

THESIS

3D Intersection operations for voxel data represented as surfaces in GIS



TU Delft – Geomatics – 3D geoinformation group

Supervisors: Ass. Prof. dr. S. (Sisi) Zlatanova
drs. M.E. (Marian) de Vries

Co-reader: dr. F.L.(Fransje) Hooimeijer

Delegate of the Board of Examiners:

Ir. E.J. (Engbert) van der Zaag

Royal HaskoningDHV

Mentor: dr. M.J.M. (Maartje) van Meeteren,
Advisor Subsurface and Environment

ABSTRACT

This thesis research focuses on semantic and attribute preserving 3D intersection operations in GIS as 3D Boolean set operations, performed on vectorised voxel data, together with 2D surface data and 2D line and point data. In GIS, semantics are the titled layer or dataset and attributes are information accompanying the geometry. In order to make the three dimensional intersection (which contains overlapping 3D geometry), all three input datasets are converted into the same data representation: 3D vector data, surface based.

Regarding the input data: GeoTOP is the voxel dataset that models the geological features of the upper 50m part of the subsurface, TOP10NL contains the 2D surfaces and is the most detailed digital topographic map, and KLIC data which consist of cables, pipes and sewer system data in the subsoil of The Netherlands.

Semantics and attribute information of the input datasets needs to be maintained in the 3D intersection output, because based on the properties of the GeoTOP dataset – which are stored in the attributes – basic geohydrologic principles can be determined of the investigated, open soil locations. Via geohydrology, insights in the water infiltration and storage capacity of these 3D intersected open soil locations are provided, which is relevant regarding the increase in amount of precipitation urban areas have to deal with. The linking of the 3D intersection and geohydrology is done in the Application chapter of this thesis.

The main test environment in this research is ArcGIS, which has Multipatches as their main 3D geometry. ArcGIS incorporates 3D Boolean set operations in their 3D Analyst extension, named as the *Difference 3D*, *Union 3D* and *Intersect 3D* set operations. The Nef_Polyhedra_3 class implementation of CGAL is used by ArcGIS to (i) convert their Multipatch geometry to CGAL data structures, and (ii) to perform the 3D Boolean set operations on the converted data.

Three conceptual workflows are proposed – one for each input data set – to convert the input data into 3D surface based vector data. Also, a conceptual workflow to perform 3D intersection operations that maintains semantics and attributes, is drafted. All workflows are implemented and the 3D intersection workflow is tested and evaluated on various 'Test Cases'. The criteria on which the evaluation is made are (i) geometric validity, (ii) correct assignments of attributes, and (iii) the possibility to perform volume and surface area calculations. These criteria test the design of the proposed conceptual 3D intersection workflow, and at the same time evaluate the ArcGIS implementation of CGAL's Nef_Polyhedra_3 class.

Converting the voxel dataset to 3D surface based vector data, requires a sound elaboration, because the voxels need to be represented as points, which can be performed via a script. For the other data representations, other conversion steps are necessary: 2D surface data mainly needs to be extruded, and 2D line a point data requires buffering and additionally 3D modelling via COLLADA.

The cases drafted to test the implementation of the 3D intersection workflow, pointed out that in general the implementation can be achieved in ArcGIS: semantics and attributes are maintained on valid geometry output, which is sufficient result for the purposes of Application. The Nef Polyhedra are a suited data structure to perform 3D Boolean set operations on, and to attach the attributes belonging to the original input geometry, to the geometry output result of a set operation.

Of certain tests that adopted more complex input, geometry was invalid (non-closed) and volume calculations could not be performed. The relationship between valid, closed geometry and the possibility to perform volume calculations, is not a consequent relation, for instance it occurred that closed geometry delivered a volume of zero. This could be the case cause of the complexity of the input geometry, e.g. geometry that consists out of a large amount of surfaces. Also, the non-manifold cases pointed out that a sound theoretical Nef Polyhedron cannot be fully modelled in the chosen test environment of ArcGIS.

The Application showed the value of linking geohydrology to 3D intersected GeoTOP data. Spatial analysis via 3D intersection operations that maintained semantics and attributes on the output, resulted in a reasoned explanation on the geohydrologic potential of the subsurface.

ACKNOWLEDGEMENTS

I would like to thank all my mentors; Sisi Zlatanova, Marian de Vries of TU Delft and Maartje van Meeteren of Royal HaskoningDHV, for sharing their expert knowledge, and for their guidance.

Furthermore, I would like to thank Martijn Meijers for his guidance with Python programming, Fransje Hooimeijer for her urban water management and terminology expertise, Niels van der Vaart of ESRI for his accurate and very informative responses and data sharing. My gratitude goes out to Tom van Steijn for assisting me with geo-hydrologic matters, and to Ingeborg van Oorschot, Arno Peekel and Henk Koster of the RHDHV Advice Group Subsurface for their support in this research.

CONTENT

Abstract	3
Acknowledgements	4
List of Figures	7
List of Tables	8
Acronyms	9
1. Introduction	
1.1 Research Scope	
1.1.1 Motivation and Relevance	10
1.1.2 Research Requirements	11
1.1.3 Problem Statement	12
1.2 Research Approach	
1.2.1 Research Questions	12
1.2.2 Application	13
1.2.3 Method Description	13
1.2.4 Expected Results	14
1.3 Related Work	14
1.4 Reading Guide	15
2. Background Information	
2.1 CAD/GIS	17
2.2 Data: Vector and Raster	
2.2.1 Vector and raster representation principles	17
2.2.2 Three-dimensional Vector data	18
2.2.3 Validity: 2D Polygon and 3D Polyhedra	19
2.2.4 Voxel data	19
2.2.5 Multipatch in ArcGIS	20
2.2.6 Multipatch validity	22
2.3 3D Intersection operations	
2.3.1 Clipping	23
2.3.2 3D Boolean set operations	25
2.3.3 Nef Polyhedra main theories	26
2.3.4 Nef Polyhedra in CGAL	28
2.3.5 3D Analyst in ArcGIS	29
2.3.6 ArcGIS and CGAL	33
2.4 Resume Background Information	33
3. Used data	
3.1 Voxel GeoTOP	
3.1.1 Subsurface Models of TNO	35
3.1.2 GeoTOP Introduction	35
3.1.3 Modeling GeoTOP	36
3.1.4 GeoTOP voxels	36
3.1.5 GeoTOP litho classes as attribute	38
3.2 2D surface: TOP10NL	38
3.3 2D line and points: KLIC	40
4. Conceptual Workflows	
4.1 Conversion Workflows	
4.1.1 A workflow for voxel data to 3D vector data, surface based	42
4.1.2 A workflow for 2D surface data to 3D vector data, surface based	42

4.1.3 A workflow for 2D line data to 3D vector data, surface based	43
4.2 3D Intersection workflow	
4.2.1 A workflow for 3D intersection operations on surfaces	44
5. Testing and Evaluation	
5.1 Implementing Conversion Workflows	
5.1.1 Voxel conversion	47
5.1.2 2D surface conversion	48
5.1.3 2D line and point conversion	49
5.2 Implementing 3D Intersection Workflow	50
5.3 Testing and Evaluating the 3D Intersection Operations	51
1. Simple objects	54
2. Complex object	56
3. Overlapping surface/edge/point	60
4. Complex object – Difference	64
5. Construct objects – non-manifolds	64
Overview table	67
6. Application	
6.1 Geohydrology	69
6.1.1 Infiltration capacity	70
6.1.2 Storage capacity	71
6.2 Working Demo	72
6.2.1 Groundwater levels Dordrecht	73
6.2.2 Building the working demo	73
6.2.3 Working demo results	75
6.2.4 Web publishing	78
6.3 Application Findings	79
7. Conclusions	
7.1 Conclusions and Answered Research Questions	
7.1.1 Main research question	80
7.1.2 The sub questions	81
7.1.3 Findings regarding the Test Cases	82
7.2 Discussion	83
7.3 Recommendations for Future Work	83
7.4 Reflection	84
References	85
Appendices	88
A: Python Conversion Script	88
B: Groundwater surface points	89
C: Union 3D	90
D: Data, GeoTOP voxel dataset	91
E: Other investigated areas/ interpolations	92
F: Photos, illustration of precipitation puddles accompanied with 3D intersections	93

LIST OF FIGURES

Figure 1 Urban water system	10
Figure 2 Part of the 2D Soil Map and part of the 3D GeoTOP	11
Figure 3 Raster and vector data	17
Figure 4 Raster data	18
Figure 5 Volume and Surfaces	18
Figure 6 Valid polyhedra and invalid polyhedra	19
Figure 7 Multipatches	22
Figure 8 Is Closed 3D operator	23
Figure 9 Clipping and splitting plane	24
Figure 10 Clipping on screen	24
Figure 11 Point distance	24
Figure 12 Inside outside, 4 steps	24
Figure 13 Sutherland-Hodgman-Algorithm	25
Figure 14 3D Boolean set operations	26
Figure 15 Non-manifold vertex, edge, face	27
Figure 16 A Nef Polyhedra	27
Figure 17 Local pyramids	27
Figure 18 Sphere map principle	28
Figure 19 Half-edge structure	29
Figure 20 Intersect 3D Line With Multipatch	30
Figure 21 Buffer 3D	30
Figure 22 Difference, intersect and union in 3D	31
Figure 23 Difference 3D	31
Figure 24 Intersect 3D	32
Figure 25 Union 3D (ESRI ArcGIS, 2015)	32
Figure 26 overview of NL with the division of the GeoTOP	35
Figure 27 GeoTOP section	38
Figure 28 GML TOP10NL prototype example	39
Figure 29 Top10NL terrain	39
Figure 30 KLIC data	40
Figure 31 Sewer system in KLIC	41
Figure 32 COLLADA workflow	44
Figure 33 3D vectorized GeoTOP	48
Figure 34 3D vector of TOP10NL	49
Figure 35 3D vector of KLIC	50
Figure 36 Test Cases in diagram	52
Figure 37 Main 3D GeoTOP intersection	53
Figure 38 Vectorized GeoTOP and TOP10NL	54
Figure 39 Valid, closed Multipatch output	54
Figure 40 Correct attribute assignment	54
Figure 41 Correct volume calculations	55
Figure 42 Possible summarize on TYPE attribute for volumes	55
Figure 43 Oblique input	55
Figure 44 Correct attribute assignment and volume determination	56
Figure 45 Complex geometry as input	56
Figure 46 Valid complex geometry	57
Figure 47 Volume determined as zero	57
Figure 48 Long cylinder geometry	57
Figure 49 Input datasets	58
Figure 50 KLIC data, and as 3D line	58
Figure 51 Buffers	58
Figure 52 Buffers 2	59
Figure 53 Volume zero for non- closed features	59
Figure 54 Cylinder with many, and cylinder with few surfaces	59
Figure 55 Different volume output	60
Figure 56 Input datasets	60
Figure 57 Editing via vertex	61
Figure 58 Overlapping surfaces and equal color pattern surface	61

<i>Figure 59 Output error message</i>	61
<i>Figure 60 Geometry output</i>	62
<i>Figure 61 Geometry set back (before boundaries)</i>	62
<i>Figure 62 Geometry set over boundaries</i>	62
<i>Figure 63 Sharing edge</i>	63
<i>Figure 64 Multipatch with 3D line</i>	63
<i>Figure 65 Sharing point</i>	63
<i>Figure 66 Input for difference operation</i>	64
<i>Figure 67 3D Difference operation</i>	64
<i>Figure 68 Touching points – for non-manifold</i>	65
<i>Figure 69 Touching edge- for non-manifold</i>	65
<i>Figure 70 Touching surface – for non-manifold</i>	66
<i>Figure 71 3D section of the unsaturated subsurface</i>	70
<i>Figure 72 Main soil content</i>	71
<i>Figure 73 GeoTOP litho class attributes – colored</i>	72
<i>Figure 74 Groundwater levels as 3D surface between points</i>	73
<i>Figure 75 Area of interest: city district Dubbeldam</i>	74
<i>Figure 76 Demo interpolation</i>	74
<i>Figure 77 Part of the adopted, vectorised GeoTOP map sheet 44w</i>	74
<i>Figure 78 Modified TOP10NL data consisting of extruded surfaces</i>	75
<i>Figure 79 Working demo result</i>	75
<i>Figure 80 Modelling part of the sewer system</i>	75
<i>Figure 81 Modelling part of the sewer system and performed 3D intersection operation</i>	76
<i>Figure 82 Unsaturated zone and the sewer system do not overlap</i>	76
<i>Figure 83 Unsaturated zone and the sewer system overlap each other</i>	76
<i>Figure 84 Sewer pipes and sewer connection part in the unsaturated zone</i>	76
<i>Figure 85 Unsaturated zones in the Rechte Zandweg</i>	77
<i>Figure 86 WebScenes</i>	78

LIST OF TABLES

<i>Table 1 Litho classes</i>	38
<i>Table 2 Test Cases for testing</i>	51
<i>Table 3 Permeability</i>	71
<i>Table 4 Drainable porosity</i>	72
<i>Table 5 Groundwater levels via 3D points</i>	73

ACRONYMS

B-rep	Boundary Representation
CAD	Computer –Aided Design
CGAL	the Computational Geometry Algorithms Library
COLLADA	Collaborative Design Activity – interchange data format
CRS	Coordinate Reference System
GIS	Geographic Information System
GLG /GHG	Average groundwater levels – low and high

1. INTRODUCTION

1.1 Research Scope

1.1.1 Motivation and Relevance

The upper part of the subsurface in The Netherlands, yields the natural potential to take in abundant rainwater. In general, precipitation is taken up in the soil, surface waterbodies and in the sewerage. In the built environment, the natural soil is mostly covered by buildings, pavements and roads. In dense urban areas, precipitation can hardly infiltrate the soil. This phenomenon causes that the subsurface cannot use its full potential to temporary 'store' or infiltrate rain water – while through climate change there is a growing need to optimize water discharge in the subsurface during precipitation peaks (Bodemvenster, 2015).

In the context of a changing climate, rainstorm patterns have changed causing more heavy rainfall in shorter periods of time, which results in pluvial flooding especially in dense urban areas (NOS, 2015). In order to cope with and prevent pluvial flooding, multiple municipalities in The Netherlands recently planned to make substantial investments in upgrading the sewerage systems to enlarge their discharge capacity (VK, 2015). Three main sources of water can be distinguished in urban areas, namely precipitation, drinking water supply and seepage. These water sources are discharged via infiltration in the subsurface, evaporation and through runoff in the sewer system and surface water – which discharge respectively approximately 40, 37 and 23 percent in medium dense urban area. This balance of water discharges, depends on the coverage of the surface (Hooimeijer, 2014), the denser an urban area is, the more impervious surfaces are present that cause changes in the discharge balance, e.g. less discharge via infiltration is then possible. Besides man-made interventions such as the sewer system, a proper use of the upper part of the subsurface via open soil locations in and around cities can positively contribute in rainwater discharge issues.

The capacity of the subsurface to take in precipitation is related to the field of geohydrology and is more specifically dependent on (i) the type of soil and on (ii) groundwater levels or the phreatic level. While the geohydrologic conditions in The Netherlands are complex (Frapporti, 1993) – as can be seen in Figure 1 – specific knowledge about the subsurface is required to provide insights in the water infiltration and storage capacity of the subsurface.

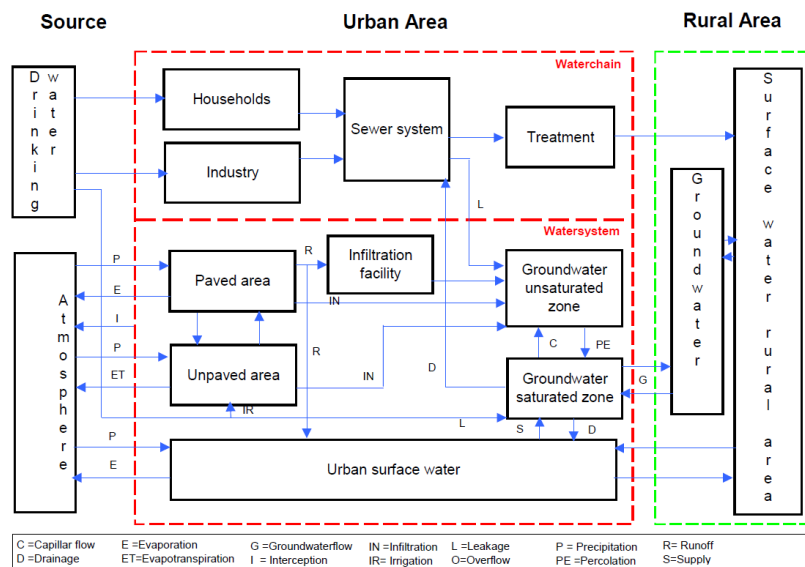


Figure 1 Urban water system (De Graaf, 2009)

Regarding the urban water system portrayed in Figure 1, 'unpaved area' and the 'groundwater unsaturated zone' are the main themes that are dealt with in the Application of this research.

Information about the soil types in The Netherlands can be found in the national Soil Map, which is distributed by Wageningen UR. The Soil Map (nld. Bodemkaart) is a two-dimensional map that provides

general classifications of the soil at about 1 meter below ground surface level. However, two significant deficiencies of the Soil Map can be drafted: (i) soil information on various, different depths are not provided and (ii) the map excludes soil type information of the built environment, areas which are solely classified as 'built area' (Wageningen UR, 2015), as seen in Figure 2.

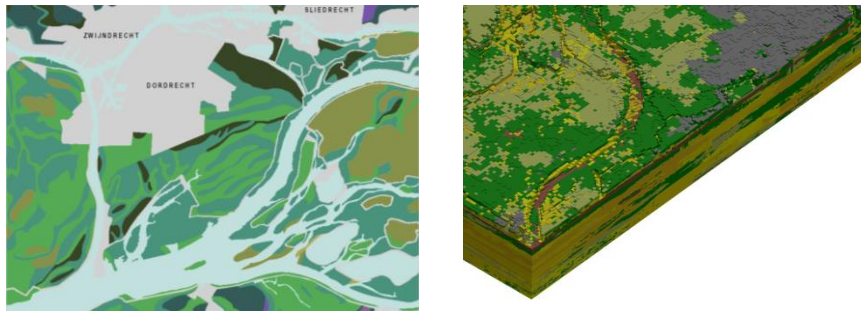


Figure 2 Part of the 2D Soil Map and part of the 3D GeoTOP

The TNO geological GeoTOP voxel dataset overcomes the aforementioned deficiencies, because it models in 3D the upper 50m of the subsurface in NL (Figure 2). Furthermore the GeoTOP model includes geological features of the soil per voxel – including the subsurface regions of the built environment. Three-dimensional intersection operations in GIS on the GeoTOP model, together with data of the built environment and with basic geohydrological principles (i.a. the permeability K and groundwater levels), can reveal new insights in the water infiltration and storage capacity of the subsurface on specified locations.

The result of the 3D spatial analysis through 3D intersection operations in GIS, can provide part of the explanation – regarding soil type properties – on why certain areas in the city that suffer from pluvial flooding after heavy rainfall. The outcome is an underpinning for the application of various suited measures that provide a resolution for rainwater discharge issues (e.g. the appliance of a wadi, or deep water infiltration – alternative measures are dealt with in section 1.2.4).

The novelty of this research is that 3D Boolean intersection operations in GIS are tested in their geometric validity and their ability to incorporate semantic and attribute information in the output. In this thesis, data conversions are proposed and built, to perform the three-dimensional intersection operations in the 3D vector surface data representation.

1.1.2 Research Requirements

The research focusses on the three-dimensional, Boolean intersection set operations in GIS. ArcGIS of vendor ESRI is used as the test environment because of the available *3D Intersect* set operation tool of the 3D Analyst extension. *Intersect 3D* adopts only 3D vector data, named as Multipatch geometry in ArcGIS. Therefore, all datasets that serve as input for the 3D intersection operations, have to be converted to 3D vector data.

The performed 3D intersection operations result in an Application that is able to unveil part of the subsurface geohydrologic potentials to cope with rain water discharge issues in urban areas. In order to provide such an Application for geohydrologic purposes, the following requirements to the research should be met.

Requirements regarding the input data:

To build the Application for geohydrology, the following input data is needed: GeoTOP that provides information about the soil types, TOP10NL that indicates open soil locations in the built environment and KLIC data which can be used to model the subsurface sewer system.

These input data sets, are of three different data representations and dimensions: 3D voxel data (GeoTOP dataset: modelled and accessed via the Netherlands Organisation for Applied Scientific research 'TNO'), 2D surface data (TOP10NL dataset: distributed by the Dutch National Mapping Agency 'Kadaster'), and 2D line and point data (KLIC data: distributed by 'Kadaster').

To make 3D intersection operations in ArcGIS, all three different input data require preprocessing steps to be converted to the same data representation and dimension: 3D surface based vector data. Three

distinct conceptual workflows – explicating step by step how to convert each different input data representation – are proposed to convert input data to 3D vector data, surface based.

Requirements regarding the 3D intersection operations:

The 3D intersection operations are executed on the georeferenced (i) 3D geological subsurface data of the GeoTOP model, and (ii) an extruded map of unbuilt terrain derived from the TOP10NL vector dataset. Also (iii) buffered KLIC data is used, comprising of subsurface cable and pipe data.

In GIS, semantics are regarded as the entitled layer (dataset) unveiling the meaning of the data. Attributes are additional information accompanying geometry, stored in a table and linked to the geometry via an ID. The intersection in 3D should maintain semantic and attribute information of the original input data, in the output. Attribute preservation is necessary to be able to link geohydrological principles (e.g. water infiltration capacity) to the attribute information provided with the 3D intersection output. Semantic preservation is needed to convey the meaning (e.g.

'3D_vectorised_GeoTOP_intersection_below_RechteZandweg') of the geometry output. To apply geohydrologic principles on the 3D intersection output of the Application, valid geometry is required to make volume and surface area calculations on the output possible (e.g. water storage capacity can be measured in volume quantity: m³).

A workflow is proposed that consists of a step by step method on how a semantic and attribute preserving 3D intersection operation can be performed. The workflow for 3D intersection operations on surfaces and the extent to which this workflow can be implemented in GIS, are investigated in this research. This investigation is done through various Test Cases that are drafted to test and evaluate the proposed 3D intersection workflow in ArcGIS based on three evaluation criteria that emanate from the aforementioned research requirements: (i) geometric validity, (ii) correct assignment of attributes and (iii) the possibility of volume/surface area calculations.

1.1.3 Problem Statement

Three-dimensional intersection operations on solid and surface based geometry, can be achieved in CAD, not in commonly used GISs. For instance in the CAD software MicroStation of vendor Bentley, an intersection between e.g. a surface and a solid can be performed via the *Compute Intersections* tool. This tool finds the intersection points between overlapping elements, and can only be used for 3D geometry (Bentley, 2015).

The datasets that are required for the aim of this thesis research, are georeferenced datasets that can be processed in a GIS. Therefore, 3D intersection operations should be performed within a widely used GIS, in order to be accessible by the users of the Application of this research.

1.2 Research Approach

1.2.1 Research Questions

The research in this thesis adopts the following main research question:

In what manner can a GIS facilitate 3D intersection operations for voxel data represented as surfaces, while maintaining semantic and attribute information in the output?

This question leads to the investigation of what data representation is necessary and suited for such 3D intersection operations, as well as what these operations entail. The main question is answered in a more general way that incorporates the investigated theories, and in a more specific way that is related to the Test Cases, the used test environment (ArcGIS) and to the Application. The main research question is divided into three sub questions:

- **What conceptual, generic workflow is required to retrieve a 3D intersection with surface represented objects?**
- **How can 3D intersection operations be performed in GIS?**
- **How can semantic and attribute information of the 3D intersection output be maintained in GIS?**

Key themes that will be dealt with in this thesis are:

- 3D Boolean set operations for spatial analysis
- Maintain semantics and attributes in the output geometry
- Valid geometry
- Voxel and vector data

These key themes will be explicated in the Background Information chapter of this thesis.

1.2.2 Application

The Application of this thesis is located in the municipality of Dordrecht, a city in the west part of The Netherlands, and which is surrounded by three rivers. Dordrecht is part of the ISDR – International Strategy for Disaster Reduction, and is therefore committed to their urban security regarding water related issues and flooding.

3D Intersection operations in GIS that meet with the evaluation criteria (valid output, with semantics, correct assignment of attributes, and volumes, or surface areas calculation), lead to the geohydrological Application of this thesis. This Application is a 3D GIS model built out of the 3D intersection operations with vectorised GeoTOP (accurate soil type information), TOP10NL and KLIC data (open soil locations and cables and pipes in the subsurface) respectively. The performed spatial analysis of the Application can provide new three-dimensional insights for the 'urban water task' (nld.: Stedelijke Wateropgave) of the city of Dordrecht. With the Application of this research, better decisions for urban water management can be made. The Application serves as a significant implementation of the performed 3D intersection operations in GIS.

Additionally, the Application incorporates groundwater level data in the 3D intersected output. This groundwater level data is represented as surfaces – the phreatic surface level – between points (groundwater measuring pipes) in the Application, to give an indication of the unsaturated zone (also stated as the drainage depth) of the subsurface. This zone is an important aspect in the determination of the geohydrologic potential of the subsurface – more refined elaboration on the applied geohydrologic principles is described in Chapter 6.

1.2.3 Method Description

The following four steps cover the approach that is adopted in this thesis research.

1. Investigate the possibilities of 3D intersection operations in GIS
 - The operations that enable a 3D intersection operation, and inquired based from theoretical perspective, and from the practical perspective which is focused on the GIS that is used.
2. Define a conceptual, generic workflows for data conversion and for 3D intersection of surface represented objects
 - Three different input datasets are used. The first is GeoTOP which are voxel, secondly the TOP10NL where only certain 2D surfaces are used, and lastly the KLIC comprising of 2D line and point data. For all three input datasets, a conceptual workflow is proposed that enables the conversion to 3D vector data, surface based.
 - Another workflow that is proposed, explicates the steps of performing a semantic and attribute preserving 3D intersection operation on 3D surface vector data.
3. Implement 3D intersection operations on vectorized voxel data with 3D vector data in a chosen GIS (ESRI ArcScene), while maintaining semantic and attribute information
 - The workflows for conversion are performed and the 3D intersection workflow is tested via various Test Cases. The Cases are tested and evaluated on valid geometry, with semantics, correct assignment of attribute and the possibility to perform volume and surface area calculation on the output. Specific cases are drafted in order to test the specific CGAL Nef_Polyhedral_3 class implementation of ArcGIS in so called non-manifold conditions.
4. Compare and evaluate the conceptual generic workflow for 3D intersection with the implemented 3D intersection in a GIS

The findings on the implemented proposed and built conversion workflows, and the implemented 3D intersection workflow, are compared and evaluated. The findings on the Test Cases and the performance of the workflows on the Application are dealt with.

The software that is used in this research are:

- Python 2.7.8
- MS Access and MS Excel 2010
- ESRI ArcGIS – ArcScene 10.3, ArcMap 10.3, ArcCatalog 10.3

1.2.4 Expected Results

3D Intersection operations performed on voxel data represented as surfaces in GIS, where the output geometry is valid geometry with maintained correct semantics and attributes. In the ideal case the output of a 3D intersection operation is of valid geometry with semantics and correctly assigned attribute. If the later conditions are not the case, post processing (e.g. COLLADA conversion or join operations on attribute tables of the input data) should overcome the issues. Furthermore, the resulting 3D intersection output offers a base to determine the following geohydrologic principles: the water infiltration and storage capacity.

Regarding the Application, insights retrieved out of the linkage between geohydrologic principles and the 3D intersection output of GeoTOP with TOP10NL, could result in guidelines regarding spatial measures or interventions in urban areas for the purpose of a better urban water system. The elaboration of these measures is not part of the scope this research.

Based on the spatial analysis outcome of the geohydrologic Application, possible spatial interventions could entail:

1. Whether an 'open soil' location should remain open to help tackle water related issues of the subsurface in an urban environment, or not.

Regarding this intervention: also initially closed locations of paved surfaces could be opened up for natural water infiltration. Or, permeable paving could be applied. The instalment of a wadi – initially dry gullies that temporary collect water after a rain – is also an intervention option.

2. To provide new insights regarding the maintenance or renewal of the sewer system.
3. To involve the potentials of the upper part of the subsurface in new spatial planning projects, e.g. to work with natural drainage principles powered by the subsurface.
4. To apply 'deep infiltration' (nld.: diepinfiltratie), in layers below impermeable soil layers that are situated just below the ground surface level.

And these four aspects should lead to:

1. A decrease in water related issues due to heavy rainfall in urban areas.
2. Possible saving in large, new investments in the sewer network.

1.3 Related Work

The voxel model is often used to model in 3D phenomena in the field of geology. A geological voxel model can describe the lithology or other geological characteristics of the subsurface (Guillen et al., 2008 and Jørgensen et al., 2013), meaning physical properties of a subsurface element can be numerically attached to a voxel. A 3D geological model is helpful in providing insights and understanding of geology (Jørgensen et al., 2013).

The following section covers found, documented research specifically related to the 3D geological GeoTOP voxel model of TNO:

- The 3D Pilot – ESRI voxel with 3D geo- objects, case RandstadRail tunnel Rotterdam, 2011:

This pilot project dealt with the question of which and what amount of soil types would be excavated by the RandstadRail tunneling in Rotterdam (Berntssen et al. 2012). In order to answer this question, two datasets needed to be used in the analysis: the 'raw' GeoTOP voxel dataset (.txt) and the central axis line of the tunnel in 3D vector data (.shp). After several conversions, various ArcGIS 3D Analyst tools were used that resulted in a GIS prototype and a graph portraying the excavated soil types together with volume calculations (Van Maren and Pluim, 2011).

Several issues occurred during this 3D Pilot case. First, the buffered cylinder geometry appeared to be no valid geometry, cause it was not closed. The buffered geometry (the RandstadRail tunnel) had to be exported to COLLADA, closed via manually editing, and subsequently imported as COLLADA in ArcGIS where it could be converted back to Multipatch geometry. Secondly, the *Intersect 3D* operation ESRI used for the 3D Pilot, did not automatically assign the attributes of the input geometry, to the intersecting output geometry. Attributes had to be assigned via after-processing (join operations) in ArcGIS. Lastly, volume calculations per selected set of attributes on the *Intersect 3D* operations were not automatically generated (Van Maren and Pluim, 2011).

- Geohydrological modeling, predictions for an area- orientated approach for groundwater contamination in the City of Utrecht, 2013:

This research project deals with the geohydrological modeling of contaminated groundwater flows – and plumes with chlorinated hydrocarbons, on a relatively large, area scale. An existing groundwater flow model of Arcadis was used and enriched by the GeoTOP voxel dataset together with its hydraulic conductivity property. Analyses by the flow model in the form of streamlines were performed in this project (Valstar & Maljers, 2013)

This thesis research builds upon the aforementioned projects by presenting a conceptual workflow that (i) converts voxel data, 2D surface data and 2D line data to 3D surface based vector data, and (ii) proposes a method to perform semantic/attribute preserving 3D intersection operations in GIS on 3D vector data.

The extent to which this workflow can be performed in ArcGIS is tested through drafted Test Cases, which are evaluated on three main evaluation criteria: (i) geometry validity, (ii) correct assignment of attributes on the intersected output geometry, and also (iii) volume/ surface area calculation.

Furthermore, this research covers an Application in which the basic geohydrologic principles of hydrological conductivity/ infiltration capacity and water storage capacity are determined, out of the 3D intersected GeoTOP model output with its litho class (soil types) attributes. The three evaluation criteria stated in the previous paragraph, are necessary to determine the geohydrologic properties of soil on a certain location. This matter can be explained via the following example:

Incorrect assignment of the GeoTOP litho class attribute (soil type), can lead to the assessment that a certain open soil location is not fit for rainwater infiltration, when it in fact is. This could be the case when instead of litho class attribute 7 (which is the soil type sand, suited for infiltration), litho class attribute 2 is assigned (which is soil type clay, not suited for infiltration).

1.4 Reading Guide

Chapter 2 includes background information that provides explanation in the overarching themes that are dealt with in this thesis; inter alia (i) data representation, and 3D intersection operations. Chapter 3 consists of a focus on the used datasets of this thesis research, which are GeoTOP, TOP10NL and KLIC. Chapter 4 contains proposed conceptual workflows for (i) data conversions, and (ii) 3D intersection operations that will be implemented on ArcGIS. Chapter 5 describes the testing and evaluating of both the conversion workflows, as the 3D intersection workflow, via various Test Cases, based on three evaluation criteria, (i) valid geometry, (ii) correct attribute assignment and (iii) volume/surface area calculation. Chapter 6 describes the Application which relates the 3D intersection output with geohydrology. Lastly, this thesis contains in Chapter 7, the overall conclusion, discussion recommendations and the reflection.

2. BACKGROUND INFORMATION

This Chapter entails the main background information and theories that are dealt with in this thesis research. Information about the distinction between CAD and GIS software, various 2D/3D data types in GIS, validity and 3D intersection operations are explicated.

2.1 CAD/GIS

Computer –Aided Design (CAD) and 3D Geographic Information System (GIS) software are initially invented for different purposes, and thus their difference in functionality is still apparent. In principle, GIS models the complex real world as it exists, while CAD models ‘man- made’ features that still need to be produced (Newell and Sancha, 1990). CAD deals with 3D design, editing and visualization. GIS has a focus on georeferenced data, attributes and spatial analysis (Pu and Zlatanova, 2006). Therefore, data that is adopted in GIS is usually of a large amount. In CAD, the data used, and the models created, are of a smaller size.

Issues in functionality difference:

- More supported 3D data representation/ primitives in CAD, such as cones, free form-curves, and also solids.
- Editing of 3D data is limited in GIS. Existing georeferenced data can be visualized and analyzed in GIS, editing this real-world data emphasized in the software.

Currently, CAD and GIS software develop into more similar systems – CAD adopts projections, attribute and more elaborate analysis, whereas GIS offers more refined 3D visualization techniques. Still, the adopted data types and file formats that CAD and GIS adopt, differ significantly. Furthermore, the multiple vendors of CAD and GIS software often use their own specific formats (Pu and Zlatanova, 2006). In the context of the limited availability of adopted primitives in GIS, the use of voxel data is not supported in commonly used GIS.

2.2 Data: Vector and Raster

2.2.1 Vector and raster representation principles

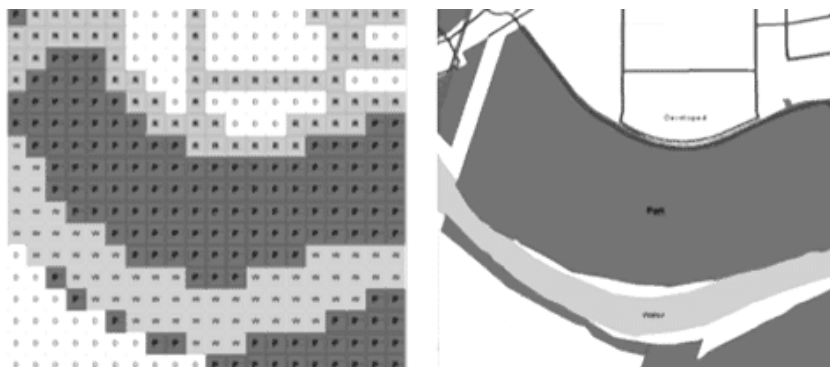


Figure 3 Raster and vector data (GIS Commons, 2015)

In GIS, the adopted data models can generally be segmented into two types: vector data and raster data, see Figure 3. Vector data uses point, lines and polygons for representation, while raster/grid data is represented by cells, pixels, or 2D squares describing a continuum (Zobl et al., 2011, Ledoux & Gold, 2007).

In vector data, lines and polygons are in fact sequences of points, raster data cannot be expressed in solely points according to geometric terms, but in pixels (Becker, 2012).

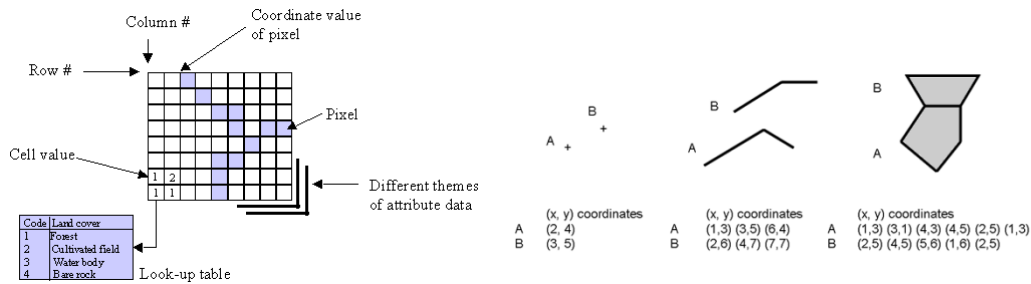


Figure 4 Raster data by (Yeung 1998) and point, line, polygon, (Restore, 2015)

In 3D, vector and raster data distinction remains: 3D vector data incorporates multiple z-values per distinct x,y position (Zobl et al., 2012), and voxel data adopts z-values per separate pixel. The diversity in both 2D as 3D data types is stated as geometric diversity (Zlatanova et al., 2013). The data models used in 3D, and 4D, GIS can be divided in two main types: objects can be represented either Surface-based, and by a Volume-based representation (Zlatanova et al., 2012, Gia et al., 2013). As seen in Figure 5:

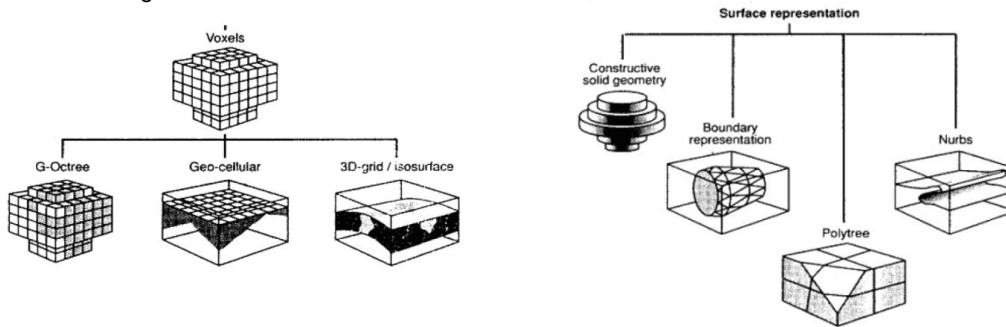


Figure 5 Volume and Surfaces (Lattuada, 2006)

Vector and raster data abstractions are fit for modelling respectively (i) 3D objects that are mainly used for objects that are internally homogeneous and discrete, and (ii) for continuous phenomena with properties of a continuous variation (Zobl et al., 2011, Zlatanova, 2012).

Surface-based representations can further be subdivided in: Constructive solid geometry (CSG), Boundary representation (B-rep), Polytree, Non Uniform Rational B-Splines (Nurbs) representation, and the Triangular Irregular Network (TIN). Volume representation is mainly practiced by Voxels, which comprehend a regular space division and which can be subdivided in a G-Octree, Geo-cellular and 3D grid or Isosurface representation (Zlatanova et al., 2012, Zobl et al., 2012), shown in Figure 5.

One of the used Surface-based representations is the B-rep – in fact a volume defined by its surrounding faces – that can practice operations such as union, subtraction and intersection (Zobl et al., 2012, Hegemann et al., 2013). The surfaces of a boundary representation are oriented in order to define interior and exterior of the shape. Becker et al. (2012) stated the hypothesis that each distinct data type, could be converted to a Boundary representation, while this transformation might not always be efficient concerning raster or voxel data. The complexity of a B-rep and the fact that no unique B-rep data structure exists, can be regarded as disadvantages of the data model (Zlatanova et al., 2012). De Cambay (1993) states that a boundary representation (B-REP) is efficient for visualization in GIS, but restricted to the polyhedral representation.

2.2.2 Three-dimensional Vector data

Vector data in 3D can be defined for a GIS in numerous formats, as stated by Zlatanova et al. (2012): multiple file formats for 3D data exist that are developed by international organizations or by vendors. Some of the 3D data formats became a standard by its widespread usage of both GIS users as software vendors, examples are .kml and the .shp file format. The .shp file originates from 1998, and is a binary format that links attributes to objects, provides vast drawing and editing functionalities, and consists generally of a main file, index file and a database file – respectively .shp, .shx, .dbf (Zlatanova et al., 2012).

Commonly, in GISs, 3D objects consisting of vector data are set as polygons of polylines in a 3D .shp file format (Hofierka & Zlocha, 2012). A 3D .shp file, stores vector data in x, y and z direction (Milner et al.,

2014), and is in a lot of cases used and practiced in an ESRI (Environmental System Research Institute) ArcGIS environment. The use of the .shp file enables geometry and attributes in a GIS, topological representation is not supported. In the latest ArcGIS 10, a .shp file can handle points, multi-points, polygons, polylines, but also Multipatches, all in accordance to the OGC standards (Zlatanova et al., 2012). While the .sph file is a vector representation, it automatically adopts the advantages of this vector data type, in 2D but also in 3D: boundaries are precisely described, real-world objects as perceived by humans are represented, the storage is compact and the overall representation of object with attributes happens directly (Arroyo Ogori et al., 2015, Zlatanova et al. 2012).

2.2.3 Validity: 2D Polygon and 3D Polyhedra

Beside points and line, a polygon is vector data. Multiple definitions and standards for valid geometry exist.

In this research, the valid polygons are defined according to the international Simple Features specification (SFS) of OGC, as formulated in (Arroyo Ogori et al. 2012),

- Closed topologically: the boundaries define one connected area (Arroyo Ogori, 2012)
- Holes can be present, when the geometry has an outer ring for its exterior boundary, and an inner ring for the interior boundary which defines the hole.
(Multiple holes can touch each other at one point, and a hole can touch the exterior boundary at one point).
- Contains no spikes and punctures
- And contains no overlap and gaps between multiple polygons (OGC, 2015)

The polyhedron is a 3D vector data representation. A polyhedron is a 3D primitive represented by boundaries and which is built out of multiple polygons, a polyhedron is thus of boundary representation. A polyhedron is defined as followed by Stoter and Van Oosterom (2006),

'A polyhedron is a bounded subset of 3D coordinate space enclosed by a finite set of flat polygons such that every edge of a polygon is shared by exactly one other polygon, adjacent polygons. The vertices and edges of the polygons are the vertices and edges of the polyhedron, the polygons are the faces of the polyhedron.'

Subsequently, in this research valid polyhedra are based on the set of rules formulated by (Stoter en Van Oosterom, 2006).

- A polyhedron that 'bounds one single volume, which means that from every point (including those on the boundary), every other point (including on the boundary) can be reached via the interior.' (Stoter en Van Oosterom, 2006). In other terms, a polyhedron is closed in Euclidean (x, y, z) space (Computer and Information Science UPenn, 2015).
- Comprises out of simplicities faces: faces should have a surface area, no (self-) intersection of faces occurs, touching can occur, no disconnected parts are present.

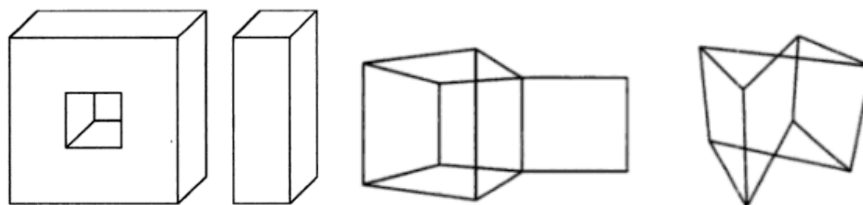


Figure 6 Valid polyhedra and invalid polyhedra (dangling face, intersecting faces) (Stoter and Van Oosterom, 2006)

2.2.4 Voxel data

A voxel is the consequence of taking a pixel value of raster data, to 3D with x, y, z values. Therefore, voxels (a merger of volume and pixel) are in fact 3D pixels that can describe objects in an array of voxels, which result on a larger scale in 3D fields (Zlatanova et al., 2012, Ledoux & Gold, 2007). A 3D field is equally split by intervals on the orthogonal coordinate axes, in order to get a x, y, z, voxel representation, which can be structured in an Octree data structure (Defu et al., 2010, Hegemann et al., 2013).

As a volume, a distinct voxel element consists of one or multiple values. Voxel models are suited to model continuum phenomena in the scope of climate, hydrology and geology (Zlatanova et al., 2012). In order to represent such continuous fields in voxels, these fields must be discretized into regular a

volume, which have as benefits: vast management, analysis and computation of the voxel model (Ledoux & Gold, 2007, Zlatanova, 2012).

The main downturns of voxels are: the enforcement of 3D regular volumes on geographical elements, causing disruption in analysis and resulting in rough visualizations, and the fact that high resolution 3D field representations result in a large data volume (Zlatanova et al., 2012, Ledoux & Gold, 2007). The smaller the size of the voxel, the higher the estimation accuracy and spatial detail level (grid resolution) becomes, for a realistic simulation, the 3D fields have a tendency to become of a large size (Hofierka & Zlocha, 2012, Varduhn et al., 2014, Glander & Dollner, 2008). Jørgensen et al. (2013) pointed out that handling a voxel model that usually consists out of millions of voxels, puts pressure on the performance of computer's hardware and software. Also, according to Ledoux & Gold (2012), the chosen resolution of the 3D field is of importance regarding the required or possible detail level, the determination of the exact position of a value with one voxel can be regarding indistinct, scalability is hard to achieve, and boundaries based on similar values are vague. Above all, voxels are best fitted to represent continuous spatial variation (Zobl et al., 2011).

Additionally, the support of voxels in a GIS is not commonly found (Neteler et al. 2011). A GIS that can work with voxel data is The Geographical Resources Analysis Support System (GRASS), an open source GIS. However, the 3D operations in GRASS GIS on voxel data is focused on 3D map algebra functions that are entered via formulas (e.g. is larger than, or is smaller than). Thus 3D difference, union and intersection Boolean set operations on voxels are not supported.

2.2.5 Multipatch in ArcGIS

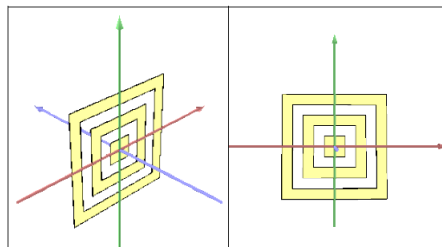
For vector representations, ESRI provides points, multipoints, polylines, polygons and the 3D geometry Multipatch. The geodatabase and graphic element systems of ArcGIS uses these geometries to construct their shape and graphics. Together with spatial references, the geometries can be located on a place on earth (ESRI, 2008).

In the commonly used GIS suite of ESRI, ArcGIS, the Multipatch geometry is the available 3D geometry for solids. Multiple patches together – which are in fact surfaces – form a Multipatch.

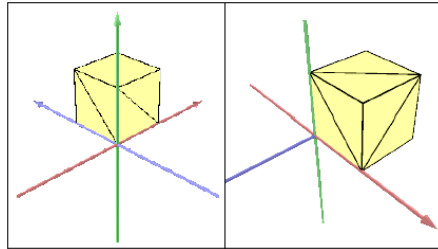
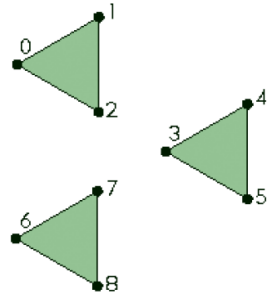
A Multipatch is built upon rings or (a collection of) the OpenGL 3D triangle primitives in the form of either strips or fans, representing 3D objects based on the polyhedron, seen in Figure 7. A combination of these surfaces compose a 3D space, thus the Multipatch feature represents a 3D object via its boundary (B-Rep). A requirement is that the points of the surface geometry on which a Multipatch is built, have a defined z- value (ESRI, 2008).

The Multipatch geometry type, may consist of:

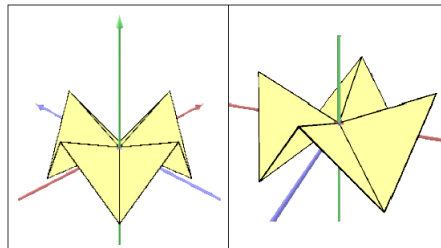
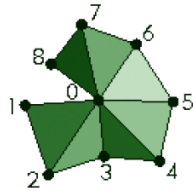
- Rings: polygons are based on rings that define an area via a closed sequence of elements. Only closed rings are valid, thus the first and the last vertex of a ring are the same. To set the interior and exterior of a polygon respectively, inner and outer rings are determined within a Multipatch. Furthermore, the vertices of a ring do not have to lie in one plane (flat surface) to be considered valid. An example:



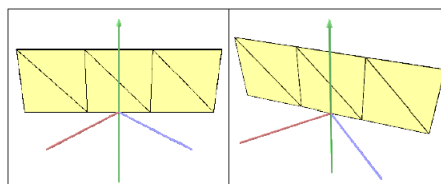
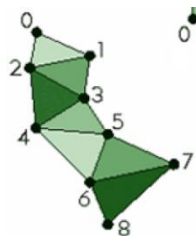
- Triangles: multiple 3D triangles, every set of three vertices define a triangle. Examples:



- Triangle Fan: a fan of 3D triangles, a first vertex defines the common point that all triangles adopt as origin. Examples:



- Triangle Strip: a strip of 3D triangles, each vertex after the first two vertices creates a new triangle. Examples:



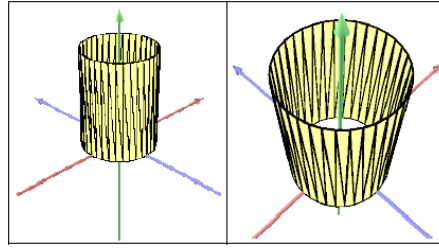


Figure 7 Multipatches (ESRI, 2008)

Geometrical simple objects (e.g. cubic objects), and more complex objects (e.g. ISO surfaces) can both be modeled as a Multipatch (ESRI, 2008).

The surfaces of a Multipatch can either be faced positive or negative – surfaces that face towards the exterior and interior respectively – determined by the distinct ordering of points, or orientation. Clockwise ordering of points establishes positive/exterior faced surfaces, while a counterclockwise ordering defines negative/interior faced surfaces. The orientation of surfaces is important in various analysis processes, such as determining interior volumes (ESRI, 2008).

Features in 3D can be stored as a table in an ESRI geodatabase with only one geometry field, and no point duplicates are stored. The fact that no repeating points of the Multipatch are stored, ensures a less expensive storage (Ford and James, 2005). However, it is possible that patches have common boundaries, which is not an issue if these boundaries do not penetrate. In the latter case, processes such as interior and exterior determination might result in errors.

With the introduction of the 3D Multipatch geometry, the availability of performing analysis in 3D was not yet available. Thus, only analysis in 2D – such as intersection operations – could only be executed on a 2D surface of the Multipatch, usually the footprint. With the arrival of the 3D Analyst extension, analysis on features in 3D became available.

2.2.6 Multipatch validity

Each of the geometry types, have restrictions (validity rules), e.g. polygon geometry has to have an interior area separated by its exterior. These constraints determine whether geometry is valid, and if the geometry is satisfied with its constraint, it is stated as 'simple' geometry. A geometry is labeled as 'not simple' (i.e. invalid), when it meets with the following two conditions: (i) a geometry restriction is violated, or (ii) it is not known whether the constraint is met (ESRI ArcGIS, 2015).

This means that a 3D Multipatch needs to be closed, and should further meet with the following requirements:

- A closed Multipatch built of surfaces, defines one distinct volume
- Multiple patches [surfaces] define the shell [B-REP] of the volume
- All patches have to have a counterclockwise orientation of coordinates
- Patches cannot intersect with each other
- No gaps or holes can be present in the shell

(ESRI ArcGIS, 2015)

The constraints for 2D geometry are different than the constraints for 3D geometry. Initially, the 2D Multipatch geometry type was set up to be unconstrained by 2D validity rules. Representing e.g. vertical oriented elements, or extrusions of 2D lines would not be possible with the solely 2D constraints (ESRI ArcGIS, 2015).

In ArcGIS, the Multipatch geometry type can be constructed via importing data from various file formats, e.g. the COLLADA format (ESRI ArcGIS, 2015). This format can be built in and exported out of a 3D modelling program, such as SketchUp by Tremble.

In ArcGIS, the main method to validate Multipatch geometry is to verify whether the geometry is closed (ESRI ArcGIS, 2015), and thus when in contains no intersecting faces or disconnections. The 3D Analyst extension provides an operator that can determine whether a Multipatch feature is closed or not, and lists this information as a new field in the attribute table of the feature. *Is Closed 3D* verifies if a Multipatch

completely encloses one volume (ESRI ArcGIS, 2015). The *Is Closed 3D* is demonstrated in the following Figure 8 (Python Script),

```

*****
Name: IsClosed3D Example
Description: This script demonstrates how to use the
            IsClosed3D tool on all multipatches in a target workspace.
*****
# Import system modules
import arcpy
import exceptions, sys, traceback
from arcpy import env

try:
    # Obtain a license for the ArcGIS 3D Analyst extension
    arcpy.CheckOutExtension('3D')
    # Set environment settings
    env.workspace = 'C:/data'
    # Set Local Variables
    for fc in arcpy.ListFeatureClasses(): # list features in workspace
        # Determine which features are multipatches
        if arcpy.Describe(fc).shapeType == 'MultiPatch':
            # Execute Is Closed 3D
            arcpy.IsClosed3D_3d(fc)

```

Figure 8 *Is Closed 3D* operator (ESRI ArcGIS, 2015)

Additionally, tools such as volume determination (*Add Z information*) or the solid operations in the 3D Analyst extension (*Difference 3D*, *Intersect 3D*, *Union 3D*), require closed Multipatch geometry as input, in order to construct valid output. Therefore, Multipatches which are not closed cannot be used for further analysis.

2.3 3D Intersection operations

2.3.1 Clipping

In other terms, 3D intersection operations can be referred to as 'clipping', or '3D clipping', terminology as used in the field of computers graphics. Clipping is defined as the complete process of determining what region of space is within another region of space. In computer graphics clipping is related to the rendering of 3D scenes for visualization on-screen. Redundant data can be neglected in the processing rendering due to clipping. Also, clipping can determine whether two features touch each other.

The Sutherland-Hodgman clipping algorithm – relevant in the context of 3D intersections of surface represented geometry and regarded as an algorithm for a Boolean operation on polygons – is often mentioned in literature (Agu, 2015, Pankaj, 2015). Its working principle offers a comprehensive description on the step by step methodology of intersection operations, and it will be explicated in this section.

Sutherland-Hodgman-Algorithm:

The basis of clipping, deals with a plane – in 2D or 3D – cutting through a 3D shape. Thus this 3D shape is split into two parts. The principle of the Sutherland-Hodgman-Algorithm will be explained here from the perspective of two dimensional geometry, though the algorithm allows an easy extension towards three dimensional geometry. A list of vertices as input geometry intersecting with list of planes, generates a new list of vertices as output, within the set of planes. When more planes are used to intersect with the input geometry, the Sutherland-Hodgman-Algorithm will clip one plane at a time. The figures below portray a simple 2D clipping process of (i) an intersection with one plane, and (ii) an intersection of multiple planes in steps, thus sequentially. Blue represents the splitting plane.

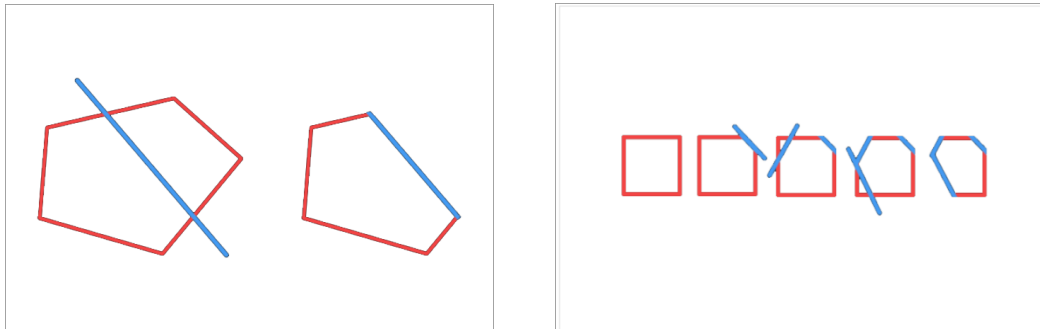


Figure 9 Clipping and splitting plane (Gaul, 2013)

More elaborated, the algorithm can be portrayed in case of clipping a polygon against a rectangle (e.g. screen). The first situation (i) portrays a clipping situation where lines are clipped without the principles of the Sutherland-Hodgman-Algorithm (linear time complexity, sequentially clipping), and (ii) which displays a correct clipping output.

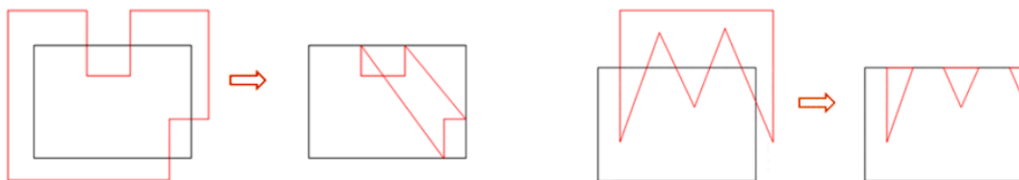


Figure 10 Clipping on screen (Sunshine, 2007)

Furthermore, orientation of the geometry relative to the splitting plane is important. This deals with the front and back – or interior and exterior/ inside or outside – of the plane, which can be determined via dot product (operation that takes two sequences of numbers and returns a single number) and thus the orientation of the normal of the plane. The result of distance d can be either positive or negative if n is not normalized. The point geometry is in front of the plane orientation if d is positive, if d is negative the point is on the back of the plane.

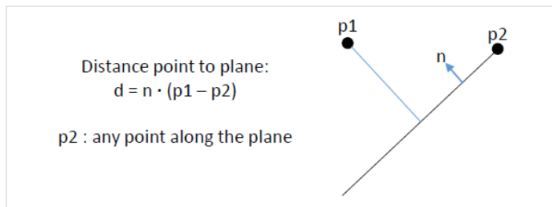


Figure 11 Point distance (Gaul, 2013)

The sequentially method of the Sutherland-Hodgman-Algorithm, adopts the following four steps, or 'scenarios':

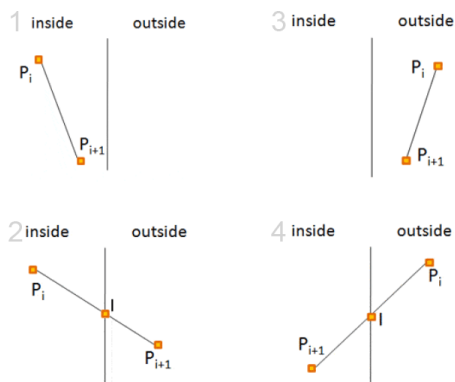


Figure 12 Inside outside, 4 steps (Sunshine, 2007)

1. Two points P_i and P_{i+1} are inside the region of the plane that performs the clipping operations. P_{i+1} can be added to the resulting polygon.
2. Point P_i is located inside the clipping region, point P_{i+1} is located outside. Point P_i of step 1 and intersection point I is added to the resulting polygon.
3. Two points P_i and P_{i+1} are outside the region of the clipping plane. In the resulting polygon, both points are neglected.
4. Intersection point I and point P_{i+1} are both added to the resulting polygon.

The above mentioned method can be described in a for-loop based pseudocode of the Sutherland-Hodgman-Algorithm:

```

for each clipping edge do
  for (i = 0; i < Polygon.length -1; i++)
    Pi = Polygon.vertex[i];
    Pi+1 = Polygon.vertex[i+1];
    if (Pi is inside clipping region)
      if (Pi+1 is inside clipping region)
        clippedPolygon.add(Pi+1)
      else
        clippedPolygon.add(intersectionPoint(Pi, Pi+1, currentEdge))
    else
      if (Pi+1 is inside clipping region)
        clippedPolygon.add(intersectionPoint(Pi, Pi+1, currentEdge))
        clippedPolygon.add(Pi+1)
    end for
    Polygon = clippedPolygon
  end for

```

Figure 13 Sutherland-Hodgman-Algorithm (Sunshine, 2007)

2.3.2 3D Boolean set operations

The term 3D Boolean set operations is affiliated with Boolean operations that refer to a framework for defining relationships based on a system of logic, and practices in e.g. in database queries or in computer graphics. Three basic Boolean operators are AND (narrows down), OR (broadens) and NOT (excludes).

The Boolean set operations on geometry are difference, union and intersection operations, but also complement and symmetric difference (Hachenberger, 2007). The difference Boolean operator removes the overlapping part of two input geometries, the union Boolean operator takes two input geometries and merges them into a single output geometry, and the intersection Boolean operator outputs only the overlapping part of two input geometries (Michigan Technical University, 2015).

Boolean operations are used in Constructive Solid Geometry (CSG), which are models built out of elementary shapes – or primitives, e.g. spheres and cubes. These simple geometric shapes can together construct more complex geometry, in CSG (De Cambray, 1993 and Biermann et al. 2001).

Only a few solid computational representations remain closed geometry after Boolean operations, resulting in an invalid geometric output. The boundary representation (B-Rep) is a commonly used surface-based representation for to perform Boolean operations on. With complex B-Rep geometry, Boolean operations can become expensive as they need to calculate overlap of parametric surfaces, subdividing those surfaces and subsequently compute new geometry as output. However implementing Boolean operations on objects should go rapidly (Biermann et al. 2001).

Figure 14 shows the difference, union, intersect 3D Boolean set operations with primitives, and with more complex geometry.

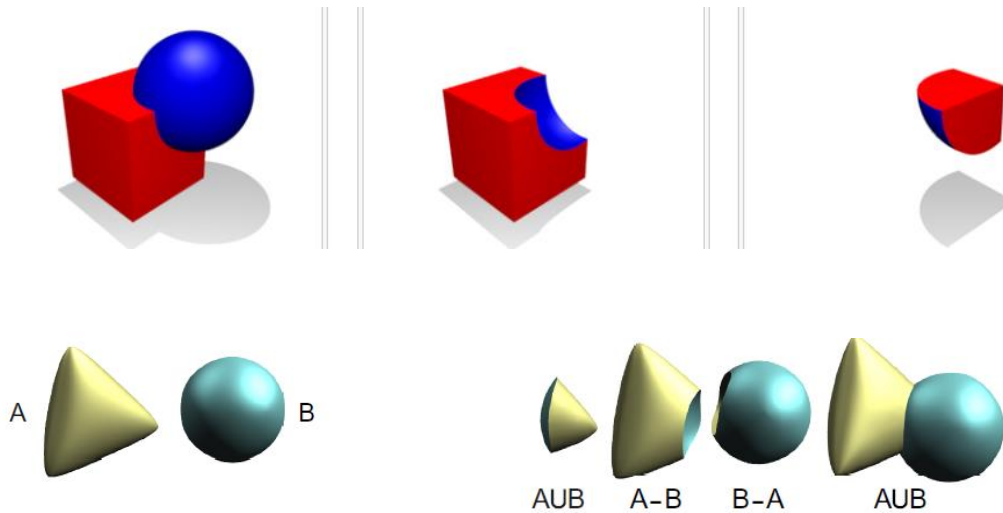


Figure 14 3D Boolean set operations (Wikipedia CSG, 2015) and (Biermann, 2001)

The three 3D Boolean set operations can be visualized as followed:

- For a difference (subtraction)
- ∪ For a union (merger)
- ∩ For an intersection (common parts)

With 3D Boolean set operations, spatial analysis can be performed by assessing relationships between geometries.

2.3.3 Nef Polyhedra main theories

A Nef Polyhedron is a comprehensive and general mathematical description of polyhedra in any arbitrary dimension, formulated by Walter Nef in 1978 (Hachenberger, 2006, CGAL Nef_3, 2015). A regular 3D Polyhedron (section 2.2.3) is mathematically less refined than the Nef Polyhedra. The Nef Polyhedra are regarded as a precise and complete model (Hachenberger, 2006).

Nef invented a new type of polyhedra that offers a general, simplistic and applicable polyhedra close to the common conception of polyhedral in Computers Graphics. Though, while meeting these aforementioned properties, the Nef Polyhedra are a fully sound mathematical theory (Bieri, 1995). Nef did not create new types of operations that form a closed scape, but adapted the definition of a polyhedron into a new one. A Nef Polyhedron can represent different dimensional features and non-manifold conditions (Hachenberger, 2006). A manifold can be described as followed: every vertex located on the boundary of a manifold solid geometry, subdivides the modelling space into an inside region and outside region of space. Whenever at any vertex the boundary of the modelling space is not subdivided into an inside and outside region, then geometry is non-manifold at that vertex (DS Spatial, 2015).

A non-manifold represents for instance a 3D Boolean unified polyhedron consisting of two cubes that only share a point (i.e. at this non-manifold vertex there are two inside regions, instead of an inside and outside region of space), or share an edge or face, as illustrated in Figure 15 (DS Spatial, 2015). An example of a complex Nef Polyhedra is shown in Figure 16 (Hachenberger, 2006).

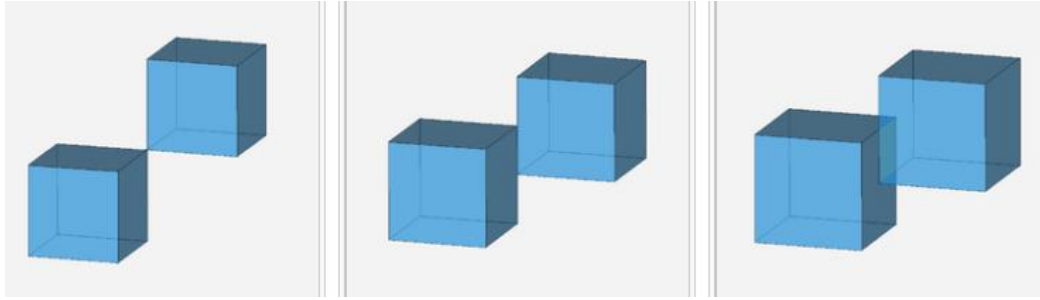


Figure 15 Non-manifold vertex, edge, face (DS Spatial, 2015)

Figure 16 shows a 3D example of a Nef Polyhedron, which is non-manifold, contains lower dimension features (dangled face), and isolated vertices (Hachenberger, 2006).

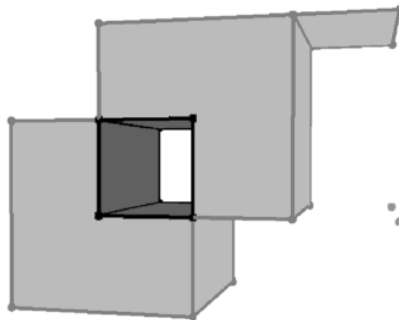


Figure 16 Nef Polyhedra (Hachenberger, 2006)

Accompanied with the new polyhedral theory, Nef founded the following notions, among others: a class for polyhedra in d -dimension, that is closed concerning the main set operations and closed regarding the topological operations closure and interior. Furthermore Nef gave methods to perform the operations complement and intersection, and a new definition of faces of a polyhedral (Bieri, 1995). A face is an equivalence class of 'local pyramids', which define the local space that surrounds a point. This notion is shown in Figure 17, for a 2D planer example of Nef Polyhedra, where only the bold edges and bold nodes define Nef Polyhedra, and the 'local pyramids' are hatched in a small circle (Hachenberger, 2006).



Figure 17 Local pyramids (Hachenberger, 2006)

Regarding data structures, Nef Polyhedra could be stored e.g. via their 'local pyramids', stated as the Wurzburg structure (Hachenberger, 2006).

The 3D Nef polyhedron is a defined subset in 3D space, which is the result of finite set intersection and set complement operations on open 3D half spaces ((Hachenberger, 2006, CGAL Nef_3, 2015). This leads to the fact that a Nef Polyhedron is closed geometry under the Boolean operations intersection, union and difference (CGAL, Nef_Polyhedra_3,2015).

A geometrical half space is one of the two sets that results out of a splitting hyperplane in a 3D target shape. Where a hyperplane is a subspace of one dimension less than the target shape (e.g. if the target shape is 3D, the hyperplane is 2D) (Wikipedia halfspace, 2015). An open half space is solely either of the two sets that was generated of the splitting hyperplane. When the splitting hyperplane is united with an open half space, a closed half space is the result of that.

The arrangement of hyperplanes is stored at vertex level, which allows storage of attributes, in the vertices, as in the spherical faces (Arroyo Ohori, 2015).

2.3.4 Nef Polyhedra in CGAL

The Computational Geometry Algorithms Library (CGAL) project houses various geometric computation algorithms and data structures, in a C++ (programming language) library. The GCAL library is adopted in various applications, in the field of computer graphics, and in GIS and CAD. Various research institutes, universities and companies are involved in the CGAL project.

CGAL provides data structures that are able to model with 3D Nef Polyhedra, and to execute Boolean and topological operations on them. Important to state that CGAL models the Nef Polyhedra as B-REP data structures – closed with Boolean set operations. Also, regarding topology, a Nef Polyhedron is closed with the operations closure, interior, exterior and boundary.

This B-REP data structure is then separated into two data structures. The first data structure comprises the local neighborhood of vertices and offers a comprehensive description of the geometry. The second data structure implements edges, facets and volumes on the neighborhoods of the first structure (CGAL Nef_3, 2015).

Related to the first data structure is the sphere map of a vertex. A sphere map is the position and mark of the vertex and the local pyramid (Hachenberger, 2006). The local pyramid of a vertex adopts the topological and geometric aspects and situation of the vertices, and is embedded on a sphere (CGAL, Nef_Polyhedra_3, 2015). Solely the sphere map structure is in theory sufficient to represent the overall geometry (Hachenberger, 2006).

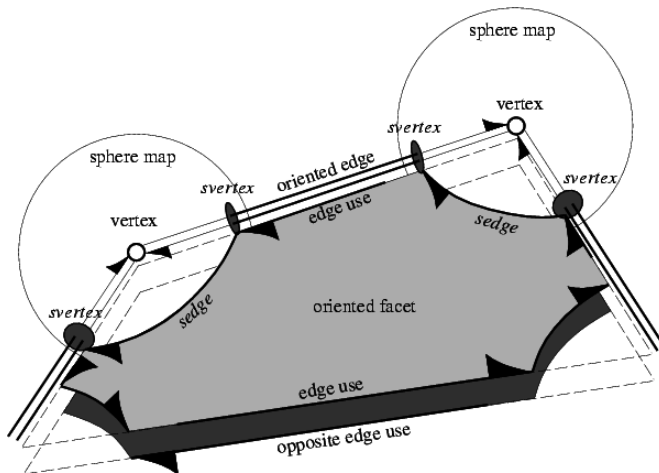


Figure 18 Sphere map principle (CGAL Nef_3, 2015)

CGAL implements the 3D Nef Polyhedra in the class `Nef_polyhedron_3`. In total, a `Nef_polyhedron_3` comprises of the following elements:

- Vertices V
- A sphere map for every vertex in V
- Edges E , where each edge is represented by two half-edges
- Facets F , where each edge is represented by two half-facets
- Volumes C
- A mark/label for every elements
- An incidence relation on every item (CGAL, Nef_Polyhedra_3, 2015)

A half-edge is generated by splitting an edge through its length. Two half edges together form a complete edge, and each of the half edge has an opposite orientation (clockwise, or counter-clockwise) (Flipcode, 2015).

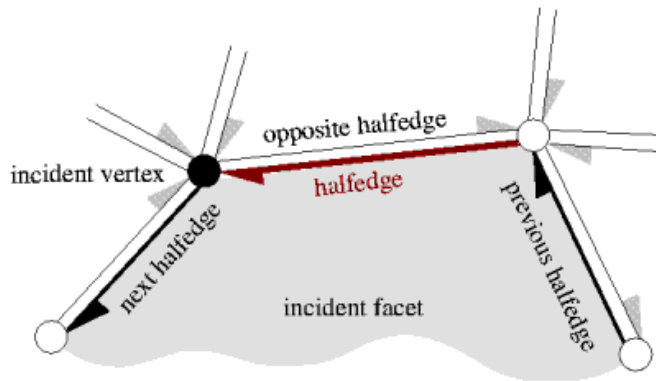


Figure 19 Half-edge structure (CGAL Halfedge, 2015)

In the CGAL half-edge data structure, an incident face and one incident vertex are stored in a half-edge. The data structure properties give insight in adjacency and incidence (CGAL Halfedge 2015).

2.3.5 3D Analyst in ArcGIS

For 3D data, ArcGIS has the ArcScene software that adopts 3D geo-data in 3D Euclidean space. ESRI ArcGIS has an extension called '3D Analyst' that comprises of various geoprocessing tools, and is subdivided into several toolsets, e.g.: the *3D Features toolset* for 3D spatial analysis on features, and the *Data Management Toolset* and *Conversion Toolset* for data management and data conversions respectively.

For this research the *3D Features toolset* of the 3D Analyst extension provides the required analytical 3D operations. In this toolset, spatial analysis can be performed through operations that adopt mainly 3D vector data (Multipatches) and evaluate their geometry and relationships. The toolset incorporates 3D Boolean operations in the form of the solid operations *Difference 3D*, *Union 3D* and *Intersect 3D* (ESRI ArcGIS, 2015).

Furthermore, the *3D Features toolset* provides the following operations:

- *Intersect 3D Line With Multipatch:*

This operation adopts 3D Multipatch geometry and 3D polyline geometry as input, and generates the intersection. This intersection output can be points which represent the intersection, and lines which are divided at the intersecting points. The *Intersect 3D Line With Multipatch* has some restrictions regarding geometry and attributes. One is that the operation only adopts 3D features as input. This means that the line which intersects with the Multipatch, has to consist out of points with x, y, z coordinates. When a line is in 2D but has z coordinates – height – defined in its attribute, a conversion is needed in order to use this line in 3D space (ESRI ArcGIS, 2015)..

Also, the attributes of specifically the line input can be maintained in the output, this joining of attributes has to be explicitly entered before the *Intersect 3D Line With Multipatch* operation is performed.

The *Intersect 3D Line With Multipatch* is demonstrated in the following Figure 20 (Python Script),

```

*****
Name: Intersect3DLineWithMultiPatch Example
Description: This script demonstrates how to
            use the Intersect3DLine tool.
*****
# Import system modules
import arcpy
import exceptions, sys, traceback
from arcpy import env

try:
    # Obtain a license for the ArcGIS 3D Analyst extension
    arcpy.CheckOutExtension('3D')
    # Set environment settings
    env.workspace = 'C:/data'
    # Set Local Variables
    inLineFC = 'sample.gdb/lines_3d'
    inMP = 'sample.gdb/test_MP'
    # Ensure a unique name is produced for output files
    outPoint = arcpy.CreateUniqueName('OutPt_3DIntersect', 'sample.gdb')
    outLine = arcpy.CreateUniqueName('OutLine_3DIntersect', 'sample.gdb')
    # Execute Intersect 3D Line with Multipatch
    arcpy.Intersect3DLineWithMultiPatch_3d(inLineFC, inMP, 'IDS_ONLY',
                                           outPoint, outLine)

```

Figure 20 Intersect 3D Line With Multipatch (ESRI ArcGIS, 2015)

- **Buffer 3D:**

This operation adopts 3D point geometry and 3D polyline geometry, and generates an array area on a specified distance. The output is a transformation of the point and line geometry to 3D Multipatch geometry. The *Buffer 3D* operation complies with several constraints on geometry. The buffer output for 3D points is a Multipatch sphere, the output for a 3D polyline is a Cylinder, and the output is a valid, thus closed Multipatch (ESRI ArcGIS, 2015). Furthermore, geometry output is not generated when the Multipatch output cannot be closed, due to e.g. a complex polyline, or a buffer distance resulting in intersecting buffers. A buffer can consist of a certain amount of segments used to construct the Multipatch geometry with.

The *Buffer 3D* is demonstrated in the following Figure 21 (Python Script),

```

*****
Name: Buffer 3D Example
Description: This script demonstrates an application of
            the Buffer 3D and Inside 3D tools.
*****
# Import system modules
import arcpy
from arcpy import env
import exceptions, sys, traceback

try:
    # Obtain a license for the ArcGIS 3D Analyst extension
    arcpy.CheckOutExtension('3D')
    # Set environment settings
    env.workspace = 'C:/data'
    # Set Local Variables
    inFC = 'lineFC.shp'
    bufferOut = 'buffer3d.shp'
    # Execute Buffer 3D
    arcpy.Buffer3D_3d(inFC, bufferOut, '15 Meters', 'Round', '30', '1 Meters')
    arcpy.Inside3D_3d(bufferOut, 'survey_pts.shp', 'inside_analysis.dbf')

```

Figure21 Buffer 3D (ESRI ArcGIS, 2015)

Also, interior the *Add Z Information* tool in the *3D Features toolset* is able to calculate e.g. the interior volume of a closed Multipatch, and its surface area.

3D Boolean operations are three solid operations in ArcGIS: *Difference 3D*, *Union 3D* and *Intersect 3D*. Their functioning an output is visualized in Figure 22:

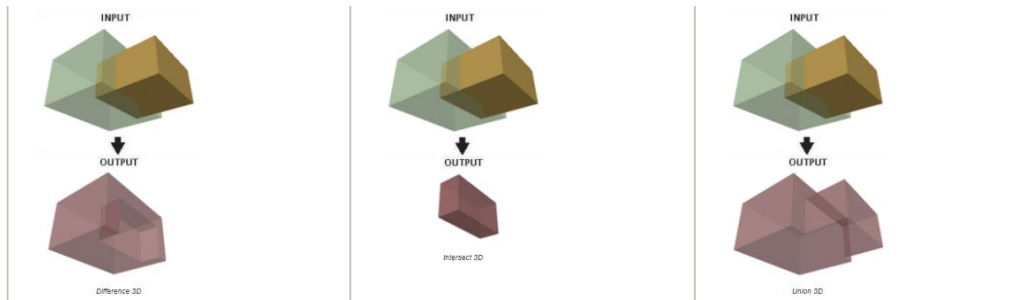


Figure 22 Difference, Intersect and Union in 3D(ESRI ArcGIS, 2015)

The solid operations can perform spatial analysis, and subsequently construct new geometry as features. Also these operations provide for geometric quality control as they required valid, closed 3D Multipatch geometry as input.

Difference 3D: input 1 is the target Multipatch and input 2 is the deduction Multipatch. This solid operation takes away that part of the target Multipatch that overlaps with the volume of the deduction Multipatch. When a deduction Multipatch is completely enclosed are disposed from the output geometry (ESRI ArcGIS, 2015).

The *Difference 3D* is demonstrated in the following Figure 23 (Python Script),

```

*****
Name: Difference3D Example
Description: This script demonstrates how to create
            shadow volumes that fall along a specified surface using the
            Difference3D tool.
*****
# Import system modules
import arcpy
import exceptions, sys, traceback
from arcpy import env

try:
    # Obtain a license for the ArcGIS 3D Analyst extension
    arcpy.CheckOutExtension('3D')
    # Set environment settings
    env.workspace = 'C:/data'
    # Set Local Variables
    inMP = 'buildings.shp'
    eraseMP = 'bldg_extensions.shp'
    outMP = arcpy.CreateUniqueName('bldgs_without_extensions.shp')
    # Execute Difference3D
    arcpy.Difference3D_3d(inMP, eraseMP, outMP)

```

Figure 23 Difference 3D (ESRI ArcGIS, 2015)

Intersect 3D: this operation requires two Multipatches as input, and constructs the geometric intersection of the two. The generated output may vary depending on the input geometry. *Intersect 3D* delivers a Multipatch output if (parts of) volumes overlap with each other.

The *Intersect 3D* is demonstrated in the following Figure 24 (Python Script),

```

*****
Name: Intersect3D Example
Description: This script demonstrates how to use the
            Intersect3D tool
*****
# Import system modules
import arcpy
import exceptions, sys, traceback
from arcpy import env

try:
    # Obtain a license for the ArcGIS 3D Analyst extension
    arcpy.CheckOutExtension('3D')
    # Set environment settings
    env.workspace = 'C:/data'
    # Set Local Variables
    inMP1 = 'Boston_MP_Small.shp'
    inMP2 = 'Boston_MP.shp'
    # Ensure output has a unique name
    outMP = arcpy.CreateUniqueName('Intersect.shp')
    # Execute Intersect 3D
    arcpy.Intersect3D_3d(inMP1, outMP, inMP2)

```

Figure 24 Intersect 3D (ESRI ArcGIS, 2015)

The operations *Intersect 3D* and *Difference 3D* both require comprehensive computation as they iterate through each distinct feature of the first input geometry, and iterate through every feature of the second input geometry. For each of these iterative steps (i) overlaps are determined, (ii) the geometric intersection is computed, and (iii) a new geometric feature is generated.

Union 3D: two input Multipatch geometries are combined if they overlap with each other, into one new output Multipatch geometry. The operation adopts an iterative process over the geometry of the shells (i.e. triangles or rings) of the input geometries that intersect, and removes these unnecessary parts. For the *Union 3D* operation, the input features need to overlap each other with volumes (ESRI ArcGIS, 2015).

The *Union 3D* is demonstrated in the following Figure 25 (Python Script),

```

*****
Name: Union3D Example
Description: This script demonstrates how to use the
            Union3D tool.
*****
# Import system modules
import arcpy
import exceptions, sys, traceback
from arcpy import env

try:
    arcpy.CheckOutExtension('3D')
    # Set environment settings
    env.workspace = 'C:/data'
    # Set Local Variables
    inMP = "multipatch.shp"
    # Ensure output multipatch has a unique name
    outMP = arcpy.CreateUniqueName("union_output.shp")
    outTbl = arcpy.CreateUniqueName("UnionTable.dbf")
    GroupField = "Type"
    optimize = "DISABLE"
    solids = "ENABLE"
    #Execute Union3D
    arcpy.ddd.Union3D(inMP, outMP, GroupField, optimize, solids, outTbl)
    arcpy.CheckInExtension('3D')

```

Figure 25 Union 3D (ESRI ArcGIS, 2015)

2.3.6 ArcGIS and CGAL

The 3D Analyst tool in ArcGIS software adopts CGAL source code to perform 3D operations. In order to do so, the Multipatch geometry type of ESRI is converted to CGAL data structures belonging to the Nef Polyhedra. This conversion is done 'behind the scenes'. CGAL is also used for other solid operators of ESRI (Crawford, 2015).

To be able to convert a 3D geometry (in another file format than that of CGAL), to a CGAL polyhedron object, an incremental polyhedron builder can be used – Polyhedron_incremental_builder_3, which outputs a CGAL Nef Polyhedra and uses a half-edge data structure. During the conversion and generation of the polyhedron, conditions of its surfaces are validated. The incremental property of the polyhedron builder implies that first all the point with coordinates are listed, and then the polygons. Edges are derived from the latter two geometry parts (CGAL, Polyhedron 2015).

Once (e.g. Multipatch) geometry is converted into a Nef_Polyhedron_3, the Boolean operations can be performed.

The actual call for a Boolean intersection operation – located in the Nef_polyhedron_3 class under Binary Set Operations – can be done in two ways:

<code>Nef_polyhedron_3< Traits ></code>	<code>intersection (const Nef_polyhedron_3< Traits > &N1) const</code>
	return the intersection of N and N1.
<code>Nef_polyhedron_3< Traits ></code>	<code>operator* (const Nef_polyhedron_3< Traits > &N1) const</code>
	return the intersection of N and N1.

These calls imply that the two inputs Nef_polyhedron_3 *N* and Nef_polyhedron_3 *N1* result in the output Nef_polyhedron_3 *N2*.

Via either $N2 = N * N1$, or via $N2 = N.intersection(N1)$

The stand-alone script in Python that expresses the Intersect 3D operation at its base:

```
import arcpy
from arcpy import env

env.workspace = 'C:/data'
arcpy.Intersect3D_3d('inMultipatch1.shp', 'outMultipatch.shp',
                    'inMultipatch2.shp')
```

The *Intersect 3D* operation in GIS automatically adopts semantic information in the form of layers, and all attribute information belonging to both of the input geometries. First, all geometry objects are found and secondly all their attributes are placed together into the intersecting output (Crawford, 2015).

2.4 Resume Background Information

The first section of this Chapter dealt with the difference between CAD and GIS software. Although the functionalities of the two different software get more similar, CAD has a focus on man-made modelling and editing with a large amount of primitives, while GIS deals with real-world georeferenced data, attributes and spatial analysis.

Two distinct data types can be adopted in GIS: vector and raster data, in 2D or in 3D. Vector data in GIS consist mainly out of points, lines, polygons and polyhedrons that model discrete boundaries, raster data are grid cells that describe a field. Two main type can be distinguished for the 3D data models used in GIS: Surface-based (e.g. B-rep) or Volume based (e.g. voxel).

A polyhedron is a 3D, boundary represented vector data representation. Its validation requires inter alia closure of the geometry. Voxel data is in fact 3D grid cells, thus 3D raster data. Only few GISs adopt the voxel data representation.

Based on polyhedra, the 3D Multipatch geometry of ESRI is built out of triangle primitives or rings. The validity of a Multipatch depends on closure of the geometry and the lack of gaps in the geometry.

The second section of this Chapter dealt with existing methods to perform a three-dimensional intersection operation on surfaces: 3D Clipping via the Sutherland-Hodgman clipping algorithm, and the theory and practice of the Nef Polyhedra, in ArcGIS. All methods have the ability to provide valid geometry as 3D intersection output.

A 3D intersection operation on surface geometry is a common operation in the field of Computer Graphics, and is commonly referred to as 3D Clipping. Here, only the data that is on-screen needs to be represented in order to improve further processing.

While the main methodology of 3D Clipping focusses on portraying geometry on-screen, it does describe the principles on how to perform a 3D intersection operation on geometry represented as surfaces. To ensure closed geometry output of a 3D intersection operation, the Nef Polyhedra theory offers a comprehensive and sound method.

Intersection in 3D is a three-dimensional Boolean operation. The Nef Polyhedra theory and CGAL implementation provides for a 3D B-REP (surface-based) model which is closed under 3D Boolean operations.

A GIS that makes use of the theories of 3D Clipping and the Nef Polyhedra for spatial analysis, should be able to offer a comprehensive method to generate valid output for 3D intersection operations on geometry represented as surfaces.

Though both Clipping in computer graphics and the mathematical description of Nef Polyhedra are very extensive methods for 3D intersection operations, their implementation is initially not built for GISs. In GIS geometry objects are accompanied with attributes, and semantics can be expressed in layers. The workflow described in Chapter 5, is proposed to investigate and resolve the matter in which 3D intersection operations on surfaces can maintain semantics and attributes.

3. USED DATA

In this Chapter, the three main input datasets – GeoTOP, TOP10NL and KLIC – used for this research are described in detail. This data will be used for the Test Cases in Chapter 5 and for the geohydrologic Application in Chapter 6.

3.1 Voxel: GeoTOP

3.1.1 Subsurface Models of TNO

The subsurface can be subdivided into the deep subsurface (at a depth of approx. 5km), the shallow subsurface (till 500m depth) and the upper part of the subsurface (till 50m depth). Depending on location, each of these subsurface parts has their distinct benefits and potentials, such as mineral resources, geothermal heat, CO₂ storage on a relative large depth, and thermal energy storage, underground spatial planning (e.g. for infrastructure: tunnels and underground constructions) and groundwater flows on a shallow depth in NL. The Geological Survey of The Netherlands – as part of TNO: Netherlands Organization for Applied Scientific Research – made four static models of the subsurface. Two of these models, Digital Geological Model (DGM) and Regional Geohydrologic Information System (Regis-II) are layer based models. The other two models are modeled in 3D via voxels: NL3D with a voxel resolution of 250*250*1 m, and the more detailed GeoTOP model with a voxel resolution of 100*100*0.5 m (TNO, 2015).

3.1.2 GeoTOP Introduction

The GeoTOP model describes that part of the subsurface that is most extensively used by people: the upper part of the subsurface (TNO, 2015). Since GeoTOP is a static model, it doesn't model dynamic groundwater flows. Each distinct voxel contains as attribute, information about the geological entity, the soil type, and the chance of occurrence of a type. The subsurface domain that GeoTOP models, mainly consists out of the litho classes, sand, grind, clay and peat, ascribed via a stochastic simulation technique.

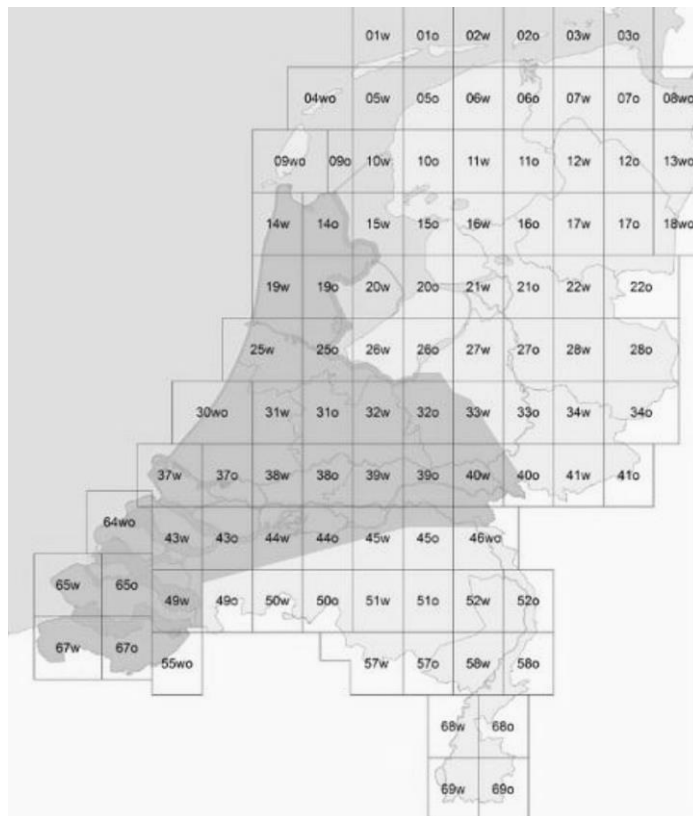


Figure 26 Overview of NL with the division of the GeoTOP grids/ map sheets. Map sheet of the Application: 44w (Stafleu et al., 2015)

Modeling GeoTOP is a multi-year project of TNO. Therefore, the GeoTOP doesn't have nationwide coverage yet as GeoTOP is still being modelled. Some parts of the Netherlands, such as the South West of the country and the major river regions, are finished and available for download. Other parts of The Netherlands will follow the coming years.

3.1.3 Modeling GeoTOP

The basis of the GeoTOP models are ground drillings performed in the Netherlands, and stored in the DINO (Data and Information of the Dutch Subsurface) database. Interpolation techniques provide the conversion of drilling description towards voxel, via automated processes.

Besides drilling data, several other sources are used to compose GeoTOP as realistic as possible. A selection of this additional data, is listed below:

- Geotechnical probe inquiries: litho classes can be derived from the cone resistance and friction ratio parameters. The advantage of probe inquiries is that they are mostly conducted in urban areas and near infrastructure.
- Geological Map of the Shallow Subsurface of NL.
- DGM, Digital Geological Model.
- REGIS-II, Regional Geohydrologic Information System.
- AHN, Actual Height file of The Netherlands: used for setting the ground surface level and its geological properties. The AHN raster of 5*5 m is scaled up towards the required raster size of 100*100 m. Surface waterbodies heights are derived from bathymetric soundings.
- Soil Map of NL: available at a scale of 1:50000, and developed until 1995 via hundreds of thousands ground drillings in the Netherlands (Alterra, 2012)
- Geomorphological Map, of NL.
- LGN, Nationwide Land use of NL.
- Topographic map 1:50000, by Kadaster: the map sheets of the GeoTOP model correspond with the sheets of this topographic map.
- Aerial Imagery: used to be able to verify the adopted ground surface level.
- Lithostratigraphic Nomenclator of the Shallow Subsurface: used for the lithostratigraphic units in NL.

Plus, in the GeoTOP work process geological experts ensure the quality control (Stafleu et al., 2015).

From interpolation onwards, a stratigraphic layer model is constituted. This layer model is then 'filled' with assigned voxels.

3.1.4 GeoTOP voxels

Map sheets of the GeoTOP voxel dataset can be downloaded from the TNO DINO website as .svp extension. This format is built for the voxel viewer software that TNO offers. The .svp file contains various data that ensure the TNO voxel viewer to read and portray the voxel data, among others layer data in the form of rasters in .asc and mv.asc, ground drilling data in .blg and voxel data in .csv. To retrieve the 'raw' voxel dataset in .csv, the .svp file needs to be zipped.

The voxels of GeoTOP describes the upper part of the subsurface in the Netherlands geometrically as a 3D grid, where each distinct voxel has uniform properties. The voxels are solids, which enclose multiple attributes per voxel. To ensure georeferenced appliance, the location of a voxel is stored in x, y, z, coordinates of the centroid of the voxel. In order to process voxel data, voxels contain numerical values only: integers or floats. Thus, to assign a string as voxel attribute, an encoding of string towards numerical value is required.

The actual voxel data is developed by TNO and is based on the ArcASCII format of ESRI, with a 3D expansion. This TNO format ensures the assignment of multiple attributes per voxel. The voxel dataset is a comma separated values extension (.csv). The .csv file of a GeoTOP voxel dataset, contains a header of 14 rows which completely describes the structure of the voxel data. Below the header, the actual voxel data is listed. Each voxel is positioned on a distinct row in the file, with its multiple attribute values, separated by commas (Stafleu et al., 2015).

The header contains the following 14 elements, in order:

- gridtype a regular or irregular shaped grid are possible
- sorting sorting order of the voxel data within the total grid/ map sheet; the z value changes first (from the bottom to the top), then the y value (from South to North), then the x value (from West to East), the '+' indicates an sorting from small to big in the context of coordinate values, a '-' would indicate a sorting from big to small
- nx, ny, nz of the grid covers the amount of voxels in x, y, z direction and therefore determines the shape of the overall grid/ map sheet; there are in total $nx*ny*nz$ voxels in the dataset
- x, y, z center of the origin 3D coordinates of the lower left (ll) corner of the grid, defined as the centroid of the first voxel in the dataset
- dx, dy, dz, of the voxel size size of a voxel, $100*100*0.5$ m
- nodata_value NULL value
- attribute names possible attribute names
- attribute types possibilities of attribute types in integers and floats

The 'raw' voxel data is described as follows:

- in integers or floats attributes are separated by commas, each second integer portrays the litho class, there are as many attributes per voxel, as described in the attribute rows in the header

Affiliated with the ASCII grid format, the voxel dataset is thus encoded as followed:

```

Header:
(14 rows)  gridtype regular
           sorting +Z +Y +X
           nx 200
           ny 250
           nz 185
           xllcenter 100050.0
           yllcenter 400050.0
           zllcenter -49.75
           dx 100.0
           dy 100.0
           dz 0.5
           nodata_value 999.0
           lithostrat,lithoklasse,kans_1_veen,kans_2_klei,kans_3_kleilig_zand,kans
           _5_zand_fijn,kans_6_zand_matig_grof,kans_7_zand_grof,kans_8_grind
           int,int,float,float,float,float,float,float,float
Voxel data:
(9.25mlj rows) 35,6,0.0,0.04,0.06,0.06,0.82,0.02,0.0
                28,6,0.04,0.02,0.04,0.23,0.67,0.0,0.0
                28,1,0.09,0.02,0.09,0.13,0.67,0.0,0.0
                12,3,0.0,0.14,0.76,0.02,0.07,0.01,0.0

```

In the upper encoding format, each distinct voxel does not have its own coordinates. Working with a header and setting only the coordinates of the centroid of the first voxel (thus the origin, having of all voxel-coordinates the lowest coordinate values), reduces the file size of the overall voxel dataset. Though, via this encoding method, the way of sorting a grid/ map sheet of GeoTOP has significant importance. Plus, voxels that contain nodata, are necessary to list as well in order to guarantee the distinct sorting of the total dataset and to fill the cubical GeoTOP grid/ map sheet completely. The section figures below portray the necessity of the nodata voxels with reference to the sorting of the voxels (Stafleu et al., 2015).

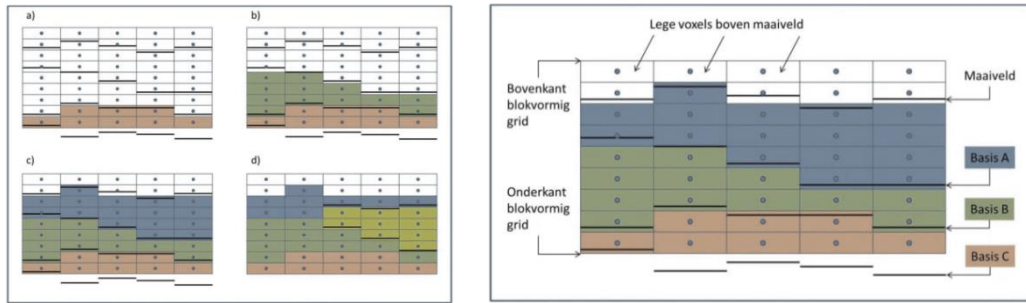


Figure 27 GeoTOP section (Stafleu et al., 2013)

3.1.5 GeoTOP litho classes as attribute

The two main attributes of a GeoTOP voxel are the lithostratigraphic unity and litho class attribute. Other attribute values described the chance of a voxel to be assigned to a certain litho class, The lithostratigraphic attribute deals with numerical assignment of a GeoTOP voxel to specific geological layers and formations. The attribute litho class of a GeoTOP voxel is the most significant in this research. A litho class is derived of a classification of soil types. This soil type indicator is encoded as a numerical value in GeoTOP, e.g. the integer 2 is linked to soil type clay, and the integer 7 is linked to coarse sand (Stafleu et al., 2015). The complete overview – including the soil type abbreviation and median of the grain size, if applicable – is as followed:

Litho class	Integer	Abbreviation (NL)	Grain size
Peat	1	o	
Clay	2	k	
Clayey sand/ Loam	3	kz	
Fine sand	5	zf	≥ 63 µm and < 150 µm
Midst sand	6	zm	≥ 150 µm and < 300 µm
Coarse sand	7	zg	≥ 300 µm and < 2 mm
Grind	8	g	≥ 2 mm
Shells	9	she	
Sand – unknown grain	10	z	unknown

Table 1 Litho classes(Stafleu et al., 2013)

In this thesis, the GeoTOP map sheet 44w is adopted. This map sheet covers those parts of the municipality of Dordrecht which are relevant for investigation in the Application. As TNO maintains the map sheet division of the Dutch Topographic map 1:50000 by Kadaster seen in Figure 26, a small part of the northern area of the city of Dordrecht is located outside map sheet 44w, but is instead present in map sheet 38o. The Application however, did not require investigations of open soil location in the north of Dordrecht, therefore map sheet 38o is not taken into account.

3.2 2D surface: TOP10NL

The TOP10NL is a 2D vector dataset of the topographic map of The Netherlands. The project around developing TOP10NL started with TOP10vector, which was solely a digital map with a nationwide coverage and thus lacked a topographic database (Kobben, 2003). The demand to use the TOP10vector in GIS, initiated the TOP10NL project as the TOPvector data-model project 2.0, initially developed by the Dutch National Mapping Agency (Kadaster), ITC Enchede and the universities of Delft and Wageningen. First, prototypes and the new data model design were proposed, which eventually lead to a new database for Top10vector. The requirements for the data structure of TOP10NL were that it should be an object-oriented data model, with singular ID's, stored in a database, attached non-spatial attributes, and the datasets had to be in line with OpenGIS standards to provide linkage and exchange (Kobben, 2003 and De Vries, 2002).

Currently, TOP10NL is accessible as open data in the OpenGIS standard GML: Geography Markup Language. GML is an open exchange modelling language formulated by the Open Geospatial Consortium – OGC, to build geographical features (OGC GML, 2015). GML can represent features with geometry and attributes, and is built out of a schema describing the data and an instance document that contains the data (ESRI, 2015), and can be read and exported to .shp file format in e.g. ArcGIS. The GML-format is based on XML, which is the Extensible Markup Language, initiated to store and transport structured data on the internet (W3.org, 2015). With GML geographic objects can be represented via its geometries, among others: points, line strings and polygons. An example GML data of the initial prototype of TOP10 NL is displayed in Figure 28:

```
<tdn:WegDeel fid="TOP10.402814">
  <tdn:top10_id>2400001</tdn:top10_id>
  <tdn:bronRef xlink:type="simple" xlink:href="metadata.xml#TOP10.9000017"/>
  <tdn:object_begindatum>2001-12-17T13:24:10+02:00</tdn:object_begindatum>
  ...
  <tdn:toegankelijkheid>Openbaar</tdn:toegankelijkheid>
  <tdn:status>In gebruik</tdn:status>
  <gml:polygonProperty>
    <gml:Polygon srsName="EPSG:28992">... </gml:Polygon>
  </gml:polygonProperty>
  <gml:geometryProperty>
    <gml:LineString srsName="EPSG:28992">
      <gml:coordinates>
        106045.396,449000 105965.228,448967.705 105619.713,448827.507
      </gml:coordinates>
    </gml:LineString>
  </gml:geometryProperty>
  <tdn:wegtype>Autosnelweg</tdn:wegtype>
  <tdn:hoofdverkeersgebruik>Snelverkeer</tdn:hoofdverkeersgebruik>
  ...
  <tdn:wegnummer>A12</tdn:wegnummer>
  <tdn:wegnummer>E25</tdn:wegnummer>
  <tdn:wegnummer>E30</tdn:wegnummer>
</tdn:WegDeel>
```

Figure 28 GML TOP10NL prototype example (De Vries, 2002)

Currently, TOP10NL adopts GML 3.2. The objects in TOP10NL account for multiple representations – point, line and polygon – that indicate topographic areas as surfaces, and e.g. a road network as lines and nodes (Kobben, 2003). TOP10NL encompasses a level of detail referring to a representation on a scale of 1:10.000, and its objects can have multiple attributes with an ID and for instance date and short descriptions.

In The Netherlands, the main topographic map TOP10NL is part of the Base Registration Topography (BRT) and built out of several topographic basic objects, defined in object classes such as: road, rail, water, building, terrain. Sources of TOP10NL are aerial imagery, panoramic imagery, field recordings and external source material (Kadaster, 2015).

Figure 29 shows two parts of the online accessible Web Map Service of TOP10NL where only the object class 'terrain' is opened:



Figure 29 Top10NL terrain (Kadaster, 2015)

In this research, a two-dimensional clip of the 2D vector data (surface-based) TOP10NL is used, in .shp file format as delivered via the municipality of Dordrecht. This particular section covers the region of the

municipality of Dordrecht, which is used in the Application. Furthermore, only the object class 'terrain' is adopted, providing areas of either built or open terrain (roads are not taken into account), represented as 2D surfaces.

3.3 2D line and points: KLIC

The Dutch National Mapping Agency (Kadaster), provides as well the KLIC data which holds information about the location of cables and pipes in the subsurface and is initiated to prevent and limit damage when excavations are performed (Kadaster KLIC, 2015)

The objects in KLIC account for two representations points and lines. KLIC is built out of the following objects: cables and pipes for water, electricity, internet, oil and gas (Kadaster, 2015). Also the sewer system is and connection parts are available in KLIC. The aforementioned objects can have multiple attributes and are all represented as 2D line strings and points.

Initially KLIC of The Netherlands is delivered topographic maps as raster data, however the original data is XML data. In practice, the XML data is often read and exported to .dgn or .dwg format by CAD software.

Figure 30 shows a part of the KLIC data.

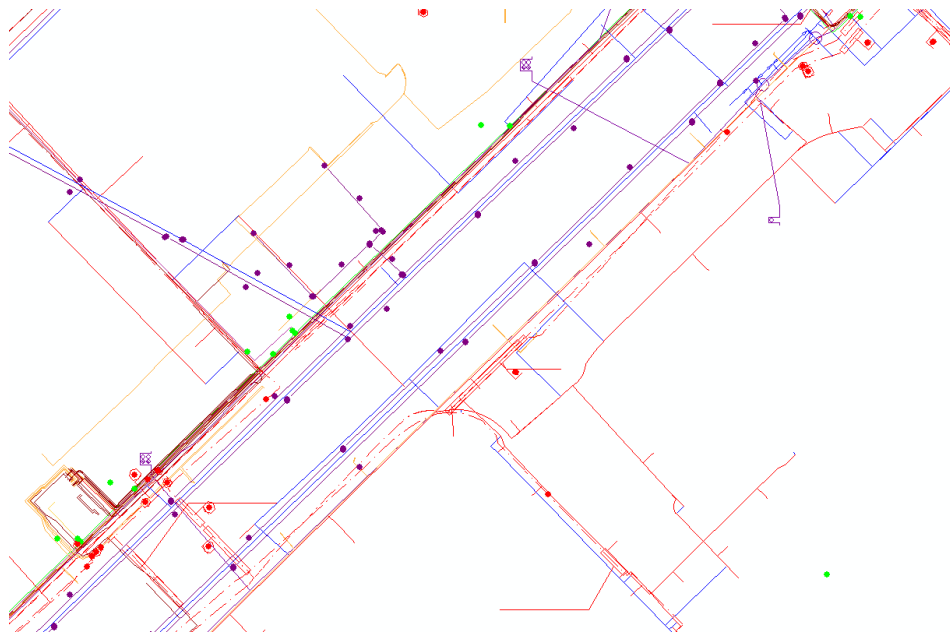


Figure30 KLIC data

Notable, the tubes of the sewer system are represented as 2D line strings, while drains and the connection parts of multiple sewer tubes are represented as (multiple) 2D points. In real-life however, the sewer connection parts are relative large elements that have a complex geometry to ensure a solid connection between 2, 3 or 4 sewer tubes. The geometry of the sewer connection parts are reconstructed in one of the cases described in section (5.2). On Figure 31, solely the sewer system of the 2D KLIC vector data is presented as positioned in the subsurface under the road network.

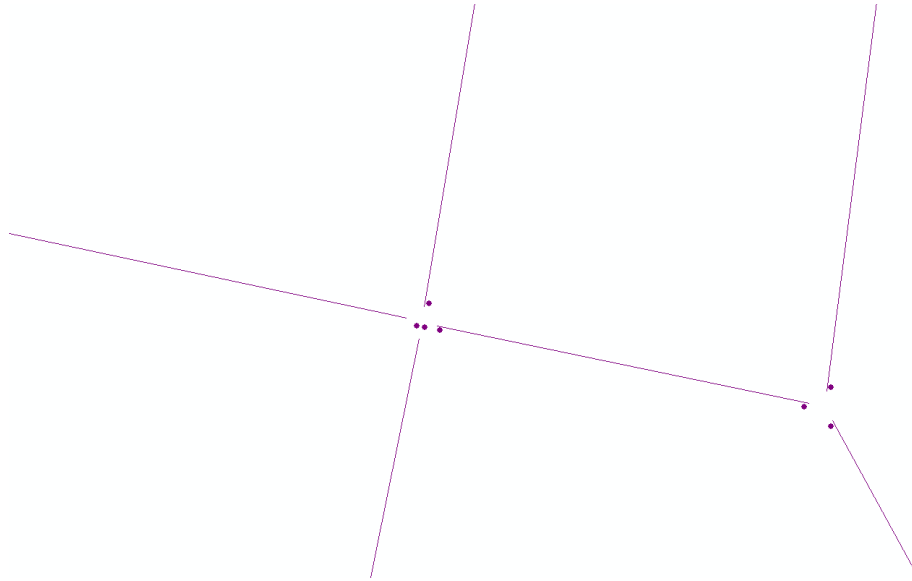


Figure 31 Sewer system in KLIC

In this research, KLIC data of 2D vector data (lines) is used, in .dgn CAD format as delivered via the municipality of Dordrecht. The KLIC data contains the cable, pipes (including sewerage) of the city of Dordrecht and its surrounding areas. Only the lines representing the sewer systems are adopted represented as 2D line strings and points.

4. CONCEPTUAL WORKFLOWS

The conceptual workflows that are proposed in this Chapter, contains steps that convert voxel data, 2D surface data and 2D line data to 3D surface based vector data.

The intended 3D intersection operations that will be performed, is regarded as a 3D Boolean set operation. The Boolean operations are considered as suited to perform on boundary representations – B-rep (Zobl et al., 2012, Hegemann et al., 2013). As the B-rep is 3D surface based vector data, the proposed conceptual conversion workflows all have 3D surface based vector data as output.




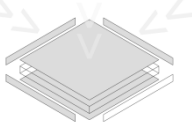
The workflows proposed in this Chapter, comprise of steps that (i) depict a manner to convert voxel data, 2D surface data and 2D line data to 3D vector data, and (ii) to perform valid 3D intersection operations in GIS that maintain semantics and attributes in the output.

This Chapter is built up as followed: first, the workflows for both the conversion as the 3D intersection operation are described in text. Secondly, a step by step proposal is presented, accompanied by conceptual diagrams.

4.1 Conversion Workflows

4.1.1 A workflow for voxel data to 3D vector data, surface based

The 'raw' voxel dataset can be used to start the conversion. Voxel data may be stored in ASCII format, where the defining properties of the dataset are stored as header. The header includes information about the voxels themselves (e.g. size), and the dataset as a whole (e.g. coordinates of the lower left voxel and the ordering of the dataset). A script that incorporates these properties can convert raw voxel data to 3D points. 3D points are a common geometry type adopted by various programs, and relatively cheap regarding data storage. The 3D points can be built as cubic, boundary represented shapes taking into account the dimensions of an original voxel.

Text	Diagram
1. Retrieve 'raw' voxel dataset [.txt / .csv] together with the distinct ordering of the file [ArcASCII sorting, or other]	
2. Interpret each voxel as being built up out of eight vertices (defining a cubic shape), or as having a centroid, Point Geometry	
3. Convert via a script the listed raw voxel dataset into 3D Point Geometry with x, y, z coordinates and with the attributes	
4. 'Build' each distinct set of points or each centroid point as a cubic B-rep – representing the boundary surfaces of a voxel	

4.1.2 A workflow for 2D surface data to 3D vector data, surface based

2D surface data can be converted into 3D surface based vector data via the following workflow. This conversion can be achieved by extrusion of the flat surfaces. First, 2D polygon data of a map needs to be retrieved, of which x and y coordinates are known. From the original data, the relevant part can be selected via 2D clipping, a process that excludes polygons that lie out of the splitting plane. Clipping can

also ensure the exclusion of point and line data which don't represent surfaces. The extrusion of 2D surface data adds z- value information to the 2D surfaces, in either positive or negative direction. The process of extrusion can be described as copying the 2D surface data on a vertical axis and between the copied and original polygons, new 'z- enabled' surfaces are generated. The conceptual conversion in diagram:

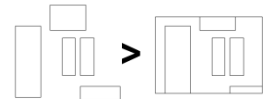
Text

1. Retrieve the original surface data consisting of polygons, load/import in a CAD or GIS format

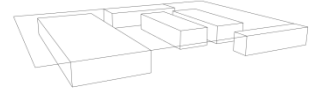
Diagram



2. Clip in 2D the relevant part of the data



3. Extrude the clipped map, meaning to expand a 2D surface in vertical direction to generate a 3D shape, where the fixed surface footprint remains intact.



4.1.3 A workflow for 2D line data to 3D vector data, surface based

2D line data can be converted to surface based 3D vector data with the following workflow. Buffering the line data is the main method to perform the conversion. Before the buffering can take place, the x and y coordinates of either the point geometry, or of the points from which a line is built, need to be retrieved. The z- coordinates can be obtained via attributes, via accompanied text or can be assigned. Coordinate information can be incorporated into a script that outputs 3D points with x, y, z coordinate. If multiple 3D points represent a line, a line between these points can be constructed. This line is now 3D vector data, to represent it by surfaces, buffering takes place. The process of generating a buffer comprises of defining a bounding shape around the line geometry, at a specified distance that serves as the diameter, and alongside all point on the line. The buffer operation delivers closed geometry when the ends of the line are enclosed by circles, which close the buffered shape via their bounding polygon. As shown in the diagram:

Text

1. Load/ import the 2D line and point data in a CAD or GIS readable format

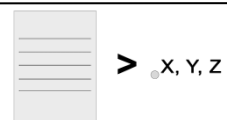
Diagram



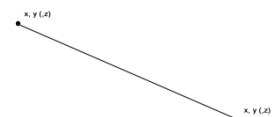
2. Retrieve the 2D coordinates of the points that the line is built of, and retrieve additionally the z coordinate of these points (as attribute or in text)



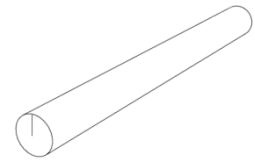
3. Convert via a script to output x, y, z points



4. Construct a line between the 3D points, or construct lines between multiple points in case of kinked line data, curved lines are not considered



5. Buffer the lines with the correct diameter, meaning a bounding shape is defined at a specified distance alongside all points on the line



6. Ensure closure of the buffered geometry. When gaps of overlapping, buffered kinked lines occur, enclose the buffer via COLLADA.



Notable regarding the point data:

Points itself can as well be converted to 3D surface vector data, via buffering as described in the latter workflow. Buffered 3D points would result in a sphere.

However, a point can indicate an object that is in fact more complex geometry, which is the case in this research. Then, converting a 2D point to a 3D point or sphere does not suffice: more elaboration is needed. This can be achieved by editing the 3D point in modelling software supporting the interchange format COLLADA.

More specific: commonly used GIS, such as ArcGIS, supports COLLADA data, and the modelling software SketchUp supports COLLADA as well. Thus the exchange between COLLADA data of the two programs is possible. This workflow is shown in Figure32.



Figure 32 COLLADA workflow

4.2 3D Intersection Workflow

In order to make 3D intersection operations with different datasets, the three input dataset are converted to one data type, which is the 3D surface based vector.

4.2.1 A workflow for 3D intersection operations on surfaces preserving semantics and attributes

A 3D intersection operation is a 3D Boolean set operation that can be conducted on 3D vector data built out of surfaces. The workflow comprises of a step by step method how such an operation can be performed. To start with, a 3D intersection operation requires at least two input geometries in the same data representation, in this case 3D surface vector data; boundary representation (B-rep).

The two input geometries must have an overlapping part, or one input object is placed within the other input object. The overlapping volume is – in this stage – bounded by the surface boundaries of the input objects. If the two geometries exactly overlap with each other (there bounding box is the same), no new intersection geometry is made.

The overlapping boundary parts should define the 'cutline', on which new points are generated that are able to define the boundary surface of the overlapping part.

Via these new points on the cutline, new surfaces can be computed, accompanied with the attributes of both the input geometries. The new geometries can be stored in a data file, which is titled to define the semantics (meaning) of the output geometry.

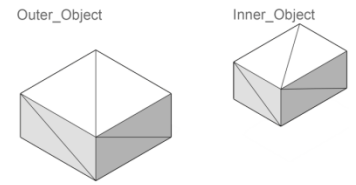
The intersection output geometry can be verified on closure of its geometry, and the attributes of the input geometries need be assigned correctly to the intersection output. As explicated in the workflow diagram:

Text

Diagram

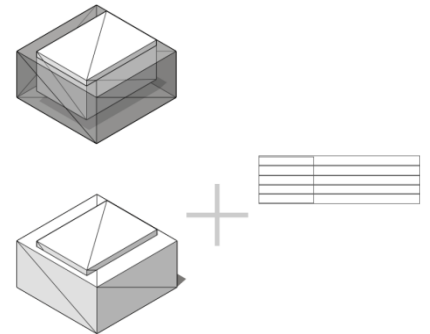
1. Both objects should be represented as surfaces (B-REP)

Else, pre-conversion steps needed

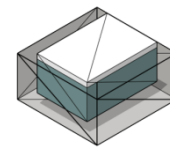


2. Place B-REP Inner_Object and B-REP Outer_Object within each other

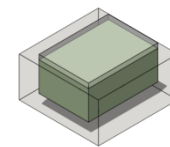
2.5 Include semantic and attribute information of both geometries (object-oriented)



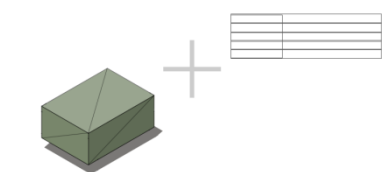
3. Cut Outer_Object with the boundaries of Inner_Object



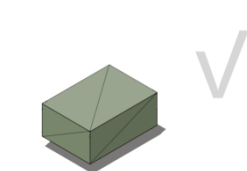
4. Create new Points in the Outer_Object which define the cutline that was generated in step 3.



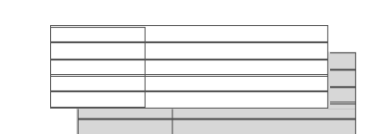
5. Generate the new Intersection_Object that adopts the semantic and attribute information of Out_Object



6. Validate Outer_object with new surfaces to close its geometry on the cutline



7. Analyse attribute information based on volume content calculation of Intersection_Object



When the intersection concerns multiple objects, step 3 (Cut Outer_Object with the boundaries of Inner_Object) and step 5 (Generate the new Intersection_Object that adopts the semantic and attribute information of Out_Object) needs to be performed one distinct object at a time.

The three conceptual conversion workflows will be implemented, in order to perform 3D intersection operations in the test environment ArcGIS. In ArcGIS, 3D vector data in boundary representation is available as the Multipatch geometry type.

5. TESTING AND EVALUATION

Chapter 2 and 3 provides the theoretical descriptions of both three- dimensional intersection operations on surface geometry, as of the main input data and further processing. Before actual implementation of 3D intersection operations on surfaces in GIS, a workflow is presented in the previous Chapter 4. This workflow should cover the vectorization of the input data and offer a base for the 3D intersection operations performed in the Test Cases.

This Chapter is built up as followed: first, the implementation of all four workflows, as proposed in Chapter 4, is described. Secondly, the testing and evaluating of the workflow for geometrically valid 3D intersection operations in GIS, while maintaining semantics and attributes, is dealt with.

5.1 Implementing Conversion Workflows

5.1.1 Voxel conversion

The first conversion workflow which covers the conversion steps from voxel data to 3D vector data comprised of four main steps. The following software is used for this conversion: Python 2.7.8 and MS Excel and Access 2010. Each distinct step number in the conversion, will be compared with its actual implementation:

1. In order to use the GeoTOP voxels, the 'raw' data in .csv needed to be retrieved. Zipping the original file ,svp of TNO, leads to the access of an entire GeoTOP map sheet document, where among .asc, .blg data the voxels data is present in .csv. One map sheet contains millions of voxel, therefore preprocessing steps (that subdivide the overall dataset) in a database software is required. Also, the .csv voxel data needs to be cleaned and the appropriate attributes need to be selected before exporting the data as plain .txt. The header of the .csv data – including the sorting of the data – can be incorporated the Python conversion script, and the voxel data as .txt can be the main input for the conversion.

To conclude: this first step of the voxel conversion can be executed as expected.

2. Each voxel can consist out of eight bordering vertices which define the cubic shape, or out of a single centroid. While eight vertices per voxel define the later B-REP shape accurately, this method would result in an eight times larger dataset, which is not convenient for further processing. By default, the header of the .csv voxel data interprets the coordinates of the first voxel in the dataset via its centroid. Building up a vectorized voxel via its centroid, is thus considered the most useful option.

To conclude: each distinct GeoTOP voxel can best be represented as its centroid for later processing.

3. The conversion of the voxel data towards 3D vector data can be accomplished via a Python Script that adopts the voxel data in .txt and converts in Point geometry with x, y, z coordinates, with the selected attribute and with the correct sorting indication of the dataset. The script (Appendices: A) which is used for this step, is based on the script ESRI built during their 3D Pilot project, although it consist of a different loop structure. The script (Voxel_Point.py) outputs a .shp file containing voxel data represented as 3D Point geometry. Due to the size of the voxel data set, the script is run ten times in order to cover one GeoTOP map sheet.

To conclude: a Python script can convert 'raw' voxel data in .txt with attributes, to 3D Point geometry in a GIS readable file format – .shp.

4. The output of the conversion can be loaded in GIS – here in ESRI's ArcScene – where the boundary representation (B-REP) of each 'voxel' should be built. In ArcGIS geometry can be symbolized as different 3D shapes. First, the Point geometry can be symbolized as being cubes with a dimension of 100*100*0.5 – the GeoTOP voxel dimensions. Then, these 3D symbols can be converted to 3D surface based of ESRI, the Multipart. This conversion can be done via a tool in 3D Analyst.

To conclude: a B-REP (surface based) representation of voxels can be built in GIS. As ArcGIS is used for the Application, a twofold procedure needed to be adopted: first symbolize the point geometry in 3D with the correct measurements, then convert to Multipart geometry.

Practical remarks:

The 'raw' voxel data requires preprocessing in a database in order to make the first conversion step – from voxels to points. This data preparation is done in MS Access and MS Excel. The complete voxel

dataset is opened as a database in MS Access, for the demo it concerned map sheet 44w. This dataset comprises of approximately 9.25 million rows (including the 14 header rows). Per voxel, the attribute that indicates the litho class (soil type) is necessary for the spatial analysis, and selecting only this voxel attribute can be done in MS Excel. MS Excel is however limited to load 1.048.576 rows. Therefore, in MS Access it is necessary to divide the 9.25 million voxel dataset into 10 smaller datasets of approx. 1 million rows. Also, 10 smaller datasets make the actual conversion of voxel to points go smoother. Important to note is that the distinct order (+Z +Y +X sorting) of the overall voxel dataset is very important to preserve. Via the Simple Query Wizard in MS Access, a new query field is generated, and the Top 1.000.000 Values are queried. The result of this query is saved in the Query Tab and exported as a new Excel readable format. This query procedure is repeated 10 times. Each of the 10 subsets of the voxel data, is loaded in MS Excel, Here, the data is 'cleaned' and a small script/ formula is adopted to retrieve per voxel only the second attribute – litho class.

Result:

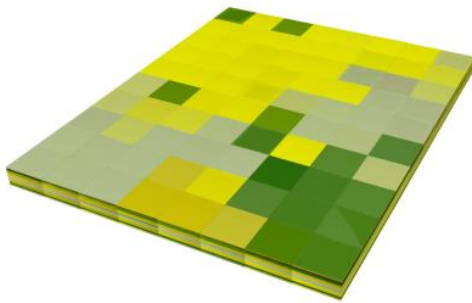


Figure 33 3D vectorized GeoTOP

5.1.2 2D surface conversion

The second conversion workflow converts 2D surface data to 3D surface vector data. For each step the implementation is described.

1. The topographic data can be retrieved via the National Mapping Agency. This data can be loaded in either a CAD or GIS program. In this research, the data was delivered as the GIS format .shp. A conversion from e.g. a CAD format to GIS format was thus not necessary.
To conclude: as the TOP10NL data was delivered in a GIS readable format, no additional exporting of the data needed to be performed.
2. In the test environment ArcGIS, the topographic can be clipped, to retrieve only the relevant 2D surfaces belonging to the same object class – in this case the class terrain is used. For the 2D clipping an initial empty .shp file is required, on which the splitting polygon can be constructed. The original data is then clipped with this splitting polygon, and the output can be stored in a new .shp file, 'clip output'. This latter 'clip output', can then be used to perform the extrusion in the next step.
To conclude: 2D clipping operations can be used to only retrieve the relevant 2D surface data.
3. In the z-coordinate aware ArcScene program, the 2D 'clip output' .shp file of step two can be extruded. In order to do so, the 2D surface can be 'symbolized' as being extruded. In this merely visual symbolizing step the vertical dimension can be determined, which can either be negative or positive. When the correct height value (z-coordinate) is determined, the extrusion symbolization can be converted to 3D Multipatch geometry, which can be stored as new data in a .shp file.
To conclude: the additional step of symbolizing the extrusion, before turning it into actual data was not considered in the conceptual workflow, but provides a useful step in determining the needed extrusion height.

Result:

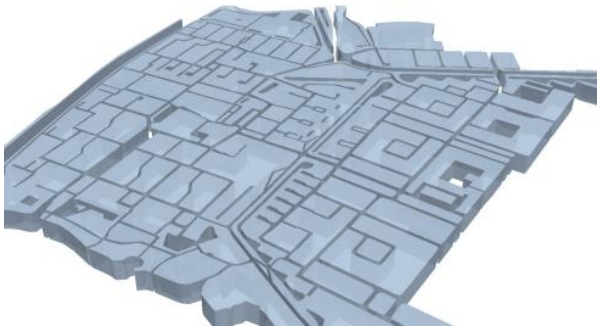


Figure 34 3D vector of TOP10NL

5.1.3 2D line and point conversion

The third conversion workflow converts 2D line and point data to 3D surface vector data. For each step the implementation is described.

1. Also the cable and line data can be obtained via the National Mapping Agency. The data can be delivered in CAD, which is the case in this research.
To conclude: the KLIC data was delivered as CAD data, no extra exporting of the data needed to be performed.
2. The CAD data is loaded into GIS, here ArcGIS. In GIS, the data can be projected on an applicable Coordinate Reference System (CRS) and then georeferenced. Whenever the attributes of geometries provide coordinate information, this information can then be retrieved instantly. Z coordinate information is given as attribute of in accompanied text. Only coordinates of points are registered, thus also only the point of which a line is built of. When coordinate attribute information is not present, coordinates of points can be retrieved by selecting the points – the GIS is capable of displaying the coordinates of the selected features.
To conclude: CAD can be loaded in GIS to project it on a CRS. Coordinates can be retrieved by selecting points in GIS, via attribute information, or plain text.
3. The coordinates of the retrieved points can be put into a Python Script that outputs the points accurately in 3D, with x, y, z coordinates. The script delivers a .shp file as output.
To conclude: a Python Script can deliver a .shp file with accurate x, y, z coordinates.
4. Regarding the initial line data, 3D lines can be constructed between the points in ArcScene, The lines may consist out multiple line strings and can be stored as new data (in a .shp file).
To conclude: through 3D points, lines can be built and stored as new data.
5. The 3D line vector data can be buffered via the *Buffer 3D* operation, part of the 3D analyst extension. Before performing the actual operation, the buffer distance can be specified, as the amount of surfaces the buffer will be constructed of.
To conclude: via three- dimensional buffer operations, surfaces can be created from 3D line data.
6. The buffer operation in ArcGIS results in a cylinder built out of triangle faces. When the buffer is performed around kinked lines, the cylinder might overlap each other at some parts, and leaving gaps at other parts. (Also a 3D intersection operation with buffered geometry may result in non-closed cylinder geometry.) When the cylinder geometry is not closed, it can be exported to the COLLADA format. This format can be loaded in a 3D modelling program such as SketchUp, were the geometry can be 'manually' closed, i.e. find and close any gaps or remove intersection surfaces. Once the geometry is closed, the new geometry can be imported in GIS again via the COLLADA format.
To conclude: 3D modelling software may be required to overcome the issues of having non-closed geometry in GIS.

Additionally, as points in KLIC data can represent complex objects (e.g. sewer connection parts), this geometry can completely be built in 3D modelling software, where the point remains the centroid of the object. A COLLADA export/import can provide the exchange between modelling software and GIS.

Result:

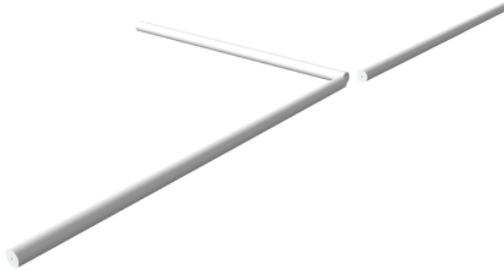


Figure 35 3D vector of KLIC

5.2 Implementing 3D Intersection Workflow

5.2.1 3D Intersection operation

The workflow that covers semantic and attribute preserving 3D intersection operations, is tested via drafted Test Cases that test and evaluate the workflow on three evaluation criteria: (i) geometry validity, (ii) correct assignment of attributes and (iii) the possibility to determine volume and surface area of the output. First, a more general description is given that sets the workflow against the implementation of the test environment ArcGIS.

As the first and second step of the 3D intersection workflow describe, all of the input features of the intersection are represented by their boundary. In ArcGIS this can be achieved via the 3D Multipatch geometry. The input geometries need to have an overlapping part – in general a volume – with each other, this overlapping part will define the 3D intersection output.

The *Intersect 3D* operation first finds which features intersect with each other after the intersecting output is constructed. The operation then places the attributes of both features in the generated output, which entails steps three up until five of the proposed workflow. However, semantics are accounted for in a later stage in ArcGIS, when the output is placed in a new layer, and can be saved as a file containing solely the output. The layer title serves for the semantic of the intersection output.

The *Is Closed 3D* operator and the *Add Z Information* tool, can provide respectively a geometric validation (based on closure of the interior volume of the geometry) and volume and surface calculations on the output, as described in steps six and seven. Notable is that the output of both *Is Closed 3D* and *Add Z Information* is placed as information in the attribute table of the intersecting output.

To conclude: overall the conceptual 3D intersection workflow can be performed in ArcGIS with the *Intersect 3D* operator. The ordering of the steps is in line with the operation in ArcGIS, deviating on the incorporating the semantics is not performed at the same time as the attribute assignment. And the set operations and tools are available in ArcGIS to verify geometric validity and to calculate volume and surface area.

5.3 Testing and Evaluating the 3D Intersection Operations

In this section, the cases as stated in Table 2 will be used to implement and test the extent to which the proposed 3D intersection workflow of Chapter 4 can be realized.

The Test Cases are set up in such a way that they are relevant to test the theories of for instance geometry validity, as stated in (2.2.3). Furthermore, the cases relate to issues that may occur in certain conditions in the Application described in Chapter 6.

Each of the case is evaluated on (i) geometry validity, (ii) correct assignment of attributes and (iii) volume/surface area calculation. The rules for valid geometry output are based upon the polyhedron validity rules in as described in section (2.2.3, also section 2.3.3), which can be evaluated in the used test environment ArcGIS via the verification whether a Multipatch is closed ('simple'), or not closed ('not simple').

The 3D Boolean set operation is the solid operator *Intersect 3D* of the *3D Features Toolset* in the extension 3D Analyst. *Intersect 3D* only adopts closed (valid) Multipatch geometry as input. In order to validate the output of *Intersect 3D*, the *Is Closed 3D* tool is used that generates a new field in the attribute table of the intersection output filled with either Yes or No values, indicating closed or open Multipatch geometry respectively. The *Add Z Information* tool is used to add volume and surface area in the attribute table of the intersected output. The main focus of the Test Cases lies on the 3D intersection operation. Additionally, the other solid operations *Difference3D* and *Union 3D* are also conducted in two separate case groups. Also the output results after the difference and union operations will be evaluated on (i) geometry validity, (ii) correct assignment of attributes and (iii) volume/surface area calculation.

	Geometry	Attributes	Volume/surface area
<i>3D intersection operations</i>			
1. Simple Multipatch objects			
1.1 Vertical	Validity	Correctness	Possibility
1.2 Oblique	Validity	Correctness	Possibility
<i>3D intersection operations</i>			
2. Complex objects			
2.1 With many surfaces	Validity	Correctness	Possibility
2.2 Cylinder long	Validity	Correctness	Possibility
2.3 Cylinder curved patch	Validity	Correctness	Possibility
2.4 Cylinder low/high detailed	Validity	Correctness	Possibility
<i>3D intersection operations</i>			
3. Overlapping surface/edge/point			
3.1 Via a Multipatch with Multipatch	Validity	Correctness	Possibility
3.2 Via a Multipatch with a surface	Validity	Correctness	Possibility
3.3 Via a Multipatch with an edge	Validity	Correctness	Possibility
3.4 Via a Multipatch with a point	Validity	Correctness	Possibility
Experiments:			
<i>3D difference operation</i>			
4. Complex objects			
4.1 Subtracted cylinder object	Validity	Correctness	Possibility
<i>3D union operations</i>			
5. Construct or subtract objects			
5.1 Non-manifold point	Validity	Correctness	Possibility
5.2 Non-manifold edge	Validity	Correctness	Possibility
5.3 Non-manifold surface	Validity	Correctness	Possibility

Table 2 Test Cases for testing


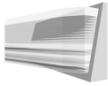
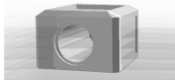


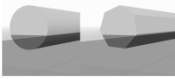



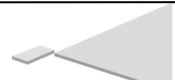




	Diagram
<i>3D intersection operations</i>	
1. Simple objects	
1.1 Vertical	
1.2 Oblique	
<i>3D intersection operations</i>	
2. Complex objects	
2.1 With many surfaces	
2.2 Cylinder long	
2.3 Cylinder curved path	
2.4 Cylinder low/high detailed	
<i>3D intersection operations</i>	
3. Overlapping surface/edge/point	
3.1 Via a Multipatch with Multipatch	
3.2 Via a Multipatch with a surface	
3.3 Via a Multipatch with an edge	
3.4 Via a Multipatch with a point	
Experiments:	
<i>3D difference operation</i>	
4. Complex objects	
4.1 Subtracted cylinder object	
<i>3D union operations</i>	
5. Construct or subtract objects	
5.1 Non-manifold point	
5.2 Non-manifold edge	
5.3 Non-manifold surface	

Figure 36 Test Cases in diagram

Remarks regarding the Test Cases in Figure 36:

- Case 1.2 'Vertical' can be regarded as the 'main 3D GeoTOP intersection': the intersection required for the Application, i.e. a 3D intersection operation on vectorized GeoTOP data with extruded TOP10NL data (on an open soil location). In diagram:

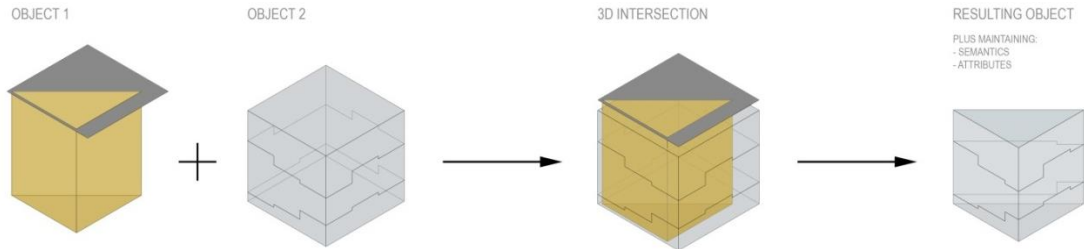


Figure 37 Main 3D GeoTOP intersection

- KLIC data is used for these cases: of group 2 and 4.1.

- Volume and area calculations are verified with simple calculations on both original input geometries, e.g. to verify a volume the area of a surface of a TOP10NL can be multiplied by the z- value of the GeoTOP data.

- The Test Cases numbers in group 1 to 3 solely comprise 3D intersection operations and are drafted to reflect on 'real- life' practical situations (e.g. modelling the sewer system) that relate to the Application in chapter 6.

- The cases drafted in numbers 4 and 5 are more experimental, and are proposed to explicitly test the CGAL Nef_Polyhedra_3 class used by 3D Analyst of ArcGIS. As the Nef_Polyhedra_3 is closed under Boolean operations and can be non-manifold, three-dimensional Union operations are performed via *Union 3D* of ArcGIS on non-manifold geometry cases.

- Symbols in ArcGIS:

In ArcGIS, symbols can be used to graphically display a certain feature into a symbol of a different shape. When a certain feature is symbolized into a different shape, the geometry of this shape is not stored as a feature. Thus a symbol in ArcGIS is solely a graphic presentation.

- Features in ArcGIS:

Features in ArcGIS can be regarded as spatial objects, which are stored in feature classes in a geodatabase. In a feature classes, features have a common spatial representation (e.g. point, line, polygon) and an attribute with coordinates.

- A feature class comprise of the geometry and attributes of each distinct feature.

- Layer stored in .gdb: a layer in ArcGIS can be packaged, that includes the layout of the layer and the data itself. Also a layer can be exported to a .shp file, which is in fact a vector data storage format (ESRI ArcGIS, 2015).

Test Cases

1. Simple objects

1.1 Vertical

Main input data: the vectorized GeoTOP with the 3D surface based vector data of TOP10NL, extruded. Figure 38 shows a screen of the two datasets. One distinct extruded polygon (a small park) will be intersected with GeoTOP.

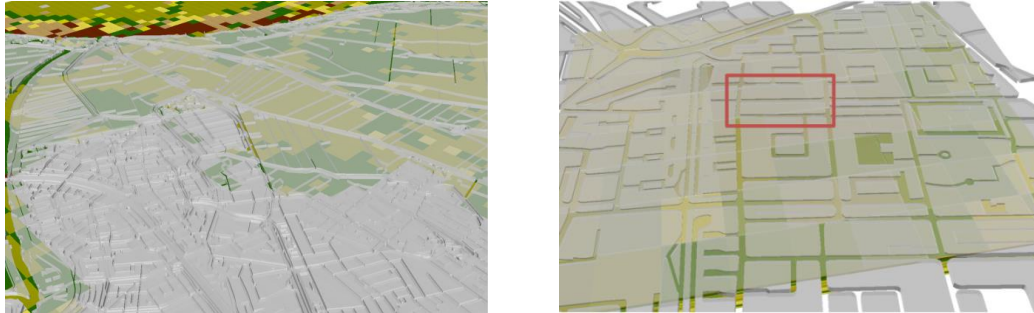


Figure 38 Vectorized GeoTOP and TOP10NL

The intersection is of closed, valid geometry. The output is automatically placed in a new layer named as 'Intersection_Output', which covers the semantics.

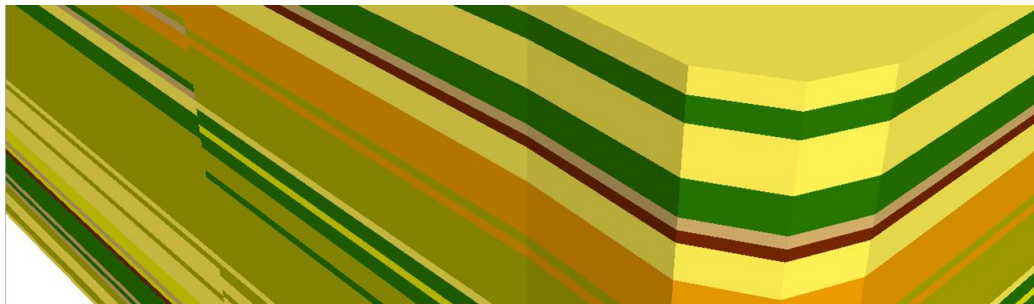


Figure 39 Valid, closed Multipatch output

The original GeoTOP attributes are assigned to the various Multipatch feature that together comprise the output of *Intersect 3D*.

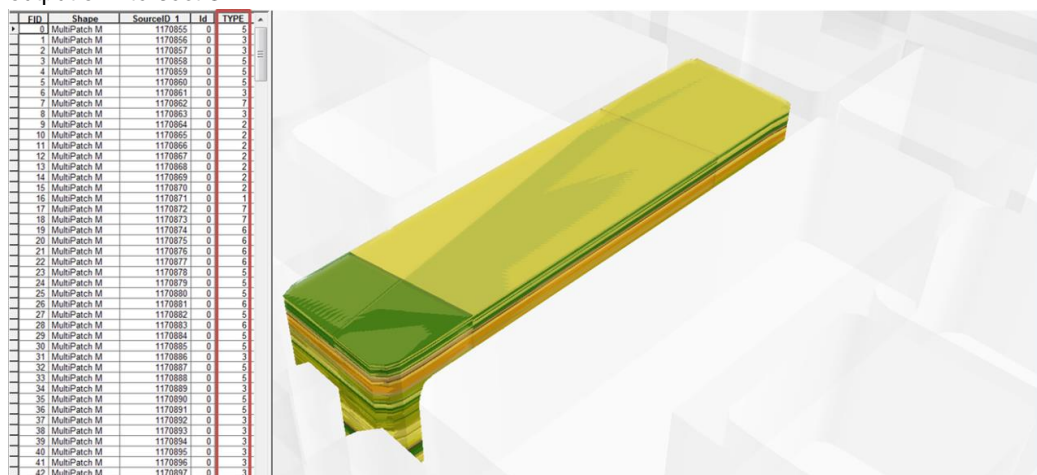


Figure 40 Correct attribute assignment

Every feature is of closed geometry, thus volume and also surface area calculations can be made via the *add Z information* tool.

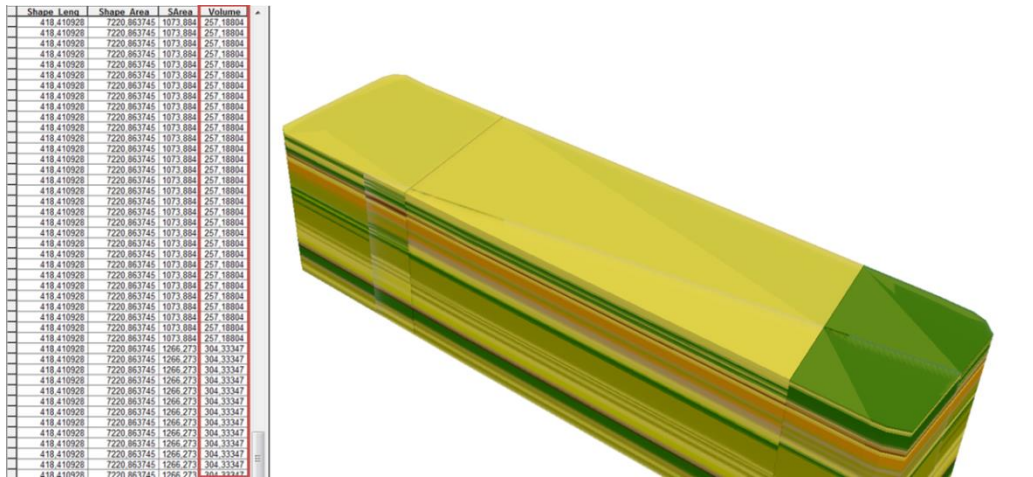


Figure 41 Correct volume calculations

The volumes can be summarized on the TYPE attribute (entailing the soil type), and this summary can be stored in a small .dbf file:

TYPE	Cnt_TYPE	Sum_Volume
1	13	7228,79905738000
2	97	57049,91915380000
3	154	97284,77644620000
5	139	81083,71319560000
6	26	14434,27227720000
7	39	22645,12023110000
8	36	23549,67691410000

Figure 42 Possible summarize on TYPE attribute for volumes

case number	Geometry	Attributes	Volume/surface area
1.1 (vertical)	Valid, all features that together form the intersection output are closed geometry	Correct assignment, the TYPE attribute belonging to GeoTOP is assigned to correct features. Colors can be linked to the value in TYPE, value '5' is sand and colored as yellow	Possible. Both volume and surface area calculations can be made, which are listed per distinct feature in the attribute table. Furthermore, a summarize can be made TYPE attribute value.

1.2 Oblique

The foundations that are used for the second Simple Multipatch object case, are based on an indication of the surrounding built environment. The input situation:

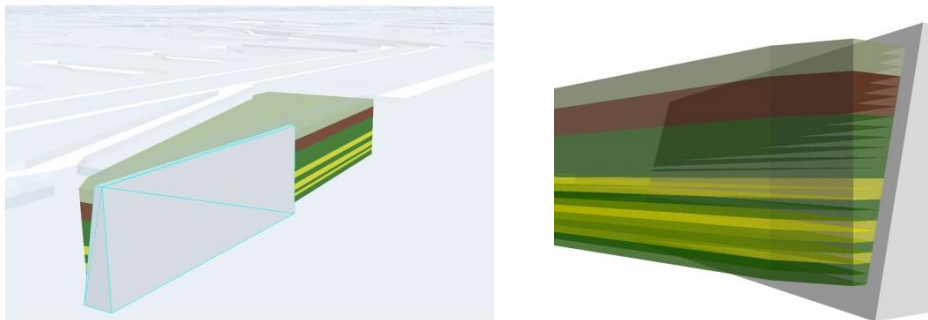


Figure 43 Oblique input

The 3D intersection operation with an oblique object as input, resulted in a valid geometry output. Attribute are assigned correctly, and calculations on interior volume can be performed.

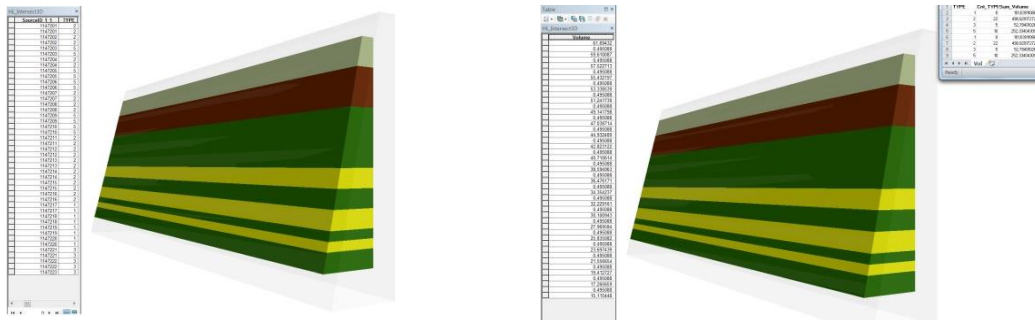


Figure 44 Correct attribute assignment and volume determination

case number	Geometry	Attributes	Volume/surface area
1.2 (oblique)	Valid, all features are closed	Correct assignment.	Possible.

2. Complex objects

2.1 With many surface/ holes

This following ‘Complex objects’ cases are based on KLIC data. The sewerage network is located in the upper part of the subsurface. The connection parts of the sewer pipes are rich of detail, however only represented as points in original KLIC data. For the scope of this case only one sewerage connector is built and used as input for a 3D intersection operation. An unconnected sewer connection part consists out of multiple holes.

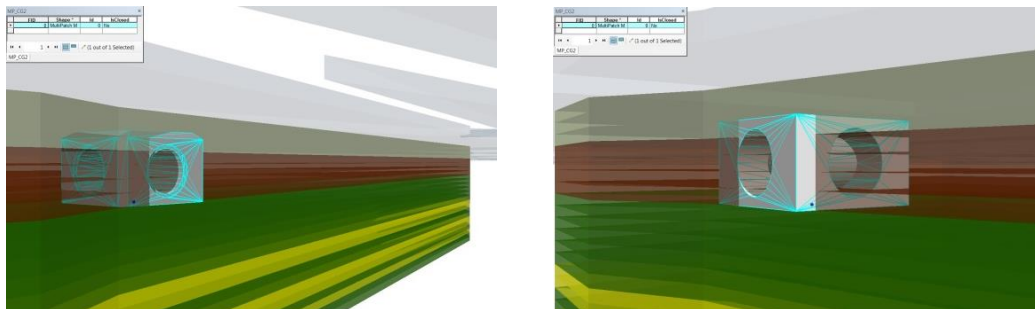


Figure 45 Complex geometry as input

Before a 3D intersection operation is performed, a verification on whether the geometry is closed or not, is performed via an *Is Closed 3D* operation. As seen in the attribute tables of both geometries of Figure 45, the geometry is not closed. Various grounds can cause this invalid geometry; the geometry does not meet the requirements of a closed Multipatch, or the edited COLLADA file contains irregularities. In the latter cases, the holes in perpendicular directions cause patches (i.e. surfaces) to overlap with one another, resulting in unclosed Multipatch geometry. Evenso, a 3D intersection operation can be performed, but without result as stated in the following warning/ error message:

Description: Nonclosed features are skipped and not used by set operator tools. The message shows the input feature class name and the OID of the input feature that is not closed.

Solution: None. Features must already be closed.(ESRI, 2015)

More simplistic geometry – without crossing holes – counts as closed, thus a 3D intersection accompanied with the correct attributes, can be performed with this geometry as input. Figure 46:

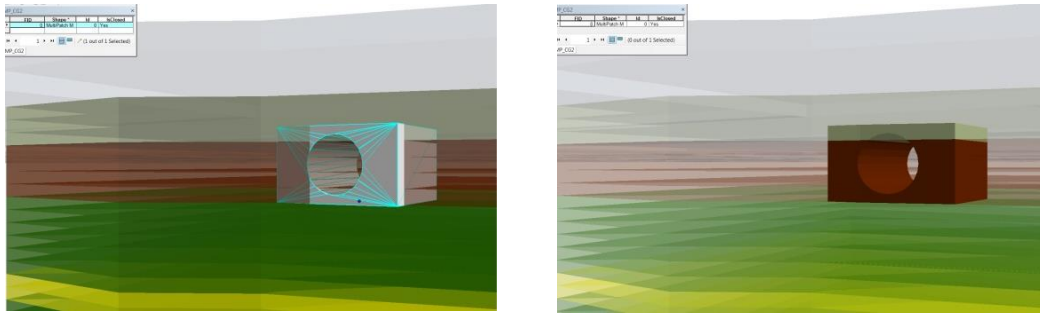


Figure 46 Valid complex geometry

Volume and surface area calculations can only be performed when the all geometries are fully closed. If the object is closed, but very complex build, determining volume gives a zero result, see Figure 47.

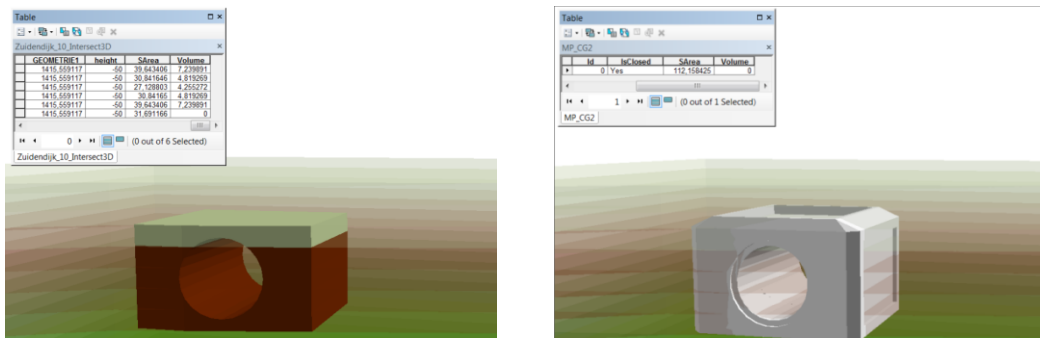


Figure 47 Volume determined as zero

case number	Geometry	Attributes	Volume/surface area
2.1	Complex objects with a lot of surfaces are prone to be invalid. COLLADA or a simplification of the modelled object can provide the solution.	Correct assignment. (With closed geometry)	Not possible with non-closed geometry, Or with too complex geometry.

2.2 Cylinder long

The sewerage may comprise of very long pipes. In this case such a long geometry is drafted, as present in multiple 'open soil' locations.

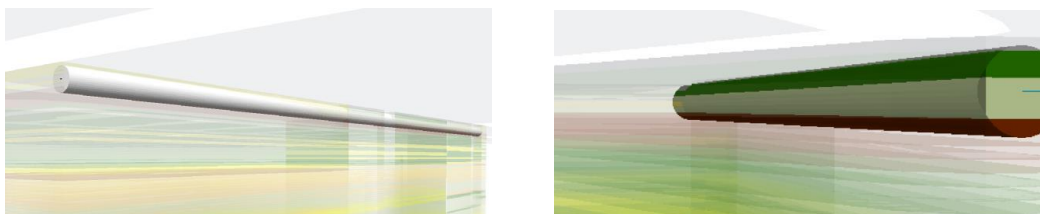


Figure 48 Long cylinder geometry, and part of it

case number	Geometry	Attributes	Volume/surface area
2.2	Valid, all features are closed.	Correct assignment.	Possible.

2.3 Cylinder curved path

The sewerage network is an apparent infrastructure in the upper part of the subsurface. Reconstructing the sewerage pipes (cylinders) in GIS can result in more accurate determination of geohydrological

characteristics. Furthermore, building the sewerage network in 3D can help in more comprehensive hydrologic sewer calculations.

This experiment case with curved geometry is located in the city district Dubbeldam of Dordrecht, an area selected by the Stadslab program of the municipality of Dordrecht for its vulnerability to flooding. The investigated open soil location is part of a water rich green area and is stated as 'Platanenlaan Park'. Part of the Dubbeldam district is built in GIS, and contains the vectorized Geotop dataset and a clipped and extruded TOP10NL, Figure 49:

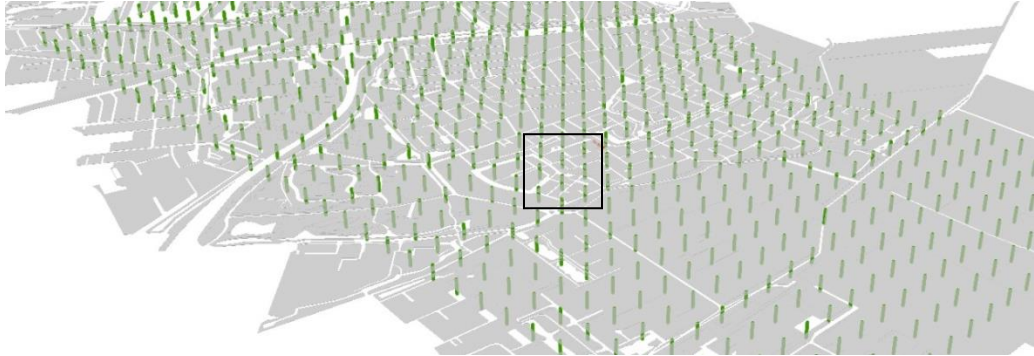


Figure 49 Input datasets

Modified 2D KLIC data is loaded into GIS to be able to retrieve the exact position of the sewer network with its pipes at the location of the 'Platanenlaan Park'. Depth values of the KLIC dataset are stated as text in the CAD data. The x, y, z coordinates that define the center line of a sewer pipe, are loaded into GIS via Python, see Figure 50:

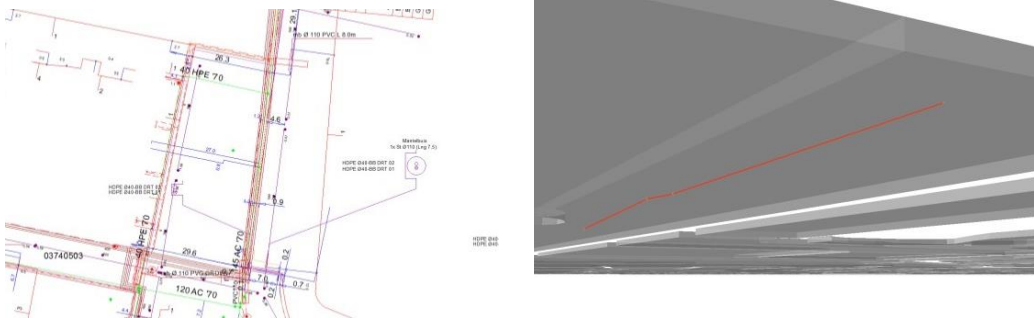


Figure 50 KLIC data, and as 3D line

Via buffer operations in 3D, the cylinder geometry of the sewer pipes can be generated. Since the concerning sewer pipe that runs through the 'Platanenlaan Park' has two nodes, buffering the sewer center line piece by piece results in three cylinders, overlapping each other see Figure 51 (left). Buffering the same sewer center line all together results in a one cylinder without any overlaps, see Figure 51 (right). On both cylinder geometries a 3D intersection operation with the vectorized GeoTOP dataset will be performed.

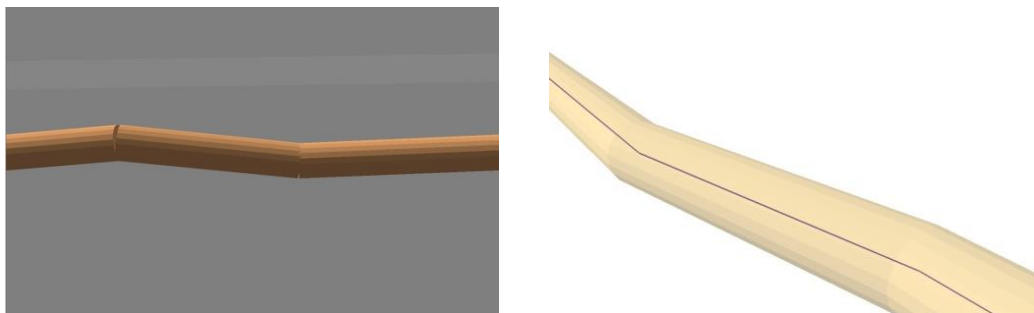


Figure 51 Three buffered cylinders around a center line – gaps between the closed cylinders (left image), one buffered cylinder, no gaps in the cylinder (right image)

3D Intersections on both cylinder geometries result in valid geometry output with correct assignment of attributes. Thus, multiple geometries may overlap one another as long as each distinct geometry is closed. Figure 52:

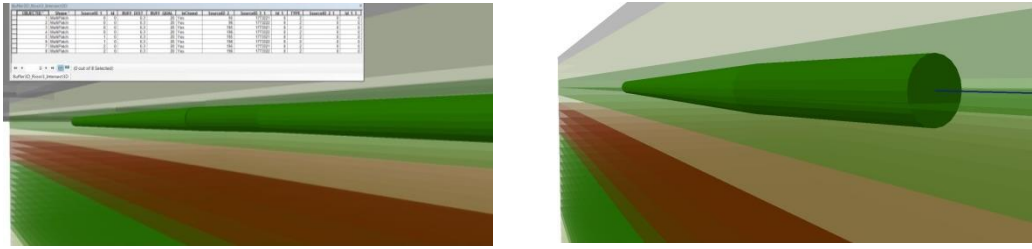


Figure 52 Correct output of three overlapping closed cylinders (left image), correct output of an exactly fitting, kinked cylinder (right image)

However, parts (which are separate features that constitute the overall geometry) of even the fitting, kinked cylinder is non-closed, as the *Is Closed* 3D operation explicated in the attribute table. Thus, volume of this feature can also not be determined, see Figure 53.

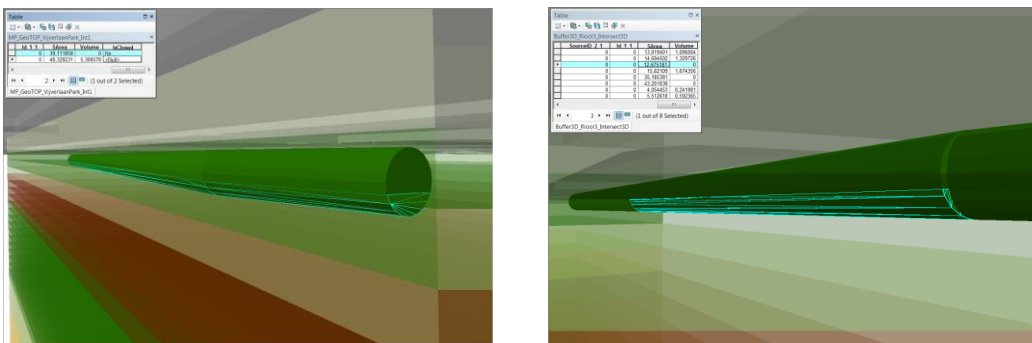


Figure 53 Volume zero for non- closed features

case number	Geometry	Attributes	Volume/surface area
2.3	Partly invalid, some features belonging to the output are non- closed.	Correct assignment.	Possible, only for the closed features.

2.4 Cylinder low/high detailed

The geometry of a cylinder can be represented by a lot of vertices (Figure 54 left), or few vertices (Figure 54 right). A more detailed and more accurate representation requires more vertices. In the following cylinder experiment case, interior volumes of a high detailed and a low detailed cylinder are calculated, in order to verify whether level of detail not only relevant in offering a more real- life representation, but also relevant in performing spatial analysis. Both cylinders represent sewer pipes which are buffered in 3D over the same center line.

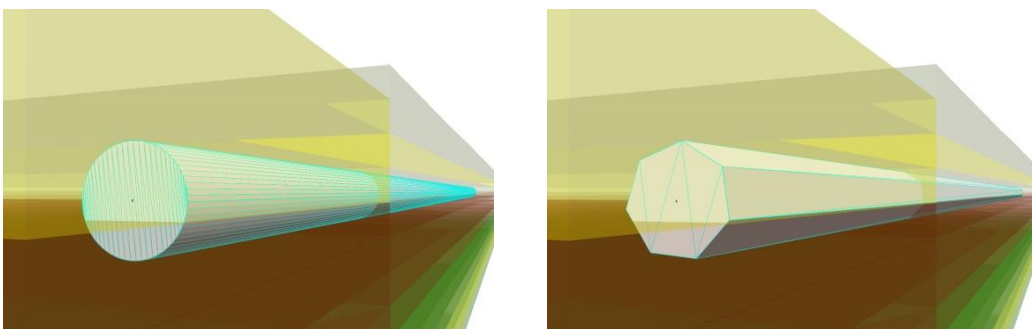


Figure 54 Cylinder with many, and cylinder with few surfaces

First a 3D intersection operation is performed with both of the cylinder shapes with the vectorized GeoTOP dataset. This operation outputs valid geometry and correct assignment of attributes. After the 3D intersection operation, interior volume is determined. The difference in volume between the two cylinders is apparent, as the cylinder which is built out of more vertices comprises of more interior volume (17.9 m³) than the cylinder with less vertices (15.7 m³), see Figure 55:

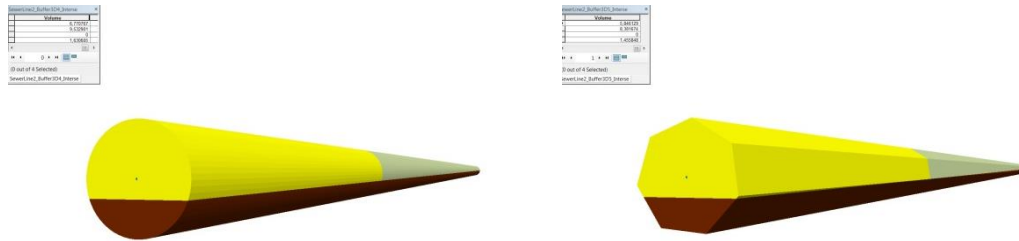


Figure 55 Different volume output

case number	Geometry	Attributes	Volume/surface area
2.4	Valid.	Correct assignment	Possible, with significant difference between the interior volume of the high detailed cylinder and the low detailed cylinder.

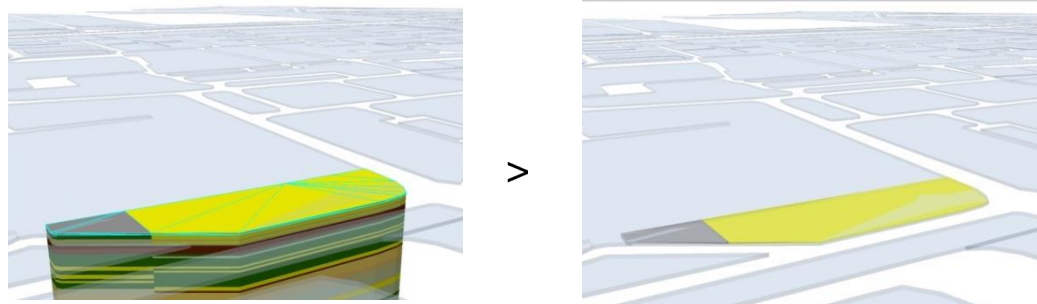
3. Overlapping surface/edge/point

3.1 Via a Multipatch with Multipatch

The vectorized GeoTOP dataset comprises of rectangular Multipatches which are formed around each distinct centroid of a voxel. Thus, per Multipatch multiple overlapping (bordering) faces occur, e.g. between neighboring Multipatches MP1 and MP2.

In 3D intersection operations, the intersecting input geometry could exactly touch with an overlapping face of the target geometry. This situation is expected to lead to difficulties in generating output geometry accompanied with the correct attributes, as each or either of the neighbors – Multipatch N1 or N2 – could be assigned to the intersecting result with their distinct attributes.

As an experiment case, an open soil location is chosen near the city district Wielwijk – a focus area of the municipality of Dordrecht for urban renewal. This unsaturated zone in the subsurface of this green terrain ‘Zuidendijk Park’ is bordered with a paved area. In this case, only the part of the open soil that borders the paved area needs to be investigated. The ‘Zuidendijk Park’ as selected in GIS with the TOP10NL and GeoTOP datasets and the resulting target geometry, can be seen in the following figures:



Figures 56 Input datasets

In order to configure an input geometry that exactly touches overlapping (bordering) faces, the vertices of the overlapping faces are used. The four vertices are snapped, and the input geometry (in orange) is

built that should capture only the soil type attribute 5 [sand - yellow], and not 0 [unknown, paved - grey] as can be seen in figure 57:

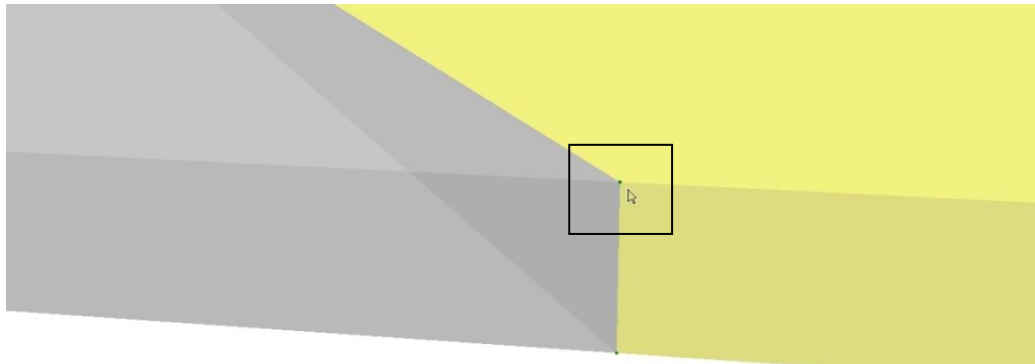


Figure 57 Editing via vertex

The input Multipatch geometry (orange) for the 3D intersection operation is positioned on the coordinates of the overlapping face. A visual verification of overlapping faces in the rendering of the surfaces inside GIS, where an equal color pattern cannot be visualized for overlapping faces (Figure 58).

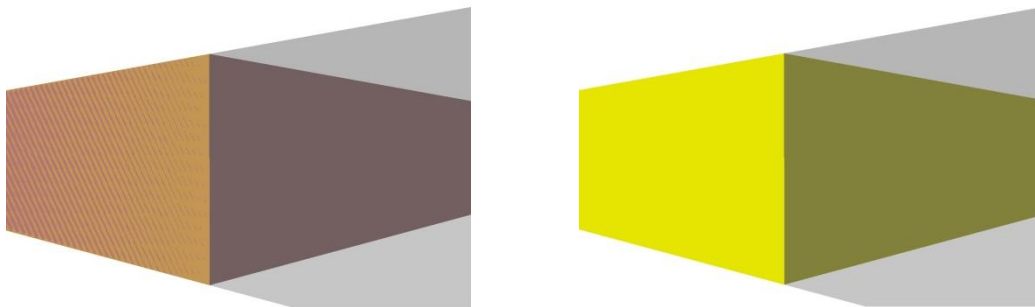


Figure 58 Overlapping surfaces and equal color pattern surface

The performed 3D intersection operation, displays the following warning/ error message

'Description: Intersect failed because the result is not simple, meaning the two features share only a single vertex or edge. This message shows the two input feature class names and the OIDs of the two input features.

Solution: None. Those two features cannot intersect.' (ESRI, 2015)

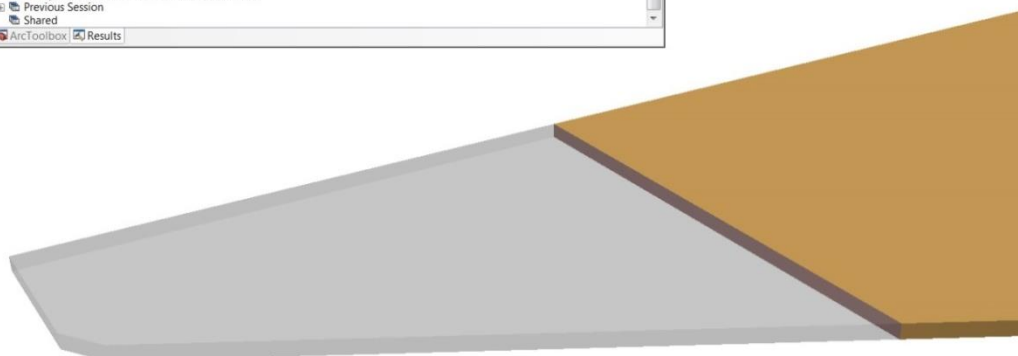
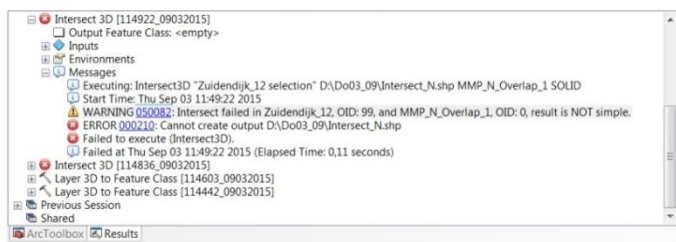


Figure 59 Output error message

Based on the displayed error message, a geometry output with correct assignment of attributes is not expected. However, a correct 3D intersection output can be created when taking the (smaller) intersecting geometry as *first* input and the (larger) target geometry as *second* input. This useful result is seen in Figure 60:

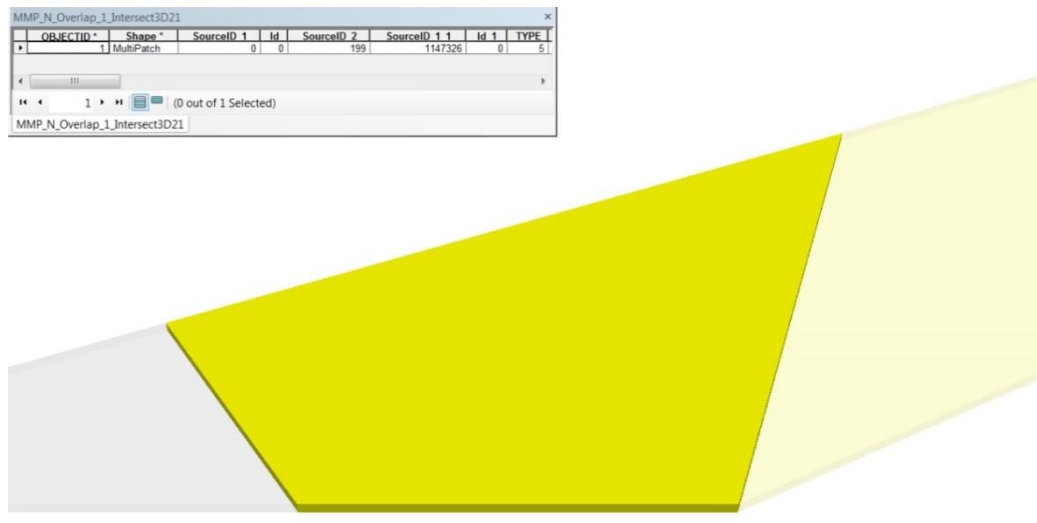


Figure 60 Geometry output

The most effective way to overcome any possible output errors, is to set the intersection input x distance over or x distance before the overlapping face. These scenarios are portrayed in Figures 61 and 62.

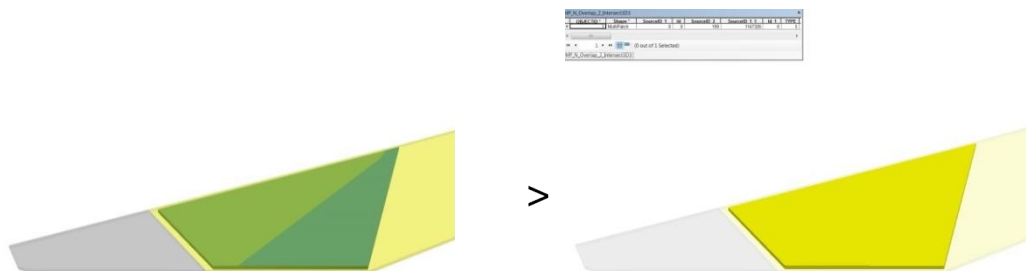


Figure 61 Geometry set back (before boundaries)

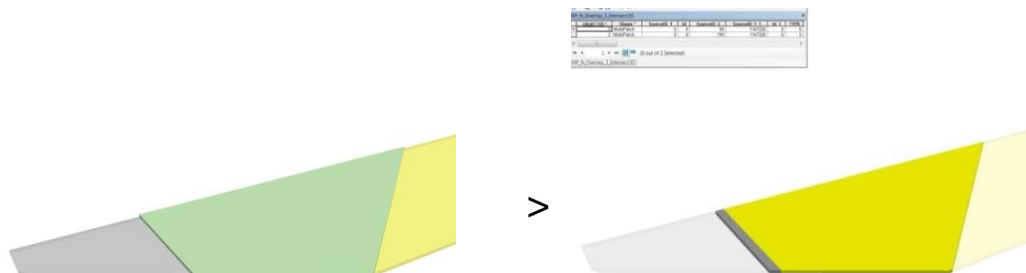


Figure 62 Geometry set over boundaries

case number	Geometry	Attributes	Volume/surface
3.1	Valid output, but with a warning: the output result is 'not simple'.	Correct assignment, even the output result is 'not simple'.	Possible.

3.2 Via a Multipatch with a surface

This drafted Test Case is focused on two GeoTOP Multipatches that are located next to each other and thus only share a surface. The *intersect 3D* set operation did not generate an intersecting surface.

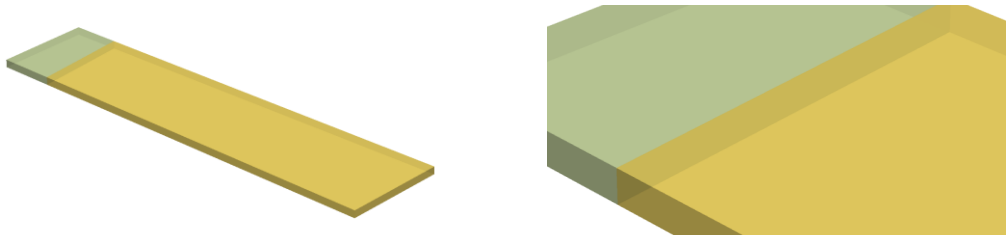


Figure 63 Sharing edge

case number	Geometry	Attributes	Volume/surface area
3.2	Not possible to compute. A warning indicating the result is not simple is given.	NP	NP

3.3 Via a Multipatch with an edge

The operator *Intersect 3D Line With Multipatch* can compute the intersection between an edge and a 3D Multipatch, as a way to penetrate and report the 3D features which are cut by the line. The output of this operation are a line and points which are divided and located at the boundaries of the 3D Multipatches they intersect with.

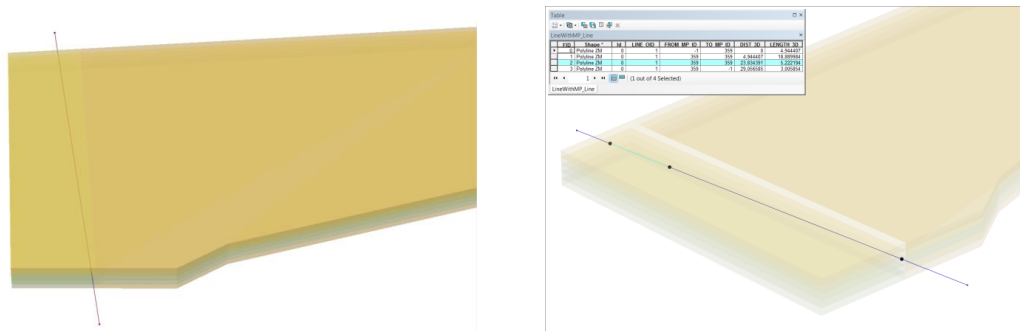


Figure 64 Multipatch with 3D line

case number	Geometry	Attributes	Volume/surface area
3.3	Valid, line geometry and additional point geometry.	Not applicable	NA

3.4 Via a Multipatch with a point

This case converges to a shared point between two Multipatches. The *intersect 3D* set operation did not generate an intersecting point.

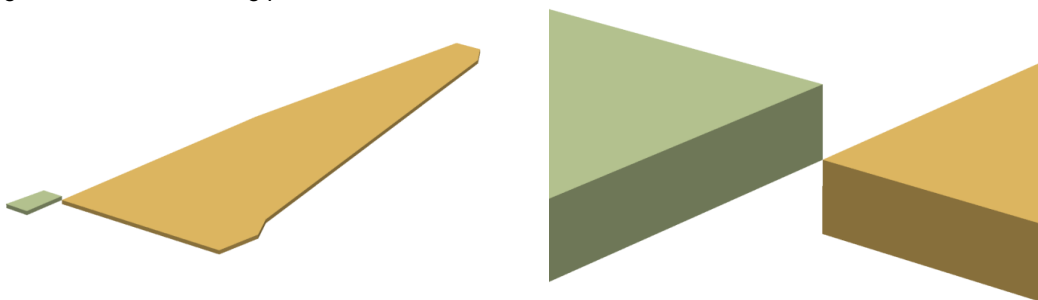


Figure 65 Sharing point

case number	Geometry	Attributes	Volume/surface area
3.4	Not possible to compute. A warning indicating the	NP	NP

	result is not simple is given.		
--	--------------------------------	--	--

Experiments:

4. Complex objects

4.1 Subtracted cylinder object

A *Difference 3D* operation is performed in order to subtract a Multipatch buffered line into two equal parts. Volume computation should be halved in this matter.

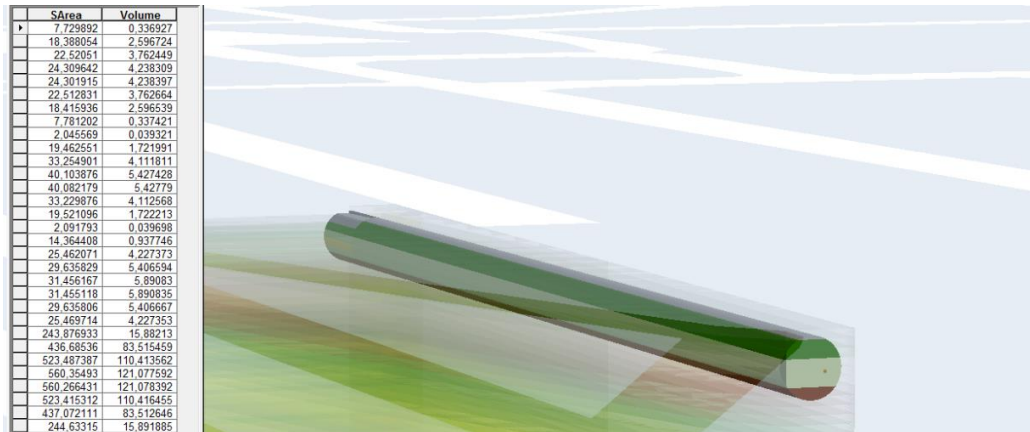


Figure 66 Input for difference operation

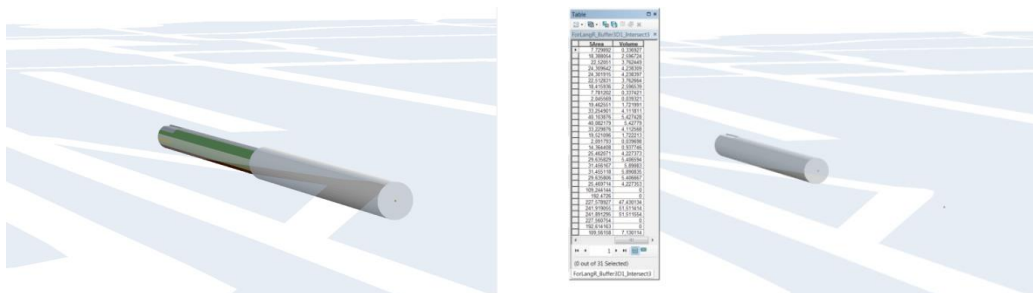


Figure 67 3D Difference operation

case number	Geometry	Attributes	Volume/surface area
4.1	Partly, a small part of the Multipatches that built the output cylinder are non-closed.	Correct assignment.	Possible, volume amount is decreased. As some features of the output cylinder do not have volume, the calculations deflect.

5. Construct objects

The following cases are proposed in order to test whether geometric output can be formed with 3D union set operations. As a Nef Polyhedra can be non-manifold, three non-manifold cases are drafted: non-manifold point, edge and surface.

**5.1 Non-manifold point
For Union 3D**

Two closed Multipatches share only a point. An *Union 3D* operation is performed.

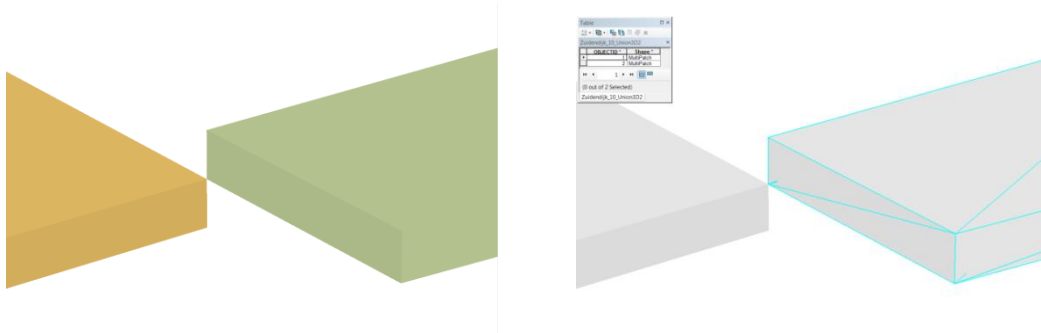


Figure 68 Touching points – for non-manifold

case number	Geometry	Attributes	Volume/surface area
5.1	Valid. Though the geometry is not unified and two features remain. A warning indicating the features share a complex relationship; only a vertex or edge.	Not assigned to the output.	Possible, for two separate features.

**5.2 Non-manifold edge
For Union 3D**

Two closed Multipatches share only an edge. An *Union 3D* operation is performed.

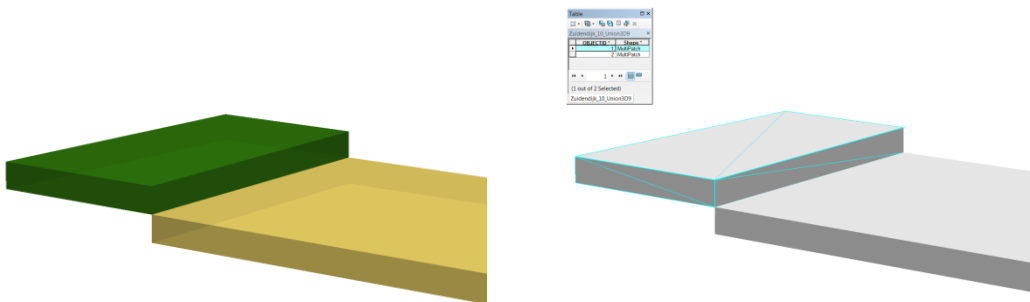


Figure 69 Touching edge- for non-manifold

case number	Geometry	Attributes	Volume/surface area
5.2	Valid. Though the geometry is not unified and two features remain. A warning indicating the features share a complex relationship; only a vertex or edge.	Not assigned to the output.	Possible, for two separate features.

**5.3 Non-manifold surface
For Union 3D**

Two closed Multipatches share only a surface. An *Union 3D* operation is performed.

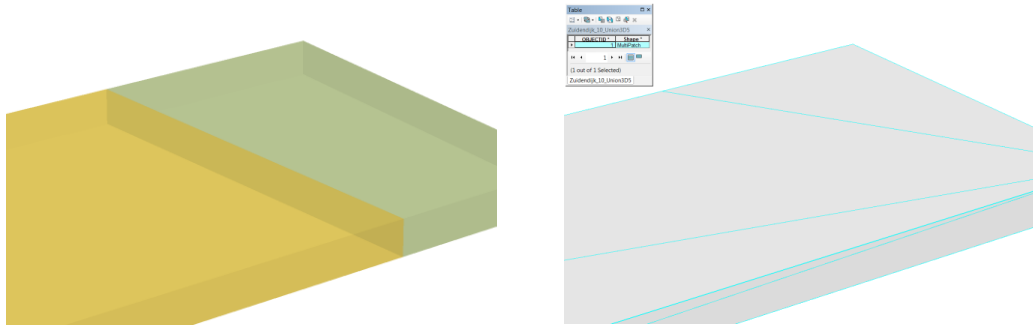


Figure 70 Touching surface – for non-manifold

case number	Geometry	Attributes	Volume/surface area
5.3	Valid. The two features are unified as one Multipatch.	Not assigned to the output.	Possible.

An overview of the evaluation tables of the drafted Test Cases:

case	Geometry	Attributes	Volume/surface area
1.1 Vertical	Valid, all features that together form the intersection output are closed geometry	Correct assignment, the TYPE attribute belonging to GeoTOP is assigned to correct features. Colors can be linked to the value in TYPE, value '5' is sand and colored as yellow	Possible. Both volume and surface area calculations can be made, which are listed per distinct feature in the attribute table. Furthermore, a summarize can be made TYPE attribute value.
1.2 Oblique	Valid, all features are closed	Correct assignment.	Possible.
2.1 With many surfaces	Complex objects with a lot of surfaces are prone to be invalid. COLLADA or a simplification of the modeled object can provide the solution.	Correct assignment. (With closed geometry)	Not possible with non-closed geometry, Or with too complex geometry.
2.2 Cylinder long	Valid, all features are closed.	Correct assignment.	Possible.
2.3 Cylinder curved path	Partly invalid, some features belonging to the output are non- closed.	Correct assignment.	Possible, only for the closed features.
2.4 Cylinder low/high detailed	Valid.	Correct assignment	Possible, with significant difference between the interior volume of the high detailed cylinder and the low detailed cylinder.
3.1 Overlap Multipatch Multipatch	Valid output, but with a warning: the output result is 'not simple'.	Correct assignment, even the output result is 'not simple'.	Possible.
3.2 Overlap Multipatch surface	Not possible to compute. A warning indicating the result is not simple is given.	NP	NP
3.3 Overlap Multipatch edge	Valid, line geometry and additional point geometry.	Not applicable	NA
3.4 Overlap Multipatch point	Not possible to compute. A warning indicating the result is not simple is given.	NP	NP
4.1 Subtracted cylinder object	Partly, a small part of the Multipatches that built the output cylinder are non-closed.	Correct assignment.	Possible, volume amount is decreased. As some features of the output cylinder do not have volume, the calculations defect.
5.1 Non-manifold point	Valid. Though the geometry is not unified and two features remain. A warning indicating the features share a complex relationship; only a vertex or edge.	Not assigned to the output.	Possible, for two separate features.

case number	Geometry	Attributes	Volume/surface area
5.2 Non-manifold edge	Valid. Though the geometry is not unified and two features remain. A warning indicating the features share a complex relationship; only a vertex or edge.	Not assigned to the output.	Possible, for two separate features.
5.3 Non-manifold surface	Valid. The two features are unified as one Multipatch.	Not assigned to the output.	Possible.

6. APPLICATION

This Chapter explicates the Application of this thesis research: geohydrology linked to the output of the 3D intersection operations on vectorised GeoTOP, TOP10NL and KLIC data in a 3D GIS model. The spatial analysis that the Application contains, unveil the subsurface geohydrologic potentials to cope with rain water discharge issues in urban areas. The geohydrological Application adopts the main research requirements – as tested via the evaluation criteria on the Test Cases in Chapter 5.

6.1 Geohydrology

Geohydrology in the subsurface is a complex field: various theories and models exist to describe the hydrology flows through the subsurface.

Through spatial analysis consisting out of geometrically valid 3D intersection operations that maintain semantics and attributes on the GeoTOP dataset, the Application aims to retrieve the following key geohydrological aspects of the subsoil:

- **hydrological conductivity/ infiltration capacity** [mm/time]
- **water storage capacity** [m³]

A selection of relevant geohydrologic terminology is dealt with in the following section, in order to provide context and basic insight.

<p>Storage – water content, the volume of water present in a specific part of the subsurface.</p> <p>Storage coefficient/ factor – the quotient of change in water storage and the change in groundwater levels.</p> <p>Capillary ascend – upwards increase of water above the groundwater level.</p> <p>Permeability – the ability of a gas or fluid (here, water) to go through subsoil.</p> <p>Permeability coefficient – the permeability expressed in a quotient as measurement of the subsoil to penetrate fluid or gas.</p> <p>Pressure altitude – pressure of the water column, equal to the relative water pressure. A negative value.</p> <p>Unsaturated zone / drainage depth – that part of the subsoil above the groundwater level. Here, the pores consist of air and water (Cultuurtechnisch vademecum, 1988).</p>
--

The Application converges on precipitation as main ‘water input’ for open soil locations. Therefore, the **storage, permeability coefficient** and the **unsaturated zone/ drainage depth** are the terms that are dealt with in this Application.

The exact zone which is relevant in this matter, is the unsaturated subsurface: the area in the subsurface between ground surface level and groundwater level, indicated by the critical z-value. Unsaturated soil comprises of solid elements, water and air, where both air and water are situated in the pores of the subsoil (Cultuurtechnisch vademecum, 1988).

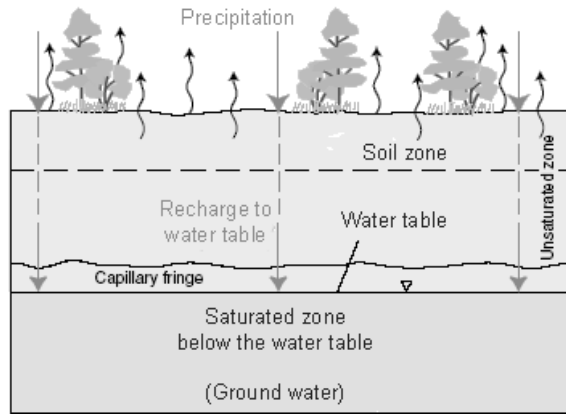


Figure 71 3D Section of the unsaturated subsurface (USGS, 2015)

6.1.1 Infiltration capacity

The infiltration capacity, is influenced by many environmental factors, such as:

- amount of rainwater – precipitation of the earth’s surface
- rainwater collection/losses
- intake by vegetation
- vaporization
- paving type
- overall intake by the subsurface

Also, the aforementioned capillary ascend and pressure altitude, but these refined hydrological factors are not taken into account in this Application. Then, the infiltration capacity can be derived/estimated via two basic geohydrologic values:

The first is the water permeability of the subsurface, the hydrological conductivity, expressed in the permeability coefficient K . The type of soil can be linked to a specific K - coefficient, from which a permeability qualification can be made. Regarding the K - coefficient: the vertical permeability is essential, K_v . The K - coefficient in the context of water (and its fluid properties) in the subsurface, depends on the following aspects:

- The particle size of the soil type
- The porosity of the soil
- The shape and size of the voids in the soil (Khan, 2005)

Secondly, In addition to the K , data on groundwater levels is needed. Groundwater flows are dynamic processes and thus change over time, aspects which are not incorporated in the static GeoTOP model. To estimate the water infiltration and storage capacity of the unsaturated subsurface, the Average Highest Groundwater level and the Average Lowest Groundwater level (abbreviated in Dutch respectively GHG and GLG) are required. Groundwater levels determine the unsaturated part of the subsurface. This unsaturated parts lies in the critical z-value: which is defined as the vertical area between ground surface level and groundwater level. Capillary water rise can occur in this subsurface zone (with an order of magnitude of 2mm a day referenced for all soil types due to surface tensions), though as stated before, this phenomenon is not taken into account in this Application.

The *permeability* K is a determined factor that can be assigned to a certain soil type, as seen in Table 3. Derived from the *permeability* K a classification can be made, that indicates whether a soil type is suited to led water through, or not – see Classification Permeability in Table 3.

<u>Litho Class / Soil Type</u>	<u>Permeability</u> [m/etmaal]	<u>Classification Permeability</u> *)
<i>klei:</i>		
sterk gescheurd (zuiderzeepolders)	10-100	zeer goed
enige poriën of scheuren	0,5 - 2	goed

zeer dicht (komklei, slechte laag)	0,005 - 0,05	slecht tot zeer slecht
zeer dicht (knipklei)	<0,005	zeer slecht
slap, ongerijpt	10^{-4} - 10^{-5}	zeer slecht
ongerijpt, samengeperst	10^{-6}	zeer slecht
lichte zavel, gerijpt	0,02 - 0,2	slecht tot matig
<i>zand:</i>		
grof, met enig grind	10-50	zeer goed
middelfijn (dekszand)	1-5	goed
uiterst fijn	0,2 - 0,5	matig
<i>veen:</i>		
ongerijpt	0,01	slecht

***) Classifications:**

<i>Permeability (k-waarde in m/etmaal)</i>	<i>Classification Permeability</i>
$k > 5$	zeer goed
$5 < k < 1$	goed
$1 < k < 0,5$	redelijk
$0,1 < k < 0,5$	matig
$0,01 < k < 0,1$	slecht
$k < 0,01$	zeer slecht

Table 3 Permeability (De Vree, 2015) Courtesy ir. d.j. biron, 2004)

6.1.2 Storage capacity

Related to the infiltration capacity of the subsurface, the ‘static’ water storage capacity of analyzed open soil locations can be determined. The water storage capacity is dependent on the porosity of the soil – a value that is also soil type dependent. The amount of space (pores) between the parts of the soil, the height of the groundwater levels and the overall volume of the analyzed location, determine the water storage capacity of a soil (Bodemvenster, 2015).

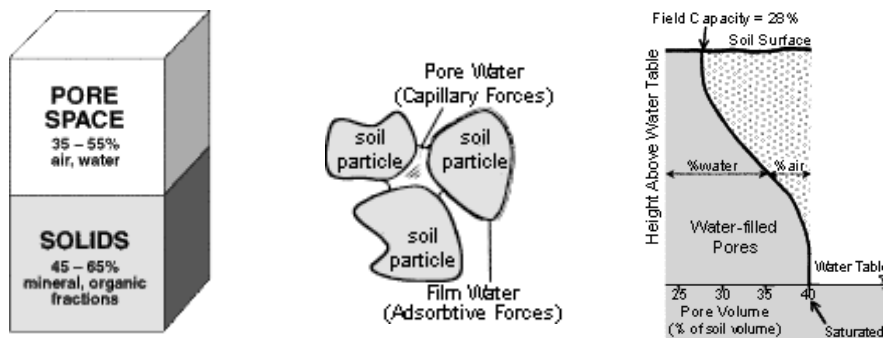


Figure 72 Main soil content (left), pores filled with water (middle) and relationship of air filled and water filled pores related to the water table (Sands, 2001)

Pores of the soil can be either filled with air or with water; rain water can be taken into the pores that are filled with air. Water in pores is held via capillary forces (Sands, 2001), when the soil is saturated by water and cannot take in more water, field capacity is reached. The volume of water a soil type can hold at field differs per type and through time (Sands, 2001), the level where soil pores are saturated is called the water table (related to groundwater levels).

An estimation about the amount of water that can be stored at a ‘static’ moment in time, can be made based on the soil type and volume of the unsaturated zone, excluding refined geohydrology such as water runoff, evaporation, absorption by vegetation.

Important for the water storage capacity is the effective (or drainage) porosity, which entails the porosity value of the soil that is available for fluid flow (Brady, 2003). Soil containing sand has a larger effective porosity than clay soils, as can be seen in Table 4.

Soil texture	Field capacity (% by vol.)	Drainable porosity (% by vol.)
clays, clay loams, silty clays	30-50%	3-11%
well structured loams	20-30%	10-15%
sandy	10-30%	18-35%

Table 4 Drainable porosity (Sands, 2001)

Based on the effective porosity percentages can be used to calculate the water storage in volume of the unsaturated zone.

6.2 Working Demo

The Application of this thesis research is an online accessible working demo that is the result of proceedings proposed and tested in Chapters 4 and 5 respectively. The GIS that is used to implement the demo is ArcGIS of vendor ESRI. In ArcScene, the GeoTOP voxel dataset will be represented here as 3D surface based vector data. The 'map of the unbuilt terrain' based on TOP10NL is represented as 2D vector data that is extruded into 3D in order to make a 3D intersection operation. The TOP10NL data indicates the open soil locations that can be analyzed.

Additionally, the 2D line and point vector data of KLIC is buffered. The KLIC data indicates the sewer system, and serves as a refinement for the working demo: by subtracting the volume of the sewerage with the volume of the unsaturated zone, the available soil volume that can take in rain water can be determined. This resulting unsaturated soil volume is required to assess the water storage capacity of the analyzed area.

Colors of the vectorized GeoTOP output geometry can be assigned based on the GeoTOP attribute, each color is affiliated with the soil type:

	1 (organische stof (veen))	Peat
	2 (klei)	Clay
	3 (kleilig zand en zandige klei)	Clayish sand
	5 (zand fijn)	Fine sand
	6 (zand midden)	Sand
	7 (zand grof)	Coarse sand
	8 (grind)	Grind
	NN (Unknown)	Unknown

Figure 73 GeoTOP litho class attributes – colored (Stafleu et al., 2013)

ArcGIS adopts – via CGAL conversions – the principles of the Nef Polyhedron, which can be regarded as one of the most complete polyhedron models (Hachenberger, 2006).

The working demo comprises of a spatially analyzed part of the city of Dordrecht via 3D intersection operations that preserves the semantics and attributes of the input data. Furthermore, the demo contains average water level flows represented as surfaces between set groundwater measuring pipes, and the demo is published online as a WebScene.

The parts of the city of Dordrecht which are analyzed are chosen for their known water related issues (via reported water disturbances), the districts Wielwijk, Dubbeldam and Sterrenburg. The working demo comprises of a part of the greenzone alongside the Rechte Zandweg, and is located near the city district Dubbeldam – a focus area for the municipality of Dordrecht.

The output of the 'main 3D GeoTOP intersection' operation between the vectorised GeoTOP and map of the unbuilt terrain is a set of GeoTOP Multipatches ranging from its surface ground level till 50m below

NAP. In order to retrieve in this intersection result, the critical z value, groundwater levels must be appointed in this 3D intersection result. The soil type in the zone which is set by the critical z value, determines the infiltration capacity of the subsoil.

6.2.1 Groundwater levels in Dordrecht

In the municipality of Dordrecht, groundwater levels are monitored via measuring pipes built in the subsurface. These pipes are divided into primary, secondary and tertiary installments. The secondary and tertiary network are installed for specific projects initiated by the municipality of Dordrecht, thus for this research the general, primary network is used.

The primary groundwater measuring pipe network of Dordrecht, is monitored frequently and in these tubes the water levels are measured and stored each month. A database of the groundwater level data with x,y coordinates of the measuring pipes, is stored by the department of ‘Urban Management’ (nld. Stadsbeheer) of Dordrecht, plus the data is published each month on the main municipality website of Dordrecht – freely accessible.

However, online location data of the measuring pipes is not stored as coordinates, but the locations are visualized on a map. For the research in this thesis, the online published groundwater level data is used. The groundwater data is used as average. In ArcScene, the measuring pipes are visualized as points, with determined coordinates, seen in Table 5 and Figure 74.

Primary network Measuring pipe	Groundwater levels in meter relative to NAP	Assigned coordinates
1	-1.74	107713, 423374
2	-1.47	107783, 423092
3	-1.84	108059, 423153
4	-1.42	108158, 423412
5	-1.91	107931, 423367
6	-1.74	107865, 423221

Table 5 Groundwater levels via 3D points



Figure 74 Groundwater levels as 3D surface between points

With simple Python scripts, the groundwater level data is generated as x, y, z points in a .shp file. This .shp file is added in the 3D intersection result in ArcScene. The adopted groundwater levels are visualized in GIS as a surfaces built out of triangle polygons.

6.2.2 Building the working demo

One of the investigated areas is the city district Dubbeldam – an area prone to water related issues, as portrayed in a map by the municipality displaying water nuisance alerts of civilians. Based on groundwater level/ phreatic level data, an isohypses map can be conducted. The ‘Rechte Zandweg’ and two open soil location alongside the ‘Nijhofflaan’ are analysed here, in order to determine the water storage and infiltration capacity of the unsaturated zone of certain areas (coloured red in Figure 75).

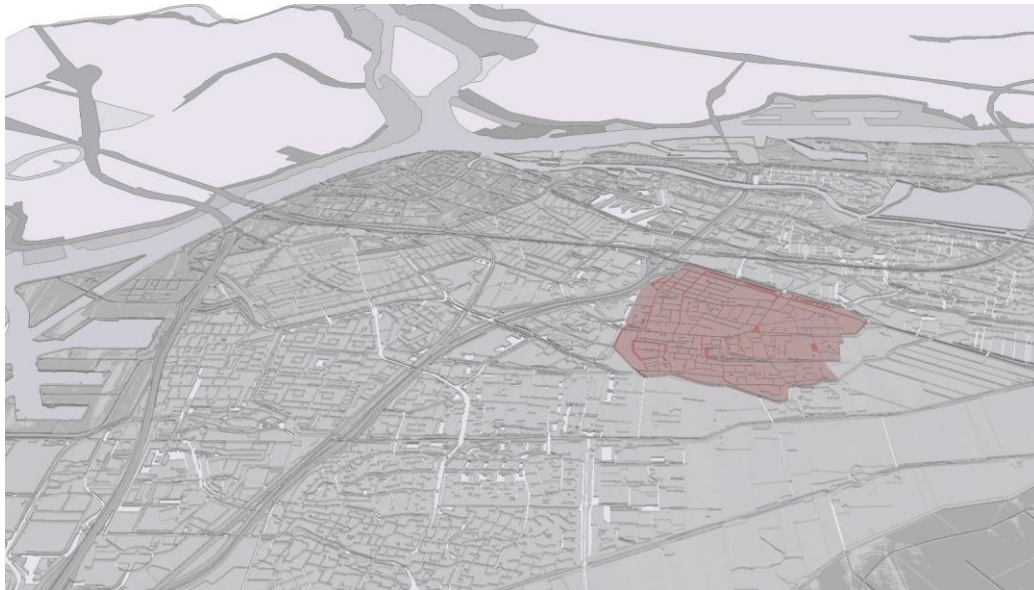


Figure 75 Area of interest: city district Dubbeldam

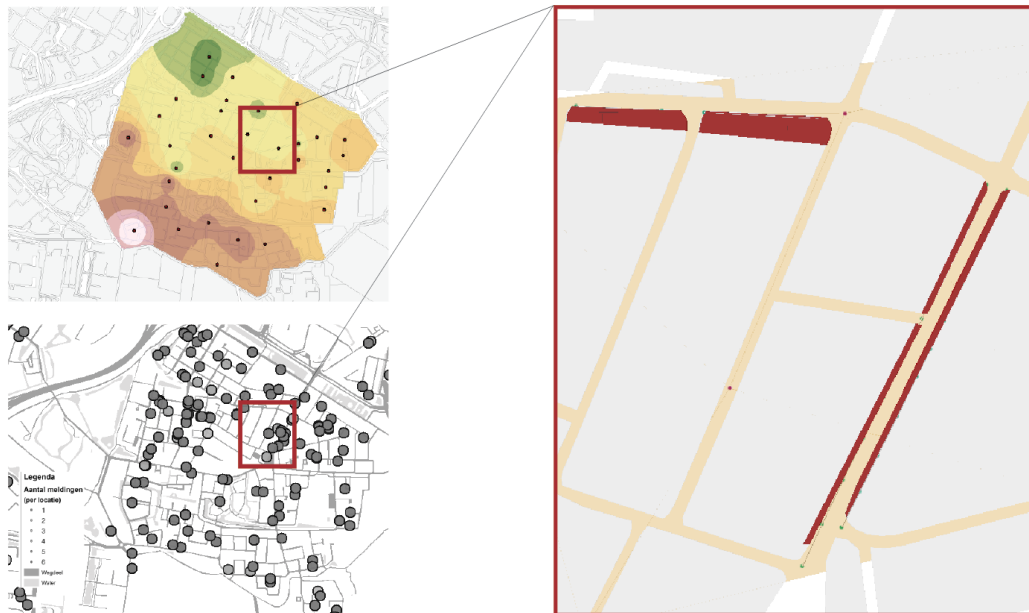


Figure 76 Demo interpolation (upper left), water nuisance on the street 2008-2015 (lower left), selected open soil locations - red strips to analyze (right)

To make the 3D intersection for the indicated red areas, modified TOP10NL terrain data is used: the red polygons are extruded vertically to make the intersection with the vectorised GeoTOP data. The surface in orange represents the average groundwater levels, based on the groundwater level data of the measuring pipes in the area.

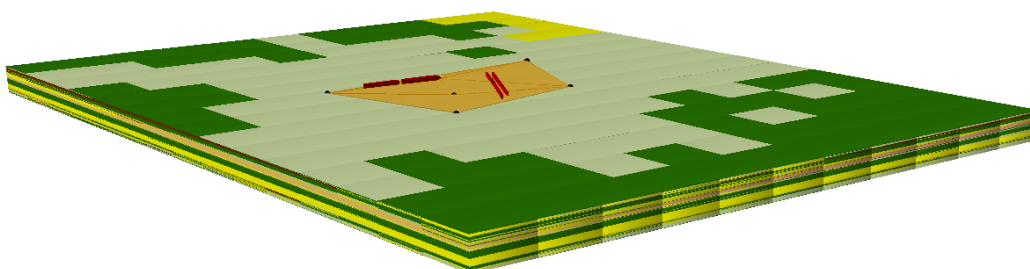


Figure 77 Part of the adopted, vectorised GeoTOP map sheet 44w

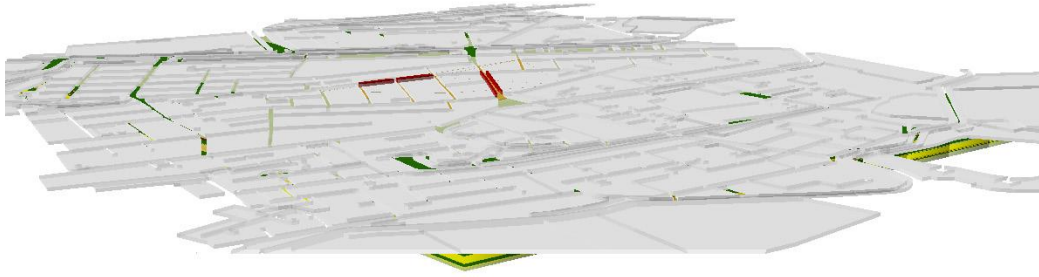


Figure 78 Modified TOP10NL data consisting of extruded surfaces

6.2.3 Working demo results

In Figure 79-86 the working demo result is shown: the 3D intersection operation result of the 'Rechte Zandweg' combined with groundwater data, represented as surfaces.

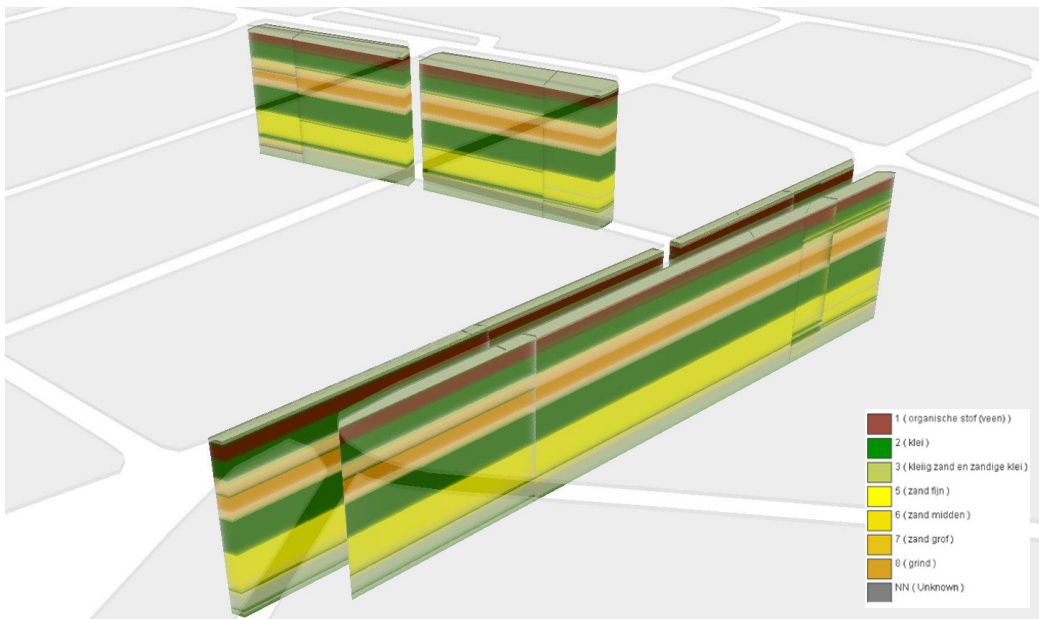


Figure 79 Working demo result

With the 3D GeoTOP-TOP10NL intersection, the unsaturated zone of the subsurface can be determined when groundwater level data is placed into the model (orange phreatic surface, Figure 77 and Figure 86 (right)). Litho class (soil type), volume and surface area can all be calculated with this intersection result. Also infiltration and storage capacity values can be linked to the intersection outcome: here light green represent 'clayish sand', which has a very small K- value/permeability. To enhance the model, sewer data of KLIC can be incorporated, and can be used for e.g. further intersections, or volume determinations (Figure 80 and 81).

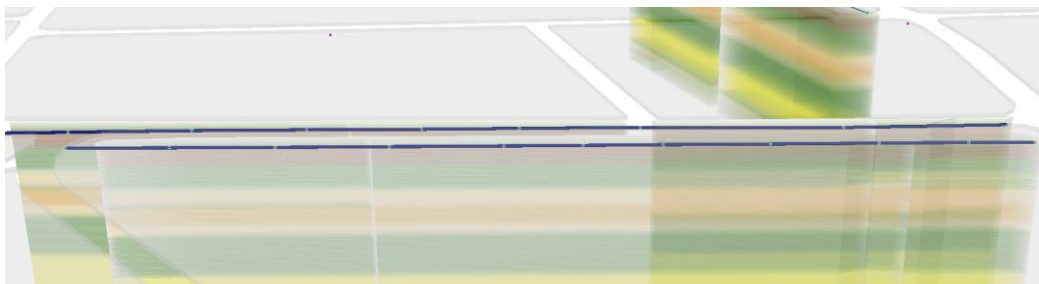


Figure 80 Modelling part of the sewer system

With the 3D modelling of the sewer system, out of which soil type the sewerage is composed can be determined. Plus volume calculations can be performed, via 3D intersection operations between the vectorised GeoTOP and the sewer geometry (Figure 83).

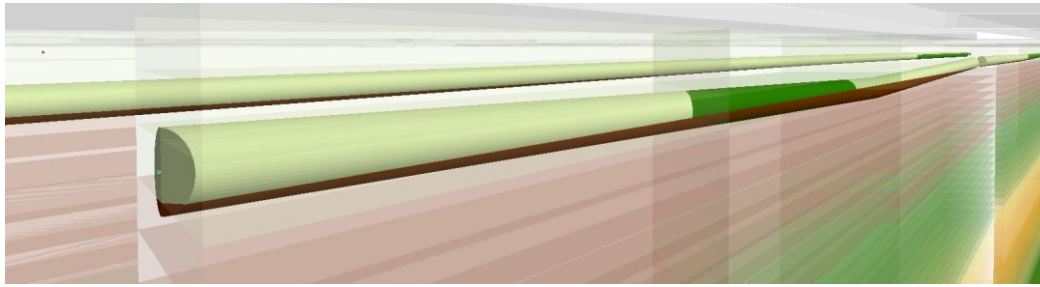


Figure 81 Modelling part of the sewer system and performed 3D intersection operation

Since large part of the sewerage consists of gravitational sewer parts (with a slope), some parts of the sewer are not present in the unsaturated zone as can be seen in Figure 82. Other parts however do overlap with the unsaturated zone, as is the case in the northern parts of this analyzed area, shown in Figure 83 and 84. In the latter cases, volume calculations of the subtraction between the unsaturated zone as the sewer system can be made. The resulting volume can be used to determine the static water storage capacity of that specific area.

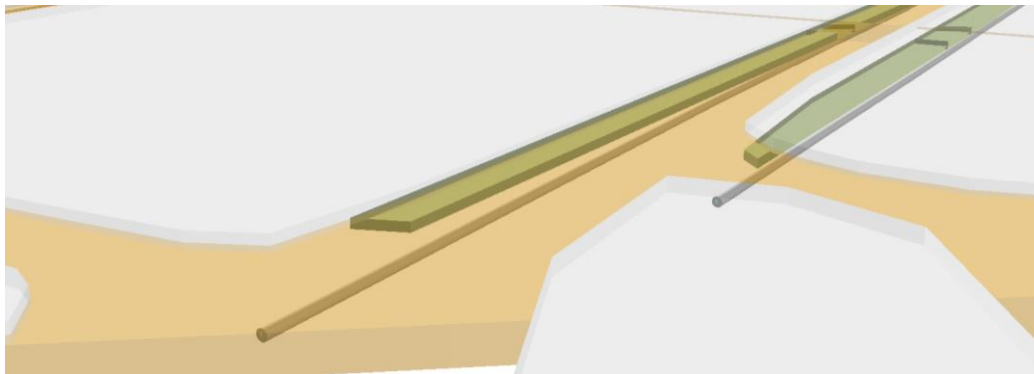


Figure 82 Unsaturated zone and the sewer system do not overlap

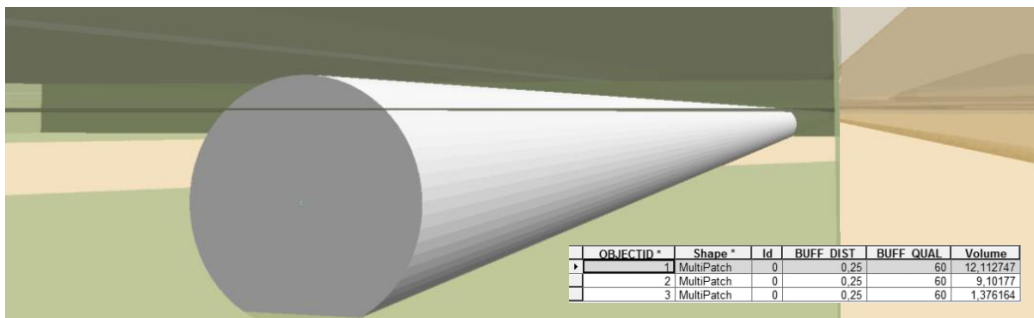


Figure 83 Unsaturated zone and the sewer system overlap each other

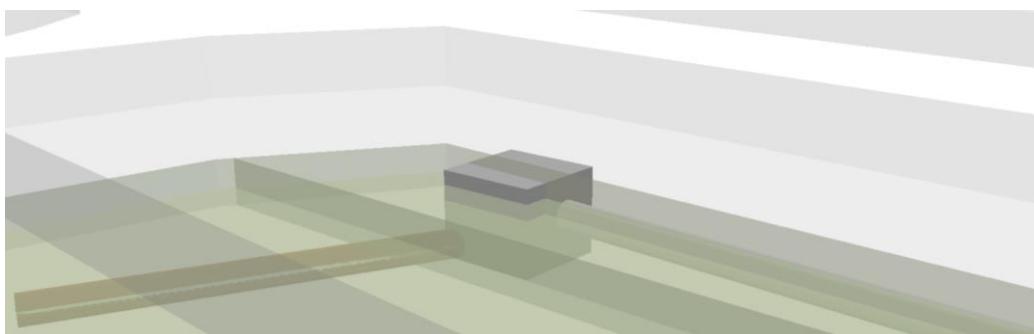


Figure 84 Sewer pipes and sewer connection part in the unsaturated zone

In order to retrieve the unsaturated zone/ drainage depth, another 3D intersection is performed between the upwards extruded groundwater surface, and the upper parts of the earlier GeoTOP intersection output. What can be seen in Figure 85, is that the unsaturated zone of the Rechte Zandweg green zone parts mainly consists out of clayish sand (GeoTOP litho class attribute 3), colored light green. A verification: the TYPE attributes of the output of the latest 3D intersection, indicate that the complete drainage depth consists out of TYPE nr 3 – clayish sand.

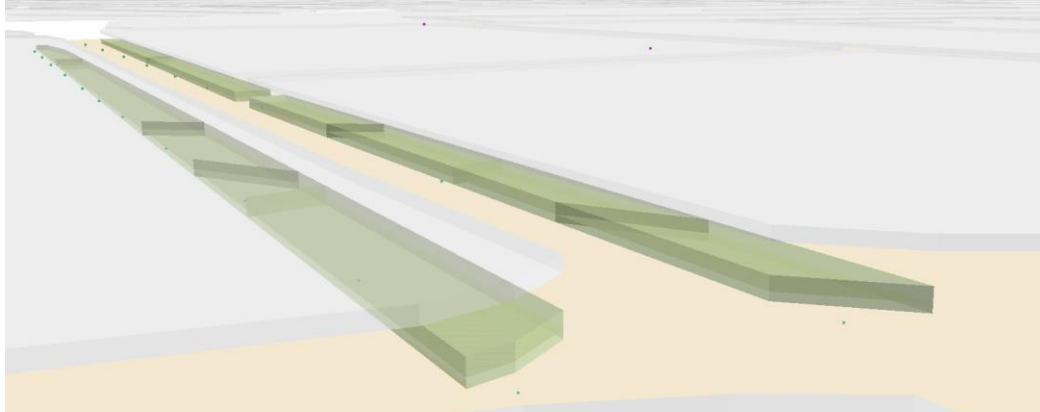


Figure 85 Unsaturated zones in the Rechte Zandweg

The interior volume of the Multipatches which form the result of the second 3D intersection operation, can be determined and added as attribute. Via the *Add Z Information* tool, an interior volume calculation is made, and placed in the attribute table.

For the determination of the water infiltration and storage capacity of the working demo, an open soil location alongside the 'Nijhofflaan' is analysed (Figure 76). The drainage depth of this green zone is partly occupied by the sewerage, which then will be subtracted in the adopted volume determination for the purpose of a statement about the water storage capacity.

Determination Infiltration Capacity – at T1000:

With two 3D intersections (one with GeoTOP and TOP10NL, and with this output result, one with (extruded) groundwater levels), the unsaturated zone/drainage depth of the subsurface on an open soil location could be determined. Via the attributes of the final intersection output, soil type with its overall volume can be determined. In this working demo of the Application, clayish sand can be linked to the *permeability K* of approximately 0.02 up to 0.2 (in the best scenario) meters/day.

Very recently, in week 35 of 2015, heavy rainfall caused disturbances in several city districts of Dordrecht. This rainfall was categorized as a T1000 shower (is expected to occur once in a 1000 years) and resulted in a measured 80 mm (0.08m) of precipitation in one day – 24 hours. The rate that will infiltrate in the soil in a medium dense urban area is approximately 40%, because 37% of precipitation is runoff and 23% is evaporated. Thus: $0.40 \times 0.08 = 0.032$ m of precipitation is the amount that can infiltrate in the soil.

Seen on a general perspective – leaving out several environment and refined hydrological factors – the infiltration capacity of the unsaturated zone of the Rechte Zandweg should be able to cope with a 0.032 m (being a value between 0.02 and 0.2 *K*) precipitation in one day.

Determination Water Storage Capacity – at T1000:

1. A T1000 shower results in 80 mm (0.08m) of precipitation (in one day)
Surface area of the 'Nijhofflaan' green zone: 570 m^2

The amount of water volume to be stored via solely soil infiltration (minus runoff and evaporation in dense urban areas) is 40%, leading to:

$$0.40 \times 0.08 = 0.032 \text{ m}$$

The amount of precipitation calculated into volume m^3 :

$$0.032 \text{ m} \times 570 \text{ m}^2 = \underline{18.24 \text{ m}^3 \text{ rain water to be stored via soil infiltration.}}$$

2.	Volume of the unsaturated zone of the 'Nijhofflaan' green zone:	801 m ³
	Volume that the sewer system occupies at the 'Nijhofflaan' green zone:	$\frac{12 \text{ m}^3}{789 \text{ m}^3}$
The effective porosity of soil type 3 is assumed to be 11% (seen in Table 4), leading to:		
0.11 x 789 = <u>86.79 m³ is available to for water storage via soil infiltration.</u>		

The soil present in the unsaturated zone of the 'Nijhofflaan' green zone should be able to store the water of a T1000 shower as 18.24 m³ < 86.8 m³. This is a statement about a static moment in time.

6.2.4 Web publishing

The working demo can entirely be published on the web, and can be shared via a link – if accessibility restrictions are not made.

The online publishing can be performed to export the scene in which the demo is built, to a .3ws extension. The CityEngine is used to perform the export and publishing. The resulting WebScene can be opened in a browser that supports WebGL, and it consists out the layers (dataset) which are used in the 'desktop' scene. Also attribute of features can be displayed, as shown in Figure 86.

<http://rhk.maps.arcgis.com/apps/CEWebViewer/viewer.html?3dWebScene=92fd8079e3de4c3abf4d5a64f5248f5c>

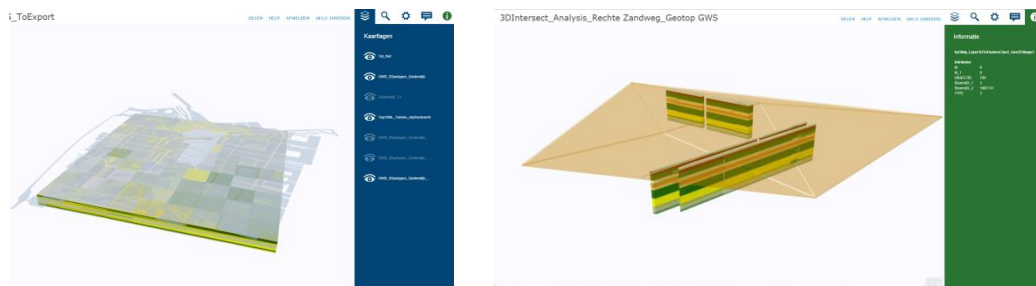


Figure 86 WebScenes

6.3 Application Findings

This Application pointed out that a working demo allows basic geohydrologic determinations, through spatial analysis via 3D intersection operations on surfaces. Maintaining semantics and especially attributes in the output, are evident for the working demo to be applied in geohydrology.

To determine the infiltration and storage capacity of an open soil location, several datasets are required. In this Application, GeoTOP and TOP10NL formed the base for the first 3D intersection. To go a step further in geohydrologic analyzing, groundwater level can be modeled that serve as a base for a second 3D intersection that unveils the drainage depth of the upper part of the subsurface. The associated *permeability K* can be attached to the initial GeoTOP litho class attribute, giving insights in the infiltration and storage capacity of an investigated part of the subsurface. Also, KLIC data is used to refine the model with the sewer system and to perform volume calculations that are used for water storage determination of an analyzed area.

The working demo of the 'Nijhofflaan' green zone showed that this open soil location is capable – due to the soil type and volume content of the unsaturated zone – to infiltrate and store precipitation of a heavy T1000 shower.

7. CONCLUSIONS

7.1 Conclusions and Answered Research Questions

As a general conclusion it can be stated that voxel data, 2D surface data and 2D line and point data can be converted to 3D surface based vector data, via different workflows. 3D Intersection operations on subsurface data represented as surfaces in commonly used GIS offer a sound base for geohydrology principles.

The Nef polyhedra are a suited data structure to perform 3D Boolean set operations with and to attach attributes with geometry – when applied in the field of GIS. The implemented Nef_Polyhedra_class of CGAL by ArcGIS does not account for all Nef Polyhedra, with respect to the theory.

More elaborate and refined conclusions are described in the answers to the research questions.

7.1.1 Main research question

In what manner can a GIS facilitate 3D intersection operations for voxel data represented as surfaces, while maintaining semantic and attribute information in the output?

In general:

Few GISs can adopt and execute extensive three dimensional spatial analysis on voxel data, the conversion of voxel data to 3D vector data is recommended regarding the availability of 3D operation tools in commonly used GISs. Converting various data representations into the 3D surface based vector data representation, asks for different conversion approaches. The conversion of 2D surface data and 2D line and point data is straightforward, most of the conversion can be done in GIS, in some cases extra COLLADA conversion and subsequently external editing in a modelling software is required. The voxel conversion to 3D vector data is more extensive, as it requires database processing and scripting.

The properties of the Nef Polyhedra data structure are suited for 3D Boolean set operations, together with attribute storage. Based on CGAL's Nef Polyhedra description, ArcGIS can perform 3D Boolean operations that provide closed geometry as output. With 3D intersection operations, it is possible to maintain semantic information (in the form of layers) and attribute information in the output.

A GIS that fully incorporates the theory of the Nef Polyhedra, offers a sound method to perform 3D intersection operations on surfaces with semantics and attribute information attached to the output. Thus, voxel data represented as 3D vector surfaces can output valid geometry, with semantics and assigned attributes. Commonly used GIS such as ArcGIS is suited for spatial analysis in the form of 3D intersection operations.

Regarding the Application:

Considering the context of this research, data that is used is from the Dutch National Mapping Agency and TNO. The input data of voxels represent the geological subsurface (GeoTOP), the 2D surface data represents terrains on ground surface level (TOP10NL) and the 2D line data represents cable and pipe data (KLIC).

The data used for the working demo is the GeoTOP voxel dataset and a (extruded) 'map of the unbuilt terrain' indicating open soil locations in the city of Dordrecht based on TOP10NL, and KLIC data. When all datasets are represented as 3D vector data (surface based), 3D intersection operations can be performed in ArcGIS of ESRI with the 3D Analyst extension. Such 3D spatial analyses offer the base for a working demo through which water infiltration and storage capacity of the drainage depth of the subsurface can be determined. The demo can give new insights in the currently increasing water related issues in urban areas.

The 3D intersection operations required for the Application are stable: valid geometry is the outcome, with semantics and correctly assigned GeoTOP attributes.

Regarding ArcGIS:

For 3D Boolean set operations, Multipatch geometry in ArcGIS is converted to the CGAL Nef Polyhedra data structure – on the background, while performing a Difference 3D, Union 3D or Intersect 3D operation. From the performed Test Cases in this research, the Multipatch affiliates to the polyhedron rules as stated by Van Oosterom en Stoter (2006) seen in section (2.2.3). For instance, three tested non-manifold cases as described in section (2.3.3) were not able to construct one unified polyhedron geometry via the set operator *Union 3D*. The non-manifold test cases delivered unintentional result for *Difference 3D* and *Intersect 3D*.

Thus, the properties of a Nef Polyhedron do not all apply in ArcGIS's implementation. The validity rules of a 'regular' polyhedron as stated in section (2.2.3) seem to apply (such as a required single bounded, closed polyhedron), but do not apply entirely. The Test Cases (2.4 and 4.1) pointed out that through 3D operations such as Intersect 3D non-closed geometry can come out. Exporting the concerning geometry to COLLADA, import it in 3D modeling software where it can be closed, and exported again in ArcGIS are the required steps to fix a non-closed 'polyhedron' in Multipatch geometry.

ArcGIS offers a comprehensive implementation of CGAL's Nef Polyhedra, though the drafted Test Cases showed the limits of this implementation: no non-manifold Nef Polyhedra could be adopted.

However, the Nef_Polyhedra_3 based *Intersect 3D* operation in ArcGIS provides a sound method to perform 3D intersection operations that result in valid geometry output, with semantics and correct assignment of attributes. Additionally, accurate volume and surface area calculations can be performed.

7.1.2 The sub questions

- What generic workflow is required to retrieve a 3D intersection with surface represented objects?

Regarding the distinct data type, a suited conceptual conversion workflow to 3D, surface based vector data can be drafted. There are different conversions for difference types of data representations. In this research a conceptual conversion workflow for (i) voxel data, (ii) 2D surface data and (iii) 2D line and point data is proposed. Each of these conversions comprises of different steps to get to 3D vector data as output. When all input data is in the same data representation (3D surface based vector data), 3D intersection operations can be performed.

- The voxel conversion workflow to 3D vector data, requires the 'raw' voxel data in a database. A script can convert a voxel datasets to centroid points with x, y, z coordinates together with their attributes. This voxel to 3D point conversion forms the main, significant step towards vectorization of the voxels. The actual surface representation requires the points to be constructed as boundary represented cube objects, in accordance to the sizes of the initial voxels. A 3D shell of a set of surfaces can form one voxel. This latter conversion step can be performed in commonly used GIS.

- 2D surface data represent a plane enclosed by a polygon. A workflow that converts 2D surface data to the third dimension, requires extrusion of the surface data in z (vertical) direction. Vertical extrusion ensures the preservation of the footprint (x, y) of a surface, while converting the 2D vector data to 3D.

- 2D line data represents a linear shape. A workflow which converts 2D line data to 3D vector data, requires buffering of the line data in all three dimensions.

3D Intersection operations that maintain semantic and attribute information, can be drafted into a conceptual workflow:

- A workflow for 3D intersection operations on surfaces, requires solely closed (valid) 3D vector data as input. The overlapping part of the two input geometries define with its boundary represented surfaces, the volume of the result of the intersection. A cut is performed, for each distinct overlapping object at a time, and the cutline defines a new generated geometry. The new geometry adopts the semantic and attribute information of the input objects. A validation on geometric validity and on correct assignment of attributes needs to be performed.

The conversion workflows that are needed in order to make a 3D intersection operation on the same data representation (3D vector data, surface based), work for the specific input data types, i.e. voxels, and surfaces and lines. Thus the workflows do not account for other data types and representations.

- How can 3D intersection operations be performed in GIS?

As a three dimensional intersection on surface based geometry is regarded as a 3D Boolean set operations, a GIS should offer a 3D geometry representation and provide an intersection tool for 3D geometry that is able to extract attributes per intersecting feature/object of the valid input geometry and to insert these attributes in the valid intersection output. ArcGIS includes the B-rep Multipatch geometry (3D vector data) and offers 3D Boolean set operations as the *Difference 3D*, *Union 3D* and *Intersect 3D* set operations, available in the 3D Features Toolset of the extension 3D Analyst. Additionally, the operation *Intersect 3D Line With Multipatch* allows a 3D polyline geometry and 3D Multipatch geometry as input.

Difference 3D, *Intersect 3D* and *Union 3D* use the CGAL implementation of the Nef Polyhedra. In order to do so, ArcGIS uses CGAL source code to convert the input Multipatches to CGAL data structures on the background, and subsequently performs the 3D Boolean set operations.

A 3D intersection operation can be performed in GIS, in the chosen test environment ArcGIS. Here, the set operation *Intersect 3D* adopts only valid geometry, which is in ArcGIS a closed Multipatch. Input geometry has to be 3D Multipatch, output geometry, can be a point, line (when *Intersect 3D Line With Multipatch* is used). Interior volume calculation is used to compute the geometric output, for each distinct intersecting object. What *Intersect 3D* does, is it locates as the geometry objects that intersect, and then assigns per object the attributes of both of the input geometries to the intersecting output. With the *Is Closed 3D* operation, the output can be verified for geometric validity.

Notable is that in this research, the CGAL implementation of the Nef Polyhedra is not tested. What is tested and evaluated (on various Test Cases) is CGAL's Nef_Polyhedron_3 class implementation of ArcGIS. Thus, issues regarding the performance of the *Difference 3D*, *Union 3D* and *Intersect 3D* of ArcGIS could be present because of the ArcGIS implementation of the Nef_Polyhedron_3, therefore those issues do not account for the CGAL implementation of the Nef Polyhedra.

- How can semantic and attribute information of the 3D intersection output be maintained in GIS?

In GIS the semantics of a feature are stated in a layer, and each feature has its accompanying attributes. A 3D intersection operation in GIS should obtain attributes of each intersection feature and place those in the new output geometry – which is indicated by a new layer.

In all cases where the input geometry was considered valid (closed Multipatch), the output was placed in a new layer (covered the semantics in GIS), attributes were correctly assignment to the geometric output. The output of 3D solid operations in ArcGIS, is placed in a new layer that covers the semantics of the output, e.g. named as '3D intersection of the Zuidendijk Park Dordrecht'. The data that is presented in this layer (i.a. the valid, closed geometry and attributes) can be exported as a .shp file, that enables the storage of the output data. The attributes belonging to the original input geometry are attached after a 3D intersection operation, to each distinct geometry output feature.

The data in the .shp file containing the geometry and attributes, can be separately opened, for further usage such as additional analysis.

7.1.3 Findings regarding the Test Cases

The focus of the Test Cases lied on 3D intersection operations as being a 3D Boolean set operation, available as *Intersect 3D* in ArcGIS. Additionally, *Union 3D* and *Difference 3D* operators are performed, operations that could be of use for further processing (e.g. comprehensive sewer capacity calculation which is outside the scope of this research). The *Intersect 3D* and *Difference 3D* operations showed a stable performance and outcome. Being solid operations, *Difference 3D* and *Union 3D* only allow solids (i.e. closed 3D Multipatches in ArcGIS) as input. For instance, the *Union 3D* operations lacked the ability to unify features that were (i) not part of the same data sets (e.g. two different layers in ArcGIS), or (ii) features of a different representation (e.g. a surface with a 3D Multipatch).

The proposed cases were tested on: valid geometry (with semantics) correct assignment of attributes and volume/ surface area calculations. The Test Cases to perform *Intersect 3D*, *Difference 3D* and *Union 3D* operations on, delivered in general stable outcome. The complex geometric cases (e.g. case number 3.1) however, delivered unstable results, indicated by warnings for a 'not simple' outcome, or even the inability to perform the 3D operations (cases 3.2 and 3.4). (Additionally, in most Test Cases, surface area calculations could be performed, and in some cases volume calculations as well).

A more experimental case regarding geometric validity, was not possible to construct in ArcGIS, or the 3D operation did not allow the actual execution of the operations. This was for example the case when a closed Multipatch is needed to be unified via *Union 3D* with a dangling face (a condition such as portrayed in Figure 16) could not be tested in ArcGIS. *Union 3D* only works with closed 3D Multipatches. Appendix C contains more background information on this experimental case.

Overall, the cases that adopted complex input, resulted in sometimes invalid, non-closed geometry were subsequently no volume calculations could be determined of. But, a complex case also pointed out that even valid, closed geometry could sometimes not deliver volume calculations, thus no consequent relation between valid geometry and volume calculations can be made. The input geometry could be too complex to determine its interior volume.

The non-manifold cases, pointed out that not all Nef Polyhedra can be obtained in the test environment, as several union operations on non-manifolds delivered invalid output.

7.2 Discussion

The chosen approach in this thesis research is converting all three input datasets to 3D surface based vector data. While two of the used datasets in this research comprises vector data (TOP10NL and KLIC), one dataset was voxel data (GeoTOP), or 3D raster. Especially the GeoTOP voxel dataset – even seen on map sheet level – is large in storage space. The map sheets of a GeoTOP voxel dataset contains approximately 9 million voxels and is thus of extensive size once loaded and modified in ArcGIS (2.00GB). ArcGIS is able to work with such large georeferenced datasets, and perform 3D operations of selected parts of the data.

The Application required the soil types of the upper part of the subsurface. In the Netherlands, GeoTOP is the most accurate model that entails such information. However, the voxel resolution is of a $100*100*0,5\text{m}$, an order of magnitude that might lead to misinterpretations of the analysis, because it results in rough borders between different soil types.

Converting an entire GeoTOP voxel map sheet to 3D points, lead to usable output in accordance to the set georeference. Processing all data point into GIS – operations that took relatively long time to achieve – lead to misplacements of some sets of voxels, a phenomenon that might occur cause of the large amount of 3D data that need to be loaded into GIS.

Regarding the Application, as groundwater levels are dynamic, modelling them as static surfaces is automatically a simplification of the real- life situations. Though some basic geohydrologic principles can be determined based on the Application, more refined calculations need more hydrologic information such as accurate precipitation data, or knowledge about capillary groundwater effects.

7.3 Recommendations for Future Work

A more accurate level of detail of the GeoTOP dataset would benefit the spatial analysis as performed in this thesis research via 3D intersection operations. More substantiated insights regarding the geohydrological properties of the upper part of the subsurface could then be set up.

As the GeoTOP map sheets represent a large area, incorporating or import the entire groundwater database as GLG/GHG values of larger areas (for instance data on city scale) would be beneficial for geohydrological analysis on GeoTOP.

7.5 Reflection

This thesis research and its topic had for me had a sound connection with the Geomatics Core Courses, as held in the first year of the Geomatics master program. Several topics that I find very interesting were dealt with in this thesis, such as three-dimensional GIS and different data representations.

From my perspective, the research opened my view on the potential of 3D Boolean set operations performed on real – life datasets, and applied on a practical case. This unveiled the power of spatial analysis on three dimensional geo-data.

More in detail, the conversion of voxel to 3D surface based vector data was more elaborate than initially imagined. Dealing with large datasets, requires significant pre – processing.

I liked the fact that in this research I could literally set refined theories against various cases. Overall, I experienced the importance of three dimensional geo- data for various purposes, and I especially liked studying and working in an interdisciplinary field where geo-engineers, specialists, hydrologists and subsurface experts all took part in.

REFERENCES

- Agu, E. (2015). Computer Graphics 3D Clipping and Viewport Transformations. Retrieved 27 July 2015, from: <http://web.cs.wpi.edu/~emmanuel/courses/cs543/slides/lecture8.pdf>
- Arroyo Otori, K., Ledoux, H., Stoter, J. (2015). An evaluation and classification of nd topological data structures for the representation of objects in a higher-dimensional gis. *International Journal of Geographical Information Science*
- Arroyo Otori, K., Ledoux, H., Meijers, M. (2012). Validation and Automatic Repair of Planar Partitions Using a Constrained Triangulation. *Photogrammetrie, Fernerkundung, Geoinformation*
- Becker, S., Walter, V., Fritsch, D. (2012). Integrated Management of Heterogeneous Geodata with a Hybrid 3D Geoinformation System. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences.* - I-2. ISPRS
- Bentley (2015). Retrieved 21 July 2015, from http://communities.bentley.com/products/microstation/w/microstation__wiki/13818.intersection-between-line-and-3d-element
- Berntssen, M., Danes, M., Goos, J., Klooster, R., Kooijman, J., Noordegraaf, L., Stoter, J., Veldhuis, C., Vosselman, G. (2012). 3D Pilot Eindrapport werkgroep 3D Use cases. NCG KNAW
- Bieri, H. (1995). Nef Polyhedra: A Brief Introduction. In *Dragstuhl*. 10: pp43-60. Vienna: Springer-Verlag.
- Biermann, H., Kristjansson, D., Zorin, D. (2001). *Approximate Boolean Operations on Free-form Solids*. NYU Media Research Labaoratory
- Bodemvenster (2015). Retrieved 15 October 2015, from: <http://www.zeeuwsbodemvenster.nl/themas/water/waterberging>
- Brady, M. M., Kunkel, L. A. (2003). *A Practical Technique for Quantifying Drainage Porosity*. PTS Laboratories
- CGAL, Nef_Polyhedra_3, (2015). Retrieved 23 July 2015, from: http://doc.cgal.org/latest/Nef_3/classCGAL_1_1Nef__polyhedron__3.html
- CGAL, Polyhedron (2015). Retrieved 23 July 2015, from: http://doc.cgal.org/latest/Polyhedron/classCGAL_1_1Polyhedron__incremental__builder__3.html
- CGAL, Halfedge (2015). Retrieved 23 July 2015, from: <http://doc.cgal.org/latest/HalfedgeDS/index.html>
- CGAL, Nef_3 (2015). Retrieved 27 July 2015, from: http://doc.cgal.org/latest/Nef_3/index.html
- Crawford, C. (2015) – Lead Product Engineer, Software Products Group ESRI
- Cultuurtechnisch vademecum (1988), Cultuurtechnische vereniging
- De Cambray, B. (1993). *Three-Dimensional (3D) Modelling in a Geographical Database*. Laboratoire PRISM
- De Graaf, R. E. (2009). *Innovations in urban water management to reduce the vulnerability of cities: Feasibility, case studies and governance*. Delft: Civil Engineering and Geosciences TU Delft
- De Volkskrant (2015). Retrieved on 03 August 2015, from: <http://www.volkskrant.nl/binnenland/rioolstelsels-op-schop-om-buien-aan-te-kunnen-a4109823/>
- De Vree, J. (2015) K-waarde. Retrieved on 12 July 2015, from: <http://www.joostdevree.nl/shtmls/k-waarde.shtml>
- De Vries, M. (2002), TOP10NL GML prototype, 2nd GML Relay, TU Delft.
- Defu, W., Pei, P., Khoon, T.Y. (2010). *The Digital Rock Engineering System Based On 3D GIS Technology*. Berlin: ISPRS
- DS Spatial (2015). Retrieved 10 September 2015, from: http://doc.spatial.com/articles/m/a/n/Manifold_and_Non-manifold_Objects_2264.html
- ESRI (2008). *The Multipart Geometry Type*. ESRI: New York.
- ESRI ArcGIS, Resource Center (2015), Retrieved 05-08-15, from <http://help.arcgis.com/EN/ARCGISDESKTOP/10.0/HELP/index.html#/Multipatches/00q8000000mv000000/> And from: <http://resources.esri.com/help/9.3/arcgisengine/dotnet/12105296-1a78-4cba-b306-c0a47d02a64b.htm> and

http://resources.arcgis.com/en/help/main/10.1/index.html#/An_overview_of_the_3D_Features_toolset/00q9000008p00000/

Frapporti, G., Vriend, P., Van Gaans, F.M. (1993). Hydrogeochemisrty of the shallow dutch groundwater: Interpretation of the National Groundwater Quality Monitoring Network. *Water Resources Research*, Volume 9 – 9 pp. 2993-3004

Flipcode (2015). Retrieved 04 August 2015, from: http://www.flipcode.com/archives/The_Half-Edge_Data_Structure.shtml

Ford, A. & James, P. (2005). *Integration of 3D petroleum datasets in commercial GIS*. Newcastle: University of Newcastle School of Civil Engineering and Geosciences

Gaul, R (2015). Understanding Sutherland-Hodgman Clipping for Physics Engines. Retrieved 05 August 2015, from: <http://gamedevelopment.tutsplus.com/tutorials/understanding-sutherland-hodgman-clipping-for-physics-engines--gamedev-11917>

Gia, T.A.N., Tran, P.V., Khac, D.H. (2013). Overview of Three and Four-Dimensional GIS Data Models. In: *Multimedia and Ubiquitous Engineering, Lecture Notes in Electrical Engineering – 240*. Dordrecht: Springer. pp 1013-1020.

GIS Commons (2015). Retrieved 01 September 2015, from <http://giscommons.org/files/2009/09/1.82.gif>

Glander, T., Dollner, J. (2008). Techniques for Generalizing Building Geometry of Complex Virtual 3D City Models. In: *Advances in 3D Geoinformation Systems, Lecture Notes in Geoinformation and Cartography*. Springer. pp 381-400

Guillen, A., Calcagno, P., Courrioux, G., Joly, A., Ledru, P. (2008). Geological modelling from field data and geological knowledge, Part II - Modelling validation using gravity and magnetic data inversion. In: *Physics of the Earth and Planetary Interiors*, Elsevier, 2008, 171 (1-4), pp.158

Hachenberger, P., Kettner, L., Mehlhorn, K. (2007). Boolean operations on 3D selective Nef complexes: Data structure, algorithms, optimized implementation and experiments. Elsevier

Hachenberger, P. (2006) *Boolean Operations on 3D Selective Nef Complexes: Data Structure, Algorithms, Optimized Implementation, Experiments, and Applications*, Universitat des Saarlandes.

Hegemann, F., Manickam, P., Lehner, K., Kock, C., A.M.ASCE, Konig, M. (2013). Hybrid Ground data Model for Interacting Simulations in Mechanized Tunneling. In: *Journal of Computing in Civil Engineering – 27*. American Society of Civil Engineers. pp 708-718

Hofierka, J., Zlocha, M. (2012). A New 3-D Solar Radiation Model for 3-D City Models. In: *Transactions in GIS – 16*, 5. Blackwell Publishing Ltd. pp 681-690

Hooimeijer, F.L. (2014). *The making of polder cities: a fine Dutch Tradition*. Rotterdam: Jap Sam Publishers

Jørgensen, F., Rønne Møller, R., Nebel, L., Jensen, N.P., Vest Christiansen, A., Sandersen, P. B. E. (2013). *A method for cognitive 3D geological voxel modelling of AEM data*. Berlin Heidelberg: Springer-Verlag

Kadaster (2015). Retrieved 09 September 2015, from: <http://www.kadaster.nl/web/artikel/producten/TOP10NL.htm>

Kadaster (2015). Retrieved 10 September 2015, from: <http://www.kadaster.nl/web/Themas/Registraties/KLIC-WION.htm>

Khan, I.H. (2005). *Textbook of Geotechnical Engineering 2nd Ed.*, Prentice-Hall of India, Pvt. Limited

Kobben, B. (2003) *SVG and the TOPNL Project*. Enschede: International Institute for Geo- information Science and Earth Observation (ITC)

Lattuada, R. 2006. Three-dimensional representations and data structures in GIS and AEC. In: Zlatanova, S. & Proserpi, D. (eds.), *Large-scale 3D data integration—Challenges and Opportunities*, London: Taylor&Francis, pp. 57-86

Ledoux, H., Gold, C.M. (2007). *The 3D Voronoi Diagram: A Tool for the Modelling of Geoscientific Datasets*. GéoCongrès

Michigan Technological University (2015). Retrieved 27 August 2015, from: <http://www.cs.mtu.edu/~shene/COURSES/cs3621/LAB/povray/csg.html>

Milner, J., Wong, K. & Ellul, C. (2014). *Beyond visualisation in 3D GIS*. London: University College London

Neteler, M., Bowman, M.H., Landa, M., Metz, M. (2011). *GRASS GIS: A multi-purpose open source GIS*. In: *Environmental Modelling & Software – 31*. Elsevier. pp 124-130

Newell, R.G. and Sancha, T.L. (1990). *The difference between CAD and GIS*. Butterworth & Co

- NOS article 1 (2015). Retrieved on 10 August 2015, from: <http://nos.nl/artikel/2032053-schade-door-regen-kockengen-weet-er-alles-van.html>
- NOS article 2 (2015). Retrieved on 10 August 2015, from: <http://nos.nl/artikel/2032004-verzekeraars-fors-meer-schade-door-regen.html>
- OGC (2015). Retrieved 12-09-15, from: <http://www.opengeospatial.org/standards/sfa>
- OGC GML (2015). Retrieved 29 August 2015, from: <http://www.opengeospatial.org/standards/gml>
- Pankaj, P. (2015). 3D Clipping. Retrieved 27 July 2015, from: http://www.academia.edu/6750506/3D_Clipping
- Pu, S. and Zlatanova, S. (2006). Integration of GIS and CAD at DBMS level. The Third Dimension, UDMS
- Restore (2015). Retrieved on 12 September 2015, from: <http://www.restore.ac.uk/geo-refer/images/Concepts%20-%20geographical%20objects%20in%20GIS.png>
- Sands, G. R. (2001). Agricultural drainage: Soil water concepts. Minnesota: Regents of the University of Minnesota
- Stafleu, J., Maljers, D., Busschers, F.S., Gunnink, J.L., Schokker, J., Dambrink, R.M., Hummelman, H.J., Schijf, M.L. (2013) GeoTop modellering. Utrecht: TNO
- Stoter, J. and Van Oosterom, P. (2006). 3D Cadastre in an International Context, Legal, Organizational and, Technological Aspects. Taylor & Francis Group
- Sunshine (2015), 2D Clipping with the Sutherland-Hodgman-Algorithm, Retrieved 05 August 2015, from: <http://www.sunshine2k.de/coding/java/SutherlandHodgman/SutherlandHodgman.html>
- TNO (2015). Retrieved 25 July 2015, from: <http://www.tno.nl/nl/samenwerken/expertise/earth-life-and-social-sciences/geomodelling/>
- Tuan, A.N.G. (2013). Overview of Three-Dimensional GIS Data Models. In: International Journal of Future Computer and Communication- 2. IJFCC
- Valstar, J., Maljers, D. (2013). Geohydrological modelling, predictions for an area- orientated approach for groundwater contamination in the City of Utrecht. City Chlor
- Van Maren, G. & Pluim, A. (2011). Demo van het combineren van voxeldata en eigen GIS-data. ESRI
- USGS (2015), Infiltration – The Water Cycle. Retrieved on 29 July 2015, from: <http://water.usgs.gov/edu/watercycleinfiltration.html>
- Varduhn, V., Mundani, R. P., Rank, E. (2014). Multi-resolution Models: Recent Progress in Coupling 3D Geometry to Environmental Numerical Simulation. In: 3D Geoinformation Science, Lecture Notes in Geoinformation and Cartography. Springer. pp 55-69
- Yeung, A. (1998). Information Organization and Data Structure. Ontario Ministry of Northern Development and Mines, Canada. Retrieved from: <http://www.ncgia.ucsb.edu/giscc/units/u051/u051.html>
- Wageningen UR (2015). Retrieved 11 October 2015, from: <http://www.wageningenur.nl/en/show/Bodemkaart-1-50-000.htm>
- W3.org (2015). Retrieved 10 September 2015, from: <http://www.w3.org/XML/>
- Wikipedia CSG (2015). Retrieved 27 August 2015, from: https://en.wikipedia.org/wiki/Constructive_solid_geometry
- Wikipedia halfspace (2015). Retrieved 25 August 2015, from: [https://en.wikipedia.org/wiki/Half-space_\(geometry\)](https://en.wikipedia.org/wiki/Half-space_(geometry))
- Zlatanova, S., Beetz, J., Boersma, A.J., Mulder, A., Goos, J. (2013). 3D spatial information infrastructure for the port of Rotterdam. ISPRS 2013
- Zlatanova, S., Stoter, J., Isikdag, U. (2012). Standards for Exchange and Storage of 3D Information: Challenges and Opportunities for Emergency Response. In: Proceedings of the 4th International Conference on Cartography & GIS – 2. International Cartographic Association. pp 17-28
- Zobl, F., Chimelina, K., Faber, R., Kooijman, J., Marschallinger, R., Stoter, J. (2011). Multidimensional aspects of GeoBIM: new standards needed. Salzburg: IAMG

APPENDICES

A: Python Conversion Script

Voxel_Point.py

Pseudo Code For-Loop

Source: ESRI / Martijn Meijers / Milo Janssen

```
import arcpy, os
voxels = open(r'D:\Vr31_07\klein.txt')
fc = r'D:\Vr31_07\First_TEM.shp'

if arcpy.Exists(fc):
    arcpy.Delete_management(fc)

arcpy.CreateFeatureclass_management(r'D:\Vr31_07', 'Tot_First_Vr.shp',
    'POINT', '', 'DISABLED', 'ENABLED')
arcpy.AddField_management(fc, 'TYPE', 'LONG')
#cur = arcpy.InsertCursor(fc)

nx=200
ny=250
nz=185

x11=100050
y11=400050
z11=-49.75

dx=100
dy=100
dz=0.5

xset=0
yset=0
zset=0

point3d=0
no_point3D=0

line_no = 0
for x in range(maxx):
    for y in range(maxy):
        for z in range(maxz):
            data = voxels[line_no].strip()
            line_no + 1

            print 'Plane ' + str(x) + ' of 200 written, ' + str(nodata) + ' NoData
            rows found, ' + str(data) + ' data rows found'
```


B: Groundwater surface points

```

import arcpy, os

fc = r'D:\Ma05_10\GWS_Ddam.shp'

if arcpy.Exists(fc):
    arcpy.Delete_management(fc)

arcpy.CreateFeatureclass_management(r'D:\Ma05_10', 'GWS_Ddam.shp',
'POINT', '', 'DISABLED', 'ENABLED')
arcpy.AddField_management(fc, 'TYPE', 'LONG')
cur = arcpy.InsertCursor(fc)

#point_1
row = cur.newRow()
pnt = arcpy.Point()
pnt.X = 107713
pnt.Y = 423374
pnt.Z = -1.74
row.SHAPE = pnt
cur.insertRow(row)

#point_2
row = cur.newRow()
pnt = arcpy.Point()
pnt.X = 107783
pnt.Y = 423092
pnt.Z = -1.47
row.SHAPE = pnt
cur.insertRow(row)

#point_3
row = cur.newRow()
pnt = arcpy.Point()
pnt.X = 108059
pnt.Y = 423153
pnt.Z = -1.84
row.SHAPE = pnt
cur.insertRow(row)

#point_4
row = cur.newRow()
pnt = arcpy.Point()
pnt.X = 108158
pnt.Y = 423412
pnt.Z = -1.42
row.SHAPE = pnt
cur.insertRow(row)

#point_5
row = cur.newRow()
pnt = arcpy.Point()
pnt.X = 107931
pnt.Y = 423367
pnt.Z = -1.91
row.SHAPE = pnt
cur.insertRow(row)

#point_6
row = cur.newRow()
pnt = arcpy.Point()
pnt.X = 107865
pnt.Y = 423221
pnt.Z = -1.74
row.SHAPE = pnt
cur.insertRow(row)

```

C: Union 3D

The aim of this case was the unify a 3D Multipatch Geometry together with a surface, which then served as a dangled surface. First, it was tried to module this situation in ArcGIS, but the available editing did not allow such a setup. Secondly, a COLLADA model was imported that represented a GeoTOP Multipatch with an attached surface via a shared edge. But peculiarly, once imported in ArcGIS the COLLADA model turned out to be shifted and of two separated volumes, not fit for editing. This case pointed out the difficulty of modelling in a GIS.

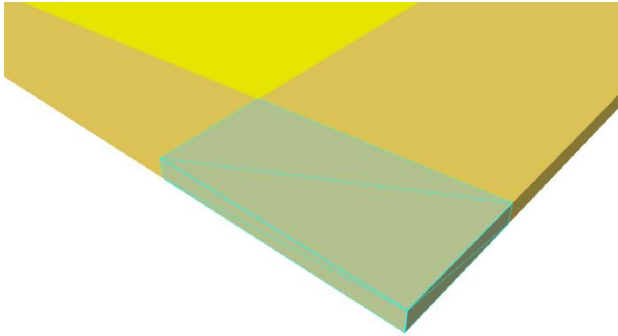


Figure A: the concerning GeoTOP Multipatch

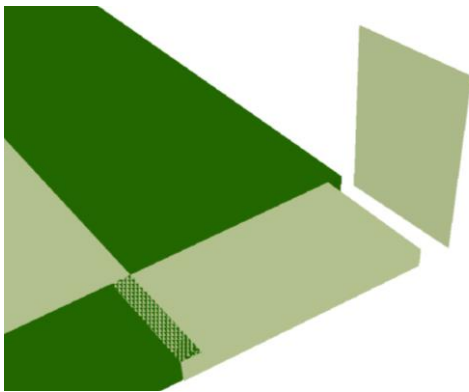


Figure B: the shifted and separated imported COLLADA model

A 3D Union operations could be performed successfully, with the original – unedited GeoTOP Multipatches as input:

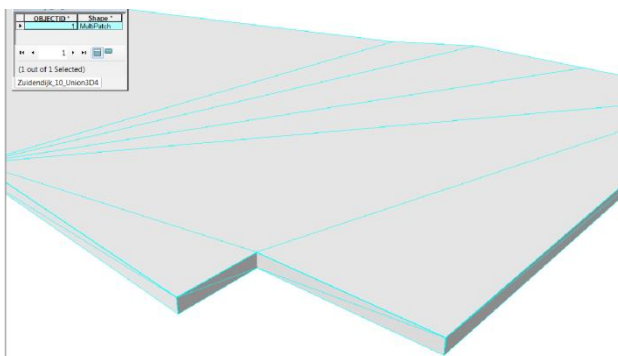


Figure C

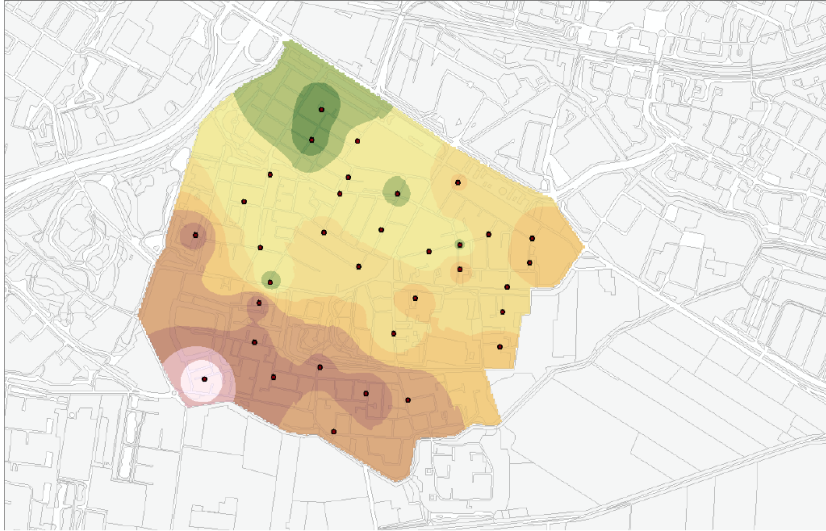
D: Data, GeoTOP voxel dataset

gridtype regular											
sorting +Z +Y +X											
nx 200											
ny 250											
nz 185											
xllcenter 100050.0											
yllcenter 400050.0											
zllcenter -49.75											
dx 100.0											
dy 100.0											
dz 0.5											
nodata_value 999.0											
lithostrat,lithoklasse,kans_1_veen,kans_2_klei,kans_3_kleilig_zand,kans_5_zand_fijn,kans_6_zand_matig_grof,kans_7_zand_grof,kans_8_grind											
int,int,float,float,float,float,float,float											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
999,999,999.0,999.0,999.0,999.0,999.0,999.0,999.0,999.0											
61,3,0.0,0.14,0.26,0.35,0.22,0.03,0.0											
61,3,0.01,0.23,0.36,0.21,0.17,0.02,0.0											
61,2,0.01,0.27,0.32,0.32,0.08,0.0,0.0											
61,2,0.0,0.28,0.32,0.26,0.13,0.01,0.0											
61,2,0.0,0.27,0.23,0.28,0.17,0.05,0.0											
61,3,0.0,0.18,0.32,0.35,0.11,0.04,0.0											
61,3,0.01,0.35,0.54,0.08,0.01,0.01,0.0											
61,3,0.01,0.22,0.47,0.18,0.08,0.04,0.0											
61,3,0.02,0.23,0.35,0.22,0.13,0.05,0.0											
61,3,0.02,0.3,0.48,0.12,0.07,0.01,0.0											
61,3,0.01,0.16,0.23,0.29,0.28,0.03,0.0											
61,3,0.01,0.19,0.3,0.29,0.19,0.02,0.0											
41,6,0.0,0.0,0.0,0.0,0.15,0.6,0.25,0.0											
41,6,0.0,0.0,0.0,0.0,0.13,0.6,0.27,0.0											
41,6,0.0,0.0,0.0,0.0,0.11,0.66,0.23,0.0											
41,6,0.0,0.0,0.0,0.0,0.1,0.73,0.17,0.0											
41,6,0.0,0.0,0.0,0.0,0.1,0.66,0.24,0.0											
41,6,0.0,0.0,0.0,0.0,0.11,0.65,0.24,0.0											
41,6,0.0,0.0,0.0,0.0,0.14,0.63,0.23,0.0											
41,6,0.0,0.0,0.0,0.0,0.11,0.66,0.23,0.0											
41,6,0.03,0.02,0.05,0.13,0.49,0.27,0.01											
41,6,0.02,0.03,0.05,0.12,0.52,0.25,0.01											
41,6,0.04,0.01,0.05,0.12,0.58,0.19,0.01											
41,6,0.01,0.01,0.08,0.1,0.59,0.2,0.01											
41,6,0.02,0.03,0.05,0.1,0.55,0.24,0.01											
41,6,0.0,0.0,0.0,0.0,0.13,0.57,0.3,0.0											
41,6,0.0,0.0,0.0,0.0,0.13,0.64,0.21,0.02											
41,6,0.0,0.0,0.0,0.0,0.13,0.58,0.28,0.01											
41,6,0.0,0.0,0.0,0.0,0.1,0.66,0.23,0.01											
41,6,0.0,0.0,0.0,0.0,0.09,0.62,0.26,0.03											
41,6,0.0,0.0,0.0,0.0,0.08,0.6,0.29,0.03											

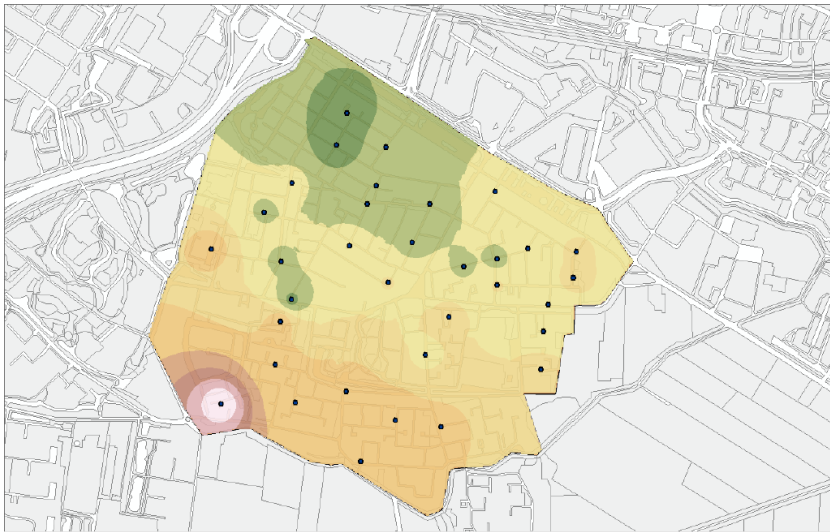
E: Other investigated areas/ interpolations

Based on the groundwater level data, interpolation can be made, here for the city district Dubbeldam in a winter and summer situation. Green surfaces represent groundwater levels on a relatively large depth, red indicate groundwater levels on a small depth. Higher groundwater levels (red/yellow) can mean a smaller unsaturated zone/drainage depth.

Interpolation 1 Summer situation:



Interpolation 2 Winter situation:



F: Photos, illustration of precipitation puddles accompanied with intersections

In the late summer of 2015, exceptionally heavy rainfall (T1000 shower) caused disturbances in the city district Sterrenburg. The park in this area, Sterrenburg Park, which is largely built up out of grass fields and trees, flooded because of the precipitation as can be seen in the photos below. 3D Intersections with GeoTOP, TOP10NL and accompanied by groundwater level surfaces, provided insights in the causes of the park flooding: the unsaturated zone of the park mainly consists out of clay layers, with right below layers of peat – all together a non-permeable soil combination.



Figure D: photos flooded Sterrenburg Park. Source: Schrandt, P. (2015) – dordrecht.net



Figure E: indication of the flooded park area

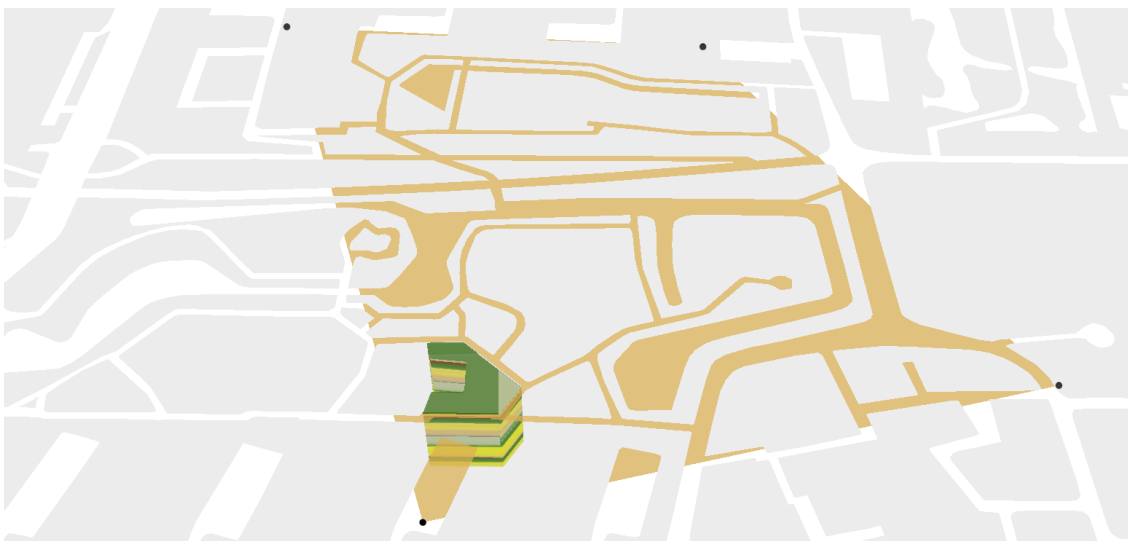


Figure F: groundwater level data as surfaces, 3D intersection with vectorised GeoTOP

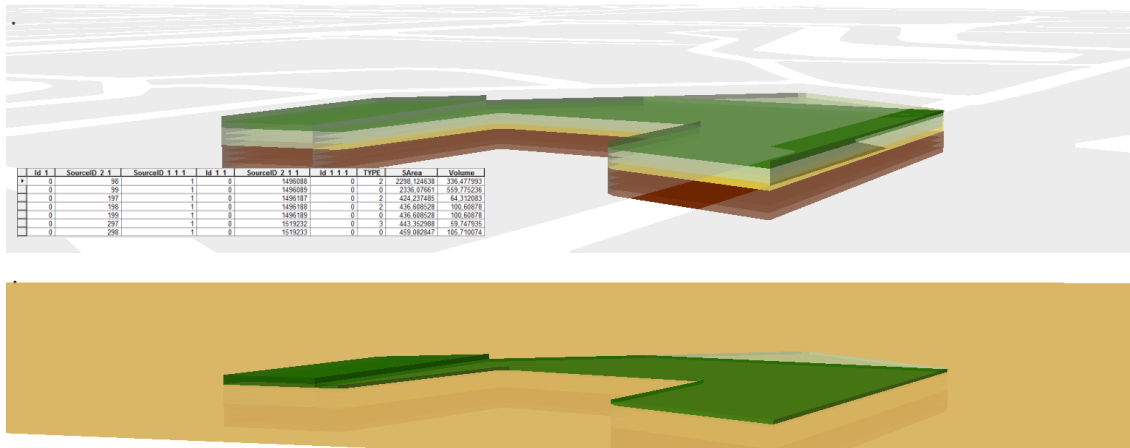


Figure F: the unsaturated zone and volume determination

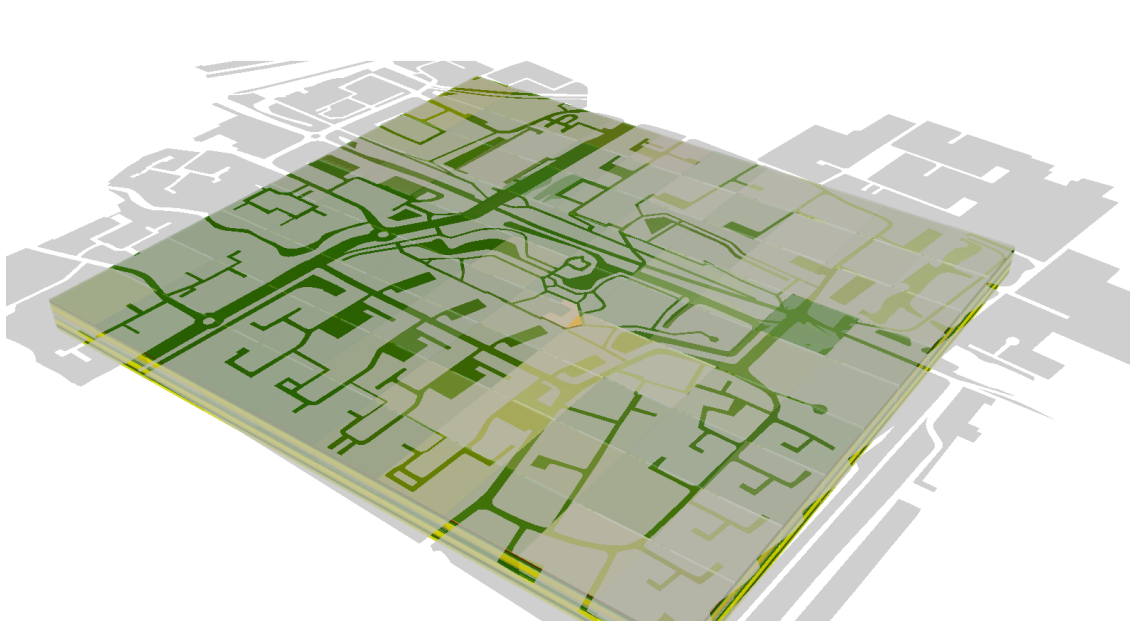


Figure D: the two main input datasets in 3D surface based vector: vectorised GeoTOP and TOP10NL

