

Modeling Terrains and Subsurface Geology

M. Natali^{†1} E.M. Lidal¹ J. Parulek¹ I. Viola^{3,1} D. Patel^{2,1}

¹University of Bergen, Norway

²Christian Michelsen Research, Bergen, Norway

³Vienna University of Technology, Austria

Abstract

The process of creating terrain and landscape models is important in a variety of computer graphics and visualization applications, from films and computer games, via flight simulators and landscape planning, to scientific visualization and subsurface modelling. Interestingly, the modelling techniques used in this large range of application areas have started to meet in the last years. In this state-of-the-art report, we present two taxonomies of different modelling methods. Firstly we present a data oriented taxonomy, where we divide modelling into three different scenarios: the data-free, the sparse-data and the dense-data scenario. Then we present a workflow oriented taxonomy, where we divide modelling into the separate stages necessary for creating a geological model. We start the report by showing that the new trends in geological modelling are approaching the modelling methods that have been developed in computer graphics. We then give an introduction to the process of geological modelling followed by our two taxonomies with descriptions and comparisons of selected methods. Finally we discuss the challenges and trends in geological modelling.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object modelling—Curve, surface, solid, and object representations

1. Introduction

Realistic appearance of natural sceneries has been a key topic in computer graphics for many years. The outcome of this research primarily targets the film and gaming industries. The modelling is often procedural and can be constructed with little user intervention, at interactive frame-rates. In most cases, only the top surface is the final product of the modelling process, even if subsurface features have been taken into account during the modelling.

Parallel to this development, the modelling of geological structures has been developed from the geological domain. This modelling process usually requires heavy user involvement and substantial domain knowledge. The model creation can often take up to one year of intensive work. The modelling process also includes data acquisition from the site which is to be modelled. The resulting model is usually a very complex 3D structure, consisting of a number of different subsurface structures.

The needs of the entertainment industry and the geoscientific domain are substantially different, although they represent similar natural phenomena. While the former one puts emphasis on interactive realistic visual appearance, the latter one focuses on the correctness from the geoscientific point of view. In recent years, research in geosciences has identified the importance of rapid prospects, i.e., fast geoscientific interpretations at early stages in exploration. For such use, the extensive development period of a typical geological model becomes a severe limitation. Rapid prospects raise a need for rapid modelling approaches that are common practice in computer graphics terrain modelling. This is one of the reasons why geoscientific modelling techniques are approaching traditional terrain modelling approaches.

One essential difference still remains: While the terrain synthesis for entertaining industries is a product of content creation carried out by artists, the geoscientific models are created based on actual measurements and are done by geologists and other geoscientists based on a substantial level of background knowledge and expertise. Moreover, when modelling based on measurements, the input data are either densely covering a certain spatial area, for instance by

[†] mattia.natali@uib.no

means of large-scale acoustic surveys, or consist of sparse samples that are completed by extrapolating known values over the areas where no measurements are taken. There is a multitude of approaches to collect geological information, for instance from seismic surveys (2D, 3D, 4D), boreholes (1D), virtual outcrops from LIDAR scans (3D), or vertical outcrop analysis (2D) [PHH*06]. It is outside the scope of this report to provide in-depth information on the acquisition processes. However, the nature of these measurements strongly influences the modelling approach from the geoscientific perspective. The aim of this report is to assess different approaches for modelling geological structures, compare methods from the computer graphics and geoscience domains and to suggest promising topics for future research agendas.

It is important to note that only a part of the research in geomodelling is of academic origin and publicly available. There is a substantial amount of geological modelling research available only in the form of ready-to-use tools in commercial software packages. This research is, unfortunately for this report, protected by commercial vendors and their algorithmic details are often not known to academia. Therefore, in this report we focus on describing those geomodelling methods that have been made publicly available. The aim is not to describe the functionality of commercial modelling packages or attempt to reverse-engineer the methodology behind them.

We firstly provide, in Section 2, a light-weight background on essential properties of geo-bodies as compared to man-manufactured objects typically modelled by means of Computer-Aided Design (CAD). After becoming familiar with essential background knowledge, we discuss our proposed taxonomies and highlight principal differences in Section 3 and Section 4. High-level distinctions between a computer graphics approach and a domain science workflow are reviewed. The aim is to give the interested reader a notion of possible future integrative tendencies between these two fields by means of mutual adaptation of methodologies typical for one or the other domain. Section 5 consists of a comparison of selected techniques for modelling surfaces and for modelling solids. Finally in Section 6 we suggest possible developments of computer graphics in geological applications.

2. Geological Elements

The study of structural geology divides the subsurface into geo-bodies [Hou94] of different categories. Central objects are layers, horizons, faults, folds, channels, deltas, salt domes and igneous intrusions.

Much of the reviewed work shows how to represent geological feature such as horizons, folds, faults and deposition. Deposition occurs when eroded particles of rock are brought by wind, water or gravity to a different place, where

they accumulate to form a new rock layer. The subsurface is composed of a set of layers with distinct material composition. The surface which delimits two adjacent layers is known as a *horizon*. Two fundamental geological phenomena involve modification of the original structure of horizons: the process of folding and the process of faulting. A fold is obtained when elastic layers of rock are compressed. It is defined as a permanent deformation of an originally flat layer that has been bent by forces acting in the crust of the Earth. Faults originate when forces acting on layers are so strong that they overcome the rock's elasticity and yield a fracture. Horizons are thereby displaced and become discontinuous across a fault. Channels are underground fluvial paths that either directly originate under the ground or are old rivers went through a sedimentation process that has created a new layer on top of them. Geological models can be divided into two different categories, *layer-based models* and *complex terrain models* [Tur06]. The layer-based models, built of multiple horizontal oriented surfaces, are typically created to model sedimentary geological environments during ground-water mapping, or oil and gas exploration. In regions with complex geological structures or where the layering is not dominant, for instance when modelling igneous and metamorphic terrains, a more complex terrain model is needed. These complex terrains are modelled when exploring for metal and mineral resources. In this report, we focus on the layer-based models as they are spatially more well defined and share similarities with terrain modelling in computer graphics.

In two viewpoint articles, Turner [Tur06, TG07] provides a thorough introduction to the challenges of creating computer tools for modelling and visualization of geological models. The article formulates the essential domain needs and the capability to interactively model and visualize: geometry of rock and time-stratigraphic units; spatial and temporal relationships between geo-objects; variation in internal composition of geo-objects; displacement and distortions by tectonic forces; and the fluid flow through rock units.

Furthermore the following characteristics of the geo-bodies are highlighted: complex geometry and topology, scale dependency and hierarchical relationships, indistinct boundaries defined by complex spatial variations, and the intrinsic property heterogeneity and anisotropy of most subsurface features. These characteristics are, according to Turner, not possible to satisfy with traditional CAD-based modelling tools. Thus, dedicated geological modelling and visualization tools are necessary.

In geological modelling there are often scenarios that lack sufficient data, so in order to build a meaningful model, the creator must interpolate between the sparse sampled or derived data available. Traditional interpolation schemes for discrete signal reconstruction are not sufficient as the process needs to be guided by geological knowledge, often through many iterations, to produce a successful result. A plausible

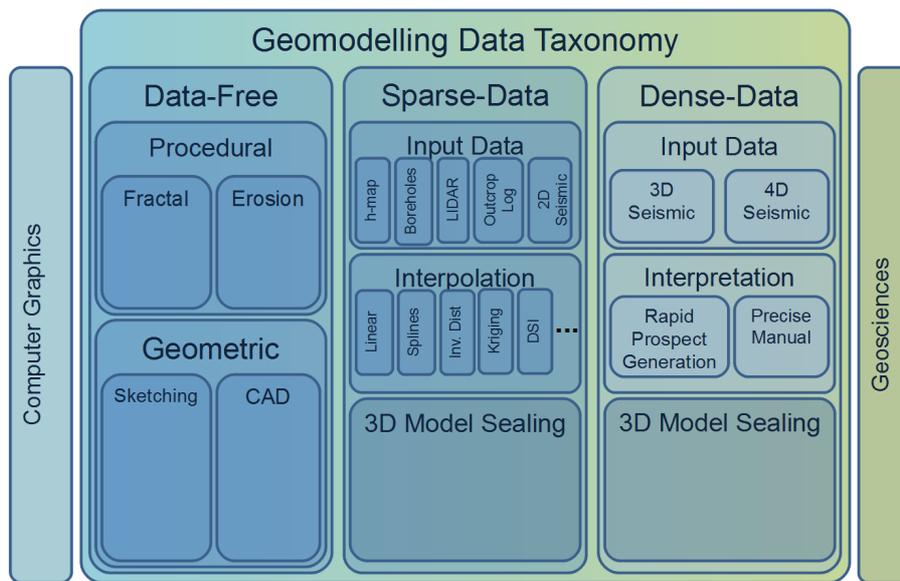


Figure 1: Several methods are classified according to which type of data they use and from which domain they originate (geosciences or computer graphics).

geological scenario has to follow certain geo-physical constraints. Caumon et al. [CCDLCdV*09] describe specific structural modelling rules for geological surfaces defining boundaries between different lithological layers. Geo-bodies exhibit spatial continuity, therefore abrupt geometric variations such as sudden change of normal orientation on the surface, and abrupt changes within a fault are not common. This implies that a structural model may be validated via reconstructing its depositional state. Caumon et al. also describe the typical process of creating a structural model. The modelling usually starts with fault modelling. The mesh can either be produced directly as triangle strips, based on the dip information or indirectly using a specific interpolation scheme. The second and most important step is to define the connectivity among fault surfaces. The last step is the horizon modelling.

In Section 3 we will briefly categorize previous works in terms of the type of data that is being addressed (see Figure 1) followed by an in depth categorization of papers according to where they fit in the workflow of geomodelling (see Figure 2) in Section 4.

3. Geomodelling Data Taxonomy

By comparing the outcome of computer graphics terrain modelling with geoscientific modelling, we can roughly divide modelling into three distinct categories: the *data-free*, *sparse-data*, and *dense-data* scenario. The first category represents current and future trends in rapid modelling, where current methodology originates from the computer graph-

ics research, while the latter two categories are developed from explicit needs in the geoscientific domain. This categorization forms one of the high-level taxonomies of the discussed modelling approaches. Figure 1 shows the taxonomy together with different modelling scenarios.

The *data-free* scenario has no ground truth information and therefore the geometric synthesis relies entirely on *procedural* [Ols04, PGGM09b] and *geometric* modelling. The typical computer graphics research agenda proposes methodologies that alleviate the user from labor-intensive tasks by automating parts of the modelling. Procedural modelling offers the modeller specific, easy to handle, input parameters which control the process of geometry generation. The geometry typically represents terrain surfaces. Procedural techniques in modelling has been facilitated mainly through *fractal* modelling [SBW06, Bel07]. In the dynamic case, simple *erosion* models [KBKv09, HGA*10] are utilized to create dynamic and realistic landscapes [PHT93, HD11].

The shortcoming of procedural modelling is usually the lack of direct control over the landscape development. The modeller has a rough idea of the landscape, but implicit parameter settings do not guarantee a match with the modeller's idea of an intended shape. Therefore a combination of explicit geometric modelling, to represent the modeller's expectations, with procedural modelling, to add realism, is a preferred strategy. On the other hand, geometric modelling can be a labor-intensive task. For rapid modelling scenarios, various forms of *sketching* metaphors [GMS09, PGGM09a, VMc*10] or modelling by example [BSS07] provide fast

ways to express the rough structure of a terrain or of a stratigraphic model [NVP12, LNP*13].

The *sparse-data* modelling scenario is the most frequently seen in the geoscience domain [Gro99]. Very often it contains networks of boreholes [LJ03, KM08, MP09], where the data needs to be interpolated between. Besides boreholes, there are often other acquisition types available, such as surface elevation models [DLC05, Zha05], obtained through the process of remote sensing. This heterogeneous pool of geoscientific data raises the challenge of data integration and data interpolation. The main interpolating methods are the B-Spline method, inverse distance method, Kriging and discrete smooth interpolation [Mal92, Mal97]. They will be discussed in Section 4.2.3. The implicit function interpolator is another increasingly popular interpolation method [MGC*07]. Turner [Tur06] demonstrates how to build a typical geological model from a *sparse-data* scenario by firstly interpreting bore-hole logs to construct triangulated surfaces of horizons [BBZ09] and then create the geo-bodies [DS03, SdK05] in sealed, boundary-representations [CLSM04] of the volumes between these surfaces. The modelling of faults [Wu03] is also very important and Turner describes the challenge of modelling the interface between the boundary representation and the fault to avoid an unwanted crossing or empty spaces between the fault and geo-bodies. Using a structured mesh representation of the boundary surface can result in discretization errors, while using unstructured grid representation [Fra06] adds computational complexity and results in slow model construction.

The *dense-data* scenario is typically based on a single- or multi-attribute volumetric seismic dataset. The first challenge is purely of computational character, i.e., how to interactively display huge amounts of volumetric data, addressed, e.g., in the work of Plate et al. [PTCF02]. Utilizing volume rendering concepts, these datasets can be displayed without prior extraction of geo-bodies. Extracting geological structures from this data is necessary for consecutive steps along the workflow, such as reservoir modelling. This process is known as geoscientific interpretation and is a very time-demanding task. Typically the original seismic dataset, consisting of the amplitudes of reflected sound waves, can be used to extract a number of derived attributes. These attributes are not geo-bodies, but their distribution over the 3D domain indicates the presence of certain geological structures. The *SHIVR* interpretation system [LLG*07] can extract geo-bodies based on scatter plots, as shown by Andersen and van Wijngaarden [Av07]. Rapid prospect generation can certainly benefit from faster interpretation. Patel et al. proposed methods for rapid horizon extraction in two [PGT*08] and three dimensions [PBVG10]. Afterwards, once the interpretation is available, 3D visualization can assist in validating the correctness of the extracted horizons with respect to original or derived attribute data [PGTG07].

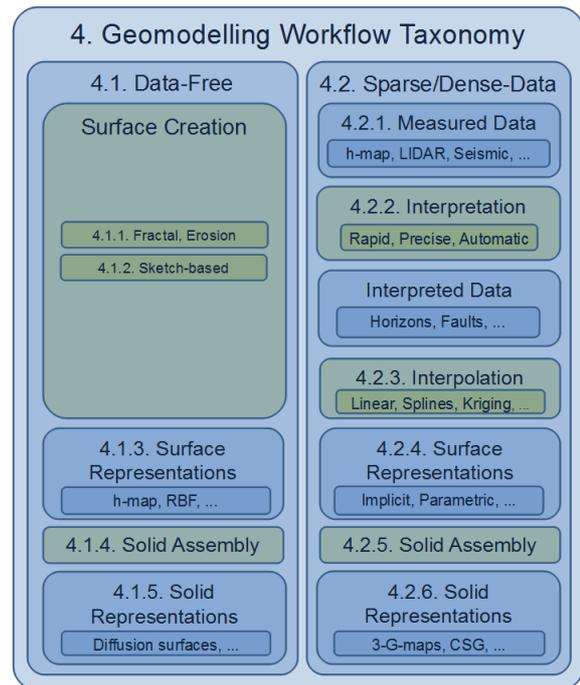


Figure 2: Workflow taxonomy. Blue boxes represent data and green boxes represent action on the data to create richer data. The smaller boxes inside show examples of methods for that subtopic.

A natural next step after 3D structural modelling is the development of a time-varying structural model. Inverse methods are often utilized in geomodelling to restore hypothetical geological scenario by going backwards in time [Wij03, GCC*08, Cau10]. Such an approach aims at restoration of deposited sedimentary layers, for example through unfolding. Restored information about palaeogeography often gives good indication where to search for hydrocarbon reservoirs.

4. Geomodelling Workflow Taxonomy

In Figure 1, methods are categorized according to which domain they arise from, which is also tightly correlated with the amount of measured geologic data they handle. However for the rest of the paper we have found it more appropriate to describe methods in the order they would be applied in a workflow for creating a geological model. Such a taxonomy is shown in Figure 2. We have defined two separate workflows, one for creating models from no data, and one for creating models when data exists. The steps in each workflow are aligned with each other and we have specified in front of each category which chapter they are described in. In the workflow where no measured data is required as input (data-free), some papers focus on surface creation, other

general papers describe different mathematical surface representations. Several sketch-based papers describe different ways of fast sketching and assembling of solid objects and some focus on compact representations and fast rendering of complex solid objects. For the case where data is being used (the sparse/dense-data column), the workflow begins with measuring data, interpreting relevant structures, interpolating these into higher-order objects and representing these in some appropriate mathematical way. The structures are then assembled into solid geometry describing the subsurface. We have devoted a subsection to each of the topics described in the geomodelling workflow taxonomy.

4.1. Data-Free

This section describes works that do not rely on any measured or sampled input data, and where the models are created from scratch, driven by imagination or concept ideas and domain knowledge.

4.1.1. Fractal and Erosion Surface Creation

There are usually three approaches to generate synthetic terrains: fractal landscape modelling, physical erosion simulation and terrain synthesis from images or sample terrain patches. Before the work by Olsen [Ols04], it was mostly possible to use simple fractal noise to obtain terrain surfaces, because computers were not fast enough to simulate erosion processes in real-time. Olsen proposes a synthesized fractal terrain and applies an erosion algorithm to this. His representation of terrains is a two-dimensional height-map. To simulate erosion, he considers the terrain slope as one of the main parameters: a high slope results in more erosion, a low value produces less erosion. Starting from a noisy surface, called the *base surface*, erosion occurs to simulate weathering on a terrain. He applies two types of erosion algorithms to the base terrain: thermal and hydraulic. After testing the two erosion methods, he decides to combine the advantages of each, namely, the rapidity of the thermal erosion and the good erosion approximation of the hydraulic erosion.

The ability to model and render piles of rocks without repetitive patterns is one of the achievements of the work by Peytavie et al. [PGGM09b]. They focus on rocks and stones, which are found everywhere in landscapes. They provide realism to the scene, reveal characteristics of the environment and hint on its age. Before this paper, the canonical way of generating rocks was to produce a few models by artists, which were then instantiated in the scene. To create piles of rocks, collision detection techniques were applied with a high computational cost and low control. The authors propose aperiodic tiling of stones to avoid repetitive patterns. Two steps define the method proposed in the paper: the first is a preprocessing step that generates a set of aperiodic tiles constrained to maintain contact between neighbours; subsequently, the rock piles are created by exploiting aperiodic tiles from the previous step. Voronoi cells are employed to

control the shape of rocks. In the construction of the Voronoi cells, an anisotropic distance to avoid round-shaped stones is used. Finally, erosion is applied to produce the final model. Meshes of the rocks are represented by using standard implicit surface meshing techniques.

Musgrave et al. [MKM89] describe the creation of fractal terrain models, avoiding global smoothness and symmetry; these two drawbacks arise from the employment of the first definition of fractional Brownian motion (fBm) as introduced by Mandelbrot [Man82]. Moreover, there is a second stage in which the surface undergoes an approximation of a physical erosion process. Terrain patches are represented as height-maps and the erosion process is subdivided into a thermal and a hydraulic part.

Concerning modelling terrains with rivers, Sapozhnikov et al. state that at the time when their paper [SN93] was written it was impossible to simulate the process of natural river network formation without making a substantial approximation; i.e. a simpler model that does not make use of the physical laws, but nevertheless reproduces the main geometrical features of a real river network. They use a random walk method to generate a set of river networks of various sizes.

Stachniak et al. [SS05] point out that fractal methods have been used to create terrain models, but these techniques do not give much control to the user. They try to overcome this by imposing constraints to the original randomly created model, according to the user's wishes. The method requires two inputs: the initial fractal approximation of the terrain and a function that incorporates the constraints to be satisfied in order to achieve the final shape. As an example, they show how to adapt a fractal terrain to accommodate an S-shaped flat region, representing a road. The constraint function defines a measure that indicates how close a terrain is to the desired shape. The final solution is provided by a minimization of the difference from the current terrain to the desired one.

Another way of combining procedural modelling with user constraints is described by Doran and Parberry [DP10]. In their work, they procedurally generate terrain elevation height-maps, taking into consideration input properties defined by the user. The model lets the user choose amongst five terrain tools: coastline, smoothing, beach, mountain and a river tool. Together, these tools can be used to generate various types of landscapes.

A terrain surface is created by fractal noise synthesis in Schneider et al.'s work [SBW06]. They aim to solve the problem that was one of the biggest disadvantages in fractal terrain generation at the time, namely the setting of parameters. They reduce such an unintuitive process of setting parameters by presenting an interactive fractal landscape synthesizer.

Roudier et al. [RPP93] propose a method for terrain evolution in landscape synthesis. Starting from an initial topo-

graphic surface, given by a height-map, they subsequently apply an erosion process to obtain the final 3D model. The erosion consists of mechanical erosion, chemical dissolution and alluvial deposition.

Chiba et al. [CM98] propose a method that overcomes the limitation of previous techniques for generating realistic terrains through fractal-based algorithms, but lack ease of handling, i.e. it is not possible to modify the surface on the basis of the user's suggestions. Such as introducing ridge or valley lines, which are usually important to characterize a mountain scenery. The topology of the landscape is created by a quasi-physically based method, that produces erosion by taking velocity fields of water flow into consideration. The whole process of erosion, transportation and deposition is derived on the basis of the velocity field. Dorsey et al. [DEJ*99] focus on erosion applied to one stone or rock, represented by its volume, taking into consideration weathering effects on it.

In the work by Benes et al. [BF01], a method for eroding terrains is described. A concise version of a voxel representation is utilized, and thermal weathering is simulated to erode the initial model. This new way to represent terrains has the advantage of being able to store caves and holes. When applying erosion, all the layers and ceilings of the caves are involved in the process.

In a subsequent paper [BF02], a technique for procedural modelling of terrains through hydraulic erosion is introduced. The purpose is to use a physically-based approach together with a high level of control. Contrary to previous techniques which tend to oscillate during water transportation, they provide a tool for hydraulic erosion that is fast and stable. They overcome oscillation by relying more on physical constraints than was previously the case. The erosion process consists of four independent steps, where each step can run repeatedly and in any order. These four steps are: introduction of new water (simulation of rain); material capture by water (erosion); transportation of material; and deposition at a different location.

Another work by Benes et al. [BTHB06] applies a hydraulic erosion fully based on fluid mechanics and thus on the Navier-Stokes equations that describe the dynamics of their studied models. They use a 3D representation provided by a voxel grid and the erosion process leads to a model that can show a static scene or part of an animation illustrating the terrain morphology. At each iteration of the process of erosion, a solution to the Navier-Stokes equations is computed to determine a pressure and velocity field in the voxels.

Interactive physically-based erosion is employed by Stava et al. [SBBK08] (Figure 3). This work is based on physics by making use of hydraulic erosion, and on interactivity, which allows the user to take an active part during the generation of the terrain. The technique is implemented on the GPU and,

because of the limited GPU memory, the terrain is subdivided into tiles, which allow then to apply erosion to local areas. Each terrain-tile is represented as a height-map.

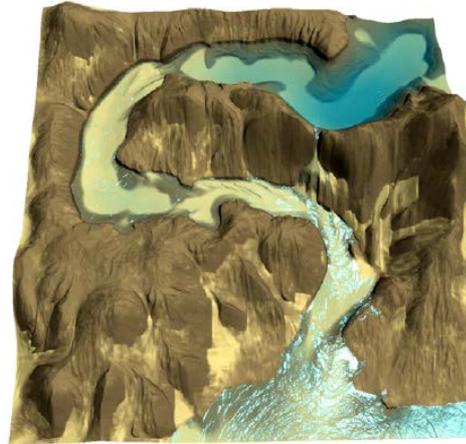


Figure 3: The eroded terrain is obtained by simulating the movement of the water flow and transportation of rock particles [SBBK08].

Kristof et al. [KBKv09] adopt 3D terrain modelling through hydraulic erosion obtained by fluid simulation using a Lagrangian approach. Smoothed Particle Hydrodynamics (SPH) [GM77, Luc77] is employed in this paper to solve dynamics that generate erosion. SPH requires low memory consumption, it acts locally, works for 3D features and is fast enough to work on large terrains.

For Hnaidi et al. [HGA*10], the terrain is generated from some initial parametrized curves which express features of the wished terrain (see Figure 4). Each curve is enriched with different types of properties (such as elevation and slope angle) that become constraints during the modelling process.

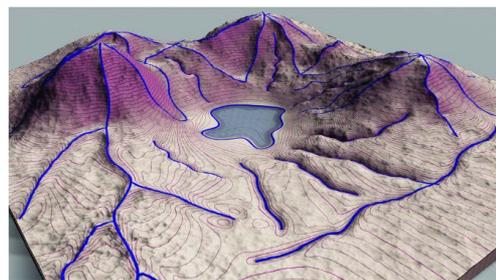


Figure 4: Sketches, that are visible in the figure as blue strokes, work as constraints during the method proposed by Hnaidi et al. [HGA*10].

Prusinkiewicz and Hammel [PHT93] address the problem of generating fractal mountain landscape, which also

includes rivers. They do it by combining a midpoint-displacement method for the generation of mountains with a method to define river paths.

Hudak and Durikovic [HD11] tackle the problem of simulating terrain erosion over a long time period. They use a particle system and take into consideration that terrain particles can contain water. Discrete Element Method (DEM) is used for the simulation of the soil material and Smoothed Particle Hydrodynamics (SPH) simulates water particles.

Instead of procedural and erosion synthesis, Brosz et al.'s paper [BSS07] introduces a way to create realistic terrains from reference examples. This process is faster than starting the terrain generation from scratch. Two types of terrains are necessary to obtain the final one: a base terrain, used as a rough estimate of the result, and the target terrain that contains small-scale characteristics that the user wants to include in the reconstruction. Brosz et al. have two common ways to generate landscapes represented as a height-map: using brush operations to bring some predefined information or action on the surface; alternatively, simulation and procedural synthesis can be applied to obtain a realistic terrain. One drawback of using simulation is that it can be slow, while in the case of procedural synthesis, expressability is reduced by a limited set of parameters. De Carpentier and Bidarra try to combine brushing and procedural synthesis in their work [dCB09] (an example is shown in Figure 5).

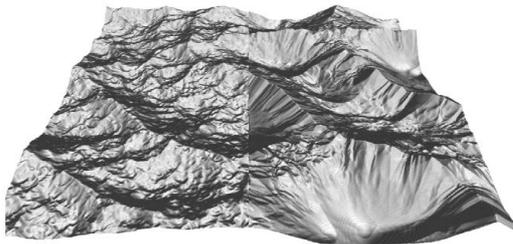


Figure 5: Two types of noise applied by de Carpentier and Bidarra [dCB09] to achieve a realistic terrain.

4.1.2. Sketch-based Surface Creation

Gain et al. present a paper [GMSe09] that describes procedural terrain generation with a sketching interface. Their approach aims to gather benefits and overcome some limitations of previous methods of sketch-based terrain modelling [CHZ00, WI04, ZSTR07]. Watanabe and Igarashi [WI04] employ straight lines and, even though they yield a boundary for landforms using local minima and maxima of the user's sketch, they do not give the user the possibility to change the proposed shape (see Figure 6). Furthermore, they apply noise onto the terrain after surface deformation, hence the obtained surface does not interpolate the user strokes exactly. Whereas landforms rarely follow straight lines, Zhou et al. [ZSTR07] allow landforms to have

more free-shape paths using a height-map sketching technique as guidance for an example-based texture synthesis of terrain. In contrast to the method suggested by Gain et al. [GMSe09], they provide low and indirect control over the height and boundary of the resulting landform.

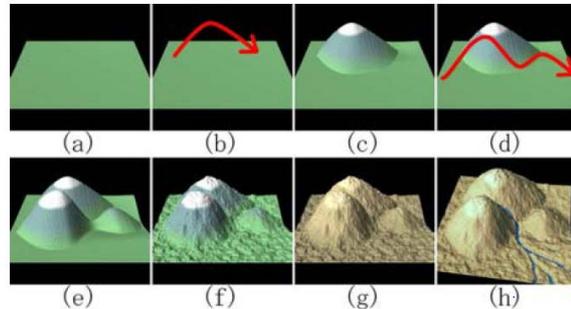


Figure 6: Watanabe and Igarashi's process for obtaining a landscape surface by sketching [WI04].

Brazil et al. [VMc*10] introduce a sketch-based technique to generate general 3D closed objects using implicit functions. They also show how to exploit their tool to obtain simple geological landscapes from few user strokes, see Figure 7.

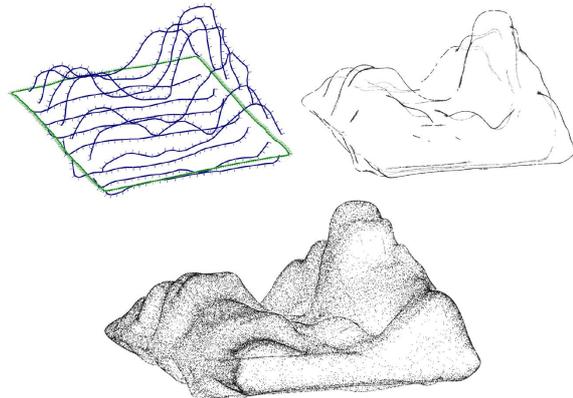


Figure 7: Landscape example generated with a sketch-based approach by Brazil et al. [VMc*10]. The model is represented with Hermite Radial Basis Functions.

4.1.3. Surface Representations

Several of the fractal and erosional surface creation methods represent the surfaces as height-maps. This is an easy-to-maintain datastructure which fits well with erosional calculations. The method by Brazil et al. [VMc*10] can represent complex surfaces with overhangs or closed objects, using implicit functions defined as a sum of radial basis functions. Based on points with normals as input, a smooth implicit

function, interpolating the points while being orthogonal to the normals, is created.

Peytavie et al. [PGGM09a] represent complex terrains with overhangs, arches and caves. They achieve this by combining a discrete volumetric representation, which stores different kinds of material, with an implicit representation for the modelling and reconstruction of the model.

Bernhardt et al. [BMV*11] presented a sketch-based modelling tool to build complex and high-resolution terrains, as shown in Figure 8. They achieve real-time terrain modelling by exploiting both CPU and GPU calculations. To represent large terrains, they use an adaptive quadtree data structure which is tessellated on the GPU.

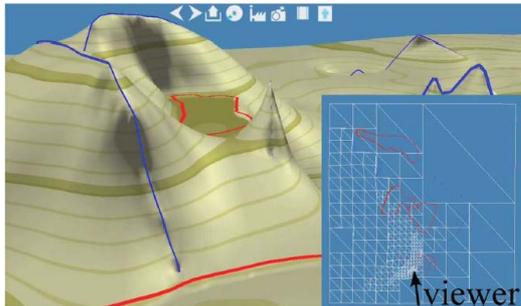


Figure 8: Surface modelling with the sketch-based tool proposed by Bernhardt et al. [BMV*11].

4.1.4. Solid Assembly

By solid assembly, we refer to the process of assembling boundary surfaces or basic solid building blocks into a complete solid object. This work process is supported by CAD based tools. An example of sketch based solid assembly of geological layer-cake models was presented in the work by Natali et al. [NVP12]. Here they describe how to obtain a boundary representation of a solid model with the use of a sketch-based technique. Their approach allows a user to sketch the stratigraphy on one side of the model which is then extruded to a solid.

4.1.5. Solid Representations

Here we consider solid representations differing from boundary representations in that they are not hollow, but have spatially varying properties. Takayama et al. [TSNI10] present diffusion surfaces as an extension of diffusion curves [OBW*08] to 3D volumes. The representation consists of a set of coloured surfaces in 3D, describing the model's volumetric colour distribution. A smooth volumetric colour distribution that fills the model is obtained by diffusing colours from these surfaces. Colours are interpolated only locally at the user-defined cross-sections using a modified version of the positive mean value coordinates algorithm. A result of the work by Takayama et al. [TSNI10] is shown in Figure 9.

In the work by Wang et al. [WYZG11], objects are represented as implicit functions using signed distance functions. Composite objects are created by combining implicit functions in a tree structure. This makes it possible to produce volumes made of many smaller inner components. This multi-structure framework lets them produce models irrespective of resolution (see Figure 10 for their geological application example).



Figure 9: A volumetric representation of a geological scenario using diffusion surfaces (Takayama et al. [TSNI10]).

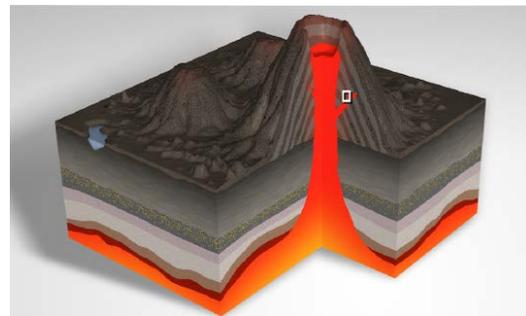


Figure 10: A volumetric representation of a geological scenario using an implicit representation (Wang et al. [WYZG11]).

4.2. Sparse and Dense Data

This section describes methods that use sparsely scattered geologically measured data or dense data, such as 3D seismic reflection volumes, for creating a subsurface model. In contrast to data-free modelling, a user's artistic expressibility is limited during model creation. The process is constrained by values in the data.

4.2.1. Measured Data

Subsurface data can be collected in several ways, at various effort and expense. Seismic 2D or 3D reflection data is collected by sending sound waves into the ground and analysing the echoes. When the sound waves enter a new material with a different impedance, a fraction of the energy is reflected. Therefore, various layer boundaries of different strength are visible in the seismic data as linear trends. Well logs are obtained by drilling into the ground and performing measurements and collecting material samples from the well. Outcrops are recorded by laser scans together with photography

(LIDAR) to create a 3D point-cloud of the side surface of geology [PHH*06]. This surface can be investigated and visible layer boundaries can be identified and outlined as curves along the surface.

4.2.2. Interpretation

Several commercial tools exist for interpreting 3D seismic data. One example is Petrel [SIS] where the user sets seed points and the system grows out a surface. The user can change the growing criteria or the seed points until a satisfactory surface is extracted. This can be time consuming. Kadlec et al. [KTD10] present a system where the user interactively steers the growing parameters to guide the segmentation instead of waiting until the growing is finished before being able to investigate it. Fast extraction of horizon surfaces is the focus of Patel et al. [PBVG10]. Their paper introduces the concept of brute-force and therefore time-consuming preprocessing for extracting possible structure candidates in 3D seismic reflection volume. After preprocessing, however, the user can quickly construct horizon surfaces by selecting appropriate candidates from the pre-processed data. Compact storage of all surface candidates is achieved by using a single volumetric distance field representation that builds on the assumption that surfaces do not intersect each other. This representation also opens up for fast intersection testing for picking horizons and for high quality visualization of the surfaces. The system allows the user to choose among precomputed candidates, but editing existing surfaces is not possible. Editing is addressed by Parks [Par09]. He presents a method that allows to quickly modify a segmented geologic horizon and to cut it for modelling faults.

Free-form modelling is achieved using boundary constraint modelling [BK04]; this is simpler and more direct than Spline modelling, which requires manipulation of many control points. Discontinuities arising from faults are created by cutting the mesh. Amorim et al. [ABPS12] allow for more advanced surface manipulation in their system. Surfaces with adaptive resolution can be altered and cut with several sketch-based metaphors. In addition, the sketching takes into account the underlying 3D seismic so that it can automatically detect strong reflection signals which may indicate horizons and automatically snap the sketched surface into position.

4.2.3. Interpolation

Key interpolating methods for surfaces in geosciences are the B-Spline method, the inverse distance method, the Kriging method, the Discrete Smooth Interpolation (DSI) method [Mal89, Mal92, Mal97] and the Natural Neighbor Interpolation method [Sib81]. Kriging is a statistical approach to interpolation that incorporates domain knowledge and is uncertainty-explicit [CD99, VCL10]. Kriging, like exemplar-based synthesis, creates a surface that has similar properties to an example dataset. Kriging uses statistical

methods. One method, based on available samples, calculates how the variation of heights between samples change as a function of the distance between the samples. In terrains, neighbouring points have more similar height values than points further away. By calculating a variogram, which has variance on the y-axis and distance on the x-axis, this variability is captured. This information is then used to interpolate values by finding height values, so that the interpolated point fits the characteristics of the variogram.

The Discrete Smooth Interpolation allows for integration of geo-physical constraints into the interpolation process. The interpolator takes as input a set of (x, y) positions, some with height values and others without which need to be calculated. After interpolation, all positions have been given height values. Discontinuities between positions can be defined so that certain points do not contribute during interpolation. Typically for a horizon surface, discontinuities would be added over fault barriers. In addition, constraints such as having points being attracted towards other points, having points being limited to movement along predefined lines or on surfaces can also be defined. These constraints are useful for interpolating geologic surface data. However the method might not be well suited for cases with very little or no observation data (as indicated by De Kemp and Sprague [DS03]), such as in the data-free scenario. Natural Neighbor Interpolation is also based on a weighted average, but only of the immediate neighbours around the position to be interpolated. A Voronoi partition is created around all known points and the weight is related to the area of these partitions around the unknown point.

An interpolation and surface representation system for geology is discussed in the work by Floater et al. [Flo98]. Scattered point measurements can come in many forms, uniformly scattered, scattered in clusters, along measurement lines or along iso-curves. Fitting a surface through the points requires interpolation. Different interpolation methods vary in quality depending on the distribution of the scatter data. Floater et al. offer interpolation in form of piecewise polynomials on triangulations, radial basis functions or least squares approximations.

Although more of a connectivity algorithm than an interpolation algorithm, Ming and Pan [MP09] present a method for constructing horizons from borehole data. Each borehole dataset consists of a sequence of regions. Each region has its start and end depth specified as well as its rock type. Figure 14a exemplifies this. One rock type might appear in several layers and also the rock type sequence might vary between boreholes. This results in several possible connectivity solutions. The challenge is to make a suitable matching of layers to create a solid layer for each rock type.

Faults define the discontinuity of horizons, however, when interpreting seismic data, fault and horizon surfaces will not be perfectly aligned and will either have gaps or overlaps between each other. Closing gaps by extending hori-

zon surfaces slightly and then cutting them at fault intersections can yield topological errors. Euler et al. [ESD98] propose to use a constrained interpolator (DSI) to control which horizons will be extrapolated and onto which faults. This creates a correctly sealed model (see Figure 11). The paper uses a test dataset called Overthrust model SEG-EAEG 1994 [ABN*94] consisting of four horizons and two faults that merge into one fault. The four horizons divide a cube into five layers. Each layer is divided into three parts by the faults resulting in fifteen distinct volumetric blocks.

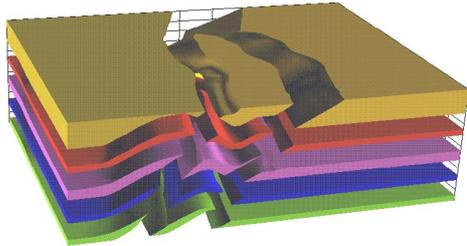


Figure 11: The work by Euler et al. [ESD98] presents a solid model made from the surfaces in the Standard overthrust SEG-EAEG 1994 model [ABN*94].

Belhadj [Bel07] models a terrain through a fractal-based algorithm. The aim of the author is to reconstruct Digital Elevation Map (DEM) models. The surface is reconstructed with constraints consisting of scattered points of elevation provided by a satellite or other sources of geological data acquisition. Furthermore, it is possible for the user to change the final shape of the terrain by intervening with sketches on the model. As the goal was to have an interactive model, the choice of the algorithm has been a fractal based approach instead of a physically based one. Specifically, part of the work is based on the so called Midpoint Displacement Inverse process (MDI), shown in Figure 12. MDI does not allow reconstruction constraints, therefore a new adapted version of this technique has been proposed, named Morphologically Constrained Midpoint Displacement (MCMD). To be able to include constraints in the computation of the interpolation, MCMD introduces changes in the order of computation of the midpoint displacement.

4.2.4. Surface Representations

In the work by Floater et al. [Flo98], surfaces can be created from scattered data and are represented either as an explicit surface ($f(x,y) = z$), parametric surface ($f(i,j) = (x,y,z)$) or a triangulation. In addition they support offset surfaces defined by one parametric surface and a function that offsets the main surface along the surface normal. Parametric surfaces can be created from triangulated surfaces.

In the works by De Kemp and Sprague [DS03, SdK05], surface modelling using traditional Bezier curves and B-splines [DB78] is discussed. Bezier curves are used as ap-

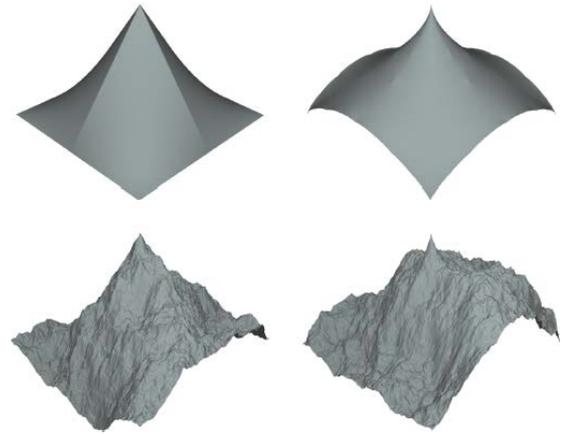


Figure 12: Example by Belhadj et al. [Bel07] of constrained midpoint displacement. Five points define the constraints for the generated surface

proximative curves, while B-splines are employed when interpolative curves are better suited. All points can be interpolated or approximated. For non-interpolated points, attraction weights can be specified. However, controlling a large number of control points individually can be tedious and lead to meaningless localized distortions. To ameliorate this issue, the authors present the technique of having hierarchical control points of decreasing resolution so that the user can move control points in the hierarchy he/she wishes to displace the surface, to avoid manipulating an excessive number of control points at the lowest resolution level. They use the concept of structural ribbons for describing a curve with normals. It can be considered as a thin strip of the surface that can be fitted on available geological information such as outcrops or map traces.

In areas of importance in an interpretation with sparse interpolated data, the paper expresses the need for expert users to be able to override and alter the coarse approximation and easily update the model when new data arrives. An expert typically attempts to get an understanding of the processes that were operative in shaping the final geometry of a given structure while at the same time respecting the local observational data.

4.2.5. Solid Assembly

Solid geometric representations of subsurface structure is important for analysis. A sealed model enables consistent inside/outside tests, providing well defined regions and good visualizations. It is also the first step for producing physical simulations of liquid or gas flow inside the model at later stages.

Baojun et al. [BBZ09] suggest a workflow for creating a 3D geological model from borehole data using commercial

tools and standards. They use ArcGIS for creating interpolated surfaces from the sparse data. They use geological relevant interpolation such as Inverse Distance Weighted, Natural Neighbor, or Kriging interpolation. This approach results in a collection of height-maps which are imported into 3D Studio Max and stacked into a layer cake model (see Figure 13). Then Constructive Solid Geometry (CSG) [PS85] operators are used to create holes (by boolean subtraction) at places where data is missing in the well logs. The model is then saved as VRML enabling widespread dissemination since it can be viewed in web browsers.

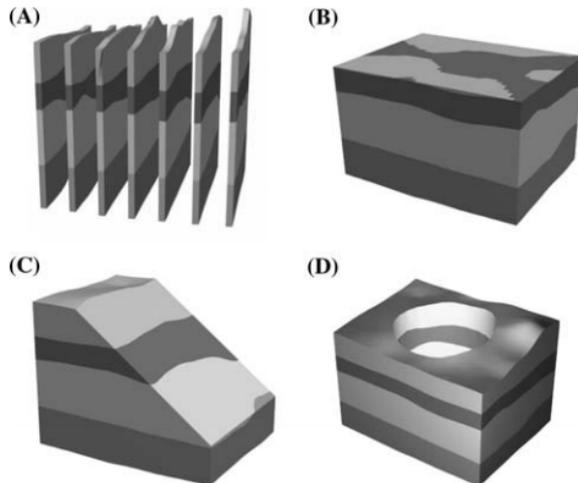


Figure 13: Geological model made with CSG operations in 3D Studio Max shown with different cut styles available in the program (Baojun et al. [BBZ09]).

When representing subsurface volumes using geometric surfaces for each stratigraphic layer, Caumon et al. [CLSM04] describe two constraints that must be followed. The constraints are that only faults can have free borders, i.e. horizon borders must terminate into other surfaces, and that horizons can not cross each other. Following the rules result in a correct and sealed model. This requires that each volume consists of triangulations with no holes and with shared vertices on seams of intersecting surfaces. Maintaining these constraints when editing horizons and faults is discussed. Mass conservation and deformation constraints during editing is also discussed. In their later work [CCDLCdV*09], additional geometric rules are introduced. The surface orientation rule states that geological surfaces are always orientable (i.e. having no twists, no Möbius ribbon topology and no self-intersections). Due to the physical process of deposition, they suggest an optional constraint requiring that horizons must be unfoldable without deformation, i.e. that they are developable surfaces with zero Gaussian curvature everywhere. They state that using implicit surfaces instead of triangulated surfaces directly enforce several

validity conditions as well as making model updates easier, however at the cost of larger memory consumption. They also discuss the importance of being aware of the varying degree of uncertainty in the different measured data modalities and, for instance, using triangulations of different coarseness according to the sparseness and uncertainty of the underlying observations. The paper presents general procedures and guidelines to effectively build a structural model made of faults and horizons from sparse data such as field observations. When creating a model, they start with fault modelling and then define the connectivity among fault surfaces. Finally horizons are introduced into the model. However, if the fault structure is very complex, they state that it is wiser to define the horizons first as if there were no faults and introduce the faults and their consequence on horizon geometry afterwards.

Lemon and Jones [LJ03] present an approach for generating solid models from borehole data (see Figure 14). The borehole data is interpolated into surfaces. For creating a closed model, they state and exemplify that CSG together with set operations can be problematic as the set operation trees grow quickly with increased model complexity. They simplify the model construction by representing horizons as triangulated surfaces while letting all horizon vertices have the same set of (x,y) positions and only varying the z positions (see Figure 15). This simplifies intersection testing between horizons and makes it trivial to pairwise close horizons by triangulating around their outer borders.

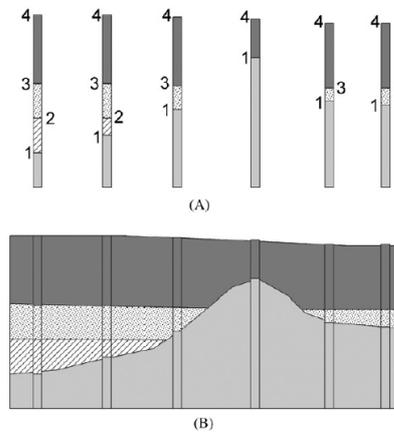


Figure 14: Borehole data in a) and resulting interpolation in b) from the method by Lemon and Jones [LJ03].

Complexity increases when models must incorporate discontinuities in the layers due to the faults. Wu and Xu [Wu03] describe the spatial interrelations between faults and horizons using a graph with horizons and faults as nodes. The graph is used to find relevant intersections and bounding surfaces which are Delaunay triangulated to form closed bodies (as shown in Figure 16). In a follow-up paper [fZH-

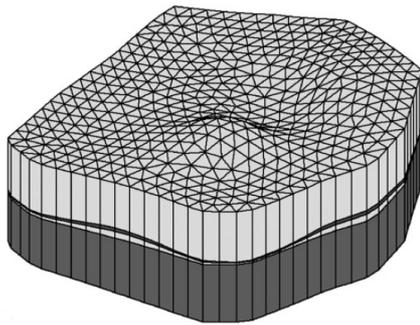


Figure 15: Example of model created with method by Lemon and Jones [LJ03]. The shared (x,y) vertex positions can be seen on the side surfaces.

PcW06], two types of fault modelling techniques are compared (based on what they call *stratum recovery* and *interpolations in subareas*) and a unified modelling technique for layers and faults is presented to solve the problems of reverse faults (i.e. convergent sedimentation blocks), syn-sedimentary faults (when slumping of sedimentary material happens before it is lithified) and faults terminated inside the model (blind faults).

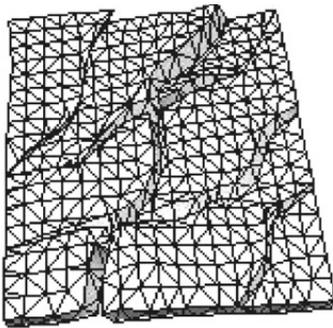


Figure 16: Mixed-mesh model employed by Wu [Wu03]. A regular mesh is used in non-boundary continuous areas, whilst an irregular triangulated mesh is adopted elsewhere.

4.2.6. Solid Representations

Solid modelling tools in CAD do not easily support subsurface features such as hanging edges and surface patches. Many papers describe data structures for representing the solid blocks that horizons and faults subdivide the subsurface into. Boundary representations are frequently used. Generalized maps, used for describing closed geological models, are introduced by Halbwachs and Hjelle [Hal02].

A 3-Generalized map (3-G-map) [Lie91] is a boundary representation appropriate for defining the topology of subsurface structures. A 3-G-map is defined as a set of darts D

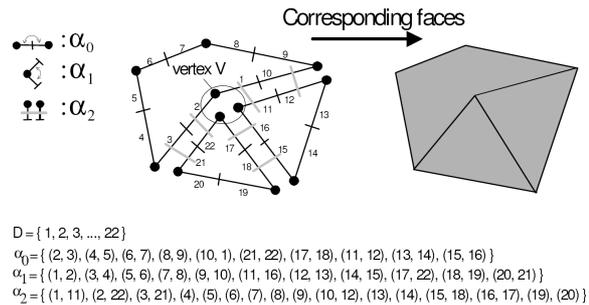


Figure 17: Example of data structure explaining the 3-G-map definition [Hal02].

and three functions on them: α_0 , α_1 , α_2 and α_3 (see Figure 17 for a 2D example). If one considers an edge as the line between two vertices, then a dart is a half-edge starting at a vertex and ending at the centre of the edge. The three functions map from darts to darts and sew half-edges into edges (by α_0), edges into polygons (by α_1), free polygons into connected polygons (by α_2), and defines neighbouring connected closed polygon volumes (by α_3). The 3-G-map is a simple yet powerful structure for defining the topology, in such a way that it is easy to traverse the space between connected or neighbouring vertices, surfaces and solids. Apel [Ape04] presents a comparison of 3-G-maps with other boundary structures.

To create a 3-G-map, relations are defined on the faults and horizons, describing how a surface is terminated onto/cut by another surface. The construction process is divided in two, first geometries are created, then they are glued together through defining topology. The 3-G-map encapsulates the topology of the final model. For 3-G-maps, topology must be described very detailed. To relieve the user from this task, several abstractions have been suggested. By letting the user instead define the relation and cuts between horizons and faults in a graph or tree datastructure, the system can then generate a detailed topology description from this. In the work by Brandel et al. [BSP*04], the user specifies a graph of chronological order for when the surfaces have been physically created. In addition a graph describing the fault network using the relation “fault A stops on fault B”, is specified. This work is extended [PZRS05] to include meta-information in the nodes of the graphs for explicitly expressing the geological knowledge attached to each of the geological surfaces. Examples of such information is that a surface is onlap, erosional, older than or stops on another surface. An example of such a graph can be seen in Figure 18.

Implicit surfaces (implicit) provide a suitable way to represent geological solids [MGC*07]. Essentially, such solids are described by implicit functions that can be expressed in different forms, e.g., distance based models, analytical

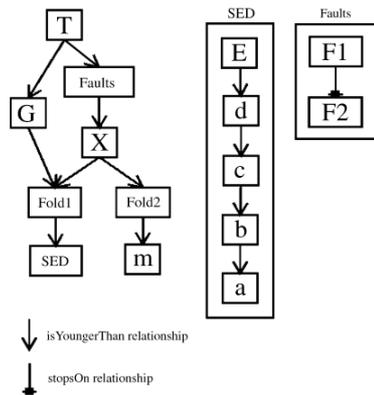


Figure 18: Example of a graph that describes the relations between geological surfaces [PZRS05].

functions, interpolation schemes like for instance RBFs, etc. Pasko et al. [PASS95] generalized the above representations, which lead to an inequality, $f \geq 0$, also called *functional representation* of solids. Kartasheva et al. [KAC*08] introduced a robust framework to model complex heterogeneous solids, which was based on functional representation. The implicit solid definition is quite broad, and for instance, the terrain modelling using a height-map can easily be represented by implicits [GM08].

5. Comparing Surface and Solid Representations for Geomodels

In this section we compare and discuss the surface representation methods in Table 1 and solid representations in Table 2, in the context of how well suited they are for modelling geologic structures. The interesting features of such representations are: how close to a natural terrain the top surface is (*Terrain realism*, i.e. the ability to portray the properties of terrain such as randomness and the occurrence of all frequencies); modelling of faults (discontinuities); interpolation of input points (*gap-filling*); support for multi-z values (*overhangs*). Ease of modelling (*control*), processing requirements, storage space requirements and the ability of simultaneous representation of high- and low-level details (*multiscale*) will also be discussed for each category. In Table 2 we compare the techniques with respect to their ability to model layers; support for tubular structures such as channels, caves or holes; ease of modelling; processing and storage requirements and multiscale support. These features are graded with plus for good support, minus for bad support or 0 if neutral.

5.1. Surfaces

This subsection discusses the surface methods described in Table 1. The methods are split in two by a gray separating

line. The three first methods produce surfaces procedurally while the four last methods (four last columns) interpolate points into surfaces. Fractal techniques create a surface from input parameters; erosional methods create a surface from an input surface and simulation parameters; while exemplar-based methods create surfaces based on a collection of surface examples. Radial-basis functions and splines are defined by control points possibly set by a user. Kriging and DSI methods are completely automatic, therefore the user control does not apply to them and they are grayed out in Table 1. For comparing the capability to model faults, although any method can support this by splitting the surface into two, we strictly evaluate the methods in their mathematical formulation, without allowing such a heuristic.

Fractal and *noise-based* methods (Section 4.1.1) are well suited for achieving a realistic appearance of the surfaces. In particular, fractals are ideal for expressing the self-similarity found in nature. In addition, noise can increase the random behaviour of real geological surfaces. Faults are difficult to represent with fractal or noise approaches, as they usually are represented by height-maps that do not allow discontinuities. For the same reason, multi-z values can usually not be expressed with these techniques. Fractal and noise-based methods do not allow intuitive or local control of the surface, but it is easy to vary few parameters to obtain a different result. There is no need to store data, fractal and noise behaviour is represented by compact analytical formulas. On the other hand, processing requirements can be high, depending on the complexity of the formula describing the surface shape. Multiscale behaviour is present in fractals by their definition.

Erosion (Section 4.1.1) is a process that affects terrain by simulating weathering. Therefore it is very well suited for modelling a natural appearance of the top layer. Erosion is modelled as a flow process and, therefore, does not handle discontinuities well. Erosion, when used with Smoothed Particle Hydrodynamics (SPH), needs to incorporate some data interpolation method to fill the gaps. Erosion processes can result into carvings, and thereby multiple z values. The erosion process is hard to control. Essentially a simulation with given parameters is initiated and the user can either accept the results or modify input parameters for a more satisfactory result. Storage requirements are low, while the resulting model can be of arbitrary size. On the other hand, erosion is a dynamic process that requires processing resources for the simulation.

Exemplar-based (Section 4.1.1) techniques can, to a certain degree, represent faults, but not real discontinuities, since the methods (mostly) uses height-maps. It can in theory synthesize terrains with abrupt changes if the exemplars contain steep cliffs. The method was not initially designed for data interpolation. However exemplar synthesis often works with having a filter expanding the border of the so-far-made-texture by filling in with parts of exemplars that

	Fractal & Noise-based	Erosional	Exemplar-based synthesis	Radial-basis function	Splines	Kriging	Discrete smooth interpolation
Terrain realism	+	+	++	-	-	+	O
Faults (discontinuities)	-	-	O	-	-	-	++
Gap-filling (interpolation)	+	-	+	++	+	++	+
Overhangs (multi-z values)	O	O	--	++	+	-	+
Control	-	-	O	+	+		
Processing requirements	-	-	-	-	+	-	--
Storage requirements	++	+	-	+	+	+	+
Multiscale	++	O	+	--	--	-	-

Table 1: This table compares the abilities of surface representation methods in terms of modelling geological features. Methods to the left of the gray separator procedurally create surfaces, while methods to the right are interpolative.

have similar neighbourhoods. Therefore interpolation can be made by starting with a texture having the interpolation values set, and letting the rest be synthesized. Multi-z values are not supported for methods using a 2D height-map. In theory, one could perform 3D texture synthesis, but this has not been explored for terrain generation. Classical exemplar-based methods offer no control at all, whereas more recent methods allow for a coarse input mesh [BSS07] and can be guided by a user-sketched feature [ZSTR07]. Brosz et al. [BSS07] show how to use a base terrain and add details by texture synthesis. Storage requirements are quite high, since many exemplars must be stored. Furthermore, creating the terrain is computationally expensive.

Radial-basis functions (RBFs) (Section 4.1.3) represent a variational interpolation technique that enables to fit/approximate an iso-surface to a given set of points and normals associated to these points. Here, the points can be given in arbitrary order unlike splines. An important feature, that might be seen as a drawback for geological models, is the C^n continuity of the resulting surface, where n is a number of points. To produce highly realistic terrains one would need to specify a substantial number of points with varying normals to interpolate. On the other hand the RBF method can easily fill the gaps in the surface model and model overhangs, which comes from the nature of the

technique [SWSJ06]. In practice, the specification of control points and normals can guide the appearance of the final surface. Moreover, modelling multiscale features is not supported due to the linear model composition. In order to visualize the final iso-surface, one needs to evaluate the function at the given point, which puts the computational burden to the surface generation step. Nevertheless, to store the implicit function, one only needs to specify the function evaluation process based on the given points and normals. Multiscale modelling is not directly supported.

Splines are defined as parametric surfaces that similarly to other interpolation techniques, produce the surface from a set of control points and the tangent or normal vectors associated with the points. Note that splines require an ordered list of points, which makes the modelling procedure somewhat tedious. Similarly to other interpolation-based surfaces, spline surfaces are continuous by their nature, which makes it hard to create discontinuous faults or realistic terrains. In comparison to RBFs, splines require a greater effort to change a surface model to fill the gaps or to produce overhangs. On the other hand, the parametric form facilitates the computation and visualization of the resulting surface. Multiscale representations are natively, similarly to RBFs, not supported by the spline model definition.

Kriging (Section 4.2.3) produces good terrain realism, because the interpolated values are correct in a statistical sense. In addition, Kriging is ideal for filling gaps in the input dataset, since the method is tailored for interpolating terrains using statistics.

Discrete Smooth Interpolation (DSI) (Section 4.2.3) belongs to the family of interpolation techniques that compute the missing information (function values) on a given graph. As such, it provides a powerful framework for modelling specific features in geology. For instance, the information about discontinuities on a set of vertices can be specified by cutting out connected nodes or by adjusting their contributing weights [Mal89]. Since the entire evaluation procedure that computes the unknown values at a graph node requires a minimization (iterative) algorithm, the processing complexity is very high compared to other interpolation techniques. However DSI is efficient in iterative modelling when one needs to adjust an existing model. Essentially, to update the node values, only few steps of the iterative minimization procedure are required. Since DSI evaluates values at nodes and not anywhere else, one stores only the graph nodes with their attributes and connectivity information. Due to this property, they do not support multiscale surface representations.

5.2. Solids

The output from the surface methods in Table 1 are input to the solid-creation methods in Table 2. Solids can then be faulted or carved after creation, if the method supports this.

Implicit solids (Section 4.2.6) do not offer any special classes of implicits aimed at geological models. Nevertheless, they offer a variety of techniques to represent such models. For instance, layers can be represented by combination of implicit primitives or by utilization of RBFs. Additionally, cavities can be realized by a subtraction operator applied to two, or multiple, compound objects [KAC*08]. The representation of implicits in multiscale models has also been successfully introduced [SWSJ06]. Moreover, the interactive modelling capabilities become more and more prominent with the introduction of sketch-based interfaces [VMc*10, KHR02]. One of the major advantages of implicits, when representing even very complex objects, is their storage requirements, which is simply represented by the function evaluation process. On the other hand, to visualize the final solids, one needs to convert the implicit models into a set of triangles or adopt a direct ray-casting method.

Constructive Solid Geometry (CSG) can be used to compose a layer-cake model with simple layers, in terms of shape definition. It is also adaptable to multiscale solutions and channels/cavities representation (e.g. employing logical set operator minus). CSG is defined by simple primitives and set operators, but the global shape is difficult to intuitively control when the model starts to become complex ([LJ03]). If the primitives are basic geometrical objects, CSG does not

require much memory, and their logical interactions are relatively quick.

3-G-maps are the representation of choice for several geological solid modelling approaches [Hal02,BSP*04]. This is a boundary representation where the boundaries are typically triangulations. Details at different scales are supported and depend on the detail level of the underlying geometry. During modelling, the triangulations and their topology must be synchronously updated. There are no particular challenges with respect to processing or storage of 3-G-maps.

A *voxel representation* is essentially a regular 3D discretized volume representation with given values in each sample. It can store layer information by simply tagging each voxel with a bit pattern defining a certain segmentation mask that defines the layer. Due to its expressiveness, it can also easily handle faults. In both horizons and faults, the final representation might need to be further processed in order to avoid visual artefacts arising from the space discretization. It can also support complex shapes, like cavities and channels. Voxel representations has been used for expressing channels [PGGM09a]. This datastructure does not offer a natural modelling approach that would be simple to use for a modeller. Therefore, it is often combined with other modelling representations. A voxel representation is very space demanding, but it does not require a computational stage for evaluation, as it explicitly stores values in memory.

Diffusion Surfaces (Section 4.1.5) apply to layered models, but the approximation introduced by Takayama et al. [TSNI10] in their tool restricts each layer to have a rotational symmetry. Diffusion surfaces lack ease of modelling when dealing with multiscale models, because a user has to define every boundary surface that delimits a piece of the solid model. Cavities are representable with diffusion surfaces, in particular if they have symmetries in shape. In producing the volumetric colour distributions, it is not necessary to perform precomputations, as opposed to Poisson approaches. Storage requirements are low, because colours are interpolated locally at cross-sectional locations.

Vector volumes (Section 4.1.5) is a volumetric representation of objects represented as a tree of signed distance functions (SDF trees). Thus, vector volumes combine the benefits of voxel and implicit models. Since each SDF tree contains the information about the interior and the exterior of an object in a hierarchical fashion, vector volumes provide a powerful way to represent solids at different levels of detail, although the storage requirements can become very high. Moreover, such a representation requires a tedious way to interactively update the model. Although a volumetric object markup language is described [TSNI10], we do not consider it being a straightforward solution. A volumetric object is usually achieved instead by developing and updating the model via different representations, e.g., boundary representation, and then performing the conversion into vector volumes. Due to the unique voxel and SDF identification

	Implicit solids	CSG	3-G-Maps	Voxel representation	Diffusion surfaces [TSNI10]	Vector volumes [WYZG11]
Layer support	+	+	++	+	○	+
Channels/cavities support	+	+	+	++	+	+
Ease of modelling	+	+	+	-	+	-
Processing requirements	--	○	○	+	+	○
Storage requirements	++	+	○	--	+	-
Multiscale	+	+	+	--	○	++

Table 2: This table shows and compares the abilities of solid representation methods in terms of modelling geological features.

when performing a volume ray-casting, this representation becomes efficient in direct visualization. Nevertheless, it is still required to evaluate the implicit function at each node in addition.

6. Challenges and trends in geological modelling

Geoscience technology on closed model representations and model updating has not progressed at the same speed as in computer graphics. Better knowledge transfer between these groups could be advantageous. Caumon et al. [CCDLCdV*09] state that beginners with 3D modelling too often lose their critical sense about their work, mostly due to a combined effect of well defined graphics and non-optimal human-machine communication. It is also important that a structural model can be updated when new data becomes available or perturbed, to account for structural uncertainties. In other words, with current modelling technology, uncertainty is difficult to express, and models are hard to update.

Researched literature from this domain emphasizes a strong need for modelling technology for communication and further analysis of the Earth's subsurface. While several matured methods are now in use by the domains of geology and geosciences, all tools require considerable effort to build structural model. Current tools focus on precise modelling in favour of rapid modelling. We believe that rapid modelling is the key for the ability of expertise exchange, especially in the early phases of the interpretation process.

As interesting research directions that would benefit from future attention of the graphics community, we point out two

distinct ones: one research direction can be procedural geological modelling that takes advantage from sparsely defined acquired information about the subsurface. Ideally an automated procedural method could be refined by the user through a series of sketches. Another research direction can be the consideration of temporal aspects in geology. Erosion has been investigated in this context, but geological processes are driven by many more phenomena than only surface erosion. Here, also the intended process can benefit from user input in the form of sketched information.

Acknowledgment

This work is partly funded by the Petromaks program of the Norwegian Research Council through the Geoillustrator project (#200512) and by the Statoil Academia Agreement. A minor part of the work has been funded by the Vienna Science and Technology Fund (WWTF) through project VRG11-010.

Biographies of the Authors

Mattia Natali is a Ph.D. student at the University of Bergen (UiB), Norway. He received his M.Sc. degree in applied mathematics in 2009 from the University of Genova, Italy. His research is focused on illustrative visualization of geological models, sketch-based modelling and computational geometry.

Endre M. Lidal is a Ph.D. student at University of Bergen. He received M.Sc. in computer science in 1997 at the Norwegian University of Science and Technology,

Trondheim, Norway. His PhD studies are within the interdisciplinary Geoilustrator research project, where his focus is on illustrative visualization of geological models, rapid geology modeling, and visual alternatives exploration. Before starting the PhD studies he has been working for more than a decade with research and development within scientific visualization. He has among other things worked with virtual reality system, volume rendering, and large visualization system development. The main application domain has been the geosciences, and the oil and gas industry. He is a member of ACM SIGGRAPH and IEEE Computer Society.

Julius Parulek is a Post-Doc fellow in the Visualization Group at Department of Informatics, University of Bergen, Norway. He received his MSc. in 2003 and his PhD study in 2008, in the area of computer science at the faculty of Mathematics, Physics and Informatics, Department of Applied Informatics, Comenius University, Bratislava. His research mainly focuses on visualization in biology (interactive 3D visualization integrated with visual analytics). Additionally, he has special interests in 3D geometrical modeling and visualization of Implicit Surfaces. Other topics cover image processing and computer vision.

Ivan Viola is assistant professor at the Vienna University of Technology, Austria, and adjunct professor at the University of Bergen, Norway. He received M.Sc. in 2002 and Ph.D. in 2005 from Vienna University of Technology, Austria. His research is focused on perception effective visualization and illustrative visualization for communication of complex scientific data. Viola co-authored several scientific works published in international journals and conferences such as IEEE TVCG, IEEE Vis, and EuroVis and acted as a reviewer and IPC member for conferences in the field of computer graphics and visualization. He is member of IEEE Computer Society, VGTC, ACM SIGGRAPH

Daniel Patel received his PhD in visualization and analysis of seismic volumetric data in November 2009. The PhD was taken jointly at Vienna University of Technology and at the University of Bergen (UIB). He has published several works dealing with fast interpretation and modelling of seismic structures. He is working as an Adjunct Assistant Professor at the visualization group in UIB, as well as a technical leader in the research company Christian Michelsen Research, Bergen, Norway. He is involved in the Geoilustrator research project which focus on geological modelling in collaboration with Statoil, the visualization department at UIB and the geology department at UIB.

References

- [ABN*94] AMINZADEH F., BURKHARD N., NICOLETIS N., ROCCA F., K W.: SEG-EAEG 3D modeling project. *The Leading Edge* (September 1994). 10
- [ABPS12] AMORIM R., BRAZIL E. V., PATEL D., SOUSA M. C.: Sketch modeling of seismic horizons from uncertainty. In *Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling* (2012), SBIM '12, pp. 1–10. 9
- [Ape04] APEL M.: *A 3d geoscience information system framework*. PhD thesis, TU Freiberg, 2004. 12
- [Av07] ANDERSEN C., VAN WIJNGAARDEN A.-J.: Interpretation of 4d avo inversion results using rock physics templates and virtual reality visualization, north sea examples. *SEG Annual Meeting* (September 2007). 4
- [BBZ09] BAOJUN W., BIN S., ZHEN S.: A simple approach to 3D geological modelling and visualization. *Bulletin of Engineering Geology and the Environment* 68, 4 (2009), 559–565. 4, 10, 11
- [Bel07] BELHADJ F.: Terrain modeling: a constrained fractal model. In *Proceedings of the 5th international conference on Computer graphics, virtual reality, visualisation and interaction in Africa* (2007), vol. 1, pp. 197–204. 3, 10
- [BF01] BENES B., FORSBACH R.: Layered data representation for visual simulation of terrain erosion. *Spring conference on Computer Graphics (SCCG)* (2001). 6
- [BF02] BENES B., FORSBACH R.: Visual simulation of hydraulic erosion. *Journal of WSCG* (2002). 6
- [BK04] BOTSCH M., KOBELT L.: An intuitive framework for real-time freeform modeling. In *ACM SIGGRAPH 2004 Papers* (2004), SIGGRAPH '04, pp. 630–634. 9
- [BMV*11] BERNHARDT A., MAXIMO A., VELHO L., HNAIDI H., CANI M.-P.: Real-time Terrain Modeling using CPU-GPU Coupled Computation. In *XXIV SIBGRAPI* (Maceio, Brazil, Aug. 2011). 8
- [BSP*04] BRANDEL S., SCHNEIDER S., PERRIN M., GUIARD N., RAINAUD J.-F., LIENHARDT P., BERTRAND Y.: Automatic building of structured geological models. In *Proceedings of the ninth ACM symposium on Solid modeling and applications* (2004), SM '04, pp. 59–69. 12, 15
- [BSS07] BROSZ J., SAMAVATI F., SOUSA M.: Terrain synthesis by-example. *Advances in Computer Graphics and Computer Vision* (2007), 58–77. 3, 7, 14
- [BTHB06] BENES B., TESINSKY V., HORNYS J., BHATIA S. K.: Hydraulic erosion. *Computer Animation and Virtual Worlds* 17, 2 (2006), 99–108. 6
- [Cau10] CAUMON G.: Towards stochastic time-varying geological modeling. *Mathematical Geosciences* 42, 5 (2010), 1–25. 4
- [CCDLCdV*09] CAUMON G., COLLON-DROUILLET P., LE CARLIER DE VESLUD C., VISEUR S., SAUSSE J.: Surface-Based 3D Modeling of Geological Structures. *Mathematical Geosciences* 41, 8 (2009), 927–945. 3, 11, 16
- [CD99] CHILÈS J.-P., DELFINER P.: *Geostatistics : modeling spatial uncertainty*. John Wiley and Sons, Inc., 1999. 9
- [CHZ00] COHEN J. M., HUGHES J. F., ZELEZNIK R. C.: Harold: a world made of drawings. In *Proceedings of NPAR '00* (2000), pp. 83–90. 7
- [CLSM04] CAUMON G., LEPAGE F., SWORD C. H., MALLET J.-L.: Building and Editing a Sealed Geological Model. *Mathematical Geology* 36, 4 (2004), 405–424. 4, 11
- [CM98] CHIBA N., MURAOKA K.: An erosion model based on velocity fields for the visual simulation of mountain scenery. *The Journal of Visualization* 194 (1998). 6
- [DB78] DE BOOR C.: *A Practical Guide to Splines*. No. v. 27 in Applied Mathematical Sciences. Springer-Verlag, 1978. 10
- [dCB09] DE CARPENTIER G. J. P., BIDARRA R.: Interactive gpu-based procedural heightfield brushes. In *Proceedings of the 4th International Conference on Foundations of Digital Games* (New York, NY, USA, 2009), FDG '09, ACM, pp. 55–62. 7

- [DEJ*99] DORSEY J., EDELMAN A., JENSEN H., LEGAKIS J., PEDERSEN H.: Modeling and rendering of weathered stone. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques* (1999), pp. 225–234. 6
- [DLC05] DHONT D., LUXEY P., CHOROWICZ J.: 3-D modeling of geologic maps from surface data. *AAPG Bulletin* 89, 11 (2005), 1465–1474. 4
- [DP10] DORAN J., PARBERRY I.: Controlled procedural terrain generation using software agents. *IEEE Transactions on Computational Intelligence and AI in Games* 2, 2 (2010), 111–119. 5
- [DS03] DE KEMP E., SPRAGUE K.: Interpretive tools for 3-D structural geological modeling part I: Bezier-based curves, ribbons and grip frames. *Geoinformatica* 7, 1 (2003), 55–71. 4, 9, 10
- [ESD98] EULER N., SWORD C. H., DULAC J.-C.: A new tool to seal a 3d earth model: a cut with constraints. In *Proc. 68th Annual Meeting* (1998), Society of Exploration Geophysicists, pp. 710–713. 10
- [Flo98] FLOATER, MICHAEL AND HALBWACHS, YVON AND HJELLE, ØYVIND AND REIMERS, MARTIN: Omega: A cad-based approach to geological modelling. *GOCAD ENSG, Conference Proceedings*. 9, 10
- [Fra06] FRANK T.: Geological information retrieval using tetrahedral meshes. *gocad.org* (2006), 12–15. 4
- [fZHPcW06] FENG ZHU L., HE Z., PAN X., CAI WU X.: An approach to computer modeling of geological faults in 3d and an application. *Journal of China University of Mining and Technology* 16, 4 (2006), 461 – 465. 11
- [GCC*08] GUILLEN A., CALCAGNO P., COURRIOUX G., JOLY A., LEDRU P.: Geological modelling from field data and geological knowledge Part II. Modelling validation using gravity and magnetic data inversion. *Physics of the Earth and Planetary Interiors* 171, 1-4 (2008), 158–169. 4
- [GM77] GINGOLD R. A., MONAGHAN J. J.: Smoothed particle hydrodynamics: Theory and application to non-spherical stars. *Monthly Notices of the Royal Astronomical Society* 181 (1977), 375. 6
- [GM08] GAMITO M. N., MADDOCK S. C.: Topological correction of hypertextured implicit surfaces for ray casting. *The Visual Computer* 24 (2008), 397–409. 13
- [GMSe09] GAIN J., MARAIS P., STRASSER W.: Terrain sketching. In *Proceedings of the 2009 symposium on Interactive 3D graphics and games* (2009), pp. 31–38. 3, 7
- [Gro99] GROSHONG R.: *3-D Structural Geology: A Practical Guide to Surface and Subsurface Map Interpretation*. Springer, 1999. 4
- [Hal02] HALBWACHS, YVON AND HJELLE, ØYVIND: *Generalized maps in geological modeling: Object-oriented design of topological kernels*. Advances in Software Tools for Scientific Computing. Springer Verlag, 2002. 12, 15
- [HD11] HUDAK M., DURIKOVIC R.: Terrain Models for Mass Movement Erosion. In *Proceedings of EG-UK TPCG Conference* (2011). 3, 7
- [HGA*10] HNAIDI H., GUÉRIN E., AKKOUICHE S., PEYTAVIE A., GALIN E.: Feature based terrain generation using diffusion equation. *Computer Graphics Forum* 29, 7 (2010), 2179–2186. 3, 6
- [Hou94] HOULDING S.: *3D Geoscience Modeling: Computer Techniques for Geological Characterization*. Springer, 1994. 2
- [KAC*08] KARTASHEVA E., ADZHIEV V., COMNINOS P., FRYAZINOV O., PASKO A.: Heterogeneous objects modelling and applications. Springer-Verlag, Berlin, Heidelberg, 2008, ch. An implicit complexes framework for heterogeneous objects modelling, pp. 1–41. 13, 15
- [KBKv09] KRIŠTOF P., BENEŠ B., KŘIVÁNEK J., ŠTĚVA O.: Hydraulic Erosion Using Smoothed Particle Hydrodynamics. *Computer Graphics Forum* 28, 2 (2009), 219–228. 3, 6
- [KHR02] KARPENKO O., HUGHES J. F., RASKAR R.: Free-form sketching with variational implicit surfaces. *Computer Graphics Forum* 21, 3 (2002), 585–594. 15
- [KM08] KAUFMANN O., MARTIN T.: 3D geological modelling from boreholes, cross-sections and geological maps, application over former natural gas storages in coal mines. *Computers & Geosciences* 34, 3 (2008), 278–290. 4
- [KTD10] KADLEC B. J., TUFO H. M., DORN G. A.: Knowledge-assisted visualization and segmentation of geologic features. *IEEE Computer Graphics and Applications* 30, 1 (Jan. 2010), 30–39. 9
- [Lie91] LIENHARDT P.: Topological models for boundary representation: a comparison with n-dimensional generalized maps. *Computer-Aided Design* 23, 1 (Feb. 1991), 59–82. 12
- [LJ03] LEMON A., JONES N.: Building solid models from boreholes and user-defined cross-sections. *Computers & Geosciences* 29, 5 (2003), 547–555. 4, 11, 12, 15
- [LLG*07] LIDAL E. M., LANGELAND T., GIERTSEN C., GRIMSGAARD J., HELLAND R.: A decade of increased oil recovery in virtual reality. *IEEE Computer Graphics and Applications* 27, 6 (2007), 94–97. 4
- [LNP*13] LIDAL E. M., NATALI M., PATEL D., HAUSER H., VIOLA I.: Geological storytelling. *Computers & Graphics* (2013). 4
- [Luc77] LUCY L.: A numerical approach to the testing of the fission hypothesis. *Astronomical Journal* 82 (1977), 1013. 6
- [Mal89] MALLET J.-L.: Discrete smooth interpolation. *ACM Transactions on Graphics* 8, 2 (Apr. 1989), 121–144. 9, 15
- [Mal92] MALLET J.-L.: Discrete smooth interpolation in geometric modelling. *Computer-aided design* 24, 4 (1992), 178–191. 4, 9
- [Mal97] MALLET J.: Discrete modeling for natural objects. *Mathematical geology* 29, 2 (1997), 199–219. 4, 9
- [Man82] MANDELBROT B. B.: *The Fractal Geometry of Nature*. W. H. Freedman and Co., New York, 1982. 5
- [MGC*07] MCINERNEY P., GOLDBERG A., CALCAGNO P., COURRIOUX G., GUILLEN A., SEIKEL R.: Improved 3D geology modelling using an implicit function interpolator and forward modelling of potential field data. In *Proceedings of Exploration* (2007), vol. 7, pp. 919–922. 12
- [MKM89] MUSGRAVE F., KOLB C., MACE R.: The synthesis and rendering of eroded fractal terrains. *ACM SIGGRAPH Computer Graphics* 23, 3 (1989), 41–50. 5
- [MP09] MING J., PAN M.: An Improved Horizons Method for 3D Geological Modeling from Boreholes. *2009 International Conference on Environmental Science and Information Application Technology* (2009), 369–374. 4, 9
- [NVP12] NATALI M., VIOLA I., PATEL D.: Rapid visualization of geological concepts. In *SIBGRAPI 2012 (XXV Conference on Graphics, Patterns and Images)* (august 2012). 4, 8
- [OBW*08] ORZAN A., BOUSSEAU A., WINNEMÖLLER H.,

- BARLA P., THOLLOT J., SALESIN D.: Diffusion curves: a vector representation for smooth-shaded images. *ACM Transactions on Graphics* 27, 3 (Aug. 2008). 8
- [Ols04] OLSEN J.: *Realtime procedural terrain generation*. Tech. rep., Department of Mathematics And Computer Science (IMADA) University of Southern Denmark, 2004. 3, 5
- [Par09] PARKS D.: Freeform modeling of faulted surfaces in seismic images. *SEG Technical Program Expanded Abstracts* 28, 1 (2009), 2702–2706. 9
- [PASS95] PASKO A. A., ADZHIEV V., SOURIN A., SAVCHENKO V. V.: Function representation in geometric modeling: concepts, implementation and applications. *The Visual Computer* 11, 8 (1995), 429–446. 13
- [PBVG10] PATEL D., BRUCKNER S., VIOLA I., GRÖLLER M. E.: Seismic Volume Visualization for Horizon Extraction. In *IEEE Pacific Visualization 2010* (2010), pp. 73–80. 4, 9
- [PGGM09a] PEYTAVIE A., GALIN E., GROSJEAN J., MERILLOU S.: Arches: a Framework for Modeling Complex Terrains. *Computer Graphics Forum* 28, 2 (2009), 457–467. 3, 8, 15
- [PGGM09b] PEYTAVIE A., GALIN E., GROSJEAN J., MERILLOU S.: Procedural Generation of Rock Piles using Aperiodic Tiling. *Computer Graphics Forum* 28, 7 (2009), 1801–1809. 3, 5
- [PGT*08] PATEL D., GIERTSEN C., THURMOND J., GJELBERG J., GRÖLLER M. E.: The seismic analyzer: interpreting and illustrating 2D seismic data. *IEEE transactions on visualization and computer graphics* 14, 6 (2008), 1571–8. 4
- [PGTG07] PATEL D., GIERTSEN C., THURMOND J., GRÖLLER M.: Illustrative rendering of seismic data. In *Proceedings of vision modeling and visualization* (2007), pp. 13–22. 4
- [PHH*06] PRINGLE J., HOWELL J., HODGETTS D., WESTERMAN A., HODGSON D.: Virtual outcrop models of petroleum reservoir analogues: a review of the current state-of-the-art. *first break* 24 (2006), 33. 2, 9
- [PHT93] PRUSINKIEWICZ P., HAMMEL M., TN C.: A fractal model of mountains with rivers. *Panorama* (1993), 174–180. 3, 6
- [PS85] PREPARATA F. P., SHAMOS M. I.: *Computational geometry: an introduction*. Springer-Verlag New York, Inc., New York, NY, USA, 1985. 11
- [PTCF02] PLATE J., TIRTASANA M., CARMONA R., FRÖHLICH B.: Octreemizer: a hierarchical approach for interactive roaming through very large volumes. In *Proceedings of the Symposium on Data Visualisation* (2002), VISSYM '02, Eurographics Association, pp. 53–ff. 4
- [PZRS05] PERRIN M., ZHU B., RAINAUD J.-F., SCHNEIDER S.: Knowledge-driven applications for geological modeling. *Journal of Petroleum Science and Engineering* 47 (2005), 89 – 104. 12, 13
- [RPP93] ROUDIER P., PEROCHE B., PERRIN M.: Landscapes synthesis achieved through erosion and deposition process simulation. *Computer Graphics Forum* 12, 3 (1993), 375–383. 5
- [SBBK08] STAVA O., BENES B., BRISBIN M., KRIVANEK J.: Interactive terrain modeling using hydraulic erosion. In *Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2008), pp. 201–210. 6
- [SBW06] SCHNEIDER J., BOLDTE T., WESTERMANN R.: Real-time editing, synthesis, and rendering of infinite landscapes on GPUs. In *Vision, modeling, and visualization 2006: proceedings, November 22-24, 2006, Aachen, Germany* (2006), p. 145. 3, 5
- [SdK05] SPRAGUE K. B., DE KEMP E. A.: Interpretive Tools for 3-D Structural Geological Modelling Part II: Surface Design from Sparse Spatial Data. *GeoInformatica* 9, 1 (2005), 5–32. 10
- [Sib81] SIBSON R.: A brief description of natural neighbor interpolation. *Interpreting multivariate data* (1981), 21–26. 9
- [SIS] SIS: Petrel seismic interpretation software. *Schlumberger Information Solutions*. http://www.slb.com/services/software/geo/petrel/seismic/seismic_interpretation.aspx. 9
- [SN93] SAPOZHNIKOV V., NIKORA V.: Simple computer model of a fractal river network with fractal individual watercourses. *Journal of Physics A: Mathematical and General* 26 (1993), L623. 5
- [SS05] STACHNIAK S., STUERZLINGER W.: An algorithm for automated fractal terrain deformation. *Computer Graphics and Artificial Intelligence* (2005). 5
- [SWSJ06] SCHMIDT R., WYVILL B., SOUSA M. C., JORGE J. A.: Shapeshop: sketch-based solid modeling with blobtrees. In *ACM SIGGRAPH 2006 Courses* (New York, NY, USA, 2006), SIGGRAPH '06, ACM. 14, 15
- [TG07] TURNER A., GABLE C.: A review of geological modeling. *Three-dimensional geologic mapping for groundwater applications. Minnesota Geological Survey Open-file Report* (2007), 07–4. 2
- [TSNI10] TAKAYAMA K., SORKINE O., NEALEN A., IGARASHI T.: Volumetric modeling with diffusion surfaces. *ACM Transactions on Graphics (proceedings of ACM SIGGRAPH)* 29, 6 (2010). 8, 15
- [Tur06] TURNER A. K.: Challenges and trends for geological modelling and visualisation. *Bulletin of Engineering Geology and the Environment* 65, 2 (2006), 109–127. 2, 4
- [VCL10] VIARD T., CAUMON G., LÉVY B.: Adjacent versus coincident representations of geospatial uncertainty: Which promote better decisions? *Computers & Geosciences* (2010). 9
- [VMc*10] VITAL BRAZIL E., MACÊDO I., COSTA SOUSA M., DE FIGUEIREDO L. H., VELHO L.: Sketching Variational Hermite-RBF Implicits. In *Proc. Sketch Based Interfaces and Modeling* (2010), pp. 1–8. 3, 7, 15
- [WI04] WATANABE N., IGARASHI T.: A sketching interface for terrain modeling. In *ACM SIGGRAPH 2004 Posters* (2004), p. 73. 7
- [Wij03] WIJNS C.: Inverse modelling in geology by interactive evolutionary computation. *Journal of Structural Geology* 25, 10 (2003), 1615–1621. 4
- [Wu03] WU Q. X. H.: An approach to computer modeling and visualization of geological faults in 3D. *Computers & Geosciences* 29, 4 (2003), 503–509. 11, 12
- [WYZG11] WANG L., YU Y., ZHOU K., GUO B.: Multiscale vector volumes. *ACM Transactions on Graphics* 30, 6 (Dec. 2011), 167:1–167:8. 8
- [Zha05] ZHANG Z.: 3D Terrain Reconstruction Based on Contours. *Computer* (2005), 3–8. 4
- [ZSTR07] ZHOU H., SUN J., TURK G., REHG J. M.: Terrain synthesis from digital elevation models. *IEEE transactions on visualization and computer graphics* 13, 4 (2007), 834–48. 7, 14