TECHNISCHE UNIVERSITÄT DRESDEN

**Faculty of Environmental Sciences** Institute for Cartography

# Master Thesis

## Automated detection and explicit modelling of structural relations for the generalisation of island groups

submitted by Benedikt Víðisson

born on 22.01.1985 in Akureyri, Iceland

submitted for the academic degree of
Master of Science (M.Sc.)

Date of Submission 12.11.2015

Supervisor Prof. Dr. Dirk Burghardt

Institute for Cartography

TECHNISCHE UNIVERSITÄT DRESDEN

**Faculty of Environmental Sciences** Institute for Cartography

# Task of Master Thesis

**Course of Studies:** Cartography

**Name of the Graduand:** Benedikt Víðisson

**Topic:** Automated detection and explicit modelling of structural relations for the generalisation of island groups

**Goals of this Study:**

The structural generalisation of isolated polygon groups was described as the generalisation problem with the highest complexity by Bertin (1974, p 312ff) in his book "Graphic semiology". A first big challenge is the automated identification of polygon groups. Within generalisation literature this process is described as data enrichment with the identification and modelling of structural relations.

The first task of the thesis will be the manual identification of representative island groups at different scales for a qualitative comparison. Following that a conceptual analysis has to be carried out to identify different island group categories.

The main part of the thesis will be dedicated to the automated detection of island groups. Auxiliary and graph based data structure should be utilised to model neighbourhood relations. Further measures has to be implemented to quantify distance, similarity in size, shape as well as the linear arrangements. Based on that a method should be developed for the identification of linear island arrangements. A weighting of influencing factors should be considered as well as a measure to describe the homogeneity of the groups.

The thesis should be submitted in two printed versions together with the digital version on CD. The digital version should include the text description and all required data and software to run the prototype. It is encouraged to publish the thesis on the publication server Qucosa of SLUB. The major findings will also be presented in the form of an A2 colour poster.

**Supervisors:** Prof. Dr.-Ing. habil Dirk Burghardt (TU Dresden)

**Beginning of thesis:** 1.5.2015
**Date of submission:** 30.9.2015

## Statement of Authorship

Herewith I declare that I am the sole author of the thesis named

**“Automated detection and explicit modelling of structural relations for the generalisation of island groups”**

which has been submitted to the study commission of geosciences today.

I have fully referenced the ideas and work of others, whether published or unpublished.

Literal or analogous citations are clearly marked as such.

Dresden, 12.11.2015 Benedikt Víðisson

## Abstract

The development of automated generalisation of island groups requires automation of structure modelling and pattern detection. For this purpose, geometric properties of island features have to be described in a machine-translatable way, effectively teaching computers to recognise island groups. In this study, structural relations between islands were formalised by choosing criteria denoting important similarity of shapes. Connections between islands were used as agents to test whether island pairs should be grouped, and those pairs then consolidated into clusters according to the grouping strategies. The criteria were used for creating grouping parameters to decide whether each pair of adjacent islands would be united according to that parameter or not. The criteria were based on relations of proximity, size, shape, orientation and straight-line continuation. The grouping strategy then consisted of a divisive hierarchical clustering, using two separate methods to perform the cluster division. Each grouping parameter was indexed to each connection with information whether parameter criteria were fulfilled or not. The first divisive strategy was to rank the parameters by importance and use each parameter in sequence to eliminate connections between islands when these connections did not fulfil the parameter criteria because of lack of similarity between the islands, thereby dividing them. The second strategy consisted of assigning a weight to each parameter, so if important parameter criteria were fulfilled, such as proximity, then connections would obtain a higher weight index than if criteria of less important parameters were fulfilled, or none. These weight indices were then applied in ascending order, containing a more and more exclusive set of connections, to divide the clusters. The results showed promise in using such parameters in combination to divide island groups into sub-groups, with proximity and linear continuation showing the best island groups. The weighted parameter application worked better than the sequential application. Such an approach can be of use in comprehensive generalisation frameworks as a component in a bottom-up approach to automated generalisation.

## Table of contents

## List of figures

## List of tables

# 1 Introduction

There have been many attempts to give cartography a proper definition as a subject, usually involving the intersection of science and art. As an attempt to be brief, cartography and map making can be described as the science and art of abstracting and modelling space for a given purpose, in order to make a map that conveys information in a clear and accessible way. Contrary to some preconceptions of many non-cartographers, map-making is far from a straightforward task, and is highly subjective and dependent on decisions made by the one or many who commission and draw the actual map. Almost every step of the cartographic process is subject to this reality.

## 1.1 Generalisation

Generalisation is a fundamental process in cartographic work. Representation of spatial data on maps of different scale levels accentuates the subjectivity of the cartographic process, as it includes decision making about said representation which always must have the goal to facilitate the usage of the map and make the information readable and accessible, often to non-experts in either cartography or the specific field for which the map production is employed. Without this readability, the map is mostly useless. The human eye detects patterns and structure in most visible things, and maps are no exception. Map reading is highly susceptible to the same principles of pattern recognition that governs other graphical creations. Sometimes, patterns in maps describe real-world processes or evidence thereof, such as the spatial patterns caused by geological and geomorphological processes, or anthropogenic patterns such as those that follow human development. Most often though, pattern recognition in maps is important for orientation and patterns existing over different scale levels serve to enable the map-reader to recognise the area shown on multiple maps as one and the same, despite the difference in representation on each map.

For these reasons, preservation of important graphical structures over different scales is important, to maintain the recognisability of areas and features at different scales. When a map is drawn by hand, these pattern preservations are done by decisions of the drawer, be they conscious or unconscious. Most pattern recognition done by the human brain is unconscious, but even when conscious, it is not always easily explained or formalised. Some attempts have been made to formally describe important factors in

structures that enable recognition of patterns, and cartographers have built on these to some extent in research about generalisation.

In modern cartography, automation plays an increasingly important role in easing tasks that can otherwise be very time-consuming if done by hand. A great deal of cartography is indeed done with the help of computers, which solves many problems that have to do with data management and storage, quick access and manipulation and output creation. However, much of the work – especially drawing – Is still done by hand for serious cartographic products. In the digital world, an ever increasing number of these tasks can be and are being automated, but there is an important limit that computer technology still has and probably will not overcome in any near future, i.e. the solution to problems of representation of complex multi-levelled data and the generalisation thereof when reducing the map scale. The high subjectivity of the work makes it impossible as of yet to delegate it completely to the computer, but steps are constantly being made to improve the computer brain to solve problems hitherto only approachable with a human one.

Automation of generalisation is a huge task and much studied. The main problem is that generalisation is not simply a simplification of lines and polygons, or omission of points (for which many algorithms already exist), but a holistic approach considering the multi-levelled structure in the map. This means that not only is it important to recognise geometric structures, but also semantic relations between features. Before the generalisation step is reached, the pattern detection must be completed one way or another. This is a much researched and discussed field, but is far from completed given the multiple complexities involved in teaching pattern recognition to a computer that compares to that of a map-reader. In this study, island groups are used as test data to develop a pattern-recognition process that could then be used to aid in later generalisation. This pattern recognition, however, is purely geometrical and does not consider semantic relations or importance of features and therefore only represents one component in the generalisation process.

## 1.2 Structures of island groups

Island groups, like most natural features, exhibit spatial structures rather unlike man-made features such as built-up areas. Each island feature is usually more irregular and complex in shape than building features, and the distribution of the islands is likewise

less regular than that of buildings, which are often aligned along strongly linear features such as roads. Yet, structures in island groups can often be distinguished.

There are several different types of island groups. The most straightforward grouping relation is arguably relation by proximity. Most or all islands in groups must be relatively close to each other in order to constitute a group or archipelago, but islands may be united by one or more structural relation parameters on top of that. Proximity relation without any other structure could be described as a proximity cluster formation. Linear relation is another structure that stands out to a map-reader. This is when islands are arranged in such a way that a line pattern can be discerned. Linear relation can further be divided into island groups depending on whether the islands are arranged in a straight line or curved line, and whether shape of the islands follows the line or not. Figures 1.1A and 1.1B show the Frisian and Aleutian islands respectively, where many of the islands have a shape that resembles the linear direction of the island distribution. Other categories exist, e.g. where islands are shaped in a dominant linear pattern, but the arrangement is non-linear (figures 1.1C and 1.1D). These particular examples in figure 1.1 show evidence of the geomorphological processes that shaped the islands. In addition, size is an important factor in this type of pattern recognition, as objects of similar size are more likely to be seen as belonging together than objects of vastly different sizes.

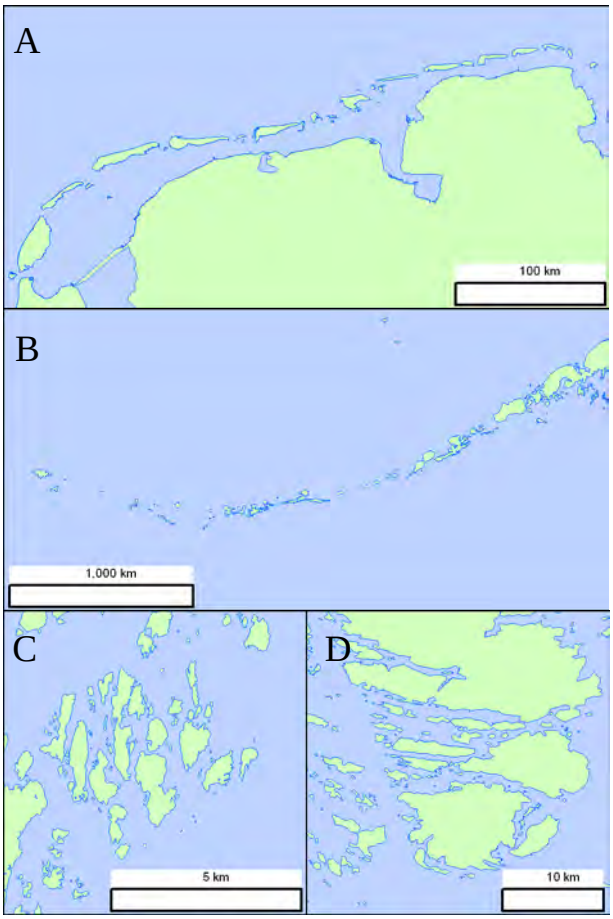


**Figure 1.1: Some examples of island groups patterns**
**A) Frisian Islands of Netherlands and Germany**
**B) Aleutian Islands of Alaska**
**C) Group of islands in the Stockholm Archipelago**
**D) Islands outside Turku, Finland**

## 1.3 Aims of thesis

The implications of the various opinions about which approach should be taken to the problem of automated generalisation are discussed further below in the state of the art and discussion chapters, but in brief, while is debatable whether geometrical pattern recognition for a very specific part of an automated generalisation process is worthwhile in the quest to fully automate the subjective and holistic process of generalisation, it can be argued that in developing the framework for automated generalisation, it is important to solve the problems of the components of the framework, and it is possible to do this separately for the purpose of later integrating the components into a comprehensive process. This approach has to a certain degree been attempted in previous research.

The approach in this paper can be described as a component in a bottom-up approach to automated generalisation, solving one problem in structure recognition as a preliminary step by modelling structural relations. The motivation for this research is the automation of island group generalisation. In order for such automation to take place, preliminary steps must be taken, such as the creation of automation algorithms for the detection of these groups and spatial patterns within the groups. To this end, the islands are considered as semantically identical two-dimensional features, and an attempt is made to group these based on their geometrical properties. The lakes around Dombes, France have featured frequently in previous research into generalisation, and therefore the algorithm will also be applied to that area for evaluation, but the main focus is on island groups. Lakes, as some other natural map features, can also be reduced to semantically identical two-dimensional features for this purpose.

The aim in the paper is to use island pairs in a larger island group as test data, reverse engineer a process of structure recognition for island grouping using unifying relations between island pairs that share some attributes, express these attributes as grouping parameters, develop grouping strategies using these grouping parameters and apply the grouping process to several test areas and then qualitatively evaluate the resulting groups and the success of the process.

## 1.4 Outline of thesis

The thesis is structured in the following way:

- After the introduction of the topic, the description of the aims of the thesis and this structure outline, previous research is referred to and discussed in the chapter called **State of the art**. The chapter is structured to discuss problems of generalisation, moving onto automated generalisation, structural pattern recognition by human perception and the automation of this detection.
- The next chapter is **methodology**. It lists the steps taken to automate the structure detection, starting by listing the software and datasets used for the purpose of this study, and giving a short background of the test areas putting the structural patterns in real-life context. Afterwards, there is a description of the general grouping strategy, which governs all subsequent method choices, before listing step-by-step how the data is structured to prepare for pattern detection, and how parameters are calculated and applied on this structure, resulting in the grouping and sub-grouping within the datasets.
- After **methodology** come the **results**. The output of the process is primarily figures, be they maps or graphs. In between the figure series, there are descriptions of the image data to give them meaning and guidance as to how they should be interpreted.
- In **discussion**, the results are put into a wider context as to what the implications are for the success or otherwise of the algorithm, and what the use of such a process could mean for future research into automation of generalisation.
- Lastly, there are some brief **conclusions**, listing the main findings of the thesis.

# 2 State of the art

This chapter will outline and discuss some research that has been done in the field of generalisation, the question about the feasibility of automating it, and how group detection and pattern recognition fits into the process.

## 2.1 Background of generalisation research

The cartographer Max Eckert wrote in 1908:

> In generalizing lies the difficulty of scientific map-making, for it no longer allows the cartographer to rely merely on objective facts but requires him to interpret them subjectively. To be sure the selection of the subject matter is controlled by considerations regarding its suitability and value, but the manner in which this material is to be rendered graphically depends on personal and subjective feeling. But the latter must not predominate: the dictates of science will prevent any erratic flight of the imagination and impart to the map a fundamentally objective character in spite of all subjective impulses. It is in this respect that maps are distinguished from fine products of art. Generalized maps and, in fact, all abstract maps should, therefore, be products of art clarified by science. (Eckert, 1908. p. 347)

This thought applies to the problem of automated generalisation inasmuch as it describes the fact that a part of the work is scientific and objective, and therefore should be machine-translatable, yet there is the subjectivity and “artistic” part, which is far more difficult to formalise in computer algorithms. McMaster and Shea (1992) were interested in the fundamental issue in automatic generalisation, whether it can be done and what approaches should be taken to the solution. They mentioned Arthur Robinson’s significant work in documenting previous research in generalisation and his speculations about the impossibility of setting a consistent set of rules for unbiased map generalisation and that generalisation would always be a creative process. Eduard Imhof wrote on a related note:

> The content and graphical structure of a complex, demanding map image can never be rendered in a completely automatic way. Machines, equipment,

> electronic brains posses [sic] neither geographical judgment nor graphic-aesthetic sensitivity. Thus the content and graphic creation remain essentially reserved for the critical work of the compiler and drawer of a map." (Imhof, 1982, pp. 357-358)

This fundamental problem is a recurring theme in generalisation research even many years later. Brassel and Weibel (1988) weighed in on the subject explaining how generalisation is a broad concept that is used in all aspects of life, not only cartography, where the extraction of the important and the general over the specific and superfluous is essential. This may shine some light on why it is hard to define and to formalise. They wrote:

> Model construction (i.e. generalization) is heavily influenced by the fact that the major goal is not mere analysis of space but communication of spatial concepts. Map generalization theory, therefore, has to consider not only spatial modelling but also visual communication theory." (Brassel and Weibel, 1988, p. 230)

According to Brassel and Weibel, previous practices tend to neglect the traditions and theoretical background in generalisation in the automation effort that sets the procedures of generalisation up as pragmatic programming procedures (Brassel and Weibel, 1988). In many respects, these are valid points but in order to hope to automate generalisation, some segmentation into solvable problems must occur. Brassel and Weibel (1988) emphasised that the different distinguishable domains, functions and controls according to useful definitions of generalisation (for its automation) are not entirely separable:

> Generalization, as an intellectual process, structures experienced reality into a number of individual entities, then selects important entities and represents them in a new form. If we want to simulate this process automatically we have first to understand the process of recognizing essential features to model this process and to implement component extraction and representation. (Brassel and Weibel, 1988, pp. 230-231)

McMaster and Shea (1992) claimed that it is a prevailing issue in efforts towards automation of generalisation that those who have attempted it…:

> …have focused on a single generalization problem in *isolation* from other aspects of generalization, and, more often than not, have dealt with *abstract* graphic entities, rather than digital graphic objects that were representations based upon an underlying geographical frame of reference. Though many authors have repeatedly emphasized that manual generalization not be conducted in isolation or in the abstract, many of the early attempts at automation often disregarded that guidance. (McMaster and Shea, 1992, p. 20)

McMaster and Shea (1992) said that the models so far put forth that try to define the problem of generalisation do not do so from a technical and a philosophical perspective, but they tried to develop a model that does just that. They divided the generalisation process into three operational areas: the philosophical objectives (why is there a need to generalise), the cartometric evaluation (when to generalise) and the selection of the appropriate spatial and attribute transformations to use (how to generalise). The first point has to do with adherence to general principles of cartography, the question of what needs to be generalised and how, and "a consideration of existing computing technology demands and capabilities" (McMaster and Shea, 1992, p. 28). The part in their model that is most relevant to this paper is the spatial and holistic measures of the cartometric evaluation, i.e. the part that examines the geometric properties of objects on the map.

There are seven listed measures:

1. Density measures are used to evaluate number of features per unit area, or "average density or number and location of cluster nuclei".
2. Distribution measures are used to examine position relative to each other with regard to clustering, randomness, complexity, distance from common feature, etc.
3. Length and sinuosity measures have to do with examination of length and angle relationships of parts within line or area features.

4. Shape measures are various methods in use to assess the shape of area features and whether they can be represented on a smaller scale.
5. Distance measures – the measure of distance between two points or buffers around points.
6. Gestalt measures – the assessment of the perceived structures of shape and distribution of features.
7. Abstract measures – evaluation of conceptual nature of spatial distribution where "possible abstract measures include complexity, homogeneity, symmetry, repetition, recurrence."

(McMaster and Shea, 1992, pp. 46-48)

The authors explained that while most of these can be translated digitally, measures such as Gestalt or abstract are more difficult. Importantly:

> In the end, it appears as though many prototype algorithms first need to be developed and then tested and fit into the overall framework of a comprehensive generalisation processing system. Ultimately, the exact guidelines on how to apply the measures designed above can not be determined without precise knowledge of the algorithms. (McMaster and Shea, 1992. p. 48)

## 2.2 Perceptions of island groups

As stated in the introduction, the question of automated detection of structural patterns in maps is not new, but nor is it solved to a large extent. Previous researchers have suggested processes and approaches, and even possibly different philosophies as to how the task should be completed, but nothing close to an industrial standard exists in cartography or GIS.

Steiniger, Burghardt and Weibel (2007) approached the topic of automated detection of island group by surveying map-readers and then building an algorithm designed to detect the pattern as identified by the users. They wrote that a key step in the beginning of the process of automated generalisation of island groups is the automatic detection thereof. This is in order to identify important structures that will then potentially be conserved during the actual generalisation. Important features that should be kept include features that form an outline of a structure and the core (as also

stated by Jacques Bertin in his book Semiology of Graphics), and isolated islands because they may be important for orientation. Steiniger, Burghardt and Weibel (2007) further mentioned the work of Max Wertheimer in gestalt theory and argued that for the purpose of map reading, the most important laws of organisation in perceptual forms are the law of similarity and law of proximity, i.e. that objects are more prone to be grouped together by a map reader if they are alike and if they are close together, compared to other object relations in the map. However, they also mentioned the difficulty in formalising attributes that constitute these relations, i.e. the question of how similarity relations should be ranked or what the relation would be between similarity and proximity.

The results of the survey made by Steiniger, Burghardt and Weibel (2007) were that spatial proximity is far more important than other variables in detecting large island groups because the member islands of the groups exhibited significant heterogeneity and are therefore were not primarily grouped using other variables. The smaller island groups were somewhat varied, with shape, size and orientation all in agreement or not, but again, some small island groups appeared to be formed primarily based on proximity rather than other variables. They furthermore mentioned that as per the law of Prägnanz, visual perception is more sensitive to horizontal and vertical patterns than patterns in other directions. The main interest of the authors in that particular research was the detection of meso-structures (island groups of around 12 islands or more), which they concluded is primarily dependent on proximity relations, while other grouping variables are much less important. Their work included a creation of an algorithm to detect the island groups identified by the surveyed people – a reverse engineering of the island group detection process. The resulting algorithm did not entirely agree with the grouping done by the human sample, but then, not all the people identified the exact same island groups. At the end, Steiniger, Burghardt and Weibel (2007) stated that recognition of micro-structures (island groups of 10 or fewer islands) remains an important task for the future of automatic generalisation.

This underlines the subjectivity of the problem and somewhat undermines the cause of building a computer algorithm aimed at identifying patterns as if it were a human map-reader. During generalisation, it is essential to maintain some structures in the map that the user recognises as important patterns by which to orientate. In order to

automate the generalisation process, the pattern or structure-recognition must therefore also be automated. Furthermore, it is important to consider the human element for evaluation of the resulting grouping done by any possible detection algorithm, and consider that a creation of a fixed algorithm to detect island groups in accordance with the experience of every map-reader is probably an unsolvable task.

The aims in this paper are partly in accordance with the point made by Steiniger, Burghardt and Weibel (2007) about future work in recognising micro-structures in island groups. The primary grouping variable for meso-structures may indeed be proximity, but then in order to then detect smaller and smaller groups within the larger groups, other variables need to be considered. The general method in this paper is also the same as in their paper, i.e. detection of relations in island groups that are important for the grouping together of islands, and a reverse engineering to enable the computer to make the same grouping. Whereas it is an important hindrance to the solution of this task – the perceived impossibility of a perfect solution – it is arguable that good approximations can be obtained in the future.

Steiniger and Hay (2008) went further into the topic of determining what characterises perceived groups in island maps with the ultimate purpose of translating the parameters into computer-understandable language for automation of detection and generalisation. Again the method was to survey a group of people using maps of island and lake features, and evaluating which grouping variables determined the resulting polygon groups. This method serves the first of three components in the development of pattern-recognition map generalisation algorithms, i.e. “Identification of the visual patterns of interest.” The other two are “formalization of patterns” and “development of pattern recognition algorithms based on the discovered principles” (Steiniger and Hay, 2008, p. 2). The results in that particular study are only presented as preliminary, but they do at least show that most of the groups (86% of groups within a test area in the Åland Islands and 90% in a test area in Maine) are so-called *proximity* groups, i.e. groups where the internal distance between group members is smaller than between group members and outside polygons. That means though that 26% and 10% respectively are *non-proximity* groups, meaning there are other variables at play, albeit less importantly.

## 2.3 Automated generalisation

Brassel and Weibel – while listing some of the difficulties in attempting to automate generalisation – did concede that automation is important and drew up a framework of the process (figure 2.1).

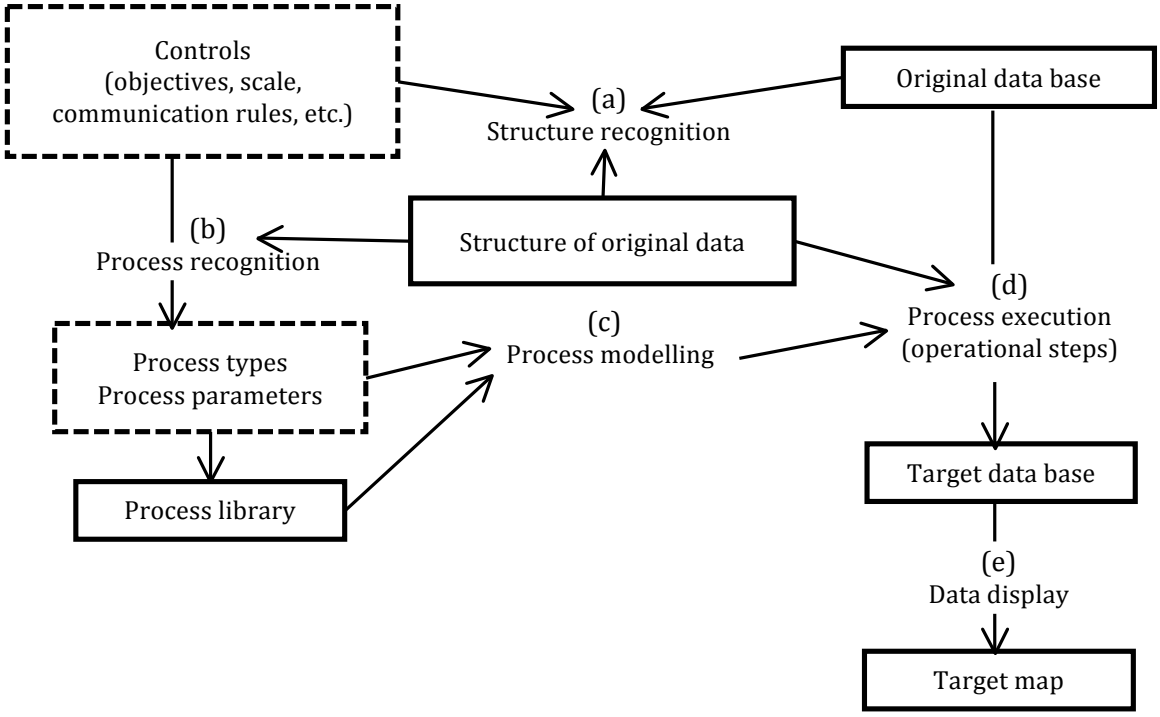


**Figure 2.1: A framework for automatic generalisation**
**From 'A review and conceptual framework of automated map generalization'**
**(Brassel and Weibel, 1988, p. 231)**

The first step in their process is labelled “structure recognition”, or in the words of the authors themselves:

> This process aims at the identification of objects or aggregates, their spatial relations and the establishment of measures of relative importance. Structure recognition is controlled by the objectives of generalization, the quality of the original database, the target map scale and the communication rules (graphic and perceptual limits). It represents a process of intellectual evaluation which is traditionally performed by visual inspection of a map. Some processes may be simulated by specific computer vision procedures, but an adequate automation of the step is non-trivial. (Brassel and Weibel, 1988, p. 231)

There are those who wish to develop and build on the idea of generalisation models that focus on specific applications and others that stress that the whole modelling needs to be redesigned in such a way that enables a more holistic approach to generalisation and synoptic processing. This is because the former simple specific algorithm-oriented approach is deemed insufficient for a multi-layered map. In the case of multi-layered maps where Brassel and Weibel considered the "reduction and simplification process" as insufficient, they discussed the need to…:

> …devise schemes that can model the complexity of the task and that are based on theoretical understanding of cartographic communication and generalization. They involve the development of processes for structure recognition, process recognition and process modelling and are to be calibrated by empirical tests of map perception and generalization effects. Only such a strategy can lead to the development of real knowledge-based systems (and not just rule-based systems) using adaptive, intelligent algorithms and flexible data models. (Brassel and Weibel, 1988, p. 240)

Müller and Wang (1992) wrote about the generalisation of semantically identical two-dimensional area patches. This could mean lakes, islands or any other map features that represent the same type of area. The title of their paper includes the term "competitive approach", which in this case refers to the competition for space between the features on the map, based on their geometric properties. The paper is focused on the generalisation process itself, as opposed to the preliminary pattern processing, and it includes interesting ideas for rules to give the generalisation algorithm. Firstly, the elimination of area patches if they fall under a certain size limit, governed by the ability of the human eye to discern small objects. Secondly, the aggregation of smaller islands to nearby larger islands, the latter being enlarged accordingly in order to maintain the figure-to-ground relationship across the scales. Other rules include the preservation of topology of the features, features being either combined or moved apart when in conflict and distance of displacement of features being inversely related to the feature size. Müller and Wang discussed attempts made up to that point in the automation of generalisation,

stating that the translation of the manual generalisation experience into machine-understandable language has so far been partially successful in larger-scale data where the transformation problems are primarily geometrical in nature, but inadequate in smaller-scale data where semantic transformation are also required. They wrote: “what appears obvious for manual cartographers, when they have to decide on selection, displacement, enhancement and symbolisation of cartographical objects, is not so obvious when those decisions have to be provided in a logical frameworks and turned into a computer program” (Müller and Wang, 1992, p. 137). This is due to contextual information governing such decisions, and according to the authors, the processing of contextual information for spatial data objects was difficult given the technology at the time. They further said about generalisation of area patches such as lakes: “the purpose of generalisation is to eliminate detail in a meaningful way. What are the factors influencing the process of generalisation? For a trained cartographer, factors such as scale, visual perception, geographical area, visualisation platform and format, regional characteristic and user’s needs are all connected.” (Müller and Wang, 1992, p. 138). The authors mentioned Jacques Bertin’s rules of generalisation of the lakes around Dombes, near Lyon, France.

1. “Only a subset of the original patches is preserved after scale reduction, and the remaining patches should be exaggerated in size”
2. “Elements of the contour of the convex hull delimiting the original distribution of patches must be recognisable after generalisation”
3. “The structure of the spatial distribution of patches must be preserved.” (Müller and Wang, 1992, pp. 138-139)

The authors discuss these rules for the sake of the interpretation of good generalisation according to Bertin. According to them, other cartographers may have other solutions in mind or might prefer to modify Bertin’s rules for their own purposes, but the bottom line is that “Rules are necessary in order to automate the process of generalisation.” (Müller and Wang, 1992, p. 139).

The results from Müller and Wang's generalisation process on the Dombes lake area are relatively good (figure 2.2). The process they outlined is a complete process of data handling and generalisation, whereas this paper is focused on one of the preliminary steps, i.e. the pattern recognition. Interestingly, in Müller and Wang's pre-processing, the decision of how to handle each island is dependent on buffer creation, where the buffer size is calculated from the compactness of the island. This is a valid alternative of determining a threshold distance between potentially conflicting polygons, but ultimately it means that the only parameters used in this pre-processing are the proximity variable and to some extent, shape (i.e. compactness or non-compactness). This is also only usable for simple area-patch generalisation, but as the authors themselves noted, the algorithm is not successful in the preservation of archipelago forms, necessitating a process of pattern recognition. Müller and Wang (1992) conclude that while it is problematic to combine individual generalisation solutions into a comprehensive generalisation, their solution can still be useful for generalisation of islands and forest patches, and can be improved and built on by future work in this field.

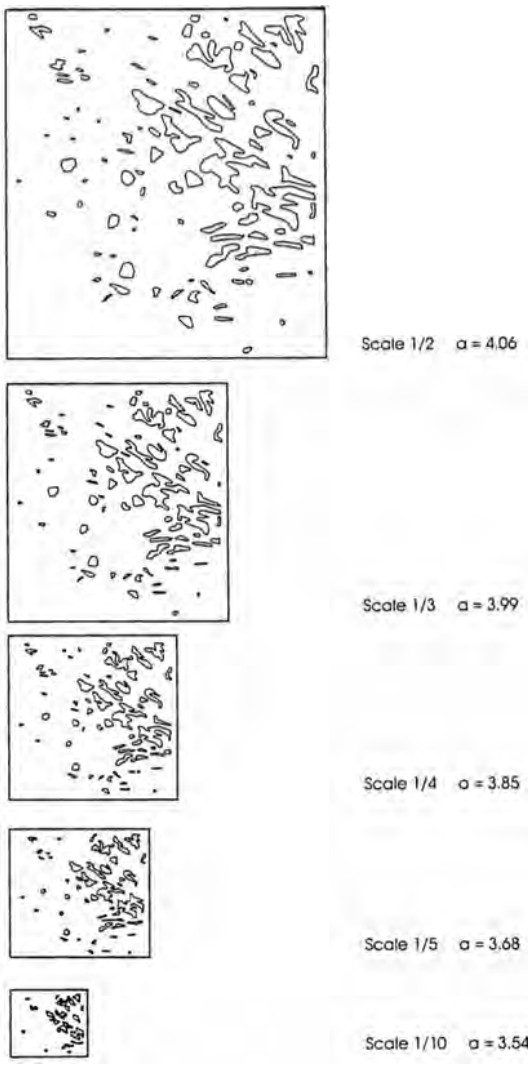


**Figure 2.2: Müller and Wang's generalisation results of the Dombes lake area.**
**From 'Area-patch generalisation: a competitive approach' (Müller and Wang, 1992, p. 143)**

Regnauld (2005) outlined a model that is relatable in this paper, in that features and edges are given attributes, which are then used to govern the clustering. In his example, the connection between features were done by a Delaunay triangulation of the feature centroids. Each node in the graph carries information about its XY coordinates and the edges that connect it to other nodes. Each edge likewise carries information

about which two nodes it connects. The clustering is performed by assigning a weight to each edge, and then eliminating the edges that are below a certain defined threshold weight. The remaining edges are then used to cluster together the nodes. This type of connection-centric weight-based divisive clustering is the same method as the main method in this paper. Regnauld did state that the weights can be assigned using various parameters, and whereas the weight in the example used in his paper is the Euclidean distance, the weight in this paper is calculated from several input parameters, as will be detailed below.

Finally, it is worth mentioning definitions of certain types of generalisation modelling as given by Harrie and Weibel (2007), namely that of the so-called *agent modelling*, *combinatorial optimisation modelling* and *continuous optimisation modelling*. *Agent modelling* is a system whereby objects in the map are considered "aware" of their own attributes and the constraints put on them by the map parameters. This enables these agents to determine whether they fulfil whichever requirements are asked of them according to given map parameters and thus, fulfil their "goals". If they do not, e.g. if a size constraint is not met, the agent performs a number of operations to rectify the situation, either by size change, displacement, self-removal or any other generalisation operations. This requires a system in which agents can be aware of their surroundings and relations to other agents in the map. *Combinatorial optimisation* modelling is an approach where the process generates possible solutions to each conflict on the map and uses the most optimal combination of solutions. *Continuous optimisation* modelling considers all map features as their comprising vertices and uses a mathematical function to reach its optimal generalisation solution on the given space. The reason to give these definitions here is to point out similarities with the approach here and the first modelling variation as explained by Harrie and Weibel (2007), i.e. the *agent modelling*. While the agents in this case to not perform generalisation operations, they do decide whether they are grouped together or not based on their attributes. The agents in this paper are the islands and the connections between the islands, and where these agents are to be thought of as active agents, in this paper it would be the connections that make the decisions of whether to group islands or not, and the islands just serve to carry information about themselves to give to the connections for each determination and then either be grouped with their neighbours or not.

## 2.4 Pattern detection

The detection of island groups within a certain scale level is dependent on formalisation of *horizontal* relations. Horizontal relations are the geometric relations between objects in the same map, whereas *vertical* relations are relations between instances of the same class of feature within maps of different scale, in which case these features have undergone some degree of generalisation (Steiniger and Weibel, 2007). Steiniger and Weibel took an example of a row of buildings, which is considered to be of a relation type *alignment*. Within that type of relation, they listed relations that make up the definition of the alignment type, i.e. distance relations (distance between objects), angle relations (deviation of objects from the alignment axis), size relations (comparison of object size), shape relations (similarity of object shape) and semantic relations (type of object) (Steiniger and Weibel, 2007, pp. 178-179). The size properties of an object can be expressed by area, diameter, perimeter or length of an object, and the distance relation is likewise relatively straightforward, being the Euclidean distance calculation (Steiniger and Weibel, 2007). The shape is a more elusive property to formalise in an object. While some simple indices can be derived, a better shape formalisation would appreciate various different aspects of what constitutes shape (AGENT Consortium, 1999). This means that a proper shape formalisation needs a more elaborate computation than e.g. simply obtaining a convex or rectangular hull.

With regard to island groups, the same relations can be seen as the main grouping variables when testing island similarity. The exception in this case is the semantic relation, as it does not pertain specifically to geometric relations, although the semantic relation could be valuable in island group detection in a multi-levelled generalisation approach.

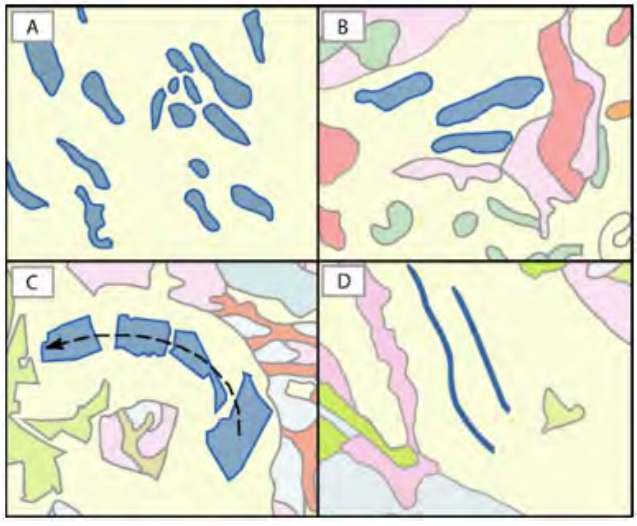


**Figure 2.3: Grouping of objects by Gestalt laws**
**From 'Relations and Structures in Categorical Maps'**
**(Steiniger and Weibel, 2005, p. 9)**

In research into generalisation of buildings and road networks, emphasis is usually

put on several important Gestalt principles that aid in the guidance of what structures in a picture will be perceived. Thomson and Richardson (1999) applied the principle of *good continuation* to road networks and managed successfully to implement it in such a way that lead to a logical result for smaller scales. Steiniger and Weibel (2005) showed examples of the effect of Gestalt laws on horizontal relations that help group objects together (figure 2.3), i.e. the law of proximity (A), principle of similarity, or a similarity in size, shape and orientation (B), direction of alignment, which does not necessarily have to be straight (C), and the law of Prägnanz, which makes similarly aligned lines appear parallel (D).

# 3 Methodology

This chapter describes how the process of island detection was designed, what software was used to develop and implement it, which data was used for the development and testing purposes and a step-by-step description of the creation of the individual parameters of the process. The process is roughly outlined in figure 3.1. The *manual input* steps list actions and decisions that need to be made by a human, before the results are fed into the algorithm. The steps in the *automated process* box then have functions that generate the end result automatically.

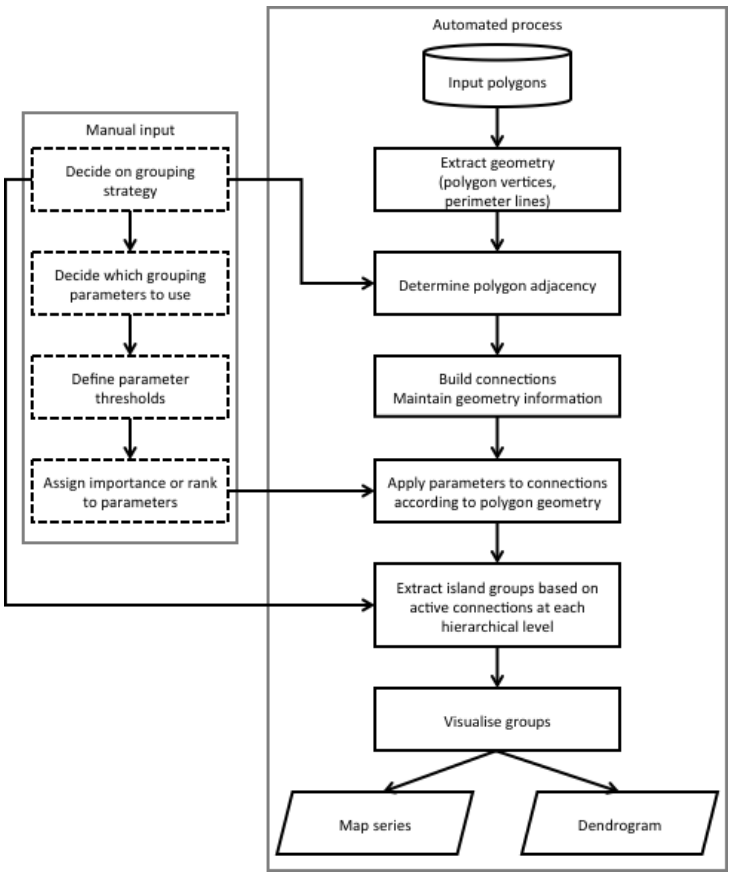


**Figure 3.1: A flow chart roughly illustrating the process of island group detection developed in this study**

## 3.1 Software

The software used for the process development was primarily ArcGIS 10.1 and its Python interface. The code for the structure recognition is written almost entirely in Python for application in ArcGIS. In the resulting code, many functions are general Python functions, possible to apply in most Python consoles, but some depend on tools provided in the ArcGIS toolbox for handling shapefiles and raster files. R was used to graph the island groups as dendrograms. R has a function for dendrogram creation, but in order to obtain the adequate dendrogram structure for grouping of islands, the Python console of ArcGIS was used to generate the CSV files with the appropriate format that could then be used in R to create the dendrogram. QGIS was used for small auxiliary tasks, mainly for its convenient Delaunay triangulation tool. Lastly, OCAD was used to draw many of the figures that illustrate the methods outlined below.

## 3.2 Data

The data used for the reverse engineering of the structure recognition and then for most study areas for the application were vector data for the islands off the east coast of central Sweden, close to the Stockholm and Lake Mälaren region (figure 3.2). This area is commonly called *Stockholms skärgård* or *Skärgården*, which can be translated as the Stockholm archipelago. The Stockholm archipelago is one of the largest archipelagos in the world when counting the number of islands, but most of the islands are small. According to Swedish statistics, the islands in the sea within Stockholm County number to close to 30,000 islands, but with neighbouring counties, the number of islands in the area rises to 60-70,000 (Statistiska Centralbyrån, 2009). In addition, neighbouring island groups belonging to Finland consist of over 40,000 islands, bringing the number of islands in the general region in the Baltic Sea to well over 100,000.


**Figure 3.2: Location of Stockholm archipelago**

The shapefile data of the Swedish islands were obtained from Metria, formerly a part of the Swedish National Land Survey agency, but now a separate state-owned enterprise, which manages geographical data for Sweden. The National Land Survey

produces vector data in several scales for several different purpose maps over all parts of Sweden. These have a varying degree of detail, depending on purpose. The property map (fastighetskartan – scale: 1:12,500) is the largest scale map, and the others in order are the terrain map (terrängkartan – scale: 1:50,000), the road map (vägkartan – scale: 1:100,000), overview map (översiktskartan – scale: 1:250,000) and Sweden maps of smaller scales. The road map was chosen for testing the grouping algorithm as it is somewhat simpler than the largest scale maps but still retains important features with regard to shape of islands in the Stockholm archipelago. The multiple feature layers and topographical detail that is included in these maps is not relevant to this work, so an automatic extraction of coast-line data was performed, providing semantically homogenous two-dimensional polygons. For the lakes around Dombes, France, the input data were vector data provided by Dresden Technical University, originally obtained from the OpenStreetMap database. All polygons were processed so that touching polygons were merged and holes were eliminated.

### 3.3 Background of test areas

In the most recent ice age, the Fennoscandian Shield (the name of the bedrock encompassing modern Norway, Sweden, Baltic Sea, Finland, Karelia and Kola Peninsula) witnessed a glaciation covering the whole shield plus areas stretching northeast to the Arctic and south to approximately where Berlin is situated. The maximum extent of this glaciation was at around 20,000 years ago, and the duration of the deglaciation was approximately 10,000 years. During this deglaciation period, most of the glacier retreated north while leaving an ever-growing lake behind, covering the south of the modern Baltic Sea and the lower lands surrounding it. This lake burst out twice and connected with the Atlantic Ocean, once over central Sweden to the Atlantic and then through the straits between Denmark and Sweden, creating what is now the Baltic Sea. During the glaciation, the land was pressed down by the heavy burden of the ice, and since the end of the last ice age, this land has been rising again in an isostatic rebound. While this isostatic rise is slowing down, it is still significant and results in a slight increase of land area for Sweden every year, both by movement of the coast and by the formation of new surfacing islands in the Baltic Sea. (Helmfrid, 1992; Sjöberg, 1996)

The lakes around Dombes are man-made ponds dating from the Middle Ages, when an extensive fish-breeding industry developed in the region (Sahrhage and Lundbeck, 2012, p. 63). The lakes therefore only follow a natural pattern inasmuch as the impervious bedrock dictates, but are more likely demarcated by anthropogenic constructions such as land boundaries and roads.

In these two regions, there are a great number of polygonal features that exhibit some visual patterns on a map, often including linear patterns. For that reason, these regions, especially the Stockholm archipelago, were considered likely to include groups of polygons that comprise patterns based on their geometric properties. Four sub-areas in the Stockholm archipelago data were examined further with regard to interesting properties that could test the algorithm.

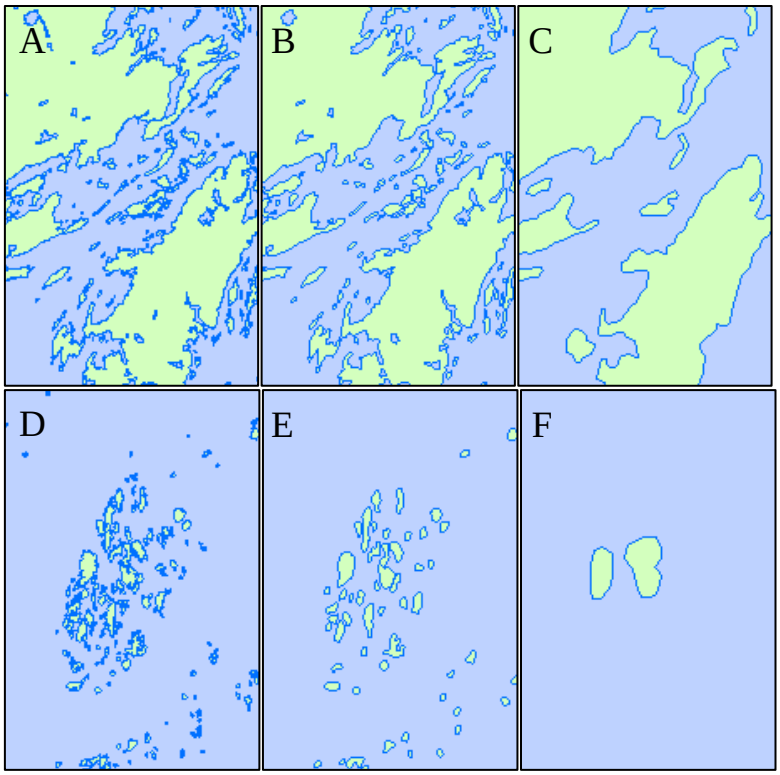


**Figure 3.3: Sub-groups in the Swedish Archipelago in different scales**

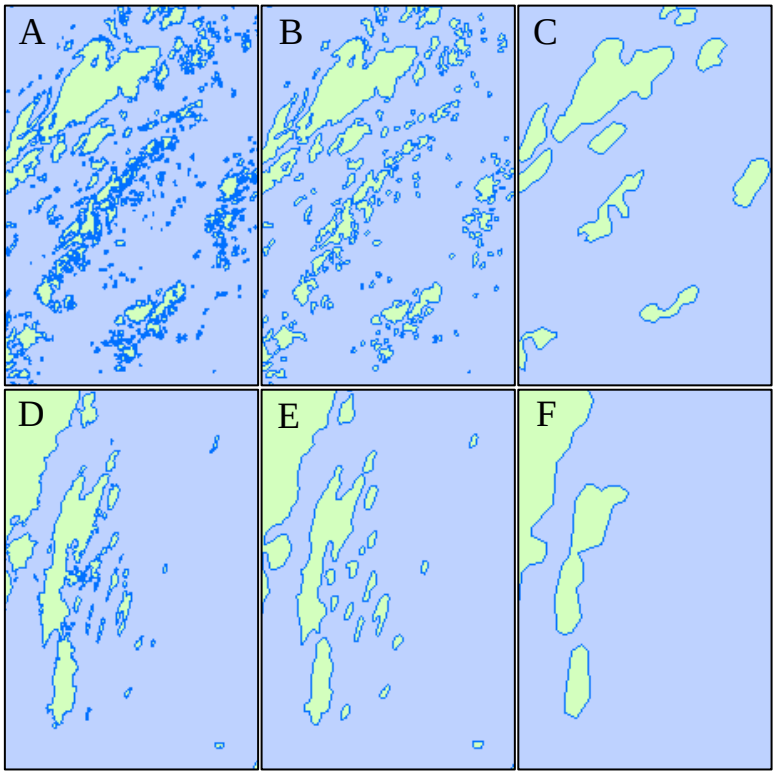


**Figure 3.4: Two additional sub-groups in the Swedish Archipelago in different scales**

Figures 3.3 and 3.4 show four different sub-areas in the Swedish archipelago. Each group is shown using the data of the Road map (made for the scale 1:100,000), then the Overview map (1:250,000) and one Swedish map (1:1,000,000). In groups containing islands of similar shape and orientation, some form seems to be preserved, especially between the first two where the generalisation operators seem to be mostly smoothing and typification. The latter scale change is a much bigger jump, to a scale where these island groups become hardly visible anymore without some elimination and simplification (e.g. figure 3.4E – F), or amalgamation (e.g. figure 3.3E – F). In the latter, most islands are eliminated, but the central islands increase in size for partial compensation.

## 3.4 Grouping strategy

For various tasks in the parameter definitions and for the island grouping strategy, a choice has to be made between several clustering possibilities. These questions are not sequential, but rather simultaneous, seeing as each choice affects the others.

One question pertains to whether the clustering is divisive or agglomerative. This has been touched upon in chapter **2** where in the study by Regnault (2005), the clustering was divisive, splitting clusters when the connections no longer fulfilled the parameters given according to that system. In brief, agglomerative clustering involves starting the process with each individual island defined as a cluster of one, and then building up the clusters using the connection to other islands and whichever connecting parameters and grouping strategies are used. Divisive clustering conversely starts with all the islands unified into one cluster, and then divides it into more and more by eliminating the connections.

Another clustering choice is whether clusters are hierarchical or not. Hierarchical agglomerative clusters make use of some calculation of distance from object to clusters. As the proximity parameter becomes more inclusive (allowed distance increases), the objects – while belonging to multiple smaller clusters in the beginning – join increasingly larger clusters, until at the top, all objects are parts of a single all-encompassing cluster. In divisive clustering, the hierarchy can be expressed by the ordering of each step during which the data are split. Hierarchical clustering can be illustrated using dendrograms.

Clustering methods can further be overlapping or exclusive. Quite simply, exclusive clustering allows each object to belong to only one cluster. Once it has joined a cluster, no other cluster stakes a claim to it. Overlapping clustering conversely allows for the possibility that objects belong to multiple clusters. In the survey done by Steiniger, Burghardt and Weibel (2007), participants identified island structures that overlapped.

Lastly, there are several different ways of assigning members to clusters during the evaluation of each object. These can be based on distances or probability computation from statistical evaluation of the properties of the cluster. There are also more simple ways of clustering, e.g. testing each object against nearby objects, or neighbouring objects that already belong in a cluster. Single-linkage clustering tests

each pair of objects that are not part of the same cluster in the order of proximity, starting with the closest pair and thus iteratively joining clusters in an agglomerative bottom-up fashion. There are variations on this type of clustering, e.g. complete-linkage, where instead of the two closest members of two clusters are checked, the two members that are furthest apart are. Average-linkage is where the average of all conceivable pairs between the two clusters is checked.

For the purpose of clustering of islands into groups, the choice has been made to apply an exclusive divisive hierarchical clustering for the parameters that pertain to the properties of each island. The choice of exclusive clustering is for two main reasons, i.e. the simplicity of illustration and application, and lower computer requirements. In principle, islands can possibly belong to more than one group at the same hierarchical level (Steiniger, Burghardt and Weibel, 2007). Therefore, a good grouping strategy would appreciate this even though it may be relatively rare. One problem with overlapping clustering in this study is the computing ability and time. In an area containing 100 islands, the number of clusters starts at 100 but quickly rises with each iteration as each cluster generates several new clusters. This occurs despite prevention of duplicates and even if clusters are removed once they are completely contained by another cluster. Further into the clustering, clusters should reach their maximum number and then start reducing until they are finally united into a single cluster containing all islands. In this study when pursuing this avenue, the set with around 100 islands would quickly grow to clusters numbering in the tens of thousands, taking a long time to process. In addition, there are some problems in displaying overlapping clustering, e.g. via dendrogram or map, and making the visualisation easily readable. The clustering is divisive because then connections can be considered to be either active or inactive depending on whether the connected islands fulfil certain parameter requirements or not. For agglomerative clustering, a more elaborate computation needs to be made, especially when combining variables and not using a single variable, e.g. proximity.

## 3.5 Connections and parameters

The clustering is performed by use of the connections between each pair of adjacent islands and a calculation to determine whether a connection is *active* or *inactive*. As an example, two islands are 5 km apart. During simple clustering where there is a fixed distance threshold of 10 km, the two islands would be close enough together to

constitute a group. Here, the term “active” is used to mean a connection that fulfils such parameter requirements and thus unites the islands on both ends. If the threshold were 3 km on the other hand, the two islands would not be united. That would conversely constitute an “inactive” connection.

The clustering will however not be performed with a single static parameter, but a combination of parameters adapted to the properties of the islands. For reasons discussed in chapters **1** and **2**, proximity is arguably the most important parameter by which to divide islands into groups. Size is another important parameter, as it tends to be important for general grouping of graphical shapes. As discussed above with examples of island group patterns, island shape can be important, especially when islands are formed in a long and narrow shape and are adjacent to other such islands. When this is the case, the orientation of the island also becomes interesting, as two long islands oriented the same way constitute a stronger pattern than ones oriented differently. Lastly, the distribution of the islands is important to whether they form a straight line, curved line or a more irregular cluster. These main principles are kept as a guide when programming the individual parameters applied to each island connection.

The parameters are programmed in such a way – using pre-defined thresholds – that each connection is tested for whether each parameter condition is fulfilled. Then, this combination of fulfilment or non-fulfilment of parameter requirements is applied on the connection in two main ways, which governs the grouping process.

1. A sequential application of parameters, dividing the island clusters with the step-wise inclusion of each parameter. The sequence of the application of parameters is an important factor in determining the end results. Here four sequences are tested.
2. An application of a combination of the parameters, with each parameter multiplied by a weight value. This results in a list of weight indices, and the division of islands into smaller groups happens when the weight threshold applied to the weight index list increases. The result is governed by the determination of the weight for each parameter, and in this paper, four different weight configurations are tried.

## 3.6 Connection creation

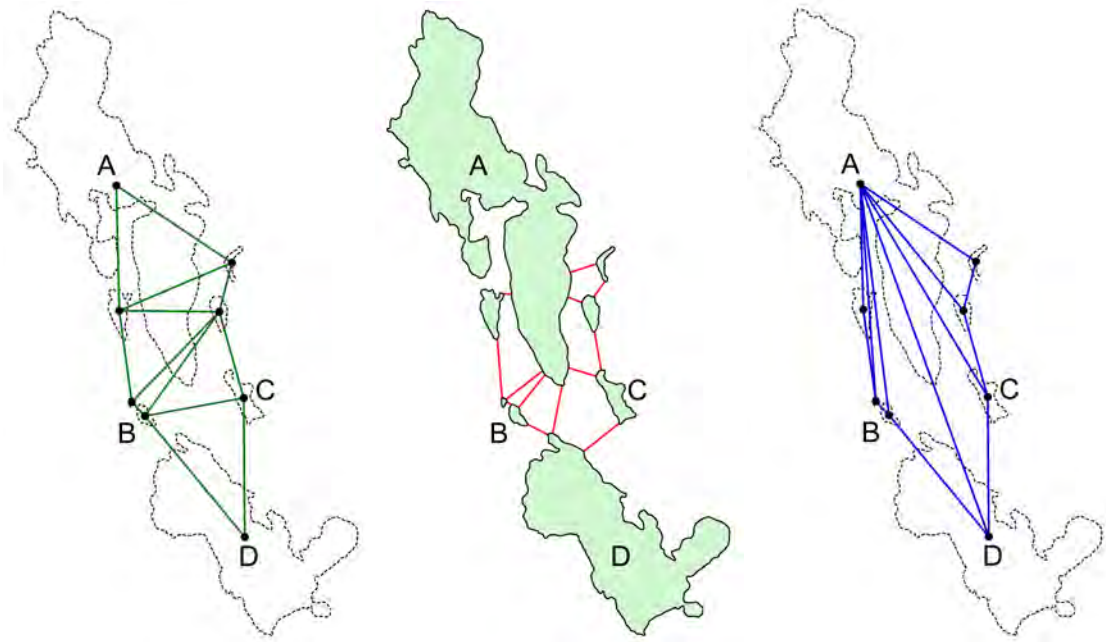


**Figure 3.5: Comparison of connection possibilities between islands**

In order to build connections between islands on which to base the grouping, it is necessary to determine which islands are adjacent to each other. In previous research with similar aims, i.e. to group polygons, the tendency has been to build these connections according to the principle of Delaunay triangulation, using the polygon centroids as the nodes by which to construct the triangles. The Delaunay triangulation however has some drawbacks for use with irregular polygons such as islands. Adjacency of nodes within Delaunay triangulation is completely dependent on distance to other nodes, without any regard to the shape that the node represents. That is, in the case of island centroids, large islands that are close to many smaller islands will not necessarily be deemed adjacent to these because as the large island is reduced to its centroid, proximity of its coastline to other coastlines is no longer relevant. This is exacerbated when two large islands lie close to each other but the two respective centroids are far apart. In the example in figure 3.5, the leftmost image shows several islands connected after a Delaunay triangulation of their centroids. Islands A and D are large, and their respective coastlines are closer together than those of islands B and C, yet because of the centroid positions, B and C are connected whereas A and D are not. Furthermore, several islands on each side of island A can connect across it as long as

their centroids are closer together than any of them is to centroid A. The middle image shows the same islands after the lines of shortest distance between coastlines have been drawn, with no crossing connections. The image on the right in figure 3.5 then shows the connections between the centroids as determined by this proximity determination. These shortest connections are obtained by the same principle as generating a near-neighbourhood polygon around each island and using the adjacencies as determined by the meeting of these polygons. The polygon vertices (e.g. of island polygons) are converted into points, making Thiessen polygons out of them to create nearest-neighbour areas for each (figure 3.6). These Thiessen polygons are used to determine which islands are adjacent to each other, and which nodes in each island pair constitute an adjacency pair. The node pair with the shortest distance in between is found and the line between them is the shortest distance path between the two islands. The principle of Delaunay triangulation can be said to apply here to some extent, as its inverse is the Voronoi diagram (Thiessen polygons) of the nodes. Here, this Voronoi diagram creation is instead done using the extent of the actual islands instead of only their centroids. Once these connections have been established, the parameters by which to determine island relations can be computed.

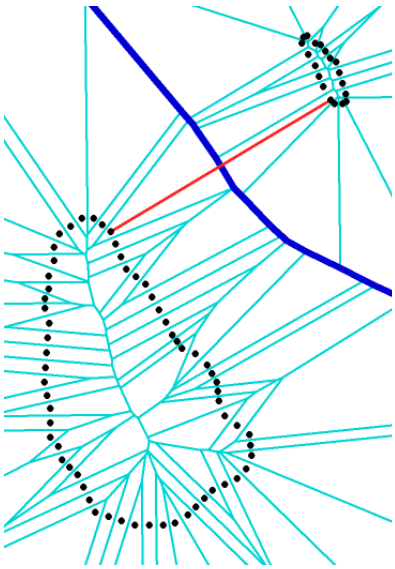

**Figure 3.6: Usage of Thiessen polygons to draw connections between islands.**

### 3.6.1 Parameter calculation

The parameters used for island grouping are meant to formalise the most important variables that describe geometric island relations. The first and most important parameter is the **proximity** parameter. In such divisive clustering, the proximity parameter is simply the test of whether the distance between islands is greater or less than a certain defined threshold distance. There are several possibilities for determining this proximity threshold, as is the case with the thresholds of most or all the other parameters. The method used here to decide whether two islands are close enough to each other to be grouped together is based on the method used by Steiniger, Burghardt and Weibel (2007), where the threshold can be expressed as the half of the length of the

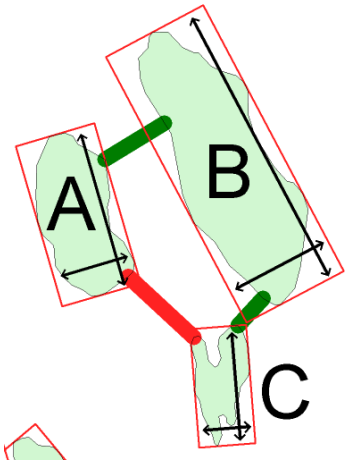


**Figure 3.7: Island proximity check using the distances and the dimensions of their minimum bounding rectangles**

minimum-bounding rectangle of the shorter island. To clarify, a minimum-bounding rectangle is created around both islands, and the longer axis of each rectangle is taken and divided by half. The shorter of the resulting two lengths is then defined as the threshold distance between those two particular islands. In the study by Steiniger, Burghardt and Weibel (2007), the description is such that each island is given a buffer zone with a radius stretching from its coast equal to half of the longer axis of the minimum-bounding rectangle, and is given a one-way connection to each island within that buffer. If an island within this buffer – after undergoing the same process and acquiring its own buffer search zone – does not acquire a buffer large enough to include the first island, the connection cannot be reciprocated and is therefore not considered. Only two-way connections are thus considered to be within this threshold. In the current study, since the method focuses on the connections between the islands and their relations to the qualities of the connected islands, the simplest way of calculating this is to take the shorter of the lengths of the minimum bounding rectangles of each island and testing it against the length of the connection line after dividing the former by half. Figure 3.7 shows three islands being checked against each other. The connections AB and BC are active, but connection AC is not because the distance between islands A and C is longer than half the length of the minimum-bounding rectangle around island C. One problem with this distance threshold is the fact that it is rather strict and will only group together islands that are quite close to each other in comparison to their size, and as a result, in an island group that has smaller and farther apart islands in general, no consideration is given to the relative distances within the island group. So, in a group of small and far away islands, while proximity clusters can be perceived because of relative internal distances compared to distances to other islands, this particular proximity calculation would not register the relative proximity. A good example of island groups exhibiting these properties is the atolls in the Pacific Ocean where islands can be very small – hundreds or thousands of kilometres apart – but still be perceived as belonging to the same group. One way to combat this effect could be to use a formula that decides the

distance threshold based on the typical distances in the whole group of islands and thus taking into consideration the relative distance or proximity between islands. Another way could be to increase the distance threshold in case of islands that exhibit other patterns and are therefore more likely to belong together, e.g. if they are distributed in a line. Then, the strict distance threshold would only be applied to island pairs or groups, whose uniting parameter is only proximity, but allow for longer distances for islands that can be united via other parameters. Given that the main study areas consist of islands in the Swedish archipelago where islands are often close together, the method for proximity determination using the minimum bounding rectangles is adequate. In the application of this parameter, each connection is tested using the minimum-bounding rectangle of each island it connects, and a list is created with indices parallel to the list of connections, with a value 0 if the proximity requirement is not fulfilled and 1 if it is.

The next parameter to generate is the **size** agreement between island pairs. This parameter may be the most straightforward to calculate, but it still requires some arbitrary decision making as to which method to apply. In this study, this parameter is simply a test whether two islands are of a similar enough size to be considered a group. First, the area of each island is obtained. Here, a threshold needs to be determined as to what the size difference between two adjacent islands can be. The calculation could be more elaborate if the clustering method consisted of using a mean or an otherwise statistic evaluation of the existing clusters, but in order to keep the task simple, each island pair as connected via each connection is simply tested for size ratio. The aim in the parameter decisions is to use a simple and straightforward method, with the help of visual inspection where methods rooted in a theoretical background are difficult to apply. Here, the threshold ratio of 1:2 is used, i.e. two islands are deemed close enough in size if the smaller island is no less than half of the size of the larger one, based on a visual inspection. Figure 3.8 shows four islands being tested, with two pairs of similarly sized islands (connected with green lines) emerging. A list is created and each connection

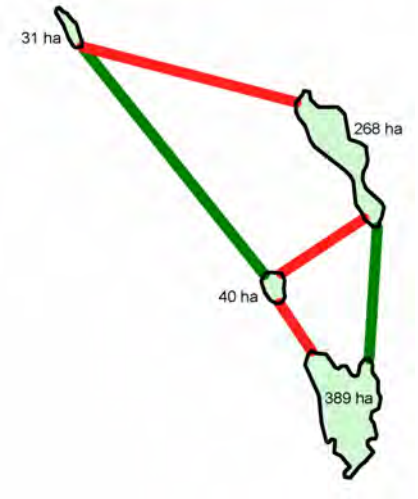


**Figure 3.8: Islands joined depending on their size similarity**

is tested, taking each island pair and testing their size ratio against the tolerance ratio 1:2. 1 is added where the ratio is larger than the tolerance ratio and 0 where it is not.

Maybe the most difficult parameter to formalise is the one of **shape and orientation** of each island. Shape and orientation are two separate parameters, but they are considered together here because of the close connection between them and the fact that the determination of them is much intertwined. Both of these parameters are somewhat fuzzy in definition, even though the concepts may appear simple. The shape of an object can be described with respect to the complexity of its perimeter, its similarity to a circle with the same perimeter or area, the ratio between the length and width of its corresponding minimum bounding rectangle (which can further be divided into the rectangle with the minimum area or minimum width) amongst other methods. These have advantages such as simplicity and quickness of computation, but drawbacks such as the non-consideration of parts of islands or mainland areas that may exhibit a continuation of island group patterns. Here, another approach is tried by creating a central skeleton of each island and extracting the skeleton parts where an island is longer than it is narrow according to a defined ratio. This ratio is arbitrarily set at 1:2, which means that an edge is considered to describe the shape of a long and narrow part of an island if the distance to the coast in the direction along the edge is twice the length of the distance to the coast perpendicular to the edge. This skeleton is created from the Thiessen polygons that were generated when building the connections, using the coastline vertices. The polygons are converted to lines, and since the central skeleton of each island is sought, all lines that either intersect with the coastline or lie at sea are deleted. The remaining lines now resemble island skeletons (black lines in figure 3.9, left), but as the desired result is lines that run through long narrow islands or parts of islands rather than all Thiessen polygon borders, each short edge is processed thus:

- To eliminate edges that are unlikely to describe a line running through a long and narrow part of the island, each edge only connected at one end with another edge where the angle difference between the edges is less than 45° is deleted. This removes all ultimate edges that only have one or no connections and edges that connect to a significant bend in the centre line.
- Coordinates of the two end vertices and the centre point of each remaining edge are put in a list.
- The end vertices are used to get the direction of the edge.
- The centre point is used as the starting point for four new lines, reaching from the centre point to the nearest coastline in a parallel and perpendicular direction with regard to edge. Each line pair is then regarded as a single line, giving one parallel line and one perpendicular (figure 3.9, centre).
- The ratio between the parallel and perpendicular lines is calculated and put into a list, corresponding to each edge index.
- The ratio corresponding to each edge is tested against the ratio 1:2 and the edges where the parallel line is at least twice as long as the perpendicular line is kept as a skeleton and the others are deleted (figure 3.9, right).

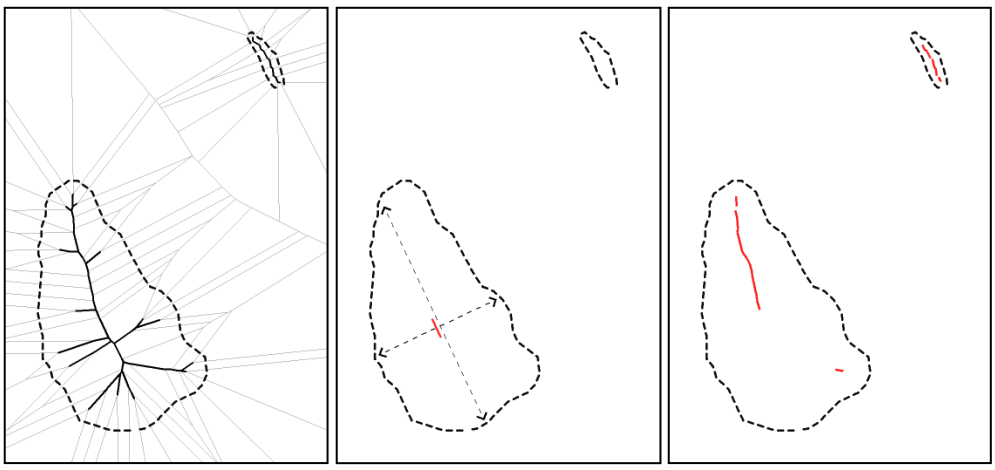

**Figure 3.9: Creation of an island skeleton and search for long and narrow island parts**

Orientation of an island or a polygon object can be derived in a number of ways. One quick way to determine an island's orientation is the determination of the direction of the length line of the polygon, the length line being the longer axis of a minimum-bounding rectangle. There are other ways, such as a PCA (principal component analysis) where the islands are converted to points, or an averaged direction calculation based on each edge comprising the coastline of each island, weighted against the edge length (Steiniger, Burghardt and Weibel, 2007). In this study, each island is not given a pure shape or orientation value according to a calculated index. Rather, the aforementioned skeleton is used to formalise the shape of an island for the purpose of testing against the shape of neighbouring islands. The components of the skeleton are still short individual edges comprising branches of a larger structure or isolated parts within or on the fringes of the skeleton structure. The edges are first clustered together in such a way that unites the branches of the skeleton, i.e. all edges that share a node are merged if the node is shared by exactly two edges. This means that all touching edges are combined into an edge cluster as long as they do not touch over an intersection of 3 or more lines. Edges in the skeleton can have been deleted in the earlier step because the position of the edge itself proved not to lie in an area exhibiting the ratio requirement even though the edge's neighbours were kept, and as a result, parts of the skeleton can in places be separated where they should still be considered a continuation of the same linear pattern. For this reason, each united skeleton branch is now tested against each other using an angle-difference tolerance as the threshold. For simplicity, a principle similar to that of single linkage clustering is used here. Each possible branch pair is put into a list ordered first by distance and then by length of the two branches combined. The latter ordering is added for cases where 3 or more branches meet in a single point, as the distance between the three (or more) pairs will be identical. Since the clustering is exclusive, i.e. each branch can only join one cluster, the potentially most important branch pair is considered first on the basis that the longer a branch is, the more important it is in describing a significant portion of the island.

The angle difference check takes each branch pair and first checks if the two branches share an end point. If they do, a simple check is performed, whether the direction of the two branches (determined by the straight line between the endpoints) are within the given tolerance angle of each other (45°). If the branches do not touch,

however, the proximal points are extracted. These proximal points are the endpoints on each branch that are closest to each other and thus constitute the closest connection between the two branches. The proximal points form an edge by themselves with a length and a direction like any other edge. This edge is then angle-checked against each branch using the 45° tolerance angle. The two branches, if united, are now considered a single cluster with the length as combined length of the constituent branches and an angle that of the direction determined by the two farthest end points. This check continues using the resulting clusters until there is no more change. When the last iteration of the branch pair check of each island is complete, the final clusters or unmerged branches are checked for length. A branch cluster is not considered a significant skeletal line of a narrow island if the length of the cluster is below a certain threshold length. In this algorithm, that length is given as half of the length of the island's minimum-bounding rectangle, but that may be too strict. When the branches on each island have been united based on the angle check and then checked for length, each pair of adjacent islands is processed so that each skeleton branch cluster on each island is tested against each cluster on the other island. The angle check performed here has two thresholds. If lines on each island fulfil the conditions of an angle check with a narrow angle (15°) as the tolerance (i.e. they have a similar direction and they follow as a good continuation of each other) they are deemed to agree in both shape and orientation and straight continuation. If the lines have a similar direction within a wider tolerance angle (30°) but do not appear to be a continuing line (e.g. two approximately parallel lines side by side) the islands are considered to agree in shape and orientation but not in continuation. Figure 3.10 shows three islands having been processed this way. Each unbroken red line on each island is created from the skeleton-creating process and constitutes a branch of the skeleton of that island. The connections between the islands are given values based on whether these skeleton branches, after being united and checked against the length of the islands, have a similar direction and if they form a continuing line. In one case, both criteria are fulfilled, and in the other, only one is.

If an island is relatively circular in shape, it is unlikely to have any of the skeletal lines that describe a long and narrow island. Two islands can however be circular in shape and therefore appear similar even though they cannot as a result have any particular orientation. A shape index is therefore applied to each island where the length of the perimeter is divided by the length of the perimeter of a circle of the same size as the island. This shape index for determining the compactness of islands is listed in AGENT Consortium (1999), and the specific formula is obtained from Zhang (2012, p. 99). Islands that have a high shape index and are therefore irregular in shape can be vastly different to each other, so the shape index is only useful in determining similarity of islands if both islands are relatively circular. There are no universal guidelines as to the values of this index and what circularity thresholds to apply, but a visual check shows that polygons with a shape index smaller than 1.2 (where 1 is a perfect circle and more irregular shapes have higher values) look circular and those with a higher value look more complicated. The circularity threshold is therefore set at 1.2. Naturally, since a circular object is not deemed able to possess orientation, this shape check only checks if two islands both have a circularity value lower than 1.2 (figure 3.11).

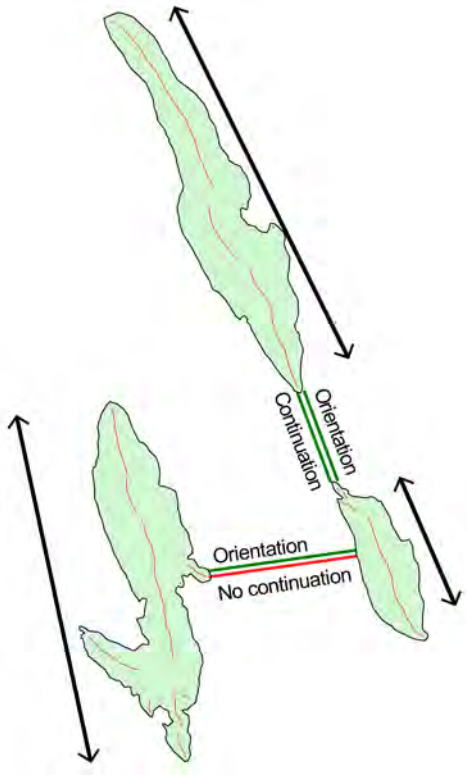


**Figure 3.10: Islands joined depending on orientation agreement or continuation**

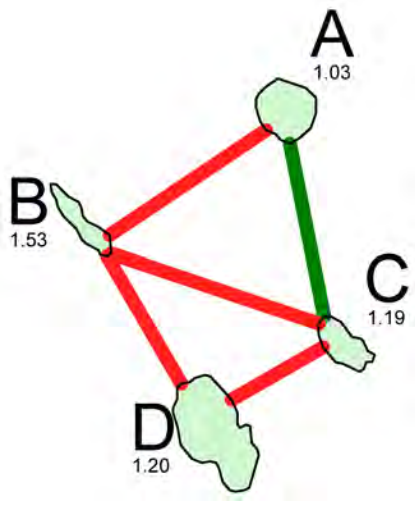


**Figure 3.11: Islands joined depending on their shape index**

The circularity agreement described above indicates an absence of orientation agreement, but not a disagreement, and is therefore used. Islands that prove to be long and narrow and therefore have an agreement in shape, but fail the 45° angle check, can be considered to disagree with regard to

orientation and are therefore not united. This approach is rather more complicated than a comparison of the length ratio and orientation of the sides of a minimum-bounding rectangle of each island and it is unknown whether such an elaborate method is necessary in order to get acceptable results. However, as islands often are irregular in shape, it seems logical to devise a way of appreciating that parts of islands or mainland areas can describe some structural relations to other islands or island parts.

This process results in two lists. One list is called **general shape**, and it checks each connection for whether the islands either fulfil the circularity similarity test (both have a shape index below 1.2) or if they have skeleton branch clusters that are oriented within a 30° angle of each other. This would mean they are either both circular in shape or they have long and narrow parts which are similarly oriented (both connections in figure 3.10 and connection AC in figure 3.11). The other list is that of **linear shape**. It requires that two islands have skeleton branches whose connecting edge (using the proximal points) has an orientation difference of less than 15° from either branch and thus constitute a relatively straight line (e.g. the upper connection in figure 3.10). Each list as the previous lists is filled with 1 or 0 depending on whether the parameter criteria are met or not.

### 3.6.2 Application of parameters

As explained in part **3.5**, two approaches are tested in the application of the parameters that have been created. The first is a sequential application where with each successive step only active connections from the previous step are processed. The second is a weight index calculation enabling the simultaneous application of all parameters without strict exclusions. Both are hierarchical in nature.

In the sequential application of the parameters, the first step is to decide the order of the parameters that should be applied to split the clusters. In previous researches into island group detection, the **proximity** parameter is deemed the most important parameter in the grouping, especially with regard to meso-structures (island groups of ca. 12 islands or more). The available parameters to further divide the clusters are therefore **size**, **general shape** and **linear shape**. It is somewhat arbitrary which order to use for these three parameters, so some experimentation is justified. Also, while proximity is the most important parameter, it is possible to downgrade it in the sequence

to evaluate other parameters. Here, four configurations of parameter sequencing are tested:

S1.
Proximity, then
size, then
general shape, then
linear shape

S2.
Size, then
proximity, then
general shape, then
linear shape

S3.
General shape, then
proximity, then
size, then
linear shape

S4.
Proximity, then
general shape, then
linear shape, then
size

In the initial state all the islands belong to a single cluster, as if all the connections were active. As the first parameter is applied, some of these connections go inactive. When enough connections have thus been deactivated, so that two islands do not share a connection either directly or via other active connections, a cluster undergoes a split. This is performed four times according to each of the four configurations above. While the configurations are meant to test different sequencing of parameter application, the parameter **linear shape** is almost always had last as it is the most exclusive parameter and therefore an initial application using it would remove too many connections in the beginning unless multiple island pairs in the test area exhibit continuing straight line patterns. The exception is the last configuration, where both **shape** parameters are put ahead of **size.**

In the weighted index application, the first step is to decide the importance of each parameter by assigning it a weight. In an example with 5 connections, four parameters are considered, the same as in this paper. Table 3.1 shows the fulfilment of parameter requirement of each connection index. Next, the importance of each parameter is determined. Table 3.2 lists the parameters and the associated weights, showing **proximity** to be most important, then **linear shape**, then **general shape**, and lastly **size**. This results in a weighted index shown in table 3.3, where the parameters have been added up after the application of the weights. The hierarchy then consists of

grouping all the islands whose connections belong in each weight step. In this example, the top cluster would include all connections (and their associated islands), the next would drop connection index 2, then 3, 1, 0, 5 and end up with a single connection (index 4), or a cluster of two islands, for the bottom level. Table 3.4 shows an initial state for each connection according to the first (leftmost) parameter in table 3.1 and the progressive change, or elimination, as each parameter is applied sequentially (moving right).

**Table 3.1: Connection indices and the fulfilment status of each connected parameter**

| | Parameters | | | |
|---|---|---|---|---|
| Connection Index | Proximity | Size | General shape | Linear shape |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 |
| 3 | 0 | 1 | 0 | 0 |
| 4 | 1 | 1 | 1 | 1 |
| 5 | 1 | 1 | 1 | 0 |

**Table 3.2: The weights applied to each parameter**

| Parameter | Weight |
|---|---|
| Proximity | 4 |
| Size | 1 |
| General shape | 2 |
| Linear shape | 3 |

**Table 3.3: Values for the connections from the parameter weighting**

| Connection Index | Weighted parameters |
|---|---|
| 0 | 5 |
| 1 | 4 |
| 2 | 0 |
| 3 | 1 |
| 4 | 10 |
| 5 | 7 |

**Table 3.4: Stepwise deactivation of connection based on the parameter sequence**

| Connection Index | Sequential parameters |
|---|---|
| 0 | 1->1->0->0 |
| 1 | 1->0->0->0 |
| 2 | 0->0->0->0 |
| 3 | 0->0->0->0 |
| 4 | 1->1->1->1 |
| 5 | 1->1->1->0 |

As each successive parameter is applied, any 0 in the list results in a 0 for the remaining steps. Like in the weighted parameter application, in this hypothetical example, the end result would be a single active connection (connection index 4).

The process is better illustrated using map results. In a hypothetical example, an island group has 11 islands, and four separate parameters have been applied to the connection between them (figure 3.12). The first experiment applies these in sequence, so that only active (green) connections from the first step are tested in the second step, and so on. As more connections go inactive (red), the groups get smaller (figure 3.13). The second experiment applies weight index thresholds to each connection. Here, all the parameters in figure 3.12 have simply been given the weight 1, so the weight index of each connection simply indicates how many of the parameters combine to unite islands

of each connection without assigning any ranked importance to the parameters. Then, as each weight threshold value (from 1-4) is applied to the image in sequence, more and more connections go inactive and the groups are split (figure 3.14).

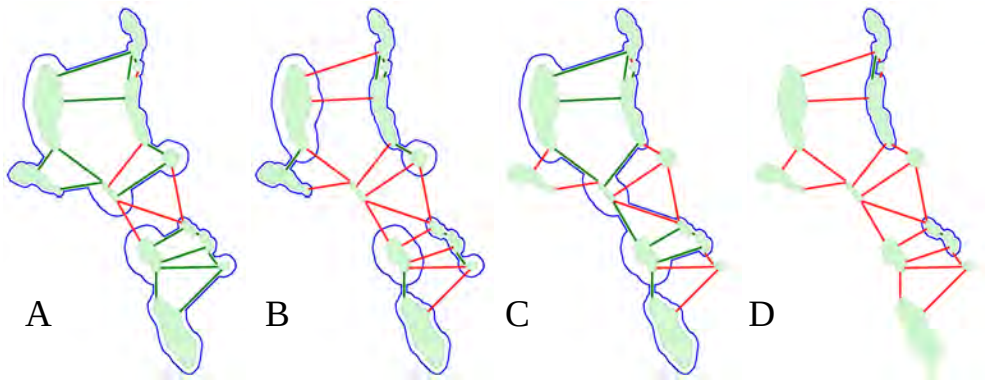


**Figure 3.12: Hypothetical example of an island group divided by four separate parameters, A, B, C and D**

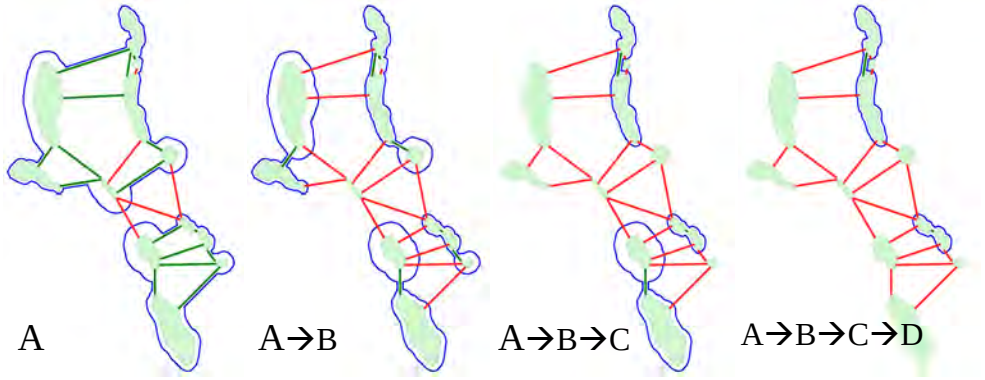


**Figure 3.13: Illustration of the division process when parameters are applied in sequence**

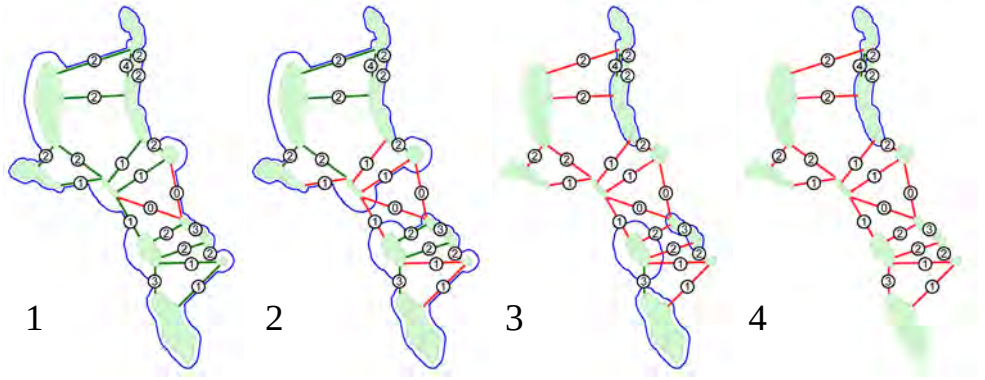


**Figure 3.14: Illustration of the division process when weights have been applied to each parameter. Numbers in circles indicate the weight index calculated for each connection. Numbers in the corner of each image indicates the weight used to divide the groups at each step.**

In this paper, four configurations are tried in order to see if there is significant difference depending on the weight designation:

W1.
- Proximity = 4
- Linear shape = 3
- General shape = 2
- Size = 1

W2.
- Proximity = 10
- Linear shape = 5
- General shape = 3
- Size = 1

W3.
- Proximity = 2
- Linear shape = 4
- General shape = 3
- Size = 1

W4.
- Proximity = 3
- Linear shape = 2
- General shape = 1
- Size = 4

## 3.7 Preventing gaps

When islands are grouped based on common properties, it is conceivable that islands connected via active connections form a cluster around islands that have no active connections to any of the cluster members. While it is possible for sub-groups to exist within island groups, conceivably within the same hierarchical level (in overlapping clustering) and especially on a different level, it is not logical for an island group to have a hole containing one or more islands that do not belong to any group. When grouping islands manually, a cartographer would presumably draw a perimeter around the group, but not holes inside that perimeter around the islands that do not belong. As a result, the algorithm performs one extra step, testing whether a group has a hole in it, and if so, merges whichever islands are on the inside with the group. Figure 3.15 shows a process where two clusters are consolidating along the active connections (shown in green). The two clusters in the leftmost picture join in the middle picture, but an island with no active connection to any other island is captured in the middle. This island is subsequently joined, even if it does not share the properties that were used to join the other islands, to prevent holes in the island group.

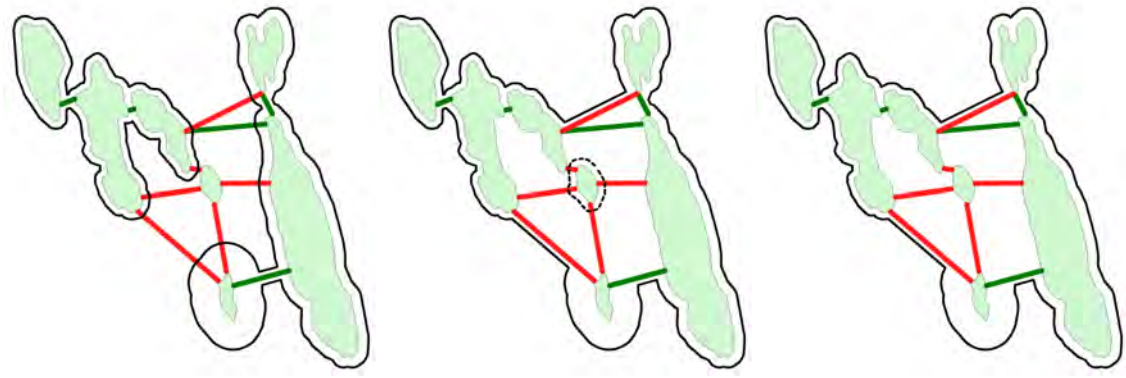

**Figure 3.15: Joining of a cluster with a surrounded island. Active connections are green and inactive connections are red.**

## 3.8 Graphing the clusters

The results of the clustering can be displayed using a dendrogram. Since the clustering is exclusive, the dendrogram is a relatively simple construction when the number of islands is not too great. In the dendrogram, the top tier represents the top step, encompassing a cluster of all islands in the map. Each step in the y-axis then represents a step in the hierarchy where a split can occur. At the bottom of the dendrogram, no islands are clustered together, illustrating the maximum division where every island is a cluster of one object. For legibility reasons however, the full output of group dendrograms will not be presented, but rather, an example using some of the smaller island groups in order to show examples of such graphing.

## 3.9 Test area descriptions

Figure 3.16 shows maps of the island groups used to test the algorithm These are the same areas as were shown in figures 3.3 and 3.4. Figure 3.17 shows two of the island groups with the polygon IDs overlaid, showing the island IDs that correspond to the dendrograms. Figure 3.18 shows a map of the Dombes lake area.

Test area 1 (TA1) consists of 73 islands, generally positioned in a line running from south-west to north-east. The islands are very variable in shape and size. Many are somewhat long and narrow but others are very irregular. Some patterns can be discerned by the continuation of lines through adjacent islands. The islands are not spread out evenly; most of the islands are noticeably gathered in a central cluster with a few islands

to the north-east and to the south-west. Some linear patterns are discernible, primarily in a south-west to north-east direction.

Test area 2 (TA2) is a larger archipelago than TA1 in number of islands. It consists of 113 islands and is generally compact in distribution. There is significant size variability, and the shape of the islands is generally irregular. There are a few examples of islands that have a shape agreement whether it is circularity or orientation, and some adjacent islands are similar in size, but the only general shape pattern discernible throughout the area is a north-south linearity pattern exhibited by a handful of islands.

Test area 3 (TA3) is the largest area here in terms of the number of islands. It includes 193 islands positioned along a direction from south-west to north-east. In the north-western half of the group, the islands are larger in area and more packed together, with the south-eastern area more spread out and consisting of smaller islands. Island shape is very irregular, especially amongst the larger islands, but even for their complex coastlines, the island group on the whole appears to form somewhat of a linear pattern running from south-east to north-west. The smaller islands exhibit no such pattern and strong linear patterns of long, narrow islands are not to be found in this group.

Test area 4 (TA4) is a small group consisting of only 39 islands. Most of the islands are situated to the west of the map, with two very large islands and the rest much smaller. Several islands are long and narrow and follow the same south-south-west to north-north-east direction, and a few are so situated that they could form a relatively straight line if a line were drawn between them.

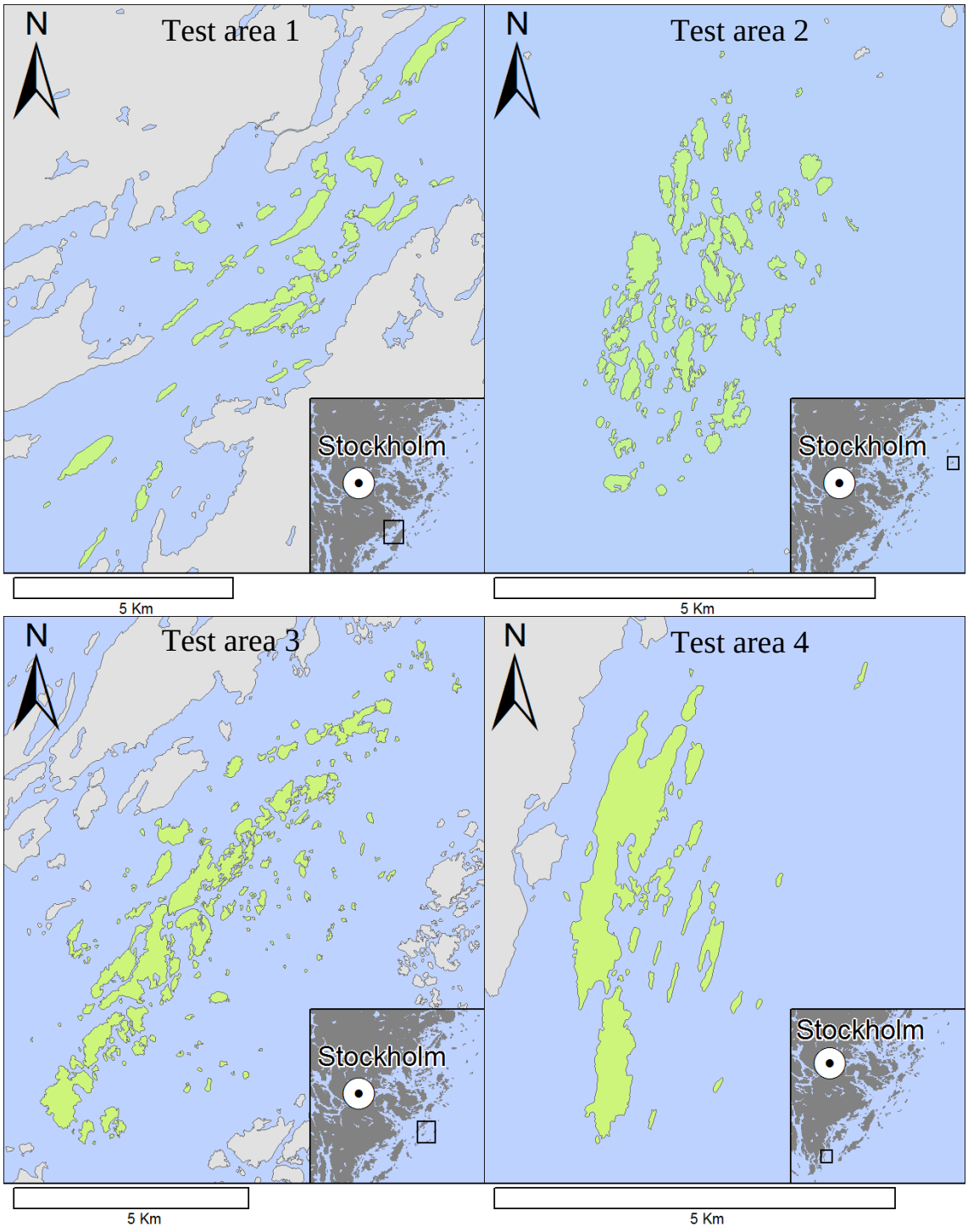


**Figure 3.16: Maps of the four test island areas and their location within the Stockholm archipelago**

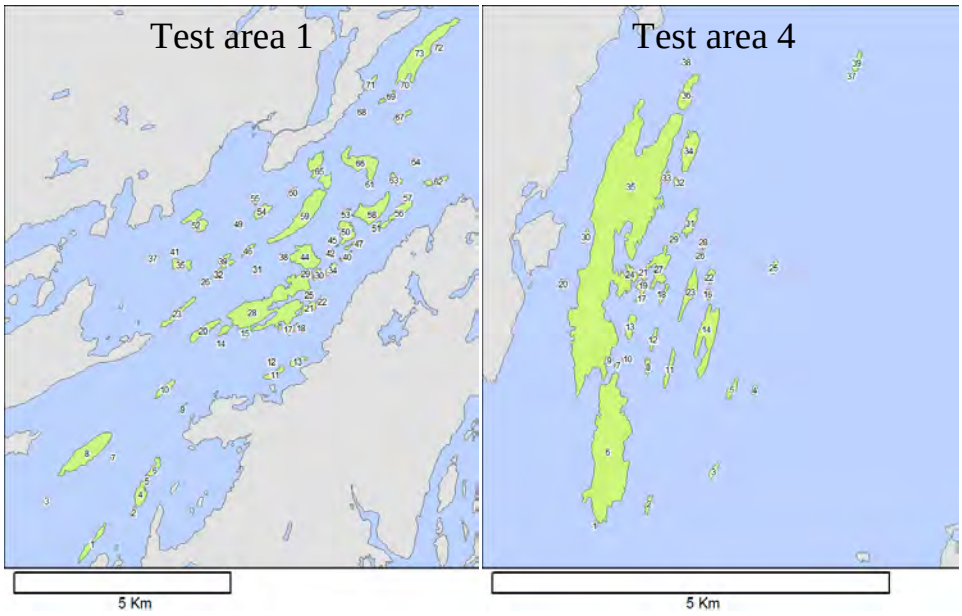


**Figure 3.17: Island test areas with ID labels**

Test area 5 (TA5) consists of the lakes around Dombes, France, a few kilometres north-east of Lyon. The number of lakes is a subject to fuzzy definitions, but using this particular dataset, the area consists of 803 lake polygons. There are a few discernible line patterns in the data, both in the shape of some lakes and in their distribution. These patterns depend on location within the area, as the direction of these line patterns is not uniform throughout the whole map.

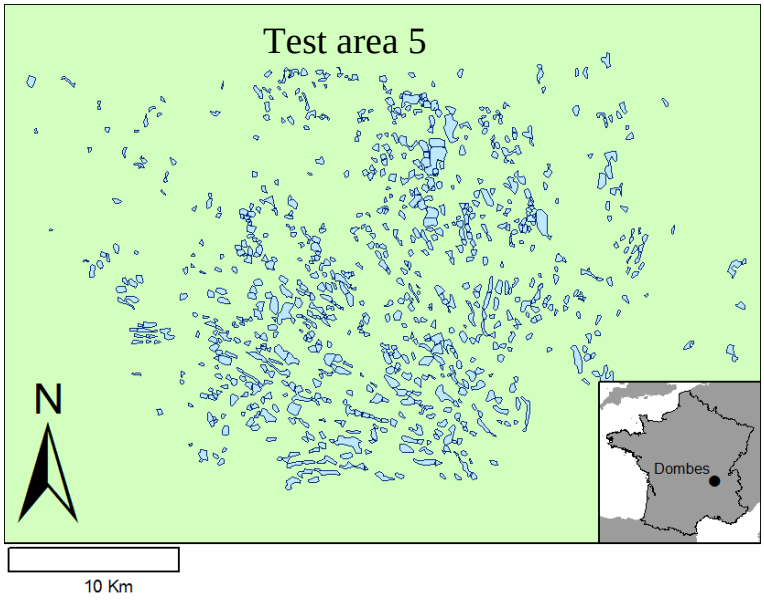


**Figure 3.18: A map of the lakes around Dombes, France**

# 4 Results

The resulting products from this process, as shown in the flowchart in figure 3.1, are graphs of the hierarchical grouping of the islands in the form of dendrograms, and map series, illustrating the corresponding groups. The actual results are therefore in picture form. In this chapter, these output pictures are presented with accompanying observations and interpretations to give them meaning. Further discussion of these results then follows in the **discussion** chapter.

The resulting figures of the grouping outputs are ordered by grouping strategies. First, there are select dendrograms resulting from the sequential parameter application, followed by maps from the same grouping, showing all test areas (TA) in succession with individual parameters applied separately in the top row, and then the grouping by parameters in sequence in the following rows. The numbers indicate the order of the parameters according to their number in the top row. Next are the dendrograms of the application of weighted parameters, followed by their accompanying maps. The numbers on the maps represent the weight threshold applied to the connections after weighing the parameters, resulting in each map. The configurations that the figure captions refer to are those listed in part **3.6.2**. The maps show a perimeter around each group, and connections inside the groups. These connections are red when active and grey when inactive. Inactive connections between groups are not shown. In test area 5, these connections are not shown due to the number of polygons in the area and instead only a red perimeter around the groups is displayed.

## 4.1 Sequential parameter application

Figure 4.1 shows the resulting dendrograms from the sequential parameter application of TA4. TA4 has fewer islands than the other test areas, and therefore the dendrograms are easier to discern and read, although the island IDs become too small when displayed in a series, so the last image (configuration W4) is shown separately in figure 4.2. The dendrograms here serve mainly as an illustration of how such a hierarchical island grouping can be graphed according to the methods in this paper, but some clues to the resulting clusters can be seen in the shape of the dendrograms. In all the sequential parameter dendrograms for all test areas, the vertical drops are generally very long as most of the branching takes place in the first two steps. There is also a different outcome depending on which sequence of parameters (which configuration) is chosen.

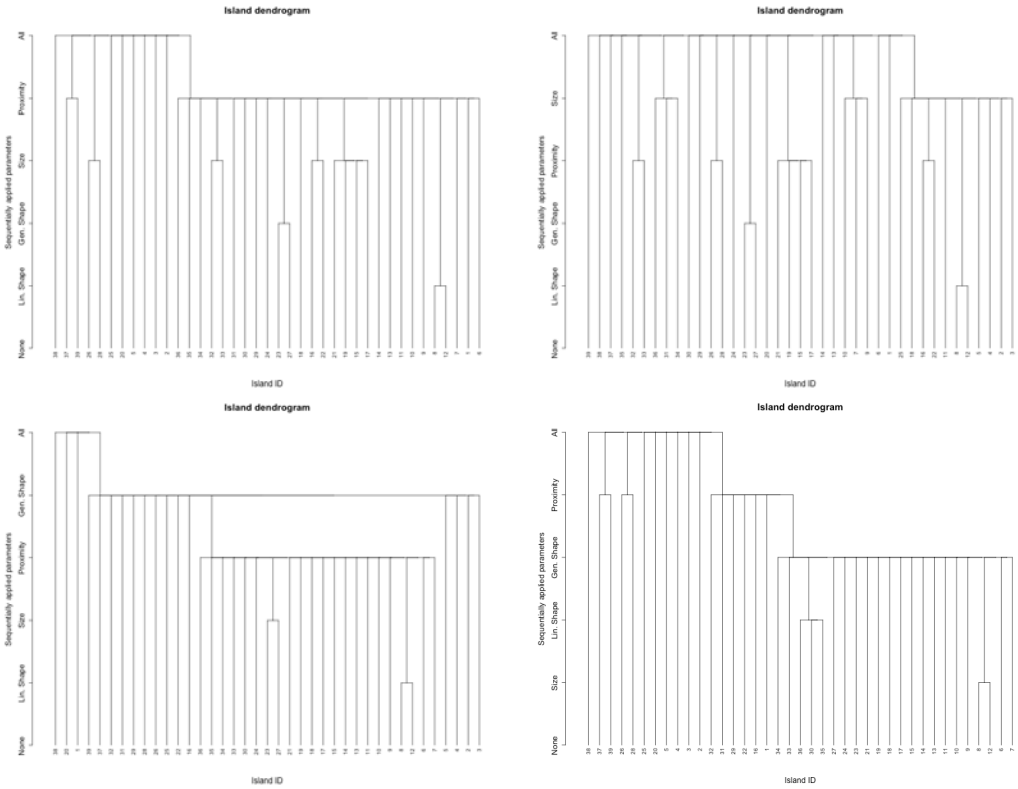


**Figure 4.1: Dendrograms of the sequential parameter applications on TA4**
**top left: configuration S1, top right: configuration S2**
**bottom left: configuration S3, bottom right: configuration S4**

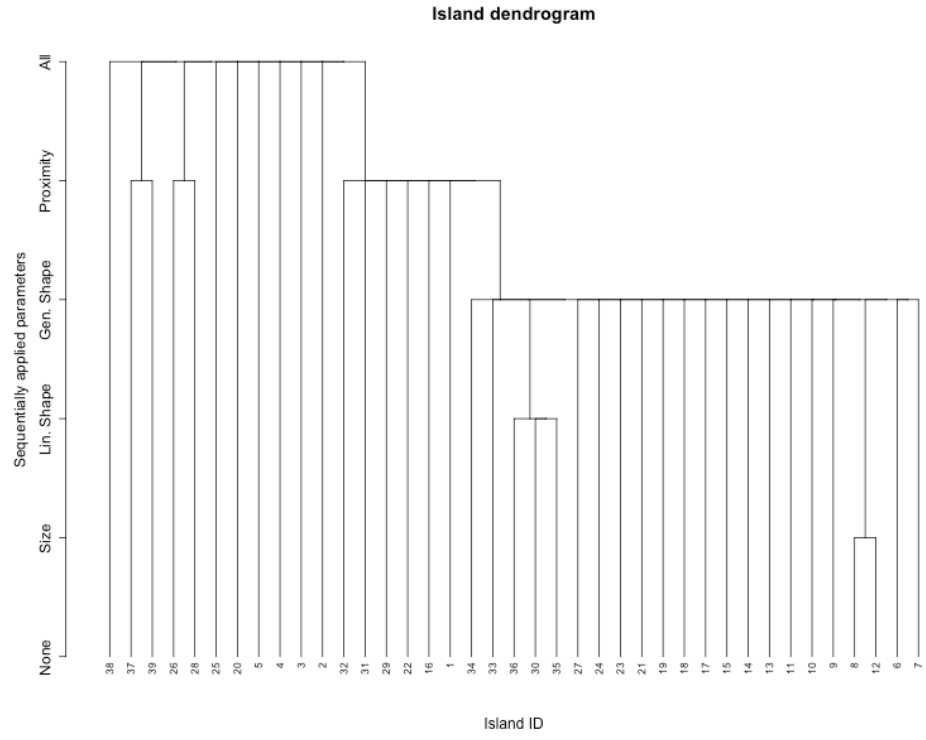


**Figure 4.2: Dendrogram of the sequential parameter application on TA4 according to configuration S4**

The next five pages show each test area in succession (figures 4.3, 4.4, 4.5, 4.6 and 4.7). The top row in each figure shows the island grouping result after applying each of the parameters – proximity, size, general shape and linear shape – separately. By design, the distance parameter shows the most compact clusters within the island groups. The size parameter results in rather fragmented clusters of potentially far-between, but similarly sized islands. The shape parameter clusters are also fragmented in places, but successfully group adjacent islands that point in a similar direction or are similarly circular. Lastly, the linear shape parameter results in island groups that look like a straight line in shape and distribution. Usually, the linear shape parameter generates very few and small groups. Each remaining row in the figures shows each island group divided by the successive application of the parameters in the order indicated by the numbers. In TA1, TA2 and TA4, the groups have become small and few in number by the third step, which conforms to the dendrograms and the tendency to be mostly split into individual leaves after the second division. Finally in TA1 and TA2, no clusters of more than one individual island remain by the application of the fourth parameter. TA3 and the Dombes area retain several clusters after the third step in all configurations, but after the last step, there are no clusters with more than two islands. There is some difference in the clusters obtained based on which of the three configurations is used, but the best results with regard to meaningful groups of islands that belong together are primarily obtained by use of the distance parameter alone, or a combination of distance and size or distance and shape. Further division seems to become too exclusive and less meaningful. In TA1 and TA4 where there are several islands forming straight lines, the **linear shape** parameter yields interesting results when applied separately. In TA2 and TA3, linear shape however does not find neat straight lines, but connects islands whose parts still form a line according to the algorithm.

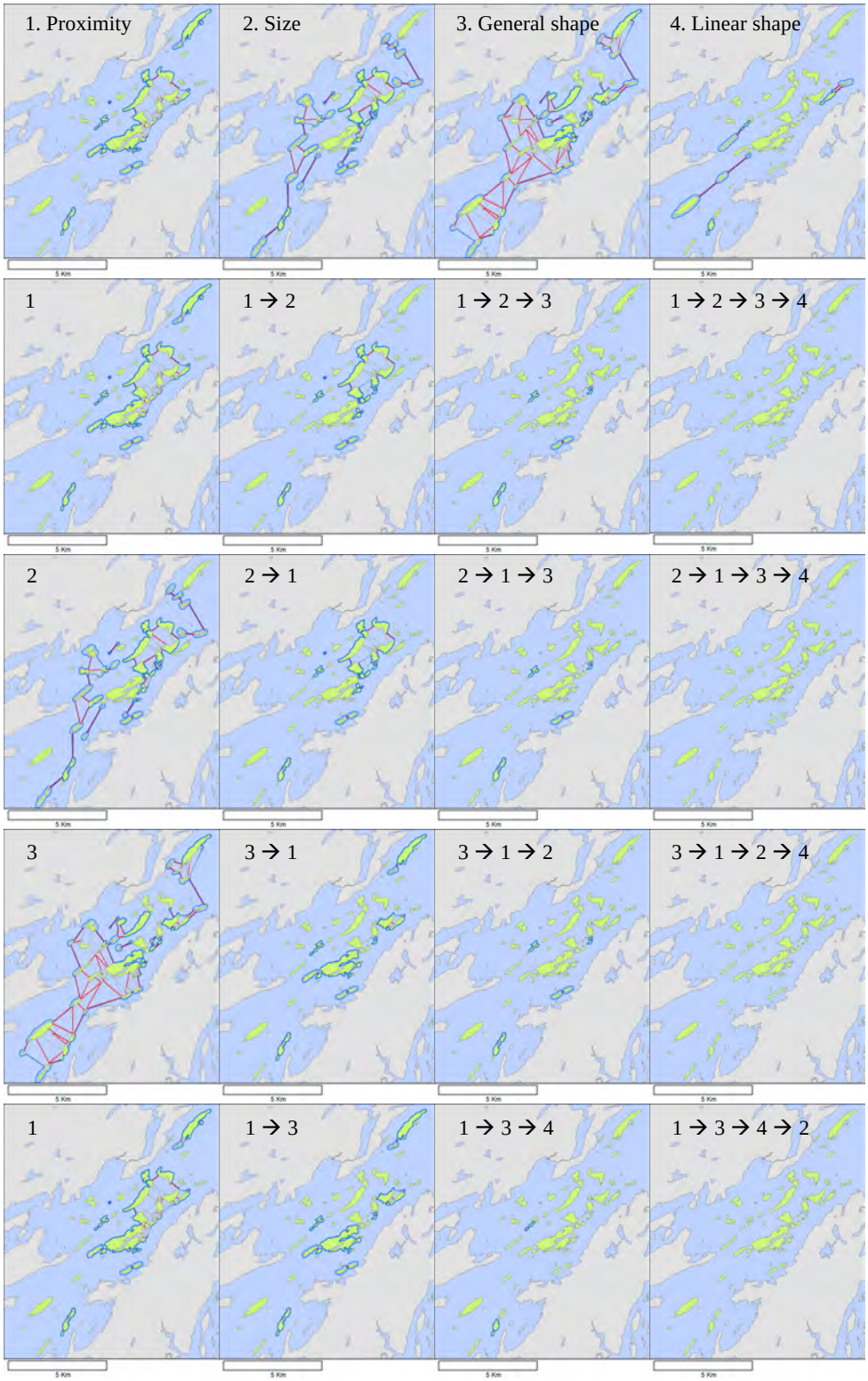


**Figure 4.3: Sequential application of parameters on TA1**
**Top row contains island groups after the individual application of each parameter. The other rows show each parameter being applied in sequence according to each of the four configurations.**

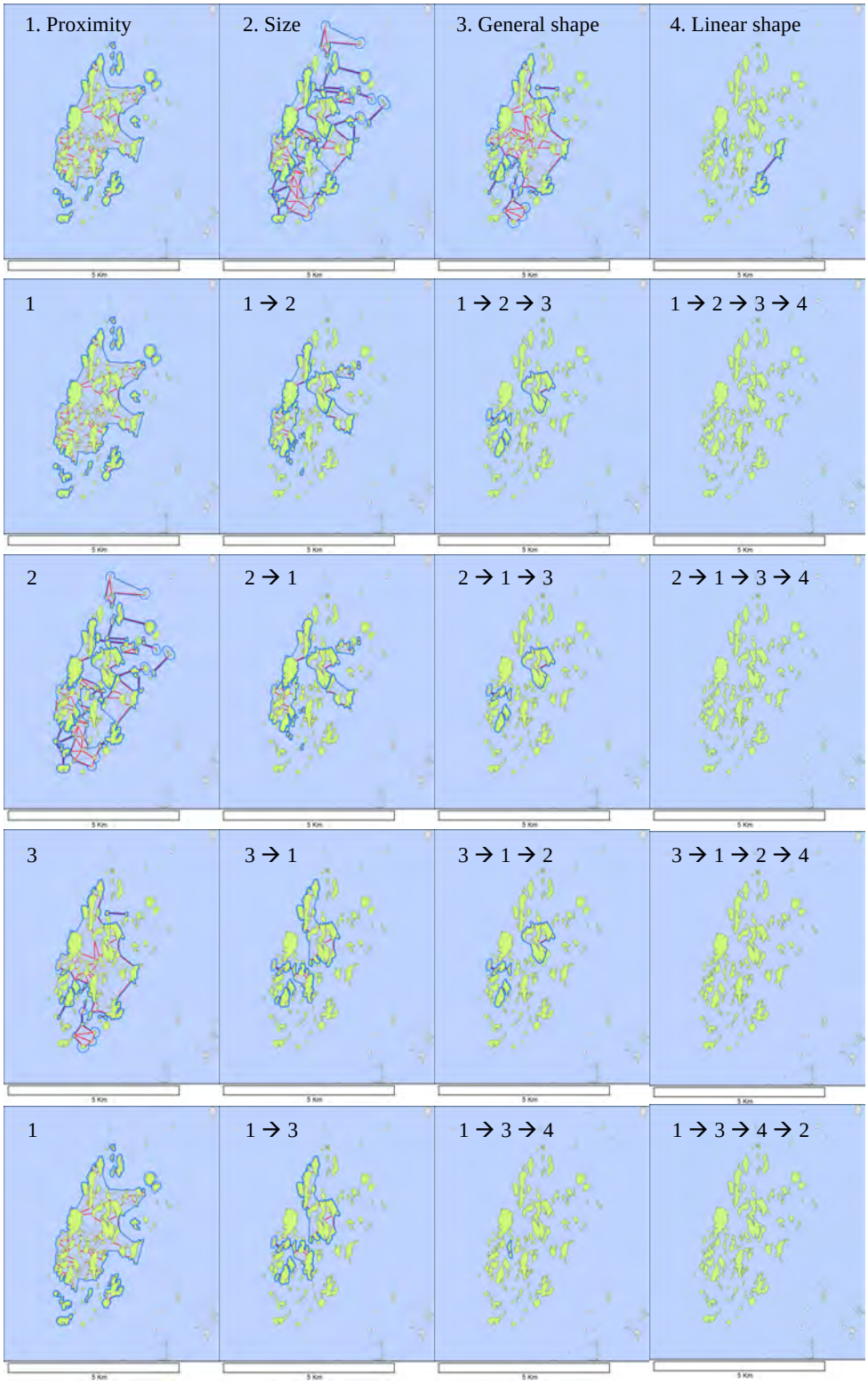


**Figure 4.4: Sequential application of parameters on TA2**
**Top row contains island groups after the individual application of each parameter. The other rows show each parameter being applied in sequence according to each of the four configurations.**

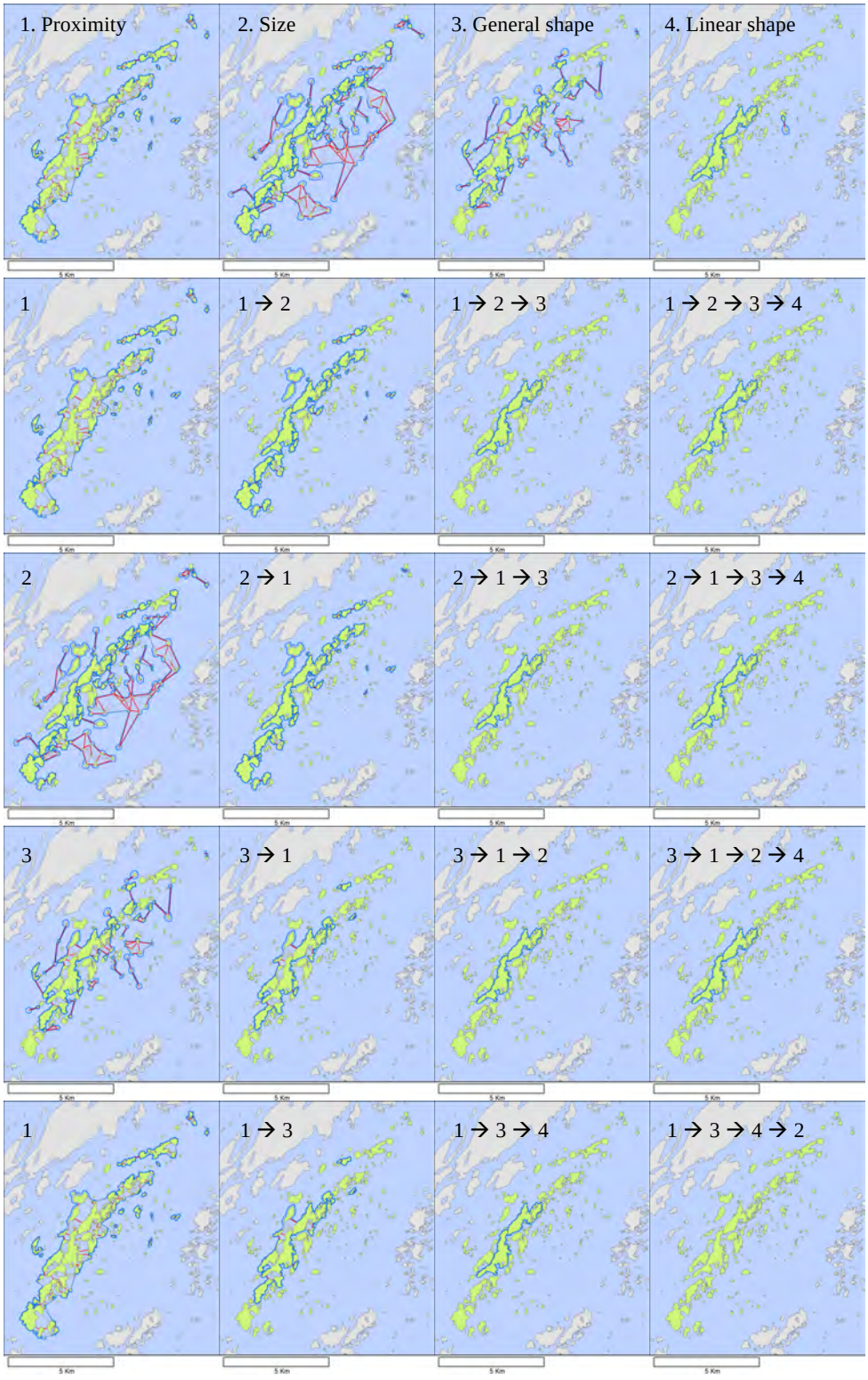


**Figure 4.5: Sequential application of parameters on TA3**
**Top row contains island groups after the individual application of each parameter. The other rows show each parameter being applied in sequence according to each of the four configurations.**

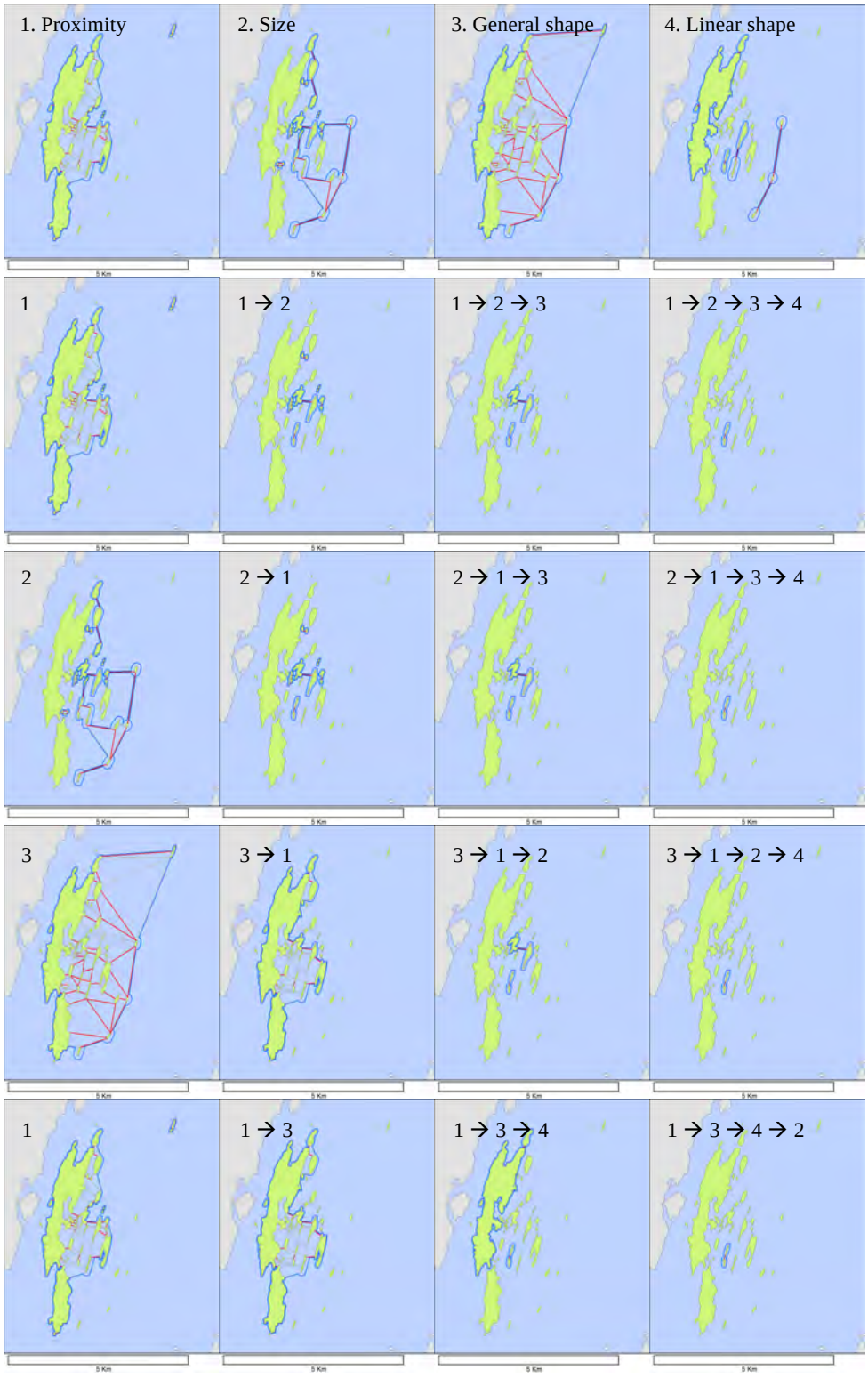


**Figure 4.6: Sequential application of parameters on TA4**
**Top row contains island groups after the individual application of each parameter. The other rows show each parameter being applied in sequence according to each of the four configurations.**

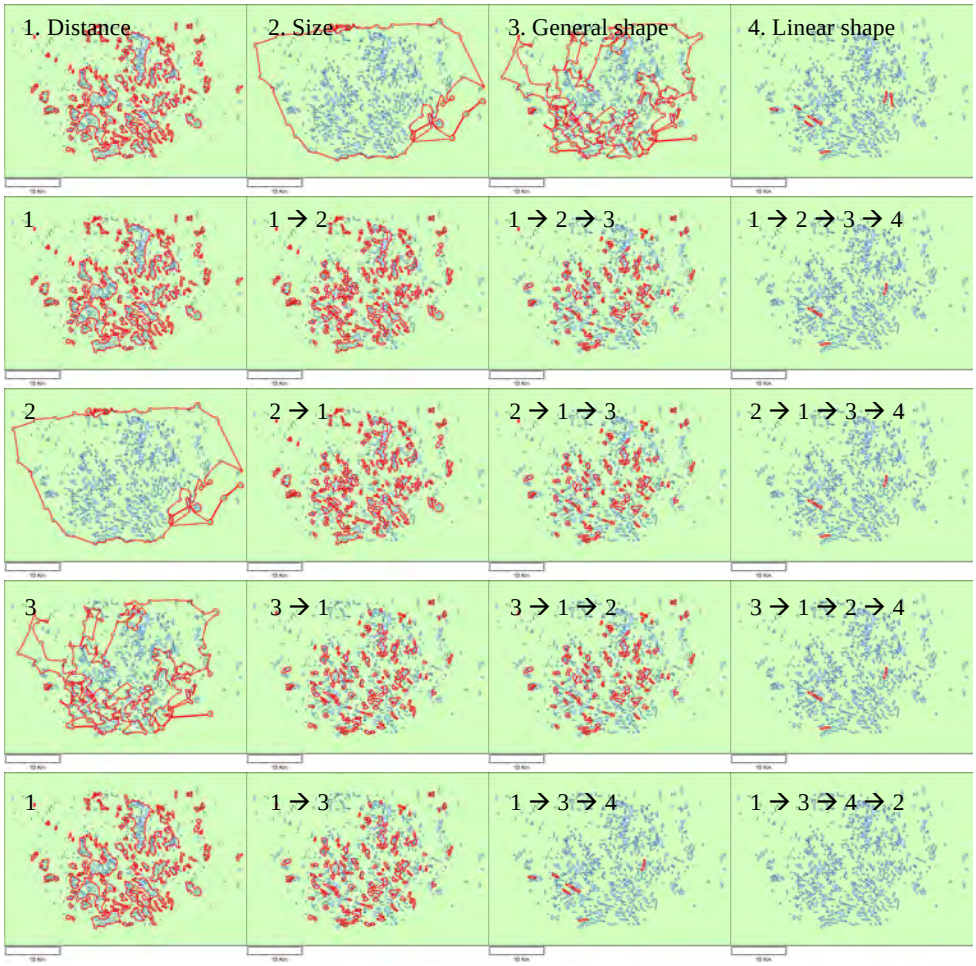


**Figure 4.7: Sequential application of parameters on TA5**
**Top row contains island groups after the individual application of each parameter. The other rows show each parameter being applied in sequence according to each of the four configurations.**

## 4.2 Weighted parameter application

The dendrograms from the weighted parameter application (figures 4.8 and 4.9) show a more balanced process than the graphs of the sequential parameter application, where most of the islands are not separated at the very beginning. The y-axis in these dendrograms to not represent the individual parameters but rather, the combined weight index as applied at each division step. By the lower middle of the graph (higher weight thresholds), most of the islands have been divided into individual clusters, except possibly for the bottom right dendrogram of each figure, where the configuration favours size similarity over distance and shape. In the other configurations, size is the least important parameter. As a result, in the other configurations, islands stay grouped longer if they are close together and there is agreement in shape, but as configuration 4 ranks size and distance top, the groups appear to stay intact for longer because there is a greater general size and proximity agreement across the areas than shape agreement. This is expected, especially with consideration of the linear shape parameter, which generates groups with a very low population. As seen in the rightmost images of the top rows of figures 4.3, 4.4, 4.5, 4.6 and 4.7, these linear shape clusters are not particularly large and when a higher weight is applied to it than other parameters, the groups will split more quickly when raising the weight value applied to each connection.

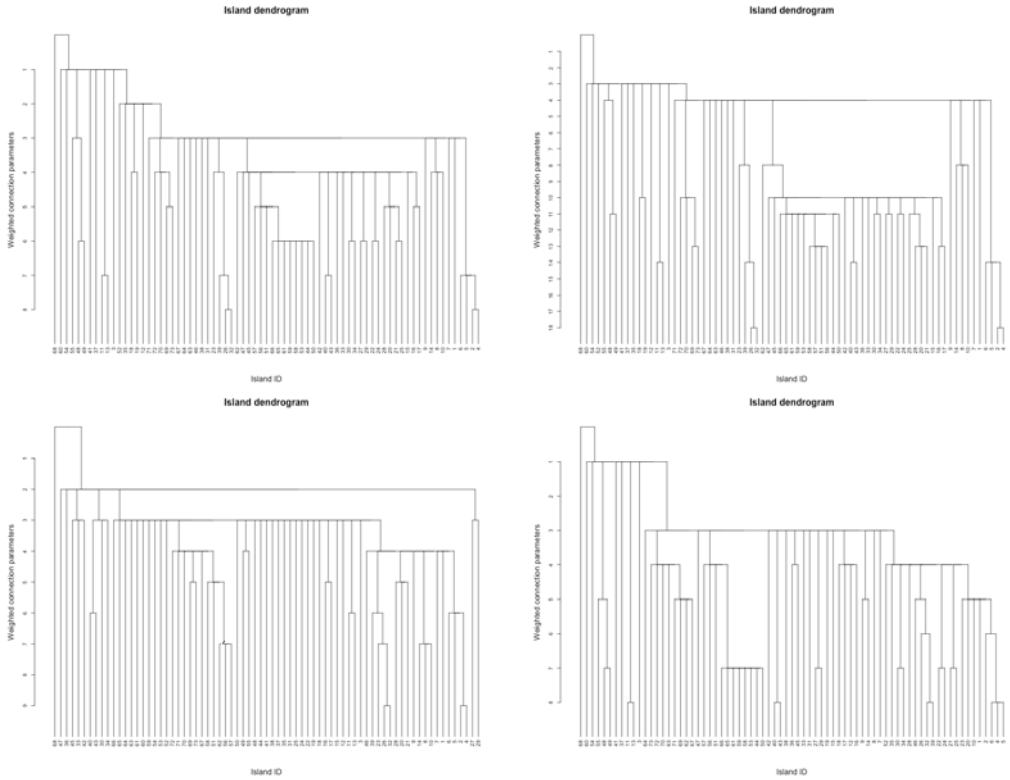


**Figure 4.8: Dendrograms of the weighted parameter applications on TA1**
**top left: configuration W1, top right: configuration W2**
**bottom left: configuration W3, bottom right: configuration W4**

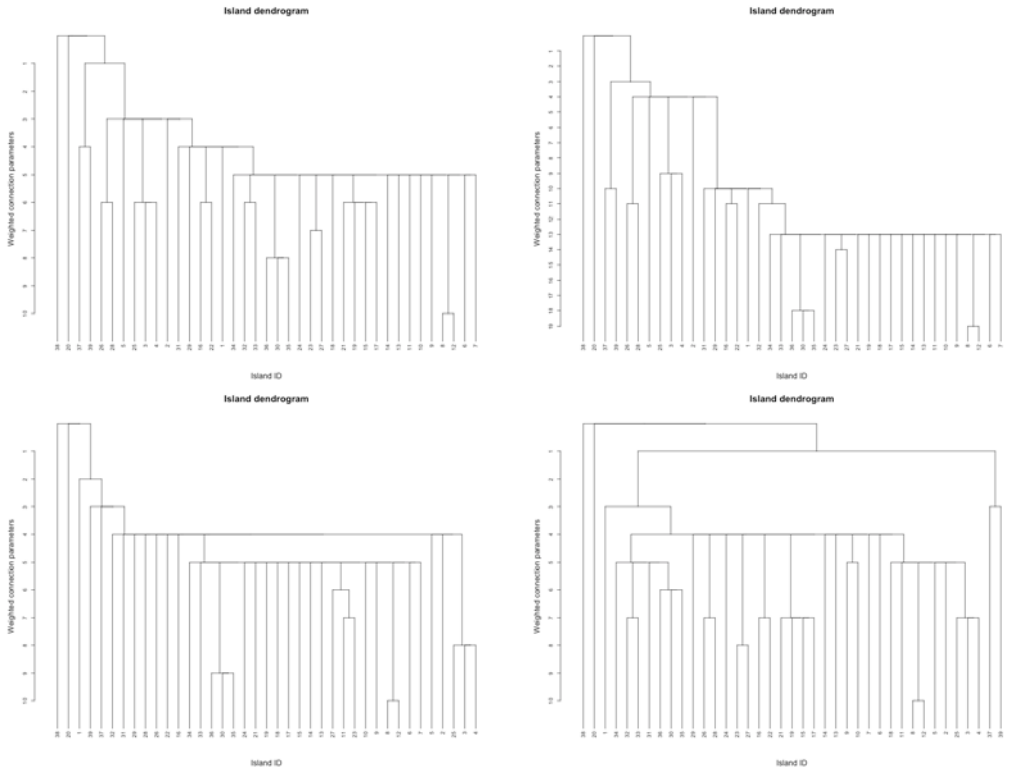


**Figure 4.9: Dendrograms of the weighted parameter applications on TA4**
**top left: configuration W1, top right: configuration W2**
**bottom left: configuration W3, bottom right: configuration W4**

In figures 4.10-4.29 on the following 9 pages, the grouping is displayed on maps. In all configurations, it is common that in the first few steps, the groups are still very big, with splits only taking place where adjacent islands have almost nothing in common. By the end, the high weight indices mean that most or all of the parameter requirements must be fulfilled in order to join islands. As a result, as in the last two steps of the sequential parameter application, the groups have become so small that there hardly are any meaningful groups. The most meaningful results are therefore near the middle in each map sequence. In these middle steps, the possibility is greatest for the parameters to work together in combination, without any single one having to be strictly applied, as in the sequential parameter application.

Configurations W1 and W2 have the same order, where proximity is the most important parameter, then linear shape and general shape and lastly size. The difference is the exact weight applied to each parameter. In configuration W1, they are simply given weight according to their order, but in configuration W2, the weights are further apart, increasing the proportion of the difference between the parameters. In the maps, this can be seen where groups exist that either have a shape agreement or are close together, but in configuration W2 the shape agreement groups disappear sooner as the weight number increases. Proximity is considered the most important parameter by which to group polygons, and this is reflected in the resulting groups here. In the most meaningful groups near the middle stages in configurations W1 and W2, the groups are quite compact, with longer-lasting relations if they are also linear. For most test areas, configuration W2 results in more logical groups than configuration W1, as the groups split more readily once the proximity parameter is required. Configuration W3 favours shape, especially linear shape, over proximity and size. In most test areas, the resulting groups from configuration W3 do not seem very logical, with the exception of TA1 and TA4, but this is to be expected because TA2 and TA3 consist of a large number of heterogeneous islands with no clear line patterns between island pairs (although arguably the archipelago in TA3 is distributed along a prevailing direction, thereby forming a potential line on a macro-level). In TA1 and TA4, there seem to be several islands that exhibit a straight linear pattern, and it becomes meaningful to retain these when splitting the groups. Configuration W4 has the least logical groups as it favours size over proximity and shape, and the resulting groups are rather fragmented.

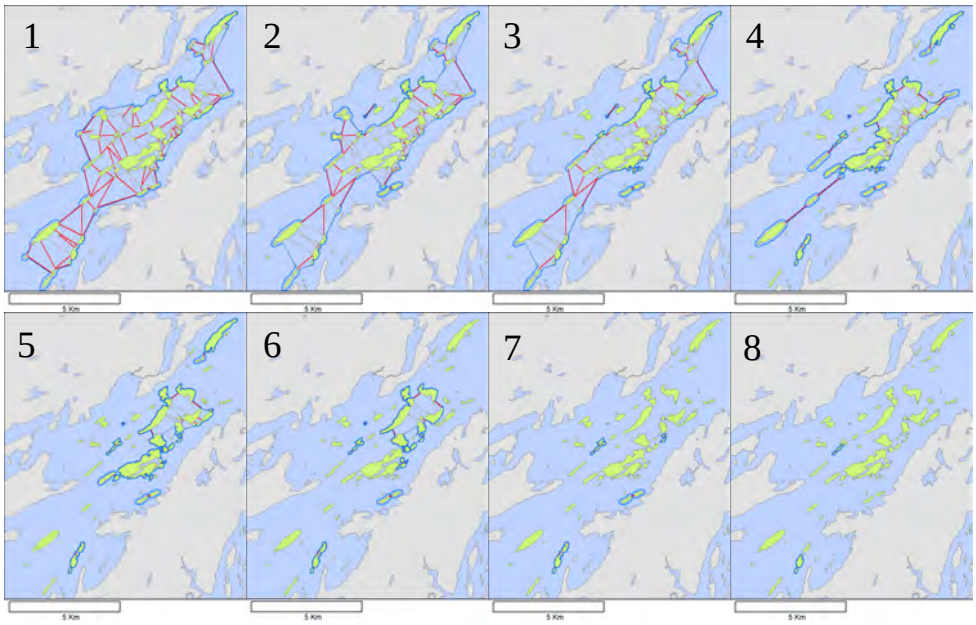


**Figure 4.10: Application of weighted parameters on TA1 using configuration W1.**

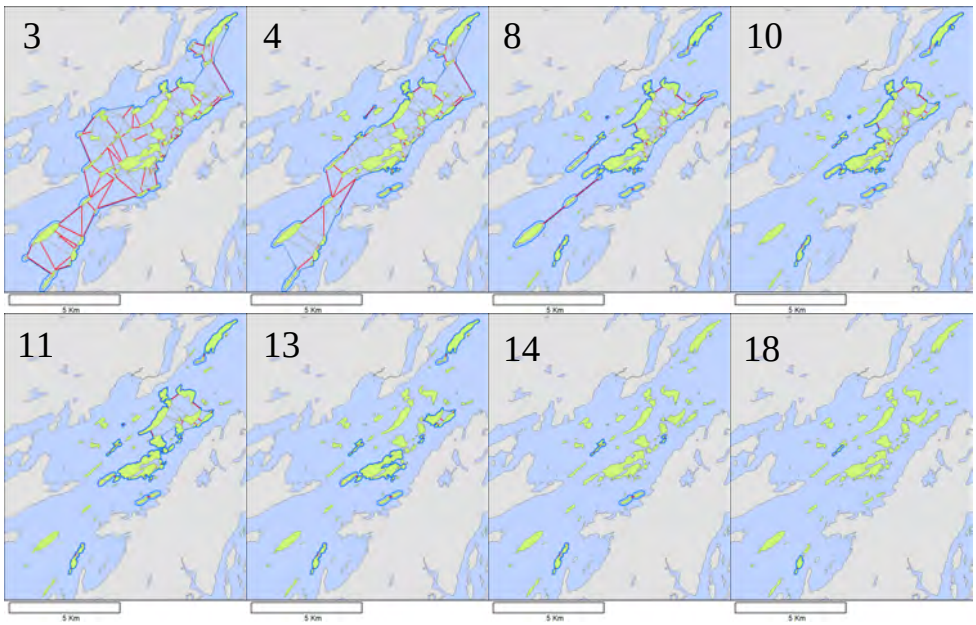


**Figure 4.11: Application of weighted parameters on TA1 using configuration W2**

First two configurations applied on test area 1. The parameter weights in these configurations are:

W1.

- Proximity = 4
- Linear shape = 3
- General shape = 2
- Size = 1

W2.

- Proximity = 10
- Linear shape = 5
- General shape = 3
- Size = 1

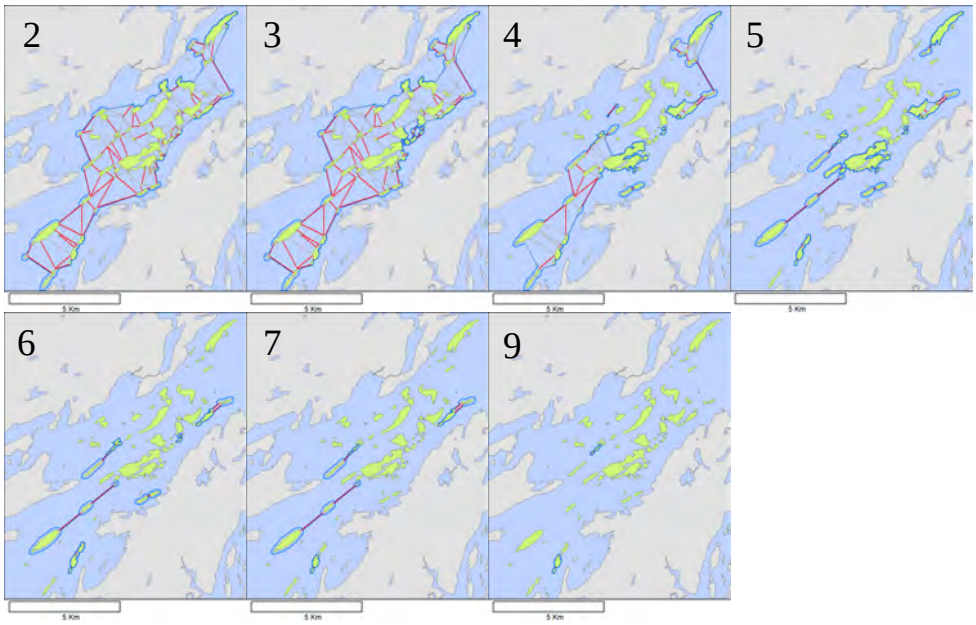


**Figure 4.12: Application of weighted parameters on TA1 using configuration W3**

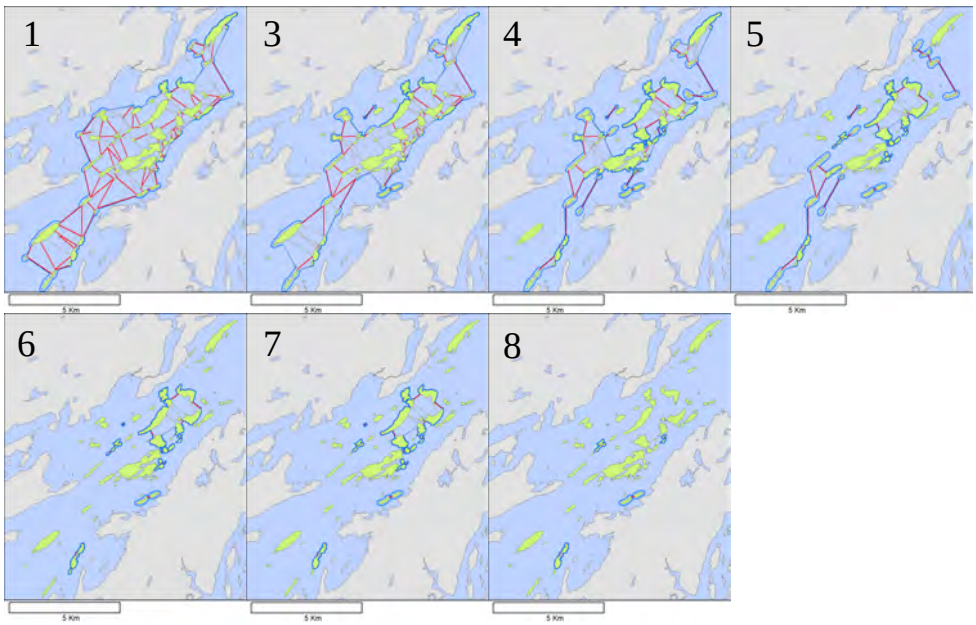


**Figure 4.13: Application of weighted parameters on TA1 using configuration W4**

Latter two configurations applied on test area 1. The parameter weights in these configurations are:

W3.

Proximity = 2

Linear shape = 4

General shape = 3

Size = 1

W4.

Proximity = 3

Linear shape = 2

General shape = 1

Size = 4

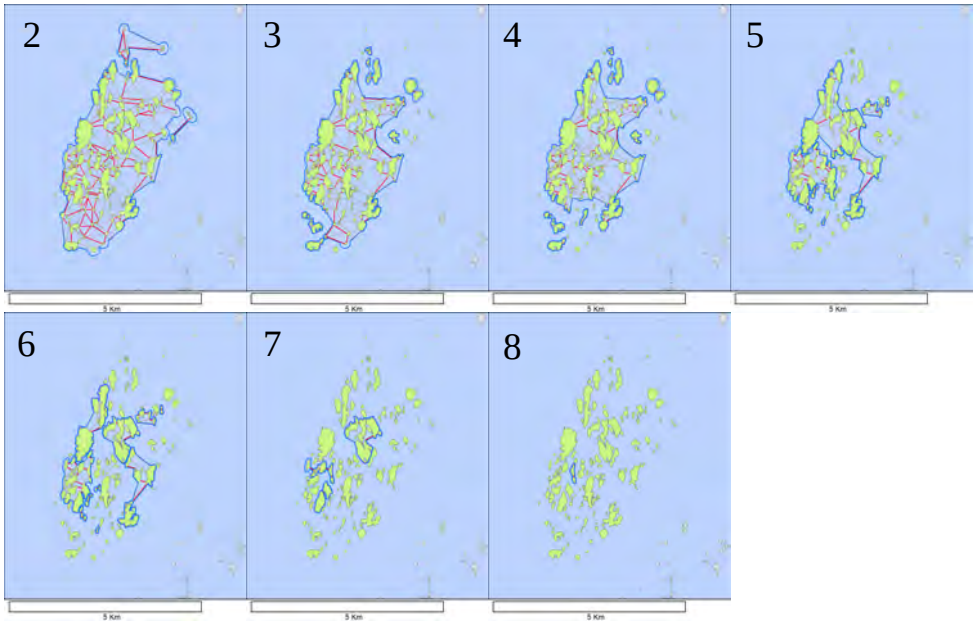


**Figure 4.14: Application of weighted parameters on TA2 using configuration W1**

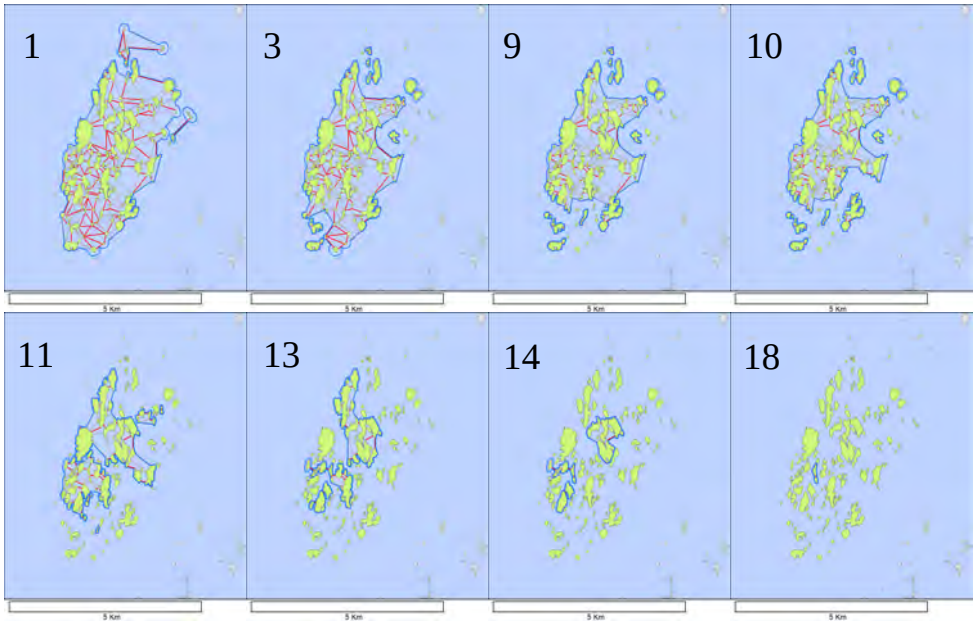


**Figure 4.15: Application of weighted parameters on TA2 using configuration W2**

First two configurations applied on test area 2. The parameter weights in these configurations are:

W1.

Proximity = 4

Linear shape = 3

General shape = 2

Size = 1

W2.

Proximity = 10

Linear shape = 5

General shape = 3

Size = 1

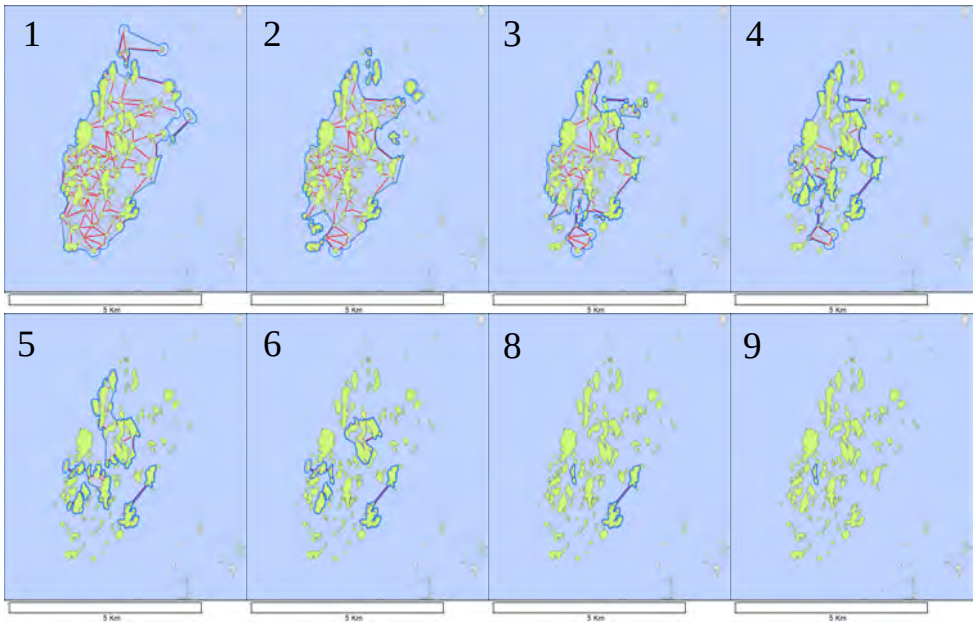


**Figure 4.16: Application of weighted parameters on TA2 using configuration W3**

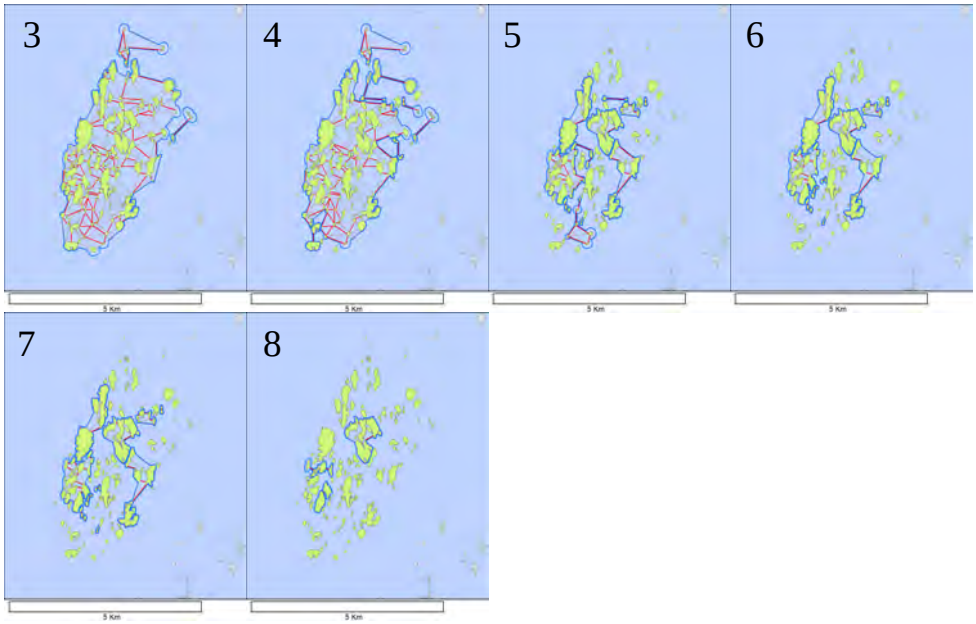


**Figure 4.17: Application of weighted parameters on TA2 using configuration W4**

Latter two configurations applied on test area 2. The parameter weights in these configurations are:

W3.

Proximity = 2

Linear shape = 4

General shape = 3

Size = 1

W4.

Proximity = 3

Linear shape = 2

General shape = 1

Size = 4

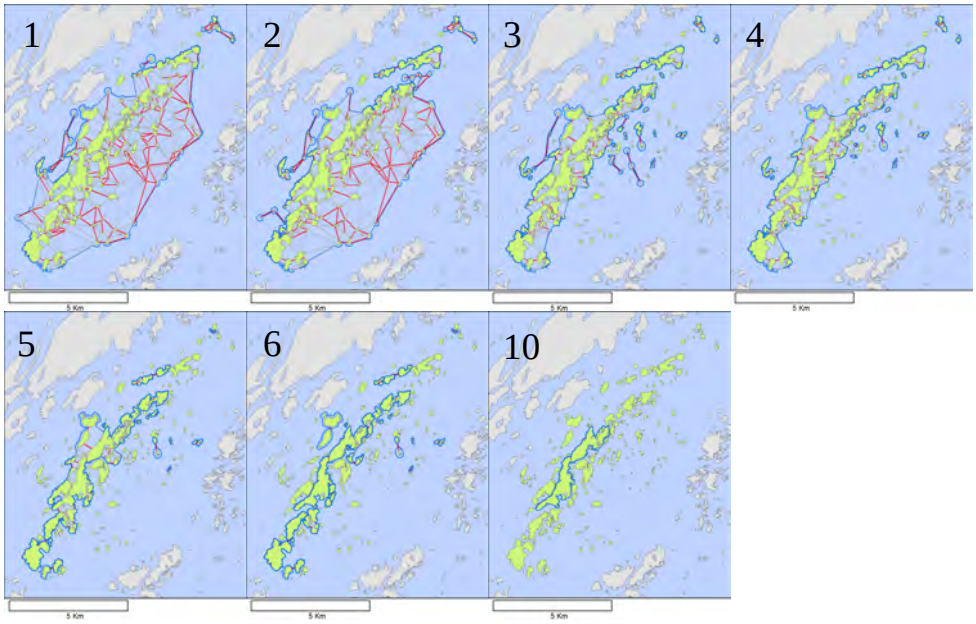


**Figure 4.18: Application of weighted parameters on TA3 using configuration W1**

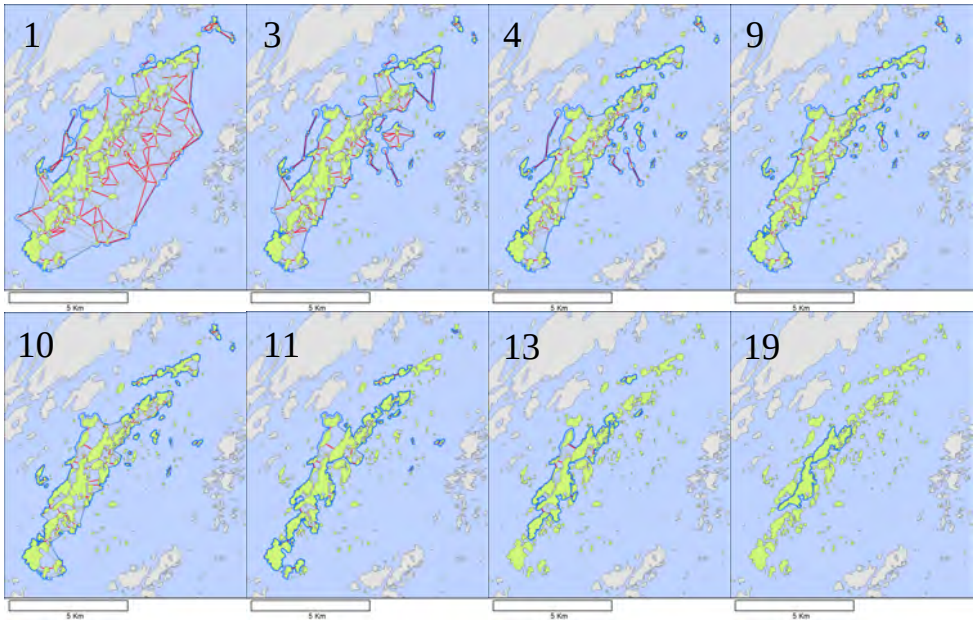


**Figure 4.19: Application of weighted parameters on TA3 using configuration W2**

First two configurations applied on test area 3. The parameter weights in these configurations are:

W1.

Proximity = 4

Linear shape = 3

General shape = 2

Size = 1

W2.

Proximity = 10

Linear shape = 5

General shape = 3

Size = 1

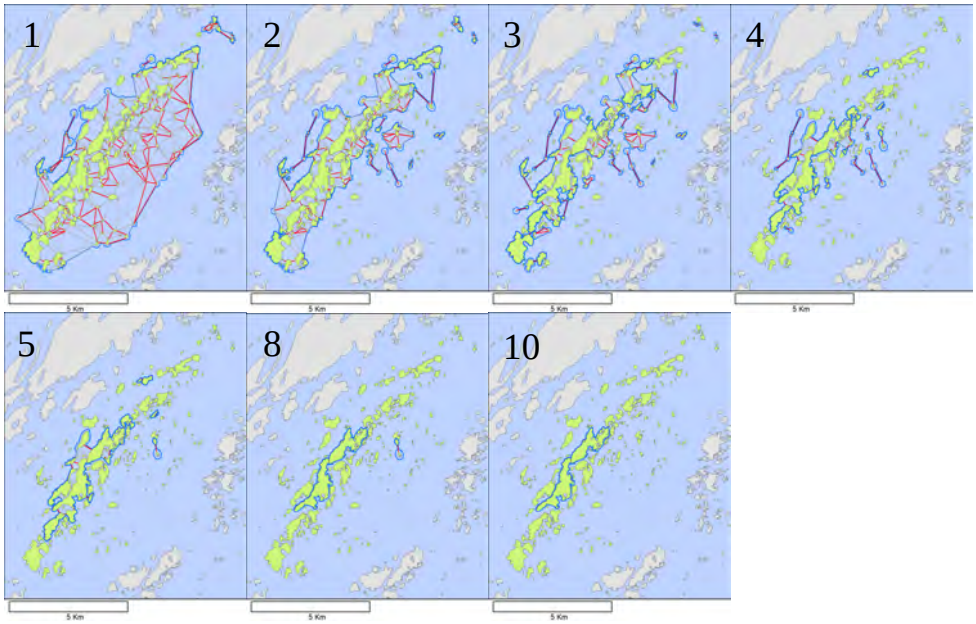


**Figure 4.20: Application of weighted parameters on TA3 using configuration W3**

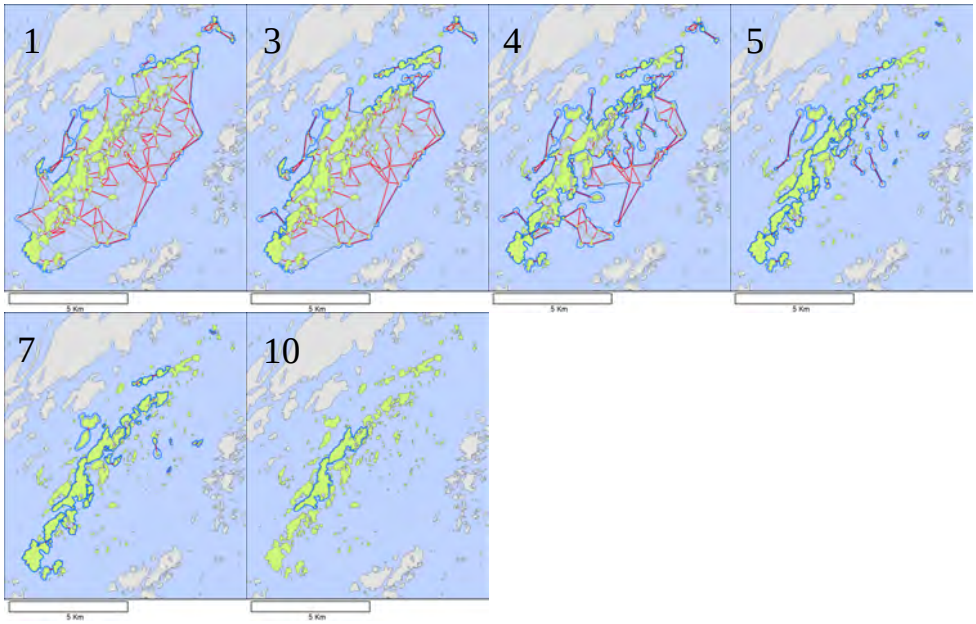


**Figure 4.21: Application of weighted parameters on TA3 using configuration W4**

Latter two configurations applied on test area 3. The parameter weights in these configurations are:

W3.

Proximity = 2

Linear shape = 4

General shape = 3

Size = 1

W4.

Proximity = 3

Linear shape = 2

General shape = 1

Size = 4

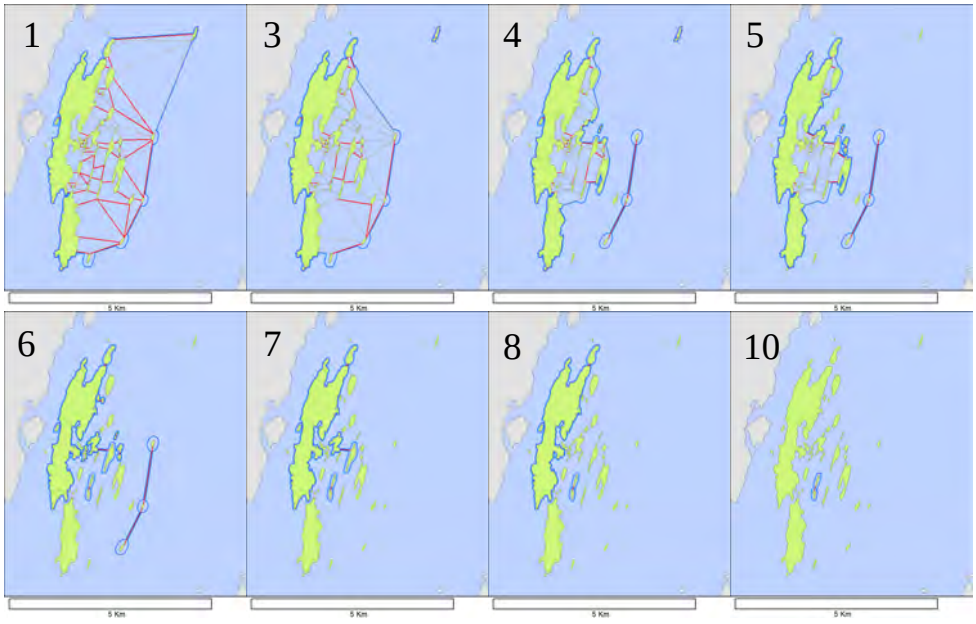


**Figure 4.22: Application of weighted parameters on TA4 using configuration W1**


**Figure 4.23: Application of weighted parameters on TA4 using configuration W2**

First two configurations applied on test area 4. The parameter weights in these configurations are:

W1.

Proximity = 4

Linear shape = 3

General shape = 2

Size = 1

W2.

Proximity = 10

Linear shape = 5

General shape = 3

Size = 1


**Figure 4.24: Application of weighted parameters on TA4 using configuration W3**

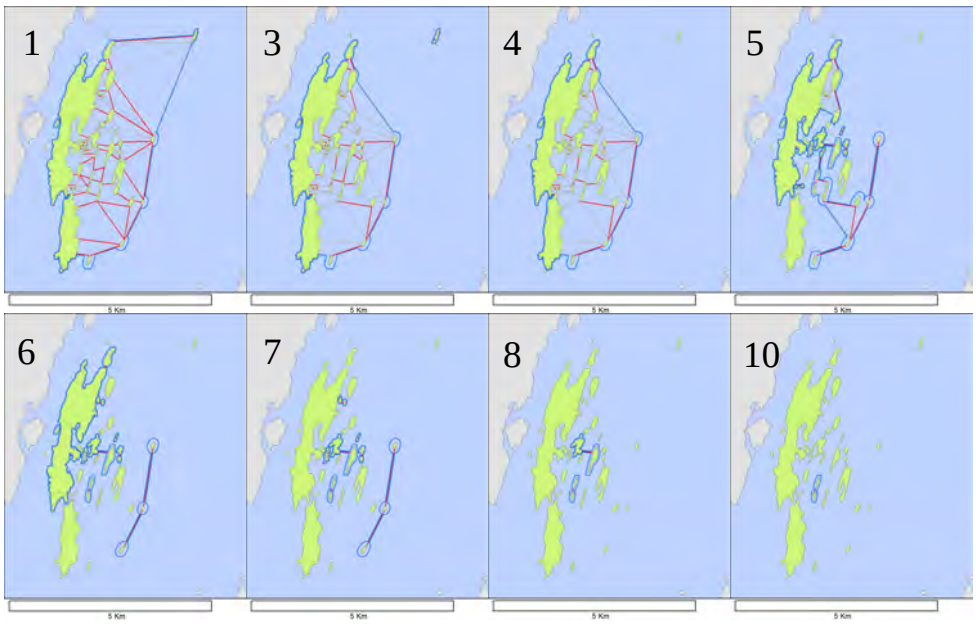


**Figure 4.25: Application of weighted parameters on TA4 using configuration W4**

Latter two configurations applied on test area 4. The parameter weights in these configurations are:

W3.

Proximity = 2

Linear shape = 4

General shape = 3

Size = 1

W4.

Proximity = 3

Linear shape = 2

General shape = 1

Size = 4

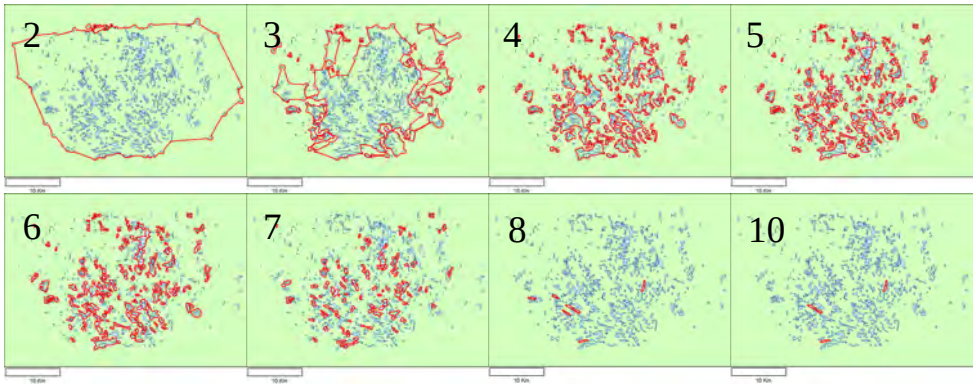


**Figure 4.26: Application of weighted parameters on TA5, using configuration W1**

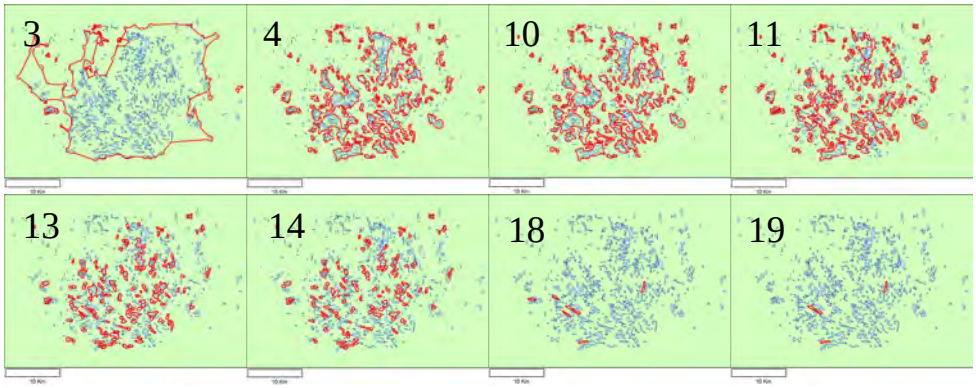


**Figure 4.27: Application of weighted parameters on TA5, using configuration W2**

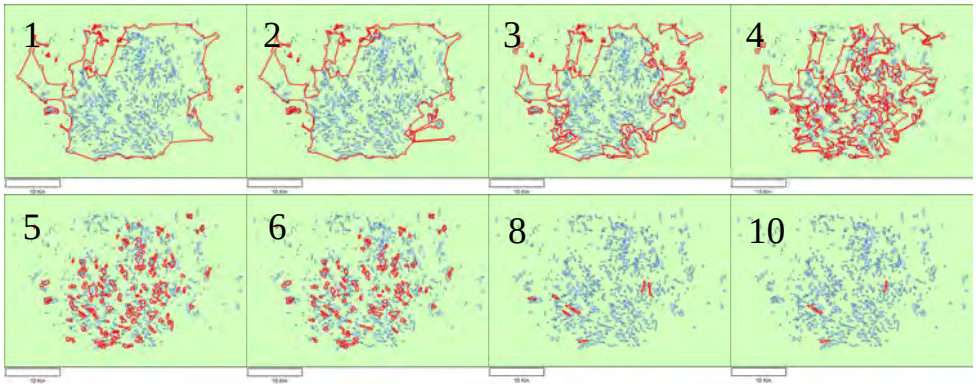


**Figure 4.28: Application of weighted parameters on TA5, using configuration W3**

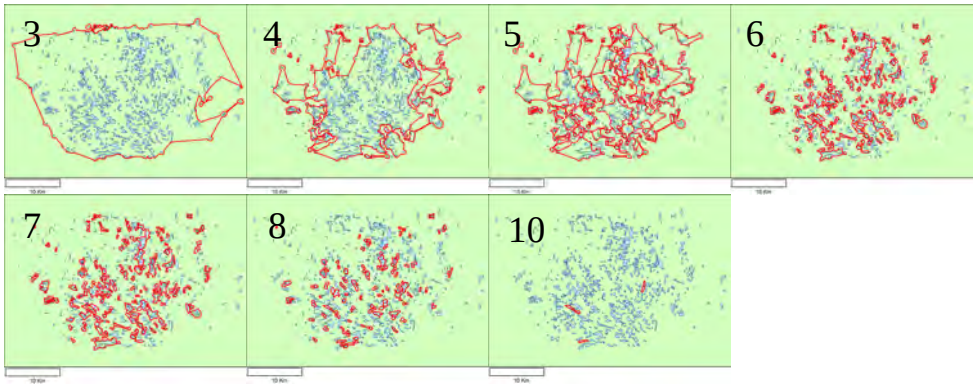


**Figure 4.29: Application of weighted parameters on TA5, using configuration W4**

# 5 Discussion

This chapter discusses first the implication of the results and what they mean for pattern recognition for those particular island groups, before moving on to what they mean for the evaluation of the group detection algorithm and thoughts about future work and improvements.

## 5.1 Implications of results

The results show that this algorithm can be useful in identifying island groups. However, not all application variants result in logical groups. Proximity is by far the most important parameter to keep in mind for grouping of islands, but other parameters, especially shape and linear shape can be useful for areas where there are strong linear patterns and thus these lines become almost as important as proximity for island grouping. The algorithm was not very useful for detection of meso-structures in test areas where island pairs exhibited no clear patterns that would override proximity in importance, but in other cases, such as in groups where there were discernible line patterns, the algorithm could successfully identify these and in combination with the proximity parameter, provide good looking island groups.

The algorithm was a success with regard to the individual parameters, inasmuch as each parameter was able to do what was expected of it to a large extent. Islands close together would be grouped by proximity, islands with a similar direction or similar circular shape would be grouped, likewise similarly sized islands, despite being far apart, and of course, islands that form a relatively straight line. There was some success in the application of the parameters in combination, either sequential or by use of a weight index, but the most meaningful results seemed to appear during the application of the proximity parameter in general, and of the linear shape parameter in groups with a propensity for clear straight line patterns. Since proximity is the most important individual parameter by which to group islands or other polygonal features, a comparison can be made between the island groups arising from the proximity parameter working alone and some select outputs of different configurations using the weighted parameter application.

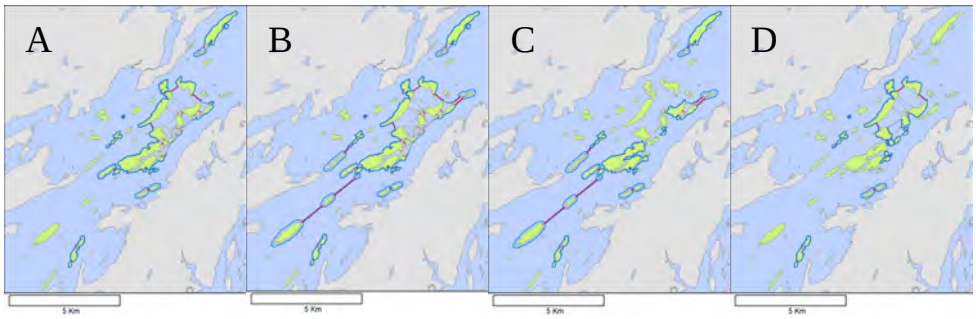


**Figure 5.1: Comparison of select grouping results for TA1**
**A) Proximity parameter only, B) Configuration W1 – weight 4**
**C) Configuration W3 – weight 5, D) Configuration W4 – weight 6**

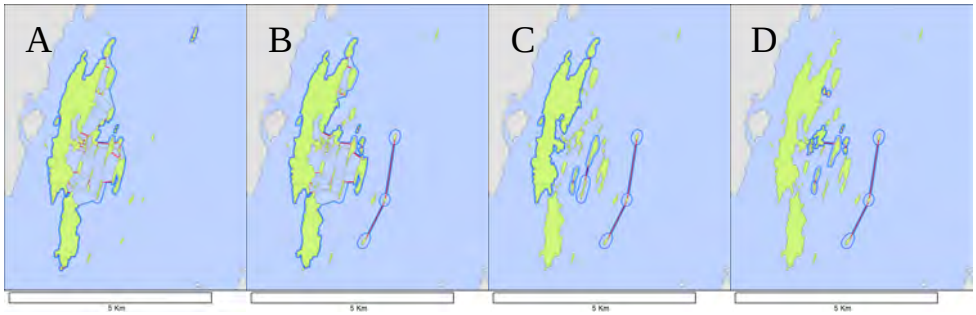


**Figure 5.2: Comparison of select grouping results for TA4**
**A) Proximity parameter only, B) Configuration W1 – weight 5**
**C) Configuration W3 – weight 7, D) Configuration W4 – weight 7**

Figures 5.1 and 5.2 show test areas 1 and 4 respectively, where some linear shape structures can be observed. In image A in both figures, only the proximity parameter has been applied to the island groups. In test areas 1 and 4, some interesting groups appear when applying the linear shape parameter in combination with proximity (image B) or with linear shape as the most important parameter (image C). Image D in both figures is representative of the 4$^{th}$ configuration where size takes precedence over proximity and shape, and the end result is not as good. Here it appears meaningful to extract the linear shapes and not only the proximity clusters for the grouping of islands.

Figures 5.3, 5.4 and 5.5 show test areas 2, 3 and 5 respectively. The features in these areas have much less prominent features that form continuing lines than test areas 1 and 4. Furthermore, there do not seem to be particular property commonalities that make an elaborate grouping algorithm particularly meaningful in comparison to the proximity parameter, such as size or general shape. As a result, the proximity parameter yields results that look more logical than the results of the weighted parameter applications by simply being compact. Some results from the weighted parameter applications shown in figures in chapter **4** do also look compact and relatively good,

resembling the output of the proximity parameter, but that puts into question the need to develop more elaborate parameters for this island clustering than the proximity grouping alone. Some linear structures seem to be correctly identified in test area 5 when assigning more importance to the linear shape parameter (figure 5.5C), but there are some linear shapes – especially linear arrangements of polygons – that are not identified and put into groups.

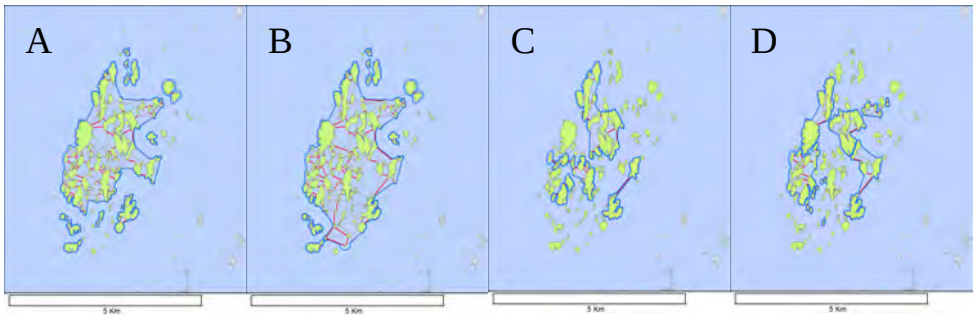


**Figure 5.3: Comparison of select grouping results for TA2**
**A) Proximity parameter only, B) Configuration W1 – weight 3**
**C) Configuration W3 – weight 5, D) Configuration W4 – weight 7**

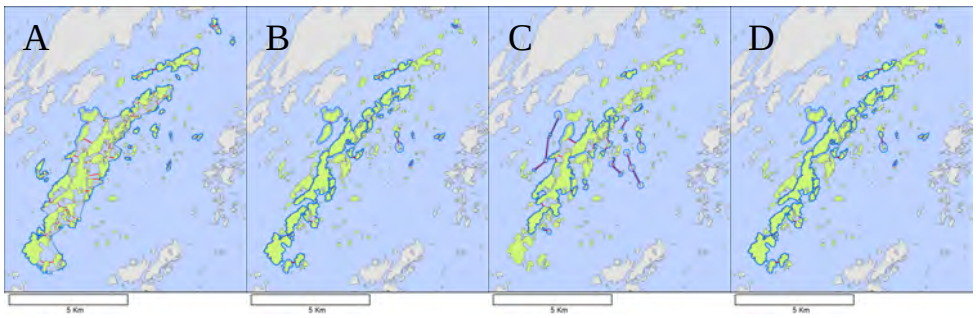


**Figure 5.4: Comparison of select grouping results for TA3**
**A) Proximity parameter only, B) Configuration W1 – weight 6**
**C) Configuration W3 – weight 4, D) Configuration W4 – weight 7**

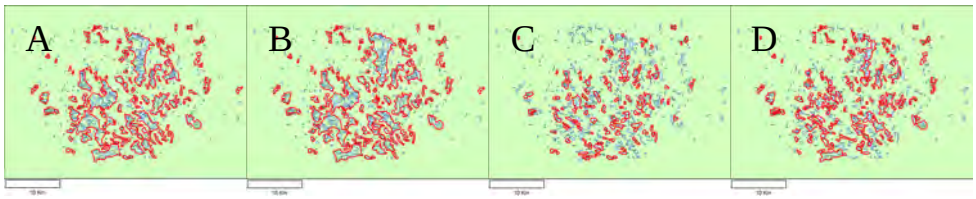


**Figure 5.5: Comparison of select grouping results for TA5**
**A) Proximity parameter only, B) Configuration W1 – weight 4**
**C) Configuration W3 – weight 5, D) Configuration W4 – weight 7**

On the whole, it looks as if this approach to island grouping is partially successful, especially with regard to proximity clustering and linear clustering in areas where these patterns are very clear. The sequential pattern application did not yield

particularly useful results except for when either the proximity or linear shape parameters were applied alone. The application of the weighted parameters worked better, as it did not suffer the main problem of the sequential parameter application, i.e. when most of the parameters contribute to the group division, almost all the clusters have disappeared. In the weighted parameter application, there were some meaningful and interesting island groups, but this depended on the configuration of the weights applied to the parameters.

The algorithm is not written to identify patterns on a macro-level, i.e. prevailing direction patterns throughout a whole map. That could be a useful addition to make to the algorithm, e.g. in the case of the Dombes lake database (test area 5) where the polygons are not necessarily shaped so that the parameters developed here recognise them as belonging together as a line, but they are distributed in such a way that can be perceived as a linear pattern. This would need further consideration of formalisation of Gestalt laws into algorithmic pattern recognition.

Despite the added complexity in developing a general shape and linear shape parameters instead of shape and orientation parameters, and despite the explanation of the former being somewhat complicated than the latter, it appears that the shape parameters, especially linear shape, have been more valuable than simpler parameters would have been in cases where there are patterns of continuing straight lines in the polygon groups.

## 5.2 Suggested calibrations and changes

Some changes that could prove useful in the improvement and development of the process outlined in this paper include use of different tools, i.e. software and coding platforms. The choice to use ArcGIS software was mainly due to convenience of Python scripting and GIS visualisation, but some bottlenecks in computer performance, especially when grouping became complicated. The slowness of group visualisation serves as a motivation to use other platforms and perhaps make development quicker and easier. Some processes and functions in the algorithm can doubtlessly also be optimised to take less time. The island detection algorithm here proved only to be suitable for island groups of relatively few islands, preferably fewer than 100. Larger island groups could be processed, but the time requirements grew to highly impractical

lengths for a home computer environment – several hours or more, depending on how large the dataset was.

For general improvement of the algorithm in terms of structure recognition and island group detection, addition of certain parameters, e.g. a parameter that identifies linear patterns in spaces between polygons and not just inside them, could prove valuable. Also, a detection of prevailing macro-structures in maps would be useful. One parameter that was alluded to in the methodology was the linear distribution parameter. In the parameter development in this paper, with the particular grouping strategy as was used here, linear distribution as a parameter only manifested as a part of the linear shape parameter, only checking island pairs and then uniting all pairs that exhibited such linear structure. For a more appropriate use of such a parameter, a different grouping strategy would be needed. Lines, be they straight or curved, consist of multiple nodes that lie in a linear arrangement. When extracting such a line from the node network, the formation of the line can reach the point where two (or more) new nodes can effectively become a part of the line, but these two nodes between them cannot be together in a single linear structure. Therefore the new line would have to duplicate to accommodate each node. This would require an overlapping clustering method since nodes can be members of multiple lines.

The process in this paper furthermore does have some flaws apart from software limitations or lack of parameters. One definition that could be changed is that of island adjacency (see chapter **3.6**). Instead of determining adjacency by a Thiessen polygon analysis, islands could be tested against every other island or at least by some criterion that is more inclusive than the one used here. This coastline-to-coastline connection adjacency used here has the drawback that islands, even if they are small in size, can disrupt identifiable patterns in shape or size by being situated between islands that otherwise would be grouped together. Also, the variables in this paper were the different island groups that were used to test the group detection process and the different configurations of how to order the parameters and assign importance to them. Different definitions of more basic values, such as angle difference tolerance thresholds or size ratio thresholds, were not systematically tested, although they were somewhat calibrated during development of the algorithm.

In this process, many small and large decisions had to be made as to which routes to take in the design of the algorithm. Seeing as the purpose of such an algorithm is to teach a computer to recognise structures as seen by the map-reader, these decisions were often based on a visual inspection. This includes the decisions for the angle thresholds which determine what constitutes a straight linear relationship or a similarly oriented lines, the shape index threshold which determines when a polygon can be considered to resemble a circle and when it cannot, the ratio when comparing the size of two polygons, the threshold of the allowable distance between two polygons and the ratio for the narrowness of a polygon part when extracting the island skeletons. Where there was no theoretical background to assist or a limited possibility of a visual determination, decisions were made for as straightforward values as could be estimated, such as the size ratio of and the narrowness ratio of 1:2, the angle difference ratio of 45° for possible direction agreement of skeleton edges and branches, 30° for a direction alignment between islands, and 15° for a straight line continuation of islands, or half the size of a right angle and fractions thereof. In further research into this topic, such thresholds can hopefully be more elaborately calibrated according to research into pattern recognition, or made more dynamic, similarly to the distance threshold used here, which calculates a different threshold based on the length of each polygon, instead of having a fixed threshold throughout the polygon group.

Of course, with the addition of more variables in such a process, the number of outcomes increases greatly, so while the decision was taken here to use a fixed set of threshold definitions in order not to explode the number of results, these could most probably be calibrated better.

## 5.3 Context of process in generalisation research

The development of strategy and use of constraints, whether strict or flexible, is essential in the computerisation, let alone full automation of generalisation. It is a possible problem that in order to have the full process of generalisation and the steps required to apply it, many specialised algorithms are necessary and it is another potential problem when those algorithms are focused on a specific problem or feature with a large deal of abstraction and do not take into account the holistic generalisation needs. Computers have been programmed to perform amazing feats and learn and respond to complex situations that were previously unthinkable for anything but a

human, and thus it seems like short-sightedness to claim that a full generalisation algorithm that fulfils the requirements of a human cartographer can never be developed. Nor is it clear however whether there is a programmable solution to the generalisation problem, seeing as creativity and art make up a large part of the cartographer's work and his/her subjective cartographic decisions. Therefore, perhaps the problem can in the foreseeable future at best be semi-automated, where algorithms for the different steps of the generalisation process are used as tools for the cartographer, who can at any point accept, reject or modify the result proposed to him/her by the algorithm, saving a large amount of manual work, but not replacing all of the work.

This paper is – in contrast to the ideas of holistic approaches to generalisation as described by McMaster and Shea (1992) – rather an example of an isolated-issue bottom-up approach to automated generalisation. The focus in this paper is only on island groups, and specifically the detection thereof and modelling of their structural relations, and as a result, the islands were treated as semantically homogenous objects where the outline geometry is the only concern (as opposed to e.g. semantic properties).

The work completed here is done under the assumption that while generalisation may not be fully automated without the creative input of the cartographer in the near future, or maybe ever, aides to automatic generalisation can indeed be developed to at best semi-automate the process. McMaster and Shea did mention – despite their insistence on the need for a holistic approach – that individual prototype algorithms for the various components of the generalisation process may be necessary. As a result, despite their argumentation seeming to contradict the basic premise of an undertaking such as the one done in this paper, this contradiction is not absolute. Müller and Wang (1992) also focused on two-dimensional, semantically identical area patches, and this approach, while not being a holistic approach to generalisation, arguably has great value in the research of automated generalisation. As Müller and Wang suggest about their process, this process could at least be useful in the case of generalisation of natural features such as forests, lakes or islands.

A process such as this could serve as a component in a wider solution to automatic generalisation by grouping islands into logical groups or to identify which island groups are united by a certain property in order to then decide which generalisation operators to apply to the island group. In generalisation processes such as

the one by Müller and Wang (1992) – see figure 2.2 – it may be largely sufficient to use only proximity, but for automation of operators such as typification, it becomes important to detect the shape and distribution patterns, and during simplification of features that are shaped or distributed along a prevailing direction, it may be useful to detect which islands have a linear relationship. Although it is frequently demonstrated by earlier research and the results in this paper that proximity is by far the most important parameter when it comes to grouping islands, it is reasonable to assume that the recognition of other parameters is significant, especially in the detection of micro-structures. With further development and calibration of these other parameters, they could prove more successful in identifying polygon groups than in this paper.

This process of reverse engineering of perceived patterns has a lot of merit, as seen in some of the results here and in the results of the algorithm created by Steiniger, Burghardt and Weibel (2007). One problem in this paper is that the pattern recognition and mental island grouping was solely performed by the author while engineering the algorithm, whereas a survey of a larger population probably provides a more representative account of what constitutes a pattern or group according to the general map reader. This does not undermine the attempt however, since the purpose here is to generate a process that then can be built on and improved using better knowledge, but in the continuation of this work, for this subject is far from being put to rest, this is one of the aspects to keep in mind.

# 6 Conclusions

The island group detection and explicit modelling relation by use of four primary parameters that check similarity between adjacent island pairs, i.e. proximity, size, general shape and linear shape, is partially successful in extracting logical sub-groups from within island groups. In some cases, the proximity parameter would be enough on its own, especially in groups where islands do not tend to be long, narrow and arranged in lines. A combination of parameters provided promising results, but the process would benefit from addition of macro-structure detection algorithms and possibly a finer calibration of the algorithm. The evaluation of the results are furthermore not automated, requiring human input as to which results could be deemed useful for further use in generalisation.

## References

AGENT Consortium, (1999). Report C1 – Selection of basic measures.
Available at: <http://agent.ign.fr/deliverable/DC1.html> [Accessed 7 October 2015].

Brassel, K. E. and Weibel, R. (1988). A review and conceptual framework of automated map generalization. *International Journal of Geographical Information Systems.* Vol. 2, No. 3 (1988), pp. 229-244.

Eckert, M. (1908). On the Nature of Maps and Map Logic. *Bulletin of the American Geographical Society*. Vol. 40, No. 6 (1908), pp. 344-351.

Harrie, L. and Weibel, R. (2007). Modelling the Overall Process of Generalisation. In Mackaness, W., Ruas, A. and Sarjakoski, L. T. (eds.), *Generalisation of Geographic Information: Cartographic Modelling and Applications*. Elsevier.

Helmfrid, S. (ed.) (1992). *The geography of Sweden. National atlas of Sweden.*
Stockholm: Almqvist and Wiksell International.

Imhof, E. (1982). *Cartographic relief presentation*. Berlin: Walter de Gruyter & co.

McMaster, R. B. and Shea, K. S. (1992). *Generalization in Digital Cartography*. Washington DC: Association of American Geographers.

Müller, J. C. and Wang, Z. (1992). Area-Patch generalisation: a competitive approach. *The Cartographic Journal*. Vol. 29, No. 2 (1992), pp. 137-144.

Regnauld, N. (2005). Spatial structures to support automatic generalisation. In *XXII International Cartographic Conference*, A Coruña, Spain 11-16 July 2005.
Available at: <http://www.cartesia.es/geodoc/icc2005/pdf/oral/TEMA9/Session%202/NICOLAS%20REGNAULD.pdf> [Accessed 13 August 2005]

Sahrhage, D. and Lundbeck, J., (2012). *A History of Fishing*. Springer-Verlag Berlin Heidelberg.

Sjöberg, B. (ed.) (1996). *Sea and coast. National atlas of Sweden.* Stockholm: Almqvist and Wiksell International.

Statistiska Centralbyrån. (2009), *Befolkning på öar, korrigerad 2009-06-18 2000-2008*. Statistiska Centralbyrån, Stockholm. Available at <http://www.scb.se/statistik/MI/MI0812/2008A01/MI0812_2008A01_SM_MI50SM0901.pdf> [Accessed 12 May 2015].

Steiniger, S. Burghardt, D. and Weibel, R. (2007). Recognition of Island Structures for Map Generalization. In Steiniger, S. (2007). *Enabling Pattern-Aware Automated Map Generalization*. PhD. Universität Zürich.

Steiniger, S. and Hay, G. J. (2008). An experiment to assess the perceptual organization of polygonal objects. In Klippel, A. And Hirtle, S. (eds) *You-are-here-maps—creating a sense of place through map-like representation, spatial cognition* Freiburg, pp. 38-44.

Steiniger, S. and Weibel, R., (2005). Relations and Structures in Categorical Maps. *The 8th ICA Workshop on Generalisation and Multiple Representation.* A Coruña (Spain), July 7-8th, 2005.

Steiniger, S. and Weibel, R., (2007). Relations among Map Objects in Cartographic Generalization. *Cartography and Geographic Information Science*. Vol. 34, No. 3 (2007), pp. 175-197.

Thomson, R.C. and Richardson, D.E. (1999). The 'Good Continuation' Principle of Perceptual Organization applied to the Generalization of Road Networks, in *Proceedings of the ICA 19th International Cartographic Conference*. pp. 1215-1223.

Zhang, X. (2012). *Automated evaluation of generalised topographic maps*. PhD. Twente University, Enschede.