sid.inpe.br/mtc-m21c/2018/05.07.12.09-TDI

GRADIENT PATTERN ANALYSIS: NEW METHODOLOGICAL AND COMPUTATIONAL FEATURES AND APPLICATION

Rubens Andreas Sautter

Master's Dissertation of the Graduate Course in Applied Computing, guided by Dr. Reinaldo Roberto Rosa, approved in May 09, 2018.

URL of the original document:
<http://urlib.net/8JMKD3MGP3W34R/3R3SGF2>

INPE

São José dos Campos

2018

GRADIENT PATTERN ANALYSIS: NEW METHODOLOGICAL AND COMPUTATIONAL FEATURES AND APPLICATION

Rubens Andreas Sautter

Master's Dissertation of the Graduate Course in Applied Computing, guided by Dr. Reinaldo Roberto Rosa, approved in May 09, 2018.

URL of the original document:
<http://urlib.net/8JMKD3MGP3W34R/3R3SGF2>

INPE

São José dos Campos

2018

Aluno (a): *Rubens Andreas Sautter*
Título: "GRADIENTE PATTERN ANALYSIS: NEW METHODOLOGICAL AND COMPUTATIONAL FEATURES AND APPLICATION".

Aprovado (a) pela Banca Examinadora em cumprimento ao requisito exigido para obtenção do Título de *Mestre* em

Computação Aplicada

Dra. **Lúbia Vinhas**
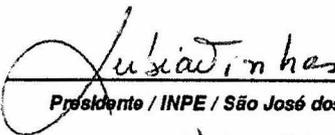
Presidente / INPE / São José dos Campos - SP

( ) Participação por Video - Conferência

Dr. **Reinaldo Roberto Rosa**
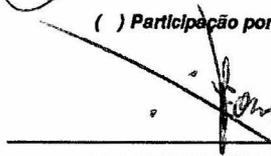
Orientador(a) / INPE / SJCampos - SP

( ) Participação por Video - Conferência

Dr. **Elbert Einstein Nehrer Macau**

Membro da Banca / INPE / São José dos Campos - SP

( ) Participação por Video - Conferência

Dr. **Erico Luiz Rempel**

Convidado(a) / ITA / SJCampos / SP

( ) Participação por Video - Conferência

Dra. **Virginia Klausner de Oliveira**
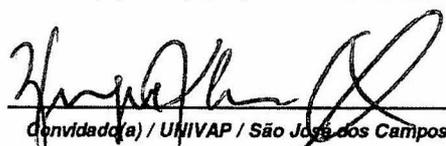
Convidado(a) / UNIVAP / São José dos Campos - SP

( ) Participação por Video - Conferência

Este trabalho foi aprovado por:

( ) maioria simples

(x) unanimidade

São José dos Campos, 09 de maio de 2018

"The greatest value of a picture is when it forces us to notice what we never expected to see."

<div align="center">

JOHN TUKEY

</div>

A meus pais Frank e Cláudia, e a minha irmã Cecilia. Pelo apoio, para que eu chegasse até aqui.

ACKNOWLEDGEMENTS

ABSTRACT

In this work it is presented the Gradient Pattern Analysis (GPA), a formalism that describes operators for analysis of spatially extended system, concerning its asymmetry. Aiming to work with large datasets, it is proposed improvements to the most popular version of GPA, with respect to the metric measurement and computational efficiency. We also review and explore the gradient moments, and propose two new operators. In order to validate the implementation of the operators G_1 and G_2, the following study cases are presented: (i) a dynamical study case in Coupled Map Lattices (CML), and (ii) a static case study in Galaxy Morphology. With respect to application (i), we analyze two system transitions: symmetry breaking and synchronization. Concerning the application (ii), it is presented a system of galaxy morphometrics named CyMorph, which has an important role on a project for studying the galaxies formation and evolution. The aim of CyMorph is to classify galaxies, between early-type and late-type using non-parametric morphometrics. G_1 and G_2 were integrated to CyMorph. We observe that G_2 is the second-best morphometric in a system with 10 metrics.

Keywords: Gradient Pattern Analysis. Dynamical Systems. Galaxy Morphology. Image Processing.

ANÁLISE DE PADRÕES GRADIENTES: NOVOS ASPECTOS COMPUTACIONAIS METODOLÓGICOS E APLICAÇÕES

RESUMO

Neste trabalho é apresentado a Análise de Padrões Gradientes (no inglês GPA), um formalismo que descreve operadores para a análise de matrizes, por meio da simetria. Com o objetivo de analisar bases de dados extensas, neste trabalho é proposto o refinamento da versão mais popular do GPA, a respeito da medida e da complexidade computacional. Neste estudo é apresentado todos os momentos gradiente, e testado o primeiro e segundo momento gradiente (respectivamente G_1 e G_2). A fim de testar o refinamento das técnicas G_1 e G_2 é apresentado os casos de estudos: (i) um estudo de caso dinâmicos em Grade de Mapas Acoplados (no inglês CML) e (ii) um estudo de caso estático em Morfologia de galáxias. Em relação aplicação (i), duas transições de estado do sistema são apresentados: quebra de simetria e sincronização. Em relação a aplicação (ii), foi desenvolvido um pipeline de análise não paramétrica de galáxias conhecido como CyMorph. O pipeline apresentado incorpora uma versão aprimorada das técnicas de análise morfologica, G_1 e G_2. O objetivo principal do CyMorph dentro do escopo do projeto de pesquisa é classificar galáxias entre elipticas (early-type) e espirais (late-type). Analisando o desempenho da técnica de GPA frente as técnicas tradicionais de morfologia, observou-se que G_2 é o segundo melhor parâmetro morfométrico no conjunto apresentado.

Palavras-chave: Análise de Padrões Gradientes. Morfometria. Cosmologia. Sistemas Dinâmicos. Caos.

LIST OF FIGURES

LIST OF TABLES

xvii

LIST OF ABBREVIATIONS

CART	–	Classification And Regression Tree
CFD	–	Central Finite Difference
CML	–	Coupled Map Lattice
FWHM	–	Full Width at Half Maximum
GPA	–	Gradient Pattern Analysis
GPGPU	–	General Purpose Graphics Processing Unit
LLE	–	Largest Lyapunov Exponent
PSF	–	Point Spread Function
RBF	–	Radial Basis Function
SDSS	–	Sloan Digital Sky Survey
SExtractor	–	Source Extractor
SOM	–	Shobu-Ose-Mori
SVM	–	Support Vector Machines

CONTENTS

1 INTRODUCTION

From large scales, such as galaxies, to small scales, for instance molecular structures, a diversity of bidimensional projected patterns are observed. These patterns are intrinsically related to the processes that drive the system. Thus, the extraction of features of these systems shall reflect its underlying pattern formation processes. However, the analysis of these patterns, which are observed in digital images, is a challenge since there is a constant increase of data, in terms of amount and resolution.

The analysis of patterns is a digital image processing (DIP) task. As a DIP task, it can be detached in two branches: the images processing, and the image analysis (PEDRINI; SCHWARTZ, 2008). The image processing includes algorithms to highlight and simplify the information in a given image, such as image segmentation and image enhancement. Whereas, the image analysis concerns the extraction of the information in a given image or set of images. Notice that, the last is a multidisciplinary task since it requires knowledge from computer science, and the data domain. In the context of this work, we explore both branches of DIP, but we highlight the image analysis as we intend to characterize patterns.

Several techniques attempts to characterize spatial patterns (DIGGLE, 2013). A commonly approach, to deal with the diversity of patterns, is to extract a set of parameters. For instance, the CAS systems have been employed to characterize galaxy morphology (DRESSLER, 1980; CONSELICE, 2003). This approach gained attention with the advent of datamining area. It has positive and negative aspects. The more parameters, the greater set of features are detected. However, the extension of a set of parameters increases the computation time. Also, a meaningless metric can perturb the analysis, specially, when it is applied an automatized analysis. Thus, we shall explore fast meaningful metrics.

In this work we explore the Gradient Pattern Analysis (GPA) an innovative technique, which have been employed in a diversity of applications. It exploits the gradient symmetry breaking in order to characterize a given pattern, which no other technique measures. GPA was proposed by Rosa et al. (1999), where it has been shown some properties and the conjectures. Further works extended the formalism by introducing the gradient moment concept (RAMOS et al., 2000; ROSA et al., 2008; ASSIREU et al., 2002), which were fundamental to improve the analysis capability. We intend in this work to review and improve GPA technique. In order to test the improved version, we apply the proposed metric to images from chaotic systems and

astrophysics, measuring its performance.

1.1 Motivation

GPA has been employed in a diversity of applications, such as: detection of solar bursts (ROSA et al., 2008), characterization of diffusive-reaction phenomena(RAMOS et al., 2000), and characterisation of nonlinear transition in Lagrangean fluctuations (ASSIREU et al., 2002). The analysis with GPA achieved the goals. However, the data analyzed in these applications were represented on small matrices, which size vary between 3x3 and 128x128. With the advance of observational capacity, and the increase of complexity of simulated data, a demand for fast and efficient algorithms increased. However, in the state of the art, this metric is overly sensitive to noise level, and the high computational cost hinder the analysis of extended systems.

1.2 Objectives

The main goal of this work is to improve the Gradient Pattern Analysis (GPA) with respect to the metric characteristics and the computational performance. In order to verify the presented technique, we selected a dynamic case study, and a static case study, which are respectively: (i) Coupled Map Lattices and (ii) galaxy morphological classification. GPA had already been employed in these applications (RAMOS et al., 2000; STRIEDER, 2010). However, some details were not explored in the previous works, such as: different maps (i), and larger datasets (ii). As a secondary objective, we intend to explore the details related to those domains. Other objectives intrinsically related to the main and secondary objectives are:

- To establish an overview of the GPA formalism, reviewing the previous definition and expanding some steps that were defined in previous works.
- To parallelize the presented operator.
- To characterize Coupled Map Lattices (CML) state transition with classical methodologies: Phase-Space analysis and Lyapunov Exponent.
- To introduce and adapt the GPA formalism to CML, aiming to detect and characterize the system state transitions.
- To develop an open-source pipeline for galaxy morphometry, based on most advanced galaxy morphometrics. Adjust the proposed operator to the galaxy morphometric analysis and compare with the current state of the art.

1.3 Dissertation overview

This dissertation is structured in following chapters:

2. GRADIENT PATTERN ANALYSIS - In this chapter, we aim to overview GPA state of the art, propose improvements to the technique, and apply the tests well-established on literature.
3. FIRST CASE STUDY - A study of Coupled Map Lattices is presented in this chapter. It is also presented some analysis techniques commonly applied to this domain.
4. SECOND CASE STUDY - In this chapter we review of state of the art on galaxy morphology. We propose improvements to the established techniques, adapt the GPA formalism to this domain and compare the results.
5. CONCLUSION - The overall performance of the presented technique is described and future works are proposed.

Additionally, it is presented in the appendix: a simple tutorial for downloading in Github the source code of all algorithms presented, a method for obtaining galaxy images, an approach to G_2 parallelism, and the technique for measuring the Largest Lyapunov Exponent.

2 GRADIENT PATTERN ANALYSIS

In this section we present Gradient Pattern Analysis (GPA), a computational operator that describes matrices in terms of gradient symmetry and alignment.

In order to understand the presented technique, consider a matrix, which is composed by a set of amplitude $A(x, y)$, being x, y the element spatial coordinates. According to the elements spatial disposition, we may observe different patterns, as shown in Figure 2.1.

In the GPA formalism the characterization of these patterns are given by the analysis of local differences (gradient) and global differences (asymmetry). For example, the patterns A and B from Figure 2.1 are symmetric, thus we may expect low values from GPA metrics. Whereas a random matrix is an example of asymmetric pattern (for example pattern E). Therefore, it is expected a higher magnitude of GPA metrics.

The main reason to analyze the gradient field instead of the amplitude field is the gradient sensitivity. Small perturbations on amplitude can be difficult to detect in the amplitude field. However, in gradient field those perturbations drastically change the pattern.

On the other hand, it is important to distinguish small perturbed patterns from patterns that are completely disordered. Here, we consider a disordered pattern, a matrix with high fragmentation degree, with asymmetrical fragments. This distinction is necessary since it is commonly observed in data acquisition and data handle small perturbations that are not related to the overall pattern. For example, the patterns C and D from Figure 2.1 are respectively the same of pattern A and B, with an addition of noise. Thus, the GPA metric for patterns C and D must be similar to the response of pattern A and B, and the difference between those patterns and pattern E must be greater.

Another important aspect of GPA operators is the distinction between asymmetric regular patterns and irregular asymmetrical patterns. In Example 2.1.E, it is shown a gradient image, which in gradient field correspond to a laminar flow. Although the gradient field of patterns 2.1.E and 2.1.F are asymmetrical, both patterns are clearly distinct. Therefore, the operator response must be different for these patterns.

Figure 2.1 - Symmetric and asymmetric gradient pattern examples
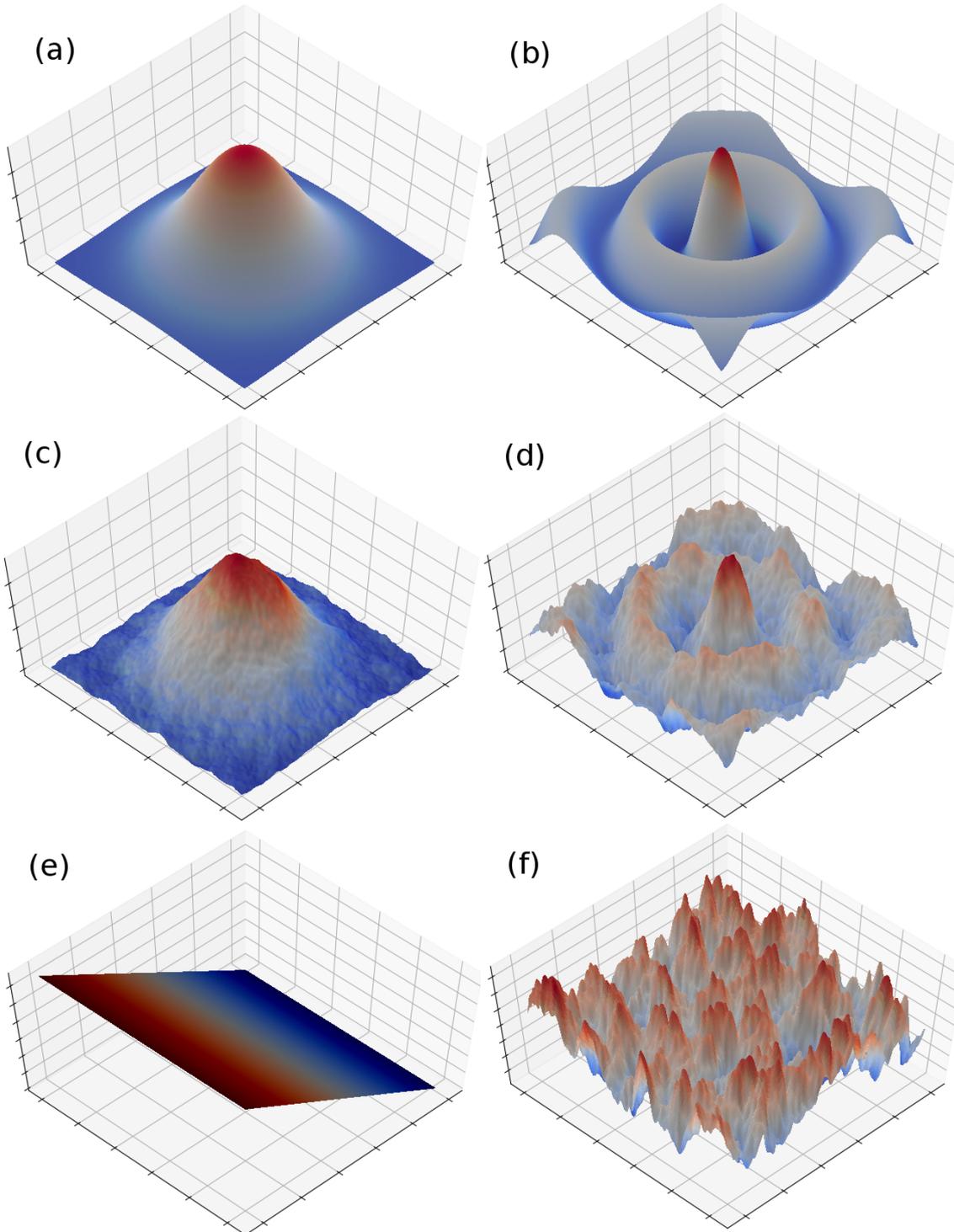
Six sample of patterns. (a) and (b) are examples of symmetric patterns, (c) and (d) show the same pattern of (a) and (b) contaminated by a Perlin noise, (e) show an asymmetrical pattern known as laminar pattern, and (f) show an asymmetrical pattern composed by Perlin noise.

SOURCE: Own author.

Besides the metric meaning, a demand for techniques for larger bidimensional systems/images is constantly increasing with the advance of the observational capacity. In this work we also intend to adapt GPA for larger systems, we aim to minimize the computational cost. The bottleneck of GPA previous version is the symmetry removal (the second step in Figure 2.6). All vectors in this step were compared with each other, the complexity for this approach is $O(W^2H^2)$, were W and H are the matrix width and height. To increase GPA computational efficiency, we review the symmetry definition in section 2.1 and introduce the concentric symmetry notion, which reduces the total number of operations. In addition, we exploit the last hybrid parallel technologies, it is presented a parallel version of this operator using OpenCL in Appendix C. In our implementation the presented symmetry removal has a computational complexity[1] of $O(W^2H^2/P)$, where P is the number of parallel process.

This chapter is organized in the following sections: a review of symmetry definition and the concentric symmetry proposal (section 2.1), the review of gradient moments (section 2.2), an overview of the proposed framework (section 2.3), the tests with noise matrices (section 2.4), and lastly some GPA conjectures (section 2.5).

2.1 Symmetry

The lack of symmetry – the asymmetry– is useful for understanding and characterizing matrices. In this section, is discussed the general symmetry definition, and the extension for the case of gradient field.

According to Barker e Howe (2007), an operator composed by a set p and an operation $R()$ is said symmetrical if it is invariant to the operation ($R(p_x) = p_x, \forall p_x \in p$). Furthermore, the number of element with the same symmetry element (p_x) defines the group order. A diversity of symmetric groups were defined in literature following this definition. Some examples of symmetry groups are shown in Figure 2.2.

---

[1]This analysis is shown in Appendix C

Figure 2.2 - Example of symmetry groups
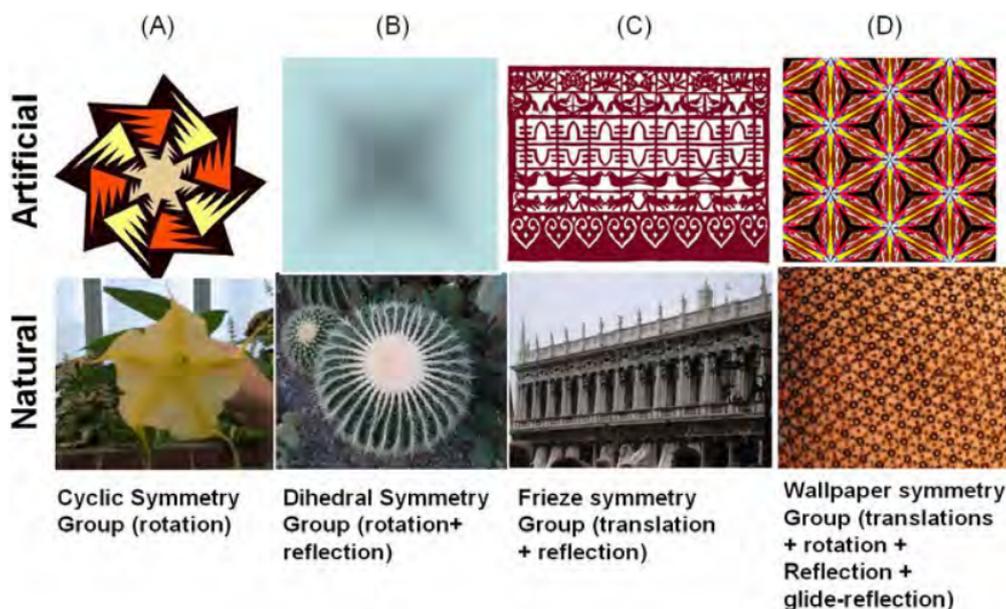
In this work we do not intend to study the differences between symmetry groups. Instead, we aim to detect all asymmetrical elements, the set of elements that are not in any group of symmetry. Since we are analyzing gradient fields, we shall consider two main properties of each element: the spatial coordinates and each element local property. The spatial position determine which set of elements are candidates for symmetry, whereas the local property determine the equality operation.

In the gradient field, each element has a spatial coordinate, modulus, and phase. The spatial position determines which set of elements are candidates for symmetry, whilst the modulus and phase are local properties that define the equality operation.

Several symmetry definitions attempt to detect asymmetrical patterns. One of the simplest spatial criterion is the bilateral symmetry. Given two elements equally distant to a separation axis, where the difference between both elements are perpendicular to the axis of separation. If both elements are equal, then they are a symmetrical pair.

It is important to note that in some systems, we may observe many axes of symmetry. Consequently, the bilateral concept leads some issues:

I-1 How to define the orientation from a generic image? (A galaxy image for instance)

I-2 How many axes of symmetry should be applied?

I-3 Given N axes of symmetry, which evaluation sequence should be applied?

To solve the issues, we introduce the concentric symmetry notion. This concept is an extension of bilateral symmetry that considers all possible axes of symmetry. The candidates for symmetry, in this criterion, are given by every element at the same euclidean distance. In this analysis we do not concern on the symmetry group order. If a candidate of symmetry has at least one equal element at same distance, then it is set as symmetrical element.

Notice that the described definition solves all issues since the operation is invariant to rotation (I-1), considers always the maximum number of axes (I-2), and is independent to the evaluation sequence, as it does not concern on the group order (I-3).

The disadvantage of the concentric symmetry is the implementation computational complexity of $O(W^2 H^2)$, where W and H are the input matrix sizes. In this work we couldn't reduce the computational complexity. However, we describe an algorithm that reduce the number of computational operations by restricting the search area of each element (see Appendix C).

Once defined the spatial disposition criteria, we need to define the equality operation. For the equality operation, we follow the criteria defined by Strieder (2010). Two vectors (for instance v_1 and v_2) are symmetric if both vectors have the same modulus ($|v_1| = |v_2| \pm \Delta v$) and opposite phase ($|\theta_1 - \theta_2| = \pi \pm \Delta\theta$). Notice that in practice, this equality test considers a tolerance in modulus (Δv) and phase ($\Delta\theta$).

In Figure 2.3, it is presented the symmetry removal on a 2D Gaussian sample. The circled vectors are an example of symmetric group since both vectors are at same distance to the center, have opposite phase, and same modulus. Moreover, all vectors in this example are symmetrical with exception of the red vectors. The red vectors describe an artificial contamination, which causes the symmetry breaking in eight vectors. Four vectors are asymmetrical as they are in the contamination neighborhood. Other four vectors in opposite side to the contamination are asymmetrical, due to a mirroring effect.

Figure 2.3 - Gradient Symmetry example
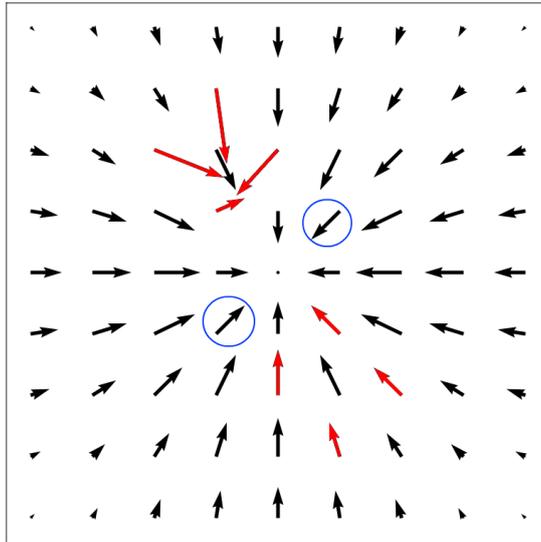
Gradient of a gaussian matrix contaminated. Red vectors are asymmetrical, blue circle is an example of symmetry pair.

SOURCE: Own author.

Here we end the review of symmetry applied to gradient field. In following section it is shown another important aspect of GPA formalism: the gradient moments.

2.2 Gradient Moments

In this section we present a set of metrics that characterize a gradient pattern. These metrics concerns on gradient alignment, gradient disposition, and variety of modulus, which are standardized according to the gradient moment. Ramos et al. (2000) introduced the gradient moment concept, where the metric is standardized according to the gradient notation, as shown in Figure 2.4. The first gradient moment is measured directly on gradient field. Whereas, the second and third gradient moment are measured respectively on the set of modulus and phases. Lastly, the fourth gradient moment is a measurement of the gradient complex representation.

Figure 2.4 - Gradient Moments
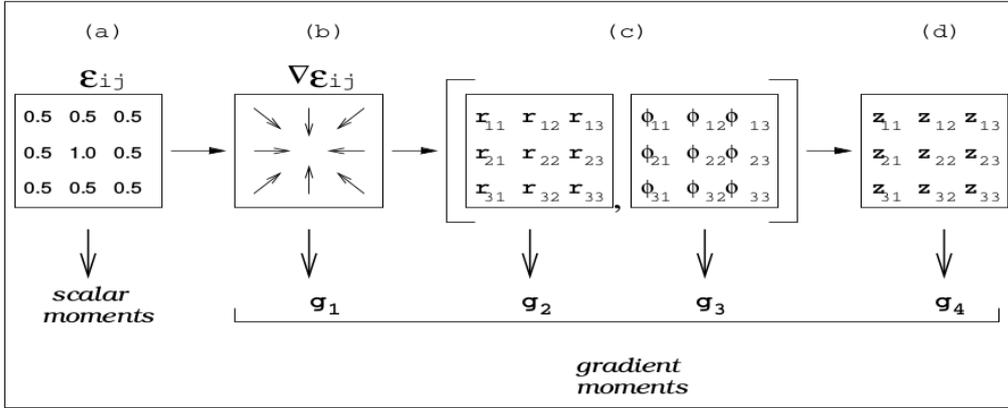
Four gradient moments, each gradient moment is measured from a gradient field nota-
tion.

SOURCE: Ramos et al. (2000).

Here it is presented the operators for all gradient moments. The first and fourth
gradient moment have been defined in literature (ROSA et al., 1999; ROSA et al., 2008;
ASSIREU et al., 2002; RAMOS et al., 2000). Since there was no operator for the second
and third gradient moments described in literature, we introduce the Asymmetrical
Magnitude Coefficient (G_2) and Asymmetrical Phase Coefficient (G_3). In following
subsections we present these operators according to the moment order.

2.2.1 Asymmetrical Gradient Coefficient

The first gradient moment describes patterns in matter of the fragmentation of
the asymmetrical gradient field. Specifically, the Asymmetrical Gradient Coefficient
(G_1) exploits the Delaunay triangulation to distinguish spatially extended patterns.

It is important to note that there is a diversity of triangulation algorithms. The rea-
son to establish the Delaunay triangulation is that this triangulation maximizes the
minimal internal angle. The resultant mesh contain triangle set, which approaches
to an equilateral triangle the most as possible. In this implementation we use the
function "scipy.spatial.Delaunay" from Scipy (JONES et al., 2001).

The Asymmetrical Gradient Coefficient is measured according to the equation 2.1,
where N_C is the total of Delaunay connections, and N_V is the total of asymmetrical

vectors.

$$G_1 = \begin{cases} \frac{N_C - N_V}{N_V} & if\, N_C \geq N_V \\ 0 & otherwise \end{cases} \qquad (2.1)$$

Rosa et al. (1999) show that G_1 is bound to the interval $[0.0, 2.0]$, where 0.0 represents a totally symmetric matrix, and 2.0 represents an asymmetrical matrix with misaligned vectors.

2.2.2 Asymmetrical Magnitude Coefficient

Here is introduced the Asymmetrical Magnitude Coefficient (G_2), a GPA operator of second order. Given the set of asymmetrical vectors (v_a^i), the number of asymmetrical vectors V_A, and the total number of vectors (V), G_2 is measured according to the equation 2.2.

$$G_2 = \frac{V_A}{V} \left(2 - \frac{\left| \sum\limits_{i}^{V_a} v_a^i \right|}{\sum\limits_{i}^{V_a} |v_a^i|} \right) \qquad (2.2)$$

Notice that this operator concerns on the asymmetry ratio (V_A/V), and the confluence ($|\sum v_a^i| / \sum |v_a^i|$). The independent analysis of both measurements is also important in some applications since these measurements concern on different properties.

The asymmetry ratio is a simple proportion of asymmetrical vectors, which is bound to $[0, 1]$. For general problems, the total of vectors (V) is given by WH, where W is the matrix width and H is the matrix height. However, in problems that require a segmentation –for example the morphometry (section 4.9) – this amount is given by the number of segmented area of the aimed object.

The confluence determines if the vectors are aligned and have the same magnitude. If all vectors are the same (v_a), then the sum of all vectors is ($|\sum v_a^i| = \sum |v_a^i| = V_A |v_a|$), thus $G_2 = V_A/V$. The described pattern is a laminar flow, an example of this pattern is shown in Figure 2.1.F. It is important to note that $G_2 = 1.0$ does not necessarily mean that the pattern is a laminar flow since we can obtain the same value for G_2 when $V/V_A = 2 - |\sum v_a^i| / \sum |v_a^i|$, an example of this case is shown on synchronization of Coupled Map Lattice (section 3.5).

Also, at limit the maximal misalignment turns the vector sum to zero ($|\sum v_a^i| = 0$),

thus $G_2 = 2V_A/V$. This case is commonly observed in random patterns, an example is shown in Figure 2.1.F.

2.2.3 Asymmetrical Phase Coefficient

In this subsection, we introduce a new operator for the third gradient moment: the Asymmetrical Phase Coefficient (G_3). Considering the same framework applied to G_2, the proposed operator is expressed in terms of the asymmetry ratio (V_A/V) and the confluence, as shown in equation 2.3.

$$G_3 = \frac{V_A}{V}\left(2 - \frac{1}{V_A(V_A - 1)} \sum_{i=0}^{V_A-1} \sum_{j=i+1}^{V_A} 1 + u_i u_j\right) \tag{2.3}$$

As described in Figure 2.4, the third gradient moment concerns on the asymmetrical gradient phase. Thus, we propose a new technique for measuring the confluence, by measuring the average of angular distance between each pair of phases (θ_i, θ_j).

In order to measure the angular distance, consider the unit vectors u_i and u_j, where $u_i = (cos(\theta_i), sin(\theta_i))$. The internal angle ($\rho$) between u_i and u_j is used to measure the distance. For instance, if both vectors are aligned ($\rho = 0$) the distance is maximum, whereas if both vectors are opposite ($\rho = \pi$) the distance is minimum.

The internal angle can be measured from the dot product between u_i and u_j, since $u_i u_j = |u_i||u_j|cos(\rho) = cos(\rho)$. Here we measured the distance as the normalized cosine function of the internal angle between u_i and u_j. In order to normalize this distance, we resize and translate this function, leading: $(u_i u_j + 1)/2$. The Figure 2.5 shows the distance function for a given internal angle ρ.

Figure 2.5 - Angular distance function
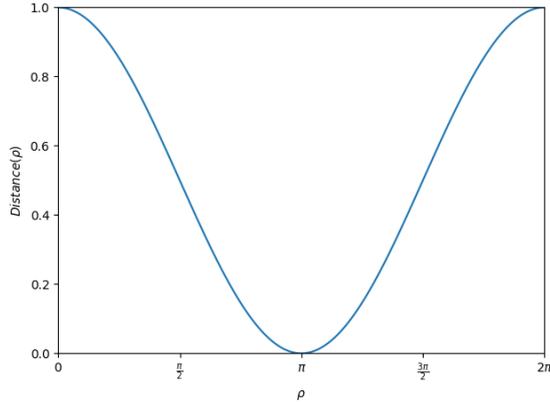
The internal angle (ρ) in x-axis, and the angular distance (($u_i u_j + 1)/2$) in y-axis.
SOURCE: Own author.

Notice that the alignment is symmetrical ($u_i u_j = u_j u_i$), thus we can reduce the number of computational operations. The minimal number of pairs to be compared is given by the combination: $V_A(V_A - 1)/2$, where V_A is the number of asymmetrical vectors. Finally, the confluence is measured as: $\frac{1}{V_A(V_A-1)} \sum \sum 1 + u_i u_j$.

2.2.4 Generalized Complex Entropic Form

The Generalized complex Entropic Form (GEF) is a fourth gradient moment operator, which is deduced from a generalization of the concept of degeneracy. The general form of GEF is given by the equation 2.4, where $|v_{i,j}|$ and $\phi_{i,j}$ are respectively the modulus and the phase of the element on spatial coordinate i, j.

$$G_4 = -\sum_{i,j} |v_{i,j}| ln(|v_{i,j}| e^{-i\phi_{i,j}k}) \tag{2.4}$$

Notice that the image from G_4 is a complex number, as shown in equation 2.5. In order to simplify the analysis and interpretation, the real and imaginary parts of G_4 are usually analyzed separately (see for example Rosa et al. (2000)).

$$G_4 = -\sum_{i,j} |v_{i,j}| ln(|v_{i,j}|) + i \sum_{i,j} |v_{i,j}| \phi_{i,j} k \tag{2.5}$$

We can see that G_4 has a constant value for the laminar case, when it has a constant

modulus. However, if we change the vector modulus in laminar case, we may observe a different a proportional response. Another important aspect to note is that G_4 is not bounded as the previous operators.

2.3   GPA Overview

In previous sections, it is described the aspects that compose the GPA formalism: the symmetry (section 2.1) and gradient moment (section 2.2). Here, we describe the framework of the formalism, which includes those aspects. A general scheme is shown in Figure 2.6, which shows three stages that are applied to the input matrix in order to measure the feature.

Figure 2.6 - GPA scheme
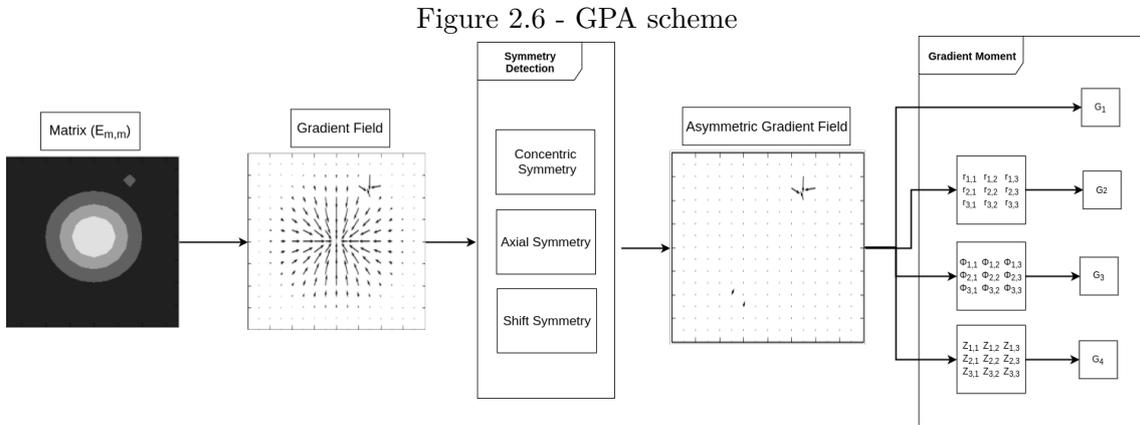
GPA overview, given the matrix E(i,j), three steps are applied: (i) gradient field measurement (ii) symmetric vectors remotion (iii) gradient moment measurement.

SOURCE: Own author.

In the first step the matrix gradient is obtained, the technique to measure the gradient must consider the data type (for example image color-scale and matrix size), and the domain of application (for example in dynamical system analysis, image processing, or general purposes). A central finite difference (CFD) scheme was employed in the presented framework since it is the simplest approach, besides being implemented in previous versions of GPA (ROSA et al., 1999; STRIEDER, 2010). The CFD scheme for the x-axis direction, in a matrix composed by elements $a_{x,y}$, where x and y are the spatial coordinates is given by $f_x(a_{x,y}) = (a_{x+1,y} - a_{x-1,y})/2$. Whereas, in y direction the CFD is $f_y(a_{x,y}) = (a_{x,y+1} - a_{x,y-1})/2$.

In the second step, vectors are removed according to an asymmetry criterion. A brief

essay about symmetry and the extension for gradient field is presented in section 2.1. In this framework the concentric symmetry were applied.

Finally, the last step consists of measuring a scalar value from a gradient field, which is given by the gradient moment. In the presented framework, all gradient moments have been implemented. However, due to time limitations, only operators from first and second gradient moment were tested. Here, we complete the overview of the framework, in following section, we start the first test with noise matrices.

2.4   Noise Matrices

In this section we present a sensitivity test to the GPA technique. As shown by Rosa et al. (1999), there is a correlation between the matrix size and G_1 sensitivity. The reason that should drive the methodology sensitivity is the number of possible patterns that arises with the matrix size. A matrix with size NxM has a finite number possible patterns for G_1, since the maximum number of asymmetrical vectors is NM, and the maximum number of Delaunay connections $2NM$. We intend to see if the proposed operator has the same property observed in previous works (ROSA et al., 1999; STRIEDER, 2010).

Notice, this test can be done with any kind of pattern family. There is a diversity of patterns described literature, and some of these patterns also had been tested with GPA (ROSA et al., 2000; VERONESE, 2011). However, here we intend to analyze the simplest patterns.

In order to remake this test, G_1 and G_2 were applied to two set of noise matrices: Random matrices and Perlin matrices. Both matrices have interesting properties. Random matrices have maximum variability between amplitudes. Therefore, it is expected the maximum G_2 value. Whereas Perlin matrices have spatial continuity that shows the same pattern for a given set of variables, which result in same average value for a given frequency.

2.4.1   Random Matrices

It is presented in this subsection the review of the sensitivity test[2] applied to G_1, with respect to the matrix size. In this test, the gradient moment is measured in groups of square matrices with a fixed length. Comparing the average (μ) and standard deviation (σ) of these groups, considering the matrix length, we may expect

---

[2]Using the same test of Rosa et al. (1999)

according to previous works μ converging the pattern characteristic value and σ converging to 0 with the increase of matrix length.

To generate random matrices, we selected matrices with MxN elements $(0, 1, ..., MN)$, and randomly permuted its position. Then, we obtain a matrix, which has no spatial correlation. Some examples of these matrices are shown in Figure 2.7. The library numpy were employed to apply the permutation, which used the Mersenne Twister random number generator (RNG). There is a diversity of RNG algorithms, which would be interesting to test, however, due to time limitations it is not explored in this work.

Figure 2.7 - Random matrices
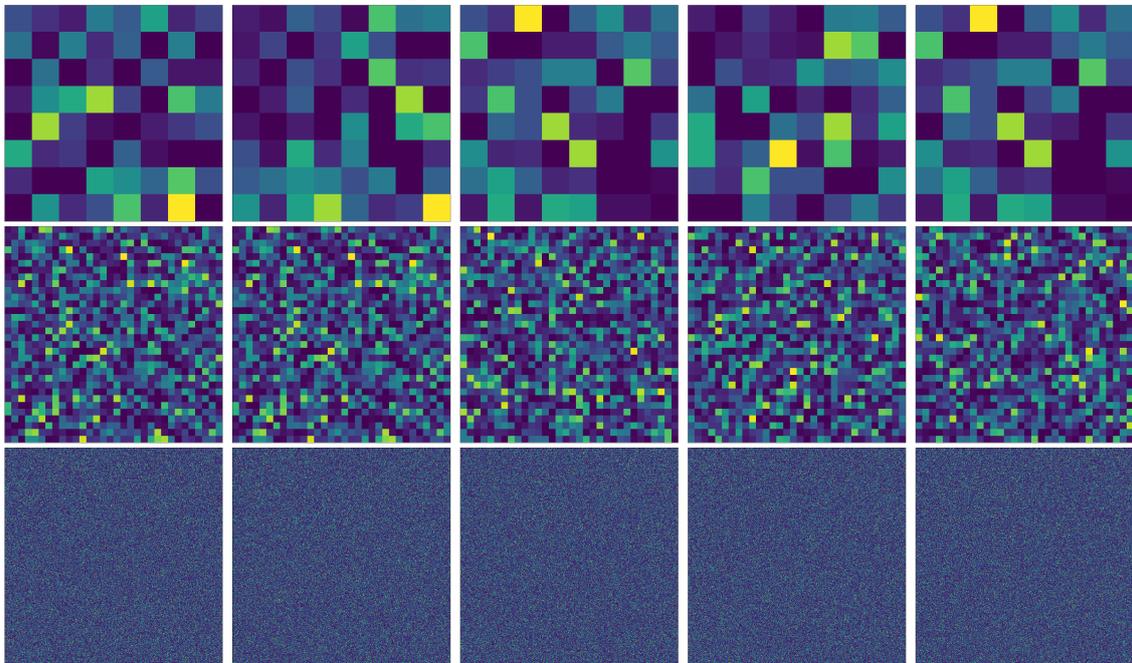
Sample of random matrices, rows ordered according to the matrix size: 8x8, 32x32, 512x512

SOURCE: Own author.

The results of this test are shown in Table 2.1. All elements in these matrices are different. However, as we increase the size of this matrix small changes are not detectable normalized gradient. The higher probability of finding similar gradients in larger matrices explains the decrease on G_2 with the matrix size.

Table 2.1 - G_2 for Random matrices

Matrix size	(i) Without tolerance		(ii) With tolerance	
	Average	Standard Deviation	Average	Standard Deviation
4x4	1.6665	0.1826	1.6034	0.2068
8x8	1.883	0.0578	1.8622	0.0674
16x16	1.9546	0.0239	1.9516	0.0248
32x32	1.9824	0.0099	1.9795	0.0104
64x64	1.994	0.0031	1.9876	0.0042
128x128	1.9978	0.0014	1.9835	0.0024
256x256	1.9992	0.0006	1.9707	0.0023
512x512	1.9999	0.0003	1.9421	0.0023

2.4.2 Perlin Matrices

Interesting solutions and algorithms were developed with the advance of computer graphic subareas. The random generation of textures and terrains is a challenge, as it requires the generation of spatially related random amplitudes. A remarkable algorithm to generate spatially related random amplitudes is the Perlin noise(EBERT et al., 1998). Besides graphical computing application, with some adaptations, Perlin noise is a good description for spiral galaxies dust (see for example the models generated by Groeneboom e Dahle (2014)). The analysis of the pattern generated by Perlin noise is essential for a better understanding of GPA results in galaxy morphology.

Here we name the matrices which are only filled with Perlin noise as Perlin matrices. These matrices were built using a component from library Pyglet (HOLKNER, 2008). The procedure for generating Perlin matrices is composed by three steps: Grid definition, Dot product, and interpolation. In the first step it set a sparse grid of vectors (the red dots in Figure 2.8), and the final grid of elements (blue dots in Figure 2.8). Each vector in sparse matrix is randomly selected from a list of predefined directions, in order to avoid defects(EBERT et al., 1998).

Figure 2.8 - Perlin noise scheme
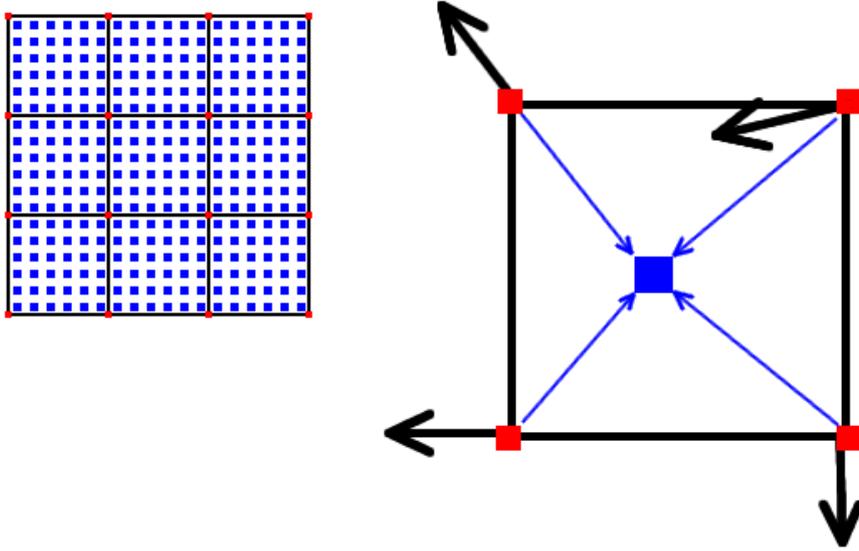
Red dots are the sparse grid elements, with vector in black. Blue dots are the Perlin matrix elements, and the blue arrows are the distance vector.

SOURCE: Own author.

In the follow steps, each element in the final grid associate a value to each nearby sparse grid element. This value is given by the dot product between the distance to the sparse grid element (blue vector in Figure 2.8) and the corresponding vector in the sparse element (black vector in Figure 2.8). The final value is measured as the interpolation between each element and the measured dot product. Some examples of Perlin matrices (generated using Pyglet component) are shown in Figure 2.9.

Figure 2.9 - Perlin matrices
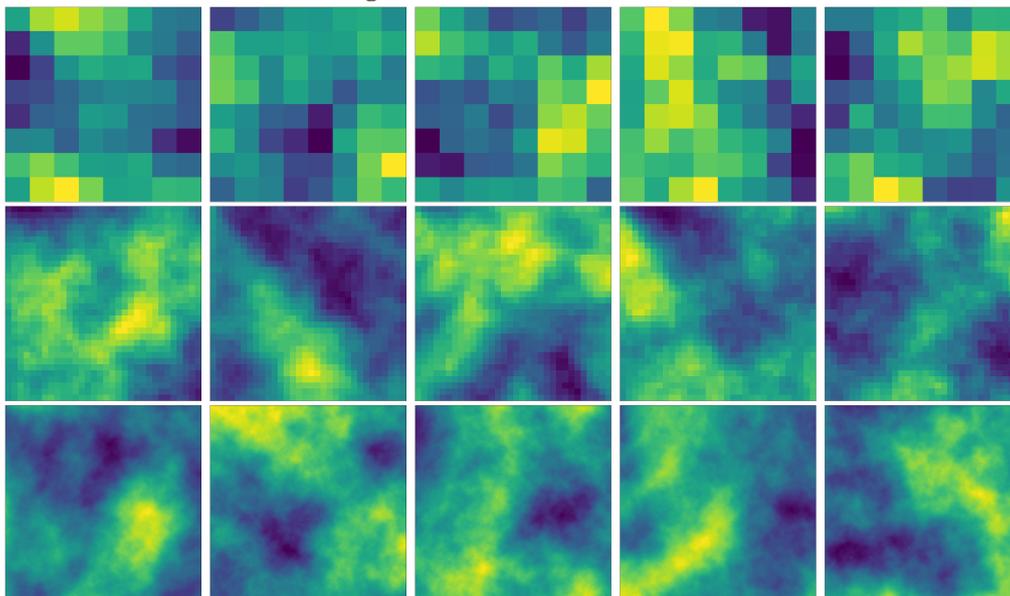
Sample of Perlin matrices, rows ordered according to the matrix size: 8x8,32x32,512x512
SOURCE: Own author.

We observe different patterns according to the sparse matrix size, as we increase the sparse matrix size, the amplitudes of higher frequencies grows. We tested G_1 and G_2, in a set of 100 matrices varying the size of the sparse matrix and the final matrix. The presented test considered two analysis: (i) without phase and module tolerance, (ii) with 0.1% of module tolerance and 0.01 rad of phase tolerance.

In table 2.2, is observed the typical values for G_2 in Perlin matrices. Notice that G_2 converges[3] to the approximate value $1.91 \sim 1.95$ when no tolerance is established, which cannot be distinguished from a totally random pattern. However, when it is considered a tolerance the typical value for G_2 in Perlin matrices is $1.78 \sim 1.88$, which is distinct in most of cases to the random matrices. Thus, the tolerance is required for measuring patterns.

---

[3]Considering a range of one standard deviation

Table 2.2 - G_2 for Perlin matrices

| | (i) Without tolerance | | (ii) With tolerance | |
Matrix size	Average	Standard Deviation	Average	Standard Deviation
4x4	1.629	0.2081	1.629	0.2081
8x8	1.7876	0.1026	1.7876	0.1026
16x16	1.8395	0.0799	1.8388	0.0806
32x32	1.8786	0.0612	1.8755	0.0609
64x64	1.8797	0.0527	1.8738	0.0525
128x128	1.8897	0.0561	1.8759	0.0565
256x256	1.9045	0.0485	1.8716	0.0499
512x512	1.9104	0.0451	1.8295	0.0459

2.5 Conjectures

Since the first work related to GPA(ROSA et al., 1999), a set of conjectures related to this metric has been proposed. Here, we do not intend to prove these conjectures since it demands a detailed mathematical analysis. However, it is important to mention these conjectures since some of these conjectures are observed in practice, and may explain the results in presented applications. In order to showcase some examples of matrices in Figure 2.10, which show four samples of matrices that exemplify the conjectures.

As described by Rosa et al. (1999), the insurgence of a central structure reduces every operator magnitude. The matrix (a) is the basis for this test, a two-dimensional symmetric structure with a random noise. When added a central structure[4] to matrix (a), we obtain the matrix (b). Notice that vectors near the central structure are conduced in direction of the center, thus some of these vectors become symmetrical.

Another important conjecture is the central misalignment, if a central structure is drifted from its position, then we observe an increase of asymmetrical vectors. This is observed in matrix (c), where the central structure from matrix (a) is replaced by the nearby elements. Notice that the number of asymmetrical vector increases when the central structure is drifted.

---

[4]A gaussian matrix with low standard deviation

Figure 2.10 - Conjecture samples
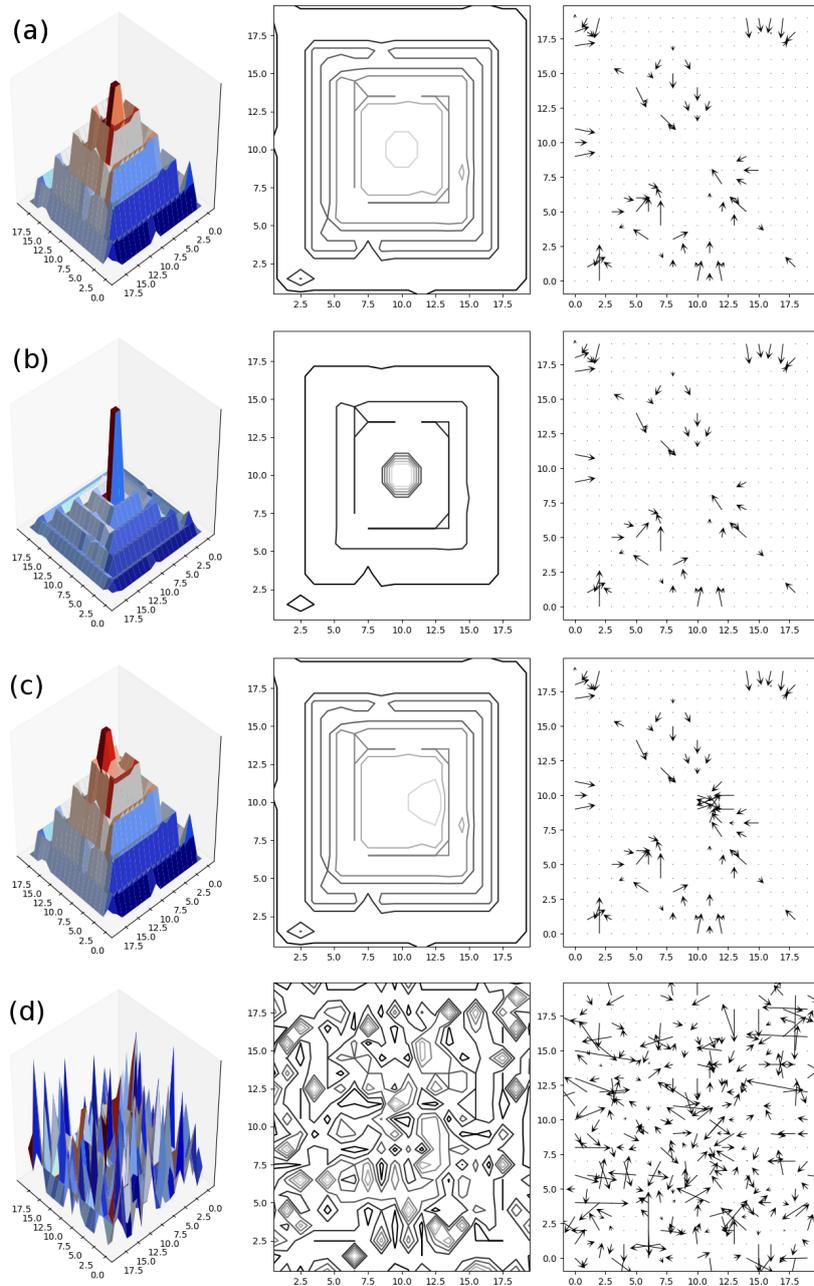
Four pattern inspired on the examples of Rosa et al. (1999). The first column represents the amplitude, the second column is the matrix contour lines, and the third column is the asymmetrical gradient. In (a) it is shown a pattern with central structure contaminated. The pattern (b) shows the pattern (a) when added a secondary central structure. Pattern (c) shows pattern (a) with central structure drifted, and finally, pattern (d) shows pattern (a) with a random permutation.

SOURCE: Own author.

At last, the fragmentation from a matrix usually increases the operator magnitude. The last conjecture can be observed on matrix (d), where this matrix were built by shuffling matrix (a). This last conjecture raise a question that still unresolved: Which matrix maximizes the GPA magnitude? We presented in example (d) a process which increases the system fragmentation. However, it is not known the degree of fragmentation expected on shuffling process, nor which initial condition maximizes GPA. Additionally, a formal prove is required to determine, whether a random matrix, or a shuffled matrix produces the highest GPA magnitude.

Considering GPA preamble consolidated, in next chapter we start the first study case. In next chapter we view the dynamical study case, we analyze Couple Map Lattices with GPA.

3 FIRST CASE STUDY

In this chapter we showcase a study in Bidimensional Coupled Map Lattices (2D-CML). This system has an important role on encryption(SUN et al., 2008), study of chaos control(DATTA et al., 2007), and network synchronization(ZHANG et al., 2018). In order to analyze this system, especially with respect to its behavior, usually the Largest Lyapunov exponent (LLE) is employed, which is a metric that indicates the chaotic behavior. However, interesting features, for example symmetry breaking, short-time series regime, and synchronization are better described by spatial analysis, such as the GPA. Also, it is important to note that 2D-CMLs are a canonical study case for GPA (see for example Ramos et al. (2000)).

This chapter is not directly related to a practical application. However, the analysis of this system is important as a benchmark for dynamical system analysis. Also, some of features are not well explored in the literature, and it have important implications on dynamical systems and the secondary applications.

In this chapter it is presented an analysis of phase-space of this system, which indicates the system dynamics with respect to its gradient. And finally, we introduce an analysis with G_2, in two phase transitions: symmetry breaking (section 3.4) and synchronization of chaotic oscillators (section 3.5). Additionally, in appendix D, it is presented an analysis with the Largest Lyapunov Exponent (LLE).

3.1 CML Overview

Coupled Map Lattice (CML) are dynamical systems discrete in space/time, and continuous state variables(KANEKO, 1992). This system is given by a regular grid of oscillators. Each state of a 2D-CML is represented by a lattice A^t composed by amplitude elements $a_{x,y}^t$, where x and y represent the spatial location and t the lattice snapshot. Each element of this matrix updates the current state with the local oscillator and the neighborhood (here denoted by $a_{i+k,j+l}^t$) according to a transition function (for instance the equation 3.1).

In equation 3.1, $f(a_{i,j}^n)$ is a map (usually a chaotic map), and ϵ is the coupling factor, which represents the neighborhood influence, typically this parameter ranges between 0 and 1.

$$a_{i,j}^{n+1} = (1 - \epsilon)f(a_{i,j}^n) + \sum_{k,l} \frac{\epsilon}{N} f(a_{i+k,j+l}^n) \qquad (3.1)$$

Notice that the equation 3.1 is a general form of CML, which considers any type

of neighborhood. For this work, it is established a 4-connection neighborhood that represents a bidimensional system. From equation 3.1, the 4-connection CML is given by equation 3.2.

$$a_{i,j}^{n+1} = (1 - \epsilon)f(a_{i,j}^n) + \frac{\epsilon}{4}(f(a_{i-1,j}^n) + f(a_{i+1,j}^n) + f(a_{i,j-1}^n) + f(a_{i,j+1}^n)) \qquad (3.2)$$

A general scheme that describes an iteration of a CML connectected in a 4x4 scheme is shown in Figure 3.1. Notice that, a given oscillator depends only on its neighbors to update the local state. However, the local state depends of all oscillator in a long time iteration, as all oscillator are connected.

Figure 3.1 - CML grid in a 4x4 connection scheme
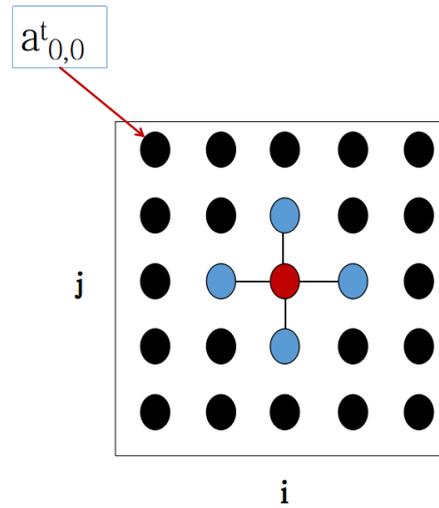
Scheme for CML transition, in grid with size (i=5,j=5). For a given element in red, the next state depends on the local amplitude and the amplitude of its neighborhood, which is composed by the elements in blue.

SOURCE: Own author

3.1.1 Maps

A diversity of visual patterns are also observed according to the chosen map. In this work we highlight three maps: Logistic, Doubling and Shobu-Ose-Mori (SOM). The

phase-space [1] for these maps are shown in Figure 3.2. We selected these maps with the described parameters due to the different dynamic near the inflection point, which is $x \approx 0.5$ for Logistic and Dyadic map, and $x \approx 0.8$ for SOM map.

Figure 3.2 - Chaotic maps
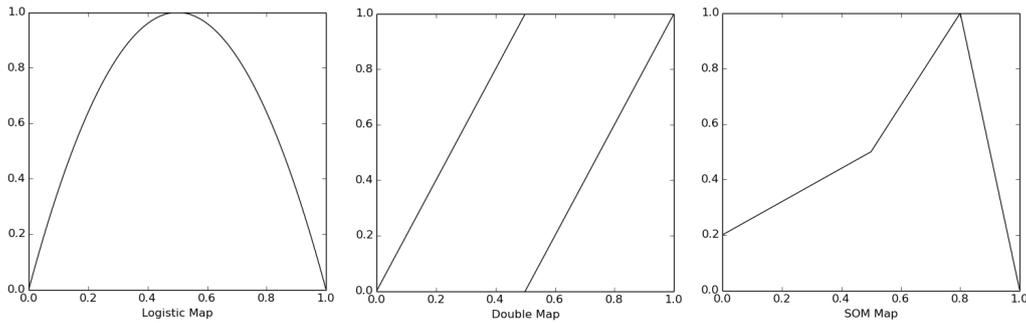
SOURCE: Own author

The leading map in which our tests were applied is the Logistic map. This map is a model for population growth, which can show a chaotic dynamics depending on the parameter (α). The Logistic map equation is: $x^{t+1} = \alpha x^t(1-x^t)$, where x^t represents the state in time t. In this work we apply the parameter $\alpha = 4.0$ for all systems, as it is known that the map dynamics with this parameter is chaotic (SMALE et al., 2003).

The Doubling map, also known as Dyadic map, Bernoulli map, 2x mod 1, and bitshift-map, is a chaotic map given by equation 3.3.

$$x^{t+1} = \begin{cases} 2x & , 0.0 \leq x < 0.5 \\ 2x - 1 & , 0.5 \leq x \leq 1.0 \end{cases} \tag{3.3}$$

The SOM map was proposed to study the intermittency phenomena (SHOBU et al., 1984), in special the intermittency of type I on maps. The equation for SOM map is shown in 3.4, where α and β are free parameters, and $\gamma = 0.8/(1 + \alpha)$. Notice that this oscillator has two main regions: a laminar flow ($0 \leq x < 0.8$), and a burst region ($x \geq 0.8$).

---

[1] The phase-space is graph that relates a state of the map (x-axis) with the next state (y-axis)

$$x^{t+1} = \begin{cases} \alpha x^t + 0.2 & , for \ x^t \leq \gamma \\ \alpha(x^t - 0.8) + 1 & , for \ \gamma < x^t < 0.8 \\ (1 - x^t)/\beta & , for \ x^t \geq 0.8 \end{cases} \qquad (3.4)$$

In this work, we apply the parameters $\alpha = 0.6$ and $\beta = 0.2$. We select these parameters due to the presence of a saddle-node bifurcation (KOBAYASHI et al., 1989). The equilibrium points, for the first return map, with these parameters are: $x = \gamma = 0.5$ and $x = 1/(1 + \beta) = 5/6$.

Here we complete the CML overview. At the following section we view the case study of this work.

3.2 Case Study

Depending on the properties of the system, different patterns are observed. In this work, we selected four cases study to describe the system dynamics. We analyze the systems, described in Table 3.1, at the following sections.

Table 3.1 - CML study cases

	A	B	C	D
Dimension	128x128	32x32	128x128	128x128
Initial Condition	Random	Gaussian-2D	Gaussian-2D	Random
Boundary	Toroidal	Toroidal	Toroidal	Toroidal
ϵ	1/2	0 to 1	1/2	1/2
Map	Logistic	Logistic	Logistic, Dyadic, SOM	Logistic
Iterations	7,000	1,000(λ), 4,100(G_2)	140	1,000
Observed	Gradient (at center)	λ,G_2	G_1, G_2	G_2

3.3 Gradient Phase Space

In this section we explore the system behavior on matter of phase-space. This analysis is one of simplest test for detection of attractors and is an important property for understanding the dynamics of this system.

We analyzed the system A, shown in Table 3.1. Since our aim is to analyze the spatial behavior, and GPA is a formalism that describes system based on the spatial gradient, we measured at every time step (t) the gradient phase (θ^t) and modulus(r^t)

of a vector at the center of the system. As we intend to analyze the long-range regime, the first 4,000 iterations were discarded. The gradient was measured with finite difference (as described in section 2.3), the vector angle was measured from gradient using the function 'atan2'. The results are shown in Figure 3.3.

Figure 3.3 - Gradient phase space
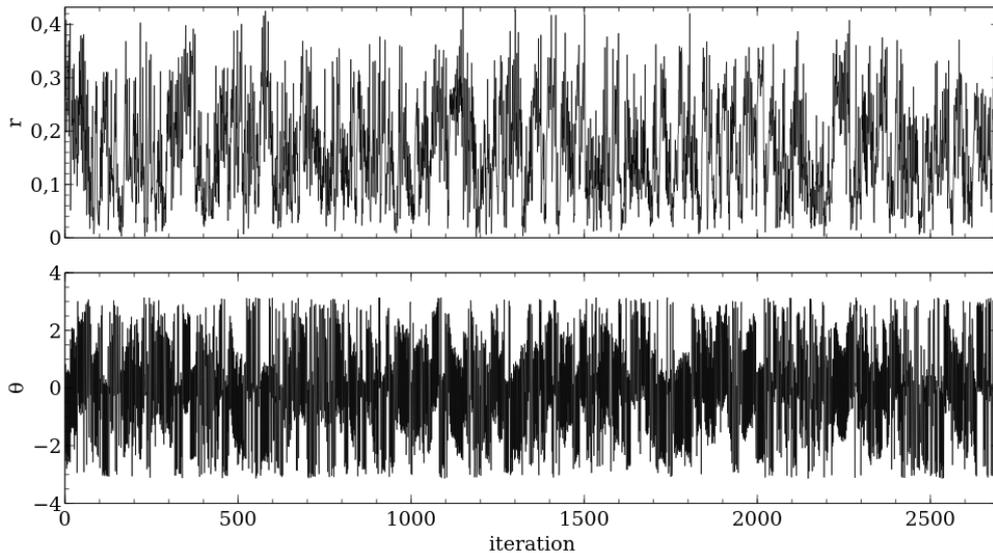
SOURCE: Own author

To inspect this system, the phase-space is plotted in Figure 3.4. Each point in this plot represents the gradient of an element in a given state system state (in x-axis) versus its following state (in y-axis), where each state is represented by the amplitude in the time series.

As shown in Figure 3.4, there are two lines in θ_n (angular coefficient $\alpha \approx 1$) with high density of points. It indicates that, given a system state, we expect the same phase for the next state ($\theta^{n+1} = \theta^n$). On the other hand, the dynamics of modulus shows a different pattern, we do not expect a regular dynamic for the modulus. Additionally, notice that the phase-space is not completely filled with points, which means that for a random dynamic of the system all states are not equally probable, the system is not ergodic.

27

Figure 3.4 - Gradient phase space
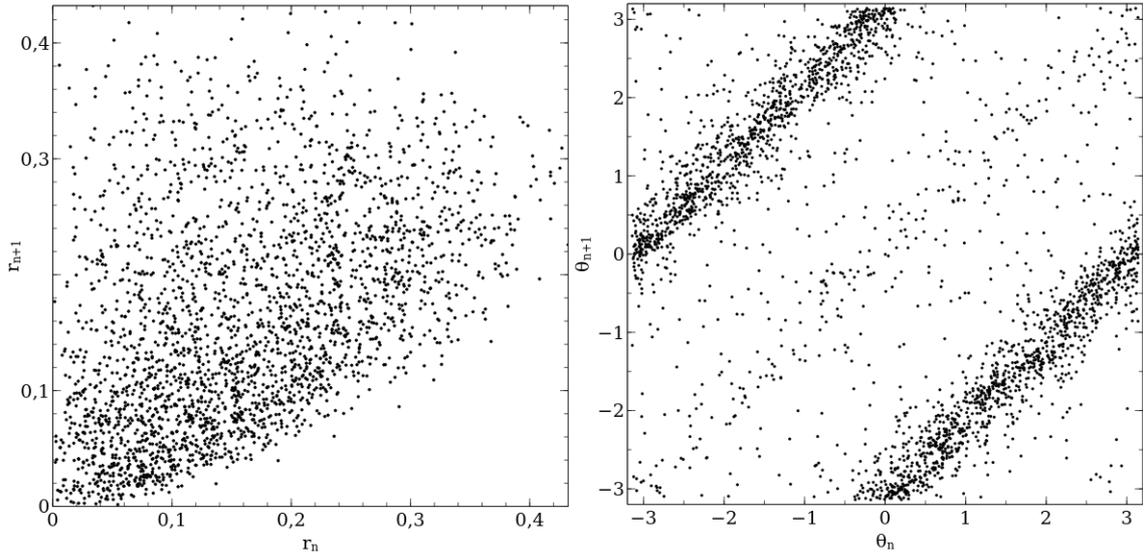
SOURCE: Own author

Here we end this brief analysis of the CML gradient dynamic. We infer that the first and second gradient moments are the best technique in GPA formalism for analysis since the gradient modulus of CML has an interesting dynamic. In the following section, we continue to explore the CML dynamic, this time with an analysis of the system symmetry with GPA.

3.4 Symmetry breaking

The sensitivity to initial conditions is an important property observed on dynamical systems. In spatially extended systems the sensitivity to initial conditions (in maps) drives system symmetry breaking. Here, we describe the spontaneous symmetry breaking, which is a phenomenon that starts from a symmetric initial condition, in CML with different maps.

Different visual patterns are observed on changing the map, and the system parameters. Here we fixed the coupling factor, type of boundary, and initial conditions according to the column C from Table 3.1. As we can observe in Figure 3.5, starting from a Gaussian matrix, a complex pattern arise, and eventually occurs the system symmetry breaks.

It is important to note that the system symmetry breaking occurs in different periods

and velocities depending on the map. Due to the intermittencyfootnoteIn this case an intermittency of type I: a dynamic between large laminar periods and small bursts periods in SOM map CML the system break starts in the same time of Doubling map. However, SOM breaks the symmetry with less speed.

Figure 3.5 - Symmetry breaking samples
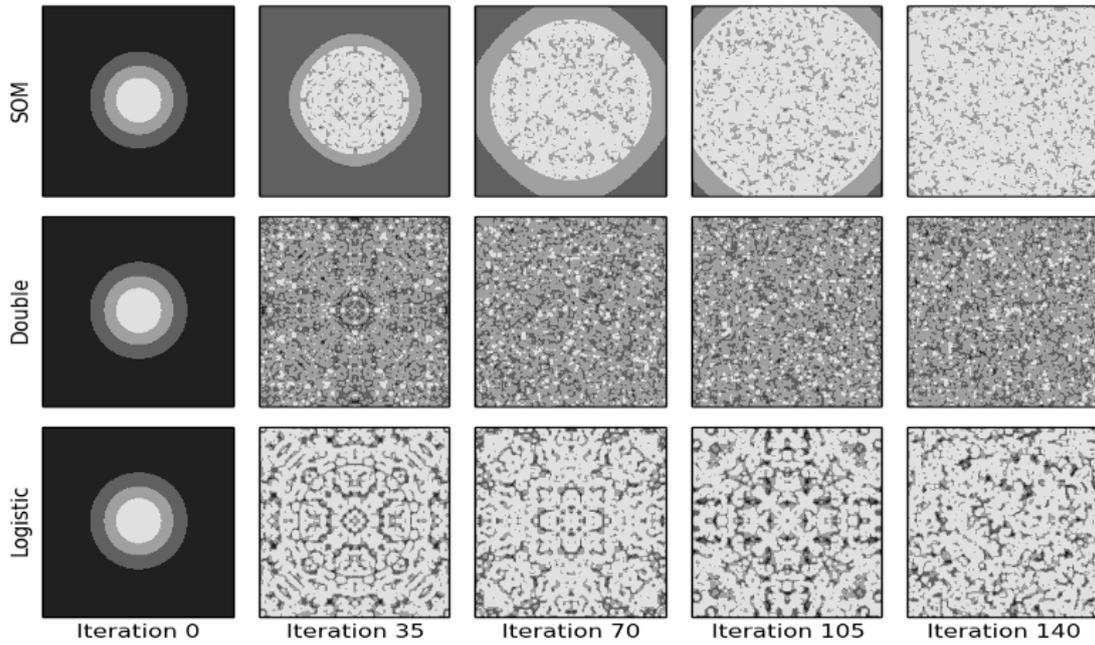
Snapshots at every 35 iterations, of CML dynamic, considering every map.

SOURCE: Own author

In order to visualize the transitions, we present in Figure 3.6, the symmetry breaking observed via G_1. Notice that the time series has three levels of G_1. The first level represents the initial condition, where the system is completely symmetric ($G_1 = 0.0$), whereas the second level is only observed in logistic and doubling maps. Lastly, the third stage represents the asymmetrical system state.

It is important to note that the intermediary state occur due to the system interaction with the boundaries. We observe the asymmetrical vectors are at boundary, at the intermediary state. The effect does not occur on SOM map CML due to the laminar phase that is state as initial condition near the boundaries.

29

Another technique for visualization of the state transition is the phase-space plot that is also shown in Figure 3.6. In phase-space we can observe two clusters of points, where the first cluster represents the boundary effects, and the second represents the symmetry breaking.

Figure 3.6 - G_1 in symmetry breaking transition
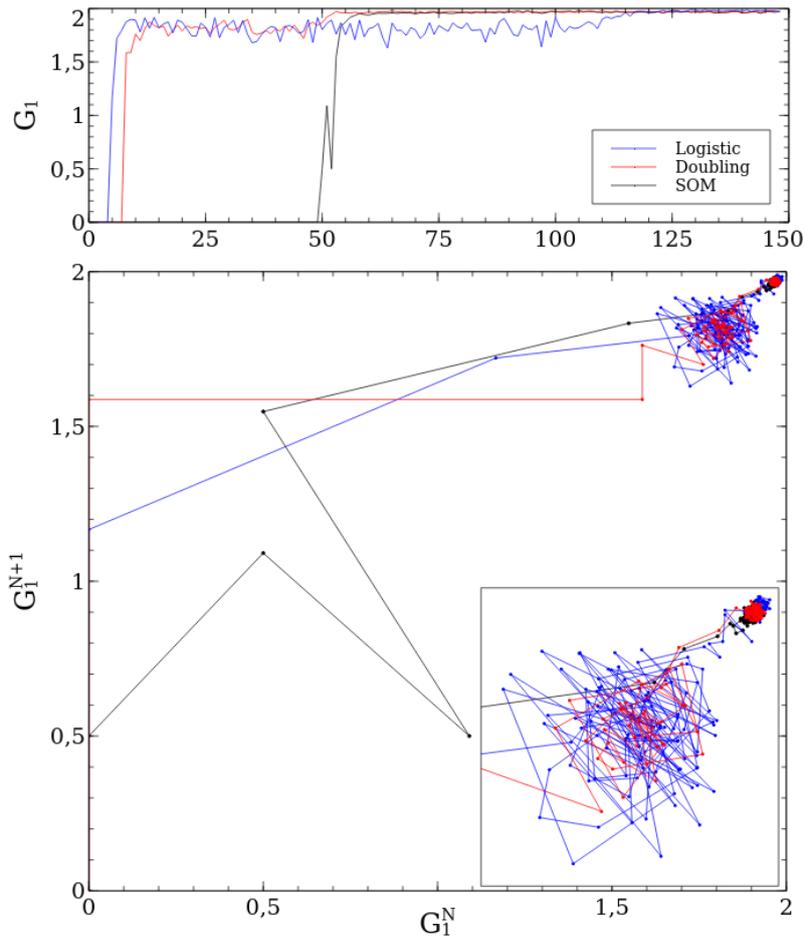
In top side, the G_1 temporal series for symmetry breaking. On bottom side, the phase-space of the temporal series. The square represented a magnified region of clusters.

SOURCE: Own author

Analyzing the same system with G_2, we obtain the time series from Figure 3.7. Notice that $G_2 \neq 0.00$ for CML with Logistic and Doubling maps after the symmetry breaking. As observed in G_1, this threshold is generated by small fluctuations that appear near the boundaries. However, G_2 response and fluctuation at this interme-

diary effect is almost constant. It indicates that G_2 is less sensitive than G_1 to small fluctuations.

Figure 3.7 - G_2 in symmetry breaking transition
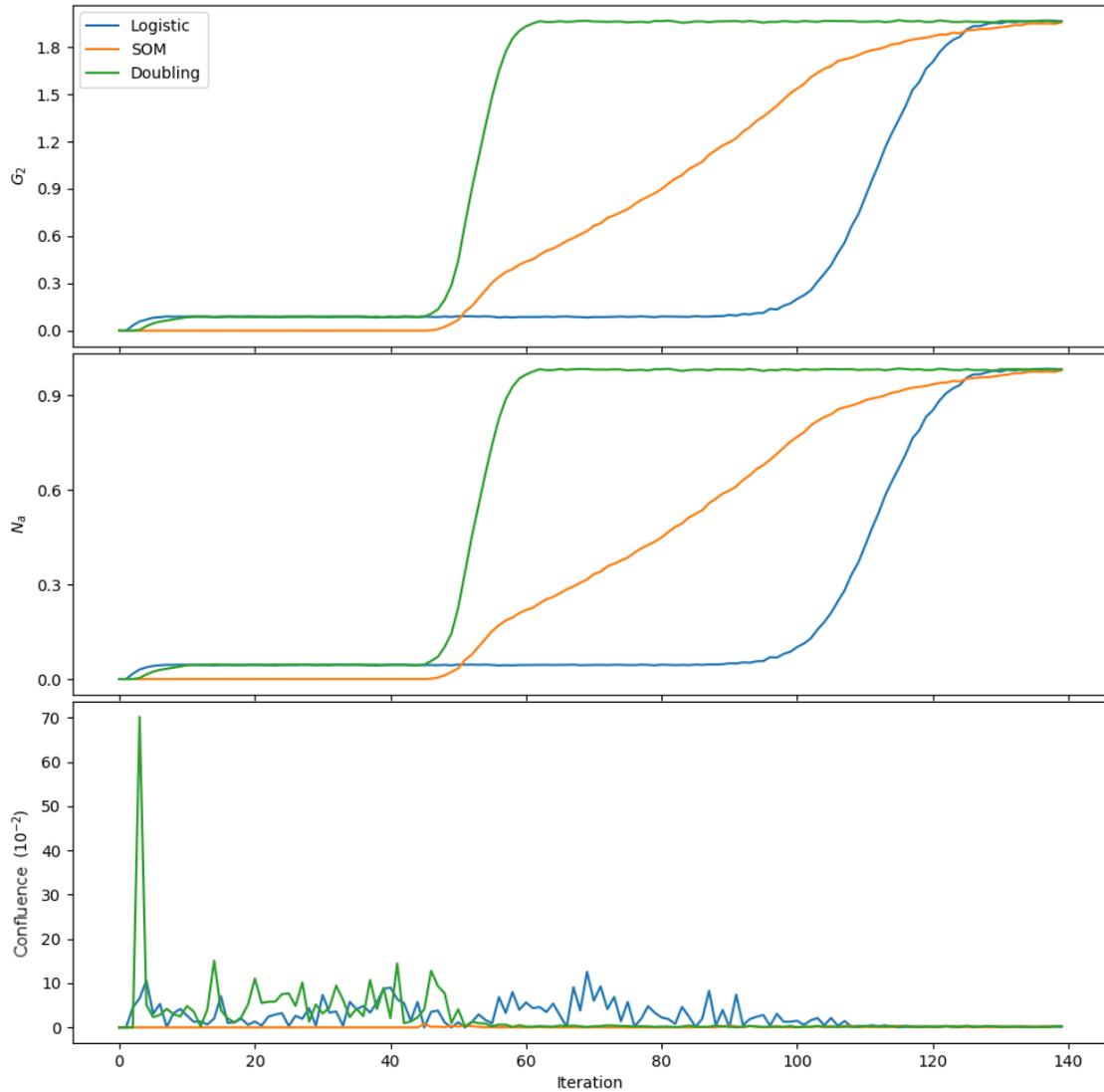
Following the analysis of G_2, notice that, it is observed a high value of similarity (especially for Doubling map) at beginning followed by small values of confluence. It means that, the system first interaction with the boundaries produces asymmet-

rical vectors with similar magnitude and direction. After the first interaction, the asymmetrical pattern has vectors in all directions with different magnitudes.

Proceeding with the study of this phenomenon, it is proposed a secondary test that concerns on system parametrization. Selecting the system B, we intend to understand the coupling (ϵ) and the map parameter (α) effect in Symmetry breaking. In this experiment, it was selected 32x32 maps starting with 2D-Gaussian, after iterating 4,000 times, we measured the G_2 average of the next 100 iterations. It is presented in Figure 3.8 the results of this test.

Figure 3.8 - G_2 phase space for 32x32 CMLs in Symmetry Breaking
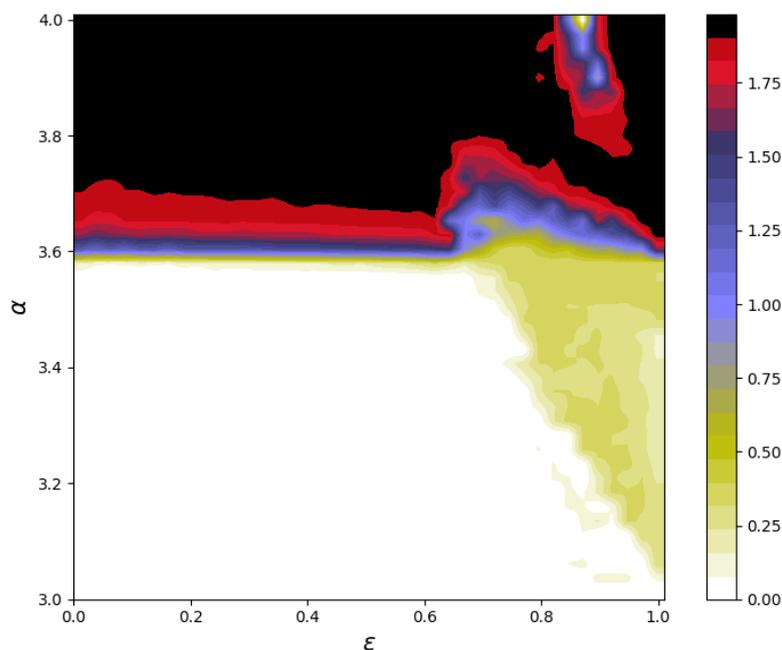
Average of G_2 measured on 100 matrices, after iterating 4,000 times. The initial condition is a 2D-Gaussian. The grid density is: $\Delta\epsilon = \Delta\alpha = 0.025$.

SOURCE: Own author

The Figure 3.8 show that the symmetry breaking depends only on map parametrization. For $\alpha > 3.6$, the value of G_2 converges to the maximal value. In fact, we can also observe the chaotic behavior for $\alpha > 3.57$ in logistic map (see the analysis of appendix D, in Figure D.2).

However, there is an exception to this statement, when the coupling factor is high enough ($\epsilon > 0.7$), complex structures arises. As a consequence, with high enough coupling factor and long iterations, the symmetry breaking also occurs.

Some studying still required to understand this phenomenon. However, we end the analysis of this state transition here since a more detailed study of this phenomenon, would be beyond the scope of this work. In the following section we showcase another type of state transition, which was also analyzed with GPA.

3.5 Synchronization

It is well known that depending on the coupling parameter and the map parameter(s), systems composed by oscillators can synchronize. This process is observed in homogeneous[2] and heterogeneous systems (see for example Zhang et al. (2018)). In this section, we showcase a homogeneous synchronization of 2D-CML, which is given by the system D in Table 3.1. The visual pattern (Figure 3.9) that arises in this process is a groove, each oscillator with high amplitude has low amplitudes neighbors and vice versa. There are several filaments in the observed pattern, these filaments are produced by the interaction between regions which are locally synchronized.

In Figure 3.10, it is shown the value of G_2 with angular tolerance of 0.01 rad and 0.1% of module tolerance for this CML. The data were plotted as a time-series with 1,000 observations, additionally it is shown the asymmetrical proportion and the similarity in the time series.

Figure 3.9 - Synchronization samples
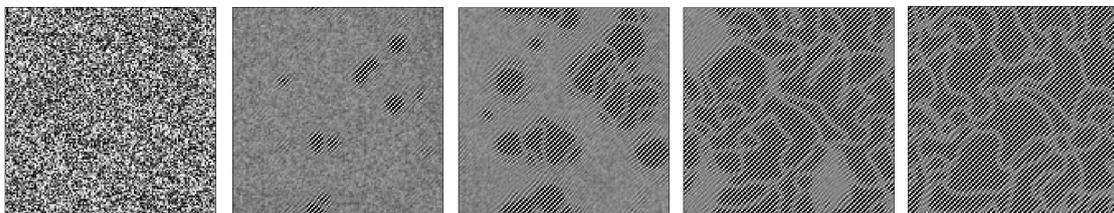
Samples at every 50 iterations in a system in Synchronization state transition
SOURCE: Own author

---

[2]Here we denote as homogeneous systems with the same map function

Notice that the pattern has a characteristic interval for G_2, the average of G_2 is 0.954 and standard deviation 0.093 between the iterations 200 to 1,000. The dynamic of this system when synchronized consists of the groove and the movement of the filaments, it explains the high fluctuations on the observed interval.

Figure 3.10 - G_2 in synchronization transition
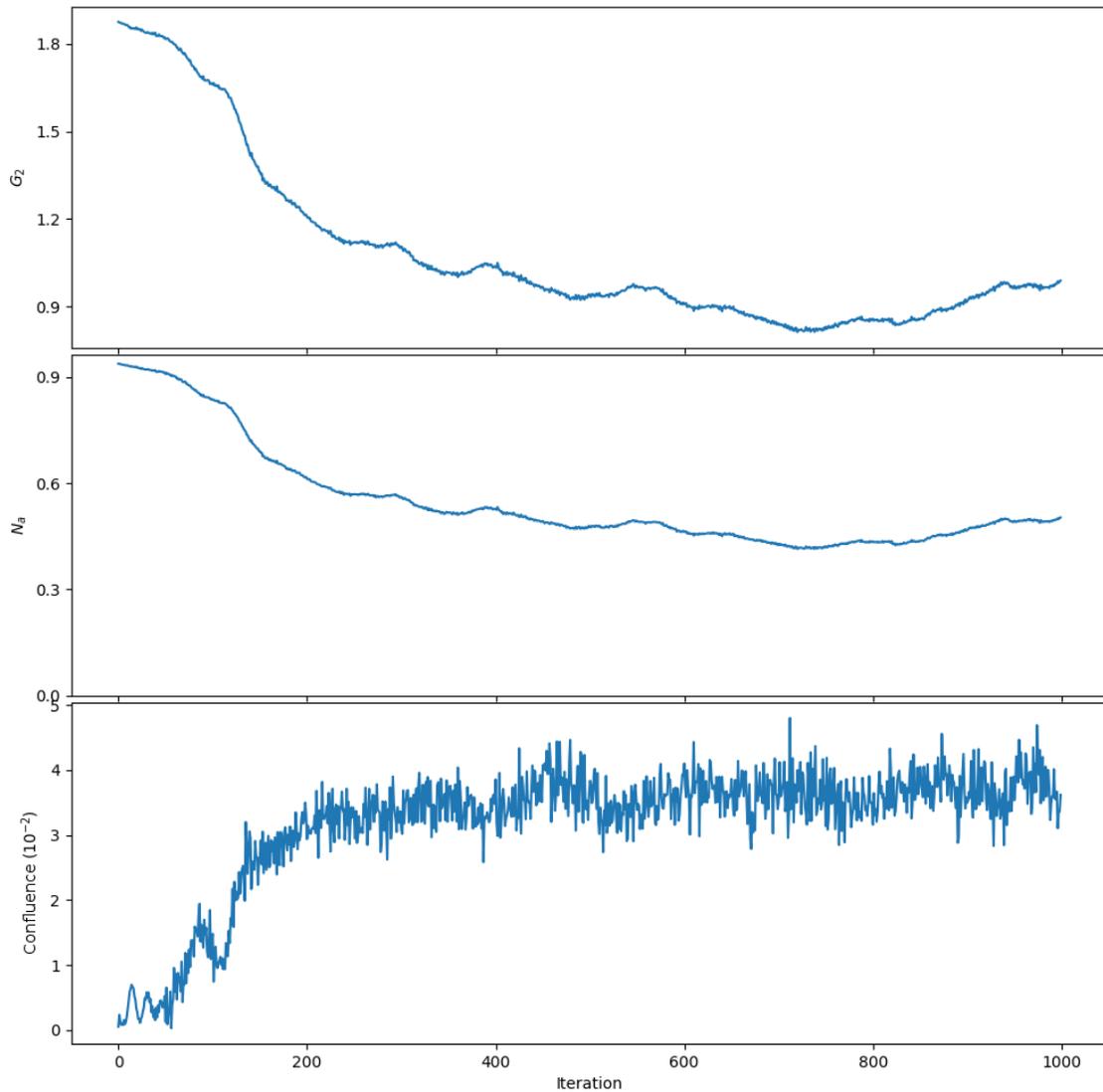
SOURCE: Own author

In order to determine the region that the synchronization occurs, and also under-

stand the system behavior with random initial conditions, a secondary test is presented in Figure 3.11. In this test for each set of parameters (ϵ, α) , it is measured G_2 average of 100 long-range iterations. The long-range iteration starts with a new randomized matrix, in which the model iterates 4,000 times. Due to the high cost for computing this test, the matrix of this test was reduced to 32x32.

Figure 3.11 - G_2 phase-space for 32x32 random CMLs
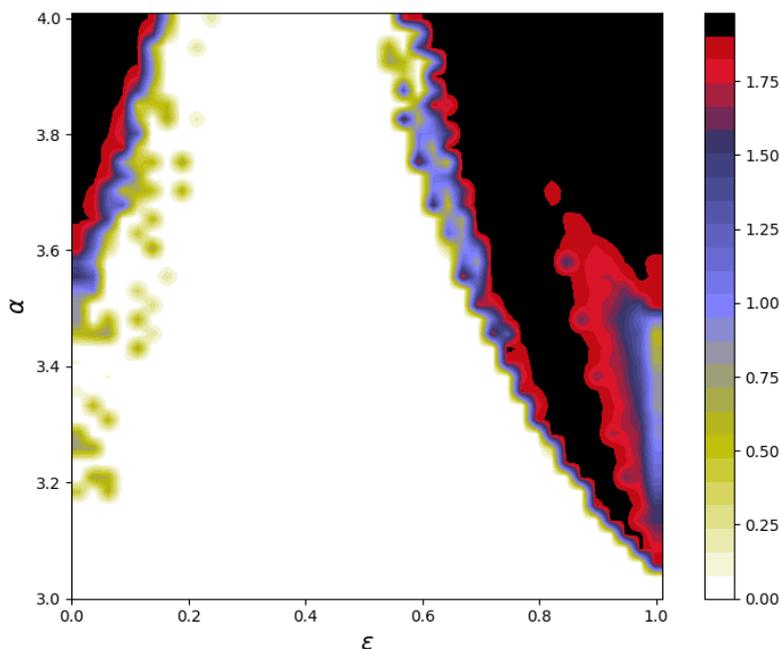
Average of G_2 of 100 random matrices, after iterating 4,000 times. The grid density is: $\Delta\epsilon = \Delta\alpha = 0.025$.

SOURCE: Own author

Notice that, in this test, the region that describes the groove pattern are the blue areas $(G_2 \approx 1.0)$, for instance the region $0.9 < \epsilon < 1.0$, $3.1 < \alpha < 3.5$.

We had shown that G_2 is able to characterize CML transient behavior. In the Appendix D, we show the Largest Lyapunov Exponent (LLE), which phase-space (Figure D.1) can be compared to the G_2 phase-space shown in Figures 3.8 and 3.11. Observing the region $0.2 < \epsilon < 0.6$ and $\alpha > 3.57$, we can conclude that G_2 does not detect long-range regimes, and it is dependent to the initial condition. However, it detect transient phenomena, which are not explicit in the LLE phase-space. An

example of the system at this region is shown in Figure 3.5, observe that the system is asymmetric. However, if we start the same system with another initial condition, we obtain the pattern from Figure 3.9. Notice that both patterns are completely different. With respect to the system dynamic in LLE space representation, the system has no significant unstable orbits. Thus, G_2 is detecting another kind of feature, being a complementary tool for regime characterization.

From this analysis of state transition in 2D-CML patterns, we end up the application of GPA on a typical chaotic dynamical system. In order to refine this application, a further analysis with other metrics is still necessary. However, for the purpose of this work the results are able to establish the GPA performance. In the next chapter, we perform the gradient pattern analysis on digital images of galaxies in order to study the GPA performance for morphological analysis.

4 SECOND CASE STUDY

A major challenge in cosmology is the environmental influence in formation and evolution of galaxies. It is well known that early-type galaxies prevail in crowded environments, and late-type galaxies prevail at barren environments (DRESSLER, 1980; MO et al., 2010; TEMPEL et al., 2011). However, interesting insights have been observed in a more accurate analysis. The most recent morphology techniques allowed the detection and environmental study of peculiar galaxies in high redshift, for example spiral galaxies with high metallicity (TEMPEL et al., 2011), unusual blue galaxies (TEMPEL et al., 2011; SHAMIR; WALLIN, 2014), and tidal-distorted morphologies (SHAMIR; WALLIN, 2014). However, the analysis of high red-shifted galaxies is a challenge, since the image quality decrease with the aimed object distance.

There is also a demand for fast classification algorithms since there is a constant increase of the data volume and variety. For instance, the project Sloan Digital Sky Survey (SDSS) constantly increase the sky coverage area and meta-data. SDSS-1 released a catalog with 53 million objects (approximately 14GB), SDSS-7 cataloged 357 million objects (approximately 15TB), and 933 million objects (approximately 28TB of raw data) were released recently in SDSS-14.

A recent study from Strieder (2010) introduces the G_1 to the morphological galaxy classification, producing suitable results. In this chapter, it is presented a comparison between G_1, G_2, and the improved version of classical non-parametric morphometrics.

Here, we present a pipeline called CyMorph (abbreviation of Cython Morphology), which is part of a project (N°2014/11156-4). The main goal of this pipeline, in the context of the project, is to pre-classify galaxies, to a more accurate analysis with Galphat(YOON et al., 2011). Additionally, two papers related to this pipeline have been published (ROSA et al., 2018; BARCHI et al., 2017).

This chapter contains a description of the dataset, a description of CyMorph, the results and interpretation. This chapter is structured in the following parts: the problem context (section 4.1), dataset description and preprocessing (sections 4.2 and 4.2.1), the objective function for parametrization (section 4.3), morphometrics (sections 4.4 to 4.9), the system usability (section 4.10), and lastly the morphometric distributions (section 4.11).

4.1 Context

Galaxies are systems composed by stars held together by a gravitational force. The galaxy morphology is a fundamental property for understanding the characteristic of these stellar populations. Moreover, in a larger scale, the morphology is important for understanding the universe formation.

Many schemes have been carried out to describe the diversity of structure and properties of galaxy morphology. These schemes explain from visual perspective, the physical-chemical similarities and discrepancies between objects. It is also important to notice that these schemes also implies in a continuity between the object classes, which should explain eventual morphological changes.

One of the earliest scheme –and most agreed to describe nearby galaxy– is the Hubble Tunning fork. As shown in Figure 4.1, the Hubble Tunning fork is majority composed by four classes of galaxies: ellipticals, normal spirals, barred spirals, and irregulars.
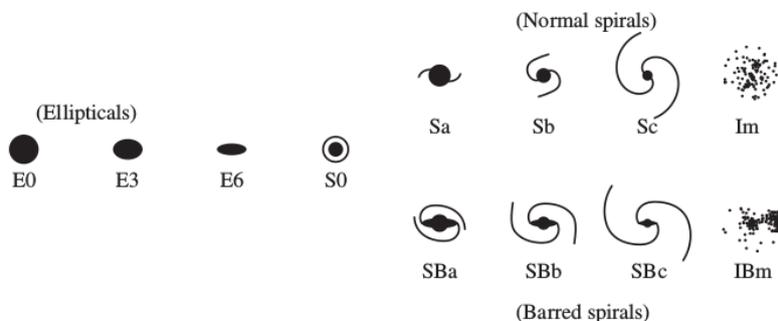
Figure 4.1 - Hubble Tunning fork

SOURCE: Mo et al. (2010).

Elliptical galaxies (also known as Early-type) are preponderantly composed by old stars, with high metallicity[1]. Despite the physical-chemical properties, elliptical galaxies are sub-classified according to the ratio between minor and major axis.

Spiral galaxies (also refereed as Late-type) are systems composed by a diversity of regions, these galaxies are sub-classified in Hubble-scheme by presence or absence

---

[1]In this context, metallicity refers to the proportion chemical elements that are not Hydrogen neither Helium.

of bars, and the spiral arms angle. Notice that these galaxies also have a diversity of regions:

- Bulge - the galaxy central region that has high metallicity stars and low stellar formation
- Spiral arm - a region with low metallicity stars and high rate of stellar formation
- Bar - a structure formed on orbit instability of bulge stellar population, this structure forces outside the bulge gas into the Spiral arms fueling the formation of new stars.
- Disc - a region preponderantly composed by gas, dust and new stars
- Halo - this region has low gas and dust density, and generally contain globular clusters

It is important to notice that Hubble system has also some unexplained galaxy types (for example galaxies with rings), and does not characterize morphologies of interacting galaxies (for example tidal distorted). Furthermore, new schemes such as Hubble-de Vaucouleurs extended the Hubble scheme by including information from other sources then visual.

Due to the objectives of this work, we evaluate a simpler model that still a challenge for classification, when we consider the data volume and complexity. This model organizes the galaxies in two main classes: early-type and late-type.

4.2 Dataset

We selected bright galaxies from Galaxy Zoo1 catalog (LINTOTT et al., 2008), a citizen science project that aims to understand the galaxy formation, especially with respect to the morphology. Galaxy Zoo1 provided a catalog with approximately 890,000 classified objects. In order to retrieve the images, we downloaded field images from Sloan Digital Sky Survey (SDSS), the code to download the set of images is described in Appendix A. The field images used in this study are 2048x2048 images in r-band, since our aim in this pipeline is to detect elliptical galaxies which commonly are brighter in r-band. An example of field image is shown in Figure 4.2, notice, this image contains many objects. In order to filter the image, a preprocessing were applied in Field images, as described in section 4.2.1.

We apply a criterion for large object selection, according to a relation between areas: $\pi R_{50\%}^2(b/a) >= 40\pi(f.seeing_r/2)^2$, where $R_{50\%}$ is the radius containing 50%

Figure 4.2 - Field image example
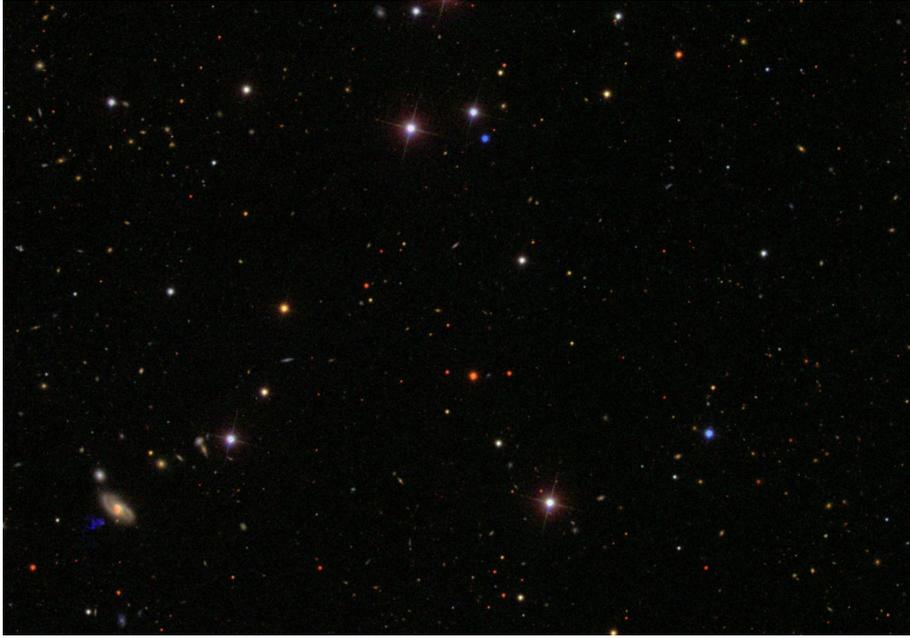
SOURCE: Blanton et al. (2017)

of Petrosian flux(PETROSIAN, 1976), b/a is the ratio between the fitted ellipse that bounds the galaxy, and $seeing_r$ is the average measure of the Full Width Half Maximum from the fitted ellipse. In virtue of the Galaxy Zoo1 preprocessing and our criteria, around 58,000 classified objects were analyzed. All galaxies of this analysis are inside the redshift range: $0.03 < z < 0.1$.

4.2.1 Preprocessing

Incisive image preprocessing techniques are mandatory for morphology, as it ensures the consistency of parameters, and improves the feature extraction. For the morphology analysis, there are two major issues are assigned to this step: the stamp cut, and secondary objects removal.

Figure 4.3 - Pre-processing fluxogram
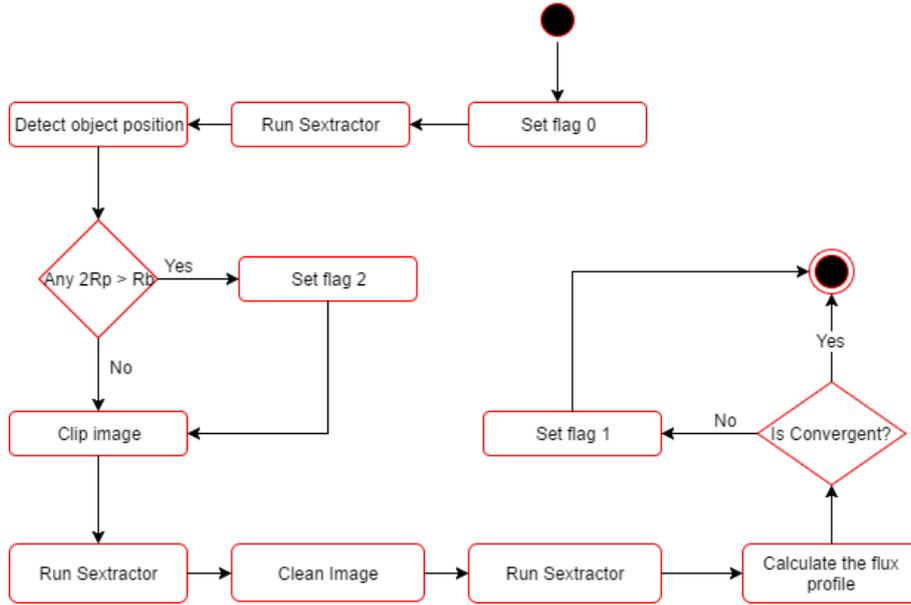
Image pre-processing fluxogram, where R_P is the petrosian distance, and Rb is the minimum distance from the galaxy to the boundary.

SOURCE: Own author

The pipeline for our image preprocessing step is given by the Figure 4.3. In the first step the aimed object is selected from the Field Image using SExtractor (BERTIN; ARNOUTS, 1996). A stamp with maximal distance of $7.5R_P$ around the aimed object is produced in the clipping process. If the object is not entirely inside this image a quality flag is set for post-processing.

In order to clean the image, the pixels from objects, which are not aimed, are replaced by its isophotal level. Where the isophotal level is given by the ellipse, which boundaries have approximately the same amplitude. The isophotal level is approached using a random value from a Gaussian distribution on the aimed object expanded ellipse that intersects the pixel.

Finally a secondary quality test is done by verifying the convergence of the concentration parameter (see section 4.4). This last criteria detect whenever a small objects footnoteHere a small object contain less than 6 pixels contaminates the image. Finally, a stamp image is generated, some examples of stamps are shown in

Figure 4.4.

Figure 4.4 - Stamp sample
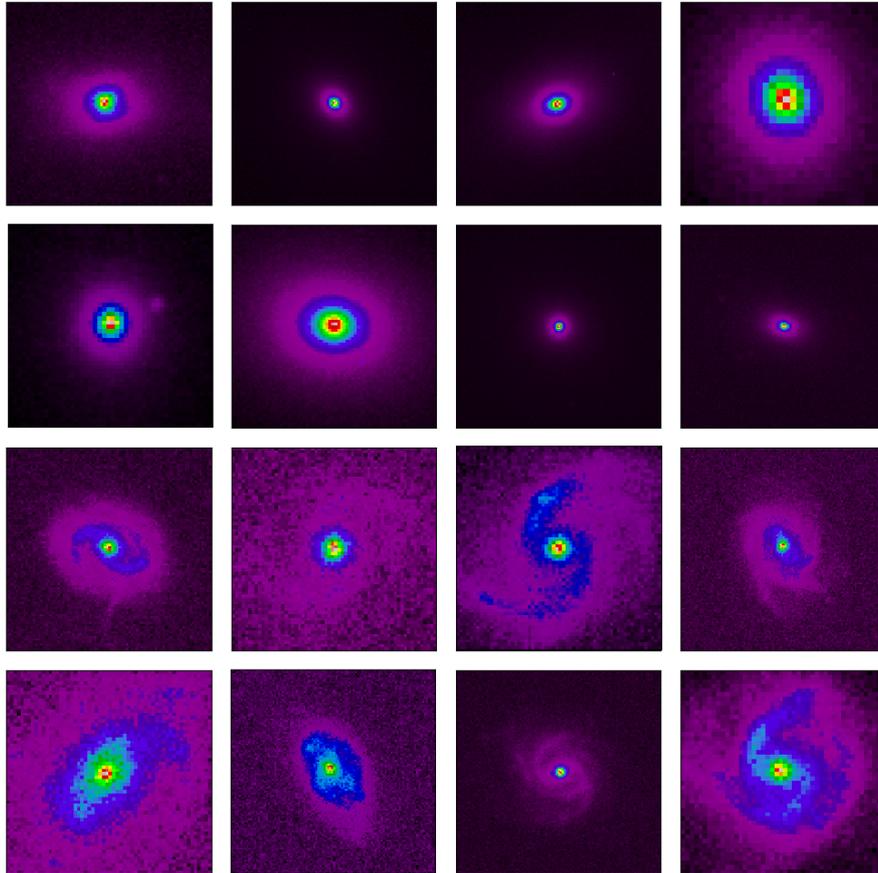
Sample of galaxy stamps(magnified for easier visualization), early-type galaxies at first and second row, late-type galaxies at the end rows.

SOURCE: Own author

4.3 Objective Function

A recurrent problem in morphological analysis is the parametrization, some morpho-metrics receives a set of parameters, for example the smooth degree of clumpiness (section 4.7). Usually an expert settle these parameters. However, this approach is subjective. Moreover, we should expect a global parametrization that optimizes the data. However, there is no unanimity in literature of which set of parameters

should be applied. In this work we present an innovative approach to this problem: we state a set of objective functions, which determines the best parametrization. The proposed framework measures binomial distance, a distance between two set of distributions.

In order to measure these metrics we randomly selected 1,000 galaxies of each type (spirals and ellipticals), and tested with a set of parameters. Since our goal is to separate each morphological class, we selected the parameters that maximizes the objective functions.

There are advantages and disadvantages to this methodology, the advantages of this methodology are:(i) it is an unbiased approach, (ii) numerically compare morphological parameters, (iii) automatize the parameterization. However, this methodology has some issues, we may observe different optimal solution, depending on data variety, and in some cases there is no consensus between the metrics (see for example the entropy in section 4.8).

We select three metrics with different properties: Kullback-Leibler divergence (equation 4.1)(KULLBACK, 1997), Hellinger distance (equation 4.2), and a new metric based on the geometrical distance.

$$KL(P_1||P_2) = \sum_{\phi \in \Phi} P_1(\phi) log_2 \left(\frac{P_1(\phi)}{P_2(\phi)} \right) \tag{4.1}$$

$$D_H(P_1, P_2) = \frac{|\sum_{\phi \in \Phi} \sqrt{P_1(\phi)} - \sqrt{P_2(\phi)}|}{\sqrt{2}} \tag{4.2}$$

The Kullback-Leibler measures the degree of divergence from a distribution P_2 to P_1. Notice that KL is not a symmetric operation $(\exists (P_1, P_2)|KL(P_1||P_2) \neq KL(P_2||P_1))$, neither bounded. A common measurement that is invariant to the referential distribution is the average between $KL(P_1||P_2)$ and $KL(P_2||P_1)$, here denoted as $D_{KL}(P_1, P_2)$. This solution leads us an unbounded symmetric operation that ranges between 0 and ∞, where 0 represents similar distributions and ∞ represents distant distributions. In the presented analysis, we have not considered $KL(P_1||P_2)$ and $KL(P_2||P_1)$, instead our analysis uses $D_{KL}(P_1, P_2)$.

The Hellinger distance quantify the similarity degree between distributions P_1 and P_2, by means of the euclidean norm. This metric is bound in the interval $[0, 1]$, where 0 represent similar distributions and 1 represents distant distributions.

In addition, we claim for a new metric that depends on the Geometric Histogram Separation (GHS). The GHS metric(δ_{GHS}) considers the area of separation (δ_{BCA}) and the length distance (δ_{BCL}). The features measured in each distribution is shown in Figure 4.5, where features of two distributions(R and B) and the intersection(RB) are measured.

Figure 4.5 - Example of binomial distribution and the GHS features
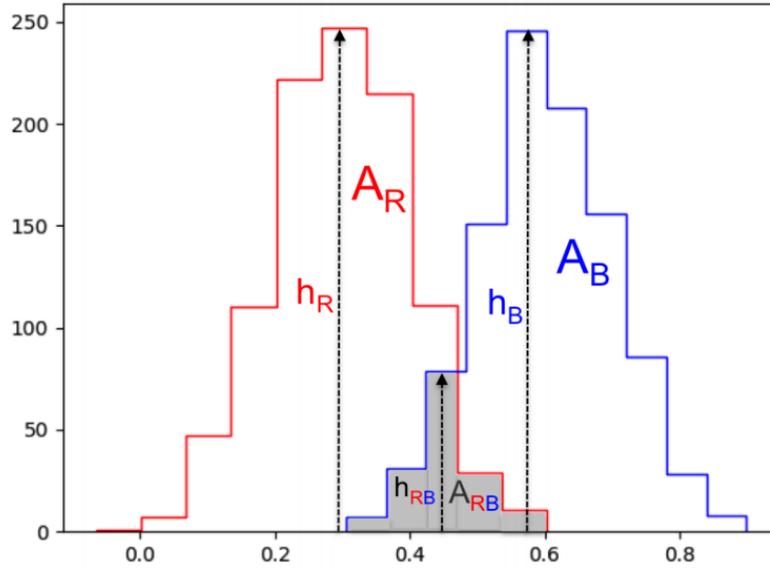
Example of binomial distance between a distribution in red (R), and another distribution in blue(B). The intersection between both distribution is in gray(RB). The values A_R, A_B, and A_{RB} are the distribution areas. The values h_R, h_B, and h_{RB} are the distribution heights.

SOURCE: Own author

Considering the distribution areas A_R, A_B, and A_{RB}, the area of separation is measured as the equation 4.3. Notice that the lower intersection area, the greater is δ_{BCA}, also notice that this feature is normalized according to the total area($A_R + A_B + A_{RB}$). As canonical examples, when both distribution are equal then, $A_R = A_B = 0$ and $A_{RB} = 1$, therefore $\delta_{BCA} = 0$. Whereas, if both distribution are detach, then $A_{RB} = 0$ and $\delta_{BCA} = 1$.

$$\delta_{BCA} = 1 - \frac{A_{RB}}{A_R + A_B + A_{RB}} \tag{4.3}$$

The length separation is measured according to equation 4.4, where h_R, h_B, and h_{RB} are the distribution lengths. Notice that δ_{BCL} is also normalized, and the extreme values are the same of δ_{BCA}.

$$\delta_{BCL} = \frac{h_R + h_B - 2h_{RB}}{h_R + h_B} \tag{4.4}$$

Finally, the GHS is measured as according to equation 4.5. Since we are comparing an one dimensional metric with a bidimensional metric, it is applied the square root of δ_{BCA} to measure δ_{GHS}.

$$\delta_{GHS} = \frac{\sqrt{\delta_{BCA}} + \delta_{BCL}}{2} \tag{4.5}$$

Since we are dealing with three metrics, and in some cases there is no unanimity in which is the optimal parameter, we established the average as criteria. As D_{KL} is not bounded, we normalized D_{KL} according to the maximal value of D_{KL} measured.

4.4 Concentration

An accumulated flux profile relates the relative flux from a given galaxy to the distance. This index explores properties of galaxies shape that are intrinsically related to it's accumulated flux profile. For example, steeper profiles are usually observed in bulge systems whereas smoothed profiles are observed in bulge+disk systems.

Figure 4.6 - Flux profile estimated in a synthetic galaxy
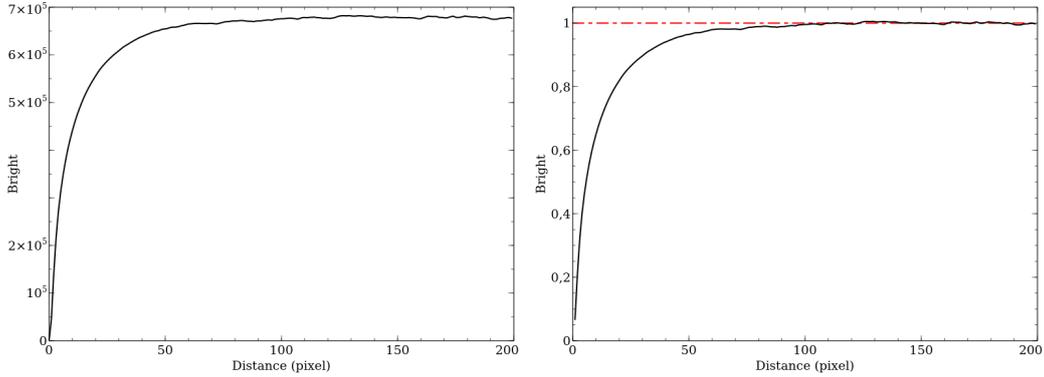
Flux profile from a simulated galaxy using Sérsic profile, $n = 6.0$, and half-light radius 10. Absolute accumulated flux profile in left side, and normalized accumulated flux profile on right side.

SOURCE: Own author.

Morgan (1958) proposed the first concentration index, which is given by the ratio of the distance that contains 80% brigthness[2] ($R_{80\%}$) of the observed galaxy, and the distance that contains 20% brightness ($R_{20\%}$) of the observed galaxy, as shown in the equation 4.6. Notice that fixing the brightness value, the ratio between these distances indicate the galaxy profile slope.

$$C_1 = log_{10}\left(\frac{R_{80\%}}{R_{20\%}}\right) \tag{4.6}$$

Further improvement in this index associate other ratios of brightness proportion radius since the distance measurement is affected by the image sky, and seeing effects in the center of the galaxies (FERRARI et al., 2015). In order to avoid the smoothing effect from galactic center, Kent (1985) proposed another concentration index (C_2), changing to the ratio between $R_{90\%}$ and $R_{50\%}$.

In this work we intend to define which ratio of brightness proportion radius produces a better characterization in the observed galaxies. To analyze the best ratio, we show in section a new methodology for setting parameters. As result, the optimal concentration index (C_3) is given by the ratio between $R_{65\%}$ and $R_{35\%}$, the

---

[2]A galaxy image brightness is the integrated flux in a given region

concentration index using this ratio were compared using the classifier.

Table 4.1 - Concentration radius ratio test

R_1	R_2	$D_{KL}(L,E)$	$D_H(L,E)$	GHS(L,E)	Average
0.55	0.15	0.626	0.681	0.813	0.692
0.55	0.25	0.873	0.744	0.874	0.810
0.55	0.35	0.947	0.783	0.877	0.847
0.55	0.45	0.66	0.823	0.892	0.776
0.65	0.15	0.644	0.712	0.872	0.728
0.65	0.25	0.653	0.769	0.869	0.748
0.65	**0.35**	**1.075**	**0.779**	**0.902**	**0.894**
0.65	0.45	0.727	0.8135	0.876	0.789
0.75	0.15	0.49	0.775	0.88	0.704
0.75	0.25	0.28	0.845	0.895	0.667
0.75	0.35	0.841	0.849	0.902	0.844
0.75	0.45	0.975	0.854	0.9	0.887

4.5 Estimating the accumulated flux profile

In literature empirical techniques are usually applied for the accumulated flux profile estimation (e.g. Ferrari et al. (2015), Conselice (2003)), another approach is to associate to a galaxy parametric model, as the Sérsic index (GRAHAM, 2001). Both techniques have issues, for instance empirical techniques have issues associated to seeing effects (FERRARI et al., 2015), whereas parametric models are inaccurate for late-type galaxies (GRAHAM, 2001).

Here we present a new empirical analysis process for the profile estimation. Our approach estimates the accumulated flux in a set of distances, and interpolates the profile to provide a distance for a given desired fraction of accumulated flux. The process to estimate a profile is composed by the steps: estimating the sky/background, sampling, estimating the galaxy total flux, interpolating the desired luminosity distance.

In order to estimate the background value we adopted the σ-clip technique from SExtractor (BERTIN; ARNOUTS, 1996). The main advantage of this solution is the invariance to the local noise, in this method the background of each pixel is given by the mean of local selected pixels. This selection is an iterative process where pixels inside a kernel that are not inside an interval around the local median are discard.

Figure 4.7 - Sampling process, and flux derivative profile
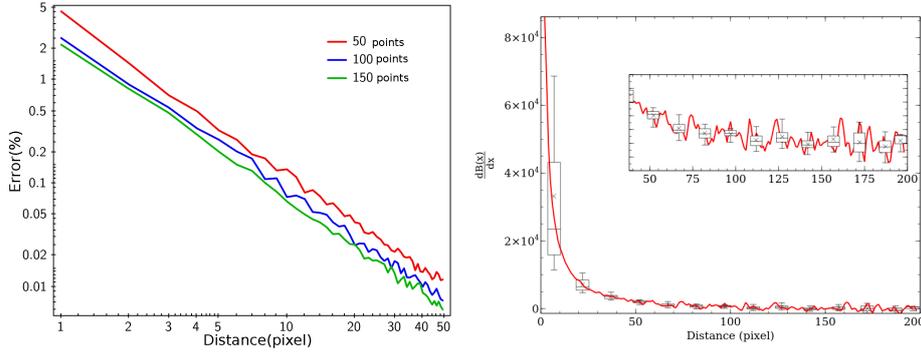
Error estimated in the sampling process versus distance, considering three densities. In right side, the derivative from the flux profile converging to 0

SOURCE: Own author.

In the sampling step, the total brightness profile is estimated (for example the profile A from image 4.6). Given a distance d, for each pixel n pixels are sampled in order to estimate the flux contribution for this distance. The total brightness of the distance d is given by the weighted sum of each pixel. The sampling density drives the calculation error, if a small density is used, the precision of this method decay, whereas if a high density is used, the computational cost increases. This leads us a question: Which sampling density achieves an acceptable error, and has a suitable computational cost?

First of all an error equation is presented: consider an image composed by the constant 1, if we estimate the brigthness inside an aperture with radius d, then the real total brightness in this aperture ($B(d)$) is the circle area (πd^2). Using the circle true area, the error equation 4.7 is proposed. The error was measured with three different pixels density, and shown in figure 4.7. Notice that with the density 100 samples/pixel, the error stabilizes since this is the minimal distance that the error converges.

$$E(d) = \frac{|\pi d^2 - B(d)|}{\pi d^2} \tag{4.7}$$

For the total flux estimation, in literature usually is set an empirical aperture (see for example Ferrari et al. (2015), Conselice (2003)), and then is calculated the sum-

48

mation of the flux inside this aperture. However, this technique may lead some issues, as the detection of problems in background estimation process. Analyzing the brightness profile derivative with respect the distance, as shown in Figure 4.7, the derivative approaches to zero as the distance from the galaxy center increases.

To accurately determine total flux in a given a profile, a technique similar to the sky σ-clipping is presented. Firstly, we measure the median flux ($\tilde{f}$) and its standard deviation(σ_f) from a set starting from a distance where the flux derivate with respect the distance is less than 10% is measured. Given $\tilde{f}$, and σ_f, the total flux from the source is measured as a mean of points that satisfies the inequation $|f - \tilde{f}| < k\sigma_f$. If there is no derivative less than 10% or less than five points satisfies the inequality, the flux profile is said non-convergent, the error flag is set 2, and the total flux is calculated as the sum of every pixel in image.

4.6   Asymmetry

The asymmetry is a typical basis on morphometric systems (see for example(ABRAHAM et al., 1996; FERRARI et al., 2015)). Since this metric is commonly applied to characterize high-redshift galaxies (CONSELICE, 2003), a diversity of equations and enhancing processes were proposed to improve this method.

A popular version of asymmetry (ABRAHAM et al., 1996) is given by the equation 4.8. In this equation, $I_{i,j}^d$ is the (i, j) pixel intensity after subtracting the background, and d is the angle of rotation in radians. Notice that each term of sum is weighted by $|I_{i,j}^0|$. This weight enhances the spiral disk region since the galaxy central region usually has the higher intensity value than the disk. However, notice the sky is also enhanced in this process since it has a low flux intensity, as result one of the main tasks is how to segmented the images. In this version, we may expect higher values of asymmetry for late-type galaxies.

$$A_1 = \sum_{i,j} \frac{|I_{i,j}^0 - I_{i,j}^\pi|}{|I_{i,j}^0|} \tag{4.8}$$

In the state of the art, the asymmetry index is measured using correlation coefficients, as described by the equations 4.9, and 4.10. The functions $r()$ and $s()$, are respectively the Pearson rank and the Spearman rank. The advantage of the correlation coefficients are the robustness to seeing effects (FERRARI et al., 2015) and the robustness to the sky interference on the measurement.

$$A_2 = 1 - r(I^0, I^\pi) \tag{4.9}$$

$$A_3 = 1 - s(I^0, I^\pi) \tag{4.10}$$

In our approach, we also introduce an enhancement process by weighting each pixel intensity according to the filtered gradient geometric mean (see section 4.7.1).

4.7 Clumpiness

The clumpiness describes the local flux dispersion from an image, which is measured as the correlation between the original image and its smoothed counterpart. Traditionally this process is done convolving a smoothing kernel with a fixed-size kernel (ABRAHAM et al., 1996),or according to the aimed object size (CONSELICE, 2003; FERRARI et al., 2015). However, in literature there is a lack of explanation/tests about the defined kernel size, and the kernel parameters. A new approach for image smoothing is presented, using the Butterworth filtering (subsection 4.15). The advantage of this method is a continuous adaptive control to the image smooth degree. In order to improve this parameter characterization, we also introduce an enhancing step using the Sobel filter (see section 4.7.1).

Originally, the clumpiness is measured as the weighted difference between the image (composed by elements $I_{i,j}$) and its smoothed version(composed by elements $I_{i,j}^s$), according to the equation 4.11.

In further works (CONSELICE, 2003; HAMBLETON et al., 2011) the sum operation was replaced by the maximum operation. This modification improves the acuity (see histograms from Hambleton et al. (2011)). However, it increased the method sensitivity.

$$S_1 = \sum_{i,j} \frac{|I_{i,j} - I_{i,j}^s|}{|I_{i,j}|} \tag{4.11}$$

In recent works, this parameter has been improved by adopting correlation coefficient(FERRARI et al., 2015). The advantages to this approach are the characterization of the flux intensity levels with respect to the linearity of the flux distribution, and the robustness to local noise. The smoothness parameters are measured according to the equations 4.12, and 4.13, where I is the flux intensity in the original image, and I^s is the flux intensity in the smoothed image.

$$S_2 = 1 - r(I, I^s) \tag{4.12}$$

$$S_3 = 1 - s(I, I^s) \tag{4.13}$$

Aiming to determine the best smoothing degree, three objective functions based on empirical histograms distance/divergence were applied (details to these measurements are shown in section 4.3). In Table 4.2 is shown the objective function for a set of smoothing parameters ($deg(S_3)$). The ratio of 0.6 is the best parametrization since it produce the highest separation ratio.

Table 4.2 - Smoothing degradation degree test

$deg(S_3)$	$D_{KL}(L,E)$	$D_H(L,E)$	GHS(L,E)	Average
0.3	0.656	**0.838**	0.655	0.573
0.4	1.429	0.796	0.594	0.627
0.5	1.923	0.805	0.527	0.665
0.6	**2.905**	**0.778**	**0.541**	**0.773**
0.7	2.361	0.737	0.679	0.743

4.7.1 Sobel filter

The Sobel filter is a 2D-gradient filter that emphasizes high-spatial frequencies (**??**). The kernel for horizontal gradient (dh), vertical gradient(dv), and both diagonals gradient are given by the kernels from figure 4.7.1.

Figure 4.8 - 3x3 Sobel filters

-1	0	1
-2	0	2
-1	0	1

1	2	1
0	0	0
-1	-2	-1

0	1	2
-1	0	1
-2	-1	0

-2	-1	0
-1	0	1
0	1	2

In order to enhance the galaxy image, a set of weights are built from the filtered gradient geometric mean. The weights ($w_{i,j}$) are measured according to the equation 4.14, where i, j are matrix coordinates dh, dv, dp, and ds are respectively the gradients of: the horizontal, vertical, principal diagonal, and secondary diagonal.

51

$$w_{i,j} = \frac{\sqrt{dh_{i,j}^2 + dv_{i,j}^2 + dp_{i,j}^2 + ds_{i,j}^2}}{max(\sqrt{dh^2 + dv^2 + dp^2 + ds^2})} \qquad (4.14)$$

Each weight($w_{i,j}$) is directly multiplied to the correspondent matrix element($I_{i,j}$), in order to obtain the enhanced matrix.

4.7.2 Butterworth filter

The Butterworth filter (KASZYNSKI; PISKOROWSKI, 2006; **??**) associated weights to spatial frequencies domain, in order to filter high frequencies. This filter is given by the equation 4.15, where ω is the amplitude from a spatial frequency $d(\omega)$. Two control parameters determine the cutoff[3] frequency (d_0), and the filter order (n).

$$G(\omega) = \frac{1}{1 + \left(\frac{d(\omega)}{d_0}\right)^{2n}} \qquad (4.15)$$

In Figure 4.9 is presented the frequency profile with four orders, and cutoff distance. Notice that increasing the order the profile approaches to a threshold function, where it is observed the formation of artifacts in the image. In this pipeline, we reparametrized d_0 as the $c\,max(W, H)$, were W, H are the matrix sizes. With the reparametrization c turns into a control parameter that ranges between 0 and 1. We fixed the filter order to 2.

---

[3]A frequency with weight 0.5

Figure 4.9 - Butterworth filter for a set of frequencies
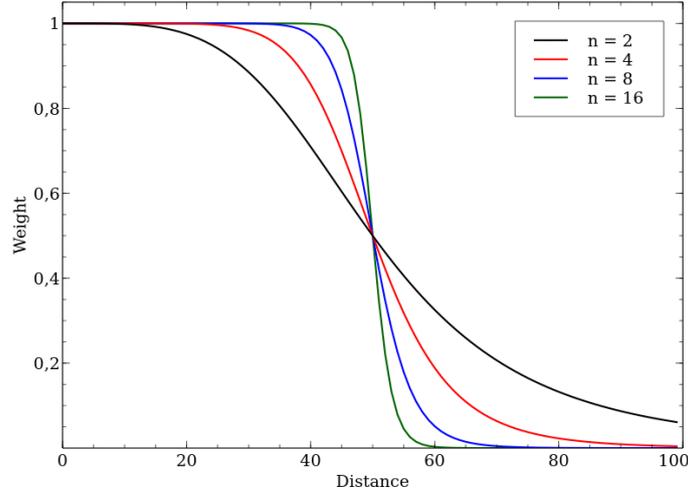
Frequency weights given by the Butterworth filter, with $d_0 = 50$, and varying order.
SOURCE: Own author.

4.8 Entropy

The entropy is a measurement of a random variable uncertainty. Here, we adopted the Shannon entropy (shown equation 4.16), as this approach were already applied to morphology providing reliable results(FERRARI et al., 2015).

Assuming the galaxy flux as the random variable, this measurement shows the heterogeneity degree in pixel distribution.

$$H = -\frac{\sum_{k}^{K} p(I_k) log(p(I_k))}{log(K)} \tag{4.16}$$

Notice that in equation 4.16, the flux intensity (I_k) is discretized in a set of K bins. As described in other morphological parameters, we apply an objective function to determine the best number of sets (K). Although different from other methods, there was no concordance between on the optimal value for K. To solve this problem, we selected the average of best K for each objective function, which is $K = 130$ bins.

Table 4.3 - Entropy bins

Bins	$D_{KL}(L,E)$	$D_H(L,E)$	GHS(L,E)	Average
100	0.683	0.753	0.873	0.764
130	**1.027**	**0.715**	**0.891**	**0.869**
160	0.549	0.743	0.856	0.711
190	0.729	0.76	0.866	0.779
220	0.818	0.743	0.879	0.806

4.9 Gradient Pattern Analysis

Concerning on GPA performance for characterizing galaxy images, a small change on G_1 and G_2 is proposed. The equations for these metrics are the described in sections 2.2.1 and 2.2.2. However, the operator has been adapted to work with segmentation masks. In symmetry detection step, if a pixel in a position (x, y) is detected as sky by SExtractor, then the gradient at position (x, y) is ignored.

In last step, for G_2, each a pixel that belongs to the sky decreases the total number of vectors (N), and the vectors at those positions are not considered on confluence summation. In G_1 all vector, which belongs to the sky, are enforced to symmetrical since G_1 depends only on the ratio between the number of Delaunay connections and the number of asymmetrical vectors.

An example of analysis is shown in Figure 4.10. The presented example shows two standard galaxies: an elliptical, and a spiral. Notice that, vectors with highest modulus are inside the galaxy disk, which are the vectors at galaxy boundary. Also, the central vectors -that belongs to the galaxy bulge- are symmetric.

Figure 4.10 - G_1 and G_2 applied to galaxy morphology
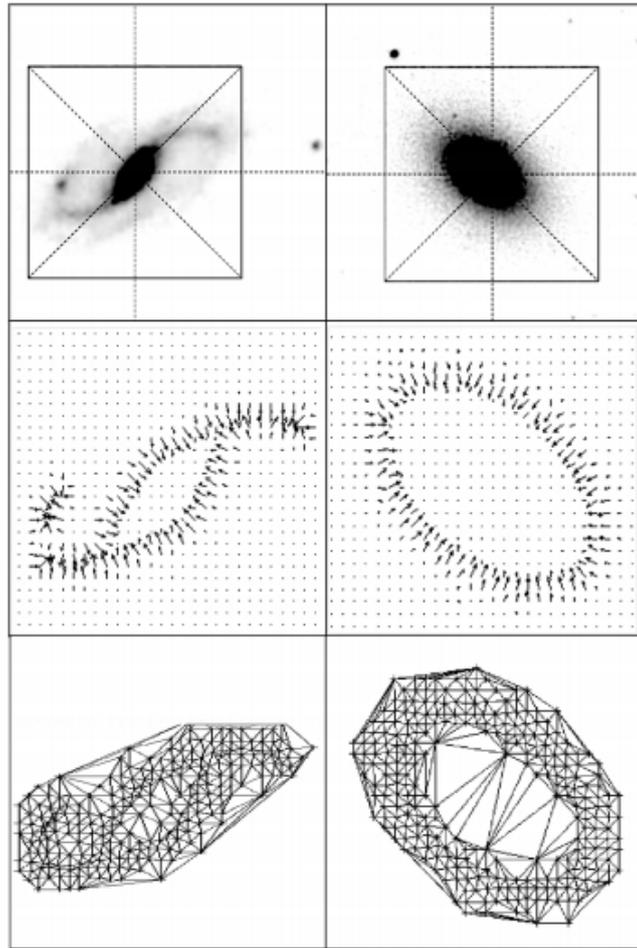
On left side is shown an example of spiral galaxy, on right side is shown an example of elliptical galaxy. The first row is the original image, the second row is the asymmetrical gradient field, and the third row is the Delaunay triangulation.

SOURCE: Own author.

Notice, the elliptical galaxy in Figure 4.10 has a large bulge, and the vectors at boundary are regularly distributed with all possible directions. In contrast, spiral galaxies have more asymmetrical vectors in proportion, and the perturbations observed at the disk increases the number of Delaunay triangulations. Thus, we may expect a higher value of G_1 and G_2 for spiral galaxies.

4.10   Usability

CyMorph is a pipeline in command-line interface that uses a configuration script as input. This pipeline was developed in Cython, a library that translates part of code in Python into C. It also allows the exchange between flexibility (from Python) and performance (from C). We can exchange the flexibility by: specifying the variable types, coding using imperative paradigm, and compiling the code (as shown in Code 4.1). Additionally, the components of this code are accessible by Python scripts.

Code 4.1 - Command for locally compile CyMorph

```
python compile.py build_ext ——inplace
```

The input script is ordered in three categories: File, Output, and Indexes. In the first configuration category the file path, indexes, and the clipping process are specify. In output, is specified if the pipeline should save the figures in each step (for example the flux profile, the smoothed galaxy image, or the correlation points), and if the pipeline should print in terminal the current step and eventual problems (for instance the concentration convergence). The Indexes category is specified each morphological parametrization (for example the smooth degree). An example of configuration file is shown Code 4.2. There are three execution modes, the user can: run a simple galaxy image (Code 4.3), run a set of images with MPI (Code 4.4), or optimize the morphological parameters.

Code 4.2 - CyMorph configuration file example

```
[File_Configuration]
Path: benchmark/
Filename: 587722952230240617.fit
Indexes: Ga,S,A3,C, H
Clip: True


[Output_Configuration]
Verbose: True
SaveFigure: True


[Indexes_Configuration]
Entropy_Bins: 180
Ga_Tolerance: 0.02
Ga_Angular_Tolerance: 0.03
Ga_Position_Tolerance: 0.01
Concentration_Density: 100
Concentration_Distances: 0.65, 0.35
butterworth_order: 2
smooth_degree: 0.5
```

Notice that in Code 4.4 the configuration file is not specified in the pipeline call. Instead, we use the file list as input. In this execution mode, the configuration is specified in "ParallelConfig.ini", a configuration file which does not specify the path, neither the filename.

Code 4.3 - Running CyMorph for a single galaxy images

```
python main.py config.ini
```

Code 4.4 - Running CyMorph for a a set of galaxy images

```
mpirun −np 3 PCyMorph.sh test500/spirals.csv
```

The optimization process is the script "optimizeIndexes.py", that generates configuration files for a distributed execution. This process requires more control than other execution modes, we should define precisely the parameter space to analyze since every execution has a high computational cost. Thus, the script must be rewrite for each parameter optimization.

4.11 Morphometric Histograms

The result for each parameter, with the optimal parametrization, is exposed in this section. For each morphological parameter, it is plotted a histogram for early-type galaxies (in red) and a histogram for late-type (in blue). The histograms shown in Figure 4.11 shows the best distribution of each morphometric type. The goal is to separate the galaxy distribution as much as possible.

Figure 4.11 - Histograms of galaxies
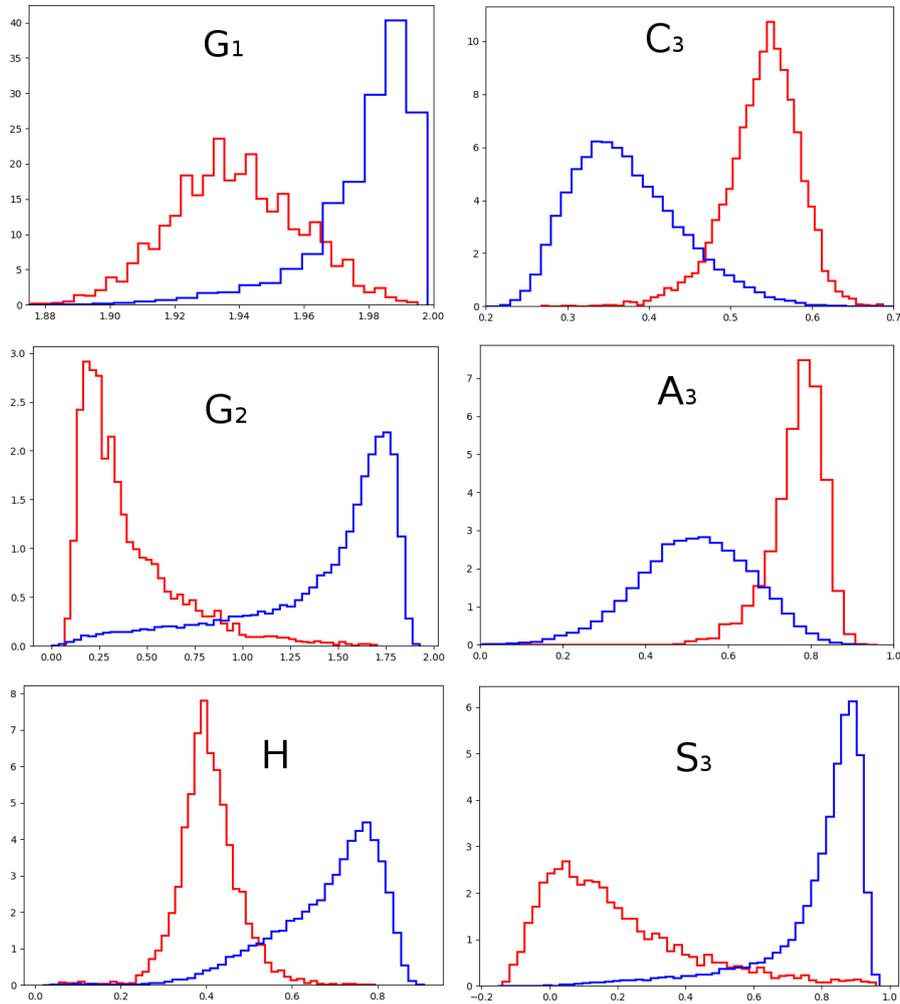
Each graph displays the distribution of a galaxy morphometric. The histogram have normalized area, since the dataset is imbalanced. Elliptical galaxy distribution is in red. Spiral galaxy distribution is in blue.

SOURCE: Own author.

It is also presented the final distance between the distributions is shown on Table 4.4. Notice, the final histogram distances are different to the optimal parametrization distances since the final distances are measured in the whole dataset.

In order to build a reliable classifier we must consider a set composed by the most significant metrics. According to δ_{GHS}, the best morphometrics are respectively: S_3, G_2, and H. Notice, the difference between separation ratio of these metrics are not significant. For instance, the difference between $\delta_{GHS}(C_3)$ and $\delta_{GHS}(G_2)$ is 0.01.

Also, it is important to note, the more diverse is the set of metric, the better is the classifier. As demonstrated, GPA retains the signature of the asymmetric gradient field, which is a unique feature. Since G_2 is one of the bests morphometrics with respect to the separation ratio, and it is a unique feature, we can conclude that we achieved our goal. We adapted G_2 to the galaxy morphometry, and we obtain a suitable galaxy separation ratio. A further study of classifiers with these metrics is required. However, due to time limitations and the goals of this work, here, we have not explored this topic.

Table 4.4 - Final histogram distances of the best morphometrics

	$\mathbf{C}_3$	$\mathbf{A}_3$	$\mathbf{S}_3$	$\mathbf{H}$	$\mathbf{G}_1$	$\mathbf{G}_2$
D_{KL}	2.957	2.274	2.828	2.632	1.500	2.056
D_H	0.747	0.667	0.744	0.737	0.585	0.723
δ_{GHS}	0.81	0.834	0.900	0.873	0.802	0.890

With this last statement, we consolidate the static study case. In the following chapter, we conclude this dissertation, reviewing the application and important topics of this work, and suggesting future works.

5 CONCLUDING REMARKS

The aim of this work has been the Gradient Pattern Analysis (GPA) adjustment to large datasets especially datasets with large matrices. These adjustments are required, as it have been observed that in the state of the art, all operators of GPA have a high computational cost, and are not effective in large matrices. The secondary objective of this work has been the dynamical system analysis and the galaxy morphological analysis, comparing with the commonly applied techniques in these applications.

5.1 Conclusion overview

At chapter 2 the GPA formalism and the classical technique (G_1) were presented, it were also proposed some improvements. The proposed improvements include the establishment of concentric symmetry, the definition of a second gradient order operator (G_2), and the code parallelization. We review the test basis for GPA, composed by: canonical matrices (Laminar, Bessel, and Gaussian) and Random matrices. It is introduced a secondary type of basic test: the Perlin Matrices, were the objective is to distinguish spatial noise to random noise. We observed that G_2 has a similar result to G_1 with respect to the Random matrices. However, unlike G_2, G_1 were not able to distinguish Random matrices to the Perlin matrices.

In chapter 3 it is presented the Coupled Map Lattice(CML), a spatially extended system that is a template for complex systems. It is presented two interesting system transitions: symmetry breaking and synchronization. It is presented a test with a simple case in each system transition, and a global test varying the system parameters. Additionally, in symmetry breaking it is presented a test with different maps, in this test it is observed a unique time and velocity for system transition. With the improvements to GPA technique, we are able to observe these transitions.

Lastly, in chapter 4 it is presented the static application: the galaxy morphology. The aim of this chapter has been the evaluation of a reliable galaxy morphometric system, based on classical techniques and the proposed operator. It is proposed some improvements that include: the preprocessing, the replacement of some basic image operations[1], and a criteria for rating each morphometric. After some adaptations, it is also introduced G_1 and G_2 in the morphometric set. We observed that, G_2 has a similar classification capability to entropy (H) and clumpiness (S_3), were according to the proposed criterion (δ_{GHS}), G_2 is the second best morphometric.

---

[1]Such as the smoothing process, and the structure enhancement

5.2   Future works

The presented work demonstrated the GPA potential to characterize spatially extended systems. We achieved the primary and secondary goals, however, some important topics emerged during this work. Due to time limitations, these topics should be pursued in future works.

Firstly, with respect to the metric, some improvements to the computational performance still required. We intend to explore different approaches to the GPA bottleneck[2] in future works. Also, in theory field, some proprieties and important questions about GPA still unsolved, such as the maximal fragmentation.

Concerning the dynamical application, it would be interesting to see if GPA is a good descriptor of other dynamical systems, for example graph dynamical systems and 3D-systems. We would also like to explore a possible scaling free (L) universality class, $G(L) \propto L^{\mu}$, to characterize different underlying physical process in the pattern formation phenomena. This is an innovative and complementary approach in the field of spatiotemporal nonlinear dynamics that deserves future systematic research using GPA.

With respect to the galaxy morphology application, it is important to see if the presented results still valid for image-datasets with lower/higher resolutions. Another important topic, which we are researching, is the classification. We expect intriguing result in an unsupervised morphological analysis.

---

[2]The symmetry removal step

REFERENCES

ABRAHAM, R.; TANVIR, N.; SANTIAGO, B.; ELLIS, R.; GLAZEBROOK, K.; BERGH, S. V. D. Galaxy morphology to i= 25 mag in the hubble deep field. **Monthly Notices of the Royal Astronomical Society**, Blackwell Science Ltd Oxford, UK, v. 279, n. 3, p. L47–L52, 1996. 49, 50

ASSIREU, A. T.; ROSA, R. R.; VIJAYKUMAR, N. L.; LORENZZETTI, J. A.; REMPEL, E. L.; RAMOS, F. M.; SÁ, L. A.; BOLZAN, M. J. A.; ZANANDREA, A. Gradient pattern analysis of short nonstationary time series: an application to lagrangian data from satellite tracked drifters. **Physica D: Nonlinear Phenomena**, v. 168, p. 397–403, 2002. 1, 2, 10

BARCHI, P.; COSTA, F.; SAUTTER, R.; MOURA, T.; STALDER, D.; ROSA, R.; CARVALHO, R. Improving galaxy morphology with machine learning. **arXiv preprint arXiv:1705.06818**, 2017. 37

BARKER, W. H.; HOWE, R. **Continuous symmetry: from Euclid to Klein**. [S.l.]: American Mathematical Society, 2007. 6

BERTIN, E.; ARNOUTS, S. Sextractor: software for source extraction. **Astronomy and Astrophysics Supplement Series**, v. 117, n. 2, p. 393–404, 1996. 41, 47

BLANTON, M. R.; BERSHADY, M. A.; ABOLFATHI, B.; ALBARETI, F. D.; PRIETO, C. A.; ALMEIDA, A.; ALONSO-GARCÍA, J.; ANDERS, F.; ANDERSON, S. F.; ANDREWS, B. et al. Sloan digital sky survey iv: mapping the milky way, nearby galaxies, and the distant universe. **The Astronomical Journal**, IOP Publishing, v. 154, n. 1, p. 28, 2017. 40

CATANZARO, B. Opencl optimization case study: simple reductions. **White Paper**, 2010. 71

CONSELICE, C. J. The relationship between stellar light distributions of galaxies and their formation histories. **The Astrophysical Journal Supplement Series**, v. 147, n. 1, p. 1, 2003. 1, 47, 48, 49, 50

DATTA, B. N.; YATSENKO, V. A.; NAIR, S. P. Model updating and simulation of lyapunov exponents. In: PROCEEDINGS... KOS: IEEE. **Control Conference (ECC)**. [S.l.], 2007. p. 1094–1100. 23, 74

DIGGLE, P. J. **Statistical analysis of spatial and spatio-temporal point patterns**. [S.l.: s.n.], 2013. 1

DRESSLER, A. Galaxy morphology in rich clusters-implications for the formation and evolution of galaxies. **The Astrophysical Journal**, v. 236, p. 351–365, 1980. 1, 37

EBERT, D. S.; WORLEY, S.; MUSGRAVE, F. K.; PEACHEY, D.; PERLIN, K.; MUSGRAVE, K. F. **Texturing and Modeling**. 2. ed. Orlando, FL, USA: [s.n.], 1998. ISBN 0122287304. 17

FERRARI, F.; CARVALHO, R. R.; TREVISAN, M. Morfometryka – a new way of establishing morphological classification of galaxies. **The Astrophysical Journal**, v. 814, n. 1, p. 55, 2015. 46, 47, 48, 49, 50, 53

GRAHAM, A. W. An investigation into the prominence of spiral galaxy bulges. **The Astronomical Journal**, v. 121, n. 2, p. 820, 2001. 47

GROENEBOOM, N.; DAHLE, H. Introducing gamer: a fast and accurate method for ray-tracing galaxies using procedural noise. **The Astrophysical Journal**, v. 783, n. 2, p. 138, 2014. 17

HAMBLETON, K.; GIBSON, B. K.; BROOK, C.; STINSON, G.; CONSELICE, C.; BAILIN, J.; COUCHMAN, H.; WADSLEY, J. Advanced morphological galaxy classification: a comparison of observed and simulated galaxies. **Monthly Notices of the Royal Astronomical Society**, v. 418, n. 2, p. 801–810, 2011. 50

HOLKNER, A. Pyglet: Cross-platform windowing and multimedia library for python. **Google Code**, 2008. 17

JONES, E. et al. **SciPy: Open source scientific tools for Python**. 2001. Disponível em: <http://www.scipy.org/>. 10

KANEKO, K. Overview of coupled map lattices. **Chaos: An Interdisciplinary Journal of Nonlinear Science**, v. 2, n. 3, p. 279–282, 1992. 23

KASZYNSKI, R.; PISKOROWSKI, J. New concept of delay equalized low-pass butterworth filters. In: PROCEEDINGS...CANADA: IEEE. **IEEE INTERNATIONAL SYMPOSIUM ON INDUSTRIAL ELECTRONICS**. [S.l.], 2006. v. 1, p. 171–175. 52

KENT, S. M. Ccd surface photometry of field galaxies. ii-bulge/disk decompositions. **The Astrophysical Journal Supplement Series**, v. 59, p. 115–159, 1985. 46

KLÖCKNER, A.; PINTO, N.; LEE, Y.; CATANZARO, B.; IVANOV, P.; FASIH, A. PyCUDA and PyOpenCL: A Scripting-Based Approach to GPU Run-Time Code Generation. **Parallel Computing**, v. 38, n. 3, p. 157–174, 2012. ISSN 0167-8191. 69

KOBAYASHI, T.; MORI, N.; HATA, H.; HORITA, T.; YOSHIDA, T.; MORI, H. Critical scaling laws of dynamic structure functions for type i intermittent chaos. **Progress of Theoretical Physics**, v. 82, n. 1, p. 1–6, 1989. Disponível em: <+http://dx.doi.org/10.1143/PTP.82.1>. 26

KULLBACK, S. **Information theory and statistics**. [S.l.]: Courier Corporation, 1997. 43

LINTOTT, C. J. et al. Galaxy zoo: morphologies derived from visual inspection of galaxies from the sloan digital sky survey. **Monthly Notices of the Royal Astronomical Society**, v. 389, n. 3, p. 1179–1189, 2008. 39

LIU, Y.; HEL-OR, H.; KAPLAN, C. S.; GOOL, L. V. et al. Computational symmetry in computer vision and computer graphics. **Foundations and Trends® in Computer Graphics and Vision**, v. 5, n. 1–2, p. 1–195, 2010. 7

MO, H.; BOSCH, F. van den; WHITE, S. **Galaxy Formation and Evolution**. [S.l.]: Cambridge University Press, 2010. ISBN 9780521857932. 37, 38

MORGAN, W. A preliminary classification of the forms of galaxies according to their stellar population. **Publications of the Astronomical Society of the Pacific**, v. 70, n. 415, p. 364–391, 1958. 46

PEDRINI, H.; SCHWARTZ, W. **Análise de imagens digitais: princípios, algoritmos e aplicações**. São Paulo: Thomson Pioneira, 2008. ISBN 9788522105953. Disponível em: <https://books.google.com.br/books?id=13KAPgAACAAJ>. 1

PETROSIAN, V. Surface brightness and evolution of galaxies. **The Astrophysical Journal**, v. 209, p. L1–L5, 1976. 40

RAMOS, F. M.; ROSA, R. R.; NETO, C. R.; ZANANDREA, A. Generalized complex entropic form for gradient pattern analysis of spatio-temporal dynamics.

Physica A: Statistical Mechanics and its Applications, Elsevier, v. 283, n. 1, p. 171–174, 2000. 1, 2, 9, 10, 23

ROSA, R.; KARLICKÝ, M.; VERONESE, T.; VIJAYKUMAR, N.; SAWANT, H.; BORGAZZI, A.; DANTAS, M.; BARBOSA, E.; SYCH, R.; MENDES, O. Gradient pattern analysis of short solar radio bursts. **Advances in Space Research**, v. 42, n. 5, p. 844–851, 2008. 1, 2, 10

ROSA, R.; PONTES, J.; CHRISTOV, C.; RAMOS, F. M.; NETO, C. R.; REMPEL, E. L.; WALGRAEF, D. Gradient pattern analysis of swift–hohenberg dynamics: phase disorder characterization. **Physica A: Statistical Mechanics and its Applications**, v. 283, n. 1-2, p. 156–159, 2000. 13, 15

ROSA, R. R.; CARVALHO, R. R. de; SAUTTER, R. A.; BARCHI, P. H.; STALDER, D. H.; MOURA, T. C.; REMBOLD, S. B.; MORELL, D. R.; FERREIRA, N. Gradient pattern analysis applied to galaxy morphology. **Monthly Notices of the Royal Astronomical Society: Letters**, 2018. Disponível em: <http://doi.org/10.1093/mnrasl/sly054>. 37

ROSA, R. R.; SHARMA, A. S.; VALDIVIA, J. A. Characterization of asymmetric fragmentation patterns in spatially extended systems. **International Journal of Modern Physics C**, v. 10, n. 01, p. 147–163, 1999. Disponível em: <http://www.worldscientific.com/doi/abs/10.1142/S0129183199000103>. 1, 10, 11, 14, 15, 20, 21

RUAN, D. Applied artificial intelligence. In: **International FLINS Conference**. [s.n.], 2006. v. 7. ISBN 9789812774118. Disponível em: <https://books.google.com.br/books?id=UOIZP1Vp2MQC>. 74

SHAMIR, L.; WALLIN, J. Automatic detection and quantitative assessment of peculiar galaxy pairs in sloan digital sky survey. **Monthly Notices of the Royal Astronomical Society**, v. 443, n. 4, p. 3528–3537, 2014. 37

SHOBU, K.; OSE, T.; MORI, H. Shapes of the power spectrum of intermittent turbulence near its onset point. **Progress of theoretical physics**, v. 71, n. 3, p. 458–473, 1984. 25

SMALE, S.; HIRSCH, M. W.; DEVANEY, R. L. **Differential equations, dynamical systems, and an introduction to chaos**. [S.l.: s.n.], 2003. 25, 77

STRIEDER, C. **Calculo do coeficiente de assimetria gradiente em C/C++: Exemplos de aplicações em astrofísica e cosmologia**. 2010. 129 p.

(INPE-14794-TDI/1237). Dissertação (Mestrado em Computação Aplicada) — Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 2010. Disponível em: <http://urlib.net/8JMKD3MGP7W/3827KLH>. Acesso em: 22 mar. 2016. 2, 8, 14, 15, 37

SU, C.-L.; CHEN, P.-Y.; LAN, C.-C.; HUANG, L.-S.; WU, K.-H. Overview and comparison of opencl and cuda technology for gpgpu. In: PROCEEDINGS... IEEE. **IEEE Asia Pacific Conference on Circuits and Systems**. [S.l.], 2012. p. 448–451. 69

SUN, F.; LIU, S.; LI, Z.; LÜ, Z. A novel image encryption scheme based on spatial chaos map. **Chaos, Solitons & Fractals**, v. 38, n. 3, p. 631–640, 2008. 23

TEMPEL, E.; SAAR, E.; LIIVAMÄGI, L.; TAMM, A.; EINASTO, J.; EINASTO, M.; MÜLLER, V. Galaxy morphology, luminosity, and environment in the sdss dr7. **Astronomy & Astrophysics**, EDP Sciences, v. 529, p. A53, 2011. 37

VERONESE, T. B. **Grade numérica generalizada: um novo conceito para representação e visualização analítica de sistemas de séries temporais**. 118 p. Tese (Doutorado) — Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2011. Disponível em: <http://urlib.net/sid.inpe.br/mtc-m19/2011/11.15.22.02>. Acesso em: 15 abr. 2018. 15

YOON, I.; WEINBERG, M. D.; KATZ, N. New insights into galaxy structure from galphat–i. motivation, methodology and benchmarks for sérsic models. **Monthly Notices of the Royal Astronomical Society**, Blackwell Publishing Ltd Oxford, UK, v. 414, n. 2, p. 1625–1655, 2011. 37

ZHANG, Y.-Q.; HE, Y.; WANG, X.-Y. Spatiotemporal chaos in mixed linear nonlinear two dimensional coupled logistic map lattice. **Physica A: Statistical Mechanics and its Applications**, v. 490, n. Suppl. C, p. 148 – 160, 2018. ISSN 0378-4371. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0378437117307070>. 23, 33

APPENDIX A - DOWNLOADING SDSS IMAGES

To download SDSS field images, it is necessary a catalog specifying each object proprieties. For this work, it was downloaded images from Galaxy Zoo 1 catalog, which can be downloaded directly by the project website. The table A.1 is an example of catalog.

The table must have the proprieties: ra,dec, run, rerun, camcol, and the filename. An Uniform Resource Locator (URL) is build to download the compressed field image. The following code recieves a catalog in ".csv" format, build the URL, download each compressed image, and descompress it.

Code A.1 - Download SDSS field images

```python
import numpy
import csv
import sys
import os

gal = list(csv.reader(open(sys.argv[1], "rb"), delimiter=','))
ndata = len(gal)
header = numpy.array(gal[0])
gal[1:] = sorted(gal[1:], key=lambda l:l[numpy.where(header == "image")[0][0]])
path = "Field/"

for line in range(1,ndata):
        imgIndex = numpy.where(header == "image")[0][0]
        fieldName = path+gal[line][imgIndex]
        fileName = gal[line][imgIndex].replace(".gz", "")
        if not(os.path.isfile(fieldName) or os.path.isfile(path+fileName)):
                print("Downloading", line)
                ra = gal[line][numpy.where(header == "ra")[0][0]]
                dec = gal[line][numpy.where(header == "dec")[0][0]]
                run = gal[line][numpy.where(header == "run")[0][0]]
                rerun = gal[line][numpy.where(header == "rerun")[0][0]]
                camcol = gal[line][numpy.where(header == "camcol")[0][0]]
                #field = gal[line][numpy.where(header == "field")[0][0]]
                dr7id = gal[line][numpy.where(header == "dr7objid")[0][0]]

                cmd = "wget --inet4-only -r -nd --directory-prefix=Field http://das.sdss.org/raw/"
                cmd += str(run) + "/"
                cmd += str(rerun) + "/corr/"
                cmd += str(camcol) + "/"
                cmd += fileName + ".gz"
                print(cmd)
                pr = os.popen(cmd)
                print(pr.read())
                # unzip the image
                cmd = "gzip -d " + path + fileName + ".gz"
                pr = os.popen(cmd)
                pr.read()
        else:
                print("Found",fieldName,line)
print("Done")
```

Table A.1 - Example of catalog table

dr7objid	ra	dec	run	camcol	rerun	field	Zoo1	image
5877396478195139 43	169.0302	31.7175	4632	4	40	178	S	fpC-004632-r4-0178.fit.gz
5877348621429638 79	146.195	7.232444	3518	2	40	37	S	fpC-003518-r2-0037.fit.gz

APPENDIX B - DOWNLOADING SOURCE CODE

The source code of the presented systems are public, under GNU Licensce 3.0. A version of CyMorph code is available at: https://github.com/rsautter/CyMorph, and the CML system is also available at: https://github.com/rsautter/CML-CoupledMapLattice. And finally, the GPA source code is available at: https://github.com/rsautter/Concentric_GPA.

Alternatively, the code can be downloaded, if installed the the application 'git', via command line:

Code B.1 - CyMorph source code download
```
git  clone  https :// github .com/ r s a u t t e r /CyMorph
```

Code B.2 - CML source code download
```
git  clone  https :// github .com/ r s a u t t e r /CML–CoupledMapLattice
```

Code B.3 - CML source code download
```
git  clone  https :// github .com/ r s a u t t e r /Concentric_GPA
```

APPENDIX C - PARALLELIZATION

In order to take advantage from hybrid parallel components, a version of this metric was implemented using the library pyOpenCL (KLÖCKNER et al., 2012). OpenCL is not the fastest GPGPU library (see the for example the tests from Su et al. (2012)), however an important propriety of this library is portability, considering GPUs and other coprocessor units.

In every step of the technique an improvement to the performance were observed, in Table C.1 is shown a comparison[1] between the time in every step and the input matrix size.

Table C.1 - Execution time in seconds of each operator step

Matrix Size	Step 1	Step 2	Step 3
16x16	0.00005	0.00105	0.00003
32x32	0.00004	0.00303	0.00002
64x64	0.00004	0.01670	0.00002
128x128	0.00011	0.21659	0.00006
256x256	0.00039	2.76907	0.00016

Notice that the second step is the step which spends more computational time, which is explained by the computational complexity described in the following subsection.

C.1 Implementation and complexity analysis

In order to measure G_2, a kernel for each step were developed. Every kernel is executed by WH threads, which is associated to a local and global index. For the gradient approximation (Figure 2.6.a), the central finite difference method were applied. In this method, each matrix component is compared to its neighborhood value. This operation has no recursion/loop in each thread, therefore the computational complexity for each thread is $O(1)$.

Vectors with same distance must be compared at the second step (Figure 2.6.b). In order to select the data according to the distance,a smaller square is defined as search-space. The bounding points are given by $(cx - r, cy + r)$ and $(cx - r, cy - r)$, where cx and cy are the center coordinates and r is the distance of the evaluated

---

[1]Tested on a desktop with the configuration:Ubuntu 16.04, intel core i7-4500U , 16GB of RAM, AMD Radeon 86000M

point. This process is summarized on code C.1, where the function 'isSymmetric' determines whether two given vectors have same absolute value and opposite phase. As result, the computational cost for each thread in the second step is $O(WH)$, where W and H are respectively the matrix width and height.

Code C.1 - Kernel for second step of GPA

```
__kernel void asymmetry_r(
    __global double *phases, __global double *mods,
    __global int *tableau, const double mtol, const double ftol
){
    int2 p1 = (int2)(get_global_id(0), get_global_id(1));
    int2 dim = (int2)(get_global_size(0), get_global_size(1));
    double2 center;
    double ldist;
    center=(double2)(convert_double(dim.x),convert_double(dim.y));
    center = center/2.0;
    double myDist = euclid_distance(center,convert_double2(p1));
    int2 p3 = (int2)(0);
    int x, y,xini,xfin,yini,yfin;

    tableau[getIndex(p1,dim)] = 0;

    if( mods[getIndex(p1,dim)] < mtol/2){
        tableau[getIndex(p1,dim)] = 1;
            return;
    }
    if(isInsideMatrix(p1,dim)!=1){
        tableau[getIndex(p1,dim)] = 1;
        return;
    }
    xini = max(0,convert_int(center.x-myDist)-2);
    xfin = min(dim.x,convert_int(center.x+myDist)+2);
    yini = max(0,convert_int(center.y-myDist)-2);
    yfin = min(dim.y,convert_int(center.y+myDist)+2);

    // lazy search
    for(x= xini; x < xfin; x++){
        for(y=yini; y < yfin; y++){
            p3 = (int2)(x,y);
            ldist=fabs(euclid_distance(convert_double2(p3),center)-myDist);
            if(ldist <1.0){
                if(isSymmetric(p3,mods,phases,mtol,ftol)==1){
                    tableau [getIndex(p1,dim)] = 1;
                    return;
                }
            }
```

```
                        }
                }
        }
}
```

In the case of G_2, the final step consists on a summation of vectors and magnitude. An efficient solution for this step is the reduce method (Figure C.1).

Figure C.1 - Reduction tree
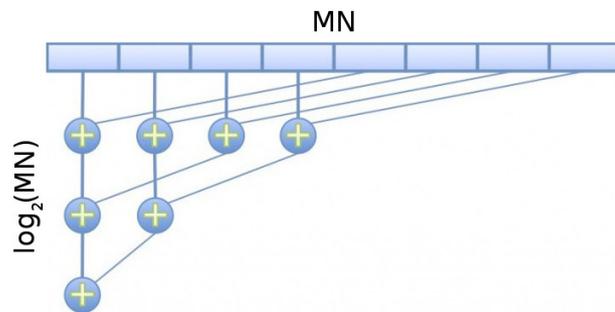
Parallel reduction tree applied to sum asymmetrical vectors.
SOURCE: Adapted from Catanzaro (2010)

The reduce method consists on recursively split the data and apply the desired operation. In this case the operation consists on accumulate the number of asymmetrical vectors, sum the partial vector coordinates, and sum the vector magnitude. To avoid the concurrency problems, a barrier is placed at the end of each loop iteration, as shown in code C.2. This method can be represented as a tree (Figure C.1), which the number of splitting sets (S) determine the tree order. As result this method has $O(log_S(WH))$ parallel time in each thread[2].

Code C.2 - Kernel for measuring G_2

```
___kernel void getG2(
    ___global double *gx, ___global double * gy,
    ___global double *mods,___global int *tableau,
    ___global double *partialX, ___global double *partialY,
    ___global double *partialMS, ___global int* countAsym,
    ___local double *tpartialX, ___local double *tpartialY,
```

_____

[2]$O(log_2(WH))$ in this case

71

```
    ___local double *tpartialMS, ___local int* tcountAsym
){
        int gid = get_global_id(0);
        int lid = get_local_id(0);
        int gsize = get_global_size(0);
        int lsize = get_local_size(0);

        //every element verify if it is a symmetric vector
        if(tableau[gid] > 0){
                tpartialX[lid] = 0.0;
                tpartialY[lid] = 0.0;
                tpartialMS[lid] = 0.0;
                tcountAsym[lid] = 0;
        }else{
                tpartialX[lid] = gx[gid];
                tpartialY[lid] = gy[gid];
                tpartialMS[lid] = mods[gid];
                tcountAsym[lid] = 1;
        }

        int oldStride = lsize;
        // measuring diversity, using partial sum method
        for (int stride = lsize/2; stride >0; stride = stride/2){
                barrier(CLK_LOCAL_MEM_FENCE); //wait everyone update
                if (lid < stride){
                        tpartialX[lid] += tpartialX[lid+stride];
                        tpartialY[lid] += tpartialY[lid+stride];
                        tpartialMS[lid] += tpartialMS[lid+stride];
                        tcountAsym[lid] += tcountAsym[lid+stride];
                }
                if(oldStride%2 != 0 && lid == stride -1){
                        tpartialX[lid] += tpartialX[lid+stride+1];
                        tpartialY[lid] += tpartialY[lid+stride+1];
                        tpartialMS[lid] += tpartialMS[lid+stride+1];
                        tcountAsym[lid] += tcountAsym[lid+stride+1];
                }
                oldStride = stride;
        }
        barrier(CLK_LOCAL_MEM_FENCE); //wait everyone update
        //merge each local sum in an array
        if(lid == 0){
                partialX[get_group_id(0)] = tpartialX[lid];
                partialY[get_group_id(0)] = tpartialY[lid];
                partialMS[get_group_id(0)] = tpartialMS[lid];
                countAsym[get_group_id(0)] = tcountAsym[lid];
```

```
        }
}
```

APPENDIX D - Largest Lyapunov Exponent

Here, we showcase an analysis with the Largest Lyapunov Exponent(LLE). From the geometrical perspective, the LLE measures the maximal separation rate between nearby points. In particular to CML, this analysis considers the variation on nearby points, which implies on gradient dynamics. A diversity of proof for chaotic behaviour in 1D-CML are presented in the literature (RUAN, 2006; DATTA et al., 2007). Part of this proof is given by the Global Largest Lyapunov Exponent measure. We adapt the analytical solution, into a local (empirical) solution for measuring the LLE in 2D-CML. Notice that this extension is not sufficient to characterize chaos, but is a good indicator of sensitivity to initial conditions. In order to measure LLE, consider the study case B, shown in table 3.1.

The LLE is given by $\lambda = \sum \frac{1}{N} ln|\Lambda(J_{2D})|$, where $\Lambda(J_{2D})$ is the largest eigenvalue of the Jacobian matrix (equation D.1), and N is the total of iterations. The Jacobian can be deduced by deriving CML general (shown in equation 3.2) with respect to each state $(a_{i,j}^n)$. To ensure that LLE is not biased by the initial condition, this value must be large. In this tests we apply $N = 1,000$ iterations. The presented analysis measures the eigenvalues from matrix D.1, using the method ' eigs' from the library Scipy.

$$J_{2D} = CA = \begin{bmatrix} C_1 & C_2 & 0 & ... & C_2 \\ C_2 & C_1 & C_2 & ... & 0 \\ 0 & C_2 & C_1 & C_2 & \vdots \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & ... & C_2 & C_1 & C_2 \\ C_2 & ... & 0 & C_2 & C_1 \end{bmatrix} A \qquad (D.1)$$

Notice that the Jacobian (equation D.1) is composed by multiplying the connection matrix (C) with the local derivative matrix (A). In this equation, the matrix A is expressed as: $diag(\alpha - 2a_{i,j}^n\alpha)$, where α is the map parameter, and $a_{i,j}^n$ is the oscillator state.

The matrix of connections is composed by block-circulant matrices C_1 and C_2, shown in D.2 and D.3.

$$C_1 = \begin{bmatrix} 1-\epsilon & \epsilon & 0 & \ldots & \epsilon \\ \epsilon & 1-\epsilon & \epsilon & \ldots & 0 \\ 0 & \epsilon & 1-\epsilon & \ldots & 0 \\ \vdots & \vdots & & & \\ \epsilon & 0 & & \ldots & \epsilon & 1-\epsilon \end{bmatrix} \tag{D.2}$$

$$C_2 = \begin{bmatrix} \epsilon & 0 & 0 & \ldots & 0 \\ 0 & \epsilon & 0 & \ldots & 0 \\ 0 & 0 & \epsilon & \ldots & 0 \\ \vdots & \vdots & & & \\ 0 & 0 & \ldots & 0 & \epsilon \end{bmatrix} \tag{D.3}$$

In this analysis, we represent the CML state as a space, which each axis is the state of an oscillator. In this representation, the signal of the Lyapunov exponent indicates if the system has unstable orbits. If $\lambda > 0.0$, then the system has unstable orbits. In a given orbit, if a small perturbation is added in a certain direction, which locally is given by the corresponding eigenvector, then the difference between the original orbit and the perturbed orbit has a tendency to increase.

On the other hand, if $\lambda = 0.0$ the system is stable. Finally, if $\lambda < 0.0$ then the system is asymptotically stable, considering an orbit and a small perturbation, the difference between a perturbed orbit and the original orbit has a tendency to decrease.

The result of this test is presented in Figure D.1. The central region of the phase-space has negative LLE, which indicates a region where the parameters result in a system with stable orbits. Whereas the region in black has positive LLE, indicating set of parameters that produces unstable orbits.

Figure D.1 - Lyapunov exponent phase space for 32x32 CMLs
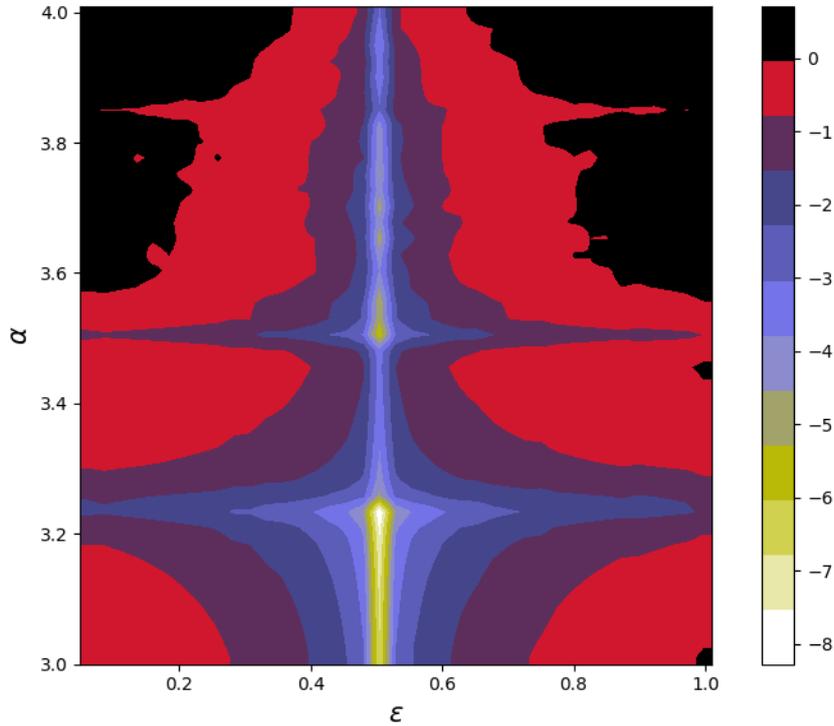
Largest Lyapunov exponent (LLE) measured in a series of CMLs, from experiment B shown in Table 3.1. Each point coordinate in this matrix represents a CML configuration, whereas the color-scaling represents the LLE. The grid density is: $\Delta\epsilon = \Delta\alpha = 0.025$.

SOURCE: Own author

Notice that the system has unstable orbits for low($\epsilon < 0.2$) and high($\epsilon > 0.8$) coupling factor. With respect to the map parameter, when the coupling factor is low or high, it is observed a similar behavior to a single map.

For instance, consider the bifurcation diagram of Logistic map (Figure D.2). This diagram associates each parameterization from a model (in x-axis) with the states of the system in a long-range iteration (in y-axis). To build this diagram a set of random initial conditions are iterated N times, then each element is plotted on the coordinates that correspond to the model parameter (in x-axis), and the iterated result (in y-axis).

Figure D.2 - Bifurcation diagram of Logistic map
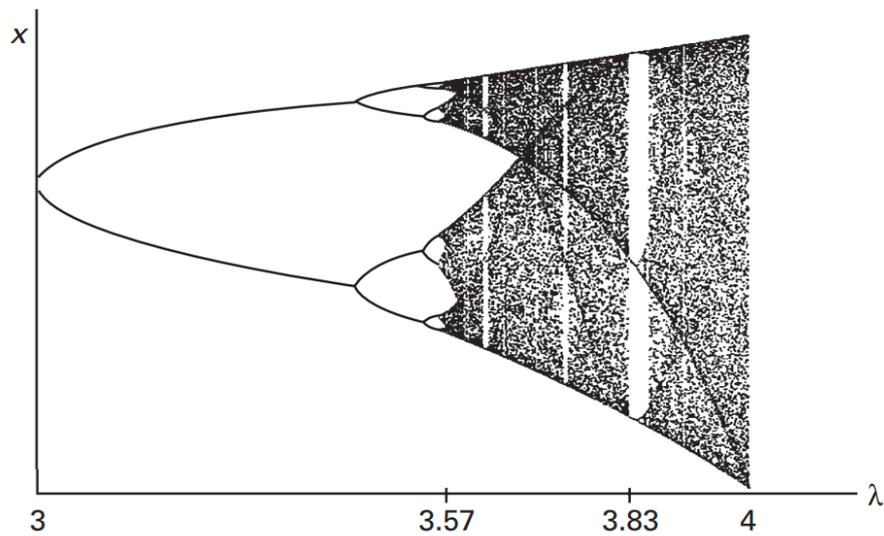
SOURCE: Smale et al. (2003)

The bifurcation diagram shows periodic orbits, which are orbits that after K iterations return to the original state, being K the period. Observe that, the period depends on the map parametrization: at $3.0 < \alpha < 3.4$ the system has period-2 orbits, at $3.45 < \alpha < 3.54$ it is observed orbits with period-4, and so on. The period duplicate as the parameter increases, from $\alpha > 3.57$, it is observed periodic orbits of all periods, which indicates chaos(SMALE et al., 2003).

Also notice that in the orbit diagram there is a blank area at $\alpha \approx 3.83$. If magnified this region, it is observed a similar bifurcation diagram when $3.0 < \alpha < 3.57$. However, this time starting with period-3 orbit, duplicating the number of stable points as α increases.

The same pattern is observed for CML with large and low coupling factors. As shown in Figure D.1, with $\alpha < 3.6$ the Largest Lyapunov exponent is negative. Whereas for $\alpha > 3.6$ the Largest Lyapunov exponent is positive. Additionally, it is also observed a window with stable orbits at $\alpha \approx 3.83$ in Figure D.1.

77

PUBLICAÇÕES TÉCNICO-CIENTÍFICAS EDITADAS PELO INPE

Teses e Dissertações (TDI)

Teses e Dissertações apresentadas nos Cursos de Pós-Graduação do INPE.

Notas Técnico-Científicas (NTC)

Incluem resultados preliminares de pesquisa, descrição de equipamentos, descrição e ou documentação de programas de computador, descrição de sistemas e experimentos, apresentação de testes, dados, atlas, e documentação de projetos de engenharia.

Propostas e Relatórios de Projetos (PRP)

São propostas de projetos técnico-científicos e relatórios de acompanhamento de projetos, atividades e convênios.

Publicações Seriadas

São os seriados técnico-científicos: boletins, periódicos, anuários e anais de eventos (simpósios e congressos). Constam destas publicações o Internacional Standard Serial Number (ISSN), que é um código único e definitivo para identificação de títulos de seriados.

Pré-publicações (PRE)

Todos os artigos publicados em periódicos, anais e como capítulos de livros.

Manuais Técnicos (MAN)

São publicações de caráter técnico que incluem normas, procedimentos, instruções e orientações.

Relatórios de Pesquisa (RPQ)

Reportam resultados ou progressos de pesquisas tanto de natureza técnica quanto científica, cujo nível seja compatível com o de uma publicação em periódico nacional ou internacional.

Publicações Didáticas (PUD)

Incluem apostilas, notas de aula e manuais didáticos.

Programas de Computador (PDC)

São a seqüência de instruções ou códigos, expressos em uma linguagem de programação compilada ou interpretada, a ser executada por um computador para alcançar um determinado objetivo. Aceitam-se tanto programas fonte quanto os executáveis.