení okna, aby došlo k vykreslení vytvořené trojúhelníkové sítě.

Metoda *createContourLinesClick* slouží k vytvoření vrstevnic a jejich vykreslení. Metodou *getDT* je ze třídy *Draw* načtena vytvořená Delaunayho triangulace. Na základě nastavených parametrů jsou metodou *createContourLines* třídy *Algorithms* vytvořeny vrstevnice, které jsou metodou *setContours* předány třídě *Draw* a na závěr je vyvolána metoda pro překreslení okna.

Metoda *analyzeSlopeClick* slouží k analýze a vykreslení sklonu trojúhelníků. Metodou *getDT* je ze třídy *Draw* načtena vytvořená Delaunayho triangulace a metodou *analyzeDTMSlope* třídy *Algorithms* je vytvořena síť trojúhelníků s analyzovaným sklonem, která je metodou *setDTMSlope* předána třídě *Draw*. Na závěr je vyvolána metoda pro překreslení okna.

Metoda *analyzeExpositionClick* funguje na stejném principu jako předchozí metoda, ale analyzuje orientaci trojúhelníka vůči světovým stranám a využívá k tomu metodu *analyzeDTMAspect* třídy *Algorithms*.

Metody *clearAllClick* a *clearClick* slouží k vymazání bodů a všech výsledků, respektive k vymazání pouze výsledků, a využívají k tomu odpovídající metody třídy *Draw*.

Metoda *setParameters* slouží k nastavení parametrů pro tvorbu vrstevnic. Pomocí dialogového okna uživatel zadá hodnotu pro minimální a maximální nadmořskou výšku pro tvorbu vrstevnic a krok, se kterým mají být vytvořeny. Tyto informace pak metoda ukládá do připravených proměnných.

### 3.2 Třída QPoint3DF

Tato třída slouží k definování bodu v 3D prostoru, dědí atributy *x* a *y* od třídy *QPointF* a přidává k nim souřadnici *z*. V inicializační metodě dochází k předání a inicializaci všech tří souřadnic a metoda *getZ* navrací *z*-ovou souřadnici bodu.

### 3.3 Třída Edge

Tato třída slouží k definici datového typu pro hranu trojúhelníka, na vstupu má argumenty pro počáteční a koncový bod hrany, oba typu *qpoint3df*. V inicializační metodě dochází k nastavení počátečního a koncového bodu hrany, které jsou vráceny pomocí příslušných metod *getStart* a *getEnd*.

Metoda *changeOrientation* vrací nově vytvořenou hranu s prohozeným počátečním a koncovým bodem vstupní hrany. Dále je definován operátor pro porovnání, zda jsou dvě zadané hrany stejné, tedy jestli si odpovídají jejich počáteční a koncové body.

### 3.4 Třída Triangle

Tato třída slouží k definování datového typu pro trojúhelník a na vstupu má tři argumenty odpovídající bodům tvořící vrcholy trojúhelníka typu *qponit3df*, dále sklon a orientaci trojúhelníka, oba datového typu *float*. V první metodě dochází k inicializaci předávaných vstupních argumentů a k vytvoření trojúhelníka. Jeho jednotlivé vrcholy, sklon a orientace jsou vráceny příslušnými metodami *getVertices*, *getSlope* a *getAspect*.

### 3.5 Třída Settings

Tato třída slouží k vytvoření dialogového okna, pomocí kterého uživatel nastavuje parametry pro tvorbu vrstevnic. Na vstupu má tři argumenty odpovídající aktuálně nastavené minimální a maximální nadmořské výšce pro tvorbu vrstevnic a jejich krok, které se zobrazují při otevření okna.

Metoda *getInputs* pak vrací uživatelem zadané hodnoty parametrů pro tvorbu vrstevnic.

### 3.6 Třída Algorithms

Tato třída v sobě implementuje algoritmy pro analýzu digitálního modelu terénu.

Metoda *get2LineAngle* má na vstupu čtyři body typu *QPoint3DF* tvořící dvě linie, mezi kterými počítá úhel. Nejprve dochází k výpočtu vektorů, jejich skalárního součinu a norem obou vektorů. Následně je spočítán argument pro funkci *cosinus*, který je zaokrouhlen na hodnotu 1 nebo -1, pokud je větší, respektive menší. Metoda pak vrací vypočítaný úhel.

Metoda *getPointAndlinePosition* slouží k určení polohy bodu vůči linii a na vstupu má analyzovaný bod a dva body typu *QPoint3DF* určující přímku. Nejprve dochází k výpočtu vektorů a následně determinantu matice tvořené těmito vektory. Pokud je determinant kladný, bod leží v levé polorovině od přímky a metoda vrátí hodnotu 1. V případě záporného determinantu bod leží v pravé polorovině a dochází k návratu hodnoty 0. Pokud bod leží na přímce, vyjde nulový determinant a metoda pak vrací hodnotu -1.

Metoda *getNearestPoint* má na vstupu bod typu *QPoint3DF* a seznam všech bodů. Metoda prochází všechny body a vrací nejbližší bod. Počítá euklidovskou vzdálenost mezi vstupním bodem a body v seznamu. Je vybrán bod, který má nejnižší vzdálenost.

Metoda *getDelaunayPoint* má na vstupu dva body - počáteční a koncový typu *QPoint3DF* a seznam všech bodů, také typu *QPoint3DF*. Metoda prochází všechny body a vrací Delaunayovský bod, který se

dvěma vstupními body svírá největší úhel a splňuje tak podmínku Delaunayho triangulace pro maximalizaci minimálního úhlu.

Metoda *updateAEL* má na vstupu hranu typu *edge* a seznam *ael* tvořený hranami typu *edge*. V metodě dochází na základě porovnání s opačně orientovanou hranou k aktualizaci seznamu dostupných hran *ael* tak, aby obsahoval pouze hrany, které mohou dále vstupovat do Delaunayho triangulace.

Metoda *createDT* má na vstupu seznam bodů typu *QPoint3DF* a vytváří Delaunayho triangulaci definovanou seznamem hran typu *edge*.

Metoda *getContourPoint* má na vstupu dva body typu *QPoint3DF*, výšku roviny typu *float* a na základě teorie vrací bod typu *QPoint3DF* představující průsečík mezi hranou trojúhelníku a zadanou rovinou tvořící základ pro tvorbu vrstevnic.

Metoda *createContourLines* má na vstupu Delaunayho triangulaci, minimální a maximální nadmořskou výšku pro tvorbu vrstevnic a krok vrstevnic, všechny typu *float*. Metoda postupně prochází všechny trojúhelníky a pro ně všechny nadmořské výšky definované vstupními parametry a vytváří hrany typu *Edge* tvořící vrstevnice, které ukládá do seznamu a ten na závěr vrací.

Metoda *computeSlope* má na vstupu tři body typu *QPoint3DF* a vrací hodnotu sklonu trojúhelníku.

Metoda *computeAspect* má na vstupu tři body typu *QPoint3DF* a vrací hodnotu expozice trojúhelníku.

Metoda *analyzeDTMSlope* má na vstupu seznam hran typu *Edge* tvořící Delaunayho triangulaci. Metoda postupně prochází všechny trojúhelníky, definuje body tvořící jejich vrcholy, počítá sklon trojúhelníka, za jeho expozici ukládá hodnotu 0 a na základě těchto parametrů vytváří trojúhelníky datového typu *Triangle*, které ukládá do vytvořeného seznamu, který na závěr vrací.

Metoda *analyzeDTMAAspect* funguje na stejném principu jako předchozí metoda, ale počítá orientaci trojúhelníka a za jeho sklon ukládá hodnotu 0.

### 3.7 Třída Draw

Třída *Draw* složí k inicializaci proměnných a zajišťuje grafické rozhraní aplikace. Inicializační metoda má dva poziční argumenty a dochází v ní k inicializaci proměnných.

V inicializační metodě jsou definovány proměnné pro ukládání bodů, Delaunayho triangulace, vrstevnic a trojúhelníků.

Metoda *getData* slouží k načtení vstupních dat. Prvně dojde k vyvolání dialogového okna, ze kterého je získána cesta k souboru vstupních dat. Když je soubor vybrán, tak je soubor načten. Následně jsou inicializovány seznamy pro ukládání souřadnic bodů. Dále po řádcích procházen celý soubor a do připravených seznamů jsou ukládány jednotlivé souřadnice. Z nichž jsou následně získány minimální a maximální souřadnice.

Metoda *getAspectColor* definuje ke každé světové straně (dané úhlovým intervalem) barvu, která se vykreslí, když má trojúhelník určitou orientaci.

Metoda *paintEvent* má na vstupu argument *QPaintEvent*. Tato metoda slouží k vykreslení bodů a výsledků analýz. Pomocí *for* cyklu jsou procházeny a vykresleny všechny body. Dále jsou procházeny všechny trojúhelníky a na základě hodnot sklonu nebo expozice je příslušná analýza vykreslena. Dále jsou pomocí *for* cyklu procházeny a vykresleny všechny trojúhelníky Delaunayho triangulace a základní a zdůrazněné vrstevnice.

Metoda *getPoints* vrací seznam vykreslených bodů.

Metoda *getDT* vrací seznam hran typu *Edge* tvořící Delaunayho triangulaci.

Metoda *clearAll* slouží k vymazání všech bodů a výsledků. Všechny odpovídající proměnné jsou tak nastaveny na prázdné seznamy a dochází k vyvolání metody na překreslení okna.

Metoda *clearResults* slouží k vymazání výsledků analýz, ale k zachování bodů. Stejně jako v předchozím případě všem proměnným kromě načtených bodů přiřazuje prázdné seznamy a poté dochází k vyvolání metody na překreslení okna.

Metoda *setDT* má na vstupu seznam hran typu *Edge* tvořící Delaunayho triangulaci, který přiřazuje do odpovídající proměnné třídy *Draw*.

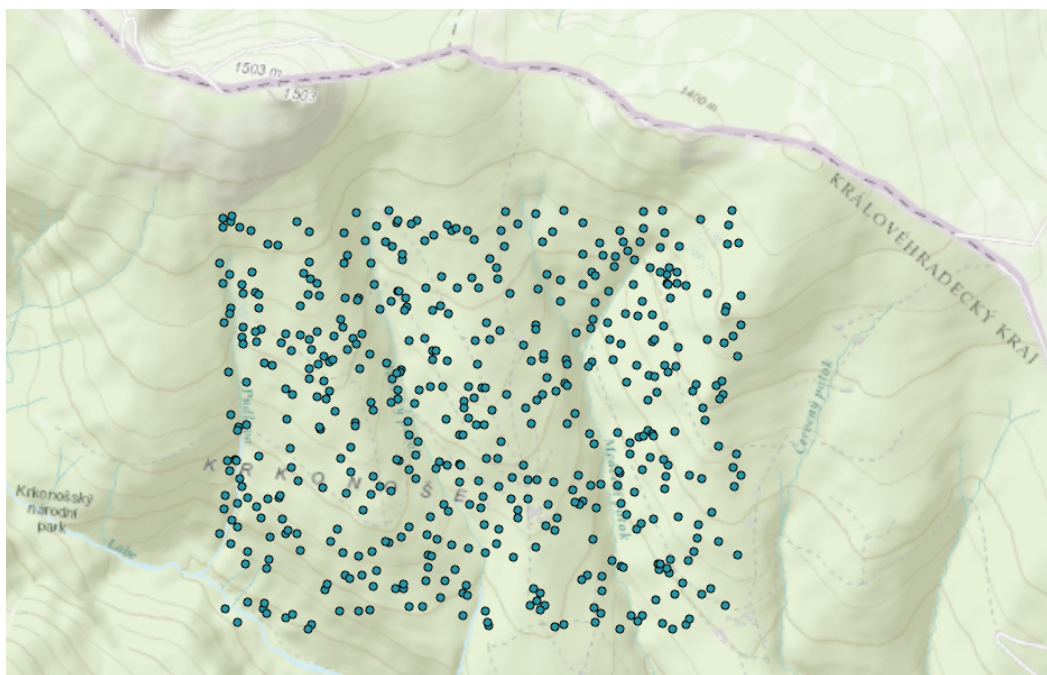
Metoda *setContours* má na vstupu dva seznamy hran typu *Edge* představující základní a zdůrazněné vrstevnice, které přiřazuje do odpovídajících proměnných třídy *Draw*.

Metoda *setDTMAsspect* má na vstupu seznam typu *Triangle* představující trojúhelníky s vypočtenou orientací, který přiřazuje do odpovídající proměnné třídy *Draw*.

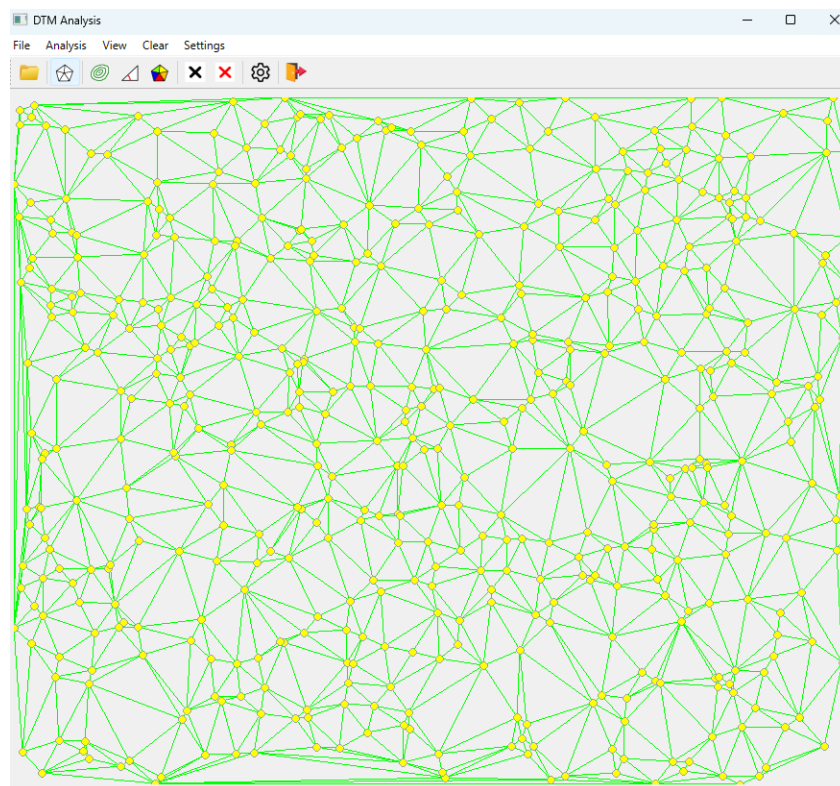
Metoda *setDTMSlope* má na vstupu seznam typu *Triangle* představující trojúhelníky s vypočteným sklonem, který přiřazuje do odpovídající proměnné třídy *Draw*.

## 4 Výsledky

Data obsahují 500 bodů, které byly vygenerovány nad DMR 5G z ČÚZK. Body se nachází v oblasti Sedmidolů v Krkonošského národního parku (obr. 4). Body na území mají nadmořskou výšku od 885 m n.m. do 1345 m n.m.

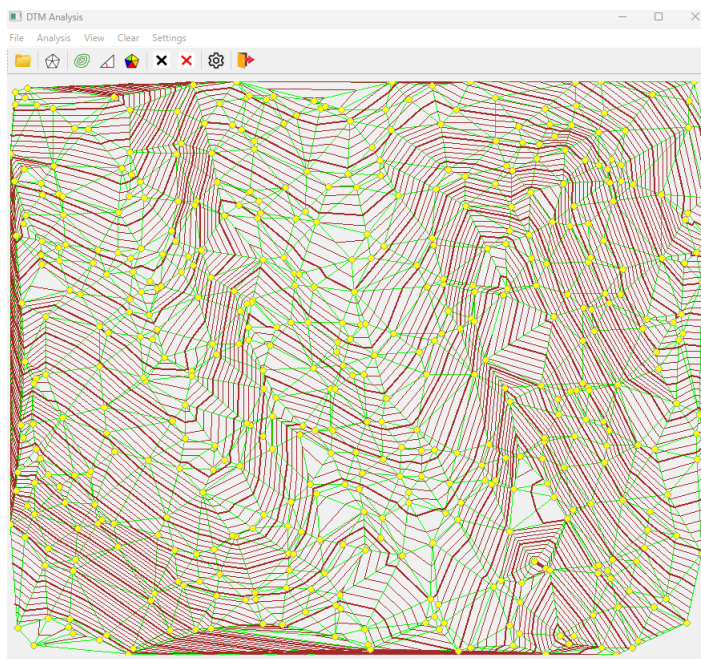


Obrázek 4: Přehled rozmístění vstupních bodových dat na podkladové mapě

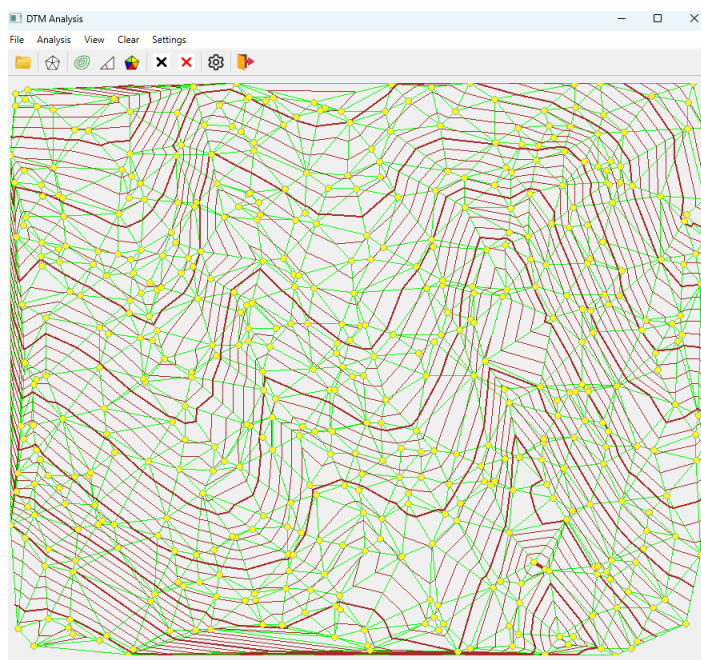


Obrázek 5: Výsledek Delaunayho triangulace pro vstupní body

Výsledek vrstevnic vytvořených s nastavením parametrů na rozsah nadmořských výšek 0 až 2000 m a s krokem 5 m je na obrázku 6. Pro zvolenou bodovou vrstvu se ukázal krok 5 metrů jako nevhodný a výsledné vrstevnice jsou až příliš podrobné, v některých místech splývají do sebe. Vrstevnice s krokem 10 m na obrázku 7 a 15 m na obrázku 8 jsou již přehlednější a dávají lepší přehled o průběhu terénu. Vrstevnice s krokem 20 m na obrázku 9 nedávají dostatečně podrobný přehled o průběhu terénu.

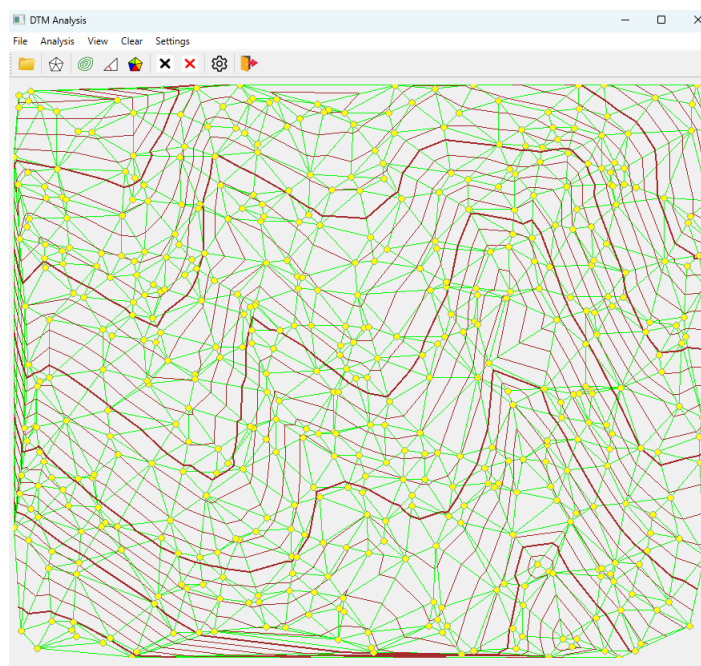


Obrázek 6: Vrstevnice vytvořené s krokem 5 m

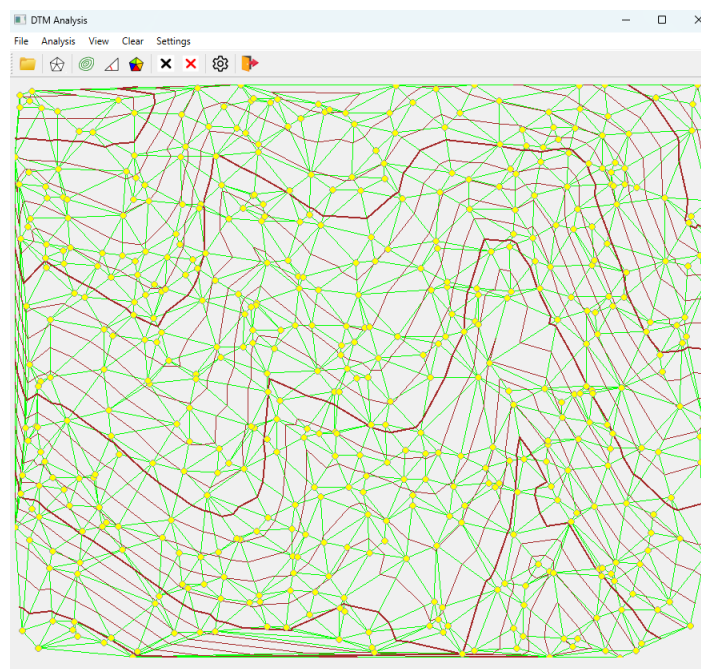


Obrázek 7: Vrstevnice vytvořené s krokem 10 m





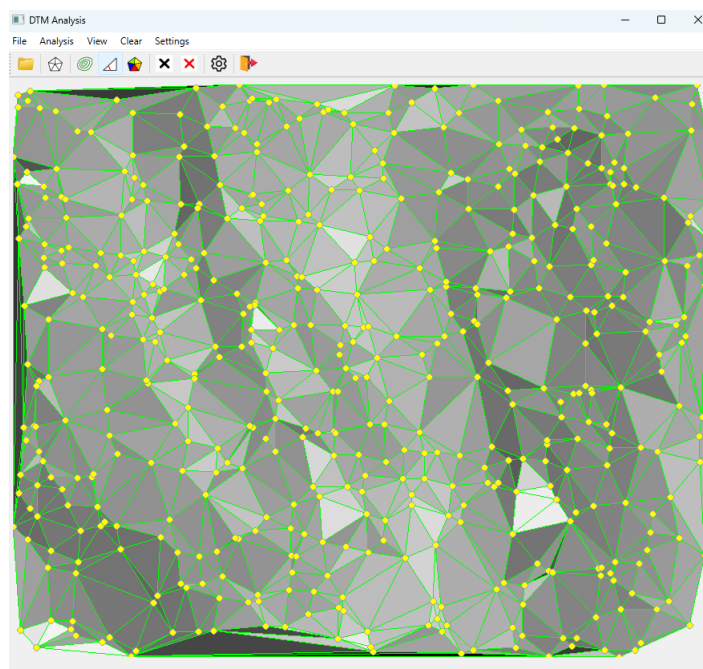
Obrázek 8: Vrstevnice vytvořené s krokem 15 m



Obrázek 9: Vrstevnice vytvořené s krokem 20 m

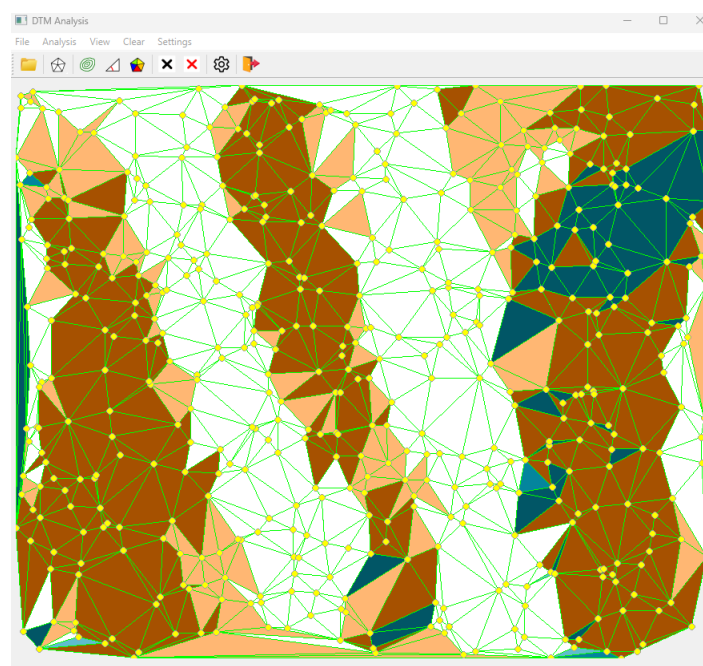
Další prováděnou analýzou je analýza sklonu jednotlivých trojúhelníků vytvořené triangulační sítě. Sklon aplikace vizualizuje prostřednictvím odstínů šedi. Výsledky pro vstupní množinu bodů jsou na obrázku 10. Čím je trojúhelník vybarven tmavší šedou, tím je sklon v místě strmější. Světlé trojúhelníky značí rovinaté

povrchy.



Obrázek 10: Analýza sklonu DMT

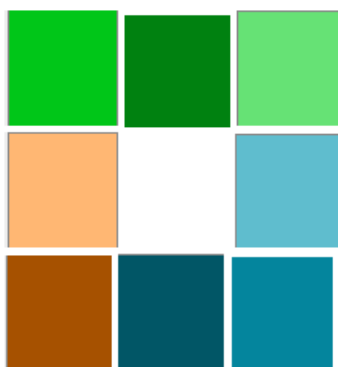
Poslední analýzou, kterou aplikace provádí, je orientace vůči světovým stranám (obr. 11). V Tabulce 1 je popsána zvolená barevná škála a na obrázku 12 jsou vyznačené barvy u odpovídajících směrů. Oblasti s bílou barvou jsou rovinaté a že byly algoritmem vyhodnoceny jako oblasti bez dominantního směru.



Obrázek 11: Analýza orientace DMT

Tabulka 1: Barevná škála orientace DMT

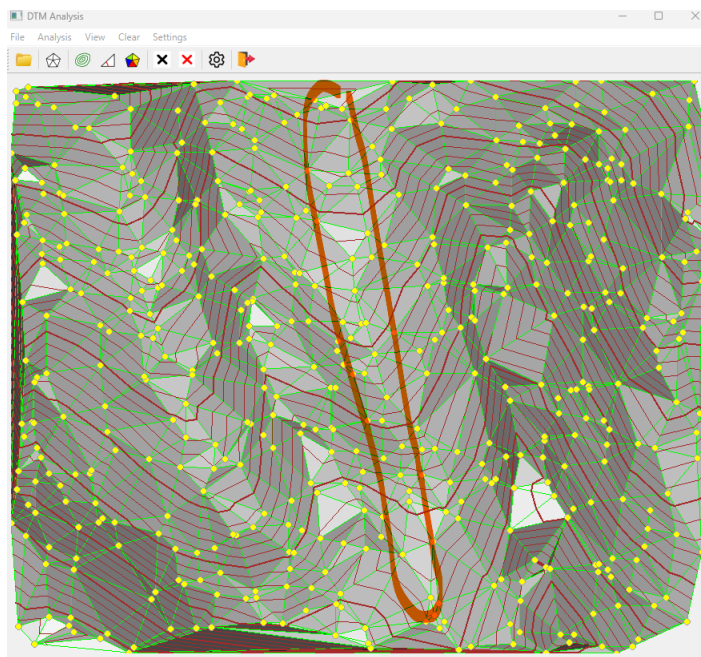
Orientace svahu	RGB kód
sever	(0,129,16)
severovýchod	(102,226,117)
východ	(95,189,206)
jihovýchod	(4,133,157)
jih	(1,86,102)
jihozápad	(166,81,0)
západ	(225,183,115)
severozápad	(0,198,24)



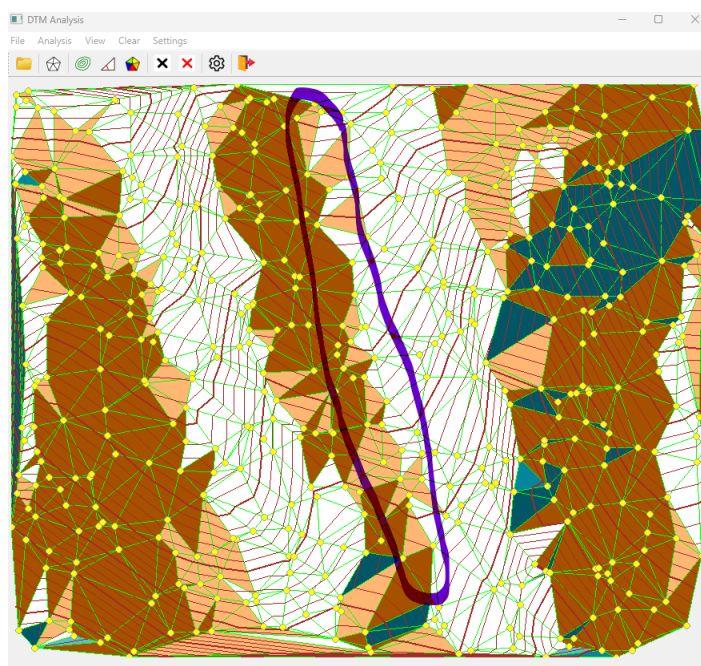
Obrázek 12: Barevná škála orientace DMT

## 4.1 Hřbet

Funkčnost aplikace byla otestována na vybraných terénních tvarech. Prvním z nich byl hřbet, pro který platí, že se od středové oblasti ke krajům nadmořské výšky snižují. Přehled zvoleného hřbetu a výsledky analýz jsou na obrázcích 13 a 14. Oblast hřbetu je v aplikaci dobře rozeznatelná z analýzy sklonu, trojúhelníky jsou v oblasti vykreslené světle šedou barvou na středu a tmavší na okrajích, to značí klesající terén. Také z vrstevnic je hřeben dobře patrný. Z analýzy orientace je patrné, že se v oblasti hřbetu mění orientace z roviny na jihozápadní.



Obrázek 13: Sklon a vrstevnice v oblasti hřbetu



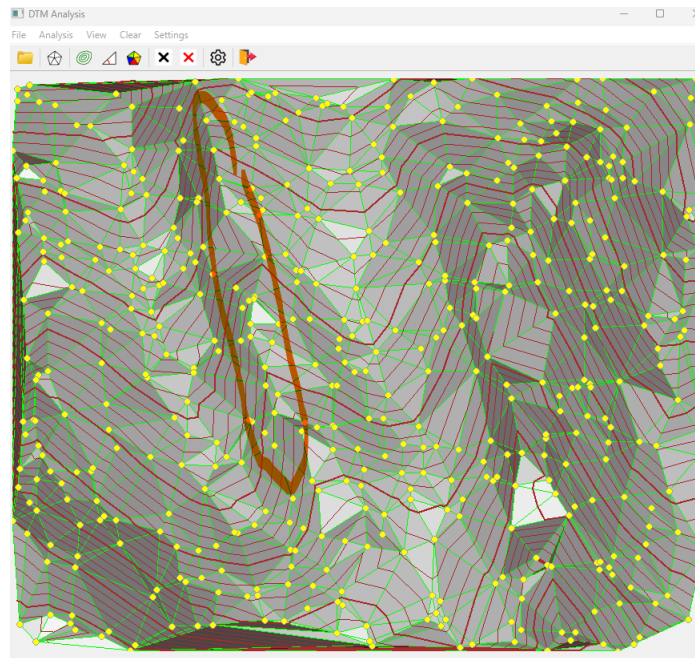
Obrázek 14: Orientace a vrstevnice v oblasti hřbetu

## 4.2 Údolí

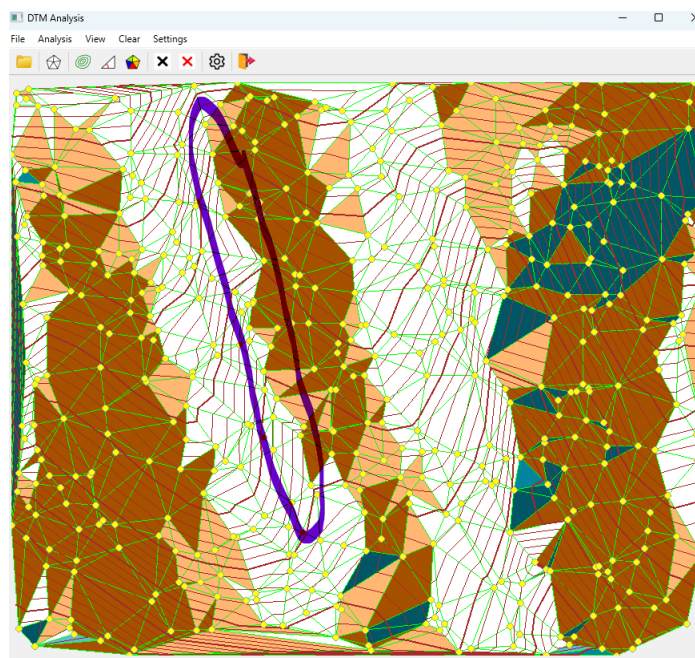
Dalším analyzovaným tvarem terénu je údolí, pro které naopak platí, že se od středové části nadmořské výšky zvyšují. Přehled zvoleného údolí a výsledky analýz jsou na obrázku 15 a 16. Údolí je z analýzy sklonu



hůře rozeznatelné než hřbet, jeho oblast není tak rozsáhlá. Pro výraznější znázornění by byla potřeba větší koncentrace bodů, tak by vznikla podrobnější triangulační síť a sklon by byl názornější. Z analýzy orientace je patrné, že se v oblasti údolí mění orientace jihozápadní na rovinu.



Obrázek 15: Sklon a vrstevnice v oblasti údolí



Obrázek 16: Orientace a vrstevnice v oblasti údolí

Aplikace nedává dobré výsledky triangulace v okrajových oblastech, kde jsou velké vzdálenosti mezi body

a dochází zde tak ke vzniku protáhlých úzkých trojúhelníků. Tento problém by mohlo řešit umělé vkládání bodů do těchto okrajových oblastí, které by zajistily vznik přibližně rovnostranných trojúhelníků a díky tomu lepšího výsledku triangulace.

## 5 Závěr

Vytvořená aplikace vytváří digitální model nad vstupní množinou dat pomocí Delaunay triangluace. Rovněž je možné nad vytvořeným modelem provádět analýzy jako je sklon a orientace terénu nebo vytvořit vrstevnice.

Pro vykreslení vrstevnic a analýzu sklonu a orientace trojúhelníků je nutné nejdříve vytvořit Delaunayho triangulaci. Možným rozšířením programu by mohlo být splnění bonusových úloh, jako je třeba 3D vizualizace terénu s promítáním. Trojrozměrný prostor by umožňoval snazší orientaci v hodně členitém terénu.

## 6 Zdroje

přednášky z předmětu *Algoritmy počítačové kartografie*, dostupné z: <http://web.natur.cuni.cz/bayertom/index.php/teaching/algoritmy-pocitacove-kartografie>

BUCKLEY, A. (2008): Aspect-slope map, ArcGIS Blog, <https://www.esri.com/arcgis-blog/products/product/mapping/aspect-slope-map/> (27. 5. 2024).

PACINA, J., BREJCHA, M. (2014): Digitální modely terénu. Fakulta životního prostředí Univerzity Jana Evangelisty Purkyně. [http://envimod.fzp.ujep.cz/sites/default/files/skripta/28e\\_finalisk.pdf](http://envimod.fzp.ujep.cz/sites/default/files/skripta/28e_finalisk.pdf) (27.5.2024).