EFFECTS OF IRREGULAR TOPOLOGY
IN
SPHERICAL SELF-ORGANIZING MAPS

A Thesis

Presented to the

Faculty of

San Diego State University

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in

Geography

---

by

Charles R. Schmidt

May 2008

SAN DIEGO STATE UNIVERSITY

The Undersigned Faculty Committee Approves the

Thesis of Charles R. Schmidt:

Effects of Irregular Topology in Spherical Self-Organizing Maps

Sergio J. Rey, Co-Chair
Department of Geography

Andr Skuin, Co-Chair
Department of Geography

Robert P. Malouf
Department of Linguistics and Asian/Middle Eastern Languages

Approval Date

TABLE OF CONTENTS

LIST OF TABLES

PAGE

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The Self-Organizing Map (SOM) is an unsupervised competitive learning process developed by Teuvo Kohonen as a technique to analyze and visualize high dimensional data sets. The applications of SOM are far reaching; Kohonen (2000) provides a thorough review of the SOM literature including applications of SOM. SOM has been used in applications ranging from speech recognition and image classification to breast cancer detection and gene expression clustering. Skupin and Agarwal (forthcoming) outline the growing interest of SOM in the GISciences, and propose that the relationship between SOM and GIScience should be bidirectional. The SOM offers a powerful method for exploring and visualizing geographic data and GIScience offers a wide array of tools and methods to enable the exploration of the SOM itself. This thesis takes advantage of GeoVisualization and GeoCompution in order to explore some of the basic properties of the SOM.

The SOM is a type of artificial neural network in which neurons are “organized” in such a way as to project the high-dimensional relationships of a set of training data onto a low-dimensional network structure. The traditional SOM uses a rectangular or hexagonal network topology (Kohonen, 2000). These topologies create a well-known problem in SOM called the boundary or edge effect. Neurons on the boundary of the hexagonal and rectangular lattices have fewer neighbors, which reduces their ability to interact with other neurons during the self-organizing process. Using a spherical lattice has been widely suggested as a solution to the problem (Ritter, 1999; Boudjemai et al., 2003; Sangole and Knopf, 2003; Nishio et al., 2006; Wu and Takatsuka, 2006). The use of the spherical lattice, however, does not completely overcome the boundary problem, and the choice of which spherical topology to use for the network can be difficult to make.

A regular network topology is one in which every node on the network has exactly the same number of adjacent nodes. Any topology involving an edge is irregular. Arranging our lattice on the surface of a sphere seems to be an obvious way to overcome the edge. However, there exist only five arrangements on the sphere which are completely regular; these are the five platonic solids (Ritter, 1999; Harris et al., 2000). Any other arrangement of neurons on the surface of the sphere will result in an irregular topology, as not all neurons will have the same number of neighbors. The classic method for minimizing this irregularity is to generate the spherical lattice by tessellating the sides of the icosahedron (Nishio et al., 2006). While this method will always result in a highly regular spherical topology, the main drawback is that the number of neurons in the network (the network size), N, grows exponentially as tessellations are applied. That results in only very coarse control over network size. Other methods for arranging neurons on the sphere allow for unlimited control over network size, but yield topologies with increased irregularity (Harris et al., 2000; Wu and Takatsuka, 2005; Nishio et al., 2006). To date the literature has largely ignored the more irregular methods in favor of the aforementioned tessellation-based methods. A topology which yields a more flexible network size may be desirable. However, in order to address this issue of network size, we must first determine the degree to which irregularity effects the SOM.

1.1 Research Objectives

The objective of this research is to determine the utility of certain irregular spherical topologies beyond offering greater control over network size. Toward that end, I develop and test new diagnostics to measure and visualize topology-induced errors in SOM. The following diagnostics are developed and implemented:

1. Compare the internal variance of observations captured by a given neuron to that neuron's first-order neighborhood size.
2. For different topologies, compare the internal variance of each neuron against a composite measure of topological regularity.
3. Develop a SOM-based visualization of the internal variance.

These diagnostics help facilitate the evaluation of both traditional and spherical SOMs. To satisfy the objective of this research, I apply these diagnostics to a series of comparable SOMs. Each SOM is trained using the same synthetic data and training parameters, but utilize different network topologies. By formally testing for difference of means and variance in the results of the diagnostics, the following questions are addressed:

1. Does the internal variance of a neuron decrease as its first-order neighborhood size, or degree, increases?
2. Is the average internal variance of a SOM higher when a more irregular topology is used?
3. Which insights, if any, can be gained from a SOM-based visualization of internal variance?

CHAPTER 2

BACKGROUND

This chapter is divided into four sections. The first provides a general introduction to the SOM algorithm and the problems created by using irregular topology. The second reviews the current spherical topologies used with SOMs. The third examines the flexibility of the various topologies with regard to network size. The fourth takes a look at the limitations of using "uniformity" to evaluate potential topologies.

2.1 Training and the Boundary Effect

The SOM algorithm uses an artificial neural network to organize high-dimensional data onto a low-dimensional lattice, or map, of neurons. Each neuron contains a reference vector that models the input data. Before training, these neuron-vectors are initialized, most commonly to random values. During the training process a randomly selected observation (input vector x) searches all neurons (reference vectors m_i) to find the one to which it is most similar, referred to as its Best Matching Unit (BMU c). The BMU and its neighborhood (N_c) are then adjusted to better match that observation (Kohonen, 2000). The training process is repeated a predefined number of times, or ideally until the map converges. The traditional SOM is laid out on a two-dimensional plane using either a rectangular or hexagonal topology. According to Wu and Takatsuka (2006), the hexagonal structure is more uniform and generally preferred.

One drawback of building the neural lattice in a discrete Euclidean plane is the boundary of the resulting lattice. A neuron located on the boundary has fewer neighbors and thus fewer chances of being updated (Wu and Takatsuka, 2006). As observed in Figure 2.1, neurons in the center of the map tend to better represent the mean of the input-space. This is arguably caused by outliers being pushed to the edges of the map, where they encounter fewer competing signals.

2.2 Spherical SOM

One way to eliminate the boundary effect is to wrap the lattice around a three-dimensional object such as a sphere or torus, thereby removing the edge entirely. The toroidal SOM was introduced by Li et al. (1993), however the torus is not effective for visualization, as maps generated from a torus are not very intuitive (Ito et al., 2000; Wu and Takatsuka, 2006). Ritter (1999) describes the torus as being topologically flat and suggests that a curved topology, such as that of a sphere, may better reflect directional data. A sphere also results in a more intuitive map, since we are accustomed to looking at geographic maps based on a sphere.

Ritter (1999) first introduced the spherical SOM, and several enhancements have since been suggested (Boudjemai et al., 2003; Sangole and Knopf, 2003; Nishio et al., 2006; Wu and Takatsuka, 2006). A good comparison of these enhancements can be found in Wu and Takatsuka (2006). All of these methods derive their spherical structure through the tessellation of a polyhedron as originally proposed by Ritter (1999). Wu and Takatsuka (2006) point out the importance of a uniform distribution on the sphere, and that it is preferable for all neurons to have an equal number of neighbors and to be equally spaced. They find generally that the tessellation method best satisfies these conditions, and specifically that the icosahedron is the best starting point (Wu and Takatsuka, 2005). Tessellation of the icosahedron results in a network of neurons, each of which having exactly six neighbors, save the original twelve which each have five neighbors. This is very close to the ideal structure in which every neuron would have exactly six neighbors. This structure has very low variances in both neuron spacing and neighborhood size.

2.3 NETWORK SIZE

The literature offers little theoretical guidance on choosing an appropriate network size for a given dataset (Cho et al., 1996). Vesanto (2005) suggests simply using a network size of $5\sqrt{n}$, where n is the number of observations. Given this lack of theoretical development, researchers should be cautious when using methods that limit the control of network size. Having a high level of control over network size allows support for such very different SOM applications as clustering versus low-dimensional spatial layout. Skupin and Agarwal (2007) demonstrate this when they use the same data to train two SOMs of different sizes. In the three-by-three (9) case the neurons act as containers clustering similar states, while in the twenty-by-twenty (400) case relationships are expressed with much finer granularity. As shown in Figure 2.2, methods for arranging an arbitrary number of points on a sphere provide a much higher degree of flexibility when choosing a network size.

The cost of relying on Ritter's tessellation method is decreased control over network size. Ritter's tessellation method results in a network size that grows at a rate of $N = 2 + 10 * 4^f$, where f is the frequency of tessellation. Wu and Takatsuka (2006) offer a slight improvement. Rather than recursively subdividing the faces, they redivide the original icosahedron with each step, resulting in $N = 2 + 10 * f^2$. In practice, 2D Euclidean SOMs also offer limited control over network size because it is undesirable to have one dimension dramatically larger than the other. Nishio et al. (2006) try to address the issue of network size granularity by departing from the tessellation method and suggesting the use of a partitioned helix to uniformly distribute any number of neurons on a sphere. A similar method proposed by Rakhmanov et al. (1994) was dismissed by Wu and Takatsuka (2005) for failing to satisfy the uniformity conditions described above.

2.4 UNIFORMITY

Wu and Takatsuka state that "[f]or SOM, it is desirable to have all neurons receive equal geometrical treatment" (Wu and Takatsuka, 2006, p. 900). To satisfy this constraint, two conditions must be met. First, each neuron should occupy the same amount of space on the given surface. Second, each neuron should be bordered by the same number of surrounding neurons, and we should maximize that number. The first condition may be important for visualization, but is irrelevant for training. During the training of the SOM only the topology of the neurons is considered.

Based solely on measures of neuron spacing, Wu and Takatsuka (2005) dismissed a method proposed by Rakhmanov et al. (1994) for distributing points on a sphere. Similarly Nishio et al. (2006) use these variance measures to support their helix algorithm for distributing points on a sphere. Table 2.1 shows that these metrics can be misleading and comparison across topologies may not be consistent. The traditional rectangular and hexagonal topologies have no variance in neuron spacing, and the generally preferred hexagonal structure displays greater variance in neighborhood size than the rectangular structure. The torus, by comparison, would have variance in neuron spacing, yet no variance in neighborhood size. The distance between two neurons is only considered during the formation of the neural network. At this stage the spacing is significant as it plays a part in determining neuron adjacency. However, using this measure to evaluate potential topologies for use in SOM may be misleading.

Table 2.1. Variances in Topologies

Topology	Grid Size	Neuron Spacing	Variance in Neighborhood Size
Rectangular	9x18	1	0.2716
Hexagonal	9x18	1	1.2138
Tessellation	162	0.25319 - 0.31287	0.0686
Rakhmanov	162	0.15779 - 0.30069	0.2908

Methods for distributing points on the sphere, which allow for fine-grained control over network size, produce slightly more irregular topologies. However, no substantive discussion of these irregularities or their effects on SOM training exists in the literature. Given that limited theoretical guidance is available for choosing network size, the desire for finer control over the network size should not be overlooked. Particularly for larger SOMs, the desired network size may not be achievable via tessellation of the icosahedron.

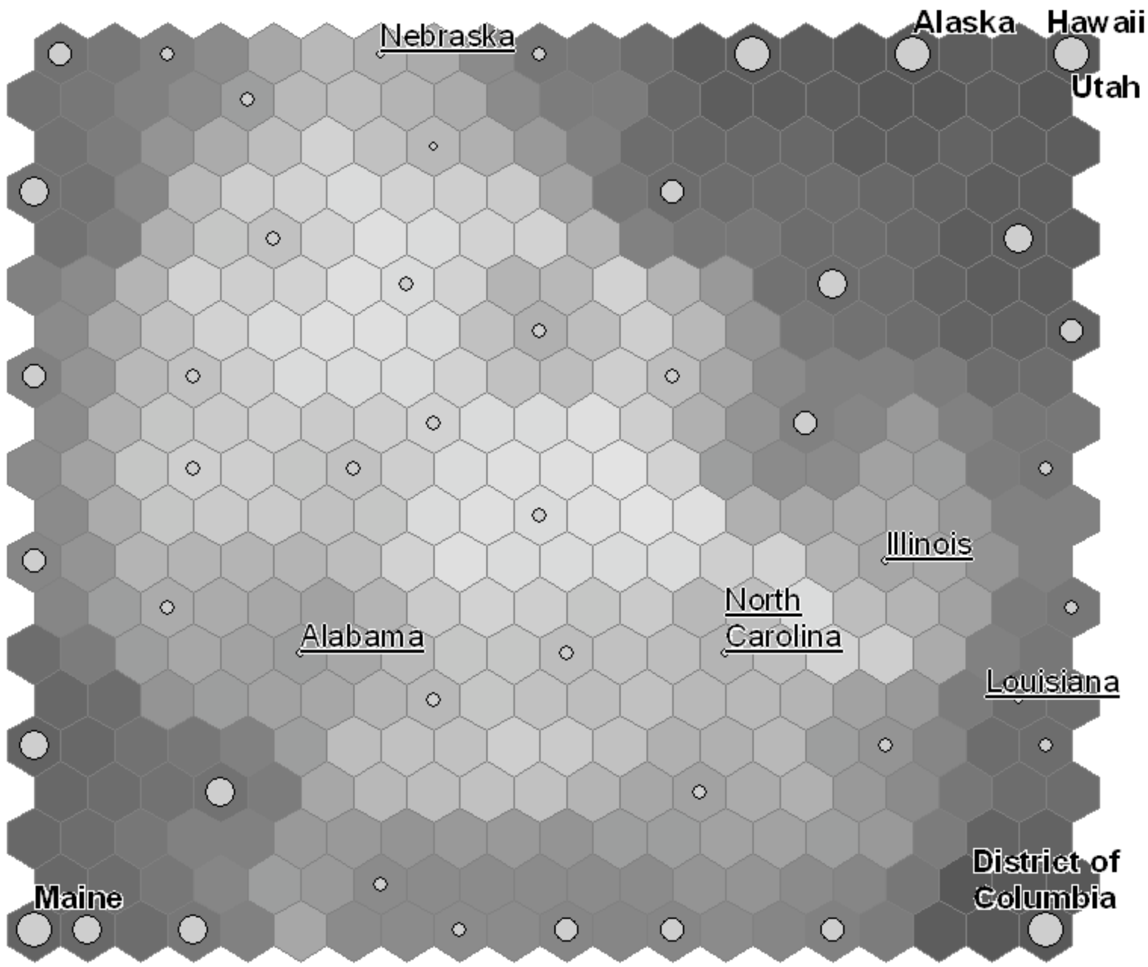


Figure 2.1. Fifty States and the District of Columbia mapped onto a SOM trained with thirty-two population census variables. Darker neurons have a relatively larger difference from the mean of the states, while lighter neurons are relatively closer. Smaller point symbols show states that are closer to the mean, while large symbols show outliers. The five states closest to the average are shown with underlined labels and the five states furthest from the mean are shown with bold labels.

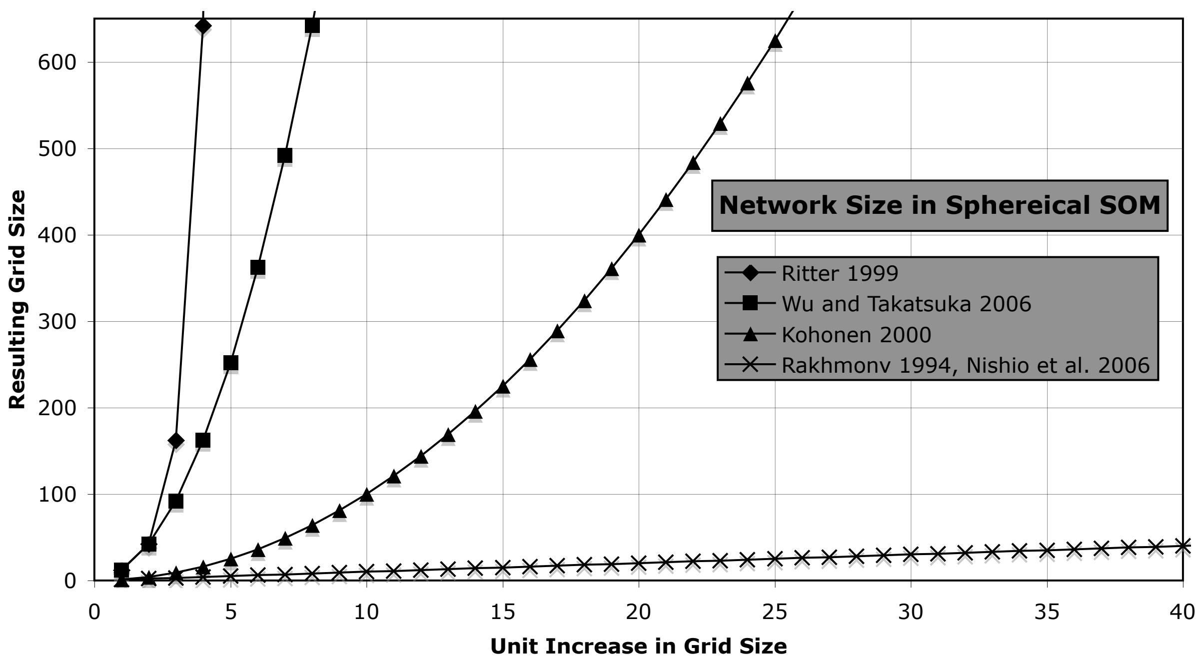


Figure 2.2. This figure demonstrates the achievable network size using various spherical topologies, in comparison with the traditional SOM. The Y-axis represents the achievable network size, the meaning of the X-axis is dependent on the topology. For the tessellation methods the X-axis represents the frequency of the tessellation. For the traditional Kohonen method the X-axis represents the size of both dimensions of the grid; for comparability the ratio between the dimensions was fixed at one ($X_{dim} = Y_{dim}$). For the Rakhmanov et al. (1994) and Nishio et al. (2006) methods the X-axis represents the exact network size.

CHAPTER 3

METHODOLOGY

This chapter is composed of three sections. Section 3.1 describes three diagnostics developed for evaluating network topologies. The diagnostics are applied to a series of trained SOMs. Synthetic training data, as described in section 3.2, is used as our training data. The parameters of the training, section 3.3, are consistent across each trained SOM.

3.1 Diagnostics

Three diagnostics are developed to explore the effect of irregular topology on spherical SOM. All three build on the idea of an internal variance IV. The internal variance of neuron (i) is defined as,

$$IV_i = \frac{2}{{n_i}^2 - n_i} \sum_{j=1}^{n_i} \sum_{k=j+1}^{n_i} ||x_{ij} - x_{ik}|| \tag{3.1}$$

where, n_i is the number of observations mapped to i, and x_i are the input vectors mapped to i.

The first diagnostic will be used to address the research question regarding the internal variance and neighborhood size. The second diagnostic will address the question concerning internal variance and topological irregularity. The third tool will help visualize the patterns between internal variance and topology.

3.1.1 Internal variance vs. first-order neighborhood size

This diagnostic will compare the internal variance of each neuron against its first-order neighborhood size. In traditional SOMs, outlying observations are pushed to the edge of the map where they encounter fewer competing signals. A prime example of this is the "Utah-Hawaii" case shown in Figure 2.1. Relying only on the SOM, one would be left to believe that the two states are similar. Upon closer inspection we see that the QError[1] from Utah to the neuron is 1.509, the QError from Hawaii to the neuron in 1.505, but the QError from Utah to Hawaii is 3.014. In this case only Utah and Hawaii were mapped to that neuron. In a case where multiple observations land on the same neuron, it is possible to measure average pairwise QErrors between those observations. This gives us a notion of internal

[1]The average quantization error (QError) is the standard method for measuring the quality of the SOM. The QError measures the distance, in attribute space, of an observation to its BMU in the trained SOM (Kohonen, 2000). The average QError provides a global measure of accuarcy for the SOM.

variance for each neuron. It would be expected that in traditional SOMs neurons closer to the edge will have higher internal variances. This can be extended to spherical SOMs by comparing the degree of a neuron ($deg(m_i)$ or the number of adjacent neurons) to its internal variance. The degree of each neuron can easily be calculated by taking the column sums of the first-order adjacency matrix (A).

Once the internal variance ($var(m_i)$) and degree ($deg(m_i)$) of each neuron have been calculated, the neurons can be separated into a small number of groups based on the degree [2]. The variance and mean will be calculated for each of these groups. The expected result is that variances and means of the groups will decrease as the degree increases. This hypothesis will be tested using random labeling as described by Rey (2004). The result will also be visualized using a box plot.

3.1.2 Internal variance vs. topological regularity

This diagnostic will compare the internal variance of each neuron against a measure of regularity for its associated topology. As mentioned above the degree of each neuron can be calculated by taking the column sums of A. A completely regular network topology (i.e. the torus) will have no variance between these column sums. For irregular networks the variance between these column sums gives us a measure of irregularity. There are many alternative ways to classify the connectivity of a network; Florax and Rey (1995) outline four such summary measures, which will be evaluated for use in this diagnostic. For each topology we can compare the internal variances as described above against a measure that summarizes the given topology's regularity.

This diagnostic is evaluated in much the same way as the last diagnostic. The internal variances are this time grouped by their topology. We can then compare the variances of internal variances and the means of the internal variances across topologies. It is expected that the distribution of internal variances will be narrower for groups trained on more regular topologies. It is further hypothesized that the means of these internal variances will decrease when the network is more regular, or when there is less variance in the column sums of A.

3.1.3 Visualize internal variance mapping

Visualizing the internal variance may yield insight into how irregular topology effects the SOM. Once the internal variance of each neuron has been calculated we can use the values to color or shade a map of the given topology. The degree of the neurons can be visualized using proportional symbols to help show patterns between internal variance and irregularity.

3.2 DATA

[2]For most topologies the number of different degrees will be limited to three or four.

The internal variance measure is sensitive to both the properties of the SOM and the properties of the training data. Therefore, a dataset with uniform properties is needed. We follow the method for generating uniform synthetic data used by Wu and Takatsuka (2006). Their method creates seven clusters in three dimensions. Each cluster is normally distributed and has a standard deviation of one. The uniform clusters generated by this method allow us to systematically compare the diagnostics under several different topologies. To ensure that we can calculate an internal variance for each neuron, we create a large number of observations relative to the number of neurons.

3.3 SOM Training

Before we can go on to address the research questions we need to train a series of SOMs. We train SOMs using four different topologies: *rectangular, hexagonal, geodesic sphere* and *spherical*. The spherical topology is based on a method, developed by Rakhmanov et al. (1994), for distributing an arbitrary number of points on to the surface of a sphere. Delaunay triangulation is then applied to these points, producing a topological structure. To yield meaningful results these SOMs must be trained with comparable parameters. The literature provides many rules of thumb for training a SOM: each SOM is trained in two stages, the first of which uses a larger initial learning rate and neighborhood search radius with a small number of training steps; the second stage uses a lower initial learning rate and neighborhood search radius, but extends the length of training.
First Stage Parameters:

- Initial neighborhood search radius of 50%, which decreases during training.
- Initial learning rates of 0.04 which decreases during training.
- 100,000 training steps.

Second Stage Parameters:

- Initial neighborhood search radius of 33%, which decreases during training.
- Initial learning rates of 0.03 which decreases during training.
- 1,000,000 training steps.

As shown in Figure 2.2, topologies differ in terms of achievable network size. For comparability, the network size of each SOM needs to be as close as possible. The achievable network size for the geodesic SOM is the most limiting of the topologies we test. We chose the eighth frequency geodesic sphere, which has 642 nodes, which is relatively close to the 644-node hexagonal and rectangular topologies achieved when the dimensions are set to $28x23$. Finally, the spherical topology was set to 642 nodes.

One problem that we face is a small sample size when the neurons of a given SOM are grouped by their degree. For example, the four corners of the rectangular topology are the only neurons that have a degree of two. The rest of the neurons have three or four neighbors depending on whether or not they are on the edge. To address the problem of small sample size for topologies with relatively few neurons of a particular degree, we will increase the sample size by combining the results of many SOMs.

Ten synthetic datasets are generated using the parameters described above. The datasets are used to train SOMs for each topology resulting in forty trained SOMs. The mean internal variance of these SOMs is summarized in Table 3.1. We find that the mean internal variance remains fairly stable, suggesting that the results of each simulation can be combined within a given topology. For the rectangular topology, we now have forty neurons with a degree of two for which an internal variance can be calculated.[3]

Table 3.1. Mean internal variance for each simulation, by topology.

Simulation	Geodesic	Spherical	Hexagonal	Rectangular
1	0.0277	0.0277	0.0285	0.0289
2	0.0281	0.0281	0.0291	0.0295
3	0.0278	0.0280	0.0286	0.0292
4	0.0280	0.0282	0.0286	0.0293
5	0.0279	0.0280	0.0289	0.0296
6	0.0278	0.0274	0.0285	0.0290
7	0.0286	0.0283	0.0294	0.0297
8	0.0284	0.0285	0.0294	0.0298
9	0.0283	0.0282	0.0293	0.0295
10	0.0285	0.0285	0.0293	0.0298
Combined	0.0281	0.0281	0.0290	0.0294

[3]We can only measure the internal variance when a neuron captures two or more observations from the training data. Therefore, it is still possible to have less than forty measurements.

CHAPTER 4

RESULTS AND DISCUSSION

This chapter applies our three diagnostics to the trained SOMs. Section 4.1 discusses the application of the first diagnostic, which looks at how internal variance changes with a neuron's first order neighborhood size, or degree. In section 4.2 the second diagnostic, which compares the mean internal variance across topologies, is applied. Finally, section 4.3 applies the third diagnostic, which visualizes the internal variance of each SOM.

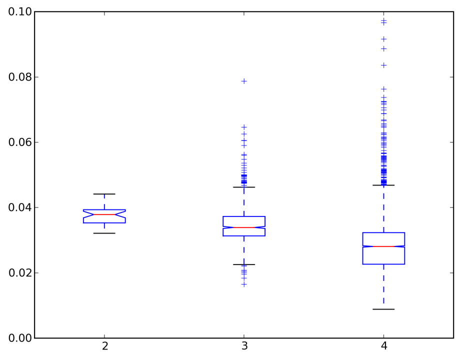
4.1 Internal variance vs. first-order neighborhood size

The *degree* is one measure of a neuron's centrality in a network. A neuron with a higher degree has higher network connectivity, and therefore is updated more frequently during the training process. We hypothesized that outlying observations would migrate to less central neurons on the map, where there is less competition. If this was true, we expect the internal variance as measured by equation 3.1 to increase at these locations, demonstrating that topological irregularity affects the placement of outliers in the SOM.

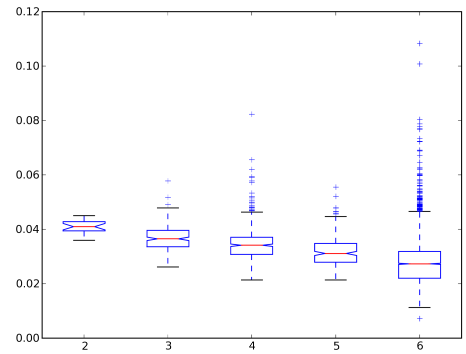
For each topology, the internal variance and degree of each neuron was calculated. We then grouped the internal variance measures by degree. These groups are treated as representative samples from a larger distribution. Table 4.1 shows the details of each sample, including its size, mean and variance. The size is the number of observations for which we were able to calculate an internal variance. The mean and variance capture the centrality and dispersion of the samples. We create a box-and-whisker diagram (boxplot) for each sample, as shown in figure 4.1, to visualize the samples summarized in table 4.1.

We see that the means of the samples seem to respond as expected for the rectangular and hexagonal, or "flat," topologies, but not in the spherical topologies. While this is not what we anticipated, it may suggest that the spherical and geodesic topologies are effectively overcoming the edge problem. The variance of the samples, however, did not respond as expected. In the flat topologies the variance increased as the degree increased. This is possibly due to the large difference in sample size. To verify these conclusions we formally test for differences in means and variance between the samples.

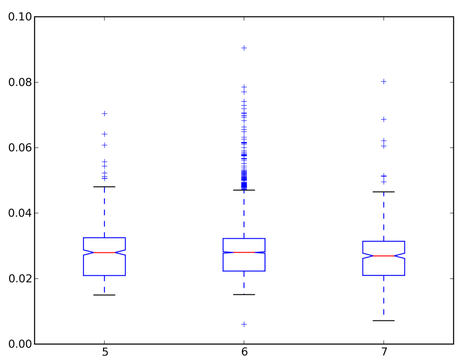
The results of the means test are presented in table 4.2. This table shows the difference in means below the diagonal, the p-value above the diagonal (with significant values in bold), and the variance of each sample along the diagonal. In the rectangular and hexagonal topologies, we observe that all sample means are significantly different. We also observed significant differences in the variance of the samples, except the case of degree size four and five in the hexagonal topology, where they were not significantly different. No variance in the spherical and geodesic topologies were significantly different. In those topologies, the only significant difference in means was between the sample with degree size six and seven in the spherical topology.



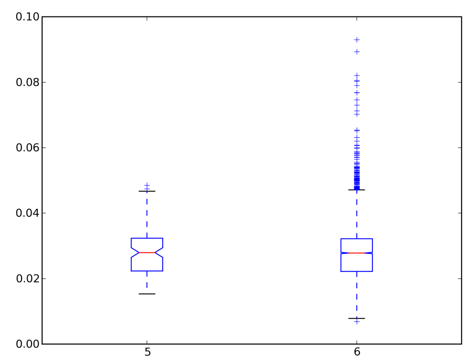
(a) Rectangular Topology



(b) Hexagonal Topology



(c) Spherical Topology



(d) Geodesic Sphere Topology

Figure 4.1. Box-and-whisker diagrams representing samples derived from forty trained SOMs. The samples within each topology were created by grouping the internal variance of each neuron of a given degree size. The diagrams how the centrality and dispersion of each sample.

4.2 INTERNAL VARIANCE VS. CENTRALITY

The degree of a node on a network is a measure of its centrality, or importance. Nodes with more connections are thought to be more central to the network and have a larger influence than nodes with fewer connections. As an artifact of the training process

Table 4.1. Size, mean and variance of each sample

Degree	Geodesic		Spherical		Hexagonal		Rectangular	
	N	Mean (Var)	N	Mean (Var)	N	Mean (Var)	N	Mean (Var)
2					20	0.0409 (5.66E-06)	40	0.0378 (7.59E-06)
3					218	0.0371 (2.18E-05)	880	0.0348 (3.59E-05)
4					489	0.0347 (4.05E-05)	4926	0.0284 (6.28E-05)
5	113	0.0283 (5.28E-05)	526	0.0279 (6.43E-05)	206	0.0319 (3.50E-05)		
6	5598	0.0281 (6.05E-05)	4758	0.0282 (6.05E-05)	4954	0.0278 (6.09E-05)		
7			417	0.0273 (6.65E-05)				

observations that are more average than others tend to be centralized. The observations that surround them tend to be more extreme. If you refer back to figure 2.1, you'll notice that observations with smaller symbols are closer to the mean of all the observations and that these observations have been centralized in the network. Using the degree as a measure of centrality does not capture this picture well, as neurons near the edge can still have a large degree. A way to capture this effect would be to look at closeness centrality, which is the inverse of the average distance of a neuron to every other neuron on the network.

Above we saw that changes in a neuron's degree was related to changes in internal variance in topologies with edges, but not in the two sphere-based topologies. The next step is to see if internal variance changes between topologies. To do this we will first order our topologies by a summary measure of the network centrality. For this diagnostic we use the average closeness centrality for each topology as the summary measure. The summary measure and the sorting of our topologies is shown in table 4.3.

In this diagnostic we group the samples from the previous section by topology. This results in one sample for each of the four topologies. We test for a difference in mean and variance between each sample using the same method of random labeling that was applied in the previous diagnostic. No significant differences were found in the variances; the difference in means are presented in table 4.4. It was expected that the mean and variance of the samples would decrease for the more regular topologies. These results generally support this hypothesis. We see that the rectangular topology has the highest mean internal variance and is the least regular as measured by closeness centrality. The geodesic and spherical topologies are the most regular and have the lowest internal variance. These two groups display very similar measures of closeness centrality and show no significant difference in mean internal variance. This suggests that even though the spherical topology is more irregular than the geodesic topology, similar levels of quality may be achieved.

4.3 Visualization of internal variance

In order to address our research questions we first created ten synthetic datasets. Each dataset was created by sampling from the same data generating process that was provided by Wu and Takatsuka (2006). We used these datasets to train ten SOMs for each of our

Table 4.2. Results of Difference of Mean Testing Within Each Topology

(a) Rectangular Topology

Degree	2	3	4
2	(0.000008)	**0.002700**	**0.000100**
3	0.002969	(0.000036)	**0.000100**
4	0.009364	0.006396	(0.000063)

(b) Hexagonal Topology

Degree	2	3	4	5	6
2	(0.000006)	**0.000800**	**0.000200**	**0.000100**	**0.000100**
3	0.003810	(0.000022)	**0.000100**	**0.000100**	**0.000100**
4	0.006253	0.002443	(0.000040)	**0.000100**	**0.000100**
5	0.009008	0.005197	0.002755	(0.000035)	**0.000100**
6	0.013067	0.009257	0.006814	0.004059	(0.000061)

(c) Spherical Topology

Degree	5	6	7
5	(0.000064)	0.485500	0.197400
6	0.000250	(0.000060)	**0.020000**
7	0.000674	0.000924	(0.000067)

(d) Geodesic Topology

Degree	5	6
5	(0.000053)	0.783900
6	0.000203	(0.000060)

topologies. The purpose of training ten SOMs was to produce large enough samples sizes for the difference of means and variance testing that was necessary to formally evaluate the results of our first two research questions. In this section we verify that combining those simulations was appropriate by visualizing the similarities between them. In the remainder of this section we focus our efforts on comparing internal variance across topologies. Since we have seen that little variation exists between the ten simulations, we choose the first of those simulations for each topology and explore them in more depth.

What information can be gained from visualizing the SOM and its internal variance.

In order to create these visualizations we used off-the-shelf GIS packages, namely ESRI's ArcGIS. Two primary challenges had to be overcome. First, to visualize the rectangular and hexagonal topologies we created polygons centered over each neuron, however, the spherical and geodesic topologies were slightly more complicated. Creating the polygons for these topologies required that we first compute the Voronoi diagram on the surface of the sphere. This is done using STRIPACK, a software program created by Ranka

Table 4.3. Measure of Topological Regularity and Sample Mean and Variance

Topology	Closeness Centrality	Mean	Variance
Rectangular	0.0603	0.0294	0.0001
Hexagonal	0.0739	0.0289	0.0001
Geodesic	0.0890	0.0281	0.0001
Spherical	0.0906	0.0281	0.0001

Table 4.4. Results of Difference of Mean Testing Across Topologies

Topology	Rectangular	Hexagonal	Geodesic	Spherical
Rectangular	(0.000064)	**0.001000**	**0.000100**	**0.000100**
Hexagonal	-0.000479	(0.000064)	**0.000100**	**0.000100**
Geodesic	-0.001329	-0.000850	(0.000060)	0.505600
Spherical	-0.001328	-0.000849	0.000001	(0.000061)

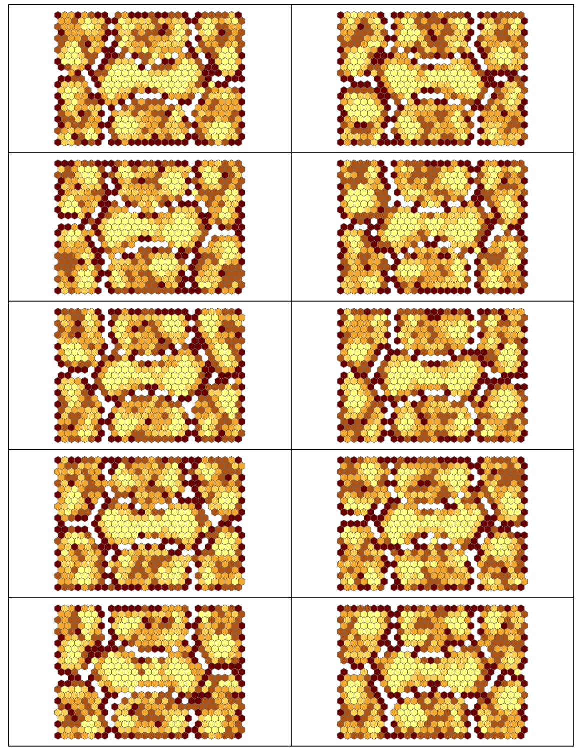
(1997). Second, ArcGIS and other common GIS packages assume Cartesian distances and thus can not handle polygons that cross the 180^{th} meridian. To accommodate this we split each polygon at the 180^{th} meridian and redrew it as two parts.

As expected, figure 4.2 reveals little variation between simulations of a given topology. This homogeneity demonstrates that our ten synthetic datasets adequately increase our sample size without introducing bias. Because of the apparent homogeneity between the simulations *within* a given topology, we move on to compare the first simulation *across* topologies. Our synthetic training data consisted of seven clusters located in three dimensional space. We examine how the SOM treats the original data through a series of visualizations. The component planes in figure 4.3 show how the SOM represents each dimension of the training data. The variations we see in the placement of high and low values, between topologies, is caused by the random initialization of the SOM. However, the relationships between the dimensions remain constant across topologies.

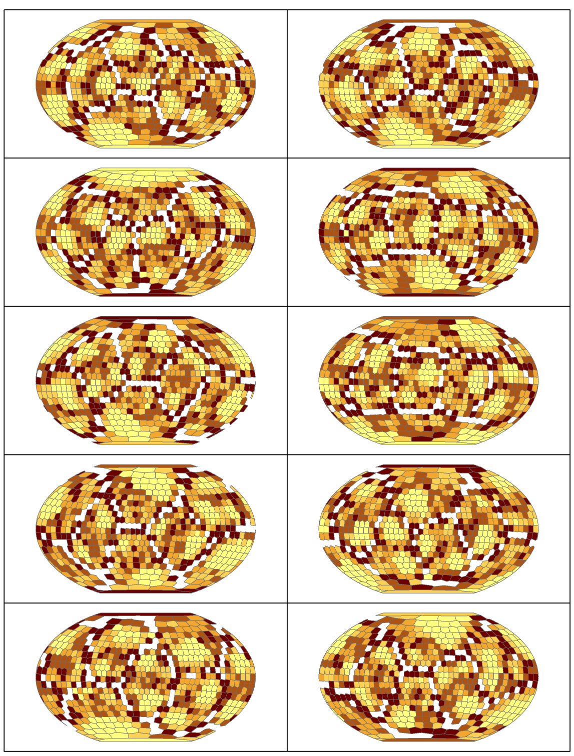
Figure 4.4 shows how the SOM detected the clustering of the input data. We compute the cluster each neuron represents by looking at the observations mapped to it. Knowing which cluster each observation belongs to allows us to see where the clusters are being mapped to on the trained SOM. For a given neuron we classify it by the cluster it captured most frequently.

It is interesting to note that the edges of the cluster exhibit high internal variance. This is somewhat intuitive, as our clusters are normally distributed; the majority of our observations will fall well within the regions observed in figure 4.4. Each cluster's outliers will be pushed toward the *edges* of these regions. In the hexagonal and rectangular topologies, some of these cluster regions share an edge with the topology itself. This allows their outlying observation to interact less with the interior neurons. In the spherical and geodesic topologies outliers that are pushed toward the edge of their cluster's region on the map are forced to interact with other neurons.

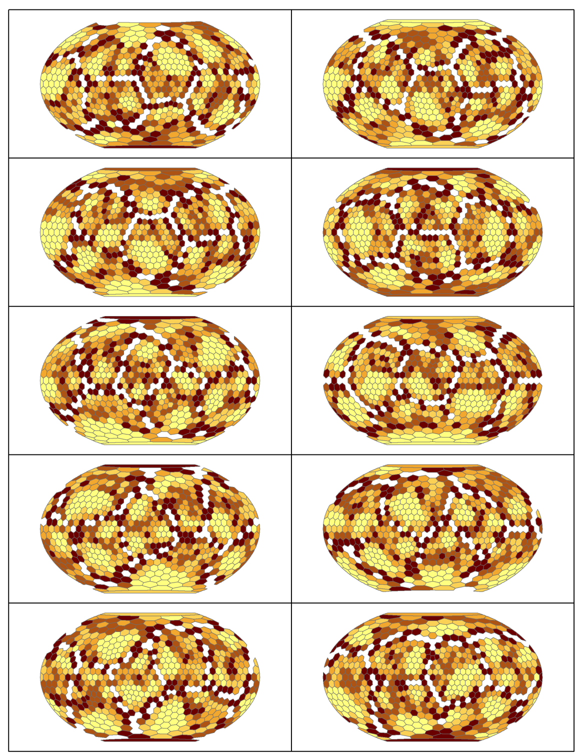
(a) Rectangular Topology



(b) Hexagonal Topology



(c) Spherical Topology



(d) Geodesic Sphere Topology

Figure 4.2. Internal variance mapping for each of the forty SOMs. Darker colors represent neurons that display larger internal variance. Neurons for which an internal variance could not be calculated are not displayed.

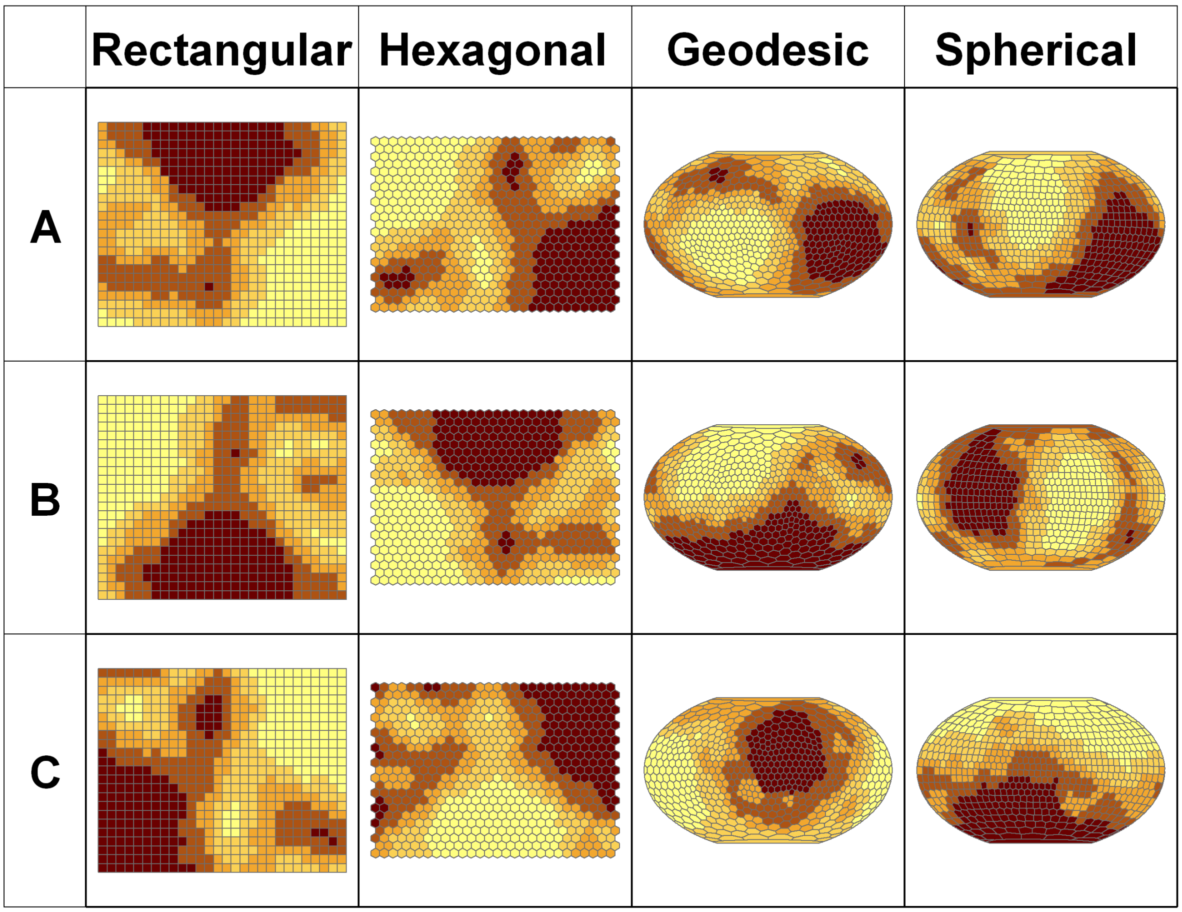
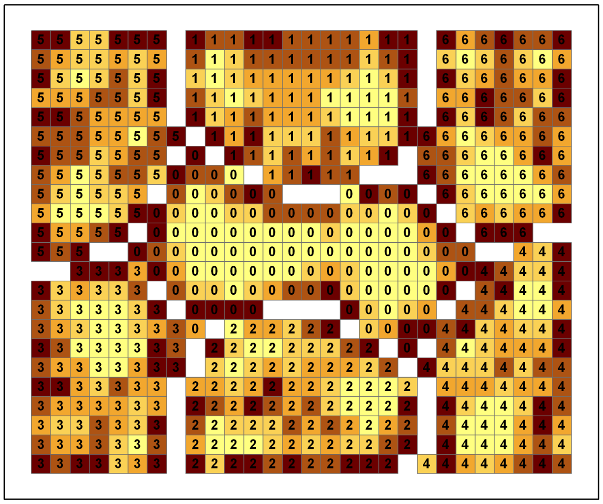
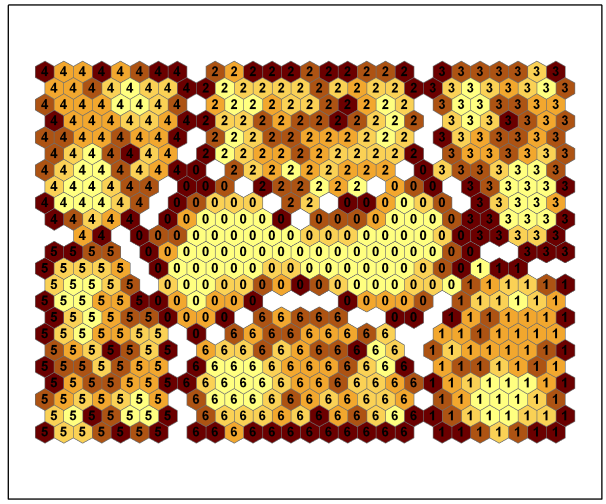


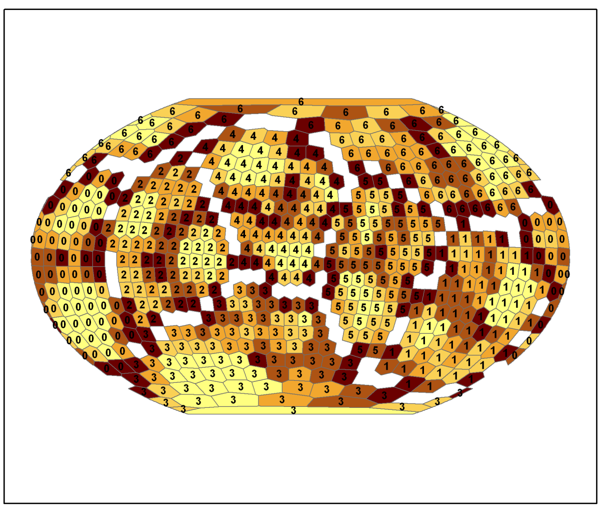
Figure 4.3. The first (A), second (B) and third (C) component planes are shown for the first simulation of each topology. These component planes show how the original dimensions are represented in the trained SOMs.



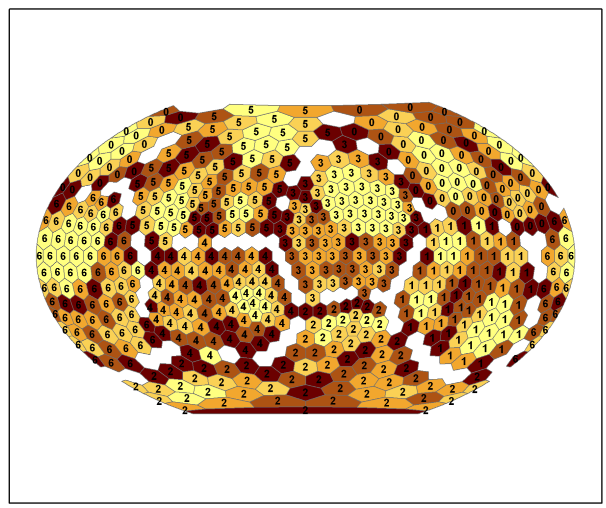
(a) Rectangular Topology



(b) Hexagonal Topology



(c) Spherical Topology



(d) Geodesic Sphere Topology

Figure 4.4. Detailed internal variance mapping for each topology. Darker colors represent neurons that display larger internal variance. Neurons for which an internal variance could not be calculated are not displayed. The numbers represent primary cluster mapped at that neuron.

CHAPTER 5

CONCLUSIONS

5.1 SIGNIFICANCE

The commonly used tessellated icosahedron based topology offers the most regular topology. However, the main disadvantage of this topology type is that it offers a limited control over network size. Alternative methods for generating the spherical topology, which can create a network of any size, have been reviewed or suggested by Wu and Takatsuka (2005) and Nishio et al. (2006). These alternative methods produce network structures that are more irregular. This research will take a closer look at the impact that the irregularity has on the training process in an attempt to address the suitability of these topologies for use in SOM.

5.2 LIMITATIONS

This research will look at the relationship between regularity in neuron connectedness and the training of a SOM. The relationship between topology and SOM visualization is not addressed. The topology chosen for a SOM has a direct link with how that SOM is visualized. When representing the topology on the surface of a sphere issues arise with the uniformity in neuron spacing and sizing. Future work may be needed to address these issues in visualization.

BIBLIOGRAPHY

Boudjemai, F., Enberg, P. B., and Postaire, J. G. (2003). Surface modeling by using self organizing maps of Kohonen. In *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, volume 3, pages 2418–2423 vol.3.

Cho, S., Jang, M., and Reggia, J. A. (1996). Effects of varying parameters on properties of self-organizing feature maps. *Neural Processing Letters*, V4(1):53–59. Available from: `http://dx.doi.org/10.1007/BF00454846`.

Florax, R. J. and Rey, S. (1995). The impacts of misspecified spatial interaction in linear regresion models. In *New directions in spatial econometrics*, Advances in Spatial Science. Springer.

Harris, J. M., Hirst, J. L., and Mossinghoff, M. J. (2000). *Combinatorics and graph theory*. Springer, New York.

Ito, M., Miyoshi, T., and Masuyama, H. (2000). The characteristics of the torus self organizing map. In *Proceedings of 6th International Conference ON Soft Computing (IIZUKA2000)*, volume A-7-2, pages pp.239–244, Iizuka, Fukuoka, Japan. Available from: `http://mylab.ike.tottori-u.ac.jp/~mijosxi/act1997-/#Oral`.

Kohonen, T. (2000). *Self-Organizing Maps*. Springer, 3rd edition.

Li, X., Gasteiger, J., and Zupan, J. (1993). On the topology distortion in self-organizing feature maps. *Biological Cybernetics*, V70(2):189–198. Available from: `http://dx.doi.org/10.1007/BF00200832`.

Nishio, H., Altaf-Ul-Amin, M., Kurokawa, K., and Kanaya, S. (2006). Spherical SOM and arrangement of neurons using helix on sphere. *IPSJ Digital Courier*, 2:133–137.

Rakhmanov, E. A., Saff, E. B., and Zhou, Y. M. (1994). Minimal discrete energy on the sphere. *Mathematical Research Letters*, 1:647–662.

Ranka, R. J. (1997). Algorithm 772. stripack: Delaunay triangulation and voronoi diagram on the surface of a sphere. *ACM Transactions on Mathematical Software*, 23(3):416–434.

Rey, S. J. (2004). *Spatially Integrated Social Science*, chapter Spatial Analysis of Regional Income Inequality. Oxford University Press, Oxford [England] ; New York :.

Ritter, H. (1999). Self-organizing maps on non-euclidean spaces. In Oja, E. & Kaski, S., editor, *Kohonen Maps*, pages 97–110. Elsevier, Amsterdam. Available from: `citeseer.ist.psu.edu/ritter99selforganizing.html`.

Sangole, A. and Knopf, G. K. (2003). Visualization of randomly ordered numeric data sets using spherical self-organizing feature maps. *Computers & Graphics*, 27(6):963–976. Available from: `http://www.sciencedirect.com/science/article/B6TYG-49S7595-3/2/0913af2271d2f83a833452eea7ebddf9`.

Skupin, A. and Agarwal, P. (2007). (In preparation) Introduction: What is a self-organizing map? In Agarwal, P. and Skupin, A., editors, *Self-Organizing Maps: Applications in Geographic Information Science*. In preparation for publication by Wiley.

Vesanto, J. (2005). Som toolbox: implementation of the algorithm. `http://www.cis.hut.fi/projects/somtoolbox/documentation/somalg.shtml`. Available from: `http://www.cis.hut.fi/projects/somtoolbox/documentation/somalg.shtml` [cited November 20, 2006].

Wu, Y. and Takatsuka, M. (2005). Geodesic self-organizing map. In Erbacher, R. F., Roberts, J. C., Grohn, M. T., and Borner, K., editors, *Proc. SPIE Vol. 5669*, volume 5669 of *Visualization and Data Analysis 2005*, pages 21–30. SPIE.

Wu, Y. and Takatsuka, M. (2006). Spherical self-organizing map using efficient indexed geodesic data structure. *Neural Networks*, 19(6-7):900–910. Available from: `http://www.sciencedirect.com/science/article/B6T08-4K719S6-2/2/f6d26607f815781a5bd3cfe5df5d6214`.