Measures for the Generalization of Polygonal Maps with Categorical Data

Beat Peter
Department of Geography
University of Zurich
Winterthurerstrasse 190
CH-8057 Zurich / Switzerland
Phone: +41 1 635 51 51
Fax: +41 1 635 68 48
Email: beatp@geo.unizh.ch

Abstract

Categorical data in the form of polygon mosaics is an important and frequent data type in GI Systems. However, tools and methods which support the automatic generalization of this type of data are not yet well developed. The work described in this paper is part of a broad initiative at the Geography Department of the University of Zurich to change this by focusing on measures and methods of spatial analysis which form the foundation of every generalization task. The goal is to provide formal mechanisms which allow describing all relevant characteristics of datasets with categorical data and respecting semantic information as well. Appropriate measures are required throughout the generalization process. We need them to identify conflicts, to guide and control transformation operations and for the evaluation of the results. After a short recapitulation of the cartographic constraints applicable to categorical data and the repetition of some basic requirements for measures in chapter 2, we start with some conceptual considerations about the potential and limitations of measures in the generalization process in chapter 3. In chapter 4, we present and discuss measures for the above mentioned purposes and describe procedures for their computation. The measures are classified according to the main characteristic they represent. Examples show how a measure or a combination of several measures can be used for a specific purpose in the transformation (generalization) process. Chapter 5 deals with the workflow in a generalization system from the measures perspective and, finally, chapter 6 presents conclusions and an outlook to the next steps in this project.

1. Introduction

Categorical data in the form of polygon mosaics or as raster data sets are an important and frequent data type in today's Geographic Information Systems. Examples cover a wide range of topics from landuse/landcover maps over datasets with geological information to maps with political units. While significant progress has been made in the development of integrated systems for the generalization of topographic maps (e.g. AGENT 2001) equivalent solutions which are specifically designed for categorical data are still missing.

Research in the field of generalization of categorical data has a long tradition at the Geography Department of the University of Zurich. A number of papers dealing with various aspects of categorical map generalization have been published in the last few years (Peter 1997, Bader 1997, Bader and Weibel 1997, Peter and Weibel 1999b). The work described here is basically the continuation of the research project we presented at the ICA Conference in Ottawa in 1999 (Peter and Weibel 1999a) where we developed a conceptual framework for a constraints based approach to categorical map generalization and exploited the potential of procedures involving transformations of the data model. Building on this general framework we will now continue by extending our knowledge and understanding of some of its key components, namely measures. We will work with polygonal data from now on.

The goal of this paper is to develop a comprehensive set of measures which describes all relevant geometric and semantic properties of selected types of categorical maps and to test them with real data. Measures are of central importance in a generalization workflow. They are needed in every step of the process: for initial evaluation of the data and basic structure analysis, to identify cartographic conflicts and to guide and control the transformation process on the generalization operator and algorithm level as well as for quality evaluation and assessment

of the results. Since both the measures themselves as well as the form in which they are represented in a generalization system are closely linked with the available operators and algorithms, we will closely cooperate and coordinate the work with the project of Martin Galanda who is working on algorithms for polygon generalization (Galanda 2001).

Chapter 2 presents a short recapitulation of the constraints to categorical data, some basic definitions and formal requirements for measures as well as a classification scheme. Chapter 3 presents conceptual considerations. We discuss the potential and limitations of measures in a generalization system and answer the question what measures are useful in what situation and for which types of categorical data. In chapter 4 we define measures for the purposes mentioned above and show ways for their calculation. Possibilities for the integration of measures in a generalization workflow are discussed in chapter 5. Finally, chapter 6 provides conclusions and an outlook to future work.

2. Constraints and Measures – Definitions, Properties and Requirements

This chapter is a condensed recapitulation of the respective chapters in Peter and Weibel (1999b) where we describe the terms in more detail. Modifications reflect the fact that we now work exclusively with data in the vector model (polygon mosaics or polygonal subdivisions). Since the terms below will be used throughout the paper and the constraints to categorical data form the foundation not only of this paper but for the whole research project, we repeat them here. This should also allow the reader to follow the argumentation more easily. Readers who are familiar with these concepts can skip this chapter.

2.1 Constraints to Categorical Data

Classification of Constraints
We classify constraints both according to their function and their spatial application scope.

Function:
- Graphical
- Topological
- Structural
- Gestalt

Spatial application scope:
- Object or polygon [1] (micro level)
- Category (macro level)
- Group of objects, region or partition of the dataset (meso and macro level)

Constraints Related to Objects
1. *Minimum size* (graphical): Objects which are too small must be either deleted or enlarged
2a. *Minimum distance* (graphical): The distance between consecutive vertices of a polygon outline should not be less than the minimum visual separability distance
2b. *Self-coalescence* (graphical): The distance between any vertices of a polygon outline should not be less than the minimum visual separability distance
3. *Separability* (graphical): The distance between two objects should not be less than the minimum visual separability distance
4. *Separation* (topological): Avoid separation of objects when deleting parts of it
5. *Islands* (topological): Objects which can be identified as islands may be deleted or enlarged but should not be amalgamated with other objects of the same category to avoid changes of topology
6. *Self-intersection* (topological): Avoid introduction of self-intersection of object outlines
7. *Amalgamation* (structural): Disjoint objects of the same category may be amalgamated if they are close enough and the area in between is not specifically protected
8. *Collapsability* (structural): The area of eliminated objects should be distributed among the neighboring objects such that no holes are created
9. *Shape/Angularity* (structural): Respect the global shape and angularity of objects

[1] We define constraints for individual objects or polygons although at least two objects are always affected by any geometric transformation because we work with (plane filling) polygon mosaics.

Constraints Related to Categories

10. *Size ratio* (structural): Respect the size ratio for each category relative to the total area
11. *Shape/Angularity* (structural): Respect typical shapes and angularity of objects of each category
12. *Size distribution* (structural): Respect the given size distribution of objects for each category
13. *Alignment/Pattern* (Gestalt): Preserve typical alignments and patterns of objects of a category

Constraints Related to Partitions or Groups of Objects

14. Neighborhood relations (topological): Preserve given neighborhood relations
15. Spatial context (structural): Avoid introduction of illogical neighborhood relations (e.g., house in a lake)
16. Aggregability (structural): Allow aggregation of categories if required and suitable super-categories exist
17. Auxiliary data (structural): Observe constraints imposed by auxiliary data (e.g., roads, rivers, point features)
18. Alignment/Pattern (Gestalt): Preserve typical alignments and patterns of objects within the map or within a group of objects
19. Visual balance (Gestalt): Avoid gross changes in shape and distribution of objects, unless required by extreme scale change
20. Equal treatment (Gestalt): Ensure equal treatment within a partition of the map and avoid highly unequal treatment across all partitions.

2.2 Measures

Key Aspects of Measures

Among the important key aspects of measures are: They

- are defined as procedures for computing *measurements* (numerical values) or binary values in the case of topological or semantic tests.
- can represent *semantic information* or *domain knowledge* in an appropriate format, for instance, as *text strings*, as *tables* or as *flags* in the database.
- allow assessing the *need* for and the *success* of generalization operations.
- are used to make *strategic* and *tactical* decisions in the generalization process.
- can be *simple* (e.g. area calculation) or *complex* (e.g. require computation of auxiliary data structures such as triangulations).
- can be *absolute* (intrinsic) or *relative* (extrinsic) depending on whether one or more states of the object in the database needs to be evaluated for a correct interpretation of a situation (e.g. before and after transformation).

Requirements for Measures

A useful measure should satisfy the following criteria: Ideally, it should

- describe the intended property as precise as possible and should not be influenced by other effects (orthogonality),
- be insensitive to outliers (robustness),
- be invariant to geometric transformations (geometric invariance),
- produce different results for different configurations of characteristics and similar results for similar configurations (differentiation),
- be easy to calculate (ease of calculation),
- be easy to use with only a limited number of parameters (ease of use), and
- be easy to interpret (ease of interpretation). Ideally for a certain value (a measurement) only one possible configuration of the measured property should exist.

Classification of Measures

Measures can be classified according to the main characteristic they represent. The classification scheme below is influenced to a large degree by the corresponding constraints for categorical data. However, some measures may express more than one property, for example, the *area* of an object in combination with its *perimeter* contains also information about the object's shape. For practical reasons we distinguish the following classes of measures:

- Size measures
- Distance and proximity measures
- Shape measures
- Topology measures

- Density and distribution measures
- Pattern and alignment measures
- Semantic measures and information

3. Conceptual Considerations for the Computation of Measures

3.1 Measures – Potential and Limitations

A well-designed set of measures is a key element in every generalization system. It allows developing an appropriate generalization strategy and, since measures can be computed faster and more easily than geometric transformations in most cases, can therefore help reducing the number of time consuming feedback loops and the extensive usage of backtracking mechanisms. By providing measures at all spatial levels from local to global (or micro to macro), we provide the means to resolve local conflicts introduced by scale reduction while *simultaneously* controlling the impact of the transformation on the involved neighborhood and the whole dataset. This approach may lead to the selection of different generalization operators or algorithms for conflicts of the same type, for instance, some objects violating the minimum size constraint might be enlarged while others are deleted. Generally speaking, the role of measures is not only to identify conflicts but also to provide clues on what options (operators) exist for their resolution and what transformation operations (algorithms) should be applied. Using measures to their full potential therefore means that we try to find *cartographically* acceptable solutions to every problem within our reach.

If we compare how a cartographer perceives the structure of a dataset he/she wants to generalize and to what degree we can represent this complex (and partly subjective) spatial reasoning process with measures, we cannot expect to be able to fully reproduce it. For practical reasons it is virtually impossible to provide measures for every theoretically possible situation. It can be expected that problems due to insufficient measures increase when the target scale decreases because difficult to formally recognize higher order structural elements become more and more important as does the *artistic* component of map generalization. A second reason that prevents us from describing every facet of a dataset formally is the variation of the different types of categorical data as well as the data acquisition methods that were used which of course also have an influence on the significance and reliability of measures in a generalization process. A third limiting factor is the comprehensive view of a cartographer on the data as well as on their spatial and thematic context. We can only partially replace this informal knowledge with auxiliary data such as digital terrain models, road networks and point data or with information acquired through expert interviews.

3.2 What Should be Measured and When?

For a comprehensive set of measures for the generalization of categorical data we think of a system consisting of *mandatory* and *optional* measures. The major part consists of mandatory measures which can be used for all types of categorical data. The *area* of a polygon or the *average size* of the polygons of a category for example are measures where we can assume that they are needed regardless of the type of categorical data we want to generalize. Optional measures are only computed for specific types of categorical data. Such measures relate to higher order spatial structures, such as *patterns* and *alignments* and to semantic information for specific data types. The reason why we propose such a strategy has several reasons. Firstly, the identification of patterns or alignments is very complex from the computational and the algorithmic perspective. Secondly, we can think of many types of categorical data where the spatial variables of a polygon (e.g. area and shape) and its category are completely independent from the one of its second or third order neighbors. Treating such polygons as a group of objects in the generalization process would then be just wrong and could also have a negative influence not only on the generalization operations which are applied to the concerned partition but also on the quality of the resulting map. The concept how we plan to implement mechanisms which allow avoiding the computation of inappropriate or misleading measures will be presented in chapter 5.

4. Measures for the Generalization of Categorical Data

4.1 Size Measures

Size measures are very important in our measures toolbox because they are useful for all types of categorical data. We use them both as "standalone" indicators and as components complex measures. They are used on all spatial levels from micro (e.g. the area of a polygon) to the macro levels where we compute statistical indicators or plot histograms for entire categories or for partitions. Since some size measures such as *area of a polygon* are computed automatically in most GI Systems, no specific methods need to be implemented for them. Resolving size related conflicts are among the major tasks in categorical map generalization. We will provide measures for conflicts identification as well as the means that allow selecting appropriate operators and generalization algorithms. For most measures we compute absolute and relative versions. While *absolute* measures are used to identify local conflicts and their severity, we need *relative* measures for comparing polygons and categories as well as for assessing global properties of a dataset. The generalization operators most often associated with area measures are *reclassify*, *delete* and *enlarge*.

Measures:

Number of polygons

n_i, N — n_i is the number of polygons of category i. N is the total number of polygons in the dataset.

Although the number of polygons of a category or the entire dataset is not size measures, we define them at this point because these numbers are components of many measures we define later in this section.

Area

$a_{\min}$ — a_{min} is the user specified minimum area of a polygon at target scale.

a_{min} can either be defined for the whole dataset or separately for each category. It is the minimum area at which the category of an object, represented by a color fill, can be clearly identified. When a_{min} (and other minimum dimensions as well) is defined we have to take into account that the outlines of each polygon are represented by a thin black line. Besides target scale, the size of a_{min} is also influenced by various other *map controls*, for instance, screen resulution.

a_{ij} — a_{ij} is the area of polygon j of category i.

$A_i = \sum_{j=1}^{n} a_{ij}$ — A_i is the total area of all polygons of category i.

A — A is the total area of the dataset.

Area is one of the most important measures in categorical map generalization. At the micro level (individual polygon) we can identify conflicts once we have defined a minimum area a_{min}. At the meso and macro levels, the area measures defined here will mostly be used as components for the computation of the measures we introduce below.

Relative area

$ra_{ij} = \frac{a_{ij}}{A_i} \cdot 100$ — ra_{ij} is the percentage of the area of polygon j of category i of the total area of category i.

$RA_i = \frac{A_i}{A} \cdot 100$ — RA_i is the percentag of the area of category i of the total map area.

Relative area is an important measure because it allows us assessing the importance of an individual object in its context. Together with other indicators we can establish formal guidelines which operator to use for its generali-

zation (e.g. *delete* or *enlarge*). As always, this process can then be modified by integrating semantic information. At the macro level, comparing the RA_i values of categories can be used for strategic decisions. We can, for example, define a dominating category with a very high RA_i value and only few but large objects as *background* and generalize it by generalizing the polygons of the *other* categories (Peter 1997). The RA_i value is therefore also a density measure which will be discussed later in this section. If we discover an extremely unequal distribution of the RA_i values we should consider a *reclassification* of the data into fewer categories before we start applying geometric transformations.

Number of polygons below minimum area per category

$$nb_i; a_{ij} < a_{\min}$$

nb_i is the number of polygons of category i with an area below the defined minimum area a_{min}.

Relative number of polygons below minimum area per category

$$rnb_i = \frac{nb_i}{n_i} \cdot 100$$

rnb_i is the percentage of the number of polygons of category i below the minimum area a_{min} compared to the total number of polygons of category i.

Total area of polygons below minimum area per category

$$AB_i = \sum_j a_{ij}; a_{ij} < a_{\min}$$

AB_i is the area of the polygons of category i with an area below the defined minimum area a_{min}.

Relative area of polygons below minimum area per category

$$RAB_i = \frac{AB_i}{A_i} \cdot 100$$

RAB_i is the percentage of the area of category i which lies in polygons with an area below the defined minimum area a_{min} compared to the total area of category i.

This group of measures is very important to decide which operator (e.g. *delete* or *enlarge*) to use for the resolution of area conflicts. Since the goal of cartographic generalization is to retain the balance of the areas of the individual categories in the target dataset, we have to assess how many polygons of each category are affected by minimum size conflicts and what total area they represent. Using appropriate criteria (see below), we can apply the operators mentioned above and limit the shift in area distribution between categories. We would, for example, rather *enlarge* polygons of a category that occurs entirely in small objects than *delete* them to prevent this category from disappearing in the generalized dataset. If the above indicators show low values for several categories, *reclassification* of the data should be considered. Both absolute and relative measures for number of polygons and area are required to assess a situation reliably.

Difference to minimum area

$$ad_{ij} = a_{ij} - a_{\min}$$

ad_{ij} is the difference between the area of a polygon j of category i and the defined minimum area a_{min}.

If ad_{ij} is positive, the polygon area is above the minimum area and the situation for that object is not identified as a conflict. If ad_{ij} is negative we have identified a violation of the minimum size constraint. The amount of ad_{ij} is a measure for the *severity* of the conflict. The strategy for conflict resolution of polygons of that category could then be to *enlarge* the objects with an area only slightly below the minimum area or within a certain bandwidth respectively and to *delete* the other ones (figure 1). Although such a procedure does not respect the spatial distribution of the polygons to be enlarged or deleted, we can assume that it will produce acceptable results in most situations. This procedure can always be modified by integrating semantic information or by defining rules (e.g. *do not delete island polygons*). The definition of the lower limit of the bandwidth can either be made automatically (e.g. 75% of the polygons or of the category area should be represented in the target map) or be based on user input. Histogram plots of the polygon area distribution per category, as shown in figure 1, can be of good use since they allow flexible and more intuitive parameter settings.

Mean polygon area

$$ma_i = \frac{A_i}{n_i}$$

ma_i is the mean polygon area of category i.

$$MA = \frac{A}{N}$$

MA is the mean polygon area of the entire dataset.

Polygon area coefficient of variation

$$acv_i = \frac{\sqrt{\sum_{j=1}^{n}\left[a_{ij} - \left(\frac{A_i}{n_i}\right)\right]^2}}{ma_i} \cdot 100$$

acv_i is the variation in % in polygon area of polygons of category i relative to the mean polygon area of category i.

$$ACV = \frac{\sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}\left[a_{ij} - \left(\frac{A}{N}\right)\right]^2}}{MA} \cdot 100$$

ACV is the variation in % in polygon area relative to the mean polygon area.

The main application scope of statistical measures like the two mentioned above is the evaluation of generalization results. In a good solution, these indicators would only change within a certain bandwidth both within and between categories. Many statistical indicators assume a normal distribution of the population (polygons). Since this will most likely *not* be the case for polygons of the various types of categorical data, such measures should only be used with care. The *polygon area coefficient of variation* (FRAGSTATS 1994) partially resolves these problems but should still only be used together with other indicators to avoid misinterpretation of the structure of the categories or the datasets.

Perimeter

$$p_{ij}$$

p_{ij} is the perimeter of polygon j of category i.

Length of common boundary with neighbor polygons

$$cb_{ij,kl}; i \neq k, j \neq l$$

$cb_{ij,lk}$ is the length of the common boundary of polygon j of category i and its neighbor polygon l of category k.

(Weighted) relative length of common boundary with neighbor polygons

$$rcb_{ij,kl} = \frac{cb_{ij,kl} \cdot w_{ik}}{p_{ij}} \cdot 100; i \neq k, j \neq l$$

$rcb_{ij,kl}$ is the percentage of the length of the common boundary of polygon j of category i and its neighbor polygon l of category k, multiplied by a factor w_{ik}, representing the neighborhood relations between categories i and k, compared to the total perimeter p_{ij} of polygon j of category i.

Two of the three measures above deal with the *context* of a polygon. *Perimeter* itself can be interpreted as a simple shape measure together with *polygon area*; a long perimeter in connection with a small area indicates a complex or elongated, non-circular shape. Better shape measures will be discussed latter in this section. Besides providing information about the neighborhood and the length of the shared segments with other polygons for every object, the later two measures are of central importance when the area of a (small) polygon needs to be split among its neighbors. The *rp* measure indicates the *relative length* of the common boundary. The weight factor w_{ij} allows respecting *semantic knowledge* about the involved category pair if desired. This measure is derived from FRAGSTATS' *Edge Contrast Index* (FRAGSTATS 1994). All perimeter and common boundary measures can also be calculated on a per category basis or for an entire dataset (FRAGSTATS 1994). While

these indicators may be useful in landscape fragmentation analysis (the original application domain of the FRAGSTATS package), we cannot think of how to usefully interpret them (or the changes respectively) in the context of cartographic generalization. Figure 2 shows the *cb* measure in a complex situation.

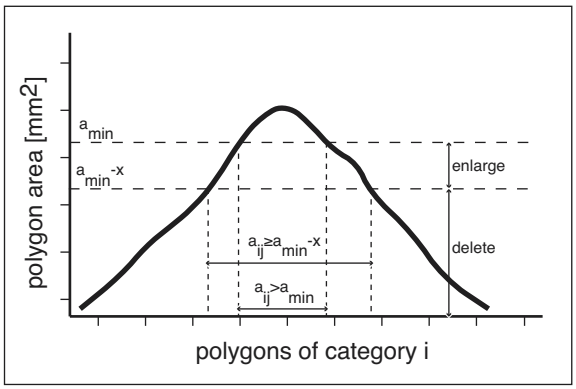


Figure 1: Histogram plot of polygon area distribution. Polygons with an area within a bandwidth x from the minimum area a_{min} are enlarged while the others are deleted.

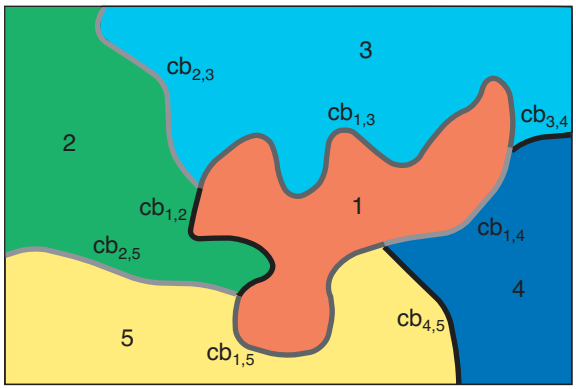


Figure 2: Length of common boundary in a situation with 5 polygons.

4.2 Distance and Proximity Measures

Distance and *proximity* measures are very important for the generalization of all types of categorical data. Their spatial application scope reaches from micro to meso and we use them to identify and resolve conflicts within and between polygons. Distance and proximity are closely related terms. Every distance related conflict is also a violation of the proximity constraints but *not* vice versa. While a distance conflict requires being resolved (e.g. because two vertices of a polygon are too close to each other), proximity expresses rather an option; we can for instance aggregate two objects of the same category because they lie within a specified distance of each other. The operators associated with distance and proximity measures are *simplification, exaggeration* and *enlargement* for conflicts within a polygon. The options when two polygons are involved are *aggregation, displacement, exaggeration* and *typification*. Which operator to choose and which generalization algorithm, (e.g. uniform or non-uniform displacement), depends on the semantic context of the situation and on the available space in the neighborhood of a conflict. We will provide measures which allow identifying conflicts reliably and resolving them in a cartographically appropriate way.

Measures

Minimum distance

$d_{\min}$ — d_{min} is the minimal visual separability distance.

d_{min} is normally identical for all categories of a dataset and depends both on target scale and on the various map controls such as the intended map purpose and output media. It is the shortest distance at which we still can clearly visually separate two polygons or identify all parts of one polygon respectively.

Consecutive vertex distance

$cvd = d_{aj,(a+1)j}$ — *cvd* is the distance between two consecutive vertices of a polygon *j*.

If *cvd* is smaller than d_{min},, we have identified a conflict. The two conflicting vertices can either be *replaced* by a new vertex at the position of the arithmetic mean of the coordinates of the conflicting vertices or the two vertices are *displaced* in such a way that no new conflicts are created. The latter option prevents the introduction of unnatural object shapes.

Vertex distance

$vd = d_{aj,bj}$ — *vd* is the distance between two non-consecutive vertices of a polygon *j*.

If *vd* is smaller than d_{min},, we have identified a part of an object that is too narrow. Because of the *duality* of problems in categorical map generalization of polygon mosaics, we have, in many cases, also identified a distance conflict *between* two polygons. The identification of problematic polygons is relatively straightforward. Using *inside buffers* of size half the minimum distance, we compute the so-called *core areas* of an object (FRAGSTATS 1994). A polygon with none or more than one core after buffer application (buffer size $1/2*d_{min}$) has at least one part which is too narrow. Bader (1997) has implemented methods which allow both identification of the conflicting vertices and resolving the conflict by displacing them from each other. Depending on the situation, *displacement* of vertices and widening gaps is in general the appropriate operator for the resolution of distance related problems within a polygon but *simplification* by removing bends in the objects' outline is also an option. Removing a narrow part by using an *aggregation* operator from the viewpoint of the neighboring polygons would mean to split one polygon into two separate objects and therefore to introduce topological error. Figure 3 shows the concept of both the *cvd* and the *vd* measures.

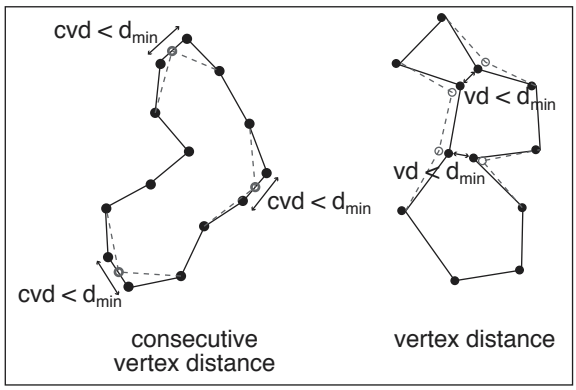


Figure 3: Distance conflicts between consecutive vertices and between non-consecutive vertices of a polygon and possible solutions (dashed line).

2nd order distance (distance between non-neighboring polygons)

$sds = d_{ia,ib}$ — *sds* is the shortest distance between two polygons of category *i*.

$sdd = d_{ia,jb}$ — *sdd* is the shortest distance between two polygons of different categories.

sds and *sdd* relate to proximity problems between separate polygons (figure 4). We distinguish two cases: proximity between two objects of the same category and between polygons of different categories. For the first case, we have (besides the possibility to *delete* one polygon) two alternatives. We can either *displace* one or both objects or we can employ the *aggregation* operator. Aggregation is in general used if both candidate objects are *quasi* island polygons and lie inside a big object of a dominating category. This is also the situation where we can use the aggregation operator to resolve the distance conflicts mentioned in the last section without introducing severe topological error. Aggregation of polygons can of course also be employed for cartographic reasons beyond pure conflict resolution to improve the quality of a map for a specific purpose. In the second case, if we have to deal with proximity problems between polygons of different categories, we have only the displacement operator available for generalization. Because we compute *2nd order distances* for all polygons up to a reasonable maximum search distance d_s, we have also enough information available to estimate if displacement is at all possible and how far we can move one or both of the conflicting polygons (or parts of them) without creating new proximity conflicts (see figure 4). The quality of the results of both operators depends on how the generalization methods work. Aggregation of polygons should be executed in such a way that the part bridging the gap is integrated in the overall shape of the resulting object in a harmonic way and the relative distance of polygons should be maintained as much as possible when they are displaced. Algorithms respecting

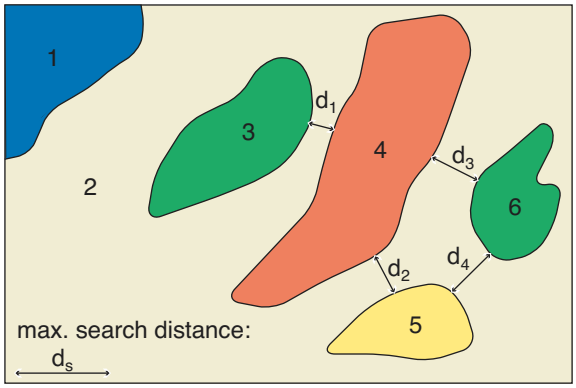
these issues have been implemented by Bader (1997, 2001) and will be adopted for polygonal data by Galanda (2001).

Weighted 2nd order distance (cost distance)

$$wsds = d_{ia,ib} \cdot w_j$$

sds is the shortest distance between two polygons of category i, adjusted by a weighting factor representing the cost to cross the area of category j.

wsds is a modification of the measure introduced in the last section. By using weighting factors, we can integrate domain knowledge and semantic information in the generalization process and compute so-called *cost distances*. These can rather be seen as a complementary measure to the simple Euclidean distances we compute above because we neither want to use them for conflict identification nor to estimate how much space he have available for the displacement of polygons. The main application scope of cost distance measures is the *theme driven aggregation* of polygons beyond minimum distance conflicts. Therefore, it is only usefully computed between objects of the same category. By assigning to a category a low weighting factor we can facilitate aggregation of other polygons across its territory while a high weighting factor acts like a barrier. In landuse maps, for example, we would normally prevent aggregation of polygons across lakes by assigning them a high weight factor while aggregation across grassland belonging to the dominating categories would be promoted by assigning this category a very low weighting factor.



Id	1	2	3	4	5	6
1						
2	N					
3	>d_s	N				
4	>d_s	N	d_1			
5	>d_s	N	>d_s	d_2		
6	>d_s	N	>d_s	d_3	d_4	

N = Neighbor

Figure 4: 2nd order distances for a situation of 6 polygons. Distances are only computed up to the maximum search distance d_s.

Difference to minimum distance

$$dd = d - d_{\min}$$

dd is the difference between a computed distance d and the defined minimum distance d_{min}

If *dd* is negative, we have identified a conflict related to distance. Similar to ad_{ij} for area measures, the amount of *dd* expresses the severity of a conflict and can therefore be used to guide the operator selection process.

4.3 Shape Measures

Shape measures can be computed for entities at various spatial levels. At the micro level we mainly want to identify characteristic shapes of individual objects, while at the macro level, we compute *shape indicators* for entire categories to compare the values before and after generalization. Except for one special case that we will discuss below shape measures are not directly used for conflict identification. One of their main application scopes is the evaluation of generalization solutions. Therefore we will compute indicators which allow assessing the *changes* in shape that have taken place during the transformation process both at the micro and the macro level. The second and more important application scope of shape measures is *shape recognition* and the identification of *characteristic shapes* of specific types of categorical data at the micro level to guide the operator selection process. If we are able to reliably identify complex structures and preserve their overall characteristics during data generalization, it can be expected that the visual appearance of our results will be of good quality.

Measures

Zero core area

zca_j — zca_j identifies a polygon j which disappears from the dataset after the application of an inside buffer of size $1/2*d_{min}$ (Boolean value).

zca_j can be interpreted as a special case of a minimum area conflict. With this measure we can identify long and thin objects. Since their *area* is above the defined minimum a_{min} we would normally *enlarge* them or select other appropriate alternatives to preserve them for the target dataset. Figure 5a illustrates the idea of this measure.

Shape index

$$si_{ij} = \frac{p_{ij}}{2\sqrt{\pi \cdot a_{ij}}}$$

si_{ij} describes the shape of polygon j of category i compared to a standard circular object of the same area.

Category shape index

$$csi_i = \frac{\sum_{j=1}^{n} p_{ij}}{2\sqrt{\pi \cdot A_i}}$$

csi_i describes the shape of all polygons of category i compared to a standard circular object of the total category area.

It is obvious that the interpretation of the above shape indicators is difficult, especially if calculated for an entire category. As it is illustrated in figure 6, the shape index *si* value of a circle is 1 which is the minimum. The higher the values are, the more complex and irregular (non-circular) is the shape of an object. Like other measures, such as *fractal dimension*, the above indicators are often used for the analysis of landscape fragmentation (FRAGSTATS 1994). One of the basic properties of the majority of shape measures is that they are not unambiguous; *different configurations can produce identical values*. In the context of cartographic generalization, the individual values have no specific meaning since it is neither useful to compare the shape of polygons to circles nor is it our wish to generate circular objects through generalization transformations. The *shape index* measures are possibly useful to compare objects and to evaluate their changes after transformations have been applied. To assess the *relative* changes between categories can be interesting as well. However, since we work with polygon mosaics, where changing the shape of one particular object inevitably means that the shape of its neighbors are modified as well, we doubt that important insight can be acquired from such measures or their changes, respectively. One of the main goals of cartographic generalization is to change the shape of the objects in a map in an appropriate and controlled way suitable for a specific target scale and observing the various map controls. Therefore we believe that it will not be easily possible to establish a connection between changes of shape indicator values and the application of unsuitable generalization methods. We will nevertheless perform empirical tests to prove this hypothesis in the course of our future work.

Number of distance conflicts per arc

ndc_{jl} — ndc_{jl} designates the number of distance conflicts of the common arc (segment) of polygon j and polygon l.

Knowledge about the number of distance related conflicts, especially those between *non-consecutive* vertices within an arc of a polygon can be very important. A high value of ndc_j characterizes a complex shape such as a wiggly structure. While resolving every conflict individually could result in the selection of a *simplification* operator in every case due to lack of space, treating the conflicts as a group allows better preservation of the objects' overall shape. We could then alternate between *simplification*, *exaggeration* or typification methods, preserve every second bend and thus the overall shape of the object for the target dataset. Figure 5b shows such a situation as well as a possible solution. Methods for the formal description of such situations, as well as generalization algorithms for this procedure are well developed because it is a frequent problem for roads in topographic map generalization (e.g. Mustière and Duchêne 2001).

Characteristic shape

cs_j	cs_j designates a polygon *j* with a characteristic shape as per definition.

One of the main aims of cartographic generalization is to preserve *characteristic shapes* in the target map. cs_j is not a measure like others because it has neither a unit, nor does it express a percentage nor is it an absolute number. Its main goal is to *flag* certain objects as characteristic for a specific type of categorical data and limit the kind of generalization operators that can be applied on them. By defining this measure, we want to integrate *semantic information and domain knowledge* about shape in the generalization process. Characteristic shapes are for example *rectangles* representing agricultural fields in a landuse map or *ring structures* in geological maps. These two structures can be identified easily with automatic methods. A polygon flagged as characteristic may not be aggregated with polygons nor is it allowed performing any other (non uniform) operations on it that dramatically change its overall appearance. It may, for instance, only be *displaced* or *scaled* as an entity and distance conflicts may only be resolved through *exaggeration*. We will develop a list of characteristic object shapes for different types of categorical data and implement methods for their reliable identification. Methods for this purpose can be found in a variety of disciplines, such as in *computer vision*.

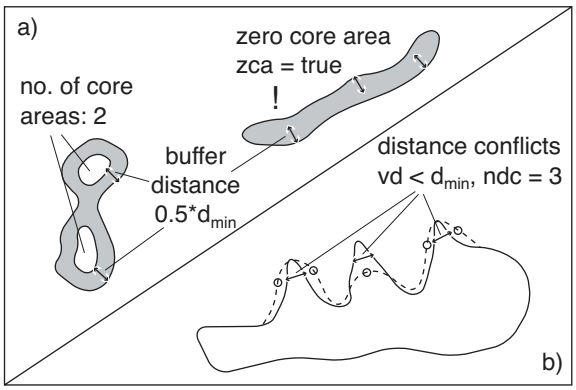


Figure 5a: Example for a polygon with zero core area = true, identifying an object with a long and thin shape.

Figure 5b: Example for the ndc measure. Three distance conflicts on the same are identified and resolved with respect to each other (dashed line).

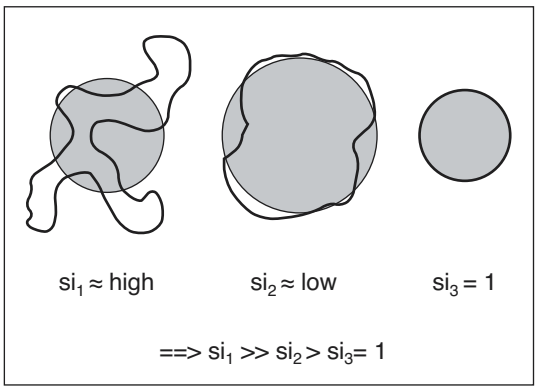


Figure 6: Shape index (si) for three objects. The minimum value of si is 1 for a circular shape. More complex shapes have higher values. Shape index is used for comparing the shapes of polygons, the value itself cannot be usefully interpreted in cartography.

4.4 Topology Measures

Topology measures have a *regional (meso* scale*)* spatial application scope because they deal with relationships between neighboring polygons. Changes in topology are a logical consequence of generalization transformations: whenever objects are aggregated or removed from the dataset, topology changes. The smaller the target scale, the more changes will occur. By defining topology measures, we do not want to prevent such modifications from taking place but rather establish guidelines to define which cases are acceptable and which ones are regarded as topological error. To do this, we have to consider geometric as well as semantic properties of the data. Our goal is to influence the operator selection process before the transformations starts by limiting the choice for the generalization of specific situations. Topology measures are non-metric, have no unit nor do they represent a percentage. Like *characteristic shape* defined in the last section they are implemented as *flags* or *tables* in the database.

Measures

Neighborhood

nh_j — nh_j contains a list with the category numbers of all neighbor polygons of a polygon *j*.

nh_j is rather a source of information than a measure. Besides using it as component for other measures such as the *length measures* defined in section 4.1, we can compute any kind of statistical analysis of the neighborhood relationships in a dataset to assess its structure at the meso or macro level.

Island polygon

i_j — i_j defines a polygon *j* as an island.

The two measures above are mentioned for completeness only since the data model of most modern GIS software packages provides topological information by default. As the name suggests, *island polygons* have only *one* neighbor and are therefore easy to identify. Since such objects should be preserved as islands in the target map for cartographic reasons in most cases, they are *flagged* accordingly. While their shape can be generalized in an appropriate way (e.g. through *simplification*), we limit the application of other potential operators; *aggregation* and *deletion* are not allowed, only *displacement* and/or *exaggeration*. Such methods could eventually also be implemented for other polygons that are not necessarily islands in order to meet specific semantic requirements.

Unwanted neighborhood

unh — *unh* contains a prioritized list of neighborhoods which are not allowed or should be avoided for each category.

This measure can be implemented as a *list* or *look-up table* containing neighborhoods to be avoided for each category. This table can be consulted for instance when the area of a polygon to be removed has to be distributed among its neighboring objects. *Deletion* is the operator most changing the topological relationships in a map especially in complex cases where several categories are involved. Although we have introduced measures which allow controlling unequal splitting of objects under size measures (e.g. the *relative length of common boundary*), we do not explicitly exclude categories from becoming neighbors. Using a structured and prioritized list will provide the means to avoid unwanted or unnatural neighborhoods *as far as possible* by integrating semantic information about the type of categorical data to be generalized. Of course, this list computed for *unh* should be in accordance with the one for nh_i at the object level. From an opposite point of view, the last items in the list for each category can be regarded as the *preferred* neighbors if it turns out to be impossible to avoid a topological conflict completely. If a category is not mentioned in the table at all, it is a preferred neighbor from a positive point of view.

Separation of polygons

sep_j — sep_j designates a polygon *j* which has been separated into two or more polygons during generalization transformations.

The value of sep_j (true or false) can only be computed *after* completing the transformations for the respective map area. sep_j indicates a *topological error* introduced by the *aggregation* operator which has been applied to two of object *j*'s neighbors. If such a case is discovered, the aggregation operation has to be revoked and an appropriate alternative method has to be selected to resolve the conflict, such as *exaggeration* of polygon *j* and *displacement* of its neighbors (see figure 7). Of course, it would be much more efficient to identify potential candidate situations for this kind of problem *before* transformations take place. However, since it is very costly from the algorithmic point of view to comprehensively describe them and because they are not frequent in most types of categorical data, we refrain from implementing them at the moment.

Self-intersection

sis_j	sis_j designates a polygon j where self-intersection of its outlines has been introduced during generalization transformations.

Some generalization algorithms produce *self-intersections* of a polygon outline (e.g. Douglas-Peucker) or between polygons and can therefore introduce topological error to a generalized dataset. Because procedures to test for self-intersections will be implemented at the algorithm level at run time or at least before the generalization operation has terminated, we do not have to take care of this problem by providing a dedicated measure.

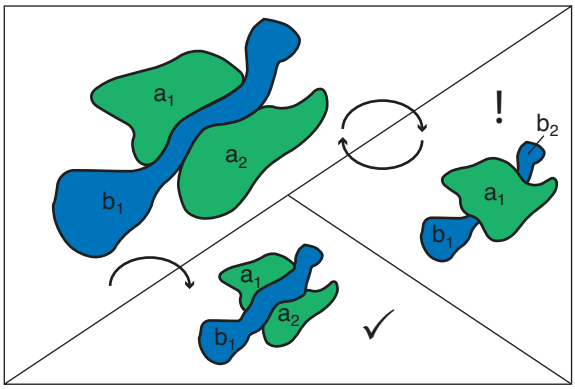


Figure 7: If sep_j *is true, a polygon has been wrongfully separated by generalization transformations. The operation is revoked and a different operator respecting topology is applied.*

Auxiliary Data and Topology

Besides providing valuable semantic information the integration of auxiliary data such as point information in the generalization of categorical data imposes strong topological constraints and therefore drastically limits the options we have available for the operator selection to execute generalization transformations. A fundamental requirement for all such generalization solutions is that the containment of points in polygons may not be altered during generalization (de Berg et al. 1995). This creates severe problems for such operators like *simplification* and *displacement*. The *deletion* of objects is no longer an option. The concept of defining *characteristic shapes* would have to be extended to include also segments of objects which could then only be transformed with respect to the point data, for instance a particular bend may be *exaggerated* but not *simplified*. On the other hand, integrating auxiliary data in the generalization of categorical data will also lead to more correct results if compared with the real world. If time allows and provided we have appropriate data available, we will perform tests and evaluate the differences between generalization solutions with and without the integration of auxiliary data and their topologic potential for different target scales.

4.5 Density and Distribution Measures

Density and distribution measures relate to higher order spatial structures and describe properties of the data at the macro level. By implementing them, we try to simulate the ability of human beings to understand the structure of a dataset holistically. According to our definition, the two terms are not complementary and describe different aspects of a dataset. While density focuses rather *on* specific objects, for example all polygons of a category, a measure for distribution deals with the space *between* them. Therefore, the values produced for density range from *low* to *high*, whereas those for distribution are interpreted on a scale from uniform to random or clustered. In this section we define a measure that allows identifying regions with a high density of objects of a category (cluster) and two measures which allow assessing the quality of generalization solutions by comparing relative indicators for categories before and after geometric transformations.

Measures

Relative area

$$RA_i = \frac{A_i}{A} \cdot 100$$

RA_i is the share in % of the area of category *i* of the total map area.

Relative area of a category compared to total map area has already been defined in section 4.1 under area measures. Its other main application is to estimate the density of a category in a dataset. Comparing changes of the values before and after generalization and changes in the relation of values between categories can provide important information. A drastic change of RA_i for a certain category indicates that the *deletion* operator has been applied with inappropriate parameters.

Relative area of the category convex hull

$$chc_i$$

chc_i represents the area of the convex hull of all polygons of category *i* in % of the total map area.

chc_i is an indicator for the distribution of the polygons of a category in a dataset. As such, it considers only the extreme objects of each category with respect to their coordinates and does not provide any information about the density relationships inside the convex hull. Therefore, chc_i has always to be used together with other measures, especially *relative area* and *absolute number of polygons*. The main application of chc_i is the identification of cases of inappropriate parameter selection for the resolution of *minimum area* conflicts (*deletion* operator). Drastic changes of chc_i, together with a considerable reduction of the number of polygons indicates that a category is no longer adequately represent in a partition of a map (see figure 8 left). Accordingly, drastic reduction of the absolute number of polygons and stable chc_i value indicates a similar problem somewhere on the *inside* of the convex hull which requires our attention. We could probably define measures that describe the distribution of objects more precisely and reliably but these would require extensive computation (and frequent recomputation) of distances between objects. For the intended purpose we believe that such procedures are too time consuming and computationally too expensive. Empirical testing will show if our convex hull approach is sufficiently reliable with real data. A second relatively simple and easy to implement method to roughly assess the distribution of the objects of a category would be to partition the map with a regular grid and to compare the number of polygons in each cell before and after generalization. As it is illustrated in figure 8 on the right, a combination of convex hull and regular grid could resolve some of the problem of the chc_i measure and would still be easy to compute. We will experiment with this method as well.

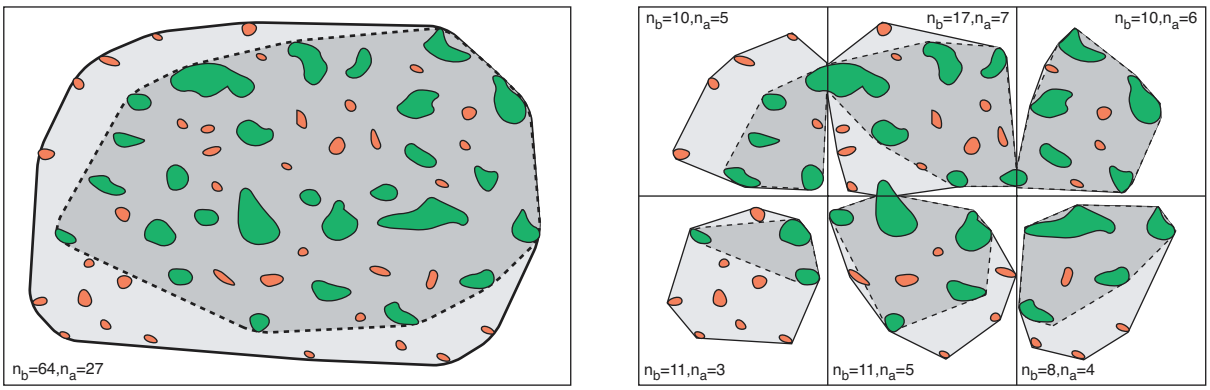


Figure 8: (left) Changes in convex hull area for the polygons of a category after generalization. Too many objects have been deleted (red/dark). (right) Applying the convex hull method in the cells of a regular grid allows localizing low-density problems more precisely.

High-density region (cluster)

hd_i	hd_i is the convex hull of polygons of category i, each of which is lying within a distance d_s to its nearest neighbor.

hd_i defines the outlines of a *cluster* of polygons of the same category. This measure is most usefully computed for binary maps or for datasets with dominating categories which can be treated as *background*. All objects of a cluster should be contained in the same (background) polygon and the presence of objects of other categories is not allowed to prevent interference during the generalization. The identification of clusters is relatively straightforward using a buffer of an appropriate size in combination with the measures for neighborhood proposed in the previous section, as it is illustrated in figure 9. The possibility to treat objects as an entity offers several advantages. We can, for example, compute a variety of the defined measures specifically for the cluster area and the polygons in it and therefore better control both the operator selection process and the generalization transformations for that area. Working with high-density regions also allows the implementation of *typification* methods. Using the outlines of the convex hull as a guideline, we can reduce the number of objects while preserving the represented area and the group's overall shape.

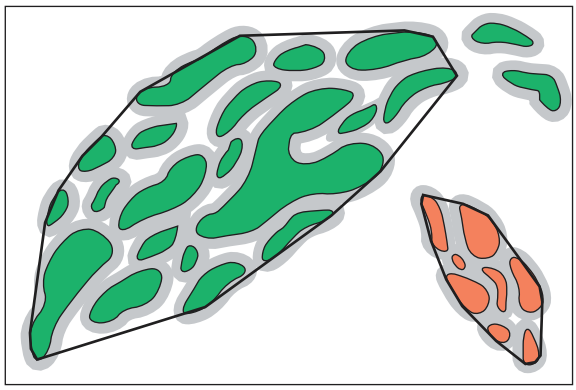


Figure 9: Clusters of polygons of the same category are identified on a homogenous background category using buffers and convex hull operations.

4.6 Pattern and Alignment Measures

The purpose of the measures we have defined so far is to formally describe the *given entities* and their structure as comprehensively as possible. The idea behind *patterns* and *alignments*, however, is different: we have *predefined* structural concepts (prototypes) in mind and try to find representatives of them in a dataset. Our definition of the term pattern designates characteristic *arrangements* and/or *sequences* of polygons in a map. The variables are category, area, shape, distance and semantic/logical relationship. Thus, an alignment can be defined as a one-dimensional pattern: the objects are arranged more or less on a virtual polyline. Pattern and alignment recognition in the context of digital cartography is a complex task. In this section we will present a pragmatic method which allows to reliably identify such structures in specific types of maps with categorical data.

Measures

Pattern

$pattern_m$	$pattern_m$ designates a user defined group of polygons which form a pattern according to geometric, semantic and logical considerations.

Recognizing global patterns in polygon mosaics, especially in those with many categories and high spatial variability is almost impossible, even for the human eye. Although we might be able to discover certain regularities in a map relating to category sequence, shape or area, we fail to explain their existence in reality in most cases with sufficient reliability. In the majority of datasets with categorical data, the *spatial variables* of a polygon are *independent* from those of other objects at some distance of it and are determined by facts we do not know (e.g.

ownership relations or soil quality). Therefore, the attempt to develop fully automated pattern recognition methods for polygon mosaics is perhaps an over-ambitious idea. We are aware that many algorithms for sub-problems of our problem exist in a variety of scientific disciplines which could eventually be adopted (many work with raster data, though). Depending on the prevailing conditions and the parameter settings, most of these methods would identify some sort of pattern but possibly only *pseudo* patterns, not existing in (cartographic) reality. Respecting non-existing patterns in generalization transformations could lead to inferior results because the number of options for the operator selection would be drastically and unnecessarily reduced in many cases. We believe that in general good and plausible generalization results can be achieved with the measures we have introduced in the preceding sections and by careful selection of the transformation algorithms with appropriate parameters.

We can, however, think of situations where patterns exist in a dataset and where treating them as such could be advantageous with respect to the final result. One example is an area with rectangular agricultural fields of different categories. While the *sequence* of categories might be random, *shape* and *area* are not. Defining a pattern in such a case is an extension of the *cluster* concept combined with the *typical shape* measure we have introduced in this chapter. A second and more complex potential application is the characteristic sequence of landuse/landcover categories with change of altitude. Whether a pattern can be identified in this case depends on the scale of the source map, on the defined categories as well as on many other variables. We assume that an expert could identify *macro scale* patterns we cannot treat with the *topological* and other measures we have introduced also in geological maps. Because of the fact that existing automatic pattern recognition methods are too unreliable for our purposes, we recommend that the user defines the objects forming a pattern interactively when he/she thinks that it exists and that preserving it in the target map would be useful. The concerned objects can then be transformed with respect to each other.

Alignment of polygons

$alignm_m$	$alignm_m$ designates a user defined group of aligned polygons based on geometric, semantic and logical considerations.

We can think of a number of situations in various types of categorical data where *alignments* occur. Lakes, for example, often form alignments in rather broad valleys. These structures satisfy several of the variables defined in the introduction since comprehensible logical connections exist between the individual polygons of an alignment and the objects' shapes are at least similar. A second example, relating to geology, is the erosion of ridges which can partly uncover the underlying stratum, resulting in a series of aligned polygons in a geological map. As for patterns, the correct identification of an alignment and its member objects depends to a large degree on semantic information and expert knowledge. This reasoning process can only partly be substituted by auxiliary data (e.g. DTMs) and rules and thus not be satisfactory automated. We therefore recommend that the (experienced) user defines the polygons belonging to an alignment. Because alignments normally consist only of relatively few objects, this task can be executed rather quickly by selecting them interactively with the mouse. The individual members of the alignment can then be treated as a group.

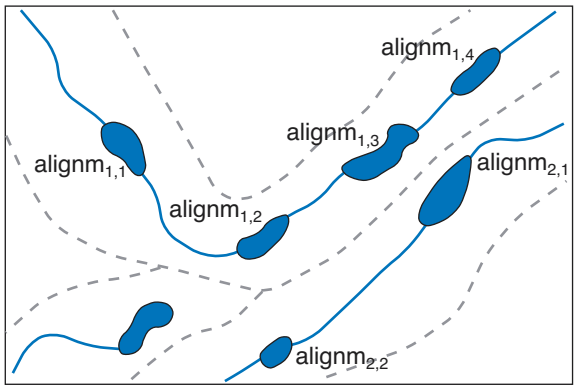


Figure 10: Alignments of objects are identified interactively by the user/system operator based on expert knowledge, represented in the figure by mountain ridges and rivers.

Displacement would only be possible in the same direction for all objects and by approximately the same amount. *Deletion* of particular polygons would only be allowed in exceptional cases while the *aggregation* of a member object and a non-member polygon would be completely prohibited. Although we do not formally require it, an alignment will be formed by objects of the same category in the majority of cases. Figure 10 shows two alignments of lakes in mountainous area. The additional information like mountain ridges and rivers represents the geographic knowledge of the user. Detailed auxiliary data and computationally expensive procedures would be necessary to achieve the same result with automatic methods. If adequate domain knowledge is not available, we recommend to rather not define alignments and to generalize the objects individually.

4.7 Semantic Measures and Information

Semantic information in relative and absolute terms has been integrated in many of the measures we have introduced throughout this chapter. We have defined measures like *characteristic shape* and *island polygon* which designate important objects according to their shape and their position relative to other objects. Therefore, all that is left to do in this section is to introduce a measure which designates an important polygon based on semantic information. Such objects can then be treated in an appropriate way, for instance, only *exaggeration* and *displacement* operators are allowed but not *deletion*.

Measures

s_j	s_j designates an important polygon j based on semantic information.

5. Measures in the Generalization Workflow

A generalization workflow with emphasis on measures is illustrated in figure 11. The first set of measures is computed as part of the initial analysis of the dataset to be generalized. At this point, we want to gain structural insight about the objects and categories in the dataset as well as the relations between them both in absolute and relative terms. Conflict identification is not yet an issue. The statistical measures we have introduced in chapter 4, visual inspection of the data and any form of expert knowledge or other information we have about the data type, the geographical location as well as the applicable map controls (e.g. target scale, map purpose and output media) are integrated to a comprehensive view of the data and a strategy for the generalization task ahead. If reclassification of the data is required, it should take place at this point, before geometric transformations start. If we reclassify the dataset, the statistical measures of the initial data analysis have to be recomputed, because they represent the *before* state of the data which is needed later to assess and judge changes.

Based on the mathematical description of the data and other formal or informal information we have available, we are now able to define our strategy (see figure 11 center). We define both the values of the measures for conflict detection (graphic parameters), such as minimum area and distance, and those representing semantic knowledge like the tables with weight factors and neighborhood information. Patterns and alignments are also defined at this stage, if they occur in the dataset. For such measures like *characteristic shape*, the system operator can specify examples interactively on the screen. Automatic procedures will then search the dataset for objects with similar shapes of the respective categories (e.g. for ring structures of some sediment layer in geology maps). Important objects based on semantic information as defined in section 4.7 can be flagged automatically or interactively depending on whether all required information is formally available or not.

A very important issue for the generalization of polygon mosaics is the processing order. We recommend working on a *per category* basis. The advantage of this strategy is that we can identify and resolve conflicts (more or less) independently from each other because objects of the same category have no common boundaries per definition [2]. In addition to that we have defined many measures on the category level and can better evaluate their changes both formally and visually if we proceed per category. This would not be the case if we chose processing the data, for instance, from upper left to lower right. The transformations we apply on the objects of one category have an influence on the polygons of other categories. Selecting an appropriate sequence is there-

[2] According to our definition, a polygon mosaic consists of a continuous network of objects. The data model uses shared primitives. No polygon has common boundaries with other polygons of the same category and the objects' outlines are graphically represented by a thin black line.

fore an important strategic decision. In general it is advisable to start with categories consisting mostly of small polygons and a small *relative area* and to treat the so-called *dominating* or *background* categories later. However, this depends on the type of categorical data to be generalized and on its specific properties. Although generalization operators and algorithms are not explicitly part of this work, we recommend an initial *cleaning* of the dataset by deleting small objects that cannot be aggregated with other polygons of the same category at a later stage (*noise removal* operation)). Depending on the structure of the dataset, we can considerably accelerate the following frequent recomputation of measures since fewer objects are present. The global structural indicators should not be affected dramatically by this procedure. The sequence of operators in a generalization workflow is an important strategic issue as well but since decisions are made based on the specific values of the measures for a given dataset and not on the measures themselves we cannot make further recommendations at this point.

In contrast to a *per category* processing order, we could also envision a *per operator* sequence, in which generalization operators are applied in sequence to all polygons of all categories. In our empirical studies, we will experiment with different sequencing options to assess the influence of the possible processing order.

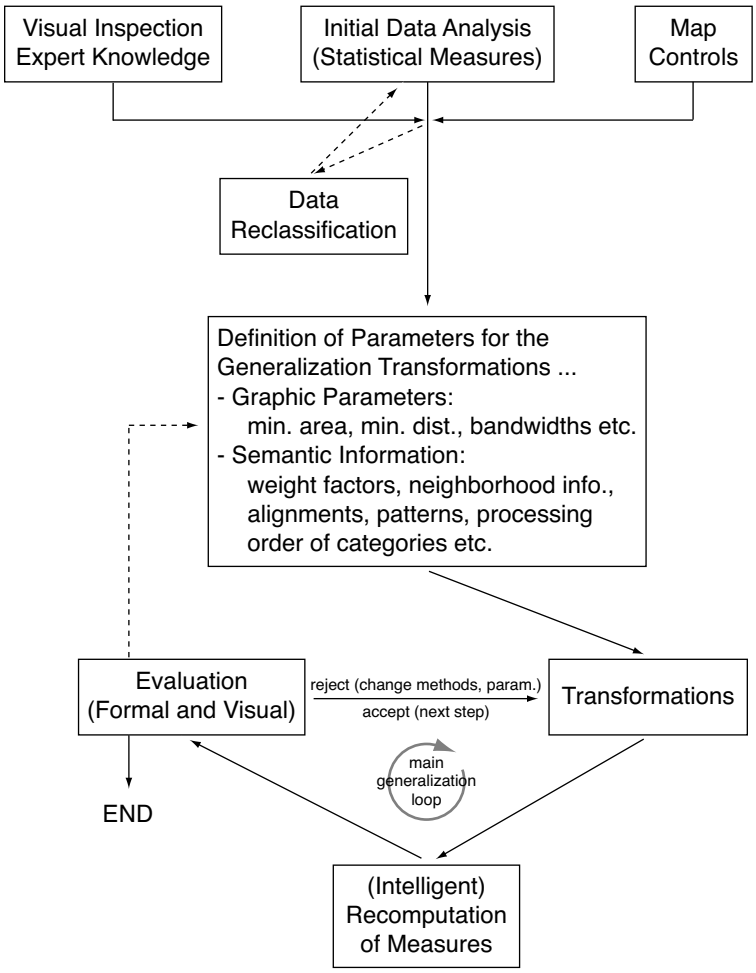


Figure 11: Measures in the generalization workflow.

After every sequence of geometric transformations, all relevant measures have to be recomputed (the *after* state of the data). The following evaluation shows whether conflicting situations have been resolved without introducing new ones and if the changes on higher spatial levels (e.g. for the respective category) lie within the defined tolerance settings. Because of the nature of polygons mosaics, values of measures can change even for objects or categories that have already been treated. This step involves quantitative as well as visual methods since it is, as we have mentioned, not possible to fully formalize concepts relating to the overall visual appearance of a map (Gestalt constraints). Furthermore, it will most likely not be possible to eliminate all conflicts completely in complex situations. Although we provide the means to assess the *severity* of a conflict automatically (e.g. priority tables for topological conflicts) and act accordingly, some problems might have to be resolved

manually on a case-by-case basis. The final result of the evaluation stage is either to accept the computed solutions and proceed with the next step or to reject them. In this situation the transformations are revoked at least partly and the step is repeated with different settings and/or different methods. If a transformation sequence fails completely, it is also possible to backtrack further and revise the strategy in an appropriate way.

6. Conclusions and Outlook

Looking at the number of measures in chapter 4 and their distribution among the different sections, we notice that it is obviously much easier to find suitable measures for the formal description of *graphic* constraints. The majority of them can be considered *generic* since they are used to identify local conflicts that occur in all types of categorical data or relate to general structural properties of a dataset. The area and distance measures are, together with some of the shape and topology indicators, part of our *core set* of measurements and we compute (and recompute) them for every map we have to generalize. The global structural measures in these sections of chapter 4, however, can be problematic. It is not yet completely clear whether and how we will be able to interpret these values correctly and establish a logical link between changes of values and a positive or negative development of the situation on the map. We expect that extensive empirical testing with datasets of different spatial variability and for different target scales will be necessary until we can use these measures reliably to their full potential. Shape is also a very difficult to formalize concept. In general it is difficult to associate a characteristic shape with a specific category in most types of categorical data. We can think of very few cases where this is possible and have therefore provided an appropriate measure. The class of shape index measures is only of limited use because their values are normally not unambiguous; many configurations can have the same value. We will use them with care and only for comparing values before and after generalization transformations. Again, empirical testing will be required to decide which changes are acceptable and which are not. Our strategy so far is to use shape measures mainly at the level of an individual polygon for conflict identification. If needed, we will also experiment with more robust shape measures such as Fourier descriptors.

The problems of the measures we have introduced for *density and distribution* and for *pattern and alignment* are of a different nature. The methods to precisely assess the density and the distribution of objects on a map are very complex and time consuming since they require extensive distance calculations between polygons. At this point, we doubt if such detailed quantitative information is really necessary. The measures we have introduced for this purpose can be seen as a compromise. The *category convex hull* measure for example allows only *estimating* changes in the distribution of objects in a qualitative way but, on the other hand, it can be computed relatively fast and we think it should be precise enough for the purpose we need it for, especially if we combine the method with the regular grid as illustrated in figure 8. A major drawback of this method is that it can only be computed after completion of the transformations. If we reject the results, the entire transformation step has to be revoked and repeated with different parameter settings or by using alternative methods. The measures we have introduced for patterns and alignments may at first be disappointing from the scientific perspective because we propose to identify such structures *manually*. Considering the complex structure of polygon mosaics with a high spatial variability, we find it rather unlikely that automatic methods will deliver correct and robust results. Therefore we have decided to concentrate on more promising issues for the time being. However, we will probably try to implement automatic methods at a later stage for specific data types such as forest maps with only two categories.

The work executed so far was focusing on *completeness* and the majority of the measures we have introduced are rather *generic*. So far, we did not yet respect any platform dependent issues nor did we consider any specific requirement of particular generalization algorithms. This will change in the next step when we start with the implementation of measures as part of a system for the (semi-) automatic generalization of categorical data. We have decided to use Laser-Scan's LAMPS2 platform. Because our department was a member of the consortium which developed an *agent-based* system for the automatic generalization of topographic maps for LAMPS2 (AGENT 2001), we hope to benefit from the existing knowledge and experience. Besides some basic measures that every GI System provides (e.g. area), we hope that other useful functionality resulting from the AGENT project like the computation of the convex hull for a set of points can easily be adopted for our purposes. One of the next steps will also be to study how topology measures can be best implemented in the LAMPS2 environment. Another very important issue is the harmonization of measures and generalization algorithms. The generalization algorithm part of the generalization system is taken care of by Martin Galanda (Galanda 2001). Since this project is also in an early phase, we cannot give further details at the moment. It is clear, however, that more flexibility is required from the measures side than from the generalization algorithm side. It can therefore be expected that the set of measures we have defined in this paper will have to be revised, extended, and modified in the course of the project.

7. References

AGENT (2001): Integrating Multi-Agent, Object Oriented, and Algorithmic Techniques for Improved Automated Map Generalization. *Proceedings of the 20th International Cartographic Association Conference*, Beijing, China, 6-10 August 2001.

Bader, Mats (1997): Methoden zur Erkennung und Lösung von Metrischen Konflikten in der Generalisierung von Polygonmosaiken. *M.Sc. Thesis*. Geographisches Institut der Universität Zürich. In press.

Bader, Mats and Weibel, R. (1997): Detecting and Resolving Size and Proximity Conflicts in the Generalization of Polygonal Maps. *Proceedings of the 18th International Cartographic Association Conference*, Stockholm (S), pp. 1525-1532.

Bader, Mats (2001): Energy Minimization Methods for Feature Displacement in Map Generalization. *Ph.D. Thesis*. Geographisches Institut der Universität Zürich.

de Berg, M., van Krefeld, M. and Schirra, St. (1995): A New Approach to Subdivision Simplification. *Proceedings of AUTO-CARTO 12*, **4**, pp. 79-88.

FRAGSTATS (1994): Spatial Pattern Analysis Program for Quantifying Landscape Structure. *Documentation of Version 2.0*. Oregon State University.

Galanda, Martin (2001): Optimization Techniques for Polygon Generalization. *Fourth ICA Workshop on Progress in Automatic Map Generalization*, Beijing, China, 2-4 August 2001.

Mustière, Sébastien and Duchêne, C. (2001): Comparison of Different Approaches to Combine Road Generalization Algorithms: GALBE, AGENT and CartoLearn. *Fourth ICA Workshop on Progress in Automatic Map Generalization*, Beijing, China, 2-4 August 2001.

Peter, Beat (1997): Ableitung von Generalisierten Bodennutzungskarten aus der Arealstatistik der Schweiz 1979/85. *M.Sc. Thesis*. Geographisches Institut der Universität Zürich.

Peter, Beat and Weibel, R. (1999): Integrating Vector and Raster-Based Techniques for the Generalization of Categorical Data. *Proceedings of the 19th International Cartographic Conferenc*e, Ottawa, Canada, 16-20 August 1999.

Peter, Beat and Weibel, R. (1999): Using Vector and Raster-Based Techniques in Categorical Map Generalization. *Third ICA Workshop on Progress in Automatic Map Generalization*, Ottawa, Canada, 12-14 August 1999.