4 Techniques for better vario-scale map content

The previous chapters covered the research where the vario-scale structure has been introduced. The main aim of the research was general functionality, performance and optimization. So far, the technical aspects had higher priority than the map content. Therefore, this chapter focuses on improving our development kit for generating vario-scale content. It presents a strategy to provide good cartographic results throughout all scales and properly stored in the structure. First, Section 4.1 specifies our target. Section 4.2 presents solutions of other researchers. Section 4.3 introduces concepts and tools which are used later in newly designed process. This is demonstrated on road features in Section 4.4. The section presents the generalization approach for the whole scale range from large scale, where roads are represented as area objects, to mid and small scales, where roads are represented as line objects. In our suggested gradual approach even for one road the representation can be mixed, both area and line, which may provide better transition phase and better impression for the user. This is shown in Section 4.5. Data density is still an issue, therefore, Section 4.6 suggests solution for preserving different feature density in the map, *e.g.* high feature density in the cities and sparsely occupied rural regions. Finally, Section 4.7 concludes the chapter and presents some open questions.

Own publications

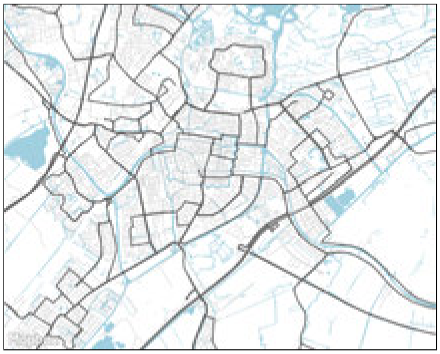
This chapter is based on the following own publications:

- Šuba, R., Meijers, M., Huang, L., and van Oosterom, P. (2014b). Continuous Road Network Generalisation. In *Proceedings of the 17th ICA Workshop on Generalisation and Multiple Representation, Vienna, Austria, September 23, 2014*, pages 1–12.
- Šuba, R., Meijers, M., and van Oosterom, P. (2015). Large scale road network generalization for vario-scale map. In *Proceedings of the 18th ICA Workshop on Generalisation and Multiple Representation, Rio de Janeiro, Brazil, 21 August, 2015*, pages 1–10.
- Šuba, R., Meijers, M., and Oosterom, P. v. (2016b). Continuous road network generalization throughout all scales. *ISPRS International Journal of Geo-Information*, 5(8):145.

§ 4.1 Objectives

The vario-scale concept has a wide range of application from navigation software, through desktop GIS to mobile applications. Recent development was focusing on on-line applications with effective vector data transfer, see Section 3.7. It is based on the idea that if features are generalized in small steps, smooth transitions between subsequent object representations can be derived (Sester and Brenner, 2005; Midtbø and Nordvik, 2007; Chimani et al., 2014; Huang et al., 2016). This provides a different approach to conventional discrete scale maps on the Internet. Those discrete maps have abrupt changes between scales, which can lead to disorientation, result in confusion and eventually the frustration of map users. Figure 4.1 shows a detailed road network in one map scale in comparison to the sparse network at the next available scale.

The development of the vario-scale data structure reaches the state where we have sufficient tools to generate meaningful map content for tGAP structure. We have well-know remove/merge, collapse/split and line simplification operations. This gives us options to get reasonable map content. The main requirement of the process stays the same over the project research. We want to create meaningful, reasonable and 'cartographically correct' maps. In more detail all generalization actions during the tGAP creation process should lead to good quality results from the cartographic point of view, *i. e.* Every intermediate map gives a proper impression to the map reader. Nevertheless, the generation of meaningful output of continuous generalization is still a challenge.



(a)

(b)

FIGURE 4.1 An example of map fragments at two slightly different scales; (a) at a larger scale, (b) at a smaller scale. Note that only the water and road network are displayed for these two scales, which were obtained directly from the original multi-scale database without any content modification (source: OpenStreetMap, styling: Mapbox Studio).

Our ambition is to develop a better strategy to create vario-scale representation. In this chapter, our working assumption is that by capturing the whole generalization process from large-scale input data, being progressively collapsed from areas (large scale) and/or merged into lines (small scale), it is possible to obtain a better representation with the following properties:

- Better preserved (road) network connectivity.
- Changes in geometry and/or classification are gradual.
- Small details are gradually eliminated (unimportant dead-ends (cul-de-sac) are removed and no new dead-ends are created).
- Road network semantics are taken into account, and the overall map impression is emphasized (remains during generalization).

This may give in theory a better impression to the user, *e.g.* while zooming in and out. Note that user testing should confirm that, and it will be investigated more in Section 6.5.

A property of the transition phase from large to small scale is that area and line representations are mixed together, *e.g.* second order roads at the same map scale may be represented partially by areas and partially by lines. Thus, by design our approach does not rely on homogeneous road segment representations. This is a non-trivial issue for conventional data structures and requires specific data modelling and modification of the data structure, both of which will be explained later in the text. In previous papers on vario-scale data using the tGAP structure (van Oosterom, 2005, p. 345) or (Meijers, 2011, p. 199), it has been mentioned several times (in the 'Future Work' section) that it should be possible and highly desirable to include features that have a line representation to improve the content of the map significantly. Now, for the first time, this is being realized and tested with real data.

To summarize briefly, our main goal is to generate good cartographic results representing all scales in the tGAP structure, to have better support and data content for smooth zooming.

To accomplish such a goal, we have the following requirements based on our vario-scale concept:

I. To introduce line feature representations.
II. To deal with mixed areas and line representations.
III. To generalize in small steps.
IV. To use an area partition as input.
V. To preserve the meaning of road network.

§ 4.2 Linear versus area representations / Related work, road network generalization

Our tGAP tools are in principle designed in a very generic way, however, in this chapter will be demonstrated mainly on road network data because the roads (similarly rivers and water channels) as linear/infrastructure objects are the 'backbone' of many map types. They improve the legibility of maps and help users to orient and recognize the depicted real-world situation more easily. On top of that, road network generalization forms a prerequisite for all other topographic generalization action (operators) and is thus a fundamental operation in the overall process of map and database production (Weiss and Weibel, 2014).

Various earlier attempts have explored ways of generalizing road networks as an important part of generalization process. The road processing has been extensively studied where at a given scale, two main representations of the roads may be found: linear and area representations. On large-scale maps, for Belgium, the Czech Republic, the Netherlands, the United Kingdom and other countries, a road segment is represented by area geometries. Together, the collection of road areas represents the road network. These areas form an implicit road network graph comprised of edges and nodes. At smaller scales, the road segments are represented by line geometries. These lines correspond more directly to the edges in the road network graph. At smaller scales (with road segments directly represented as edges), the road network generalization can have the emphasis on: (1) the linear road representation itself; or (2) the areas between the roads as regions bounded by minimal road loops or cycles, *e. g.* containing built-up area, forest or terrain. The linear emphasis approach considers the network as a set of linear connected elements, while the regions emphasis approach concentrates on the 'space between' the roads. Sometimes, the term 'area partition' (Edwardes and Mackaness, 2000) or 'mesh density' (Thomson and Richardson, 1999; Chen et al., 2009; Li and Zhou, 2012) is used. For smaller scale road network generalization, where road sections are represented as lines, both views (either linear or regions emphasis) have their own advantages and disadvantages (Edwardes and Mackaness, 2000).

Moreover, at the smaller scales there are two main perspectives within the approach with a focus on the linear network representation: (1) strokes; and (2) segment generalization. The stroke-based generalization groups the road segments into longer lines, which may cross without explicitly intersecting. The decision to intersect may be based on some criteria, such as geometry (angle between segments), topology (node degree of two), attribute or classification (Zhou and Li, 2012; Thomson and Richardson, 1999). In the segment-based generalization, the road segments (from junction to junction, where the topological degree of end nodes is greater than two) are the smallest atomic elements for removal.

The main advantage of using strokes is evident: it preserves information about the connectivity between segments. This indicates that the stroke-based approach can be a useful generalization tool. However, Turner (2007) points out that segment analysis (the creation of a segment map as known by the space syntax community) can give comparable or better output compared to a stroke-based method and even produces more meaningful results, *e. g.* better in correlation with observed vehicular flow.

From computational perspective, the strokes approach is used for all road segments present in the dataset, but for our purpose only local information is needed, *e. g.* to find the road segment with best continuity to determine the new classification. Additionally, our road network is changing dynamically during the process and it would be useless to maintain global information about strokes. Therefore, we only view a merged set of road segments as a stroke to determine its new classification, when there is no 'clearer' alternative to do this.

There has also been continuous map generalization research concerning road network, see Section 2.3. Another truly continuous solution is proposed by Li and Zhou (2012), including experiments with real data and an extensive evaluation. They compare both approaches; strokes and mesh density, and combine them in their so-called *integrated* method to create a universal solution for road network using the advantages of both approaches. Their solution generates two separate linear and areal hierarchies, combining them together to provide continuous multi-scale representations of a road network. This rather complex solution is based on the omission of features, with the analysis of what should be eliminated. This analysis takes place only at the beginning of the process. Changing parameters of the generalization process are not considered; *e. g.* rules/parameters adapted during different phases of the generalization. On the other hand, the performance test looks promising and suggests that the approach is quite feasible and will be good for on-the-fly use.

§ 4.3 Concepts and definitions

A processing strategy for generalization throughout the scales is introduced in Section 4.4. Before doing so, terminology and specific tools are explained in this section to be able to better explain our strategy. Note that Section 4.3.1 and Section 4.3.2 explain theoretical concepts first. Then Section 4.3.3, 4.3.4 and 4.3.5 will present more practical tools implemented in our developing kit.

§ 4.3.1 Granularity

Continuous generalization requires geometric changes (big or small) between generalization steps in the process. We call the number of features (data) changing in one generalization step the *granularity* (Šuba et al., 2015), and we distinguish the following levels:

- *course granularity*, when all features are involved at once, *e. g.* all roads are omitted.
- *medium granularity*, when all features of a certain class or subclass are processed, *e. g.* all local roads with speed limits are removed.
- *fine granularity* when one single feature is processed, *e. g.* one dead-end road is removed.
- *finest granularity*, when a part of a single feature is involved, *e. g.* one road segment is removed.

Since we aim at a more gradual transition without significant modification in geometry, the changes should be small. In our case, *finest granularity* is then optimal. This guarantees that the changes are as small as possible, aligning well with the vario-scale concept.

We apply merge/remove, collapse/split and simplification operations for these object parts, which can result in a road object composed from segments represented by mixed lines and faces; see Figure 4.2. From a traditional cartographic point of view this might seem less favorable, but from a vario-scale point of view, it is desirable. The differences in the representation can be compensated in the visualization by applying proper styling, when the line segment is represented by lines of the same thickness equal to the width of the adjacent area segment. Furthermore, the fact that the generalization process is performed in small steps leads to simpler problems, which are easier to compute or implement. On top of that, the history of steps is stored explicitly, and this implies that the links between generalized and original objects are present. Often, these links are missing in multi-scale implementations.

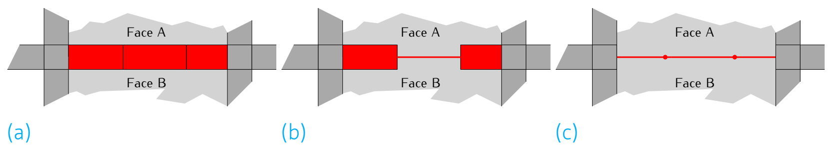


FIGURE 4.2 Side effect of the gradual transition from one scale (a) to another (c). For some reasons (perhaps different attributes; *e.g.* road surface type, name or speed limit), the red road consists of three parts. To achieve a gradual transition, the individual parts are generalized separately. It changes the representation from areal at the most detailed scale (a) to semi-linear at the 'halfway' scale (b) to linear at the final scale (c). Be aware of the fact that the complete road in one moment of the process is represented by both areal and linear parts at the same time. Note that one of the consequences is a topological change, where *Face A* and *Face B* become adjacent.

§ 4.3.2 Level of abstraction

The input datasets that are currently supported within the tGAP structure are modelled as a two-dimensional polygonal map, *i.e.* as a partition of the plane in a geometric sense, without gaps and overlaps. As a result, the data structure contains only topological primitives; nodes, edges and faces, where one area object in the map corresponds to just one topological face. The same is true for roads in the input dataset, these are represented by faces only (areas).

Besides classification and the planar area partition, there is more semantic information *implicitly* present in the large-scale input map, such as the linear networks (road infrastructure or rivers and water channels). These linear networks are implicit in the input data, and we wish to preserve their natural meaning in the target map of the smallest scale as well. Even thought these features are part of a network they are often not explicitly modelled. Therefore, we will first make the implicit information about the role features play inside the network explicit, even when not given as input.

Figure 4.3 shows a simple example of such a network in the input (large scale) (see Figure 4.3a) and in the target small scale (see Figure 4.3d). The road network can be easily derived from both figures. When two road segments are both represented by areas (large scale), then they are incident when they share an edge. Where at least one of the road segments is not represented by an area, then they are incident when they share at least a point (node). However, it is not so simple to keep track of the linear network as the map changes from scale to scale in a more gradual way; see the intermediate steps in Figure 4.3b, c. Therefore, the same situation is captured in Figure 4.4, but this time, the *linear graph* of the road network is depicted for better understanding of the relationships among the road objects. As mentioned, the road segments can be *incident* with other road segments. Depending on the number of incidences, a road segment plays the role of junction (node) or connection (edge) in the linear network graph. From Figure 4.3a–d, we can see that the geometric embedding of the objects in the map changes, while the conceptual linear network graph stays the same. This gives us an effective tool for meaningful road network generalization throughout the scales. Figure 4.5 shows actual structure to represent this phenomen.

Keep in mind that in the following description, the road segments of different dimensions can be in an interaction with each other, *e.g.* a 1D line can be in an interaction with a 2D area road segment. Therefore, the road segments can be classified (especially in gradual changing scales) based on the number of other incident road segments and are included in the data structure as follows:

- A road segment is classified as an *isolated segment* when it has no other road segments incident.
- A road segment is classified as a *dead end* if it has exactly one other road segment incident. It is represented either by a face or by an edge in the topological data structure.
- A road segment is classified as *road connection* of the network when it is incident with exactly two other road segments. It is represented either by a face or by an edge in the data structure.
- A road segment is classified as *road junction* of the network when incident to more than two other road segments. It is represented either by a face or by a node in the data structure.

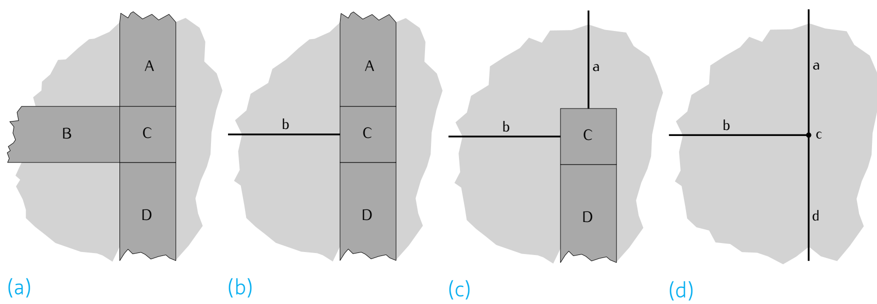


FIGURE 4.3 Example of road network generalization from the large scale (a), through intermediate step (b, c), to the final scale (d). Geometrical representation of the road segment/junction changes from 2D areas to 1D segments/0D point, but 'features' (semantic), and their role in linear network remain the same.

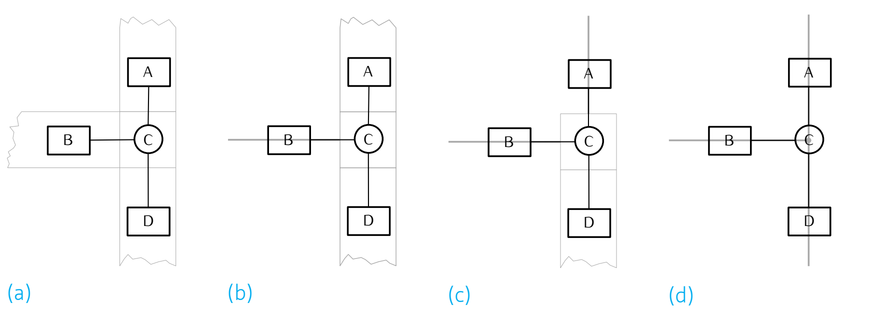
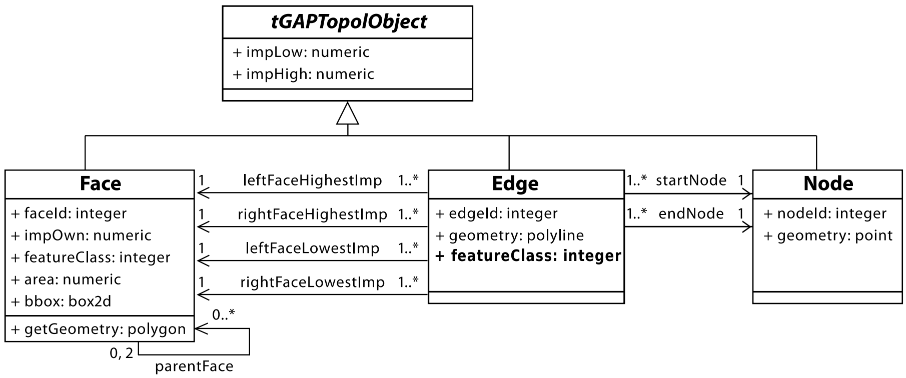


FIGURE 4.4 The linear network graph represents the same situation as Figure 4.3. It captures the topological relationships of road objects where a rectangle indicates a road connection and a circle a road junction.

In this way, we can associate any selected feature in the map at any stage of the process with a topological feature. The assumption related to the input data is that connections and junctions of a road network are well defined in the input dataset; see Figure 4.6a. However, when the input data do not conform to this assumption (see for an example



(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);
```

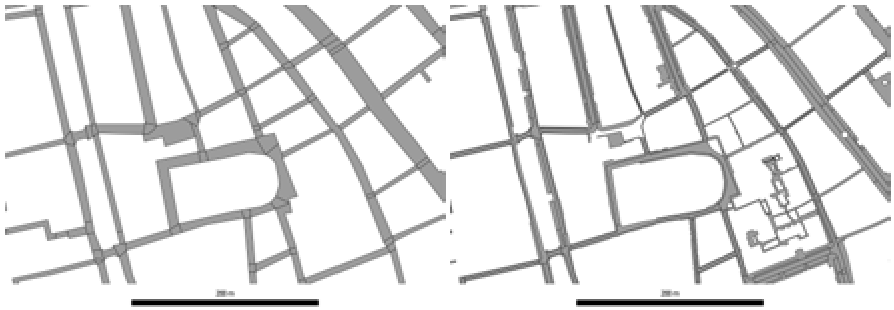
(c)Face hierarchy table

```
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 4.5 Road representations in UML diagram and the database tables for tGAP (It is inspired by (Meijers, 2011, p. 159)).

Figure 4.6b), an additional pre-processing step must be taken. This can be done by applying the constrained Delaunay triangulation to obtain properly-classified (connection or junction) area road segments, as proposed by Uitermark et al. (1999).



(a) (b)

FIGURE 4.6 Two map fragments of different possible input showing the city center of Leiden, the Netherlands: first (a), the topographic map (TOP10NL) intended for use at a 1:10,000 map scale; second (b), the BGT base map (in Dutch: Basisregistratie Grootschalige Topografie) intended for use at a 1:500–1:5,000 map scale. Note that only road network features are displayed.

§ 4.3.3 I-neighbours

Operations in our process are performed locally based on the global criterion of least important feature. To identify how many objects are involved in an operation we introduce $i - neighbours$. This is similar to a breadth-first search, *e. g.* $1 - neighbours$ checks only the direct neighbours of the chosen object, $2 - neighbours$ checks the neighbours of the direct neighbours, $3 - neighbours$ checks also the neighbours of the $2 - neighbours$, etc.

This topological measure can be used in two ways either for faces or for edges. Figure 4.7 shows the principals for each representation.

§ 4.3.4 Connectivity

When collapsed features (lines) interact, *e. g.* the classification for a newly-merged road must be picked, we define how well they fit together by computing the $connectivity$ value. For a specific road segment it is defined by how many routes between other segments go via this segment. More passing routes within segments means higher $connectivity$. This measure is known from space-syntax theory (Turner, 2007), which is used in analysis for urban planning and implemented in software, such as depthmapX[1]. There exists a modified version of the connectivity value, called *radius connectivity*. Here a geometric criterion ϵ, where $\epsilon \in \mathbb{R}$, is used. For the chosen segment only the

1 https://varoudis.github.io/depthmapX/

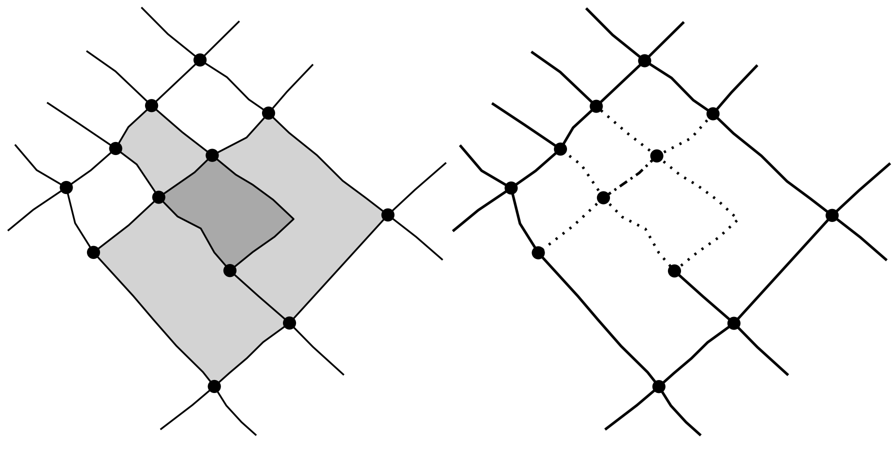


FIGURE 4.7 $1 - neighbours$ for a face (a), where the selected face is in dark grey. In (b), $1 - neighbours$ (dotted) for one selected edge (dash-dotted).

other segments within a radius ϵ of the segment are considered. It seems to hide the fact that segments on the edge of dataset have by definition a lower $connectivity$ value than segments in the middle of the dataset.

In our case, we compute the $connectivity$ locally. But instead of using a geometric radius ϵ, we use a topological measure, only considering $i - neighbours$, see Section 4.3.3. Figure 4.8 illustrates the computation of the $connectivity$. Let's assume that we want to identify the connectivity for selected road segment (in red). In this example, red segment has $connectivity = 6$ for the $1 - neighbours$ and $connectivity = 16$ for the $2 - neighbours$. Note that in our implementation other aspects such as classification or length of segments are involved.

§ 4.3.5 Deflection angle

To find the best continuing neighbouring road segment (for lines), we use a simple geometric principle, that is called the deflection angle. Figure 4.9a illustrates that this is the deviation from $180\,^\circ$ of the angle between two road segments. This is inspired by the strokes approach (note that many strategies for stroke building exist, cf. Zhou and Li (2012)). By using the deflection angle, we form locally a 'stroke' where a neighbouring segment appears to follow the same direction as the merged segment if the deflection angle is small. However, if the deflection angle is large (*e.g.* more $90\,^\circ$) the road segments will not be going in the same direction. We chose a $60\,^\circ$ threshold, if an adjacent, neighbouring segment has a deflection angle larger than this, it will not be considered.

Figure 4.9 demonstrates the principles of determining the new classification based on the deflection angle. Let's assume that the decision has been made to merge newly collapsed roads e_2 and e_3, see Figure 4.9b. There are three classification possible for the new segment e_{23}; 'A','B' and 'C', where 'A' is the highest. The segment e_1 has the

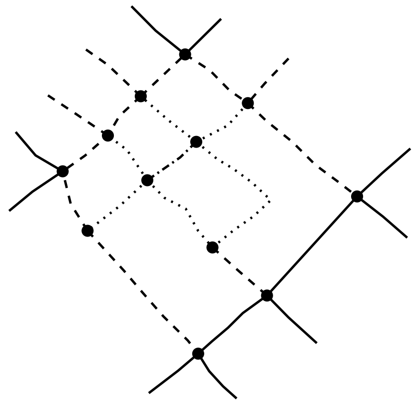
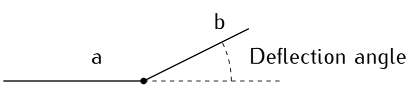
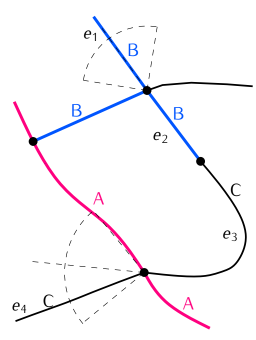


FIGURE 4.8 An example of defining connectivity for one selected road segment where only collapsed features (roads) exist. The dash-dot segment is selected. $1-neighbours$ are dotted. $2-neighbours$ are dashed.



(a) The principle of deflection angle. A smaller angle means a better connectivity.



(b) An example of merging two road segment. The deflection angle is indicated with the dashed lines.

FIGURE 4.9 Determining the classification of a merged segment, where the original segments are classified differently best deflection angle of all adjacent segments and its classification is used to determine the class of the newly merged segment. Note that the classification e_1 only indicates what the new classification should be and it thus not used directly.

One of the three possibilities can happen; class of $e_1 \geq$ maximum of classes e_2, e_3, class of $e_1 \leq$ minimum of classes e_2, e_3, or class of e_1 is between classes e_2, e_3. In the first case, the final class will be the maximum of the classes e_2, e_3. In the second case, the final class will be the minimum of classes e_2, e_3 and in the third case, the class will be based on the other aspects such as the *length* of segments e_2, e_3. In example 4.9b, the new merged segment will carry 'B', because $e_1[B] \geq e_2[B]$.

§ 4.4 Strategy for complete scale range of road network

This section introduces our proposed processing strategy for road network generalization. The generalization process to generate content for the vario-scale data structure is based on the tGAP-principle (find the least important object and merge it with a most compatible neighbor) extended using linear network knowledge. Note that the principle is designed in a very generic way, such that it is possible to mix different generalization operations. It is also integrated in the sense that all features for which the operations are performed are treated together (all features are geometrically integrated in the same planar area partition that is used as input). There are many design decisions in the development of this process, and we will label these as *DESIGN DECISION I* (with *I* a sequence number). Often, there are several alternative options, but based on our experience and some limited testing, we present our initial 'best guesses' for these decisions.

DESIGN DECISION

DESIGN DECISION 1: We distinguish in the creation of the vario-scale content for the data structure three classes of objects only: roads (sub-classified as either junction or connection), water and other objects. This will reduce the number of object types during the creation of the vario-scale to three, which makes decisions more transparent. Besides road network processing, this also allows us to treat water differently from other non-road classes. Note that the original classification of the other classes is kept and used later on in visualization (but not during the creation of the vario-scale structure). Alternative processing design decisions here could have been made: two classes (road, other: even simpler), three classes (roads, water and other: more refined, with two subclasses for water: junction, connection), more classes (as present on common topographic base maps, where the classification of these other classes is used; *e. g.* for selecting the most compatible neighbour to merge with).

At the beginning of the process, every face in the structure gets an importance value based on the type and the size (area) of the feature (in the initial large-scale map, there are only area features as a constraint). Note that the computation of the importance value can be refined; see Section 4.7. Based on the importance value for every face, the process starts picking one face after another and performs specific actions based on the type of the chosen face. The face with the lowest importance value is processed first.

DESIGN DECISION

DESIGN DECISION 2: As in the integrated data structure, both area and line representations of roads (and other features) are possible; an alternative to having just faces in the importance queue is also having line or node features in the importance queue.

Depending on the type of a face, there are the following processing options; see Figure 4.10.

- The selected face is a *road junction* and will be either merged with the adjacent road junction or preserved until all adjacent road connections are collapsed. If the former, then the face itself can be collapsed. If the latter, then the importance is raised and the face is put back in the queue (and will be processed later). Note that this can cause infinite loop, therefore an additional measures are taken such as queue reordering.

  *Design Decision 3:*Instead of postponing the processing of the junction, it would be possible to directly collapse it to a node (even if not all incident connections are collapsed). Design Decision
- The selected face is a *road connection* and will be merged with the adjacent road connection. If there is no such face, then it is collapsed to a line.
- The selected face is *water* and will be merged with another adjacent water face. If there is no such face, then it is collapsed to a line.
- The selected face is the *other* object and will be merged with an adjacent other object, if there is any; otherwise, the face is collapsed.

  Design Decision 4: If no adjacent face of type *other* is present, an alternative design decision instead of collapse would be to raise the importance and put it back in the queue. Later on, when one (or more) of the neighbour road faces have been collapsed to a line, then the selected face might have an adjacent *other* face. Design Decision

  Adjacent *other* objects with no collapsed roads lying between are the most optimal to merge with. If there is no such adjacent option, another object with the least important collapsed road (edge) between is selected. When the collapsed road lies between, the faces will be merged (and the collapsed road will be removed).

This recipe guarantees to generalize the roads in a meaningful way and is continuous for all faces in the structure. Roughly speaking the following happens with the roads: Initially, the area road segments are collapsed, and later on, the merging of the other areas takes place. To which neighbour the other area is merged depends on the edge between. If there is no collapsed (line) road segment in between, this has preference. If all edges represent collapsed road segments, then the least important one is selected, and this decides with which neighbour area it should be merged. With this type of merge, the unimportant linear road segments are 'automatically' removed, as well.

It is important that the least important collapsed road (line) is determined by looking at its classification, the local configuration (connectivity) and length. Specifically, for every potentially collapsed road, we look at its classification first. If this gives a 'winner', we pick this road as the least important collapsed road. If this results in a tie (collapsed roads having the same classification), we compute for every collapsed road a connectivity value. The connectivity value for one road is defined by how many routes between other roads go via this road, with more passing routes meaning higher connectivity, see Section 4.3.4. The least important collapsed road is the one with the lowest connectivity value. As a last resort, in the case of roads also having the same connectivity

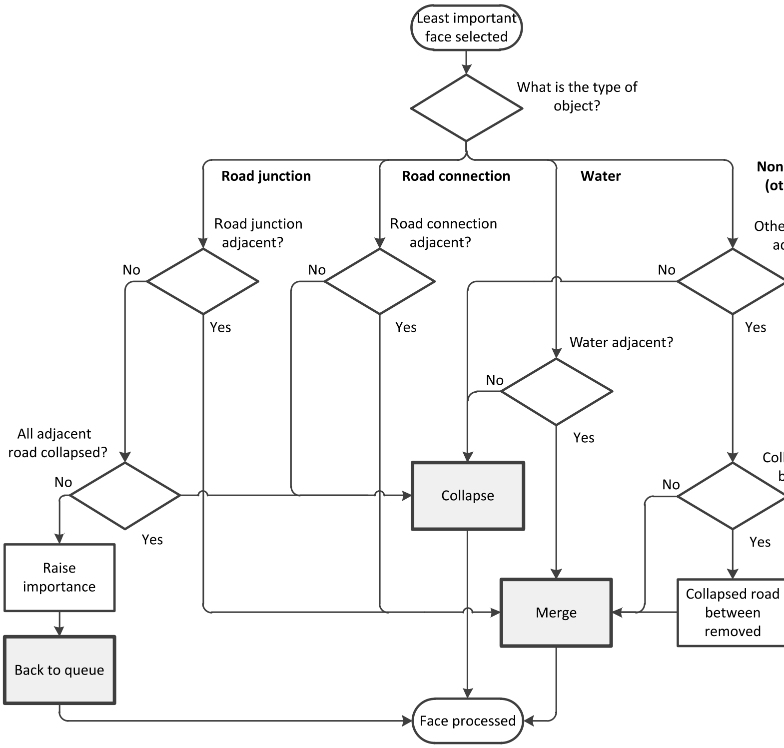


FIGURE 4.10 The way one generalization step is performed. Note that the road area will never merge with a non-road (other) area.

value, the shortest of the collapsed roads is selected as the least important one. An in-depth description is given in (Šuba et al., 2014b), where this has been extensively tested.

Note that another (non-road) area is collapsed if and only if it is completely surrounded by areas of roads, *e.g.* a face of grass is between two faces of road (grass strip between lanes of a highway). This is a rare case where the collapse of a non-road feature is the most favourable. Another option would be to return the non-road face back into the queue and wait until at least one road nearby is collapsed (this is *Design Decision 4*). However, the collapse operation is preferred in this case, because it makes sense to assign the parts of this unimportant face to the neighboring road faces, and it will reduce the number of faces by one.

Design Decision

Design Decision 5: In the beginning of this section, we defined road junctions and road connections. During the generalization process the configuration changes, which gives two options: (1) faces keep their original road (junction/connection) subclassification even if it is in contradiction with our definitions; or (2) faces are reclassified when needed in order to remain consistent with our definitions for junctions and connections. We opt for the first option, because this results in a slightly better vario-scale cartographic quality according to our visual inspection; *cf.* Section 4.5.3.

The above-described iterative generalization proceeds in steps. Normally, there is one face fewer after every step, and the number of faces never increases. There is some delay when a road junction area still has a road connection area as a neighbour, causing the processing of the junction to be postponed. However, the neighbouring road connection areas will all be collapsed to lines at a certain moment in the process, and after that, the delayed processing of the road junction areas can take place.

§ 4.5 Experiences with a data set with terrains and integrated roads

This section addresses the cartographic quality (Section 4.5.1), the quantitative analysis carried out (Section 4.5.2) and additional discussion items (Section 4.5.3).

§ 4.5.1 Cartographic quality

For our experiments, we loaded a subset of the Dutch topographic map (TOP10NL) intended for use at 1:10,000. Two following regions were used:

- A rural region, area of 7×7 km with 11,300 faces as input; see Figures 4.11 and 4.12.
- A city center (of the city of Leiden). It is a 1×1 km region with 19,400 faces as the input; see Figure 4.13.

These datasets are provided as simple feature polygons, where terrain, water and road network layers together form a complete planar area partition. The road segments are present in the dataset with geometries for road junctions and connections. The layer of buildings lies on top of those layers. First, we 'fused' all data layers together and created a planar partition as input that we converted to a topological data structure with the help of software FME[2]. At this stage, all objects are represented by areal geometries. Then, this input is processed into a vario-scale structure with the help of merge/remove and split/collapse (Meijers et al., 2016) generalization operations. Line simplification could be included in the process; see Figure 4.14c, d. However, this is not the case at this stage of the research process in order to see the features' geometry without any additional effects.

Figure 4.11 shows the impact of our method on real data. It demonstrates that by making small generalization steps, we got incrementally a simpler map. At intermediate scales two representations for road objects (areas and lines) are mixed.

Figure 4.12 presents a sequence of maps for a small part of the rural region retrieved from the tGAP structure as generated by our algorithm. It demonstrates the outcome of the algorithm (vario-scale content), but it does not correspond to the correct user impression as the scale is fixed in this figure and vario-scale map use should be an interactive experience during zooming. At least it gives an impression of the content of the different map scales. Figure 4.13 shows a proportional re-sized map sequence to give a better impression of how a user should perceive the derived maps.

2 http://www.safe.com/fme/

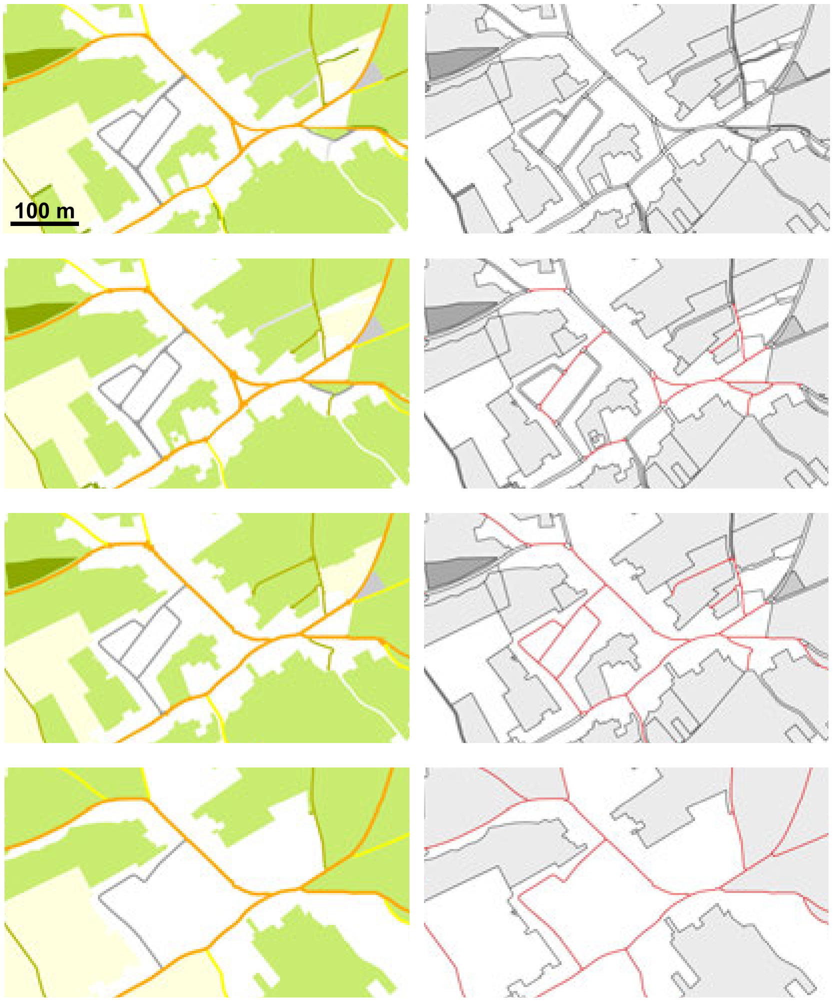
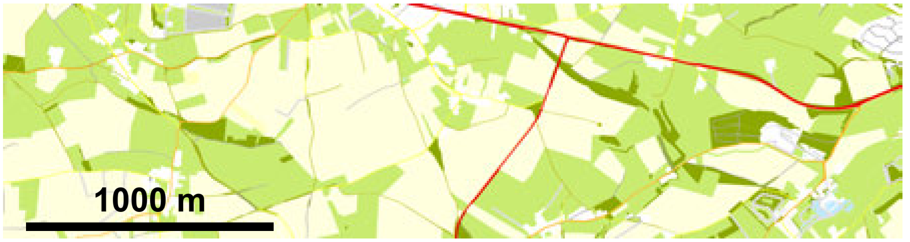
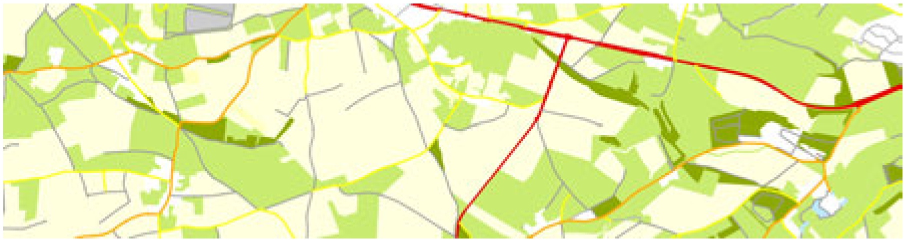


FIGURE 4.11 Detailed situation throughout the scales. All maps are at the same scale (and the exact scale is not very relevant). It illustrates how the structure evolves. The top maps are at the input scale, where all objects are represented by areas. The intermediate scales are in the middle, where the representation of roads is mixed (by areas and by lines) and generalized scales are at the bottom, where roads are represented only by lines. The styled map (left) corresponds to the situation in the tGAP structure (right). Red lines (right) indicate collapsed road edges. Note: all map fragments are displayed at the same scale to clearly show the effect of the generalization process (in reality, the bottom map fragments should be shown at smaller scales).

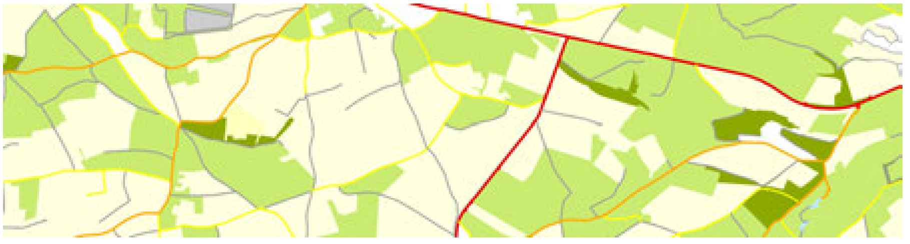
Figure 4.14 presents a detailed situation of a simple roundabout. Note that these types of infrastructure objects are not present as separate entities, nor classified as such; their road segments are dealt with individually.



(a)



(b)



(c)

(d)

FIGURE 4.12 An example of the generalization process in the rural region (presented at the same scale). The top figure (a) shows the input. The other figures (b), (c) and (d) show map fragments after 86.9%, 94.4% and 98.2% of the process, where 0% not generalised (a), with 11,300 faces and 100% is reduced to one face.

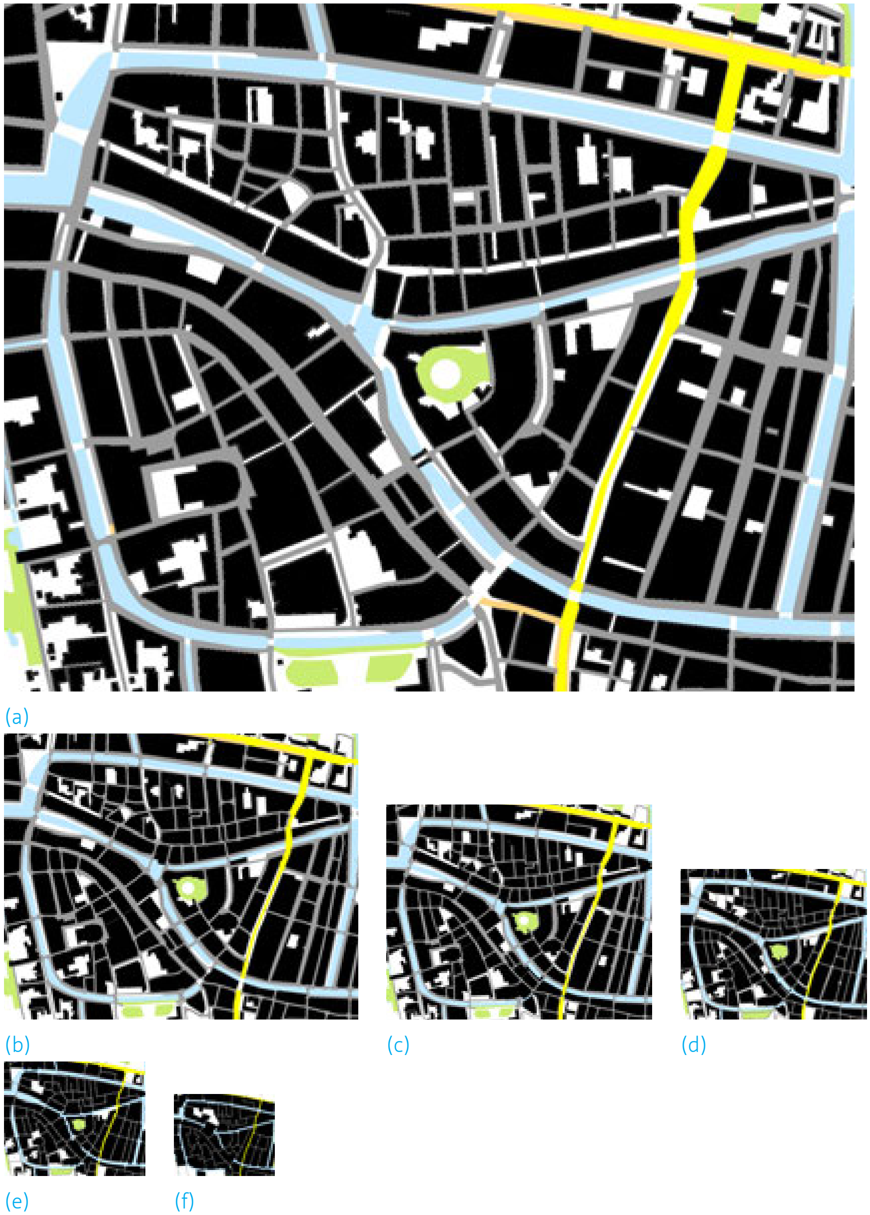


FIGURE 4.13 An example of the generalization process for the city center dataset. (a) shows the input; The other subfigures (b–f) represent 60.6%, 68.7%, 80.8%, 88.9% and 95.0% of the process, where 0% is the input with 19,400 faces and 100% is reduced to one face. Note: the map fragments are displayed at the intended target scales; (a) 1:10 k; (b) 1:16 k; (c) 1:18 k; (d) 1:23 k; (e) 1:30 k; (f) 1:45 k.

FIGURE 4.14 Example of the process for a roundabout. It changes the representation from areal in the initial partition in (a), partial areal and partial linear in (b) to linear (c). (d) demonstrates the option when line simplification is used.

§ 4.5.2 Quantitative analysis

Due to the nature of illustrations on paper, only specific intermediate map scales can be shown. The intention is to use the vario-scale data in combination with interactive zooming and panning operations over the map. However, there are only a few existing measures to evaluate continuous map generalization in general. Thus, we used visual inspection and compared our results to previous developments. To give a better quantitative notion of the process, we also generated some graphs, which give better insights into the whole generalization process.

First, Figure 4.15 shows the proportion of feature classes throughout the generalization process. For every generalization step, we count how many objects in the tGAP structure there are for a certain feature class. Then, those feature classes are grouped into 'water', 'terrain', 'buildings', 'roads' and 'other' super classes. This graph corresponds to the example depicted in Figure 4.12. Similarly, Figure 4.16 shows the area these objects cover in the structure. Note that a collapsed road has no area (even if it is still a map object) and that measured in the area of the roads has a smaller share than when expressed proportionally (%). Near the end of the graph in Figure 4.16, road objects do not occupy any area, although there are still road line segments. The water bodies are small and occupy only a small portion of dataset; therefore, they do not survive long in the process.

Second, Figures 4.17 and 4.18 provide another indication of the same example from Figure 4.12. With proper styling and color schema, it is not obvious which roads are still areas and which have been already collapsed. Therefore, these graphs present absolute numbers of road objects. Figure 4.17 captures only roads objects represented by areas. Figure 4.18 shows the number of edges in the structure (representing the collapsed road objects). It shows that the process collapses the majority of road areas first (creating new road lines). Later in the process – part *b*, the road lines between two merged faces are removed. This corresponds to our designed strategy.

Graphs presented so far have shown only small value changes. It indicates a gradual process that corresponds to our goal presented earlier. It also suggests that our generalization rules in the overall strategy were quite reasonable.

Finally, Figure 4.19 shows the usage of the different generalization operators throughout the process. Exactly one operator is applied in every generalization step, either the merge/remove or the split/collapse. The graph summarizes what happened ev-

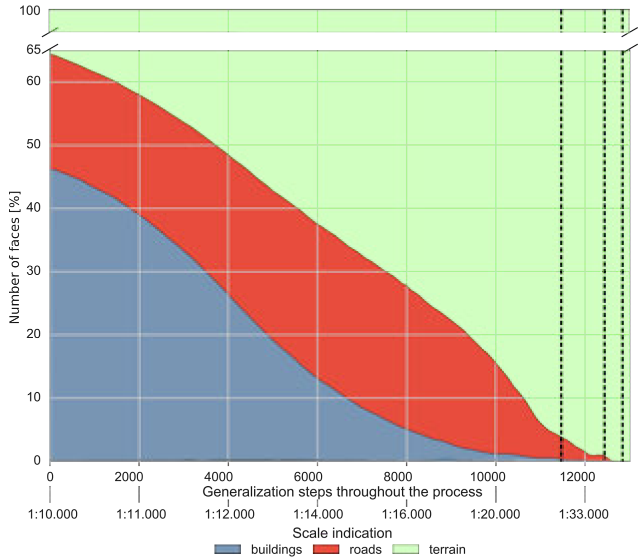


FIGURE 4.15 Ratio of feature classes throughout the generalization process in topological Generalized Area Partition (tGAP). The numbers are relative (number of faces per feature class divided by all faces present in the map). Note that the total number of objects decrease throughout the process. The vertical dashed lines indicate the map scales shown in Figure 4.12b–d.

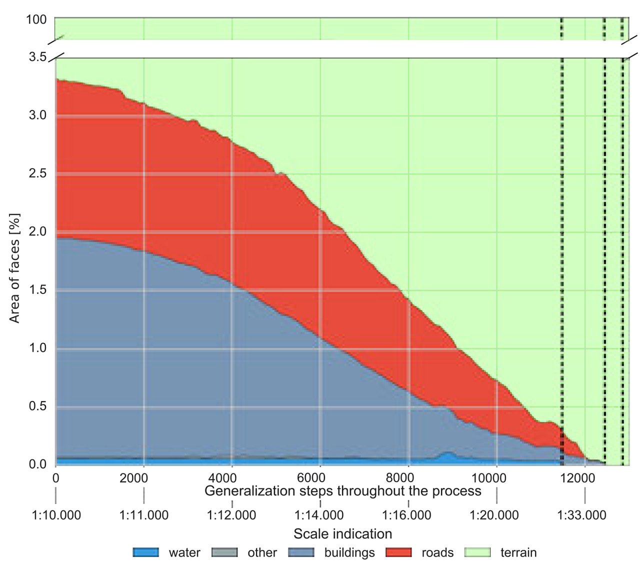


FIGURE 4.16 Ratio between the covered area in tGAP structure for feature classes throughout the process. The graph relates to the data of Figure 4.12. The vertical dashed lines indicate the map scale of Figure 4.12b–d. The numbers are relative.

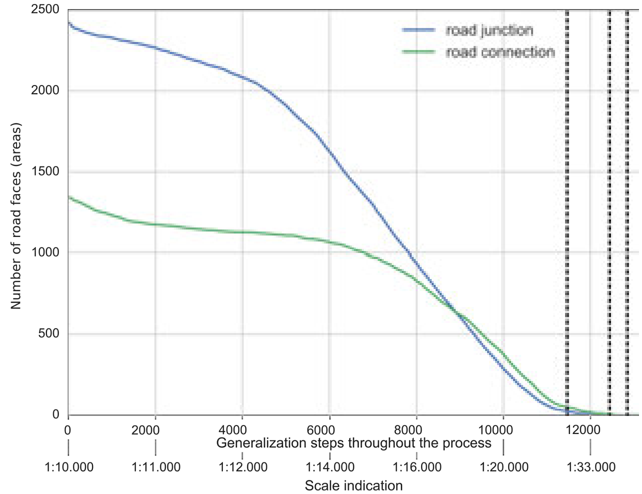


FIGURE 4.17 Number of road faces represented by areas throughout the process in Figure 4.12.

ery 500 steps and shows the ratio between the operators. One can observe that a lot of merge/remove operations happen at the beginning when tiny faces are merged. Those faces are mainly slivers from the preprocessing step when layers of buildings were 'fused' together with other layers. Since they are small, they have low importance and are processed first, but this could have been an additional preprocessing/cleaning step.

Later in the process, the split/collapse operator is more dominant because road objects are processed. Finally, the merge operator becomes more significant again because most of the roads are collapsed, and other objects are then merged together, removing any collapsed roads between.

§ 4.5.3 Additional discussion points

The results above have shown some reasonably good outcomes in automated and continuous map generalization. However, there are still quite a number of design decisions (as mentioned above) and some additional issues, which have been encountered during the design and implementation. In most cases, the best solution is not yet known, and further research is needed. The list of additional issues includes:

ROAD CLASSIFICATION Road objects were classified as *road junction* or *road connection*, based on the number of incident road segments, where a *junction* should have more than two road neighbours. There are two options when this classification can take place (*Design Decision 5*). In the first option, the objects are classified in a preprocessing step, and then the same knowledge is used throughout the whole process. In the second option – more dynamic, the objects are reclassified during the process, when needed.

Figure 4.20 shows the processing sequence for both approaches, starting from an initial configuration where humans would recognize the strip of grass between two roads running in parallel. We can see that static classification (on top in the figure) identifies two junctions and two connections at the beginning of the process. Then, connections

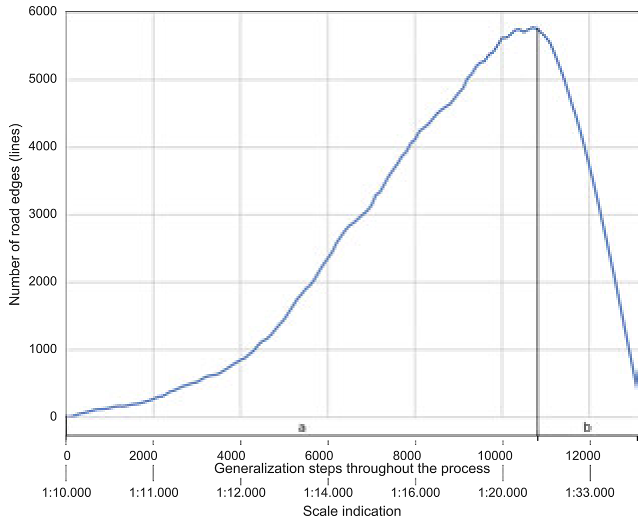


FIGURE 4.18 Number of road line edges in the structure (collapsed roads), related to Figure 4.12. Graph part *a*, where the number of collapsed road connections increases, represents the initial stage of the process, where roads are split/collapsed. Part *b*, where the graph decreases, indicates a situation where two area objects merge, intentionally removing the collapsed road lines between.

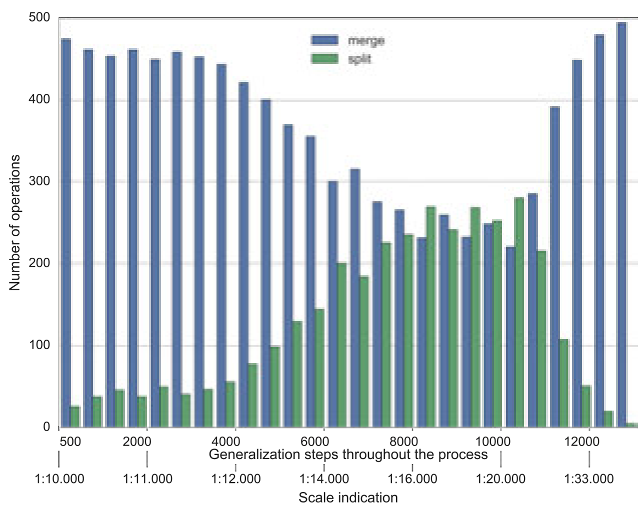


FIGURE 4.19 Application of the merge/remove and the split/collapse generalization operator throughout the generalization process. It is related to the example in Figure 4.12.

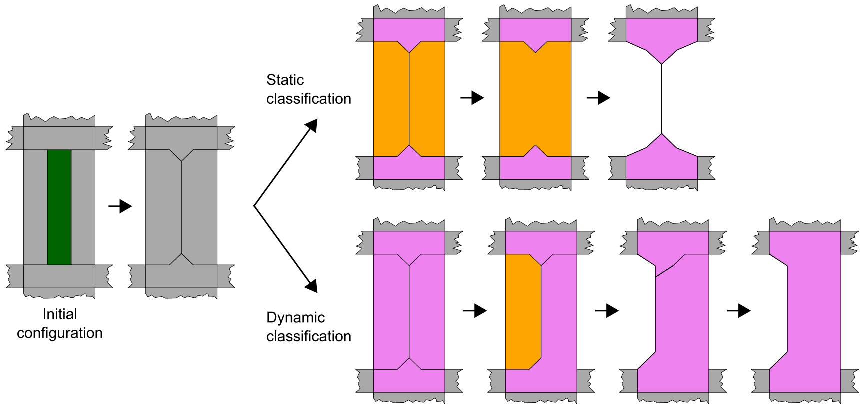


FIGURE 4.20 Alternate road object classification during the generalization process. It starts with collapsing the strip of grass between roads. Road junctions in pink and road connections in orange.

are merged together and then collapsed. On the other hand, dynamic classification (at the bottom) recognizes everything as junction objects (all objects have more than two incident roads). Then, two road junctions are merged, and a new classification identifies one new road connection, which is collapsed later. The last step is the merge of the two remaining road junctions.

The generalization process would continue and remaining junction(s) would be collapsed. Note that for dynamic classification this would happen much later because the final junction is larger (higher importance) and would lead to a different geometry of collapsed roads.

Besides technical aspects, such as memory use or time complexity, a static approach may lead to incorrect classification (of junctions and connections according to our definition) during the process, but it gives a slightly better overall cartographic impression during visualization. Therefore, we used it in our implementation.

RECLASSIFICATION AFTER SPLIT/COLLAPSE *Design Decision 6:* When the face of the road object is collapsed, the newly-created edges should carry a correct classification; see Figure 4.21. The face is transformed based on the skeletonization to a set of edges. Most of the time, one 'main' branch corresponds to the shape of the original face. However, the most appropriate reclassification for new edges is not so clear. Should all branches receive the same classification or should only the main branch be classified? Design Decision

New classification for all edges guarantees good connectivity, because it is more easily detected in the topology. The implementation is simpler and more obvious for further processing. Therefore, we used it in our approach. Nevertheless, it slightly deforms the original network. The network is 'spreading'; see the middle in Figure 4.21 and the detail in Figure 4.22. On the other hand, another option is to classify only the main branch of the collapsed object. This way, the road network is prevented from unwanted spreading. However, the whole network in the domain becomes more and more shattered throughout the process and connectivity analysis will be more difficult.

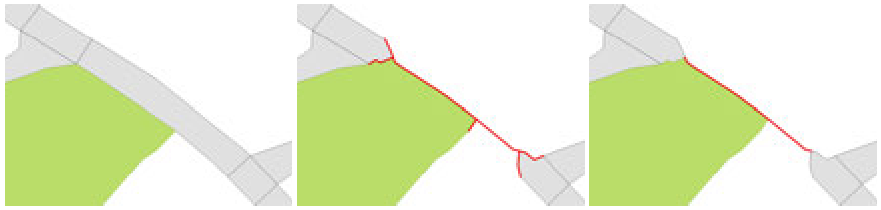


FIGURE 4.21 Reclassification after split/collapsed operation. One road connection object from the initial configuration (a) can be reclassified in two ways. Either all branches (in red) carry road classification (b) or only the main branch carries information (c).

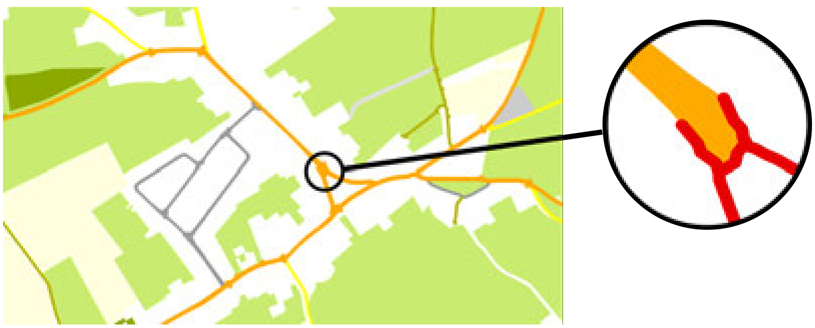


FIGURE 4.22 Side effect of all branches reclassification. This detail relates to Figure 4.11.

CROSSING OF MULTIPLE NETWORKS The input map is a projection of 3D space into a 2D map. This means that the information about linear networks crossing each other at multiple height levels should be preserved somehow in the map, allowing us to use that knowledge in the processing. However, what should we do if there is no so such knowledge available? Obtaining data is one thing (*e. g.* if it is available in TOP10NL), but how do we keep that knowledge during the process when objects change their representation? Additionally, how should the priority of individual networks be set? Another not really clear aspect is the solution of the case where more important networks cross less important ones, *e. g.* a road network crossing rivers and water channels. What to do in such a case is not really clear at this moment.

It is interesting to note a possible gain in the connectivity of two linear networks: roads and water. At the larger scale and with a single classification per face, it is not possible to model the fact that both the road network and the river network are properly connected. However, at the smaller scales, when the road and water segments are collapsed to lines and nodes, both networks stay connected at these locations (and better represent the nature of both networks).

§ 4.6 Better classification with groups

The current approach lacks a solution where the map features such as groups of buildings forming built-up area should stay longer in the generalization process rather than falling in the shattered pieces as seen in Figure 4.23. Here, we will presented better strategy for aggregating features.

The creation of tGAP structure is driven be two main aspects; First, the global order of the features based on importance value, *i. e.* in every step the least importance feature is selected and it is processed alone (collapsed) or interacted with adjacent features. Second, the most compatible neighbour(s). Our assumption based on visual inspections is that the first has 20% influence in the appearance of the final result compare to 80% for the second. Therefore, the second aspect is more relevant and has been researched before. Other previously developed strategies for vario-scale process can be summarized as following:

COMPATIBILITY MATRIX van Putten and van Oosterom (1998) provided full description of principle and it is captured in Section 3.2. The principle is based on most compatible feature class which is used for picking the most compatible neighbour. All possible combination are captured in the predefined compatibility matrix.

On one hand, the main advantage is the simplicity of the solution where same set of rules captured in the matrix stay the same throughout the process where the matrix defines relations between the feature classes which is based on analysing of existing data (and collect statistic about objects) . There is also a classification hierarchy in the matrix by definition. On the other hand, it suffers from the fact that the matrix for the whole process is designed based on educated guess and trial and error.

Figure 4.23 demonstrates the result of generalization with simple a compatibility matrix. Note that resulting map 'falls apart' into pieces quite early during the generalization process. The small objects are eliminated, other faces are merged into objects of similar sizes and patterns such as streets, villages or grouped settlements are hard to distinguish, giving the user an artificial impression. This problem is even more obvious later, see Figure 4.23c and d.

CONSTRAINED TGAP Haunert et al. (2009); Dilo et al. (2009) proposed an other method of capturing better map content in the generalization process, see Section 3.3. Beside the source scale it uses a final generalized map (small scale). This final small scale map is obtained either from other sources or computed in advance. Then it generates all intermediate scales between in such a way that the final sequence of maps progresses between the large and small scale maps. The most significant limitation of using constrained tGAP is in the requirement to use additional knowledge derived from other solution. In theory, it means that the solution cannot run independently. Moreover, the results are very dependent on the 'target' small scale approach obtained from complex and expensive preprocessing step or other source.

Besides the mentioned approaches above there are other researchers who tackled similar problems in an automated generalization process. Haunert and Wolff (2006) used a very expensive aggregation function to identify the possible groups of features which should be handled and preserved throughout the process. Zhang (2012) used a method based on geometric configuration only, *i. e.* clusters of building or build-up

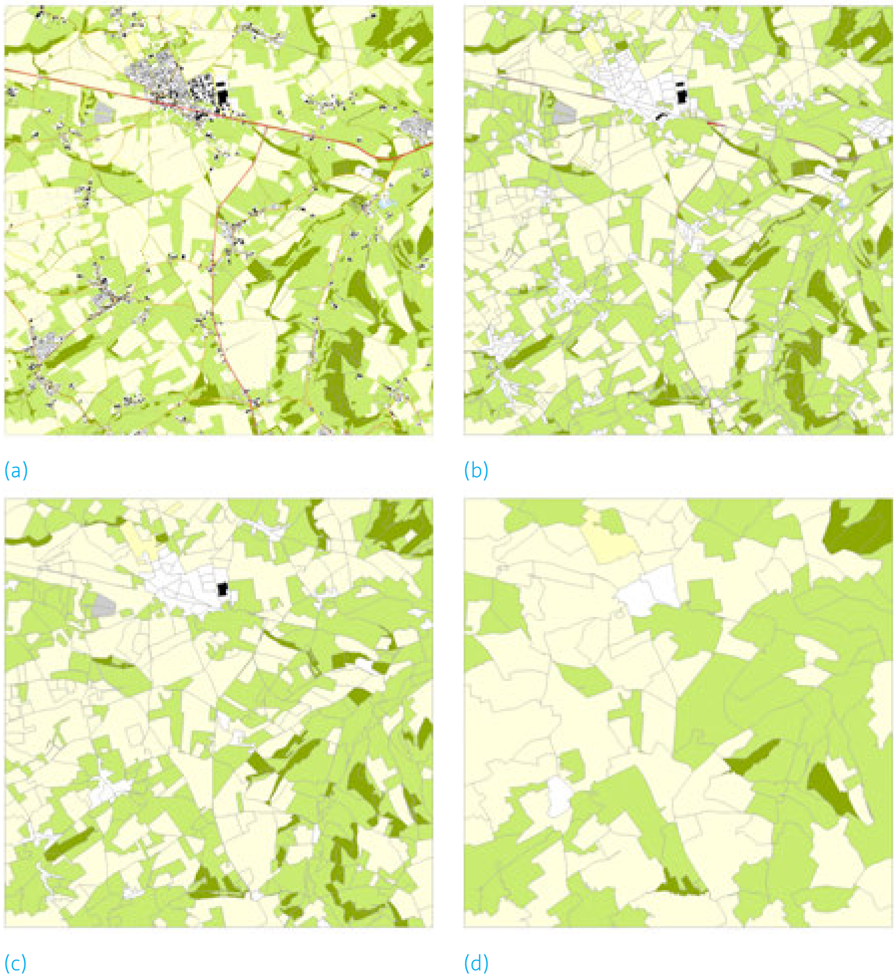


FIGURE 4.23 Generalization process where compatibility matrix been used. (a) shows the input. The other figures (b), (c) and (d) show map fragments after 87 %, 93% and 98% of the process, where 0% is the input (a) with 11,300 faces and 100% is one face.

FIGURE 4.24 The same test dataset as in Figure 4.23. An example of groups definition as shown in Figure 4.25 took place in the rectangle region.

areas. It uses Delaunay triangulation and Minimum Spanning Trees structures to construct a proximity graph defining the building clusters.

Even with those proposed strategies we want something more generic. We seek a very generic solution which can; 1) be easily implemented and used for arbitrary input data, 2) discourage certain merges to happen while features are potentially quite compatible, and 3) be obtained from multiple sources such as classification, geometry, or other. Therefore we propose a new method for *the most compatible neighbour* selection. The innovative idea is based on the object's membership in predefined sets and we call the method *groups*. It uses the fact that objects are members of the same groups to influence compatibility value. Note that the individual groups can be overlapping and one object can be member of the multiple groups. More practically, for the selected object we compare all adjacent objects and the one with the highest number of common groups is chosen as *the most compatible neighbour*. Other parts of the generalization process stays the same and the generalization actions are performed the same way.

At the beginning of the process the groups have to be defined. They can be obtained from attributes of the input data and/or as a result of the geometrical test. Selection of all local roads can be the example of the first, faces completely fitting a polygon representing build-up area can be the second. Obtaining the groups from attributes is straightforward, however obtaining the groups based on geometrical test requires a layer of geometry to test against. There are more options how this layer can be obtained. The considered options follow (together with indication of our experiences within brackets):

- A layer of geometry is obtained from existing maps of target scale (not tested). Then the map objects are tested against the layer *i. e.* faces completely fitting the polygons will be members of the group. This option has same limitation as the constrained tGAP method; (1) the quality of result is dependent on additional source material and (2) the fact that it cannot run independently.

- Triangulation of the building geometries layer first and then elimination of triangulation edges crossing the roads (not tested). The group membership is defined based on remaining clusters of objects still connected by triangulation edges. The method is inspired by similar strategy for the displacement operation Zhang (2012).
- A layer of geometry is created by buffer of the building geometries (tested). It is inspired by research conducted by geo-ICT Geodan[3].

Figure 4.25 shows an example of the groups generated in a test dataset. See the overview in Figure 4.24, where four groups have been created. First three g_1, g_2 and g_3 are retrieved from the *feature class* attribute of the input. g_1 with the smallest regions is defined by all roads, see Figure 4.25a. g_2 with bigger regions is defined only by local and the most importance roads, see Figure 4.25b. Large blocks in g_3 are defined by the main roads (highway) and hydrology, see Figure 4.25d. The urban regions in $g4$ are defined by geometrical test against a combinations of buffers around the buildings; We used buffers of the building geometries (to grow) and the buffers of negative value (to shrink). The combination of buffers is used to create a geometry representing build-up area where buildings are close to each other and details and holes in geometry are removed .

In the phase when the selection of the most compatible neighbour take place it uses the following formula for every possible neighbour:

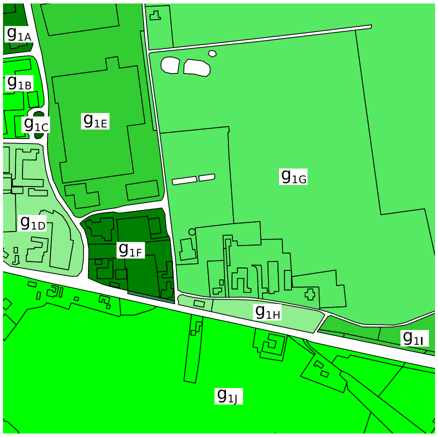
$$compatibility_{f,n} = \frac{|G_{common}|}{|G_{total}|} + l^{-10}$$

where $compatibility$ defines a value between the least important face f and neighbour n. $|G_{common}|$ specifies the number of common groups of f and n and $|G_{total}|$ shows total number of the groups in dataset. l is the length of the common boundary between f and n. The value l plays minor role in breaking a tie between two possible neighbours with the same $compatibility$. The potential neighbour with the higher compatibility will be chosen for the most compatible neighbour. Note that for a newly merged face a membership in the groups must be defined. We used attribute intersection of previous two memberships, no new testing is performed.

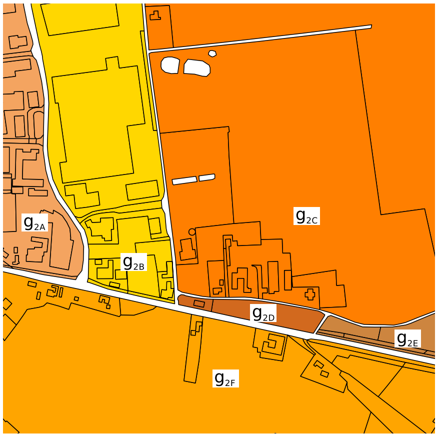
Figure 4.26 shows complete generalization process where groups were applied. To make it more transparent only the merge operation is used. A good indication can be the dominant highway in the top half of the figure running from west to east. It is preserved throughout the entire process, which was our initial intention. We can observe that road features are present much longer in the process, instead of approx. 60% in the process captured by Figure 4.23. On the other hand the roads features become more dominant in later stage (depicted by the yellow and orange regions in Figure 4.26d). Note that this would probably not happen when split operation is used as well.

We can observe two interesting things; The first, the faces are merged within the same groups if possible. The second, since the size of the features drives the whole generalization process it has a side effect on the groups. The groups with smaller regions

3 http://research.geodan.nl/sites/bag-tiles/



(a)Green polygons are members of g_1. The group is defined by all roads.



(b)The objects in shades of orange are in the group g_2. Provincial roads, highways and hydrology define the group.

(c)The objects in red are in the group g_3. Hydrology and highway define the group.

(d)The objects (in purple) define urban regions of g_4.

FIGURE 4.25 Four defined groups g_1, g_2, g_3 and g_4.

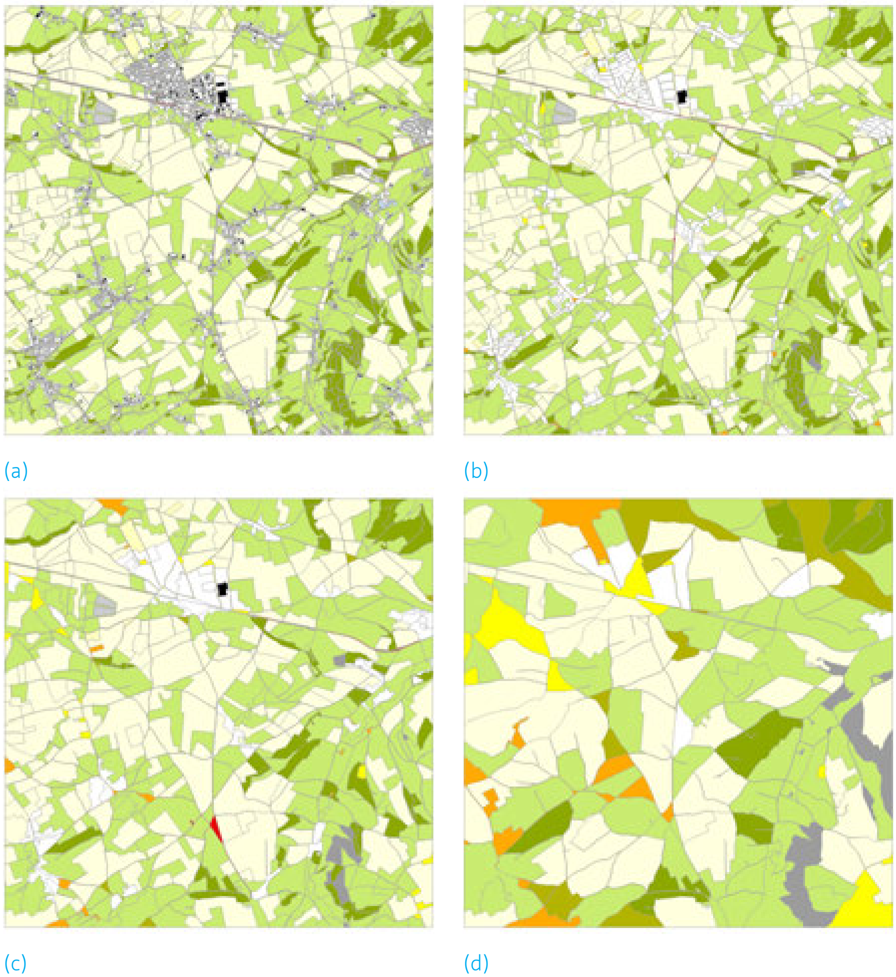


FIGURE 4.26 Generalization process where only merge operation with groups have been used. The figure (a) shows the input. The other figures (b), (c) and (d) show map fragments after 87 %, 93% and 98% of the process, where 0% is input (a) with 11,300 faces and 100% is one face.

are processed first followed by the groups with bigger areas, and so on. This puts the groups in implicit hierarchical order. In our case g_1 is dissolved first and one group covering all of the dataset remains at the end.

Another advantage comes from the fact that additional information is obtained from the same input. Moreover, it is not dependent on the setting of a precise threshold, only checking object membership. This makes the group method very generic, cheap and effective, and worthy of future investigation. In this section, we have presented the group method only to show the principle. However, we assume better results can be obtain in combination with the compatibility matrix, where the matrix steers the process at local level (within small building blocks) and the group method at global level (urban - rural regions).

§ 4.7 Conclusions

Up to now, our vario-scale method could only be used to represent area features. We have shown that line features in our current vario-scale solution may be introduced. In Section 4.4 we have designed an algorithm which provides the fully-automated generalization process that preserves a road network throughout all scales, and which gives reasonable results. The algorithm maintains knowledge about a road network even in situations where roads are partially represented by lines, as well as partially represented by areas. We presented the necessary modifications that have to be applied to accommodate such an algorithm and still follow the idea of small generalization steps. For this, we have introduced new concepts, techniques and more processing strategies in Section 4.3 and presented six design decisions. We also have suggested that it is sufficient to use a large-scale planar partition with only area objects and their classification as input. We have validated our approach on a test dataset together with some quantitative measurements. With the generated results, we now have an opportunity to conduct user testing of the vario-scale principle. However, there are still the following issues that need to be researched:

- Other options of the various design decisions as mentioned in this paper could be further analysed.
- The advanced treatment for water networks (also junctions and connections), rail networks, buildings or other feature types could be included. It might be very well the case that during the generalization process additional knowledge or different treatment is needed for these types.
- Line simplification in the generalization process could be included to create the vario-scale data structure. However, standard line simplification techniques could introduce extraneous line intersections. Therefore more care needs to be taken with non-trivial simplification, considering topological correctness of results.
- Besides `Top10NL` data, the proposed approach should also be applied to other datasets; *e. g.* Corine (smaller starting scale) or Dutch BGT (Basisregistratie Grootschalige Topografie) (larger starting scale).
- The proposed strategy using groups requires more testing. Neither the optimum number of groups nor limitation of the size of the dataset is known, *i. e.* can we obtain same result for 200 step process as for 2,000 step process?