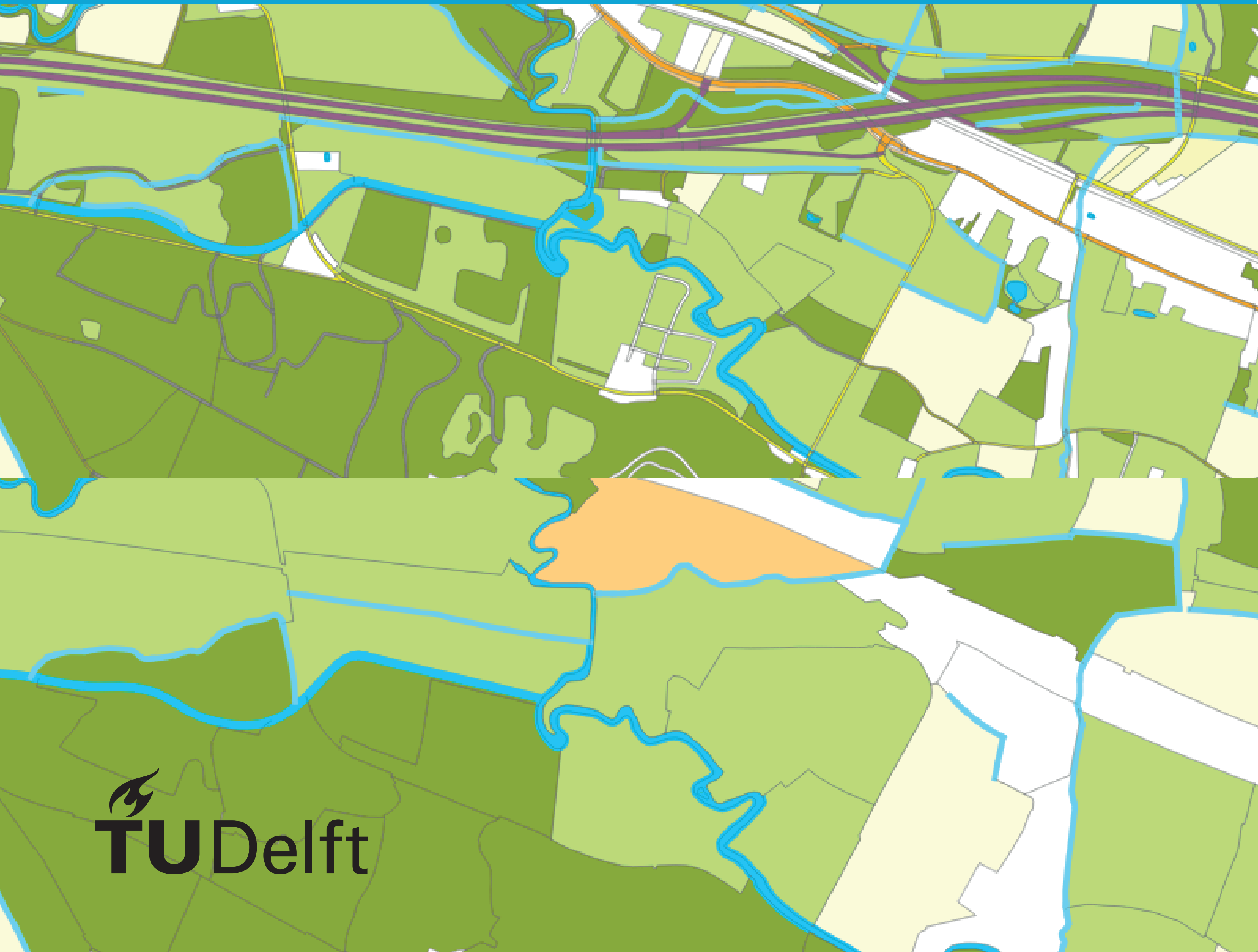
MSc thesis in Geomatics for the Built Environment

GENERALISATION OF HYDROGRAPHY NETWORKS

FOR A VARIO-SCALE BASEMAP

IJ.D.G. Groeneveld

2018



GENERALISATION OF HYDROGRAPHY NETWORKS FOR A VARIO-SCALE BASEMAP

A thesis submitted to the Delft University of Technology in partial fulfillment
of the requirements for the degree of

Master of Science in Geomatics for the Built Environment

by

IJ.D.G. Groeneveld

2018

IJ.D.G. Groeneveld: *Generalisation of Hydrography Networks for a Vario-scale Basemap* (2018)

An electronic version of this thesis is available at:
https://repository.tudelft.nl/

The work in this thesis was made in the:

Geo-Database Management Centre
Department of the OTB
Faculty of Architecture & the Built Environment
Delft University of Technology

Supervisors:	Prof.dr.ir. P.J.M. van Oosterom
	Dr.ir. B.M. Meijers
	Ing. R.Šuba
Co-reader:	Dr.ir. P. Nourian

ABSTRACT

The vario-scale concept has proved an alternative for the well known discrete multi-scale solution. However the vario-scale concept is still ongoing research and requires further development e.g. on how to generate better content for vario-scale maps? This research does focus on how to incorporate hydrography networks in the vario-scale solution and how to continuously generalise hydrography networks throughout all scales.

The pre-processing of topographic data is adjusted, information on hydrography features that are below another feature is stored and hydrography line features are added. The planar partition created in the pre-processing is used for the creation of the initial topological Generalised Area Partition (tGAP).

Hydrography graphs are created by iterating over all hydrography features in the data structure and storing the hydrography features as objects and the connection between two neighbouring hydrography features as links. Based on the links between objects the connected components, in this case the hydrography networks are created. There can be multiple hydrography networks in one data set, which are used for the decisions made in the generalisation process. The proposed improvements and decisions for hydrography features do result in meaningful hydrography networks throughout all scales and are an improvement to the vario-scale approach without these improvements and decisions.

ACKNOWLEDGEMENTS

I would like to take the opportunity to express my gratitude to the people who provided help and supported me during the MSc Geomatics Graduation period.

First of all, I would like to thanks Peter van Oosterom for being my main supervisor. Thanks for the helpful discussions and feedback.

Second, I would like to thank Martijn Meijers for all the guidance he provided.

Next, I would like to thank Radan Šuba for his help in Shenzhen, China when I was learning the vario-scale code. Thanks for all the discussions we had on vario-scale, which I really appreciated.

Furthermore, I would like to thanks Pirouz Nourian for being my co-reader and providing a lot of comments on the draft version of this thesis which really helped me to write the final version.

Finally, I would like to thanks my family for providing support. Also my housemates and friends, thanks for all the support and the coffee sessions we had.

Thank you all
IJsbrand Groeneveld

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

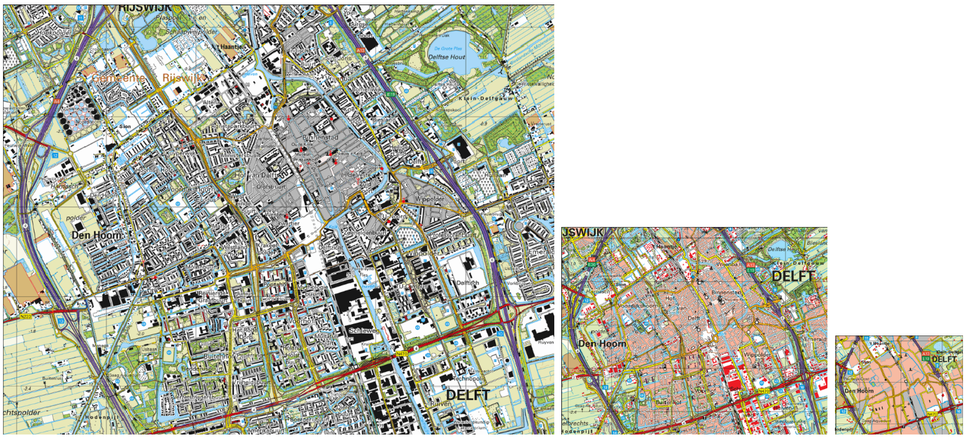
LoD	Level of Detail	2
tGAP	topological Generalised Area Partition	3
DEM	Digital Elevation Model	4
BLG	Binary Line Generalisation	9
DAG	Directed Acyclic Graph	9
SSC	Space-Scale Cube	10
SDK	Software Development Kit	11
BBOX	Bounding Box	31
SRTM	Shuttle Radar Topography Mission	35
LiDAR	Light Detection And Ranging	35
AHN3	Actueel Hoogtebestand Nederland	47

1 INTRODUCTION

Hydrography features like e.g. rivers and lakes have always been an important part of our physical environment and their importance is still growing with the expanding world population. Water from hydrography features is a basic natural resource and essential for humans and various human activities. As a result the banks of hydrography features have attracted humans since ancient times. The hydrography features are since used for e.g. irrigation, navigation and the transportation of goods and people. Due to the importance of hydrography features they have been a major component of maps created since ancient times. While ancient maps were published on paper most maps are nowadays published and transferred via the internet as digital maps on which users can perform e.g. zooming and panning operations. However the creation, retrieval and visualisation of maps more or less didn't change until van Oosterom [2005] introduced the concept of vario-scale. The focus of this research will be to investigate how to better incorporate hydrography features in the map creation needed for the vario-scale concept.

1.1 PAPER MAPS AND DIGITAL MAPS

Maps are representations of the world around us and are used as essential tools by humans and algorithms to help them e.g. to navigate. Nowadays maps are even more important because they are used in many fields like e.g. urban planning, transportation and resource management. All the different users want a map in a format and scale that is as close as possible to what they need. Therefore cartographers make maps in many different formats and scales ranging from large scales (very detailed) to small scales depending on the intended usage of the map, see Figure 1.1. The scale of a map is defined as the ratio of the distance on the map to the corresponding distance in the real world.



(a) 1:25k

(b) 1:50k

(c) 1:100k

Figure 1.1: Region of Delft for different map scales

When cartographers make maps they have to decide which features appear on the map as an abstraction of part of the world. This abstraction of part of the world requires exaggeration of features regarded as important and removal of features regarded as unimportant. This process is called generalisation and is applied when the scale of the map has to be reduced [Kraak, 2010]. It is important to notice that generalisation entails information loss, however it should be tried to preserve the essence of the contents of the original map [Kraak, 2010]. Generalisation involves the use of different generalisation operators/operations, e.g. the reduction of the amount of features in the map, simplification and selection. Simplification is applied to simplify the boundaries between features and selection is used to select features that are regarded as important for the intended use of the map.

Nowadays cartographers are making digital maps with the same techniques as they used for the creation of paper maps. These techniques involve the usage of a large scale topographic data sets and generalisation tools to create digital maps for different scales e.g. the input topographic data set has a scale of 1:1k and is used to construct maps with a scale of 1:10k, 1:25k, 1:50k, etc. These digital maps are mostly stored in a multi-scale database where each map is used for a specific scale interval, see Figure 1.2. The main drawback of this approach is that redundant data is stored and that only a fixed number of map scales i.e. Level of Detail (LoD) are available. Google Maps[1], Bing Maps[2] and OpenStreetMaps[3] all have around 20 zoom levels or LoDs available. Another consequence of this approach is that when a user is zooming in or out a new predefined map is retrieved from the server. Due to many fixed map scales a lot of information in the map is the same in successive map scales, however all information needs to be transferred to the user every time he requests another map scale. This results in duplicate data that has to be sent to the the user, affecting response time and the bandwidth needed for the requested map.

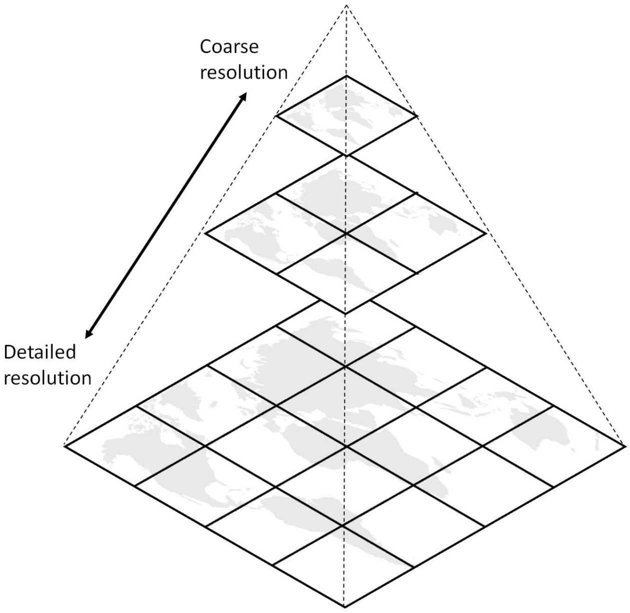


Figure 1.2: Concept of traditional tile pyramid (Scales or LoDs are fixed, no geometry between the layers in the pyramid), adjusted from Garca et al. [2012]

Maps also need to be maintained and updated which is labour intensive in the classical approach described above because there is no connection between the same feature appearing in maps at different scales. This lack of links between features in maps at different scales also make analysis, search operations and other processing more complex to perform.

There is however an alternative approach to these discrete/fixed map scales in the digital environment called variable-scale (vario-scale for short) which deals with the issues encountered in the classical approach.

[1] https://developers.google.com/maps/documentation/static-maps/intro#Zoomlevels

[2] https://msdn.microsoft.com/en-us/library/bb259689.aspx

[3] http://wiki.openstreetmap.org/wiki/Zoom_levels

1.2 VARIO-SCALE

Van Oosterom and Meijers [2013] introduced the concept of smooth topological Generalised Area Partition (tGAP) which is based on the concept of a tGAP data structure which is presented as a vario-scale data structure by van Oosterom [2005]. The tGAP structure starts with a planar partition at the most detailed level which corresponds to the largest available scale. The least important object/feature is selected and either merged with the most compatible neighbour or splitted and divided among the neighbours (importance of the neighbours and the length of their shared boundary can be taken into account). This process is repeated until one single object remains. The tGAP data structure contains all the results of the map generalisation operations, where feature by feature is generalised which is progressively leading to a more simple map. The result is that all the possible map scales are stored in a single vario-scale data structure.
The vario-scale approach is ongoing research and can work with area features and road networks which are added by Šuba et al. [2016]. Networks are considered a collection of features which can be represented as graphs which are mathematical structures used to model pairwise relations between objects. The real world objects like e.g. road features correspond to mathematical abstractions called vertices or nodes and the connection between a pair of related vertices is called an edge or a link. The edges of the graph may be undirected or directed, see Figure 1.3. A directed graph is often referred to as a Digraph. The networks considered in this research are Digraphs and when not specified the edge between the vertices is directed in both directions.

(a) Undirected Graph

(b) Digraph

Figure 1.3: A undirected graph and a directed graph with both 3 vertices and edges as connections between the vertices [Wikimedia Commons, 2007a] [Wikimedia Commons, 2007b]

In the vario-scale concept there is no special treatment for e.g. hydrography networks, utility networks and rail networks. As hydrography features are important for the navigability on a basemap extra research is needed to investigate the correct treatment of hydrography networks in the vario-scale solution while maintaining a meaning full hydrography network as long as possible throughout the generalisation process.

1.3 RESEARCH QUESTION

Research on the vario-scale concept has already been carried out for a few years by the GIS Technology Group at TU Delft. Publications, software and other information on the research and the concept can be found at `http://varioscale.bk.tudelft.nl`. This MSc Geomatics thesis aims to contribute to this ongoing research with the following main research question:

To what extent can hydrography networks be better incorporated in the vario-scale concept for creating a vario-scale basemap while maintaining the network structure?

The goal of this research is to study the possibility to incorporate hydrography networks in the vario-scale concept for creating a vario-scale basemap while maintaining the hydrography network structure. To achieve this goal and be able to answer the main research question the following sub-questions are relevant:

SUB 1: How to create a hydrography network based on hydrography features in the large scale topographic input data?

SUB 2: Water normally flows from areas with higher elevation to areas with lower elevations, except in some man-made hydrography networks. How to include the flow direction in the hydrography network and how does this influence the generalisation result? Are additional data like e.g. elevation needed?

SUB 3: How to implement the generalisation method for hydrography networks in the vario-scale concept? Which generalisation decisions need to be made in the process?

SUB 4: What are the differences in the generalisation results with the introduced treatment of hydrography networks compared to the version that doesn't have this functionality? How to assess the hydrography networks throughout the scales in the vario-scale approach?

1.4 SCOPE OF THE RESEARCH

The items in the following list define the scope of this research, what will be done and what will be delivered:

- The aim is to create a vario-scale basemap based on a large scale topographic input data set.
- The focus is on the hydrography network structures (Digraphs) and how they can be better included in the vario-scale concept so that the hydrography network structure is not torn apart during the generalisation process.
- The whole research will be carried out with 2D vector data only for all the features in the map. A Digital Elevation Model (DEM) will be used for determination of the flow direction.
- The main deliverable will be a report (MSc Thesis) describing the generalisation of hydrography networks for a vario-scale basemap. The code developed for the prototype to run this generalisation will be made available under an open source license.
- The generalisation method will be tested with real world topographic sample data and elevation data when needed.
- If the current tGAP data structure is not ready for the special treatment of hydrography networks, the structure will be enriched to be able to deal with the special treatment of hydrography networks.

It is also important to note what is not in the scope of this research:

- 3D data and the generalisation of 3D input data.
- Processing of large data sets.
- Labels are important for map readability but will not be addressed.
- Other networks like e.g. utility networks and rail networks will not be addressed, however could partially be dealt with in the same way.

- The temporal aspect of hydrography features will not be addressed as the research does focus on a vario-scale basemap.
- Map updates are important but will not be addressed in this research.

1.5 THESIS OUTLINE

This thesis document is structured in the following way:

- Chapter 2 provides the THEORETICAL BACKGROUND AND RELATED WORK. The tGAP concept is explained and the implementation of road networks is described in detail as this will be used as basis for the generalisation of hydrography networks.
- Chapter 3 explains the methodology used and the development of the special treatment of hydrography features in the vario-scale concept.
- Chapter 4 deals with the pre-processing of the data, the flow direction in the hydrography network and the creation of a hydrography network itself in the initial tGAP data structure.
- Chapter 5 shows the results of the implemented special treatment for hydrography networks in the vario-scale concept.
- Chapter 6 concludes the thesis with the conclusions drawn from the research. Also relevant future work is suggested.

2 THEORETICAL BACKGROUND AND RELATED WORK

The THEORETICAL BACKGROUND AND RELATED WORK chapter aims to provide the relevant theoretical knowledge on the vario-scale concept and the work than by others that relates to the topic of this research. More specific, Section 2.1 describes the tGAP concept and structure. Section 2.2 deals with the implementation of road networks in the vario-scale concept developed and implemented by Šuba et al. [2016]. The last section, Section 2.3 briefly describes some other related work.

2.1 THE TGAP BACKGROUND

Van Oosterom [2005] introduced the concept of tGAP and presented it as a vario-scale data structure. The tGAP structure starts with a planar partition at the largest available scale which contains the most details. At the moment only features that can be seen from above are used for the construction of the planar partition. First, the importance function (Equation 2.1) is used to determine which feature is least important based on its size (Area) and the relative importance of the feature class it belongs to (WeightClass). The WeightClass is determined by the user, e.g. if forest areas are more important than cornfields according to the user, the WeightClass of the forest area will have a higher relative importance value.

$$\mathcal{I}mportance(a) = Area(a) * WeigthClass(a) \tag{2.1}$$

Second, neighbour b is selected based on the highest value of the collapse function (Equation 2.2) with Length(a,b) being the length of the common boundary between feature a and feature b. The CompatibleClass(a,b) is taken from a compatibility matrix created by the user before the start of the generalisation process. This compatibility matrix indicates how well different feature classes can be merged with each other.

$$\mathcal{C}ollapse(a,b) = Length(a,b) * CompatibleClass(a,b) \tag{2.2}$$

The least important feature is merged with the most compatible neighbour found with the collapse function (Equation 2.2). This process is repeated until one single feature remains, see Figure 2.2. The merging of the features is recorded in the tGAP face tree structure and the last remaining feature is the top of the tree, see Figure 2.1.

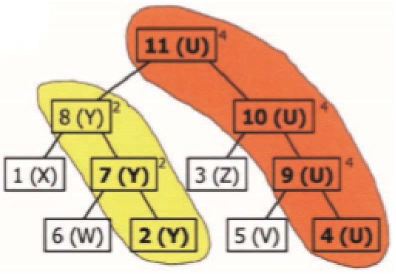


Figure 2.1: tGAP-face tree, a new object id whenever a face changes in the generalisation process (the old object id is at the upper right of a node in the tree). The feature class is shown in brackets after the object id. Adjusted from van Oosterom [2005]

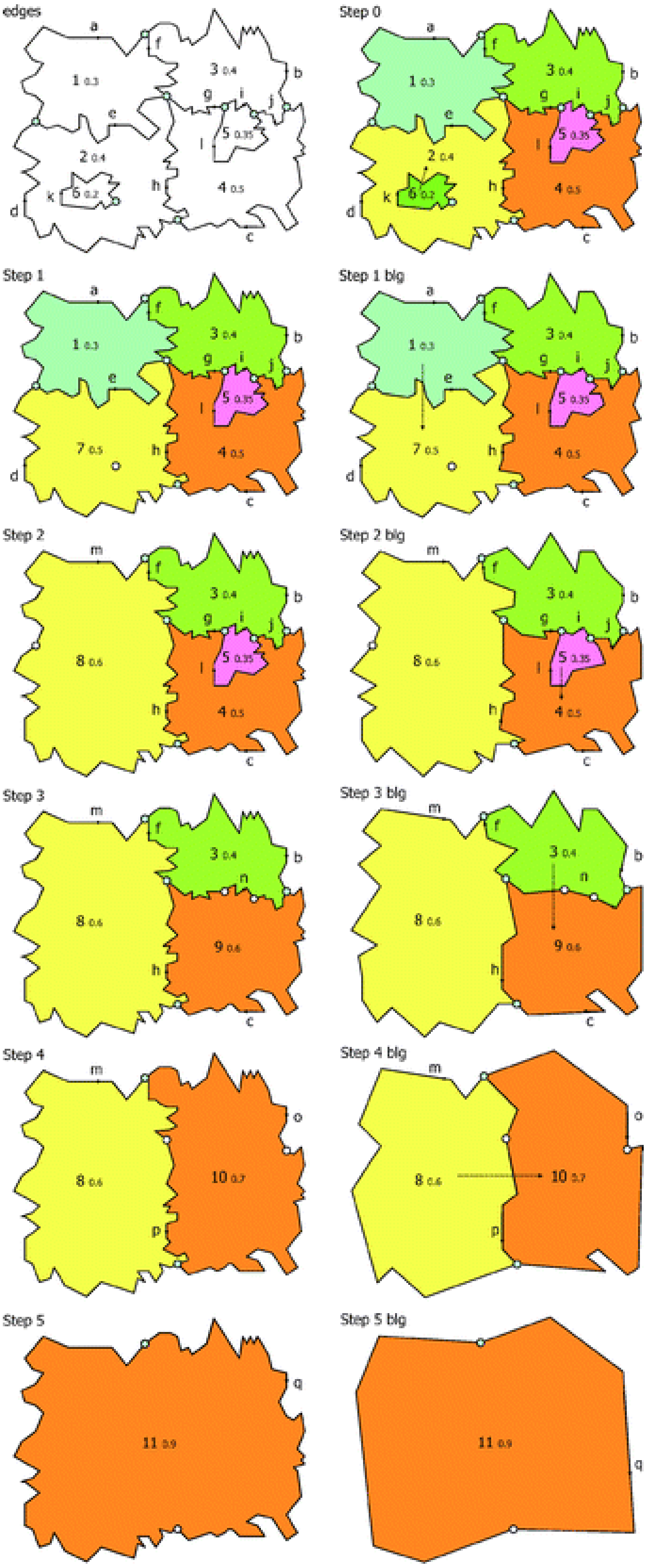


Figure 2.2: Generalisation example shown in five steps. First for every step a merge operation is performed. Second the boundaries are simplified via the BLG tree. Note that nodes are shown in green and that nodes that are removed are shown in white for the next generalisation step. The computed importance value is shown in a smaller font next to the face id. [van Oosterom, 2005]

The nodes in the tGAP-face tree contain no explicit geometry, only the topological face information is stored. The stored edges do contain geometry and have topological references to the faces left and right as proposed by Vermeij et al. [2003]. This makes that the faces can refer to the edge(s) which form the boundary of the face. During the generalisation process when two faces are merged three different things can happen to the edge(s) involved:

1. An edge is removed; e.g. edge *e* in step 2 of Figure 2.2.

2. Two or three edges are merged to form a new edge; e.g. edges *a* and *d* are merged to form edge *m*, see Figure 2.2.

3. The references of the edge are changed; e.g. the reference to the right face of edge *h* changes from face *7* to face *8* in step 2 of Figure 2.2.

The edges at the start of the generalisation process contain a lot of detail (lot of points which form a polyline) therefore the edges are also simplified during the generalisation process. The Binary Line Generalisation (BLG) tree is a data structure used for line generalisation and is well suited for polylines, continuous in detail level, and can be implemented with a simple binary tree [van Oosterom and van den Bos, 1989]. The algorithm to create a BLG tree is based on the Douglas-Peucker algorithm [Douglas and Peucker, 1973]. Figure 2.3 shows the resulting BLG tree for the edges of the scene in Figure 2.2. The BLG tree is traversed in order to produce the appropriate level of detail which depends on the requested tolerance value.

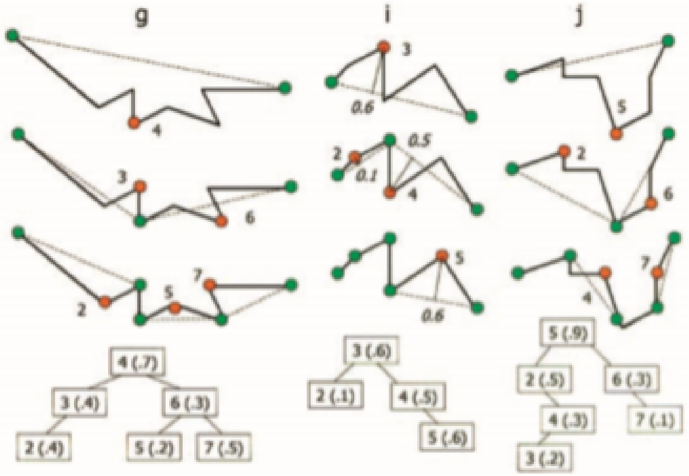


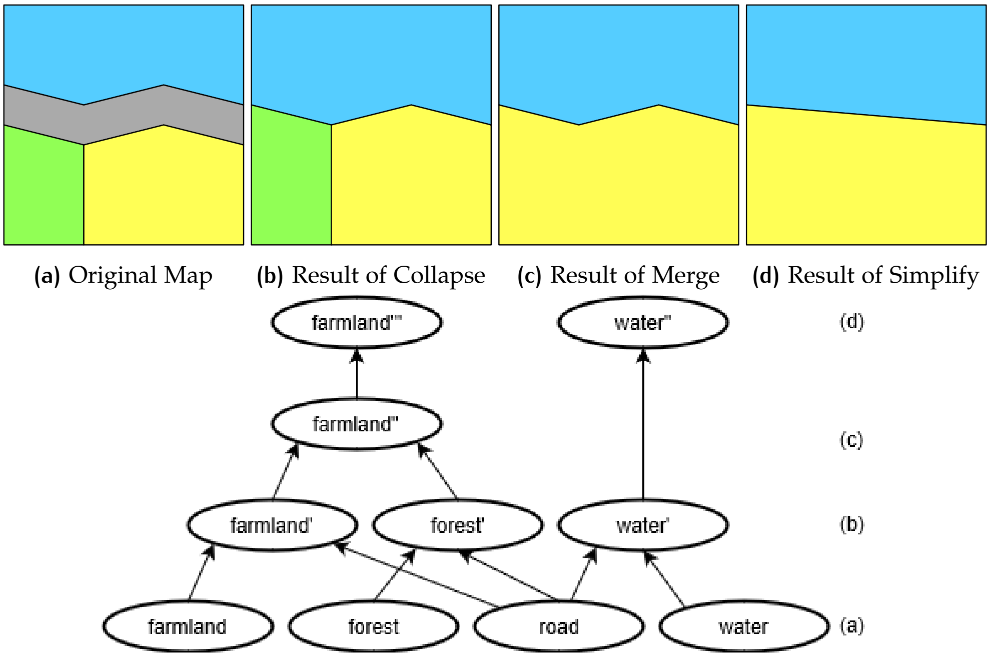
Figure 2.3: Three examples of the BLG trees for the edges *g*, *i* and *j* of the scene in Figure 2.2. Every node in the BLG tree contains a point (number) and a tolerance value (in brackets). [van Oosterom, 2005]

Ai and van Oosterom [2002] noted that merging the least important object with just the most compatible neighbour is not always the best option and that it may result in a sub-optimal map representation. Therefore the split operation is introduced which divides the feature along its skeleton and assigns the different parts to different neighbours. This improvement made that the tGAP tree changed from an hierarchical tree structure to a tGAP Directed Acyclic Graph (DAG) as the split operation causes objects to have several parents. Together with the BLG tree it is called the tGAP structure. The result of the generalisation process is the tGAP structure, which can be used to select a representation at any requested LoD or scale. Figure 2.4 shows four map fragments and the corresponding tGAP structure in which the following generalisation operations have been applied:

1. Collapse of the road feature from area to line, former road area is split and assigned to the neighbours.

2. Merge of the forest feature with the most compatible neighbour, in this case farmland.
3. Simplification of the boundary between the farmland and the water feature.

In the current implementation of the generalisation process the simplification operation and the merge/split operation are combined in one generalisation step. However for making the principle clear these operations are illustrated separately in Figure 2.4.

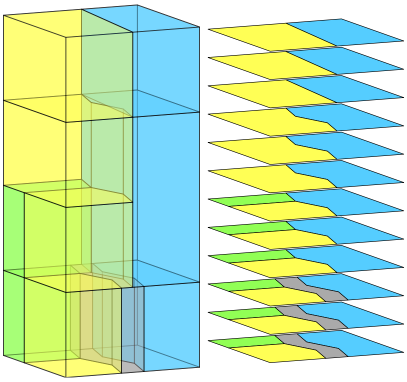


(a) Original Map **(b)** Result of Collapse **(c)** Result of Merge **(d)** Result of Simplify

(e) Corresponding tGAP structure

Figure 2.4: Four map fragments and the corresponding tGAP structure, adjusted from van Oosterom and Meijers [2013]

Vermeij et al. [2003] introduced the idea that map generalisation of 2D polygonal regions can be seen as extrusion into the third dimension. Meijers and van Oosterom [2011] used this idea for the tGAP structure where the scale is depicted as the third dimension in the integrated Space-Scale Cube (SSC) representation, see Figure 2.5a. A map can be obtained from the SSC by a horizontal slice, see Figure 2.5b



(a) SSC **(b)** Slices

Figure 2.5: SSC for the classic tGAP structure and slices from the SSC. [van Oosterom and Meijers, 2013]

2.2 ROAD NETWORKS IN THE VARIO-SCALE CONCEPT

Roads features and similarly hydrography features are the so called backbone of many maps. They improve legibility and navigation on the map for the users. Besides this road features are also used by algorithms like e.g. those of Google and TomTom. These algorithms are using shortest path algorithms like e.g. Dijkstra, A*, etc. all of which require network/graph data structures.
The current vario-scale concept however can handle only polygon features, therefore Šuba et al. [2016] did research on how to implement road line features in the vario-scale concept which are at the largest scale represented as polygon features while at the mid and small scale they are represented by line features. So throughout the generalisation process the polygon features representing roads are changing to line features representing the road features.
The processing strategy for the continuous generalisation of road networks throughout all scales developed by Šuba et al. [2016] is described in Section 2.2.2. Concepts and definitions needed for this are first explained in Section 2.2.1.

2.2.1 Concepts and Definitions

First two theoretical concepts will be explained which will be used in the generalisation process.

Granularity

The continuous generalisation requires geometric changes between successive steps in the process. Šuba et al. [2016] call the number of features that are changing in one generalisation step the granularity and distinguishes the following levels of granularity:

- coarse granularity: all features are processed at once (one step) in the generalisation process, e.g. all roads are removed.
- medium granularity: all features of a certain feature class or subclass are processed together, e.g. all local roads with a certain speed limit.
- fine granularity: a single feature is processed, e.g. one dead-end road.
- finest granularity: a part of a single feature is processed, e.g. a segment of a certain road.

The finest granularity is optimal for vario-scale as this guarantees that the changes between successive generalisation steps are as small as possible, this complies very well with the vario-scale concept.
In the generalisation process merge or split and simplification operations are performed, these operations can have as a result that a road object is composed of different segments which can be represented by a mix of lines and polygons, see Figure 2.6. From the traditional cartographic point of view this might seem less favourable, however from a vario-scale point of view this is good. When visualising the data the difference in representation can be made less visible by applying the proper styling to the line features. The history of the generalisation steps is stored explicitly, which means that original objects and generalised objects are linked. These links are often missing in a multi-scale implementation.

Level of Abstraction

The input data sets currently supported by the tGAP structure are modelled as 2D polygonal data structures, i.e. as a partition of a plane without gaps and overlaps, a so called clean planar partition. The vario-scale Software Development Kit (SDK)

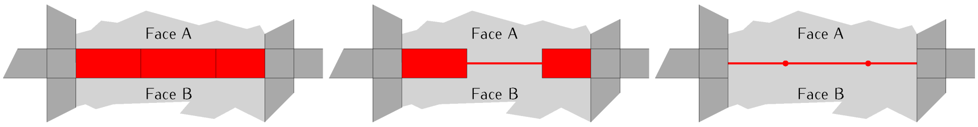
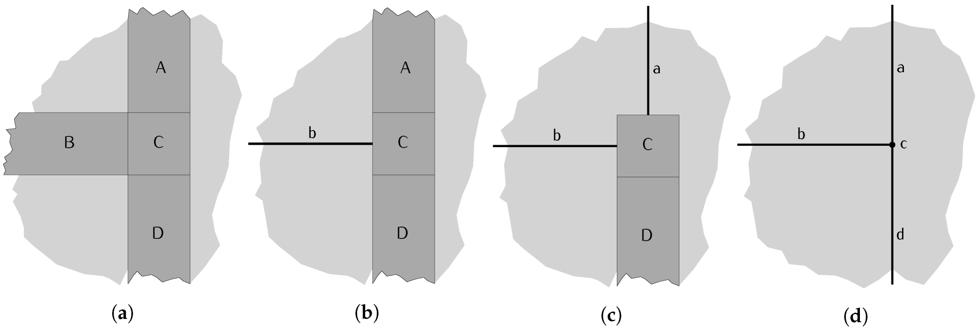


Figure 2.6: Gradual transition from one scale to another. The red road consists of three different parts. In the generalisation process each part is generalised separately. First the representation changes form area to semi-linear and to linear at the final scale. During the generalisation process the road is represented both by area and line parts. Note that also topological change is taking place: Face A and Face B become adjacent. [Šuba et al., 2016]

used for the generalisation process which is filling the tGAP data structure can handle only clean planar partitions. As a result, the tGAP data structures only contain the topological primitives vertexes, edges, and faces. One polygon object in the input data set corresponds to just one topological face. It is important to note that the same is true for all the road features in the input data set, they are all represented by faces.

Beside the feature class of the polygons in the planar partition more information or semantic information is implicitly presented in the large scale input data set e.g. the road and hydrography features in the planar partition can be networks however these networks are not explicitly modelled. Šuba et al. [2016] wish to preserve the natural meaning of the road network(s) in the target map at a small scale, even tough the features that are part of a network are not explicitly modelled. Therefore Šuba et al. [2016] aims at making this implicit information on the role a road feature plays in the road network explicit also when this information in not present/modelled in the input data set.

Figure 2.7 shows an example of an input data set which contains some road features which form a network (see Figure 2.7a) and the target after the generalisation process (see Figure 2.7d). If two road segments which are both represented as faces share an edge, than they are neighbours and connected. When at least one road feature is not represented as a face than they are connected when they share at least a vertex. It is not always simple to keep track of the road network as the map changes during the generalisation process from scale to scale in a gradual way, see the intermediate generalisation steps in Figure 2.7b,c. Figure 2.8 shows the constant network graph and the relations between the road objects of the scene in Figure 2.7.



(a) (b) (c) (d)

Figure 2.7: Generalisation of road network from the large scale input data (a), trough the intermediate steps in the process (b,c), to the final scale (d). Note that the representation of the road segments changes from 2D to 1D, and the representation of the junction changes from 2D to 0D. The semantics and the role of the features in the networks stay the same. [Šuba et al., 2016]

Road segments can be incident with other road segments. Depending of the number of incident road segments a certain road segment can be a junction (node) or connection (edge) in the linear road network graph. As can be seen in Figure 2.7

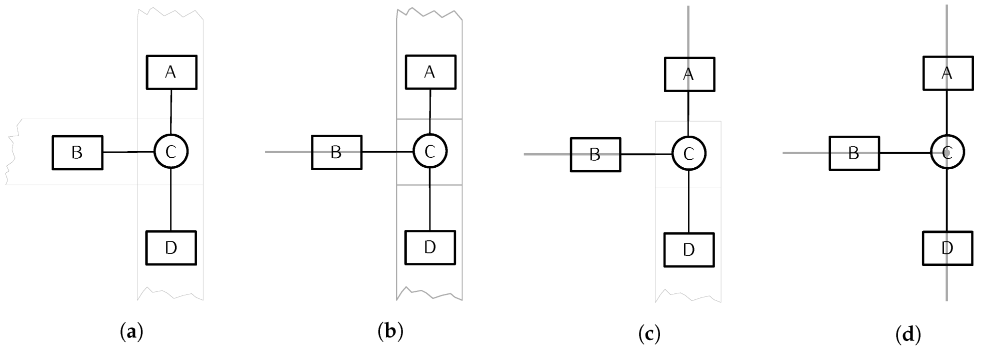
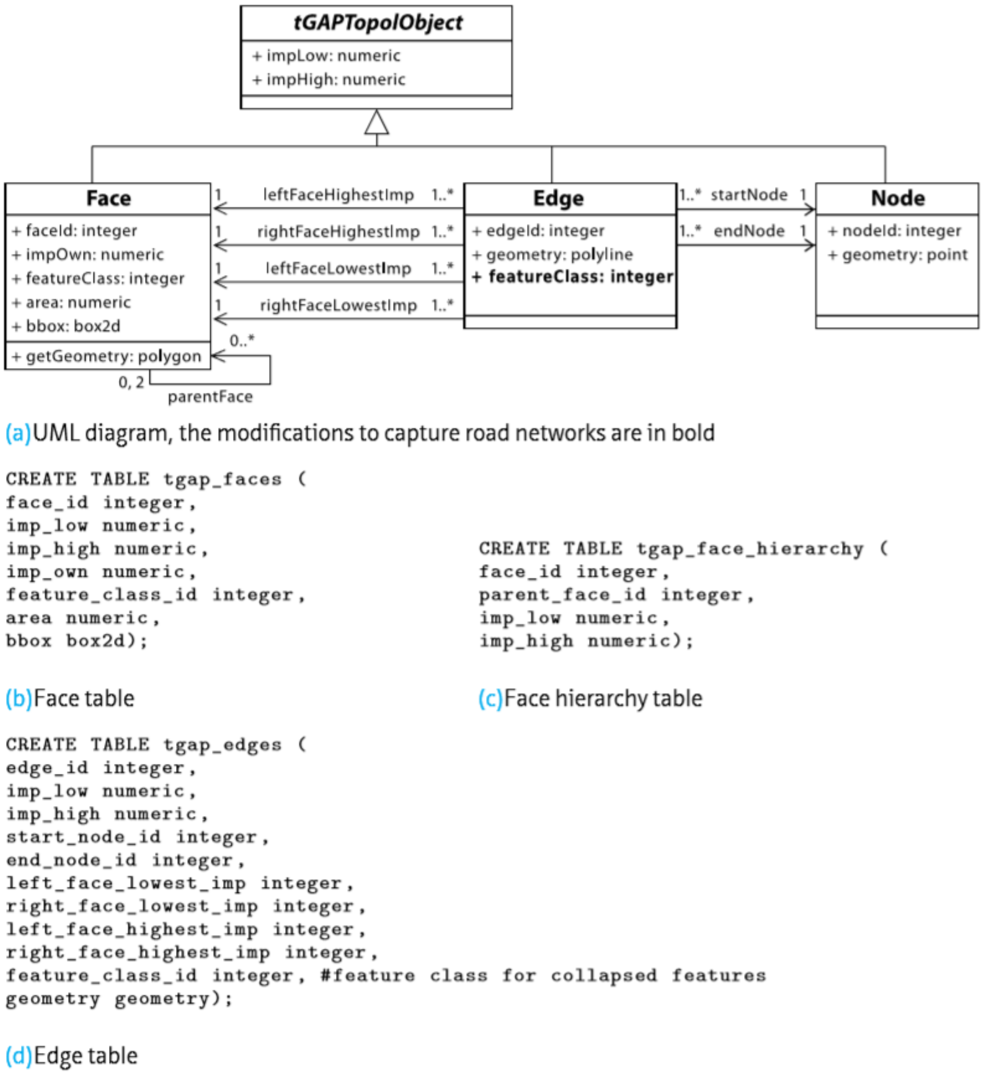


Figure 2.8: Linear network graph of the road network in Figure 2.7. It shows the topological relations of the road objects, a rectangle indicates a road connection and a circle indicates a road junction. [Šuba et al., 2016]

the objects in the map gradually change during the generalisation process while the linear network graph stays the same. This is considered an effective tool which can be used for the meaning full generalisation of road networks throughout the scales.



(a) UML diagram, the modifications to capture road networks are in bold

```
CREATE TABLE tgap_faces (
face_id integer,
imp_low numeric,
imp_high numeric,
imp_own numeric,
feature_class_id integer,
area numeric,
bbox box2d);
```

(b) Face table

```
CREATE TABLE tgap_face_hierarchy (
face_id integer,
parent_face_id integer,
imp_low numeric,
imp_high numeric);
```

(c) Face hierarchy table

```
CREATE TABLE tgap_edges (
edge_id integer,
imp_low numeric,
imp_high numeric,
start_node_id integer,
end_node_id integer,
left_face_lowest_imp integer,
right_face_lowest_imp integer,
left_face_highest_imp integer,
right_face_highest_imp integer,
feature_class_id integer, #feature class for collapsed features
geometry geometry);
```

(d) Edge table

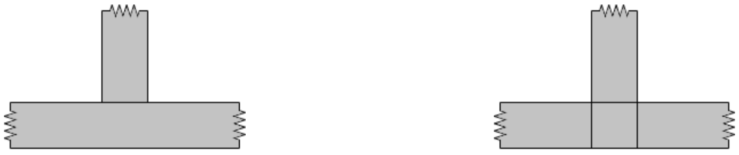
Figure 2.9: UML diagram and the database tables for tGAP. [Šuba, 2017]

In the current database tables needed for tGAP no feature class information about an edge is stored, therefore Šuba et al. [2016] adjusted the structure, see Figure 2.9,

to be able to store feature class information of the edge which is needed for the described generalisation of road networks throughout all scales.
Šuba et al. [2016] classifies the road segments based on the number of other incident roads and includes this in the data structure, they make the following classification:

- isolated segment: no other road segments are incident.
- dead end: road segments has exactly one other incident road segment. It is represented by a face or an edge in the topological data structure.
- connection: road segment is incident with exactly two road segments. It is represented by a face or an edge in the data structure.
- junction: the road segment is incident with more than two other road segments. It is represented with a face or a node in the topological data structure.

With this classification it is possible to define the role of the road feature in the map at any stage in the generalisation process. It is assumed that the input data contains well defined connections and junctions as illustrated in Figure 2.10b, if this is not the case as illustrated in Figure 2.10a an additional pre-processing steps needs to be taken. This can be done by applying the constrained Delaunay triangulation to obtain properly classified road segments, as proposed by Uitermark et al. [1999].



(a) No defined road junction

(b) Defined road connections and junction

Figure 2.10: Illustration of road features with and without well defined road connections and junctions

2.2.2 Processing Strategy for Roads

The generalisation process to generate the content for the vario-scale structure is based on the tGAP principle which Šuba et al. [2016] extended using linear network knowledge which is applied to roads. They used the following design decisions in the development process:

- Only three object classes are used for the creation of the content for the vario-scale structure, the object classes are: roads, water and other. In the process roads will be sub classified to either connection or junction. This limited amount of object classes make the decisions that have to be made during the generalisation process more transparent.
- At the start of the generalisation process every face in the input data set gets an importance value. The face with the lowest importance value will be processed first and the decision on the operations performed on the face are based on the feature type of the selected face.
- If the selected face is a road junction it will be either merged with an adjacent road junction or it will be preserved until all road connections to the road junction are collapsed, see Figure 2.11. If all incident road connections are collapsed that road junction itself can also be collapsed. When not all incident road connections are collapsed the importance value of the road junction is increased and the face is put back in the queue of the faces that have to be processed. Note that this could cause an infinite loop, which Šuba et al. [2016] prevents using additional measures such as queue reordering.

- If the selected face is a road connection it will be merged with an adjacent road connection. If there is no adjacent road connection the face will be collapsed, see Figure 2.11.
- If the selected face is water it will be merged with an adjacent water feature. If there is no adjacent water feature the face will be collapsed, see Figure 2.11.
- If the selected face is of the type other it will be merged with an adjacent other type feature. If there is no adjacent feature of the type other the face is collapsed, see Figure 2.11. Other features with no collapsed road between them are the most optimal to be merged with, if there is an collapsed road between them adjacent other feature with the least important road between will be chosen for the merge and the collapsed road between them will be removed.

These decisions make sure that the generalisation of road networks takes place in a meaningful way and is continuous for all faces in the generalisation process. Šuba et al. [2016] concluded after testing that the developed strategy which creates road line features in the generalisation process works in the vario-scale concept, and gives reasonable results.

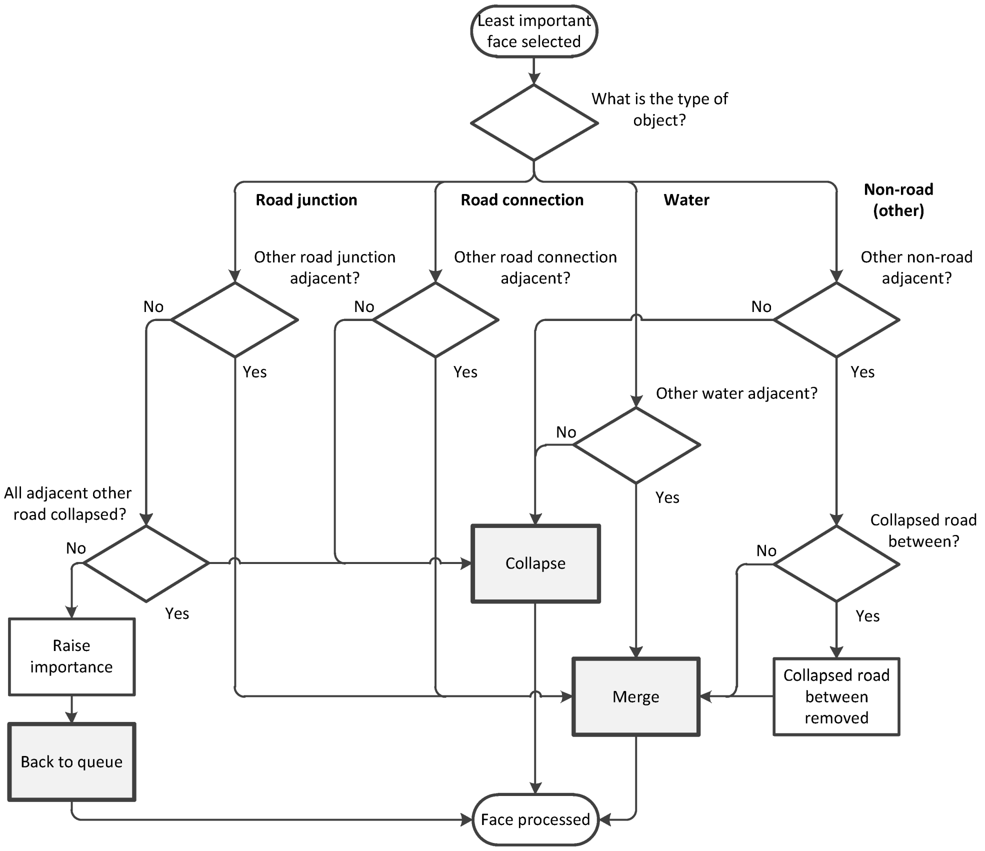

Least important face selected
What is the type of object?
Road junction
Road connection
Water
Non-road (other)
Other road junction adjacent?
Other road connection adjacent?
Other non-road adjacent?
No
Yes
Other water adjacent?
All adjacent other road collapsed?
Collapsed road between?
Collapse
Raise importance
Back to queue
Merge
Collapsed road between removed
Face processed

Figure 2.11: Processing strategy for roads in the vario-scale concept. [Šuba et al., 2016]

2.2.3 Road Networks versus Hydrography Networks

The described method of Šuba et al. [2016] introduced line features in the vario-scale concept and applied it to road networks. In the developed method the advanced treatment for e.g. hydrography networks, utility networks and rail networks could be included, however it might be the case that additional knowledge or a different treatment is needed for these networks during the generalisation process. The following differences can be noted between hydrography networks and road networks:

- Water in natural hydrography networks is always flowing towards lower elevations e.g. a river flowing from the mountains to the sea. The direction of the water flow could possibly be incorporated in the generalisation process to obtain a better generalisation of the hydrography network so that the network structure in the result is meaningful.
- The shapes of the features in the hydrography network are a mixture between man-made and natural phenomena while roads are man-made features which tend to be more straight compared to hydrography features which tend to follow the path of least resistance.
- There can be area's like lakes in a hydrography network, which is not the case in road networks. This makes that there are different topological relationships between the hydrography features which need to be preserved in the generalisation process.

2.3 RELATED WORK ON HYDROGRAPHY

Hydrography features as network objects are together with road networks considered as the backbone of many map types. They help users with orientation and recognition of real world objects, besides this they improve the overall legibility of the maps. Two main representations of hydrography features exists: linear and area. In the past different methods have been described for the generalisation of hydrography networks represented by linear and area features.

McAllister and Snoeyink [2000] describe the use of the medial axis of a polygon (described by the left and right banks) to automatically generate river centerlines and to derive river attributes. They also did experiments to approximate the medial axis by a Voronoi diagram and computed the approximation through a robust implementation of the Voronoi diagram. The result is an approximation of the medial axis of the river network which can be used for further analysis.

Gold and Snoeyink [2001] adjusted the crust algorithm of Amenta et al. [1998] to extract the skeleton from unlabelled vertices. They found that by applying the algorithm as a local test on the Voronoi diagram the crust and the skeleton can be found.

Haunert and Sester [2007] describe a generalization method based on straight skeletons to obtain the centerlines of the considered features. The construction of the straight skeleton is based on the step wise shrinking process of the polygon which can be performed by simultaneous parallel offsets of the polygon edges until the skeleton remains. They also describe how this method can be used to partially collapse a feature while preserving the topological relationship e.g. the connection between a river and a lake.

Strahler [1952] developed a method to classify the branches of a river network through using a counter which increases when two branches with the same number meet in case two branches with different numbers meet the section will get the same number as the highest of the two branches, see Figure 2.12. This method is known as the Strahler order and is widely used as enrichment method for river networks. The Horton order can be constructed after the Strahler order is known. For the Horton order, the highest order N corresponds to the main stream, the N-1 order to the second most important stream, and so on [Horton, 1945].

Savino et al. [2011] describes a method that uses the Strahler order, the width of the rivers, the flow direction, the longest distance in the network, the length of the river, the number of branches upstream and the density of the network for

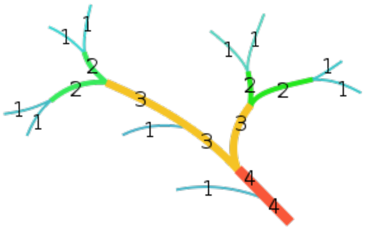


Figure 2.12: Strahler stream order [Wikimedia Commons, 2011]

the generalization of river networks. They describe a method to calculate the flow direction of the rivers based on the z-coordinate which is extracted from a DEM.
Regnauld and Mackaness [2006] describe a method to automatically create a topologically connected hydrography network from hydrography features 'broken' into parts by features such as e.g. bridges. They join hydrography features based on rules of continuity, proximity and the flow direction. For the construction of the flow direction they used an underlying DEM.
Ai et al. [2006] consider the order (Strahler and Horton), the length, the distribution pattern and other parameters such as distribution density and distance between proximity channels. They come to the conclusion that the for generalization needed geometric parameters are easy to compute, while the geographic parameter (consisting of watershed information) is usually difficult to compute because it requires a complex model to get useful information for the generalization process.
Van Altena and Stoter [2016] studied if the automatic generalization of man-made water networks can be improved by pruning based on the landscape. They showed that it is possible to improve the generalization of man-made hydrography networks after taking into account the landscape type.
For the generalization of Dutch municipal data (1:1k) to a 1:10k map van Altena et al. [2014] describe which steps need to be carried out to automatically generalise hydrography features.
Hydrography network generalisation has mainly been used to generalise a map from a fixed large scale to another fixed smaller target scale. Ai et al. [2017] however follows the idea of vario-scale but in a different way than the approach of van Oosterom [2005]. They build a matrix model to store the LoDs from multiple generalisations in a hybrid hierarchical structure which allows vario-scale representations of hydrography networks over a wide range of scales. In the matrix structure, generalization processes of hydrography features accompanied with the geometric smoothing of the hydrography features are hierarchically constructed as the row and columns which include an explicit scale range [Ai et al., 2017]. This means that the generalization and simplification are done separately, while in the vario-scale approach based on the tGAP both operations are combined.
Meijers et al. [2016] describe a algorithm called SplitArea which can be used to compute linear representations and at the same time split the old area of the feature and assign it in a 'fair' way to neighbours based on the compatibility of these neighbours. This algorithm can be used to change the representation of hydrography area features (e.g. wide rivers at large scale map) into hydrography line features (for smaller scale maps).

3 METHODOLOGY AND DEVELOPMENT

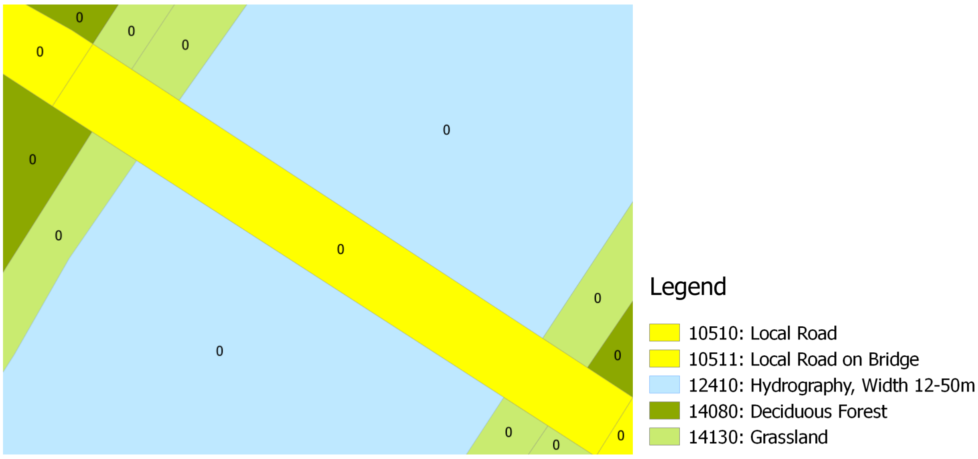
Chapter 2 described the concept of the vario-scale structure and the research that has been done so far. The main aim of Chapter 3 is to describe what has been done to be able to answer the Research Question which will help to improve the vario-scale concept when implemented. This chapter will described from a theoretical point of view, details on the implementation can be found in the Chapter 4.

3.1 HYDROGRAPHY FEATURES IN THE PLANAR PARTITION

Hydrography features can be represented by polygon and/or line features in a data set. The subsections respectively describe these representation and their inclusion in the planar partition which is used as input for the tGAP data structure.

3.1.1 Hydrography Polygon Features

The creation of the clean planar partition from a large scale topographic data set until now only involves polygon features (see Chapter 2) seen from above. To differentiate between features that are on top of each other in the topographic data set height levels are used. Feature seen from above have a height level of zero, while feature that are directly beneath (and thus can't be seen from above) have a height level of minus one. Figure 3.1a shows features in a topographic data set with a height level of zero, while Figure 3.2a shows the features with height level minus one that are in this case beneath a bridge.

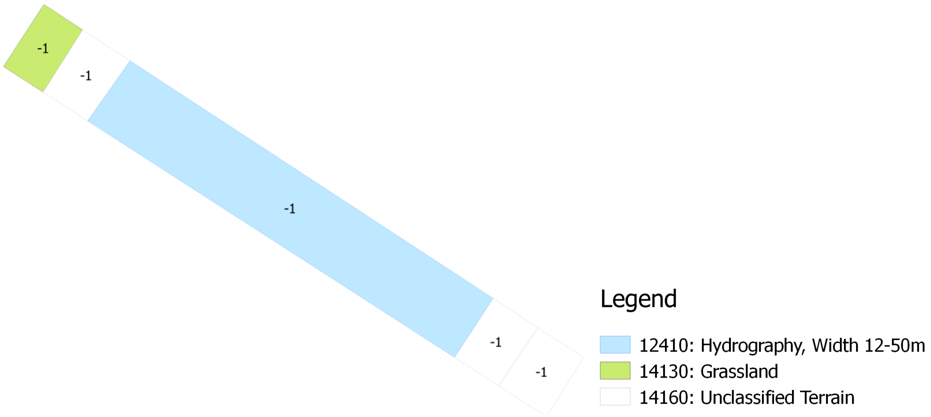


(a) Features with height level zero

(b) Legend

Figure 3.1: Features with height level zero in a topographic data set.

Figure 3.3a shows a representation of the bridge in Figure 3.1 and the information about the features that is stored and used in the construction of the tGAP data structure can be found in Table 3.1. However, the hydrography features that are directly below another feature are crucial features for the construction of a hydrography network. Therefore this information needs to be stored in the pre-processing so that it



(a) Features with height level minus one

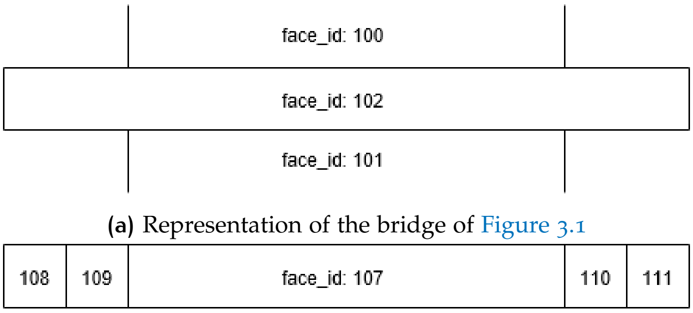
(b) Legend

Figure 3.2: Features with height level minus one in the same topographic data set as used for Figure 3.1.

can be used in the construction of the tGAP data structure. The following steps are adjusted or added to the pre-processing in order to store this information:

- The polygons with height level of zero are intersected with the polygons with a height level of minus one. Figure 3.3a contains a schematic representation of the bridge in Figure 3.1 and is intersected with the features that the located beneath, see Figure 3.3b. The result of the intersection is shown in Figure 3.4.
- The information on the polygon features is stored in database table at the end of the pre-processing. Table 3.1 shows the feature attributes stored, however this database table does not store the feature class of the features (see Table 3.2) that are located below. Therefore the features get an extra attribute called feature_class_below in which the feature class of the feature that is beneath is stored, see Table 3.3. Only the feature class of hydrography features with a height level of minus one are considered, all other features with a height level of minus one are not considered for this research.

The implementation details are described in Section 4.1. It can be noted that road junctions can be more complex, with multiple road features on top of each other. As this research does focus on hydrography networks these complex junction are not considered, however the feature class of the road features on top of each other could probably be stored as an array in the feature_class_below attribute and processed.



(a) Representation of the bridge of Figure 3.1

(b) Representation the the features with a height level of minus one of Figure 3.2 which are located beneath the bridge.

Figure 3.3: Representations of the bridge of Figure 3.1 and the polygons of Figure 3.2 which are located beneath the bridge.

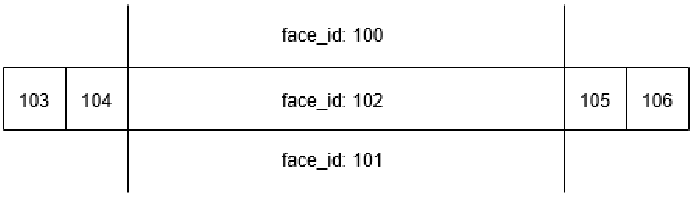


Figure 3.4: Representation of the bridge after the intersection of the features with height levels of zero and minus one.

Table 3.1: Database table of the features in Figure 3.3a.

face_id	feature_class	area(m^2)	geometry
100	12410	>300	polygon
101	12410	>300	polygon
102	10511	428.516	polygon

Table 3.2: Database table of the features in Figure 3.3b.

face_id	feature_class	area(m^2)	geometry
107	12410	288.843	polygon
108	14130	31.945	polygon
109	14160	37.297	polygon
110	14160	34.519	polygon
111	14160	35.911	polygon

Table 3.3: Database table of the features in Figure 3.4. In this database table the feature class of a hydrography feature that is beneath another feature is stored.

face_id	feature_class	feature_class_below	area(m^2)	geometry
100	12410		>300	polygon
101	12410		>300	polygon
102	10511	12410	288.943	polygon
103	10511		31.945	polygon
104	10511		37.297	polygon
105	10511		34.519	polygon
106	10511		35.911	polygon

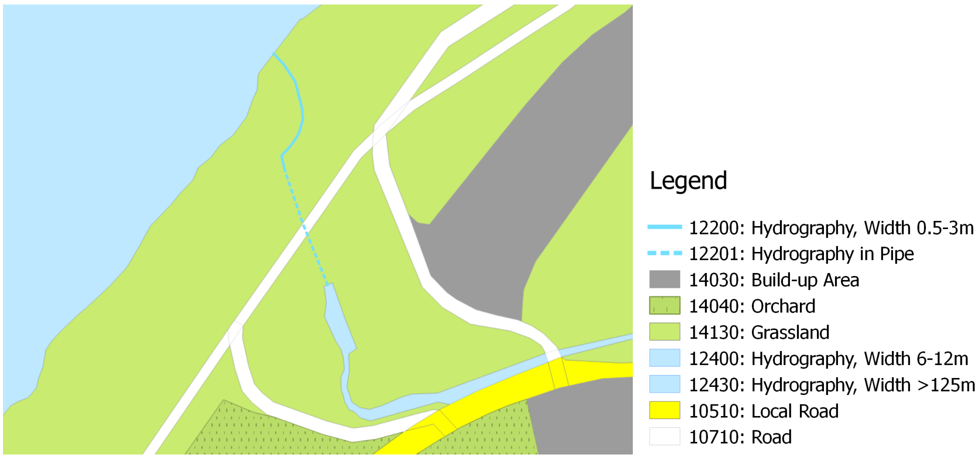
3.1.2 Hydrography Line Features

Hydrography features can also be represented by lines in the input large scale topographic data set used for the creation of the planar partition. At the moment line representations of features are not used in the pre-processing of the topographic data set for the creation of a planar partition. This means that e.g. that the connection between the two hydrography features represented by polygons in Figure 3.5a is lost. This connection however is important for the construction of a hydrography network. The following list describes what is added or adjusted in the pre-processing in order to include the hydrography line features in the planar partition so that it can be used for the construction of the tGAP data structure:

- All line hydrography line features are intersected with the polygon features in the topographic input data set. Breaking lines and polygons when an intersection occurs, and at the meantime overlapping line segments are combined if present in the input data. See Figure 3.6 where a schematic overview is given of the scene of Figure 3.5a.

- All hydrography line features that are connected to only one other hydrography polygon or line are iterative removed in the pre-processing as the vario-scale SDK can't handle these in the construction of the tGAP data structure.

- Now that all polygons are splitted into separate polygons when there is hydrography line feature going through it is needed to store the hydrography lines features so that they can be used for the construction of the tGAP data structure. Šuba et al. [2016] extended the tGAP data structure so that the feature class of the road faces, which are in the generalisation process collapsed to edges, can be stored in the attribute called edge_class. The database tables created at the end of the pre-processing (the last step of the pre-processing is to construct the topology of the planar partition and store the nodes, edges and faces in separate database tables) do store the feature class of the hydrography line features in the edge_class attribute, see Table 3.4.


Legend
12200: Hydrography, Width 0.5-3m
12201: Hydrography in Pipe
14030: Build-up Area
14040: Orchard
14130: Grassland
12400: Hydrography, Width 6-12m
12430: Hydrography, Width >125m
10510: Local Road
10710: Road

(a) (b) Legend

Figure 3.5: Example of a large scale topographic data set in which two polygon hydrography features are connected by a hydrography line feature. Where the hydrography feature is crossing the road it is below the surface in a pipe.

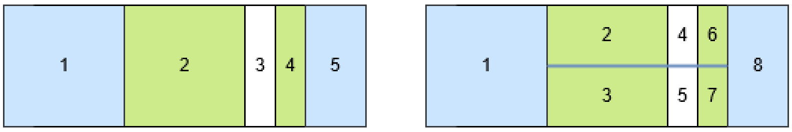


Figure 3.6: Schematic representation of the two polygon hydrography features and the hydrography line features connection them, and splitting the polygons which intersect with the hydrography line features.

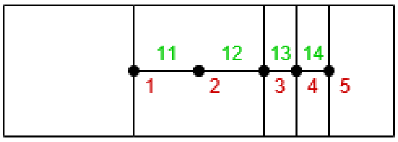


Figure 3.7: Nodes and edge which correspond to the hydrography line features of Figure 3.6 and Figure 3.5a.

Table 3.4: Edge_class for hydrography line features stored in database table.

edge_id	right_face_id	left_face_id	start_node_id	end_node_id	edge_class
11	3	2	1	2	12200
12	3	2	2	3	12201
13	5	4	3	4	12201
14	7	6	4	5	12201

3.2 HYDROGRAPHY NETWORKS IN TGAP DATA STRUCTURE

The input data sets that are currently supported by the vario-scale SDK are modelled as a two-dimensional polygonal map, i.e. a planar partition in the geometrical sense, without silvers or gaps and overlapping polygons. The resulting initial tGAP data structure contains only the topological primitives; nodes, edges and faces, where one polygon in the input data set corresponds to just one topological face in the initial tGAP data structure. The hydrography features in the input data set are represented by polygons and lines, which in the tGAP data structure are respectively faces and edges.

Beside the feature class of the faces and edges and the feature class of faces which are below another feature there is more information implicitly present in the initial tGAP data structure such a hydrography networks. These hydrography networks are implicitly in the input data, in the generalisation process the aim is to preserve their meaning throughout the generalisation process. As the hydrography networks are not explicitly modelled the implicit information about the role a feature plays in a hydrography network has to be made explicit.

Figure 3.8 shows a simple example of such a hydrography network in the in the initial tGAP data structure (see Figure 3.8a) and in the target small scale map (see Figure 3.8c). When two hydrography faces are incident they share an edge and they are considered connected. If this shared edge is an hydrography edge this edge is not connected to the hydrography faces on both sides, in the real world this hydrography feature is running beneath or above. When a hydrography feature is represented by an edge it incident with another hydrography edge when they share a node. An hydrography edge is incident with an hydrography face when the hydrography edge at the start or end node share this node only with an hydrography face and not with an hydrography edge. It is however not easy to keep track of linear network as the features in the map gradually change from scale to scale, see the intermediate step in Figure 3.8b. The same scene is shown in Figure 3.9 but this time the hydrography network is shown for better understanding of the relationship between the hydrography features during the generalisation process. From Figure 3.8 it can b seen that the geometrical representation of the hydrography features gradually changes throughout the generalisation process while the hydrography network stays the same, see Figure 3.9. Therefore the creation of hydrography networks can be used as an effective tool for the meaningful generalisation of hydrography features throughout all scales.

For the construction of the hydrography network or graph Algorithm 3.1 is used in which topo_map is the initial tGAP data structure. See Section 4.3 for details on the implementation. It is important to note that there can be several disconnected hydrography networks in the initial tGAP data structure.

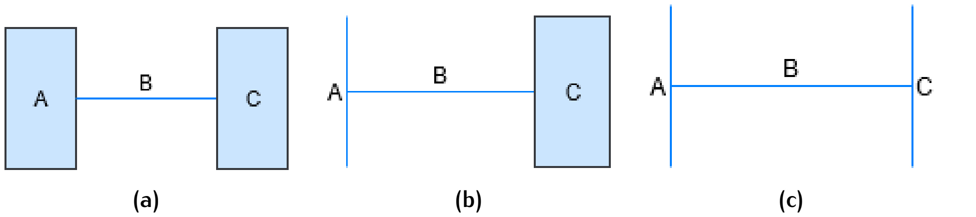
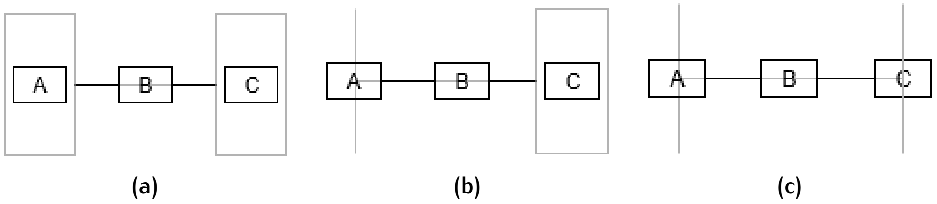


Figure 3.8: Example of hydrography network generalisation from the large scale **(a)**, through the intermediate step **(b)**, to the final scale **(c)**. Geometrical representation of the hydrography features changes from 2D to 1D, however the semantic and their role in the hydrogaphy network or graph remain the same.



(a) (b) (c)

Figure 3.9: The hydrography network or graph represents the same situation as in Figure 3.8. The topological relationships of the hydrography objects (represented by rectangles) are captured.

Algorithm 3.1: HYDROGRAPHYNETWORK

```
Input: topo_map and list with the hydrography feature classes
Output: Hydrography Network
Create empty Graph()
for face in topo_map do
    if not face unbounded/universe then
        if face.feature_class in hydrography_feature_class_list or
          face.feature_class_below in hydrography_feature_class_list then
            Graph.add_object((face.id, face ← True))
for edge in topo_map do
    if edge.edge_class in hydrography_feature_class_list then
        Graph.add_object((edge.id, face ← False))
for object in Graph do
    if object is face in topo_map then
        neighbours ← topo_map.faces[object[0]].neighbours_no_same_face
        for n in neighbours do
            if not face unbounded/universe then
                if (n.id, face ← True) in Graph then
                    Graph.add_link(object, (n.id, face ← True))
    if object is edge in topo_map then
        for vertex in Graph do
            if vertex is edge in topo_map then
                if start/end node of object == start/end node of vertex and object !=
                  vertex then
                    Graph.add_link(object, (vertex.id, face ← False))
    if object is edge in topo_map and object connectivity <= 1 then
        non_hydrography_neighbour_edge ← set()
        for egde in topo_map do
            if start/end node of object == start/end node of edge and
              (edge.id, face ← False) not in Graph then
                non_hydrography_neighbour_edge.add(edge)
        for non_hydrography_edge in non_hydrography_neighbour_edge do
            if (right/left face of non_hydrography_edge, face ← True) in Graph
              and right/left face of object != right/left face of non_hydrography_edge
              then
                Graph.add_link(object, (right/leftface.id
                  ofnon_hydrography_edge, face ← True))
```

3.3 FLOW DIRECTION IN THE HYDROGRAPHY NETWORKS

In natural hydrography networks the water is always flowing from higher elevations towards lower elevations. In man-made parts of the hydrography network the water flow can be different as pumping stations and sluices allow humans to control the water flow. If the flow direction in a hydrography network is known, this information could be used in the generalisation process to make generalisation decisions in such a way that meaningful hydrography networks are preserved longer in the generalisation process.

The aim of the vario-scale concept is that it should be generic, therefore the flow direction should be determined based on input data that is generally available. This means that no stream gauge data will be used as this data is not available everywhere. Instead a freely accessible DEM will to used to try to determine the flow direction in the hydrography network. Section 4.2 describes the experiments done for determination of the flow direction in the hydrography networks.

The outlet point of the hydrography network can be determined if the flow direction is known. When the outlet point of the hydrography network is known it is possible to construct the Strahler order (see Section 2.3) for all the hydrography features in the network with a breath-first search traversal of the hydrography network. Algorithm 3.2 and Algorithm 3.3 are used to assign the Starhler order to each hydrography feature and thereby indicates the importance of that feature in the hydrography network. This additional information can be used in the decision making during the generalisation process.

Algorithm 3.2: Go_upstream

```
Input: Hydrography Object
object.visited ← True
if number of links == 1 and object != outlet then
    object.strahler ← 1
    return
strahler_values ← empty list
for vertex in connected objects do
    if not vertex.visited then
        go_upstream(vertex)
        strahler_values.append(vertex.strahler)
object.strahler ← get_strahler_number(strahler_values)
```

Algorithm 3.3: Get_strahler_number

```
Input: strahler_values from Algorithm 3.2
Output: Strahler number or order
unique_strahler_values ← set(strahler_values)
if length of strahler_values > 1 then
    if length of set unique_strahler_values == 1 then
        return strahler number + 1 from unique_strahler_values
    else
        return maximum strahler number of unique_strahler_values
else
    return strahler number from strahler_values
```

3.4 DEVELOPMENT OF GENERALISATION DECISIONS

The aim of the research is to study the possibilities to better incorporate hydrography networks in the vario-scale concept for the creation of a vario-scale basemap while maintaining the hydrography network structure.

The creation of the content for the vario-scale data structure is based on the tGAP principle, where the least important object is merged with the most compatible neighbour or splitted and divided among the neighbours. Šuba et al. [2016] extended the generalisation process to be able to deal with road network knowledge, see Section 2.2. This research is used as a basis for the development of the generalisation decisions that have to be made for the better generalisation of hydrography networks. The design decisions made for the development are described below:

Design Decisions One: Šuba et al. [2016] developed a method for the generalisation of road networks throughout all scales, as the focus in this research is on hydrography networks the road networks are not considered. Therefore for the creation of the vario-scale content two different object classes are considered: hydrography features (input: polygon and line) and other features (input: polygon). This makes sure that there are not many different object types, which makes the generalisations decision that have to be made in the generalisation process more transparent and more easy to implement. In the future it can always be made more complex with more object types and decisions that have to be made. Further research could first focus on how to combine the generalisation decisions for road networks and the generalisation decisions for hydrography networks in the generalisation process.

Design Decision Two: At the start of the process every face in the initial tGAP data structure gets an importance value based on the size (area) and feature class of the feature (in the initial large-scale map there are also line features, however as the current generalisation process is driven by faces these are not considered and therefore do not get an importance value). Note that the computation of the importance value can be adjusted, see Section 6.2. The face with the lowest importance value will be processed first and the importance values of all the faces, that are involved with the processing of this face, will be updated.

Design Decision Three: The feature class of the selected face determines which processing option is done, see Figure 3.15 for a schematic overview of the possible options. The possible processing options are:

- The face is a hydrography feature and will be merged (see Figure 3.10) with an adjacent hydrography face feature (hydrography features with same Strahler order have priority to be merged with) or it will be collapsed to a line if there are no hydrography face adjacent (see Figure 3.11).

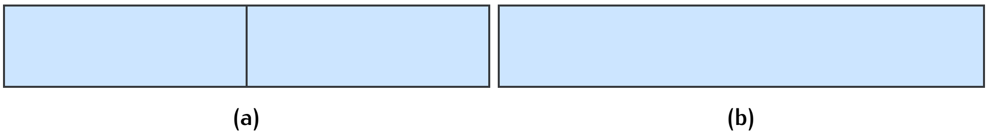


Figure 3.10: Two hydrography features **(a)** are merged **(b)** as they are adjacent.

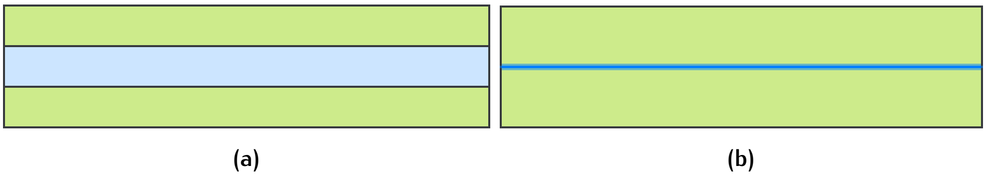


Figure 3.11: Hydrography face feature which is collapsed to a line as there are no other hydrography face features adjacent.

- If two hydrography features need to be merged, see e.g. the features **5** and **6** of Figure 3.12a, first the common boundary between them is checked. If this common boundary is a hydrography feature it means that the hydrography line feature is not connected (due to the data structure) to the selected faces in the process. Figure 3.12a shows an example of underground hydrography pipe connecting the hydrography faces **1** and **10**, this hydrography pipe is however not connected to the hydrography faces **5** and **6**. If this is the case the other adjacent hydrography features of hydrography face **5** in Figure 3.12a will be checked as well, if there is a neighbour with no hydrography edge as common boundary (hydrography face **4** in Figure 3.12a) these features will be merged, see Figure 3.12b for the result. When all adjacent hydrography features share a hydrography edge as boundary the hydrography face will be merged with a adjacent hydrography feature with which is shares the shortest common boundary.

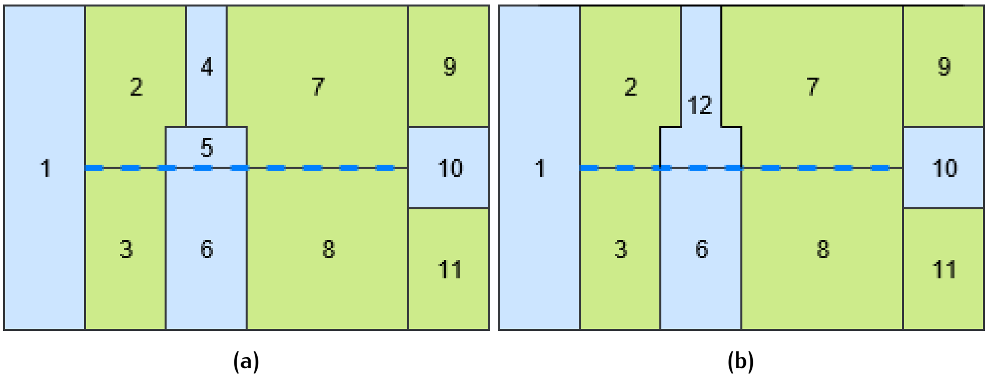


Figure 3.12: If a feature should be merged, it merges with the neighbouring face with which it shares the longest common boundary. If two hydrography faces are selected for the merge operation first the common boundary between them is checked, if this common boundary is a hydrography edge the merge should be done with a adjacent hydrography feature which doesn't share an hydrography edge. In this way the hydrography networks are preserved as long as possible in the generalisation process.

- The selected face is of the type other (e.g. face **2** of Figure 3.13a) and the neighbouring face selected (face **1** of Figure 3.13a) for the merge operation is also of the type other, before processing first the common boundary of these faces will be checked. If the common boundary is a hydrography edge all other neighbours of type other will be checked to see if there is a neighbouring face without a hydrography edge between them. If there are neighbouring faces (face **3** of Figure 3.13a) without a hydrography edge as common boundary the merge operation is done with the neighbouring face which has the longest non-hydrography common boundary, see Figure 3.13b for the result. If there is no non-hydrography common boundary the face will be merged with the neighbour with which is shares the shortest common hydrography boundary.

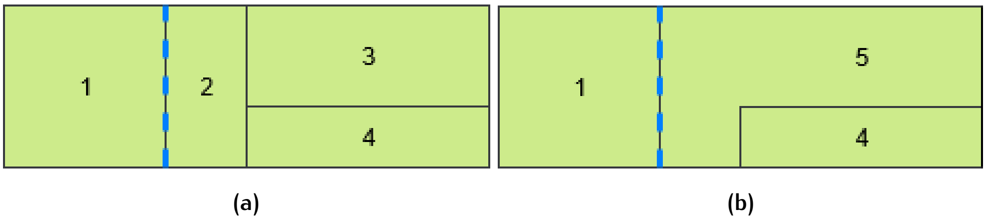
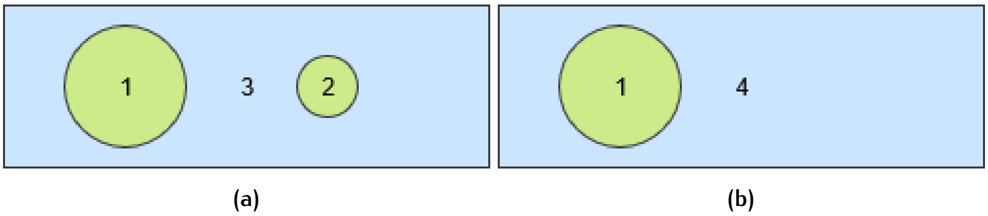


Figure 3.13: Merge of faces of type other with check of the common boundary for hydrography. If hydrography boundary, select neighbour of type other with a non-hydrography common boundary.

- The selected face is of the type other (face 2 in Figure 3.14a) and the adjacent face(s) is/are hydrography, the face will me merged with a hydrography feature, see Figure 3.14.



(a)

(b)

Figure 3.14: Feature of type other (face 2) selected for processing, it is merged with hydrography feature as there are no neighbouring faces of the type other.

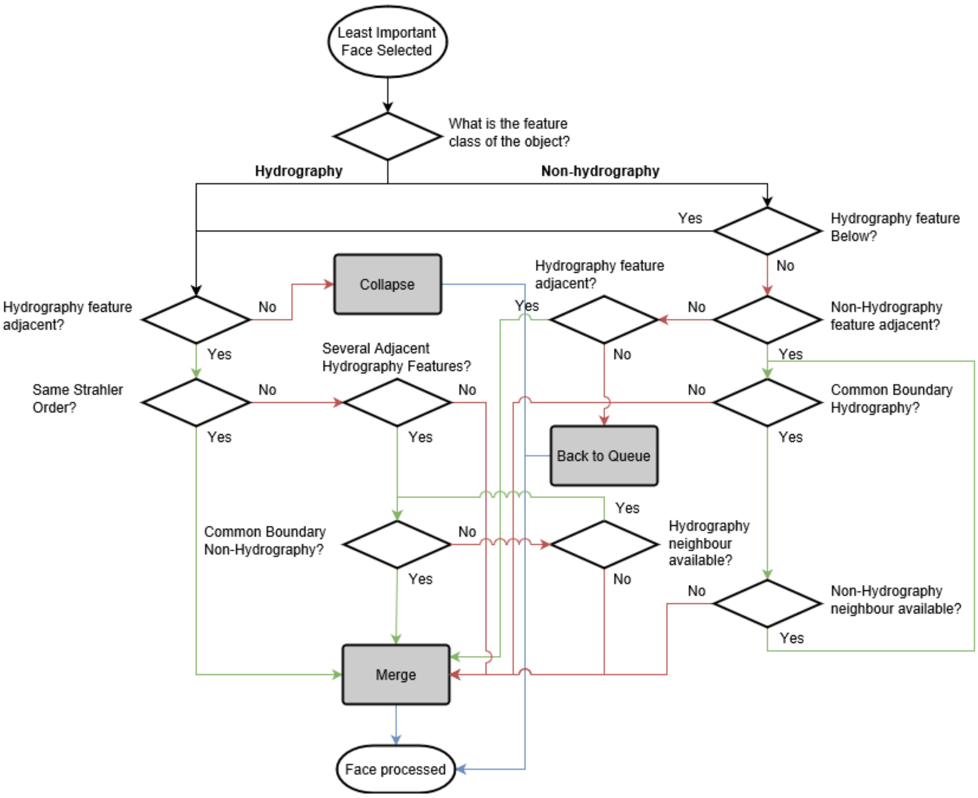

Least Important Face Selected
What is the feature class of the object?
Hydrography
Non-hydrography
Yes
Hydrography feature Below?
No
Collapse
Hydrography feature adjacent?
Hydrography feature adjacent?
No
Yes
No
Non-Hydrography feature adjacent?
Yes
Several Adjacent Hydrography Features?
No
Yes
Same Strahler Order?
No
No
No
Common Boundary Hydrography?
Yes
Yes
Yes
Back to Queue
Yes
Hydrography neighbour available?
Common Boundary Non-Hydrography?
No
Yes
No
No
Non-Hydrography neighbour available?
Yes
Merge
Face processed

Figure 3.15: Schematic representations of the decisions made for one generalisation step

Design Decision Four: The iterative software development model (see Figure 3.16) is used for the development of the generalisation decisions in the vario-scale concept. First a generalisation decision is designed after which the decision is coded in the vario-scale SDK. Next the generalisation decisions is tested and verified with real data and the result is critically assessed. In the next iteration more generalisation decisions are added or existing decisions are improved.

After a certain number of iterations the generalisation result probably has significantly improved. It is a design decision that the development iterations are stopped when the main hydrography network in the real world test data set is present in at least 90% of the SSC.

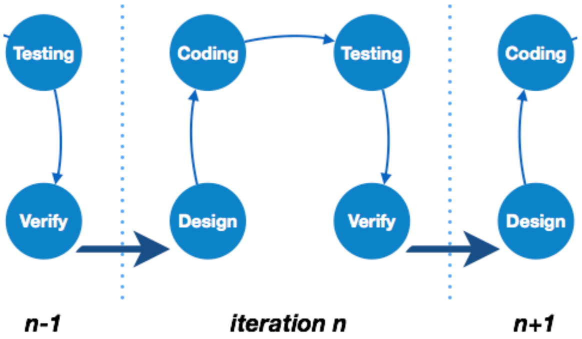


Figure 3.16: Iterative software development model [Tutorials Point, 2017].

3.5 QUALITY OF THE GENERALISATION

The quality of the generalised vario-scale basemap with the generalisation decisions described in Section 3.4 will be visually and quantitatively compared to a generalised vario-scale basemap without these decisions. For the quantitatively assessment the following are used:

- Statistics are gathered on the amount of hydrography features throughout the generalisation process.
- Algorithm 3.1 is used to construct the hydrography graph. This graph is used to get the number of connected components, which indicates how many hydrography networks are present in the data set.

The hydrography network structure should not be torn apart during the generalisation process. It is however possible that a complete network disappears in the generalisation process because it is not important enough to be in the tGAP data structure throughout all the scales.

4 IMPLEMENTATION AND EXPERIMENTS

Chapter 4 describes the details of the pre-processing and processing stage. Also the experiments done to determine the flow direction based on a DEM are described.

4.1 PRE-PROCESSING OF THE DATA

Information on the used data set can be found in Section 4.1.1. The pre-processing of the data set is described in Section 4.1.2.

4.1.1 Data set

The input data set that is used for this research is the municipality of Valkenburg which is part of the province of Limburg, the Netherlands. The data set used is a TOP10NL (scale 1:10,000) vector data set produced and distributed by the 'Netherlands Kadaster'[1]. The data layers used for this research are: hydrography area, hydrography line, roads, and terrain. These layers need to be combined to form the clean planar partition which is needed to construct the edge and face table which are the base of the tGAP data structure.

4.1.2 Creation Clean Planar Partition and Tables needed for tGAP

Software from the company Safe called FME[2] is used to create the clean planar partition from the input data layers. This clean planar partition is afterwards used in FME to construct the node, edge and face table which are needed as input for the creation of the initial tGAP data structure. The main difference with previous pre-processing is that now information of hydrography polygon features that are below another feature is stored and that hydrography line data is used. The pre-processing is described in detail steps:

Municipality Data

Outlines of the municipalities in the province of Limburg, the Netherlands are imported and the municipality of Valkenburg is selected. Next the Bounding Box (BBOX) of the municipality of Valkenburg is created which will be used as the extent for the data set used in this research, see Figure 4.1.

Hydrography Data

The hydrography data layers contains hydrography point features, hydrography line features and hydrography polygon features. For this research the hydrography line and polygon features are used (see Figure 4.1). Hydrography point features which are e.g. wells are not considered.

[1] https://www.kadaster.nl/-/top10nl

[2] https://www.safe.com/fme/fme-desktop/

Road Data

The road data layers contains line features, polygon features and collections. First the polygon features and collections are selected, the road polygon features are directly suitable while the collections, which are road polygon features together with the road centerlines, need to be processed to extract the area features. See Figure 4.1 for a schematic overview of the processing steps.

Terrain Data

The terrain data layer has holes and gaps at the locations of hydrography and road features. Buildings which are not considered for this research are on top of the terrain features and would require extra processing when they are considered.

Extend and Attributes

Now that all the needed data is available in the correct geometry type the data is clipped to the extend described in Municipality Data. After clipping to the extend the data does still contain all attributes of the original input data, however not all these attributes are needed. The feature class and height level (see Section 3.1.1) attributes are needed in the process, the rest of the original attributes is removed, see the last step in Figure 4.1.

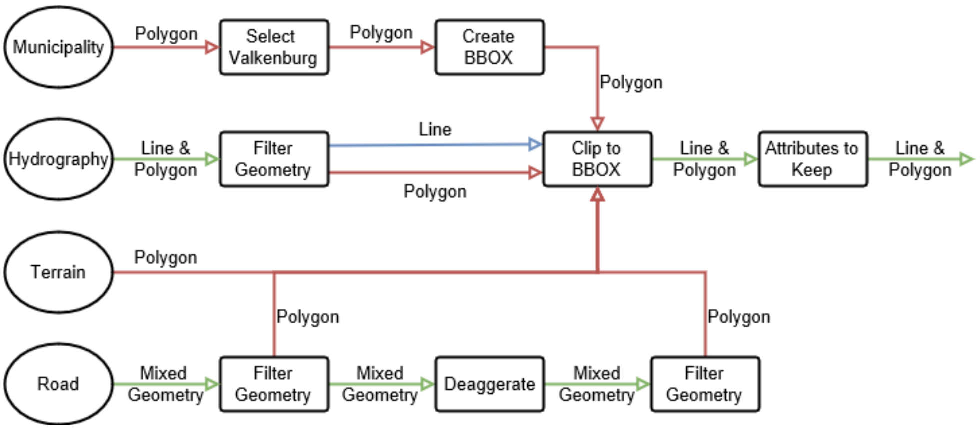


Figure 4.1: Pre-processing in FME: Filtering of input data layers, clipping to the correct extend and attribute selection.

Snapping and Removal of Spikes

Segments are snapped together when the distance between them is less than 0.1m, this operation could produce spikes in some cases. The spikes are removed if the angle between them is less than 0.5 degrees.

Filtering: Height Level, Feature Class and Geometry

All features with a height level of minus one, which indicates that the feature is below some other feature, are selected. The selected feature are filtered based on the feature class, for this research only hydrography features that are below another feature are considered. The feature class of hydrography features that are beneath another feature is a negative number between -12999 and -12000. All other features with a different feature class that are below another object are not considered and are lost at this stage in the pre-processing. See Figure 4.2 for a schematic overview. All features with a height level of zero are filtered based on their geometry because the line features and polygon features need to be treated differently for the coming steps in the pre-processing.

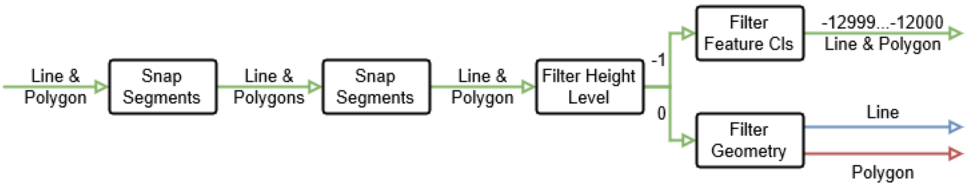


Figure 4.2: Pre-processing in FME: Continuation of Figure 4.1, snapping of segments, spike removal, height level filtering, selection hydrography features with height level of minus one and geometry filtering of features with height level zero.

Hydrography Polygon Features Processing

The hydrography polygon features with a height level of minus one are filtered based on geometry, see Figure 4.3. The feature class of the hydrography polygon features is a negative number, this is not wanted anymore so the feature class is changed to the absolute value of the feature class, see Figure 4.3. Next an extra attribute called hydro_below is created and assigned the feature class value, in the next processing step all other attributes are removed (see, Figure 4.3). Next these hydrography polygon features are intersected with the features which have a height value of zero. This makes sure that hydrography features that are below another feature have exactly the same size as the feature that is on top. For example the bridge in Figure 3.1a is divided into three parts due to the hydrography polygon (see, Figure 3.2a) that is beneath the bridge.
With the intersection of the polygon features the attributes of the polygons that intersect are merged.

Extra Attribute for Hydrography Line Features

All the hydrography line features get an extra attribute called hydro_edge, see Figure 4.3. The value of this new attribute is set to one.

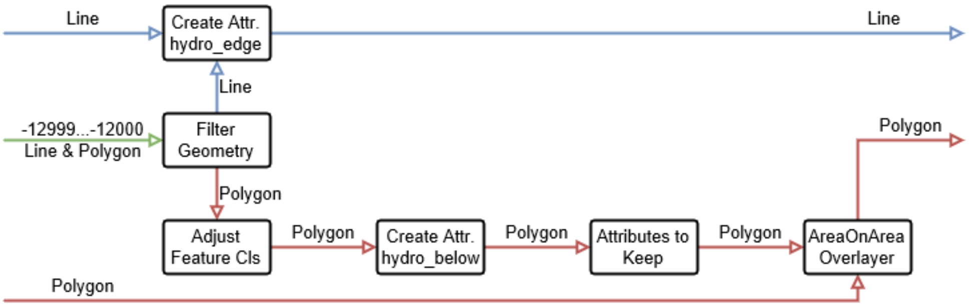


Figure 4.3: Pre-processing in FME: Continuation of Figure 4.2, creation of extra attributes for hydrography lines and polygons and the intersection of hydrography polygons (with height level of minus one) with all other polygon features.

Set Hydrography Feature Class When Below

Now that hydrography features that are below other features have exactly the same size it is possible to store the feature class of this hydrography features. An extra attribute called feature_class_below (see Figure 4.4) is created which gets the value of the attribute hydro_below. So if the feature_class_below contains a value it is known that a hydrography feature with that feature class is beneath.

Intersect Hydrography Lines with Hydrography Polygon Features

The polygon features are intersected with the hydrography line features, see Figure 4.4. The result of the intersection will be edges and nodes. The edges are used to construct polygon features and all edges that have the attribute hydro_edge are selected. After polygons have been build from the edges, some edges will not be part of an polygon as they are dangling (have a connectivity of one) these are called incomplete (see Figure 4.4). These edges are overlaid with all edges having the attribute hydro_edge, the overlaying tool creates a new attribute called overlap. If two input edges are overlapping the value of the overlap attribute will be two. These are the hydrography lines/edges that are not wanted, so they will not be used anymore in the processing.

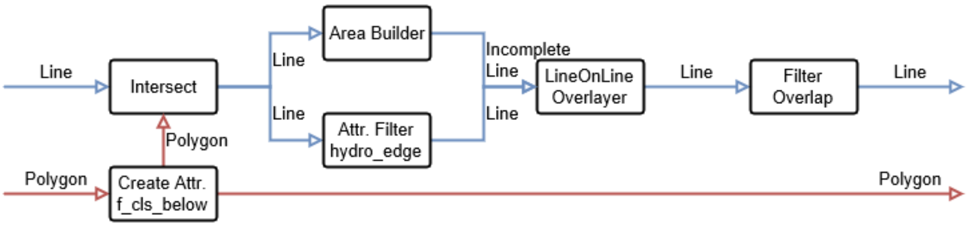


Figure 4.4: Pre-processing in FME: Continuation of Figure 4.3, storing the feature class of the hydrography polygon features that are below other features in their attributes and detection of the hydrography lines that are not wanted.

Correct Attributes and Topology Builder

The hydrography line features regarded as correct and use full are intersected with the polygon features, see first block in Figure 4.5. The resulting edges are used by the area builder to create the correct clean planar partition. Next the newly build polygons are overlaid with the polygons created before. The results will be that polygon features which were on top of other features are now dividing them into two parts, e.g. the terrain polygon feature next to the hydrography feature in Figure 3.6 is now divided into two part by the hydrography lines features.
The last step is to construct the node, edge and face tables needed for the creation of the initial tGAP data structure. The topological structure is constructed, see Figure 4.5 and the resulting tables are stored in a database. The structure of the tables is shown in Figure 4.6. Compared to previous pre-processing the feature_class_below is new and the edge_class does now contain data for the hydrography edges from the beginning.

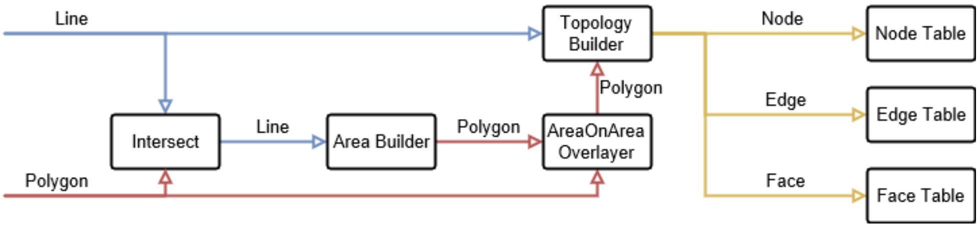


Figure 4.5: Pre-processing in FME: Continuation of Figure 4.4, last steps to make the input data ready for the creation of the node, edge and face tables.

```
CREATE TABLE _node (
node_id integer);
```

```
CREATE TABLE _edge (
edge_id integer,
right_face_id integer,
left_face_id integer,
start_node_id integer,
end_node_id integer,
edge_class integer);
```

```
CREATE TABLE _face (
face_id integer,
feature_class integer,
area float,
feature_class_below integer);
```

Figure 4.6: Node, Edge and Face tables in the database needed for the initial creation of the tGAP data structure.

4.2 FLOW DIRECTION HYDROGRAPHY NETWORK

It is important to keep in mind that the creation of the tGAP data structure should be generic and possible with data sets from around the globe. As not all countries have an detailed DEM, a global DEM is used to get the height values of the hydrography features, which in turn can be used to determine the flow direction as water normally flows from higher elevations towards lower elevations. The Shuttle Radar Topography Mission (SRTM) DEM with a 30m resolution is used for determining the heights of the hydrography features as this DEM is available for all countries.

Several experiments were conducted to store the height of a hydrography features. Average height for hydrography features was calculated and stored, after which these height values are used to determine the flow direction. Another option tested is to store the height values for all nodes of a hydrography face and based on these height values determine the flow direction of the water. The main challenge is determination of the flow direction and after several experiments it appeared that this approach is not suitable for the test data set due to the too coarse resolution of the SRTM DEM compared to the size of the hydrography features.

A second experiment has been conducted to see if it is possible to determine the flow direction when a much more detailed DEM is used. The Dutch provinces, central government and the waterboards cooperated to create a point cloud with the heights of the Netherlands which is called AHN3[3]. AHN3 is used by the 'Netherlands Kadaster'[4] to construct a DEM which is available with a spatial resolution of 0.5m and 5.0m. Both are used to see if the flow direction can be determined with a more detailed DEM. Something important to note is that these DEMs are constructed based on Light Detection And Ranging (LiDAR) measurements, as LiDAR is absorbed by water no data is available for hydrography features, only the heights of the features directly adjacent to the hydrography features can be used. In experiments these height values are used for determination of the flow direction, however difficulties in doing so again arise. As most waterways in the Netherlands are embanked and there are many man-made waterways the height values of the banks are often higher than the surrounding terrain and are varying a lot along a waterway due to different embankments (stone/earthen). After several tests it appears that even with a detailed DEM it is not feasible to determine the flow direction for all hydrography feature with the chosen approach.

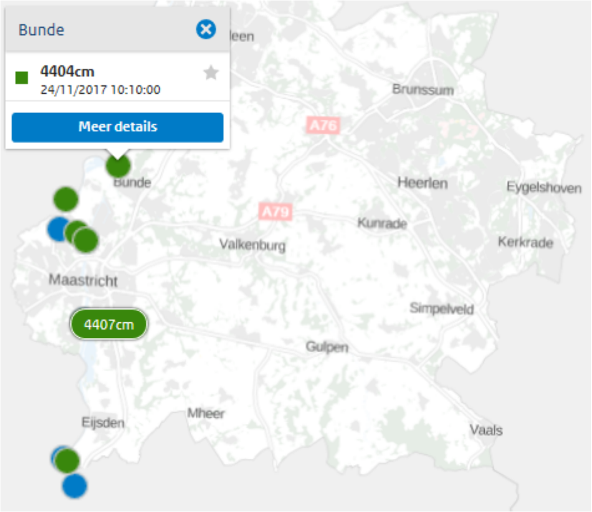


Figure 4.7: Map of part of the province of Limburg, the Netherlands. The map shows the locations where water levels are measured, and shows the water level at two different locations.

[3] http://www.ahn.nl/common-nlm/over-ahn.html
[4] https://www.pdok.nl/nl/ahn3-downloads

As last option it should be possible to get water levels from the waterboard(s), see Figure 4.7 for water levels published by the Dutch waterboard[5]. As can be seen in Figure 4.7 the water level does not difference a lot between the two locations shown. As the test data is from the area around Valkenburg there is no water level data available there. Therefore experiments with this data are skipped. The main motivation behind this is that water level data is not available everywhere on the globe, while the vario-scale approach should be generic and applicable everywhere. Due to the difficulties of determining the flow direction for the real world test data set, it is decided not to use it for this research.

4.3 CREATION HYDROGRAPHY NETWORKS

The feature class of all the faces and edges that are hydrography are known and stored in the database tables used for the creation of the initial tGAP data structure. This makes it possible to construct network(s) for the connected hydrography features. Algorithm 3.1 is used to construct the hydrography graphs. For construction of the hydrography graphs the Python Software Package NetworkX[6] is used in the vario-scale SDK. NexworkX is a Python Software Package used for the creation, manipulation, dynamics, function and structure of (complex) networks. After construction of the hydrography network it is e.g. possible to determine to which hydrography network faces and edges belong and it is possible to get the number of connected components (with NetworkX) which is the amount of hydrography networks present in the data set.

The hydrography networks in the data set are now known, would it be possible to construct the flow direction based on the feature class in the hydrography network? Water flows naturally from higher towards lower elevations and most rivers end up in the sea (there are some exceptions however). In the Netherlands this is however much more complicated due to pumps and mills altering the water flow. One option would be to check which feature of the hydrography network is connected to the sea, if this is the case we can say that this feature is the outlet point for the considered hydrography network. The second option would be to search in the network for signs that indicate which feature may be the outlet point of the hydrography network. Signs like this are e.g. feature class of the hydrography features. Hydrography features have a feature class based on the width of the feature, if not there in the input topographic data this can be added in the preprocessing step. Natural hydrography networks often start with all kinds off small streams which join in rivers which tend to get wider and wider the further from the source.

In the data set used for this research there is no sea feature or lake feature as outlet point of the hydrography network, so the outlet point is determined based on the feature class if possible. In the hydrography network all features with a connectivity of one are visited, if there is a single hydrography feature with a feature class that represents a width of the feature wider than all other hydrography features this feature is selected as outlet point of the hydrography network. In the real world test data set used for this research it is however with above described methods not possible to determine the outlet point of the hydrography network, which means that the flow direction cannot be constructed and therefore is not used.

[5] https://waterinfo.rws.nl/#!/kaart/waterhoogte-t-o-v-nap/
[6] https://networkx.github.io/documentation/networkx-1.9.1/overview.html

5 RESULTS AND ANALYSIS

Chapter 5 provides the results of the research and gives an analysis of the obtained results. The first section shows the used real world test data set, and provides statistics in the data in the data set. The second section shows the generalisation results, while the last section compares the generalisation results with the result of the tradition vario-scale.

5.1 TOPOGRAPHIC INPUT DATA

For the experiments a subset of the Dutch topographic map (TOP10NL) intended for usage at a 1:10.000 scale is used[1]. The subset used is the region around the municipality of Valkenburg in the province of Limburg, the Netherlands, see Figure 5.1. The data is pre-processed as described in Section 4.1 resulting in a test data set, see Figure 5.2 with an area of 9.8km x 7.0km consisting of 16.208 nodes, 26.239 edges and 10354 faces. The building features are not used in the test data set, as pre-processing of them is more demanding and generalisation of buildings is not the aim of this research.

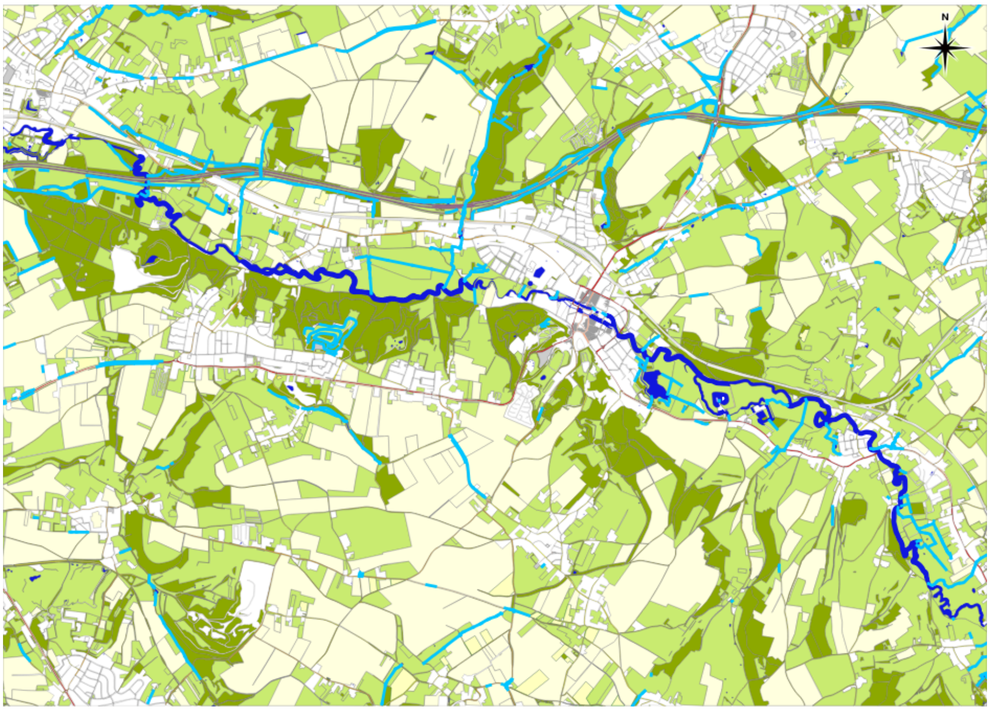


Figure 5.1: Topographic data from the municipality of Valkenburg in the province of Limburg, the Netherlands. Hydrography polygon features (dark blue) and hydrography line features (light blue) are enhanced to make them more visible.

Table 5.1 shows the amount of each feature type in the initial tGAP data structure. There are 33 non-hydrography features which have an hydrography feature directly below them. As only hydrography line features are used in the pre-processing of

[1] https://www.kadaster.nl/-/top10nl

the data all non-hydrography edges in the initial tGAP data structure do not contain information, so no value for edge_class.

Table 5.1: Initial edges and faces in the tGAP data structure. Last column indicates how many non-hydrography features have a hydrography feature directly below.

	Terrain	Hydrography	Road	Hydrography Below
Edge	-	1169	-	-
Face	4557	260	5537	33

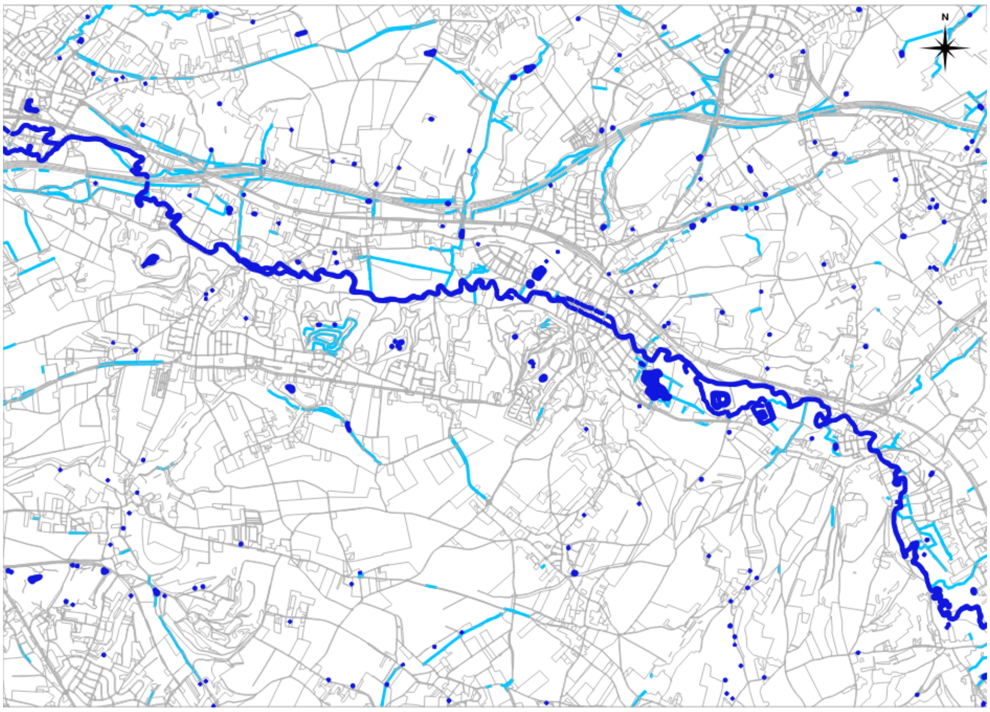


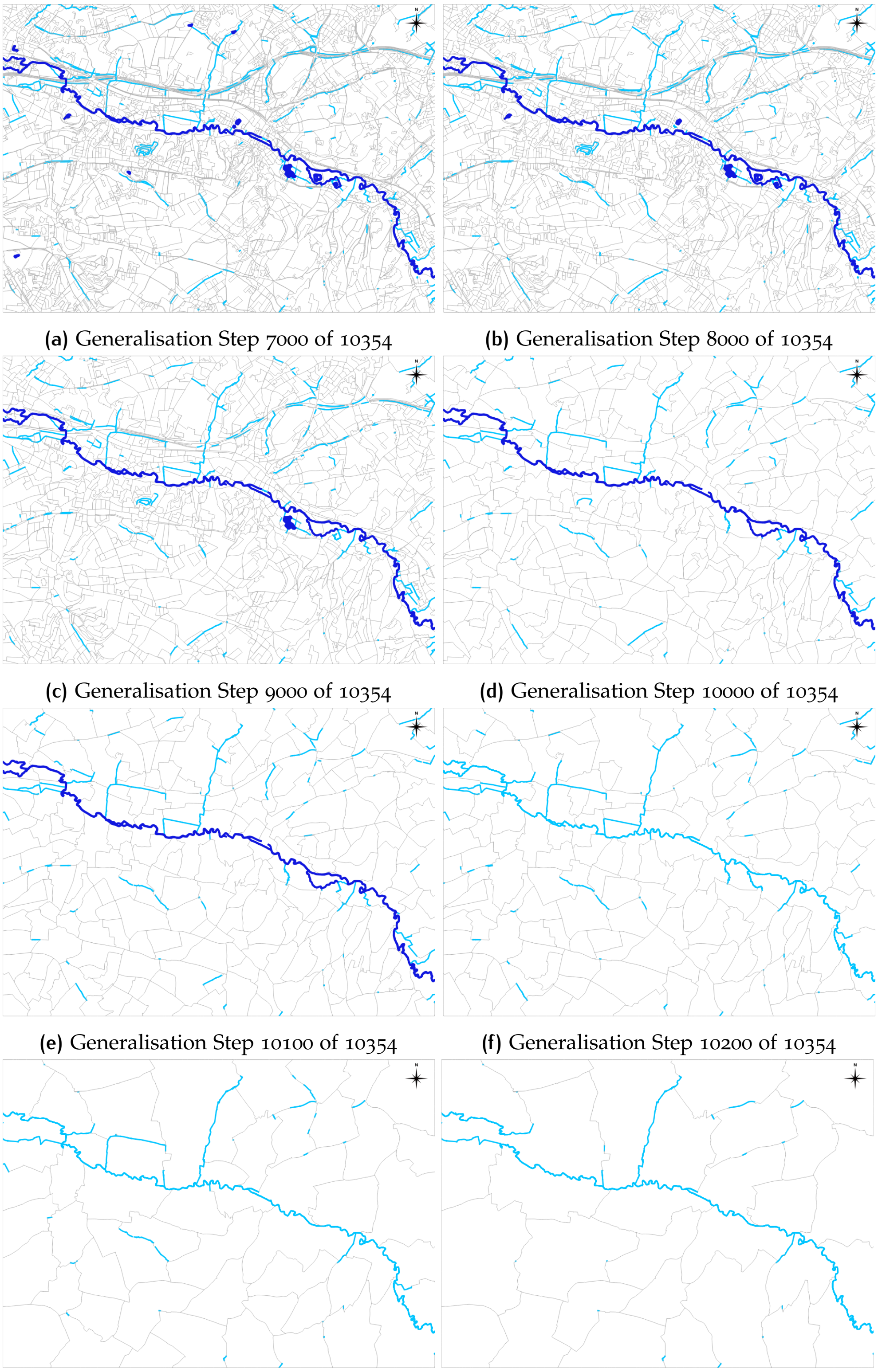
Figure 5.2: Initial tGAP data structure, amount of features and their feature class are described in Table 5.1. For readability purposes all non-hydrography faces are displayed white and all hydrography features are enhanced. Hydrography faces are dark blue, while hydrography edges are light blue. All other edges are grey.

5.2 GENERALISATION RESULTS

The topographic input data is processed with the vario-scale SDK into a vario-scale data structure with the use of the merge and split generalisation operations. Line simplification could also be included in the generalisation process. However this is not included in the generalisation process used in this research as the focus is on how hydrography network features are processed without any additional effects.

Figure 5.3 shows the developed treatment for hydrography networks on the test data set. It demonstrates that through making small generalisation steps an incrementally simpler map is generated.

The total area of the hydrography faces as percentage of the total area of all faces throughout the generalisation process is shown in Figure 5.4. At the end of the generalisation process no hydrography faces are left, because all hydrography faces are collapsed (see Figure 5.3) in the generalisation process.



(a) Generalisation Step 7000 of 10354

(b) Generalisation Step 8000 of 10354

(c) Generalisation Step 9000 of 10354

(d) Generalisation Step 10000 of 10354

(e) Generalisation Step 10100 of 10354

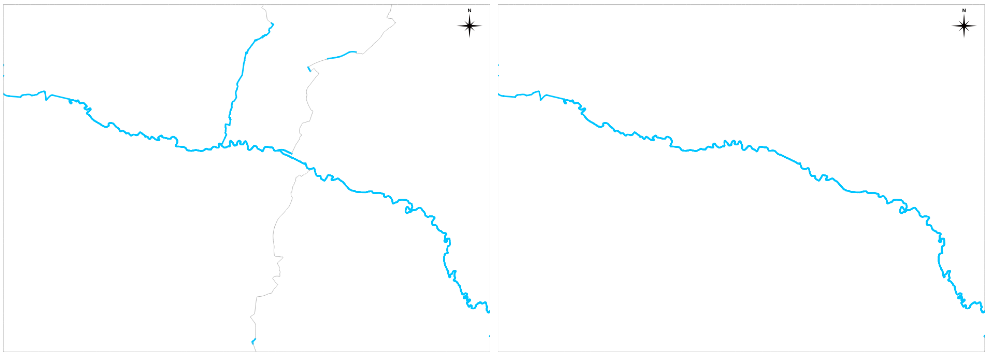
(f) Generalisation Step 10200 of 10354

(g) Generalisation Step 10300 of 10354

(h) Generalisation Step 10325 of 10354

Figure 5.3: Situations at different steps and scales in the generalisations process. The subfigures illustrate how the vario-scale data structure evolves. Note that all maps are displayed at the same size to clearly show the effect of the generalisation process. In reality the maps at a higher step in the generalisation process should be shown at a smaller scale.

The number of hydrography faces and hydrography faces that are below another feature throughout the generalisation process can be found in Figure 5.5.



(i) Generalisation Step 10350 of 10354

(j) Generalisation Step 10353 of 10354

Figure 5.3: Situations at different steps and scales in the generalisations process. The subfigures illustrate how the vario-scale data structure evolves. Note that all maps are displayed at the same size to clearly show the effect of the generalisation process. In reality the maps at a higher step in the generalisation process should be shown at a smaller scale.

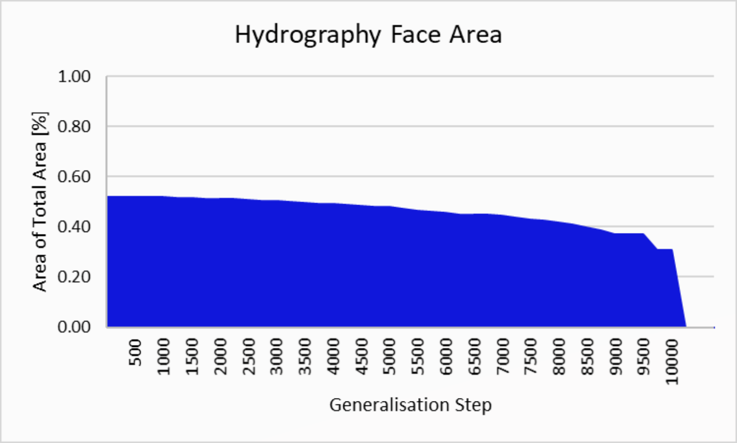


Figure 5.4: Hydrography face area as percentage of the total area of the faces in the data set. At the end of the generalisation process there is no hydrography face area, as all hydrography faces are collapsed in the generalisation process, see also Figure 5.3.

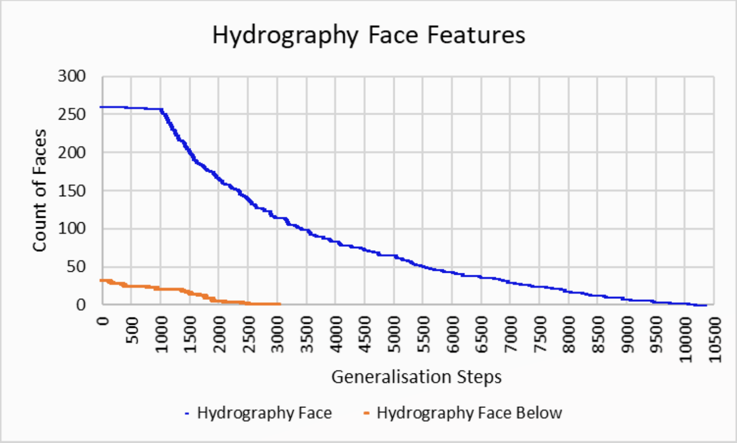


Figure 5.5: Number of hydrography face features and hydrography feature that are below another feature throughout the generalisation process.

Figure 5.6 shows the number of hydrography edges throughout the generalisation process. At the moment the large hydrography face is collapsed (see, Figure 5.3e and Figure 5.3f) an enormous amount of hydrography edges is created which is indicated by the line in Figure 5.6 going 'out' of the chart. This artefact is due to the fact that there is no line simplification involved in the generalisation process.

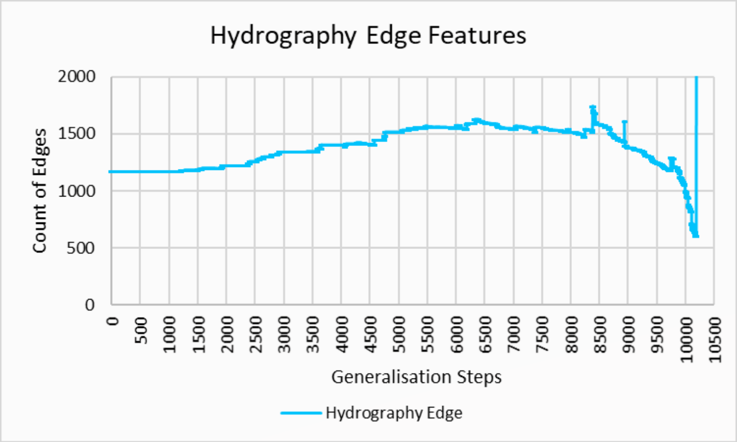


Figure 5.6: Number of hydrography edges throughout the generalisation process.

The number of connected components which indicates the number of hydrography networks in the data set can be found in Figure 5.7.

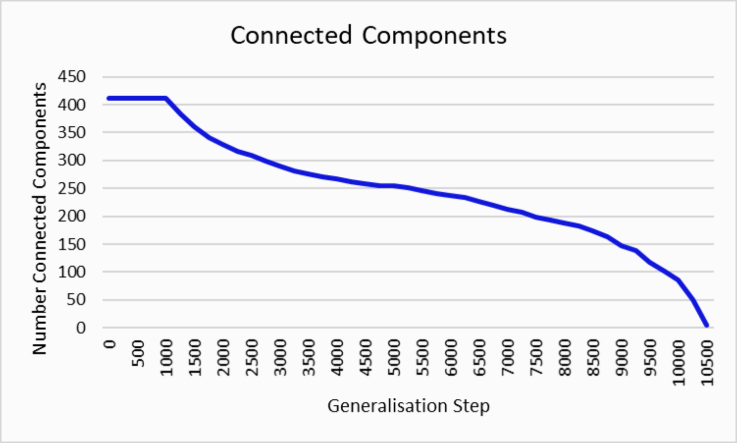
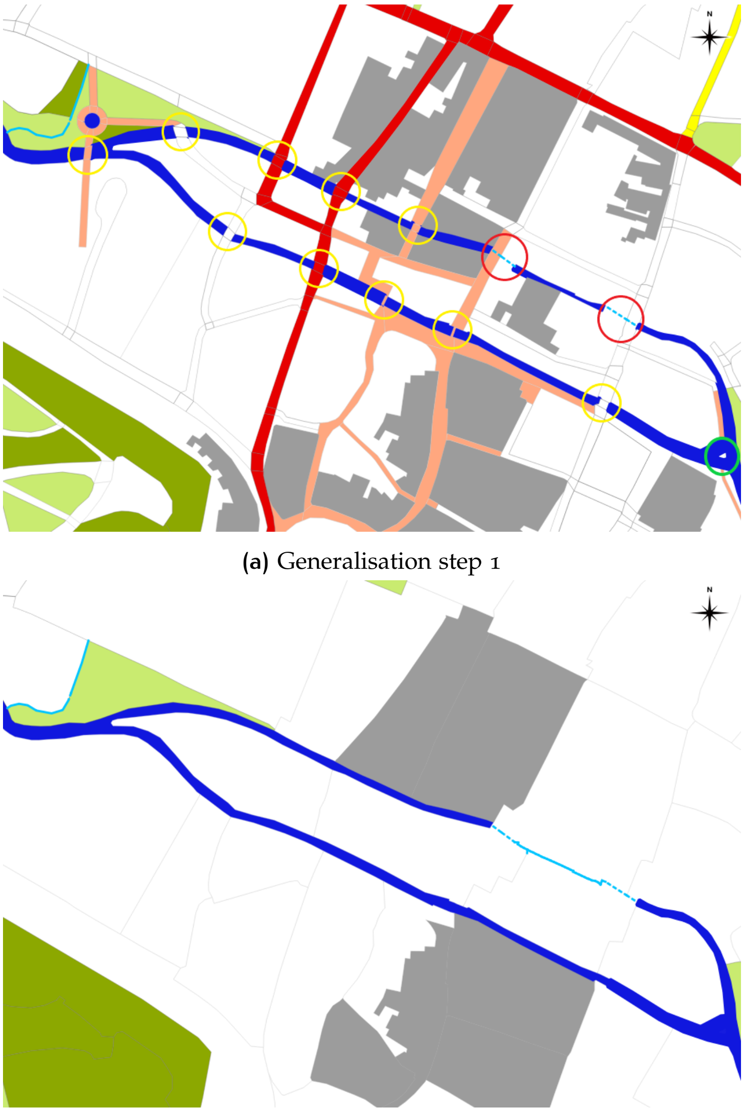


Figure 5.7: Number of connected components throughout the generalisation process.

The yellow circles in Figure 5.8a indicate that there is a hydrography face feature beneath the road feature. The red circles indicated a hydrography edge (which is a pipe in the real world) which is connecting two hydrography faces. The hydrography network is preserved well throughout the first 8000 steps of the generalisation process as can be seen in Figure 5.8b.

The green circle in Figure 5.8a indicates a terrain feature that is surrounded by hydrography features, in the generalisation process this feature is merged with the surrounding hydrography feature.

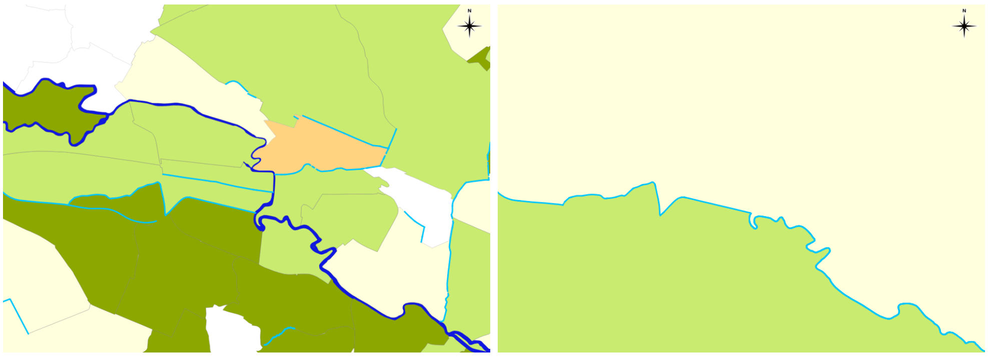
Figure 5.9 shows the difference between generalisation step **10000** and step **10350**. This is not an ideal generalisation as the original hydrography face feature is lost and a less important hydrography edge is preserved.



(a) Generalisation step 1

(b) Generalisation step 8000

Figure 5.8: Map at generalisation step **1** and step **8000**. The yellow circles in Figure 5.8a indicate that there is a hydrography face feature below the road feature. The hydrography network is preserved well between the two generalisation steps.



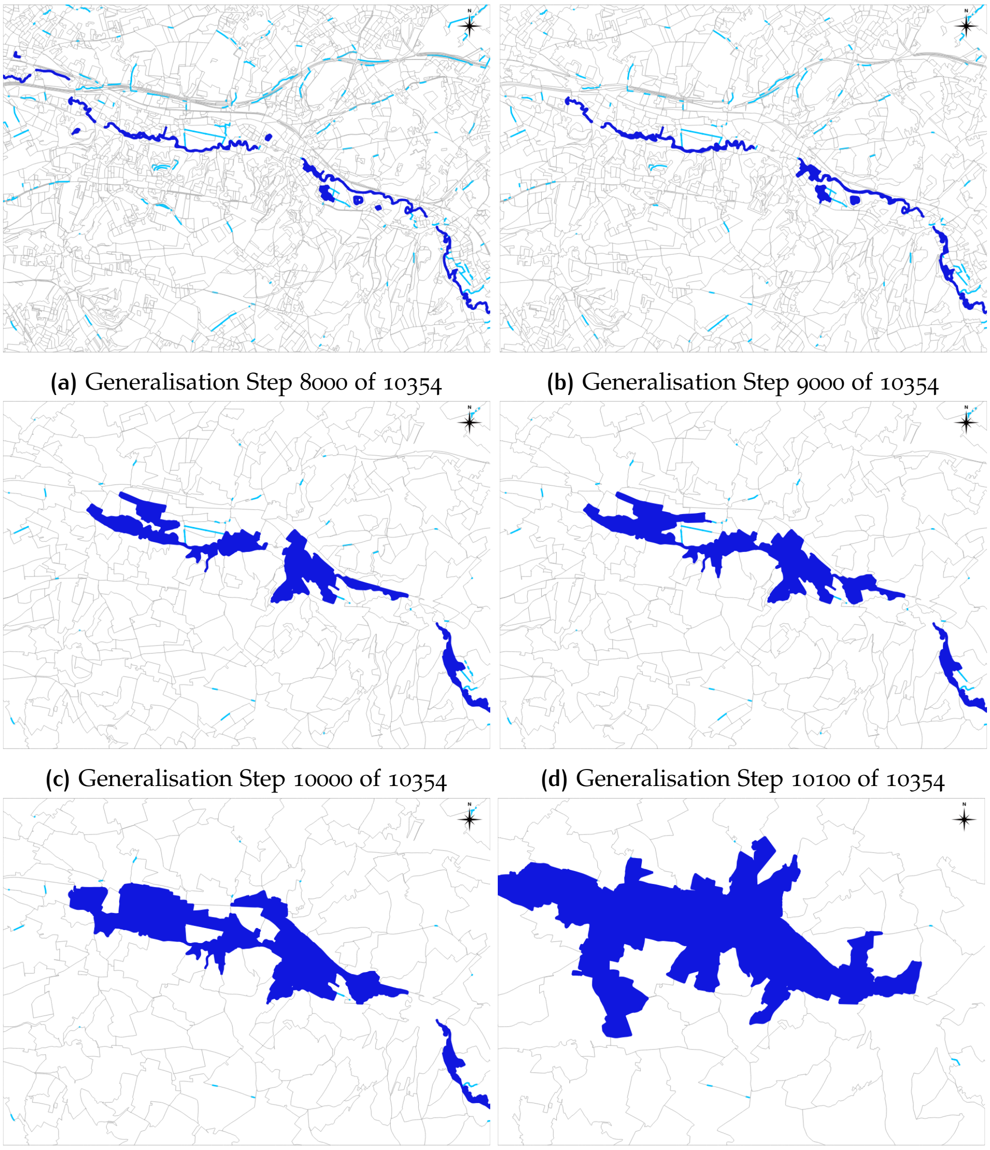
(a) Generalisation step 10000

(b) Generalisation step 10350

Figure 5.9: Map at generalisation step **10000** and step **10350**. Notice the hydrography face and edge features in Figure 5.9a and the same scene later in the process.

5.3 COMPARISON WITH TRADITIONAL VARIO-SCALE

The traditional vario-scale approach is also used to generalise the test data set so it can be used for comparison. The traditional vario-scale approach only uses the merge operation in the generalisation process. The result of the generalisation can be found in Figure 5.10. As hydrography features are often represented as long polygon features, they act as 'magnets' for smaller features next to them which will merge with the hydrography features making them larger and larger. As can be seen in Figure 5.10 the hydrography features start to 'eat' all other features and at the last step in the generalisation process the map will be one hydrography feature. After comparing Figure 5.3 and Figure 5.10 it is visible that there is a huge difference between the two generalisations of the same test data set. The generalisation process with the decisions for hydrography features makes sure that the hydrography feature are not acting as 'magnets' for other features and that the hydrography network is better preserved throughout the generalisation process. Hydrography features which are connected in the input data set are disconnected throughout most of the generalisation result with the traditional approach, see Figure 5.10a,b,c,d,e.



(a) Generalisation Step 8000 of 10354

(b) Generalisation Step 9000 of 10354

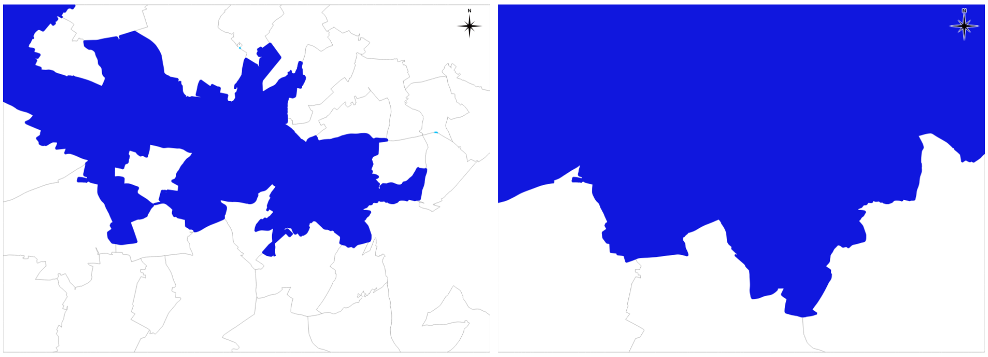
(c) Generalisation Step 10000 of 10354

(d) Generalisation Step 10100 of 10354

(e) Generalisation Step 10200 of 10354

(f) Generalisation Step 10300 of 10354

Figure 5.10: Continued on the next page



(g) Generalisation Step 10325 of 10354

(h) Generalisation Step 10350 of 10354

Figure 5.10: Situations at different steps and scales in the generalisations process. The subfigures illustrate how the traditional vario-scale structure evolves. Note that all the map are displayed at the same size to clearly show the effect of the generalisation process. In reality the maps at a higher step in the generalisation process should have been shown at a smaller scale.

Beside visual difference between the two generalisation methods there are also statistical/qualitatively differences. Figure 5.11 shows that the number of hydrography edges in the traditional approach gradually decreases, this is as expected because the merge operation does not take into account hydrography edges. The number of hydrography face features is more or less the same throughout the generalisation process for both methods.

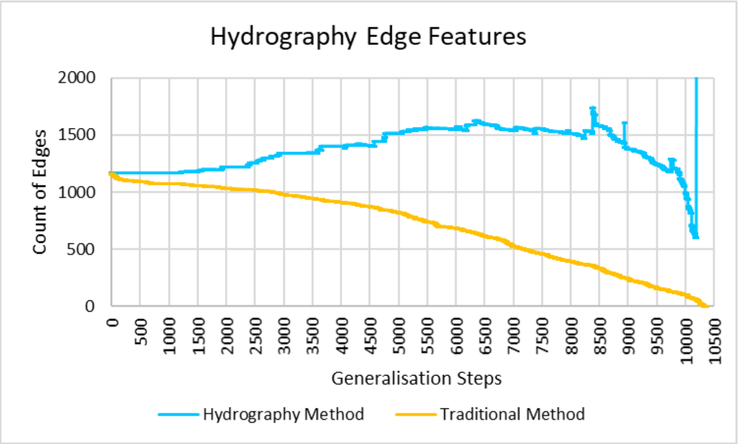
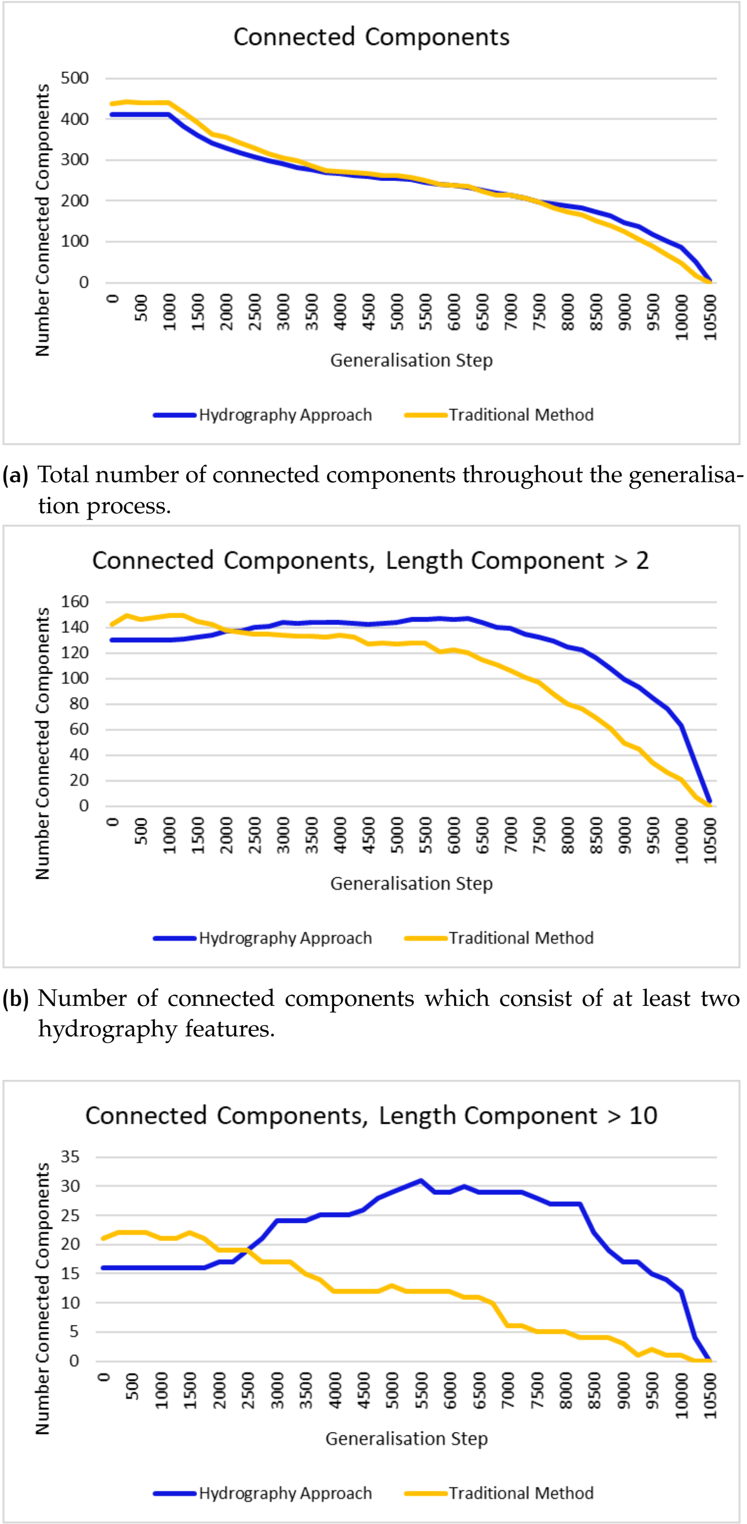


Figure 5.11: Comparison of the hydrography edges between the traditional vario-scale approach and developed approach with decisions for hydrography features.

The total number of connected components is more or less the same throughout the generalisation process for both methods, see Figure 5.12a. When the length of the components is also considered, which is the number of hydrography features connected to form a network, it appears that the vario-scale approach with the decisions for hydrography features has bigger hydrography networks throughout the generalisation process. This means that less hydrography networks are torn apart compared to the traditional approach, and thus that the hydrography networks are more meaningful.

The overall connectivity of the hydrography features throughout the generalisation process done with the decisions for hydrography is higher compared to the traditional vario-scale approach, see Figure 5.13.

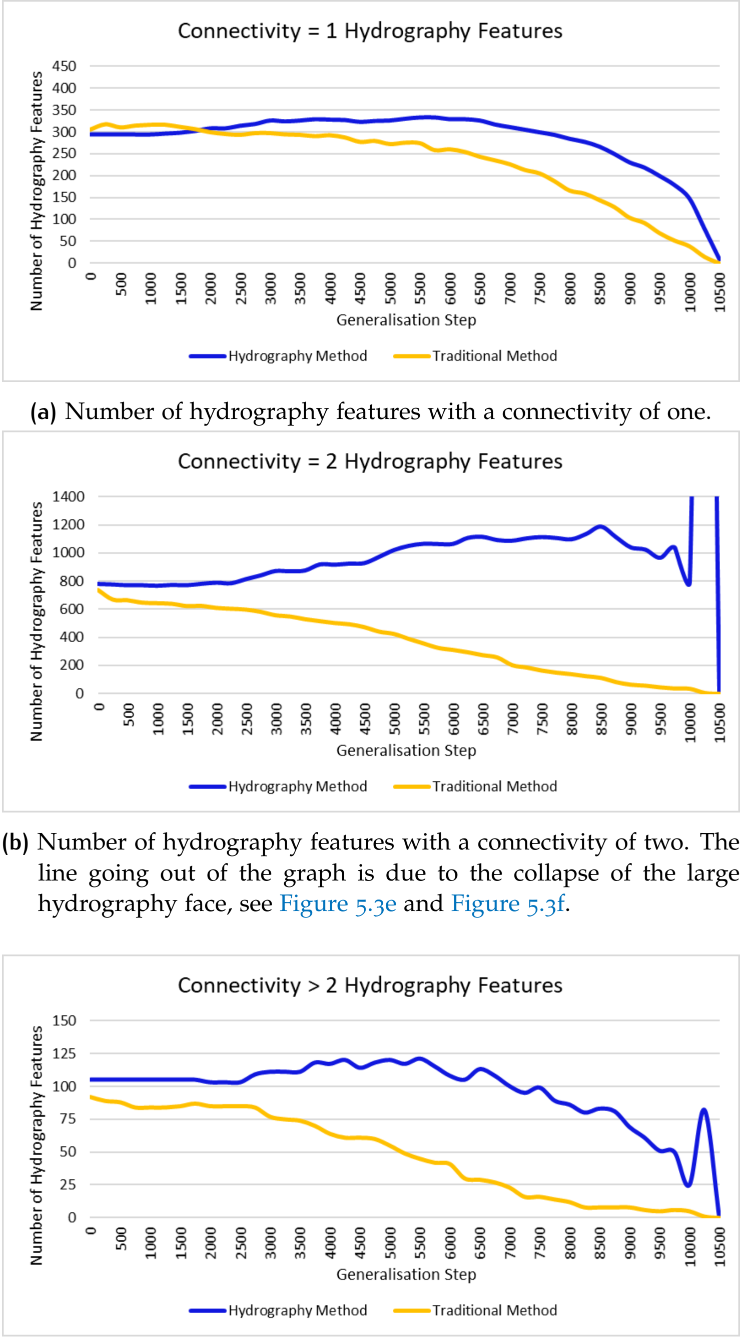


(a) Total number of connected components throughout the generalisation process.

(b) Number of connected components which consist of at least two hydrography features.

(c) Number of connected components which consist of at least ten hydrography features.

Figure 5.12: Comparison of the total number of connected components throughout the generalisation process.



(a) Number of hydrography features with a connectivity of one.

(b) Number of hydrography features with a connectivity of two. The line going out of the graph is due to the collapse of the large hydrography face, see Figure 5.3e and Figure 5.3f.

(c) Number of hydrography features with a connectivity of at least three.

Figure 5.13: Comparison of the connectivity of the hydrography features throughout the generalisation process.

6 CONCLUSION AND FUTURE WORK

The research is concluded in the CONCLUSION AND FUTURE WORK, Chapter 6. The conclusions are given in Section 6.1. Recommendations and future work are described in Section 6.2.

6.1 CONCLUSIONS

The vario-scale concept has proved an alternative for the well known discrete multiscale solution. However the vario-scale concept is still ongoing research and requires further development e.g. how to create better content for vario-scale maps? Šuba et al. [2016] introduced line features in the vario-scale concept and used it to better incorporate road networks in the vario-scale solution. This is used as the starting point for research on the better incorporation of hydrography networks in the vario-scale concept. So that hydrography networks are more meaningful throughout the scales without being torn apart in the generalisation process.

In the pre-processing of topographic input data for the initial tGAP data structure an extra attribute is created in which the feature class of a hydrography feature is stored if it is located beneath another feature e.g. a road feature (bridge). Compared to earlier initial tGAP data structures not only area features, but also line features are used.

Hydrography networks are created by iterating over all hydrography features in the data structure and storing the hydrography objects and the links which indicate a connection between two neighbouring hydrography features. Based on the links, the connected components in the hydrography graphs are found. There can be multiple hydrography networks in the data structure.

To determine the flow direction in the hydrography networks two DEMs are used, namely the Shuttle Radar Topography Mission (SRTM) and a DEM based on the Actueel Hoogtebestand Nederland (AHN3). After several experiments in which the average height values or the height values for the nodes of an hydrography feature were added to the feature it appeared that this was not sufficient enough for determining the flow direction in the hydrography networks in the real world test data set. The alternative approach for constructing flow direction is to find the outlet point of the hydrography network, if this point is found the Strahler Order can be computed which also indicates the flow direction in the hydrography network. If the outlet point cannot be found flow direction is not used in the generalisation process. Determining flow direction appeared to be not feasible for the real world test data set, and is therefore not used.

Generalisation decisions have been formulated and implemented in the vario-scale approach. If an hydrography feature is selected it is either merged with another hydrography feature or splitted and the area divided among the neighbouring features leaving a hydrography line behind. If a non-hydrography feature is selected in the generalisation process it is merged with another non-hydrography feature, in determining which other non-hydrography feature to merge with the common boundary with the neighbours is checked for absence of hydrography edges. If hydrography edges are present another neighbour will be searched for with a non-hydrography common boundary, if not there the merge operation will take place with the neighbour with the shortest common hydrography edge as boundary.

The generalisation results with the implemented generalisation decisions for hydrography features does improve the generalisation results for a vario-scale basemap. In the traditional vario-scale method the hydrography network is not preserved as the hydrography features act as 'magnets' and 'eat' all other features in the generalisation process. After generalising the real word test data set of the municipality of Valkenburg, the Netherlands the river 'de Geul' is preserved throughout the generalisation process, which shows that hydrography networks can be incorporated in the vario-scale concept for creating a vario-scale basemap with the special focus on hydrography networks.

Contribution: This MSc research contributed to the ongoing research on the vario-scale concept that has already been carried out for a few years by the GIS Technology Group which is part of the staff of the MSc Geomatics at TU Delft. In this research the advanced treatment of hydrography features in the vario-scale concept has been introduced. This can be used in the further development of the vario-scale concept and applied to other networks like e.g. roads, rail and utility networks.

6.2 FUTURE WORK AND RECOMMENDATIONS

In this section several suggestions and recommendations are provided for future research on the vario-scale concept.

- In the current solution for the generalisation of road networks and the developed generalisation for hydrography networks in the vario-scale concept, many decisions and operations have been hard coded in the generalisation software. This makes the vario-scale approach less generic because variables have to been changed in the code for usage of other data sets. Research is needed on how to make these generalisation tools generic so they can be used with all input data sets. An option could be that the user creates an detailed database table as input for the process which indicates e.g. what feature class represents what kind of objects and which objects/feature class are most important for the map the user wants to generalise.
- Line simplification could be included in the generalisation process, research is needed for the best strategy to do this.
- Also other data sets should be tested with the developed generalisation for hydrography features. These data sets can e.g. have a smaller or a larger starting scale.
- Generalisation of hydrography networks and roads networks need to be integrated. How to deal e.g. with hydrography features running parallel with road features?
- Road networks also have a certain flow direction could this be introduced in the generalisation of road network features to give a better generalisation result?
- Processing of large data sets with the developed method for the generalisation of hydrography networks. Large data sets will be processed in chunks, research is needed on how to exchange the hydrography network information between the different chunks.
- Labels are important for the readability of a map and could be included for the hydrography features.
- Every generalisation step starts with picking the least important face for which an generalisation operation is performed. This selection could be improved.

Instead of faces, also the selection of edges should be possible. A more advanced function for determining the importance using e.g. size, feature class, connectivity and other semantic information could be considered and implemented.

- The aim of this research is a vario-scale base map, however is it also possible to make generalised maps for a different purpose? Navigability on waterways can be such a map, which also needs extra generalisation decisions and information on e.g. the depth of the waterway, the temporal aspect (waterways can disappear sometimes of the year).
- In the Netherlands many pumps and windmills exist altering the natural flow of water. How to include this information in the generalisation process?

BIBLIOGRAPHY

Ai, T., Huang, L., Yan, X., and van Oosterom, P. (2017). A matrix-based structure for vario-scale vector representation over wide range of map scales: the case of river network data. *International Journal of Geographical Information Science*.

Ai, T., Liu, Y., and Chen, J. (2006). *The Hierarchical Watershed Partitioning and Data Simplification of River Network*, pages 617–632. Springer Berlin Heidelberg, Berlin, Heidelberg.

Ai, T. and van Oosterom, P. (2002). GAP-tree extensions based on skeletons. In *Advances in spatial data handling*, pages 501–513. Springer.

Amenta, N., Bern, M., and Eppstein, D. (1998). The crust and the β-skeleton: Combinatorial curve reconstruction. *Graphical Models and Image Processing*, 60(2):125–135.

Douglas, D. H. and Peucker, T. (1973). ALGORITHMS FOR THE REDUCTION OF THE NUMBER OF POINTS REQUIRED TO REPRESENT a DIGITIZED LINE OR ITS CARICATURE. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10(2):112–122.

Garca, R., de Castro, J. P., Verd, E., Jess, M., and Mara, L. (2012). Web map tile services for spatial data infrastructures: Management and optimization. In *Cartography - A Tool for Spatial Analysis*. InTech.

Gold, C. and Snoeyink, J. (2001). A one-step crust and skeleton extraction algorithm. *Algorithmica*, 30(2):144–163.

Haunert, J.-H. and Sester, M. (2007). Area collapse and road centerlines based on straight skeletons. *GeoInformatica*, 12:169–191.

Horton, R. E. (1945). Erosional development of streams and their drainage basins; hydrophysical approach to quantitative morphology. *Geological Society of America Bulletin*, 56(3).

Kraak, M. (2010). *Cartography: Visualization of Geospatial Data*. Harlow ; New York : Prentice Hall, 3rd ed edition.

McAllister, M. and Snoeyink, J. (2000). Medial axis generalization of river networks. *Cartography and Geographic Information Science*, 27(2):129–138.

Meijers, B. M. and van Oosterom, P. J. M. (2011). THE SPACE-SCALE CUBE: AN INTEGRATED MODEL FOR 2d POLYGONAL AREAS AND SCALE. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXVIII-4/C21:95–102.

Meijers, M., Savino, S., and van Oosterom, P. (2016). Splitarea: an algorithm for weighted splitting of faces in the context of a planar partition. *International Journal of Geographic Information Science*, 30(8):1522–1551.

Regnauld, N. and Mackaness, W. A. (2006). Creating a hydrographic network from its cartographic representation: a case study using ordnance survey MasterMap data. *International Journal of Geographical Information Science*, 20(6):611–631.

Savino, S., Massimo, R., Fabio, C., Giovanni, L., and Marco, R. (2011). Model generalization of the hydrography network in the CARGEN project. In *Lecture Notes in Geoinformation and Cartography*, pages 439–457. Springer Berlin Heidelberg.

Strahler, A. N. (1952). Hypsometric (Area-Altitude) Analysis of Erosional Topography. *Geological Society of America Bulletin*, 63(11):1117.

Šuba, R. (2017). *Design and development of a system for vario-scale maps*. PhD thesis, Delft University of Technology, Faculty of Architecture and the Built Environment.

Šuba, R., Meijers, M., and van Oosterom, P. (2016). Continuous road network generalization throughout all scales. *ISPRS International Journal of Geo-Information*, 5(8):145.

Tutorials Point (2017). Software development life cycle. http://www.tutorialspoint.com/software_engineering/software_development_life_cycle.htm. Last accessed on June 17, 2017.

Uitermark, H., Vogels, A., and van Oosterom, P. (1999). Semantic and geometric aspects of integrating road networks. In *Interoperating Geographic Information Systems*, pages 177–188. Springer Berlin Heidelberg.

van Altena, V., Bakermans, J., Lentjes, P., Nijhuis, R., Post, M., Reuvers, M., and Stoter, J. (2014). Generalisation of a 1:10k map from municipal data. In Burghardt, D., Duchesse, C., and Mackaness, W., editors, *Proceedings of the 17th ICA Workshop on Generalisation and Multiple Representation*, pages 1–9.

van Altena, V. and Stoter, J. (2016). Context-aware thinning of artificial water networks for map generalization. *GI_Forum*, 1:12–29.

van Oosterom, P. (2005). Variable-scale topological data structures suitable for progressive data transfer: The GAP-face tree and GAP-edge forest. *Cartography and Geographic Information Science*, 32(4):331–346.

van Oosterom, P. and Meijers, M. (2013). Vario-scale data structures supporting smooth zoom and progressive transfer of 2d and 3d data. *International Journal of Geographical Information Science*, 28(3):455–478.

van Oosterom, P. and van den Bos, J. (1989). An object-oriented approach to the design of geographic information systems. *Computers & Graphics*, 13(4):409–418.

Vermeij, M., Van Oosterom, P., Quak, W., and Tijssen, T. (2003). Storing and using scale-less topological data efficiently in a client-server DBMS environment. In *7th International Conference on GeoComputation, Southampton*.

Wikimedia Commons (2007a). Directed graph. https://commons.wikimedia.org/wiki/File:Directed.svg. Last accessed on January 13, 2018.

Wikimedia Commons (2007b). Undirected graph. https://commons.wikimedia.org/wiki/File:Undirected.svg. Last accessed on January 13, 2018.

Wikimedia Commons (2011). Flussordnung (strahler). https://commons.wikimedia.org/wiki/File:Flussordnung_(Strahler).svg. Last accessed on May 04, 2017.

COLOPHON

This document was typeset using LaTeX. The document layout was generated using the `arsclassica` package by Lorenzo Pantieri, which is an adaption of the original `classicthesis` package from André Miede.

TU Delft