AN ALGORITHMIC APPROACH FOR SIMULATING REALISTIC IRREGULAR TILINGS

Por

Alejandro Betancourt

Proyecto de investigación enviado al departamento de ciencias básicas como requerimiento parcial para la obtención del título de:

MÁSTER

EN MATEMÁTICAS APLICADAS

Supervisada y dirigida por:
Juan Carlos Duque y Freddy Hernan Marín

Elaborada en el grupo de investigación:
RiSE (Research in Spatial Economics)

Universidad EAFIT
Medellín, Colombia

Septiembre 2012
(Para graduación en Diciembre 2012)
Este trabajo está dedicado a mi familia por sus esfuerzos y sacrificios durante mis estudios, sin ellos nada de esto sería posible.
ACKNOWLEDGMENT

It is a pleasure to thank the many people who helped to complete this project effectively and moreover on time. This project would not have been possible without the support and supervision of Prof. Juan C. Duque and Prof. Freddy Marín. Also I would like to thank each member of RiSE group for their comments and suggestions to improve this project.

I would like to gratefully acknowledge the high performance computing team “Apolo” of EAFIT University for the computational support on the experiments that made possible this thesis.

Last but not the least, I would like to thank the Research unit of EAFIT University for giving me the chance to study the master’s program of Applied Mathematics and supporting my research activities at EAFIT during the last 6 years.
ABSTRACT

There exist a wide variety of experiments, or simulations, in which regional scientists require regular and irregular lattice structures with a predefined number of spatial units \( n \). While many GIS software offer the possibility to generate a regular lattice, to obtain an irregular tiling of a given size is not always a straightforward task. The utilization of an irregular lattice usually requires that the researcher finds a large enough real map that matches as closely as possible the required number of spatial units. This practice is usually carried out without considering whether or not the selected lattice is a good representation, in terms of its connectedness characteristics, of a “typical” real map of polygons.

In this project we propose an algorithm, written in Python as a toolbox for clusterpy, that combines fractal theory, theory of stochastic processes and computational geometry for simulating realistic irregular tiling with a predefined number of spatial units. The irregular tiling generated with this algorithm guarantee consistency in their connectedness characteristics and, therefore, reduce potential distortions in the results of the experiments due to an inappropriate selection of the lattice structure.
# CONTENTS

ACKNOWLEDGMENT .............................................................. iv

ABSTRACT ................................................................. vi

LIST OF FIGURES .......................................................... x

LIST OF TABLES ........................................................... xii

1. Introduction .............................................................. 1

2. About tilings and space partitions ................................. 5
   2.1 Space partition, consensus taxonomy of tilings ................ 5
   2.2 Measuring the topological characteristics of tilings ......... 7

3. Tilings in regional science ............................................ 11
   3.1 Commonly used tilings ........................................... 11
   3.2 Topological comparison of tilings ................................. 13

4. Generating realistic tilings .......................................... 17
   4.1 Introduction to the mean reverting process .................. 17
      4.1.1 Brownian motion ............................................ 17
      4.1.2 Stochastic differential equations ......................... 18
      4.1.2.1 Existence and Uniqueness of solutions .............. 19
      4.1.2.2 Lineal Stochastic Differential Equations .......... 21
   4.2 Introduction to the fractal dimension .......................... 23
   4.3 Mean reverting areas (MR-Polygons) .......................... 25
      4.3.1 Algorithm .................................................. 25
      4.3.2 Parametrization of MR-Polygons ......................... 27
      4.3.3 MR-Polygons with python ................................. 30
   4.4 Recursive Irregular maps (RI-Maps) ............................ 32
      4.4.1 Algorithm .................................................. 32
      4.4.2 Parametrizing RI-Maps .................................... 36
      4.4.3 RI-Maps with python ..................................... 38
5. Results ................................................................. 41
   5.1 Topological characteristics of *RI-Maps* ...................... 41
   5.2 *RI-Maps* examples .............................................. 41
   5.3 Application of *RI-Maps* ......................................... 46
6. Conclusions .......................................................... 51
7. Future research ...................................................... 53
8. Python library ......................................................... 55
   8.1 Installing clusterpy ................................................ 55
   8.2 Examples of clusterpy ............................................. 55
# LIST OF FIGURES

1.1 Randomly sampled maps of 900 areas ........................................... 2

2.1 Examples of different type of polygons ................................. 6

2.2 Consensus taxonomy of tilings .................................................. 7

3.1 Topological differences of maps from Spain and United States ...... 13

3.2 Base map and example of a random irregular maps obtained from it 14

4.1 Geometric problem to preserve the length and the fractal dimension of mean reverting process when is used to create an irregular polygon 26

4.2 Illustrative example of Irregularities explained by the Structural and Textural dimension ......................................................... 28

4.3 Empirical frequency distribution of structural fractal dimensions of real polygons ................................................................. 29

4.4 Fractal dimension of simulated polygons in function of $\alpha$ and $\sigma$ 30

4.5 Examples of stochastic polygons generated using (Pseudocode 1) with different values of $\sigma$ and $\alpha$ .................................................... 31

4.6 Fitness function across the iterations .......................................... 38

5.1 Comparison of topological characteristics of reals maps, RI-Maps, Voronoi diagrams and regular tilings .................................................. 42

5.2 SAR(0.7) process over a RI-Map of 400 areas ............................ 43

5.3 SAR(0.7) process over a RI-Map of 1.600 areas .......................... 43

5.4 SAR(0.7) process over a RI-Map of 3.600 areas .......................... 44

5.5 SAR(0.7) process over a RI-Map of 7.200 areas ........................ 44

5.6 RI-Map using United States external boundary as initial polygon (3.304 polygons) ................................................................. 45

5.7 RI-Map using Colombia external boundary as initial polygon (1.124 polygons) ................................................................. 46

5.8 RI-Map using Italy external boundary as initial polygon (7.308) ... 48

5.9 RI-Map of 20.000 areas ......................................................... 49
5.10 Execution times of AMOEBA over regular tilings and \textit{RI-Maps} of 900 areas ................................................................. 50
LIST OF TABLES

2.1 Tilings examples .......................................................... 8
3.1 Number of maps from each group for a given number of areas. ..... 15
3.2 Average connectedness characteristics for real maps, regular lattices and Voronoi diagrams ....................................................... 16
4.1 Examples of RI-Maps of 400, 1.600 and 3.600 areas using different combinations of parameters ................................. 35
5.1 Average connectedness characteristics for simulated maps ........ 42
1. Introduction

A main task in order to propose regional science methodology is to guarantee a good behaviour of them over real data. This task becomes more difficult when we take into account the fact that the geographies used to test algorithms and methodologies could mislead their valuation in different ways such as the significance of hypothesis tests (Anselin and Moreno, 2003; Mur Lacambra, 1992), the computationally efficiency (Duque et al., 2010), or could violate some of the assumptions used to construct the mathematical theories as the assumption of stationarity in spatial econometrics (Hooper and Hewings, 1981). In order to avoid these consequences different types of tilings such as real maps from specific parts of the world, regular tilings, or Voronoi diagrams, are commonly used by regional scientists however because their topological characteristics are far of being considered as those of real maps then the consequences does not dissapear.

The most intuitive approach is using real maps divided by normative regions; e.g; Europe by countries, United States by counties, Spain by municipalities.\(^1\) In one hand these structures has random realistic connectedness characteristics but in the other hand they do not necessary represent the topological characteristics of different parts of the world, as example, Fig. 1.1 shows two set of 900 areas extracted from two different parts of the world, on the left from the normative division of United States by counties with an average number of neighbors of $5.81 \pm 0.96$ and maximum of neighbors of 9, on the right from Spain divided by municipalities with an average number of neighbors of $6.015 \pm 2.60$ and a maximum number of neighbors of 31. These differences are more critical because it is common to find that researchers usually use maps from the countries where they reside. Another disadvantage with this approach is that it could be difficult to construct large sets of maps, even more if different sizes are required. Last problem is

\(^1\)According to Duque et al. (2007) normative regions are defined as: “the expression of a political will; their limits are fixed according to the tasks allocated to the territorial communities, to the sizes of population necessary to carry out these tasks efficiently and economically, or according to historical, cultural and other factors”
usually faced with some sampling approaches (Anselin and Moreno, 2003).

Fig. 1.1: Randomly sampled maps of 900 areas

Regarding regular lattices, they are easy to be generated, do not present restrictions about the number of areas and their over simplified structure usually make it easier to summarize mathematical theories and explain the behaviour of tested methodologies (Whittle, 1954; Bartlett, 1975; Griffith, 1987). The disadvantages of using these tilings are that they make it necessary to set stricter assumptions when the tested methodologies are going to be used for real applications; e.g., the problem of spatial stationarity in STARMA models (Hooper and Hewings, 1981), improper conclusions of the properties of statistics in hypothesis testing (Anselin and Moreno, 2003; Mur Lacambra, 1992) and over qualification of computational efficiency of algorithms (Duque et al., 2010; Aldstadt and Getis, 2006). A middle point between regular tilings and real maps are Voronoi diagrams, which are theoretical structures with random properties that do not restrict the amount and sizes of maps, however as is showed in section 3.2 their topological characteristics are far of those of real maps.\(^2\)

Due to the inappropriateness of using these tilings and the importance of using general realistic structures to evaluate algorithms and methodologies in re-

\(^2\)Voronoi diagrams could be considered as an special case of a birth and growth stochastic process known as Johnson-Mehl tessellations when both, birth and growth rates, are equal for each area (Moller, 1995; Chiu, 2011)
An algorithmic approach for simulating realistic irregular tilings

In chapter 2 of this thesis the basic definitions of polygons and tilings are presented, proposing a consensus taxonomy of tilings, also we propose a set of indicators commonly used to characterize tilings in different fields such as graph theory (Garrison, 1964; Tinkler, 1972), geographical sciences (Gould, 1967; Boots, 1982, 1984, 1985), and metallography (Le Caer and Delannay, 1995; Aste et al., 1996; Peshkin et al., 1991; Le Caer and Delannay, 1993). In chapter 3 the relationship between tilings and regional science is exposed, latter the average values of the indicators of maps from different parts of the world is presented as the benchmark values and compared with the indicators of commonly used tilings. In chapter 4 a brief theoretical background of Stochastic Differential Equation and Fractal dimension is presented, the rest of the chapter propose two algorithms, the first one (MR-Polygon) to generate irregular polygons and the second (RI-Map) to generate tilings with similar characteristics of real maps using MR-Polygons. In chapter 5 the indicators of RI-Maps are obtained and compared with those of regular tilings, Voronoi diagrams and real maps. In chapter 6 and 7 some conclusions and ideas for future research are presented. Finally in chapter 8 is showed how to install the clusterpy and some examples are presented.
2. About tilings and space partitions

2.1 Space partition, consensus taxonomy of tilings

In order to define tessellations, also known as tilings, it is necessary to briefly introduce the concept of polygon, which is a plane figure enclosed by the set of finite straight line segments. Polygons can be categorized according to its boundaries, convexity and symmetry.

i) Respect to its boundary, a polygon can be classified as simple if it is formed just by one simple plain figure without holes (Fig. 2.1 a,b and c). Non-simple polygons, also known as complex, are all of those that have holes or are formed by multiple parts (Fig 2.1.d). The concept of complex polygons is commonly used in geography to define areas with lakes or islands.

ii) According to its convexity, polygons can be convex or non-convex. Polygons are convex if every pair of points in the polygon can be connected by a straight line without crossing the boundary of the polygon (Fig. 2.1 a and b), (Johnson, 2001), otherwise are non-convex (Fig. 2.1 c and d).

iii) Based on its symmetry, polygons on which all the angles have the same magnitude and all the sides are of equal length are regular (Fig. 2.1.a), otherwise, irregular (Fig. 2.1 b,c and d) (Johnson, 2001).

A tessellation, in a bidimensional space, is a set of polygons with no gaps and no overlaps covering a subspace or the entire space. Formally, a tessellation of any subspace $S \subseteq \mathbb{R}^n$ is the division into $k$ subsets $i \subseteq S$ such that $\cup_i = S$ and $\cap_i = \phi$ where $\phi$ is the empty set of $\mathbb{R}^n$ (Grunbaum and Shephard, 2011), hereinafter only cases where $n = 2$ are taken into account. This definition does not restricts the shapes of the polygons which form the tessellation (Ghyka, 2004; Penrose, 1974; Berger, 1966; Robinson and Berkeley, 1971).

---

3. Complex polygons do not refer to polygons which exist in the Hilbert plane (Coxeter, 1974).
4. Concave polygons are simple and non-convex.
5. Tilley (2006) imposes the condition that a tessellation must be formed by identically shaped polygons.
There exist different taxonomies of tilings, the use of one or another depends on the study field. In Fig. 2.2 consensus taxonomy is proposed classifying tilings according to the shape of its polygons, the spatial relationship between them, and the use, or not, of symmetric relationships to construct the tessellation.\footnote{An alternative category is proposed for tilings formed by fractal polygons that are informally defined by Mandelbrot (1982) as rough fragmented geometric shapes which could be infinitely divided into scalable parts.} Table 2.1 shows an example of each category of this consensus taxonomy.

i) According to the variety of the shapes of polygons forming the tiling: If all the polygons have the same shape, then the tiling is denoted as \textit{homomorphism}, otherwise \textit{polymorphism}.

ii) According to the regularity of the polygons forming the tiling, and the way that they intersect each vertex, considering a vertex as all the points of tiling that intersects three or more polygons: \textbf{Regular} tilings are formed by regular polygons in which all the vertex join the same arrangement of polygons Tilley
An algorithmic approach for simulating realistic irregular tilings

If the polygons are regular, but there are different configurations of vertexes, then the tiling is semi-regular, otherwise irregular (Ghyka, 2004).

iii) According to the existence of symmetric relationships within the tiling. Symmetrical relationships are three: Translation: the tiling is formed by translating a subset of areas; reflection: there are axes of reflection in the tiling; rotation: it is possible to obtain the same tiling after a rotation process of less than $2\pi$ (Radin, 1993). A symmetric tiling implies the presence of, at least, one symmetric relationship, otherwise the tiling is asymmetric.

iv) According to the symmetric relationship of translation. A tiling is periodic if and only if it implies the use of translation without rotation or reflection, otherwise aperiodic (Tilley, 2006).

2.2 Measuring the topological characteristics of tilings

A common framework in fields such as regional sciences and metallography is to use some sparse matrix indicators to summarize and compare the topological
Table 2.1: Tilings examples
characteristics of tilings through its contiguity matrix (Gould, 1967; Boots, 1982; Le Caer and Delannay, 1995; Aste et al., 1996; Peshkin et al., 1991).\(^7\) In our case six indicators are used, each of which is briefly described below.

The first three indicators are the maximum \((M_n)\), minimum \((m_n)\) and average \((\mu_1)\) number of neighbors per area. The fourth indicator, Sparseness \(S\), is the percentage of non-zero elements of the contiguity matrix. The fifth indicator is the first eigenvalue of \(W\) \((\lambda_1)\) which, due to its ability to summarize different aspects of the structures, were originally used to study the connectivity matrix of networks in graphs theory (Garrison, 1964; Tinkler, 1972), and subsequently adopted by regional scientist to characterize topological characteristics of the contiguity matrix (Gould, 1967; Boots, 1982, 1984, 1985). The last indicator is the variance of the number of neighbors per polygon \((\mu_2)\) which is commonly used as a measurement of the spatial disorder within a tiling.

\(^7\)For more information about contiguity matrix see Anselin (1988)
3. Tilings in regional science

Within the field of regional science tilings are commonly used with two purposes, in one hand, real maps are used to study real phenomena; e.g. analyze spatial patterns, confirm spatial relationships between variables, detect spatio-temporal regimes within a data panel. On the other hand, they are used to evaluate the behaviour of statistical tests (Anselin and Moreno, 2003; Mur Lacambra, 1992); algorithms (Duque et al., 2010); and topological characteristics of maps (Aste et al., 1996; Le Caer and Delannay, 1995, 1993). In the last case it is necessary to use sets of tilings that satisfy some requirements imposed by the regional scientist; e.g, number of areas, regularity or irregularity of the polygons, number of maps, among others. To accomplish this it is common to use polymorphism irregular aperiodic asymmetric, such as real maps and Voronoi diagrams; or homomorphism regular periodic symmetric such as regular lattices, as geographical base for real or simulated data. In next section we will be focused on the second use of tilings.

3.1 Commonly used tilings

The process of propound an appropriate set of tilings to evaluate methodologies is not always a trivial task, specially when different topological characteristics may differ even within tilings considered in the same category; e.g; real maps from different parts of the world; regular lattices formed by hexagons or squares. An ideal scenery is to be able to collect a set of tilings with representative characteristics of maps from different parts of the world, which is not always possible because the availability of maps and its number of areas. Different strategies are followed in order to face this problem, as an example a common practice is to use a map with a large number of areas and algorithmically construct the desired number of maps with the desired number of areas. The problem with these approaches is that different parts of the world could have different topological characteristics and hence different results and conclusions could be drawn if the base map is not properly selected. This situation becomes more critical when reviewing the litera-
ture is found that those topological characteristics are not taken into account and the selection of the large map is influenced by the location where the researcher applies; e.g; Duque et al. (2011a) uses maps from Colombia and United States to propose the regionalization algorithm MAX-P; Anselin and Moreno (2003) use United States maps to compare the performance of an statistical test over regular and irregular lattices; Mur Lacambra (1992) compares different methods to detect spatial autocorrelation over two artificial contiguity matrix and the map of Spain by regions. Fig. 3.1 shows the distribution of the Average number of neighbors, the first eigenvalue and the Aboav-Weaire coefficient of maps generated using United States by counties and Spain by NUTS3. In summary it is possible to conclude that maps of Spain are less connected and the number of neighbors are more correlated than maps of United States. As an example, selecting maps of United States instead of Spain could considerably increase the execution time of algorithms which its computational complexity increase with the number of neighbors (Aldstadt and Getis, 2006; Duque et al., 2010)

Because the difficulty of collecting an appropriate map and to take advantage of more controlled structures some authors use alternative approaches such as regular tilings or Voronoi diagrams. Regarding to regular lattices, they are easily generated, do not present restrictions about the number of areas and its over simplified structure usually make it easier to summarize mathematical theories and explain the behaviour of tested methodologies (Whittle, 1954; Bartlett, 1975; Griffith, 1987). The problem of using regular tilings is that they have some desirable theoretical properties that usually are not present in real maps and hence some assumptions or requirements need to be guaranteed before studding real phenomena using methodologies tested over them; e.g; the problem of spatial stationarity in STARMA models (Hooper and Hewings, 1981), improper conclusions of the properties of statistics in hypothesis testing, (Anselin and Moreno, 2003; Mur Lacambra, 1992), over qualification of computational efficiency of algorithms (Duque et al., 2010; Aldstadt and Getis, 2006). Regarding Voronoi diagrams, as will be shown in section 3.2, although they have random topological properties they are far from those of real maps.
Fig. 3.1: Topological differences of maps from Spain and United States

3.2 Topological comparison of tilings

By now it is clear the advantages and disadvantages of the alternative approaches (regular tilings and Voronoi diagrams) to select a set of tilings to evaluate a methodology, however the question about if those categories of tilings are truly different remains open. To answer this, we propose a benchmark of topological indicators that a tiling must reproduce in order to be considered real and then compare the values of regular lattices and Voronoi diagrams. To obtain the bench-
mark the indicators proposed in section 2 are calculated over 1,000 instances of maps with 100, 400, 900, 1,600, 2,500, 3,600 areas, obtained from a sampling process over a world map at the smaller administrative division available in Hijmans et al. (2011), Fig. 3.2 illustrates the sampling process and Table 3.1 shows the number of maps sampled from each continental area. For Voronoi diagrams the same number of tilings with the same number of areas are used. For regular lattices just one instance is enough. Table 3.2 shows the expected value of the indicators for each type of tiling grouped by number of areas, additionally, for real maps and Voronoi diagrams the standard deviation is calculated and presented bellow the expected values. In order to avoid the effect caused by the boundary to the values of $M_n, m_n, \mu_1, \mu_2$ the boundary areas are allowed to be neighbors of other areas but are not considered to calculate the indicators, however for $S$ and $\lambda_1$ all areas are used because they are matrix properties.

Fig. 3.2: Base map and example of a random irregular maps obtained from it.

For regular lattices it is evident that they are not capable to emulate the Voronoi diagrams where generated using random uniform points.
An algorithmic approach for simulating realistic irregular tilings

<table>
<thead>
<tr>
<th>Group</th>
<th>Number of areas</th>
<th>81</th>
<th>100</th>
<th>400</th>
<th>900</th>
<th>1,600</th>
<th>2,500</th>
<th>3,600</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: America</td>
<td>28,041</td>
<td>240</td>
<td>240</td>
<td>240</td>
<td>240</td>
<td>244</td>
<td>244</td>
<td>244</td>
</tr>
<tr>
<td>2: Europa</td>
<td>26,245</td>
<td>225</td>
<td>225</td>
<td>225</td>
<td>225</td>
<td>229</td>
<td>229</td>
<td>229</td>
</tr>
<tr>
<td>3: Africa</td>
<td>35,943</td>
<td>308</td>
<td>308</td>
<td>308</td>
<td>308</td>
<td>313</td>
<td>313</td>
<td>313</td>
</tr>
<tr>
<td>4: Asia</td>
<td>24,503</td>
<td>210</td>
<td>210</td>
<td>210</td>
<td>210</td>
<td>214</td>
<td>214</td>
<td>214</td>
</tr>
<tr>
<td>5: Oceania</td>
<td>1.880</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>17</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Total</td>
<td>116,612</td>
<td>1,000</td>
<td>1,000</td>
<td>1,000</td>
<td>1,000</td>
<td>1,000</td>
<td>1,000</td>
<td>1,000</td>
</tr>
</tbody>
</table>

Table 3.1: Number of maps from each group for a given number of areas.

topological characteristics of real maps in any of the indicators because they are absolutely ordered tilings, $\mu_2 = 0$, with 4 and 6 neighbors per area over the entire structure. Respect to $\lambda_1$ and $S$ it is also important to note that regular tilings formed by hexagons are more connected than real maps while those formed by squares are less connected. Also note that in both cases the variations of the indicators while the number of areas increase is much lower than the variations of real maps, which means that its topological characteristics are not highly affected by the size of the map.

About Voronoi diagrams it is found that they are not capable to generate atypically connected areas, values of $M_n$ tending to 12 and $m_n$ greater than 3. Regarding $\mu_1$ they have a similar value of real maps, $\mu_1$ close to 6 which is supported by the tenet of previous studies which suggest irregular tilings have an average number of neighbors close to 6 (Weaire and Rivier, 2009). From the values $\mu_2$, close to 1.7, is possible to infer that they are more ordered than real maps. Additionally note that the magnitude of the change of the indicators while the size of the maps increase is smaller than real maps, showing that the topological characteristics of Voronoi diagrams are more stable for different sizes of maps. Finally the standard deviations evidence that the random properties of Voronoi diagrams are less disperse than on real maps.
Table 3.2: Average connectedness characteristics for real maps, regular lattices and Voronoi diagrams

<table>
<thead>
<tr>
<th></th>
<th>Number of areas</th>
<th>81</th>
<th>100</th>
<th>400</th>
<th>900</th>
<th>1,600</th>
<th>2,500</th>
<th>3,600</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Real Maps</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$M_n$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$m_n$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\mu_1$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\mu_2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$S$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\lambda_1$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Reg. Lattice (Squares)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$M_n$</td>
<td></td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>$m_n$</td>
<td></td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>$\mu_1$</td>
<td></td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>$\mu_2$</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$S$</td>
<td></td>
<td>4.44</td>
<td>3.64</td>
<td>0.95</td>
<td>0.43</td>
<td>0.24</td>
<td>0.16</td>
<td>0.11</td>
</tr>
<tr>
<td>$\lambda_1$</td>
<td></td>
<td>3.80</td>
<td>3.84</td>
<td>3.96</td>
<td>3.98</td>
<td>3.99</td>
<td>3.99</td>
<td>3.99</td>
</tr>
<tr>
<td><strong>Reg. Lattice (Hexagons)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$M_n$</td>
<td></td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>$m_n$</td>
<td></td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>$\mu_1$</td>
<td></td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>$\mu_2$</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$S$</td>
<td></td>
<td>6.30</td>
<td>5.19</td>
<td>1.39</td>
<td>0.64</td>
<td>0.36</td>
<td>0.23</td>
<td>0.16</td>
</tr>
<tr>
<td>$\lambda_1$</td>
<td></td>
<td>5.55</td>
<td>5.62</td>
<td>5.88</td>
<td>5.94</td>
<td>5.96</td>
<td>5.97</td>
<td>5.98</td>
</tr>
<tr>
<td><strong>Voronoi Diagrams</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$M_n$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$m_n$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\mu_1$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\mu_2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$S$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\lambda_1$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3.2: Average connectedness characteristics for real maps, regular lattices and Voronoi diagrams.
4. Generating realistic tilings

In this section we propose a procedure to generate polymorphism irregular aperiodic asymmetric tilings with similar characteristics of real maps. To attain this we divide the section in four parts, the first and second parts briefly introduce the theoretical concepts which support the algorithms, the third part introduces a novel way to generate irregular polygons based on a mean reverting process in polar coordinates and the fourth part proposes a recursive procedure to generate tilings with similar topological characteristics of real maps using the irregular polygons of the third part.

4.1 Introduction to the mean reverting process

In order to propose the $MR$ – $Polygon$ algorithm it is necessary to briefly introduce some formal background to state the mean reverting process. Next some basic concepts about Brownian motion and stochastic differential equations are introduced. The following definitions and concepts are constructed based on Mao (1997).

4.1.1 Brownian motion

**Definition 4.1.1.** Let $(\omega, \mathcal{F}, P)$ a probability space with a filtration $\{\mathcal{F}_t\}_{t \geq 0}$. A standard unidimensional Brownian motion $\{B_t\}_{t \geq 0}$ is a continuous real valued process $\mathcal{F}_t$ – adapted with the following properties.

1. $B_0 = 0$

2. $\{B_t\}_{t \geq 0}$ has stationary increments, which means that $B_t - B_s$ is identically distributed to $B_{t+h} - B_{s+h}$ for each $s, t \in T$ and each $s + h, t + h \in T$

3. $\{B_t\}_{t \geq 0}$ has independent increments, which means that for each $t_i \in T$ with $t_1 < t_2 < \cdots < t_n$ and $n \geq 1$, $B_{t_2} - B_{t_1}, \ldots, B_{t_n} - B_{t_{n-1}}$ are independent random variables.
4. For $0 \leq s \leq t < \infty$, the increment $B_t - B_s$ follows a normal distribution with mean 0 and variance $t - s$.

5. $\{B_t\}_{t \geq 0}$ is continuous and not differentiable at any point.

A Brownian motion has other important properties that are important to mention. Among them one of the most important is the self similarity which means that a Brownian motion preserves, at different scales, the patterns of a Brownian instance. It is important to note that using this concept it is possible to establish an important relation between Brownian motions and fractals (see section 4.2).

1. A Brownian motion is a Gaussian process.

2. The covariance of a Brownian motion is $\text{cov}(B_s, B_t) = \min(s, t)$.

3. A Brownian motion is $0.5 - \text{selfsimilar}$ which means

$$\left(T^{\frac{1}{2}}B_{t_1}, \ldots, T^{\frac{1}{2}}B_{t_n}\right) \overset{d}{=} \left(B_{T_{t_1}}, \ldots, B_{T_{t_n}}\right)$$

4.1.2 Stochastic differential equations

In this section a brief introduction to stochastic differential equations is made. To attain this, first a general n-dimensional stochastic differential equation and its solutions are showed. Second the theorems of existence and uniqueness of solutions are stated. Third the Linear Stochastic Differential Equation is presented. Finally the solution of the linear equations in strict sense are presented and related with the mean reverting process used to propose the MR-Polygon algorithm.

Let $(\omega, \mathcal{F}, \mathbb{P})$ be a complete probability space with a filtration $\mathcal{F} \geq 0$. Let $B(t) = (B_1(t), B_2(t), \ldots, B_m(t))^T, t \geq 0$ an m-dimensional Brownian motion over the probability space. Let $t_0 < T$ between $0, \infty$. Let $x_0$ be an $\mathcal{F}_{t_0}$-measurable random variable such that $E[x_0]^2 < \infty$. Let $f : \mathbb{R}^2 \times [t_0, T] \rightarrow \mathbb{R}^n$ and $g : \mathbb{R}^n \times [t_0, T] \rightarrow \mathbb{R}^{n \times m}$, both of them Borel-measurable.
Consider the n-dimensional Stochastic Differential Equation given by (4.1) with initial value \( x(t_0) = x_0 \). By definition this (4.1) is equivalent to the stochastic integral equation (4.2).

\[
dx(t) = f(x(t), t)dt + g(x(t), t)dB(t); \quad t_0 \leq t < T \tag{4.1}
\]

\[
x(t) = x_0 + \int_{t_0}^{t} f(x(s), s)ds + \int_{t_0}^{t} g(x(s), s)dB(s); \quad t_0 \leq t < T \tag{4.2}
\]

**Definition 4.1.2.** A \( \mathbb{R}^n \) valued stochastic process \( \{x(t)\}_{t_0 \leq t \leq T} \) is known as solution of (4.1) if the following conditions are true:

I) \( \{x(T)\} \) is continuous and \( \mathcal{F}_t \) adapted

II) \( \{f(x(t), t)\} \in \mathcal{L}^1([t_0, T]; \mathbb{R}^n) \) and \( \{g(x(t), t)\} \in \mathcal{L}^2([t_0, T]; \mathbb{R}^{n \times m}) \)

III) Equation (4.2) is true for each \( t \in [t_0, T] \)

It is said that \( \{x(t)\} \) is unique if any other solution \( \{\bar{x}(t)\} \) is exactly the same as \( \{x(t)\} \).

### 4.1.2.1 Existence and Uniqueness of solutions

**Theorem 4.1.3.** Suppose the existence of two positive constants \( K \) and \( \bar{K} \) that meets the next conditions, then exists a unique solution of equation (4.1) and it belongs to \( \mathcal{M}^2([t_0, T]; \mathbb{R}^n) \).

**Lipschitz criterion** For every \( x, y \in \mathbb{R}^n \) and \( t \in [t_0, T] \)

\[
|f(x, t) - f(y, t)|^2 \vee |g(x, t) - g(y, t)|^2 \leq \bar{K}|x - y|^2
\]

**Lineal growth condition** For every \( (x, t) \in \mathbb{R}^n \times [t_0, T] \)

\[
|f(x, t)|^2 \vee |g(x, t)|^2 \leq K(1 + |x|^2)
\]
**Theorem 4.1.4.** Suppose that the linear growth conditions are met, but the Lipschitz criterion is replaced by the **Local Lipschitz Criterion**:

For each integer number $n \geq 1$, exists a positive constant $K_n$ such that for every $t \in [t_0, T]$ and for every $x, y \in \mathbb{R}^n$ with $|x| \vee |y| \leq n$,

$$|f(x, t) - f(y, t)|^2 \vee |g(x, t) - g(y, t)|^2 \leq K_n|x - y|^2$$

Then exists a unique solution $x(t)$ to (4.1) in $\mathcal{M}^2([t_0, T]; \mathbb{R}^n)$.

**Theorem 4.1.5.** Suppose that the Lipschitz local criterion is met, but the lineal growth condition is replaced by the **Monotone condition**:

Exists a positive constant $K$ such that for every $(x, y) \in \mathbb{R}^n \times [t_0, T]$,

$$x^T f(x, t) + \frac{1}{2} |g(x, t)|^2 \leq K(1 + |x|^2)$$

Then exists a unique solution $x(t)$ to (4.1) in $\mathcal{M}^2([t_0, T]; \mathbb{R}^n)$.

On one hand the Lipschitz local criterion ensures that the solution exists in $[t_0, \tau_\infty]$, where $\tau_\infty = \lim_{n \to \infty} \tau_n$. On the other hand the monotone condition ensures that $\tau_\infty = T$, which means that the solution exists in the interval $[t_0, T]$.

Consider the Stochastic Differential Equation given by (4.3) with initial value $x(T_0) = x_0$. If the conditions of existence and uniqueness are met for every finite subinterval $[t_0, T]$ into $[t_0, \infty]$, then it has a unique solution, called **global solution**, into $[t_0, \infty]$.

$$dx(t) = f(x(t), t)dt + g(x(t), t)B(t) \quad (4.3)$$

**Theorem 4.1.6.** Suppose that for each real number $T > t_0$ and for each integer $n \geq 1$, exists a positive constant $K_{T,n}$ such that for every $t \in [t_0, T]$ and every $x, y \in \mathbb{R}^n$ with $|x| \vee |y| \leq n$ then $|f(x, t) - f(y, t)|^2 \vee |g(x, t) - g(y, t)|^2 \leq K_{T,n}|x - y|^2$.

Also suppose that for every $T > t_0$, exists a positive constant $K_T$ such that for every $(x, t) \in \mathbb{R}^n \times [t_0, T]$,

$$x^T f(x, t) + \frac{1}{2} |g(x, t)|^2 \leq K_T(1 + |x|^2) \quad (4.4)$$
Then exists a unique global solution $x(t)$ of (4.3) and it belongs to $\mathcal{M}^2([t_0, T]; \mathbb{R}^n)$

### 4.1.2.2 Lineal Stochastic Differential Equations

Consider the $n$-dimensional stochastic differential equation given by (4.5) over $[t_0, T]$ where $F(\cdot), G(\cdot)$ are matrix valued functions of $n \times n$, $f(\cdot), g(\cdot)$ are functions with values in $\mathbb{R}^n$ and $B(t) = (B_1(t), B_2(t), \ldots, B_m(t))^T$ is a $m$-dimensional Brownian motion. It is said that (4.5) is homogeneous if $f(t) = g(t) = \cdots = 0$, it is said lineal in strict sense if $G_1(t) = G_2(t) = \cdots = 0$ and it is said autonomous if $F, f, G_k, g_k$ are $t$ independents. Also it is assumed that $F, f, G_k, g_k$ are Borel-measurable and bounded over $[t_0, T]$. Under this framework, due to the existence and uniqueness theorem, equation (4.5) has unique solution in $\mathcal{M}^2([t_0, T]; \mathbb{R}^n)$ for each initial value $x(t_0) = x_0$.

$$dx(t) = (F(t)x(t) + f(t))dt + \sum_{k=1}^{m}(G_k(t)x(t) + g_k(t))dB_k(t)$$  \hspace{1cm} (4.5)

In this thesis we are only interested in linear equations in strict sense because, as be shown latter, the Mean Reverting process used to propose the MR-Polygon is included in this category. Consider the Linear Stochastic Differential equation in strict sense given by (4.6) over $[t_0, T]$ with initial value $x(t_0) = x_0$.

$$dx(t) = (F(t)x(t) + f(t))dt + \sum_{k=1}^{m}g_k(t))dB_k(t)$$ \hspace{1cm} (4.6)

The linear homogeneous equation for (4.6) is the ordinary differential equation given by

$$dx(t) = F(t)x(t)dt,$$ \hspace{1cm} (4.7)

And the solution of (4.6) is given by

$$x(t) = \phi(t) \left( x_0 + \int_{t_0}^{t} \phi^{-1}(s)f(s)ds + \sum_{k=1}^{m} \int_{t_0}^{t} \phi^{-1}(s)g_k(s)dB_k(s) \right)$$ \hspace{1cm} (4.8)
Particularly when $F(t)$ is $t$ independent, which means $F(t) = F$ is a constant matrix of $n \times n$ the fundamental matrix $\phi(t)$ has the form $\phi(t) = e^{F(t-t_0)}$ and it inverse $\phi^{-1}(t) = e^{-F(t-t_0)}$. Then equation 4.6 is solved by

$$x(t) = e^{F(t-t_0)}\left( x(t_0) + \int_{t_0}^{t} e^{-F(s-s_0)} f(s) ds + \sum_{k=1}^{m} \int_{t_0}^{t} e^{-F(s-s_0)} g_k(s) dB_k(s) \right)$$

(4.9)

A mean reverting process is a stochastic process that takes values following a long term tendency with presence of short term variations where the value of the process $x$ at the moment $t$ is given by the solution of the stochastic differential equation (4.10), where $\mu$ is the long term tendency, $\alpha$ is the mean reversion speed, $\sigma$ is the gain on the diffusion term, $x(t_0)$ is the value of the process when $t = 0$ and $\{B_t\}_{t \geq 0}$ is an unidimensional Brownian motion. As can be seen (4.10) is formally a linear stochastic equation in strict sense over $[t_0, T]$, being $F(t) = -\alpha$, $f(x) = \alpha \mu$, $g_t = \sigma$ and is in just one dimension, which means $m = 1$.

$$dX_t = \alpha(\mu - X_t) dt + \sigma dB_t \quad (4.10)$$

Under this framework, using (4.9), the solution of (4.10) with $\phi = e^{F(t-t_0)} = e^{-\alpha}$ is given by

$$x(t) = e^{-\alpha(t-t_0)}\left( x(t_0) + \int_{t_0}^{t} e^{\alpha(s-s_0)} \alpha \mu ds + \int_{t_0}^{t} e^{\alpha(s-s_0)} \sigma dB(s) \right), \quad (4.11)$$

Particularly to propose the MR-Polygon algorithm we can assume $t_0 = 0$ and then
An algorithmic approach for simulating realistic irregular tilings

\[ x(t) = e^{-\alpha t} \left( x_0 + \int_0^t e^{\alpha s} \alpha \mu ds + \int_0^t e^{\alpha s} \sigma dB(s) \right) \]

\[ = x_0 e^{-\alpha t} + e^{-\alpha t} \int_0^t e^{\alpha s} \alpha \mu ds + e^{-\alpha t} \int_0^t e^{\alpha s} \sigma dB(s) \]

\[ = x_0 e^{-\alpha t} + \frac{e^{-\alpha t} e^{\alpha t} \alpha \mu}{\alpha} + \int_0^t e^{-\alpha(t-s)} \sigma dB(s) \]

\[ = x_0 e^{-\alpha t} + \mu + \int_0^t e^{-\alpha(t-s)} \sigma dB(s). \] (4.12)

Evaluating (4.12) when \( t_0 = 0 \) is obtained the value of \( x_0 \)

\[ x(0) = x_0 + \mu \rightarrow x_0 = x(0) - \mu, \]

Substituting again into (4.12) the solution is given by

\[ x(t) = \mu + (x(0) - \mu)e^{-\alpha t} + \mu + \int_0^t e^{-\alpha(t-s)} \sigma dB(s). \forall t \in [0, T] \] (4.13)

For practical purposes generating polygons the use Euler approximation given by (4.14) is used, where \( \epsilon_t \sim N(0, 1) \),

\[ X_t = X_{t-1} + \alpha (\mu - X_{t-1}) \Delta_t + \sigma \sqrt{\Delta_t} \epsilon_t \] (4.14)

4.2 Introduction to the fractal dimension

In this section the concept of fractal dimension as a way to measure a subjective feeling of how densely a fractal occupies a metric space is introduced, understanding a fractal as rough fragmented geometric shape which could be infinitely divided into scalable parts (Mandelbrot, 1982). The fractal dimension is a very important measure because it is a way to connect fractals with real world data, for example it is possible to get the fractal dimension of a coastline, an irregular polygon, a cloud, a tree or the colors emitted by the sun. Next the fractal dimension is
formally introduced, following Barnsley (2000).

Let \((X, d)\) be a complete metric space. Let \(A \in \mathcal{H}\) be a nonempty compact subset of \(X\), being \(\mathcal{H}\) the space whose points are the compact subsets of \(X\), other than the empty set. Let \(\epsilon > 0\). Let \(B(x, \epsilon)\) denote the closed ball of radius \(\epsilon\) centered at \(x \in X\). Define \(N(A, \epsilon)\) the least number of closed balls of radius \(\epsilon\) needed to cover the set \(A\). In other words,

\[
N(A, \epsilon) = \text{the smallest positive integer } M \text{ such that } A \subset \bigcup_{n=1}^{M} B(X_n, \epsilon),
\]

For a set of points \(x_n : n = 1, 2, 3, \ldots M \subset X\). To prove that \(N(A, \epsilon)\) exists, surround each point \(x \in A\) by an open ball of radius \(\epsilon\) covering \(A\) using opens sets. Due to the compactness of \(A\) that cover has a finite sub-cover of \(\hat{M}\) open balls. The closure of those balls we have a cover of \(\hat{M}\) closed balls of radius \(\epsilon\). Let \(C\) the covers of \(A\) with at most \(\hat{M}\) balls of radius \(\epsilon\), then \(C\) contains at least one element. Let \(f : C \to 1, 2, 3, \ldots, \hat{M}\) be defined by \(f(c) = \text{number of balls in the cover } c \in C\). Then \(f(c) : c \in C\) is a set of positive integers having at least an integer \(N(A, \epsilon)\).

It is said that a set \(A\) has fractal dimension \(D\) if for some positive number \(C\) \(N(A, \epsilon) \approx \epsilon^{-D}\). Where \(\approx\) is used as follows. Let \(f(\epsilon)\) and \(g(\epsilon)\) be real valued functions, then it is said that \(f(\epsilon) \approx g(\epsilon)\) if \(\lim_{\epsilon \to 0} \frac{\log(f(\epsilon))}{\log(g(\epsilon))} = 1\). Solving for \(D\) then

\[
D \approx \frac{\log(N(A, \epsilon)) - \log(C)}{\log(\frac{1}{\epsilon})} \approx \frac{\log(N(A, \epsilon))}{\log(\frac{1}{\epsilon})} - \frac{\log(C)}{\log(\frac{1}{\epsilon})}
\]

(4.15)

Using the definition of “\(\approx\)” it is clear that the second term in the right hand side of (4.15) tends to zero and then (4.16).

\[
D \approx \frac{\log(N(A, \epsilon))}{\log(\frac{1}{\epsilon})}
\]

(4.16)
Definition 4.2.1. Let $A \in \mathcal{H}(\mathbb{X})$ where $(\mathbb{X}, d)$ is a metric space. For each $\epsilon > 0$ needed to cover $A$. If (4.17) exists, then it is called the fractal dimension of $A$.

$$D = \lim_{\epsilon \to 0} \left( \frac{\log(N(A, \epsilon))}{\log(\frac{1}{\epsilon})} \right)$$ (4.17)

In our case we are interested in the fractal dimension of the irregular boundary of a polygon, in this case then $A$ is the boundary, $\epsilon$ is the length of the measurement instrument, and $N(A, \epsilon)$ is the length of $A$ using a measurement instrument of length $\epsilon$, denoted as $L(A, \epsilon)$. Under this framework the fractal dimension is restated as (4.18).

$$D = \lim_{\epsilon \to 0} \left( \frac{\log(L(A, \epsilon))}{\log(\frac{1}{\epsilon})} \right)$$ (4.18)

4.3 Mean reverting areas (MR-Polygons)

4.3.1 Algorithm

The problem of characterizing the shape of irregular polygons, as those that comes from processes of urban growth or administrative borders, is commonly faced in two ways, evaluating its similitude with a circle (Haggett, 1977) or describing its boundary roughness with the fractal dimension (Batty and Longley, 1994; Frankhauser, 1998). Chen (2011) established a relationship between both ways. In our case both of them are combined, the first approach to propose a way to create irregular polygons and the second to parametrize them under the concept of fractal dimension.

To compare irregular polygons with circles different classic indexes are used such as elongation ratio (Weeitty, 1969), form ratio (Horton, 1932), circularity ratio (Miller, 1953), compactness ratio (Gibbs, 1961; Cole, 1964; Richardson, 1961), ellipticity index (Stoddart, 1965) and the radial shape index (Boyce and Clark, 2012). According to Chen (2011) these indexes are based on the minimum circumcircle and the circle with equal area to the irregular polygon. Under this
Approach an irregular polygon can be interpreted as an irregular boundary which follows a circle with a representative random noise. This interpretation guides us to use a mean reverting process, equation (4.14), to create irregular polygons.

Following the procedure showed in Pseudocode 1 it is possible to construct an irregular polygon $P$ in polar coordinates where the points follow a circle with radius $\mu$ and where the distance, following the process, between any two points of $X$ is exactly the same as the distance, following the frontier in counterclockwise direction, between their equivalent points in $P$. Angle $\phi_1$ and $\Delta_r$ of Pseudocode 1 are the result of solving the geometric problem showed in Fig. 4.1 in order to define (4.19). Points of $P$ are denoted as $P_\theta$ with $\theta$ between 0 and $2\pi$. Regarding the effect of the parameters of $X$ on $P$; $\alpha$ is the speed at which the process revert to the circle with radius $\mu$, $\sigma$ is the scaling factor of the irregularity of the polygon, $X_0$ is the initial radius. $\Delta_t$ can not be directly interpreted but it is affecting the magnitude of the angular steps of $P$.

$$P_{\theta+\phi_1} = \begin{cases} P_\theta + \Delta_R & \text{if } X_{t+\Delta_t} \geq X_t \\ P_\theta - \Delta_R & \text{if } X_{t+\Delta_t} < X_t. \end{cases}$$

(4.19)
Pseudocode 1: MEANREVERTINGAREA

PARAMETERS:
\( \alpha \): Mean reversion speed
\( \sigma \): Noise scaling factor
\( \mu \): Long term tendence
\( X_0 \): Initial point
\( \Delta_t \): Time step

\[ X_{t-\Delta_t} = X_0, \text{ Initial point of the mean reverting process} \]
\[ P = \{(0, X_0)\}, \text{ Irregular polygon in polar coordinates} \]

while \( \theta < 2\pi \) do
\[
\begin{align*}
\epsilon_t & \leftarrow \text{N}(0,1) \\
X_t &= X_{t-\Delta_t} + \alpha(X_{t-\Delta_t} - \mu)\Delta_t + \sigma\sqrt{\Delta_t}\epsilon_t \\
d & \leftarrow \text{distance between } X_t \text{ and } X_{t-\Delta_t} \\
R_{\theta} & \leftarrow \text{Last radius of the irregular polygon} \\
\phi_1 &= \arccos\left(\frac{2R_{\theta} - \Delta^2}{2R_{\theta}^2}\right) \\
\text{if } X_t \geq X_{t-\Delta_t} \text{ then} \\
\Delta_R &= d \left( \cos\left(\arcsin\left(\frac{\Delta^2}{d}\cos\left(\frac{\phi_1}{2}\right)\right) - \sin\left(\frac{\phi_1}{2}\right)\frac{\Delta^2}{d}\right)\right) \\
\text{else} \\
\Delta_R &= -d \left( \cos\left(\arcsin\left(\frac{\Delta^2}{d}\cos\left(\frac{\phi_1}{2}\right)\right) + \sin\left(\frac{\phi_1}{2}\right)\frac{\Delta^2}{d}\right)\right) \\
R_{\theta+\phi_1} &= R_{\theta} + \Delta_R \\
\text{Add } (\theta + \phi_1, R_{\theta+\phi_1}) \text{ to } P \\
\text{increase } \theta \text{ in } \phi_1
\end{align*}
\]

Add last point of \( P \) to \( (0, X_0) \)
return \( P \)

4.3.2 Parametrization of MR-Polygons

The second approach to characterize irregular polygon is through the fractal dimension \( D \) of its irregular boundary, which is a number between 1, for smooth boundaries, and 2 for rough boundaries. In general, an object is considered as fractal if it is dotted with irregular characteristics that are present at different scales of study (Mandelbrot, 1982). The concept of fractal dimension of boundaries refers to Richardson (1961) who infers that the length \( L \) of an irregular boundary, like a coast, is given by (4.20) where \( \epsilon \) is the length of the measurement instrument, \( \hat{C} \) is a constant and \( D \) is the fractal dimension of the irregular shape. For practical purposes \( D \) is obtained using (4.21) and is given by 1 minus the slope of \( \log(L(\epsilon)) \), this procedure is commonly known as the Richardson plot. For the formal framework of the fractal dimension see section 4.2. Note that (4.21) uses \( \hat{C} \) and \( (1 - D) \) instead of the constant \( C \) and \( -D \) as the formal definition, this is obtained making \( \hat{C} = \frac{C}{\epsilon} \).
Practically in all cases the Richardson plot can be explained with two line segments having different slopes, then two fractal dimensions can be obtained; textural, for small scales, and structural, for large scales (Kindratenko and Treiger, 1996). As an example, Fig. 4.2, shows a segment of the United States east coast taken from Google maps on two resolutions. Note that while the resolution increase some irregularities that were imperceptible on low resolution become visible. In this sense it may be said that irregularities at low resolution define the general shape and are related with structural dimension, while the irregularities at high resolution give the noise and are related with textural dimension. Because of this is that regional scientists are used to use highly sampled maps which preserves the general shape but removes the little variations looking for some benefits without changing the topological configuration of maps (Douglas, 1973).

In the field of stochastic signals some approaches based on different approximations of the length have been made in order to characterize them through its fractal dimension. In our case an experimental approach based on the fractal di-
mension of real polygons is proposed to select an appropriated combination of the
d parameters $\alpha$ and $\sigma$ that generates realistic polygons, because our interest is on
general shape more than little variations only structural dimension is taken into
account.\footnote{To calculate the structural dimension we use the EXACT procedure (Allen et al., 1995) with
a is a small value of $\Delta t$, later both dimension were divided using a k-means clustering algorithm
over the points of the Richardson plot.} The experiment is divided in two parts, in the first part the frequency
histogram of fractal dimensions of the real polygons is constructed. In the sec-
ond part we propose a range of possible values of $\alpha$ and $\sigma$, given $\mu, X_0, \Delta t$, that
generates fractal dimensions close to the obtained in the first part. Because the
level of the long term tendency, $\mu$, does not affect the length of $X$ and Pseudocode
1 guarantee that length is preserved then $\mu$ can be defined as constant without
affecting the fractal dimension, hereinafter it is assumed $\mu = X_0 = 10$. Regarding
$\Delta t$ it must be a small number in order to properly infer both fractal dimensions,
from now on it is defined as 0.001.

The empirical distribution of the fractal dimension of the irregular polygons
is calculated over a random sample of 10,000 polygons from the world map used
in section 3. The result of this empirical distribution is showed in Fig. 4.3.

![Empirical frequency distribution of structural fractal dimensions of
real polygons.](image)

To summarize the fractal dimension of the stochastic polygons obtained with
Pseudocode 1 a surface, of the average dimensions, is generated while both un-
known parameters, $\alpha$ and $\sigma$, changes from 0.01 to 5 with steps of 0.1, Fig 4.4. Based on the surface it can be inferred that fractal dimension is mainly affected by changes to $\sigma$ specially when looking at small dimensions, additionally it is possible to guarantee that fractal dimensions close to 1.23 are obtained when $\sigma$ takes values between 1.2 and 1.5 regardless of the value of $\alpha$. Fig 4.5 shows some examples of polygons using different values of $\alpha$ and $\sigma$. Note that polygons of the middle row, realistic combinations of $\alpha$ and $\sigma$, look similar between them which confirms the conclusions made about the influence of the parameters to the structural dimension, additionally in the same figure, both, original (gray line) and sampled (black line) polygons are showed reinforcing the fact that sampling a polygon does not affects the structural dimension. From now on and looking for computational efficiency only sampled polygons are used.

![Fractal dimension of simulated polygons in function of $\alpha$ and $\sigma$.](image)

**Fig. 4.4:** Fractal dimension of simulated polygons in function of $\alpha$ and $\sigma$.

### 4.3.3 MR-Polygons with python

Using clusterpy is possible to create *MR-Polygons* with just a few lines of code. Next some examples are presented.\(^{10}\) Due to the objective of this project no

\(^{10}\)To install clusterpy please follow the instructions of section 8.1
An algorithmic approach for simulating realistic irregular tilings

Fig. 4.5: Examples of stochastic polygons generated using (Pseudocode 1) with different values of $\sigma$ and $\alpha$.

visualization tool is exposed, however for more information see python site.

**Basic MR-Polygon** Creating a basic MR-Polygon using just $\alpha$, $\sigma$ and keeping the others parameters as default.

```python
g>>> import clusterpy
g>>> from clusterpy.core.toolboxes import mrpolygon
g>>> pol = mrpolygon(0.5, 1.2)
```
**Complete MR-Polygon** Creating a MR-Polygon defining $\alpha, \sigma, \mu, X_0, \Delta_t, nPoints$.

```python
>>> import clusterpy
>>> from clusterpy.core.toolboxes import mrpolygon
>>> pol = mrpolygon(0.5, 1.2, mu=10, X_0=10, dt=0.001, nPoints=30)
```

**Getting the coordinates of MR-Polygon** The function `mrpolygon` return 6 elements into a python list `[non-sampled angles, non-sampled ratios, sampled angles, sampled ratios, $t$, $X_t$]`. To get the sampled polygon it is necessary to get the second and third elements of the result and zip them into a single list. At the end of the example the polar polygon is transformed to Cartesian coordinates.

```python
>>> import clusterpy
>>> from clusterpy.core.toolboxes.rimaps import mrpolygon, polarPolygon2cartesian
>>> pol = mrpolygon(0.5, 1.2)
>>> sampledAngles = pol[2]
>>> sampledRatios = pol[3]
>>> polarPolygon = zip(sampledAngles, sampledRatios)
>>> polarPolygon
[(0.0, 10), (0.23, 9.47), (0.44, 9.51), \ldots, (6.28, 10)]
>>> cartePolygon = polarPolygon2cartesian(polarPolygon)
>>> cartePolygon
[[10.0, 9.22, 2.18], (8.58, 4.10), \ldots, (10.0, 0)]
```

### 4.4 Recursive Irregular maps (RI-Maps)

#### 4.4.1 Algorithm

On section 3 it was showed that most commonly used tilings are not capable to emulate the topological characteristics of real maps highlighting the necessity of new tools to create realistic tilings. Looking to satisfy that necessity and based on the fractal characteristics of geographic objects, such as cities, exposed in Batty and Longley (1994) we conceptualize a recursive algorithm on which an irregular frontier is recursively divided using *MR-Polygons* which at the end of the recursion are returned to divide the original frontier into the desired number of areas. Our
conceptualization of the algorithm was made under three principles. i) **Scalability:** Preserving the complexity of the algorithm when the number of areas increases to make possible the construction of tilings with a large number of areas. ii) **Fractality:** Constructing maps from a hierarchically perspective on which each polygon is divided recursively, looking for the fractality observed in real maps which are divided on different scales i.e: United States is divided in States which in turn are divided in counties which also are divided in townships. iii) **Correlativity:** Encouraging the presence of spatial agglomerations of areas with similar sizes.

In Pseudocode 2 we present the *RI-Maps* algorithmic approach to create a polymorphism irregular aperiodic asymmetric with similar topological characteristics of real maps dividing an irregular polygon $pol$ into $n$ irregular polygons. As notation $p_a$ is referring to the area of the polygon $p$. Because the recursive division of $pol$ is made using *MR-polygons* then it is necessary to define the values of $\alpha$ and $\sigma$ to be used, in this case and based on the experiments made in Section 4.3 they are taken from a uniform distribution between 0.1 and 0.5 for $\alpha$, and 1.2 and 1.5 for $\sigma$. About $\mu X_0$ and $\Delta t$ they are defined with the same values established in Section 4.3. Regarding the sampling process to preserve the computational treatability of geocomputational operations *MR-Polygons* are sampled to 30 points.

$$\alpha \in [0.1, 0.5]$$

$$\sigma \in [1.2, 1.5]$$

As can be observed in the *RI-Maps* procedure there are three parameters that have not been explained yet ($p_1, p_2, p_3$). Regarding $p_1$ it is related with the scaling factor to be used over the *MR-Polygons* which divides the irregular frontiers, the higher value of $p_1$ the bigger irregular polygons to be used. About $p_2$ and $p_3$ they allow the presence of the spatial agglomerations of areas with similar sizes. $p_2$ define how probable it is to increase the number of polygons to create into a divisor polygon and $p_3$ is the percentage of the missing areas on which the divisor polygon is increased. Note that if $p_2$ or $p_3$ take values of zero then the map will be homogeneously divided into areas with similar size and no disorder will be
present.

For a better illustration of the parameter selection effect on the topological characteristics of RI-Maps some examples are presented in Table 4.1. These examples are presented in small pictures for easy comparison of the general topological differences, however in the following sections more detailed examples are presented. In the first row of the table \( p_2 \) and \( p_3 \) are defined as 0 and as can be observed generated tilings are very organized and do not show spatial correlation. The second and third combinations show spatial agglomerations of areas with similar sizes and are less ordered than those of the first row, however in the second row the groups of areas with similar sizes are less frequent and marked. As will be shown in section 4.4.2 the tilings generated with the third combination are the more realistic in terms of their topological characteristics.
An algorithmic approach for simulating realistic irregular tilings

Table 4.1: Examples of RI-Maps of 400, 1,600 and 3,600 areas using different combinations of parameters

i) $p_1 = 0.1, p_2 = 0.0, p_3 = 0.0$

ii) $p_1 = 0.1, p_2 = 0.1, p_3 = 0.5$

iii) $p_1 = 0.02174, p_2 = 0.0676, p_3 = 0.3077$
### Pseudocode 2: RecursiveIRregularMap

\[
\text{if } n == 1 \\
\quad \text{then } \text{areas} \leftarrow [\text{pol}] \\
\text{else if } n == 2 \\
\quad \text{then } \begin{cases} 
\text{pol}_1, \text{pol}_2 \leftarrow \text{Divide pol in 2} \\
\text{areas} \leftarrow [\text{pol}_1, \text{pol}_2]
\end{cases} \\
\text{missingAreas} \leftarrow n \\
\text{uncoveredArea} \leftarrow \text{pol} \\
\text{coveredArea} \leftarrow \text{Empty pol} \\
\text{areas} \leftarrow \text{Empty list} \\
\text{scalingFactor} \leftarrow \sqrt{\frac{\text{pol} \times \text{coveredArea}}{n \times \text{uncoveredArea}}} \geq 0.03 \\
\text{while } \frac{\text{uncoveredArea}}{\text{pol}} \geq 1.5 \\
\quad \text{if } \text{missingAreas} \times \frac{\text{uncovered2select}}{\text{uncoveredArea}} <= 1.5 \\
\quad \quad \text{then } \begin{cases} 
\text{Append } \text{uncovered2select } \text{to } \text{areas} \\
\text{coveredArea} \leftarrow \text{coveredArea} \cup \text{uncovered2select} \\
\text{missingAreas} \leftarrow \text{missingAreas} - n_i \\
\alpha_r \leftarrow \text{Random number between } \alpha_{\text{min}} \text{ and } \alpha_{\text{max}} \\
\sigma_r \leftarrow \text{Random number between } \sigma_{\text{min}} \text{ and } \sigma_{\text{max}} \\
\text{poli} \leftarrow \text{MeanRevertingArea}(\alpha_r, \sigma_r, \mu, X_0, \Delta t) \\
\text{poli} \leftarrow \text{Multiply each ratio of poli by scalingFactor} \\
\text{poli} \leftarrow \text{Center poli, randomly into uncovered2select} \\
\text{poli} \leftarrow (\text{pol} - \text{coveredArea}) \cap \text{pol} \\
\text{poli} \leftarrow \text{Bigger part of poli} \\
\text{poli} \leftarrow \text{MissingAreas} \times \frac{\text{poli}}{\text{uncoveredArea}} \\
\text{if } \text{Uniform}(0, 1) < p_2 \\
\text{then } n_i = n_i + \text{missingAreas} \times p_3 \\
\text{if } n_i \geq 1 \\
\quad \begin{cases} 
\text{areasi} \leftarrow \text{RecursiveIRregularMap}(n_i, \text{poli}) \\
\text{Extend areas with areasi} \\
\text{coveredArea} \leftarrow \text{coveredArea} \cup \text{areasi} \\
\text{missingAreas} \leftarrow \text{missingAreas} - n_i 
\end{cases}
\end{cases}
\end{cases}
\text{uncoveredArea} \leftarrow \text{pol} - \text{coveredArea} \\
\text{Append interior holes of coveredArea to areas} \\
\text{coveredArea} \leftarrow \text{Union of areas} \\
\text{while } \text{Length of areas} < n \\
\quad \begin{cases} 
\text{Append the smaller area to its bigger neighbor} \\
\text{while } \text{Length of areas} > n \\
\quad \begin{cases} 
\text{Divide the bigger area}
\end{cases}
\end{cases}
\text{return } (\text{areas})
\]

#### 4.4.2 Parametrizing RI-Maps

To properly define \(p_1, p_2, p_3\) to get realistic topological characteristics using RI-Maps we use a standard genetic algorithm, where the population \(\gamma\) of the iteration \(i\), denoted as \(\gamma^i\), is formed by the genomes \(\gamma^i_j = [p_{j1}^i, p_{j2}^i, p_{j3}^i]\), where \(p_{j1}^i, p_{j2}^i, p_{j3}^i\)
are real numbers between 0 and 1 representing instances of \( p_1, p_2, p_3 \) denoted as phenomes. In this case \( i \in \mathbb{N} \) between 0 and 20 and \( j \in \mathbb{N} \) between 0 and 200. In order to evaluate the quality of each genome the fitness function is defined as (4.22), where \( \theta \) is a set of number of areas, \( \phi_k \) is the relative importance for a maps of \( k \) areas, and \( f_k(\gamma^i_j) \) is a function given by (4.23) that measures how different are, in average, the topological indicators of RI-Maps formed by \( k \) areas using the phonome \( \gamma^i_j \). Looking for simplicity stating (4.23) we denote the vector of real indicators as \( \Psi_k = [M_n, m_n, \mu_1, \mu_2, S, \lambda_1] \) and \( \Psi_k(\gamma^i_j) \) as the same vector for the mean values RI-Maps with \( k \) areas using \( \gamma^i_j \), to refer the \( l^{th} \) indicator \( \Psi^l_k \) and \( \Psi^l_k(\gamma^i_j) \) is used. Regarding \( n_s \) it is the number of simulations to be generated with each genome.

\[
F(\gamma^i_j) = \frac{\left(\sum_{k \in \theta} \phi_k f_k(\gamma^i_j)\right)}{\sum_{k \in \theta} \phi_k} \quad (4.22)
\]

\[
f_k(\gamma^i_j) = \frac{\sum_{l=1}^{6} (\sum_{s=1}^{n_s} \Psi^l_k(\gamma^i_j)) - n_s \Psi^l_k}{n_s \Psi^l_k} \quad (4.23)
\]

To initialize the algorithm an initial random population of 100 genomes is generated and the four genomes with lower fitness function are obtained. The subsequent populations are formed by two parts, the first 64 genomes are all the possible combinations of the last best 4 genomes, the others 36 genomes are random modifications of those 64 genomes. Due to the computational time required to evaluate (4.22) only maps of 400 and 1,600 were used with an importance of \( \phi_{400} = 1 \) and \( \phi_{1,600} = 2 \) respectively. Fig. 4.6 shows the best fitness functions of each iteration. As conclusion the best phoneme is \( p_1 = 0.02174, p_2 = 0.0676, p_3 = 0.3077 \). As is observed the fitness function shows a quick decrease in the first 5 iterations to stay constant until 13th iteration and achieve the optimal value in the 15th iteration. In section 5, the topological indicators of IR-Maps with the optimal parameters are calculated as a complement of 3.2 and some examples are showed.
4.4.3 RI-Maps with python

Using clusterpy is possible to create RI-Maps and calculate the topological indicators of tilings such as real maps or simulated instances. Next some examples are presented.\footnote{As in the previous sections it is assumed that python and clusterpy is installed.}

**Basic RI-Map** Creating and exporting to a shape file a basic RI-Map defining just the number of areas and keeping the others parameters as default.

```python
>>> import clusterpy
>>> rimap = clusterpy.rimap(1600)
>>> rimap.exportArcData("<shapeFile>"")
```

**Complete RI-Map** Creating and exporting to a shape file a RI-Map defining the following parameters, \( N_p, [\alpha_{min}, \alpha_{max}], [\sigma_{min}, \sigma_{max}], \Delta_t, p1, p2, p3, boundary \), where \( N_p \) is the number of points to sample of each polygon, \([\alpha_{min}, \alpha_{max}] \) and \([\sigma_{min}, \sigma_{max}]\) are the minimum and maximum values of \( \alpha \) and \( \sigma \) to be used for the MR-Polygons, and \( boundary \) is a path pointing to a map stored in a
shapefile with a single polygon to be used as initial irregular polygon. If any of the parameters is omitted then the value proposed in section 4.4 is used, if boundary is omitted then the initial polygon will be a MR-Polygon.

**Example 1:** Using a MR-Polygon as initial boundary

```python
>>> import clusterpy
>>> rimap = clusterpy.rimap(1600,N=30,
   alpha=[0.1,0.5],sigma=[1.2,1.5],dt=0.1,pg=0.02174,pu=0.0675,su=0.307701)
>>> rimap.exportArcData("<shapeFile>")
```

**Example 1:** Using an existing irregular map as initial boundary

```python
>>> import clusterpy
>>> rimap = clusterpy.rimap(1600,N=30,
   alpha=[0.1,0.5],sigma=[1.2,1.5],dt=0.1,pg=0.02174,pu=0.0675,su=0.307701),
   boundary="<boundary path>")
>>> rimap.exportArcData("<shapeFile>")
```

**Topological indicators** In order to obtain the topological indicators of a tiling using clusterpy is necessary to use the function `topoStats` which returns a list with 6 indicators in the following order \([M_n,m_n,\mu_1,\mu_2,S,\lambda_1]\).

**Example 1:** Getting the maximum number of neighbors and the sparseness for a regular lattice

```python
>>> import clusterpy
>>> grid = clusterpy.createGrid(10,10)
>>> stats = grid.topoStats(regular=True)
>>> stats[0]
4
>>> stats[5]
0.03636
```

**Example 2:** Getting the maximum number of neighbors and the sparseness for a map stored in shapefile

```python
>>> import clusterpy
>>> layer = clusterpy.importArcData("<shapefile>")
>>> stats = layer.topoStats()
>>> stats[0]
>>> stats[5]
```
Example 3: Getting the maximum number of neighbors and the sparseness for a RI-Map

```python
>>> import clusterpy
>>> layer = clusterpy.rimap(1600)
>>> stats = layer.topoStats()
>>> stats[0]
23
>>> stats[5]
0.0035
```
5. Results

As was exposed in section 3.2 using Table 3.2 the topological characteristics of most common used tilings in regional sciences can not be considered realistic environments. In this section we present the topological characteristics of RI-Maps in the same tabular structure and using plots of the average values of irregular tilings, Voronoi diagrams, real maps and RI-Maps. At the end of the section a set of examples of RI-Maps are presented and based on the computational results of (Duque et al., 2010) a computational application is exposed.

5.1 Topological characteristics of RI-Maps

In Table 5.1 the average and standard deviation of the 6 indicators of RI-Maps using the optimal parameters ($p_1 = 0.02174$, $p_2 = 0.0675$, $p_3 = 0.3077$) are presented. Additionally in Fig. 5.1 the average values are plotted for easy comparison. To construct the indicators of RI-Maps a total of 100 instances were used for each size of map. In Fig. 5.1 it is observed that the maximum number of neighbors of RI-Maps is close to the value of real maps. Regarding the average number of neighbors, RI-Maps value is closer to Voronoi diagrams than real maps, however for practical purposes having values of $\mu_1$ close to 6 or 5.2 is clearly not a big difference. Additionally it is important to note from $\mu_2$ that RI-Maps are disordered than Voronoi diagrams making them more appropriate to emulate real maps. From Fig. 5.1 it is important to highlight the value of $\lambda_1$ close to those of real maps, specially for maps with more than 400 areas. Finally from the values of $S$ clearly all the compared tilings show similar values. Finally from Table 5.1 it is found that standard deviations are small, which means that RI-Maps topological characteristics are less disperse than those of real maps.

5.2 RI-Maps examples

In order to complement the topological characteristics showed before three groups of RI-Maps examples are presented. In the first group Fig. 5.2 to 5.5 maps
### Table 5.1: Average connectedness characteristics for simulated maps

<table>
<thead>
<tr>
<th>Number of areas</th>
<th>81</th>
<th>100</th>
<th>400</th>
<th>900</th>
<th>1,600</th>
<th>2,500</th>
<th>3,600</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_n$</td>
<td>17.24</td>
<td>16.82</td>
<td>25.26</td>
<td>30.50</td>
<td>37.71</td>
<td>46.72</td>
<td>51.60</td>
</tr>
<tr>
<td>$m_n$</td>
<td>± 9.40</td>
<td>± 6.40</td>
<td>± 8.20</td>
<td>± 6.87</td>
<td>± 6.12</td>
<td>± 6.24</td>
<td>± 6.86</td>
</tr>
<tr>
<td>$\mu_1$</td>
<td>1.98</td>
<td>1.83</td>
<td>1.43</td>
<td>1.06</td>
<td>1.02</td>
<td>1.01</td>
<td>1.00</td>
</tr>
<tr>
<td>$\mu_2$</td>
<td>± 1.03</td>
<td>± 0.99</td>
<td>± 0.79</td>
<td>± 0.28</td>
<td>± 0.14</td>
<td>± 0.10</td>
<td>± 0.00</td>
</tr>
<tr>
<td>$\mu_2$</td>
<td>5.52</td>
<td>5.59</td>
<td>5.82</td>
<td>5.87</td>
<td>5.90</td>
<td>5.91</td>
<td>5.92</td>
</tr>
<tr>
<td>$\mu_2$</td>
<td>± 0.25</td>
<td>± 0.22</td>
<td>± 0.07</td>
<td>± 0.05</td>
<td>± 0.04</td>
<td>± 0.04</td>
<td>± 0.04</td>
</tr>
<tr>
<td>$S$</td>
<td>± 8.08</td>
<td>± 4.15</td>
<td>± 2.26</td>
<td>± 1.19</td>
<td>± 1.35</td>
<td>± 1.81</td>
<td>± 1.38</td>
</tr>
<tr>
<td>$S$</td>
<td>0.064</td>
<td>0.052</td>
<td>0.014</td>
<td>0.006</td>
<td>0.004</td>
<td>0.0022</td>
<td>0.0016</td>
</tr>
<tr>
<td>$\lambda_1$</td>
<td>6.64</td>
<td>6.70</td>
<td>7.50</td>
<td>7.87</td>
<td>8.39</td>
<td>9.00</td>
<td>9.28</td>
</tr>
<tr>
<td>$\lambda_1$</td>
<td>± 0.69</td>
<td>± 0.47</td>
<td>± 0.55</td>
<td>± 0.44</td>
<td>± 0.37</td>
<td>± 0.38</td>
<td>± 0.44</td>
</tr>
</tbody>
</table>

Fig. 5.1: Comparison of topological characteristics of reals maps, **RI-Maps**, Voronoi diagrams and regular tilings.

with 400, 1,600, 3600, 7,200 are presented with an Spatial Autoregressive Process with $\rho = 0.7$.

In the second group of examples, Fig. 5.6 to 5.7, a set of maps, created taking advantage of the fact that **RI-Maps** can use any irregular polygon as the external boundary. In Fig 5.6(a) United States by counties is presented and Fig 5.6(b) shows the result of use main single part of the external boundary of United
An algorithmic approach for simulating realistic irregular tilings

Fig. 5.2: SAR(0.7) process over a RI-Map of 400 areas

Fig. 5.3: SAR(0.7) process over a RI-Map of 1.600 areas
Fig. 5.4: SAR(0.7) process over a RI-Map of 3,600 areas

Fig. 5.5: SAR(0.7) process over a RI-Map of 7,200 areas
States as input to construct an *RI-Maps* with the same number of areas as counties into the single part (3304 polygons). In 5.7 and Fig 5.8 the same exercise is made using Colombia dividing by municipalities (1124 polygons) and Italy in third administrative level (7308 polygons). Finally to present the capacity of our algorithm of generating maps with a large number of areas a *RI-Map* of 20,000 areas is presented in 5.9.

![RI-Map using United States external boundary as initial polygon (3.304 polygons)](image)

(a) United States

(b) *RI-maps* in United States boundary

Fig. 5.6: *RI-Map* using United States external boundary as initial polygon (3.304 polygons)
By now it is clear the regional scientists necessity of a new tool to generate tilings with similar topological characteristics of real maps, however in this section an example of use is exposed, proposing an improvement to the computational efficiency analysis made by Duque et al. (2010) of the AMOEBA (A Multidirectional Optimal Ecotope-Based Algorithm). In summary AMOEBA is an algorithm originally proposed by Aldstadt and Getis (2006) to detect spatial clusters and posteriorly computationally improved by Duque et al. (2010), where using a constructive algorithm managed to make it feasible in maps with a high number of areas. To present the results Duque et al. (2010) proposed 3 computational experiments one of them analyzing the running time of AMOEBA while the number of areas of regular tilings increase. Understanding AMOEBA is easy to figure that the computational efficiency is highly affected by the maximum number of neighbors handled during the iterative process, which means that regular lattices are not hardly enough computational tests for AMOEBA. To illustrate this, on one hand a
tiling of 900 areas is constructed and over it 30 Spots processes of 4 clusters are generated, on the other hand, a set of 30 RI-Maps with 900 areas are constructed, each of them with a Spots process of 4 clusters. Latter AMOEBA is running over each Spot process and the running time recorded, Fig. 5.10 shows the differences in the execution times. This figure reinforce confirms the fact that using regular tilings to infer the computational efficiency of an algorithm could over-evaluate the real behaviour. It is important to note that our experiment is to show that robust conclusions can be made using RI-Maps and not invalidate the results obtained in (Duque et al., 2010).
Fig. 5.8: *RI-Map* using Italy external boundary as initial polygon (7.308)
Fig. 5.9: RI-Map of 20,000 areas
Fig. 5.10: Execution times of AMOEBA over regular tilings and *RI-Maps* of 900 areas
6. Conclusions

From this project different conclusions can be established, some of them are based on the theoretical background of tilings in regional sciences and others from a computational point of view.

- Based on the topological characteristics of tilings it is evident that commonly used tilings are highly restricted to emulate realistic space partitions. Taking into account these restrictions different consequences of an inappropriate selection of the tilings to test the methodologies emerge, for example mislead the analysis of an indicator or over-evaluate the computational efficiency of an algorithm.

- Despite of the importance of tilings to made a good evaluation of methodologies and algorithms there are not enough tools to allow researchers to properly infer the capabilities of their methods in real applications.

- A new tool to generate realistic tilings based on stochastic processes, fractal theories and geometric computation is proposed in order to allow researches to properly evaluate their algorithms and methodologies.

- From the implementing process in python it was found that currently geocomputational tools are not powerful enough to deal with quite irregular geometries, making difficult to develop these types of algorithms and forcing developers to seek for alternative solutions for simple theoretical geometric operations complicating the codes and making them lose some computational efficiency.
7. Future research

During the development process of RI-Maps different challenges and ideas for future researches emerged. Some of them are briefly described bellow.

- In section 4.3.2 the surface of fractal dimension of a mean reverting process is constructed in an empirical way, Fig. 4.4, however an ideal way to get that surface would be through an analytical procedure. The problem of the analytical procedure is that the euclidean distance of an stochastic polygon can not be obtained due to the square root is not a lineal function and is not possible to pass from (7.2) to (7.3), being (7.3) the expression obtained using the stochastic calculus for a mean reverting process. A future research is to use a different distance definition to avoid the square root and state a analytical formulation for the fractal dimension of a mean reverting process (4.13).

\[
d(X_t) = d(X_0, X_1) + d(X_1, X_2) + d(X_2, X_3) + \cdots + d(X_{T-1}, X_T) \quad (7.1)
\]
\[
= \sqrt{d^2(X_0, X_1) + d^2(X_1, X_2) + \cdots + d^2(X_{T-1}, X_T)} \quad (7.2)
\]
\[
\neq \sqrt{d^2(X_0, X_1) + d^2(X_1, X_2) + \cdots + d^2(X_{T-1}, X_T)} \quad (7.3)
\]

- As is exposed in section 3.2 the topological characteristics of maps from different parts of the world may differ, see Figure 3.1. Based on these differences a future research is to deeply study how the 7 indicators proposed in section 2.2 change in space.

- Due to the computational problem of geometric packages it is clear the necessity of new improvements to geocomputational methods.
8. Python library

In order to make the algorithms proposed in this thesis available for researchers they are implemented in python taking advantage of the basic functions of clusterpy, library of spatial clustering algorithms developed by RiSE group (Duque et al., 2011b). In this section is explained how to install clusterpy and additional codes required to execute RI-Maps, latter a briefly introduction to the basic operations with clusterpy is made and posteriorly some examples are presented.

8.1 Installing clusterpy

ClusterPy is a library of algorithms for spatial clustering, however it has some functions that make easy the use of spatial algorithms, for example, importing and exporting maps. clusterpy (Version 1.1.0) incorporates MR-Polygons and RI-Maps algorithms. In order to install clusterpy (Version 1.1.0) follow the instructions of http://rise-group.org/section/Software/clusterPy/.\(^\text{12}\):

8.2 Examples of clusterpy

Using clusterpy different operations can be made, for example import and export shape files, simulate spatial processes and generate regular tilings, next some examples of use are presented, however for more advanced examples please refer to “http://www.rise-group.org/risem/clusterpy/”. Additionaly for more examples of how to create MR-Polygons and MR-Maps see the section 4.3.3 and 4.4.3 respectively.

**Importing shape files** Creating a new Layer from a shapefile

```python
>>> import clusterpy
>>> layer = clusterpy.importArcData("<SHAPE FILE>")
```

**Regular tilings** Creating a new Layer with a regular lattice of 10 × 10

```python
>>> import clusterpy
>>> grid = clusterpy.createGrid(10, 10, lowerLeft=(0, 0), upperRight=(100, 100))
```

\(^\text{12}\)It is assumed that python 2.7 is already installed in your computer and can be executed from the terminal, for more information about python see “http://www.python.org/”
Simulated Spots process  Creating a grid of $10 \times 10$ with an spots process with 4 clusters, a level of compactness of 0.7, and significance of 0.9

```python
>>> import clusterpy
>>> grid = clusterpy.createGrid(10, 10, lowerLeft=(0, 0), upperRight=(100, 100))
>>> grid.generateData("Spots","rook", 1, 4, 0.7, 0.9)
```

Exporting data to Shapefile  Creating a grid of $10 \times 10$ with an spots process with 4 clusters, a level of compactness 0.7, and significance of 0.9 and exporting it to a shape file

```python
>>> import clusterpy
>>> grid = clusterpy.createGrid(10, 10, lowerLeft=(0, 0), upperRight=(100, 100))
>>> grid.generateData("Spots","rook", 1, 4, 0.7, 0.9)
>>> grid.exportArcData("gridFile")
```

Exporting RI-Map to Shapefile  Creating a RI-Map of 1.600 and saving it to a shape file

```python
>>> import clusterpy
>>> grid = clusterpy.createGrid(10, 10, lowerLeft=(0, 0), upperRight=(100, 100))
>>> grid.generateData("Spots","rook", 1, 4, 0.7, 0.9)
>>> grid.exportArcData("gridFile")
```
BIBLIOGRAPHY


An algorithmic approach for simulating realistic irregular tilings


Miller, V. (1953). A quantitative geomorphic study of drainage basin characteristics in the clinch mountain area Virginia and Tennesse. *ONR geography branch*, PROJECT NR.


