
TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky a mezioborových inženýrských studií

Studijní program: B2612 – Elektrotechnika a informatika

Studijní obor: 1802T007 – Informační technologie

**Varianty řešení automatizace postupů v tvorbě
modelových sítí**

**Possibilities of solutions in automatization
method for model mesh creation**

Diplomová práce

Autor: **Jan Dostál**

Vedoucí práce: RNDr. Blanka Malá Ph.D.

Konzultant:

V Liberci 20. 5. 2011

Prohlášení

Byl(a) jsem seznámen(a) s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval(a) samostatně s použitím uvedené literatury a na základě konzultací s vedoucím diplomové práce a konzultantem.

Datum

Podpis

Abstrakt CZ

Tato diplomová práce seznamuje s typy dat a problematikou převodu dat z GIS (Geografický informační systém) pro účely tvorby plně objemového modelu krajiny. Data se převádějí do vybraného profesionálního modelovacího programu 3ds Max na základě vytvořené aplikace, která je cílem této práce a to pomocí vimportovaného skriptovacího jazyka maxscript, který dovoluje přizpůsobit a rozšířit funkcionalitu programu 3ds Max. Prostředí 3d studia dovoluje zobrazovat změny okamžitě, takže uživatel přímo vidí výsledek převodu a lze tak pochopit model v 3d prostředí a umožnit další přehlednou práci hlavně s jejich další editací a úpravou prostřednictvím vytvořené aplikace. Tato práce dále obsahuje rozbor konkrétních problémů spojených s převodem z GIS do CAD systému (3ds Max) a jejich řešení, převodem geometrie do programu Gmsh, který se používá na katedře NTI pro další práci. Pomocí vytvořené aplikace kromě možností převodů a vizualizace můžeme model rozřezat na více samostatných částí podle zvolené definice, v případě převodu do Gmsh lze definovat objemy a vytvářet podskupiny prvků, kterým se v Gmsh přiřazují fyzikální vlastnosti. Pro přehlednost a efektivnost s ovládáním vytvořené aplikace tato práce obsahuje podrobný manuál.

Klíčová slova: GIS, 3dsmax, vizualizace, 3d model, maxscript

Abstract ANG

My diploma paper introduces different types of dates and the problems of date transfer from GIS (Geographical Information System) for the purposes of creating a fully volumetric landscape model. The dates are transferred into the professional modeling program 3ds Max on the basis of the created application being the main goal of this diploma paper, all of these with the help of the vimport script language maxscript that allows to adapt and extend the functionality of 3ds Max program. 3D studio setting allows to display the changes immediately, so that the user can directly see the result of the transfer. Like this, it is possible to understand the model in 3D setting and facilitate another clearly organized work mainly its successive editing and modification by means of the created application.

This work also contains the analysis of the concrete problems connected with the transfer from GIS into CAD system (3ds Max), their solution and the transfer of geometry into Gmsh program used to create entrance files for hydro-geological and transport models. With the help of the created application, except for the transfer possibilities and visualization, the model can be cut into several separate parts according to the chosen definition. In case of Gmsh transfer, we can define the volumes and create subgroups of elements to which physical features are matched in Gmsh. This work also contains a detailed user manual to make the manipulation clear and effective enough.

Key words: GIS, 3ds Max, visualization, 3d model, maxscript

Obsah

PROHLÁŠENÍ	3
ABSTRAKT CZ	4
ABSTRACT ANG	5
SEZNAM OBRÁZKŮ	9
SEZNAM SYMBOLŮ A ZKRATEK.....	11
ÚVOD.....	12
1 MODELOVÁ SÍŤ	13
1.1 STAVBA MODELOVÉ SÍTĚ.....	14
1.2 STAVBA GEOMETRIE.....	15
1.3 MODELOVÁNÍ GEOMETRIE.....	16
1.4 TRIANGULACE	16
2 CO JE TO GIS	18
2.1 GEOGRAFICKÁ DATA	19
2.2 REPREZENTACE DAT V GIS.....	19
2.2.1 Rastrová data.....	19
2.2.2 Vektorová data.....	21
2.3 JAK JSOU DATA V GIS UKLÁDÁNA.....	23
2.4 RELAČNÍ DATABÁZE	24
2.4.1 Klíčové atributy	24
2.4.2 Polohou.....	25
2.4.3 Zařazení do třídy geoprvků.....	25
2.4.4 Ostatní atributy.....	25
3 CO JE TO GMSH.....	26
3.1 FORMÁT *.GEO PRO GMSH.....	27
3.2 OBJEMY V GMSH	28
4 VÝBĚR PROGRAMU PRO DALŠÍ PRÁCI.....	29
4.1 VYBRANÝ PROGRAM 3DSMAX.....	29
4.2 CO JE TO MAXSCRIPT , HISTORIE A VÝVOJ	30
4.2.1 Proč byl maxscript vimplementován.....	31
4.2.2 Časové hledisko	31
4.2.3 Funkční hledisko.....	31
4.3 UŽIVATELSKÉ ROZHRANÍ MAXSCRIPTU.....	32
4.3.1 Maxscript Listener.....	32

4.3.2 Maxscript editor.....	32
4.3.3 Visual Editor.....	33
4.3.4 Maxscript help	33
5 STAVBA GEOMETRIE V PROSTŘEDÍ 3DS MAX.....	34
5.1 OBJEMY V 3DSMAX	35
6 POPIS PRAKTICKÉHO ŘEŠENÍ ZPRACOVÁNÍ A VIZUALIZACE DAT.....	36
6.1 IMPLEMENTACE DOTNET PRVKŮ VE SCRIPTU	36
6.2 ZDROJOVÁ DATA A POŽADAVKY NA PŘEVOD.....	36
6.2.1 První zdrojový soubor.....	37
6.2.2 Druhý zdrojový soubor	37
6.3 MOŽNOSTI NAČÍTANÝCH FORMÁTŮ DO APLIKACE	39
6.4 PRÁCE S DATY	40
6.5 ČÍM SE ŘÍDÍ SPRÁVNÉ ZOBRAZENÍ POLYGONU V 3DS MAX	41
6.6 PROBLEMATIKA TVOŘENÍ ELEMENTŮ POLYGONU V 3DS MAX.....	42
6.6.1 Převedení bodové reprezentace převáděné geometrie.....	43
6.6.2 Převedení do polygonové reprezentace	43
6.7 METODIKA ŘÍZENÍ NORMÁL POLYGONŮ	45
6.7.1 Co jsou to normály a možnosti řízení	45
6.8 METODIKA ŘEŠENÍ SPRÁVNÉHO ZOBRAZENÍ POLYGONU.....	46
6.8.1 Oprava sítě	50
7 STAVBA OBJEMU V 3DSMAX.....	52
8 PRAKTICKÁ REALIZACE EXPORTU DO GMSH	54
8.1 PODMÍNKY NUTNÉ PRO EXPORT DO GMSH	55
8.2 ROZDĚLENÍ ŘEŠENÍ EXPORTU GEOMETRIE DO GMSH	56
8.3 POPIS ALGORITMU PRO EXPORT PLOCH.....	56
8.3.1 Popis převodu bodů.....	56
8.3.2 Popis převodu linií.....	56
8.3.3 Popis převodu Line Loop a Plane Surface	57
8.4 POPIS ALGORITMU PRO OBJEMOVÝ MODEL	61
8.4.1 Nástin zápisu správných prvků	64
9 MANUÁL VYTVOŘENÉ APLIKACE	66
9.1 ZPŘÍSTUPNĚNÍ SCRIPTU APLIKACE	66
9.2 JEDNOTLIVÉ ROLETKY SCRIPTU	67
9.3 POSOUVÁNÍ UVNITŘ ROLETEK	67
9.4 ROLLETKA - ZDROJE DAT	68
9.5 ROLLETKA – ÚPRAVA A MATERIÁL SÍTĚ.....	69
9.5.1 Objekty scény.....	69

9.5.2 Aktivace pravé části obrazovky.....	70
9.5.3 Barvy podskupin polygonů geometrie.....	70
9.5.4 Uzavření geometrie.....	72
9.6 ROLLETKA – VRSTEVNICE	74
9.7 ROLETKA - ŘEZ OBJEKTU	75
9.7.1 Auto řezná plocha.....	76
9.7.2 Vytvořit vlastní plochu.....	78
9.8 ROLETKA – EXPORT DAT PRO GMSH	80
9.9 VYTVOŘENÍ OBJEMŮ	82
9.10 EXPORT DEFINOVANÝCH OBJEMŮ DO *.GEO	83
10 VÝSLEDKY MĚŘENÍ KLÍČOVÝCH UDÁLOSTÍ.....	84
11 ZÁVĚR.....	87
SEZNAM POUŽITÉ LITERATURY	88
PŘÍLOHA – A: PŘEVEDENÝ ZÁKLAD SÍTĚ Z *.DBF.....	90
PŘÍLOHA – B: UKÁZKY MODELOVÝCH SÍTÍ.....	91
PŘÍLOHA – C: UKÁZKY MODELOVÝCH SÍTÍ.....	92
PŘÍLOHA – D: UKÁZKY POVRCHŮ	93

Seznam obrázků

Obr.1.1: Ukázka zobrazení modelové sítě.	14
Obr.1.2: Zobrazení triangulace povrchu se započítáním tektonických linií.	17
Obr.2.1: Reprezentace rastru a vektoru.	23
Obr.3.1: Prostředí Gmsh se zobrazením sítě.	26
Obr.3.2: Jednotlivé elementy geometrie v programu Gmsh.	27
Obr. 3.3: Renderovaná ukázka reprezentace objemu v Gmsh.	28
Obr.5.1: Ukázka polygonu a jeho jednotlivé elementy (body, line a ploška).	34
Obr.5.2: Renderovaná ukázka reprezentace objemu v 3ds Max	35
Obr.6.1: První zdrojový soubor s ukázkou dat x,y,z souřadnice bodů.	37
Obr.6.2: Ukázka druhého souboru. Identifikace polygonu pomocí bodů.	38
Obr.6.3: Možné pořadí pro tvorbu polygonů.	41
Obr.6.4: Zobrazení bodové a polygonové reprezentace geometrie.	44
Obr.6.5: Příklad nevytvoření a otočení polygonu v geometrii.	44
Obr.6.6: Zobrazení normál po ploškách jednotlivých polygonů	45
Obr.6.7: Zobrazení chyb v síti (ohrazená oblast červenou barvou).	47
Obr.6.8: Vývojový diagram tvorby polygonů v geometrii.	48
Obr.6.9: Vývojový diagram tvorby sítě.	49
Obr.6.10: Výběr oblasti se stejnou orientací, první a druhá iterace.	50
Obr.6.11: Vývojový diagram pro opravu sítě.	51
Obr.8.1: Příklad řešení pro vytvoření jednoho objemu v Gmsh.	54
Obr.8.2: Vývojový diagram první část exportu pro definic bodů a linií.	59
Obr.8.3: Vývojový diagram pro řazení linií a Line Loop.	60
Obr. 8.4: Příklad zobrazení prvků, které jsou pro sjednocování hledány.	61
Obr. 8.5: Sestavení základních bloků scriptu pro analýzu modelové sítě.	63
Tabulka 8.6: Přepisy bodů, linií a ploch modelové sítě.	65
Obr. 8.7: Přepočítaný objem odpovídající *.geo souboru.	65
Obr.9.1: Zpřístupnění scriptu.	66
Obr.9.2: Jednotlivé rollouty aplikace.	67
Obr.9.3: Roletka Zdroje dat a záložka Zdroj polygonu s jejím obsahem.	68
Obr.9.4: Levá část roletky Úprava a materiál sítě, s objekty sítě, světla	69
Obr.9.5: Záložka barvení podskupin polygonu.	71

Obr.9.6: a) Záložka Uzavření geometrie s nastavením.	73
b) Importovaný model 2,5D.	73
c) Špatné použití záložky.	73
d) Nezarovnané plochy modelu.	73
e) Korektní uzavření.	73
Obr.9.7: Nastavení roletky Vrstevnice a příklad výsledku.	75
Obr.9.8: Roletka řezné roviny s nastavením metody Auto řezná plocha.	76
Obr.9.9: Řezná rovinná plocha první metody řezu pro daný model.	77
Obr.9.10: Řezná rovina příčného a cyklického vybrání hran a ploch.	79
Obr.9.11: Možný výsledek řezání modelu.	80
Obr.9.12: Zobrazení roletky pro export modelu do programu Gmsh.	81
Obr.9.13: Příklad výsledku exportu z 3dsmax do Gmsh.	81
Obr.9.14: Přeložený script starající se o převod modelové sítě.	83
Tab.10.1: Zobrazení konfigurací prvního testovaného počítače.	84
Tab.10.2: Tabulka výsledů pro první konfiguraci.	84
Tab.10.3: Konfigurace druhého testovaného počítače.	85
Tab.10.4: Výsledky pro druhou konfiguraci.	85

Seznam symbolů a zkratek

GIS – Geografický informační systém (Geographical information system)

2,5D – trojrozměrná reprezentace modelu v GIS s pouze jednou interpolací Z souřadnice

3D – plně trojrozměrná reprezentace

GPS – Global position system

ASCII – kódová tabulka znaků (znaková sada)

TIN – Triangulated irregular network – nepravidelná trojúhelníková struktura

*.xls – formát pro Microsoft office 2003

*.dbf – databázový formát dBase

TIFF- Tagget image file format (obrazový formát rastru)

GPS - Global Positioning System

SQL – Dotazovací jazyk pro relační databáze

*.ms – formát scriptového souboru pro maxscript

Úvod

Tato práce se zabývá možnostmi realizace výstavby a zpracování modelových sítí, jejichž základem jsou reálně naměřená data konkrétního území, která jsou zpracována do speciální struktury v GIS (Geografický informační systém).

V rámci výstavby modelových sítí je hledán vhodný pracovní postup efektivnosti, kdy na základě dat uložených v GIS je nutné vytvořit soubor geometrie modelové sítě a soubor sítě, která budou obsahovat všechna důležitá data.

Jelikož není možné modelovou síť vytvořit přímo v GIS, zejména proto, že takovéto systémy umožňují jen takzvané 2,5D modelování, kdy není možné přiřadit jednomu bodu se souřadnicemi $[x, y]$ více jak jednu hodnotu interpolovaného jevu (z-souřadnice), vytvářejí se tak pouze objemově neuzavřené vrstvy, takže je nemožné vytvořit plné 3d, v případě modelové sítě tedy objemovou strukturu, která je nutná pro tvorbu modelové sítě.

Je tedy nezbytné data reprezentující základní strukturu modelové sítě dále zpracovat, nejlépe za účasti aplikace, která by umožňovala nejen import dat z GIS a jejich úpravu, ale i vizualizaci, umožnění výstavby modelové sítě a hlavně následný export do formátu pro program Gmsh, který byl na katedře NTI Liberecké univerzity vybrán proto, že nejvíce vyhovoval kritériím pro tvorbu sítí. Praktickým výsledkem práce bude tedy aplikace, která bude vyhovovat zmíněným podmínkám.

Výsledná modelová síť, která má požadované vlastnosti, podle předem stanoveného účelu, ke kterému má sloužit, je složena z několika různých dílčích prvků a využívá se společně s dalšími prostředky k provádění simulací zejména v úlohách proudění a transportu tekutin v horninovém prostředí, kdy na jejich základě lze například zjistit, za jak dlouho tekutina pronikne daným prostředím. Zkoumají se tedy sanační vlastnosti různých geologických struktur v reálných územích.

1 Modelová síť

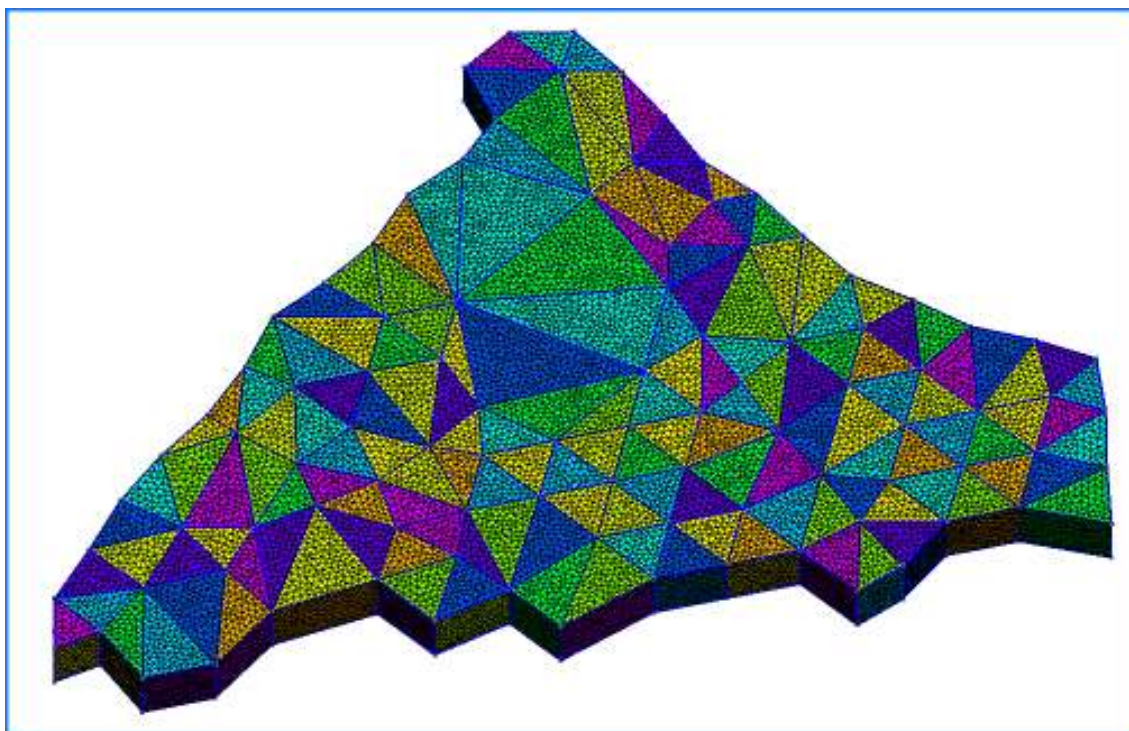
Tato práce se zabývá zpracováním dat a procesem vytváření modelových sítí, které jsou vytvářeny na základě reálně naměřených dat reprezentující konkrétní geografické území. Nejdříve je však nutné vysvětlit pojmy, které tuto problematiku provázejí.

Modelová síť je geoinformatický model, který se vyznačuje speciální geometrií a uspořádáním dat, kdy je tato geometrie tvořena konečným počtem dvourozměrných a trojrozměrných prvků v prostoru, které pokrývají území podle stanovených kritérií a v daném prostorovém rozlišení, kdy je modelová síť vždy konkrétním modelem území s jasně definovaným účelem [13].

Příkladem použití, modelových sítí v praxi je například úloha proudění a transportu v horninovém prostředí, kdy jsou tyto úlohy modelovány právě metodou konečných prvků a vychází z konkrétního modelované oblasti to na základě zpracování reálných dat daného území [15] [11].

Při procházení procesem tvorby je hledáno řešení, jak co nejefektivněji, tedy co možná nejrychleji nebo pohodlněji na základě zpracovávaných dat z geografického informačního systému vytvořit soubor geometrie a modelové sítě a soubor sítě, které nesou informace potřebné pro budoucí modelové výpočty právě pro výše zmiňované úlohy.

Základem pro tvorbu modelové sítě jsou jak už bylo zmíněno reálná data, přičemž je důležité, že každá takováto síť je tvořena pro konkrétní účel, kdy má za úkol postihnout důležité vlastnosti, které jsou pro danou oblast charakteristické, zde je vždy kladen důraz na to, aby výstavba v každém kroku zachovávala návaznost na reálná data z geografického informačního systému.



Obr.1.1: Ukázka zobrazení modelové sítě.

1.1 Stavba modelové sítě

Tvorba modelové sítě prochází určitým procesem, který se dá rozdělit do několika úkonů, které vedou k požadovanému výsledku.

- Prvním úkonem je tvorba geoinformačního systému pro konkrétní modelované území, kdy vzorem pro modelovou síť není území samo, ale jeho model, který je zobrazením právě daného systému přesně určeného a aplikovaného na daný objekt [15].
- Následuje stanovení jasných podmínek, které bude modelová síť splňovat. Do toho spadají odpovědi na otázky, které jsou spojeny zejména s geometrií (velikost celé sítě, hustota sítě nebo konečný počet prvků). Dále se sem řadí požadavky na výběr dalších prvků a jevů, které jsou pro danou modelovou síť vzhledem k účelu nezbytné (geografické, geologické nebo hydrogeologické), kdy se může jednat například o zahrnutí tektoniky, vodních toků a podobně.
- Třetí krok souvisí s předzpracováním dat z geografického informačního systému do vhodného formátu. Výběr formátu vždy závisí na modelovacích nástrojích vhodných pro další práci. Přistupuje se také k vývoji potřebných aplikací, která daný formát dokáží zpracovávat.

- Následuje stavba geometrie modelové sítě. V procesu tvorby je možné vytvářet geometrii v různých obměnách v souvislosti na stanoveném účelu a použití modelové sítě, kdy se liší svými vlastnostmi.
- Předposledním krokem je vytvoření souboru konkrétní modelové sítě, která je jak už bylo zmíněno tvořena množinou dvourozměrných a trojrozměrných prvků v daném rozlišení, prvky jsou přesně definovány v prostoru. Dále se odlišují svou topologií a tvarem podle předem stanovených podmínek a popřípadě dalšími vlastnostmi [15].

1.2 Stavba geometrie

Jedním z problémů, které vyvstávají při výstavbě geometrie a modelových sítí je efektivita, kdy je nutné geometrii z dat v GIS (Geografický informační systém) vytvořit. Tvorba je z pravidla velmi zdlouhavá, kdy se provádí zpravidla ručně, navíc pokud se přihodí, že výsledná modelová síť nesplňuje všechny stanovené požadavky, pak se musí vytvářet znovu, což celý proces prodlouží. Proto se hledají možnosti, jak celý proces urychlit, v nejlepším případě zautomatizovat. Podle jsou uvedeny a vysvětleny metody pro efektivní předzpracování potřebných dat v GIS.

Převod těchto dat je následně realizován na základě dalších metod a speciálně k tomu vytvořených aplikací, které výsledek dokáží zapsat do formátu *.geo, což je formát pro program Gmsh, ve kterém je možná modelovou sítí vytvořit podle konkrétních požadavků, tato modelová síť je následně využita pro modelování proudění a transportu tekutin.

1.3 Modelování geometrie

Jak už bylo naznačeno, vše začíná tvorbou geoinformačního systému a báze dat. V systému jsou definovány prvky reálného světa s jejich zásadními vztahy, které jsou relevantní pro účely tvořeného modelu, přitom zásadní roli hrají podmínky kladené na tento model, které musejí být splněny, aby zůstala zachována návaznost na realitu.

Předzpracování dat, které je nutné pro tvorbu geometrie je využíváno nahrazení relevantních vlastností množinami bodů v souladu s rozlišením [14]. Výsledkem zpracování je vrstva trojúhelníků, kdy se body pomocí metod triangulace pospojují do právě takové struktury. Tato vrstva je pak základem pro budoucí geometrii, která je exportována do formátu dbf, kdy struktura v tomto formátu je popsána podle předem daných podmínek, která jsou pro správnou reprezentaci, možnou vizualizaci a další použití nezbytná. Tento formát je pak pomocí speciálních aplikací zpracováván a z této struktury geometrie vytváříme modelovou síť, kterou bude možné zobrazit v programu Gmsh (viz. kapitola 3).

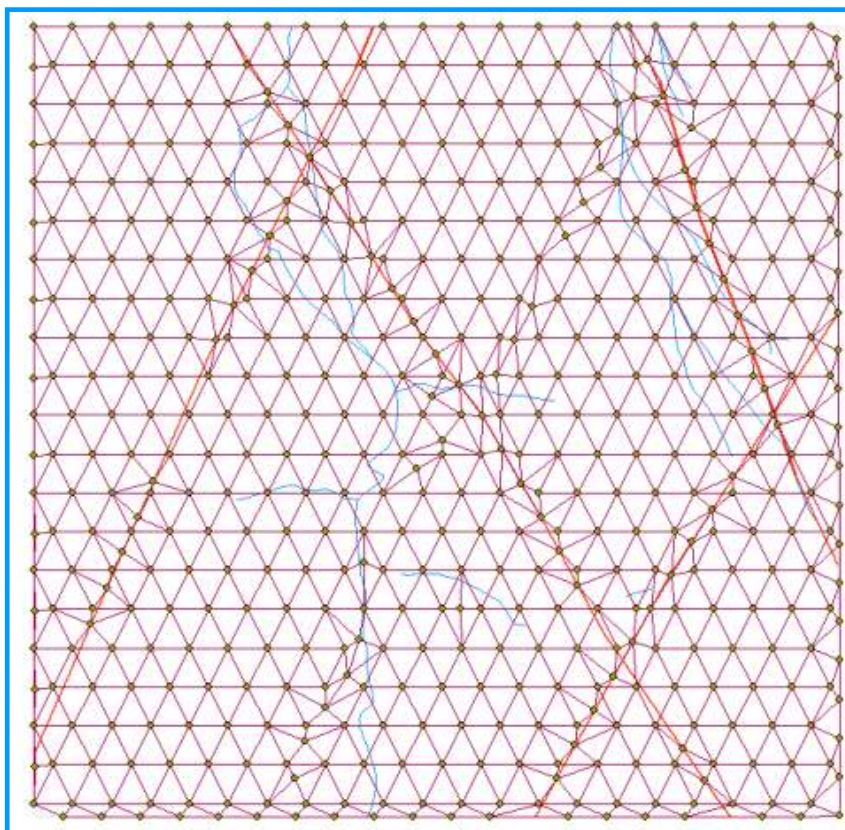
1.4 Triangulace

V předcházející kapitole bylo uvedeno, že základem celé geometrie je vrstva trojúhelníků. Jedná se o vektorovou strukturu, která je v GIS systémech využívána k vyjádření povrchové morfologie krajiny a je tedy situována do x,y,z souřadnic. Tato struktura je založena na vektorizaci digitálních zeměpisných dat a je zkonstruována pomocí metod triangulace do konečného množství bodů/uzlů které mají své prostorové souřadnice, uzly jsou navzájem spojeny množstvím hran tak, aby bylo dosaženo trojúhelníkové sítě, z nichž se trojúhelníky navzájem nepřecházejí. Při sestavování takovéto struktury je používáno množství různých metod interpolace. Jednou z těchto metod je tzv. Delaunayho triangulace celá dokumentace k této problematice je rozebrána v Diplomové práci pana Zábranského [12] nebo [5].

V zásadě existují dvě metody triangulace, jednou z nich je tzv. TIN z anglického triangulated irregular network (nepravidelná trojúhelníková struktura), která klade důraz na morfologii, znamená to, že v oblastech, kde se výška povrchu příliš nemění, algoritmus který vytváří trojúhelníkovou strukturu vytváří větší trojúhelníky (mezery mezi jednotlivými uzly jsou větší a spojující hrany jsou delší), protože v daném místě nejsou tak markantní výškové změny a není tak potřeba velká míra interpolace, pokud jsou však oblasti výškových změn naopak velké, jednotlivé uzly si jsou blíže a hrany

mezi nimi se zkracují, takže vzniklá síť trojúhelníků je v daném místě hustší aby postihla všechny změny v morfologii, což mohou být hory, propasti, údolí a jiné krajinné změny v dané oblasti.

Druhá metoda, je pro tuto práci zásadnější, protože právě takovou strukturu měla poskytnutá data, která byla k dispozici pro vytvoření aplikace pro výstavbu modelových sítí. Vytvářená trojúhelníková struktura je v tomto případě co možná nejrovnoměrnější. Důležitým bodem při triangulování povrchu je, že pokud chceme, aby se algoritmus převodu zabýval nějakou konkrétní oblastí detailněji, (v naprosté většině případů) je nutné, aby byly také zahrnuty body popřípadě přímo čáry ležící na těchto důležitých místech terénní kostry (terénní zlomy, tektonické linie, rozsedliny atd.), kdy tak přímo definujeme důležité oblasti a algoritmus vezme při vytváření trojúhelníkové vrstvy tyto body/čáry v úvahu [14], přičemž strany trojúhelníkových elementů pak leží na hranicích území, tedy např. na tektonických liniích nebo hydrogeologických oblastech. Problematika zpracování je rozepsána v [15].



Obr.1.2: Zobrazení triangulace povrchu se započítáním tektonických linií.
(Převzato z časopisu studia.OECOLOGICA č.3 1.11. 2010)

2 Co je to GIS

Pojem GIS je zkratka pro termín Geografický Informační Systém z anglického Geographic Information System.

GIS je podle definice ESRI

„GIS je organizovaný soubor počítačového hardware, software a geografických údajů (naplněné báze dat) navržený pro efektivní získávání, ukládání, upravování, obhospodařování, analyzování a zobrazování všech forem geografických informací..“

Ačkoliv GIS můžeme pojmout také podle definice GRASS jako

„Geografický informační systém (GIS) představuje obsáhlou sadu nástrojů pro sběr, ukládání, zpracování, transformaci a zobrazování prostorových dat reálného světa. Přitom jsou tyto digitálně kódované jevy a objekty reálného světa primárně uloženy podle své geografické polohy a lze je tedy umístit do vzájemného vztahu.“

V takovýchto systémech lze provádět nejrůznější operace, pomocí nichž získáváme další relevantní data pro další činnost, data tedy nejen ukládáme, ale dále zpracováváme a třídíme, abychom se dostali ke kýženému výsledku. Lze provádět analýzy nejrůznějšího charakteru popsanych v [1] a [4] v závislosti na tom, jaké vztahy jsou mezi jednotlivými objekty. Důležité je říci, že při práci v těchto systémech se prochází napříč nejrůznějším spektrem úkonů, z toho vyplývá že zpracování dat od začátku práce až po jeho konečný výsledek prochází procesem, jehož průběh se liší podle toho jaký výsledek je pro nás v danou chvíli rozhodující [3].

Geoinformační softwary pracují v takzvaném 2,5D prostředí. Každý bod tvořící základní část modelu je v takovémto prostředí definován souřadnicemi X a Y (vytvářejí tedy plošný obraz). Souřadnice Z je v těchto geoinformačních modelech poněkud problematická, neumožňuje jednomu bodu o souřadnicích X a Y dvě hodnoty interpolovaného jevu např. výšky souřadnice Z. Nejde tedy vytvářet klasický objemový model tak jak je obvyklé, ale můžeme vytvářet pouze jakési vrstvy, proto tedy 2,5D, proto se hledají cesty, jak vytvořit model objemově plně ve 3d.

2.1 Geografická data

V zásadě existují dva druhy geografických dat, které GIS využívá. Jsou to data prostorová a popisná.

Prostorová data obsahují v první řadě informace o pozici daného prvku v prostoru (dále to jsou informace o tvaru nebo geometrické reprezentaci) podrobně rozebráno v [6]. Popisná data nebo atributy jsou přiřazovány prostorovým datům a vyjadřují nejrůznější informace o daném prvku. Ke každému prvku nebo skupině prvků může být přiřazeno nejrůznější množství atributů. Atributem může být například typ prvku (hornina, vodstvo, město atd.), podtyp prvku (řeka, jezero v případě vodstva a žula, pískovec v případě typu horniny). Tyto příklady jsou pouze ukázkou, ve skutečnosti lze přiřadit jako atribut jakoukoli relevantní vlastnost daného prvku, která je z hlediska geografie a dané úlohy, která je zpracovávána důležitá.

2.2 Reprezentace dat v GIS

Základní reprezentace geografických dat, se kterými GIS systémy pracují jsou:

- Rastrová data
- Vektorová data

Tato data jsou principiálně velmi rozdílná tudíž se liší nejen svými vlastnostmi ale i použitím. Následující text osvětluje nejdůležitější rozdíly obou těchto typů [5],[6].

2.2.1 Rastrová data

Rastrová nebo také v souvislosti v GIS grid data, je druh obrazových dat která jsou rozdělena mřížkou a vytvářejí tedy jakousi matici.

Nejmenší jednotkou informace je buňka/pixel, která nese datovou informaci.

O každé konkrétní buňce v matici lze jednoznačně určit kde se uvnitř matice nachází, každá buňka uvnitř matice nese kromě informace o svojí poloze ještě tzv. atributovou informaci, což je v barva dané buňky, která reprezentuje nějakou číselnou hodnotu, díky hodnotě uvnitř této buňky jí lze přiřadit nějaký konkrétní význam.

Další důležitou informací je velikost dané buňky/pixelu a samozřejmě velikost celé matice, tedy rozměr počtu řádků a počtu sloupců tedy rozliší rastru.

V případě rastrových dat je důležitou vlastností to, že obrazová data nejsou parametrická, to znamená, že neexistují vztahy mezi jednotlivými pixely a tvary v obrazu. V praxi se jedná o to, že se změnou velikosti obrazu rastru, který bychom

uložili a poté chtěli obrázek dostat opět do původního stavu, už se nám to nemůže podařit právě z důvodu absence vztahů mezi jednotlivými tvary v obraze.

Rastry se dají využít pro některé typy analýz, pro tvorbu určitých modelů, pro mapování atp. Nejčastějšími formáty rastrových dat jsou především soubory typu JPEG nebo TIFF, které dávají výsledky z hlediska kvality a obrazové informace. Příklady použití pro rastrové podklady jsou uvedeny např. v [7].

2.2.2 Vektorová data

Nejdůležitější vlastností vektorových dat je, že existují vzájemné matematické vztahy mezi entitami v obraze, je tedy možné s obrazem pracovat, aniž by vlivem transformace velikosti obrazu docházelo k datovým ztrátám.

V případě vektorových dat je základním typem informace bod, který je dán souřadnicemi x a y . Druhým typem dat jsou linie, ty jsou tvořeny body, které jsou spojeny čarami. Třetím druhem dat jsou plochy které jsou dány hranicí které jsou tvořeny liniemi, kdy se dá za hranici označit linie, které jsou cyklicky uzavřeny, výplně daných objektů jsou dány plochami. Vektorová data se získávají v podstatě dvojím způsobem. Prvním způsobem je automatická vektorizace rastrových map, kdy se o vektorizaci postará počítač. Výhodou je rychlost vektorizace, i když vždy velice záleží na počátečních podmínkách procesu a složitosti rastrových dat. Nevýhodou je v některých případech menší přesnost a hlavně také že nemáme přímou kontrolu nad celým procesem, to znamená, že se vektorizují i obrazové prvky, které pro nás v danou chvíli nemusejí být relevantní. V takovém případě je nutné vždy zasáhnout.

Druhým způsobem vektorizace je přímo uživatelem. Kdy všechny prvky, které vás v danou chvíli zajímají tvoříte ručně. Nevýhodou v tomto případě je větší časová náročnost.

V GIS systémech jak už bylo naznačeno máme v podstatě tři základní typy vektorových dat. Která se zobrazují jako jednotlivé oddělené vrstvy. Jsou to data:

- Bodová
- Liniová
- Polygonová

Každý typ těchto dat má svoje opodstatnění, dá se říci, že bez těchto typů dat by GIS nemohl existovat.

Základními informacemi, kterými disponují všechny typy dat bez rozdílu jsou informace o pozici každého objektu. Tyto informace jsou naprosto nezbytné pro každý objekt, protože GIS systémy pracují v určitém souřadnicovém systému a jelikož pracujeme s geografickými aplikacemi, je tedy jasné že každý objekt bude mít svou jasně definovanou pozici, příklady využití pro vektorové podklady jsou uvedeny např.v [7].

Bodová

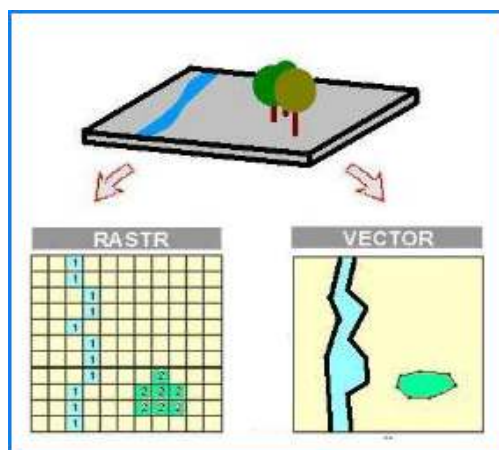
Bodová data se používají jako výrazový prostředek k určení objektů v krajině. Každý takový bod má svoji X,Y a někdy i Z souřadnici (záleží na tom v jakém souřadnicovém systému právě pracujeme). Můžeme tak ve vytvářených mapách zobrazovat některé vlastnosti dané krajiny. Vzhledem k tomu, že jsou to data bodová, používají se pro zaměření většinou malých objektů, ale není to vždy pravidlem. Vždy totiž záleží na tom, s jakou úrovní rozlišení se právě pracuje. Znamená to, že body můžeme označovat např. studny, prameny, vrty, pomníky, památky v případě že budeme pracovat s malým rozlišením ale i např. pozemky, budovy nebo celé oblasti kdy se pracuje s velkým rozlišením.

Liniová

Tato struktura dat, se používá jinak než tomu je u bodové reprezentace, je spojená zejména s vymezením určité oblasti, která má průběhový charakter. Hodí se dobře například k zobrazení inženýrských sítí, tedy k vymezení kabelového nebo potrubního a energetického vedení, může se jednat např. o telefonní, elektrické nebo plynové rozvody, televizní nebo internetové sítě, vodovodní a kanalizační potrubí atd. Může se dobře využít k zobrazení rychlostních komunikací, dálnic, polních cest, turistických tras, hranic pozemků nebo se může použít jako prostředek vyjadřující informace související přímo s geologickou strukturou krajinného rázu, kdy se liniová struktura dat velmi často používá pro zobrazení např. hranic vymezeného území, dále se používá pro zaměření říčních toků v krajině, různých zlomů atp., může se také použít pro zobrazení tzv. vrstevnic, což jsou křivky, které s určitými předem danými rozestupy reprezentují v mapě stejnou nadmořskou výšku.

Polygonová

Tato třetí struktura je neméně důležitým prostředkem pro definování určitých objektů v mapě nebo krajině. Používá se zejména pro vymezení určitých spojitých plošných oblastí. Příkladem takových oblastí mohou být například zalesněné části území, vodní plochy, jako jsou rybníky, přehradní nádrže nebo moře, mohou to být také i parcely nebo další vymezená území, tedy další podobné prvky, které se lépe definují plochou než liniovou nebo bodovou strukturou v dané úloze, záleží ale na tom v jakém rozlišení pracujeme, jelikož stejnou oblast v jiném rozlišení může být reprezentován i bodem.



Obr.2.1: Reprezentace rastru a vektoru.

(Přejato z webu o GIS –www.vysocina.eu/gis.asp?p1=27188)

2.3 Jak jsou data v GIS ukládána

Nejčastějším způsobem uchovávání informací v GIS systémech a to včetně ArcGIS je ukládání do databází. Databáze je v podstatě soubor dat, která jsou organizována tak, aby bylo dosaženo efektivní ukládání a dotazování k jednotlivým informacím. Kromě těchto požadavků je dále kladen vysoký důraz na spolehlivost a datovou nezávislost, tedy aby byla a ukládána data různého typu (číselné, textové) a bylo možné ukládat více formátů. Dalším požadavkem na databáze v GIS musí být vysoký výkon, vzhledem k tomu že můžeme v závislosti na úloze pracovat s velkým množstvím nejrůznějších dat, je nutné, aby zpracovávání probíhalo rychle. Nejčastějším typem databáze jsou tzv. relační databáze. Do databází lze jednotlivé záznamy vkládat, již vložené pak upravovat, mazat nebo k získání z databáze samozřejmě vybírat.

Důležité je, že v databázích se uchovávají veškeré informace o dané geometrii [6]. V případě vektorových dat jsou uchovávány informace o bodech, liniích a polygonech, tedy jejich poloha, jejich přesné vyjádření v prostoru, dále to je tvar, zařazení plus další popisné údaje. Takto uchovávaná data se pak dají efektivně využívat, třídit atd., navíc se při provádění analýz nad modelem data dají dobře zpracovávat.

2.4 Relační databáze

Základem všech relačních databází jsou tabulky, které obsahují různorodá data všemožných typů včetně číselných i textových. Každá tabulka se skládá z řádků a sloupců.

Ke každému prvku, který je vytvořen v prostředí GIS je přiřazeno v geodatabázi určité množství informací právě v tabulkách. Každý objekt je tedy popsán určitými informacemi, například svou pozicí nebo dalšími popisnými vlastnostmi. Řádky tabulky odpovídají jednotlivým záznamům libovolných prvků/objektů. V databázích se vlastnosti objektů nazývají atributy, které představují sloupce tabulky. Každý atribut je vlastně tedy pole jedné vlastnosti, které může odpovídat více objektům v tabulce. Kromě vlastních popisných informací tabulky obsahují tzv. vlastní nebo cizí klíče, které jsou nezbytné pro správný chod a udržitelnost databáze v případě že jsou správně navrženy. Tabulky jsou spolu provázány vztahy, díky nimž spolu dokáží díky speciálním dotazům komunikovat a vybírat z nich informace, které jsou pro nás v danou chvíli důležité z hlediska další práce [6]. Data lze tímto způsobem třídit a na základě toho tedy získávat další data pro nejrůznější následné analýzy.

V tabulce jsou nejčastěji data tohoto typu:

- Klíčové atributy
- Poloha
- Zařazení do třídy geoprvků
- Ostatní vlastnosti/atributy

2.4.1 Klíčové atributy

Klíčovými atributy jsou v případě relačních databází tzv. primární a cizí klíče. Primární klíč v tabulce jednoznačně identifikuje záznam, tedy řádek tabulky. Primární klíč může být v tabulce více než jednou, závisí to především na tom, zda je záznam jedním klíčem jednoznačně určen. Vlastností primárních klíčů je to, že musí obsahovat hodnotu. Nemůže se tedy stát, že by záznam nebyl identifikován. Musí být také zajištěno, aby každý klíč byl tzv. unikátní, to znamená, že v rámci tabulky není možné aby některý záznam měl stejnou hodnotu primárního klíče.

Cizí klíče plní funkci vztahovou, jedná se o vztahy mezi jednotlivými tabulkami. Pomáhají tedy k vyjádření, které tabulky spolu navzájem souvisí.

2.4.2 Polohou

Vzhledem k tomu že pracujeme především s geoprvky, které jsou v rámci určitého souřadnicového systému obsaženy v mapě nebo krajině, znamená to tedy, že mají svou jasně danou pozici, která jasně definuje přesné umístění daného prvku a to tedy ať už v krajině, mapě nebo modelu.

2.4.3 Zařazení do třídy geoprvků

Tato vlastnost je důležitá vzhledem k systému GIS. Kdy systém musí vědět v jaké třídě je daný objekt nebo prvek zařazen. Souvisí to především s operacemi, které vzhledem k jednotlivým třídám můžeme provádět. Příkladem takové třídy je například bodová, liniová nebo polygonová struktura prvku.

2.4.4 Ostatní atributy

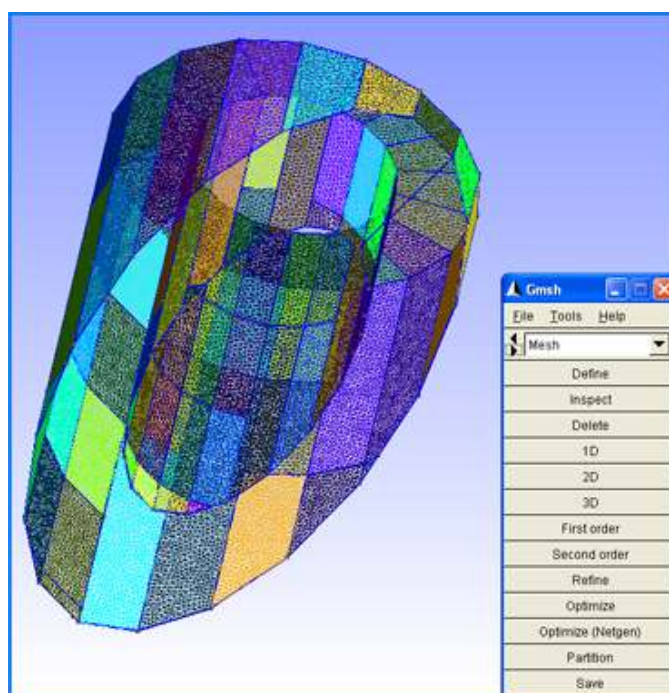
V případě této kategorie jsou pod tímto pojmem brány všechny ostatní popisné vlastnosti prvků v tabulce. Může se jednat například o poznámky o zaměření objektu, lokalitu, typ geologie nebo vodstva, chemické složení, datum měření atp. Tyto informace jsou dobré z hlediska vytváření nejrůznějších seznamů a přehledů v mapě nebo průzkumů. Například v případě pramenů v krajině můžeme ze všech vybrat jen ty, které mají jen určité chemické složení, místo pramenu kde se nachází, typu pramenu atd. a to pomocí speciálních dotazů právě do databáze a promítnout je do digitální mapy se kterou pracujeme, obarvit je jinou barvou atp.

Vybírání konkrétních dat z databáze se provádí na základě databázových dotazů. Nejčastějším jazykem pro práci s databázemi v GIS je SQL jazyk. GIS většinou pro dotazovací dotazy obsahují návod nebo průvodce, který pomáhá při sestavování správného dotazu.

3 Co je to Gmsh

Požadavkem na tuto práci bylo, aby byla daná trojúhelníková geometrie převeditelná právě do programu Gmsh a to na základě aplikace která je praktickým řešením této práce. Vytvářená aplikace tedy bude plnit úlohu prostředníka, ve které bude možné danou geometrii převést z GIS na základě dbf souborů, dále bude možné ji upravovat a originální nebo upravenou geometrii vyexportovat do tohoto programu. Podle definice je Gmsh generátor sítě konečných 3d prvků, který má zabudován post-procesor a je navržen za cílem poskytnout rychlý, lehký a uživatelsky přívětivý nástroj pro tvorbu sítí s možností parametrického vstupu a možnostmi pokročilejší vizualizace [29]. Gmsh se skládá v podstatě ze čtyř modulů: geometrie, síť, tzv. řešitelem a postprocesingem.

Každý tento modul se specifikuje a provádí pomocí klasického uživatelského rozhraní nebo je možné je specifikovat pomocí textových souborů založených na ASCII ve vlastním scriptovacím jazyce, který Gmsh obsahuje.

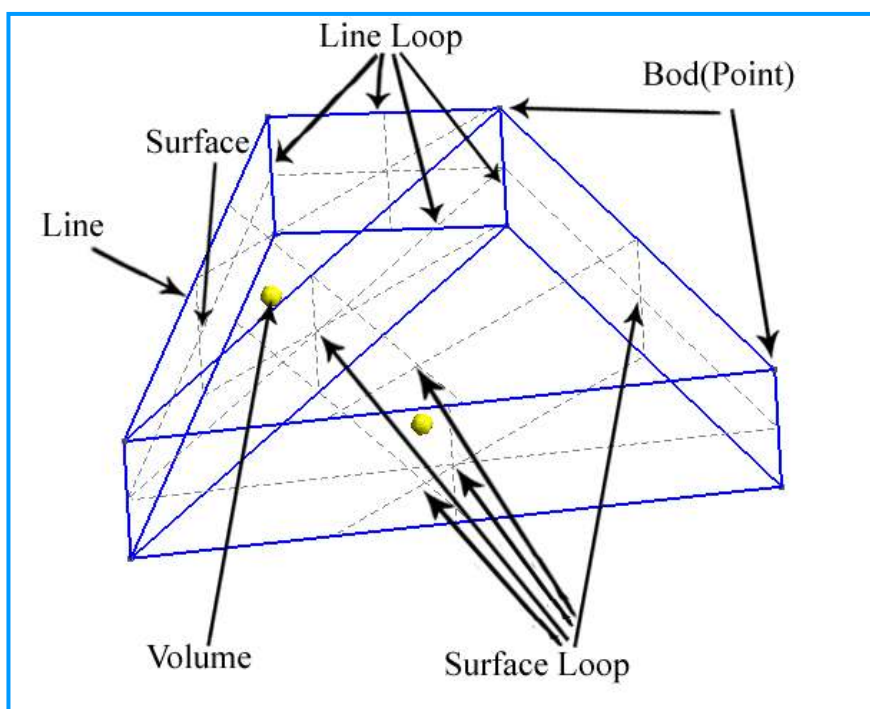


Obr.3.1: Prostředí Gmsh se zobrazením sítě.

3.1 Formát *.geo pro Gmsh

Pro převod geometrií jak už bylo zmíněno dříve je v tomto případě využíván formát *.geo, který je k tomu určen. Tento typ formátu je založen na ASCII. V Gmsh se geometrie skládá z jednotlivých bodů, linií, ploch a objemů. Kromě těchto čtyř základních prvků existují ještě další, které jsou důležité, jedná se o tzv. line loop a surface loop, které jsou jakýmsi mezistupněm mezi čtyřmi základními, nicméně jsou stejně důležité a nelze bez nich požadovanou strukturu vytvořit.

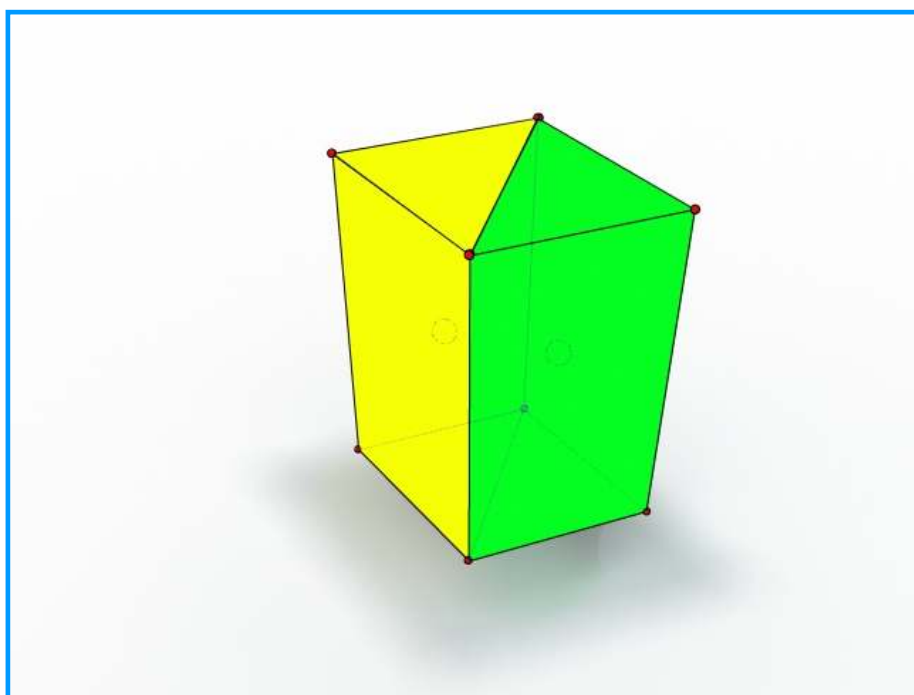
Prvním prvkem a základním kamenem jsou nejdříve definovány body geometrie, následně jsou definovány linie, které mají návaznost na tyto body, linie ohraničují třetí prvek kterým jsou povrchy geometrie.



Obr.3.2: Jednotlivé elementy geometrie v programu Gmsh.

3.2 Objemy v Gmsh

Objemy, tedy volumes jsou v programu definovány skupinou ploch (Surface Loop) viz. obr. 3.2, které mají daný objem uzavírat. Díky provedení triangulace mají trojúhelníkovou základnu, odpovídají tedy trojúhelníkové geometrii při předzpracování v GIS. Stěny trojúhelníku jsou vertikálně vytaženy podle uživatele a na konci opět uzavřeny trojúhelníkovou plochou. Důležitou vlastností seskupených objemů je, že plochy stěn, které sousedí s dalším objemem (definovaným trojbokým hranolem) jsou pro oba objemy společné a jsou tak spolu pevně spojeny.



*Obr. 3.3: Renderovaná ukázka reprezentace objemu v Gmsh.
(trojboké hranoly jsou spolu pevně spojeny)*

4 Výběr programu pro další práci

Vzhledem k tomu, že některé operace v GIS s prostorovým modelem není možné provádět. Konkrétně se jedná o vytvoření plného 3d prostorového modelu, kdy je geometrie ze všech stran obklopena polygony. GIS totiž pracují nejen s modely ve vrstvách, znamená to tedy, že se strukturou např. TIN, která reprezentuje morfologii krajiny, vytváří jen jakousi vrstvu... takováto struktura se nazývá pseudo 3d nebo také 2,5d jelikož právě není dotvořena spodní nebo jiná část dané geometrie a není tak uzavřena (nelze vytvářet objemové prvky). Kvůli tomu, že byl v této práci požadavek na to, aby byla daná data převedena dále do programu Gmsh, kde v prostředí tohoto programu lze efektivně vytvářet celé geometrie, důležité však je, že v něm lze vytvářet plné 3d a objemové prvky, hledal jsem tedy prostředek k tomu, aby se data dala převést, dále s nimi pracovat a upravovat a zároveň je i zobrazit v ucelené formě přímo v reprezentované struktuře, bylo tedy nutné, aby byl vybrán prostředek, který dokáže pracovat s 3d objekty, je možné vstupovat do tvorby geometrií a automatizovaně data zpracovávat. Program Gmsh byl na katedře NTI Liberecké univerzity vybrán proto, že nejvíce vyhovoval kritériím pro tvorbu sítí [13].

Volba se tedy zúžila na skupinu programů zabývajících se přímo modelováním ve 3d, které obsahují možnosti scriptování v daném programu.

Nakonec jsem zvolil program 3ds Max popsáném v [18], který obsahuje kromě profesionální práce ve 3d také implementovaný scriptovací jazyk. Mojí volbu také podpořil fakt, že tento program znám delší dobu a modelováním se zabývám.

4.1 Vybraný program 3DsMax

V podstatě se jedná o ucelené softwarové řešení, které se dá rozdělit na několik částí, modelování, animace, vytváření a zpracování materiálů, video postprodukce, renderování a scriptování. Toto rozdělení lze však brát jen jako rozčlenění z hlediska funkčnosti, protože se jedná o jeden kompaktní produkt [19]. Vzhledem k tomu, že tento produkt je hojně využíván nejrozličnějšími odborníky hlavně z oboru 3d grafiky, vycházejí stále nejrozličnější plug-iny pro určité specializované problémy.

Je zaměřen především na 3d modelování různorodých objektů, ať už se jedná o modelování živých nebo neživých modelů. Zvládá ovšem i modelování 2D prvků, jako jsou křivky různých typů. Tento software je tedy zaměřen na obecné modelovací

prostředí. Kromě toho se specializuje také na animování jednotlivých modelů. 3DsMax využívají hlavně 3D umělci, filmová a herní studia, protože dovoluje naprosto volnou ruku pro libovolnou tvorbu modelů, vytváření animací a videosekvencí.

4.2 Co je to Maxscript , historie a vývoj

Maxscript je jak už předcházející text naznačil scriptovací jazyk, který je přímo vimplementován do prostředí 3d modelovacího a animačního programu 3dsmax. Tento nástroj byl vytvořen za účelem lepší kontroly nad každým aspektem práce v 3d grafice.

Nebylo tomu tak ale vždy, vůbec první možnosti scriptování sahají ještě do DOS verze tohoto programu, s tímto obdobím se předchůdce programu jmenoval 3dsDos, zajímavé je, že tato možnost nebyla umožněna díky společnosti Autodesk, ale ze strany developera, v té době nic takového společnost Autodesk neplánovala.

Po změně filozofie a příchodem operačního systému Windows se stal vývoj 3d studia tajným a vydání prvního 3ds Max R1 opět žádný nástroj pro programování nástrojů opět implementován nebyl.

Vůbec první možností práce pomocí scriptování se objevila až s verzí R2, kdy bylo možné přistupovat k vlastnostem scény.

Pokud se dá mluvit o plnohodnotném scriptování v programu, jde o období vydání 3ds Max 4, kdy bylo možné přistupovat k objektům, mapám, nabídkám programu, byl vytvořen macro recorder pro záznam příkazů a možnost vytvářet macrosripty s možností vlastního nastavení kam je umístit do stávajících nabídek. Poslední novinkou bylo možnost práce s ActivX prvky.

Další vývoj maxscriptu byl zaznamenám s každou novou verzí 3dsMax přičemž za zmínku stojí například verze 8, kdy byl představen nový Debugger a verze 9 kdy byla představena podpora pro třídy, objekty, prvky a metody dotNet. Kompletní přehled historie vývoje je detailně zpracován např. v [23].

V současnosti jsou jeho schopnosti velice rozsáhlé [20]. S jeho pomocí lze zasahovat do samotného modelování objektů, tak i do dalších dílčích procesů práce, jako například animace, nastavování světel, přidávání efektů, práce s kamerou až po vytváření materiálů a renderování/vykreslování samotné finálové scény. Uživatel má jeho prostřednictvím tedy plnou kontrolu nad všemi procesy od počátku až k hotovému procesu práce.

4.2.1 Proč byl maxscript vimplementován

Maxscript byl do samotného těla programu 3dsmax vimplementován za účelem rozšíření funkčnosti už tak velmi funkčně propracovaného 3dstudia zejména proto, aby uživatel mohl zpracovávat takové procesy, které by ručním způsobem práce, byly velmi náročné na provedení a to zejména z časového a funkčního hlediska.

4.2.2 Časové hledisko

Jedná se hlavně o takový typ práce, která je v zásadě stále stejná nebo postavená na stejném základu s různými drobnými modifikacemi. Jsou to tedy hlavně stále stejné se opakující rutiny.

4.2.3 Funkční hledisko

Z hlediska funkčního je maxscript důležitý hlavně v takové případě, když prostředky a samotné uživatelské prostředí 3dsmaxu nestačí.

Využívá se hlavně v případech úloh, které jsou nějakým určitým způsobem netypické a je nutné funkce pro úspěšné dokončení práce doprogramovat.

Pro lepší představivost uvedu následující příklad:

V prostředí 3dstudia potřebujeme vytvořit například právě model krajiny, kdy data o jednotlivých prvcích modelu máme uloženy v externím souboru. Stojíme tedy před problémem jak vytvořit efektivně obrovské množství prvků, které tento model reprezentují.

Takováto úloha je hlavně z časového hlediska velmi náročná, protože vytvářet každý prvek krajiny ručním způsobem je velmi neefektivní. Právě s takovýchto případech je užitečné zvolit jako řešení problému právě maxscript, kdy můžeme pomocí cyklů vytvořit potřebné množství prvků, které jsou v souboru uloženy. Pak lze každému z těchto vytvořených modelů přiřadit libovolné množství upravujících modifikátorů, které nám pomohou model dál editovat.

4.3 Uživatelské rozhraní Maxscriptu

Při vytváření scriptů lze efektivně využívat hned několik produktových řešení, které jsou uzpůsobeny pro usnadnění práce při programování.

Všechna tato řešení jsou přístupná ve verzi 3dsmax 2009 z hlavního horního menu ze záložky Maxscript a nebo ho lze zpřístupnit z bočního tzv. Command panelu v záložce Utilities vybráním opět položky Maxscript.

4.3.1 Maxscript Listener

Listener je opticky rozčleněn do dvou částí.

Horní část okna obsahuje seznam, tedy historii předchozích vykonaných příkazů scriptu. Listener tady jak už název napovídá, naslouchá co se děje ve scéně a veškeré změny které jsou provedeny automaticky přepisuje do maxscript příkazů. Této vlastnosti lze velmi dobře využít jako nápovědy při sestavování kódu. Pokud neznáme přesnou syntaxi nějakého příkazu, můžeme využít právě Maxscript Listeneru, a to tak že ve scéně vytvoříme cílovou vlastnost kterou potřebujeme přepsat a přidat přímo do Maxscriptu, horní okno horní okno listeneru přepíše děj na scéně právě do cílových příkazů a to se správnou syntaxí. Této vlastnosti lze však využít jen v tom případě, že máme zapnutý Macro Recorder, což je vlastnost právě Listeneru, která zajišťuje právě zaznamenávání činností které se odehrávají ve scéně.

Spodní část Maxscript Listeneru obsahuje výpis chyb v syntaxi, ale i informace o všech modifikátorech a jeho vlastnostech, komponentách, objektech ve scéně a podobně. Stačí jen napsat klíčové slovo hledané komponenty nebo modifikátoru ve správném tvaru, které jsou v maxscriptu dostupné. V této části lze také výpisy na obrazovku pomocí příkazu print.

4.3.2 Maxscript editor

Maxscript editor plní hlavní součást řešení, protože do tohoto okna se zapisuje kód samotného maxscriptu vytvářené aplikace. Kód aplikace se překládá pomocí menu Tools a výběru položky Evaluate All. Syntaxe základních programátorských prvků, tedy cyklů a podmínek je velmi podobná syntaxi jiných jazyků.

4.3.3 Visual Editor

Tato miniaplikace je určena pro skládání komponent pro připravovaný script. Lze vkládat tlačítka, rollouty a ostatní kompilované komponenty, vše je podobné jako například vkládání komponent v Delphi. Takto lze připravit návrh kostry po script, který budeme vkládat později. Vkládání tímto způsobem však není podmínkou, všechny komponenty lze nadefinovat samozřejmě i ručně přímo v Maxscript Editoru. Podle mého názoru má však Visual Editor jednu obrovskou nevýhodu, můžeme v něm pracovat bohužel pouze s komponentami, které jsou v 3dsmaxu kompilovány. Jelikož maxscript dokáže pracovat také s komponenty, třídami a objekty dotNetu a Visual Editor s nimi pracovat nedokáže, je nutné všechny dotNet prvky psát do scriptu ručně, tedy tak že vše je vytvářeno za běhu. Není tedy zajištěna taková přehlednost jako v případě předkompilovaných komponent. Důvodem podpory dotNetu je jeho obrovská variabilita, obsahuje obrovské množství komponent a tříd, lze s nimi tedy komfortněji pracovat, obsahuje také mnohem více událostí jednotlivých komponent [26].

4.3.4 Maxscript help

Nápověda pro Maxscript obsažená v programu je velmi přehledná. Všechny oblasti scriptu od modelování, animaci až po rendering. Všechny metody, vlastnosti a objekty jsou uspořádány podle logických souvislostí a velmi detailně popsány. V samotné nápovědě jsou umístěny i jednotlivé užitečné příklady řešení nebo nástin jednotlivých problematik včetně vyobrazení. Vyhledávat lze buď podle klíčových slov, jednotlivých kategorií a nebo podle rejstříku. V případě důkladnějšího prostudování je možné shlédnout sérii videotutoriálů [24].

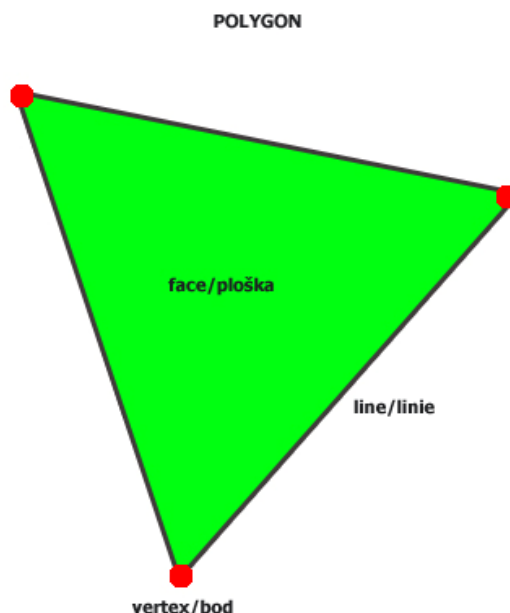
5 Stavba geometrie v prostředí 3ds Max

Jelikož převádíme geometrii, která je tvořena trojúhelníkovou sítí, pak budou jednotlivé polygony převáděné geometrie trojúhelníkové. Podobně jako v GIS se polygon skládá z bodů, hran a plošek.

Prvním elementem polygonu je vertex/bod, každý z těchto bodů tvoří vrchol polygonu (nejmenší počet bodů jsou tedy 3, což odpovídá trojúhelníku).

Druhým elementem polygonu je hrana, každá hrana spojuje body polygonu a ohraničuje polygon (nejmenší počet hran v polygonu jsou opět 3).

Třetím elementem polygonu je ploška/face, ta je uprostřed ohraničované oblasti hran a vyplňuje tak tento prostor.



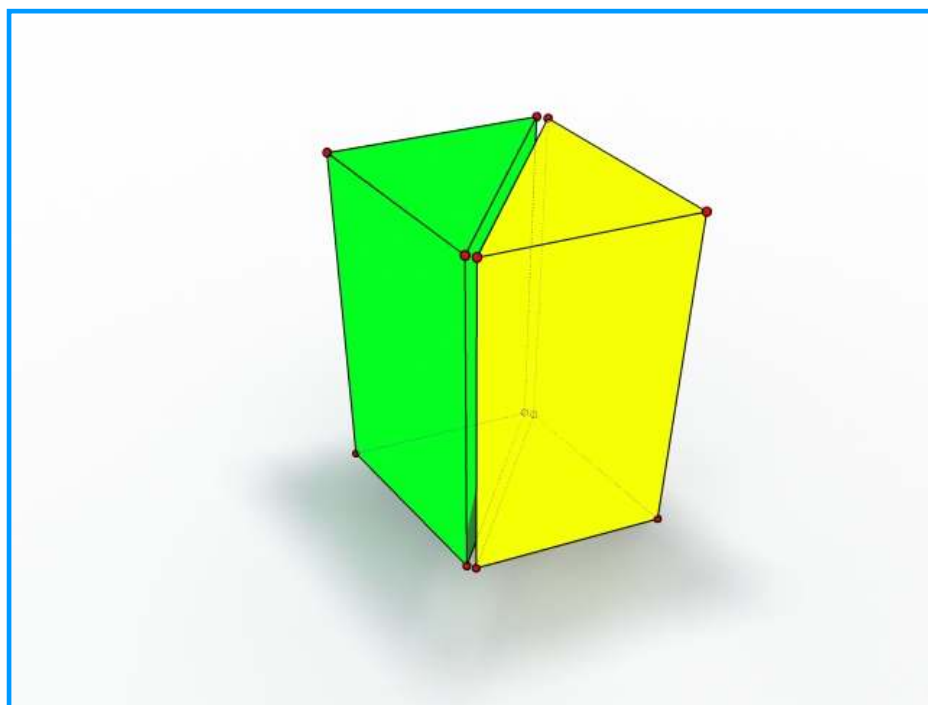
Obr.5.1: Ukázka polygonu a jeho jednotlivé elementy (body, line a ploška).

Každý bod má jasně stanovené prostorové souřadnice, tedy souřadnice X, Y a Z, přičemž každý takový bod/vertex má svoje číslo, které bod jednoznačně identifikují v rámci modelu. Stejná logika identifikace probíhá v rámci jednotlivých linií, u těch se definují body, které linii tvoří. Každá linie, jak je vidět podle obrázku je tvořena dvěma body, počátečním a konečným. Důležité je zmínit, že editace jednotlivých čísel elementů není možná, o číslování se stará jádro programu. Číslování začíná od čísla 1. Problematika stavby je rozebrána například v [19],[21],[22].

5.1 Objemy v 3dsMax

Objem je v programu reprezentován podobně jako v gmsch uzavřenou skupinou polygonů, nicméně zde podobnost také končí, protože obě struktury si jsou sice podobné ale odlišně definované. V gmsch se dá struktura objemu vyplnit trojúhelníkovou strukturou, naopak v 3dsMax toto není možné, zde je dutý prostor. Stavba objemů je v obou programech také odlišná, nelze totiž objemy reprezentované uzavírajícími plochami seskupit tak přímo jako v Gmsch, protože pravidla seskupování sítí neumožňují svařit (sjednotit) body takto uzavřených celků. To však není problém pouze 3dsMaxu, ale také např. AutoCadu a dalších Cad systémů.

Jednotlivé objemy jsou zde v podobě trojbokých hranolů od sebe tedy odděleny, vznikají tak duální prvky (body, linie, plochy), které nesmějí být při zpracovávání dat na základě aplikace počítány, respektive musejí být započítány takovým způsobem, že při generování souboru *.geo pro program Gmsch dojde k jejich eliminaci! Duální prvky mají stejné souřadnice nebo jiné vlastnosti. Obrázek 5.2 ukazuje takovou reprezentaci, pro větší zřetelnost jsou od sebe hranoly odděleny mezerou, aby bylo zřetelné, že nejsou součástí jednoho celku, v praxi se překrývají.



*Obr.5.2: Renderovaná ukázka reprezentace objemu v 3ds Max
(oddělené trojboké hranoly).*

6 Popis praktického řešení zpracování a vizualizace dat

Na následujících stránkách je zpracována práce na aplikaci, problémy které vyvstávají při převodu dat do programu 3dsMax zejména díky tomu, že GIS není CAD systém a politika tvorby modelu je odlišná a dále je zde zpracována metodika řešení při převodu a celé tvorby aplikace.

6.1 Implementace dotNet prvků ve scriptu

Za normálních okolností se jednotlivé ovládací prvky maxscriptu vkládají buď z pomocné aplikace Visual editor nebo se píše přímo do okna pro psaní kódu. Tyto komponenty jsou předkompilovány přímo v 3dsMax, mají ale tu nevýhodu, že jsou ochuzeny o většinu událostí a vlastností. Příkladem je např. komponenta tlačítko (button), kdy jedinou událostí je událost click, tyto jednoduché komponenty stačí v případě, že tvoříme menší scripty. V případě, že však píšeme obsáhlé scripty, které mohou být provázány mezi více roletek, je vhodnější využívat možnosti implementace dotNet komponent, tříd a objektů, které maxscript nabízí. Druhým důležitým faktem je, že ActiveX prvky jsou kompilovány jen pro 32-bitové systémy, tudíž je vhodnější využít dotNetu. Každá verze programu 3ds Max má nové možnosti ohledně této implementace, je vhodné si tedy zjistit jaké možnosti v dané verzi možné [20][25].

Tato práce je tedy jakýmsi hybridem, kdy klasické příkazy pro práci se scénou, geometrií, atd. jsou tvořeny příkazy maxscriptu a oproti tomu UI (uživatelské rozhraní), tedy komponenty, jsou společně se svými událostmi, vlastnostmi ve spolupráci s objekty a třídami psány pomocí syntaxe dotNetu. Při práci bylo využíváno seznamu možností implementace dle [26][27], protože ne vše jde z dotNetu použít pro psaní v maxscriptu.

6.2 Zdrojová data a požadavky na převod

Script, který se stará o vytvoření modelu čte informace ze dvou zdrojových souborů, které jsou k úspěšnému převedení modelu nezbytné. Data mohou být uložena v excelovských tabulkách nebo databázových tabulkách formátu dbf, které jsou ideální pro uchovávání informací tohoto typu a zároveň se používají v GIS systémech.

6.2.1 První zdrojový soubor

Data, která jsou obsažena v prvním souboru jsou informace o jednotlivých bodech/vertexech modelové sítě krajiny. Každý bod je definován v prostoru, má tedy tři souřadnice jsou to osy X,Y a Z. V každém řádku souboru jsou tedy čísla souřadnic, na kterých se každý daný bod nachází. Soubor má tedy tři sloupce dat.

Vzhledem k tomu, že jednotlivé body sítě jsou v souboru uloženy v řádku, předpokládá se, že informace v prvním řádku odpovídají bodu č.1 v druhém č.2 atd. To je také jedna z důležitých podmínek pro správné vytvoření.

V prvním sloupci jsou X-ové souřadnice bodů, v druhém sloupci Y-ové souřadnice a ve třetím Z-ové souřadnice.

	A	B	C
1	X	Y	Z
2	-683222,36694300000	-1098752,73804000000	250
3	-683034,07841200000	-1098809,38039000000	250
4	-682840,20099000000	-1098857,05158000000	250
5	-682660,88237000000	-1098937,90665000000	250
6	-682489,51472500000	-1099040,64729000000	250
7	-682304,92710100000	-1099112,15306000000	250
8	-682109,03024800000	-1099151,29597000000	250
9	-681958,70761800000	-1099272,47188000000	250
10	-681790,21591100000	-1099376,91894000000	250
11	-681601,93370600000	-1099443,84222000000	250
12	-681447,88561400000	-1099565,53546000000	250
13	-681280,88102000000	-1099631,32456000000	250
14	-681134,83464000000	-1099496,85729000000	250
15	-680986,84573800000	-1099600,76821000000	250
16	-680936,86243200000	-1099788,67972000000	250
17	-681034,23324200000	-1099960,44126000000	250
18	-681148,15389700000	-1100124,63206000000	250
19	-681279,95450300000	-1100269,49607000000	250
20	-681356,35320600000	-1100453,94620000000	250
21	-681464,03237900000	-1100621,72934000000	250
22	-681567,58892700000	-1100790,65925000000	250
23	-681603,02319400000	-1100986,02009000000	250
24	-681653,65253100000	-1101178,12096000000	250

Obr.6.1: První zdrojový soubor s ukázkou dat x,y,z souřadnice bodů.

6.2.2 Druhý zdrojový soubor

V druhém souboru jsou data, která obsahují informace o tom, které body/vertexy se mají spojit tak, aby vytvořily polygon, jelikož se geografické modely, díky triangulaci skládají z trojúhelníkové sítě, tedy ohraničenou plochu, která daný trojúhelníkový polygon tvoří. Data v souboru jsou tedy uspořádána tak, že každý řádek obsahuje informace o jednom polygonu. Relevantní jsou tři sloupce dat. V každém jsou čísla vertexů, který má být součástí polygonu.

V prvním sloupci jsou čísla jednotlivých polygonů, tato data jsou tu jen pro pořádek. V prvním datovém řádku prvního sloupce je tedy uloženo číslo 1, v druhém 2, 3, 4 Důležité je, aby první polygon byl označen jako polygon č.1, což je také důležitý požadavek k práci. V druhém sloupci jsou první čísla bodů, které tvoří

polygon, v třetím sloupci také druhá čísla bodů a ve čtvrtém jsou třetí čísla bodů. Tento soubor může obsahovat ještě pátý sloupec, v něm jsou udávány číselné hodnoty, které reprezentují vlastnost polygonu, která je v aplikaci dále využitelná, pomocí této vlastnosti se dá rozdělovat geometrie na podskupiny polygonů na základě barevného rozlišení.

	A	B	C	D	E
1	CISLO	POLYGON A	POLYGON B	POLYGON C	MAX_HORNIN
2	1	73	1010	74	100
3	2	68	73	72	100
4	3	69	71	70	100
5	4	69	72	71	100
6	5	74	1231	75	200
7	6	75	1232	1231	200
8	7	76	1232	77	200
9	8	68	72	69	100
10	9	77	1232	1009	200
11	10	1009	1232	1231	200
12	11	74	1230	1010	200
13	12	68	1010	73	100
14	13	77	1009	78	100
15	14	67	1010	68	100
16	15	1008	1231	1009	200
17	16	74	1231	1230	200
18	17	1008	1231	1230	200
19	18	78	1009	1008	100
20	19	1010	1230	1229	200
21	20	67	1229	1010	200
22	21	78	1008	79	100
23	22	1007	1230	1008	200
24	23	1007	1230	1229	200

Obr.6.2: Ukázka druhého souboru. Identifikace polygonu pomocí bodů.

Důležité je, aby oba soubory s daty spolu korespondovaly, nesmí se tedy stát, aby např. v druhém souboru byly informace o polygonu, který se skládá z bodů, které v prvním souboru nejsou zapsány.

Druhým důležitým kritériem pro bezproblémové načtení dat je, aby všechny informace byly uloženy ve správných sloupcích.

Třetím pravidlem pro bezproblémové zpracování je, aby první polygon byl označen jako polygon č. 1 a stejné pravidlo platí o označení bodů. Polygon tedy nesmí být označen například takto:

234, 0, 112, 9, 3 (číslo polygonu, první bod, druhý bod, třetí bod, vlastnost), kdy čárka označuje konec sloupce. Nejmenší číslo bodu je tedy 1, nulová hodnota, tak jak je uvedena v příkladu nebude akceptována a převod neproběhne

6.3 Možnosti načítaných formátů do aplikace

První možností načítání dat je ve formátu xls, tedy ve formátu excel 2003. Tento typ souboru má však jisté omezení, které částečně limituje převáděnou síť.

Protože script načítá data po řádcích, maximální počet řádků v excelu je omezen na 65 536. Z hlediska práce v excelu je to více než dostatečný počet, nicméně v případě uchovávání dat geometrických objektů se může stát u větších sítí, že kapacita nebude stačit. Jelikož se data potřebná pro vytvoření geometrického modelu týkají dvou důležitých elementů (bodů a polygonů), jsou rozděleny do dvou samostatných souborů, což je přehlednější, díky tomu je také možné uložit více dat, ale i tak není možné větší síť převádět.

Z tohoto důvodu byla do scriptu přidána i možnost otevření a načtení dat ze souboru s příponou dbf (typ databázového souboru), který dovoluje uložit větší počet řádků. Tento typ souboru je také přímo podporován ArcGIS, jedním z GIS, ze kterého byla data pro tuto práci pořízena. Tento formát podporuje velká množina GIS, takže není problém tento formát využívat a je ideální pro převod dat mezi GIS a 3ds Max. Soubor typu dbf je tedy primárním formátem pro načítání do vytvořené aplikace, protože nemá takové omezení jako v případě formátu xls, kdy je počet převáděných prvků limitován počtem řádků, který by nemusel vždy při převodu velkých geometrií stačit.

V případě nemožnosti načítat dbf formát lze v případě velkého množství dat, tedy více než zmiňovaných 65 536 řádků načítat i jako xls, tedy jako excel soubor, ale není to příliš ideální způsob a je komplikovaný na úpravu dat v souboru, právě proto byla přidána možnost načítat dbf soubor, aby se předešlo problémům. Data v takovém případě musíme rozdělit do více těchto dvojic souborů (body/polygony) a celou geometrii převádět postupně jednu část po druhé, což sebou nese několik problémů. V případě takového oddělení je každý další soubor chápán jako nová geometrie, musí tomu tedy odpovídat také úprava každého souboru a správná návaznost na změnu indexace elementů geometrie v 3dsMax, tak jak bylo popsáno v kapitole **Zdrojová data**. Po převodu všech takto rozdělených částí modelu pomocí aplikace lze všechny části spojit ručně, spojení už ale neprobíhá na úrovni vytvářené aplikace, ale v nabídkách programu 3ds max a to prostřednictvím tlačítka weld v modifikátoru Editable Poly v části vertex, problematika spojování je popsána v [19], kdy se svařují body, které jsou pro jednotlivé části geometrie společné.

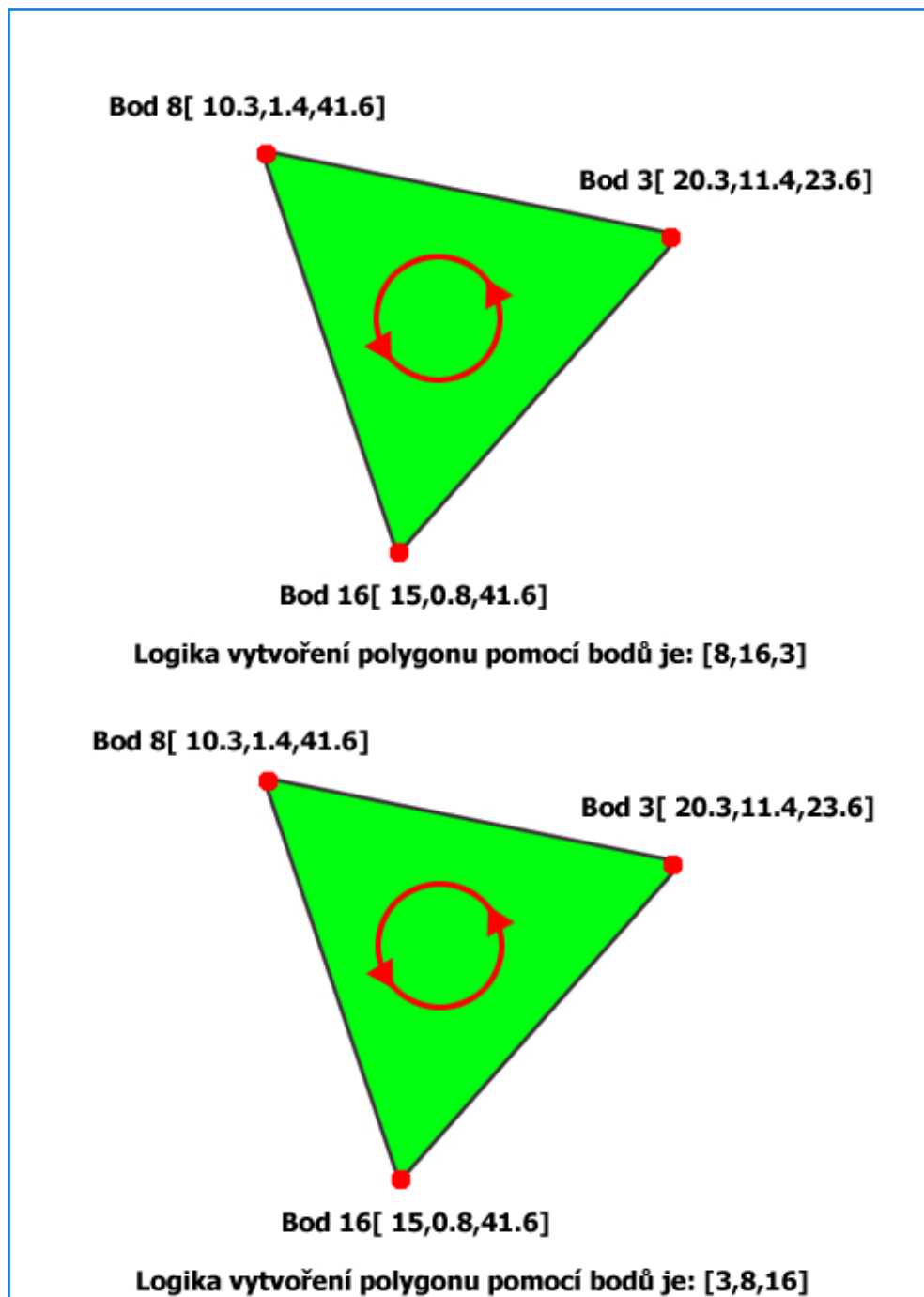
6.4 Práce s daty

Vzhledem k obrovskému objemu dat, které mohou být uloženy v obou výše zmiňovaných souborech, by bylo neefektivní, aby se v souboru přečetl vždy jen jeden řádek a vytvořil se podle něj požadovaný element. Proto se při čtení dat ze souboru automaticky data po otevření načtou a uloží do datového typu `SafeArrayWrapper` detailně popsáném v [20] a teprve potom jsou informace o jednotlivých elementech zpracovávány. Zajistí se tím tedy rychlejší práce, navíc data jsou dále potřebná pro další operace, je tedy nezbytné, aby byla takto uchována.

Elementem je v prostředí 3ds Max myšlen prvek modelu, který je nezbytný k jeho sestavení. V tomto konkrétním případě se jedná o polygon, drouhou možností kdy mluvíme o elementu je prvek polygonu, což může být prvek vertex/bod sítě nebo hrana polygonu, tedy prvek plošky triangulované sítě. Pro lepší přehlednost bude v textu vždy zdůrazněno o jaký typ elementu se jedná.

6.5 Čím se řídí správné zobrazení polygonu v 3ds Max

Zobrazení polygonů se řídí následujícím pravidlem. Pokud máme jakýkoli polygon o libovolném počtu vrcholů, pro správné zobrazení polygonů je nutné, aby byl polygon vytvořen v určitém pořadí bodů v prostoru, ze kterých je sestaven. Při testování a zkoumání jsem zjistil, že je nutné, aby byl polygon vytvořen tzv. proti směru hodinových ručiček. Obrázek vysvětluje tuto problematiku.



Obr.6.3: Možné pořadí pro tvorbu polygonů.

Pokud je polygon sestaven podle tohoto pravidla, ploška která znázorňuje viditelnou část geometrie se je orientována čelem k uživateli, pokud se provede opačný postup tvorby, pak je viditelná část polygonu orientována opačným způsobem.

Pokud se tvoří daný polygon uprostřed sousedících polygonů, není možné vytvořit plošku s opačnou orientací. Konkrétní problémy spojené s tvorbou geometrie jsou rozebrány v následujících kapitolách.

6.6 Problematika tvoření elementů polygonu v 3ds Max

Oba programy mají v případě vytváření 3d modelů některé odlišnosti které musely být ve scriptu zohledněny tak, aby byl model v prostředí 3dstudia úspěšně vytvořen.

V podstatě jsou dva způsoby, jak v prostředí 3d studia definovat polygon. Jelikož chceme převádět polygony automatizovaně a využíváme k tomu maxscript, můžeme využít dvě metody pro tvorbu polygonu.

První metoda využívá definice polygonu typu linie. Znamená to, že musíme znát konkrétní hrany, které mají být spojeny.

Druhá metoda využívá k definici polygonu typ bodu. V takovém případě to znamená, že musíme znát konkrétní body, které mají tvořit daný polygon.

Nakonec byla zvolena metoda tvorby číslo dvě, tedy metoda, která definuje polygon jednotlivými body a to zejména proto, že naprosto přesně odpovídá datům, která odpovídají vstupu, tedy datům, která jsem měl pro tvorbu k dispozici a která byla z GIS vygenerována, tento formát vstupu má ještě jednu výhodnou vlastnost a to sice tu, že zjednodušuje proces tvorby polygonu než v případě první metody.

Pro definici polygonu musíme vytvořit body celé geometrie a pro definici plošky musíme znát identifikace bodů, které celý polygon tvoří.

Pokud by byla zvolena metoda číslo jedna, musely by být vytvořeny nejdříve jednotlivé body celé geometrie, poté hrany u kterých je nutné definovat body které mají být spojeny a naposled pro definici plošky, v tomto případě tedy hrany, které uzavírají polygon.

Z toho tedy vyplývá, že celá konstrukce vybrané metody je jednodušší, protože odpadá nutnost definovat přesně hrany polygonu a usnadňuje to tedy celý proces, není nutné mít další soubor pro definici hran a šetří tedy i paměťové nároky na aplikaci.

6.6.1 Převedení bodové reprezentace převáděné geometrie

V případě prvního typu elementu, tedy jednotlivých bodů, které tvoří základ polygonu se nevyskytují žádné problémy při převodu do prostředí 3d studia. Logika je v tomto případě naprosto stejná, informace, které je třeba znát jsou prostorové souřadnice X,Y,Z. Algoritmus prochází pole, ve kterých jsou uložena data o jednotlivých bodech, čte data souřadnic a následně tento typ elementu vytvoří.

6.6.2 Převedení do polygonové reprezentace

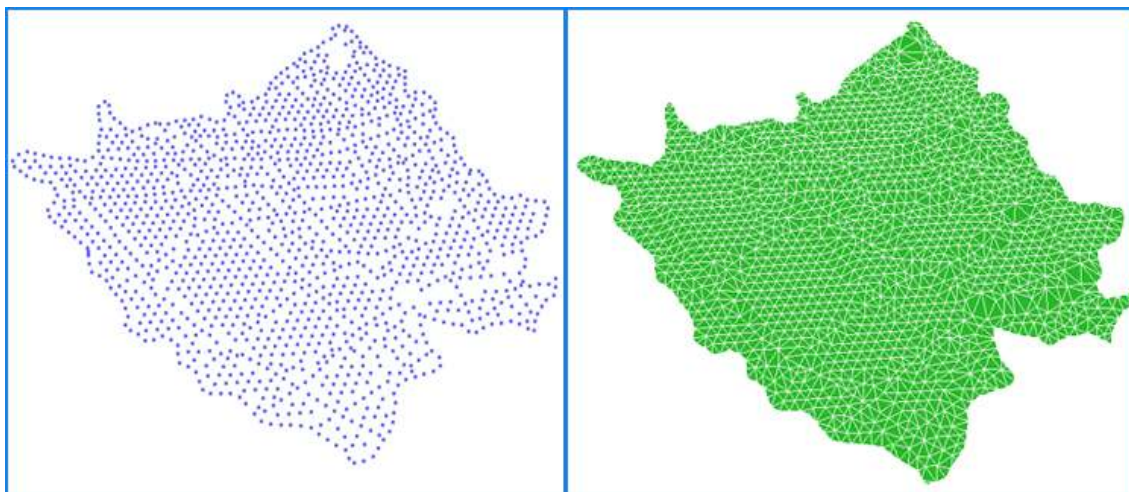
Jak už bylo řečeno, bylo pro tvorbu polygonu zvolena metoda definice každého z nich pomocí identifikačních čísel bodů. V ideálním případě by se měly polygony při čtení dotvořit a převedení celé geometrie by bylo kompletní.

Vytvoření polygonu už ale představuje značný problém, protože logika vytvoření tohoto typu elementu je odlišná v ArcGIS a je jiná v 3Dstudiu.

Důležité je zmínit, že jelikož GIS pracuje s daty ve vrstvách, struktura morfologie krajiny např. v případě TIN nemusí řešit, jestli je ze spodní strany struktury geometrie vidět nebo ne.

V případě 3ds Max je ale situace jiná, protože studio dokáže pracovat v plném 3d, rozlišuje v případě polygonů jejich vyplňující plošku na viditelnou část a odvrácenou část, která je odlišena jiným odstínem, pro vysvětlení definice viditelné strany plošky a její odvrácené strany stačí zmínit, že se řídí podle orientace normály (přímky), která je kolmá na rovinu polygonu a míří z jeho středu jedním směrem, Problémem vytváření polygonů v prostředí 3d studia je tedy to, že polygon je z jedné strany neviditelný (reprezentuje tmavší část plošky a v některých situacích se nevykreslí) a z jedné strany viditelný zobrazovaný stejně jako v GIS.

Při neošetřených situacích se tedy stávalo hned několik nežádoucích jevů, které bylo potřeba ošetřit pro správnou orientaci polygonů, kdy tyto problémy mohou ve výsledku vést k degradaci geometrie a její nesprávné reprezentaci právě z důvodů toho že data z GIS neobsahují informace o orientaci polygonů a komplikují vizualizaci v CAD systémech které taková data potřebují.



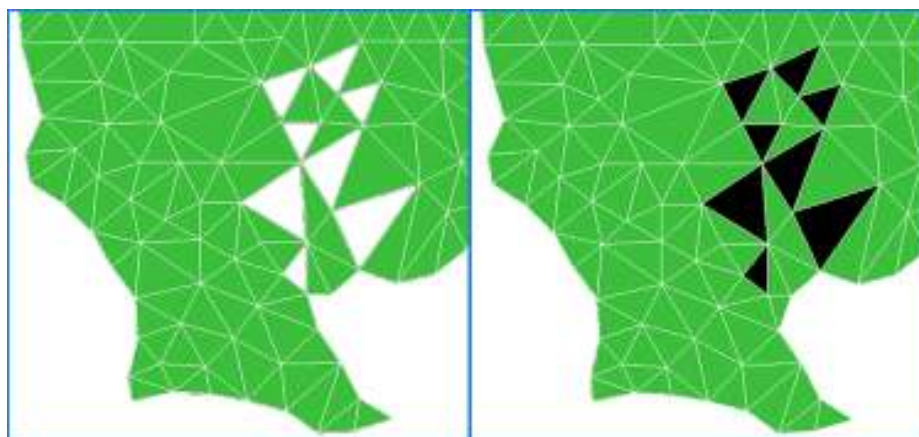
Obr.6.4: Zobrazení bodové a polygonové reprezentace geometrie.

Otočení polygonu

Prvním problémem, který bylo nutné vyřešit bylo nežádoucí otočení polygonu opačným směrem než by měla být jeho správná orientace. Stávalo se tedy, že některé polygony byly po přečtení dat ze souboru a jeho následnému převedení otočeny na opačnou stranu vytvářeného modelu. To znamená, že jeho viditelná část tedy byla přetočena opačně.

Nevytvoření polygonu

Druhým významným problémem bylo nevytvoření požadovaného polygonu. Tento problém se děl hlavně v některých oblastech vytváření modelové sítě a to zejména v oblastech, kde byly některé z polygonů otočeny opačným směrem. Stávalo se tedy, že aplikace byla neočekávaně ukončena, protože příkaz scriptu, který se staral o vytvoření polygonu v přítomnosti polygonu, který byl otočen a vedle něj měl být vytvořen aktuální polygon nedokázal vzhledem k pravidlům vytvoření příkaz vykonat.



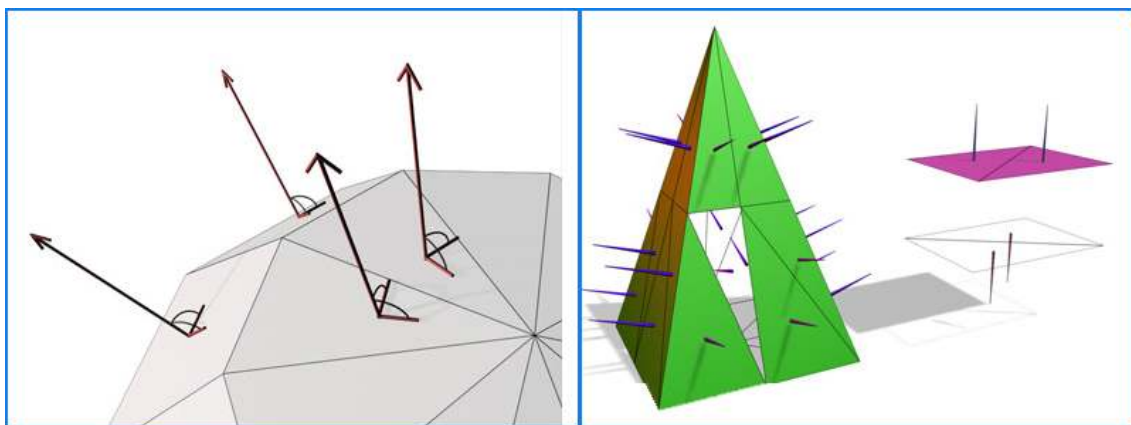
Obr.6.5: Příklad nevytvoření a otočení polygonu v geometrii.

6.7 Metodika řízení normál polygonů

V prvopočátku byla hledána metodika jak správně zobrazovat orientaci polygonů, z toho důvodu byla studována možnost řídit orientaci pomocí normál jednotlivých plošek.

6.7.1 Co jsou to normály a možnosti řízení

Normála je vektor, který definuje směr, kterým je ploška polygonu orientována, tedy viditelnou část plošky. Charakteristické pro normálu je fakt, že je kolmá na rovinu daného polygonu, svírá tedy úhel 90 stupňů a v maxscriptu je možné upravovat orientaci plošek právě takovýmto způsobem.



*Obr.6.6: Zobrazení normál po ploškách jednotlivých polygonů
(přejato z dokumentu help pro 3ds Max)*

Pokud bychom se snažili na základě dat predikovat orientaci jednotlivých normál plošek dojdeme k tomu, že jen na základě zdrojových dat to není možné, protože řada bodů, které definují polygon ne vždy odpovídá směru, kterým by měla být ploška orientována, dochází k tomu vlivem převodu z GIS vs. 3dsMax (jako jeden z CAD systémů) a není možné provádět kontrolu před vytvořením geometrie.

Byl tedy zvolen trochu odlišný způsob a to, že nejdříve se pokusí geometrie vytvořit i se všemi chybami a pak se na základě analýzy normál, konkrétně tedy sousedících plošek okolo té aktuálně kontrolované algoritmus pokusí upravit jejich směr. Bohužel ani tento přístup se neukázal jako přínosný, protože v případě že celá geometrie nereprezentuje rovinnou plochu ale členitý povrch, pak mohou být okolní normály polygonu otočeny naprosto opačným způsobem, v lepším případě by se vypočítala pouze pravděpodobnost správné orientace, ale ne jistota správného natočení (orientace normály), jak je vidět i z předcházejícího obrázku.

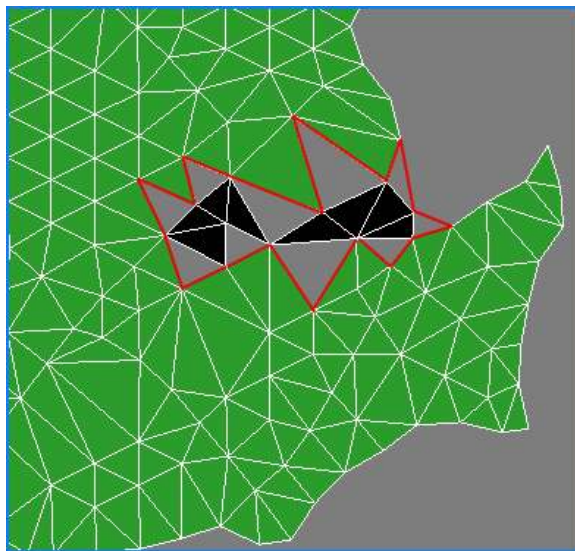
6.8 Metodika řešení správného zobrazení polygonu

Aby při automatizované tvorbě nedocházelo k chybám, které jsou popsány výše, konkrétně v podkapitole **Problematika tvoření elementů polygonu v 3dsMax**, tedy aby toto řešení správně vyhodnocovalo různé naklopení polygonu například při zobrazování pohoří, členitého povrchu, dutin, vypuklin nebo jeskyní, bylo nutné se zaměřit na vlastnosti tvorby v geometrie a jejich modifikátorů, protože zdrojová data neobsahují informaci o jednotlivých normálách polygonů. Nakonec byl tento problém vyřešen jiným způsobem, který jsem vymyslel na základě studia vlastností sítí v 3dsMax, toto řešení však není zcela obvyklé. Celý algoritmus, který se stará o vytváření geometrické sítě je prováděn nad modifikátorem Editable Poly, který jako jeden z nástrojů 3dsmax umožňuje modifikovat a editovat geometrické sítě. Princip vytváření polygonu je poměrně specifický pro daný problém. Nejdříve se polygon tzv. snaží vytvořit. Nejprve jsem příkaz pro vytváření polygonu vložil do bloku try, který by detekoval úspěšné vytvoření nebo nevytvoření polygonu a mohl bych tak pohodlně kontrolovat tvorbu sítě. Bohužel v tomto případě je v maxscriptu zřejmě chyba, protože při testech jsem zjistil, že příkaz s blokem try nepracuje korektně, nezáleží na tom, jestli se daný polygon vytvoří nebo ne, pokaždé skončí ve stejné části kódu (jako úspěšné vytvoření) a proto nelze kontrolovat tvorbu sítě a dochází k chybám popsaným v části **Problematika tvoření elementů polygonu v 3dsMax**.

Detekce vytvoření polygonu je realizována jeho následným vybráním a označením. Zároveň se počítají řádky, které byly načteny ze zdrojového souboru jako kontrola. Pokud se některý polygon nevytvoří, pak přestane souhlasit index řádku ze zdrojového souboru, na kterém se program právě nachází a počet již vytvořených polygonů, v takovém případě se provede přepočítání a zaznamená do zvláštního pole index řádku, u kterého došlo k nevytvoření daného polygonu. Tímto způsobem se tedy zjistí, který polygon nebyl vytvořen.

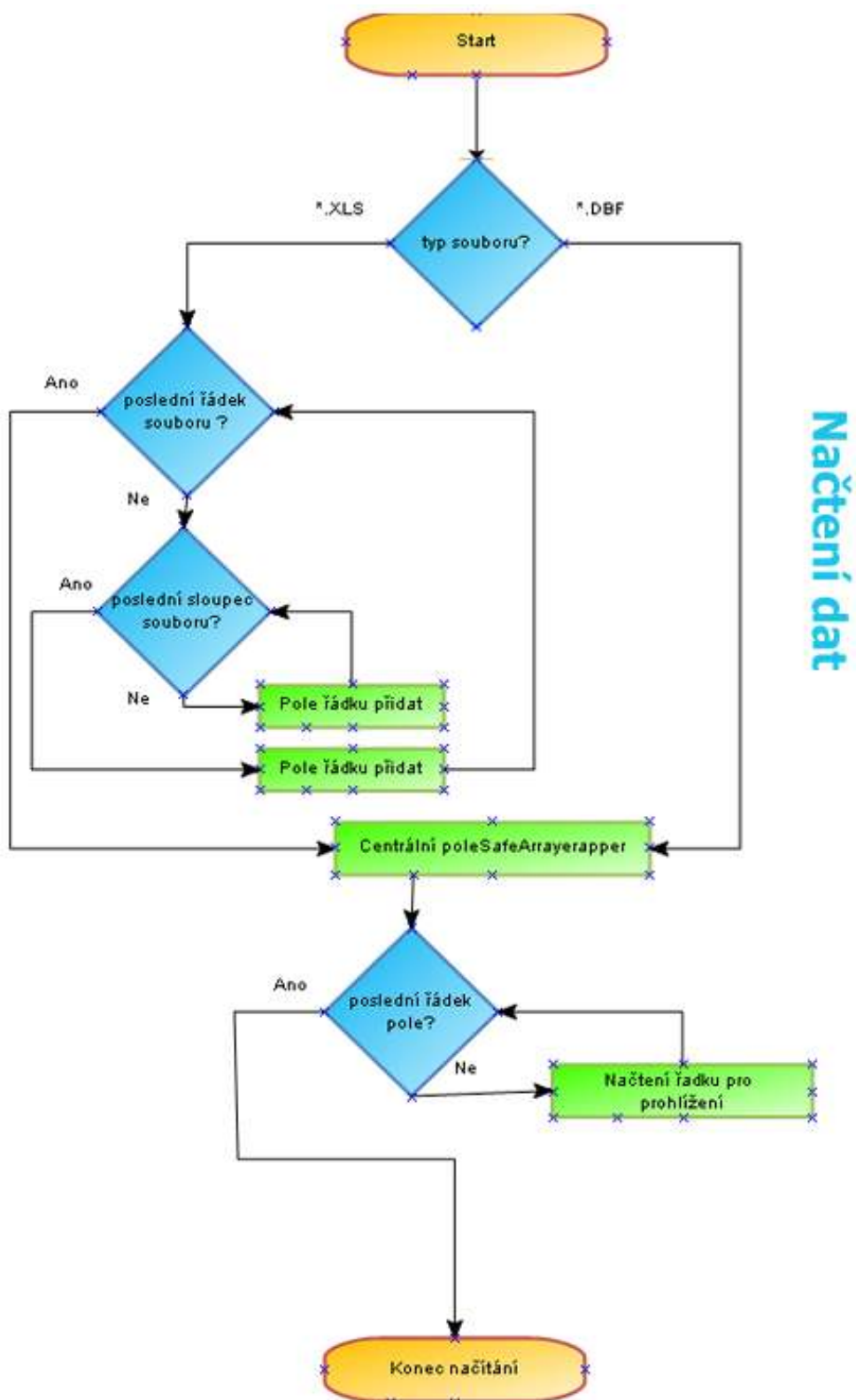
Jistou komplikací je fakt, že polygony jsou indexovány automaticky podle toho jak přijdou po sobě ze zdrojových dat. Není tedy možné jejich označení žádným způsobem měnit programově, což právě komplikuje kontrolu nevytvořených polygonů, protože pokud se vyskytne právě situace nevytvořených polygonů, pak se začnou lišit indexy těch vytvořených i označením jaké mají ve zdrojových souborech, protože ty nevytvořené polygony se dotvářejí až jako poslední.

V tomto bodu tedy vytvořená síť ještě neopravuje chyby, které vznikají vlivem jiných pravidel sestavování polygonů, v rozdílných programech, nicméně zjišťuje, které z polygonů se vlivem převodu jinak chápaných geometrií nevytvořily. Vzniklá síť tedy může obsahovat i nesprávně orientované polygony.

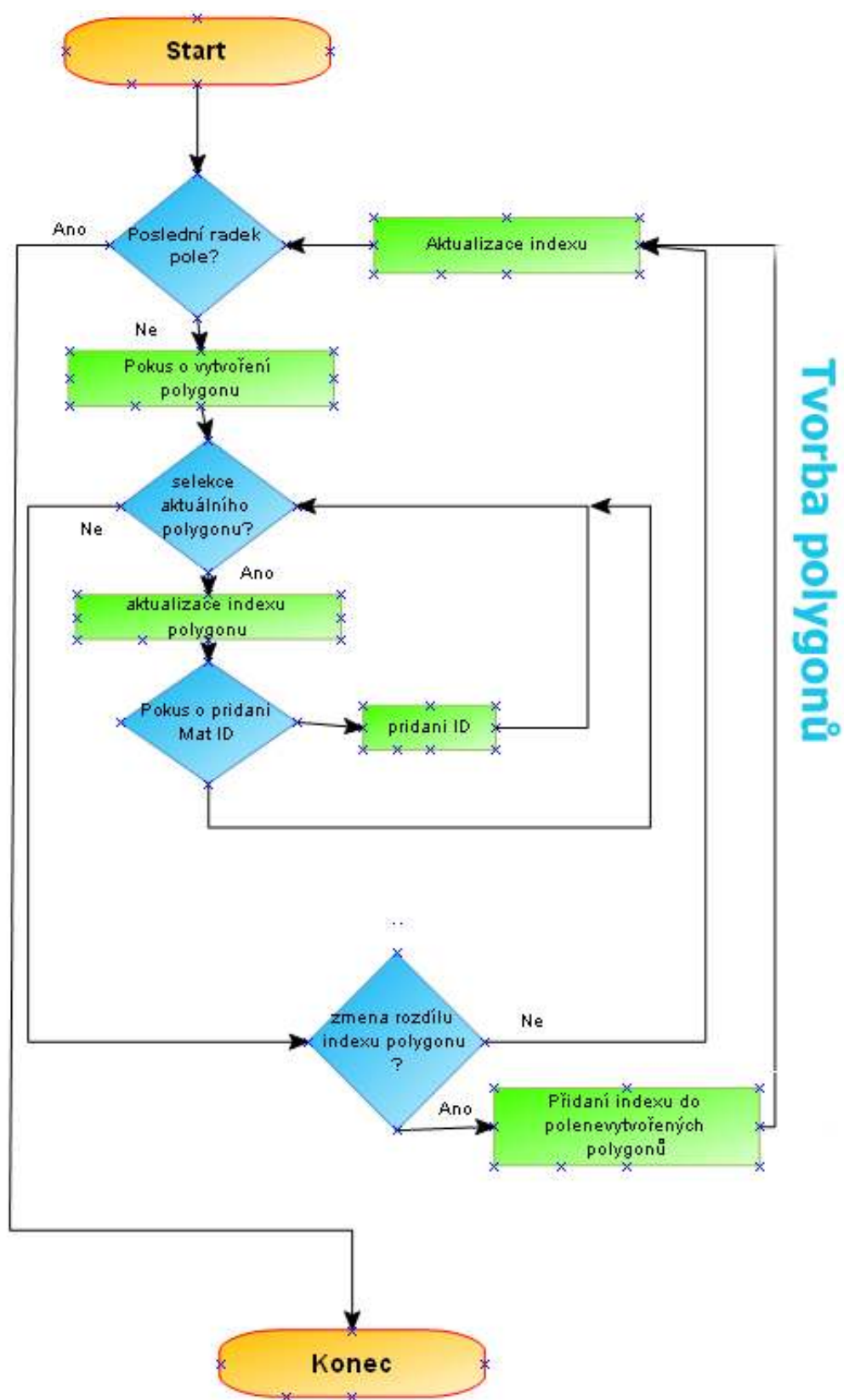


Obr.6.7: Zobrazení chyb v síti (ohraničená oblast červenou barvou).

Druhým krokem při opravě geometrie je tedy nejdříve otočit nesprávně orientované polygony, které mají svou normálu otočenou opačným směrem, protože právě jejich vlivem se sousední polygony nevytvářejí (tak je postaven příkaz pro tvorbu polygonů v 3dsmax). Nesprávně otočený polygon je tedy zásadním problémem při automatizovaných převodech sítí. Jelikož nevíme, které z polygonů jsou otočeny, vzniká nám další problémový bod a pro řešení je v tomto případě nutná částečná účast uživatele.



Obr.6.8: Vývojový diagram tvorby polygonů v geometrii.



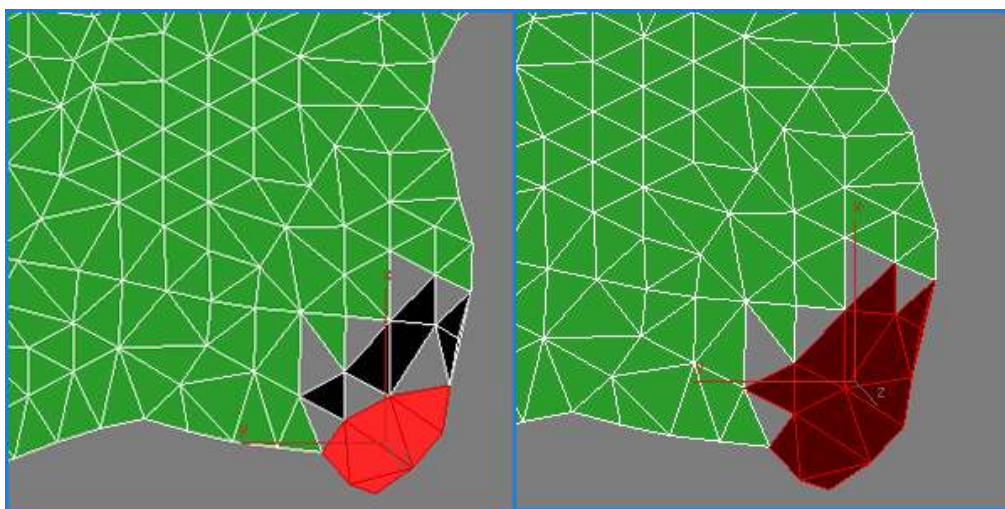
Obr.6.9: Vývojový diagram tvorby sítě.

6.8.1 Oprava sítě

Fyzicky je oprava problémů souvisejících s otočenými a nevytvořenými polygony realizována samostatným tlačítkem. Algoritmus starající se o opravu sítě je prováděn v několika krocích. Při pozorování vlastností sítí, které obsahují otočené nebo nevytvořené polygony jsem zjistil, že takováto síť je rozdělena na několik samostatných oblastí, které se dají pomocí vlastností modifikátoru samostatně vybírat.

Tyto vybrané oblasti mají tu vlastnost, že obsahují jen ty polygony, které jsou otočeny jen jedním směrem, vyloučí tedy ty, které mají normály opačně.

Algoritmus v dané geometrii nejprve vybere první polygon, tedy takový s indexem číslo 1. Následně se vybere oblast, do které patří tento první polygon a otočí se všechny normály polygonů v tomto vybraném oddílu.

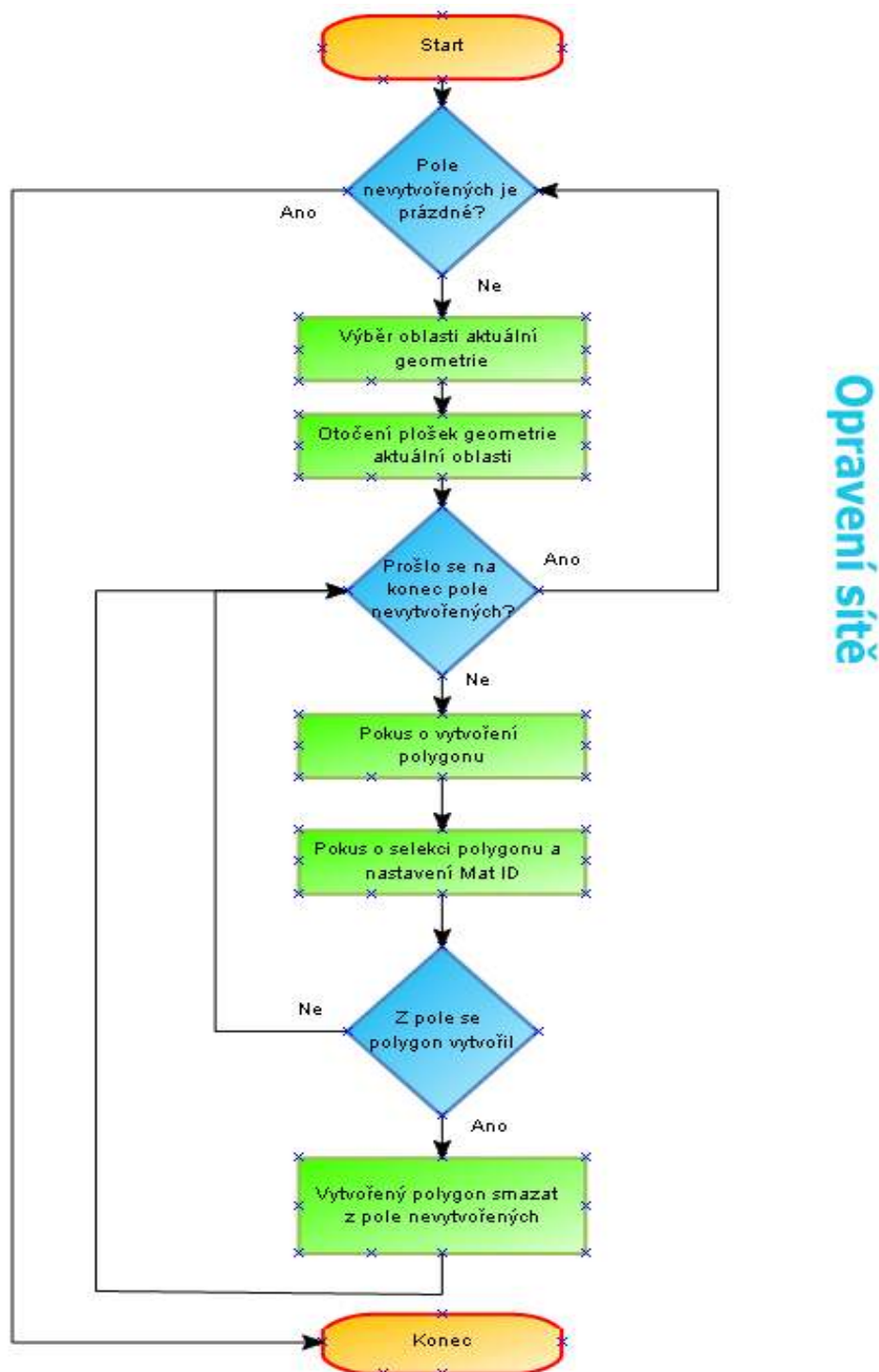


Obr.6.10: Výběr oblasti se stejnou orientací, první a druhá iterace.

Tímto procesem je docíleno toho, že se normály otočí stejným směrem jako sousedící špatně orientované polygony. Je tím docíleno obejití pravidla vytvoření polygonů. Dalším krokem je pokus vytvoření polygonů, které nebyly vytvořeny. Díky tomu, že oblast je v tuto chvíli otočena opačným způsobem, vznikají podmínky k dotvoření některých polygonů z tohoto pole, proto se polygony zkouší vytvořit znovu, výhodou tohoto systému je to, že neexistuje možnost, že by se znovu nevytvořil alespoň jeden polygon s tohoto pole. Ty polygony, které se dotvořily zmizí z pole nevytvořených a celý cyklus se takto opakuje, přičemž při výběru prvního polygonu síť a následnému vybrání oblasti už neobsahuje stejné množství polygonů, ale rozšířené o právě dotvořené a to i včetně těch, které byly v oblasti v minulém cyklu otočeny nesprávně. Tímto procesem postupného rozšiřování oblasti a ubírání tedy prvků

nevytvořených polygonů z pole nakonec dostaneme prázdný seznam, což znamená, že celá síť je kompletní a otočena jen jedním směrem.

Může se stát, že dotvořená geometrická síť je jako celek otočena nesprávně. Opravu lze provést znovu stisknutím stejného tlačítka, které celou síť polygonů otočí správně.



Obr.6.11: Vývojový diagram pro opravu sítě.

7 Stavba objemu v 3dsMax

Možnost stavby objemu, tak jak byla definována v kapitole 5. (Objemy v 3dsMax) má poměrně specifické nároky, jelikož je geometrie kterou díky hlavnímu scriptu importujeme do 3dsMax spojitá a podle zmíněné kapitoly musí být všechny trojúhelníky základu modelové sítě odděleny, aby mohla vzniknout korektní reprezentace. Samotná ruční tvorba oddělených trojbokých hranolů je celkem komplikovaná, proto stavbu objemu usnadňuje script oddělený od hlavního řešení. Script musí splňovat několik požadavků.

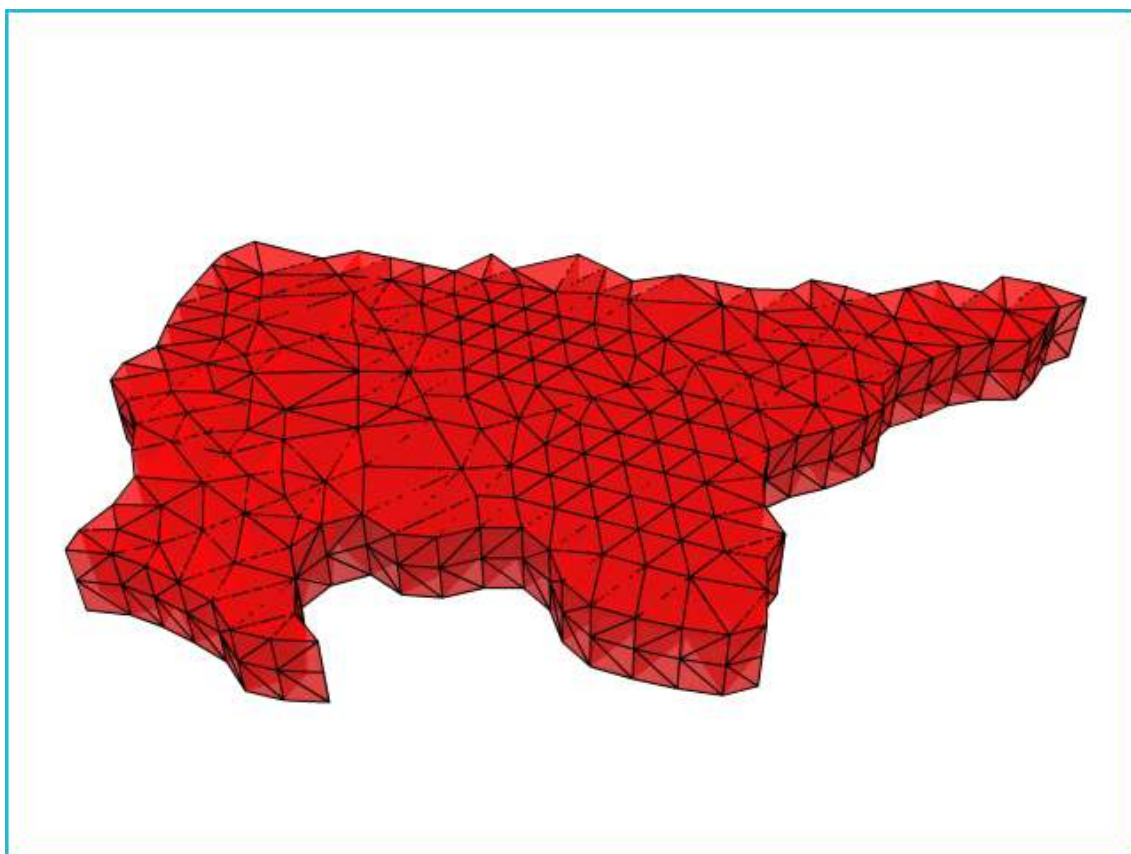
- Script musí umět jeden každý trojúhelník geometrie oddělit.
- Oddělení musí probíhat na základě extrakce plošek v originální geometrie
- Vytažení stěn trojúhelníků do určité výšky. (Stěny nejsou na opačných stranách uzavřeny spodní trojúhelníkovou základnou)
- Aplikování speciálního modifikátoru (Symmetry), který vytvoří symetrickou a otočenou kopii čelních plošek i čelních stěn.
- Opětovné seskupení teď již uzavřených trojbokých hranolů
- Odstranění nežádoucích hran a bodů, které vznikají spojením symetrických polovin.

Oddělení všech plošek trojúhelníků do samostatných geometrií, které jsou potřebné ke správnému svařování prvků probíhá pomocí speciálních příkazů.

Plošky, které se oddělí jsou pak součástí nové geometrie, přičemž každá geometrie obsahuje vždy jen jeden trojúhelníkový polygon. Tento fakt, je velice důležitý, protože při nesprávném použití, tedy při neoddělení polygonů se stávalo, že byly spojeny nesprávné body celé geometrie a došlo k degradaci celé geometrie.

Následně dojde k vytažení hran každé nové geometrie (polygonu) do požadované výšky poloviny celého modelu. Poté je aplikován speciální modifikátor, který umožňuje vytvořit symetrickou kopii otočenou tak, aby byla vytvořena druhá polovina trojbokých hranolů.

Odmazávání hran a bodů, které vzniknou při svařování vrcholů se provádí ručně, nicméně toto upravení je již jednoduché a lze dosáhnout požadovaného výsledku velice rychle. Postup je uveden v manuálu k celé aplikaci.

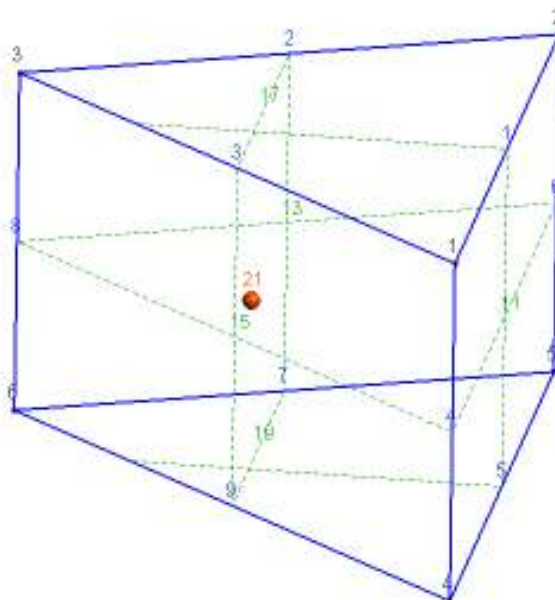


Obr.7.1: Objem ještě před odmazáním nežádoucích prvků v bočních stěnách.

8 Praktická realizace exportu do Gmsh

Samotný export do formátu *.geo je řízen jinými pravidly než jak tomu je v 3dsmaxu nebo ArcGISu. V tomto případě je export definován opět body a jejich souřadnicemi, ale potom už jsou plochy definovány pomocí linií a ne bodů. Bylo tedy nutné zvolit jiný přístup. Celková formátovaná úprava je také jiná. Systém tvorby je naprosto odlišný [30].

```
lc = 2;  
Point(1) = {-13.2557, 9.53674e-007, 19.6867, lc} ;  
Point(2) = {23.4487, 9.53674e-007, 19.6867, lc} ;  
Point(3) = {11.7244, 28.069, 19.6867, lc} ;  
Point(4) = {-13.2557, 0.0, 0.0, lc} ;  
Point(5) = {23.4487, 0.0, 0.0, lc} ;  
Point(6) = {11.7244, 28.069, 0.0, lc} ;  
Line(1) = {2, 1} ;  
Line(2) = {3, 2} ;  
Line(3) = {1, 3} ;  
Line(4) = {1, 4} ;  
Line(5) = {4, 5} ;  
Line(6) = {5, 2} ;  
Line(7) = {5, 6} ;  
Line(8) = {6, 3} ;  
Line(9) = {6, 4} ;  
Line Loop(10)={4,5,6,1};  
Plane Surface(11) = {10} ;  
Line Loop(12)={6,-2,-8,-7};  
Plane Surface(13) = {12} ;  
Line Loop(14)={8,-3,4,-9};  
Plane Surface(15) = {14} ;  
Line Loop(16)={2,1,3};  
Plane Surface(17) = {16} ;  
Line Loop(18)={5,7,9};  
Plane Surface(19) = {18} ;  
Surface Loop(20)={11,13,15,17,19};
```



Obr.8.1: Příklad řešení pro vytvoření jednoho objemu v Gmsh.

První řádek na obrázku je konstanta, tato hodnota udává velikost prosítování modelu. V Gmsh se totiž zobrazení polygonu, tedy jeho vyplnění plochou se provádí jinak, dochází k sesítování prostoru mezi liniemi v prostoru polygonu sítí linií, které barevnou plochu nahrazují. Hodnota konstanty se může měnit v závislosti na rozprostření bodů a požadavkům na hustotu prosítování.

Dalších šest řádků definuje body, každý bod je definován opět jako v 3dsmax , tedy souřadnicemi x, y, z, navíc ale obsahuje i hodnotu lc, tedy výše zmíněnou konstantou. Dalších devět řádků definuje linie, ve kterých je popis bodů které linie spojují. Řádek definovaný jako LineLoop nastavuje pořadí linií, kdy je nutné, aby na sebe linie daného polygonu navazovaly (prostřednictvím bodů uvedených v Line), aby se mohl polygon vytvořit. Obrázek na předchozí stránce ukazuje i zápornou orientaci linií, kdy znaménko mínus otočí pořadí bodů v definici Line právě pro Line Loop. Řádek definovaný jako PlaneSurface pak tento polygon vytvoří právě na základě Line Loop.

Pro definici objemu/volume je zapotřebí dvou příkazů, prvním z nich je Surface Loop, v němž je uveden seznam Plane Surface (ploch), které daný objem vytvářejí, příkazu Volume je pak uvedeno číslo Plane Surface. V dokumentaci k programu gmsh je uvedeno, že by se také měla dodržovat záporná orientace ploch, nicméně zde není možné žádným způsobem definovat správný směr, protože možností jak si ploch všimnout je mnoho (nejde označit jednoznačně pořadí ploch od začátku do konce), v tomto bodu lze tedy zapisovat jen kladná čísla, aniž by docházelo k chybám.

Po vybrání modelu, který chceme do prostředí programu Gmsh převést algoritmus prochází jednotlivé elementy celého modelu a sestavuje formát požadovaný pro gmsh. Tento formát je automaticky přepisován do textového souboru s příponou .geo, který je nutný pro otevření v gmsh. Kompletní dokumentace problému výstavby je popsána v [30].

8.1 Podmínky nutné pro export do gmsh

První podmínkou pro export dat z 3dsmaxu do gmsh je, že objekt který chceme exportovat bude vybrán.

Druhou podmínkou je že objekt, který chceme exportovat je, že na tento objekt bude aplikován modifikátor Editable poly.

8.2 Rozdělení řešení exportu geometrie do Gmsh

Jak už bylo popsáno, formátování dat o geometrii pro formát *.geo pro program Gmsh je jiné než v 3ds max. Následující text popisuje algoritmus pro export polygonové geometrie do jednoho textového souboru. V této části je nutné rozdělit popis řešení na dvě části, protože praktická část, obsahuje dva scripty, první je zaměřen zejména na převádění pouze plošné reprezentace (nepřepočítává duální prvky) a nehodí se pro přepis objemů, nicméně tento algoritmus je mnohem rychlejší. Script reprezentující objemy je umístěn samostatně mimo hlavní řešení, tento script už duální prvky započítává a korektně přepisuje indexy pro zápis do *.geo souboru, nicméně díky prohledávání všech prvků a vnořených cyklů je také pomalejší.

8.3 Popis algoritmu pro export ploch

Na následujících stránkách je vysvětlen způsob, jak algoritmus zpracovává vizualizovaný a vybraný model, na který byl aplikován modifikátor Editable Poly. Tento algoritmus nepřepočítává objemy, ale pouze povrchy modelu (duální prvky modelu budou do výpisu souboru také zahrnuty). Výpis souboru *.geo končí definicí Plane Surface.

8.3.1 Popis převodu bodů

Začátkem výpisu do souboru samotné geometrie je nutné definovat všechny body geometrie, začátek zpracování je jednoduchý, cyklus postupně prochází bod po bodu celou geometrii a zapisuje kromě konstanty k bodu X, Y, Z souřadnici každého bodu. Kromě těchto informací se zapisuje indexace každého bodu. Výsledný formát bodu je tedy například **(Point(5)={22.2423, 0.182, 100, 0.009};)**, jedná se tedy o definici pátého bodu se souřadnicí X: 22.2423, Y:0.182, Z:100 a konstantou 0,09. Důležité je ještě zmínit, že definice bodů v algoritmu je do souboru zapisován za sebou, takže začátek souboru obsahuje pouze definice všech bodů.

8.3.2 Popis převedu linií

Pokud algoritmus projde všechny body/vertexy dané geometrie, následuje procházení jednotlivých linií celé geometrie, přičemž bloky algoritmu jsou podobné těm, co se starají o zpracovávání bodů. U každé linie je nutné zaznamenat body, z kterých se daná linie, tedy hrana polygonu skládá. Výsledný formát tedy vypadá

například (**Line(12)={121, 201};**), jedná se tedy o linii/hranu číslo dvanáct a skládá se z bodů geometrie číslo 121 a bodu 201, přičemž indexace jednotlivých linií se řídí podle procházení v geometrii. V souboru jsou za definicí všech bodů definovány všechny hrany celé geometrie.

8.3.3 Popis převodu Line Loop a Plane Surface

Po ukončení zápisu všech linií do souboru se definuje pořadí hran v jednotlivých polygonech a příkaz pro vytvoření dané plošky polygonu. Jak už bylo zmíněno v kapitole **Export do Gmsh**, definice Line Loop pro řazení hran polygonu musí obsahovat všechny hrany daného polygonu v takovém pořadí, aby na sebe vzájemně navazovaly, to znamená, že konec jedné hrany musí navazovat na začátek hrany další, tak to pokračuje dál, dokud se takto dostaneme k začátku první hrany polygonu. Dobře ukázána je tato problematika v obrázku v kapitole **Export do Gmsh**. Začátek a konec hrany se pozná podle definice dané linie, tedy bodů, které tuto hranu tvoří, první bod značí začátek hrany a druhý bod její konec. Jelikož 3dsMax takto polygon nedefinuje, algoritmus aplikace který tuto problematiku řeší je velmi složitý, vývojový diagram znázorňující tento problém ukazuje základní bloky, řízení tohoto algoritmu je dosti složité a všechny náležitosti se do něj nevejdou, aniž by vývojový diagram neztratil přehlednost.

Jelikož upravená geometrie již nemusí obsahovat pouze trojúhelníkové polygony a počet vrcholů polygonu tedy není omezen, vyvstává tu problém toho, že v převáděné geometrii může být proměnlivý počet hran v každém polygonu, což komplikuje řešení a zvyšuje nároky na dynamičnost algoritmu.

Algoritmus v této části prochází jeden polygon za druhým, přičemž nejdříve zjistí čísla hran, které tento polygon ohraničují a uloží ho do samostatného pole, které je považováno za neseřazené hrany, přičemž první hrana se přesune z tohoto pole do pole, které je považováno za pole seřazených hran polygonu a podle ní se algoritmus dále řídí.

Následuje rozhodovací blok, který kontroluje, kde se v poli neseřazených hran nacházíme, tedy která hrana má přijít ke kontrole seřazení, tento konkrétní blok se řídí poměrně složitě, jde totiž o to, že ne každá následující hrana v tomto polygonu navazuje na aktuální hranu seřazenou, takže pokud se nehodí, algoritmus musí umět dynamicky

reagovat a zaznamenávat pozice kontroly, kdy může docházet k několika různým stavům, které odpovídají jinému chování.

Následně se u aktuální kontrolované hrany zjistí oba body této linie, tedy jejich indexační čísla, u poslední seřazené hrany se tyto body zjistí také a vzájemně se tyto body zkontrolují, přičemž mohou nastat tři stavy.

Prvním stavem je, že aktuálně kontrolovaná neseřazená hrana a tedy i její první bod odpovídají koncovému bodu aktuální hrany seřazené, v takovém případě se hrana přesune do pole seřazených podle čísla této hrany, index prohledávání se nastaví na číslo 1, kdy tento index řídí zaměření na počáteční daný bod v následující kontrole hrany a je zaznamenána kontrolovaná pozice.

Druhým stavem je, že kontrolovaná hrana i její bod odpovídají počátečnímu bodu aktuální hrany seřazené, v takovém případě se tato hrana také přesune do pole seřazených, ale ne pod jejím indexačním číslem, ale číslem opačným, tedy pokud se jedná např. o linii číslo 10, do pole seřazených se zaznamená jako -10, tak totiž gmsh chápe otočení linie. Index prohledávání se změní na 2, takže kontrola další linie se zaměří na bod koncový a je také zaznamenána kontrolovaná pozice.

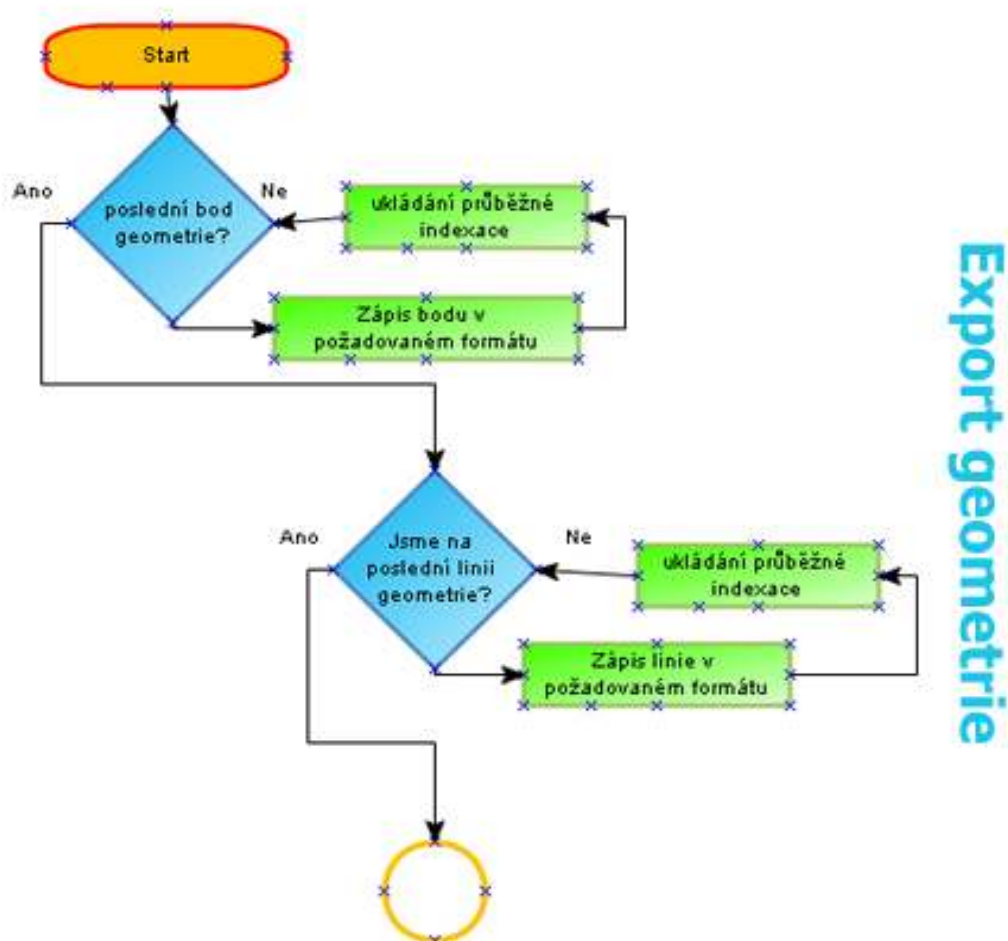
Třetím stavem, který může nastat je, že daná kontrolovaná hrana neobsahuje ani jeden bod, který je v hraně seřazené, v takovém případě se hrana nepřesune do pole seřazených, indexace kontrolovaného bodu se nezmění, musí být ale zaznamenána daná kontrolovaná pozice.

Po této části následuje rozhodovací blok, který zjišťuje, jestli je pole neseřazených hran prázdné, pokud se tak stane, je jasné, že v poli seřazených hran jsou zaznamenány všechny hrany daného polygonu a algoritmus do souboru zapíše obsah pole do požadovaného formátu pro výpis Line Loop a Plane Surface, který vytváří plošku polygonu a následně se algoritmus zaměří na další prohledávání hran dalšího polygonu.

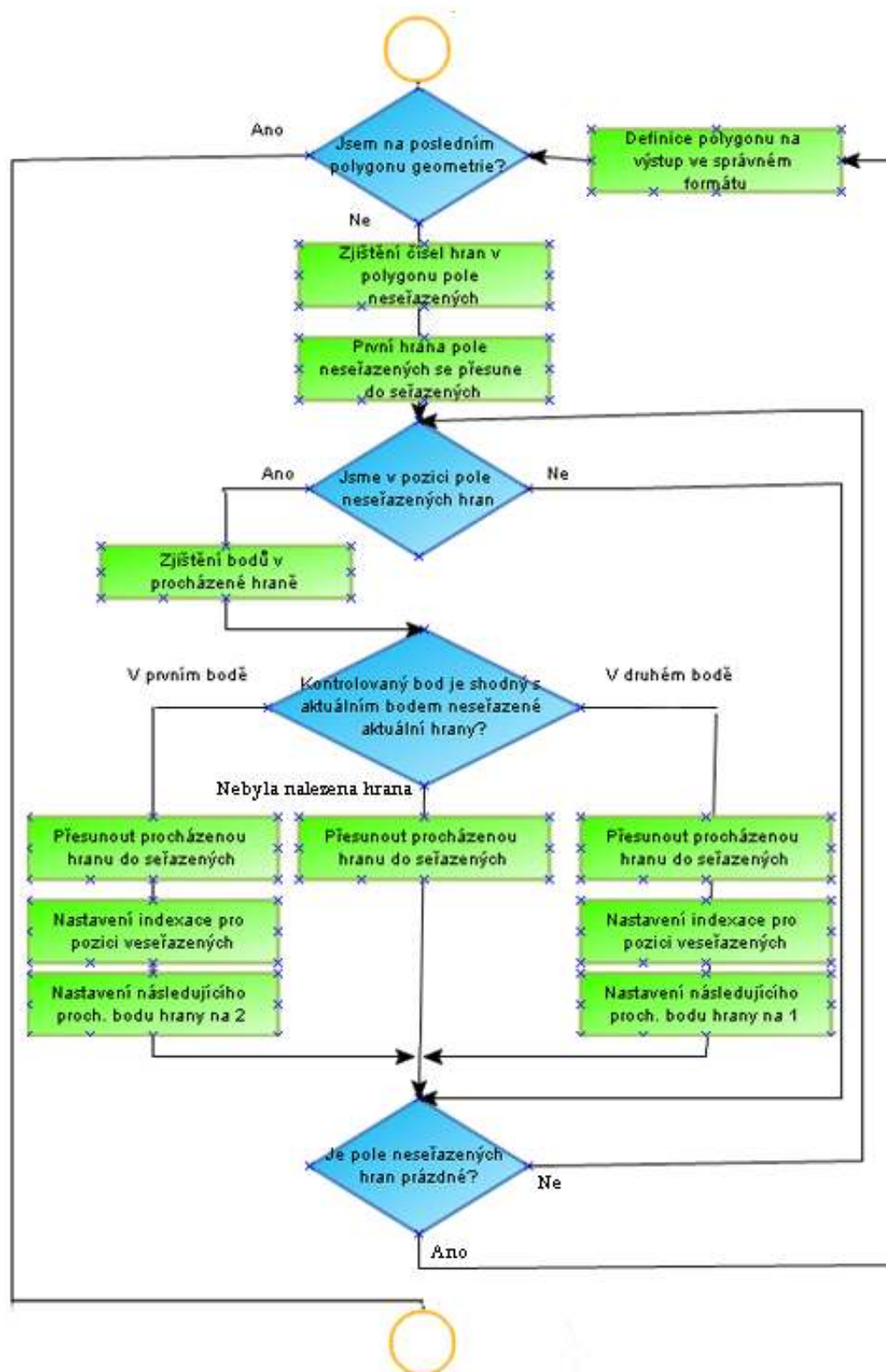
Výpis do souboru pak vypadá např. (**Line Loop(454)={12,-123,2,-82}**; tedy číslo line loop a seřazené hrany a **Plane Surface(455)={454}**; tedy číslo plošky a které lineloop se daná ploška týká). Pokud pole neseřazených prázdné není, algoritmus se vrátí k prohledávání hran daného pole, přičemž se znovu nastaví řídící dynamické proměnné tohoto algoritmu a znovu celé pole projde, toto prohledávání pokračuje tak dlouho, dokud není toto pole prázdné, může se tedy stát, že celé pole se prohledá

několikrát, dokud se všechny hrany seřadí, závisí hlavně na tom, kolik má daný polygon hran a jak jsou vzájemně orientovány.

Tento nástin popisu není zcela kompletní, pro jeho komplexní pochopení je nutné prostudovat samotný zdrojový kód, protože dynamika procesu řešení je poměrně komplikovaná a aby se došlo k požadovanému výsledku definice Line Loop, řídicí proměnné se v jeho průběhu tohoto procesu několikrát mění.



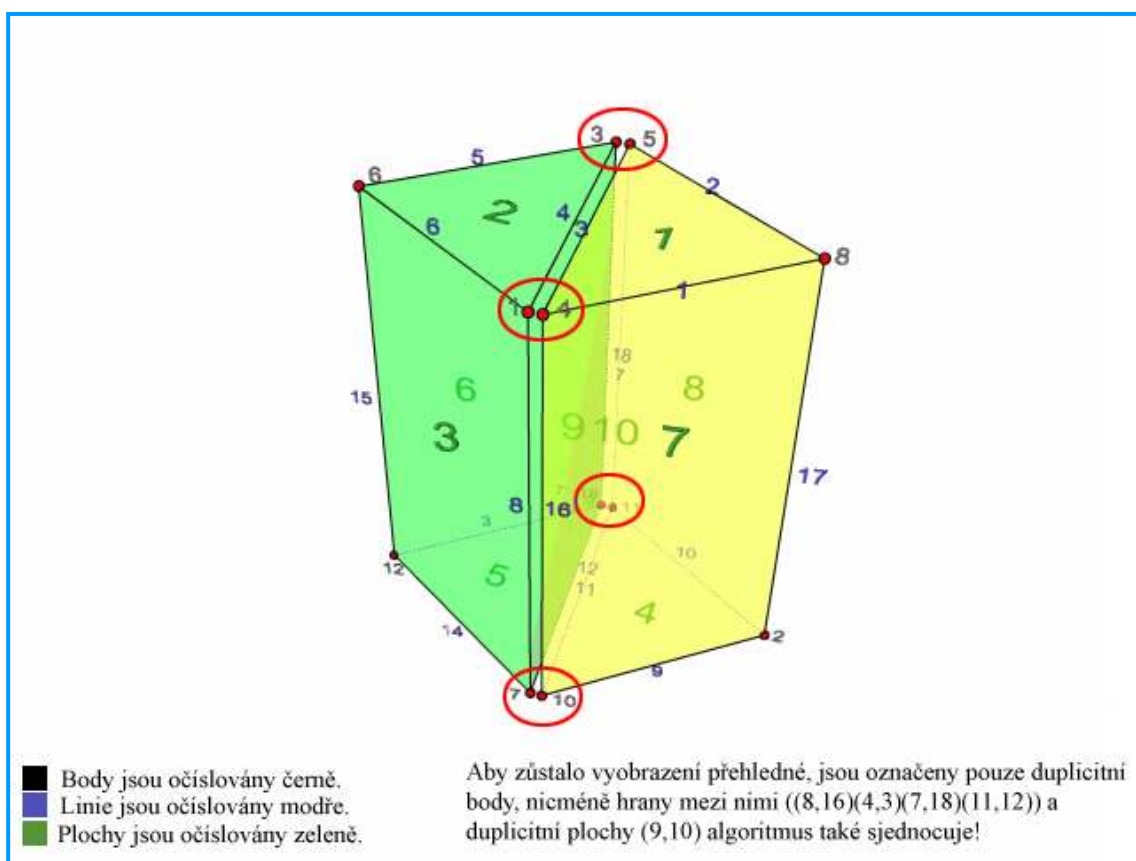
Obr.8.2: Vývojový diagram první část exportu pro definici bodů a linií.



Obr.8.3: Vývojový diagram pro řazení linií a Line Loop.

8.4 Popis algoritmu pro objemový model

Druhé řešení exportu modelové sítě do programu gms, duální prvky (body, linie a plochy) také započítává, nicméně algoritmus prochází procesem eliminace, kdy se právě takovéto duální prvky seskupují a při generování *.geo souboru k jejich výpisu nedochází. Seskupování neprobíhá fyzicky (v okně scény), ale virtuálně. Pro pochopení problematiky je nutné přečíst předcházející kapitolu, která seznamuje s tím, jak se vypisují jednotlivé prvky do souboru *.geo. Algoritmus je zde odlišný, protože seskupování a hledání duplicit je spolu s pravidly pro výpis do souboru komplikovaný. Množiny stejných prvků jsou hledány na základě stejných vlastností (stejně souřadnice bodů), protože linie se skládají z bodů a lze takto definovat i plochy. Vzhledem k tomu, že modelová síť neobsahuje jen trojúhelníkové prvky, nelze se opřít o nějakou jednotnou vlastnost ploch, je tedy nutné aby byl algoritmus schopen analyzovat různou topologii ploch.



Obr. 8.4: Příklad zobrazení prvků, které jsou pro sjednocování hledány.

Algoritmus pro správnou reprezentaci objemů pro generování *.geo souboru má za úkol:

- Dokázat vyhledat body s identickými prostorovými souřadnicemi a virtuálně je seskupit pod jednotný identifikátor.
- Musí být zachována logika stavby a posun indexů bodů, které mají být do geo souboru zapisovány, přičemž důležitá je zpětná vazba, která plní důležitou úlohu při odkazování na reálné prvky ve scéně.
- Dokázat identifikovat duální linie a virtuálně je seskupovat pod jednoznačný identifikátor, identifikace je řešena na základě bodů, které mají stejné souřadnice a mají být vypsány.
- V případě linie, která obsahuje bod, který byl seskupen s nějakým jiným bodem, popř. se skupinou bodů a byl tak změněn jeho index, vypsát jeho index správně.
- V případě duplicitních linií vypisovat do geosouboru správné číslo linie i změněných bodů, které ji tvoří.
- Dokázat identifikovat Line Loop, tzv. správný směr linií pro definici plošky pro gmsh (orientace a směr všech hran tvořících plošku), tedy správně je seskupovat podle bodů těchto linií s využitím zápornosti zápisu a také opět s ohledem na formátování souboru.
- Vyhledávat duplicitní plochy, opět je virtuálně seskupovat, přičemž musí se opět odkazovat na reálný stav ve scéně a zapisovat je ve správném formátu změněných indexů.
- Správně identifikovat plochy, které budou tvořit objemy (Surface Loop a Volume) podle nastavených čísel elementů, které byly nastaveny automatizovaně při stavbě objemů s ohledem na formátování *.geo souboru.

Tato konkrétní část této práce je velmi komplexní, proto byl script oddělen od hlavního scriptu a je prezentován jako třetí samostatný celek. Obsahuje cca 1600 řádků kódu a nelze je jednoduše popsat, protože se neustále odkazuje na jiné prvky. Vývojový diagram obsahuje pouze nejdůležitější bloky a vystihuje alespoň základ řešeného problému.



Obr. 8.5: Sestavení základních bloků scriptu pro analýzu modelové sítě.

8.4.1 Nástin zápisu správných prvků

Zápisu prvků do souboru, jak bylo uvedeno v minulé kapitole, předchází vždy analýza, kde se hledají duplicitní prvky a seskupují se virtuálně pod tzv. pseudoindexy, zároveň se vždy označí první prvek (jeho reálný index), který do dané skupiny pseudoindexů patří. V části scriptu, který pak zajišťuje správný zápis do *.geo souboru se pak zapisují ty prvky, které buď nepatří do žádné takové skupiny, nebo takové, které do některé pseudo skupiny patří, ale je zapsán do této skupiny jako první, ostatní prvky, tedy ty, které patří do některé pseudoskupiny jsou z výpisu vynechány, přičemž platí, že všechny vypisované indexy jsou posloupností za sebou jdoucích čísel, to znamená, že tam, kde dochází k přeskočení přes prvek, který nemá být vypisován dojde k přečíslování vypisovaného indexu prvku. Toto platí pro Points, Line, LineLoop, Surface. Speciálně u Line, Line Loop a Surface je proces složitější, protože se algoritmus odkazuje nejen na sebe sama (přepisuje vlastní indexy), ale odkazuje a hledá přepisované indexy bodů (definování stavby geometrie v 3dsMax). V případě Line Loop pak dochází ještě k sestavování správné kombinace linií pro správnou orientaci a návaznost bodů plošek dané modelové sítě tak jak to žádá formátování generovaného souboru pro definici Line Loop (viz. začátek této kapitoly).

Následující tabulky na další stránce jsou příkladem modelové sítě v 3dsMax, tak jak byla vyobrazena na obrázku 8.4 v této kapitole a vyobrazení již převedené modelové sítě tak jak vychází z generovaného *.geo souboru pro Gmsh.

V každé části tabulky (body, linie, plochy), vždy první řádek **Reálné** odpovídá číslům jednotlivých prvků a poslední řádek **Správné**, odpovídá vypisovanému prvků do souboru.

Například bod, který je ve scéně 3dsMax označen jako bod číslo 12, odpovídá vypsanému bodu číslu 8 ve vygenerovaném souboru.

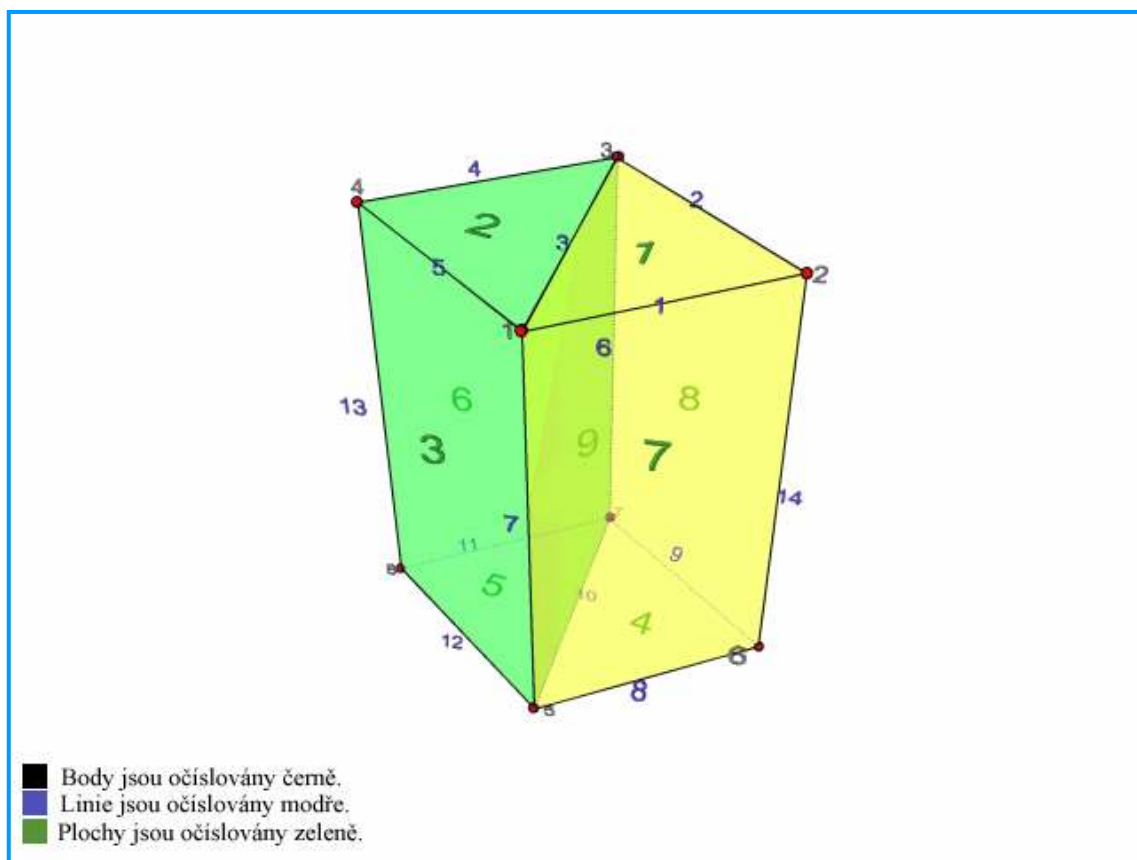
Řádek tabulky **Virtuální** (body, linie, plochy), jsou obsažena čísla těch prvků, které se budou vypisovat. Řádek **Pseudo ID** seskupuje stejné prvky a **První ze skupin** označuje první prvek který je v dané skupině. Řádek **Správné** pak ukazuje jaké prvky se budou vypisovat a znázorňuje, které pseudo prvky ukazují na první prvek dané skupiny.

Tabulka 8.6: Přepisy bodů, linií a ploch modelové sítě.

BODY												
Reálné	1	2	3	4	5	6	7	8	9	10	11	12
Virtuální	1	2	3	X	X	4	5	6	7	X	X	8
Pseudo ID	10		20	10	20		30		40	30	40	
První ze skupin	A		A	N	N		A		A	N	N	
Správné	1	2	3	1	3	4	5	6	7	5	7	8

LINIE																			
Reálné	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	
Virtuální	1	2	3	X		4	5	6	7	8	9	10	X	11	12	13	X	14	X
Pseudo ID				10	10			20	30			40	40			30		20	
První ze skupin				A	N			A	A			A	N			N		N	
Správné	1	2	3	3	4	5	6	7	8	9	10	10	11	12	13	7	14	6	

PLOCHY										
Reálné	1	2	3	4	5	6	7	8	9	10
Virtuální	1	2	3	4	5	6	7	8	9	X
Pseudo ID									10	10
První ze skupin									A	N
Správné	1	2	3	4	5	6	7	8	9	9



Obr. 8.7: Přepočítaný objem odpovídající *.geo souboru.

9 Manuál vytvořené aplikace

Díky poměrně velké komplexnosti vytvořené aplikace je v následující kapitole uveden celkový přehled toho, jak se aplikace ovládá a jakým způsobem je možné s ní v prostředí 3d studia pracovat. Kapitola je rozdělena na jednotlivé podkapitoly, které souhlasí s jednotlivými rollauty v aplikaci, aby byla zajištěna co možná největší přehlednost popisu jednotlivých ovládacích prvků.

9.1 Zpřístupnění scriptu aplikace

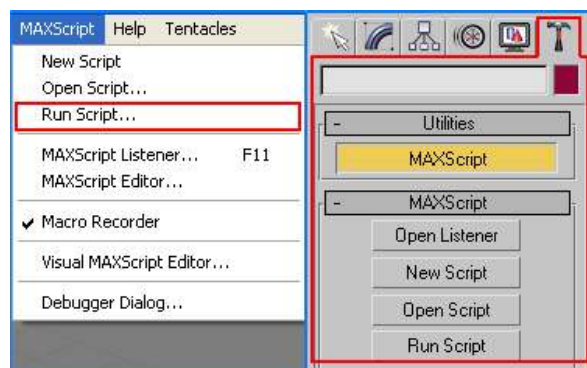
Jak už bylo vysvětleno dříve, vytvořená aplikace je ve formě scriptu. Tyto scripty mají příponu daného souboru *.ms. Jedná se tedy o speciální formát přímo pro 3dsMax, pro zpřístupnění scriptu aplikace je tedy nutné otevřít právě tento program.

Po jeho otevření máme pro spuštění scriptu dvě možnosti odkud je možné daný script rozběhnout.

První možností, jak spustit script s aplikací je přes hlavní lištu záložek programu v horní části obrazovky. Záložka, která je určená pro práci se scripty, ať už jejich tvorba, editace nebo spouštění je záložka s názvem Maxscript. Pro spuštění samotného scriptu je pak určena položka Run script... , po kliku na zobrazí dialogové okno pro otevření scriptu, tedy tzv. open file dialog, který umožní vybrat daný script.

Druhou možností jak rozběhnout script s aplikací je pomocí Command panelu v pravé části obrazovky. Tento panel je centrálním prvkem 3dstudia pro práci s grafikou. Pro spuštění scriptu je určena záložka Utilities a v této záložce je pak nutné vybrat tlačítko Maxscript, otevře se také ovládání pro práci se scripty, stačí pak také jako v prvním případě vybrat tlačítko Run script pro navigaci k danému scriptu.

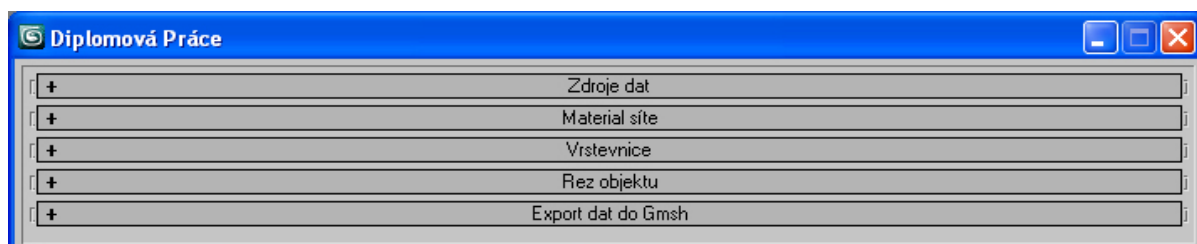
Vytvořený script této diplomové práce má název **Diplomová_práce.ms**.



Obr.9.1: Zpřístupnění scriptu.

9.2 Jednotlivé roletky scriptu

Celé řešení scriptu je rozděleno do jednotlivých oddílů, tzv. rolloutů nebo roletek, které oddělují funkční celky daného řešení. Výhodou použití těchto rolloutů je zejména to, že se dají tzv. sbalit nebo rozbalit klikem na jejich názvy. Název rolloutu je umístěn vždy do jeho hlavičky, na kterou se dá kliknout, čímž právě sbalíme nebo rozbalíme daný rollout. Hlavní výhodou takovéto konstrukce je zejména to, že se tak dá ušetřit místo na pracovní ploše a script tak zbytečně nezakrývá plochu obrazovky, je tak také zaručena větší přehlednost v aplikaci.



Obr.9.2: Jednotlivé rollouty aplikace.

9.3 Posouvání uvnitř roletek

Navigace uvnitř roletek se provádí jejich posuvem je důležitá především v tom případě, že se obsah roletky, která je právě používána nevejde do velikosti okna, řešením tohoto problému je roztáhnutí celého okna aplikace do větší výšky. Toto řešení však může být nepraktické z hlediska zabránění většího místa na pracovní ploše, každá roletka, která je rozbalená, je tedy vybavena v pravé části posuvníkem, pomocí kterého je možné obsah dané roletky posouvat aniž by okno aplikace zabíralo více místa. Dalším způsobem jak posouvat obsah roletky je kliknout do prostoru roletky a rolovat kolečkem myši, posuvník bude reagovat podle směru rolování.

9.4 Rolletka - Zdroje dat

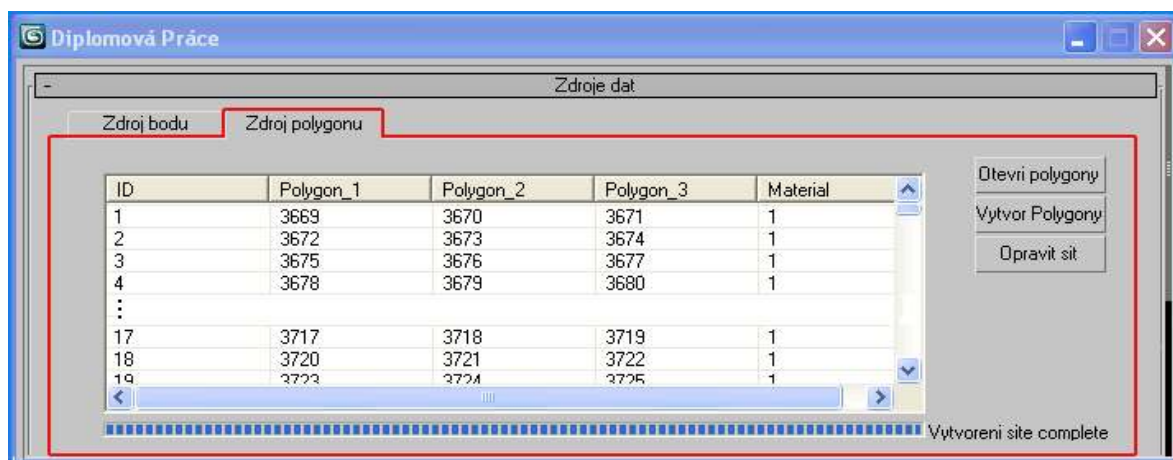
Tato roletka je první ve vytvořené aplikaci. V ní jsou umístěny ovládací prvky jak pro import dat ze zdrojových souborů, tak i tvorbě samotné geometrické trojúhelníkové sítě nebo jejich opravu.

Jelikož jsou pro tvorbu sítě potřeba dva zdrojové soubory (data o bodech, data o polygonech) zvolil jsem pro ušetření místa záložkové menu pro otevření obou souborů, navigace je tedy velmi jednoduchá.

Záložka **Zdroj bodu** obsahuje dvě tlačítka, tlačítkem **Otevři points** vyberete po objevení open file dialogu zvolený soubor dat ve formátu **.xls**, tedy formátu excel nebo ve formátu **.dbf**, která obsahují data o jednotlivých bodech dané trojúhelníkové sítě, data se tak začnou načítat do aplikace, průběh načítání je vidět ve spodní části roletky, kterou zobrazuje progress bar, po dokončení načítání se objeví nápis **načítání hotovo** a také se pro přehlednost v připravené komponentě zobrazí data daného souboru.

Druhé tlačítko **Vytvoř vertex** je určeno pro vytvoření daných dat, tedy všech bodů/vertexů, kterou jsou v souborech zaneseny. Jelikož může být proces tvorby vertexů zdlouhavý, což záleží hlavně na počtu bodů v souboru, je proces tvorby opět jako v případě načítání zobrazen v progress baru ve spodní části roletky.

Záložka **Zdroj polygonu** je co do obsahu podobná, obsahuje tlačítka **Otevři polygony**, **Vytvoř polygony**, které plní stejnou funkci jako v případě první záložky, tedy načtení dat a dokončení tvorby sítě. Tato záložka obsahuje ještě tlačítko **Opravit síť**, které v případě nedotvoření korektní sítě opraví na základě analýzy chybná místa.



Obr.9.3: Roletka Zdroje dat a záložka Zdroj polygonu s jejím obsahem.

9.5 Rolletka – Úprava a materiál sítě

V pořadí druhou roletkou, která je v aplikaci zařazena je roletka Úprava a materiál sítě, která slouží zejména k úpravě vytvořené geometrické sítě importované ze zdrojových dat.

Tato roletka je rozdělena do dvou stěžejních částí, z nichž druhá polovina se ještě dělí na další podcelky.

9.5.1 Objekty scény

Levá část obrazovky obsahuje malé okno, které obsahuje všechny objekty scény a jeho prostřednictvím se také dostáváme k dalším možnostem jednotlivých objektů.

Každý záznam v tomto okně v řádku odpovídá jednomu objektu scény, přičemž je složen v podstatě ze tří součástí.

První částí záznamu je checker, tedy rámeček k zaškrtnutí, který plní ve scéně hlavně funkci přehledovou, při vytvoření geometrie je záznam objektu v okně se zaškrtnutým rámečkem, znamená to, že objekt je ve scéně viditelný, pokud však rámeček tzv. odškrtneme, objekt se ve scéně stane neviditelným. Tímto způsobem se tedy dá velice rychle ty objekty, se kterými právě nepracujeme a které by mohli buď svým umístěním nebo počtem geometrií ve scéně rušit v práci jednoduše schovat tak nebo popřípadě opět zviditelnit pro další práci.



Obr.9.4: Levá část roletky Úprava a materiál sítě, s objekty sítě, světla

Druhou částí záznamu v tomto přehledovém okně je malý obrázek, který pomáhá rychleji se zorientovat mezi jednotlivými objekty v okně. Rozlišení spočívá v podstatě ve čtyřech různých obrázcích, které pomáhají v rozlišení objektů mezi skupinami. Jedná se o obrázek geometrie, světla, kamery a helperu ve scéně.

Třetí a poslední část záznamu objektu je nejdůležitější, protože jeho prostřednictvím se dá dostat k dalším možnostem práce s importovanou geometrií. Tato část obsahuje samotný název objektu a tím tedy zásadně přispívá k rozlišení mezi ostatními, každý z těchto záznamů je však ještě doplněn o barevnou složku, která koresponduje s barvou sítě daného objektu a tím tedy ještě pomáhá rozlišit jednotlivé podobné objekty mezi sebou. Klikem na tento záznam se aktivuje pravá část roletky, která obsahuje záložkové menu, které některé možnosti další práce s jednotlivými objekty.

Poslední složkou levé části roletky je tlačítko Refresh, pomocí kterého se obnovuje právě toto okno se záznamy. Je tedy důležité, aby po vytvoření nové geometrie bylo v případě potřeby práce s pravou částí roletky bylo kliknuto na toto tlačítko, které aktualizuje záznam v okně a zpřístupní tak možnost dostat se k vlastnostem pravé části roletky.

9.5.2 Aktivace pravé části obrazovky

Jak už bylo řečeno výše, k vlastnostem sítě se dostaneme klikem na název dané geometrie v okně objektů scény v roletce Úprava a materiál sítě. Je třeba upozornit, že možnosti další práce se objeví jen v případě, že se jedná o geometrie polygonového typu, znamená to, že aplikace nebude reagovat na klik záznamu typu světla nebo kamery, protože možnosti práce v pravé části roletky se týká tohoto typu objektu. Aplikace toto rozpozná a podle toho se objeví nebo neobjeví záložkové menu pro další práci. Tyto objekty jsou totiž napojeny na modifikátor Editable poly, který umožňuje práci právě s polygonovými typy geometrie nebo lépe řečeno s takovými, na který je daný modifikátor aplikovatelný.

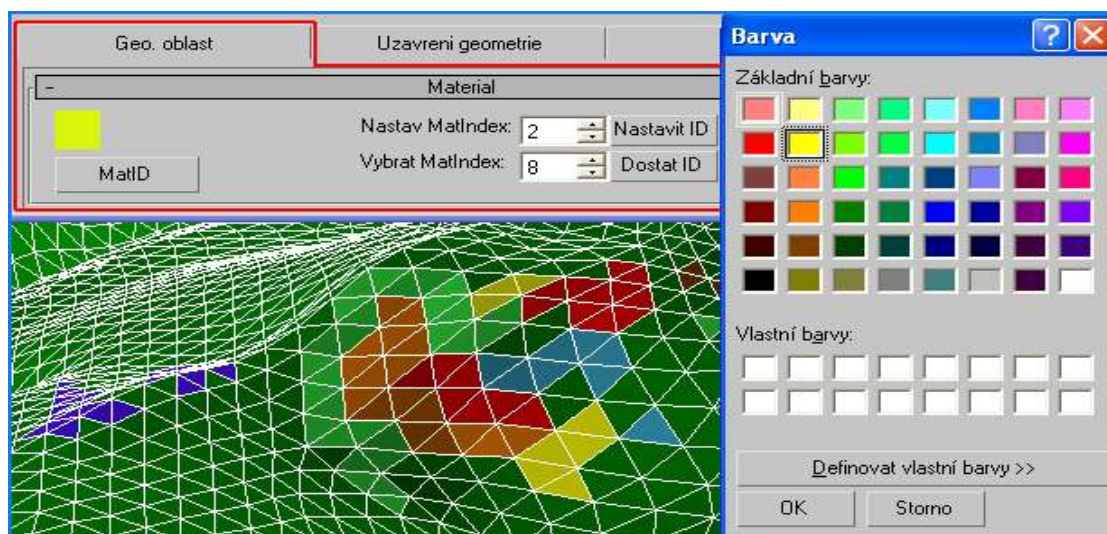
9.5.3 Barvy podskupin polygonů geometrie

První možností práce s danou importovanou geometrií je možnost buď automatizované nebo ruční možnosti přiřazení barev jednotlivým polygonům nebo jejich skupinám. Jedná se o první složku záložkového menu, které se objeví po kliku na záznam dané polygonové geometrie v okna objektů scény v levé části obrazovky.

Jméno dané záložky se mění v závislosti na tom, na který záznam v okně bylo kliknuto. Např. pokud bylo v okně scény kliknuto na záznam Vrstva geometrie, objeví se záložkové menu, kde se první záložka bude jmenovat stejně jako daný objekt, v praxi to znamená, že celé záložkové menu pak bude určeno právě pro práci s touto konkrétní geometrií, pro změnu práce s jiným objektem je pak nutné kliknout na záznam v okně levé části roletky.

Jak už bylo řečeno, změna barvy probíhá buď částečně automatizovaně nebo čistě ručním způsobem. Automatizované přiřazení se provádí už při převodu geometrie a to v případě, že zdrojový soubor týkající se polygonů ve svém pátém sloupci obsahuje číslo dané skupiny kanálu. Jeden kanál může obsahovat buď jen jediný polygon nebo určitou skupinu polygonů, není tak nutné přiřazovat kanály ručně. Pak již stačí přiřadit právě touto záložkou jednotlivé barvy pro každý kanál. Druhou možností je nastavit kanály jednotlivým polygonům ručně nebo již nastavené znovu přenastavit.

Nastavení barvy jednotlivým kanálům se provádí tak, že pokud jsou pomocí automatizace již kanály přednastavené, nejdříve v části záložky s názvem Vybrat MatIndex zvolíme číslo kanálu, kterému chceme barvu přidat a vzápětí klikneme na tlačítko Dostat ID, tímto úkonem se na modelu vyberou ty polygony, které korespondují s daným číslem. Následuje úkon přiřazení barvy, ten se provádí tak, že klikneme na barevný čtverec v levé části záložky a vybereme danou barvu v pro nastavený kanál. Samotné přiřazení pak provedeme klikem na tlačítko MatID, které provede barevnou změnu v modelu pro vybrané polygony. Tímto způsobem pokračujeme v případě všech kanálů, které máme nastaveny.



Obr.9.5: Záložka barvení podskupin polygonu.

V případě čistě ručního přiřazování, kdy zdrojový soubor neobsahoval informace o jednotlivých kanálech nebo v případě přenastavení je nutné kanály nejdříve nastavit. To se provádí tak, že polygony nebo jejich skupiny nejdříve ve scéně označíme. (Provádí se klikem na polygon nebo v případě vybrání skupiny tahem myši, přičemž přičítat výběr se dá pomocí klávesy CTRL a kliku na polygon, v případě odečtu od výběru je to klávesa ALT.) Když máme výběr hotov přiřazení kanálu se provádí v části záložky s názvem Nastav MatIndex, kde zvolíme číslo daného kanálu a samotné přiřazení provedeme tlačítkem Nastavit ID. Přiřazení barvy se pak provádí již standardně jak už bylo napsáno výše.

Důležité v těchto úkonech je fakt, že vytvořený nebo převedený geometrický objekt, který neměl nastavenou hodnotu kanálů ve zdrojovém souboru má defaultně nastavenou hodnotu kanálu všech polygonů na číslo 1. Při přiřazování jednotlivých kanálů je dobré na toto pamatovat, pokud tedy necháte nastaveno na 1 a provedete změnu barvy, změní barvu celé geometrie.

9.5.4 Uzavření geometrie

Druhou záložkou v pořadí v záložkovém menu v pravé části obrazovky je záložka s názvem uzavření geometrie. Jak už název napovídá, jedná se o část aplikace, která se stará o tvorbu uzavřené geometrie, nicméně tato záložka se nepoužívá pro definici objemu jak je definována pro gmsH (trojboké hranoly). Důležitou okolností je fakt, že tato část aplikace dokáže uzavřít jakoukoli geometrii, která se vyznačuje 2,5D, tedy přesně takovým objektem, který vypadá jako vrstva, je tedy sice orientována v 3D prostoru ale nemá objem a vypadá tedy jako jakási vyprofilovaná vrstva.

Ovládání je v tomto případě pro uživatele velice jednoduché, protože má jen několik ovládacích prvků. Operace, které důležité pro uzavření jsou opět napojeny na modifikátor Editable Poly, proto je tato funkce v aplikaci umístěna do záložkového menu.

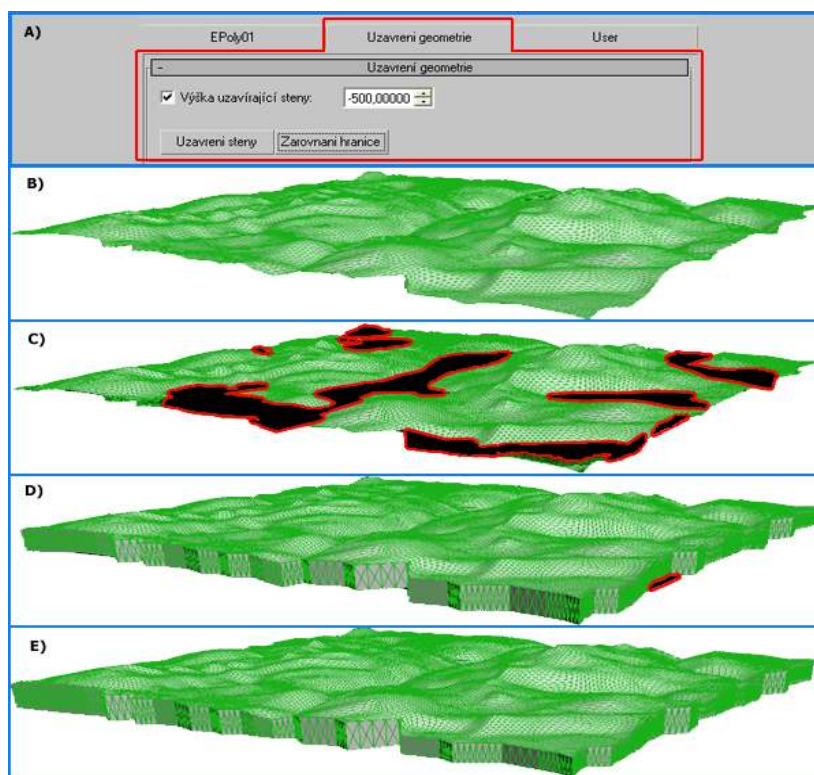
Prvním úkonem pro uživatele je vybrání hranice uzavření v daném modelu, to provedeme tak, že klikneme na jednu z hran, která patří právě do hraniční oblasti uzavření modelu. Dále je uvažováno s dvěma možnostmi, které mohou v daném modelu nastat.

První věcí, která může nastat je taková, že chceme, aby se model uzavřel právě v ohraničující oblasti, to ale jen v případě, že má model takovou morfologii, že nebude

vzniklá plocha překrývat po uzavření jinou část modelu, což by bylo nepříjemné. V takovém případě stačí jen kliknout na tlačítko Uzavření stěny a model se tak automaticky uzavře.

Druhou možností je stav jak už bylo napsáno výše, že by uzavírající plocha modelu překryla nežádoucně jinou jeho část, což je asi naprosto drtivá většina případů. V takovém případě je nutné kliknout na checker/zaškrtnutí v části Výška uzavírající stěny a nastavit číselnou hodnotu bočních hran, které se vytvoří v místech hranice modelu, aby odsadily spodní nebo vrchní uzavírající plochu. Nastavit se dají záporné nebo kladné hodnoty, záporné se zadávají v případě, že uzavírající plochou je dno modelu, kladné pak v případě vrchu modelu.

Posledním ovládacím prvkem je tlačítko zarovnání hranice, které se dá použít před i po vytvoření uzavírající plochy, ale jen v tom případě, že máme stále vybranou hranici uzavírající model! Po kliknutí tlačítkem se nerovnoměrně zarovnané hranice podle osy Z zarovnají, tlačítko se tedy dá použít jen v případě osy Z.



Obr.9.6: a) Záložka Uzavření geometrie s nastavením.

b) Importovaný model 2,5D.

c) Špatné použití záložky.

d) Nezarovnané plochy modelu.

e) Korektní uzavření.

9.6 Rolletka – Vrstevnice

Další částí aplikace je roletka Vrstevnice, tato roletka za základě určitého modelu vytváří linie vrstevnic na základě uživatelského nastavení. Pokud chce uživatel vytvořit linie vrstevnic, stačí prostřednictvím roletky Úprava a materiál sítě v okně pro objekty scény v levé části obrazovky. Pokud je daný objekt geometrií, pak se v roletce **Vrstevnice** v části **Vybraný objekt** objeví jméno dané geometrie, tím je zaručeno, že aplikace bude reagovat na tento model a budou se nad tímto modelem vytvářet vrstevnice. V části roletky **Výška objektu** podle Z se zároveň objeví výška tohoto modelu v jednotkách scény podle Z osy, tato informace je důležitá z hlediska uživatele hlavně pro výpočty počtu vrstevnic podle uživatelského přání. Další částí této roletky je **Počet vrstevnic**, kde se zadává číselná hodnota, podle které se aplikace řídí a podle této hodnoty pak vytvoří určitý počet vrstevnic.

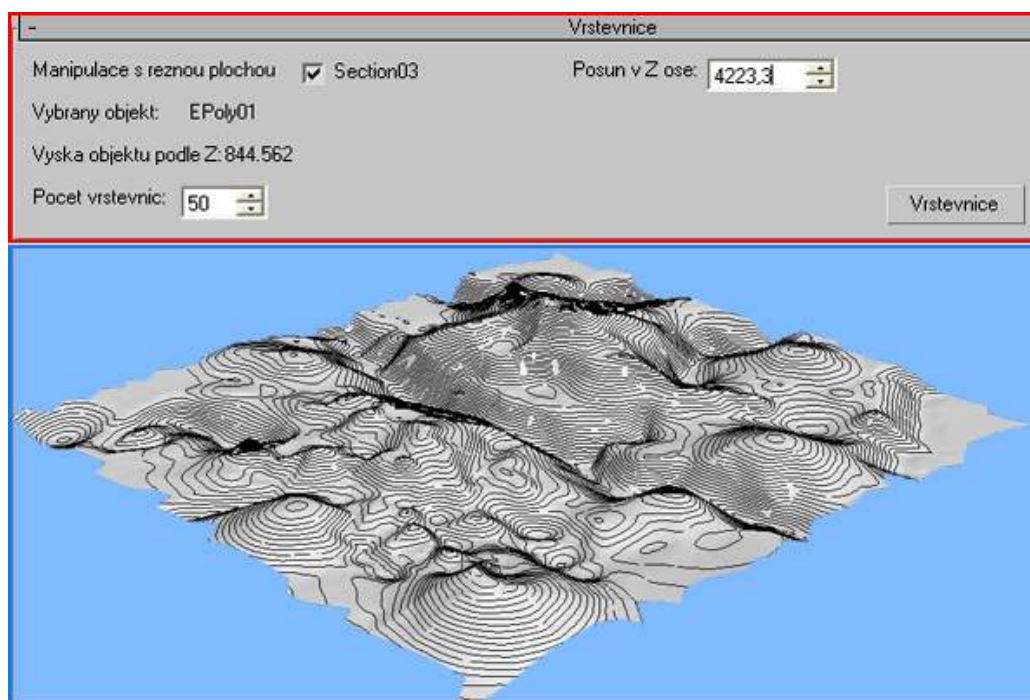
Samotnou tvorbu vrstevnic pak zahájíme klikem na tlačítko **Vrstevnice**.

Velmi důležité je, vrstevnice se začínají vytvářet od spodní části modelu a po vytvoření první vrstevnice pak pokračují vzhůru!

Pokud z jakéhokoli důvodu potřebujeme začít vytvářet vrstevnice od jiného místa v modelu než od jeho spodní hrany, pak je nutné, aby bylo zaškrtnut checker/zaškrťavátko **Manipulace s řeznou plochou**. Tento prvek aktivuje část roletky **Posun v ose Z**, pomocí kterého se dá posouvat počátek vytváření vrstevnic vzhledem k modelu. Pokud tedy zaškrtneme **Manipulaci s řeznou plochou**, objeví se kolem daného modelu rovina, pokud začneme posouvat posuvník v části **Posun v ose Z** nahoru nebo dolů (kladné nebo záporné hodnoty), začne se tato rovina jako počátek tvorby vrstevnic posouvat požadovaným směrem v ose Z. Další postup je pak stejný jako v případě vytváření vrstevnic od spodní strany modelu. Tato funkce je určena hlavně pro tvorbu obrazové dokumentace.

Tip:

Pokud bychom chtěli vytvářet vrstevnice od začátku změn morfologie modelu, je lépe začít vytvářet vrstevnice ještě předtím než model uzavřeme, tedy než uděláme z 2,5D vrstvy importovaného modelu 3D reprezentaci a zvýšíme tak model přídatnými bočními plochami, protože pak by se začaly vrstevnice vytvářet až od spodní části tohoto zvýšeného modelu a bylo by nutné posouvat rovinu vrstevnic do místa změn morfologie. Lze si tak ušetřit práci s posouváním roviny vrstevnic a až pak model uzavřít.



Obr.9.7: Nastavení roletky Vrstevnice a příklad výsledku.

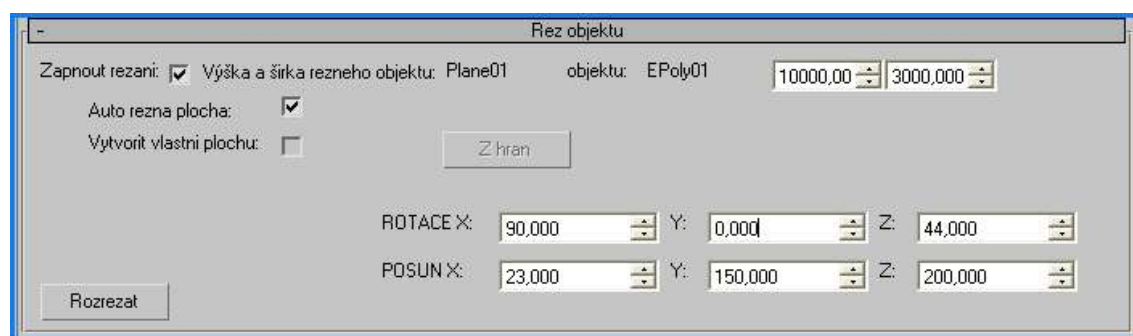
9.7 Roletka - Řez objektu

Pokud potřebujeme importovaný model, který například představuje morfologii krajiny rozdělit na několik částí podle určitých kritérií, pak je pro nás jako uživatele důležitá roletka Řez objektu, tato roletka je totiž určena pro řezání geometrií. Podmínkou pro rozřezání modelu je opět jako v ostatních případech fakt, že objekt, který potřebujeme rozřezat, musí být polygonovou geometrií.

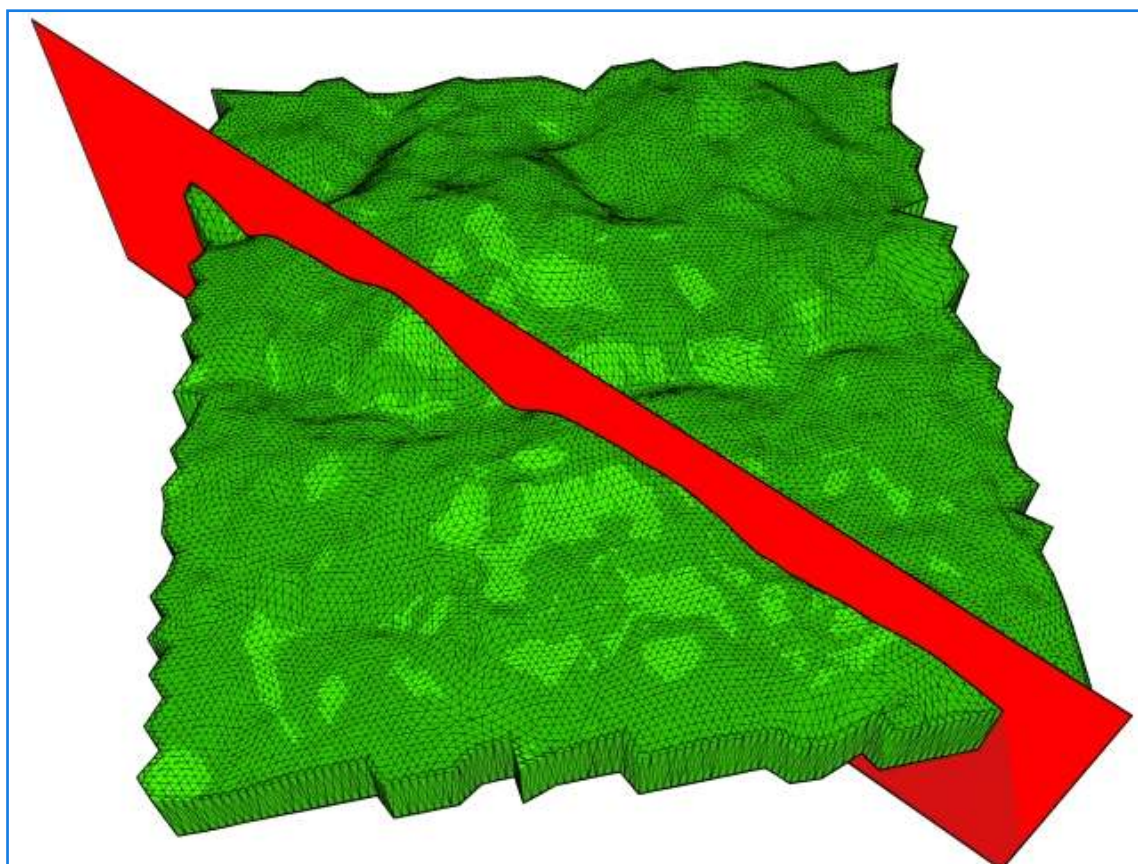
Postup pro úspěšné rozříznutí modelu je následující. Prvním prvkem, je checker/zaškrtnutí **Zapnout řezání**, kdy jeho zaškrtnutím se aktivují dva hlavní ovládací prvky této roletky. Aktivované prvky představují dvě možnosti, jak vybraný model rozřezat, tyto dvě možnosti jsou v roletce zastoupeny checkry s názvem Auto řezná plocha a Vytvořit vlastní plochu. Pokud bychom chtěli vybrat jeden z následujících postupů, je potřeba mít na paměti, že před zaškrtnutím jednoho nebo druhého způsobu řezání je důležité aby byl vybrán objekt který chceme řezat, pokud by se stalo, že není vybrán žádný objekt pro řezání nebo že objekt není typu polygonové geometrie, budeme upozorněni hláškou, která nás na toto upozorní!

9.7.1 Auto řezná plocha

Pokud tedy máme vybrán správný model a zaškrtneme způsob pro řezání právě **Auto řeznou plochu**, pak se provede několik kroků. Prvním krokem je to, že se aktivují další nezbytné prvky pro úspěšné rozřezání modelu tímto vybraným způsobem. Další úkon, který se provede, je to, že se vytvoří řezná plocha, podle které se daný model bude řezat, zároveň se v horní části roletky objeví jména obou objektů, tedy geometrie, kterou chceme řezáním rozdělit a řezné plochy podle které budeme danou geometrii řezat. Velmi důležitým faktem je, že vytvořená řezná rovina musí být v daném místě, kde bude dělit objekt ve všech směrech větší než řezaný objekt, jinak se rozřezání modelu neprovede korektním způsobem! Pokud se stane, že vytvořená řezná plocha bude po zaškrtnutí této metody řezání menší, pak musíme nastavit jinou délku a šířku této roviny, to provedeme zadáním číselné hodnoty posunutím hodnot dvou spinnerů v horní pravé části roletky hned za jménem řezaného objektu, které se objeví po zaškrtnutí této řezací metody. Řezná plocha je zarovnána na spodní část modelu a neprotíná tak žádnou jeho část, důležité je tedy zajistit, aby protínala model. To provedeme ovládacími prvky pro posun a rotaci dané roviny, které jsou umístěny ve spodní části roletky a označeny ROTACE a POSUN, které se aktivují také po zaškrtnutí této metody řezání. Rotaci i posun lze provádět ve všech osách, tedy v osách X,Y,Z, kdy ihned při posouvání hodnot držením spinneru se okamžitě začne model s případně posouvat požadovaným směrem ať už záporných nebo kladných hodnotách (doleva, doprava, nahoru, dolů). V případě rotace řezné roviny je dobré spíše nastavit číselnou hodnotu ručně na klávesnici, protože rotace se provádí z aktuálního úhlu o námi zadaný úhel, znamená to tedy například, že zadáním hodnoty 30 ve spinneru pro osu X rovina zarotuje v ose X o 30 stupňů, pokud bychom následně zadali hodnotu 20, model se posune o dalších 20 stupňů, bude tedy svírat úhel s rovinou scény 50°.



Obr.9.8: Roletka řezné roviny s nastavením metody Auto řezná plocha.



Obr.9.9: Řezná rovinná plocha první metody řezu pro daný model.

Pokud máme danou řeznou rovinu umístěnou v požadovaném místě a protíná v tomto místě model všude, tedy je větší než model, stačí už jen kliknout na tlačítko Rozřezat, které je umístěno v levé spodní části roletky a model se rozřízne na dvě části a zároveň dojde ke smazání řezné roviny a vymazání jmen původního objektu a řezné roviny a nastavení roletky se uvede opět do původního stavu a je tak připravena pro znovupoužití.

9.7.2 Vytvořit vlastní plochu

Pokud zvolíme druhou metodu řezání a zaškrtneme tedy možnost **Vytvořit vlastní plochu**, v roletce se pouze objeví jméno objektu, který chceme řezat jak tomu bylo i v případě první metody, pak se ale kroky vedoucí k rozřezání liší, protože se řezná rovina vytváří naprosto odlišným způsobem, než tomu bylo v prvním případě a neobjeví se tak tedy ihned po zakliknutí této metody.

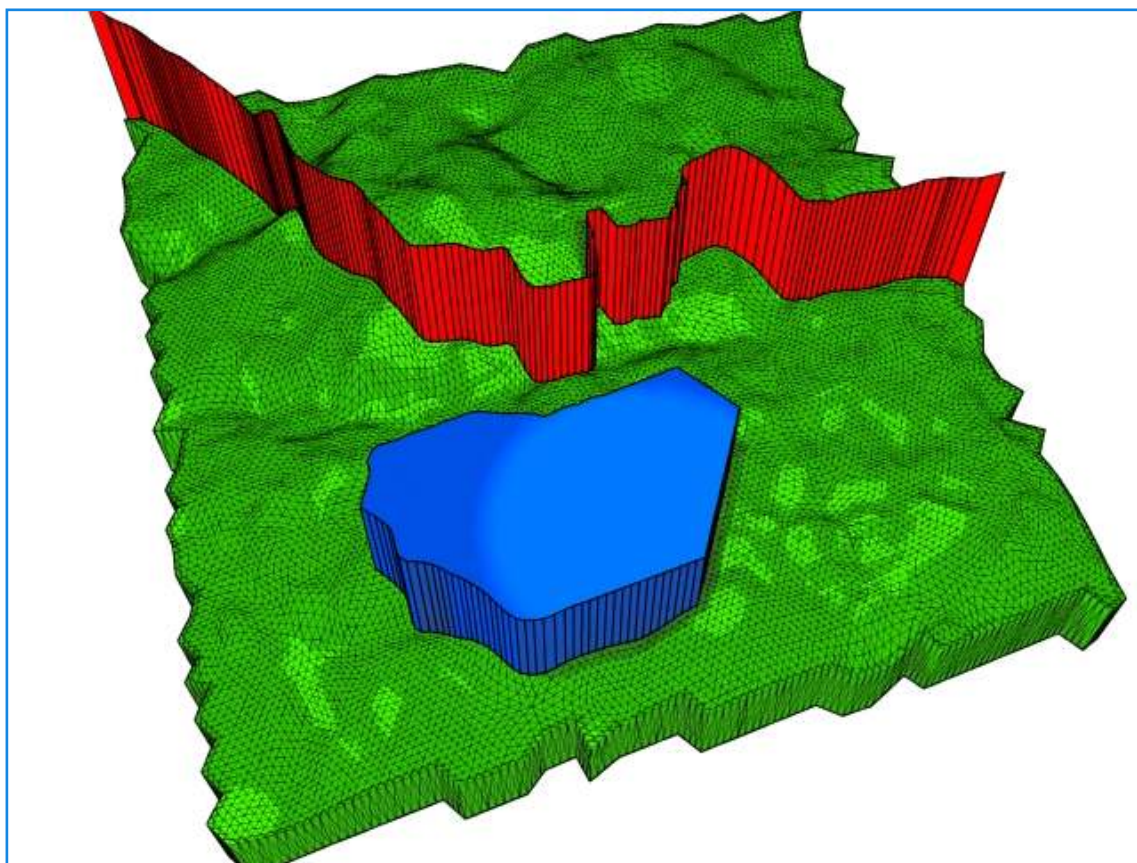
Co je pro tuto metodu důležité je, že je určena pro takové řezání, kdy chceme model rozdělit podle linií/hran daného geometrického modelu! Dá se tedy využít hlavně v případě, že model chceme rozdělit podle geolinií, které jsou reprezentovány některými z hran modelu. Je potřeba říci, že po zaškrtnutí této metody se aktivují jiné ovládací prvky než tomu bylo u metody první.

Pokud chceme tedy vytvořit řeznou rovinu podle hran, musíme nejdříve dané hrany modelu vybrat. Vybrání provádíme klikem na hrany modelu, přičemž neustále držíme klávesu CTRL, která je v programu 3dsmax určena pro přičítání výběru! Vybírání hran pro tvorbu vlastní řezné roviny je podmíněno několika pravidly!

Prvním pravidlem je, že musíme vybrat kontinuální řadu hran, tedy hran, které na sebe navzájem navazují, nesmí se tedy stát, že bude mezi vybranými hranami mezera!

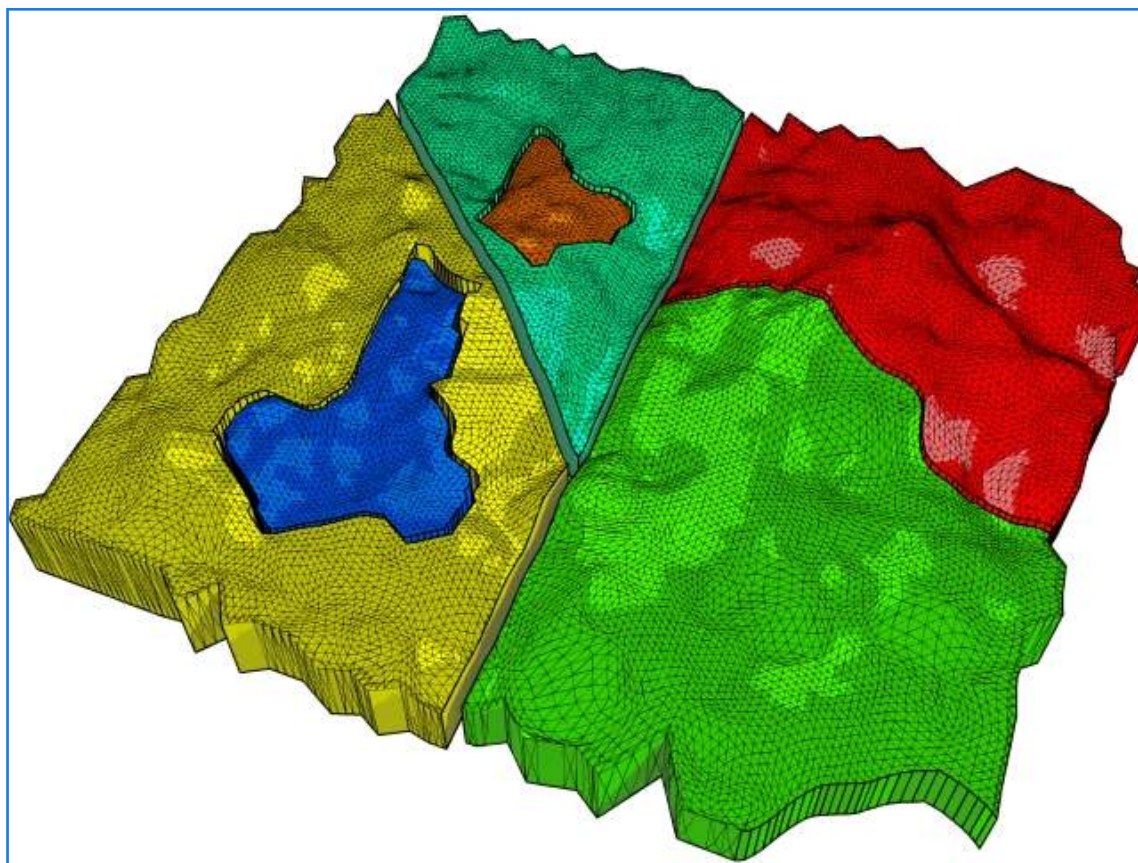
Druhým pravidlem je, že vybrané hrany musí pokud chceme model rozříznout napříč, musí být předpokládaná rovina alespoň tak dlouhá jako je model!

Třetím pravidlem je, že pokud chceme z modelu vyříznout například ohraničenou oblast, která je uprostřed modelu, pak musíme vybrat hrany, které tuto oblast ohraničují, přičemž musíme vybrat opět kontinuální hrany, tedy v tomto případě uzavřenou oblast vybraných hran!



Obr.9.10: Řezná rovina příčného a cyklického vybrání hran a ploch.

Pokud máme dané hrany již vybrány, musíme z těchto hran vytvořit tedy řeznou plochu, to provedeme tak, že nejdříve zadáme číselnou hodnotu do spinneru, který byl u první metody řezání určen k definování délky plochy, v tomto případě definuje výšku řezné plochy, přičemž platí, že je lépe zadat i nepřiměřeně velkou hodnotu, aby vytvořená plocha byla opět zaručeně větší a překrývala model! Po tomto zadání klikneme na tlačítko Z hran, které danou řeznou plochu podle naší definice vytvoří a jméno řezné plochy se objeví opět v horní části roletky, pak již stačí jen kliknout na tlačítko Rozřezat, a model se rozdělí podle námi definované hrany, následně řezná plocha zmizí i se jmény objektu a řezné plochy v horní části roletky a celé její nastavení se uvede do počátečního stavu, kdy je připravena na potenciální další řezání.



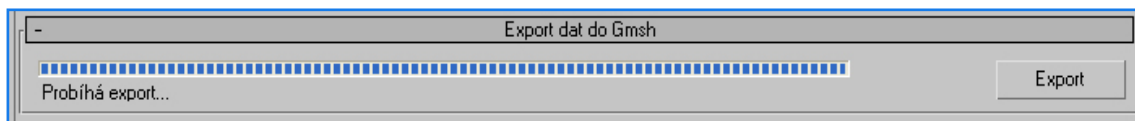
Obr.9.11: Možný výsledek řezání modelu.

9.8 Roletka – Export dat pro Gmsh

Poslední roletkou aplikace je Export dat pro Gmsh a je určena jako prostředek k exportu dat geometrických objektů z 3dsmax do programu Gmsh, což je volně dostupný 3d generátor konečných prvků sítě. Pro export geometrického modelu je důležité, aby na daný objekt byl aplikován modifikátor Editable Poly, protože algoritmus pro převod využívá některé vlastnosti právě tohoto modifikátoru. Jak už bylo uvedeno výše, importované modely které můžeme převést do 3dsmax mohou být jen složeny z trojúhelníkových polygonů, algoritmus pro export z 3dsmax je navržen tak, že může být převedena jakákoli geometrie, počínaje trojúhelníkovými polygony ale horní hranice počtu vrcholů polygonů není omezen, můžeme tedy převést libovolný model, není ani nutné, aby model obsahoval jen např. čtyř, pěti, šesti úhelníkové polygony, ale může obsahovat polygon od polygonu různý počet vrcholů, což je nesporná výhoda, protože to dává téměř neomezené možnosti převodu modelů.

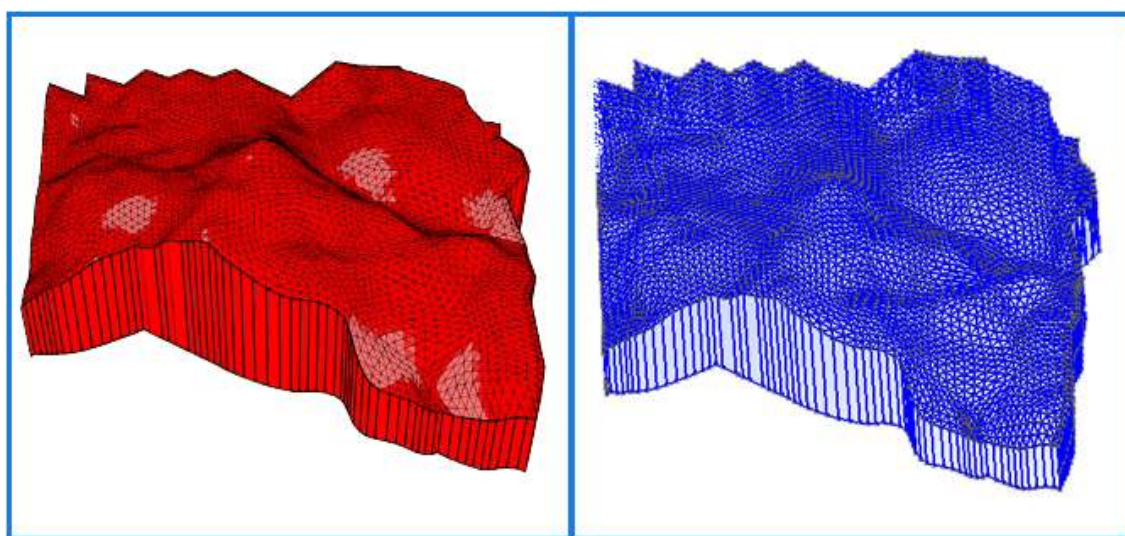
Tato roletka je co se týká uživatelského rozhraní velmi jednoduchá. Co se týká ovládání, obsahuje pouze tlačítko Export. Pro export daného modelu do Gmsh je nutné,

aby byl model vybrán/označen, pak jen stačí kliknout na tlačítko, kdy se objeví dialog pro uložení souboru.



Obr.9.12: Zobrazení roletky pro export modelu do programu Gmsh.

Co se týká možností uložení do různých typů souboru, tak je možné ukládat buď do souboru formátu *.txt, tedy klasického textového formátu, tak i do formátu *.geo. Po zvolení jména a typu souboru který se má uložit se začne pomocí algoritmu pro převod automaticky soubor zpracovávat. Průběh převodu daného modelu je průběžně zobrazován na progress baru, který je v roletce pro převod také obsažen, v případě dokončení převodu se zobrazí hláška o dokončení převodu vybraného modelu. Pak již stačí soubor otevřít přímo v programu gmsh v hlavní liště programu pomocí File-Open... a model se zobrazí v prostředí programu.



Obr.9.13: Příklad výsledku exportu z 3dsmax do Gmsh.

Toto řešení exportu však dokáže exportovat pouze plochy, objemy vygenerovat do souboru nedokáže. Jedná se o jednodušší verzi exportu. Druhá verze exportu už dokáže objemy do souboru *.geo zapisovat (oddělené trojboké hranoly). Aby to však bylo možné, je důležité tyto struktury z importovaného základu vytvořit.

9.9 Vytvoření objemů

Pro vytvoření objemů se využívá samostatného scriptu, ten je velmi jednoduchý na ovládání, obsahuje pouze jedno tlačítko a jeden číselný posuvník, pro vytvoření objemu musíme mít vybrán importovaný základ geometrie a musí být aplikován modifikátor Editable Poly, kliknutím na tlačítko **Vytvořit objem**, se provede rozdělení všech spojitých plošek a nastaví se jednotlivá ID pro pozdější rozlišování vzniklých objemů. Zároveň dojde vytažení stěn trojúhelníkových polygonů a to na základě číselného údaje, který je uživatelsky zadáván. Poté je aplikován modifikátor symmetry, který vytvoří symetrickou kopii dané geometrie a zároveň svaří vrcholy stěn, takže se spojí a vytvoří trojboké hranoly.

V druhé části je nutný zásah uživatele, je totiž nutné aby byly vymazány body a hrany, které vznikly navíc při aplikování modifikátoru. Nejlépe je odstraníme, pokud se přepneme do čelního pohledu (klávesa F -front) a zaměříme se na náš upravovaný objekt. Manipulace je uvede na např. [19]nebo [21]. Přepneme se do módu edge v pravé části obrazovky v modifikátoru Editable Poly (měly by být vybrány hrany uprostřed trojbokých hranolů, pokud ano pak klikneme na tlačítko Remove v modifikátoru), pokud se stane, že hrany vybrány nejsou, nejdříve se přepneme na položku Vertex a vybereme všechny prostřední body trojbokých hranolů, podržíme klávesu CTRL a klikneme na obrázek Edges (prázdný trojúhelníček) v modifikátoru Editable Poly, tímto úkonem se vyberou hrany, které jsou danými vybranými body spojeny. Bohužel se ale vyberou nejen prostřední hrany. Abychom dosáhli správného výběru, stiskneme klávesu ALT a necháme stlačenou a zároveň vybereme ty hrany které mají být od výběru odečteny. Tak abychom dostali výběr jen prostředních hran. Pokud je všechno v pořádku klikneme na tlačítko Remove a prostřední hrany zmizí.

Pozor, je však důležité smazat ještě body, které po vymazaných hranách vznikly, to provedeme tak, že klikneme na obrázek bodů, modifikátoru Editable Poly, pokud jsou prostřední body vybrány, klikneme na tlačítko Remove správná reprezentace je hotova, pokud body vybrány nejsou, musíme je vybrat ručně, opět je vhodné vybírat body z čelního pohledu, protože máme lepší výhled na geometrii.

Vzniklou modelovou síť je však ještě nutné seskupit do původního tvaru, to provedeme pomocí tlačítka seskupit. Po těchto krocích je modelová síť připravena na převod do souboru formátu *.geo.

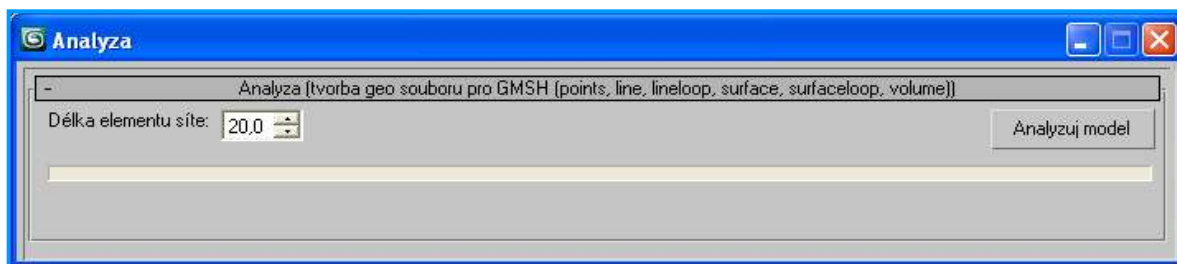
9.10 Export definovaných objemů do *.geo

Pokud jsme provedli všechny kroky předcházející kapitoly korektně, je modelová síť připravena pro převod. Script, který se stará o převod je jmenuje **Kompletace exportu do gmsb** a po jeho překladu je ihned možné ho začít používat.

Pokud máme danou modelovou síť vybránu a je na ní aplikován modifikátor Editabe Poly, je možné klikem na tlačítko **Analýzuj model** ihned zahájit analýzu a výpis do souboru, kdy připravený progress bar ukazuje provedené kroky.

Algoritmus prohledává duální prvky dané geometrie (Points, Line, Line Loop, Surface, Surface Loop a Volume) a vypisuje jen ty, které jsou nezbytné pro správnou reprezentaci modelové sítě v programu Gmsb.

Pro definování správné konstanty délky elementu je možné nastavení této délky zadáním číselné hodnoty do připraveného spinneru/posuvníku. Celá modelová síť se převede po zobrazení nápisu Complete v pravém spodním rohu aplikace.



Obr.9.14: Přeložený script starající se o převod modelové sítě.

10 Výsledky měření klíčových událostí

V průběhu testování byla vytvořená aplikace zkoušena na několika počítačích, platformách a několika verzích 3ds Max, aby byly otestovány rozdíly hlavně při načítání dat mezi jednotlivými formáty a při zpracování těchto dat.

V aplikaci byly měřeny zejména časy načítání a tvorby elementů. Pro úplnost tabulka obsahuje časy obou typů souborů, tedy *.xls formát pro Excel 2003 a databázový soubor *.dbf, kromě toho je doplněn ještě i informacemi o převedení dané geometrie do formátu *.geo, tedy formátu pro program Gmsh. Testování řezů, vytváření objemů a vrstevnic nebylo měřeno, protože probíhá v řádech milisekund až sekund a rozdíly by nebyly příliš patrné a nelze z nich příliš vyvozovat žádné závěry.

Prvním ze zkoušených počítačů byl notebook s touto konfigurací:

Tab.10.1: Zobrazení konfigurací prvního testovaného počítače.

Operační systém	Windows Xp Profesional service pack 2
Použitý software	3ds Max 2009, 3ds Max 2008, 3ds Max 9
Typ	32 bit
Cpu	Intel
Cpu (počet jader)	1
Cpu (označení)	T1350
Cpu (frekvence)	1,86 GHz
Ram (paměť)	1 GB
Ram (frekvence)	800 MHz

Tab.10.2: Tabulka výsledů pro první konfiguraci.

První zdrojový soubor (Definice bodů geometrie) – 14420 řádků ,3 sloupce dat						
SOUBOR	NAČTENÍ DAT			TVORBA BODU		
	3dsMax 2009	3ds Max2011	3dsMax9	3dsMax 2009	3ds Max2011	3dsMax9
xls	2m 35s	2m 24s	-	0m 11s	0m 12s	-
dbf	0m 14s	0m 16s	-	0m 11s	0m 12s	-
Druhý zdrojový soubor (Definice polygonů geometrie) – 28 336 , 5 sloupců dat						
SOUBOR	NAČTENÍ DAT			TVORBA POLYGONU		
	3dsMax 2009	3ds Max2011	3dsMax9	3dsMax 2009	3ds Max2011	3dsMax9
xls	5m 26s	5m 26s	-	2m 24s	2m 34s	-
dbf	0m 32s	0m 37s	-	2m 24s	2m 34s	-
Export do *.geo pro program Gmsh (Převedení geometrie)						
SOUBOR	NAČTENÍ DAT			OTEVŘENÍ		
	3dsMax 2009	3ds Max2011	3dsMax9	3dsMax 2009	3ds Max2011	3dsMax9
geo	4m 10s	4m 6s	-	sekundy	sekundy	-

Jak je vidět z tabulky, položka vytvoření a její hodnota je stejná v obou případech, protože o tvorbu elementů sítě v 3ds Max řídí jádro programu a typ souboru na tvorbu samotnou nemá žádný vliv, tato položka slouží k porovnání časů mezi jednotlivými sestavami na kterých bylo měřeno.

Z tabulky je také patrné, že pro verzi 3ds Max 9 nebyly časové údaje naměřeny, důvodem je ne zcela kompletní implementace dotNet tříd do prostředí maxscriptu v této verzi programu, kdy některé prvky jsou podporovány, ale ne zcela vše, v této verzi tedy nelze aplikaci využít.

Druhá sestava, na které byla aplikace testována bylo stolní PC. Byl na něm nainstalován operační systém Windows 7 Home ve své 64 bitové verzi, na tomto operačním systému bylo hlavně testováno, jestli se výsledná aplikace bude chovat stejně jako na 32 bitovém systému.

Tab.10.3: Konfigurace druhého testovaného počítače.

Operační systém	Windows 7 Home 64 bit
Použitý software	3ds Max 2011
Typ	64 bit
Cpu	Intel
Cpu (počet jader)	4
Cpu (označení)	i7 870
Cpu (frekvence)	2,93 GHz
Ram (paměť)	4 GB
Ram (frekvence)	1600 Mhz

Tab.10.4: Výsledky pro druhou konfiguraci.

První zdrojový soubor (Definice bodů geometrie) – 14420 řádků ,3 sloupce dat		
SOUBOR	NAČTENÍ DAT	TVORBA BODU
	3ds Max 2011	3ds Max 2011
xls	2m 12s	8s
dbf	-	-
Druhý zdrojový soubor (Definice polygonů geometrie) – 28 336 , 5 sloupců dat		
SOUBOR	NAČTENÍ DAT	TVORBA POLYGONU
xls	5m 21s	1m 1s
dbf	-	-
Export do *.geo pro program Gmsh (Převedení geometrie)		
SOUBOR	NAČTENÍ DAT	OTEVŘENÍ
geo	1m 52s	V řádech sekund

Díky implementaci dotNet prvků nedošlo k žádným problémům v oblasti ovládacích prvků a aplikace se chovala standardně. Nicméně k problémům došlo v části načítání *.dbf, kdy aplikace nerozeznala ovladač ODBC, který zprostředkovává komunikaci s tímto formátem. Důvodem tohoto problému je fakt, že ovladač ODBC je primárně kompilován pro 32 bitové systémy a pro 64 bitový systém Windows 7 není bohužel podporován.

Z výsledků, které jsou zde zobrazeny a pozorování, které bylo prováděno v průběhu tvorby a ladění aplikace lze prohlásit, že načítání dat z externích souborů není příliš ovlivněno rychlostí CPU ani počtem jader, k jistému zrychlení v oblasti načítání sice dochází, ale nelze ho přičíst rychlosti CPU, ale především zpracování jádra programu 3ds max. Co se týká samotného procesu tvorby a převodu povrchových sítí do formátu geo, je vidět významnému urychlení.

11 Závěr

V této diplomové práci na téma varianty řešení automatizace postupů v tvorbě modelových sítí, ve které se zabývám převodem dat z GIS (Geografický informační systém), který není CAD systémem, se mi podařilo úspěšně tato data zobrazit na základě aplikace, jež byla vytvořena za účelem převodu, úpravy a dalším zpracováním vizualizovaného krajinného modelu. Aplikace byla vytvořena pod vybraným 3d modelovacím programem 3ds Max, který je jedním ze zástupců CAD systémů, což samo o sobě přináší problémy při převodu a vizualizaci, protože model je reprezentován a vystaven podle odlišných pravidel. Zároveň ale umožňuje vytvářet objemové prvky, které v GIS není možné vytvářet. 3dsMax obsahuje vimplementovaný scriptovací jazyk maxscript, na jehož základě byla aplikace vytvořena.

Vytvořené scripty, které jsou uloženy na přiloženém CD obsahují okolo třiapůl tisíce řádků kódu, mají vimplementovány dotNet prvky. Ty dovolují ovládat aplikaci lépe než pomocí prvků, jež je možné využít přímo v 3ds max.

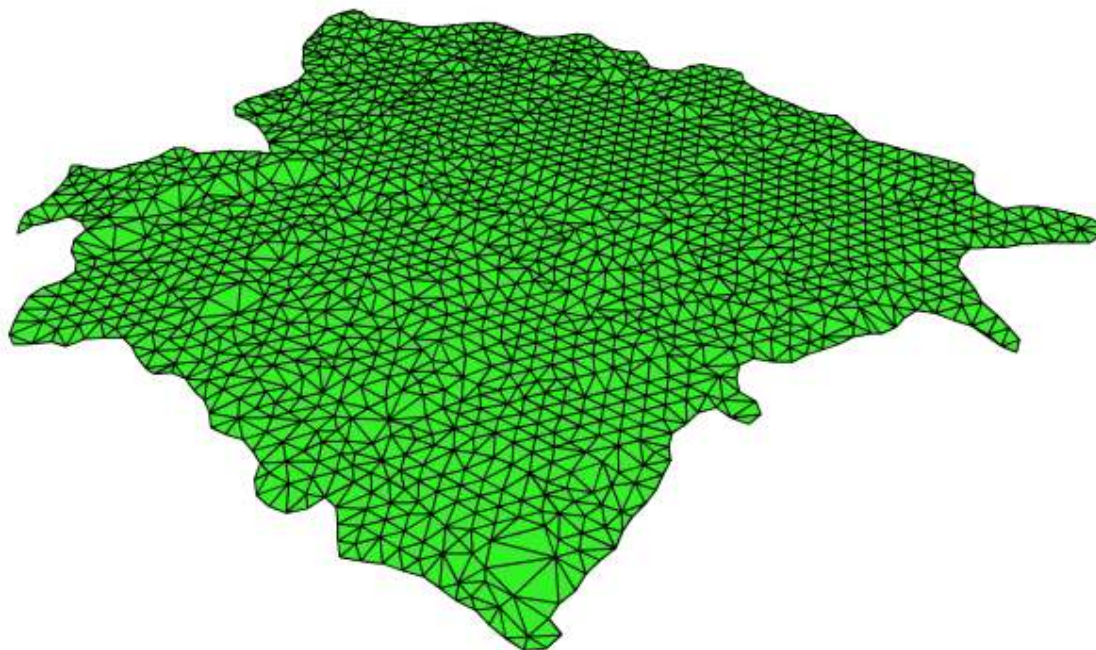
Samotný script po překladu dovoluje načíst data ze dvou souborů, prvním z nich je formát *.xls (Microsoft office 2003), druhým pak *.dbf (databázový formát dBase). Script dovoluje vytvořit a vizualizovat model v prostředí 3dsMax a dále ho upravovat, řezat model nebo vytvářet vrstevnice, popř. ho uzavřít pro obrazovou dokumentaci přímo v prostředí programu 3dsMax. Přeložený script pak dále obsahuje organizátor objektů, který dovoluje spravovat objekty ve scéně (vybírat je, např. světla, geometrie, kamery atd., schovávat ty, se kterými právě nepracujeme) jiným způsobem než organizér programu 3ds Max, kdy je možné pracovat zároveň s tímto organizérem a upravovaným modelem. Další důležitou částí programu je pak export modelu do programu gmsb, pro vytvoření objemových prvků, kdy na základě scriptu můžeme definovat podskupiny polygonů barevným odstínem a na základě takovéto definice se objemové prvky společně s celou geometrií exportují do formátu *.geo právě pro program gmsb, který se používá na katedře NTI pro další zpracovávání takovýchto geometrií.

Seznam použité literatury

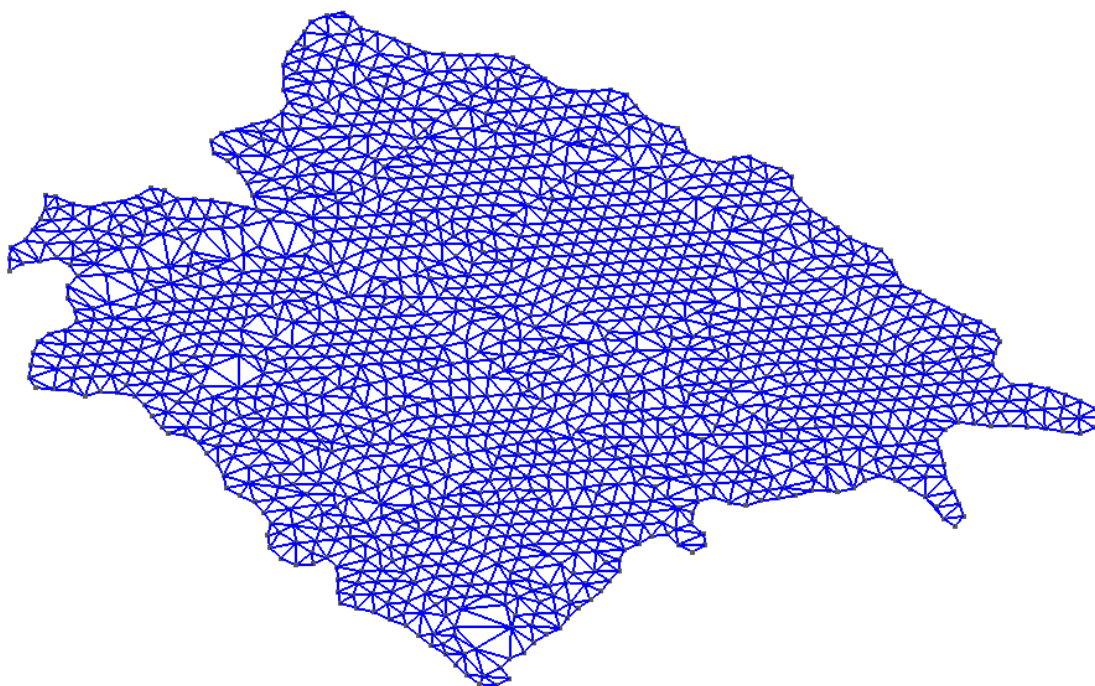
- [1] ESRI [online]. Oficiální stránky firmy ESRI o GIS 21. 10. 2010
<<http://www.esri.com/>>
- [2] Stránky o geomatice na ZČU-Úvod do problematiky.[online] 17.11.2010
<<http://gis.zcu.cz/studium/ugi/>>
- [3] Komponenty, funkčnost, aplikace. Datové modely v GIS .[online] 12.11.2010
<<http://geoinformatika.webnode.cz/news/gis-komponenty-funkcnost-aplikace-datove-modely-v-gis/>>
- [4] Referenční příručka pro ArcGIS 9.2 . 3.10.2010
- [5] Detailní informace o GIS včetně výukových materiálů západočeské univerzity. [online] 4.11.2010
<<http://gis.zcu.cz/studium/ugi/cviceni/index.html>>
- [6] Scripta věnující se GIS problematice. [online] 4.11.2010
<<http://gis.zcu.cz/studium/ugi/e-skripta/ugi.pdf>>
- [7] Mapový portál města plzně. Úvod do problematiky GIS .[online] 12.12.2010
<<http://mapy.plzen.eu/gis/o-gis/uvod-do-gis/>>
- [8] Stránky katedry biologie ekosystémů Jihočeské univerzity- Informace o GIS a problematika dat.[online] 20.12.2010
<<http://kbe.prf.jcu.cz/files/prednasky/>>
- [9] Stránky zabývající GIS ,popisem struktury dat modelů.[online] 20.12.2010
<<http://www.ian-ko.com/>>
- [10] TALHOFER, V. Modelování reálného světa. Brno 2005
- [11] MALÁ B.-PACINA J. Projekt Poohří.Výstavba modelových sítí a automatizace v rámci tvorby modelových sítí. 10.2.2011
- [12] ZÁBRANSKÝ, J. Triangulace povrchů a úlohy na nich 23.11. 2011
- [13] MARYŠKA J. – MALÁ B. Výstavba modelové sítě a její naplnění hodnotami z GIS SURAO a stanovení počátečních podmínek pro různé varianty migrace. 29.10.2008
- [14] MALÁ B- PACINA J. Článek časopisu Studia Oecologia č.3 1.11. 2010
- [15] MALÁ B.- CAPEKOVÁ Z. Geoinformatické modelování a jeho přístupy v tvorbě mesh modelu území. TUL 2008
- [16] Informace o GIS , zejména o programu ArcGIS [online] 23.11. 2011
<<http://www.arcdata.cz/uvod/>>

- [17] České stránky a forum o 3D grafice a v 3DsMax [online] 12.11.2010
<www.maxarea.com>
- [18] Stránky firmy a distributora programu 3dsMax [online] 12.11.2010
<<http://www.cadstudio.cz/3dsmax>>
- [19] Referenční příručka 3DsMax v.2009 a v.2011 15.4.2010
- [20] Referenční příručka maxscriptu programu 3dsMax v. 2009.
- [21] KŘÍŽ J. 3ds Max-Hotová řešení pro v. 2009. Kniha o modelování a programování v maxscriptu pod 3ds Max
- [22] MURDOCK, K. 3ds Max 2011 Bible- Kniha o modelování, animaci a programování v 3ds Max
- [23] Historie a vývoj maxscriptu. [online] 28.12.2010
<<http://wiki.cgsociety.org/index.php/MAXScript>>
- [24] WAINWRIGHT, J. [online] Tutoriály tvorby scriptů v 3dsmax
<<http://vimeo.com/album/1514565>>
- [25] Forum o maxscriptu.[online] 2.2.2011
<<http://forums.cgsociety.org/archive/index.php/f-98.html>>
- [26] Web se seznamem komponent událostí, a metod dotNet v maxscriptu a některé příklady použití. [online] 23.1.2011
<<http://www.scriptspot.com/bobo/mxs9/dotNet/index.html>>
- [27] Tutoriály pro maxscript zaměřen většinou na dotNet implementaci. [online]
<<http://lonerobot.net/?cat=6>>
- [28] Základy maxscriptu formou tutoriálů. [online] 5.1. 2011
<http://www.maxforums.org/threads/dub_maxscript_tutorial_index/0001.aspx>
- [29] Hlavní stránky programu Gmsh včetně referenční příručky.
[online] 10.2.2011 <<http://www.geuz.org/gmsh/>>
- [30] Tutoriály zaměřené na program Gmsh. [online] 10.2.2011
<http://amcg.es.ic.ac.uk/index.php?title=Local:Gmsh_tutorial>

Příloha – A: Převedený základ sítě z *.dbf



Obr. PA1: Data z GIS importovaná aplikací.

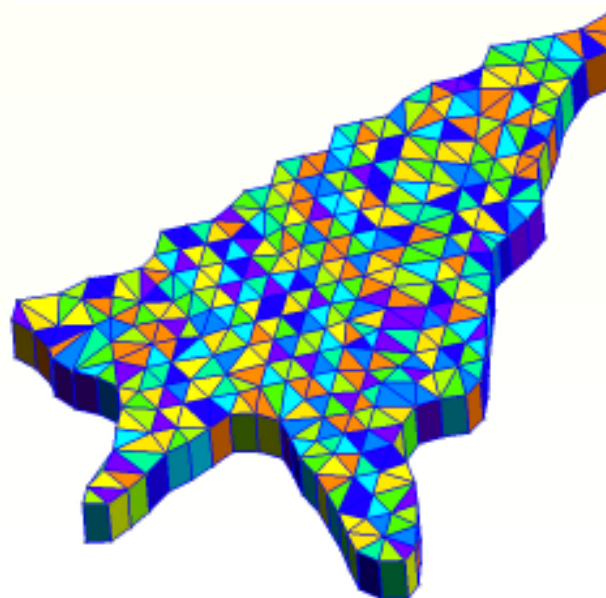
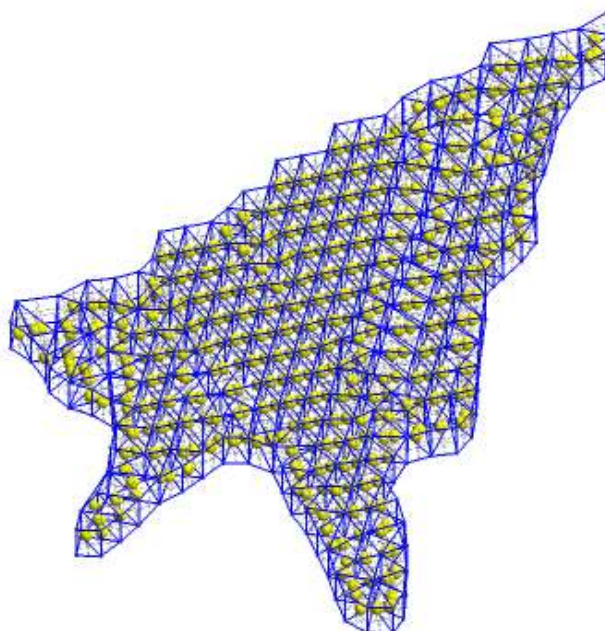


Obr. PA2: Export z 3dsMax do Gmsh.

Příloha – B: Ukázky modelových sítí



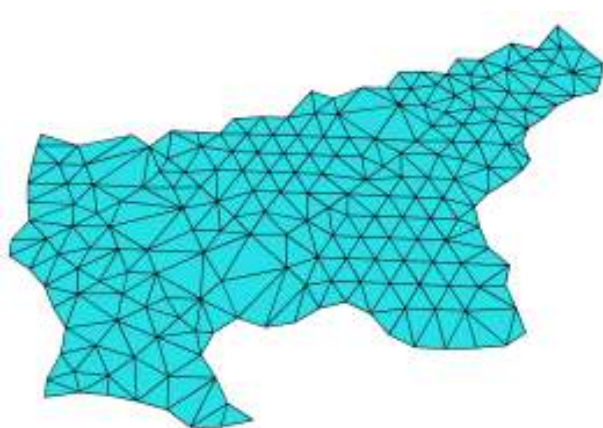
Obr.PB1: Ukázka základu sítě v 3dsMax. Obr.PB2: Ukázka modelové sítě v 3dsMax.



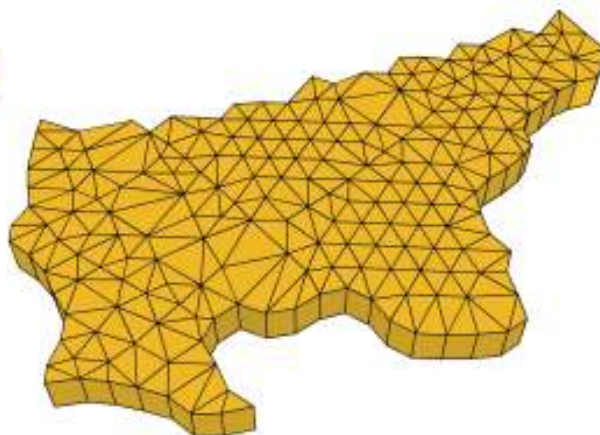
Obr.PB3: Převod mod. sítě v Gmsh.

Obr.PB4: Ukázka mod. sítě v Gmsh

Příloha – C: Ukázky modelových sítí



Obr. PC1: Základ sítě v 3dsMax.



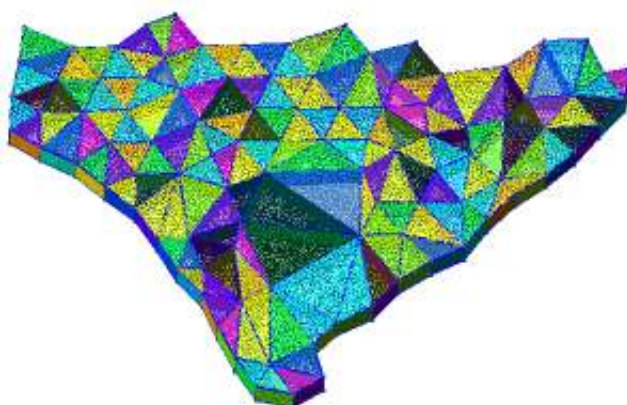
Obr. PC2: Modelová síť v 3dsMax (Neprůhledná).



Obr. PC3: Převedená modelová síť do Gmsh.

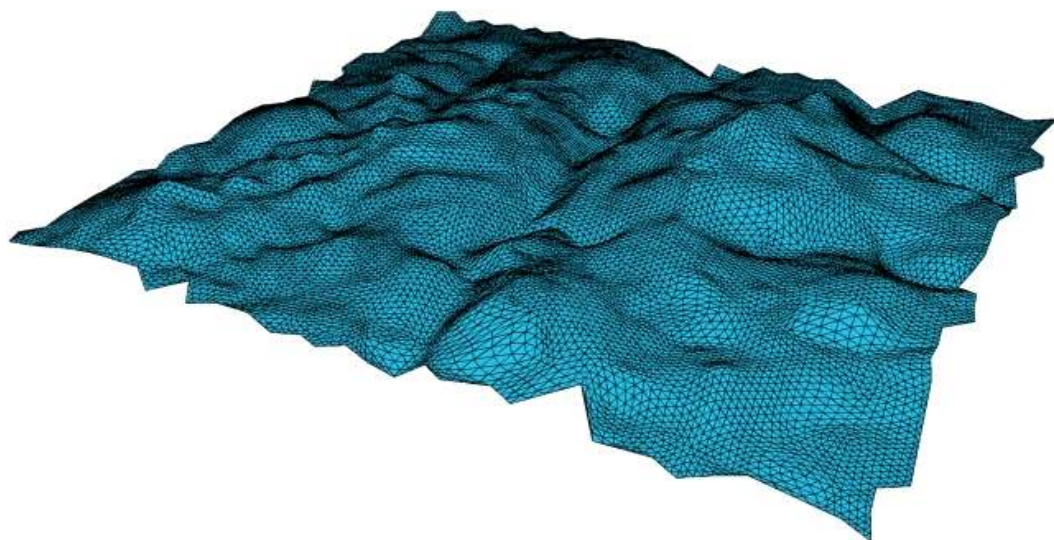


Obr. PC4: Převod mod. Sítě do Gmsh

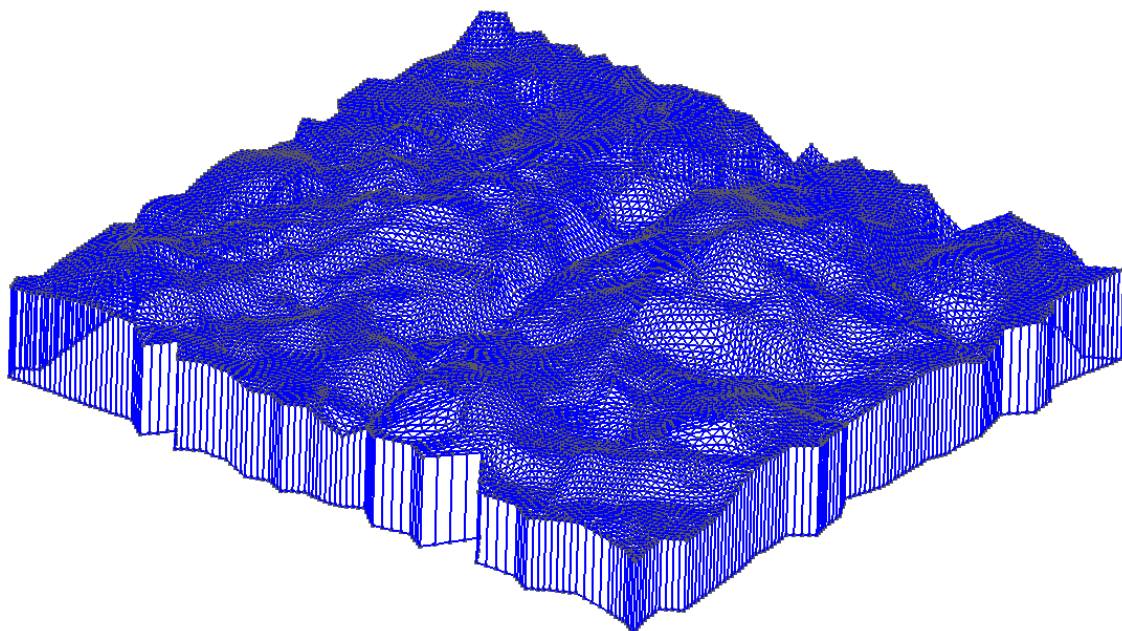


Obr. PC5: Převedená modelová síť do Gmsh.

Příloha – D: Ukázky povrchů



Obr. PD1: Ukázka neuzavřené povrchové sítě v 3dsMax.



Obr. PD2: Převod uzavřené povrchové sítě z 3dsMax do Gmsh