Feature-based multi-resolution topography

Víceměřítková reprezentace topografického povrchu založená na

geoprvcích

Lukáš Brůha

Department of Applied Geoinformatics and Cartography, Faculty of Science

Charles University in Prague

Dissertation thesis

Supervisor of the thesis: Tomáš Bayer

Prague, 2016

Prohlášení

Prohlašuji, že jsem závěrečnou práci zpracoval samostatně a že jsem uvedl všechny použité informační zdroje a literaturu. Tato práce ani její podstatná část nebyla předložena k získání jiného nebo stejného akademického titulu.

V Praze, 10. 7. 2016

Podpis

Abstract

A three-dimensional (3D) representation of the topographic surface is an important element in planning, civil engineering and mapping. Many state-of-the-art representations provided graphical, 3D model of entire planet. However, the existing solutions to 3D topographic surface lack the geometric flexibility and accuracy on boundaries with models of other geographic features. Most of the contemporary approaches to digital Earth solutions focus on the visualization performance. The high visual performance is achieved by the use of special data structures optimized for rendering. However, this optimization towards visualization hampers the data management of spatial data, their analysis and distribution.

Therefore, the introduced solution reflects on multiple requirements of digital Earth systems. In addition to the visualization performance, the requirements regarding data interoperability, data management and distribution, data analysis and the multiple level of detail (LOD) are considered as essential for the design of the new solution.

The topographic surface is central to the proposed method. It provides the defining surface in terms of which other features can be geographically referenced to. Therefore, this work introduces a new, more functional data representation of multi-resolution topographic surface. This representation is globally applicable, allows to populate the terrain surface with new geographic features and supports the multiple LOD of both features and terrain.

The presented solution to representation of spatial features for multiple LOD environment is rooted in the concept of footprint. Exploitation of this concept alleviates from the structural complexity of pure 3D solutions, however, it also supports the extension to true 3D, when needed. This work extends the existing usage of footprints by providing support for multi-resolution representation. Consequently, the footprint can be represented with variable resolution along its course in the reconstructed graphic scene. In a pursuit of interoperability, the method for footprint analysis is developed

to deal with features originating from disparate data sources. Thereafter, the algorithm for simultaneous simplification of a set of footprints is proposed. It is designed to build the multiple LOD database of features in such a manner, that the topological relations between features are preserved in the multiple LOD environment, which is reconstructed on its grounds.

Since the topological relation between the feature and terrain is essential for many geo-spatial analyses, this thesis proposes two methods to carry out the generalization of distinct spatial features with respect to the geometry of terrain. It is also shown, how the important morphological structure of the terrain can be obtained and subsequently preserved on coarser levels of resolution.

The method introduced in this thesis employs the Global Indexing Grid (GIG) as an indexing and paging mechanism. GIG determines for any position of the observer within the 3D virtual environment the position-dependent LOD of currently visible features and underlying terrain.

Abstrakt

Datové reprezentace topografického povrchu ve třídimenzionálním (3D) prostoru jsou důležitým nástrojem v plánování, stavebním inženýrství či mapování. Množství nejnovějších reprezentací povrchu poskytlo grafický 3D model celé planety. Existující 3D řešení nicméně postrádají flexibilitu a přesnost na hranicích s modely jiných geografických objektů. Většina současných přístupů k digitálním modelům Země se zaměřuje na rychlost vizualizace. Vysoký vizuální výkon je dosahován užitím speciálních, za tímto účelem navržených datových struktur, které jsou optimalizovány na vykreslení grafické scény. Takováto optimalizace jedním směrem ovšem brání efektivní správě prostorových dat, jejich analýze a distribuci.

Proto předkládané řešení zohledňuje četné požadavky takovýchto geo-informačních systémů s globálním prostorovým pokrytím. Kromě nutnosti vysokého výkonu vizualizace dat jsou zohledněny požadavky na interoperabilitu dat, řízení správy dat a jejich distribuce, analýzu dat a podporu více úrovní rozlišení.

Topografický povrch má v navrhované metodě klíčovou, sjednocující roli. Všechny ostatní prostorové objekty jsou k němu georeferencovány. Z tohoto důvodu dizertační práce představuje novou datovou reprezentaci topografického povrchu s podporou víceúrovňového rozlišení. Tato reprezentace je aplikovatelná globálně okolo sféry, umožnuje integrovat model terénu a prostorové objekty k němu vztažené a podporuje víceúrovňové rozlišení jak terénu, tak i těchto objektů.

Řešení víceúrovňové reprezentace objektů vychází z konceptu "otisku" geometrie modelu objektu na model terénu, tedy geometrie obrysu jejich prostorového průniku. Využití tohoto přístupu umožnuje vyhnout se strukturální složitosti ryzích 3D řešení, avšak současně umožňuje integraci s modely vytvořenými plně ve 3D (objemovými modely) tehdy, kdy se takovému řešení nelze vyhnout. Tato práce rozšiřuje existující přístupy, využívající tento koncept, poskytnutím podpory pro víceměřítkovou reprezentaci otisku. V důsledku tak umožňuje reprezentovat jeho geometrii v rekonstruované grafické scéně s různou úrovní detailu podél jeho průběhu.

Součástí řešení je i metoda analýzy geometrie a vybraných atributů obrysů objektů tak, aby byla umožněna další manipulace s objekty pocházejícími z různých zdrojů, vytvořených různými autory.

Následující krok navržené metody představuje algoritmus pro zjednodušování geometrie vstupní množiny obrysů. Algoritmus vytváří databázi geometrií s více úrovněmi rozlišení. Charakter výsledné databáze je takový, že grafická scéna rekonstruovaná na základě této databáze zachovává topologické vztahy mezi objekty. To je zajištěno na všech úrovních rozlišení, které jsou v takové scéně znázorněny, a to pro libovolnou pozici pozorovatele, pro kterou je databáze dotazována a vůči které je scéna vykreslena.

Topologické vztahy mezi geometriemi objektů a terénu jsou klíčové pro řadu prostorových analýz, ale mnohdy i pro kvalitu výsledné vizualizace. Tato práce proto navrhuje dvě metody, které umožňují zohlednit při procesu generalizace objektů charakter okolního terénu. Nadto je popsán postup extrakce důležitých prvků terénu, které popisují jeho morfologii, a funkčnost vybrané metody extrakce je rozšířena pro účely této práce. Na příkladu takto získané strukturní sítě je ukázán postup, umožňující zachování důležitých prvků terénu na nižších úrovních rozlišení, tedy i v částech scény se zjednodušenou geometrií a na hranicích mezi úrovněmi rozlišení.

Dizertační práce pro účely indexování prostoru a tvorby prostorové databáze adaptuje mechanismus pro globální indexování (GIG). Na základě této metody indexace je možné pro libovolnou pozici pozorovatele ve 3D virtuální scéně definovat úroveň rozlišení aktuálně viditelných objektů a terénu.

Acknowledgements

Every thesis has its own history and suddenly, one story has reached its end. This story started several years ago, when Jan Kolář introduced me into the topical research of digital Earth technologies. Great thanks belongs to him for the enthusiasm, open communication and down-to-earth attitude.

I have to stress my gratitude to my supervisor Tomáš Bayer, who always was an inexhaustible source of ideas from the field of digital cartography and computational geometry. Moreover, our discussions were insightful, and he continuously encouraged me to take a more formal and scientific stance.

My thanks are addressed to all members of the department, who contributed in one way or the other, and created pleasant environment in which I have worked for many years. I would also like to acknowledge the thousands of individuals who have coded all the open-source libraries and softwares. It is also due to their efforts and their attitude that we can test our ideas and contribute to the technological foundations.

On a personal level, I would like to thank my family for the love and support over the years and my friends for providing distractions outside university. Without all the support mentioned, this project could not have been successful.

Contents

List of Figures

List of Tables

1

Introduction

The early mapping methods fitted the nature of human perception of the ambient world surprisingly well. Even as early as 1618, Claes J. Visscher (1586 - 1652) first printed the picturesque map of the city of Paris, whose fragment can be seen in Figure 1.1. All buildings, trees or fortification are displayed as volume objects related to the ground with an elevated perspective view. Also the detail in the map gradually decreases with the growing distance from the observer, as it is natural for human's view.

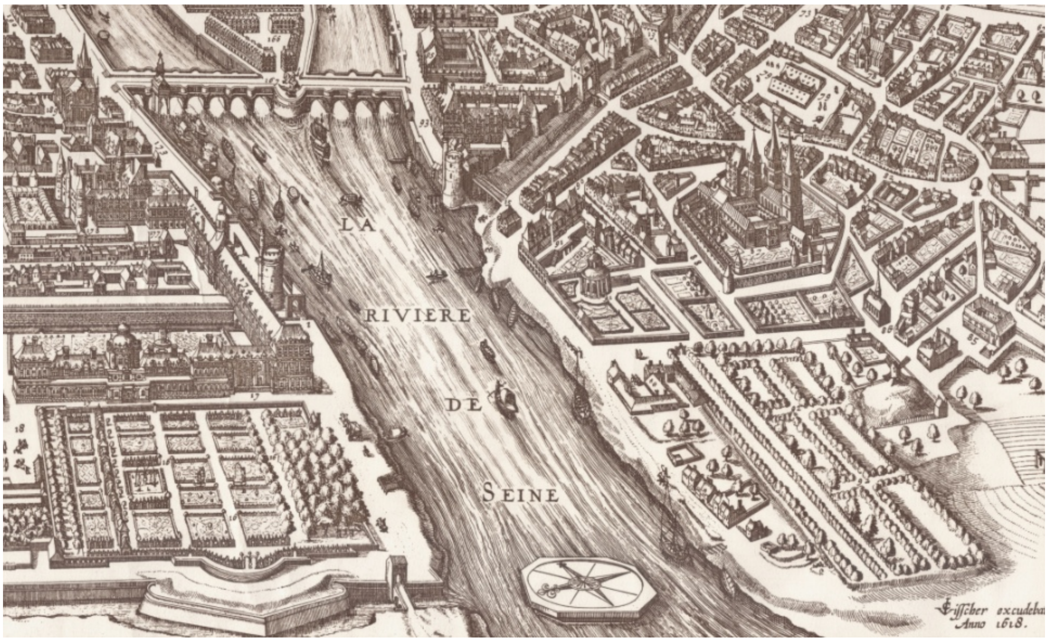

Figure 1.1: Map of the city of Paris - Segment of the map of Paris in an elevated view from 1618, scale ca. 1:2.000

In the following periods of time, the maps, that depicted the space in two-dimensions (2D), were significantly preferred. The return to the third dimension was supported by the onset of digital technologies, which also marked the domain of the visualization of space. With the introduction of geographic information systems (GIS), the maps ceased to be the means of space depiction only, but became a tool for analysis and planning. This thesis puts forward the effort for the use of spatial information in a uniform and global context, which is intended as the next step towards the natural human perception of reality within three-dimensional (3D) GIS.

The efforts for the use of spatial information in such a context are not entirely new. Research activities focusing on the development of techniques for a digital representation of the planet Earth used to be gathered under the domain of *digital Earth* since 1998, when the the term was first mentioned by Al Gore in (44). Contributions from several well-established fields of science meet in the back-end of digital Earth solutions, for example geodesy, cartography or computer graphics. Neither the techniques for data acquisition, management and distribution can be overlooked.

In this chapter, on the basis of the motivation, discussed in section 1.1, the objective of research (section 1.2), the scope of the study (section 1.3) and the structure of the thesis (section 1.4) will be discussed.

1.1 Motivation

Advances in data acquisition techniques have made huge geometric datasets ubiquitous in terrain modeling (Light Detection And Ranging (LiDAR), synthetic aperture radar), computer-aided engineering (laser short-range scanning) or vector topographic data (OpenStreetMap project). These datasets provide nearly a global coverage for applications within the field of geographic information science (GISc). There is an enormous number of spatial data sources, which differ in dimensionality, spatial coverage, resolution, accuracy and many other factors, including the data format, they stored in.

Nevertheless, all the various spatial data sources have one thing in common. In the 3D space, the topographic surface possesses a unique role. It has a similar importance as the mapping plane has in 2D cartography. It provides the global defining surface in terms of which other features can be geographically referenced (georeferenced).

Therefore, the topographic surface is central to this study. According to various dictionaries, the meaning of *topography* slightly differs. The Oxford online dictionary (25) states the following definition: *"The arrangement of the natural and artificial physical features of an area"*. In this thesis, under the term *topographic surface*, it is understood the result of association of the distinct features and terrain [1] .

In this context, the connection of words "feature-based topography" in the title of the thesis may be seen as redundant. It was used intentionally to emphasize, what current solutions fail in. Namely, the representation of features.

Many representations of topographic surface, for example in (2, 17, 19, 31, 90), provided graphical, multi-resolution model of entire planet. These works allowed for a new and uniform way to visually explore data referenced to arbitrary location on Earth at any scale and perspective. Applications of these solutions for scientific communication, education, news presentation, proved to be useful in practice through commercial products such as GeoFusion or Keyhole. Nevertheless, the state-of-the-art solutions focus on the visualization aspect neglecting the strong property of traditional GIS, the analysis.

As a consequence, the existing solutions lack geometric flexibility on boundaries with models of any other geographic features, which are connected to the terrains surface. The precise representation of such boundaries on the surface is critical for many analytical applications and engineering domains, for example in civil engineering during planning, construction and maintenance of municipal infrastructure; for delimiting of geological boundaries; for management of legal boundaries in 3D cadaster; in general for any analytical processing traditionally provided by 2D-based geographic information systems (GIS).

Even regarding storage, maintenance and distribution, existing approaches to global solutions imitate the concept of scale series of traditional maps (e. g. 1:10 000, 1:25 000, 1:50 000, etc.) to achieve multiple levels of detail (LOD) of spatial objects. For each level, distinct representation is needed, stored and maintained. In the traditional 2D domain, especially for purposes of national mapping agencies, solutions to this

[1]Frequently, the term *topography* used to be swapped with the term *terrain*. The Oxford online dictionary defines *terrain* as *"A stretch of land, especially with regard to its physical features"*. In this work, we use the term *terrain* to refer to the representation of bare Earth, which, contrary to the term *topographical surface*, does not consider distinct features.

inconsistency appeared in a form of multi-representation databases (53) or variable-scale geoinformation (97) kept in the tGAP structure (138). It is however achieved at the cost of employment complex data structures. Problem with these structures is that they are difficult to be maintained. Furthermore, the topological relations between geometrical objects, they store, were proven as an important restriction for multi-resolution visualization (38).

The storage of topological information further increases the vast data volumes to be handled within the digital Earth system, which easily exceed the usual capacity of data memory available at one site. Together with the fact, that spatial information are more and more used in a network centric environment, the need for distributed solution and techniques, that support it, is apparent. And, the topology is particularly difficult to deal with in a distributed manner.

The data management issues also manifest themselves in the need for an indexing mechanism, that addresses the topographic surface in all directions in a uniform matter. Therefore, the traditional flat approximation of Earth's shape must be avoided to eliminate geometric distortions and singular points. A step in the right direction is the Global Indexing Grid (GIG), first introduced in (76), which provides direct subdivision of the 3D space. GIG also supports multiple LOD. This concept was further elaborated in (77), where the multi-resolution representation of the bare Earth was introduced on the basis of GIG. To set the scene of this thesis, the work (77) of Jan Kolář is its starting point.

1.2 Objectives and main research questions

Currently, there is an open research question on how to address data management of various kinds of spatial features along with the underlying terrain for the sake of visualization and analysis. Therefore, the overall aim is to provide such theoretical framework for features on multiple LODs, that would enhance the bare Earth representation introduced in (77). Considering the drawbacks of current solutions and the observation on the key role of the topographic surface, the main question to be answered in this thesis is:

How can we realize the global feature-based topographic surface
with support of the position-dependent level-of-detail?

There are several partial questions, whose resolution leads to the overall aim. To get a knowledge about the latest scientific approaches of related scientific domains, we raise a question:

1. What is the state-of-the-art in large spatial data management and generalization?

To build a theoretical background of the proposed method, the second and third question needs to be answered:

2. What indexing and paging data structure should be adopted, so as to simultaneously support the multiple LOD and fulfill the preference for non-projected solution?

3. How can be formally described what is multiple LOD environment?

In order to introduce such a methodological framework, which would grant maximum flexibility, functional extensibility and general applicability, we bring into question:

4. How to design a conceptual data model that describes the feature-based topography in the multiple LOD environment?

To simplify the preconditions and to facilitate addition of independent features, the validation of data input comes into question:

5. How can be the valid input data created by as much automatic manner as possible?

Afterwards, a method for features' geometry decimation becomes an issue for further consideration:

6. How can be the database of features created so that the topological relations between features will be preserved in the multiple LOD environment reconstructed on its grounds?

Having the database of features and underlying bare Earth model:

7. How can be the topographic surface with features synthesized for given observer position?

The sixth question, that covers the topological relations between features, can be extended to cover relations between the geometry of terrain and the geometry of features:

8. How to guide the simplification of features with regard to terrain?

With respect to the data representation of bare Earth to be adopted in this work, for a plenty of GIS applications, the ability to decrease the rate of simplification or even maintain the full resolution of distinct terrain structure, is important:

9. How to preserve selected morphological characteristics of terrain on coarser levels of detail?

1.3 Research scope and limitations

Although a relatively high number of scientific fields blend in this study, the key problems solved in this thesis fall within GISc and the research is performed from the geoinformation technology perspective. In order to provide clear distinction of what are the limits of this work, the following topics are explicitly included.

1. Formal description of multiple LOD global indexing and paging mechanism.
2. Topological consistency between geometries of features (mutually) and terrain in the multiple LOD environment.
3. Suitability of the concept for spatial analysis and distributed data management.
4. Introduction of the theoretical and technological framework, which is based on 2D geometry in 3D space, provides a non-projected solution and avoids data storage redundancy.
5. Illustration of the functionality through technological implementation.

The following topics are explicitly excluded:

1. Gridded (raster) data and continuous (fields) representations.
2. Although the overall concept is designed to comply with requirements on dynamic and temporal features, or be easily extensible for these purposes, this aspect is not explicitly covered by this thesis.
3. The framework for representation of features is designed to allow true 3D (volumetric) features to be included, but the thesis does not provide methods for multiple LOD of such 3D objects.

4. Implementation of the production-quality software or the client for 3D navigation and rendering of results are beyond the scope of this work.

1.4 Structure of the thesis

The rest of the thesis is organized as follows:

Chapter 2 focuses on literature review. The chapter starts with discussion on different perspectives to spatial data modeling, which is followed up with the review of works related to modeling of dimensionality. An overview of existing data representations for multi-resolution topographic surface is provided. Furthermore, the focus is laid on approaches suitable for handling distinct features of the terrain. Great number of methods were developed for purposes of cartographic generalization. Those, which can provide an inspiration to handling the geometric detail of spatial objects on top of the bare Earth representation, are examined. The literature review concludes with methods for big data management and access.

Chapter 3 states the requirements on the solution. The arguments for three fundamental choices for the character of the solution are built, namely the preference for minimal data redundancy, the preference for procedural solution and the preference for non-projected solution. Based on the preferences, the indexing mechanism with support for LOD and the bare Earth representation, which are adopted by the method of this work, are formalized.

Chapter 4 introduces the concept of footprint, in order to provide means for population of topographic surface with any geographic features, that can influence its shape. Method, for analysis of footprints to ensure initial data validation and adjustment, is proposed. The simultaneous simplification algorithm is presented as well as the method for the reconstruction of the topographic surface. The surface reconstructed at runtime contains topologically consistent approximations of features for arbitrary position of an observer, with the intensity of geometry simplification growing with the distance from the observer. The proposed solution does not introduce any data redundancy.

Chapter 5 explores the relation between geometry of spatial objects and the geometry of bare Earth in the multiple LOD environment. The solution from previous chapter is extended in two ways. The enhancement of the simplification algorithm, which allows to restrain the simplification of features with respect to the geometry of terrain, is introduced. Furthermore, the method, which allows for preservation of terrain morphological structure on coarser LODs is described. The consequences of the procedural approach on visualization and spatial analysis are discussed.

Chapter 6 demonstrates the implementation and experiments carried out in order to prove the concept and verify the results achieved in this work. More specifically, the implementations of footprints' analysis, multiple LOD database creation and run-time reconstruction of the topographic surface are described and evaluated. The second part of the experiment focuses on the implementation and evaluation of terrain morphological structure retrieval and its application.

Chapter 7 summarizes the main achievements and shows the possibilities for future research.

2

Research background

This chapter reviews related work to set the scene of the thesis. Some of the studied works are mentioned in order to put this work in context of latest developments in the particular domain of research. Such knowledge is then instrumental in the decision on, which works should be adopted, or even extended in the thesis.

At first, the chapter introduces different perspectives on real world modeling in section 2.1, afterwards, in section 2.2, the dimensionality of data comes under scrutiny.

The multi-resolution digital terrain models are essential for proposed concept. Therefore, initially the section 2.3 pays attention to works dealing with multiple levels of detail of large terrains. Secondly, it is followed by an overview of methods for cartographic generalization of distinct spatial objects in section 2.4. And thirdly, in section 2.5, the works dealing with interaction of terrain models and distinct spatial objects are examined.

Because large data with even a global spatial coverage are one of the leading themes of the thesis, the literature review is concluded by an overview of methods for the management and access to such big data in section 2.6.

The terminology used in this chapter relates to the described study, where further detail can be found.

2.1 Modeling geographical space

The significance of the data structures for GIS can be suitably conceptualized considering the following hierarchy of definitions adopted from Donna J. Peuquet's (113):

"**Reality**: The phenomenon as it actually exists including all aspects which may or may not be perceived by individuals.

Data Model: An abstraction of the real world that incorporates only those properties thought to be relevant to the application or applications at hand, usually a human conception of reality.

Data Structure: A representation of the data model often expressed in terms of diagrams, lists, and arrays designed to reflect the recording of the data in computer code. The data structure is built upon the data model, and details the arrangement of the data elements. This structural arrangement is the heart of the GIS storage and retrieval subsystem.

File Structure: The physical representation of the data in storage hardware. This representation is usually predetermined by the makers of the GIS software and is not usually under the control of the user."

During the last five decades a rapid increase of existing GIS data models, data structures, and discussions on geographic representation and ontology could have been seen. Vector and raster representations provided early description of the real world, as did Sinton's three-dimensional schema (127). Topological data structures were studied in the 1970s. The implementation of topological structures in commercial GIS (148) was supported by the introduction of the relation model a decade later. Extension of relational model by means of object-oriented model (29, 148) occurred in the 1990s.

As far as the data model is concerned, the field - object distinction observed by (18) is essential to human perception of the world. People understand the surrounding real world as populated by discrete objects with distinct properties and functionality. However, other characteristics of the environment, especially various physical quantities including noise levels, atmospheric temperatures or precipitation, can rather be described as continuously varying fields.

(97) distinguishes three main approaches to object-based data modeling: object-first approach, space-first approach and a hybrid approach.

The object-first approach is rooted in an a priori definition of object classes to be modeled. Such classes of objects have its own set of thematic attributes and added geometry attribute. The geometric description of an object is independent from all the others. Additional rules can be associated with objects to avoid topological conflicts.

The major drawback of the object-first approach is the absence of explicitly described topological relations within the model.

On the contrary, the space first approach explicitly models the topological and geometrical relations between objects during the subdivision of the space domain. The geometries of objects are assigned attributes, which allows to classify them. However, the set of attributes is the same for all objects.

The hybrid approach utilizes the benefits from both previous approaches. It allows to share the geometry between objects, but also to design a unique thematic attribute set for object classes. (101) described this model as Formal Data Structure (FDS) theory, which is now implemented in Oracle Spatial Topology (82).

The object-based perception of the geographic domain is clearly the choice, when the description of some instances like human-made structures on top of the Earth surface is concerned. But the actual bare Earth is classically seen as an elevation field. Also its physical properties or many physical laws to be described in the geographical domain concern continuous fields and are specified as partial differential equations (42).

Obviously, the integration of both, objects and fields, within a single framework, would provide numerous advantages. Some authors have commented on this possibility (14, 35, 74, 148). The operations, that create fields from objects and vice versa, were thoroughly reviewed in (14); they cover (the list is adopted from (42)):

- density estimation, which creates a continuous field of density from a collection of discrete objects;
- object extraction algorithms in image processing and pattern recognition that extract discrete objects from a field of reflectance or radiation; and
- algorithms for identification of surface-specific points and lines (peaks, pits, passes, ridges, etc.).

The notable contribution to conceptualization of continuous-fields and discrete-objects integration was introduced through the concept of geo-atoms (42). Lately, a novel approach based on Geospatial Managed Objects (GMO) supporting both field- and object-based geospatial models in a uniform way was presented by (79). It is grounded in the byte-code unification of the GMOs. This engineering solution is inherently associated with a particular virtual machine. Moreover, through the concept of *managed code* the very low-level interoperability is achieved (73, 78).

2.2 Dimensions of spatial data

Another view on the geographical data modeling is from the perspective of data dimensionality. Geographical phenomena traditionally was and still most frequently is (e.g. cities, roads and houses) modeled as embedded in the 2D Euclidean space using three geometric primitives: (i) points (0-dimensional objects); (ii) line segments (1-dimensional objects); and (iii) polygons (2-dimensional sequence of line segments). This prevalence is retained due to the highly cost-effective systems for data acquisition like remote sensing, but slowly eroded thanks to advances in technologies like LiDAR or indoor positioning. Moreover, the third topographic dimension is often handled only by imposing the assumption that elevation is a function of location.

Therefore, the definition of dimension within geographic information science is not straightforward. (114) or (109) distinguish two types of dimensions of a spatial model, the internal and external dimension. The internal dimension describes the highest dimension of primitives that describe the modeled object. The external is the dimension of space, in which the object is modeled. The term *codimension* applies to subspaces in vector spaces. Meaning, that if W is a linear subspace of a finite-dimensional vector space V, then the codimension of W in V is the difference between the dimensions: $codim(W) = dim(V) - dim(W)$. Accordingly, (114) defines the dimension of a model as:

- 2D model: modeling with 2D primitives in 2D space (dim 2, codim 0)
- 2.5D model: modeling with 2D primitives in 3D space (dim 3, codim 1)
- 3D model: modeling with 3D primitives in 3D space (dim 3, codim 0).

However, due to the inability of 2.5D models to represent features like vertical walls or overhanded geological structures and, in addition to the fact, that it is much less trivial to reconstruct objects in 3D in comparison to 2.5D, other model dimensions were established by several authors.

(126) applied so-called multiple 2.5D model based on the usage of several 2.5D layers to simulate the 3D topographic data. The ISO 19107:2003(E) (2003) standard defines GM_TIN data type, which allows vertical faces in triangulated irregular network (TIN) to be additionally computed and added to the original triangulation. This approach is usually called 2.5D+ model. (87, 137) presented approach still based on primitives of dimension two or lower, but with no limits on vertical walls or overhangs, designated as 2.75D model.

Increased demand for 3D GIS applications stressed the structural complexity of pure 3D solutions. In order to avoid it several solutions utilize the simpler 2D topology as far as possible. Several authors followed the path of a hybrid models to keep the simplicity of 2.5D for large terrains and to model the complex features in true 3D only where needed. In (87), the triangulation was used as a foundation for extrusion of buildings' models (full polyhedra or 2.75D surfaces) on the basis of their footprints on the 2.5D TIN terrain model. The Tetrahedronized Irregular Networks (TEN) also became an object of geo-informatic research and a mean of third dimension representation, when (108) proposed integrated TIN/TEN approach. It was further elaborated in (109). The TEN concept was also addressed in (110) using the mathematical field of simplicial homology, which provides a better control over orientation and boundaries derivation of 3D features. Complete and correct axioms for '2.8-D maps', the extension of 2.5D maps of walls and overhangs, maintaining the simplicity of 2D topology were presented in (47). Cases of tunnels or bridges that exceed the limits of 2D topology are addressed in (49) by the concept of handles. 2D footprints are also considered to ensure the topological consistency of 3D city models obtained by extrusion in (87).

2.3 Data representations for multi-resolution topographic surface

Methods for digital representation of the topographic surface are an open research problem especially regarding global solutions. This subject is closely related with the capability to represent the surface at multiple levels of detail (LOD), thus with algorithms producing coarser resolution from the original geometry. The decisions on topological consistency of the coarser versions of the surface, and the consistency on boundaries between different LODs in the case of multiple LODs within one scene, are the key issues. Individual works typically focus either on the performance of visualization or stress the analytical processing.

The aim of the digital terrain model (DTM) simplification is the reduction of vertices (alternatively edges or faces), that constitute its geometry. The nature of surface mesh or TIN is well suited for the representation and simplification of DTM, because of its adaptive resolution. The TIN to be simplified will have exactly such a number of vertices that is required to model the surface within a given tolerance. This process is called mesh-simplification. In (55), Heckbert and Garland presented a complex survey on existing simplification methods of polygonal surfaces. Moreover, in (37) they came up with greedy insertion algorithm based on simple local metric to determine the vertices' importance.

Numerous representations of topographic surface providing graphical, 3D, multi-resolution model of entire planet have been developed and some of them mentioned in introduction (2, 17, 19, 31, 90). These works allowed to visually explore data related to any location on Earth. Furthermore, the VAriable Resolution Interactive ANalysis of Terrain system was proposed by (22) as a solution to LOD terrain suitable for analysis. The system is based on progressive refinement or simplification of a general TIN by addition or removal of vertices of the mesh and subsequent re-triangulation. The visual performance was essential criterium for the planetary application Batched Dynamic Adaptive Meshes (17), which employed bintree cells filled with TIN to support LOD. (36, 89, 116) provided partial solutions to the issue of stitching different LODs of terrain by introduction of continuous LOD rendering of height fields.

(77) introduced procedural representation of terrain suitable for analysis and visualization of spatial data with global extent. Out-of-core paging of big data, which is common in computer graphics, requires complex data structures efficient for rendering. The drawbacks associated with maintenance of such structures are effectively avoided thanks to the procedural character of the solution.

Existing solutions to global terrain models mostly refer to bare Earth without representation of features of the terrain or objects on top of it. Topological consistency between objects on coarser resolutions is usually solved separately within the fields of the cartographic generalization or, more recently, 3D city modeling. Works, that bring aspects of geospatial objects generalization on top of the terrain and the interaction of the outcome with the terrain, are rare. Therefore, as the next step (in section 2.4), the cartographic approaches to generalization of features will be under scrutiny, followed by the review (in section 2.5) of methods for combined terrain with features.

2.4 Cartographic generalization

As opposed to the mesh-simplification introduced in the previous section, feature- or shape-simplification, which applies various generalization operators to distinct features or groups of features. Works on the data management of geospatial objects at multiple resolutions, or scales, were traditionally associated with 2D cartography and cartographic generalization. In the context of digital era, the terms digital landscape model (DLM) and digital cartographic model (DCM) help clarify the distinction between data that is modeled for GIS and data for mapping (93). DLM is a product of object generalization, i.e. it is a specification, which defines everything relevant from the real world to be modeled. The processing of data produced for the sake of DLM, which frequently leads to coarser LODs so as to allow for more efficient computation, is termed model generalization. When such a data model is to be visualized, the DCM provides the geometric representation of objects suitable for visual presentation. This process is termed cartographic generalization within a field of cartography.

Algorithms of cartographic generalization were deeply studied and many of which later served as a background for further extensions within (3D) GIS. On the following lines, without a pretension of being complete, only several prime works are mentioned. McMaster and Shea in (94) introduced a classification of generalization operators, an abstract description of atomic generalization functionality. Atomic in such a meaning, it only affects predefined and isolated aspects of a feature in an indivisible way. Their categorization was based on spatial and attribute transformations.

Comprehensive description of generalization operators was also given by (34). While some operators adjust a geometry of an object to produce a more convenient representation (simplification, enhancement, displacement), others are far more complex applying higher-level meaning of objects' character (reclassification, typification, amalgamation).

Since 1960', the point reduction algorithms were developed to cut down the hardware demands during the vector data computing. Among the first algorithms in digital cartography, several approaches to line generalization were designed, e.g. algorithms proposed in (26, 67, 106, 118). Their objective is the best possible approximation of the line's shape by removal of less important vertices. Usually, the so called critical vertices are preserved based on geometrical criteria (global maxima, maximum curvature or points of inflection).

Currently, the Douglas-Peucker algorithm (26) is probably the most used poly-line simplification method. It was devised as a basic poly-line simplification algorithm applicable for a single, isolated poly-line. The solution uses the recursive subdivision of a line. Having the endpoints of the line marked as to-be-kept and given all the points between the first and the last point, the procedure finds the furthest point from the line segment, which is determined by the endpoints. If such furthest point p is closer than given threshold value, all intervening points are marked as to be removed. Otherwise the p is kept, the line is split into two parts, for which p becomes a new endpoint, and the algorithm recursively calls itself upon these parts.

The algorithm, however, can easily cause topological conflicts when applied to a set of neighboring poly-lines. This type of poly-line modification in isolation is termed *in vacuo* by (121) or (97). This drawback led to the development of methods that modify single poly-line in context (*en suite*) of other poly-lines, e.g. (21). However, the outcome of the algorithm of de Berg is dependent on the order of processing. Therefore, the *en masse* solution was sought to overcome this imperfection. It was found by (85, 97), who achieved it by considering the complete set of poly-lines and all mutual topological relationships.

2.5 Multi-resolution terrain with features

Although many concepts from traditional cartography can provide an inspiration for digital GIS, the shift from the second to the third dimension unveiled some new challenges to deal with. These are especially related to the determination and representation of detailed features of the terrain or population of the terrain with objects on top of it.

The TIN representation of terrain is a classical type of 2D GIS layer desirable already by early GIS applications from 1970'.

Throughout the history, many procedures for computation of Delaunay triangulations (DT) and constrained Delaunay triangulations (CDT) were developed to build TIN with distinct properties (9, 16, 23, 24, 63, 117, 124, 145). DT are known as a dual of Voronoi diagrams and they fulfill the empty circle criterium stating that the circumcircle of any face in the triangulation encloses no vertex. Constrained triangulations allow for enforcement of some edges in the triangulation, where the mandatory edges are defined by the input.

Constrained Delaunay triangulations (CDT) satisfy the constrained empty circle property meaning that the circumscribing circle encloses no vertex "visible" from the interior of a triangle, where mandatory edges represent visibility obstacles. CDT were proven to be the main tool used in meshing problems.

Many effective algorithms have been developed within the realm of computer graphics even for the simplification of 3D surfaces, and some of them were mentioned above in section 2.3.

Nevertheless, they often result in a poor approximation, when a model is simplified to a very low level of detail. Consequently, it may be desired to preserve some selected features of the surface. To overcome these limitations, (70) presented a user guided simplification to maintain features of the surface, which are geometrically small but semantically important. The method is built upon the quadric based simplification algorithm, an extension of the previous work of (38). There are two parts within their method. First, the geometric constraints of three types: point constraint, contour constraint, and plane constraint. Their purpose is to keep the position vertices and edges unchanged or to maintain the overall shape of the region. Second, the adaptive simplification, which basically allows the user to determine the places to be simplified less than others.

From the perspective of GIS, these points, lines or areas are typically some unique features of the terrain, e.g. the valley lines, the ridge lines, or general important terrain break-lines. Furthermore, the (man-made) spatial object on top of the terrain affects the shape of the terrain surface as well.

The morphology of terrain. The discovery of terrain features was widely studied within GISc. Two main groups of approaches to break-lines modeling can be identified.

The first one is based on the analysis of raster DTM, the second one uses the irregular point clouds typically acquired by LiDAR technology. The raster-based analysis are computationally less expensive. A forfeit for this advantage is the decreased accuracy of detected lines as a consequence of smoothing and suppressing the details of a terrain shape during rasterization. Important portion of these methods aims at computing the drainage information that is needed to define the ridge network (99, 104).

Another approaches usually adopt algorithms utilized for edge detection in digital image processing (40). The implementation applies discrete convolution together with the use of some type of kernel mask (Sobel, Prewitt, Laplacian operator).

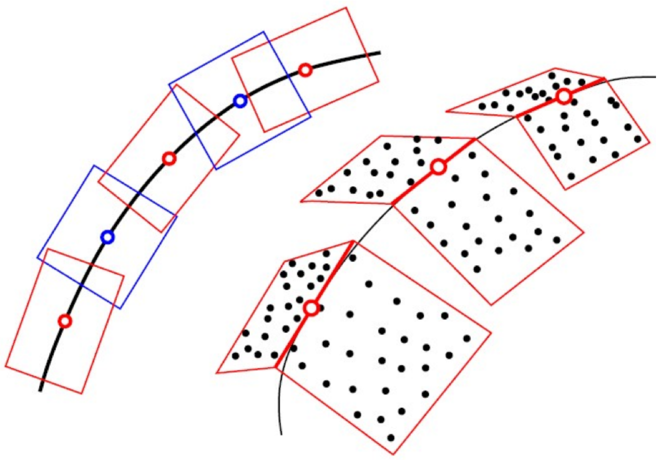

Figure 2.1: The depiction of the intersecting tangent planes principle - Figure adopted from (10)

An alternative algorithms to raster DTM processing focus on point cloud analysis. These methods are dominated by one fundamental principle, which is depicted in Figure 2.1. Its idea is based on the approximation of tangent planes of the surface that are contiguous to the searched terrain break-line. Most works that employ similar approach, first introduced by a couple of authors Kraus and Pfeifer in (83, 84), demand some approximation or rough estimate of searched break-line. The authors suggest the raster-based methods as a source of the estimate. Works that do not require any auxiliary data beyond the point cloud for the detection of break-lines were proposed, e.g. (10, 13), however, at the cost of increased computational burden.

Spatial objects on top of the terrain. Within spatial data infrastructures, the data representations of distinct geospatial objects are usually stored and maintained separately from the terrain representation itself. Moreover, they were acquired with different precisions or have different resolutions. Therefore, integration objects and raw terrain data may give imprecise results, unless the semantics of objects and local constraints, they impose on terrain shape, are considered. Flat surface of roads and lakes, or rivers flowing downhill, stand for an example of such restrictions.

For the sake of virtual reality systems, many works focused merely on visualization aspects of polyline-terrain integration (3, 12, 69, 123, 134). Generally, this option converts the vector data to a texture image layer and combines this polyline image layer with the primary terrain image layer. Within the GIS field, which concentrates more on a potential of spatial analysis, the integration of actual geometries is desired. Geometry based methods insert the vector data directly into the terrain mesh itself (144, 146). Two general approaches can be identified for integration of poly-lines into TIN.

The first, which inserts poly-lines directly into the triangulation as constrained edges, is presented for example by (20). However, this approach can cause inappropriate trenches or ridges in the resulting TIN-based terrain model. In the second approach, poly-lines are overlaid on top of the terrain, see (144) for example. (30) achieved this by adding Steiner points to the poly-lines inserted into the TIN. Neither this approach is perfect. For example, it can cause the rivers flowing locally upwards. The suitability of chosen approach to feature-terrain association is data-dependent and the semantics of integrated datasets must be considered like in (75). The work (132) classifies this approach according to the triangulation method that is used for the integration of features with the terrain. Accordingly, the possibilities of the constrained triangulation, conforming triangulation, and refined constrained triangulation were explored. They also suggested the requirement for integrated generalization of terrain and features, however, the proposed method still decimated terrain and features as separate entities without mutual respect.

Scientific works, that would consider geometric properties of both, spatial objects and terrain, in 3D and topological relations between them, are very scarce. Despite the fact, that many analyses, visualization or simulations may benefit from this conjunction. (45) demonstrated this by the case study on 3D city model.

They introduced an aggregation method of 3D building models to reduce the complexity of a city model. The method dealt with the visualization artifacts like immersion or protrusion of the aggregated block of building into, resp. from the terrain surface, see Figure 2.2. Their strategy was to introduce the aggregation error to take into

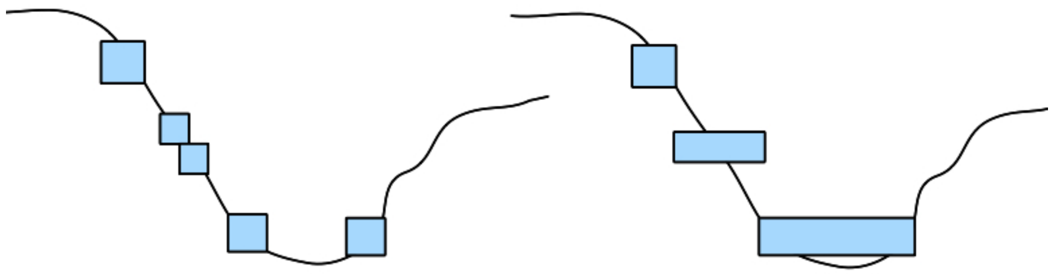

Figure 2.2: Illustration of a visual artifacts - the result of building aggregation in a steep terrain.

consideration the desired height of a resulting building block and the elevation of surrounding tiles of a gridded terrain model. This approach was further elaborated in (51) by employing linear programming, so as to optimize the building aggregation.

2.6 Multiscale data access

With existing terrain datasets having even a global coverage and various thematic layers of geospatial objects nearly the same extent, the resulting DSMs are too large to fit in a main memory. Many conventional algorithms cannot be simply applied, therefore, some researchers have focused on the development of the data management and spatial indexing techniques. Indexing techniques generally are designed to provide rapid access to the data by systematically subdividing the space into smaller, manageable parts. (77) distinguishes two approaches to the decomposition of the space. The data-driven method, whose output depends on the input data. And the space-driven decomposition, which is dependent strictly on the space. The work of (128) stands as an example of a data-driven solution for manipulation with large geometric data, which was also tested on large terrains. It is rooted in a clustering process, which identifies groups in the data. This solution even supports the LOD by building the hierarchy of clusters. However, the variability of data-driven decompositions is its disadvantage, as the decomposition changes whenever the data are updated. Within GIS domain, this can be avoided easily. The space extent is well known in advance, that is why the space-driven decomposition used to be the natural choice within GIS. Moreover, the complex task of computing the data-driven decomposition, which can become very tedious regarding large spatial data stored in distributed databases, is also avoided.

Among space-driven solutions two approaches, the projected subdivision and the direct subdivision, can be distinguished (77). The nature of the first approach is grounded in a projection of the topographic surface to the plane, where the subdivision is accomplished. This kind of solution provides the possibility to use a regular grid of quadrangles as indexing structure, which is the major advantage of planar subdivisions. In terms of computational performance, such a grid is a highly efficient data structure that can be easily deployed by common data structures. However, the disadvantage of projected subdivision for global solution is associated with the projection itself. The flattening of the curved surface to the plane inevitably introduces a geometrical errors. Furthermore, numerical errors can be introduced during the inverse transformation between projections. (136), (1), (60) or (102) presented such global solutions based on projected divisions.

Direct subdivisions of the globe present an alternative, that avoids the flattening of the earth surface. This approach deals directly with 3D space, therefore, it provides an exact mapping of the real world geometry and makes it suitable for various spatial analyses. This is achieved at the cost of higher complexity of geometrical relations in 3D in comparison to the flat 2D surface.

The most prominent direct spatial divisions of the globe, which were designed for geographical applications, are based on discrete global grid system (DGGS). Term DGGS stands for a hierarchical tessellation of regular cells to partition and address the globe. Therefore, a DGGSs provide a uniform environment for sampling, combining, and transforming geospatial data within digital Earth systems (41). Existing DGGS approaches and suitable geometric structures for their deployment were reviewed in (122). An elegant method for indexing spatial data distributed radially around the origin of a coordinate system was introduced by (76). The Global Indexing Grid (GIG) is a DGG based on Voronoi diagram on a sphere in order to tesselate its surface. This method was earlier suggested by (92). In contradistinction with these works, the GIG supports tessellation on multiple resolutions. The GIG method was elaborated in (76, 77). Especially, the GIG was associated with a precise model of geoid to reference the heights of topographic surface correctly. As this method is a starting point of the thesis, it will be described with a finer detail in chapter 3.

2.7 Starting points for feature-enhanced topographic surface

This chapter answered the first research question:

1. *What is the state-of-the-art in large spatial data management and generalization?*

The contemporary research works on conceptual modeling of spatial data, that were carried out with focus on the digital Earth technologies, revealed the need for reconsideration of some of existing approaches. The overwhelming number of spatial features and phenomena, for which the unifying digital representation is needed, led to a design of an entity-oriented paradigm of geospatial managed objects (71, 73, 147). These objects bear not only the geometry information, but also the thematic attributes, temporal characteristics, functionality and behavior of a feature within the system and with respect to other features. This thesis does not extend this concept, but holds it in mind and designs the own methods in that respect.

With regards to the dimensionality of modeling, the most important lesson learnt, refers to the combination and integration of 2.5D and 3D models. 2.5D approach is efficient for modeling of terrain heights, where a single height value at every x,y location is sufficient. At a large scale, true 3D approach to the detailed modeling of complex features is needed. The variety of methods exist, that allow for storing objects within a TIN. The state-of-the-art works in this domain proved the feasibility of integration 2.5D and 3D geometries through the example of TIN/TEN hybrid model (109).

A serious weakness of existing solutions to global surface representation is the lack of geometric flexibility on boundaries with distinct geometric features. It is mainly caused by their focus on visualization performance, whose consequence is the negligence of requirements of the data management and analysis.

To improve these imperfections, it is necessary to have means for data management of vector-based geometries around the globe. That also means support for LOD and careful treatment of topology. The topology preservation is important for many applications dealing with geographic analysis and it may also contributes to the visual quality of the resulting graphic scene. However, the topology was proven to hinder the distributed data management and to cause problems with geometry stitching on boundaries between LODs.

Based on the review, it can be hypothesized, that the procedural approach can help to resolve these contradictory requirements. Some works went in this direction, like (116) or (77). Therefore, the multi-resolution bare Earth representation of Kolář (77) is the starting point of this thesis. However, the solution, that would allow to populate the topographic surface with multiple LOD features and would operate on such principles, is missing.

On the basis of drawbacks of existing solutions and requirements on a new system reviewed in the first two chapters, in Chapter 3, the fundamental choices for the solution proposed in this thesis will be formulated.

3

Conceptual requirements and theoretical foundations for the multi-resolution topography

This chapter formulates arguments for system characteristics, the digital Earth solutions should possess. Especially three main preferences are examined, namely the preference for minimal data redundancy 3.1, the preference for procedural solution 3.2 and the preference for non-projected solution 3.3. It is shown, that the call for the minimal redundancy is closely related with fundamental choices regarding the design of a data representation of spatial objects on multiple levels of resolution. Furthermore, it is presented that procedural solutions play important role in data management and allow for maintenance of a single data model of an object. It is also observed, that the projections are inherent and unavoidable feature of existing approaches, which, from the perspective of geo-spatial information systems, causes complications.

With respect to the preferences, the indexing mechanism in section 3.4 and the representation of bare Earth in section 3.5, which are adopted by the method presented in this work, are formalized. The GIG is utilized as indexing and hashing mechanism. Due to its support to multiple LOD, the multi-resolution environment can be defined on its grounds. The preferences for the design of the solution were discussed mainly from perspective of distinct spatial objects. However, the underlying model of bare Earth must comply with them as well. Therefore, the foundations of suitable procedural bare Earth representation are formally introduced.

3.1 The preference for minimal redundancy

The producers and users of spatial data deal with data capture, data management and data visualization. The strategy of many spatial data producers focuses on building the central database, from which fit-to-purpose data products are derived on demand. Such demands may vary and include visualization as well as different kinds of analyses.

The distinction between the modeling of real world and its visualization was apparent since (50). Within the domain of cartography, the digital spatial models can be subdivided into digital landscape models (DLM) which present an alphanumerical depiction of the landscape, and digital cartographic models (DCM) which present a scale related visualization of the landscape.

Regarding this theory, the term model generalization stands for the derivation of a coarser primary representation of environment (the DLM), from a finer resolution one. That allows for a delineation of model generalization from a generalization for the sake of visualization.

Although this DLM - DCM separation is considered as a theoretically optimal way for maintaining datasets at multiple scales (131), the producers and spatial database administrators still have to answer what to explicitly store, so as to efficiently manage the data on multiple LODs.

The options how to classify the possible approaches to maintain data for single scale systems are quite straightforward. However, when considering multiple scale database, the transformation process that guarantees no graphical conflicts gets more complicated and data managers have to deal with questions what to store. The originally captured representation of geographic objects (DLM) or the geometry optimized for visualization (DCM). The situation gets even more intricate when considering 3D virtual environment with LOD varying within one graphic scene. From cartographic point of view, this type of maps is termed mixed-scale maps (97). The inspiration for the following four possible approaches can be taken from the approaches of multiple national mapping agencies, which struggle for automation of traditional map production on multiple scales.

Pure dynamic approach. The first choice is to store the geometry of real world objects in its original form, utterly non-adjusted for any kind of visualization. Although having the original geometry available is an advantage for analytical operations, the derivation of graphic form of object would have to be performed entirely at run-time. Although plenty of research focused on automation of geometrical changes of graphical objects between different LOD, this approach is manageable at reasonable time responses for limited data volume only. Especially computationally demanding operations are such, that require the complex changes in object's geometry or even a creation of a (partly) new geometry, e.g. aggregation, collapse or displacement of objects.

In the multi-scale or vario-scale environment some acceleration can be achieved by storing various representations of object geometries, thus representations already simplified to coarser resolutions. As a consequence the computational time for creation of generalized geometry is saved and only the topological conflicts between objects generalized for certain LOD has to be handled dynamically. The data redundancy is an obvious consequence. As well, it is necessary to create and maintain links between the corresponding objects on multiple LODs, otherwise the data inconsistency may be induced (98). Moreover, the adjustment of the geometrical conflicts is still too expensive operation for run-time applications when dealing with big data.

The situation is even more complicated, when dealing with the mixed-scale environment. In case of multi-scale system, the geometrical conflicts between objects within each distinct LOD must be resolved. In case of the mixed-scale environment, in addition to the previous adjustment, the conflicts between different LODs (at borders between LODs) must be dealt with. When different representations of object for each LOD is used, the reconstruction of an object on a boundary between LODs has to be accomplished from two different representations. Thereafter, the adjustments of geometry caused by topological conflicts with surrounding objects must be performed with respect to the transitional geometries reconstructed on the boundaries between LODs.

Pure graphic-oriented approach. The second alternative is to store only the graphical representation of spatial objects.

On the mutual intersection of computer graphics and GIS scientific domains, there was a lot of effort aiming towards the visualization performance of the terrain representation. The representation of distinct 3D spatial objects like 3D city models, that are optimized for graphical performance, are especially the subject of computer graphic domain aiming at fast rendering of 3D scenes in games or movies. Graphically optimized solutions of 2D representations of spatial objects datasets is also a domain of cartographers.

The consequences of the approaches in all these scientific domains have some similar features. A common drawback is an extensive preprocessing of the original data, which results into special data structures efficient for fast rendering. This is achieved at the cost of mixing original representation of real world objects with visualization details. Such object representations are unsuitable for analytical purposes and may cause some limitations to applications that deal with distributed data.

Considerable disadvantage is the ability to maintain, particularly update, datasets in a uniform manner. In a mixed-scale environment a fixed representation of an object have to be stored considering every LOD and the transitional representations of object lying on boundaries between LODs as well. It is important to understand, that there is not only one such a transitional representation needed, but several representations have to be created with respect to all distinct positions of observer. The number of such positions is conditioned by the system of LOD space distribution, typically space indexing system.

Combined approach. For the sake of completeness, it must be mentioned that both, the original representation and the preprocessed representation optimized for visual performance, can be stored persistently. This approach enables rapid access to both the geographic objects and the graphic objects, nevertheless, significant redundancy is induced. As well, the data management and creation and maintenance of links between representations of corresponding objects, is non-trivial.

Adjusted dynamic approach. The last option is a compromise between pure dynamic and pure graphic approaches that leans towards the dynamic approach. This approach is rooted in dynamic reconstruction of the scene. However, it allows to preprocess data according to defined rules and tolerances, that would respect the required quality of the dataset. The key step in this regard is the mechanism of the assignment of "weights" to geometrical primitives, according to which the process of geometry simplification between LODs is driven. Proper setting of the "weights" is supposed to guarantee the conflict-free visual representation at coarser LODs. This evaluation of geometry does not induce any data redundancy.

The general motivation of this approach is the minimization of data redundancy, the minimization of the need to establish the links between objects on various LODs and the minimization of the changes of the original geometry.

For example, insertion of a vertex into the edge is a change of original geometry, but it does not affect the accuracy of original representation, thus does not hinder any analytical application using such data. However, it may be helpful to preserve the topological relations between objects or treat the changing LOD at the borders between LODs.

In case some higher level generalization operations are desired, the need for a new geometry representation may become inevitable, e.g. object's aggregation, collapse or typification operations. In this case, the geometrical distortions should appear within defined limitations. After all, even the simplification of a poly-line by omitting some of its vertices at coarser LODs, induces a distortion of original geometry and change of representation's accuracy.

Finally, the adjusted dynamic approach makes the storage of the original object's representation persistent. For a mixed-scale environment with determined spatial indexing structure and accuracy limitations for every LOD, the graphical representation can be derived from the original at run-time. Although this approach cannot entirely prevent from multiple LOD representations of object, it facilitates the establishment and management of links between them. The main argument for this approach is, it makes the data update and maintenance easier because only a single source data set is concerned.

3.2 The preference for procedural solution

The expanding capacity to collect geometric data resulted in such datasets that just do not fit in the computer's main memory. The in-memory and external memory (out-of-core paging) are two major approaches for the use of memory.

The in-memory operation mode is the computing system that relies on main memory for data processing. This mode provides high performance, but only when data of limited size are concerned. This limit is set by the size of available main memory. Owing to the data volume of spatial data with global extent coverage this operation mode is unsuitable in the context of this work.

The external memory operation mode is a system based on algorithms that are designed to process data that are too large to fit into a computer's main memory at one time. Such algorithms are optimized to efficiently access data, which are stored in a slower bulk memory such as tapes or hard drives (142).

Applications in geographical information systems, especially digital terrain or digital surface models, where the full data set reaches several terabytes of data, are usual example of usage. The solution is rooted in the concept of pages, which are elementary patches of data, i.e of the terrain data. In GISs, the decomposition is typically space driven, in such a case the paging is associated with some space subdivision. The page of decomposed geometrical data set fits into the main memory, which also allows for the distributed management in a network environment. Usually the square tiles (58, 107) or Morton ordering (5) methods are employed to decompose the terrain data sets. The recent research efforts in the topographic surface representation have been focused on the visualization performance. Works of (17, 22, 91) are grounded on the out-of-core mode paging. Although the extensive data preprocessing is suitable for the sake of data rendering, it simultaneously imposes limitations on analytical and distributed applications, which are cardinal for GIS field.

The procedural approach is an operation mode, which also deals with data sets exceeding the main memory capacity. However, it tries to avoid some of the disadvantages of the out-of-core paging of the previous operation mode, especially storing pre-generated pages of memory, which imposes limitations on the size of database and data management (77).

Isenburg presented in (61) an approach for streaming Delaunay triangulations, which differs from external memory algorithms. The main difference inheres in the fact that nothing is temporarily paged out to external memory. A streaming algorithms make a small number of sequential passes over an input data file (typically just one pass), and process the data using a memory buffer whose size is a fraction of the stream length. (62) proved that streaming can be applied to huge point clouds for the sake of generation of Delaunay Triangulation or gridded DEMs.

The crucial advantage of the procedural mode inheres in the LOD management of large geometries, which is the key prerequisite to manage large geometries of spatial data effectively. Not only that the coarser levels of detail requires less data volume to be handled. The LOD in 3D GIS is also desirable for data understanding in the same way, as generalization in traditional cartography is a desirable feature for map's legibility.

The major problem for geometric data is the need to "stitch" the pieces of different LOD together, as observed in (116). That is because of the lack of topological correlation between different LODs. As most attention of researchers were paid to visualization aspects of large terrain, there is only few works focusing on dynamic management of LOD, which offers a solution of this issue.

An elegant solution was introduced by (77, 80) for the sake of the multiple LOD bare Earth representation. The approach is based on extremely simple data structure of unstructured points. The used procedural approach reconstructs the most of geometric detail only when needed. The storage of topological information is completely avoided and reconstructed by Delaunay triangulation only at run-time. This solution demonstrated great flexibility for data management.

The work (77) addressed a remedy of earlier work of (116), whose solution was applicable only for areas of limited extent with planar offset. (116) did not maintained distinct geometric models for every LOD, but updated them procedurally. The update operation was driven by a sytem of "grades", which were assigned to every point of DTM depending on the importance of a point in the terrain approximation.

3.3 The preference for non-projected solution

The 3D graphic environment directly enables visualization and exploration of the 3D geometry of spatial data. This absence of a flat media, that in a paper form accompanied cartography for centuries recently allowed to reconsider some fundamental approaches to the visualization of spatial data. The essential one is the need for the data projection.

The objective of cartographic projections is to systematically transform locations on a sphere or an ellipsoid into locations on a plane. (72) or (77) thoroughly reviewed the drawback of projections, namely "geometric distortion, numerical imprecision, limited spatial extent, theoretical heterogeneity, interoperability issues and computational overhead, which are common to all two-dimensional geographic systems, but also to systems that exploit three-dimensional graphics on top of a flat-map approximation."

From the perspective of digital Earth systems, the use of projection or, by contrast, its avoidance, is an important aspect regarding the way of geographic space decomposition. The decomposition of space allows for efficient access to data by processing only a small partition of space. The usage of projection in the space partitioning is well-established. The most common decomposition of the earth surface mapped to the plane is the hierarchical subdivision using quads. It results in a regular grid, also termed as raster data structure. This kind of data structure is supported by nearly all GIS softwares. Great number of various spatial datasets was stored as 2D rasters. These are the key advantages of this approach. However, the disadvantages related to cartographic projection become an integral part of raster datasets. Whereas for GIS applications of limited spatial extent this drawbacks can be manageable, for global solutions, it poses a major restriction.

The direct subdivision represents an alternative solution, which is based on coordinate system associated with 3D space. The number of existing approaches to direct subdivision is quite limited.

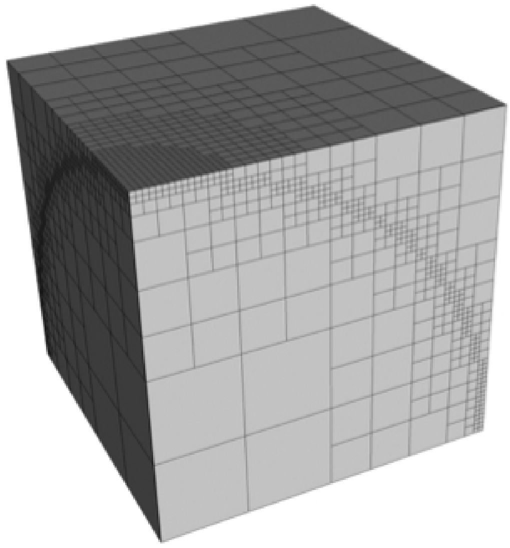

Figure 3.1: The octree - the subdivision of space based on an octree (135).

The octree is probably the most used hierarchical subdivision of space. It is based on Cartesian coordinate system as depicted in Figure 3.1. The use of octree was introduced by (95) to the field of computer graphics, but application for GIS are also frequent. However, for digital Earth solutions, there are two main arguments against the octree approach (77). First, the upwards direction on the Earth surface is difficult to be represented by octree, as it has a variable orientation. Second, the octree approach may become inefficient, because most of the data to be indexed are situated near the surface. These arguments support direct divisions of space rooted in spherical coordinate system, which is in fact a natural approach considering the spherical distribution of geographic features as well as their orientation. A subdivision of space through tessellation of sphere brought forth a construction of discrete global grids (27, 76, 92, 122).

3.4 Global Indexing

On the basis of requirements on the data management, which were stated in the foregoing text, the Global Indexing Grid (GIG) will be adopted in this work and employed as an indexing and paging mechanism. The multiple LOD environment is defined on its grounds.

GIG is a DGGS first described by Kolář in (76) and further documented in (77). It represents a system of sphere subdivision based on Voronoi DGGS, whose important property is a support for applications with multiple LOD. The level of detail of an indexing grid is driven by a division coefficient dc, which determines the distribution of the centroids defining the resultant Voronoi diagrams. As was defined by (77), the distribution of the GIG centroids is driven by two rules.

Starting from the south pole of a spherical approximation of the Earth proceeding to the North by $\Delta\phi$ along prime meridian, the increment in latitude $\Delta\phi$ for given dc is obtained by

$$\Delta\phi = \frac{\pi}{dc} \tag{3.1}$$

The longitude increment $\Delta\lambda$ corresponds to a number of centroids $n(\phi)$ along the ϕ parallel

$$n(\phi) = \lfloor 2dc \cos\phi + 0.5 \rfloor = \frac{2\pi}{\Delta\lambda} \tag{3.2}$$

GIG Cell. Voronoi diagrams divide the space according to the nearest-neighbor rule. Works within natural sciences address Voronoi diagrams under different names specific to the respective area. The use of the diagram can be traced backed to Descartes in 1644. Peter G.L. Dirichlet in 1850 used 2-dimensional and 3-dimensional Voronoi diagrams in his study of quadratic forms, therefore, the Voronoi diagram is also sometimes termed as Dirichlet tessellation. In physical geography, the Voronoi diagram is

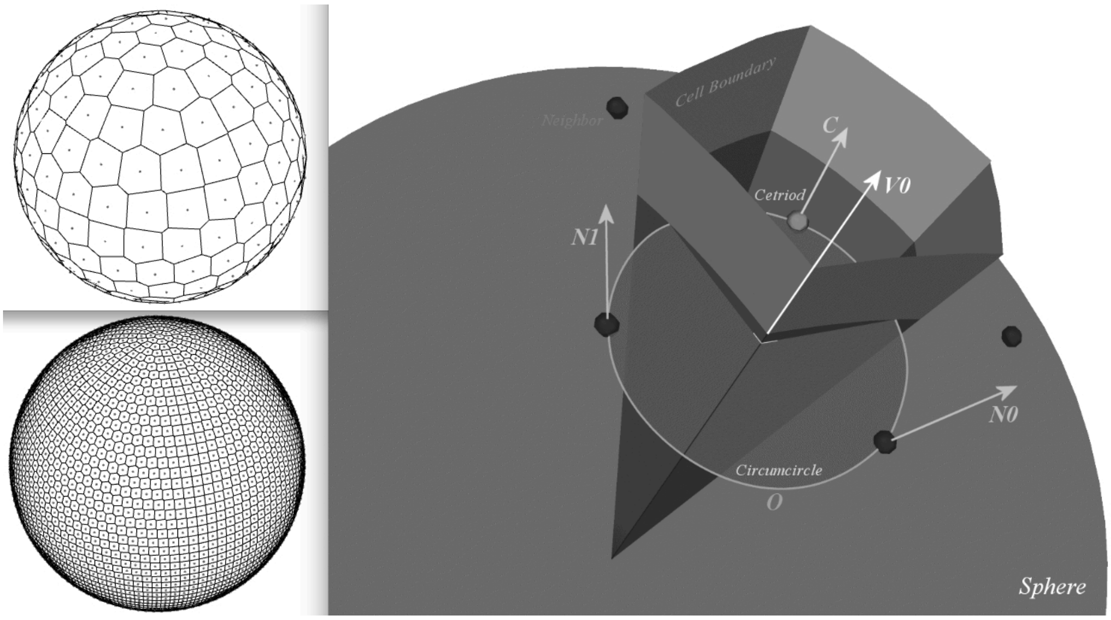

Figure 3.2: Illustration of a GIG DGGS - (left) GIG scheme and GIG centroids at subdivision levels 16 (top) and 64; (right) visualization of GIG cell and of the relationship between cells vertex vector and neighboring centroids. Figure drawn for (11).

called Thiessen polygon after a meteorologist Alfred H. Thiessen. Let us start with the formal definition of Voronoi cell V in a general metric space χ with distance function d, e.g. (43). Let K be a set of indices and let $(P_k)_k \in K$ be a tuple of nonempty subsets (points) in the space χ. The Voronoi cell V_k associated with the point P_k is the set of all points in χ whose distance to P_k is smaller or equal to their distance to the other points P_j, where $j \neq k$. In the particular case of 2-dimensional (planar) Euclidean space, we can write for the Voronoi cell V_k:

$$V_k = \{r \in \mathbb{R}^2 \mid \mathrm{d(r, P_k) < d(r, P_j), for\ all\ j \neq k}\}, \tag{3.3}$$

for d denoting the euclidean distance function.

While in planar Cartesian coordinates, the distance d between two points a and b is given by the relation $d_{AB}^2 = (x_A - x_B)^2 + (y_A - y_B)^2$, for the indexing along the sphere surface, the angular distance is essential, as it provides a measure for the construction of a Voronoi diagram on the sphere (77).

For angular distance $\hat{d}$ between two vectors C and N_0 using 3D Cartesian coordinates (x, y, z) holds:

$$\cos \hat{d}_{CN_0} = \frac{x_C x_{N_0} + y_C y_{N_0} + z_C z_{N_0}}{||N_0||||C||}, \quad (3.4)$$

where $||C||$ is the norm of C.

The application of a method for distribution of centroids documented above provides a Voronoi diagram on the sphere, for illustration see 3.2. Where, in other words, a single (GIG) cell is a set of all vectors having an angular distance from a given centroid C smaller than from any other centroid.

Moreover, this kind of centroid's distribution enables multi-resolution tessellation in accordance with multiple GIG levels. The levels are determined by their respective division coefficient dc. The value of dc spans from 1 for the coarsest level and with the upper limit for finest LOD given only by the numerical precision of the computer implementation. For further detail, see (77).

Indexing with GIG. The target cells of GIG queries are determined through the centroid, which makes the GIG queries simpler and more robust than in case of determination based on geometrical boundaries of an index's cell. The GIG queries were designed with respect to needs of instant visualization of topographic surface and its analysis. For this sake, the proximity queries like nearest-centroid, 4-Nearest-Neighbors or Cell Neighbors were presented. The output of GIG methods is a cell identifier. This can be denoted in a following way:

$$GIG\{x,\ y,\ z,\ level_{GIG}/accuracy\} \rightarrow CID_{GIG}. \quad (3.5)$$

The positional accuracy is an inherent part of the CID definition as it is a measure inseparable from the location referenced by the identifier (77). The nearest-centroid (NC) GIG methods can be utilized as a hash function. The NC method assigns an arbitrary point $p(x,\ y,\ z)$ to a unique GIG cell c based on the proximity to the centroid

$$NC(x, y, z) \quad = \quad \mathbf{c}|C \in \mathbf{c}. \quad (3.6)$$

Exploiting GIG can be perceived as hash indexing. Nevertheless, the retrieval of data already associated with particular GIG cell is achieved through traversing a sorted order of cell identifiers.

GIG queries play important role during data processing, when all neighbors of processed cell are required. For the cell c the NN query returns all adjacent cells on given GIG level L

$$NN(c) \quad = \quad \{c_1, ..., c_j\} \mid NN \subseteq L, \tag{3.7}$$

where j is the index of the last neighbor cell.

3.5 Bare Earth data representation

One of the primary concerns of this thesis is a more functional representation of topographic surface, which would support geometrical changes of the topographic surface in connection to other geographic features of various kinds. To fulfill the preferences stated in foregoing sections of this chapter, the proper representation of the bare Earth must be selected.

It is advantageous to avoid terrain representations based on a recursive subdivision of a regular grid, because it is a core concept that restricts having arbitrary polygonal edges on the surface. Furthermore, the need for pre-computed patches of the surface should be minimized and as a consequence, the ability to adapt to new features improved.

The surface topology reconstructed only at run-time using Delaunay triangulation is one approach, which can also address the topological issues of the surface on the boundaries between different levels of detail (LOD). For these reasons, the bare Earth model produced in (77) will be adopted for the experiment of this work, see Chapter 6, as a surface, to which other spatial objects will be referenced.

Kolář in (77) introduced a representation of the topographic surface of a global extent with multiple LODs that uses unstructured points as the primary geometric data structure.

Every surface point that describes the terrain is stored as one record. It is formed by a three-tuple

$$\{dc,\ CID_{GIG},\ \{x, y, z\}\} \in \tau, \tag{3.8}$$

where dc stands for the GIG division coefficient introduced in previous section. CID_{GIG} stands for cell identifier and $\{x, y, z\}$ denotes the geocentric Cartesian coordinates of the point. The implementation introduced in (77) allows for less redundant storage by avoiding use of both dc and CID_{GIG} due to the unique character of the identifier.

The desirable feature of this data structure is its great flexibility regarding data management. Update operations are straightforward thanks to the absence of stored topological relations, which are reconstructed only at run-time. The GIG indexing method introduced in the previous section is also a part of the reconstruction. On the basis of the query position (observer's location), the paging mechanism retrieves the records from relevant cells on multiple LODs. The illustration of such cells relevant to 4NN query, which consequently delimit the boundaries between LODs in the reconstructed scene, can be seen in Figure 4.3.

3.6 Closing remarks

This chapter answered the two following questions formulated in the introductory chapter:

2. What indexing and paging data structure should be adopted, so as to simultaneously support the multiple LOD and fulfill the preference for non-projected solution?

3. How can be formally described what is multiple LOD environment?

The fundamental choices for the solutions of a digital Earth kind were discussed, namely:

- the preference for minimal redundancy,
- the preference for procedural solution,
- the preference for non-projected solution.

Furthermore, this chapter provided insight into the global indexing and paging mechanism, which the method proposed in this thesis adopts for multiple LOD data management of feature-enhanced topographic surface. The procedural representation of bare Earth, which properties are in agreement with requirements laid on the solution proposed in this thesis, was adapted.

4

Feature-based enhancement of multi-resolution topographic surface

This chapter proposes an original theoretical framework for multi-resolution topography. It starts in section 4.1 with the formalization of the concept of footprint, which can be seen as a logical and functional interface between the spatial object and terrain. Further in this chapter, the 4.2 section introduces a method for analysis of the footprint, in order to ensure an initial data validation and adjustment. The primary data structure of footprint's geometry is defined in 4.3. The paging of footprints, which is fundamental for the creation of a multiple LOD database, is proposed in 4.4, followed by the method for a reconstruction of the topographic surface in Section 4.5. The properties of the presented solution are discussed in detail, as well as the contributions to the existing solutions from several scientific domains.

4.1 The concept of footprint

The provision of a topographic surface representation, which is capable of accommodating any geographic feature, that can influence its shape, relies on the concept of footprint. The choice of this concept is the combined outcome of the preferences, explained in the previous chapter, and the pragmatic modeling approach to spatial objects. The initial feature-oriented modeling is similar to the Formal Data Structure

(101), which enables the representation of four types of features: point features (0D), line features (1D), area features (2D) and volume features (3D). The footprint is defined here as the outline of the geometry resulting from the intersection between the Delaunay triangulation of topographic surface and the 3D geometry of the feature. An illustration of a footprint is shown in Figure 4.1. It can be understood as the link between the topographic surface and an independently modeled spatial object. The inspiration can be taken from works of (108, 109, 110) on 3D topography, whose solution is built upon the footprint-based integration of TIN/TEN producing hybrid 2.5D/3D model. This is a pragmatic approach, because objects modeled in 0D, 1D, or 2D can be represented in a TIN. 3D objects are modeled as separate TENs only in such cases, when its absolutely necessary (bridges, tunnels). The (108, 109, 110) works, however, did not address the issues related to multiple levels of detail.

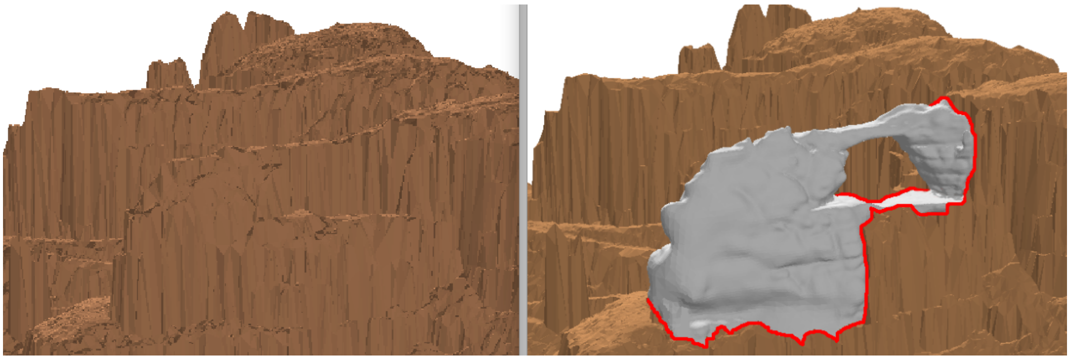

Figure 4.1: Illustration of a footprint - The figure illustrates the overall concept of footprint, which is rooted in the integration of TIN with poly-lines in 3D space. (left) The view on the rock city in the National Park České Švýcarsko. The 2.5D TIN terrain model acquired by airborne laser scanning techniques is not able to represent complex nature of the area. (right) 3D mesh object of Pravčická brána, sandstone rock arch, modeled from terrestrial laser scanner data, integrated with the 2.5D model of surrounding rock city through the footprint depicted by the red line. Figure first used in (11).

Although the method proposed by this work is designed to be generally applicable, it considers the GMO concept (73, 78), which associates geometric and thematic properties and even behavior of a spatial objects within a single entity. From the perspective of this work, especially the ability to convey the information about the desired appearance of the GMO on multiple LODs, is convenient. The need for geometrical simplification on coarser LODs inherently requires the ability to distinguish the importance of features and to determine the importance of every building element of their geometries.

This is usually achieved through the concept of weights. The weight can be inferred from the geometrical importance like in (26), or consider the semantic influence like in (85). There is, however, no uniform solution for weight assignment, as this problem is task driven and should be tailored to particular case study or user's needs. The determination of weights becomes even more difficult, when considering independently modeled objects and their interaction with each other, or with field-based models of terrain heights. This issue is not satisfactorily solved by any existing study within GIS field of science. Our design idea for a footprint is such, that it acts as a logical and geometrical interface between features mutually, and between feature and the bare Earth model. The concept of footprint also complies with the preference for a procedural solution, which employs the constrained Delaunay triangulation as the method of topographic surface reconstruction.

Accordingly, the data representation is propsed, which complies with the preferences stated in Chapter 3. So as to ensure the functional extensibility and general applicability, the fundamental concept is kept as simple as possible.

The footprint's building element is a *poly-line* P, which is a finite, non-empty sequence of m points p in $\mathbb{R}^3$

$$p(x, y, z) \quad \in \quad P = \{p_1, ..., p_m\} \mid P \subseteq \mathbb{R}^3 \wedge P \neq \emptyset. \tag{4.1}$$

Weight of footprint points. To take into account importance of different parts of footprints every point p is assigned a *weight* ω. The weights corresponding to points of P constitute the non-empty sequence W of weights ω

$$\omega \quad \in \quad W = \{\omega_1, ..., \omega_m\} \mid W \neq \emptyset. \tag{4.2}$$

Footprint definition. Then the formal definition of footprint F is a non-empty set of pairs of poly-lines and weights' sequences, which reads

$$F = \{(P_1, W_1), ..., (P_s, W_s)\} \mid F \neq \emptyset. \tag{4.3}$$

The method requires that every feature provides one footprint F.

4.2 Analysis of footprint's topology

This section discusses the initial processing and analysis of input data and present an automated method for data validation and adjustment. This need is driven by the relative independence of features and the preference for minimal redundancy. The analysis of topology between features' footprints is an important step towards a data representation for multiple LOD environment.

4.2.1 Need for a data validation

Most spatial databases are based on object-first approach, following the Simple Features standard (57). Often, spatial objects are stored in databases as a set of individual object. Moreover, since one of the objectives is high flexibility of proposed system, means that allow to incorporate various data sources from possibly different providers must be available. Thus, one can assume the datasets were not managed with mutual respect and the topological relationships between objects are not stored. Inconsistencies will often be introduced when geometries of such spatial objects are associated within one graphic scene. Consequently, due to the relative independence of the features, cases such as identical points, overlapping or intersecting edges are allowed, and must be detected and handled. Two meanings of such inconsistencies can be distinguished.

1. Geometrical and topological errors introduced due to the differences in data acquisition accuracy or numerical precision, are the first one. In practice, they result in overlapping polygons; gaps between polygons; distinct concurrent lines, or multiple points that should be unified, as they represent only one phenomenon.

   Proposed method for footprint analysis relies on a simple approach, which is frequently employed by dynamic triangulation methods, e.g. (68), to handle the degenerations such as edge overlapping, self-intersections or duplicated points, and automatically fix them. The machinery precision, based on an epsilon value and standard floating-point arithmetics, is used to define the threshold, from which multiple points are considered to be an identical one, or a standalone point to be located in the edge. It is assumed that the user is able to set the proper epsilon value to reflect the required precision of the method and the nature of the input data.

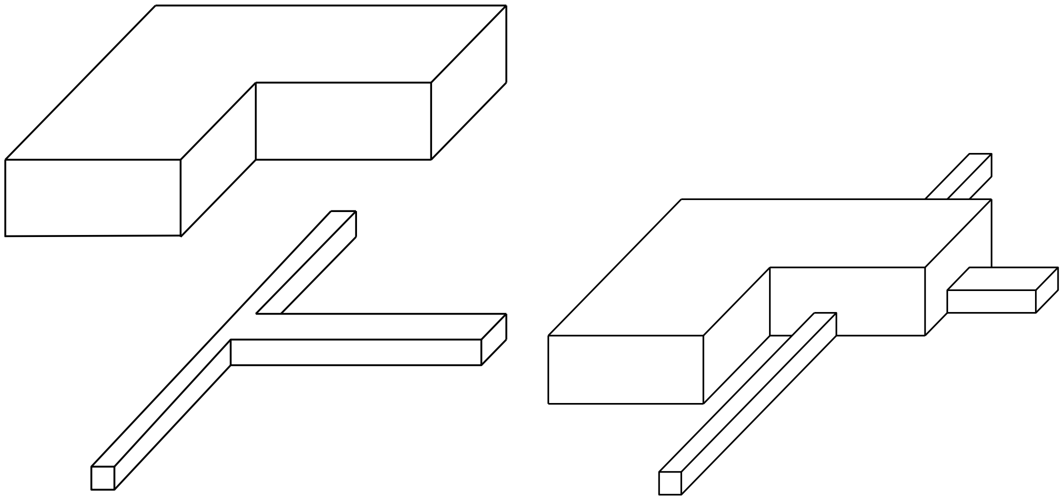

Figure 4.2: Two distinct datasets. First one might represent the footprint of an oil pipeline on the ground. The second high-pressure stations' buildings that are entered by such pipelines.

For some cases of disparate datasets, more elaborate methods may be needed. Beyond the solutions based on geometrical and topological validation rules, there are methods based on constrained triangulation and triangles flagging, e.g. (86).

2. The second meaning of inconsistencies may not be errors, but rather a consequence of different semantics of objects. An example is depicted in Figure 4.2. Two stand-alone datasets of objects, which describe, different phenomena, each can be visualized without the other one. However, if the character of an application requires appearance of both layers of objects in the scene, the mutual topology must be reconstructed and further preserved.

4.2.2 Weight adjustment

In order to preserve the mutual topology between the footprints' poly-lines in multiple LOD environment, the weights ω of some points are adjusted with respect to other features. The adjustment of weights is done prior to simplification of any points for a coarser LOD according to the rules listed in Table 4.1.

Table 4.1: Weight adjustment rules

Rule	Topological phenomenon description	Weight adjustment
1.	The first and the last point p of poly-line.	$\omega(p) \leftarrow \infty$
2.	Point p, in which 3 or more non-overlapping edges meet.	$\omega(p) \leftarrow \infty$
3.	Intersection point p inserted into edges intersecting beyond any of their endpoints	$\omega(p) \leftarrow \infty$
4.	Point p inserted into the edge due to the presence of a point v on that edge	$\omega(p) \leftarrow \omega(v)$
5.	Identical points p, v	$\{\omega(p), \omega(v)\} \leftarrow \max(\omega(p), \omega(v))$

Through the first three rules, the weights of some points are set to the maximum possible value. This concerns such points, that encode the connectivity between poly-lines. It aims at preservation of these topologically important points during the simplification of geometry, when the coarser representations of footprints are being created.

Rule 4 deals with points touching another edge. Such point is inserted into the edge and assigned the weight of the existing point, unless 3 or more non-overlapping edges meet in this point (Rule 2). This corresponds to the case of point d in Figure 4.3. Rule 5 handles the case of multiple points at one location, like points b in Figure 4.3. These matched points accept the weight of the point with the highest weight, unless Rule 1, 2 or 3 occasion. As the result of Rules 4 and 5, the weights are equalized to keep the boundary common and without gaps after potential simplification.

4.2.3 Footprint analysis of independent features

In this subsection, the method for automatic analysis of footprints is proposed. The procedure is formalized by the means of the Algorithm 1. The poly-lines of footprints F (4.3) present an input of the algorithm. The algorithm detects multiple points at one location and intersecting edges. Moreover, it handles the (partially) overlapping edges and as a consequence unifies the common boundary between footprints.

The final output consists of the list of unique vertices (*vertex_list*) and the list of unique edges (*edge_list*), which are relevant to all footprints. Furthermore, in the lists

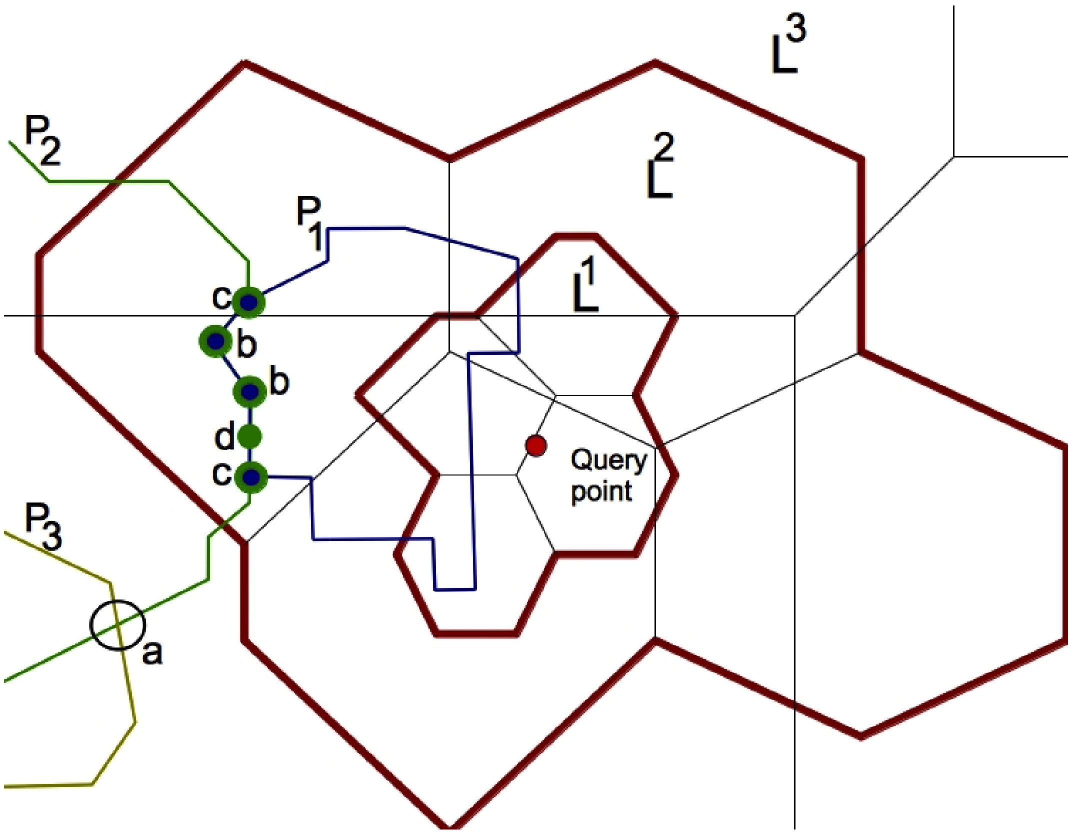

Figure 4.3: A schema of 4NN planar partitions. - A schema of planar partitions corresponding to the four-nearest-neighbor GIG cell query crossed by four poly-lines with various topological relations. Point a encodes the result of intersection of two edges beyond endpoints. Points c represent the change of neighborhood. Points b and d are internal points of common boundary of P_1 and P_2 poly-lines. Points b represent the matched identity points, originally presented on both poly-lines. Point d, originally part of P_2, occurs on edge of P_1, as an element of shared common boundary.

Algorithm 1 Footprint analysis

INPUT: A set of footprints

OUPUT: Edge lists and vertex lists with weights adjusted

```
for all P do
    for all p ∈ P do
        if (p is the first point of P) ∨ (p is the last point of P) then
            p.ω ← ∞
            location_result ← locate_point(p)
        if (location_result is an existing vertex) then
            vertex ← location_result.located_vertex
            vertex.ω ← max (vertex.ω, p.ω)
            vertex_list_P.add(vertex_list.get_index_of(vertex))
        else
            vertex_list.add(p)
            vertex_list_P.add(vertex_list.get_index_of(p))
Create edge_list and edge_list_P for all footprints
for all vertex_list_P do
    for all p ∈ vertex_list_P do
        location_result ← locate_point(p)
        if (location_result is on existing edges) then
            vertex = insert_point_in_edge(location_result.locate_edges, p)
            vertex.ω ← p.ω
for all edge_list_P do
    for all e ∈ edge_list_P do
        intersected_edges ← crossed_by(e)
        for all edges ie ∈ intersected_edges do
            if (intersection point v is beyond e endpoints ) then
                insert_point_in_edge(e, v)
                insert_point_in_edge(ie, v)
                v.ω ← ∞
            if e is identical to ie then
                unify_edges_in_edge_lists(e, ie)
        if more than three unique edges meet in v then
            v.ω ← ∞
return {vertex_list, vertex_list_P, edge_list, edge_list_P}
```

vertex_list_P the indices to points in *vertex_list* are held, and in the lists *edge_list_P* the indices to edges in *edge_list* for every poly-line on input are kept. [1]

The decimation importances of points are inferred accordingly to the rules in Table 4.1. As a consequence, the term unification of input poly-lines refers to the geometry as well as to the asserted weights.

In practice, the analysis is performed in the plane, on which the footprints are orthogonally projected. With respect to the means of the final topographic surface reconstruction, the tangent plane, taken at the centroid of corresponding GIG cell, is the most suitable in context of this work.

Algorithm 1 has three main loops. The first loop (lines 1-12) matches identical points and lists all unique vertices. The resulting point is assigned a weight according to Rule 5 defined in previous section. In the *vertex_list* only unique points are kept. *vertex_list_P* contains lists of indices to points in *vertex_list* which in order constitutes individual poly-lines of associated footprints.

The second loop (lines 14-19) inserts the points located in the edge beyond its endpoints into this edge updating relevant vertex list, edge lists and associated weight ω as for the Rule 4. As a result, there are no partially overlapping edges.

In the third loop (lines 20-31) intersecting edges are handled. First, additional vertices are created, when edge intersects another edge beyond endpoint. Relevant vertex lists and edge lists are updated. The intersection between two edges beyond the endpoints always indicates the change of neighborhood or the connectivity of footprints. Such points are to be preserved due to the Rule 3. Thus weights of such common points are set a maximum value. If the edges are matched as identical, the edges are unified in associated edge lists.

Based on the unified *edge_list*, if three or more unique edges meet in a point, its weight is set a maximum value according to Rule 2.

As a result of the the third loop only unique edges are kept. The unique edges enforce sharing a vertex by all adjacent footprints. Each vertex is associated with its decimation importance ω.

[1] *vertex_list_P* and *edge_list_P* are important as the links between the representation of the original spatial object (which may have been associated with some behavior, functions or complex 3D volumetric geometry) and the resulting footprint. This connection is important for applications and possible extensions of the method presented in this thesis, but not central to the method itself.

In the Algorithm 1 the routines *locate_point* and *crossed_by* symbolically denotes the detection of identical points, overlapping and intersecting edges. The detection of point-on-edge, and edge-edge-intersection are the most expensive parts of Algorithm 1. For all points n, constituting all poly-lines on the input, there can asymptotically be up to $O(n^2)$ intersection points, if every edge intersects every other edge. The sorting can be used to accelerate the computation of edges intersection or point on edge detection. The sweep-line algorithm for detection of intersecting edges like Bentley-Ottmann (7) reaches the $O((n+k)\ log(n))$ time, where k is the number of intersections. For k less than or equal to n the Bentley-Ottmann is the expected $O(n\ log(n))$ time algorithm. However, when k is of n^2 order, the algorithm takes $O(n^2\ log(n))$, thus even worse then the brute force algorithm. Such a case is highly unlikely considering the character of input spatial data.

Point-on-edge detection is the extreme variant of edge-edge-intersection detection. It reaches asymptotically $O(n^2)$ time. For identity points matching, using linear ordering of points, $O(n\ log(n))$ time can be reached.

On the basis of the footprint analysis outcome, the next section introduces a method for creation of multiple LOD database of footprints.

4.3 Primary data structure

The primary geometric structure for poly-line representation is the record

$$\{IDP, \{(p_{s1}, ord_{ps1}), ..., (p_{e1}, ord_{pe1})\}, ..., \{(p_{sz}, ord_{psz}), ..., (p_{ez}, ord_{pez})\}\} \in \quad \rho_{cid}, \tag{4.4}$$

which holds the footprints' poly-lines geometry relevant only to the LOD of the GIG cell c_{cid}. The record ρ is the pair of the unique poly-line identifier and the set of sequences of points together with their order in poly-line. The separation of a single sequence of poly-line's points into a set of subsequences happens, if one or more intermediate points of poly-line lies out of the cell c_{cid}.

4.4 Paging of footprints

The objective of this section is to introduce a method for creation of the multiple LOD database of footprints, which would be associated with the paging mechanism. A geometry simplification of the footprints' geometry is central to this method. Consequently, the simplification procedure formalized here is a crucial component in pursuit of effective run-time synthesis. The synthesis returns the topographic surface with features, that have their topological relations preserved for arbitrary query position of the observer.

4.4.1 The need for simplification and the stop criterion of the simplification

Within the virtual Earth environment, for the viewer's position, a paging mechanism can be obtained, which provides data for a relevant neighborhood at multiple LODs. The objective of geometry simplification procedure is to guarantee, that the LOD hierarchy of pages will give back just the manageable amount of data for visualization. Two main strategies can be identified for the decision-making on whether the goal of simplification has been reached.

Count stop. The first one relies on setting a specific number or a portion of points to be removed. The simplification procedure then tries to remove sufficient number of points to achieve the goal. This can be reached from the local perspective, when for every input poly-line certain percentage of points is removed. Or, from the global point of view, when for all input poly-lines the share of decimated points is counted. The procedure typically starts with removal of points having the least weight according to an applied importance measure.

Epsilon-based stop. The second approach is based on the determination of threshold ϵ associated with corresponding LOD. All points having the weight $\omega < \epsilon$ are supposed to be removed. In case the weights are set according to a purely geometrical criterium, the interpretation of the ϵ threshold can be described as the maximum allowed geometrical error, that can be introduced by a point removal. Therefore, the thresholds guarantee certain level of accuracy for given LOD. In proposed method, the second strategy is followed.

4.4.2 An overview of the simplification approach

Existing approaches to poly-line simplification were reviewed in 2.4, where the need for "en mass" solution was mentioned. The other approaches consider only a single poly-line, or their outcome depends on an order of poly-line processing. This "en mass" solution modifies the complete set of poly-lines simultaneously, while respecting and preserving their topological relationships. The method introduced in this section is based on this approach and adopts some key ideas described by (28) , (96). Solution proposed here also extends some thoughts, we have first published in (11).

A brief characteristic of existing approaches. The solution in (85) is based on algorithm of (141) enhanced by the possibility to moderate the simplification process by the semantic influence of the map ontology, and removed the dependency of the simplification result on the order of the poly-lines input, which is an inherent property in (8).

Meijers in (96) further contributed by employing the concept of unconnected graph of poly-lines, and addressed the merge and split operations for map generalization, which often cause problems during the simplification. Meijers' method (96) identifies poly-lines influenced by a given simplification with help of an auxiliary kd-tree data structure, which facilitates fast access to the influenced poly-lines. In contrast, (28) avoids the overhead associated with the management of auxiliary data structures by encoding the poly-lines neighborhood relationships using unconstrained edges from the underlying triangulation.

Since Delaunay triangulation is the basic concept in our surface representation, the method presented in (28) for testing of relationships between poly-lines can be adapted. However, in order to avoid the re-triangulation of the approximate set of poly-lines after removal of every point, a list of points that block the point removal is maintained, similar to the (96) solution.

Removable points. The test, whether the removal of a point causes any new intersection, uses constrained Delaunay triangulation (CDT). The CDT is constructed against the tangent plane of the concerned GIG cell's centroid. It has the footprint poly-lines as input, and edges of the triangulation provide information about connected points. The removal condition test employs two lists of points connected to the candidate for removal q. The first list S starts with point p and proceeds counter-clockwise around q ending with point r. The complementary second list T starts with r and proceeds counter-clockwise to p.

Removal of q results in a new segment $\overline{p,r}$ without any new intersections with other poly-lines only if all $s_i \in S$ hold the same topological position relatively to $\overline{p,r}$ and all $t_i \in T$ have the same but topologically opposite position to $\overline{p,r}$. The left-or-right topological position of an arbitrary point v relatively to segment $\overline{p,r}$ can be expressed by a determinant

$$det(v) = \begin{vmatrix} x_r - x_p, & y_r - y_p \\ x_v - x_p, & y_v - y_p \end{vmatrix}, \tag{4.5}$$

where x, y are Cartesian coordinates of the CDT plane. Then the non-intersecting removal condition (as in (28)) is satisfied when

$$det(s_i) > 0 \quad \wedge \quad det(t_i) < 0, \quad or \tag{4.6}$$

$$det(s_i) < 0 \quad \wedge \quad det(t_i) > 0. \tag{4.7}$$

4.4.3 Simultaneous simplification of poly-lines for the multi-resolution environment

The main aim of the method presented in this section is to associate every footprint's point only with a page on appropriate LOD, which means coupling the point with corresponding GIG cell from one appropriate GIG level. The basic idea is to gradually decimate the geometry of footprints.

To simplify the preconditions and to facilitate addition of independent features, multiple points at one location, intersecting and overlapping poly-lines were detected and processed in Algorithm 1. Moreover, the weights of distinct points were adjusted accordingly within Algorithm 1. In this sense, the outcome of the Algorithm 1 represented in *vertex_list* and *edge_list* provides valid input data to the decimation procedure, which creates a multiple LOD database.

The process of assigning the portion of footprints' geometry, which is relevant only to certain LOD, to such particular LOD, is closely related to the paging mechanism.

Levels of detail. Index cells of GIG are employed to delimit the partitions of space that correspond to each LOD in the multiple LOD environment. The required number of LODs and the required spatial extent to be covered by the GIG cells of each LOD is application dependent. Such demand determines the set Z of LODs L

$$Z = \{L^1, ..., L^l\} |\ Z \neq \emptyset \wedge c^i \in L^i \tag{4.8}$$

where L^1 corresponds to the finest LOD, L^l to the coarsest LOD, see Figure 3.2 (left).

A threshold ϵ is related to every L. It provides a sequence of thresholds $\epsilon_1 < \epsilon_2 < ... < \epsilon_{l-1}$, the limiting values, which restrict the possible simplification for $l-1$ coarser LODs.

Mapping the footprint. The GIG methods can be used in two ways. First as a hash function, which assigns an arbitrary point $p(x, y, z)$ to a unique GIG cell c based on the proximity to the centroid, cf. 3.6. Second as a paging mechanism, which supports access to the persistent data. The multiple LOD database creation algorithm employs the hash function to map every input poly-line onto GIG cells at all relevant GIG levels. The nearest-centroid function NC is used for this purpose, see Equation 3.6. The mapping is denoted by function κ and reads:

$$\begin{aligned} \kappa : P \times \{NC^1, ..., NC^l\} \quad &\rightarrow \quad \{(p_1, \{c_j^1, ..., c_k^l\}), ..., (p_n, \{c_o^1, ..., c_p^l\})\} \\ &\quad\; c^1 \in NC^1 \wedge c^l \in NC^l. \end{aligned} \tag{4.9}$$

The function κ leaves every point associated with l GIG cells, one for every LOD.

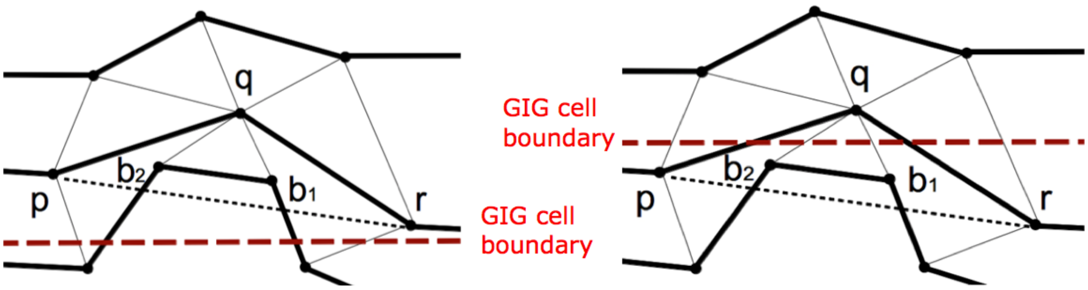

Figure 4.4: The geometry of the point q neighborhood - the point q removal results into the new $\overline{p,r}$ segment, which causes new intersections. Therefore, point q must be preserved to avoid the change of topology. However, in case the blocking points b_1 and b_2 are simplified, it also allow for removal of q in the first (left) case. In the second variant (right), the blockers b_1 and b_2 lies in the GIG cell different then the point q. Therefore, in the (right) case the point q will always be preserved to prevent from introduction of the topological error, when the cell with q is on coarser LOD. The boundary between GIG cells is depicted by the red fine dashed line.

Algorithm. The task of the algorithm is to associate every input point p with a cell c at just one GIG level L^i. There is a constraining requirement for the association. Given a set Z of LODs L, topological relationships between poly-lines P will be preserved in the finally reconstructed footprint-enhanced topographic surface. It will be

Algorithm 2 Decimation

INPUT: Adjusted footprints

OUTPUT: Multiple LOD database of footprints

Map the input upon the GIG indexing structure as in κ function.

for all $i = 1, ..., l-1$ **do**
    for all c^i **do**
        CDT($vertex\_list, edge\_list$) $\leftarrow \{\mathrm{NN}(C \in c) \cup \mathrm{NC}(C \in c)\}$ ▷ Generate triangulation with P as constraints for spatial domain resulting from the union of NC and NN query on the centroid C of c
        for all $p \in c$ **do**
            if $\omega(p) < \epsilon_i$ **then**
                if p is not blocked **then** ▷ Decision based on equations 4.6 and 4.7
                    add p to *Remove*
                else
                    for all u blocks p **do**
                        add u to $p.blocked\_by$
                        add p to $u.blocks$
        while *Remove* is not empty **do**
            $p \leftarrow Remove.pop$
            for all $u \in p.blocks$ **do**
                remove p from $u.blocked\_by$
                if $u.blocked\_by$ is empty **then**
                    add u to *Remove*
            add p to records $c^i.\rho$ of all $P : p \in P$
        Update $c^i.\rho$ with points resulting from intersection of P and GIG cell boundary edges
    for all $p \in L^i$ **do**
        remove p from $vertex\_list$
        update $edge\_list$
for all $c^l \leftarrow \kappa$ **do**
    build records $c^l.\rho$ from $vertex\_list$ ▷ Remaining points form the coarsest level l
return database of all $c.\rho$ of all levels $L \in Z$ ▷ Output

synthesized 4.5 for a neighborhood of an arbitrary query position d on the basis of data associated with GIG cells, which delimit the neighborhood of d at all L LODs, 4.3.

Therefore, the awareness of, what GIG levels form the paging structure, is important, because the cells of the index also deliniate the boundaries between different LODs. In the course of the creation of pages by means of geometry simplification algorithm, particular attention must be paid to the preservation of topological consistency at the LOD boundaries.

The schema of the algorithm is illustrated by Algorithm 2. The algorithm receives all adjusted poly-lines, the output of Algorithm 1, as an input. The mapping of footprints onto the indexing structure, as described by κ function 4.9, is applied and as the result, every input point is associated with matching GIG cells. Consequently, also the list of cells to be processed is at disposal. The processing of all cells c starts at the finest LOD and gradually reaches the coarser LODs. At given LOD i the procedure gradually walks through all cells c^i. For a currently processed cell c the NC query (Equation 3.6) is used to retrieve footprints to be simplified within this step. Furthermore, the NN query (Equation 3.7) retrieves footprints from all neighboring cells of c.

The cells, which correspond to the result of the query, also determine the domain of constraint Delaunay triangulation. It is built to encode the topological relationships between footprints that act as constraints in CDT. Footprints analysis in the previous section resulted in a set of unique points and a set of unique edges, which meet only at their endpoints. With this prerequisite it is always possible to construct CDT that contains all points and every edge of features' footprints as constraints in the triangulation. The constraints in the triangulation result from the orthogonal projection of P to a tangent plane taken at the centroid of processed cell.

CDT is also the most expensive procedure of the Algorithm 2 with achievable $O(n\,log(n))$ time complexity, where n is the number of points in $vertex\_list$.

The algorithm consists of two main loops.

The first loop (lines 4-11) determines points, which should be removed based on their importance ω and the threshold ϵ_i of a given LOD, while preserving topological consistency.

To avoid the creation of new intersections between footprints, the decision about removal must be done in context with the neighboring poly-lines, see Figure 4.4. That includes self-intersecting poly-line scenario. The neighboring poly-lines, or their points respectively, can be of two kinds.

First, the points in the same GIG cell as p. Then, such a point u can block the removal of p, however, it can also be a candidate for a removal. If the candidate for removal u is be removed, it enables the removal of p. Consequently, the removal of vertices is evaluated with respect to the simplified version of interfering poly-line.

Second, the blocking point lies in the neighboring cell. In this case, the decision about removal must be done with respect to the non-simplified version, because the neighboring cell may be visualized with finer resolution.

With every blocked p, two lists are associated. First, the list of vertices, which p is *blocked by*. And second, the list of vertices, which p *blocks*. To determine, whether the removal of point p is blocked, the geometrical criterion based on the point p neighborhood defined in equations 4.6 and 4.7 is applied. The points determined to be removed from given LOD are placed to *Remove* list.

In every iteration of the second main loop (lines 12-18), the first element is taken from the *Remove* list and relevant *blocked_by* and *blocks* lists are updated. Removal of a point can empty *blocked_by* lists of blocked vertices allowing for another point removal. Such a point is added to the *Remove* list. The pieces of poly-line's geometry relevant for given LOD are stored as records 4.4, which are associated with the GIG cell.

Finally, before the algorithm proceeds with subsequent level of detail, vertices in *Remove* list are removed from *vertex_list* and *edge_list* is updated.

Topological consistency. The outcome of the algorithm is the multiple LOD database of footprints' poly-lines. Its character guarantees, that the synthesis, performed on the basis of such a database, will for any query position d preserve the connectivity between poly-lines, all existing intersections will be kept fixed and no new intersections will be introduced.

Some other cases of topological consistency, however, should be discussed. Most authors of poly-line simplification algorithms within traditional cartography prevent from a complete removal of a poly-line. Usually the endpoints are never decimated. This is typically achieved by setting their weights to a maximum value. The reason for this approach is given by the fact that the generalization process is composed of multiple operations and a feature removal is a distinct operation with a distinct meaning. The rules in Table 4.1 are set in this respect.

However, for some applications it may be too strict, as the semantic importance of endpoints is not crucial, e.g. the stream network, or network of terrain break-lines representing important features of the terrain itself (ridge lines, valley lines).

The introduced method can be easily adjusted to allow for simplification of end-points. It can be achieved by correcting the first rule in Table 4.1 in such a fashion, that the weights of all first and last points of poly-lines will not be set to a maximum value. Only those points p, in which two or more endpoints of two or more distinct poly-lines meet, will have the maximum weight assigned. This adjustment of the first rule in Table 4.1 allows for a removal of an isolated poly-line, or its simplification from a lose end.

Two other aspects, that concern topological consistency of the poly-line simplification algorithm, are related to the areal collapse. Using the notation from 4.4.2 and having three consecutive points p, q, r, the removal of q results in a new segment $\overline{p, r}$. As characterized by Figure 4.5, the $\overline{p, r}$ segment may already exist.

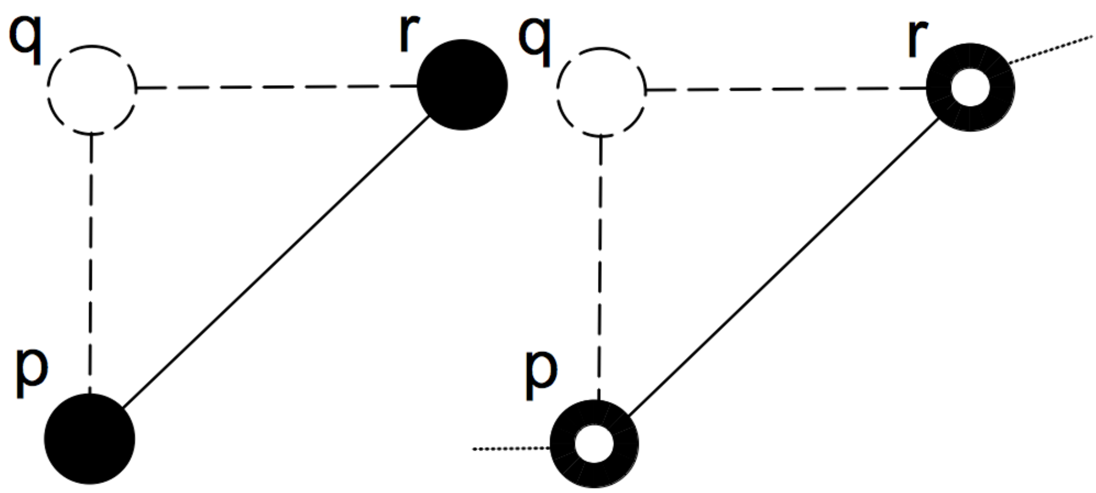

Figure 4.5: Areal collapse - (left) closed poly-line and (right) two neighboring poly-lines.

First, it maybe an edge of the same poly-line. In that case, it is a closed poly-line (having identical end-points) scenario. Second, it maybe an edge of another poly-line. In both cases the geometrical interpretation is a collapse of an area delimited by $\overline{p, r}$,

$\overline{p,q}$ and $\overline{q,r}$ edges, which is allowed by the simplification procedure and considered by an *edge_list* update step. As well, the collapse to a single point of an isolated looped poly-line is enabled.

4.5 Synthesis of topographic surface with features

At run-time, the reconstruction of feature-based topographic surface is based on the query on observer position, which returns only the target GIG cells on multiple levels with associated geometry. This procedural solution to the topographic surface with poly-lines reconstruction and visualization allows for progressive refinement of geometric detail only where needed. The number of LOD can be set arbitrarily and with respect to the needs of the application.

In other words, the solution is based on a discrete LOD approach. As characterized in Figure 4.3 by the example of two LODs, the GIG cells of coarser LOD have apparently bigger spatial coverage then the those of finer LOD. The geometry associated with a coarser LOD is capable to delineate the most important variations of the surface shape. The data associated with smaller cells at the finest LOD provide an additional detail to the shape.

This is an important characteristic of this LOD method, since the data from all coarser LODs are needed to properly reconstruct the finest detail. The fact, that the data from coarser LODs are reused on a finer level introduces the dependency of finer levels on the coarser ones. However, reusing the geometry prevents from a data redundancy.

4.5.1 Reconstruction of poly-lines

When the position d is associated with the viewer's position, a paging mechanism is obtained, which provides records ρ from relevant GIG cells at multiple LODs. The paging mechanism can utilize several types of GIG queries (NC, NN, 4NN), cf. (77), according to the visualization needs of the application.

The reconstruction of footprints on the basis of records retrieved from the database starts from the coarsest LOD. For each LOD, a set of all sequences of points, corresponding to poly-line P with certain IDP, is retrieved from queried database pages.

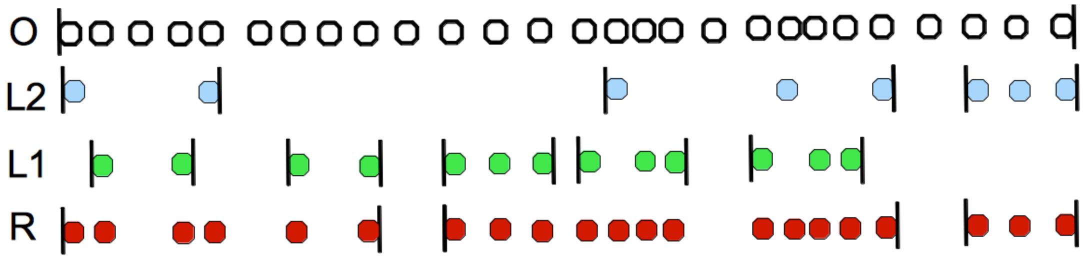

Figure 4.6: Poly-line reconstruction - illustrated through an example of fetched sequences of geometries for two LODs (L1, L2) and the reconstructed (R) geometry of poly-line resulting from the original (O) full-resolution representation.

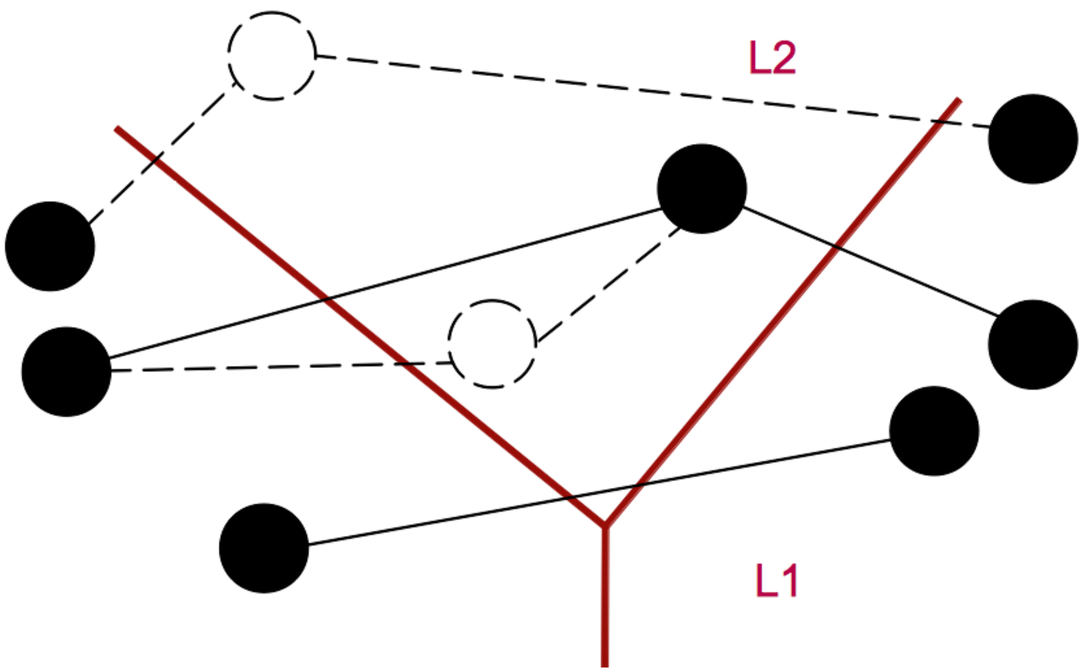

Figure 4.7: Handling poly-line reconstruction on GIG boundaries - the sequences of poly-line at the bottom were joined, since they directly succeed. As well, the sequences of the poly-line in the middle were merged, because the intermediate point from the coarser *L*2 level was retrieved from the database. Unlike the case of the the poly-line at the top, where no intermediate point can be found, therefore the poly-line consists of two detached sequences.

Within each sequence the points are ordered (according to order in the original poly-line) and the sequences contained in the record ρ_{cid} are ordered according to the first element. First, as the algorithm retrieves records from multiple pages on the given LOD (e.g. 4NN query), the sequences, which belongs to the same poly-line identifier IDP, are sorted according to the first element.

Further, the procedure goes through all the sequences and if the endpoint of a sequence is an immediate predecessor to the start point of the next sequence, they are joined. In other words, the join of two sequences is carried out, if there are no intervening points in the original poly-line.

When all LODs are processed in a described manner, the merging of sequences across the LODs commences. The process merges those sequences from different LODs, that

1. overlap according to the order of some points of the poly-line,
2. are succeeded with a sequence on a coarser level.

In other words, the sequences detached on one LOD remain separated after merging, if no intermediate point from different LOD was retrieved from the database. These characteristics of the reconstruction are visually presented in Figures 4.6 and 4.7. The output of the merging procedure is a set of poly-lines, that act as constraints to the next step of topographic surface reconstruction, the constrained Delaunay triangulation.

Algorithm 3 Poly-line reconstruction

INPUT: Records ρ retrieved from GIG-based database
OUTPUT: Set of poly-lines

for all $i = l, ..., 1$ **do**
  for all P associated with cells c answering the query on observer position on processed level **do**
    Get all ρ with the same IDP and sort all their sequences S according to the order of first element
    for all S **do**
      if $S.end\_point.order + 1 = S.next.start\_point.order$ **then**
        Join S and $S.next$
Merge overlapping or subsequent sequences S from all LODs with corresponding IDP
return Set of poly-lines formed by merged sequences

Observations on the properties of the synthesis. The paging mechanism retrieves the data from only certain pages (GIG cells). The information about data from pages beyond the relevant neighborhood is missing. That concerns the course of a poly-line beyond the target indexing cells as well.

The unknown course of a poly-line may lead to several situations during the synthesis of topographic surface with features, which are depicted in Figure 4.8. Among them, the unclear way of reconstruction of the poly-line course can introduce unfinished lines on the limits of GIG cells. The ϵ threshold, which acts as a limit to allowed deviation from the original course of poly-line and which guarantees certain accuracy, can also be exceeded on the boundaries of cells, if situations depicted by the Figure 4.9 occur.

These situations at the limits of given level of detail are not problematic generally for all kinds of data, rather they can be understood as a smooth transition to a coarser LOD. However, for some applications the named situations may cause problems.

Therefore, for selected applications some special treatment maybe be desired. There are three possible general approaches to the remedy of the situations described in Figures 4.8 and 4.9. First, to fetch the missing data from neighboring cells and reconstruct the course of poly-lines on their basis. Retrieving the data from additional pages at run-time would, however, severely compromise the very basic purpose of paging and could overload the system due to the character of big data.

The second option is to shift the responsibility upon the data object. Which means enforcing the desired behavior on the object's design level, e.g. by creating the proper geometry and setting of the geometry's weights.

The described behavior of poly-lines on the index cells boundaries can also be treated by means of the method itself. It can be achieved relatively easily.

In the equation 4.9, the κ function was introduced, which mapped every input poly-line onto the GIG cells and associated every point with corresponding cell. The same mechanism, even from the implementation point of view, can be extended to identify intersection points of GIG cells boundaries with poly-line edges. As is illustrated in 4.9, the artifacts of poly-line reconstruction on the index cells boundaries can be avoided, if the intersection points are added to the representation of poly-line's geometry and stored within poly-line records ρ in all (usually two) concerned neighboring cells.

This kind of solution however introduces a completely new geometry to the representation of the poly-line, which is not desirable in general. Furthermore, the data redundancy is brought in, since the inserted intersection points were not deemed necessary for the description of the original object. The data redundancy is further enhanced, as such intersection points must be stored in all concerned cells. Obviously, the addition of new points goes against the very objective of simplification. Lastly, the intersection points cannot be simplified on coarser level of detail, to be able to fulfill the role of preventing the ϵ-based accuracy violation or introduction of undesirable edges 4.8.

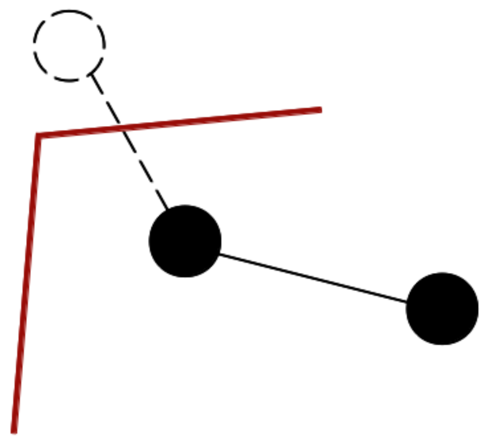

Figure 4.8: Handling the course of poly-line at the boundaries of the GIG cell - introduction of a gap, when the poly-line does not end on the boundary edge (GIG boundary marked by the red line, points simplified from the original poly-line, which were not fetched from the database, are drawn as empty circle).

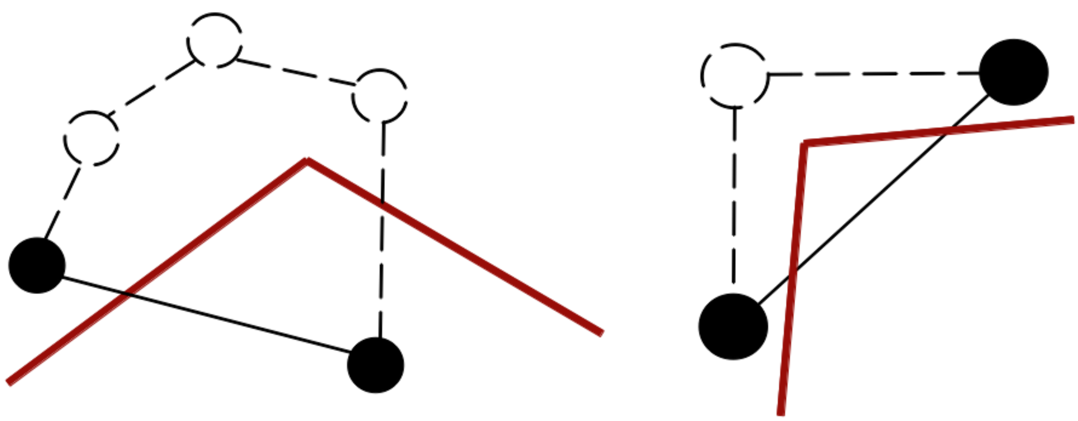

Figure 4.9: Handling the course of poly-line at the boundaries of the GIG cell - breaking of the ϵ-based accuracy of the poly-line's course, at the boundaries of GIG cells.

Nevertheless, such adjustment of the simplification algorithm 2 and its implementation are quite straightforward. It can be achieved by inserting the GIG cell boundaries geometry into the triangulation (line 3 of Algorithm 2) as constraints. Consequently, the geometry from neighboring cells is not needed for processing of given GIG cell, since the GIG cell edges prohibit any undesirable simplification across them.

The intersection point, which the poly-line reached the boundary of a GIG cell at, is associated with poly-line record in all affected cells. This fact can be utilized in the adjustment of the geometry reconstruction procedure. Namely, in Algorithm 3 the *if* condition on line 5 should allow joining of the sequences, if the endpoint of a sequence is identical to the start point of the next sequence.

4.5.2 CDT-based reconstruction of the topographic surface

The resulting geometry of the feature-based topographic surface is reconstructed by means of constrained Delaunay triangulation. The view synthesis of the 3D scene is accomplished by the run-time control application, which provides the viewer component and interactive navigation.

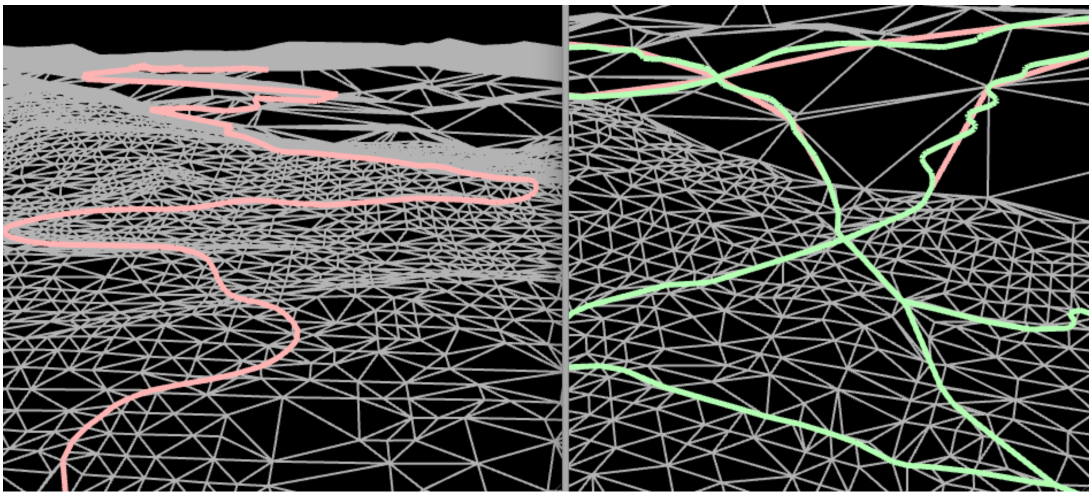

Figure 4.10: Topographic surface reconstructed by means of CDT. - OSM railroads data layer is associated with SRTM elevation data on multiple LOD, both simplified in distant areas (left). Closer view on the threshold between two LOD with the OSM road network visualized also by the green color to illustrate the original course of the road centerline footprints in comparison to the simplified ones (red color)(right). The figure was first used in (11).

The input for the CDT varies according to the observer's position, cf. 4.10. It consists of a set of poly-lines retrieved from the multiple LOD database and it is synthesized for given position of the observer, as described in previous subsection. The second input of the triangulation is a set of all elevation points associated with GIG cells of interest, cf. 3.5. The clusters of elevation points were modeled and persisted in compliance with definition 3.8.

There is no need to advocate for the usefulness of triangulations. Regular triangulations turn out to be an elegant solution for a shape reconstruction when dealing with non-uniform samples. Throughout the history, many procedures for computation of Delaunay and constrained Delaunay triangulations were developed. In this work, the two-dimensional triangulation package in CGAL (63) is employed.

Triangulations in CGAL are built around on-line insertion of vertices, therefore the main algorithmic issue is the point location (117) in the triangulation. A concise review of methods tested by CGAL developers can be found in (63), further enhancements on incremental triangulations in (23) and method improvement on points deletion was documented in (24).

The technique of the applied triangulation incrementally inserts points and constraints and locally re-triangulates. It is efficient for both addition and removal of points and constraints from the triangulation, which is suitable for our method. If auxiliary data structures are maintained by triangulation to optimize point location, the incremental algorithm achieves the expected time of O(n log n) (23).

The triangulation plane changes according to the observer position. The chosen plane corresponds to the concept introduced by Kolar (77) and, as was already mentioned earlier, is defined as "tangent plane to a unit sphere taken at the centroid of the cell from the coarsest LOD".

4.6 Closing remarks

This chapter formally described the foundations of the method and provided the theoretical framework for populating large multi-resolution terrains with different kinds of features. That was achieved by scrutinizing following research questions:

4. How to design a conceptual data model that describes the feature-based topography in the multiple LOD environment?

5. How can be the valid input data created by as much automatic manner as possible?

6. How can be the database of features created so that the topological relations between features will be preserved in the multiple LOD environment reconstructed on its grounds?

7. How can be the topographic surface with features synthesized for given observer position?

The the geo-spatial information systems with global coverage, like the digital Earth, are relatively new technologies. The approaches from multiple research domains meet in the design of such a platform for sharing of spatial data. Platform, that is supposed to be a universal tool for sharing of various types of data of various origin.

During the design of the data representation of the feature-enhanced topographic surface, it has become clear, that existing works that deal with generalization of vector data lay down relatively strict requirements on the character of the input - no intersections, no self-intersections, no overlaps or multipart polygons are allowed as an input for these solutions. Such preconditions, however, are relatively strict and prevent an easy addition of new features to the topographic surface. The creation of valid geometry for every new feature type causes an extra overhead, is less practical, and a solution for given feature types might be complex or even unknown. To simplify the preconditions and to facilitate addition of independent features, the method for analysis of footprints was proposed. It deals with possible geometrical inconsistencies by keeping track of intersecting and overlapping poly-lines.

Moreover, the introduced analysis of footprints provides means for processing the attributes of spatial objects, which encodes the required behavior or functionality of the object in the system and during its interaction with other objects. This was manifested through the example of weights that mark the importance of the constituting geometries of object's footprint. The analysis of footprint enables the adjustment of such weights. The outcome of such adjustment depends on what objects enter the analysis. Such analysis guarantee, that afterwards such objects will act in the system in mutual respect, e.g. the original mutual boundaries of the objects will remain mutual on coarser LODs.

Secondly, the concept of footprints was extended, so as to be applicable in the multi-resolution environment. For this purposes, the primary geometric structure of the footprint and the geometry simplification algorithm were proposed. The single record ρ always holds a subset of points of the original footprint's representation, which is relevant only to a given LOD. It is upon the generalization algorithm to determine the subset of points for the record associated with each LOD.

In order to assign the constituent points to a distinct LOD, the simultaneous simplification algorithm was designed, based on the method earlier described in (85) and further extended by maintaining the list of points that block the point removal similar to (96). Presented algorithm, in contradistinction to the Meijers' solution (96), is not limited to a simplification of valid planar partition. Or, unlike the approach of Kulik (85), it is not limited to a connected graph. Together with the adjustment of weights introduced in the procedure for footprint analysis, presented approach is applicable to an arbitrary set of poly-lines. The main improvement over existing methods lies in the ability to produce topologically consistent geometry for an environment (graphic scene), which consists of multiple LODs.

The procedural aspect of the solution presented across this work presented another shift of current approaches. This shift is rooted in the procedural character of the topographic surface reconstruction, which the primary geometric structure was tailored to.

When current solutions to multi-scale databases are examined, they obviously employ sophisticated data structures to maintain the multi-scale representations. For example, the state-of-the-art solution like tGAP (59, 139), which Meijers' solution (97) utilized to deal with vario-scale representations of boundaries of planar partitions. Although (97) stressed the aspect of minimal data redundancy, a constant trade off between storage and calculations must have been considered, i. e. what information should be stored implicitly and what derived procedurally from the stored information. The auxiliary data structures, that keep the informations about the topology, especially about the topological correlation between features on different scales, were shown as very expensive for data storage. Meijers (97) confirmed, that the more the geometry was simplified, the more expensive was its storage. It was also shown, that the resulting number of stored elements not only depends on the schema of data structure, but also on the chosen simplification strategy. Another complication of solutions dealing with topology is related to the distributed data management. Yet another auxiliary structures are needed to maintain the relations between distributed pages.

The primary geometric structure in this work was designed for the procedural way of topographic surface reconstruction. In this way, the redundancy in data storage was entirely avoided. The procedure for run-time reconstruction of poly-line's geometry relevant only to given query on observer's position d was proposed. As a consequence, an effective construction of the footprint-enhanced topographic surface with multiple LOD can be synthesized for a neighborhood of a position d. When the position d is associated with the viewer's position, the paging mechanism is obtained, which always provides data for a relevant neighborhood at multiple LODs. This supports data access but also visualization for possibly huge databases, which cannot be entirely loaded into memory and visualized, which can reside on a network, can be distributed, and/or decentralized.

5

Terrain-aware generalization of features

This chapter provides an insight into the relationships between the bare Earth model and the footprints of spatial objects on multiple levels of detail. While in the previous sections the means for guiding the geometry changes in mutual respect among features were studied, the following text will pay attention to how the change of geometry representation of a spatial object affects the model of terrain and, the other way around, how the terrain characteristics influence the course of simplification of spatial objects.

First, in section 5.1 the ways of multiple LOD terrain creation are re-examined. Second, two approaches to geometry simplification of objects are introduced, which take into consideration the underlying terrain model and assess the suitability of these approaches. In section 5.2, the criteria are defined, which evaluate the topological change that is introduced by the simplification of the feature into the feature-terrain relation. Afterwards, the section 5.3 pays attention to the morphological structure of the terrain, proposes a method for a retrieval of selected structural features of the terrain and shows their application for multi-resolution topography.

5.1 The multiple LOD terrain

In the foregoing chapters, the bare Earth was perceived as a global surface, which the other spatial objects are referenced to. The method of topographic surface reconstruction utilized the fact, that the points of terrain, which are linked with coarser level, can be deemed as more important for the shape description, than the points on a finer level. It is based on a very same principle as the proposed method for spatial objects.

The knowledge of the method used for multiple LOD terrain database creation is important. It predetermines the geometry of terrain, which the spatial features will interact on various LODs with.

Prior to examining the means of terrain simplification, which leads to multiple LOD terrain database creation, several ways to evaluate the terrain approximation quality, should be reviewed. Different applications have different requirements, therefore, the optimal measure for the similarity between the simplified model and the original model may also vary.

5.1.1 Assessment of the quality of simplified models

Several approaches can be applied to evaluate the suitability of a simplified model. The measures of the similarity between the simplified model and the original model should be chosen according to the desired application of the model.

Geometric errors of simplified model. Two norm and infinity norm are vector norms, whose errors can be utilized for evaluation of fidelity result between a model and approximations of the model. Following error metrics were used in an extensive evaluation of solutions to the terrain approximation performed by Garland and Heckbert in (37).

On the basis of the two-norm (also known as the L2-norm, mean-square norm, or least-squares norm) the L2 error between two models represented by two n-vectors $\overrightarrow{u}$ and $\overrightarrow{v}$ can be defined as

$$\|\overrightarrow{u} - \overrightarrow{v}\|_2 = \sqrt{\sum_{i=1}^{n}(u_i - v_i)^2}. \tag{5.1}$$

The root mean square or RMS error is the L2 error divided by $\sqrt{n}$.

The infinity norm (also known as the L_∞-norm, max norm, or uniform norm) can be used for the definition of L_∞- error between two n-vectors $\overrightarrow{u}$ and $\overrightarrow{v}$ that reads:

$$\|\overrightarrow{u} - \overrightarrow{v}\|_\infty = max_{i=1}^{n}\{|u_i - v_i|\}. \tag{5.2}$$

Modeling optimizations with respect to the $L2$ and L_∞ metrics are termed as least squares and minimax optimizations.

Perceptual similarity of simplified model. If the objective of application emphasizes the visualization aspect, then the appearance on the screen becomes the key measure. In this case, the straightforward way to measure the similarity between original and simplified model is the comparison of intensity between corresponding rendered pixels. Let us have an $m \times n$ image rendered on the basis of original model I_o, the image rendered on the basis of simplified model I_s and the value (RGB vectors) of a pixel $P(x, y)$ in image I. Then, the perceptual similarity measure between the two images can be written as:

$$\|I_o - I_s\| = \sqrt{\sum_u \sum_v (I_o(x,y) - I_s(x,y))^2}. \tag{5.3}$$

Such a measure is viewpoint dependent. Therefore, if the model is to be rendered from arbitrary location (the infinite number of points), the representative set of locations must be selected to run the assessment. The selection of such points is application dependent.

Visibility-based quality of simplified model. An example of the application-driven simplification measure is the visibility similarity. Visibility analysis over digital terrain models is distinctive for GIS domain. It answers the question, what can be seen by the observer from given location on top of the surface. Creation of simplified terrain models with respect to visibility properties holds similar aspects to perception similarity, which used to be applied in computer graphics domain. Especially if the assessment is concerned.

Having the terrain model T and its simplification T' and the points p and q on the surface of T and T'. The points p and q are said to be visible, if the entire line segment $\overline{p, q}$ occurs on or above the surface of T, or T' respectively. Let ξ be the set of pairs of points in IR^3 and τ the subset of points pairs $\tau \subseteq \xi$ such, that the visibility of those point pairs in τ is different. Then the visibility error E_{vis} reads:

$$E_{vis} = \frac{||\tau||}{||\xi||}, \tag{5.4}$$

where $||.||$ denotes the L2 norm. Having the complement of the set τ owing to ξ, which is denoted as τ', the visibility similarity with respect to ξ can be defined as $VS = ||\tau'||/||\xi||$.

5.1.2 Importance measures for terrain simplification

Having some means for a judgement of the produced model quality, what additional criteria shall be chosen to select the importance measure of model's geometrical primitives, which will be employed to guide the process of simplification?

Inclusion of various features like valley lines, peaks, ridge lines, water streams or roads into the simplification process of the terrain was the subject of several works (6, 33, 36, 70). This approach results into the optimization of terrain approximation with respect to these particular features. Taking into consideration the existence of selected features, however, one-sidedly effects the simplification outcome in favor of these features.

In the concept proposed in this thesis, the terrain model has the referential role; all other spatial objects are referenced to its surface. One of the key features of proposed system is the enhanced flexibility of the topographic surface representation on contact with arbitrary spatial features. Therefore, such measures of terrain geometry importance, which utilize an implicit knowledge based on only some features, must be dismissed.

The importance measure should be applicable to general height fields and produce generally good approximations. Owing to the high data volumes, the method should be also quite fast and easy to evaluate, since every repetition unduly increases the processing time.

Consequently, in one of our experiments, the results of Kolář in (77), which were achieved by the means of Garland and Heckbert's approach (37), were employed. In this approach, the importance of a terrain point $p(x, y)$ is measured as the difference between the actual value of the original surface point $OS(x, y)$ and the interpolated value on the approximated surface $TS(x, y)$

$$E_{loc}(x, y) = |OS(x, y) - TS(x, y)|. \tag{5.5}$$

This local error measure E_{loc} meets the requirement for a simple and fast evaluation. Also the local character is desirable, as terrains are high frequency surfaces, where the local characteristics are more important than the information about remote features. Garland and Heckbert in (37) also concluded, that the produced results are more accurate than of any other tested methods (global error, curvature error and product error measure). The simplification quality was assessed by the means of $L2$ and L_∞ errors, cf. Equations 5.1 and 5.2 . In order to ensure a rapid processing, a greedy insertion algorithm was also described by (37). The algorithm applies the refinement method, which for given LOD stops, when certain error threshold is reached. Resulting lists of points form distinctive LODs.

5.2 Analysis of feature vs. terrain relationship

The objective in this section is to present a measure, that would allow to guide the simplification of features with regard to terrain. The task can be defined as following. What error will be inflicted by a removal of a point from a poly-line? And, how should such error be measured?

In the previous chapter, the topology-preserving simplification method of footprints was introduced. Although the topology between poly-lines is preserved in the multiple LOD environment, the topology between geometrical primitives (points), that constitute the terrain model, and the footprint's primitives (poly-lines), is not guaranteed to be kept unchanged. Example of such situation is illustrated in Figure 5.1. The requirement to carefully maintain topology owing to arbitrary point of terrain would be very restrictive. For certain applications, for instance medical imaging, preserving the object topology can be essential. However, for applications based on multi-resolution rendering systems, strict maintaining of topology is a frequent limitation for the purposes of simplification.

Hence, the solution presented in this section provides the system user with means to decide, what is suitable for the particular application. Therefore, the criteria presented further should be deemed as an auxiliary tool for the decision on the progress of the geometry simplification procedure. The weights of points may be influenced during the database creation by the surrounding terrain in a similar manner, the occurrence of a feature influences the progress of simplification of another feature. Such approach does

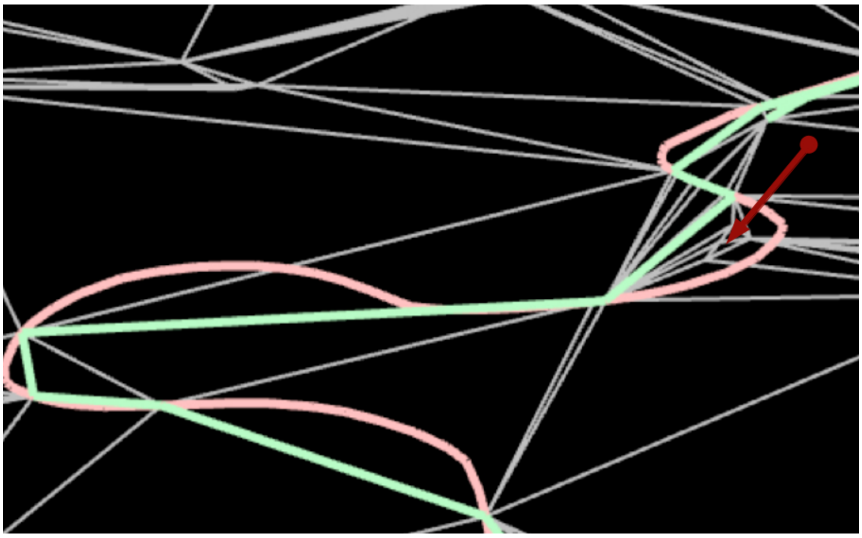

Figure 5.1: The change of topological position between feature's geometry and an elevation point. - The figure illustrates the original (red) footprint's geometry simplified (green), resulting in a changed orientation of a terrain points towards the poly-line (marked by red arrow).

not restrict the user (or spatial object designer) to a precomputed solution of a terrain model, which would be optimized for one distinct application.

5.2.1 Preliminary considerations

The straightforward approach to propose the error measure is to evaluate the change inflicted to the mesh as a result of the change of poly-line course, c.f. Figure 5.1. Nevertheless, although the change of the poly-line's course may be significant, simultaneously, it can theoretically cause no change to the mesh, as is illustrated in Figure 5.2 (left). However, the position of the simplified poly-line owing to the intervening elevation point will be changed. Even if the concerned points of poly-line and terrain were not identical, as in Figure 5.2 (left), but rather close, the introduced change of a mesh would be very small.

Therefore, in order to evaluate the importance of the change of poly-line's course owing to the terrain, first, the importance of terrain points, owing to which the poly-line position changes, must be deduced.

In the introduction to this section 5.2, the (37) approach for guiding the simplification of terrain was described and and it was reasoned, that the simplification of terrain

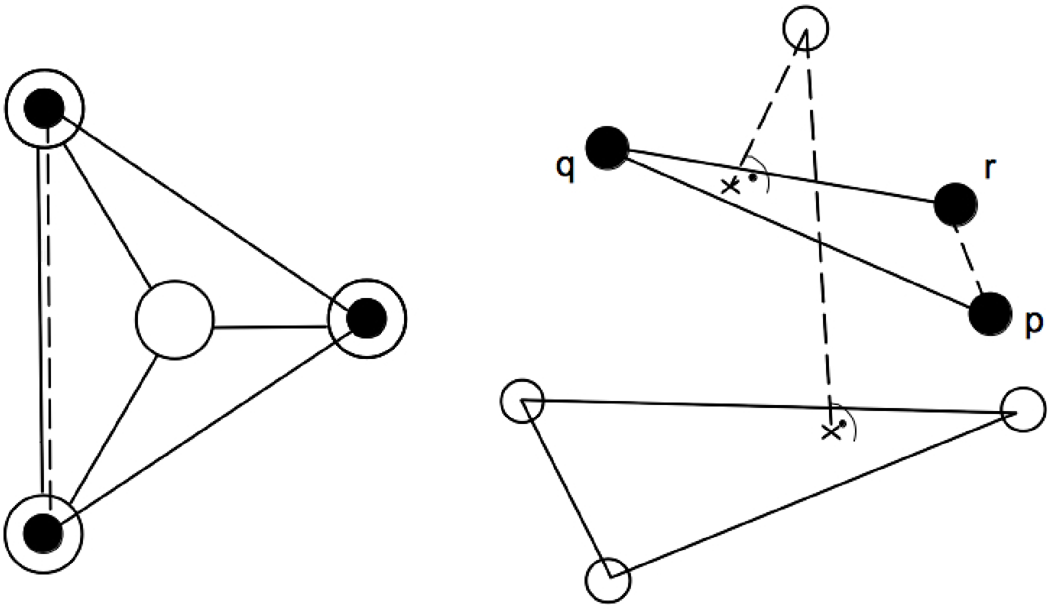

Figure 5.2: The spatial arrangement between terrain points and the geometry of a footprint - Bigger empty circles depict the elevation points, smaller full black the points of a footprint. The figure on the (left) illustrates the theoretical possibility, that the concerned points of a footprint are identical to the elevation points. Consequently, the simplification does not introduce any change to the mesh. The figure on the (right) describes the difference between two importance values of the elevation point. First, the importance inferred by means of (37) approach for the sake of terrain simplification (measured as the difference between the actual value of the original surface point and the interpolated value on the approximated surface). Second, the importance deduced from the distance of an elevation point measured from the triangle formed by three consecutive points of the poly-line.

should be generally applicable and not to a priori anticipate specific features of terrain or features related to terrain. This approach utilized the importance measure, which was assigned to every constituting point. Is such a measure directly applicable for the evaluation of topological change importance?

For Garland and Heckbert's approach (37), the triangle is a very basic element. Each triangle is responsible for the selection of a candidate point, the update of errors of the points when the triangle is newly created, and the maintenance of its importance measure.

Let us consider a poly-line P of a feature to be simplified. The removal of a point q from P results in a new segment $\overline{p, r}$. As a consequence, segments $\overline{p, r}$, $\overline{p, q}$, $\overline{q, r}$ form a triangle $\overline{p, q, r}$.

Furthermore, if the $\overline{p, q, r}$ triangle is compared with the triangle of the terrain approximation, owing to which the importance of the terrain points was derived by Garland and Heckbert's approach (37), it is obvious, that they may considerably differ in their orientation and distance to the points of terrain, c.f. Figure 5.2 (right).

Consequently, the importance value associated with terrain vertices during the terrain decimation procedure cannot be deemed as suitable.

5.2.2 The measure of topological change importance

As suggested on the lines above, simple and fast criteria for an evaluation of the importance of topological change between feature to be simplified and the terrain geometry, will be proposed. In order to determine the cost of a point q removal, which results in a new segment $\overline{p, r}$, the locality of the triangle structure will be exploited.

Orthogonal distance is defined as the perpendicular distance between a point and the plane of a triangle. We define the first measure of the topological change between a poly-line to be simplified and the terrain geometry as the highest orthogonal error among all the terrain points in the territory of $\overline{p, q, r}$

The maximum distance criterium μ is based on the orthogonal distance between a point and the plane of a triangle. μ is defined as the highest orthogonal distance among all the terrain points p' in the territory of $\overline{p, q, r}$ triangle T

$$\mu = \max_{i=1}^{n} D(p'_i, T), \tag{5.6}$$

where $D(p'_i, T)$ is an orthogonal distance of i-th point from the plane of triangle T, n is the number of points contained by the triangle. This criterium can be applied for example in the case of a road navigation. In such an application, sudden change of the road's position with respect to a significant terrain feature (like the hill) can be confusing for a user. Or, an inappropriate simplification of a road can result in a cut through a mountain ridge, causing unsuitable visual artifact, see Figures 6.11 or 6.13.

However, in some cases, the overall inflicted change should be factored in the decision, whether the poly-line's points is allowed to be removed. For this sake, the second criterium σ is defined as a root mean square of the orthogonal distances between terrain points p' in the territory of $\overline{p, q, r}$ triangle T and the plane of T, which reads:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n} D^2(p'_i, T)}{(n)}}, \tag{5.7}$$

where $D(p'_i, T)$ is an orthogonal distance of i-th point from the triangle's plane, n is the number of points contained by the triangle.

The alternative criterium, which takes all points in a triangle T and the area of T into consideration, is the volume criterium υ. The volumetric criteria used to be implemented in different ways. Several criteria were listed for example in (88).

In this work, it is proposed to reuse the σ criterium for the computation of the volumetric criterium. The volume criterium υ amounts to the value of root mean square of the orthogonal distances σ multiplied by the value of an area $a(T)$ of triangle T

$$\upsilon = \sigma \cdot a(T). \tag{5.8}$$

5.2.3 Application of the measure in the simplification procedure

The simultaneous simplification procedure that leads to the creation of multiple LOD database of footprint, has to be slightly adjusted, in order to take account of the measure of topological change. The procedure centers its attention on the evaluation of a point p removal, cf. Algoritm 2, line 6. The original assessment, whether the point p is allowed to be removed, is based on Equations 4.6 and 4.7.

In order to consider the terrain properties, this decision must be extended by incorporating the criteria of importance defined in Equations 5.6, 5.7, or 5.8.

To enable the computation of the importance of a change, the points that constitute the representation of terrain, must be counted in the decision-making. When the polylines are processed on LOD L^i (thus being simplified for the coarser detail L^{i+1}), all terrain points constituting L^{i+1} LOD are considered. That is achieved by the addition of these points into the CDT, cf. Algoritm 2, line 3. However, individual terrain points do not act as blockers. The point p removal, which results in a new segment $\overline{p, r}$, is blocked by the terrain points in the territory of $\overline{p, q, r}$, only if a threshold specified on the basis of criteria μ, σ or υ is exceeded.

5.3 Application-driven simplification of topographic surface with features

The method introduced in this section presents certain duality comparing to the approach presented above in the subsection 5.2.2.

The previous section answered the question, what is the effect of feature's geometry simplification on the resulting surface and how to measure it. In other words, the objects integrated with the terrain model were perceived as a cause of a topological dislocation. Which, of course, they are. Especially in cases, when the weights of the geometries of terrain and various classes of objects are not set in mutual respect.

The following text will focus on the opposite influence, which the features have on the terrain. Namely on the fact, that the delayed simplification of a feature generally helps to refine the shape of the topographic surface. This characteristic is inherent to footprint of any spatial object on the terrain, as long as its representation is more accurate than the one of underlying terrain. In other words, it also concerns the geometries, that represent some characteristic feature of the terrain itself, for instance peaks, saddles or ridge lines. Some methods for visibility-preserving simplification of a terrain models, like the one proposed in (6), utilize the observation, that the blocking of the view from certain point is attributable to the presence of ridges. Therefore, the preservation of the most eminent ridges gets the priority in such application.

Moreover, the approaches to mesh simplification, which allow to delay the simplification of selected regions of the mesh like (70) should stand as an inspiration. (70)

enhanced the mesh simplification algorithms based on iterative edge contractions developed earlier (36). It was achieved through the adaptive simplification of a model by controlling the order of contractions selected by the underlying algorithm. The order is manipulated by assigning a higher weights to user defined areas. This helps to delay the simplification of semantically important regions.

Such approach is applicable to terrain simplification. However, it produces such approximation of the terrain model, which was simplified adaptively only to certain specified features. Hence, it irrevocably preserves selected regions. As was already mentioned earlier, this one-sidedly effected approximation is not desirable for the purpose, the terrain has in our system.

In contrast, the approach presented in this work achieves similar outcome by integration of selected features with the terrain model. The weights of such a feature have to be set in accordance with the desired preservation on given LOD. Because the topographic surface is reconstructed only at run-time, it is upon the user, what feature classes will be retrieved from the database to constitute the result [1].

Let us inspect the possibilities of the integration of selected spatial objects with the terrain at run-time, in other words, selected only on demand of the user. The following consideration focuses on geometries, which represent unique characteristics of the terrain itself, including point features (e.g., maxima, minima, saddle points) as well as line features (e.g., ridges, valleys, coastlines or important terrain break-lines in general) represented at multiple LODs.

The role of these features in our method is of three kinds.

1. The first, and the essential one in context of this chapter, is the ability of a feature to prevent spatial objects geometries from changing their topological position owing to important parts of the terrain. This property is integral to the simultaneous simplification method introduced in previous chapter, see 4.4.3, as the geometry of a terrain feature can be added to the simplification process, acting like a block.

[1] Owing to the data volumes of terrain and spatial object classes having even a global spatial coverage, it is beyond current capabilities of computing systems to evaluate all potential topological relationships among arbitrary objects, adjusting weights accordingly and check for topological conflicts on coarser LODs, all at run-time. However, not all serialized (classes of) objects have to be retrieved from the database to be visualized on every user's demand.

2. The second role of terrain features is the ability to enhance the visual perception of the resulting surface model. While the terrain simplification algorithms employed in this work produce plausible results in most cases, especially at coarser LODs their approximations do not preserve the visual appearance of the original model very well. The perceptual importance is often conditioned by semantics of features, which may have been geometrically less important, therefore decimated by the general terrain simplification mechanism.

   According to user's demands, the detail can be increased by selection of features to be integrated with the referential terrain at run-time. An example of such can be the representation of ridge lines, since they have a potential to stabilize the shape of a surface. Even at the coarsest LOD, where the unwavering display of the scene's horizon is concerned. The ability to populate the terrain with distinct features on-demand also allows for increased decimation rate of the general version of terrain model.

3. The third role is related to analyses of the topographic surface. Analyses like navigation, signal transmission modeling, or the drainage conditions exploration demand increased preservation of some geometrical characteristics of the terrain. In fact, the visibility analysis requirements are similar to those on perceptual quality of the surface approximation.

The simplification of topographic surface with features, which is customized for distinctive application, and its reconstruction, consist of the following steps:

1. Computation of the network of break-lines and distinguishing the importance of distinct break-lines
2. Creation of the multiple LOD database of both the terrain break-lines and spatial objects' footprints
3. On-demand reconstruction of the surface.

5.3.1 Computing the terrain break-lines network

Break-lines define behavior of the surface in terms of smoothness and continuity. As their name implies, break-lines are linear features, which can be defined as lines, along which the tangent planes show a discontinuity [1] (10). Among break-lines, valley lines and ridge lines are especially important structure elements for high quality surface modeling. They correspond to such lines, where two adjacent planes with the opposite aspect connect, forming the valley, or the ridge respectively.

The detection of terrain break-lines already is a deeply studied problem in geoinformatics. This section focuses especially on the analysis of point cloud data, which is known for being able to provide accurate representation of selected terrain break-lines (valley lines and ridge lines). It is motivated by the need of detailed representation of these types of break-lines for the finest level of detail. Such representation can be eventually simplified for purposes of applications, that do not require such an accuracy, or generally on coarser LODs.

Nevertheless, the ability of algorithms implemented over the regular grid data structure to efficiently process large terrains, can become helpful in a preprocessing step. With respect to the overview of related methods in 2.5, point cloud analysis algorithms usually require a primal approximation of a sought break-line. Raster-based algorithms frequently suffer from a decreased detail of represented terrain shape as a consequence of rasterization. However, they retrieve the approximation of break-line in a computationally efficient way. In the experimental evaluation of the method in section 6.2, the raster-based method documented in (66) was adopted for the retrieval of the primal approximation [2].

The (66) method provides means to:

- detect and distinguish various kinds of terrain break-lines,

[1] In terms of differentiability classes, the surface has C^0 continuity, but not C^1 continuity.
The term differentiability class is used to classify functions due to the properties of their derivatives (133). The function f has the differentiability class C^k if the derivatives $f', f'', ..., f^k$ exist and are (with exception to f^k) continuous.

[2] It is based on the decomposition of the input DTM in a raster form into the series of elevation profiles. The cross-sections are conducted in all four major directions (rows and columns of a raster and both diagonals). By this mean, the elevation function is converted to a function of one variable. If the course of a terrain break-line is perpendicular to the cross-section, then, in case of the valley lines, their intersections correspond to local minima. Respectively, in case of the ridge lines, they correspond to local maxima. Generally, all break-lines can be detected as a sudden change in value of the first derivative of the elevation function. The appearance of a break-line in a profile is the less obvious the more its course deviates from the direction perpendicular to the profile. This fact is mitigated by utilizing four directions, which the profile is gained in.

- evaluate the importance of the detected break-line,
- efficiently process large terrain datasets.

5.3.1.1 Analysis based on the point cloud

The point cloud analysis proposed in this subsection is based on the algorithm described in (112) under the supervision of the author. The method was originally developed for the sake of accuracy improvement of the water stream course based on LiDAR data having ZABAGED [1] data layer as the first approximation. This work utilizes the same approach to increase the accuracy of valley lines and ridge lines, with the first approximation originating from the raster-based analysis.

The approach to break-lines modeling is based on the intersections of pairs of patches, which represent approximations of the tangent planes to the surface in the neighborhood of a break-line. The illustration of the concept could be seen in Figure 2.1. It utilizes the advantage of a relatively plain representation of a plane, that requires only a few parameters.

Existing approaches use the buffer of a fixed width around the approximation. For instance, (10) used an implicit knowledge about the approximation accuracy (1m) on each side from the break-line approximation. This value was a consequence of the expected error during the manual digitization from image data, which provided the approximation).

In the experiment 6.2, the usage of an implicit knowledge about the approximation accuracy was found inadequate, as this assumption is too strict for purpose of the proposed system and expected input data. Therefore, initial approximations with possibly higher (or varying) deviation from the actual course of a break-line needs to be allowed.

The procedure deals, by parts, with the input break-line approximation and surrounding slopes. The processed line is split into segments of given length, which partially overlap. For every segment, the algorithm determines one key point that forms the new, more accurate shape of break-line. The parameters of planes, which approximate the surface in a neighborhood of a break-line, were inferred by means of least

[1] Základní báze geografických dat České republiky - the fundamental geographic database of the territory of the Czech Republic maintained as a digital vector model by the Land Survey Office.

squares fitting to a point cloud subset. The principle of overlapping segments and planes, approximating the neighborhood of the break-line was illustrated in Figure 2.1.

Determination of the patch width. An iterative method was proposed for selection of points, that are most convenient for determination of plane parameters of the patches. To compute the parameters, all points within a patch are used. Because the position of a break-line is most influenced by the terrain points in its closest vicinity, the procedure starts with narrow width of the patch. However, the decreasing width increases the risk, that the positional deviation of the approximating line from its real location will be higher than the width itself. Such situation is demonstrated in Figure 5.3, where both patches contain points belonging to one slope only.

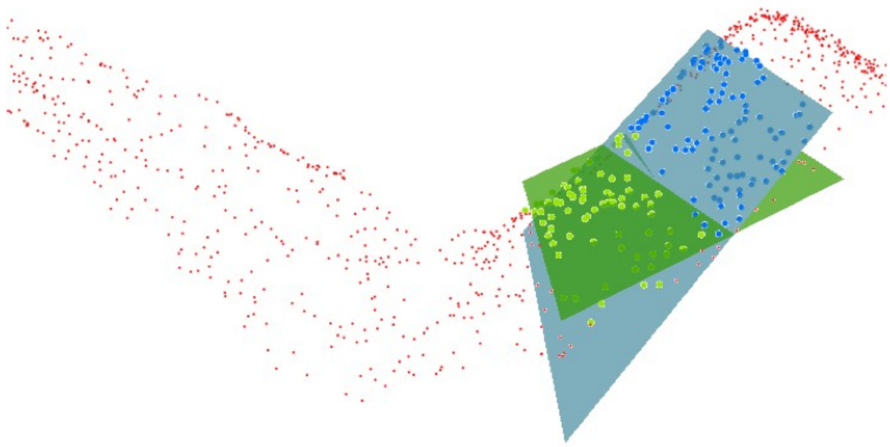

Figure 5.3: Determination of the patch width - an example of a higher distance of the actual course of a valley-line from its primal approximation, than the value of the patch width. Figure adopted from (112).

The method is based on the iterative extension of the patches' widths. The final extent of a patch used to determine the points, on which basis the plane parameters are computed, is discovered adaptively to the accuracy of the original approximation. Moreover, also the weights of points are set adaptively to required width extension.

For the sake of the decision, whether the planes approximate the surface around the sought line sufficiently, the angle between these two planes is evaluated. The observation, that the terrain shape in a direction perpendicular to the direction of a break-line is supposed to have concave (convex) character, in case of valley line (ridge line), is counted in the decision. The direction vector measured perpendicularly to the line of planes intersection can have either negative or positive value in a z direction. In case of valley line, both direction vectors should have positive value, in case of ridge line, the negative values should be found.

If this condition is not fulfilled, it is obvious that the plane(s) do not approximate the searched line well. As a consequence, the concerned patch is extended. The angle between the tangent vectors to the approximation planes is used as one of the algorithm's stop criteria. The process terminates, when acceptable angle is reached. The stop value is set depending on the terrain character (steepness of the slopes in processed area). If the angle is lower than given threshold, the patches are further extended. In the implementation 6.2, the width is multiplied by a factor of 1.5.

Moreover, the limit to the maximum possible width of the patch and the number of iterations was set. It is necessary especially in such cases, when the original break-line approximation is placed with such an error, that the valley lines or ridge lines from an entirely different, neighboring valley are detected.

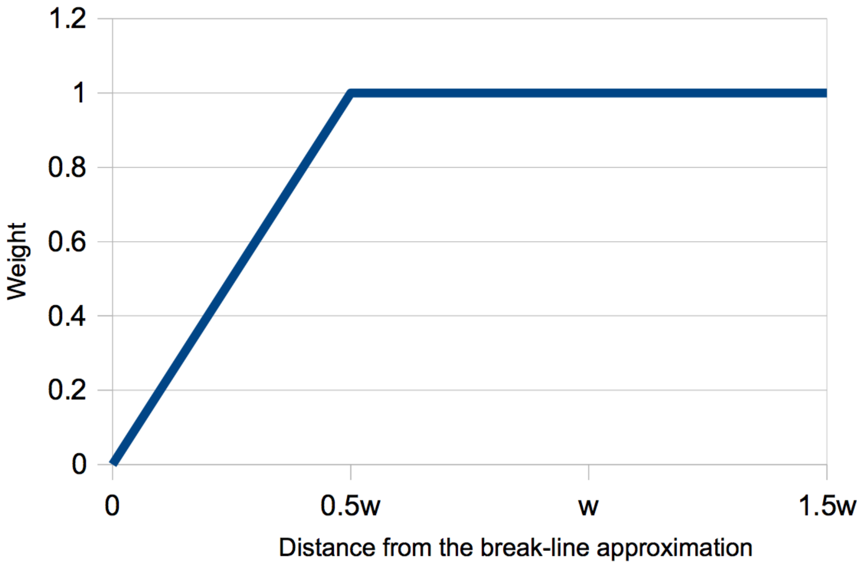

Figure 5.4: The weight function - the function is linearly growing up to the half of the previous patch width, the weight of more remote points is kept unchanged (multiplied by 1).

Weight function. Theoretically, if the approximative break-line split the point cloud correctly into two parts, the three closest points within each patch would be sufficient to define the pair of planes. The suitability of points for the determination of plane's parameters may differ significantly. Solution to this issue is the application of a weight function, which allows to guide the influence of various points. Therefore, the weight function was introduced to reduce the influence of points close to the original approximative line, which were found in the incorrect slope. As a suitable weight function was determined the function linearly growing up to the half of the width w, which the patch reached in the previous step, followed by a constant value for more distant points, as can be seen in Figure 5.4. The application of weights also manifests faster convergence to the sought valley line or ridge line, cf. Figure 5.5 and 5.6.

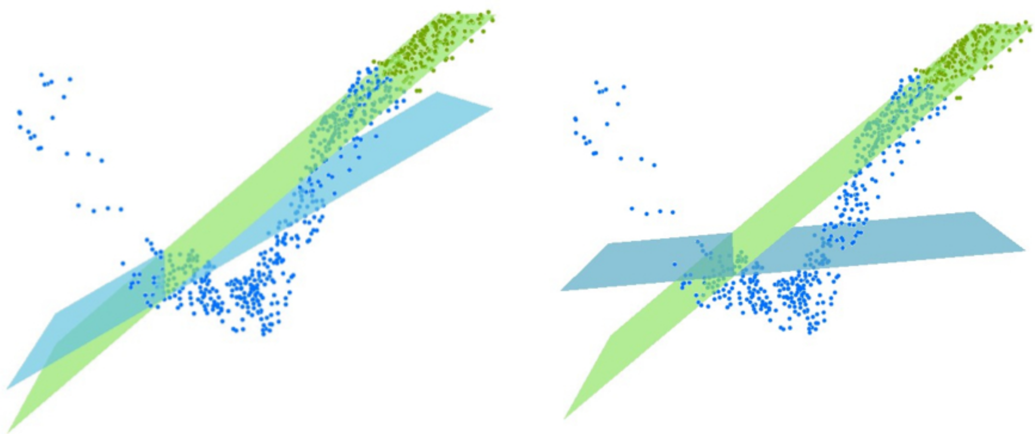

Figure 5.5: 3D view on the surface approximation planes in case of great inaccuracy of primal approximation of a break-line - the widths of the patches are 10m (the unextended side), respectively 50.6m (the extended side), using unweighted (left), respectively weighted least squares fitting. Figure adopted from (112)

5.3.2 Multiple level of detail of break-lines and surface reconstruction

The result of presented framework for break-line modeling constitutes a detailed description of the break-line with potentially a very high point density. Prior to incorporation of the terrain break-lines into the process of multiple LOD database creation, data reduction may need to be considered in practice. The 3D variant of the Douglas-Peucker algorithm suits well for this task. Moreover, the adjusted Douglas-Peucker algorithm can be employed to assign appropriate geometrical importance to the preserved points, which constitute the break-line.

At this point, there is a straightforward way to prevent various spatial objects from changing their topological position owing to key features of the referential terrain model.

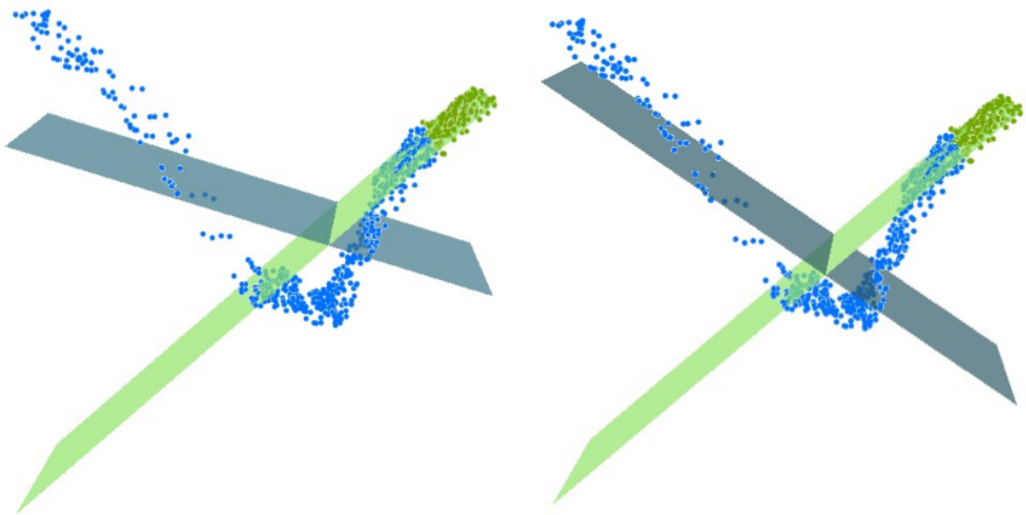

Figure 5.6: 3D view on the surface approximation planes in case of great inaccuracy of primal approximation of a break-line - the widths of the patches are 10m (the unextended side), respectively 75.9m (the extended side), using unweighted (left), respectively weighted least squares fitting. Figure adopted from (112)

That is, because an arbitrary poly-line inserted into the simultaneous simplification procedure 2 acts as a constraint. Therefore, the insertion of break-lines into the process of multiple LOD database creation influences the removal of points in break-line's neighborhood. Namely, it blocks the removal of such points from the footprints of neighboring objects, which would cause any topological inconsistency owing to the break-line.

It is at the discretion of the system operator to assess the needs of the application for particular LODs. The weights of break-lines' points should be set according to the requirements and, moreover, with respect to other layers' settings. The tuning of the weights' settings may be a complex, iterative process. It may require the evaluation of the terrain approximation quality prior to and after the insertion of break-lines into the reconstructed model. The means of such evaluation are, again, application dependent. An example of such evaluation measure was defined by Equation 5.4.

The stored multiple LOD representation of break-lines, however, can be retrieved from the database only for such application, which requires them. Either for the sake of enhanced visual perception of the rendered surface, or the spatial analyses to be run over it. This property is inherently provided by the procedural character of the solution, namely, by the reconstruction of the topographic surface topology only at run-time.

5.4 Closing remarks

In this chapter, following research questions were studied:

8. How to guide the simplification of features with regard to terrain?

9. How to preserve selected features of terrain on coarser levels of detail?

In the section 5.2, several criteria were introduced, which allowed to assess the significance of a topological dislocation between feature to be simplified and the terrain geometry. The main idea was to exploit the locality of the changes and to calculate the error considering only terrain points effected by the poly-line approximation. The modification of the simultaneous simplification algorithm 2 was outlined, in order to extend its functionality in this regard.

Further in section 5.3, an attention was paid to the method for identification of important features of the terrain morphology. First, the approach formerly introduced by Jaroš in (65) was adopted, which provided general break-lines in the terrain shape. The valley lines and ridge lines from the first method were further re-used as a primal approximation for the second method, which was presented for the sake of break-lines accuracy enhancement. The presented method, rooted in the point cloud analysis, enhanced the existing solutions especially in terms of the independence on the primal approximation accuracy. This was achieved through the iterative method, which acts adaptively to the accuracy of primal approximation and takes into consideration the shape of surrounding terrain.

Furthermore, it was shown, that the insertion of the detected terrain feature (break-line) into the database creation process, prevents from a topological dislocation, which could occur on coarser LODs between a spatial object's footprint and such feature of terrain. Moreover, the advantages of the topology reconstruction at run-time were discussed, focusing on the on-demand retrieval of selected feature classes from the database.

6

Experiment

This chapter provides an insight into the implementation and application of the methods, which were proposed in the foregoing text, and performs an experimental evaluation. The specifics of the implementation are reviewed as well as the description of the data and technologies that entered the experiments. Several tests are carried out to evaluate the characteristics of proposed methods and their implementations. The results appear in the order they were introduced in the preceding text. Namely, the processing and analysis of footprints and the creation of multiple LOD database are examined in the first experiment 6.1 mainly through an application of OpenStreetMap (OSM) data. The second experiment in section 6.2 deals with analysis of terrain morphology. The applications of the terrain morphology network for the multiple LOD topography are presented and evaluated.

6.1 Experiment with OpenStreetMap data

The motivation for employment of OSM data in the experiment is given particularly by their global spatial extent, which makes it suitable and logical choice for global geospatial solutions. Also their availability and relative reliability support such a choice. In this section, on the basis of OSM data characterized in subsection 6.1.1, the implementation of the input data preprocessing and analysis are reviewed in subsection 6.1.2. The choices for terrain data, which enter the final synthesis of topographic surface, are explored in 6.1.3. The creation of multiple LOD database and evaluation of multiple generalization scenarios are scrutinized in subsection 6.1.4. This section closes with the implementation and evaluation of the run-time reconstruction of the topographic surface in 6.1.5.

6.1.1 OpenStreetMap

The OpenStreetMap (OSM) is a freely available database of vector topographic data with nearly global coverage. Its content is created and updated via crowd-sourcing through GUI editors (105) and public APIs. As such, it has the following pros and cons:

1. Its quality is spatially variable. The better quality is available with the growing number of volunteers contributing around the particular site. The centers of large cities and popular tourist sites are often better mapped in OSM than in many commercial database, whereas rural areas and developing countries are mapped with far worse quality, whether it comes to completeness, reliability, temporal or attribute accuracy (103).

2. Its completeness is spatially variable. The varying completeness comes from the non-systematic mapping.

3. The spatial accuracy is relatively high, as is further specified in subsection 6.1.1.2. OSM data frequently serve as a primary source, which is filled with data acquired by the means of precise vectorization or GPS traces (56).

The OSM database is thematically very wide; only some layers are employed by the experiment:

1. The transport networks are such a part of OSM, which is mapped with highest detail and best coverage. These data are usually available even in areas with no other features. The positional and attribute accuracy is high. They are frequently used source for analyses (120).

2. In most larger European cities, the buildings are mapped with high detail. In the Czech republic, the coverage nearly complete is thanks to imports from the national cadastral database.

6.1.1.1 Data format

The custom OSM XML is used as the primary OSM data format. It utilizes the topological model of nodes, ways, relations and tags (105), where:

1. Nodes are used for representation of points. They can be a standalone object or a constituting part of ways.

2. Ways consist of nodes and represent lines or polygons.

3. Relations group geometries (ways and nodes) with particular roles for various purposes, resulting in multi-point or multi-polygons.

4. Tags carry the attribute information of nodes, ways and relations.

There are many services available, that convert the required portion of data, typically for specific countries such as Geofabrik (39), into other formats. The geometric component of data is distributed in geographic coordinates using the WGS-84 ellipsoid.

6.1.1.2 Spatial precision analysis

To supplement the literature on OSM precision, Jan Šimbera in his master thesis (125), which was supervised by the author of this work, ran a small-scale analysis of OSM roads against the Czech national database, ZABAGED, (the Czech Basic Map sheet ZM 14-11-08).

The results of the analysis on a sample of nearly 5000 snaps between OSM and ZABAGED layers can be sum up as there was mean snap distance equal to 1,3 m, 96 % of snaps were shorter than 4 m. The remaining 4 % are mostly caused by different linearization of wide roads, squares and roundabouts or by insignificant roads in industrial areas.

Consequently, both OSM Roads and Buildings data layers covering the area of the Czech republic can be deemed as suitable for the experiment.

6.1.2 OSM dataset preprocessing

The initial stage in the experiment is the OSM data processing. It corresponds to the validation of the input described in section 4.2 on footprint analysis.

The tools for analysis of topological relations between objects' geometries are a standard component of most of contemporary GIS software products. Nevertheless, in our implementation, we rely on the CGAL libraries (15), as they provide highly efficient implementation of computational geometry algorithms, which is suitable for processing of large datasets. The CGAL implementation of curves intersection supports detection of all kinds of degenerate cases including overlapping poly-line segments.

The implementation was tested on selected part of the OSM dataset, which corresponds to the spatial extent of 15.00E, 50.25N and 16.00E, 50.80N. The OSM's Roads and Buildings data layers were analyzed. The tested area was delimited by the union of GIG cells of the finest LOD, which covered the chosen area. The implementation reads as input the OSM geometries in the well-known text representation.

Although the specification of the Buildings layer states that it is formed by closed, non-intersecting poly-lines, several exceptions were detected. An example of detected unclosed and overlapping polygon in a *Buildings* layer can be seen in Figure 6.1. The problem of validating the input data is theoretically simple and boils down to validation of the defined set of geometrical rules. Nevertheless, the tools for automatic repair are not available and their implementation is far from being trivial.

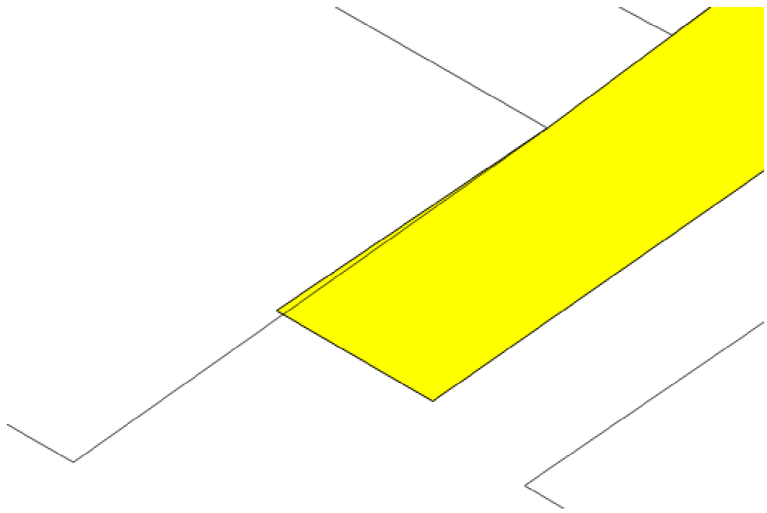

Figure 6.1: Topological error in OSM Buildings layer - the polygon representing the footprint of a building was found both unclosed and overlapping with a neighbor at 15.4741E, 50.6273N location.

The solution to the unclosed polygons can be achieved through employment of the triangulation library of CGAL. It is particularly elegant, because the triangulation is, in the presented solution, the tool that guarantees the prevention from the topological errors caused by poly-line simplification. The polygon closing is guaranteed by the triangulation library, if the input constraint to the triangulation is of *CGAL2Dpoly* class. Another advantage of this approach is, that the potential holes/islands inside polygons are easily handled by he triangulation library, with no other auxiliary data structure or special mechanisms needed.

Several inappropriate overlays of buildings footprint were removed manually. Nevertheless, technically it was not necessary, as the triangulation mechanism (both during geometry simplification and topology reconstruction for the sake of visualization) can deal with these fragments alike any other constraint. Without the removal, the resulting visualization would, however, show some disruptive visual artifacts.

For the sake of assigning the values of weights to individual points, the implementation relies on the squared distance cost class of the CGAL library. The only things needed is to have access to the individual vertices and to be able to add or adjust attributes of the vertices.

We discuss its employment in finer detail in the next subsection, which focuses on the creation of multiple LOD database of OSM data. Lastly, the heights of the OSM geometry were derived from the digital terrain model constructed from the SRTM data 6.1.3.1, as these are the elevation source in other stages of our experiment and they also enter the final synthesis.

6.1.3 Terrain data

For some time, the digital terrain models (DTM) that have nearly a worldwide coverage, have been available. Due to the advances in remote sensing and proliferation of such data sources, some are even provided on a free-to-use basis. ASTER and SRTM are freely available data sources with adequate spatial resolution for scientific use.

6.1.3.1 Datasets review

SRTM (Shuttle Radar Topography Mission) digital elevation model (DEM) has been acquired in the 2000s during the NASA Space Shuttle mission. The third and newest version can be downloaded from (129). It is provided in the WGS-84 coordinate system in 1-degree tiles with 1-arcsecond resolution (this corresponds to approximately 20x31 m pixels in Central European latitudes). The raw data contain the void parts in places with steep slopes, which keep them in the shadow of the radar beam. Methodology for supplying the missing information is available (130). The resulting grid is referenced by the geographic latitude and longitude on the WGS84 ellipsoid. The SRTM height values are related to the EGM96 geoid.

ASTER (Advanced Space-borne Thermal Emission and Reflection) is a joint venture of NASA and the Japanese Ministry of Economy. It was produced by overlaying stereo-pair images from the ASTER satellite optical and thermal imaging instrument (100). The second and current version can be downloaded per 1-degree tiles. For Central Europe region the resolution reaches approximately 20x31 m pixels. However, the scrutiny carried out by (32) showed that this resolution is mainly nominal and there are many artifacts present in the data. (119) confirmed this observation and concluded, that the advantage of ASTER over SRTM may be in a better coverage of mountainous regions and higher latitudes.

As a conclusion, SRTM seems to be a better choice, with exemption of higher latitudes and high mountain regions.

6.1.3.2 SRTM data preprocessing

The final decision to use the SRTM data was also supported by the available implementation of the SRTM data processing, which was introduced by (81) and implemented within the Grifinor project (46). The implementation follows the concepts introduced by (37) (described in Section 5.1.2), it utilized the code from the Visualization Toolkit (VTK) (143), namely the class vtkGreedyTerrainDecimator.

For the sake of our experiment, we exploited this SRTM processing implementation to acquire the preprocessed data for corresponding levels of detail and area of the same extent as of the OSM dataset. The executable Java application outputs the plain text files, each corresponding to the given GIG cell on a particular LOD, bearing coordinate pair and height on every line for a single elevation point. As the next step, this multiple LOD representation of bare Earth is imported to the PostgreSQL database, in a similar way as described in the following subsection for purposes of footprints' geometry persistence.

6.1.4 Multiple LOD database creation

The following text focuses on the implementation details of the geometry simplification procedure, the character of the resulting database and evaluation of several cases of tested simplification.

6.1.4.1 Geometry simplification

For the sake of implementation of the geometry simplification method, which is the core of our approach to the creation of multiple LOD database of features, we got back to the CGAL library. During the time period of this work, the Dyken's approach (28) to poly-line simplification "en mass" was implemented for the CGAL project and it can be found at (15). It makes use of the effective implementation of triangulation solution in the *Tringulation_2* CGAL library.

As the detection of topological conflicts, around which the Dyken's approach (28) was built, stood as an inspiration for the solution to the poly-line simplification proposed in this work, it was decided to utilize the existing implementation of detection of neighboring poly-lines in CGAL. On top of the CGAL's poly-line simplification solution, we implemented an extension for multiple-resolution database creation, which corresponds to the Algorithm 2 introduced in section 4.4.3. The implementation reads as input the representation of footprints from the well-known text files.

The actual implementation of such simplification of footprints lies in the *simplifyVLOD* method. Each poly-line is incrementally inserted into the triangulation. The method allows for determination of the subsets of vertices in the triangulation, that in a given simplification step act as unremovable blockers. That is achieved in a method *initialize_unremovable*, where the vertices, which cannot be removed are determined and their *cost* is adjusted, so as to prevent from the undesirable simplification.

The experimental implementation reuses the original CGAL implementation to handle the termination of the simplification process, i.e. reaching the maximum simplification error (ϵ-based approach), or number of simplified vertices (count stop approach), are provided to the user.

Dyken's solution (28) introduced three cost functions to measure the deviation between the original poly-line and the simplified version of the poly-line. The maximum squared distance, the scaled maximum squared distance and the hybrid maximum squared distance are also implemented in the original CGAL poly-line simplification method. They are, as well, reused by our implementation. Nevertheless, as in CGAL the cost function is a template argument of the simplification function, the implementation of some custom cost functions is straightforward.

The experimental implementation used this opportunity and defined a very simple adjustment of the original cost function based on squared distance. For specified poly-lines it allows to multiply the original cost by a particular constant, in order to differentiate between various types of features and to take into consideration some semantics or attributes of input features. In the experiment, this possibility is utilized to increase the weights (inferred here merely from geometric characteristic) of points belonging to the Buildings layer, in other words, to delay their simplification in relation to the Roads layer. It is also employed in the next case study 6.2 to differentiate the weights of the terrain break-lines from the other entries.

For the construction of the LOD of the resulting representation of the features' footprints, the *simplifyVLOD* method is called iteratively. The extent of the input data is driven in every step by the cells of GIG index, which cover the spatial domain of current interest.

The output for every processed cell is the text file named in accordance with the CID of the GIG cell. It has on every line the record of the point; the pair of coordinates, the height, the order of the point in the original shape of the poly-line and ID of the poly-line. If the poly-line crosses the borders of the cell and returns back, it is marked as an empty line (split into sequences). An example (here writen in two columns) of resulting record of single building in Špindlerův Mlýn for the cell with $cid = 383779642$ (on the level 15000), which crosses the boundary with cell $cid = 383746874$:

```
50.7303304 15.6011183 792 2 143

50.7301641 15.6016478 791 4 143

50.7301641 15.6016478 792 16 143
50.7303264 15.6018154 793 18 143
50.7303349 15.6019234 793 19 143
50.730396 15.6019992 794 20 143
50.7304963 15.6020415 794 22 143
50.7304939 15.6020464 794 23 143
50.7305506 15.6022541 794 25 143
50.7305655 15.6022738 795 26 143
50.7306116 15.6022644 795 27 143
50.7306103 15.602245 795 28 143
50.7306173 15.6019349 796 30 143
50.7306794 15.6019239 796 31 143
```

```
50.7304732 15.6017354 795 33 143
50.730368 15.6016069 793 34 143
50.7303639 15.6015534 793 35 143
50.7303941 15.6015475 793 36 143
50.7303802 15.6013774 793 37 143
```

And the coarser level representation (cell $cid = 63963411$, level 5000):

```
50.7303507 15.6013833 793 1 143
50.7301972 15.6016547 791 17 143
50.7304915 15.601981 794 21 143
50.7305106 15.6022616 794 24 143
50.7306407 15.6022383 796 29 143
50.7306623 15.6016992 797 32 143
50.7303507 15.6013833 793 38 143
```

The implementation of the geometry topology control and the geometry simplification was conducted in a C++ programming language (contrary to other parts of the implementation, which is in Java) because of the availability of the topological conflicts detection in CGAL and the general robustness of this library. It must be noted, that our implementation is experimental and is not ready for a production use.

As a result, each point of the footprints' geometry is assigned to one of the GIG levels. On this basis, the multiple LOD database of features is built. Such a database can be regarded as the main result of the experiment. It validates the entire approach for representation of spatial features' footprints on multiple levels of detail.

The simplest mean of long term data persistence, namely the I/O files storage resulting from the CGAL processing, is not suitable for the solution, that is intended for the distributed environment and client/server applications, or even enterprise-level storage. Therefore, the PostgreSQL database platform was used as a third party technology for the storage of the data. For the Java-based applications the JDBC driver for PostgreSQL is available.

An elegant technical solution, that allows to remain on the object level and to avoid a use of a proprietary formats (e.g. shapefile), is available in a form of Java Data Objects (JDO). JDO technology provides an Application interface (API) for a persistence of a plain Java technology object and database access. The Java developers are separated from the database manipulation, which is done by calling the JDO interface methods, and can focus on data manipulation by working with Java domain objects.

6.1.4.2 Evaluation

For the sake of the evaluation of the implemented method, the OSM's Roads and Buildings data layers between 15.00E, 50.25N and 16.00E, 50.80N (the same extent as in the case of the footprint analysis) were tested, approximately covering the area of northeastern Bohemia. The implementation tested two alternatives, which dealt with two, respectively three GIG levels. From the finest level 15000 the coarser 5000 was inferred. In the second scenario, levels 15000 - 5000 - 2500 were applied. The area was covered by 2260 GIG cells from the level 15000, 220 cells from the level 5000 and 55 cells for level 2500.

To test the implementation, several geometry simplification scenarios were tested and the results summarized in tables 6.3 - 6.6. Table 6.1 shows the number of poly-lines, the original average number of points per poly-line and total number of points for the datasets used in the experiment. The Roads layer and combined Roads and Buildings layer were tested. The stop strategies for performed test scenarios, either for the ϵ based approach, or the count based, are described in Table 6.2.

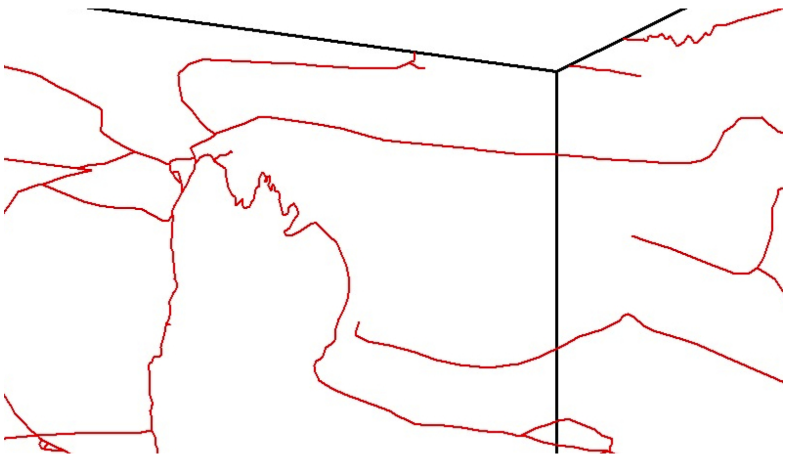

Figure 6.2: The original resolution of Roads geometry in Krkonoše mountains - the section taken on the domain boundaries delimited by the GIG cells of level 5000. 2D visualization in ArcGIS Desktop.

Figures 6.2 - 6.6 show some results from several tested alternatives. In the Figure 6.2, the original geometry of the Roads layer on the boundaries of the GIG cells of the level 5000 is depicted. It is followed in Figure 6.3 by the very coarse version simplified by means of the count-stop strategy (50% of points simplified for the coarser LOD).

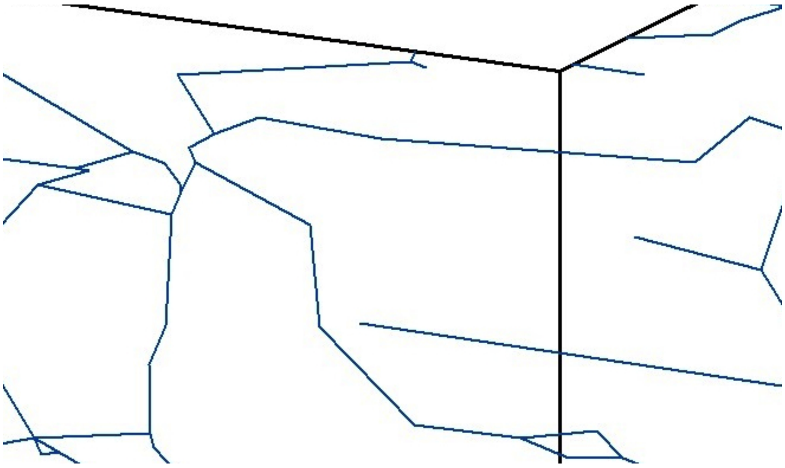

Figure 6.3: The simplified version of Roads geometry in Krkonoše mountains - identical spatial extent as in the previous figure. The count stop strategy (50%) was applied. 2D visualization in ArcGIS Desktop.

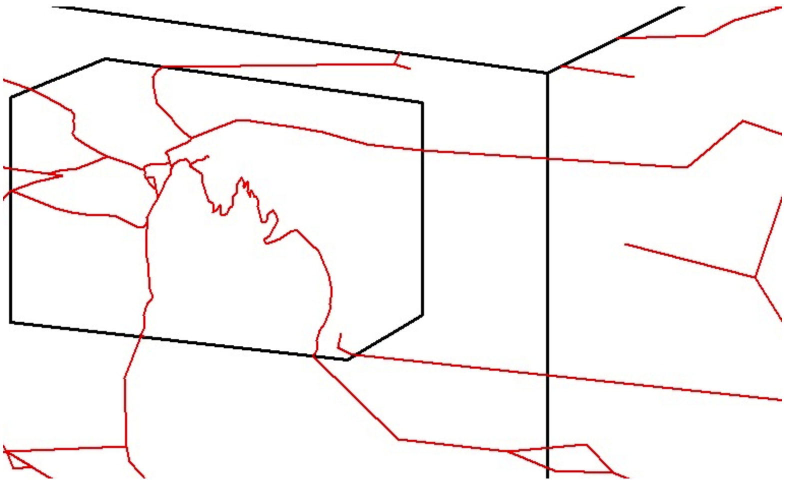

Figure 6.4: The synthesis of two LODs - the finer LOD delimited by a GIG cell of 15000 level. 2D visualization in ArcGIS Desktop.

The statistics of several variants of the single Road layer simplification are written in Table 6.3. The results of combined Roads and Buildings simplification variants were entered in the Table 6.4.

The fit-to-purpose generalization algorithms for buildings datasets were an objective of intensive research in the cartographical community, in order to consider the special character of this kind of feature. Therefore, it is not surprising, that the high rate of the geometry simplification of layers Roads and Buildings leaves the shape of buildings very unnatural. This outcome can be seen in Figure 6.6, in comparison to the original geometry in Figure 6.5. In the experiment, the cost function (maximum squared distance) was adjusted and the adjusted function applied to Buildings layer (while keeping the cost of Roads unadjusted). The adjustment increased the simplification cost of Buildings and mitigated the negative appearance. The results of the simplification with the cost of Buildings layer increased by a factor of 1.25 are summarized in Tables 6.5 and 6.6. As a consequence, the simplified appearance of Buildings corresponded to the result of ϵ_a strategy application (0.00009° as a maximum allowed cost), whose outcome can be seen in Figure 6.7.

Figure 6.5: View on Roads and Buildings in the original resolution above the Špindlerův Mlýn - in the taken picture also the boundaries of level 15000 GIG cells are drawn. 2D visualization in ArcGIS Desktop.

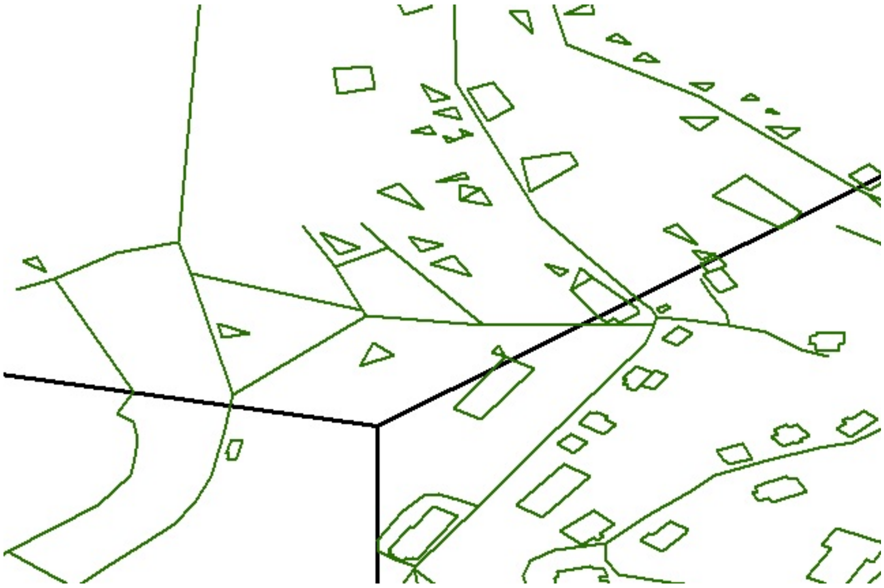

Figure 6.6: Example of an excessive simplification of Roads and Buildings above the Špindlerův Mlýn - beyond the areas of finest LOD, the outcome of the ϵ_b strategy allowing 0.0003° cost, leaving the shape of buildings entirely inappropriate. 2D visualization in ArcGIS Desktop.

It should be observed, that the count-stop (50%, 50%) approach is little bit more aggressive than the ϵ_c one based on the median value of all the costs. This is given by the fact, that the ϵ based simplification stops, when no other points exceeding the ϵ limit are found. And that is usually earlier, than the half of them (median value) is reached, because of the points blocked from the simplification. On the other hand, the count based approach continues until the 50% criterium is met (unless more than 50% of points are blocked, which did not happen).

Input layers	**avg # points per poly-line**	**total # points**
Roads	9.6	149587
Roads & Buildings	9.2	204549

Table 6.1: Average number of points per poly-line and total number of points in the original datasets used in the experiment

Figure 6.7: View on Roads and Buildings above the Špindlerův Mlýn - the layers were synthesized on two LODs. In the extent of the two GIG cells of level 15000 at the bottom, the geometry of the finest LOD is visualized. Above, the coarser representation simplified on the basis of 0.00009° maximum allowed cost. 2D visualization in ArcGIS Desktop.

Stop strategy	**2 LODs**	**3 LODs**
ϵ_a	0.00009°	0.00009°; 0.0004°
ϵ_b	0.0003°	-
ϵ_c	median of all weights	-
count-stop	50%; 50%	30%; 30%; 40%

Table 6.2: The tested simplification alternatives - the values for the ϵ-based simplification stop in degrees and the count-stop as percentages of points assigned to every LOD; two variants were tested with two, resp. three LODs.

Stop strategy	**avg # points per poly-line**	**total # points**	**blocked intra-cell**	**blocked extra-cell**
ϵ_a	6.6	103209	633	94
ϵ_b	3.9	59880	1472	136
ϵ_c	4.9	75753	847	113
count-stop	4.8	74793	850	113

Table 6.3: Result of Roads simplification for the coarser of two LODs - columns in the table corresponds to average number of points, total number of points associated with the coarser level, number of prevented simplifications because of blockers inside the processed cell, resp. outside the cell in the last column.

Stop strategy	**avg # points per poly-line**	**total # points**	**blocked intra-cell**	**blocked extra-cell**
ϵ_a	6.4	143191	954	108
ϵ_b	3.9	85915	2133	157
ϵ_c	4.6	103547	1146	127
count-stop	4.6	102274	1155	129

Table 6.4: Result of Roads and Buildings simplification for the coarser of two LODs - columns in the table describes the same attributes as the Table 6.3.

Stop strategy	**avg # points per poly-line**	**total # points**	**blocked intra-cell**	**blocked extra-cell**
ϵ_a	6.6	146212	975	108
ϵ_b	4.0	89156	2159	157
ϵ_c	4.6	103605	1204	127
count-stop	4.6	102274	1155	129

Table 6.5: Result of Roads and Buildings simplification for the coarser of two LODs (with the cost of Buildings multiplied by a factor of 1.25) - columns in the table describes the same attributes as the Table 6.3.

Stop strategy	avg # points per poly-line	total # points	blocked intra-cell	blocked extra-cell
ϵ_{a1}	6.6	146212	975	108
ϵ_{a2}	3.8	84806	2331	170
cs_1	6.4	143164	947	108
cs_2	3.7	81819	2296	161

Table 6.6: Roads and Buildings for three LODs (with the cost of Buildings multiplied by a factor of 1.25) - columns in the table describes the same attributes as the Table 6.3. First two rows correspond to ϵ thresholds between first and second and second and third LOD. The count stop approach results are described on the third and fourth line (leaving 30% points for L^1, 30% for L^2 and 40% for L^3).

6.1.5 Synthesis and visualization

This phase of the experiment focuses on the implementation and evaluation of the data synthesis, and the technical issues related to the run-time control application including the viewer component are presented.

The Java-based control application has a form of a processing pipeline. It utilizes the Apache Commons component *Pipeline*, which objective is to provide a set of utilities for parallelized data processing and several multithreaded processing models. The pipelines allows the data objects to be processed by a series of *Stages*, which are the primary unit of execution in a processing pipeline. They act as an independent user-defined components, the threading model of each stage is independent on the threading model of the other stages. The reusability of stages in other pipelines is another advantage.

The control application has four stages:

1. 4NN Query

   This stage is responsible for initialization of the actual spatial extent based on the position of the observer and is implemented as $Get4NNQueryStage.java$. As the name suggests, the stage employs the four nearest neighbors GIG query to retrieve the identifiers of always four cells on all required LODs, which are closest to the position of the observer.

2. Retrieve elevation points

On the basis of cell's identifiers retrieved in the previous stage, the stage *ReadHgtStage.java* fetches the terrain data from the database. The terrain database corresponds to the database created in 6.1.3.1. In other words, for each cell identifier we retrieve the data object bearing all elevation points corresponding to the spatial extent of given cell and its level of detail.

The data retrieved by this stage are passed to the stage number four.

3. Retrieve features

   Similarly to the previous stage, on the basis of index cells of interest, the data are retrieved from the database. In this case, the *ReadFeaturesStage.java* fetches the data objects holding the geometries of footprints corresponding to given level of detail and spatial coverage of the GIG cell.

   The course of the poly-lines is reconstructed on the basis of fetched geometries, the reconstruction procedure 3 is implemented as *PLReconstruction* method and invoked within this stage. The reconstructed geometry of footprints, that is relevant to the given position of observer, is passed to the next stage.

4. Reconstruction

   The last stage $FeatureEnhancedSurfaceStage.java$ focuses on the reconstruction of the topographic surface. The terrain points and the geometry of footprints retrieved in previous stages become an input to the constrained Delaunay triangulation. The CDT is utilized to reconstruct the topology of the surface at run-time of the application.

For the sake of visualization of results, the application utilizes the thin client developed within the Grifinor project (46). It provides the viewer component and interactive navigation. It is build around the Java 3D API, which provides constructs for building and manipulating 3D geometry and creating 3D graphics applications. It also makes use of Aviatrix3D Toolkit, which is the Java scene graph API over the top of Java OpenGL bindings. The triangulation utilized by the Grifinor library is the implementation of the divide and conquer algorithm introduced by (52).

The Reconstruction stage invokes the Grifinor client and passes the geometry reconstructed by the means of CDT to the viewer component. Some snapshots of the scene can be found in Figures 6.10 or earlier in 4.10.

It must be noted, that the triangulation is the most expensive operation in relation to the run-time visualization, as follows from the Table 6.7. Table 6.7 summarizes the performance of the stages leading to the reconstructed topographic surface covering the selected area. The statistics correspond to the observer position 15.6E, 50.7N and the SRTM, Roads and Buildings data prepared for two LODs 15000 - 5000 and the three LODs 15000 - 5000 - 2500 version, always as a result of nearest centroid query (1x2, 1x3 GIG cells) and four nearest neighbors (4x2, 4x3 GIG cells).

Query type	**Query** [sec]	**P-L reconstruction** [sec]	**triangulation** [sec]	**# points**
$1x2$	0.07	0.02	0.09	660
$4x2$	0.11	0.07	0.38	2172
$1x3$	0.15	0.15	0.91	4427
$4x3$	0.31	0.37	2.92	13036

Table 6.7: Performance statistics of the topographic surface reconstruction.

The total numbers of points retrieved upon the query may vary significantly based on the observer's position. This is not extraordinary, since the distribution of spatial data is naturally variable. More spatial data concentrate to urban areas, while in the rural are more sparse. The position-dependent LOD, however, poses the ability to magnify this characteristic. It is easy to imagine the situation of an observer above the city, being visualized on the finest LOD, and the distant rural areas simplified. As the result, the amount of geometry simplification may not be sufficient for the application's purposes. In other words, there is still room for improvements. The optimization of the simplification strategy, in order to keep the information density between scenes in balance, is an example of such.

6.2 Terrain morphology analysis and application

This experiment pursues two major objectives. First, to verify the ability of proposed method to guide the simplification of spatial objects' geometry with respect to important features of terrain morphology (like ridge-lines). Second, to test the ability of the system to enhance the visual quality of the simplified terrain on users demand and to demonstrate its potential for spatial analysis.

For this sake, the first step of the experiment will be the analysis of terrain morphology on the basis of LiDAR data. As its result, the network of the most important ridge-lines will be acquired. The ridge-lines are the intended input to the process of multiple LOD database of features creation. Their involvement will guarantee the preservation of topological position between the features of terrain and the other features.

6.2.1 Terrain data

The LiDAR data, originating from local imaging of the National park Krkonoše, are the primary data source for this experiment [1]. The point cloud density was 5 points per square meter, covering the area of $550km^2$. The details on the data processing can be found in (115).

6.2.2 Raster-based analysis

As was proposed in 5.3.1, the location of break-lines through the raster-based analysis provides the primary approximation of the searched terrain feature. This case study focuses on ridge-lines, which are a subset of general break-lines.

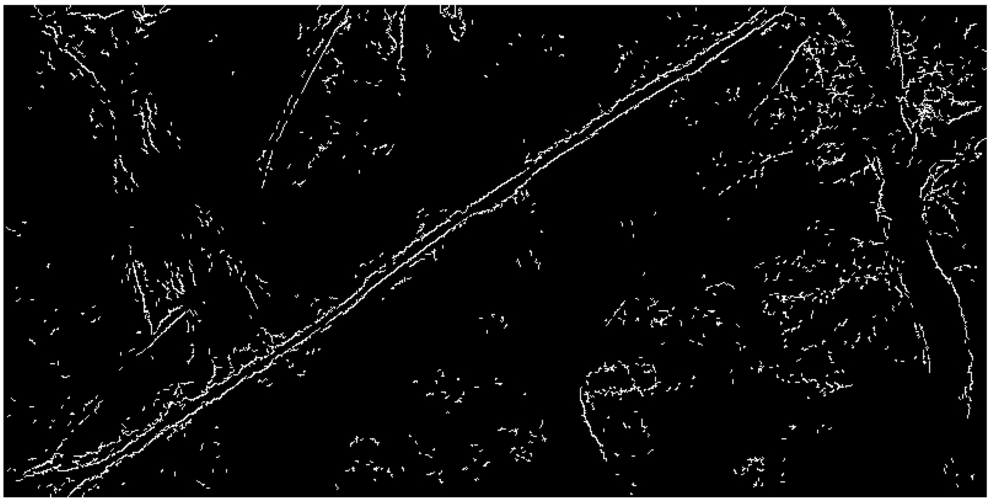

Figure 6.8: Detected general break-lines - the matlab visualization of a raster bearing the information of all detected break-lines.

[1] They were acquired within the project Krkonoš v INSPIRE společný GIS v ochraně přírody, which was supported by the operational program Přeshraniční spolupráce ČR-Polsko 2007-2013

For this sake, from the source point cloud, as an intermediate step, the TIN was derived and translated into the raster-based digital terrain model with 1m pixel size, referenced by UTM33 coordinate system together with WGS84 ellipsoidal heights system.

The matlab source code (64) for break-lines detection designed by (65) was adopted. The output can provide general break-lines, cf. Figure 6.8. On the basis of analysis of surrounding slopes, important ridge-lines can be selected as well, as illustrated in Figure 6.9. Results were, however, further cross-analysed with the slope characteristics and watershed boundaries in ArcGIS Desktop. Rather minor manual processing and corrections could not be avoided at the end. The results were transformed to the shapefile data format, which is the input to the point cloud analysis.

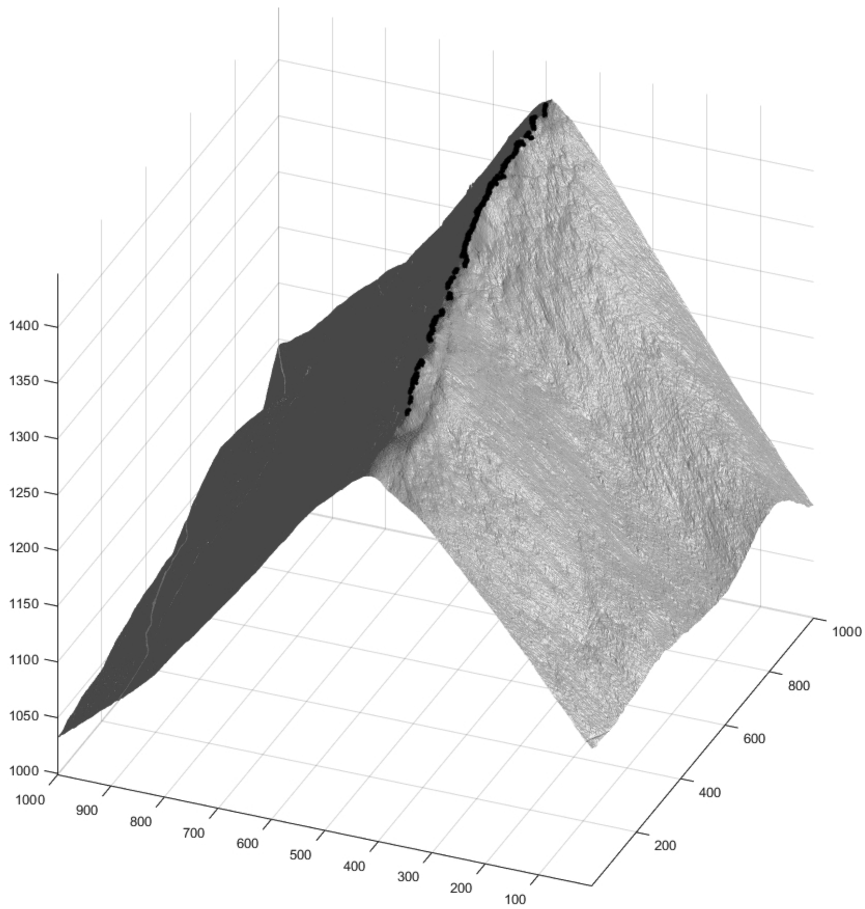

Figure 6.9: Detected ridge-line - the Matlab visualization of the ridge with the found ridge-line.

6.2.3 Point cloud analysis

In order to increase the accuracy of the primal approximation, the next stage of the ridge-line network preparation employed the analysis of the lines course over the point cloud. For this sake, the experiment utilized the Python source code (111) developed within the master thesis project (112) supervised by the author of this work.

The implementation uses the ArcPy library (4). The core of the analysis is contained in the Detection.py script, where the geometrical operations described in 5.3.1.1 were implemented. As the original work was more interested with valley line, several adjustments had to be carried out. These relate to the changes of convexity (concavity) of the slopes neighboring to the explored ridge-line in comparison to valley-line and dealing with the gradient. As well as in the previous case of raster-based detection, the results had to be visually controlled and in some places manually adjusted.

Generally both, the point cloud and the raster-based analyses, worked well for detection of sharp edges (ridge lines, valley lines, as well as general break-lines), but provided debatable results, when even significant elevation difference (deep valley, high mountain) has rounded ending (valley bottom, mountain ridge).

6.2.4 Multiple-resolution topographic surface with ridge-lines

Several data layers, whose total numbers of points in tested area are summarized in Table 6.8, enter the process of the multiple LOD database creation. In this case, the ϵ_b stop strategy was applied. The cost of the ridge-lines is determined by the maximum squared distance function as well as all the other layers. However, similar to the adjustment of the simplification cost of Buildings layer in previous section, also the cost of the ridge-lines was adjusted. It was multiplied by a factor of 5 to highly limit the intensity of simplification of ridge-lines. This decision was motivated by the endeavor to preserve the important shapes of terrain and visibility characteristics on the coarser level, as well as to enhance the overall visual perception of the scene.

The analysis of the simplification, cf. Table 6.9, identified 21 blocked simplifications of points, out of which 17 concerned changing the position of a road owing to the ridge-line (the rest for buildings). These blocked points relate to 7 segments of roads, which would shifted to a neighboring valley or cut through a salient ridge, unless blocked by the ridge-line. The number of prevented simplifications closely depends on selected

input layers	total # points
Ridge-lines	1817
Roads & Buildings	43734
SRTM	20766

Table 6.8: Input layers and numbers of points

layer	total # points	blocked intra-cell	blocked extra-cell	blocked by the ridge-lines
Ridge-lines	1809	0	0	0
Roads & Buildings	30434	227	20	21

Table 6.9: Statistics of the geometry simplification - on the Roads, Buildings and Ridge-lines layers the ϵ_1 stop strategy was applied. SRTM data were decimated according to a strategy leading an equal number of points fetched from the database for each LOD constituting the scene.

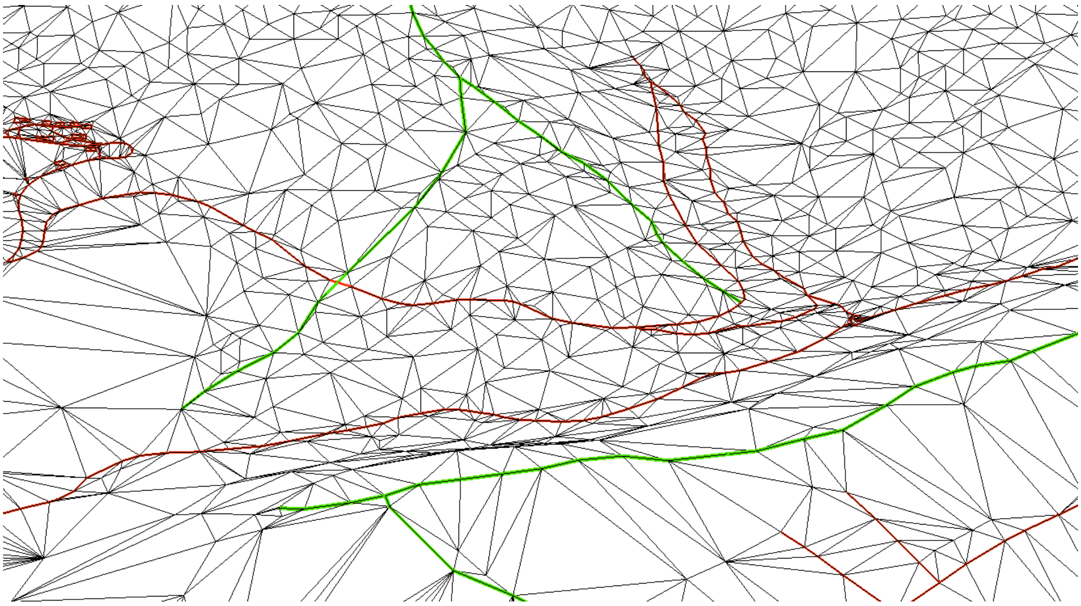

Figure 6.10: A view on a 3D scene synthesized for Grifinor viewer component - the view of the observer taken above Špindlerův Mlýn towards the near hills of Krkonoše mountains; the finest resolution of SRTM terrain, ridge-lines (yellow) and roads with buildings (violet color).

detail and completeness of the terrain morphological structure, which is required by certain application. In this test, only eminent ridge-lines (and no other kinds of terrain morphology) were used. The 3D scene illustration can be seen in Figure 6.10 to provide a graphical impression of the local situation. The visual results of the simplification based on ϵ_b stop strategy are displayed in Figures 6.11 and 6.12. The 6.11 figure exposes the situation, when the ridge-lines were not part of the multiple LOD database creation. As a result, the simplified road unnaturally cut its way through an important hill, significantly changing the shape of terrain and the relation between terrain and the feature. On the contrary, in Figure 6.12 the incorporation of ridge-lines prevented the undesirable simplification of the road. The ridge-line blocked removal of two points from the directly influenced road, which consequently blocked removal of another point from neighboring road.

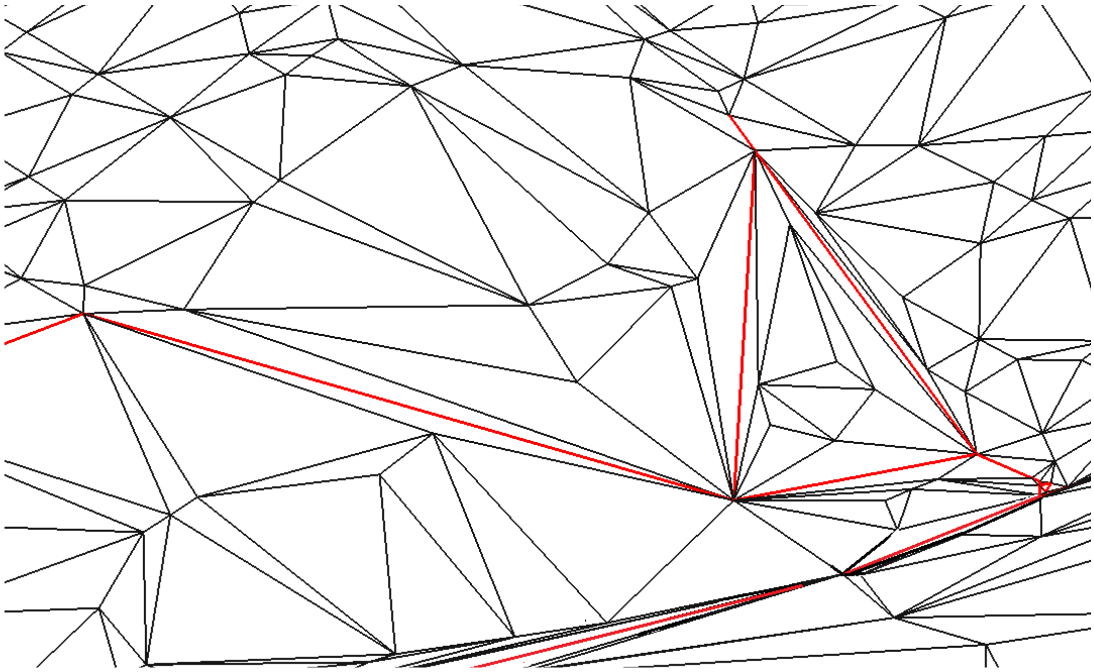

Figure 6.11: The snapshot of a coarser LOD of the topographic surface without ridge-lines - only Roads and Buildings layers entered the creation of the multiple LOD database. The simplified road (violet color) making an abrupt shortcut.

The changed impression of local situation described in Figures 6.11 and 6.12 is also apparent in an overall view on the scene. The visual artifact is apparent in the centre of Figure 6.13 in contradistinction to the Figure 6.14, where the ridge-lines were accounted

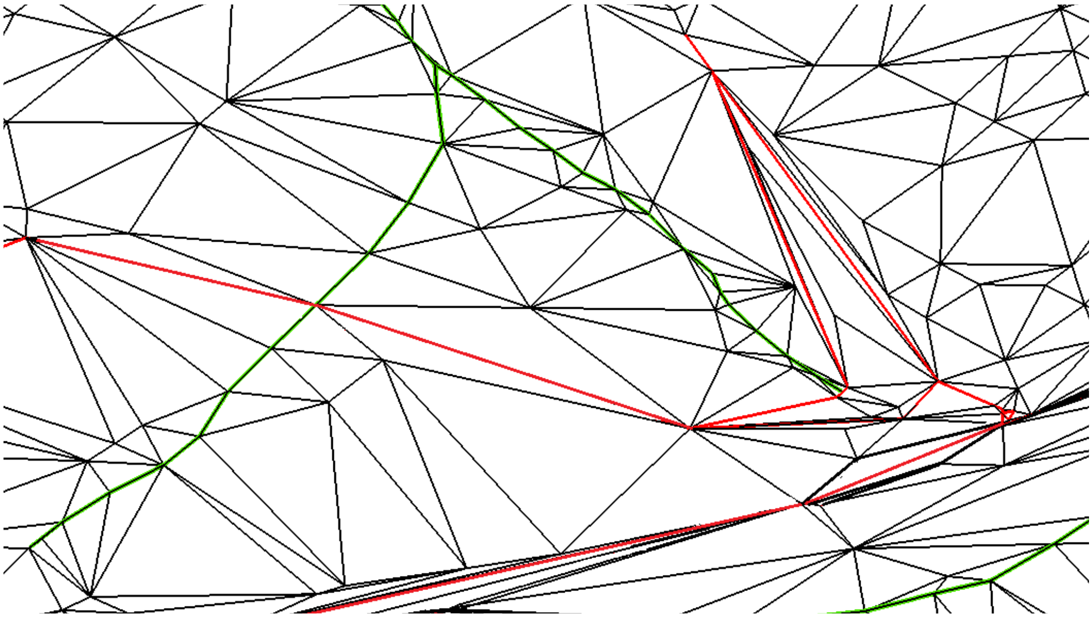

Figure 6.12: The snapshot of a coarser LOD of the topographic surface with ridge-lines - the presence of ridge-lines (the violet color) forced the concerned roads (red lines) to take account of the terrain character.

for. In the illustrated scene, also the ability to enhance the visual appearance of terrain and maintain its shape is demonstrated in 6.14 in comparison to 6.13.

Visibility analysis. As an example of an spatial analysis of the reconstructed multi-resolution topographic surface, the visibility test was performed. The motivation for this test is also given by the close relation between the visibility similarity and the perceptual similarity of the original and simplified topographic surface. The visibility analysis was carried out in ArcGIS Desktop, where the geometry of terrain from multiple LODs was imported and on its basis the DTM was reconstructed for ten selected observations points.

In order to evaluate the effect of the addition of ridge-lines, two variants of terrain model were dealt with. The first one referred to the SRTM model, the second one to the SRTM model populated with the ridge-lines.

In the test, as the referential surface, the digital terrain model built from the finest resolution SRTM data, was used. Afterwards, the visibility analysis results were compared with the outcome of visibility analyses performed over two variants of multi-resolution surfaces; with and without the ridge-lines. The visibility analysis

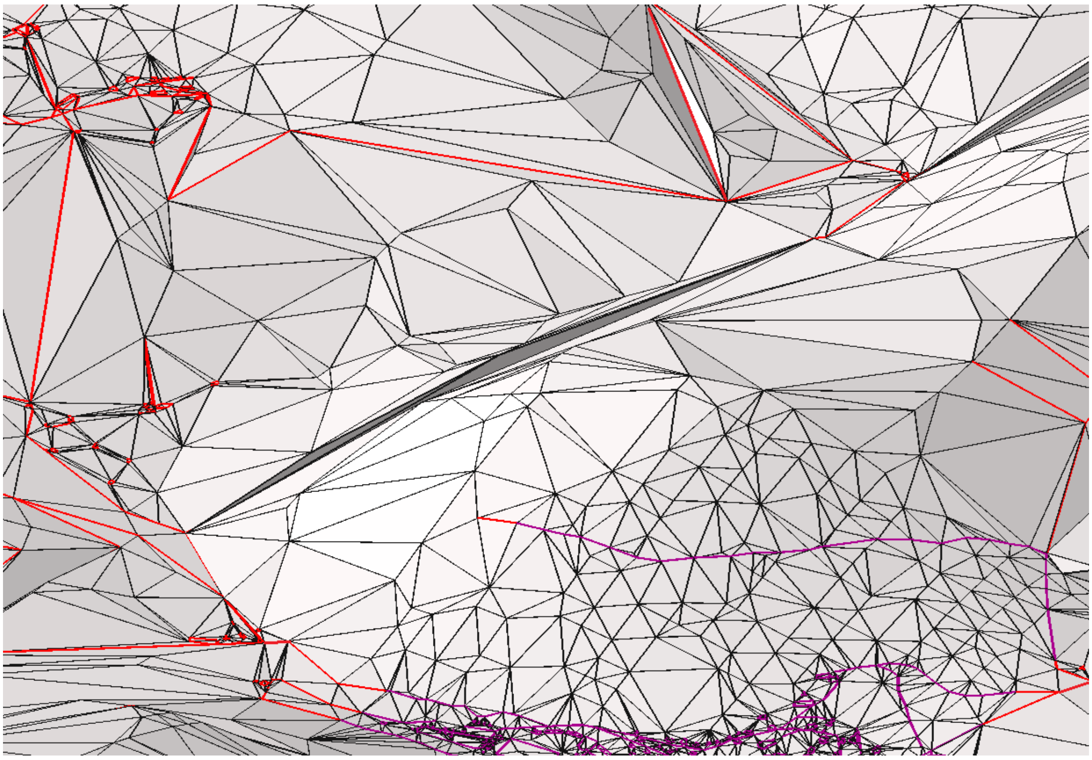

Figure 6.13: The view on a scene with two levels of detail without ridge-lines - with the finer resolution at the bottom of the figure and with distant ares from the observer simplified. The Roads and Buildings on a finer resolution drawn with violet color, in the coarser LOD areas as red. The situation from Figure 6.11 is also apparent at the top with an unnatural cut into the ridge.

input	**same visible** [%]	**same invisible** [%]	**different visible** [%]	**different invisible** [%]	**overall change** [%]
SRTM	7.32	87.20	1.73	3.75	5.48
SRTM + R-L	7.80	87.67	1.49	3.04	4.53

Table 6.10: Visibility analysis results - percentages of pixels bearing the same or different values of visibility over the multi-resolution models (the second enhanced with ridge-lines) in comparison to the full resolution terrain model analysis. The values are the average from 10 measurements (10 observation points).

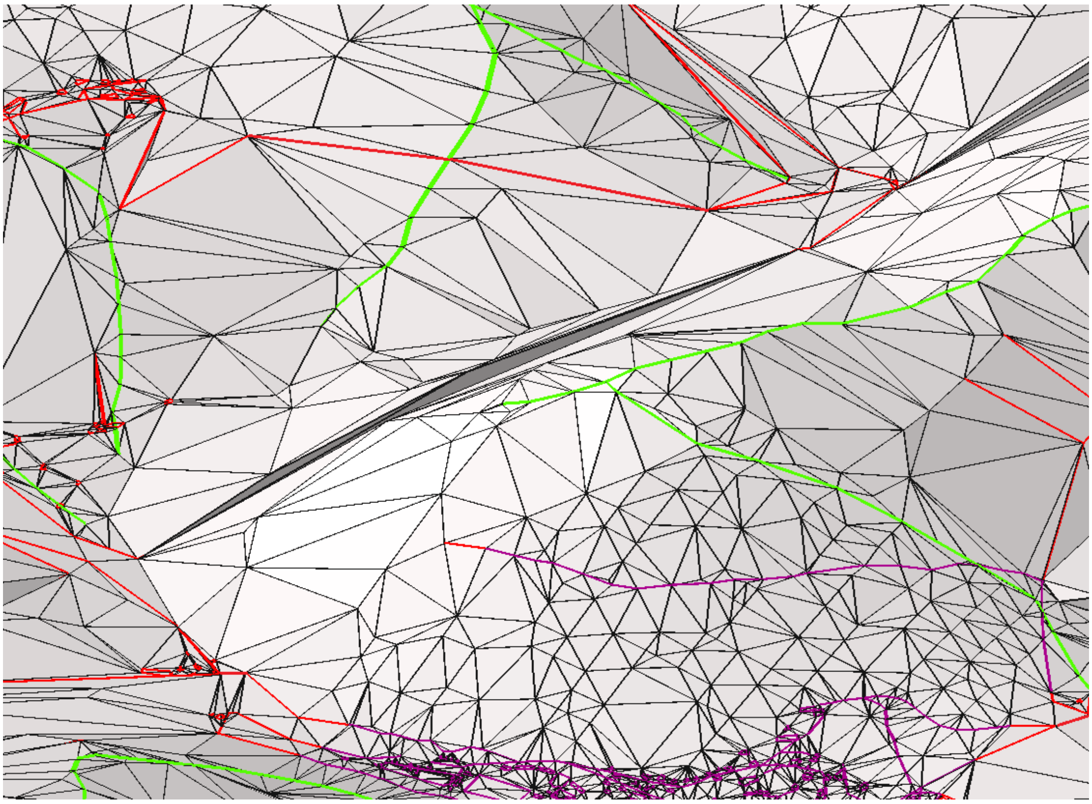

Figure 6.14: The view on a scene with two levels of detail with ridge-lines - the ridge-lines (green color) preserve the shape of terrain and the overall appearance of the scene, cf. with Figure 6.13. At the top, the situation from Figure 6.12 from a more distant perspective.

was performed from 10 randomly distributed observation points over the terrain model corresponding to the spatial extent of 4NN GIG cells of the level 5000.

The GIG cells of the level 5000 held the finest resolution of the SRTM data for the sake of the visibility analysis over the referential surface. The multi-resolution surfaces held the finest resolution in the scope of 4NN GIG cells of the level 15000, the rest of the spatial extent (delimited by the 4NN GIG cells of the level 5000) was on the coarser resolution.

The outcome of the visibility analysis was a binary raster, marking the visible and invisible pixels from given observation point. The overlay of visibility rasters was the next step. First, the rasters of visibility over the referential (full resolution) surface and the raster of visibility over the multi-resolution surface without ridge-lines, were overlaid. Second, the overlay of referential raster was repeated, this time with the multi-resolution surface with ridge-lines.

It provided new rasters bearing information of pixels with the same visibility, same invisibility, different visibility and different invisibility over the multi-resolution models in comparison to the referential one, cf. Figure 6.15. The results summarized in Table 6.10 show nontrivial improvement of preservation of visibility characteristics, when employing the ridge-lines in the multi-resolution surface creation.

More profound tests of visibility characteristics stand for future challenge. Such tests should, for example, compare two multi-resolution models with the same number of vertices. This condition was not fullfiled in the described test, since the incorporation of ridge-lines increased the total count of vertices.

6.3 Closing remarks

This chapter proposed several experiments to show the functionality and behavior of the methods presented in the thesis. The functionality of the multi-resolution concept of topographic surface with features was verified by means of the created multi-resolution database. Multiple scenarios were tested:

- various input feature layers and their combinations were used as an input to the multiple LOD database creation,
- the database was created with the use of different stop criteria to the simplification,

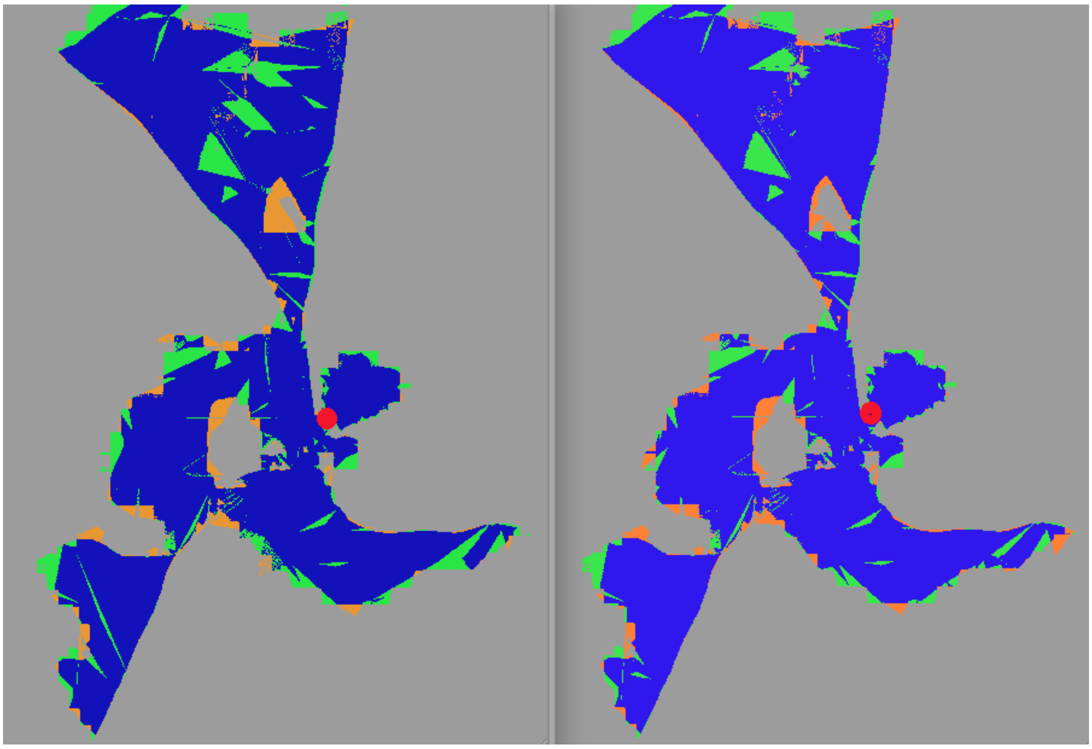

Figure 6.15: The resulting overlay of the visibility analysis rasters - the rasters hold four values. The blue color represents places visible from the observation point using either of terrain models; the referential one, or the multi-resolution one ((left) - multi-resolution without ridge-line; (right) - multi-resolution with ridge-lines). The grey color represents places invisible over both models. The green color places invisible on multi-resolution model, but visible on full-resolution. Orange color places visible on multi-resolution model, but invisible on full-resolution. Red dot marks the observation point.

- the cost functions of selected layers were adjusted,
- databases with two as well as three LODs were provided.

The outcome of buildings layer simplification suggested the need for advanced generalization operations, in spite of the fact, that the poor resulting appearance of highly simplified buildings was dealt with, by employing the cost function adjustment. This is an expected result. The simultaneous simplification method together with the concept of footprint are intended to be a framework, upon which other functional extensions should be built.

On the basis of the multiple LOD databases of features and bare Earth, the scene of multi-resolution topographic surface was reconstructed at run-time and several performance statistics were provided. However, the more proper tests are required, especially regarding the distributed data management. Such tests would have to consider variable network characteristics, as well as different hardware performances on both, the client and the server side.

Results of visibility analyses over the multi-resolution surfaces were compared, in order to evaluate the ability to enhance the quality of terrain approximations on coarser LODs on-demand. The test confirmed the improvement and proved the possibility to improve the representation of terrain due to the procedural character of terrain reconstruction. It should be noted, that it does not compare the influence of added ridge-lines and the addition of the same amount of terrain points (the most important from those removed).

From an implementation point of view, this work contributed especially on two levels.

The first one was the implementation of the generalization of footprints, that fulfills all the requirements, that were laid upon it by the character of the demanded resulting multi-resolution topographic surface.

The second one was the implementation of the procedural topographic surface with features, namely the reconstruction of such surface on the grounds of multiple LOD database (potentially distributed) of features as well as terrain points.

These main themes required several complementing implementations, especially regarding input data creation or preprocessing. Namely, the analysis of footprints; preparation of the terrain points database; terrain features extraction; and the scripting for automation of data analyses carried out in ArcGIS; contributed to the evaluation of the overall solution.

Consequently, the experimental source codes do not form a compact, ready-for-use software solution. The source codes, however, can be provided on request, as well as the insight of the author into their usage.

Since this project started as certain part of the Grifinor project (46), it is intended to incorporate some of the implementation results, which were achieved with the use of other programming environments, into the Grifinor libraries. That will also make them a part of a deployment-ready solution. This especially refers to the methods achieved with support of CGAL libraries. CGAL proven itself to be enormously powerful tool, however, it introduces similarly enormous dependency to any system built upon it. This dependency restrain the quest of interoperability, one of the main thoughts of this thesis.

7

Conclusions and future research

This work has introduced a procedural data representation for the feature-based topographic surface, which is globally applicable and supports position-dependent level of detail. The introduced solution reflects on multiple requirements of digital Earth systems. In addition to the visualization performance, the properties regarding data interoperability, data management and distribution, data analysis and the multiple LOD were considered as essential for the design of the geographic information system with global coverage.

This chapter points out the major achievements of the dissertation thesis. It sums up the contributions to the foundations of digital Earth technology and highlights the gained insights to the multi-resolution topography. It provides the answer to the main research question in section 7.1 and closes with an outlook of future research in section 7.2.

7.1 Conclusions

The overall objective of the thesis was expressed in the main research question:

How can we realize the global feature-based topographic surface
with support of the position-dependent level-of-detail?

The closing sections of preceding chapters provided answers to declared subquestions:

1. What is the state-of-the-art in large spatial data management and generalization?

2. What indexing and paging data structure should be adopted, so as to simultaneously support the multiple LOD and fulfill the preference for non-projected solution?

3. How can be formally described what is multiple LOD environment?

4. How to design a conceptual data model that describes the feature-based topography in the multiple LOD environment?

5. How can be the valid input data created by as much automatic manner as possible?

6. How can be the database of features created so that the topological relations between features will be preserved in the multiple LOD environment reconstructed on its grounds?

7. How can be the topographic surface with features synthesized for given observer position?

8. How to guide the simplification of features with regard to terrain?

9. How to preserve selected morphological characteristics of terrain on coarser levels of detail?

In the following text the synthesis of the conclusions will be performed. In summary, following key factors extended the state-of-the-art of multi-resolution topography.

The aspect of data management. The data management in the proposed work exploited the GIG as an indexing and paging method. It is suitable to remark the original motivation for adoption of GIG, instead of more frequently used quad-tree indexing.

The first reason was the elimination of dependency on cartographical projections. GIG method divides the three dimensional space with an arbitrary resolution into cells, which are radially distributed around the origin. Afterwards, the cartographic projection can be selected independently for each cell or entirely avoided, if suitable. This is an inverse process than usual. Traditionally, the projection is the first step, which is followed by the search for a suitable way of indexing. This makes the projection an inherent and unavoidable feature of the system, which, from the perspective of geo-spatial information systems, causes serious complications.

The second reason for GIG employment was the fact, that on each level of detail, the shape of GIG cells differs. Generally, this characteristic can frequently be a disadvantage. However, if a data model of an arbitrary geographical phenomenon uses GIG, then it is surely independent on the used indexing structure. That is because there is no topological or shape property of GIG cells, which any prospective data representation could be founded on. The only systematic geometric feature of GIG are the centroids, thus the points, which from its nature do not have any shape.

Consequently, the mechanism of proposed data representation is applicable to any other indexing structure, which supports multiple LODs. From the information system architecture point of view, this is a valuable property.

Data representation of features. The proposed solution to representation of spatial features for multiple LOD environment is rooted in the concept of footprint. The choice for such solution is based on the analysis of multiple aspects related to the character of digital Earth systems.

The exploitation of the concept of footprint allowed to populate the bare Earth model with any geographic feature, which can influence its shape. Conceptually, footprint alleviates from the structural complexity of pure 3D solutions, but it also supports the extension to true 3D, when needed. It can be achieved via integration of the 2.5D terrain with independently modeled 3D objects, where the footprint has a role of the geometric interface.

Considering the surface reconstruction at run-time, the query result to the database of footprints is a set of points (and their order), which can easily address local irregularities of the surface. This is a very flexible approach from perspective of both the bare Earth and the spatial objects. Moreover, the data management of points is trivial in comparison to sophisticated hierarchical data structures, which are the contemporary state-of-the-art, when dealing with multi-resolution properties.

Therefore, the concept of multi-resolution footprint contributes to the technological foundation as an universal and interoperable geo-spatial data structure. Because of its conceptual simplicity, it is easily extensible and other solutions can be build on the grounds of this concept. Prospectively, it supports an exchange of more functional data representations in contrast to static models, which are an obstruction for the cooperation between different geographic information systems.

Lastly, this work extended the existing usage of footprints by providing support for multi-resolution representation. Consequently, the footprint can be represented with variable resolution along its course in the reconstructed graphic scene.

Multiple LOD of features. The analytical applications are the key attribute of GIS. Increasingly, also the suitability of such systems for the distributed environment is accentuated. Most of contemporary approaches to the global solutions, like the digital Earth systems, focus on the visualization performance. The high visual performance is achieved by the use of special data structures optimized for rendering and extensive preprocessing of data into these structures. However, this optimization towards visualization hampers the data management of spatial data, their analysis and distribution. It was concluded in several works like those of Rabinovich and Gotsman (116) or Kolář (77), that the topology is the cause of increased the complexity of the solutions, especially from the perspective of data management. With this reasoning in mind, this work extended the existing approaches to procedural topographic surface by incorporating the poly-lines into the concept.

It was shown in this thesis, that the order of points is the only topological information about the poly-line, which is needed for the procedural reconstruction of multi-resolution topographic surface.

The consequence of this finding clearly manifested itself in the primary geometric structure proposed for poly-line representation. And, it also has a significant impact on the factor of data storage, e.g. storing multiple representations for each LOD, auxiliary geometry or auxiliary (hierarchical) data structures. Neither the used primary data structure, nor the proposed method of poly-line reconstruction, introduce any redundancy. As was also observed, that reusing the geometry from a coarser LOD for the reconstruction of a finer detail, which prevents from a data redundancy, introduces the dependency of the finer LOD on the coarser LOD(s).

The introduced procedural reconstruction of the poly-line's topology also allowed to avoid the known issue of "stitching" the geometry from different LODs together. Stitching was proven as a serious complication for approaches dealing with topology, since the topological correspondence between LODs is missing.

Lastly, the presented approach to the solution was, that the storage of the topology should be avoided, whenever possible. Despite this fact, for the resulting reconstructed topographic surface with features, the consistency of topological relations between features is preserved. It was shown, that after the initial analysis and validation of input footprints, it is necessary to guarantee topological consistency under generalization operations. For example, the proposed algorithm for creation of multiple LOD database of features (which simultaneously simplifies geometries of input poly-lines) does not introduce any topological errors. Meaning, the procedural topographic surface reconstructed on the grounds of such database preserves for arbitrary query point the original topological relations between features on all LODs of the scene.

Analysis of terrain vs. feature relation. The role of a footprint as a functional interface of the spatial object applies also to the mutual interaction between the feature and the terrain surface, which the feature is referenced to.

This work studied this influence particularly from the perspective of generalization of the feature. It proposed several criteria that evaluate the importance of a topological dislocation between feature to be generalized and the terrain geometry. By incorporation of these criteria into the generalization of feature's geometry, the outcome of the generalization is influenced in such a way, that the shape of the terrain is respected and better preserved. The related adjustment of the simultaneous simplification algorithm was discussed. It must be noted, that the experimental tests of the criteria were not finished yet. As is described in section 7.2 on possible extensions, a research in progress, which extends this work by introducing the aggregations of areas, deals with proposed criteria to manage aggregation of blocks of buildings in a steep terrain.

Furthermore, another method that allows to preserve the shape of terrain, or even to enhance the quality of the terrain approximation on coarser LODs, was presented. It was achieved through exploitation of such features of terrain, which describe its morphological structure. For this sake, the enhancement of existing methods to extraction of valley lines and ridge lines was proposed. The presented solution acts adaptively to the accuracy of the data input of the algorithm, especially to the primal approximation of the sought feature. This is in compliance with the overall approach of the thesis to limit the requirements on the input data of various kinds, having interoperability issues in mind.

It was shown, that the terrain break-lines can prevent from undesirable simplifications of other features and enhance the visual quality of multi-resolution terrains. The experiment with terrain features also demonstrated the elegancy of procedural approach in terms of on-demand selection of data layers. Thus, the scene is reconstructed at run-time with layers selected adaptively to the needs of particular application.

7.1.1 Contributions

The following contributions to the global multi-resolution topography were achieved throughout the conducted research. The thesis:

- formalized the concept of footprints in the multiple LOD environment,
- shown, how to obtain valid input data for the generalization of footprint's poly-lines,
- proposed simultaneous simplification of poly-lines for the multiple LOD environment, which is aware not to introduce topological errors,
- proposed the primary geometric structure for footprints on multiple LODs and shown, that when this representation is used in conjunction with procedural topographic surface, any data redundancy can be avoided,
- presented two approaches to guide the generalization of features with respect to terrain,
- and demonstrated, how to enhance the quality of terrain approximation on coarser LODs on-demand.

7.2 Future research

In comparison to existing solutions, this work focused on a more functional representation of a multi-resolution topographic surface. The proposed method allowed to populate the bare Earth with spatial objects, which also can influence its shape. It was deliberately designed as general and simple as possible, in order to grant maximum flexibility and functional extensibility in the future. Several extensions of this work suggest themselves.

Generalization operations. A single algorithm for poly-line simplification cannot contain the variability of spatial objects classes and complexity of their relations. However, a more sophisticated behavior of spatial objects can be designed on the grounds of the concept of footprint, which is based on simple poly-lines and the weights of individual points. Moreover, some rules for adjustment of weights and topology between objects modeled independently were added. It was demonstrated through the experiment with the layer of buildings, that a customized generalization solution is needed to achieve appropriate visual results.

1. The aggregation of polygonal areas is one of the known approaches applicable to buildings, when more intense generalization is needed. Several open problems arise, when multiple LOD environment is considered. Can the aggregation be performed across the boundaries between two levels of detail? How to stitch the geometry from two LODs?

   The relation of the result of aggregation to the conditions of surrounding terrain is another issue to be dealt with. Potential problems with aggregated buildings' blocks in a steep terrain were illustrated in Figure 2.2. To evaluate the importance of terrain-feature topological change, the criteria 5.6, 5.7, 5.8 in subsection 5.2.2 were introduced, having such possible applications to generalization operations in mind.

   Open problems mentioned within this paragraph already are a subject of a master thesis in progress, which the author supervises.

2. The collapse of an areal representation of an object to a linear representation (roads, land-use units) is an other example of generalization operation. There are known algorithm, frequently used on the skeleton-based analysis of the feature (54). Such solutions also deal with the split operation over multiple neighboring areas.

   For both operations, the collapse and the above mentioned aggregation, the question of data redundancy should also be considered. Can the generalized geometry be derived procedurally, or such operation would be too time consuming for runtime application and big data?

3D generalization and topological consistency in 3D. The design of methods for generalization of 3D spatial objects for the environment with changing LOD is another open challenge. Dealing with topology becomes even more complicated, when the third dimension is concerned. For example, topological consistency between objects like overhangs, cliffs, balconies or roofs overhangs must guaranteed.

It is also closely related to generalization operations like aggregation of buildings into blocks of buildings, which outcome will be a completely new footprint. In these cases, the topology check on the level of objects' footprints does not have to be sufficient, since objects like bridges spatially exceed their footprint on the terrain in the third dimension. Axioms, that guarantee consistency of 3D city models (at one distinct LOD), were introduced in (48).

Very topical extension of 3D topography is the indoor topography, particularly due to the possible applications, for example in the domain of disaster management. As this field of study is also highly focused on visualization performance, it would be interesting to exploit some ideas of this work for reconsideration of current solutions in favor of data analysis and distributed management.

Dynamic features. The procedural topographic surface, where the topology is reconstructed at run-time, also opened opportunities for representation of dynamic features.

The term dynamic may refer to two aspects. The editing of the surface geometry at run-time is the first one. This includes the updates of geometry of terrain or features, as well as the addition or removal of features. Furthermore, moving the features for the sake of modeling of certain phenomena, e.g. urban planning, which deals with a number of variants for a new road construction, can be implemented over the procedural solution. Such analytical application can provide visual support for the decision, but also the required calculations in relation to the terrain like volumes of moved soil, materials, etc. Moreover, tools for interactive manipulation provided to the user would certainly be an added value.

The second aspect related to dynamic features is the modeling of time. Moving features in the graphic scene according to a timestamp is one thing, which boils down to geometry editing, similar to the cases discussed above. However, the character of 4D spatio-temporal representation is more complex. It is supposed to answer questions like 'Where were you yesterday at 8 p.m.?' or 'Have you ever been in that pub?'.

The possibility to add the time and the level of detail as fully integrated fourth and fifth dimensions should be explored and actually there already are works going in this direction (140).

References

[1] R. Aasgaard. Projecting a regular grid onto a sphere or ellipsoid. In *Symposium on Geospatial Theory, Processing and Applications*, 2002. 21

[2] R. Aasgard and T. Sevaldrud. Distributed handling of level of detail surfaces with binary triangle trees. In *ScanGIS 2001*, Aas, Norway, 2001. 3, 14

[3] A. Agrawal, M. Radhakrishna, and R. Joshi. Geometry-based mapping and rendering of vector data over lod phototextured 3d terrain models. In *Proceedings of WSCG*, 2006. 19

[4] ArcGIS. ArcPy Reference, 2016. Available from: http://pro.arcgis.com/en/pro-app/arcpy/main/arcgis-pro-arcpy-reference.htm [retrieved 2016-06-10]. 104

[5] L. Balmeli, S. Ayer, and M. Vetterli. Efficient algorithms for embedded rendering of terrain models. *Proceedings IEEE International Conference on Image Processing*, pages 914–918, 1998. 29

[6] B. Ben-Moshe, M. J. Katz, J. S.B. Mitchell, and Y. Nir. Visibility preserving terrain simplification— an experimental study. *Computational Geometry*, **28**(2–3):175 – 190, 2004. Special Issue on the 18th Annual Symposium on Computational Geometry - SoCG2002. 68, 74

[7] J. Bentley and T. Ottmann. Algorithms for reporting and counting geometric intersections. *IEEE Trans. Computers*, **28**:643–647, 1979. 46

[8] M. De Berg, M. Van Kreveld, and S. Schirra. Topologically correct subdivision simplification using the bandwidth criterion. *Cartography and Geographic Information Science*, **25**(243–257), 1998. 48

[9] A. Bowyer. Computing dirichlet tesselations. *The Computer Journal*, **24**(2):162–166, 1981. 16

[10] C. Briese. *Breakline modelling from airborne laser scanner.* PhD thesis, Technische Universität Wien, 2004. 18, 77, 78

[11] L. Brůha and J. Kolář. Feature-based enhancement of multi-resolution topographic surface. In *Proceedings of 6th International Conference on Cartography & GIS, Albena, Bulgaria*, 2016. 33, 38, 48, 60

[12] E. Bruneton and F. Neyret. Real-time rendering and editing of vector-based terrains. *EUROGRAPHICS*, **27**(2), 2008. 19

[13] A. Brzank, P. Lohmann, and C. Heipke. Automated extraction of pair wise structure lines using airborne laserscanner data in coastal areas. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, **36**, 2005. 18

[14] G. Camara, A. M. V. Montero, J. A. Paiva, J. Gomes, and L. Velho. Towards a unified framework for geographical data models. In *Geoinfo 2000: Workshop Brasileiro de Geoinformatica, Sao Paulo*, 2000. 11

[15] CGAL. CGAL 4.8.1. Manual, 2016. Available from:
http://doc.cgal.org/latest/Manual/index.html [retrieved 2016-06-10]. 87, 90

[16] L. P. Chew. Constrained delaunay triangulations. In *Proceedings of the Third Annual Symposium on Computational Geometry.*, 1987. 16

[17] P. Cignoni, F. Ganovelli, E. Gobbetti, F. Marton, F. Ponchio, and R. Scopigno. Planet-sized batched dynamic adaptive meshes (p-bdam). In *Visualization, 2003. VIS 2003. IEEE*, pages 147–155, Seattle, WA, USA, 2003. IEEE Computer Society Press. 3, 14, 29

[18] H. Couclelis. People manipulate objects (but cultivate fields): Beyond the raster-vector debate in gis. In *Proceedings of the International Conference GIS - From Space to Territory: Theories and Methods of Spatio-Temporal Reasoning on Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, pages 65–77, 1992. 10

[19] C. Crawford, S. Bayarri, and D. Petrovic. Fast 3-d visualization of large image datasets in a gis. In *ASPRS 2003 Annual Conference Proceedings*, Anchorage, Alaska, USA, 2003. 3, 14

[20] M. Daehlen, M. Fimland, and M. Hjelle. A triangle-based carrier for geographical data. In Peter Halls, editor, *Spatial Information and the Environment,*, pages 105–120. Taylor and Francis Books Ltd, 2001. 19

[21] M. de Berg, M. van Kreveld, and S. Schirra. Topologically correct subdivision simplification using the bandwidth criterion. *Cartography and Geographic Information Science*, **25**:243—257, 1998. 16

[22] L. DeFloriani, P. Magillo, and E. Puppo. Variant: A system for terrain modelling at variable resolution. *Geoinformatica*, **4**(3):287–315, 2000. 14, 29

[23] O. Devillers. Improved incremental randomized delaunay triangulation. In *Proc. 14th Annu. ACM Sympos. Comput. Geom.*, pages 106–115, 1998. 16, 61

[24] O. Devillers. Vertex removal in two dimensional delaunay triangulation: Asymptotic complexity is pointless. Technical Report Research Report 7104, INRIA, 2009. 16, 61

[25] Oxford Dictionaries. Oxford university press, 2016. 3

[26] D. H. Douglas and T. K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica:The International Journal for Geographic Information and Geovisualization*, **10**(2):112–122, 1973. 15, 16, 39

[27] G. H. Dutton. *A hierarchical coordinate system for geoprocessing and cartography.* Spriger-Verlag Berlin Heidelberg, Printed in Germany, 1999. 32

[28] C. Dyken, M. Daehlen, and T. Sevaldrud. Simultaneous curve simplification. *Journal of Geographical Systems*, **10**(2):112–122, 2009. 48, 49, 90

[29] M. J. Egenhofer and A. U. Frank. Object-oriented modeling for gis. *Journal of the Urban and Regional Information Systems Association*, **4**:3–19, 1992. 10

[30] D. Eppstein. Approximating the minimum weight steiner triangulation. *Discrete and Computational Geometry*, **11**(2):163–191, 1994. 19

[31] N. Faust, W. Ribarsky, T. Y. Jiang, and T. Wasilevski. Real-time global data model for the digital earth. In *Proceedings of the International Conference On Discrete Global Grids*, Santa Barbara, California, USA, 2000. 3, 14

[32] J. Ferranti. Aster digital elevation data review. 2016. Available from: http://www.viewfinderpanoramas.org/reviews.html#aster [retrieved 2016-04-20], 2011. 89

[33] S. Filin, N. A. Akel, K. Kremeike, M. Sester, and Y. Doytsher. Interpretation and generalization of 3d landscapes from lidar data. *Cartography and Geoinformation Science*, **34**(3):231–243, 2007. 68

[34] T. Foerster, J. Stoter, and B. Köbben. Towards a formal classification of generalization operators. In *ICC 2007: Proceedings of the 23rd International Cartographic Conference: Cartography for everyone and for you.* International Cartographic Association (ICA), 2007. 15

[35] A. P. Galton. Desiderata for a spatio-temporal geo-ontology. *Spatial Information Theory: Foundations of Geographic Information Science, Springer, Berlin*, pages 1–12, 2003. 11

[36] M. Garland. *Quadric-Based Polygonal Surface Simplification.* PhD thesis, School of Computer Science, Carnegie Mellon University, 1999. 14, 68, 75

[37] M. Garland and P. Heckbert. Fast polygonal approximation of terrain and height fields. Technical report, Carnegie Mellon University, 1995. 14, 66, 68, 69, 70, 71, 72, 89

[38] M. Garland and P. S. Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, pages 209–216, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co. 4, 17

[39] Geofabrik. OpenStreetMap Data Extracts. Geofabrik, 2016. Available from: http://download.geofabrik.de/ [retrieved 2016-04-20]. 86

[40] R. C. Gonzalez and R. E. Woods. *Digital Image Processing (3rd Edition)*. Prentice-Hall, Inc., 2006. 18

[41] M. F. Goodchild. Geographical information science. *International Journal of Geographical Information Systems*, **6**:31—46, 1992. 21

[42] M. F. Goodchild, M. Yuan, and T. J. Cova. Towards a general theory of geographic representation in gis. *Int. J. Geogr. Inf. Sci.*, **21**(3):239–260, January 2007. 11

[43] J. E. Goodman and J. O'Rourke, editors. *Handbook of Discrete and Computational Geometry*. Chapman and Hall/CRC, second edition edition, 2004. 33

[44] A. Gore. The digital earth: understanding our planet in the 21st century. Keynote at Grand Opening Gala of the California Science Center, 1998. 2

[45] T. Götzelmann, R. Guercke, C. Brenner, and M. Sester. Terrain-dependent aggregation of 3d city models. In *Proceedings of the ISPRS/COST Workshop on Quality, Scale and Analysis Aspects of City Models*, 2009. 19

[46] Grifinor. Project Geospatial Internet Unified Geospatial Context for Network Computing and Grifinor project - geospatial internet, 2016. Available from: http://grifinor.net/ [retrieved 2016-06-10]. 89, 100, 113

[47] G. Gröger and L. Plümer. How to get 3-d for the price of 2-d - topology and consistency of 3-d urban gis. *GeoInformatica*, **9**(2):139–158, 2005. 13

[48] G. Gröger and L. Plümer. How to achieve consistency for 3d city models. *Geoinformatica*, **15**:137–165, 2011. 121

[49] G. Gröger and L. Plümer. Topology of surfaces modelling bridges and tunnels in 3d-gis. *Computers, Environment and Urban Systems*, **35**(3):208–216, 2011. 13

[50] D. Grünreich. Atkis – a topographic information system as a basis for gis and digital cartography in germany. In R. Vinken, editor, *From digital map series to geo-information systems, 122 of Geologisches Jahrbuch Reihe A*, pages 207–216, 1992. 25

[51] R. Guercke, T. Götzelmann, C. Brenner, and M. Sester. Aggregation of lod 1 building models as an optimization problem. *ISPRS Journal of Photogrammetry and Remote Sensing*, **66**(209–222), 2011. 20

[52] L. Guibas and J. Stolfi. Primitives for the manipulation of general subdivisions and the computation of voronoi diagrams. *ACT TOG*, **4**(2), 1985. 100

[53] M. Hampe, M. Sester, and L. Harrie. Multiple representation databases to support visualisation on mobile devices. In *Proceedings of the XXth ISPRS Congres, volume 35 of International Archives of Photogrammetry, Remote Sens- ing and Spatial Information Sciences*, pages 135–140, 2004. 4

[54] J. H. Haunert and M. Sester. Area collapse and road centerlines based on straight skeletons. *Geoinformatica*, **12**(2):169–191, 2008. 120

[55] P. S. Heckbert and M. Garland. Survey of polygonal surface simplification algorithms, 1997. 14

[56] M. Helbich, C. Amelunxen, P. Neis, and A. Zipf. Comparative spatial analysis of positional accuracy of openstreetmap and proprietary geodata. In *Proceedings of GI Forum*, 2012. 85

[57] J. Herring. The opengis abstract specification, topic 1: Feature geometry (iso 19107 spatial schema), version 5, 2001. 40

[58] H. Hoppe. Smooth view-dependent level-of-detail control and its application to terrain rendering. *Proceedings IEEE Visualization*, pages 35–42, 1998. 29

[59] L. Huang, M. Meijers, R. Šuba, and P. van Oosterom. Engineering web maps with gradual content zoom based on streaming vector data. *ISPRS Journal of Photogrammetry and Remote Sensing*, **114**:274–293, 2016. 63

[60] Google Inc. S2 geometry library, 2016. Available from: https://code.google.com/archive/p/s2-geometry-library/ [retrieved 20-06-2016]. 21

[61] M. Isenburg, Y. Liu, J. R. Shewchuk, and J. Snoeyink. Streaming computation of delaunay triangulations. *ACM Transactions on Graphics*, **25**(3):1049—1056, 2006. 29

[62] M. Isenburg, Y. Liu, J. R. Shewchuk, J. Snoeyink, and T. Thirion. Generating raster dem from mass points via tin streaming. In *Geographic Information Science—GIScience 2006, Lecture Notes in Computer Science*, **4197**, pages 186–198, Münster, Germany, 2006. 29

[63] O. Devillers J. Boissonnat, M. Teillaud, and M. Yvinec. Triangulations in cgal. In *Proc. 16th Annu. ACM Sympos. Comput. Geom.*, pages 11–18, 2000. 16, 61

[64] J. Jaroš. Break-line modelling source code, 2011. Available from: https://is.cuni.cz/webapps/zzp/download/120056517/?lang=en [retrieved 2016-06-10]. 103

[65] J. Jaroš. *Detekce linií terénní kostry z dat leteckého laserového skenování*. Master's thesis, Přírodovědecká fakulta, Univerzita Karlova v Praze, Available from: https://is.cuni.cz/webapps/zzp/detail/103527/ [retrived 20-06-2016], 2011. 83, 103

[66] J. Jaroš and J. Lysák. Detekce vybraných terénních čar z dat leteckého laserového skenování. *Geodetický a Kartografický Obzor*, **58**(11), 2012. 77

[67] G. F. Jenks. Geographic logic in line generalization. *Cartographica: The International Journal for Geographic Information and Geovisualization*, **26**(1):27–42, 1989. 15

[68] M. Kallmann, H. Bieri, and D. Thalmann. Fully dynamic constrained delaunay triangulations. In *Geometric modeling for scientific visualization*, pages 241–257. Springer, 2004. 40

[69] O. Kersting and J. Dollner. Interactive 3d visualization of vector data in gis. In *ACM international symposium on advances in geographic information systems (GIS)*, pages 107–112, 2002. 19

[70] Y. Kho and M. Garland. User-guided simplification. In *Proceedings of the 2003 Symposium on Interactive 3D Graphics*, I3D '03, pages 123–126. ACM, 2003. 17, 68, 74

[71] E. Kjems, L. Bodum, and J. Kolář. Managed objects for infrastructure data. In J. Lee and S. Zlatanova, editors, *Lecture Notes in Geoinformation and Cartography, 3D Geo-Information Sciences*. Springer, Berlin Heidelberg, 2009. 22

[72] E. Kjems and J. Kolář. From mapping to virtual geography. In S. E. Batty, editor, *CUPUM '05: Computers in Urban Planning and Urban Management : Abstracts of the 9th International Conference*. Center for Advanced Spatial Analysis, University College London, 2005. 30

[73] E. Kjems and J. Kolář. Prototyping a sensor enabled 3d citymodel on geospatial managed objects. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, **II-2/W1**:187–192, 2013. 12, 22, 38

[74] K. Kjenstad. On the integration of object-based models and field-based models in gis. *International Journal of Geographical Information Science*, **20**:491–509, 2006. 11

[75] A. Koch and C. Heipke. Semantically correct 2.5d gis data-the integration of a dtm and topographic vector data. *Developments in Spatial Data Handling*, pages 509–526, 2005. 19

[76] J. Kolář. Global indexing of 3d vector geographic features. In *International Society for Photogrammetry and Remote Sensing 20th Congres*, pages 669–672, Istanbul, Turkey, 2004. 4, 21, 32

[77] J. Kolář. *Global Indexing Of Topographic Surface For 3D Visualization And Analysis*. PhD thesis, Aalborg University, DK, 2006. 4, 14, 20, 21, 23, 29, 30, 31, 32, 33, 34, 35, 36, 55, 61, 68, 117

[78] J. Kolář. Managed code interoperability in digital earth technology. In *Proceedings of the 5th International Symposium on Digital Earth*. University of Berkeley, CA, 2007. 12, 38

[79] J. Kolář. Bytecode unification of geospatial computable models. In *Proceedings of FOSS4G*, **11**, Nottingham, 2013. 11

[80] J. Kolář and P. Ilsoe. Flexible terrain representation using runtime surface reconstruction. In *WSCG Conference*, Pilsen, the Czech Republic, 2005. 30

[81] J. Kolar, E. Kjems, L. Bodum, and E. Sorensen. Global surface model using spaceshuttle radar topographic mission dataset and global indexing grid. In *Strategic Integration of Surveying: Services FIG Working Week 2007*, Hong Kong, 2007. 89

[82] R. Kothuri, A. Godfrind, and E. Beinat. *Pro Oracle Spatial: The essential guide to developing spatially enabled business applications*. Apress, 2004. 11

[83] K. Kraus and N. Pfeifer. Determination of terrain models in wooded areas with airborne laser scanner data. In *ISPRS Journal of photogrammetry and remote sensing*, 1998. 18

[84] K. Kraus and N. Pfeifer. Advanced dtm generation from lidar data. In *International Archives of the Photogrammetry and Remote Sensing*, **3/W4**, pages 23–30, 2001. 18

[85] L. Kulik, M. Duckhamm, and M. Egenhofer. Ontology-driven map generalization. *Journal of visual languages and computing*, **16**:245–267, 2005. 16, 39, 48, 63

[86] H. Ledoux and K. Arroyo Ohori. Edge-matching polygons with a constrained triangulation. In *Proceedings of Symposium GIS Ostrava*, pages 72–81, 2011. 41

[87] H. Ledoux and M. Meijers. Topologically consistent 3d city models obtained by extrusion. *International Journal of Geographical Information Science*, **25**(4):557–554, 2011. 13

[88] X. Li. *A hybrid algorithm for terrain simplification*. PhD thesis, The University of British Columbia, 2003. 73

[89] P. Lindstrom, D. Keller, L. F. Hodges, W. Ribarsky, N. Faust, and G. l'bmer. Real-time, continuous level of detail rendering of height fields. In *SIGGRAPH'96*, 1996. 14

[90] P. Lindstrom, D. Koller, W. Ribarsky, L.F. Hodges, A. Op den Bosch, and N. Faust. An integrated global gis and visual simulation system. Technical Report GVU Technical Report GIT-GVU-97-07, Georgia Institute of Technology, 1998. 3, 14

[91] P. Lindstrom and V. Pascucci. Terrain simplification simplified: A general framework for view-dependent out-of-core visualization. *IEEE Transactions on Visualization and Computer Graphics*, **8**:239–254, 2002. 29

[92] H. Lukatela. A seamless global terrain model in the hipparchus system. In *International Conference On Discrete Global Grids*, Santa Barbara, California, USA, 2000. 21, 32

[93] R. B. McMaster. Automated line generalization. *Cartographica: The International Journal for Geographic Information and Geovisualization,*, 1987. 15

[94] R.B. McMaster and K.S. Shea. *Generalization in Digital Cartography.* Washington, DC: Association of American Geographers, 1992. 15

[95] D. Meagher. Octree encoding: A new technique for the representation, manipulation and display of arbitrary 3-d objects by computer. Technical Report IPL-TR-80-111, Rensselaer Polytechnic Institute, 1980. 31

[96] M. Meijers. Simultaneous & topologically-safe line simplification for a variable-scale planar partition. In S.C.M. Geertman et. al., editor, *LNGC: Advancing Geoinformation Science for a Changing World*, pages 337–358. Springer Verlag, 2011. 48, 63

[97] M. Meijers. *Variable-scale Geo-information.* PhD thesis, Technische Universiteit Delf, 2011. 4, 10, 16, 25, 63

[98] M. Meijers and P. van Oosterom. The space-scale cube: An integrated model for 2d polygonal areas and scale. In *28th Urban Data Management Symposium, International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, **38**, pages 95–102, 2011. 26

[99] A. Meisels, S. Raizman, and A. Karnieli. Skeletonizing a dem into a drainage network. *Computers Geosciences*, **21**(1):187–196, 1995. 18

[100] D. Meyer. Aster global digital elevation model version 2 — summary of validation results. nasa land processes distributed active archive center. Available from: http://www.jspacesystems.or.jp/ersdac/GDEM/ver2Validation/Summary_GDEM2_validation_report_final.pdf [retrieved 2016-04-20], 2011. 89

[101] M. Molenaar. A formal data structure for three dimensional vector maps. In *4th International Symposium on Spatial Data Handling, Zurich*, pages 830–843. Columbus, OH: International Geographical Union IGU, July 1990. 11, 38

[102] NASA. Cobe quadrilateralized spherical cube, 2016. Available from: http://lambda.gsfc.nasa.gov/product/cobe/skymap_info_new.cfm [retrieved 20-06-2016]. 21

[103] P. Neis and A. Zipf. Analyzing the contributor activity of a volunteered geographic information project—the case of openstreetmap. *ISPRS International Journal of Geo-Information*, **1**(2):146–165, 2012. 85

[104] F. J. O'Callaghan and M. D. Mark. The extraction of drainage networks from digital elevation data. *Computer Vision Graphics and Image Processing*, **28**(3):323 – 344, 1984. 18

[105] OpenStreetMap. Wiki. OpenStreetMap Foundation and contributors, 2016. Available from: http://wiki.openstreetmap.org/wiki/Main_Page [retrieved 2016-04-20]. 85, 86

[106] H. Opheim. Fast data reduction of a digitized curve. *Geoprocessing*, pages 33–40, 1982. 15

[107] R Pajarola. Large scale terrain visualization using the restricted quadtree triangulation. *Proceedings IEEE Visualization*, pages 19–26, 1998. 29

[108] F. Penninga. Towards 3d topography using a feature-based integrated tin/ten model. In F. Toppen and M. Painho, editors, *AGILE 2005, 8th Conference on Geographic Information Science*, **25**, pages 243–257, Estoril, Portugal, May 26-28 2005. 13, 38

[109] F. Penninga. *3D Topography: A Simplicial Complex-based Solution in a Spatial DBMS*. PhD thesis, TU Delft, 2008. 12, 13, 22, 38

[110] F. Penninga and P. J. M. Van Oosterom. A simplicial complex-based dbms approach to 3d topographic data modelling. *Int. J. Geogr. Inf. Sci.*, **22**(7):751–779, January 2008. 13, 38

[111] T. Peterková. Point cloud analysis project source code, 2015. Availble from: https://is.cuni.cz/webapps/zzp/download/120182243/?lang=en [retrieved 2016-06-10]. 104

[112] T. Peterková. *Zpřesnění průběhu vodních toků z dat leteckého laserového skenování*. Master's thesis, Univerzita Karlova v Praze, 2015. 78, 79, 81, 82, 104

[113] D. J. Peuquet. A conceptual framework and comparison of spatial data models. *Cartographica: The International Journal for Geographic Information and Geovisualization*, **21**(4):66–113, 1984. 9

[114] M. Pilouk. *Integrated Modelling for 3D GIS*. PhD thesis, ITC Enschede, 1996. 12

[115] J. Puchrik and J. Nýdrle. Snímkování a letecké laserové skenování krkonošského národního parku. *Geodis news*, **12**(1):24, 2013. 102

[116] B. Rabinovich and C. Gotsman. Visualization of large terrains in resource-limited computing environments. *IEEE Visualization*, pages 95—102, 1997. 14, 23, 30, 117

[117] F. H. Razafindrazaka. *Delaunay triangulation algorithm and application to terrain generation*. PhD thesis, United Nations University, 2009. 16, 61

[118] K. Reuman and A.P.M. Witkam. Optimizing curve segmentation in computer graphics. In *International computing symposium, North-Holland*, pages 467–473, 1974. 15

[119] M. Rexer and C. Hirt. Comparison of free high resolution digital elevation data sets (aster gdem2, srtm v2. 1/v4. 1) and validation against accurate heights from the australian national gravity database. *Australian Journal of Earth Sciences*, **61**(2):213–226, 2014. 89

[120] K. Rosina. *Improved method for population disaggregation based on European land monitoring services*. PhD thesis, Slovak Academy of Sciences, 2015. 85

[121] A. Saalfeld. Topologically consistent line simplification with the douglas-peucker algorithm. *Cartography and Geographic Information Science,*, **26**(7–18), 1999. 16

[122] K. Sahr, D. White, and A. J. Kimerling. Geodesic discrete global grid systems. *Cartography and Geographical Information Science*, **30**:121–134, 2003. 21, 32

[123] M. Schneider and R. Klein. Efficient and accurate rendering of vector data on virtual landscapes. In *Proceedings of WSCG*, 2007. 19

[124] J. R. Shewchuk. General-dimensional constrained delaunay and constrained regular triangulations. *Combinatorial Properties*, **39**:580–637, 2008. 16

[125] J. Šimbera. *Modeling population with topographic data*. Master's thesis, Charles University in Prague, 2016. 86

[126] M. Simonse, E. Verbree, P. van Asperen, and J.W. van der Vegt. Construction of the 3dtop10 - integration of countrywide planimetric data and laseraltimetry data to support 3d-visualisation and analysis. In M. Molenaar and K.J. Beek, editors, *XIXth Congress ISPRS 2000, Amsterdam, XXXIII of International Archives of Photogrammetry and Remote Sensing*, pages 995—1002, 2000. 13

[127] D. Sinton. The inherent structure of information as a constraint to analysis: mapped thematic data as a case study. *Harvard Papers on Geographic Information Systems*, 1978. 10

[128] J. Skála. *Algorithms for manipulating large geometric data*. PhD thesis, Západočeská univerzita v Plzni, Fakulta aplikovaných věd, 2012. 20

[129] SRTM. Shuttle Radar Topography Mission (SRTM) 1 Arc-Second Global. United States Geological Survey, 2016. Available from: https://lta.cr.usgs.gov/SRTM1Arc [retrieved 2016-04-21]. 88

[130] SRTM. Data Processing Methodology. The CGIAR Consortium for Spatial Information, 2016. Available from: http://srtm.csi.cgiar.org/SRTMdataProcessingMethodology.asp [retrieved 2016- 04-20]. 89

[131] J. Stoter, M. Meijers, P. van Oosterom, D. Grunreich, and M. Kraak. Applying dlm and dcm concepts in a multi-scale data environment. In B. Buttenfield, editor, *Proceedings of GDI 2010: symposium on generalization and data integration*, 2010. 25

[132] J. Stoter, F. Penninga, and P. van Oosterom. Generalization of integrated terrain elevation and 2d object models. In *11th International Symposium on Spatial Data Handling: Developments in spatial data handling*, 2005. 19

[133] G. Strang. *Calculus*. Wellesley-Cambridge Press, 1991. 77

[134] A. Szofran. Global terrain technology for flight simulation. In *Game Developers Conference*, 2006. 19

[135] C. Thieulot. Douar. 2016. Available from: http://www.cedricthieulot.net/douar.html [retrieved 20-06-2016]. 31

[136] W. Tobler and Z. Chen. A quadtree for global information storage. *Geographical Analysis*, **18**(4):360—371, 1986. 21

[137] R.O. Tse and C. Gold. Tin meets cad - extending the tin concept in gis. *Future Generation Computer systems (Geocomputation)*, **20**:1171—1184, 2004. 13

[138] P. van Oosterom. Variable-scale topological data structures suitable for progressive data transfer: The gap-face tree and gap-edge forest. *Cartography and Geographic Information Science*, **32**:331—346, 2005. 4

[139] P. van Oosterom, M. Meijers, J. Stoter, and R. Šuba. *Abstracting Geographic Information in a Data Rich World*, chapter Data Structures for Continuous Generalisation: tGAP and SSC. Springer International Publishing Switzerland, 2014. 63

[140] P. van Oosterom and J. Stoter. *5D Data Modelling: Full Integration of 2D/3D Space, Time and Scale Dimensions*, pages 310–324. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. 122

[141] M. Visvalingam and J. D. Whyatt. Line generalisation by repeated elimination of points. *The Cartographic Journal*, **30**(1):46–51, 1993. 48

[142] J. S. Vitter. External memory algorithms and data structures: Dealing with massive data. *ACM Computing Surveys*, **33**(2):209–271, 2001. 29

[143] VTK. The Visualization Toolkit. Kitware, 2016. Available from: http://www.vtk.org/ [retrieved 2016-06-10]. 89

[144] Z. Wartell, E. Kang, T. Wasilewski, W. Ribarsky, and N. Faust. Rendering vector data over global, multi-resolution 3d terrain. In *Proceeding VISSYM '03 Proceedings of the symposium on Data visualisation 2003*, pages 213–222, 2003. 19

[145] D. F. Watson. Computing the n-dimensional delaunay tessellation with application to voronoi polytopes. *The Computer Journal*, **24**(2):167—172, 1981. 16

[146] A. Weber and J. Benner. Interactive generation of digital terrain models using multiple data sources. In *First International Symposium on Digital Earth Moving*, pages 60–64. Springer-Verlag, 2001. 19

[147] W. Wen, E. Kjems, L. Bodum, and J. Kolář. Dynamic features in a 3d city model as an energy system. In *ISPRS Conference: International Conference on 3D Geoinformation*, Berlin, Germany, 2010. 22

[148] M. F. Worboys and M. Duckham. *GIS: A Computing Perspective*. CRC Press, Inc., Boca Raton, FL, 2nd edition edition, 2004. 10, 11

Curriculum vitae

Lukáš Brůha was born on April 28, 1984 in Klatovy. In 2003 he obtained his high school diploma. After this, he studied Geography and Cartography at the Charles University in Prague and obtained the Bachelor's degree. He continued his studies on the same university at the Department of Applied Geoinformatics and Cartography and got the Master's degree in 2008. Afterwards, he was introduced by Jan Kolář into the state-of-the-art research of digital Earth technologies, which motivated him to start the doctoral studies under the supervision of Dr. Tomáš Bayer at the same Department of Applied Geoinformatics and Cartography and in cooperation with 3DGI group at Aalborg university. During the years 2009 and 2010, he was granted the opportunity of an internship at the Aalborg university. Moreover, he could present several research ideas and partial results on several workshops, thanks to the participation of 3DGI group in the InfraWorld project. In 2011, he has become a research assistant at the Department of Applied Geoinformatics and Cartography. Since then, he participated in several research projects of the department. That enriched himself of many experiences from different research domains spanning from historical cartography, metadata catalogues, geospatial web technologies, or LiDAR data analyses. As well, from these reasons, the doctoral work, rooted in an amazing world of digital Earth technologies, took more time, than originally expected.

Reviewed papers:

L. Brůha a J. Kolář. Feature-based enhancement of multi-resolution topographic surface. In *Proceedings of the 6th International Conference on Cartography & GIS*, Albena, Bulgaria, 2016.

L. Brůha. Large geospatial images discovery: metadata model and technological framework. *Geoinformatics FCE CTU*, 14(2):21-36, 2015.

L. Brůha. Automation of geospatial raster data analysis and metadata updating: in-database approach. *AUC Geographica*, 49(2), 2014.

Certified methodology:

L. Brůha. Certifikovaná metodika pro tvorbu metadat kartografických dokumentů. Prague: Faculty of Science, 2015. Available from: http://www.nusl.cz/ntk/nusl-188893 [retrieved 2016-07-07].

Software:

MTDTRasPub, 2014. Software pro správu, vyhledávání, publikaci a on-line prezentaci kartografických dokumentů. RIV/00216208:11310/14:10274844. Available from: http://web.natur.cuni.cz/gis/temap/index.php/sprava-map [retrieved 2016-07-07].

Participation in research projects:

INASAMP (TAČR) - Inovativní nástroje pro automatizovanou správu, aktualizace a korekce mapových podkladů nejen pro navigační systémy, Program na podporu aplikovaného výzkumu a experimentálního vývoje EPSILON, 2015 - 2017.

Moderní geoinformační metody ve výuce GIS a kartografie, Univerzita Karlova v Praze. 2015 - 2017.

TEMAP - Technologie pro zpřístupnění mapových sbírek ČR: metodika a software pro ochranu a využití kartografických děl národního kartografického dědictví je projektem Programu aplikovaného výzkumu a vývoje národní a kulturní identity NAKI - identifikační kód DF11P01OVV003. Poskytovatelem finančních prostředků je Ministerstvo kultury. 2011 - 2015.