Optimization techniques for polygon generalization

Martin Galanda
Department of Geography, University of Zurich
Winterthurerstr. 190, 8057 Zurich
mgalanda@geo.unizh.ch

1 Introduction

Recent developments in cartography and geographic information science demand increased research efforts in map generalization. On the one hand map production on demand has grown into an important technique of exploratory work style in geo science and to a service in commerce appreciated by an increasing number of (web) users. Map production on demand could be facilitated and speeded up by the use of a single master database from which arbitrary desired scales could be derived by automated generalization. But some of the key generalization methods are missing so far - especially for categorical data. On the other hand Internet domain still suffers from low data bandwidth and screen resolutions (Arleth 1999, Hardy 2000), e.g. GIS applications in web browsers or location based services on small PDAs. Automated generalization of transmitted and displayed data could help to ensure short loading times and high cartographic quality in spite of restrictions by technology (Buttenfield 1999).

Categorical data (e.g. land use, geology, vegetation etc.) are a common data type in many GIS applications, particularly in thematic and topographic maps. The generalization of such data in the vector model is still in the early stages of research. Nowadays usually line based algorithms are used to generalize these data. Since their specific structure and topology is often ignored, nonsatsifying results are not surprising. Existing approaches concern either very specific kinds of data (e.g. Peter 1997, Mueller and Wang 1992) or the isolated implementation of single geometric algorithms (e.g. Bader 1997, Jones et. al. 1995). Comprehensive techniques for polygon generalization are still missing. Thus this paper focuses on the generalization of categorical data in the vector model (polygon generalization).

Algorithms especially designed for use with polygonal data are the main interest of the presented research. In concern with algorithms the goal covers the evaluation of snakes for generalizing polygon mosaics. Starting from an inventory of existing algorithms (sect. 2.3) necessary algorithmic developments are shown in section 2.4. The main part related to algorithms introduces and discusses the application of snakes, one particular optimization technique for polygon generalization (sect. 3). Finally some conclusions are drawn and an outlook on the next steps of the research project is presented (sect. 4).

2 Algorithms for polygon generalization

While operators designate abstract transformations in the generalization process algorithms put the geometric and semantic transformation of the data set into execution. They are the basis for performing any generalization.

2.1 Requirements of algorithms

In the design of algorithms for generalizing categorical data the idea of a modular and flexible construction system is followed can be adapted for the solution of as many different cartographic conflicts as possible related to the generalization of polygonal maps. So it is preferred to implement several simple algorithms instead of a complex one whenever possible. Algorithms must solve a cartographic conflict embedded in the whole polygon mosaic, i.e. the result should not only be cartographically pleasing but also represent a consistent state of the data set. For example, the enlargement of a single polygon can be easily applied by scaling it. But then topology and attributes in the polygon mosaic must be updated, as a solution like a simple graphic overlay is rarely acceptable and useful. Algorithms also have to produce predictable and repeatable solutions in order to enable their automatic use. Last but not least there are the generic and self-explanatory requirements: efficiency related primarily to computing time and ease of use. Some of the above specifications are in conflict with one another. In practice, every algorithm has to find a compromise between these different requirements.

2.2 Automation levels of algorithms

Looking at the whole generalization process three different automation levels of an algorithm are feasible:

- *Interactive algorithms*
  Algorithms are applied similarly to traditional generalization done by hand. All decisions, like the choice of the algorithm, parameter setup, conflict detection etc., are up to a cartographic expert. The algorithm only carries out the data transformation itself.
- *Automated algorithms*
  Every algorithm is a stand-alone application. That is parameter setup, geometric or semantic transformation and the updating of the polygon mosaic are performed automatically. The cartographer, however, is still responsible for developing a strategy and for defining the sequence of appropriate algorithms as well as the evaluation of the final result.
- *Integrated algorithms*
  Algorithms are modules/libraries in an automated generalization process. A generic framework coordinates all the modules that automatically detect conflicts, develop a strategy at a global and local level, initialize sequences of operators and algorithms and supervise the interplay of algorithms. Experts in cartographic generalization are responsible for the final evaluation, if need be some interactive post-processing and fine-tuning of the software.

In the context of this paper algorithms of the first and second class are relevant. An upgrade of the below presented algorithms to the highest automation level is aspired in this research project later on.

2.3 Existing algorithms for polygon generalization

The next section gives a short overview of existing approaches for polygon generalization with a focus on algorithms. They are grouped according to the cartographic operator[1] that they can mainly be associated with.

2.3.1 Algorithms for Reclassification

The smaller the map scale the less different classes can be distinguished in a map. The appropriate number of displayed classes is a key factor to ensure proper communication between the cartographic representation and the target audience. Thus polygon generalization requires algorithms for performing semantic data transformation, i.e. for combining different classes into one "super" class. In this process metadata concerning the original classification is indispensable for achieving a satisfying solution. From my point of view reclassification is the most important operation when generalizing a categorical polygon mosaic. By grouping semantically related polygons the number of polygons decreases and a lot of potential conflicts vanish due to the semantic and geometric combination of these polygons. These algorithms can also execute the aggregation of adjacent polygons or the elimination of a polygon.

---

[1] A detailed discussion of generalization operators is found among others in McMaster and Shea (1992), Agent Report D2 (1999).

2.3.2 Algorithms for Aggregation

With respect to polygons, aggregation is often termed in literature as the most important algorithm for generalization. Thus, it is the operator best explored and examined in the past (Bader 1997, Cottingham 1997, DeLucia and Black 1987, Jones et al. 1995). Figure 1 shows the principle of the algorithm proposed by Jones et al. (1995). It is based on a constrained Delaunay triangulation that models topological relations and provides new edges for the combined geometry.

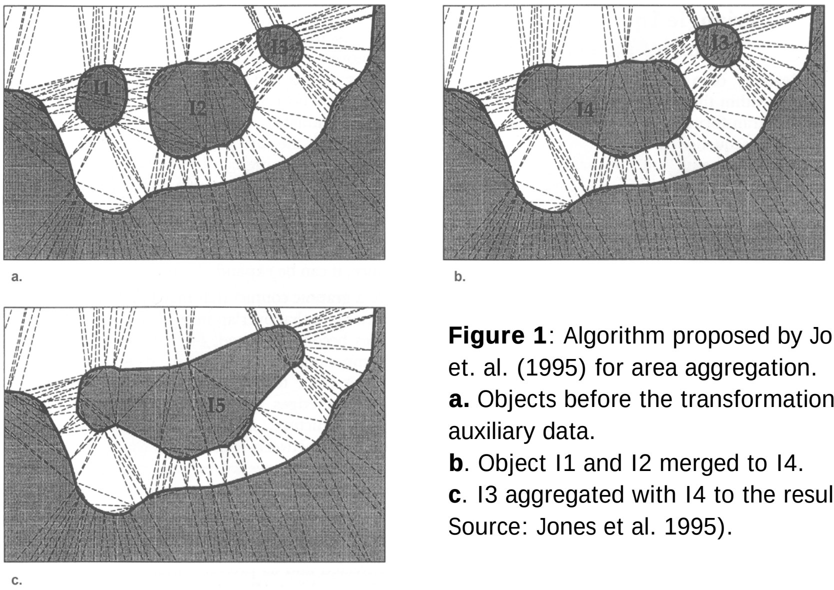


Figure 1: Algorithm proposed by Jones et. al. (1995) for area aggregation.
a. Objects before the transformation plus auxiliary data.
b. Object I1 and I2 merged to I4.
c. I3 aggregated with I4 to the result I5.
Source: Jones et al. 1995).

2.3.3 Algorithms for Collapse

Polygons can be reduced in their geometric dimension either to lines or to points (Bader 1997, DeLuzia and Black 1987, Jones et al. 1995). The first case is solved in all approaches by calculating the skeleton of the object – see the middle of figure 2. The point chosen to represent a polygon is usually set on the object's center of gravity or the object's geometric center.

2.3.4 Algorithms for Displacement

Displacement involves the movement of the whole polygon. The displacement of one object needs to be spread out to the object's neighborhood; therefore so-called propagation has to be handled by displacement algorithms, too. Jones et al. (1995) proposed a geometric solution for polygonal data while Bader (1997) tested models inspired by physics to calculate displacement.

2.3.5 Algorithms for Enlargement and Exaggeration

The increase resp. the decrease of an object's geometry and consequently its thematic importance is a common technique when performing map generalization. For polygons a global[2] (enlargement) and local[3] (exaggeration) transformation can be distinguished. While the first case is similar to a scaling of the object the second one requires methods similar to those implemented in the algorithms for displacement (Bader 1997, Jones et al. 1995).

2.3.6 Algorithms for Selection or Elimination

To emphasize important objects and to omit less important objects is one of the generic principles of cartographic generalization. Due to the space exhaustive nature of categorical data the selection (elimination) of polygons demands methods for designing the area of an omitted polygon to one or several polygons of the remaining classes. A semantic approach is to perform reclassification (see

---

[2] It is related to a whole polygon.

[3] It' related to a part of a polygon object.

above), a geometric solution is to divide the object to be deleted among the neighboring polygons (Bader 1997) – compare figure 2.

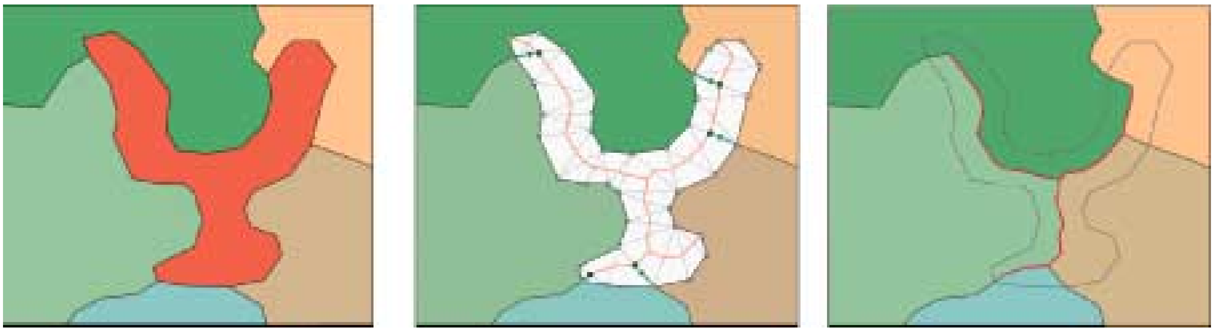


Figure 2: Algorithm proposed by Bader (1997) for eliminating a polygon based on the construction of a skeleton of the object (Source: Bader 1997).

2.3.7 Algorithms for Simplification and Smoothing

The operators simplification and smoothing are not a primary topic of the research project presented here. For details refer to Douglas and Poiker (1973), de Berg et al. (1998), Saalfeld (1999) etc.

2.3.8 Algorithms for Typification

Beyond any doubt this operator requires the most complex key decisions and measures. The pure geometric transformation can be regarded as a combination of elimination, aggregation and displacement processes. Until now only Mueller and Wang (1992) have proposed an approach for polygonal data resp. islands. Empirical experiments and further research are needed.

2.4 Starting points for research

Recapitulating the review of already existing algorithms for generalizing polygonal data in the previous section the following points stand out:

- Most of the algorithms only work with few isolated polygons. The application in the context of a polygon mosaic and of several cartographic conflicts is still largely missing (e.g. Jones et al. 1995, Bader 1997).
- Algorithms offered in commercial GIS products often produce merely graphical results, e.g. after the enlargement of a polygon the new object is saved to a separate class/layer and the polygon mosaic resp. the geometry of the neighboring objects is not updated.
- Papers dealing with categorical data generalization often present very specific solutions for one kind of data or a particular scale range. Examples are algorithms in the raster domain Peter (1997), Jaakkola (1998).
- Most approaches solve conflicts by considering geometry only, semantic and its significance for the whole generalization process is often neglected. (e.g. Jones et al. 1995).

However, the need for improved or new algorithms for polygon generalization seems obviously. The research required can be pursued towards different directions:

- the implementation of already existing algorithms that are not part of the chosen GIS - first of all the snake algorithm presented by Bader (2001),
- the development resp. improvement of algorithms that are missing or don't meet the requirements (e.g. an algorithm for typification or reclassification),
- the acquisition of additional knowledge for automated polygon generalization (e.g. guidelines for the number of different classes shown at a specific scale).

In the following, this paper concentrates on the discussion of optimization techniques - particularly so-called snakes - as a basis for generalization algorithms for polygonal data.

2.5 The algorithms' implementation

In this research the object-orientated GIS software Lamps2 by LaserScan serves as a test bed. Thus, algorithms can be based on the generalization methods already available within this GIS. With respect to polygonal data algorithms for aggregation, classification, collapsing, displacement and enlargement are available. The main drawback of all these tools is the missing possibility to update the polygon mosaic after either a semantic or geometric transformation of one or several polygons. For instance, the classification algorithm allows to group different classes into a new class, but doesn't initialize the geometric combination of adjacent polygons belonging to the same class. In spite of the choice of Lamps2 as testing and developing environment the following discussion is not related solely to this specific product but represents an attempt to implement tools for polygon generalization within a commercial GIS.

3 Optimization techniques for polygon generalization

Optimization techniques are well known methods in physics and engineering. In general, they are concerned with the determination of local or global optima (maxima or minima) of a function. Recently, optimization methods have been introduced to cartographic generalization research especially related to linear objects, e.g. least squares adjustment (among others Harrie 1999, Sester 2001), steepest gradient method and simulated annealing (Ware and Jones 1998), energy minimizing splines or so-called snakes (Burghardt and Meier 1997, Burghardt 2000, Bader 2001) and elastic beams (Bader 2001). The next section intends to highlight the quality of optimization techniques for cartographic generalization by discussing both their main advantages and disadvantages. The following parts concentrate on studying the snake method, a particular type of optimization techniques, and demonstrate its use for polygon generalization by presenting some first results of experiments with real world data.

3.1 Properties of optimization techniques

Optimization methods support a global generalization approach, i.e. several conflicts can be solved at once resp. it is hoped "that the weighted sum of local corrections guarantees also a good global solution (Bader 2001)". Thus, the transformation of one object is not calculated detached from the whole polygon mosaic. On the contrary, the polygon interacts with its adjacent and nonadjacent neighbors to find the (mathematically) optimal result. The formation of new conflicts resulting from the change of an object's geometry is prevented. This quality is shown in figure 3: Object O_2 is displaced from O_1. As a consequence the distance from O_2 to O_3 falls below the minimum threshold, too. Using global optimization methods O_3 is pushed away resp. reduced in its area besides the displacement of O_2. When the (iterative) process stops O_2 has reached the desired distance to O_1 and at the same time, the minimal distance to O_3 is ensured.

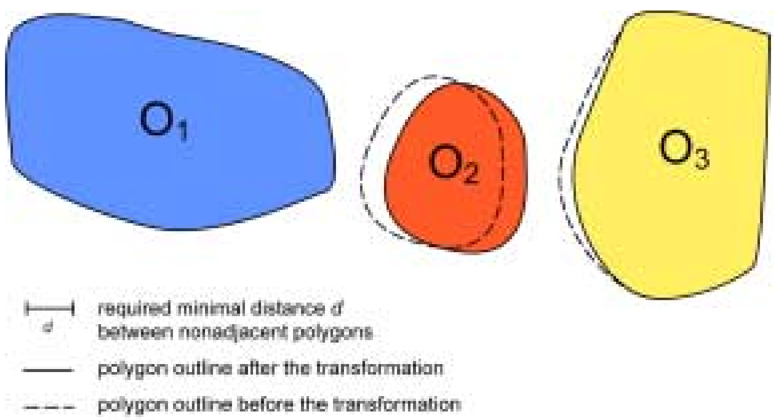


Figure 3: Interaction of a displaced object with its disjoint neighbors when using global optimization.

Using optimization methods has the second advantage that the method propagates directly a change of a polygon's geometry to all adjacent objects. Propagation is attached directly to the geometric transformation: both processes are merged and executed at the same time (Bader 2001). Figure 4 visualizes the enlargement of a polygon object with multiple adjacent neighbors: In the same computation that polygon O_2 is enlarged the geometry of O_1 and O3 is adapted, i.e. the displacement offset at the common points of O_2 with O_1 and O_3 is diffused and cushioned in the polylines automatically.

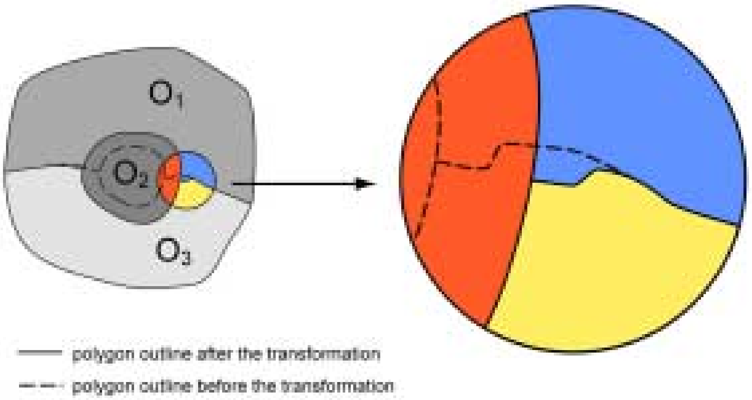


Figure 4: Propagation of geometric changes to adjacent objects shown in the example of polygon enlargement.

So-called shape parameters help determining the flexibility of objects in position and shape (internal constraints) within optimization techniques. The violation of a cartographic constraint is mapped to forces acting on the observed object (external enforcement). Optimization methods strive to reach a balance between both internal constraints and external enforcements. But there is no guarantee that the state of balance represents a cartographically pleasing solution for all problems.

Optimization techniques can greatly facilitate cartographic generalization. Table 1 compares exemplarily the main steps necessary for enlarging a polygon by an algorithm based on optimization (a global approach) and one based on scaling (a sequential approach).

polygon enlargement	
by optimization techniques	**by scaling**
1. Choose the object to be enlarged. (neighboring polygons are selected automatically)	**1.** Choose the object to be enlarged.
2. Set the method's parameters.	**2.** Apply the scaling algorithm.
3. Run the algorithm.	**3.** Define new end points for borders originally connected to the object.
4. Evaluate the result.	**4.** Propagate the displacement to these borders.
	5. Search for new conflicts due to the transformation. (possibly loop for the solution of new conflicts)
	6. Evaluate the result.

Table 1: Main steps necessary for enlarging a polygon by optimization methods and by scaling.

Enlargement by optimization techniques requires fewer input by the cartographic expert: Just select the object that should be enlarged, set the method's parameters, maybe fine tune them and finally trigger the algorithm. The direct result again is a logically and topologically consistent polygon mosaic. Of course, several polygons can be processed at once and the creation of new conflicts is prevented. A sequence of algorithms is needed to enlarge a polygon based on scaling. Beyond any doubt the scaling itself is straightforward. A separate algorithm must realize the ‘tricky’ update of the polygon’s neighborhood – step 3 and 4 in the right-hand column of table 1. High-level measures resp. several key decisions are needed that are not available until now or difficult to implement in an automated way. Then, a routine has to search for new conflicts resulting from the transformation of the polygon mosaic and if need be initialize the conflict’s solution. Optimization methods are based on solid mathematical models. Consequently, they are very robust and hardly fail (Bader and Barrault 2000). However, as their complexity and power raise very high computational costs it is recommended to always keep alternative algorithms in mind that could provide similar results much faster especially in the case of less complex conflicts. For instance, the enlargement of a single island polygon is accomplished more easily by a scaling algorithm than by an optimization method.

In conclusion, it seems legitimate to state that optimization techniques represent the holistic nature of map generalization due to the listed characteristics much better than traditional sequential approaches. In the following, the paper focuses on the application of one particular optimization technique for polygon generalization, namely the snake method. It provides a compromise between complexity on the one hand and both computational costs and possibilities of cartographic steering resp. ease of use on the other hand.

3.2 The Snake method

The concept of snakes has been developed in the field of computer vision (Kass et al. 1987). Burghardt and Meier (1997), Bader and Barrault (2000) as well as Bader (2001) have transferred the method to cartographic generalization adapting it primarily to the displacement of linear objects. Burghardt (2000) also applied snakes to polygons. However, his approach was restricted to the displacement of disjoint (isolated) polygons. A snake is an iterative process – striving to improve the solution over time - with an energy minimizing spline guided by internal constraint forces (inner energy) and influenced by external enforcement (external energy) (Bader 2001). Inner energy results from the difference between the initial object and the object after the displacement or deformation. Parameters allow to determine the elasticity and rigidity in the model. External energy defines the sum of forces acting on an object. They are determined by cartographic analysis of the object's neighborhood - e.g. forces could result from two polylines that are too close or from available map space around the object. The aim is to minimize the sum of internal and external energy. Snakes compromise excellently between allowing an object to fulfill cartographic principles and to resemble its initial shape at the same time.[4] Not only line generalization as shown in the previous works but also polygon generalization may benefit from the use of this method.

As the snake method basically works on line objects, every polygon is split into polylines of its borders, to which all the polygon’s attributes are mapped. Nevertheless, the implementation of the snake method meets the requirements of taking into consideration the topological structure of the polygon mosaic. The generalization process by snakes is guided by method parameters and external parameters. They allow a cartographic steering of the generalization.

3.2.1 Method parameters[5]

The shape parameters ‘alpha’ and ‘beta’ define the flexibility of the snake. The higher they are the less deformation of the line objects can be expected. Figure 5 gives an impression of how shape parameters can influence the result of a snake-based transformation.

---

[4] For more details on the theory behind snakes refer to Bader (2001).

[5] For a detailed description of these parameters and its influence on the generalization process refer again to Bader (2001).

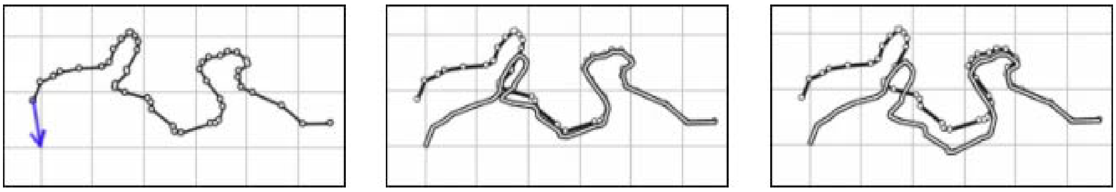


Figure 5: Low shape parameters may produce cartographically unacceptable results (middle) while an increase of alpha and beta can stiffen parts of an object, compare the narrow bend in the right figure. (Source: Bader and Barrault 2000)

By now ‘alpha’ and ‘beta’ are assumed to be the same for all objects. An object specific variation of the shape parameters may be tested later. But Bader (2001) generally concludes, “the influence of the shape parameters is less significant that might be hoped”. The ‘gamma’ value sets the inertia of the snake method. A low ‘gamma’ results in a weaker and more local displacement in one calculation step. The attraction term ‘psi’ controls the length used to cushion a displacement. The higher it is the more positional accuracy is obtained – compare figure 6. An increase of the shape parameters is equivalent to a decrease of ‘psi’ when ‘alpha” and ‘beta’ are constant. If need be an adjustment of the shape parameters is recommended (Bader 2001).

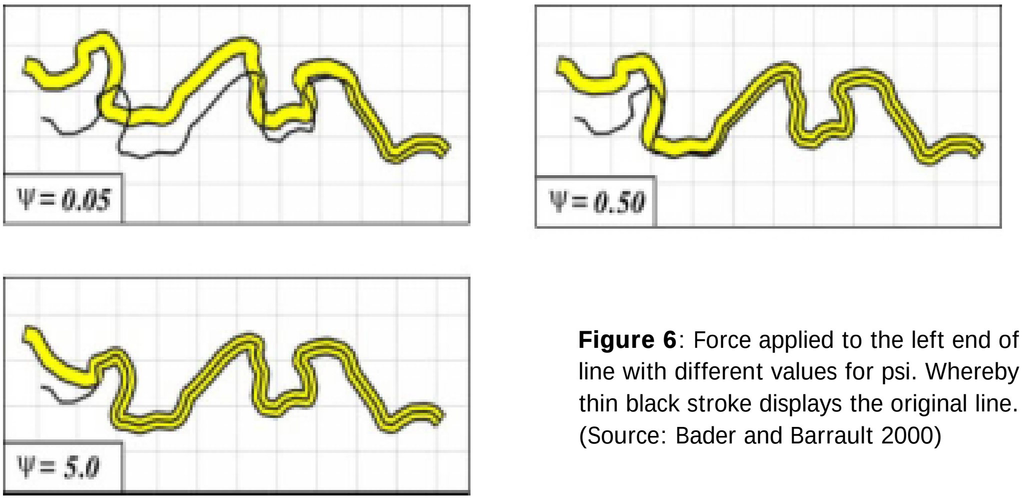


Figure 6: Force applied to the left end of the line with different values for psi. Whereby the thin black stroke displays the original line. (Source: Bader and Barrault 2000)

The fine-tuning of the parameters and the definition of the best ratio of the parameters to each other needs further investigations. It is one of the goals of this research to provide some generic rules for parameter setting at least with respect to polygonal data of different scales and type.

3.2.2 External parameters

The preliminary basis for optimization techniques is external energy. In map generalization forces acting on geometric objects describe this energy source. In general forces result from the violation of a cartographic constraint (e.g. symbol overlap, congestion). At the moment they are computed by virtue of the shortest distance between map objects. The concept may be extended in order to also consider the availability of free map space and that conflicts within a single object are regarded as well – e.g. a narrow winded river that borders a country. In fact forces applied to the objects depend on the chosen generalization operator – see also section 3.3.

The cartographic steering of the method can be enhanced by the utilization of object specific attributes. Each object holds a so-called weight attribute that controls its activity in the snake process. Values between 0.0 and 1.0 are valid, whereby 0.0 indicates no change of the object’s position, i.e. the object acts as an obstacle for all the other objects, and 1.0 indicates as much change of the object’s position as calculated during one iteration. Values in between specify a continuous transition between these two extreme behaviors. Node objects can be stiffened in two different ways. On the one hand they can be fixed totally, that means no displacement of the node and no change of the directions to the connected line segments[6] occurs at them. On the other hand only the directions can be fixed. This behavior is among others relevant for estuaries in topographic maps, for instance, where rivers border land cover units.

3.3 Snakes for polygon generalization

A study of the results of previous projects related to snakes and generalization (Burghardt and Meier 1997, Burghardt 2000, Bader 2001) and of operators needed for polygon generalization (section 2.1) identifies those generalization operators possibly suited for an implementation by snakes:

Displacement	- to change a polygon's position in map space.
Enlargement	- to expand or reduce an object equally in each direction.
Exaggeration	- to widen or reduce parts of objects (change of shape!).

These operators are all primarily used to remove metric conflicts resp. to avoid the violation of a metric constraint[7] - e.g. the operator 'enlargement' may be used when a polygon's area is too small, 'displacement' can ensure the minimal distance between two objects. Section 2.1 has already mentioned those algorithms currently used resp. known from literature for displacement, enlargement and exaggeration. Snakes can serve as basis for all these transformations. A variation of the objects' weights and force models applied then specifies the generalization operator. The use of weights for polygon generalization based on snakes has already been discussed in section 3.2.2.

force model	**description**
0 **---**	An object is only transformed as a consequence of a change of a neighboring object. *Example: neigboring object of an object that is enlarged*
1 **vertex_line**	The force for one vertex of the displaced object is given by the shortest distance to the push object and the distance's direction. If the shortest distance is greater than the required minimal distance the force becomes 0. *Example: enlarge a polyon*
2 **line_vertex**	The distance of every vertex of the push object is calculated to the nearest vertex of the displaced object. If this distance is greater than the shortest distance to the line multiplied by 1.5 a new force is evaluated between the vertex on the push object an the nearest position on the displaced object. If the nearest position is within the first/last 30% of the corresponding line segment the force is maped on the start/end vertex of the segment; if the position lays in between half the force is maped on the start and the end vertex. *not used alone*
3 **combined**	The force on one vertex of the displaced object results from the sum of forces on that vertex computed by force model 1 and 2. *Example: exaggerate a polygon*
4 **vertex_line_max**	The force acting on all vertices of the displaced object is equal to the maximum force computed by force model 1 for one of the vertices. *Example: displace a polygon*

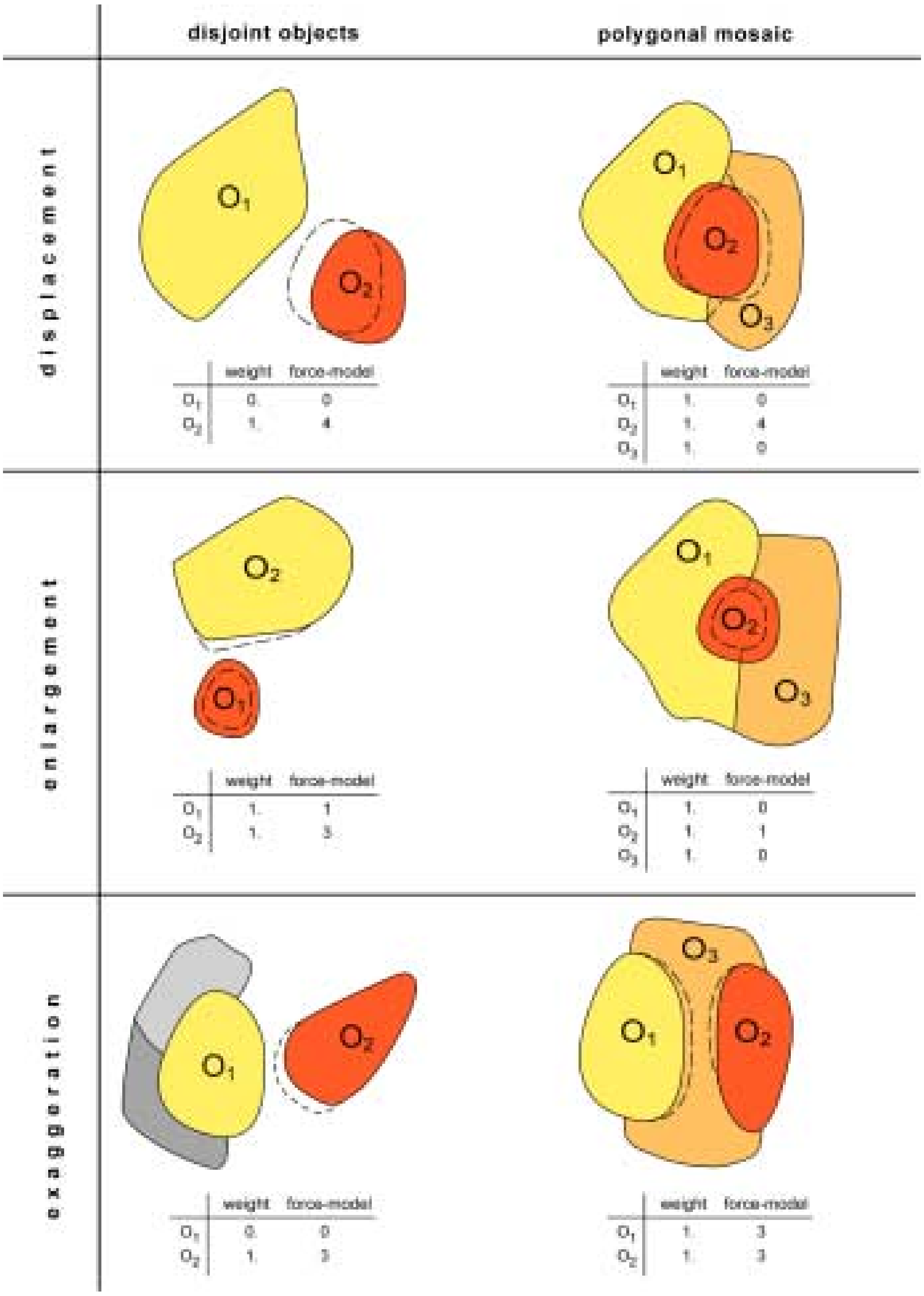
Table 2: Different force models needed for the adaptation of the snake method to the generalization operators displacement, enlargement and exaggeration.

---

[6] We recall that a snake is a specific type of a spline. Thus, it is not the case that vertices next to such nodes are fixed in their position automatically.

[7] Metric constraints mainly deal with aspects of perceptibility such as size, width and separability (Peter and Weibel 1999).

In all 4 different force models ways to compute forces between objects are needed for the realization of these three operators. A force model value ‘0’ denotes that a polygon is at least transformed as a consequence of a change of the neighboring objects, but only if the object’s weight is greater than 0, that is, the object reacts to transformations of the polygon mosaic but doesn’t try to solve a conflict actively. The model ‘vertex_line” designates the basic approach of the force computation, while model 2 is an additional extension to model 1 and model 3 the mathematical sum of model 1 and 2. The model ‘vertex_line_max’, on the other hand, returns only the maximum force and its direction of model 1. Details on the different ways to compute forces are summarized in table 2 as well as in application examples given below. The object to be displaced is named ‘displaced object’ and the object that pushes it away is called ‘push object’ for reasons of clarity.



displacement, disjoint objects:

	weight	force-model
O_1	0	0
O_2	1	4

displacement, polygonal mosaic:

	weight	force-model
O_1	1	0
O_2	1	4
O_3	1	0

enlargement, disjoint objects:

	weight	force-model
O_1	1	1
O_2	1	3

enlargement, polygonal mosaic:

	weight	force-model
O_1	1	0
O_2	1	1
O_3	1	0

exaggeration, disjoint objects:

	weight	force-model
O_1	0	0
O_2	1	3

exaggeration, polygonal mosaic:

	weight	force-model
O_1	1	3
O_2	1	3

Table 3: Displacement, enlargement and exaggeration realized by the snake method. While the solid areas describe the situation after the transformation the dashed lines show the polygons’ outline before.

The concept of the application of the snake method for polygon generalization is summarized in table 3. It illustrates the realization of the displacement, enlargement resp. exaggeration operator by the snake method through a variation of the weights and force models assigned to the individual objects. That principle of diversifying weights and forces allows either to use only one operator, e.g. the displacement graphics at the top of table 3, or to apply different snake based operators at once - for instance the left figure in the middle row demonstrates the enlargement of object O_1. As snakes support a global approach – see above - a conflict with O_2 as a consequence of the increase of O_1 is avoided by exaggerating O_2. For each operator two theoretical examples are given that each displays another conflict that requires a different setup of weights and forces. While the left column of table 3 generally shows some ‘basic’ conflicts involving disjoint objects the right column intends to give an impression how more complex conflicts in polygon mosaics can be solved by snake-based algorithms.

To prove the proposed concept the snake method was implemented in the development environment, the commercial GIS Lamps2. The implementation follows the approach proposed by Bader (2001) and is kept generic in order to support different generalization tasks at once. First tests are solely concerned with the evaluation of the geometric transformation accomplished by snakes. Real world data, a land cover partition at the scale 1:200000 with 12 different categories, have been used. Other aspects of cartographic generalization like semantics or following a global strategy have been neglected at this point.

As a basis for the displacement of a polygon force model 4 is used, i.e. equal forces are applied to all vertices of the displace object (table 3). Forces do sometimes not result in an equal displacement of all the vertices due to the different length of the polylines’ segments. Consequently, a distortion of the object’s shape can occur. Hence the maximally calculated translation is used for all vertices of the object. Figure 7 illustrates the application of a snake-based displacement within the test data. A group of 4 polygons embedded in object O_5 is given whereby the metric constraint of a minimal distance between objects is violated several times, e.g. O_1 and O_2 or O_3 and O_4 (left image in figure 7). Because of the snake method’s global character all the conflicts can be solved during one calculation. The built-up area O_3 is fixed in its position. The forest areas O_1, O_2 and O_4 are rearranged by the algorithm in such a way that the minimal separation distance is ensured for all objects (right image in figure 7).

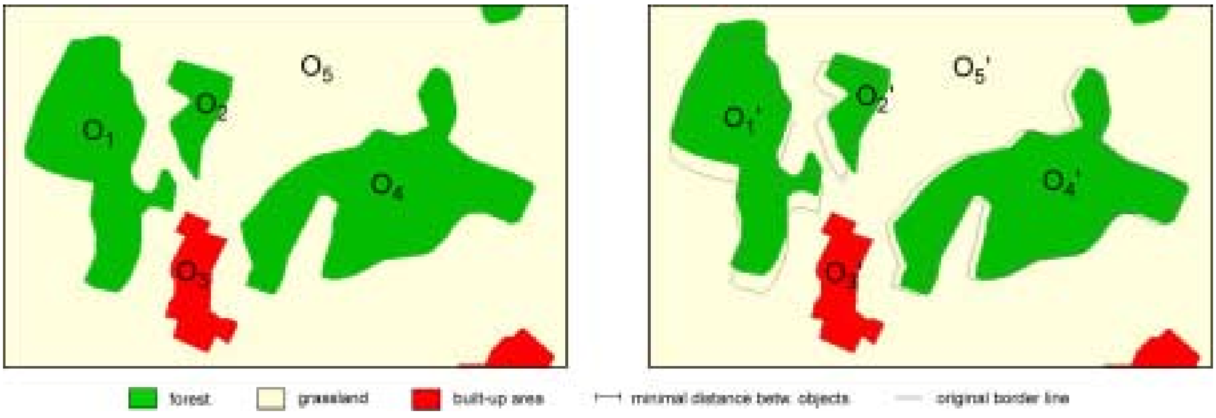


Figure 7: The displacement of several polygons based on snakes in order to ensure the minimal separation distance.

An alternative algorithm can be a displacement by either interactively defined or automatically calculated vectors. Its application may help to solve very quickly less complex conflicts, e.g. if only one object consisting of one polyline is displaced and enough empty map space is available.

The enlargement of a polygon is straightforward if there is no risk of additional cartographic errors as its consequence, e.g. an island polygon built of one polyline and further away from any nonadjacent neighbor than the minimal distance plus the amount of enlargement. In such a case an algorithm based on a scaling function is best suited.

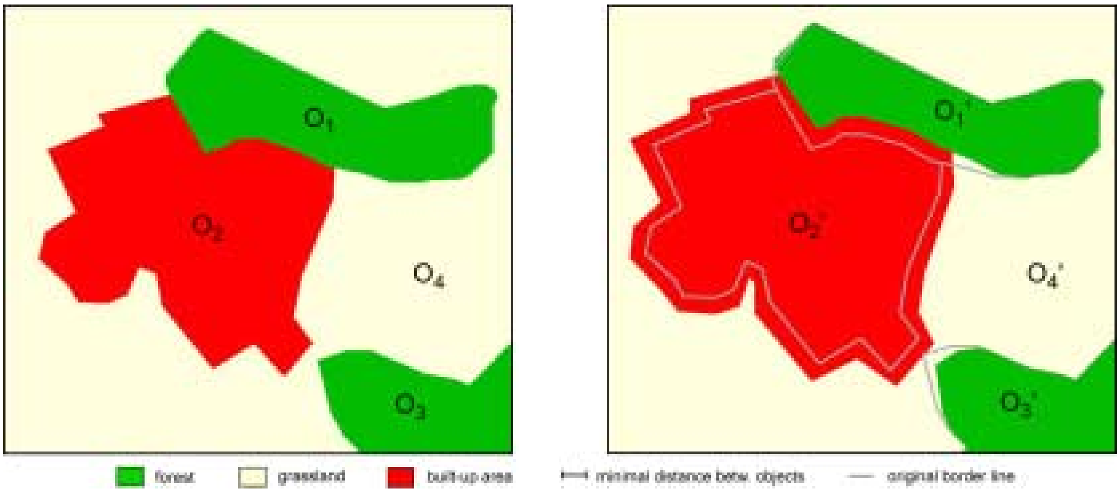


Figure 8: The enlargement of a polygon embedded in the polygon mosaic demonstrates the global approach and automated propagation supported by the snake method.

But if either the polygon object consists of multiple polylines or other objects are nearby an approach based on optimization techniques like snakes offers those advantages already discussed in section 3.1, namely a global solution, automated propagation and ease of use. These characteristics are demonstrated in figure 8, where the built-up area O_2 is enlarged and consequently the polygonal objects O_1 and O_3 are transformed in order to ensure legibility and minimal separation. For concave objects the enlargement by snakes may sometimes produce cartographically nonsatisfying solutions resp. cause conflicts between parts of an object – e.g. compare the inlet at the lower border of O_2 and O_2'. Presently, efforts are undertaken to develop a strategy to avoid such conflicts in the future. Convex objects work in general rather well.

To widen or reduce parts of an object is very similar to the line displacement investigated by Bader (2001). Therefore, exaggeration is most straightforward to be implemented by the snake method. The weights assigned to the objects allow to control the amount of exaggeration covered by the single objects – compare section 3.2.1. The method parameters described in section 3.2.1 determine the flexibility of the object's shape and the perimeter around the conflict used to cushion the transformation. In figure 9 an alternative way for ensuring the minimal separation distance between two polygons is shown. The passage between the forest areas O_1 and O_2 is too narrow (left image in figure 9). For widening it an exaggeration algorithm based on snakes is applied to these objects. The right hand image in figure 9 shows the achieved solution, illustrating well the propagation of the shape distortion at the objects O_1' and O_2'.

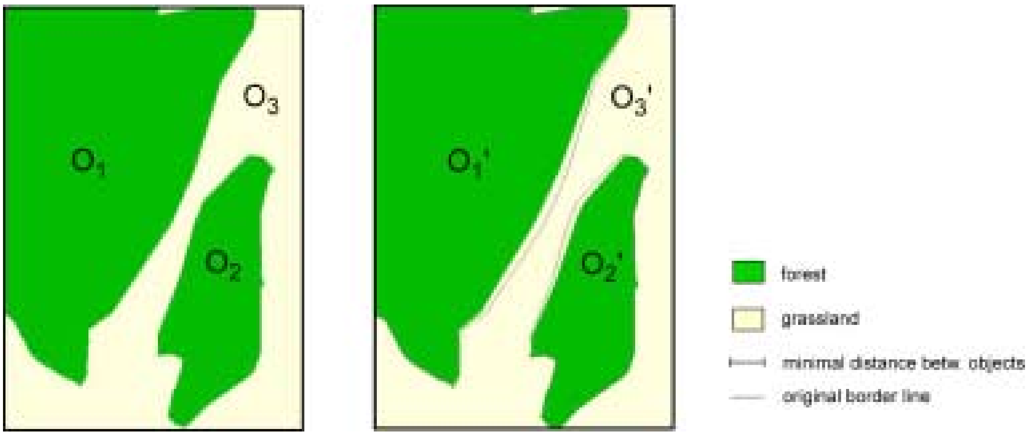


Figure 9: Exaggeration of polygons via snakes. The left figure shows the conflict before the transformation while the right one presents the achieved result.

From a cartographic point of view snakes provide cartographically pleasing results – see especially figure 6 to 9. The concept of varying the weights and force models assigned to the individual objects has both stood the test and proven the flexibility of the snake method. The experiments with real world data have shown that snakes are not only qualified well for the displacement of linear objects as indicated by Bader (2001) but also for the displacement, enlargement and exaggeration of polygons. It seems that the snake method - as basis for displacement, enlargement and exaggeration algorithms - is valuable for polygon generalization.

4 Conclusion and Outlook

At the beginning of this paper requirements for algorithms for polygon generalization were defined after having analyzed the state of the art. Optimization techniques for polygon generalization were generally evaluated. Subsequently a concept for the displacement, enlargement and exaggeration of polygons based on snakes, one particular optimization technique, was presented and discussed. Finally, first experiments with real world data substantiated the proposed ideas. The perhaps abstract discussion is complemented by a live demo during the presentation at the workshop.

In the near future more substantial experiments with both different types of real world data and different scale ranges are planned. Next, a comprehensive set of algorithms is envisaged that contains at least one algorithm for every cartographic operator needed when generalizing polygon mosaics. Afterwards the intention is to focus on topics that lead towards an automated generalization process. Thus, 3 main objectives may be dealt with:

1. Semantics and its influence on key decisions in polygon generalization.
2. Automated conflict detection based on measures proposed by Beat Peter in our research group (Peter 2001).
3. Strategies for an automated polygon generalization process resp. the development of a first coherent framework especially designed for categorical data in the vector format.

Acknowledgement

This research is part of project ‘GENDEM: Map Generalization for Thematic and On-Demand Mapping in GIS’ supported by the Swiss National Science Foundation under contract 20-52759.97.
Thanks go to Mats Bader, Mathieu Barrault, Alessandro Cecconi, Beat Peter and Robert Weibel for valuable assistance and comments.

References

AGENT REPORT D2, 1999, Selection of basic algorithms. http://agent.ign.fr/deliverable/DD2.html.

ARLETH, M., 1999, Problems in screen map design. *Proceedings Ottawa ICA 1999, 19th Int. Cartographic Conference. Ottawa.*

BADER, M., 1997, Methoden zur Erkennung und Lösung von metrischen Konflikten in der Generalisierung von Polygonmosaiken, MSc Thesis, Department of Geography, University of Zurich.

BADER, M. and BARRAULT, M., 2000, Improving Snakes for Linear Feature Displacement in Cartographic Generalization. *Proceeding GeoComputation 2000*, http://geocomp.gre.ac.uk/gc2000/e034-bader.htm.

BADER, M., 2001, Energy Minimizing Methods for Feature Displacement in Map Generalization, PhD Dissertation, Department of Geography, University of Zurich.

BURGHARDT, D. and MEIER, S., 1997, Cartographic Displacement Using the Snakes Concept. In *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps* edited by W. Foerstner and L. Pluemer (Basel: Birkhaeuser Verlag).

BURGHARDT, D., 2000, Automatisierung der kartographischen Verdraengung mittels Energieminimierung, PhD Dissertation. Institut fuer Planetare Geodaesie. TU Dresde.

BUTTENFIELD, B.P., 1999, Sharing Vector Geospatial Data on the Internet. *Proceedings 19th International Cartographic Conference Ottawa.*

COTTINGHAM, S., 1997, Generalisation of Categorical Maps: A Phenomenological Approach to Soil Mapping, PhD Dissertation. University of Edinburgh.

De BERG, M., Van KREVELD, M. and SCHIRRA, S., 1998, Topologically Correct Subdivision Simplification Using the Bandwidth Criterion. *Cartography and Geographic Information Systems* **25**(4), 243-257.

DeLUCIA, A.A., BLACK, R.T., 1987, A Comprehensive Approach to Automatic Feature Generalization. *Proceedings 13th Conference of the International Cartographic Association. Morelia*, Vol. 4., pp. 173-191.

DOUGLAS, D.H., POIKER, T.K., 1973, Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or Its Character. *The Canadian Cartographer* **10**(2), 112-123.

HARDY, P., 2000, Multi-Scale Database-Generalisation for Topographic Mapping, Hydrography and Web-Mapping, using active object techniques. http: //www. Laser-scan.com/papers/isprs2000pgh_1436.

JAAKKOLA, O., 1998, Multi-Scale Categorical Databases with Automatic Generalization Transformations Based on map Algebra. *Cartography and Geographic Information Systems* **25**(4), 195-207.

JONES, C.B. and WARE, J.B., 1998, Nearest Neighbour Search for Linear and Polygonal Objects with Constrained Triangulations. *Proceedings 8th International Symposium on Spatial Data Handling, Vancouver*, pp. 13-21.

JONES, C.B., BUNDY, L. and WARE, J.M., 1995, Map Generalization with a Triangulated Data Structure. *Cartography and Geographic Information Systems* **22**(4), 317-331.

KASS, M., WITKIN, A. and TERZOPOULOS, D., 1987, Snakes: Active Contour Models. *Proceedings of the First International Conference on Computer Vision*, IEEE Computer Soc. Press. pp. 259-269.

McMASTER, R.B. and SHEA, K.S., 1992, Generalization in Digital Cartography, Resource Publication in Geography, Association of American Geographers, Washington D.C..

MUELLER, J.C. and WANG, Z., 1992, Area-Patch Generalization: A Competitive Approach. *The Cartographic Journal* **29**(2), 137-144.

PETER, B., 1997, Ableitung von generalisieren Bodennutzungskarten aus der Arealstatistik der Schweiz 1979/85, MSc. Thesis, Department of Geography, University of Zurich.

PETER, B. and WEIBEL, R., 1999, Using Vector and Raster-Based Techniques in Categorical Map Generalization. *Third ICA Workshop on Progress in Automated Map Generalization. Ottawa*.

PETER, B., 2001, Measures for the Generalization of Polygonal Maps with Categorical Data. *Fourth ICA Workshop on Progress in Automated Map Generalization. Beijing*.

SAALFELD, A., 1999, Topologically Consistent Line Simplification with the Douglas-Peucker Algorithm. *Cartography and Geographic Information Science* **26**(1).

WARE, J.M. and JONES, C.B., 1998, Conflict Reduction in Map Generalization Using Iterative Improvement. *GeoInformatica* **2**(4), 383-407.

WEIBEL, R., 1991, Amplified Intelligence and Knowledge-Based Systems. In *Map Generalization: Making Rules for Knowledge Representation* edited by B.P.Buttenfiled and R.B. McMaster (London: Verlag Longman), pp. 172-186.