



Universidade Estadual de Maringá
Centro de Ciências Exatas
Departamento de Física

Dissertação de Mestrado

**Uso das técnicas de mapas de recorrência e
análise de Fourier para estudar a influência
da mobilidade no jogo Pedra-Papel-Tesoura**

Acadêmico: José Vítor de Oliveira Silva

Orientador: Prof. Dr. Luiz Fernando Cótica

Coorientador: Prof. Dr. Breno Ferraz de Oliveira

Maringá, 4 de maio de 2018



Universidade Estadual de Maringá
Centro de Ciências Exatas
Departamento de Física

Dissertação de Mestrado

Uso das técnicas de mapas de recorrência e análise de Fourier para estudar a influência da mobilidade no jogo Pedra-Papel-Tesoura

Dissertação apresentada ao Departamento de Física da Universidade Estadual de Maringá, sob orientação do professor Dr. Luiz Fernando Cótica e sob coorientação do professor Dr. Breno Ferraz de Oliveira, como parte dos requisitos para obtenção do título de Mestre em Física

Acadêmico: José Vítor de Oliveira Silva

Orientador: Prof. Dr. Luiz Fernando Cótica

Coorientador: Prof. Dr. Breno Ferraz de Oliveira

Maringá, 4 de maio de 2018

Lista de Figuras

1	Ilustração dos 3 tipos de lagartos e suas respectivas gargantas [1].	1
2	Ilustração adaptada da referência [1] mostrando a hierarquia cíclica entre os três lagartos.	2
1.1	Ilustração mostrando a interação cíclica que acontece no jogo pedra-papel-tesoura ou RPS [8].	4
1.2	Ilustração mostrando uma variação do jogo RPS: o RPS, <i>lizard</i> e Spock [10].	5
1.3	Ilustração de um tabuleiro de xadrez [11].	5
1.4	Ilustração de um tabuleiro de xadrez na forma toroidal, ou seja, com condições de contorno periódicas [14].	6
1.5	Ilustração da vizinhança de von Neumann. Ao centro, representado pela letra P na posição (x, y) temo o sítio central. Nas posições $(x \pm 1, y)$ representados pela letra D temos os primeiros vizinhos horizontais. Nas posições $(x, y \pm 1)$ também representados pela letra D temos os primeiros vizinhos verticais [16].	6
1.6	Ilustração demonstrando uma rede com 16 sítios ocupados aleatoriamente com indivíduos das espécies pedra, papel e tesoura e espaços vazios.	7
1.7	Ilustração da ação de mobilidade.	8
1.8	Ilustração da ação de reprodução.	8
1.9	Ilustração da ação de predação.	9
1.10	À esquerda, a rede no estado inicial e, à direita, a mesma rede após os seis passos. Note que foram sorteadas sete ações porém, apenas seis ações foram possíveis de ocorrer.	9
2.1	Representação gráfica da série numérica $S(i)$ e de um mapa de recorrência não preenchido.	11
2.2	Representação gráfica da série numérica $S(i)$ e de um mapa de recorrência com a primeira linha preenchida.	12
2.3	Representação gráfica da série numérica $S(i)$ e de um mapa de recorrência com a primeira e segunda linha preenchida.	12
2.4	Representação gráfica da série numérica $S(i)$ e do mapa de recorrência totalmente preenchido.	13
2.5	Ilustração mostrando três mapas de recorrência de três sistemas diferentes. À esquerda, um sistema periódico, ao centro, um sistema caótico e, à direita, um sistema estocástico.	14
3.1	Ilustração da função $g(x)$ e sua área abaixo da curva S	20
3.2	Ilustração de uma integração numérica utilizando a regra do retângulo. . .	21
3.3	Ilustração da integração numérica utilizando a regra do trapézio.	21

4.1	Ilustração de uma rede recém gerada.	27
4.2	Ilustração da rede após 20 mil gerações mostrando como a adição de mobilidade altera os padrões formados na rede	28
4.3	Ilustração mostrando como a adição de mobilidade altera a série temporal e o mapa de recorrência.	30
4.4	Ilustração mostrando a razão de recorrência em função da mobilidade. A área sombreada representa o erro padrão.	31
4.5	Ilustração mostrando o determinismo em função da mobilidade. A área sombreada representa o erro padrão.	31
4.6	Ilustração mostrando o tamanho médio das linhas diagonais em função da mobilidade. A área sombreada representa o erro padrão.	32
4.7	Ilustração mostrando o tamanho da maior linha diagonal em função da mobilidade. A área sombreada representa o erro padrão.	32
4.8	Ilustração mostrando o divergente em função da mobilidade. A área sombreada representa o erro padrão.	33
4.9	Ilustração mostrando a entropia de Shannon em função da mobilidade. A área sombreada representa o erro padrão.	33
4.10	Ilustração mostrando a laminaridade em função da mobilidade. A área sombreada representa o erro padrão.	34
4.11	Ilustração mostrando o tempo de aprisionamento em função da mobilidade. A área sombreada representa o erro padrão.	34
4.12	Ilustração mostrando como a mobilidade interfere na função de autocorrelação.	35
4.13	Ilustração mostrando como o tempo de aprisionamento e o tempo característico são semelhantes.	35
4.14	Ilustração mostrando a distribuição $x_i - x_j$ formando uma curva Gaussiana. Em pontilhado está as frequências de $x_i - x_j$ e a linha vermelha é o ajuste da gaussiana	36
4.15	Ilustração mostrando como a mobilidade interfere na função de autocorrelação.	36
4.16	Ilustração mostrando o comprimento característico em função da mobilidade.	37
4.17	Ilustração mostrando o espaço de fase (já normalizado) do sistema RPS. Cada cor do gráfico representa uma espécie. Cada figura mostra como a mobilidade interfere em cada espaço de fase.	37
4.18	Figura ilustrando número efetivo e estatístico de mobilidades por predação	38
4.19	Figura ilustrando a densidade de vazios ρ_0 ao longo do tempo para diferentes valores de mobilidade.	39
4.20	Ilustração na escala di-log para MoP_E/MoP_R	39
4.21	O gráfico da densidade espectral pela frequência	40
4.22	Figura com o gráfico da densidade espectral pela frequência no intervalo de 0 Hz até 0,1 Hz.	40

Sumário

Agradecimentos	v
Resumo	vi
Abstract	vii
Introdução	1
1 Jogo RPS Estocástico	4
1.1 RPS	4
1.1.1 Rede e Vizinhança	5
1.1.2 RPS estocástico	7
2 Mapas de Recorrência	10
2.1 Mapas de Recorrência	10
2.1.1 O limite ϵ	14
2.2 Quantificações dos mapas de recorrência	15
2.2.1 Medidas baseadas na densidade	15
2.2.2 Medidas baseadas nas estruturas diagonais	15
2.2.3 Medidas baseadas nas estruturas verticais	17
3 Função de Autocorrelação	19
3.1 Função de autocorrelação	19
3.2 Integração numérica	19
3.3 Transformada de Fourier	21
3.3.1 Propriedades das transformadas de Fourier	22
3.3.2 Transformada da autocorrelação	23
3.3.3 Densidade espectral	24
3.3.4 Transformada de Fourier Discreta	24
4 Resultados	27
4.1 Rede	27
4.2 Mapas de Recorrência e suas respectivas séries temporais	29
4.3 Quantificações baseadas nos mapas de recorrência	29
4.3.1 Razão de Recorrência	29
4.3.2 Determinismo	31
4.3.3 Tamanho médio das linhas diagonais	31
4.3.4 Divergente	32
4.3.5 Entropia de Shannon	33

4.3.6	Laminaridade	33
4.3.7	Tempo de Aprisionamento	33
4.4	Função de autocorrelação	34
4.4.1	Função de autocorrelação temporal	34
4.4.2	Função de autocorrelação espacial	34
4.5	Espaço de Fase	36
4.6	Mobilidade real e mobilidade aparente	38
4.7	Densidade Espectral	40
Conclusões		41
A Códigos		43
Referências Bibliográficas		56

Agradecimentos

Agradeço ao meu orientador Dr. Luiz Fernando Cótica, pelo voto de confiança; ao meu co orientador, Dr. Breno Ferraz de Oliveira pela orientação, paciência e dedicação a meu trabalho. Aos meus colegas de pós-graduação, pelo companheirismo durante o curso; à Capes pelo suporte financeiro. E por último, mas não menos importante, sou grato à minha família pelo apoio.

Resumo

Neste trabalho, estudamos a ação de mobilidade no sistema Pedra-Papel-Tesoura. O objetivo principal foi observar como a mobilidade altera o sistema Pedra-Papel-Tesoura. Procedemos ao uso da análise de Fourier e dos mapas de recorrência e verificamos que o acréscimo de mobilidade gera espirais na rede, e conforme aumentamos a mobilidade, as espirais ficam ainda maiores e mais bem estruturadas. A análise de Fourier e os mapas de recorrência mostraram que o sistema com mobilidade baixa é estocástico e pouco correlacionado, e por outro lado, o sistema com mobilidade alta, é correlacionado.

Palavras chave: RPS, dinâmica de população, mapas de recorrência, transformada de Fourier.

Abstract

In this work we study the mobility action on RPS system. The main objective in this work is to know how the mobility action change the RPS system. In order to carry out such a study, we used the recurrence maps and the Fourier analysis. We observed that increase the mobility generate spirals. If we continue increase mobility the spirals become more bigger and stable. The Fourier analisis and Recurrence plot show us that with low mobility the system is stochastic and non correlated. For other side the system with high mobility is more periodic, correlated.

Keywords: phisycs, dinamic population, recurrence plot, Fourier transform.

Introdução

Durante o ano de 1996, B. Sinervo e C. M. Lively publicaram um trabalho [1] que abordava uma competição tipo pedra-papel-tesoura, doravante chamado de RPS (*Rock Paper Scissors*). Neste trabalho, os autores observaram os padrões de uso do território e comportamento de lagartos machos da espécie *Uta stansburiana*.

A maneira como cada lagarto defende seu território é definido de forma genética e pode ser identificada pela coloração da garganta dos machos. Ao todo, existem 3 cores de gargantas: azul, laranja e amarela, como mostra a Figura 1.

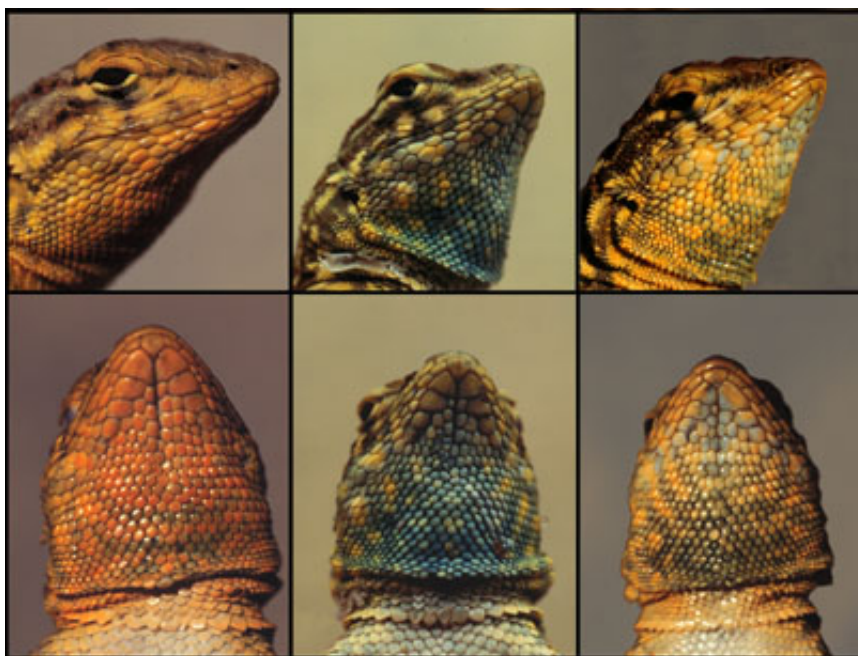


Figura 1: Ilustração dos 3 tipos de lagartos e suas respectivas gargantas [1].

Os lagartos machos de garganta amarela não possuem nenhum comportamento territorial. Os de garganta azul têm um comportamento menos agressivo e menos territorial se comparado aos de garganta laranja. Os lagartos machos de garganta laranja possuem o comportamento mais agressivo dentre os outros, além de serem muito territoriais.

O lagarto de garganta amarela se aproveita do fato de ter coloração parecida com a das fêmeas e do fato do lagarto laranja possuir um território muito grande e, por consequência, difícil de vigiar, para se mesclar com as fêmeas e se reproduzir. O lagarto de pescoço azul, por possuir um território menor, consegue vigiar todas as fêmeas e, evita que o lagarto de pescoço amarelo se mescle no grupo. O lagarto de garganta laranja, por ser mais agressivo, conquista o território do lagarto azul e fica com as fêmeas deste.

As relações de acasalamento observadas possuem uma ordem cíclica: o lagarto amarelo tem vantagem sobre o laranja que, tem vantagem sobre o azul, que tem vantagem

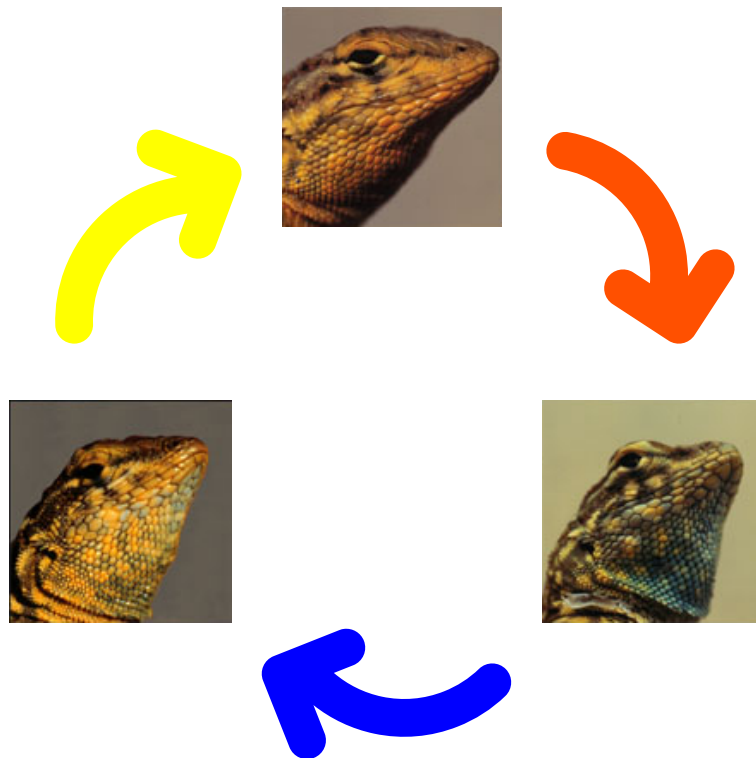


Figura 2: Ilustração adaptada da referência [1] mostrando a hierarquia cíclica entre os três lagartos.

sobre o amarelo. A Figura 2 ilustra essa interação cíclica. Essas interações promovem a diversidade biológica do sistema. Cada tipo de lagarto depende de outro tipo de lagarto para se reproduzir. Caso haja a extinção de um tipo de lagarto, a interação cíclica se desequilibra e assim temos o fim da diversidade biológica do sistema.

No ano de 2002, M. W. F. B. Kerr, M. A. Riley e J. M. Bohannan publicaram um experimento [2] com três tipos de bactérias. Nesse experimento foram observados os padrões de crescimento dessas bactérias em uma única placa de pétri. Havia uma interação cíclica para predação em que a espécie 1 predava a espécie 2, a espécie 2 predava a espécie 3 e a espécie 3 predava a espécie 1. Por meio de interações de predação e de reprodução foi possível explicar o padrão de crescimento das bactérias na placa de pétri. Em outro trabalho M. M. T. Reichendach e E. Frey [3], por meio de simulações, propuseram um novo tipo de mecanismo, a mobilidade. Além de se reproduzir e predação, as bactérias também podem se mover. Utilizando-se de simulações, apontaram como a mobilidade pode remodelar de forma significativa os padrões de crescimento e, ainda, levar ao fim da biodiversidade.

Nesse trabalho, utilizaremos os mapas de recorrências e suas quantificações junto com a análise de Fourier para entender como o sistema RPS comporta-se à medida que a ele adicionamos mobilidade.

Os mapas de recorrência, e suas quantificações constituem uma ferramenta de estudo de dados não lineares para a investigação de sistemas dinâmicos. Esses mapas são confeccionado comparando cada estado do sistema com os demais estados do próprio sistema em diferentes tempos.

Para complementarmos as quantificações, calculamos a função de autocorrelação espacial e temporal do sistema. A função de autocorrelação pode ser calculada por meio da expressão

$$C(\vec{r}) = \int \psi(\vec{r})\psi(\vec{r} + \vec{r}')d\vec{r}', \quad (1)$$

em que $\psi(\vec{r})$ é um campo escalar que descreve o valor de alguma grandeza física em todo seu espaço ou tempo. Quando calculada, ela tem um custo operacional muito alto, capaz de inviabilizar o processo computacional. Uma forma de contornar esse problema se baseia em calcular a função de autocorrelação no espaço de Fourier [4, 5].

A função de autocorrelação poder ser calculada no espaço de Fourier da seguinte maneira,

$$S(\vec{k}) = \psi(\vec{k})\psi^*(\vec{k}), \quad (2)$$

em que $\psi(\vec{k})$ é a transformada de Fourier de $\psi(\vec{r})$ e $\psi^*(\vec{k})$ é o complexo conjugado de $\psi(\vec{k})$ no espaço de Fourier. Notemos que não precisamos mais calcular uma integral. A transformada inversa de $S(\vec{k})$ nos entrega a função de autocorrelação. Tanto a transformada de Fourier de $C(\vec{r})$ quanto a transformada inversa da função $S(\vec{k})$ podem ser realizadas por meio da biblioteca FFTW (*Fastest Fourier Transform in the West*). Essa biblioteca se apoia em um algoritmo que reduz o número de processos para diminuir o tempo de cálculo [4, 5].

O procedimento numérico do trabalho realizado por M. M. T. Reichenbach e R. Frey [3] é utilizado nesta pesquisa para obtenção da série temporal que descreve a evolução do número de indivíduos de cada espécie. Por meio dessa série temporal, pretendemos montar seu mapa de recorrência, calcular suas quantificações, sua função de autocorrelação temporal e medir o seu tempo característico. Também calcularemos sua função de autocorrelação espacial e mensuramos seu comprimento característico.

No capítulo 1, dissertaremos sobre o modelo RPS estocástico, apresentaremos as definições de rede, espécie, geração, vizinhança e ações. Abordamos também as regras que foram utilizadas no código para a realização das simulações. No capítulo 2, discorreremos sobre os mapas de recorrência, sua montagem, as estruturas verticais e diagonais e suas quantificações. No capítulo 3, versamos a cerca da função de autocorrelação, do método numérico utilizado para seu cálculo, a transformada de Fourier, a transformada da autocorrelação e a transformada rápida de Fourier. No capítulo 4, apresentaremos os resultados das simulações obtidos por meio do mapa de recorrência e da função de autocorrelação. Na conclusão, discutiremos como a mobilidade interfere na rede e tecemos as considerações finais, bem como propostas para futuros trabalhos. Os códigos utilizados durante as simulações podem ser encontrados no apêndice A.

Capítulo 1

Jogo RPS Estocástico

No ano de 2002, foi publicado na revista Nature [2] um artigo que demonstrava como o padrão de crescimento de colônias de bactérias podia ser descrito por meio das regras de predador e presa do jogo RPS. Em 2007, também na revista Nature [3], foi publicado um artigo que mostrava como a introdução da mobilidade alterava a formação desses padrões. Com o passar dos anos vários outros estudos foram realizados com o modelo do jogo RPS. Na literatura há diversos estudos com o modelo do jogo RPS, alguns com foco na interface entre as espécies [6], outros com foco na mutação [7], além de estudos que generalizam o modelo RPS com mais de três espécies [6], entre outros.

Neste capítulo apresentaremos o modelo RPS estocástico, suas regras e diretrizes de evolução temporal e o conceito de rede e vizinhança de von Neumann.

1.1 RPS

Primeiramente, convém esclarecer o que é RPS. RPS é a abreviação de pedra papel e tesoura em inglês, *rock, paper e scissors*. É um jogo de mãos que se baseia na interação cíclica entre três elementos - pedra, papel e tesoura. Nesse jogo, a pedra vence a tesoura, a tesoura vence o papel e o papel vence a pedra. Na Figura 1.1 ilustramos como funciona a interação cíclica entre os elementos.



Figura 1.1: Ilustração mostrando a interação cíclica que acontece no jogo pedra-papel-tesoura ou RPS [8].

Existem variações do jogo RPS. Talvez variação mais famosa, seja o RPS *Lizard* e *Spock*. Nesse jogo, temos a adição de mais dois novos elementos: o lagarto e o Spock.

Com a adição desses novos elementos, o número de possíveis interações é aumentado. A Figura 1.2 ilustra as possíveis interações entre os elementos. Observamos que apesar do aumento do número de elementos a característica cíclica é mantida. Neste trabalho, focaremos no modelo RPS clássico, com três espécies interagindo de maneira cíclica [6,9].

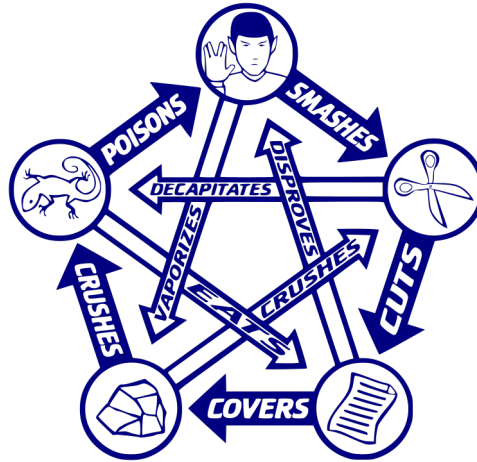


Figura 1.2: Ilustração mostrando uma variação do jogo RPS: o RPS, *lizard* e Spock [10].

Nesta pesquisa, esses elementos irão interagir em uma rede. Nesse sentido, a seguir serão introduzidos os conceitos de rede e vizinhança.

1.1.1 Rede e Vizinhança

Utilizaremos neste trabalho uma rede quadrada bidimensional. Imaginemos uma rede quadrada bidimensional como um tabuleiro de xadrez, como mostra a Figura 1.3. Cada espaço do tabuleiro é chamado de sítio. A princípio, cada sítio se encontra vazio.



Figura 1.3: Ilustração de um tabuleiro de xadrez [11].

A rede possui condição de contorno periódica, ou seja, os últimos sítios à direita da rede são conectados aos primeiros sítios da parte esquerda da rede. O mesmo se aplica aos sítios da parte superior e inferior. Usamos a rede com condição de contorno periódica para tentar simular uma rede infinita e para desprezarmos os efeitos de borda. Na Figura 1.4 apontamos como seria um tabuleiro de xadrez com condições de contorno periódicas [11]. Fizemos uso de um tabuleiro de xadrez para ilustrar como seria uma rede quadrada bidimensional, porém não significa que a rede precisa ter exatamente o mesmo

tamanho do tabuleiro e tampouco que deve ser quadrada e bidimensional. Na literatura existem estudos com redes tridimensionais [12,13]. Em nossos estudos, utilizaremos redes quadradas bidimensionais com 500×500 sítios. Se escolhêssemos uma rede muito pequena teríamos problemas de extinção precoce. Por outro lado se optássemos por uma rede muito grande teríamos problemas com o uso de memória do computador.



Figura 1.4: Ilustração de um tabuleiro de xadrez na forma toroidal, ou seja, com condições de contorno periódicas [14].

Trabalhamos com a vizinhança de von Neumann, a qual consiste nos quatro sítios ortogonais ao redor de um sítio central em uma rede bidimensional. John von Neumann foi o criador dessa vizinhança, utilizada originalmente para definir o autônomo de von Neumann e do construtor universal de von Neumann [15].

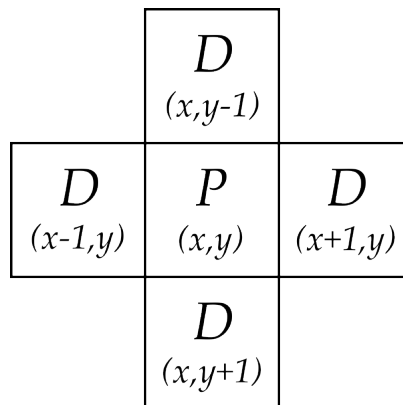


Figura 1.5: Ilustração da vizinhança de von Neumann. Ao centro, representado pela letra P na posição (x,y) temos o sítio central. Nas posições $(x \pm 1, y)$ representados pela letra D temos os primeiros vizinhos horizontais. Nas posições $(x, y \pm 1)$ também representados pela letra D temos os primeiros vizinhos verticais [16].

Os autómatos celulares são ferramentas para representar sistemas físicos compostos por elementos discretos com interações locais. O autônomo de von Neumann tem como base uma rede bidimensional infinita, em que cada sítio está ligado a seus quatro vizinhos ortogonais [17].

O construto universal de von Neumann tinha como idéia inicial uma máquina que é capaz de se auto replicar. Isto é, uma maquina que fosse capaz de cópias idênticas de si, e assim como a maquina original, as cópias da também poderiam se reproduzir, criando cópias idênticas a maquina original [17, 18].

Com as definições das regras do jogo RPS e com a definição dos conceitos de rede e vizinhança, estamos prontos para abordar o modelo RPS estocástico.

1.1.2 RPS estocástico

No modelo estocástico, usamos o método de Monte Carlo, que se baseia-se em amostragens aleatórias massivas para se obter resultados numéricos [19, 20]. Neste trabalho, utilizaremos apenas o modelo RPS estocástico, embora na literatura haja também outros modelos RPS. Um modelo bem conhecido é o que se baseia na resolução de um sistema de equações diferenciais acopladas [2, 6, 21, 22].

Para o jogo RPS estocástico, assumiremos que os elementos pedra, papel e tesoura podem ser considerados como três espécies diferentes: espécie pedra, espécie tesoura e espécie papel. Por simplicidade, podemos chamar de espécie 1, 2 e 3, respectivamente. Em cada sítio da nossa rede é sorteada uma das três espécies para ocupar ou ser deixado vazio. Na Figura 1.6 visualizamos uma rede quadrada de 4×4 preenchida aleatoriamente com 3 espécies diferentes e com espaço vazio, que é representado por \otimes [6, 9, 12, 21].

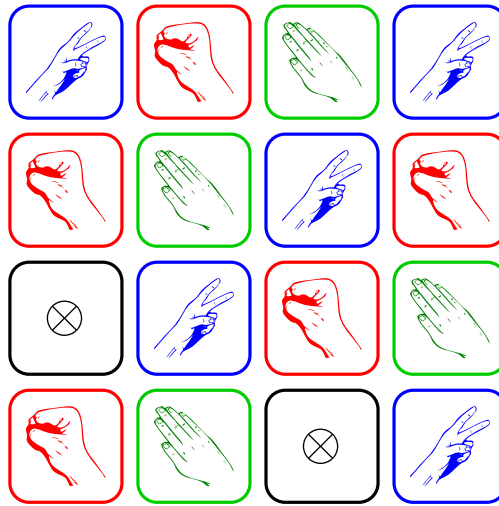


Figura 1.6: Ilustração demonstrando uma rede com 16 sítios ocupados aleatoriamente com indivíduos das espécies pedra, papel e tesoura e espaços vazios.

Cada espécie pode se reproduzir, preda e se movimentar. Para que cada ação ocorra, são necessários alguns pré requisitos, apresentados na sequência. A evolução temporal dessa rede é contada pelo número de ações, a cada N^2 ações é contabilizada uma geração, em que N é o tamanho da rede. No exemplo da Figura 1.6, cada geração tem 16 interações [6, 9, 12, 21].

A evolução da rede é regida pelo seguinte algoritmo:

1. Um indivíduo é escolhido aleatoriamente e recebe o nome de ativo, pois realizará a ação. Caso o indivíduo sorteado seja um espaço vazio, realiza-se um novo sorteio.
2. Um dos indivíduos da vizinhança de von Neumann é sorteado ao acaso. O indivíduo sorteado recebe o nome de passivo, pois sofrerá a ação do ativo.

3. A seguir, uma ação é sorteada. Essas ações podem ser a predação, a mobilidade e a reprodução. Cada ação tem uma determinada probabilidade de ser sorteada e não necessariamente elas têm o mesmo peso probabilístico. Como em nosso estudo estamos buscando investigar como a mobilidade interfere no modelo RPS estocástico, manteremos a probabilidade da predação igual à da reprodução.
4. Caso a ação sorteada seja a mobilidade, o indivíduo ativo e o indivíduo passivo trocam de posição. A Figura 1.7 ilustra a ação de mobilidade. Para que ocorra a troca de posição, não faz diferença se o passivo for vazio ou qualquer espécie. Assim que ocorre a troca de posição, é contabilizada uma ação.

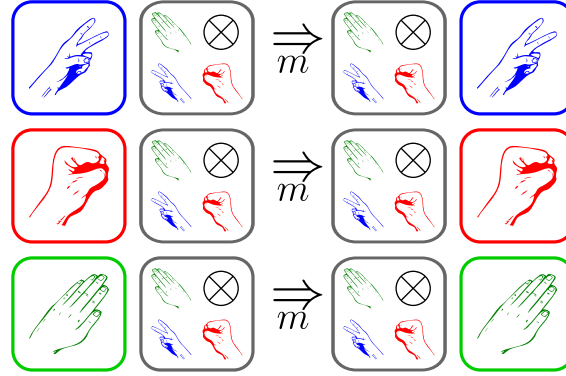


Figura 1.7: Ilustração da ação de mobilidade.

5. Caso a ação sorteada seja a reprodução, é verificado se o indivíduo passivo é um espaço vazio. Se sim, o elemento vazio se torna uma cópia do elemento ativo; caso contrário, nada acontece e não é contabilizada a ação. Na Figura 1.8, apontamos o funcionamento da ação de reprodução. Note que para que a ação de reprodução ocorra, é necessário que o elemento passivo seja um espaço vazio.

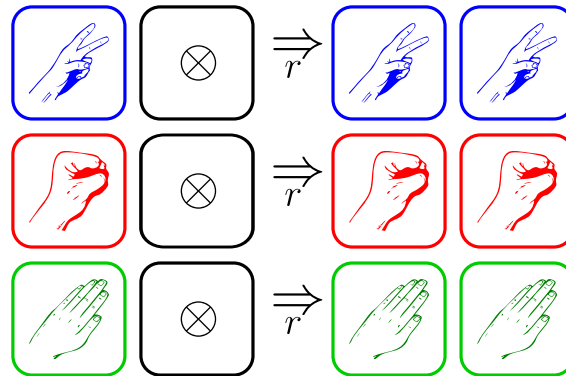


Figura 1.8: Ilustração da ação de reprodução.

6. Caso a ação sorteada seja predação, é verifica-se se o indivíduo ativo é o predador do indivíduo passivo. Se for, o elemento passivo se torna um espaço vazio e é contabilizada uma ação. Se o indivíduo ativo não for o predador do indivíduo passivo nada acontece e a ação não é contabilizada. Na Figura 1.9 ilustramos a ação de predação.

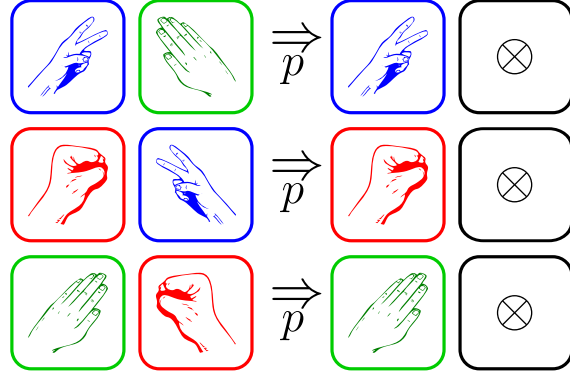


Figura 1.9: Ilustração da ação de predação.

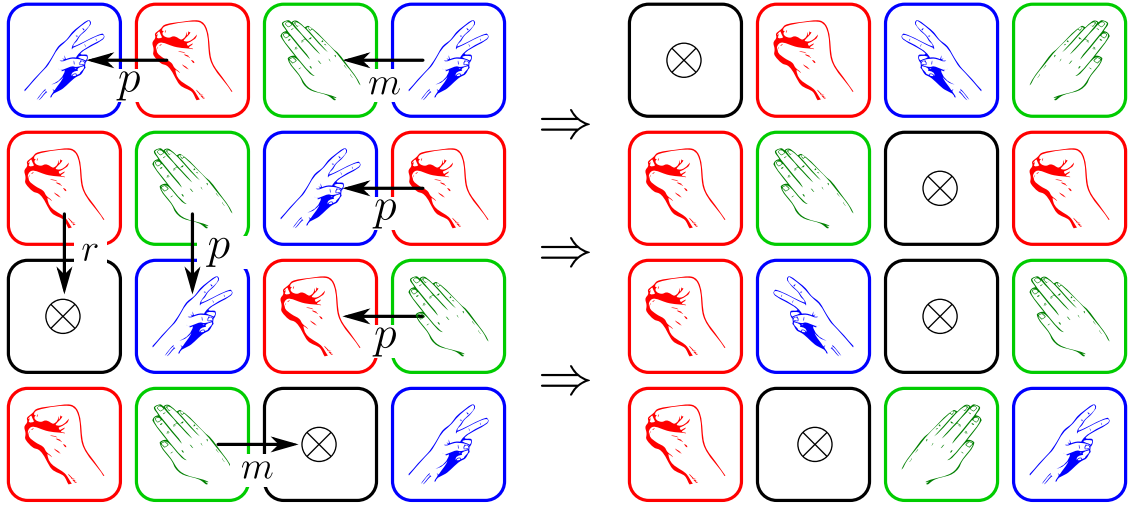


Figura 1.10: À esquerda, a rede no estado inicial e, à direita, a mesma rede após os seis passos. Note que foram sorteadas sete ações porém, apenas seis ações foram possíveis de ocorrer.

Na Figura 1.10 temos, à esquerda, uma rede em um estado inicial com algumas ações a serem realizadas e, à direita, temos a mesma rede após seis ações.

Ao fim de cada geração era contabilizada a quantidade de indivíduos de cada espécie. Usaremos esses dados para montar o mapa de recorrência.

Em nosso trabalho, utilizamos o modelo de May-Leonard, em que a reprodução e a predação ocorrem de maneira independente. No modelo Lotkva-Volterra, as ações de mobilidade e reprodução ocorrem simultaneamente, ou seja, sempre que houver predação haverá reprodução.

Capítulo 2

Mapas de Recorrência

Neste capítulo, discorreremos sobre uma das ferramentas que serão utilizadas para a análise do sistema RPS. A ferramenta em questão é o mapa de recorrência e suas quantificações. Esse mapa é uma forma de analisar sistemas não lineares. Esse tipo de análise é muito útil para sistemas que evoluem com o passar do tempo, como por exemplo o sistema RPS. Nesse sentido fizemos o uso dos mapas de recorrência para analisar o sistema RPS.

O conceito formal de recorrência foi introduzido pela primeira vez por Henri Poincaré no ano de 1890. Embora todo o esforço empreendido nos anos seguintes, a descoberta de Poincaré teria de esperar por mais 70 anos para ter seu grande avanço. Esse tempo de espera somente acabaria com o desenvolvimento de computadores rápidos e eficientes [23].

2.1 Mapas de Recorrência

Em 1987, Eckman introduziu o método de mapas de recorrência ou *Recurrence Plot*, o qual permite visualizar recorrências em sistemas dinâmicos [23].

Supomos que haja uma trajetória \vec{x}_i^N de um sistema em seu espaço de fase. A evolução desse sistema é descrita por uma série desses N vetores \vec{x} representando a trajetória em um espaço matemático qualquer. O mapa de recorrência para esse sistema é representado pela seguinte matriz:

$$R_{i,j} = \begin{cases} 1 : \vec{x}_i \approx \vec{x}_j \\ 0 : \text{do contrário} \end{cases} . \quad (2.1)$$

A “distância” em que \vec{x}_i é próximo \vec{x}_j é chamada de limite e simbolizada pela letra ϵ [23–25]. Usamos $R_{i,j}$ para representar um ponto de recorrência. Se dois pontos forem suficientemente próximos, eles são recorrentes e $R_{i,j} = 1$; do contrário, eles não são recorrentes e $R_{i,j} = 0$. Notemos que o valor de ϵ é de fundamental importância para a construção do mapa de recorrência. Adiante discutiremos qual deve ser o critério para a escolha do valor de ϵ . Reescrevendo a (2.2) em termos de ϵ , temos

$$R_{i,j} = \begin{cases} 1 : |\vec{x}_i - \vec{x}_j| < \epsilon \\ 0 : \text{do contrário} \end{cases} . \quad (2.2)$$

O mapa de recorrência pode ser representado por uma matriz $N \times N$. Essa matriz é construída comparando estado a estado do sistema. A matriz de recorrência mostra

quando os estados similares ocorrem. Fica evidente que se compararmos um ponto com ele mesmo sempre teremos uma recorrência; logo, ignoramos os casos com $i = j$ [23–26]. A fim de expurgar as possíveis dúvidas sobre a forma com a qual um mapa de recorrência é confeccionado, citamos um exemplo.

Para visualizar de maneira mais clara o mapa de recorrência, trocaremos a matriz por uma rede quadrada bidimensional. O número 1 na matriz de recorrência será representado por um quadrado preto e 0 por um quadrado em branco.

Imaginemos uma série numérica com os seguintes elementos $S(i) = 22, 20, 25, 18, 38, 45, 11$. Nesse exemplo $S(i)$ fará o papel de X_i .

Precisamos estabelecer o tamanho da nossa rede. Como a nossa série possui 7 elementos, a rede terá o formato 7×7 , conforme mostra a Figura 2.1.

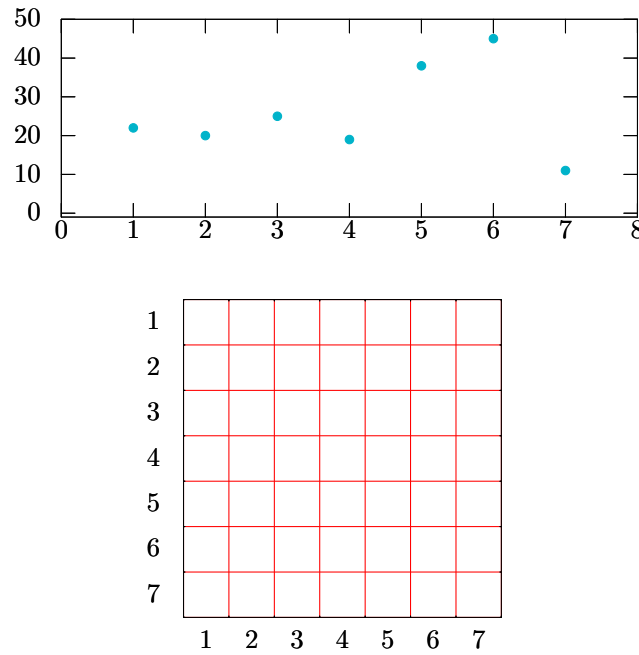


Figura 2.1: Representação gráfica da série numérica $S(i)$ e de um mapa de recorrência não preenchido.

Nesse exemplo vamos tomar o valor de ϵ como sendo 7. Primeiro fixamos $S(1)$ e o comparamos ele com demais $S(i)$. A fim de facilitar a comparação entre $S(1)$ e os demais $S(i)$, pintamos uma faixa lateral com $S(1) \pm \epsilon$, conforme mostra a Figura 2.2. Nessa Figura 2.2, vemos que os pontos $S(2)$, $S(3)$ e $S(4)$ estão dentro do alcance de recorrência. Logo preenchemos a intersecção da linha 1 com as colunas 2, 3 e 4. Lembrando que não podemos comparar $S(1)$ com ele mesmo.

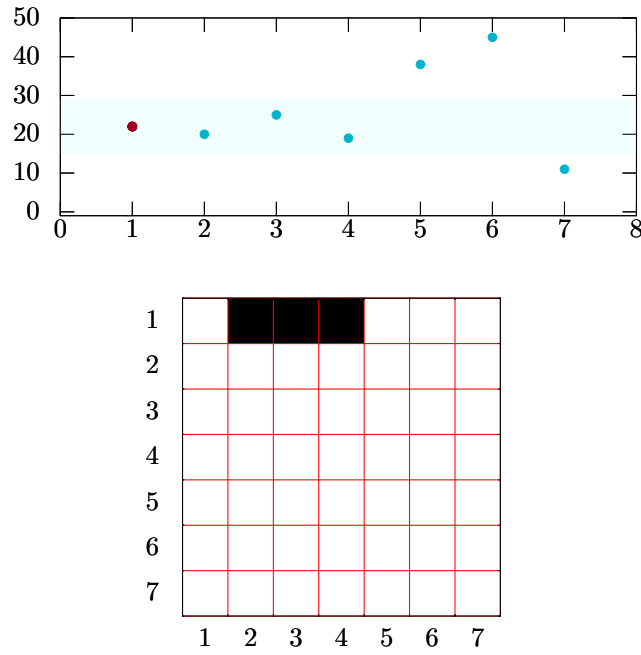


Figura 2.2: Representação gráfica da série numérica $S(i)$ e de um mapa de recorrência com a primeira linha preenchida.

No próximo passo, fixamos $S(2)$ e o comparamos com todos $S(i)$ restantes. Novamente, com finalidade de facilitar a visualização, pintamos uma faixa lateral com $S(2) \pm \epsilon$, como visualizamos na Figura 2.3. Nessa Figura 2.3, verificamos que, $S(1)$, $S(3)$ e $S(4)$ estão dentro do alcance de recorrência. Logo preenchemos a intersecção da segunda linha, com as colunas 1, 3 e 4 são preenchidas.

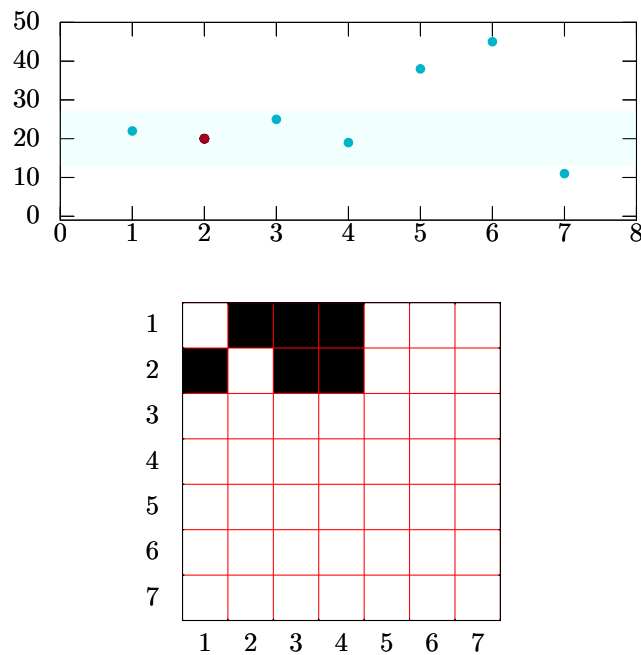


Figura 2.3: Representação gráfica da série numérica $S(i)$ e de um mapa de recorrência com a primeira e segunda linha preenchida.

Na Figura 2.4 ilustramos o mapa de recorrência concluído. A matriz de recorrência possui linhas verticais e linhas diagonais, que formam certos padrões. Em nosso exemplo usamos uma série de comprimento 7 para criar o mapa de recorrência, mas nada impede de usarmos séries de tamanho arbitrário e ver esses padrões.

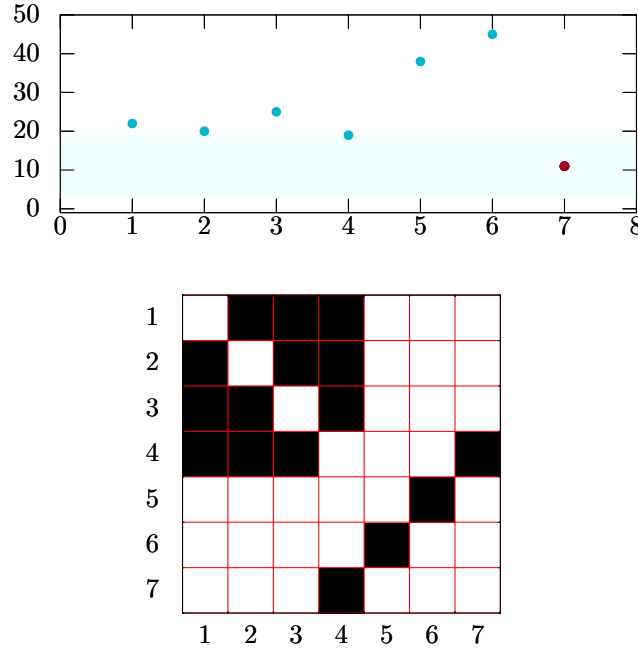


Figura 2.4: Representação gráfica da série numérica $S(i)$ e do mapa de recorrência totalmente preenchido.

Antes de seguirmos adiante, discutimos sucintamente sobre os padrões gerados nos mapas de recorrência.

Analisamos o mapa de recorrência de três sistemas diferentes, um com característica periódica (determinista), um com característica caótica (um sistema com regras mas com uma alta sensibilidade as condições iniciais) e um com característica estocástica (um sistema com uma ou mais partes aleatórias). Cada um dos sistemas referidos possui um mapa de recorrência com padrão característico, como ilustramos na Figura 2.5. Na Figura 2.5a, o sistema periódico, é possível notar longas linhas diagonais ininterruptas e quase nenhum ponto isolado. Passando para a Figura 2.5b, o sistema caótico, é possível notar linhas diagonais semelhantes ao sistema periódico, porém essas linhas diagonais não são tão longas, elas são curtas e interrompidas. Entretanto, se repararmos na parte superior esquerda da Figura 2.5b, é possível notar um pequeno retângulo que se parece com o mapa de recorrência de um sistema periódico. Passando para a Figura 2.5c, o sistema estocástico, temos um mapa de recorrência com muitos pontos isolados e nenhuma diagonal. Se olharmos para as linhas verticais veremos que os três mapas também se diferem. O sistema periódico possui linhas verticais com tamanho constantes. O sistema caótico, assim como o periódico, possui linhas verticais, mas elas têm tamanhos irregulares. O sistema estocástico não possui nenhuma linha vertical [23, 25].

Ao considerar todos os casos ditos assinalados, podemos ter uma primeira impressão sobre as linhas diagonais. Quando as linhas são curtas, podemos inferir que o sistema é menos previsível. Por outro lado, quanto maiores as linhas diagonais, mais previsível é o sistema [23, 24, 27].

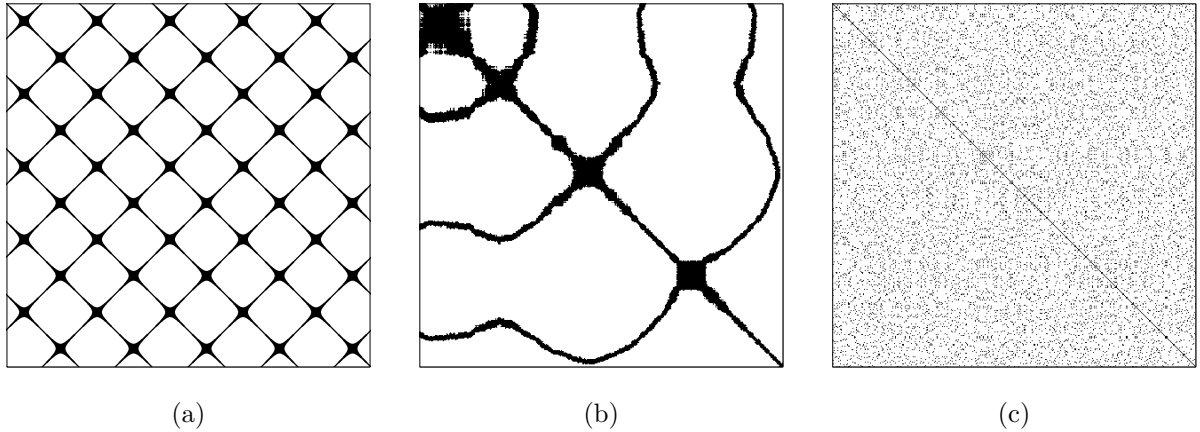


Figura 2.5: Ilustração mostrando três mapas de recorrência de três sistemas diferentes. À esquerda, um sistema periódico, ao centro, um sistema caótico e, à direita, um sistema estocástico.

Ainda podemos empreender uma análise inicial acerca das linhas verticais. No sistema periódico tínhamos linhas com tamanhos regulares. No sistema caótico, as linhas possuem tamanhos irregulares. Já o sistema estocástico não possuía linhas verticais. Logo, podemos inferir que as linhas verticais estão relacionadas ao período de oscilação do sistema [23, 24, 27].

Na sequência verificamos como é possível relacionar o tamanho das linhas diagonais e verticais com o grau de previsibilidade e periodicidade do sistema. Nossa primeira impressão nos mapas de recorrência serviu apenas para mostrar como os mapas se relacionam com seus respectivos sistemas.

Os mapas de recorrência podem ser aplicados a várias áreas tais como: medicina [28], biologia [29], cardiologia [30], acústica [26], economia [31], entre outros.

2.1.1 O limite ϵ

Apesar da escolha do valor de ϵ ser “arbitrária”, é preciso ter uma atenção especial para a escolha do mesmo. Uma seleção “errada” do valor de ϵ pode gerar um mapa de recorrência com muitos pontos de recorrência ou sem nenhum pontos de recorrência. Para ilustrar como a escolha de um ϵ errado pode interferir no resultado, imaginamos um ϵ com valor muito grande. Como o valor de ϵ é muito grande, teremos muitos pontos de recorrência, mas que na verdade não são “reais” recorrências. Por outro lado, se optarmos por um valor de ϵ muito pequeno, teremos um número muito baixo de pontos de recorrência, e não poderemos tirar informação dos mapas. Então, temos que escolher um valor para ϵ que não comprometa de forma negativa o mapa de recorrência. Na literatura existem diversas sugestões para a escolha dos possíveis valores de ϵ . Dentre estas sugestões, há uma que indica que, para comparar diversos mapas de recorrência, deve-se escolher o valor de ϵ que mantenha a densidade do mapa em torno de 10%. Como em nosso trabalho compararemos vários mapas de recorrência adotamos essa. Escolhemos um valor de ϵ que mantém o valor da densidade de pontos no mapa em torno de 10% [23, 24, 27].

2.2 Quantificações dos mapas de recorrência

Nas seções seguir, abordamos as quantificações do mapa de recorrência. Usamos as estruturas verticais, diagonais e o número de pontos na rede para realizar as quantizações.

2.2.1 Medidas baseadas na densidade

A equação (2.3) mostra como é calculada a razão de recorrência, ou RR . A razão de recorrência é a medida mais simples do sistema, e é calculada por:

$$RR(\epsilon) = \frac{1}{N^2} \sum_{i,j=1}^N R_{i,j}(\epsilon), \quad (2.3)$$

Ela mede a densidade de pontos de recorrência no mapa. Notamos que a razão de recorrência não informa sobre o grau de complexidade do sistema. Para comparar vários mapas de recorrência, é aconselhável escolher um valor de ϵ que mantenha a razão de recorrência constante [23].

2.2.2 Medidas baseadas nas estruturas diagonais

As próximas medidas apresentadas são baseadas nos histogramas $P(\epsilon, l)$ das linhas diagonais de comprimento l , que pode ser calculada por:

$$P(\epsilon, l) = \sum_{i,j=1}^N (1 - R_{i-1,j-1}(\epsilon))(1 - R_{i+1,j+1}(\epsilon)) \prod R_{i+k,j+k}(\epsilon) \quad (2.4)$$

Por simplicidade, substituiremos $P(\epsilon, l)$ por $P(l)$, omitindo ϵ . A seguir, apresentamos as quantificações baseadas nos histogramas gerados com as linhas diagonais. As medidas que serão apresentadas dizem respeito ao grau de complexidade e previsibilidade dos sistemas [23, 24, 27].

Determinismo

Como vimos, processos não correlacionados ou fracamente correlacionados e estocásticos não geram longas linhas diagonais. Entretanto, processos determinísticos, correlacionados geram longas linhas diagonais [23, 24]. A razão entre os pontos de recorrência formando linhas diagonais e todos os pontos é chamada de determinismo ou DET [23–25]. A equação (2.5) mostra como é calculado o determinismo,

$$DET = \frac{\sum_{l=l_{min}}^N lP(l)}{\sum_{l=1}^N lP(l)}, \quad (2.5)$$

em que l_{min} é o tamanho mínimo que a linha diagonal deve ter. Se $l_{min} = 1$, então $DET = 1$ independente do sistema.

O determinismo pode ser interpretado como a previsibilidade do sistema. Um sistema determinístico terá o valor de DET próximo de 1. Se o sistema for descorrelacionado, terá um valor mais próximo de 0 [23, 25, 32].

Tamanho Médio das Linhas Diagonais

Uma diagonal com comprimento l significa que durante um número l de passos temporais temos l estados recorrentes.

O tamanho médio das linhas diagonais ou $\langle L \rangle$ é a média de tempo que dois segmentos estão próximos um do outro [23, 25, 32]. A equação (2.6) mostra como é calculado o tamanho médio $\langle L \rangle$ das linhas diagonais,

$$\langle L \rangle = \frac{\sum_{l=l_{min}}^N lP(l)}{\sum_{l=l_{min}}^N P(l)}. \quad (2.6)$$

Essa pode ser interpretada também como tempo de previsibilidade médio do sistema [23–25].

Divergente

Para a próxima medida, utilizamos a maior linha diagonal. A equação (2.7) mostra como a calculamos

$$L_{max} = \max(\{l_i\}_{i=1}^{N_l}) \quad (2.7)$$

em que $N_l = \sum_{l \geq l_{min}} P(l)$. Por meio do inverso do valor de l_{max} , podemos calcular o divergente pela da seguinte equação (4.8),

$$DIV = \frac{1}{L_{max}}, \quad (2.8)$$

estas medidas são relacionadas ao expoente de Lyapunov no espaço de fase [23, 25, 32].

O expoente de Lyapunov indica a taxa de divergência das trajetórias no espaço de fase, apontando o quão sensível o sistema é em relação as condições iniciais [33].

Entropia de Shannon

A entropia de Shannon pode ser interpretada como sendo a medida de complexidade baseada na estrutura determinística de um sistema dinâmico. A entropia de Shannon é calculada a partir da frequência de distribuições das linhas diagonais no mapa de recorrência. Sistemas mais complexos e mais determinísticos contêm um valor alto de entropia. Sistemas menos determinísticos e mais caóticos e estocásticos têm um valor menor de entropia [23, 32, 34]. A equação (2.9) mostra como é calculada a entropia de Shannon,

$$ENTR = - \sum_{l=l_{min}}^N p(l) \ln(l), \quad (2.9)$$

em que $p(l)$ é a distribuição de linhas diagonais, calculada da seguinte forma

$$p(l) = \frac{P(l)}{\sum_{l=l_{min}}^N P(l)}. \quad (2.10)$$

RATIO

O razão entre o determinismo e a razão de recorrências é chamada de *RATIO*. Este é dado pela equação (2.11)

$$RATIO = N^2 \frac{\sum_{l=l_{min}}^N lP(l)}{\sum_{l=1}^N lP(l)} . \quad (2.11)$$

RATIO é utilizado mostrar dinâmicas de transições. Há certos tipos de transições nas quais a razão de recorrência diminui enquanto o determinismo se mantém constante [23, 25, 32].

Até o momento, somente discutimos medidas baseadas nas estruturas formadas com linhas diagonais no mapa de recorrência. Na próxima seção introduzimos as medidas baseadas nas estruturas verticais.

2.2.3 Medidas baseadas nas estruturas verticais

Para as medidas baseadas nas estruturas verticais utilizaremos o histograma das linhas verticais, o qual, no mapa de recorrência é calculado da seguinte forma

$$P(\epsilon, v) = \sum_{i,j=1}^N (1 - R_{i,j})(1 - R_{i,j+v}) \prod_{k=0}^{v-1} R_{i,j+k} . \quad (2.12)$$

Novamente, por simplicidade omitiremos o termo ϵ de $P(\epsilon, v)$, mantendo apenas com $P(v)$ [23, 25, 32].

Laminaridade

Análoga à definição de determinismo, temos a laminaridade *LAM*. A laminaridade é definida como sendo a razão entre as linhas verticais por todos os pontos de recorrência. A equação (2.13) mostra como é calculada a laminaridade

$$LAM = \frac{\sum_{v=v_{min}}^N vP(v)}{\sum_{v=1}^N vP(v)} . \quad (2.13)$$

A laminaridade pode ser interpretada como sendo a ocorrência de estados laminares no sistema. Porém, não descreve nada sobre o tamanho desses estados laminares [23, 25, 32].

Tempo de aprisionamento

O tempo de aprisionamento *TT* estima o tempo médio no qual o sistema fica “preso” em um estado [23, 25, 32].

A equação (2.14) mostra a forma como é calculado o tempo de aprisionamento.

$$TT = \frac{\sum_{v=v_{min}}^N vP(v)}{\sum_{v=v_{min}}^N P(v)} \quad (2.14)$$

Neste capítulo, discorremos acerca de uma das ferramentas de análise que utilizaremos no sistema RPS. No próximo capítulo, contemplamos a análise de Fourier.

Capítulo 3

Função de Autocorrelação

Neste capítulo, tratamos de outra quantificação utilizada para estudar o modelo RPS, a análise de Fourier. No decorrer deste capítulo, abordaremos a função de autocorrelação, transformada de Fourier e densidade espectral.

3.1 Função de autocorrelação

A função de autocorrelação é uma ferramenta matemática que possui várias interpretações. É capaz de avaliar o quanto uma função se relaciona com ela mesma ao longo do tempo ou espaço. Também pode ser utilizada para identificar sinais periódicos omitidos por ruído e identificar um sinal fundamental entre outras coisas. [4, 20, 35]

O valor da função de autocorrelação pode variar dentro no intervalo de -1 e 1. Nos pontos 1, 0 e -1 temos, respectivamente, a correlação perfeita, a não correlação e a anti correlação perfeita [36].

Supondo um sinal contínuo $\psi(t)$, a sua função de autocorrelação é definida segundo a equação (3.1).

$$Corr(\psi_t, \psi_t) = \frac{\int_{-\infty}^{\infty} \psi(\tau)\psi(t + \tau)d\tau}{\int_{-\infty}^{\infty} \psi(\tau)\psi^*(\tau)d\tau} . \quad (3.1)$$

Em nosso trabalho, pretendemos calcular a função de autocorrelação da rede e da série temporal dos indivíduos. Esse cálculo pode ser inviável devido ao grande número de elementos da rede. Para contornarmos esse problema, usamos a transformada de Fourier e da transformada da autocorrelação. Neste capítulo, abordaremos mais sobre essa técnica.

Utilizamos em nosso trabalho a linguagem de programação C a qual não trabalha com funções contínuas. Nessa direção, teremos que utilizar a forma discreta das funções. Como pretendemos calcular a transformada de Fourier, precisamos primeiro abordar a forma discreta da integral.

3.2 Integração numérica

A integral definida de uma função escalar $g(x)$ da variável real x é definida pela equação (3.2).

$$I_{ab} = \int_a^b g(x)dx = \lim_{\Delta x_i \rightarrow 0} \sum_{i=1}^n g(x_i)\Delta x_i. \quad (3.2)$$

O intervalo (a, b) é o intervalo de integração no qual definimos um conjunto de pontos, $x_1 = a$, $x_2 = x_1 + \Delta x_1$, $x_3 = x_2 + \Delta x_2$, ..., $x_{i+1} = x_i + \Delta x_i$, ..., $x_n = b$ [4, 20, 37].

Graficamente, calcular a integral corresponde a calcular a área abaixo da curva, conforme ilustra a Figura 3.2.

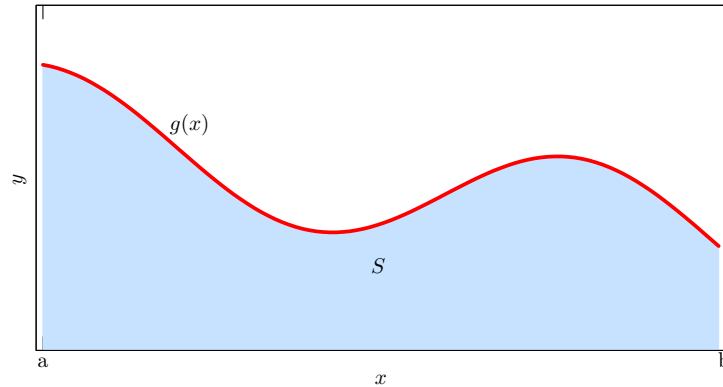


Figura 3.1: Ilustração da função $g(x)$ e sua área abaixo da curva S .

A nossa ideia é, por meio da equação (3.2), calcular a integração no limite de Δx_i “suficientemente pequeno”. Mas o que seria esse “suficientemente pequeno”? O intervalo é “suficientemente pequeno” quando a diferença no valor calculado de forma numérica e de forma analítica é muito pequena.

Ao supormos que a nossa função $g(x)$ é seja continua e diferenciável, não há motivos para não escolher $\Delta x_i = \Delta x$; em outras palavras, tomar o intervalo Δx_i como sendo apenas um Δx [4, 37]. Com isso a equação (3.2), fica

$$I_{ab} = \Delta x \sum_{i=1}^n g(x_i). \quad (3.3)$$

Podemos, por simplicidade de notação, trocar $g(x_i)$ por g_i , assim a equação (3.3) fica

$$I_{ab} = \Delta x \sum_{i=1}^n g_i. \quad (3.4)$$

Graficamente, a equação (3.4) corresponde a construir um retângulo entre x_i e x_{i+1} , com altura g_i para cada x_i . Assim, aproximamos a área abaixo da curva $g(x)$, como a soma da área dos retângulos [4], conforme ilustra a Figura 3.2.

Regra do trapézio

Na seção anterior, discorremos sobre o método de integração por meio de retângulos. Nesta seção, versamos acerca de um método que, ao invés de retângulos usamos trapézios [4, 20].

Para tal aproximação, realizaremos uma interpolação linear entre os pontos consecutivos $(x_i, g_i)\Delta x$. Lembrando que a área do trapézio é $\frac{1}{2}\Delta x(g_i + g_{i+1})$ [4, 37]. Somando as áreas de todos os trapézios, isto é,

$$I_{ab} = \frac{\Delta x}{2}(g_1 + g_2) + \frac{\Delta x}{2}(g_2 + g_3) + \frac{\Delta x}{2}(g_3 + g_4) + \cdots + \frac{\Delta x}{2}(g_{n-1} + g_n), \quad (3.5)$$

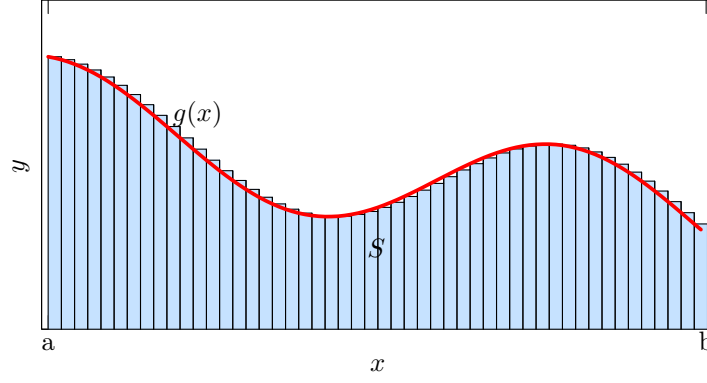


Figura 3.2: Ilustração de uma integração numérica utilizando a regra do retângulo.

colocando o termo Δx em evidência e somando os termos $\frac{1}{2}g_i$, temos

$$I_{ab} = \Delta x \left(\frac{1}{2}g_1 + g_2 + g_3 + \dots + g_{n-1} + \frac{1}{2}g_n \right), \quad (3.6)$$

Reagrupando todos os termos g_i em uma somatória $\sum_{i=1}^n g_i$, temos

$$I_{ab} = \Delta x \left(\sum_{i=1}^n g_i - \frac{1}{2}(g_1 - g_n) \right). \quad (3.7)$$

Quando escolhermos Δx_i iguais, as equações (3.4) e (3.7) são muito semelhantes. A única diferença entre ambas é que a equação (3.7) tem o termo $\frac{1}{2}(g_1 - g_n)$. Notemos que, n muito grande o termo extra se torna muito pequeno e passa a ser desprezível [4]. Na Figura 3.2 visualizamos a integração numérica utilizando o método do trapézio.

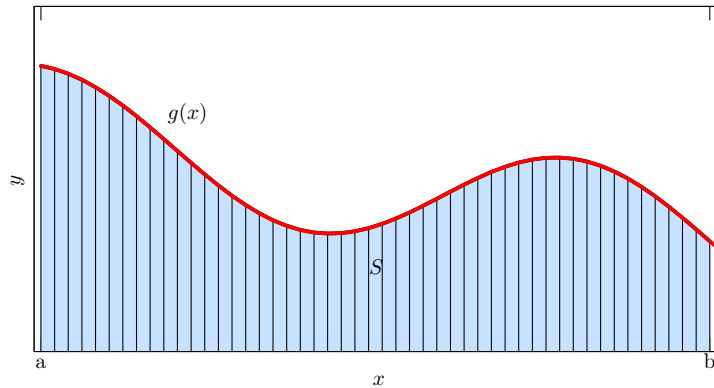


Figura 3.3: Ilustração da integração numérica utilizando a regra do trapézio.

3.3 Transformada de Fourier

Na seção anterior, estudamos o método de integração numérica com o intuito de o utilizarmos na transformada de Fourier. Nesta seção, apresentamos a transformada de

Fourier, suas propriedades, sua forma discreta e a transformada de Fourier rápida. Usamos a letra t para denotar a variável independente da função $f(t)$. Ressaltamos que t não precisa necessariamente indicar o tempo, pois a transformada de Fourier também pode ser aplicada no espaço. Usaremos como variável independente para a transformada de Fourier a frequência angular $\omega = 2\pi\nu$. Caso ocorra da variável independente ser espacial, trocaremos a frequência angular ω pelo número de onda $k = 2\pi/\lambda$ onde λ é um “comprimento de onda”. Para simplificar, chamaremos ω de apenas frequência [4, 38].

A transformada de Fourier pode ser expressa pela seguinte integral

$$h(\omega) = \int_{-\infty}^{+\infty} e^{-i\omega t} f(t) dt, \quad (3.8)$$

em que $h(\omega)$ é chamada de transformada de Fourier da função $f(t)$. O termo $e^{-i\omega t}$ é, por vezes, chamado de núcleo da transformada [38]. A transformada de Fourier da função $f(t)$ somente está definida se $f(t)$ satisfizer certas propriedades. A função deve ser integrável em qualquer região finita. Na prática, isso significa que a $f(t)$, na pior das hipóteses, tem descontinuidades de salto ou descontinuidades infinitas razoáveis [38].

Todas as funções $f(t)$ possuem uma correspondência biunívoca com suas respectivas transformadas. Tal correspondência é chamada de transformada de Fourier inversa, que pode ser expressa pela seguinte equação

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} e^{i\omega t} h(\omega) d\omega. \quad (3.9)$$

O termo $1/2\pi$ pode ser alocado na equação (3.8) em vez da equação (3.9) ou ainda, pode-se adicionar o fator $1/\sqrt{2\pi}$ nas duas equações [4, 38].

Na próxima seção abordaremos algumas propriedades da transformada de Fourier. Antes de seguirmos adiante, vamos introduzir uma nova notação, utilizaremos $\mathcal{F}[f(t)]$ para representar a transformada de Fourier da função $f(t)$.

3.3.1 Propriedades das transformadas de Fourier

Abordaremos na sequência, uma série de propriedades básicas da transformada de Fourier. Estas propriedades podem ser utilizadas para otimizar a programação em C. Utilizamos algumas dessas propriedades para explicar a transformada de Fourier rápida.

Paridade

Se a função $f(t)$ é real,

$$\text{Re}[h(-\omega)] = \text{Re}[h(\omega)] \quad (3.10)$$

e

$$\text{Im}[h(-\omega)] = -\text{Im}[h(\omega)]. \quad (3.11)$$

Com as duas equações, (3.10) e (3.11), podemos concluir que

$$h(-\omega) = h^*(\omega). \quad (3.12)$$

Com o auxílio das equações (3.10), (3.11) e (3.12), podemos concluir uma série de relações [4, 38]:

1. Se a função $f(t)$ for real e par, logo sua transformada de Fourier $h(\omega)$ será real [4,38].
2. Se a função $f(t)$ for real e ímpar, logo sua transformada $h(\omega)$ será imaginária [4,38].
3. Se a função $f(t)$ for imaginária, então as propriedades 1 e 2 estarão válidas se trocarmos a parte real pela parte imaginária e vice versa.

Linearidade

A transformada de Fourier da soma é igual à soma da transformada de Fourier. A equação (3.13) ilustra a propriedade da linearidade,

$$\mathcal{F}[af_1(t) + bf_2(t)] = a\mathcal{F}[f_1(t)] + b\mathcal{F}[f_2(t)], \quad (3.13)$$

sendo a e b constantes.

Inversão temporal

$$[f(-t)] = h(-\omega) = g^*(\omega) \quad (3.14)$$

A última parte da igualdade é somente válida quando a função $f(t)$ é real [4], conforme foi visto na equação (3.10).

Transformada da convolução

A convolução da função f_1 com a função f_2 é definida como a seguinte função de t

$$f_1 \otimes f_2 = \int_{-\infty}^{\infty} f_1(\tau) f_2(t - \tau) d\tau. \quad (3.15)$$

A sua transformada de Fourier é assim descrita

$$\mathcal{F}[f_1 \otimes f_2] = \mathcal{F}[f_1]\mathcal{F}[f_2]. \quad (3.16)$$

3.3.2 Transformada da autocorrelação

Reescrevendo a função de autocorrelação (3.1) com $f(t)$ temos

$$Corr(f_t, f_t) = \frac{\int_{-\infty}^{\infty} \psi(\tau) \psi(t + \tau) d\tau}{\int_{-\infty}^{\infty} \psi(\tau) \psi^*(\tau) d\tau}. \quad (3.17)$$

A variável de integração τ está presente em ambos os argumentos de f e só difere da equação (3.15) por um sinal de negativo. Utilizando a propriedade de inversão temporal (3.14) em conjunto com a transformada da convolução (3.15), obtemos

$$\mathcal{F}[Corr(f_t, f_t)] = h(-\omega)h(\omega) = h^*(\omega)h(\omega). \quad (3.18)$$

Para usar a propriedade da linearidade estamos supondo que $f(t)$ é real [4].

3.3.3 Densidade espectral

A densidade espectral, ou espectro, é definida como sendo a transformada de Fourier da função de autocorrelação. Ela é função real e positiva, associa uma frequência variável a um processo estocástico ou a uma função determinista no tempo.

A densidade espectral é calculada da seguinte forma

$$\Phi(\omega) = \frac{1}{2\pi} h^*(\omega) h(\omega). \quad (3.19)$$

Se olharmos para a equação (3.18) e (3.19), vemos que elas apenas diferem por uma divisão por 2π [4, 39, 40].

3.3.4 Transformada de Fourier Discreta

Neste trabalho, versamos explicitamente das funções $f(t)$ com suporte finito $[t_0, t_N]$, de tal modo que $f(t) = 0$ se t não estiver entre $[t_0, t_N]$. Para discretizar a transformada de Fourier usamos a regra do trapézio. Então, combinando a equação (3.7) com (3.8) e supondo um número n suficientemente grande para ignorarmos o termo $-1/2(f_1 + f_n)$ da equação (3.7), temos [4]

$$h_k = \Delta t \sum_{j=1}^n e^{-i\omega_k t_j} f_j. \quad (3.20)$$

Por simplicidade, utilizamos $f_j = f(t_j)$ bem como $h_k = h(\omega_k)$. Como desejamos manter a correspondência biunívoca entre f e h , devemos manter o mesmo número n , t_j para ω_k . Para se caracterizar uma variação na função $f(t)$, é preciso que exista um mínimo de deslocamento do ponto f_j original. Tal deslocamento afeta a $f(t)$ num intervalo igual a $\tau = 2\Delta t$, este, portanto, é o mínimo período de oscilação da $f(t)$ que pode ser percebido na $h(\omega_k)$. Consequentemente a máxima frequência $\nu = 1/(2\Delta t)$, como $\omega_n = 2\pi\nu$ então temos $\omega_n = \pi/\Delta t$. Também devemos levar em conta as possíveis frequências com valor negativo, com módulo de até $\pi/\Delta t$. Curiosamente, se escolhermos $\omega_1 = -\pi/\Delta t$ e $\omega_n = \pi/\Delta t$, então $h(\omega_1) = h(\omega_n)$, pois os t_j são números inteiros de Δt e, portanto, $\pm(\pi/\Delta t)t_j$ diferem por apenas números inteiros de 2π e por consequência $e^{-i(\pi/\Delta t)t_j} = e^{i(\pi/\Delta t)t_j}$. Para $g(\omega_k)$ possuir, característica periódica em k , precisamos que $\omega_k = \omega_{k'}$ onde $k' = k + 2\pi$ [4, 5]. Para que a situação descrita anteriormente seja uma realidade, escolheremos

$$\Delta\omega = \frac{2\pi}{n} \quad (3.21)$$

e

$$\omega_1 = \frac{-\pi}{\Delta t}, \quad \omega_{k+1} = \omega_1 + k\Delta\omega, \quad \omega_n = \frac{\pi}{\Delta t} - \Delta\omega. \quad (3.22)$$

Destacamos que o intervalo $\Delta\omega$ entre dois pontos consecutivos, ω_k e ω_{k+1} , independe do espaço de tempo Δt entre dois pontos consecutivos t_j e t_{j+1} . No máximo, o módulo da frequência ω_n tem uma dependência exclusiva de Δt , com independência do tamanho de t . O fato apresentado anteriormente é de grande importância para orientar a discretização de t . Caso os pontos ω_k do gráfico de $h(\omega)$ esteja muito afastados, não produzindo um gráfico suave, devemos aumentar o intervalo de tempo $t_n - t_1$, no qual definimos a $f(t)$ [4].

Reformulamos a equação (3.20) na sua forma matricial. Para tal feito, definiremos a matriz quadrada \mathcal{W} de elementos

$$\mathcal{W}_{kj} = e^{-i\omega_k t_j}, \quad (3.23)$$

e os vetores coluna f , de elementos f_j e h , de elementos forma

$$h = \mathcal{W} * f \quad (3.24)$$

em que o sinal $*$ representa o produto matricial. A transformada inversa é obtida aplicando h a matriz inversa \mathcal{W}^{-1} [4],

$$f = \mathcal{W}^{-1} * h. \quad (3.25)$$

Transformada de Fourier rápida

Supomos que temos uma função $f(t)$ cheia de detalhes, precisaremos de um número N muito grande de pontos para discretizá-la em t para obtermos uma representação fiel. A matriz \mathcal{W} que foi introduzida na secção anterior, terá N^2 elementos complexos. Se, por acaso, $N \approx 10^5$ pontos, então $N^2 \approx 10^{10}$. Tratando-se de computadores, a quantidade de memória necessária para armazenar todos estes pontos e o tempo calcular o produto $\mathcal{W} * f$ torna a operação inviável. Este problema se torna muito pior quando a variável independente possui duas ou três dimensões como, por exemplo, transformadas de Fourier espacial [4, 5].

A (*fast Fourier transform*) *fft* é um algoritmo rápido e eficiente para realizar a transformada de Fourier discreta. A *fft* tem sido desenvolvida com a colaboração de vários autores, abundantemente usada a partir da metade da década de 1960. Todavia há referências a tal procedimento, iniciando por Gauss, em 1805 [4].

Apresentaremos brevemente a premissa básica com a qual o algoritmo trabalha. Supomos que podemos escrever o número N como uma potência de 2, ou seja $N = 2^n$, sendo n um número inteiro. Como estamos trabalhando com funções $f(t)$ de suporte limitado, N pode não ser uma potência de 2. Caso isso ocorra, completamos $f(t)$ com zeros até que N possa ser escrito como uma potência de 2. Para simplificar o processo, escolhemos unidades tais que $\Delta t = 1$ e as frequências ω no intervalo $[0, 2\pi)$. Excluiremos o valor de 2π , pois devido às condições de contorno periódicas, $\omega = 2\pi$ é equivalente a $\omega = 0$. Dora-vante, as seguintes notações, $f_j = f(t_j)$ e $h(k) = h(\omega_k)$, com $\omega_k = 0, 2\pi/N, \dots, 2k\pi/N, \dots$. Assim, reescrevendo a equação (3.20) temos

$$h_k = \sum_{j=0}^{N-1} e^{(-2\pi i/N)kj} f_j. \quad (3.26)$$

Como N é par, podemos separar o somatório acima em dois, com cada parte contendo metade dos termos, isto é,

$$h_k = \sum_{j=par}^{N-2} e^{(-2\pi i/N)kj} f_j + \sum_{j=impar}^{N-2} e^{(-2\pi i/N)kj} f_j, \quad (3.27)$$

podemos, ainda, reescrevê-la como

$$h_k = \sum_{j=0}^{N/2-1} e^{(-2\pi i/N)2kj} f_{2j} + \sum_{j=0}^{N/2-1} e^{(-2\pi i/N)k+2j} f_{2j+1}. \quad (3.28)$$

Introduzimos a notação de $\mathcal{W} = e^{-2\pi i/N}$ e $N' = N/2$. Reescrevendo a equação (3.28), temos

$$h_k = \sum_{j=0}^{N'-1} e^{(-2\pi i/N')^{2kj}} f_{2j} + \mathcal{W}^k \sum_{j=0}^{N'-1} e^{(-2\pi i/N')^{kj}} f_{2j+1} , \quad (3.29)$$

ou seja,

$$h_k = h_k^{par} + \mathcal{W}^k h_k^{impar} . \quad (3.30)$$

Como podemos ver, separamos a transformada de Fourier em duas partes, uma parte com os pontos pares de t , e outra parte com os pontos ímpares de t . Usando esse método a equação (3.24) passa a ter $2N'^2 = N^2/2$ multiplicações ao invés de N^2 da forma original. Como $N' = N/2$ e $N = 2^n$ temos, $N' = 2^{n-1}$, fazendo com que N' também seja um número par. Logo é possível transformar cada transformada da equação (3.30) em duas, cada uma para $N/4$ pontos. Podemos continuar o processo até que cada transformada se reduza a um único termo. Pode-se verificar que para que o procedimento seja completo são necessários $N \log_2 N$ multiplicações de números complexos, ao invés dos N^2 da forma original. Para N muito grande, essa redução é enorme, podendo reduzir semanas de computação para alguns minutos. Quase todas as linguagens de programação já possuem uma rotina otimizada para calcular a transformada rápida de Fourier [4].

Capítulo 4

Resultados

Neste capítulo, apresentaremos os resultados das nossas simulações, nas quais utilizamos uma rede quadrada de 500×500 , com condições de contorno periódicas. As condições iniciais da rede foram geradas de maneira aleatória. Ao final de cada geração contabilizamos o número indivíduos de cada espécie. Inicialmente, aguardamos a rede evoluir por 10 mil gerações para, então começar a contabilizar a quantidade de indivíduos de cada espécie. Essa espera de 10 mil gerações serviu para a formação dos padrões.

Antes de analisarmos os resultados obtidos por meio dos mapas de recorrência e da análise de Fourier, observaremos como a introdução da mobilidade altera o sistema RPS.

4.1 Rede

A Figura 4.1 ilustra uma rede que acabou de ser gerada. À direita da rede temos uma escala de cor. Cada espécie é representada pelas cores que correspondem ao número 1 (vermelho), 2 (azul) e 3 (amarelo); o vazio é representado pelo número 0 (branco).

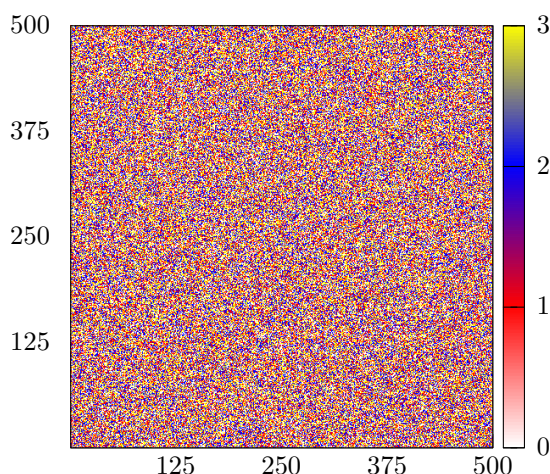


Figura 4.1: Ilustração de uma rede recém gerada.

Apresentamos as redes com diferentes valores de mobilidade. A soma probabilística da mobilidade m , predação p e reprodução r deve ser 1, como mostra a equação (4.1).

$$m + p + r = 1 \quad (4.1)$$

Utilizamos a equação (4.2) para escolhermos o peso estatístico da predação e reprodução.

$$r = p = \frac{1 - m}{2} \quad (4.2)$$

A Figura 4.2 ilustra como a ação de mobilidade altera os padrões de crescimento da rede. Na Figura 4.2a, que representa a rede sem nenhuma mobilidade, vemos pequenas “ilhas isoladas” de cada espécie. Na Figura 4.2b, referente a rede com 30% de mobilidade, há o início de formação de espirais. Notemos que, na Figura 4.2b, as espirais ainda não estão bem estruturadas. Na Figura 4.2c, que representa a rede com 60 % de mobilidade, as espirais estão mais bem formadas, estão maiores e melhor estruturadas do que na Figura 4.2b. A Figura 4.2d, relativa à rede com 90% de mobilidade, vemos que as espirais estão maiores do que as nas Figuras 4.2b e 4.2c.

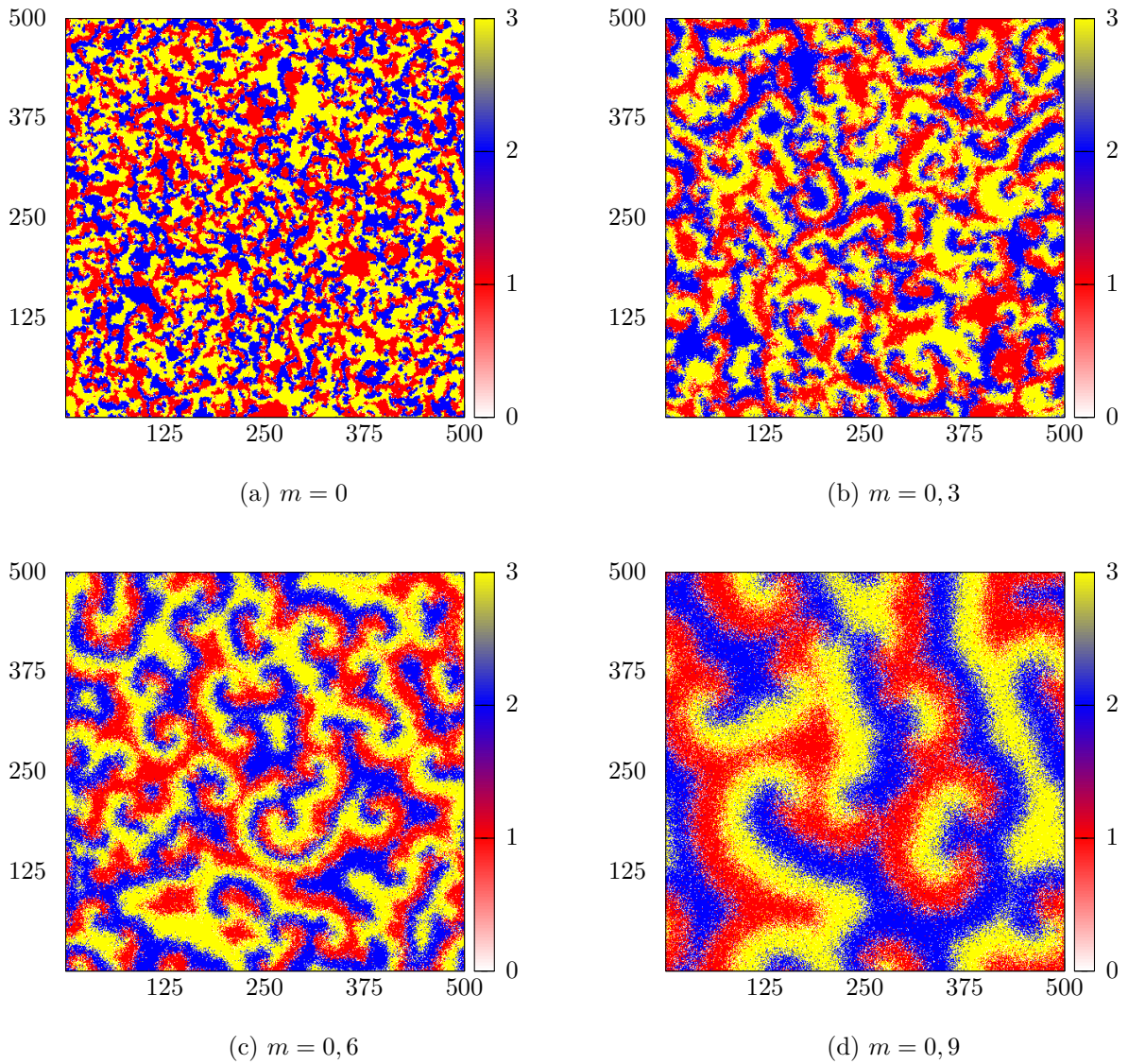


Figura 4.2: Ilustração da rede após 20 mil gerações mostrando como a adição de mobilidade altera os padrões formados na rede

O acréscimo de mobilidade no sistema gera padrões em forma de espiral. Conforme

aumentamos o valor da mobilidade na rede, maior e melhor formadas, são as espirais. Esse comportamento, é relatado em vários trabalhos, [2, 3, 6, 9, 22]. É perceptível que a mobilidade gera algum tipo de ordem na rede.

4.2 Mapas de Recorrência e suas respectivas séries temporais

Nesta seção, discorremos sobre os mapas de recorrência. Por simplicidade versamos acerca dos resultados de somente uma das três espécies. O comportamento apresentado é o mesmo para todas as espécies.

Na Figura 4.3 ilustramos 4 séries temporais com diferentes valores de mobilidade, na sequência de cada série temporal está seu mapa de recorrência. O valor do limite ϵ , utilizado para montar os mapas de recorrência, foi $\epsilon = \sigma/6$ em que σ é o desvio padrão amostral da série temporal, calculamos σ da seguinte forma,

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}} . \quad (4.3)$$

Escolhemos esse valor de ϵ de forma que a densidade de pontos de recorrência na rede seja de aproximadamente 10%.

Ao analisarmos a série temporal da Figura 4.3, sem nenhuma mobilidade, vemos que a série temporal oscila bruscamente e em um intervalo de tempo muito curto. Analisando a série temporal da Figura 4.3, sistema com 30% de mobilidade, vemos que a série temporal começa a oscilar menos, mas ainda oscila de forma brusca. Analisando a série temporal da Figura 4.3, sistema com 60% de mobilidade, vemos a série oscilar de forma mais suave. Na Figura 4.3, o sistema com 90% de mobilidade, temos a série temporal oscilando de forma suave, a série também oscila menos se comparada ao sistema sem mobilidade.

Até o momento observamos que a adição de mobilidade faz com que o sistema se ordene. Na seção seguinte, abordamos as quantificações do mapa de recorrência.

4.3 Quantificações baseadas nos mapas de recorrência

Apresentamos as quantificações geradas a partir do mapa de recorrência: razão de recorrência, determinismo, tamanho médio das linhas diagonais, divergente, laminaridade e entropia de Shannon.

4.3.1 Razão de Recorrência

A primeira medida que abordaremos é a razão de recorrência RR . Como vimos no capítulo 2, a razão de recorrência é a densidade de pontos de recorrência na rede [23, 24, 27]. Para a comparação de vários mapas de recorrência é aconselhável fixar o valor de RR . Em nosso trabalho, decidimos fixar o valor de RR próximo de 10%. Para isso, ajustamos o valor do limite ϵ de forma que $RR \approx 10$. O valor de ϵ então ficou como sendo $\sigma/6$, em que σ é o desvio padrão amostral da série temporal.

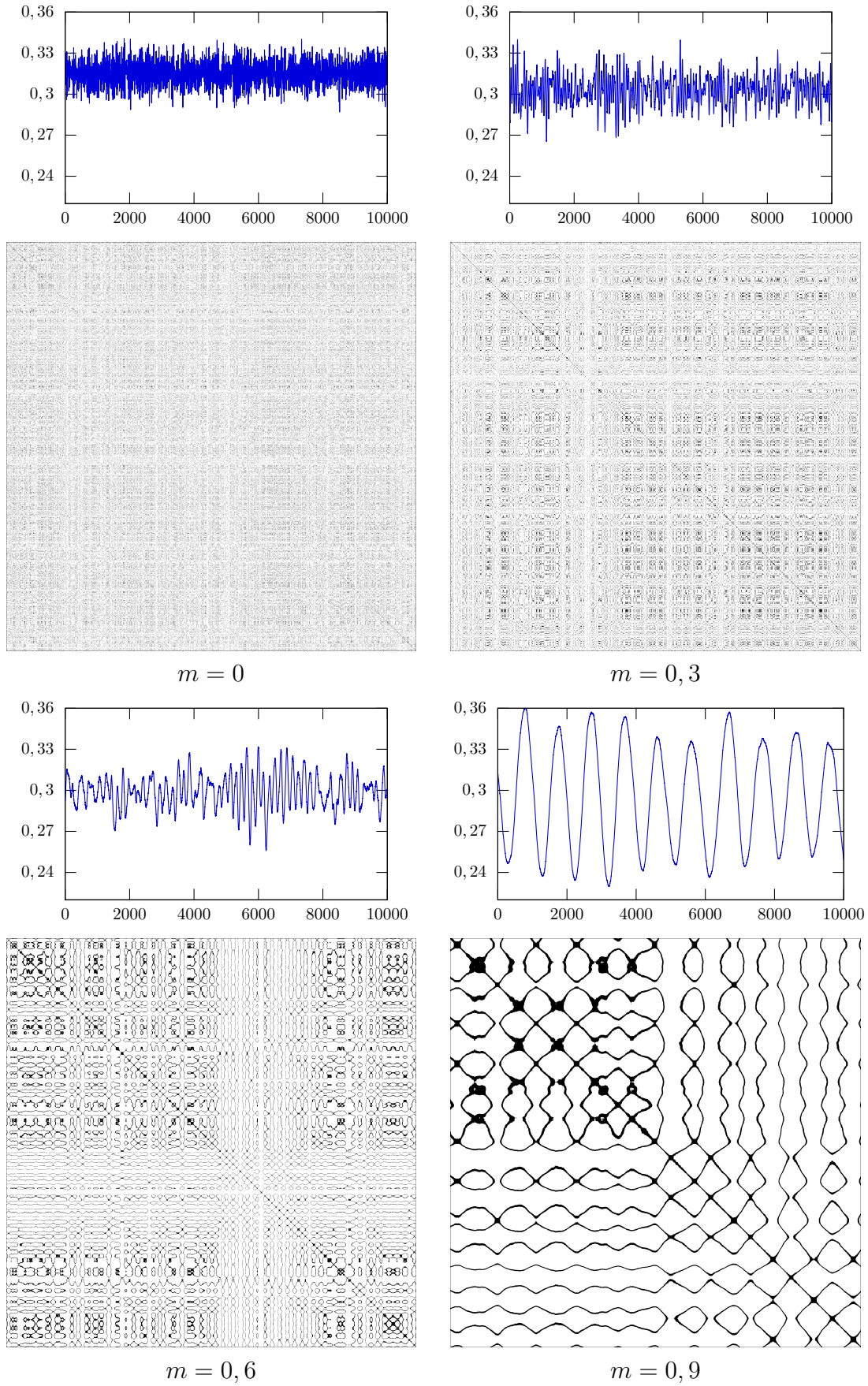


Figura 4.3: Ilustração mostrando como a adição de mobilidade altera a série temporal e o mapa de recorrência.

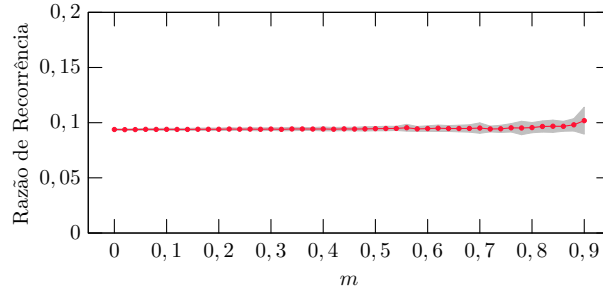


Figura 4.4: Ilustração mostrando a razão de recorrência em função da mobilidade. A área sombreada representa o erro padrão.

Na Figura 4.4 é possível ver que o valor de RR ficou muito próximo de 10%, como desejado. O valor de RR permanece constante durante quase todos os valores de mobilidade, ela apenas oscila quando a mobilidade é muito alta.

4.3.2 Determinismo

O determinismo como vimos no capítulo 2, pode ser interpretado como o grau de previsibilidade do sistema. Sistemas periódicos e correlacionados têm o valor do determinismo próximo de 1, processos descorrelacionados e estocásticos, por sua vez, têm o valor do determinismo próximo de 0 [23–25].

Podemos analisar pela Figura 4.5 que o acréscimo de mobilidade faz com que o sistema passe de um estado pouco correlacionado para um sistema periódico, correlacionado. Quando verificamos as Figuras 4.5 e 4.2, conforme o sistema fica mais periódico e correlacionado, maiores e mais bem formadas são as espirais. Isso vai de encontro com a idéia de que, aumentando a mobilidade estamos gerando uma ordem no sistema.

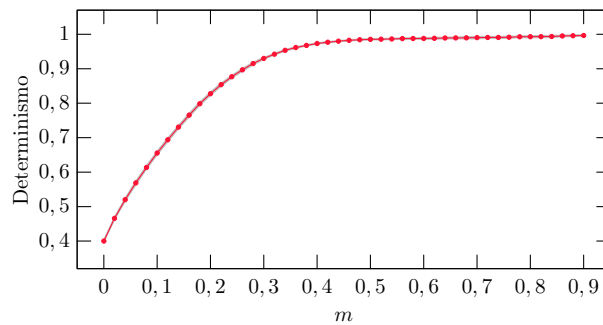


Figura 4.5: Ilustração mostrando o determinismo em função da mobilidade. A área sombreada representa o erro padrão.

4.3.3 Tamanho médio das linhas diagonais

O tamanho médio das linhas diagonais $\langle L \rangle$ está ligado ao tempo médio de previsibilidade do sistema [23, 25, 32].

Ao Analisarmos a Figura 4.6, vemos $\langle L \rangle$ aumentando conforme aumentamos o valor da mobilidade. Ou seja, com mobilidade baixa, o tempo de previsibilidade do sistema é pequeno. Por outro lado, no sistema com mobilidade alta, o tempo de previsibilidade

do sistema é alto quando comparado ao sistema com baixa mobilidade. Nas Figuras 4.2, 4.5 e 4.6, quanto maior é a mobilidade, maior é a correlação e periodicidade do sistema e maior é seu tempo de previsibilidade. Isso novamente reforça a idéia de que, quando aumentamos a mobilidade, estamos gerando alguma ordem no sistema.

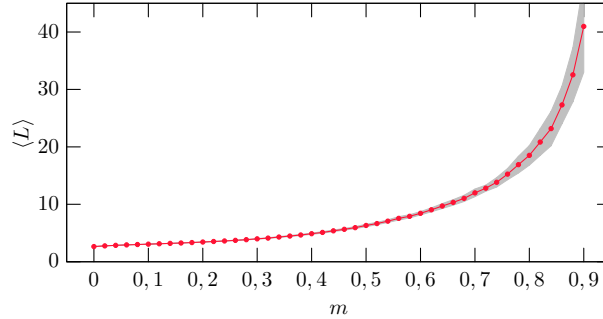


Figura 4.6: Ilustração mostrando o tamanho médio das linhas diagonais em função da mobilidade. A área sombreada representa o erro padrão.

4.3.4 Divergente

Antes de discutirmos sobre o divergente DIV , abrangemos o tamanho da maior diagonal L_{MAX} . A Figura 4.7, ilustra o tamanho da maior diagonal L_{MAX} . Quando a mobilidade está baixa, a maior diagonal é pequena. Conforme acrescentamos mobilidade ao sistema, o L_{MAX} aumenta até seu limite de 9999.

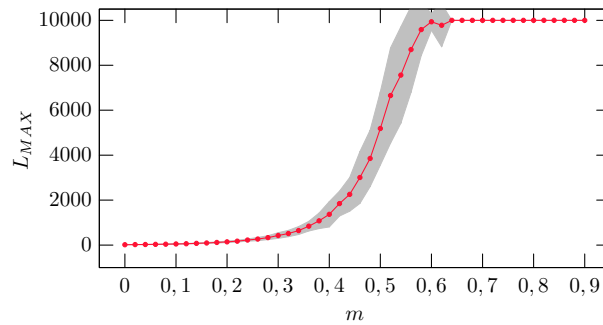


Figura 4.7: Ilustração mostrando o tamanho da maior linha diagonal em função da mobilidade. A área sombreada representa o erro padrão.

O divergente DIV é calculado por meio do inverso do valor da maior linha diagonal do mapa de recorrência. O divergente está relacionado com o maior expoente de Lyapunov no espaço de fase [23, 25, 32].

Na Figura 4.8 podemos notar que, quando aumentamos a mobilidade, diminuimos o divergente. Pensando no expoente de Lyapunov, a distância entre as trajetórias no espaço de fase com mobilidade baixa deve ser maior do que quando comparada ao sistema com mobilidade alta.

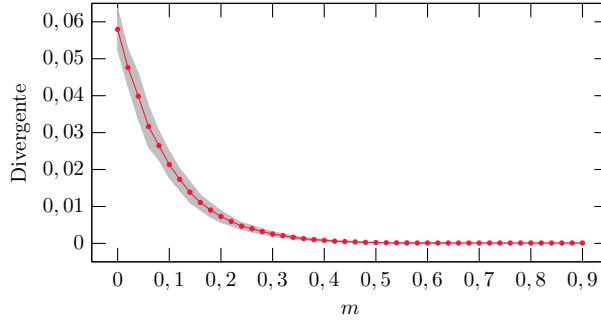


Figura 4.8: Ilustração mostrando o divergente em função da mobilidade. A área sombreada representa o erro padrão.

4.3.5 Entropia de Shannon

A entropia de Shannon é calculada a partir da probabilidade de se encontrar estruturas diagonais no mapa de recorrência e nos informa o quão complexo é o sistema [23, 32, 34].

Ao analisar a Figura 4.9 é possível notar que conforme aumentamos a mobilidade, aumentamos o valor da entropia. Com a mobilidade baixa, o sistema se comporta de maneira menos complexa. Em contrapartida, quando a mobilidade é alta o sistema é mais complexo. Novamente, isso reforça a idéia de que ao acrescentar mobilidade estamos gerando alguma ordem no sistema.

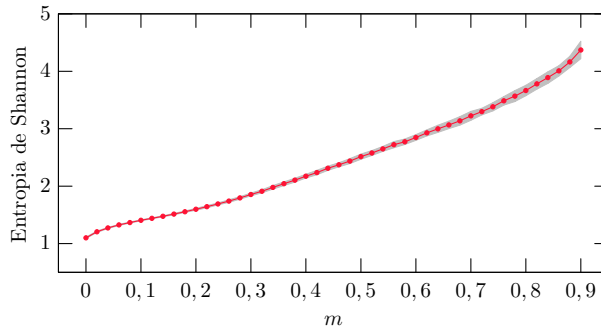


Figura 4.9: Ilustração mostrando a entropia de Shannon em função da mobilidade. A área sombreada representa o erro padrão.

4.3.6 Laminaridade

A laminaridade LAM , é calculada forma análoga ao determinismo DET . É calculada com base nas linhas diagonais no mapa de recorrência. A laminaridade mostra a ocorrência de estados laminares no sistema [23, 25, 32]. Na Figura 4.10 podemos notar que, conforme aumentamos a mobilidade no sistema, aumentamos a ocorrência de estados laminares.

4.3.7 Tempo de Aprisionamento

O tempo de aprisionamento TT como vimos, indica o tempo no qual o sistema se encontra preso em determinado estado. Na Figura 4.11, podemos concluir que, conforme aumentamos a mobilidade, aumentamos o tempo de aprisionamento. Novamente, isso reforça a idéia de que a mobilidade gera alguma ordem ao sistema.

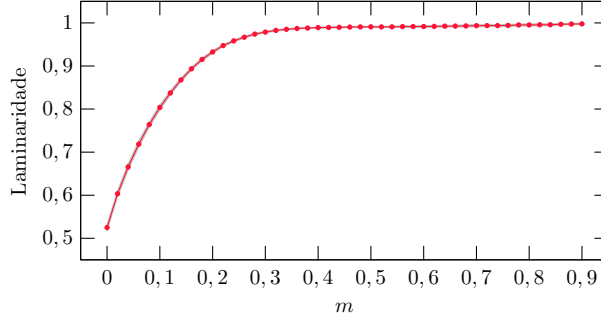


Figura 4.10: Ilustração mostrando a laminaridade em função da mobilidade. A área sombreada representa o erro padrão.

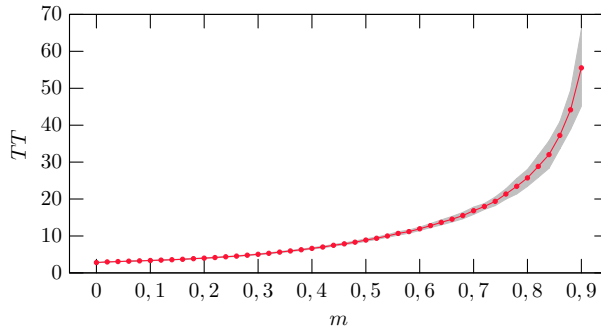


Figura 4.11: Ilustração mostrando o tempo de aprisionamento em função da mobilidade. A área sombreada representa o erro padrão.

4.4 Função de autocorrelação

4.4.1 Função de autocorrelação temporal

A Figura 4.12 mostra como a mobilidade interfere na função de autocorrelação temporal. Nessa Figura 4.12 verificamos que, quanto maior é a mobilidade, maior é a correlação de uma geração sobre a outra [4, 39, 40]. Em cada gráfico da Figura 4.12, colocamos uma seta em $C(t) = 0,5$ que indica o tempo característico. Esse tempo representado por τ , é calculado medindo a distancia no eixo x , entre a origem e $C(t) = 0,5$. Pela Figura 4.12 podemos concluir que, conforme acrescentamos mobilidade ao sistema, aumentamos o tempo característico.

O tempo característico e o tempo de aprisionamento possuem um significado semelhante. Para tal relação, variamos o valor de ϵ até que o tempo de aprisionamento coincidisse com o comprimento característico. Para que o tempo de aprisionamentos e o comprimento característico coincidam. Para isso utilizamos $\epsilon = \sigma/2, 2$.

A Figura 4.13 mostra como o tempo característico e o tempo de aprisionamento coincidem. Essa relação somente acontece quando a diferença $x_i - x_j$ na série temporal apresenta uma distribuição gaussiana, como ilustra a Figura 4.14. Estamos estudando o motivo pelo qual o tempo característico coincide com o tempo de aprisionamento quando $\epsilon = \sigma/2, 2$.

4.4.2 Função de autocorrelação espacial

Calculamos a função de autocorrelação espacial e, para tal, nos brasemos na rede.

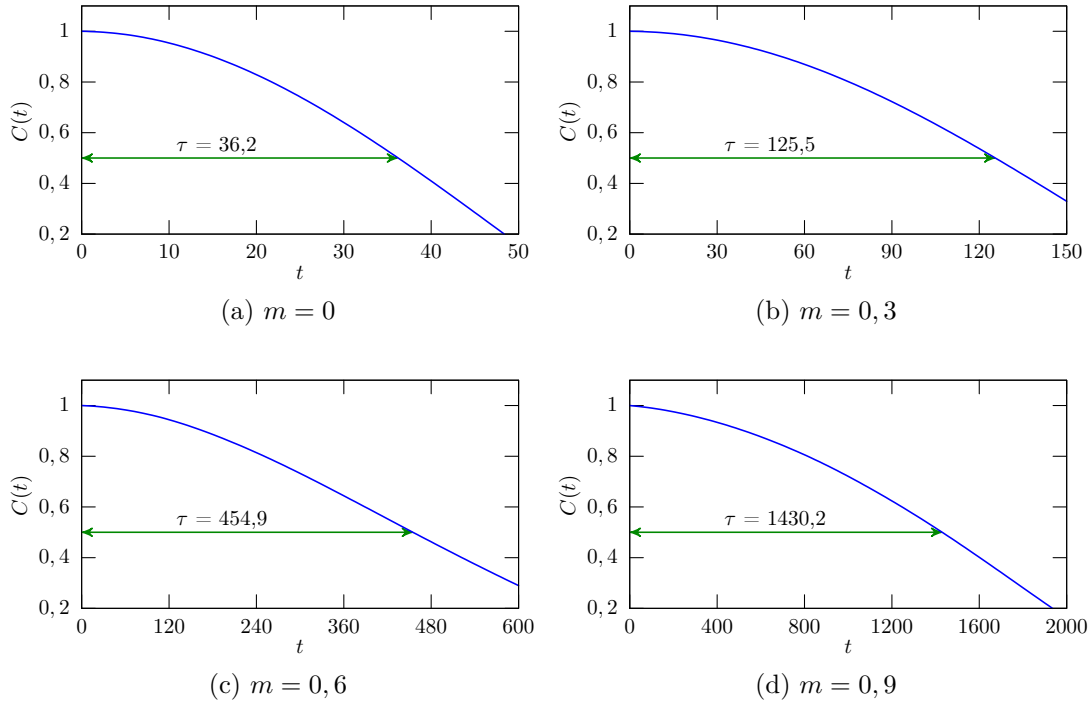


Figura 4.12: Ilustração mostrando como a mobilidade interfere na função de autocorrelação.

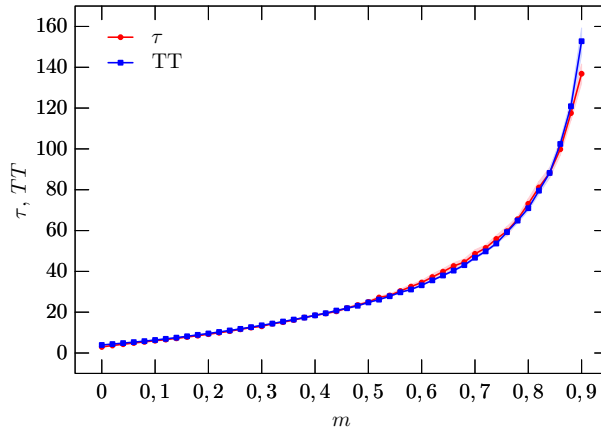


Figura 4.13: Ilustração mostrando como o tempo de aprisionamento e o tempo característico são semelhantes.

Na Figura 4.15 vemos que, no início a função de autocorrelação tem uma queda muito brusca independente do valor da mobilidade. Essa queda ocorre devido a mudanças abruptas nos valores dos campos em cada ponto da rede.

O comportamento da função de autocorrelação espacial é análogo ao da função de autocorrelação temporal. Conforme aumentamos a mobilidade, vemos que a função de autocorrelação decresce de maneira mais suave. A correlação de cada sítio com sua vizinhança é cada vez maior.

Assim como na autocorrelação temporal, também medimos o comprimento característico ℓ para cada mobilidade. Como temos um salto no início do gráfico, decidimos manter o comprimento característico próximo a $C(r) = 0,15$. A Figura 4.16 ilustra como

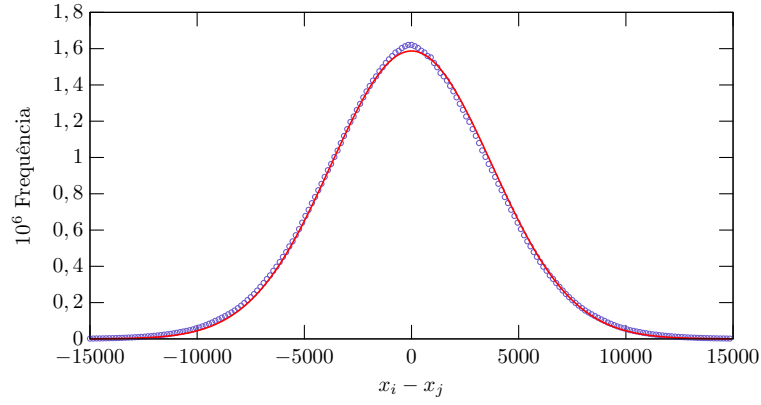


Figura 4.14: Ilustração mostrando a distribuição $x_i - x_j$ formando uma curva Gaussiana. Em pontilhado está as frequências de $x_i - x_j$ e a linha vermelha é o ajuste da gaussiana

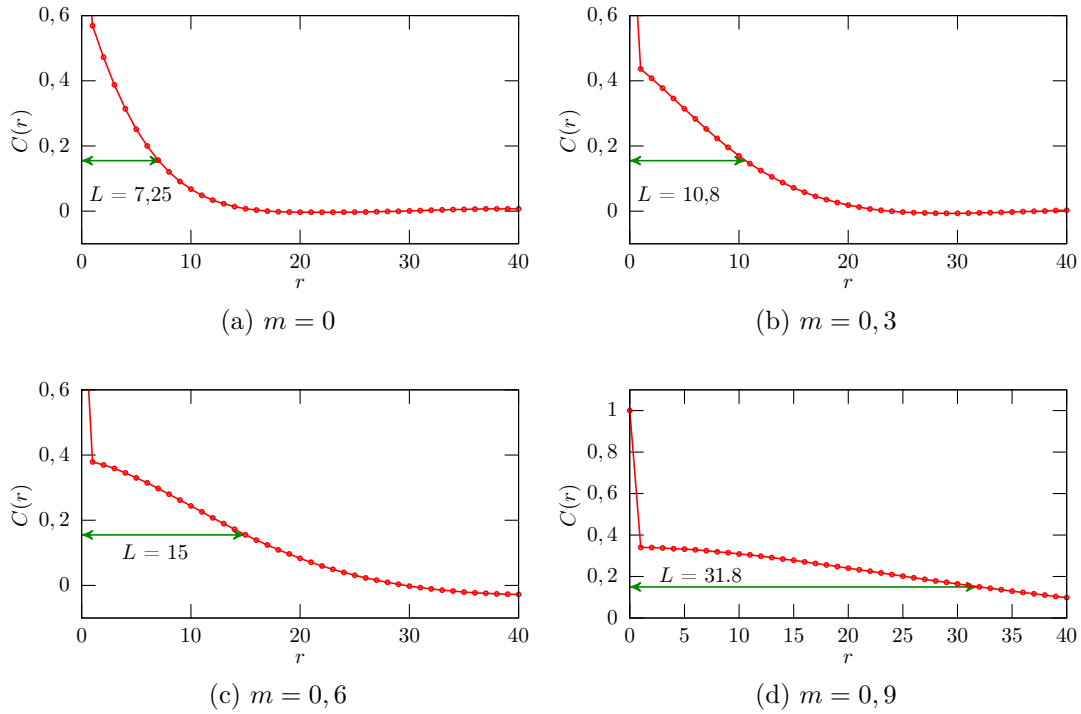


Figura 4.15: Ilustração mostrando como a mobilidade interfere na função de autocorrelação.

o comprimento característico varia de acordo com a mobilidade do sistema. No início, com mobilidade baixa, temos um comprimento característico pequeno, mas à medida que aumentamos a mobilidade do sistema, seu comprimento característico aumenta. Na função de autocorrelação temporal e na espacial o acréscimo de mobilidade teve um efeito muito semelhante.

4.5 Espaço de Fase

Tratamos da trajetória da densidade populacional no espaço de fase. Essa trajetória no espaço de fase ilustra a evolução temporal do sistema por meio da evolução temporal

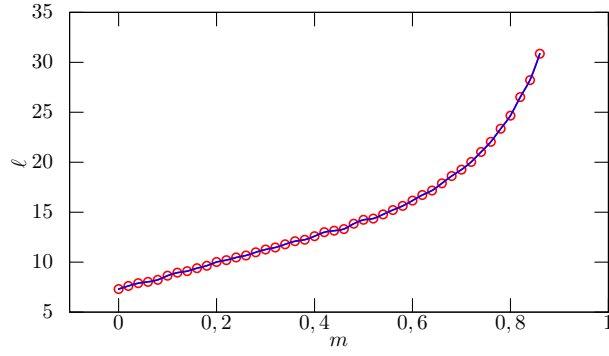


Figura 4.16: Ilustração mostrando o comprimento característico em função da mobilidade.

de suas variáveis relevantes.

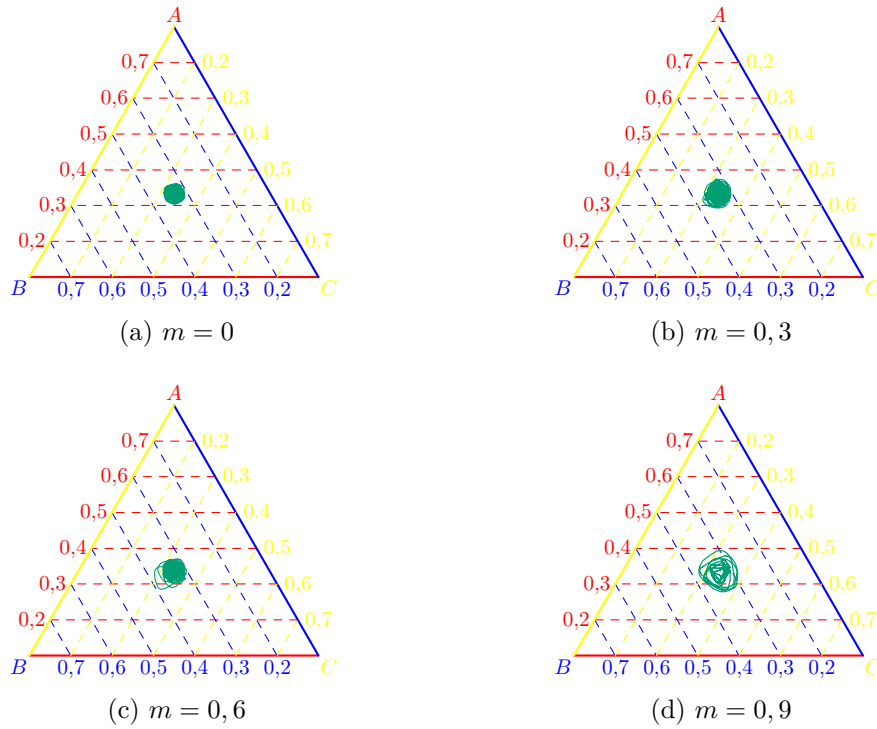


Figura 4.17: Ilustração mostrando o espaço de fase (já normalizado) do sistema RPS. Cada cor do gráfico representa uma espécie. Cada figura mostra como a mobilidade interfere em cada espaço de fase.

Na Figura 4.17 vemos que, quando temos a mobilidade baixa, a trajetória da densidade oscila próxima ao centro da gráfico.

Conforme aumentamos a mobilidade, vemos a trajetória da densidade se afastar do centro da gráfico e se aproximar das bordas do gráfico. Se continuarmos aumentando a mobilidade, haverá um momento em que a trajetória do espaço de fase encostará em uma das bordas. Quando isso ocorrer, teremos a extinção de uma espécie. Consideramos como mobilidade crítica o valor de mobilidade máximo em que não ocorra a extinção de uma espécie. Acima da mobilidade crítica sempre haverá extinção.

4.6 Mobilidade real e mobilidade aparente

Em nossas simulações, utilizamos a equação (4.1) para escolhermos a probabilidade de que ocorra predação, reprodução e mobilidade. Por exemplo: o sistema RPS com $r = p = 0,2$ e $m = 0,6$. Teoricamente haveria 60% de ações com mobilidade, 20% de ações com predação e 20% com reprodução, mas na prática, isso não acontece. Na Figura 4.18 comparamos o número de ações de mobilidade realizadas pelo número de predações realizadas MoP_R e comparamos com o peso estatístico da mobilidade pelo peso estatístico da predação MoP_E . Pelo que se pode observar, MoP_R é maior que MoP_E .

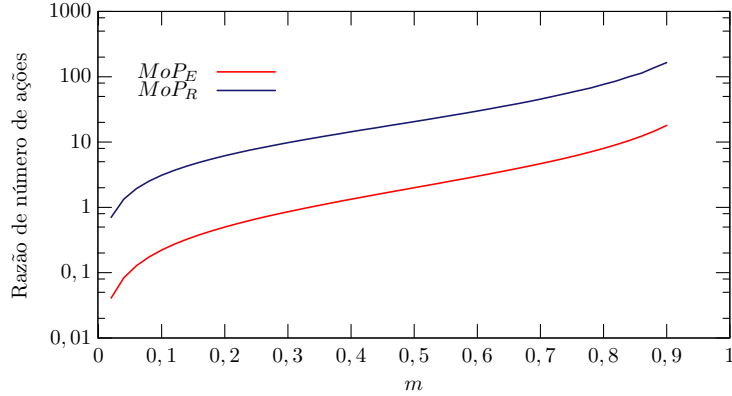


Figura 4.18: Figura ilustrando número efetivo e estatístico de mobilidades por predação

Para entender essa divergência, analisaremos a Figura 4.2. As ações de predação e reprodução ocorrem nas interfaces, pois somente ali há o encontro de predador e presa e espaços vazios. Em contrapartida, a mobilidade ocorre em quase toda a rede, na maioria dos casos haverá mais ações de mobilidade do que predações e reproduções. No entanto, ainda resta uma questão: “Por que o número de mobilidade efetiva cresce tanto com mobilidade alta?” com pouca mobilidade, temos ilhas isoladas de espécies ou espirais pequenas, o número de interfaces é grande e favorece a predação e reprodução. À medida que acrescentamos mobilidade ao sistema, aumentamos o tamanho das espirais e reduzimos a quantidade de espirais, em outras palavras, reduzimos o número de interfaces. Com menos interfaces, desfavorecemos a predação e reprodução.

Como assinalamos, é intuitivo pensar que a densidade de vazios na rede decresce conforme aumentamos mobilidade. A Figura 4.19 mostra como esse raciocínio é equivocado, pois a densidade de vazios aumenta na rede. A explicação para esse fato é que sem mobilidade cada elemento era fixo em um sítio; se esse elemento não tiver nenhum vizinho predável ele não irá preda nada. O mesmo vale para a reprodução, se o elemento não tiver nenhum vazio, ele não vai se reproduzir. Quando acrescentamos mobilidade ao sistema, damos a possibilidade de cada elemento se movimentar assim criando oportunidade de preda, ser predado e se reproduzir. Essa movimentação é responsável pelas estruturas espirais.

Mas, afinal, é possível relacionar MoP_E com MoP_R ? A resposta é sim.

Na Figura 4.20, temos MoP_R/MoP_E no gráfico di-log, nela podemos observar que a curva decresce como uma lei de potência, conforme aumentamos a mobilidade. Em outras palavras, podemos escrever

$$\frac{MoP_R}{MoP_E} = \alpha m^\beta, \quad (4.4)$$

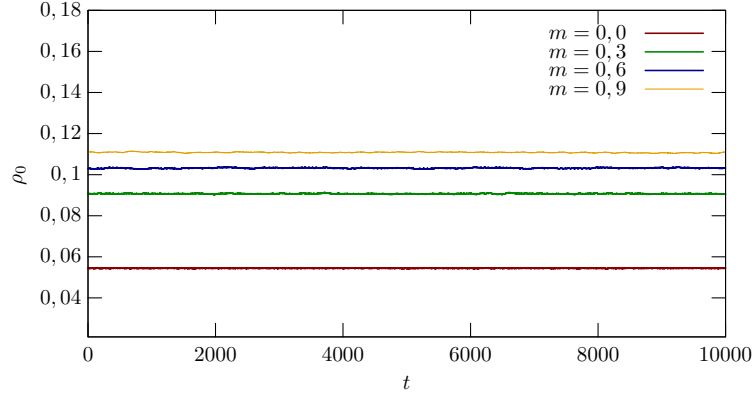


Figura 4.19: Figura ilustrando a densidade de vazios ρ_0 ao longo do tempo para diferentes valores de mobilidade.

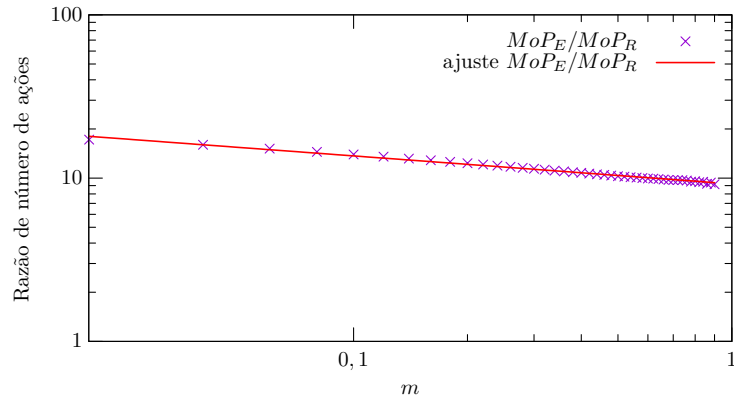


Figura 4.20: Ilustração na escala di-log para MoP_E/MoP_R .

em que α e β são duas constantes. Em nossas simulações usamos a equação (4.1) para definir os valores de predação, mobilidade e reprodução. Além disso, sempre usamos a predação com o mesmo peso da reprodução. Podemos reescrever MoP_E em termos da mobilidade m

$$MoP_E = \frac{2m}{1-m}. \quad (4.5)$$

Ao combinarmos as equações (4.4), (4.5) e realizarmos algumas manipulações algébricas, conseguimos chegar a uma expressão para a MoP_R em função da mobilidade

$$MoP_R = 2\alpha \frac{m^{1+\beta}}{1-m}. \quad (4.6)$$

Ao efetuarmos o ajuste gráfico, conseguimos o valor das constantes $\alpha = 9,21$ e $\beta = -0,17$. A equação (4.6) nos permite relacionar as duas mobilidades. Em outros termos, é possível escolher o quanto de ações de mobilidade efetiva por predação efetiva desejamos no sistema apenas baseando-se na mobilidade estatística. Em nosso estudo decidimos focar em mobilidade por predação, mas nada impede de usarmos mobilidade por reprodução, o comportamento é o mesmo para ambas.

Ao retornarmos a nosso exemplo, com $m = 0,6$ e $p = 0,2$ por meio da equação (4.6) calculamos que $MoP_R \approx 30,13$. Ao realizarmos a comparação entre $MoP_E = 3$ e MoP_R , vemos que MoP_R é quase 10 vezes maior que MoP_E .

4.7 Densidade Espectral

Por último e não menos importante, temos a densidade espectral. A partir da série temporal, calculamos a densidade espectral para diferentes mobilidades.

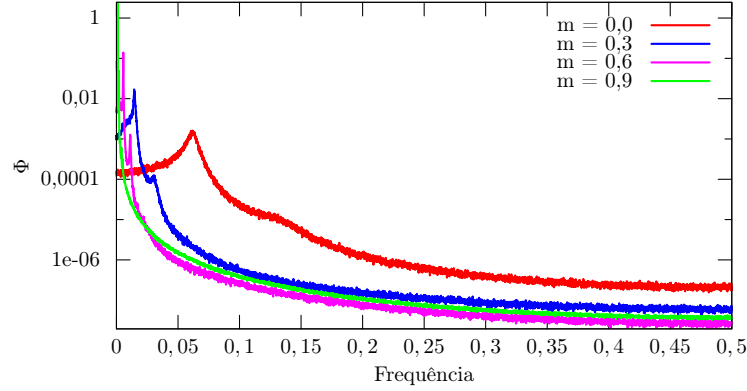


Figura 4.21: O gráfico da densidade espectral pela frequência

A Figura 4.21 mostra como é a densidade espectral para diferentes valores de mobilidade. Para facilitar a visualização do início do gráfico da Figura 4.21 o refizemos com foco na região de frequência de 0 Hz até $0,1$ Hz na Figura 4.22.

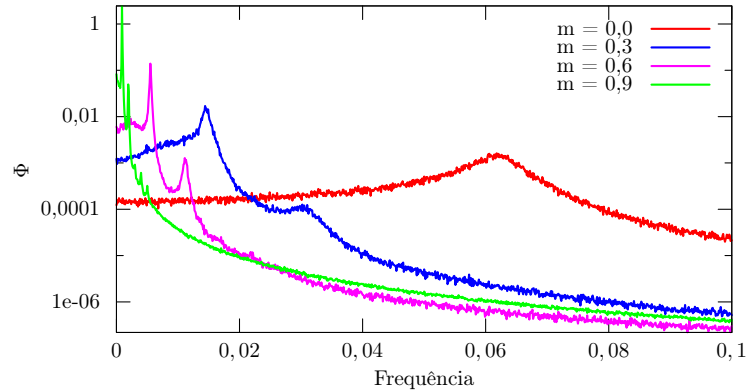


Figura 4.22: Figura com o gráfico da densidade espectral pela frequência no intervalo de 0 Hz até $0,1$ Hz.

É possível observar que, para todos os valores de mobilidade, temos 2 picos. Quando a mobilidade está baixa, esse pico é suave e, conforme aumentamos a mobilidade, tornamos o pico cada vez menos suave e mais brusco. Esses picos representam a frequência característica da série temporal; a localização desses picos depende do valor da mobilidade. À proporção que aumentamos a mobilidade, deslocamos o pico da densidade espectral para mais próximo do início do gráfico. O pico com baixa frequência está relacionado com a mudança da espécie dominante na rede. Por outro lado, o pico de alta frequência está relacionado com o ruído da série temporal [41].

Conclusões

O trabalho da realizado por M. M. T. Reichenbach e E. Frey [?] indica como a mobilidade é um fator crucial para a biodiversidade no modelo RPS. Em nossa pesquisa, estudamos por meio dos mapas de recorrência e da análise de Fourier como a mobilidade influencia no modelo RPS estocástico. Ao todo, empreendemos cerca de 9000 simulações. Nessas simulações partimos do sistema sem nenhuma mobilidade até o sistema com 90% de mobilidade. Obtivemos um resultado semelhante trabalho ao realizado por M. M. T. Reichenbach e E. Frey e demonstramos como a mobilidade pode interferir na rede. No início, com mobilidade baixa, observamos que temos ilhas isoladas de espécies. Conforme aumentamos a mobilidade, surgem estruturas espirais e, quanto mais aumentamos a mobilidade mais bem formadas são essas espirais.

Em cada simulação medimos a densidade de indivíduos de cada espécie por geração e criamos a série temporal. Comparamos séries temporais de diferentes mobilidades e observamos como a mobilidade afeta a série temporal. Por meio da série temporal, verificamos que com a mobilidade baixa a série temporal oscila muito bruscamente em um curto período de tempo. À medida que aumentamos a mobilidade a série temporal começa a oscilar de maneira mais suave. Por meio das séries temporais criamos os mapas de recorrência para todas as mobilidades estudadas. Novamente observamos como a mobilidade interfere no mapa de recorrência. Com base nas linhas diagonais, verticais e densidade de pontos aplicamos as quantificações dos mapas de recorrência. Podemos concluir que, no início, com mobilidade baixa, o sistema é pouco correlacionado (estocástico), conforme aumentamos a mobilidade vemos que o sistema para a ser mais correlacionado e mais complexo.

Calculamos a função de autocorrelação temporal, observamos a influência da mobilidade e medimos seu tempo característico. Verificamos ainda que, manipulando o mapa de recorrência é possível fazer com que o tempo de aprisionamento coincida com o tempo característico. Essa convergência entre o tempo de aprisionamento e o tempo característico somente ocorre quando a distribuição $x_i - x_j$ apresenta a forma de uma curva gaussiana. Medimos também a função de autocorrelação espacial e o comportamento foi análogo ao da autocorrelação temporal. Mensuramos o comprimento característico, notamos que o comportamento é análogo ao tempo característico. Quanto maior é a mobilidade, maior é o comprimento característico.

Analizamos também a trajetória no espaço de fase. Pela trajetória, observamos que conforme aumentamos a mobilidade, mais afastamos a trajetória do centro do gráfico. Predizemos que se aumentássemos ainda mais a mobilidade, haveria a extinção de uma espécie.

Constatamos que o peso estatístico da mobilidade não relata de forma fiel o número de ações na rede. A mobilidade real não é a mesma mobilidade estatística. Por meio da equação (4.7), conseguimos relacionar a mobilidade real com a mobilidade estatística.

$$\frac{MoP_R}{MoP_E} = 9,21m^{-0,17} \quad (4.7)$$

Analisamos também a densidade de vazios na rede e o resultado foi que conforme aumentávamos a mobilidade, a densidade de vazios aumentava. Com base em todas os resultados, pudemos concluir que a mobilidade gera uma ordem no sistema RPS. Conforme aumentávamos a mobilidade, aumentávamos o encontro de diferentes espécies na rede. Esse aumento de número de encontros levava ao surgimento das estruturas espirais.

Calculamos a densidade espectral para diferentes valores de mobilidade. Observamos que independente da mobilidade sempre haverá dois picos na densidade espectral. O pico de baixa frequência que se relaciona com a dominância da espécie na rede e o pico de alta frequência que se relaciona com o ruído.

Entre as perspectivas de trabalhos futuros, pretendemos aumentar o número de espécies nas simulações, utilizar novas regras de predador e presa, incluir possíveis ações extras, adicionar uma mobilidade direcionada, encontrar uma solução analítica que explique a relação entre o tempo de aprisionamento e a função de autocorrelação espacial, investigar mais sobre a trajetória no espaço de fase e utilizar um modelo sem rede.

Apêndice A

Códigos

Seguem os códigos dos programas, escritos em linguagem C. Os mesmos foram utilizados para gerar os resultados apresentados neste trabalho. Para compilar os códigos é preciso o uso das bibliotecas *math*, *FFTW*, e *Gnu scientific Library* .

Autocorrelação temporal

```
1  /*Bibliotecas*/
2  #include <stdio.h>
3  #include <stdlib.h>
4  #include <math.h>
5  #include <gsl/gsl_statistics.h>
6
7  /*Início do problema*/
8
9  int main(int argc, char **argv){
10
11     /*Declaração de variáveis*/
12     int i, j, k, N;
13     double a, b, c;
14     double mean, sd, div;
15     double *inp, *acf;
16     FILE *file;
17     char name[100];
18
19     /*Número de linhas do arquivo*/
20     N= atof(argv[1]);
21
22     /*Alocação dinâmica de memória*/
23     inp= (double *) calloc(N, sizeof(double));
24     acf= (double *) calloc(N, sizeof(double));
25
26     /*Abertura do arquivo de entrada*/
27     file= fopen(argv[2], "r");
28     /*Leitura dos arquivos de entrada*/
29     for(i= 0; i< N; i++){
```

```

30     fscanf(file, "%lf %lf %lf %lf\n", &inp[i], &a, &b, &c);
31 }
32 fclose(file);
33
34 /*Cálculo da função de Autocorrelação*/
35 mean= gsl_stats_mean(inp, 1, N);
36 for(i= 0; i<N; i++){
37     c[0]+=(inp-mean)-(inp[i+j]-mean);
38 }
39 div= c[0];
40
41 sd= gsl_stats_sd(inp, 1, N);
42 for(j= 0; j< N; j++){
43     for(i= 0; i< (N-j); i++){
44         acf[j]+= (inp[i]-mean)*(inp[i+j]-mean);
45     }
46     acf[j]/= div;
47 }
48
49 /*Abertura do arquivo de saída*/
50 file= fopen(argv[3], "w");
51
52 /*Escrita dos arquivos de saída*/
53 for(i= 0; i< N; i++){
54     fprintf(file, "%e\n", acf[i]);
55 }
56 fclose(file);
57
58 /*Impressão do tempo característico */
59 i=0;
60 while(acf[i] > 0.5){
61     i++;
62 }
63 if((acf[i-1]-0.5) < (0.5-acf[i])){
64     printf("%d\n", i-1);
65 }else{
66     printf("%d\n", i );
67 }
68
69 /*Liberação de memória*/
70 free(acf);
71 free(inp);
72
73 /*fim do programa*/
74 return 0;
75 }

```

Mapa de recorrência e suas quantificações

```
1  /*Bibliotecas*/
2  #include <stdio.h>
3  #include <stdlib.h>
4  #include <math.h>
5  #include <gsl/gsl_statistics.h>
6
7  /*início do programa*/
8  int main(int argc, char **argv){
9
10     /*Alocação de variáveis*/
11     int i, j, k, k_u, k_d, l, N, w;
12     int l_min, v_min;
13     double rr, sl_min, sv_min, sv, sl, sL1, sL2, sV1, sV2;
14     double RR, DET, LAM, RATIO, L, TT, DIV;
15     double v_entr, l_entr;
16     long long int *rp, *v, *d;
17     double e;
18     double *p;
19     FILE *file;
20     N= atoi(argv[2]);
21     l_min = atoi(argv[3]);
22     v_min = atoi(argv[4]);
23     w= 1;
24
25     /*Alocação dinâmica de memória*/
26     rp= (long long int *) calloc(N*N, sizeof(long long int));
27     v = (long long int *) calloc(N+1, sizeof(long long int));
28     d = (long long int *) calloc(N+1, sizeof(long long int));
29     p = (double *)          calloc(N,   sizeof(double));
30
31     /*Abertura e leitura do arquivo de entrada*/
32     file= fopen(argv[1], "r");
33     for(i= 0; i< N; i++){
34         fscanf(file, "%lf\n", &p[i]);
35     }
36     fclose(file);
37
38     /*Abertura, cálculo e impressão do mapa de recorrência*/
39     file= fopen("rp.dat", "w");
40     e= gsl_stats_sd(p, 1, N)/6.0;
41     for(i= 0; i< N; i++){
42         for(j= 0; j< N; j++){
43             if(fabs(p[i]-p[j])< e){
44                 fprintf(file, "%d ", 1);
45                 rp[i*N+j]= 1;
46             }else{
```

```

47         fprintf(file, "%d ", 0);
48         rp[i*N+j]= 0;
49     }
50 }
51 fprintf(file, "\n");
52 }
53
54 if(w == 1){
55     for(i= 0; i<N; i++){
56         rp[i*N+i]= 0;
57     }
58 }
59 fclose(file);
60
61 /*Contagem do número de linhas diagonais*/
62 for(l= 0; l< N; l++){
63     k_u= 0;
64     k_d= 0;
65     for(i= 0, j= l; j< N; i++, j++){
66         if(rp[i*N+j] == 1){
67             k_u++;
68         }else{
69             d[k_u]++;
70             k_u= 0;
71         }
72         if(i!=j){
73             if(rp[j*N+i] == 1){
74                 k_d++;
75             }else{
76                 d[k_d]++;
77                 k_d= 0;
78             }
79         }
80     }
81     d[k_u]++;
82     d[k_d]++;
83 }
84
85 /*Contagem do número de linhas verticais*/
86 for(j= 0; j< N; j++){
87     k= 0;
88     for(i= 0; i< N; i++){
89         if(i!=j){
90             if(rp[i*N+j] == 1){
91                 k++;
92             }else{
93                 v[k]++;
94                 k= 0;

```

```

95     }
96     }
97 }
98     v[k]++;
99 }
100
101 /*Contagem da maior linha vertical*/
102 int V_MAX;
103 for(i= 0; i<= N; i++){
104     if(v[i] != 0){V_MAX= i;}
105 }
106
107 /*Contagem da maior linha diagonal*/
108 int L_MAX;
109 for(i= 0; i<= N; i++){
110     if(d[i] != 0){L_MAX= i;}
111 }
112
113 /*Cálculo do recurrence rate*/
114 rr= 0.0;
115 for(i= 0; i< N; i++){
116     for(j= 0; j< N; j++){
117         if(rp[i*N+j] == 1){
118             rr+= 1.0;
119         }
120     }
121 }
122 RR= rr/(double)(N*N);
123
124 /*Cálculo do determinismo */
125 sl_min= 0.0;
126 for(i= l_min; i<= N; i++){
127     sl_min+= i*d[i];
128 }
129 DET= sl_min/rr;
130
131 /*Cálculo da Laminaridade */
132 sv_min= 0.0;
133 for(i= v_min; i< N; i++){
134     sv_min+= i*v[i];
135 }
136
137 sv= 0.0;
138 for(i= 1; i< N; i++){
139     sv+= i*v[i];
140 }
141 LAM= sv_min/sv;
142

```



```

143  /*Cálculo do ratio */
144  RATIO= DET/RR;
145
146  /*Cálculo do tamanho médio das linhas diagonais */
147  sL1= 0.0;
148  sL2= 0.0;
149  for(i= l_min; i<= N; i++){
150      sL1+= d[i]*i;
151      sL2+= d[i];
152  }
153  L= sL1/sL2;
154
155  /*Cálculo do Trapping time */
156  sV1= 0.0;
157  sV2= 0.0;
158  for(i= v_min; i< N; i++){
159      sV1+= v[i]*i;
160      sV2+= v[i];
161  }
162  TT=sV1/sV2;
163
164  /*Cálculo do Divergente */
165  DIV= 1.0/L_MAX;
166
167  /*Cálculo da entropia através das linhas diagonais*/
168  v_entr= 0.0;
169  for(i= v_min; i< N; i++){
170      if(v[i] > 0){
171          v_entr+=(double)(v[i])*(log(v[i])-log(sV2));
172      }
173  }
174
175  /*Cálculo da entropia através das linhas verticais*/
176  l_entr=0;
177  for(i= l_min; i<N; i++){
178      if(d[i] != 0){
179          l_entr+=(double)(d[i])*(log(d[i])-log(sL2));
180      }
181  }
182  /*Abertura e impressão das quantificação do mapa de recorrência*/
183  FILE *out;
184  out = fopen("dat/data.dat", "w");
185  fprintf(out,"%e ", RR);
186  fprintf(out,"%e ", DET);
187  fprintf(out,"%e ", RATIO);
188  fprintf(out,"%d ", L_MAX);
189  fprintf(out,"%e ", L);
190  fprintf(out,"%e ", l_entr/sL2);

```

```

191     fprintf(out,"%e ", DIV);
192     fprintf(out,"%e ", LAM);
193     fprintf(out,"%e ", LAM/DET);
194     fprintf(out,"%d ", V_MAX);
195     fprintf(out,"%e ", TT);
196     fprintf(out,"%e\n", v_entr/sV2);
197     fclose(out);
198
199     /*Fim do programa*/
200     return 0;
201 }

```

R.P.S. e função de autocorrelação espacial

```

1  #include "../rps.h"
2
3  /*definições*/
4  #define FL (int)(NG/NF)
5  #define q pow(NG, 0.01)
6
7  /*funções*/
8  void ic(const gsl_rng *w, int *); /* inicio a rede */
9  void cr(int , int , int *, double *); /* correlação espacial */
10 void op(int , int *); /* saída de dados */
11 void wf(int , int *); /* função saída */
12
13 /* início do programa*/
14
15 int main(int argc, char **argv){
16
17     /*chamada de variáveis*/
18     int i, j, k, n, t, ac, ps, tp, gd, nb;
19     int *phi, *dst;
20     double at, a0= 1.0;
21     double *c_r;
22     PROBABILITY_OF_PREDATION
23     while((a0*q-a0) < 10.0){
24         a0*= q;
25     }
26
27     /*Alocação de memória*/
28     phi= (int *) calloc(NxNyNz, sizeof(int ));
29     dst= (int *) calloc((NS+1), sizeof(int ));
30     c_r= (double *) calloc((100*(Nx+Ny+Nz-2)), sizeof(double));
31
32     /*gerador de número aleatório*/
33     gsl_rng_default_seed= (argc == 2) ? atoi(argv[1]) : time(NULL);
34     gsl_rng *w= gsl_rng_alloc(gsl_rng_taus);

```

```

35
36  /*início da rede e impressão da rede no tempo $t=0$*/
37  ic(w, phi);
38  op(0, phi);
39
40  /*primeira contagem de indivíduos da rede*/
41  FILE *file= fopen("dat/dst.dat", "w");
42  for(i= 0; i< NxNyNz; i++){
43      dst[phi[i]]++;
44  }
45
46
47  for(i= 1; i< (NS+1); i++){
48      fprintf(file, "%d ", dst[i]);
49  }
50  fprintf(file, "%d\n", dst[0]);
51
52  t= n= 0;
53
54  /*laço para as gerações*/
55  while(t < NG){
56      gd= 0;
57      /*número de sorteios por geração*/
58      while(gd < NxNyNz){
59
60          /*Sorteio do ativo*/
61  #if Nz == 1
62          i= gsl_rng_uniform(w)*Nx;
63          j= gsl_rng_uniform(w)*Ny;
64          ac= j*Nx+i;
65  #else
66          i= gsl_rng_uniform(w)*Nx;
67          j= gsl_rng_uniform(w)*Ny;
68          k= gsl_rng_uniform(w)*Nz;
69          ac= (k*Ny+j)*Nx+i;
70  #endif
71          if(phi[ac] != 0){
72
73              /*sorteio do passivo*/
74  #if Nz == 1
75              nb= gsl_rng_uniform(w)*4;
76              switch(nb){
77                  case 0:
78                      ps= j*Nx+((i+1)%Nx);
79                      break;
80                  case 1:
81                      ps= j*Nx+((i-1+Nx)%Nx);
82                      break;

```

```

83         case 2:
84             ps= ((j+1)%Ny)*Nx+i;
85             break;
86         default:
87             ps= ((j-1+Ny)%Ny)*Nx+i;
88     }
89 #else
90     nb= gsl_rng_uniform(w)*6;
91     switch(nb){
92         case 0:
93             ps= (k*Ny+j)*Nx+((i+1)%Nx);
94             break;
95         case 1:
96             ps= (k*Ny+j)*Nx+((i-1+Nx)%Nx);
97             break;
98         case 2:
99             ps= ((k*Ny)+(j+1)%Ny)*Nx+i;
100            break;
101         case 3:
102             ps= ((k*Ny)+(j-1+Ny)%Ny)*Nx+i;
103            break;
104         case 4:
105             ps= (((k+1)%Nz)*Ny+j)*Nx+i;
106            break;
107         default:
108             ps= (((k-1+Nz)%Nz)*Ny+j)*Nx+i;
109     }
110 #endif
111     /*ação, mobilidade, predação e reprodução*/
112     at= gsl_rng_uniform(w);
113     if(at < pm){
114         tp= phi[ps];
115         phi[ps]= phi[ac];
116         phi[ac]= tp;
117         gd++;
118     }else{
119         if(at >= pm && at < (pm+ pp)){
120             if(phi[ps] != 0 && p[phi[ac]-1][phi[ps]-1] == 1){
121                 dst[phi[ps]]--;
122                 dst[0]++;
123                 phi[ps]= 0;
124                 gd++;
125             }
126         }else{
127             if(phi[ps] == 0){
128                 phi[ps]= phi[ac];
129                 dst[phi[ps]]++;
130                 dst[0]--;

```

```

131         gd++;
132     }
133 }
134 }
135 }
136 }
137 t++;
138
139 /*atualização de indivíduos da espécie i e do vaziao*/
140 for(i= 1; i< (NS+1); i++){
141     fprintf(file, "%d ", dst[i]);
142 }
143 fprintf(file, "%d\n", dst[0]);
144
145 /*chamada da função de autocorrelação*/
146 if(t >= (int)a0){
147     a0*= q;
148     cr(n++, t, phi, c_r);
149 }
150
151 /*impressão de arquivos da rede e função janela*/
152 if((t%FL) == 0){
153     op(t/FL, phi);
154     wf(t/FL, phi);
155 }
156 }
157 fclose(file);
158 /*normalização da função de autocorrelação e impressão*/
159 file= fopen("dat/c_r.dat", "w");
160 for(j= 0; j< (Nx+Ny+Nz-2); j++){
161     for(i= 0; i< n; i++){
162         fprintf(file, "%+e ", c_r[j*100+i]);
163     }
164     fprintf(file, "\n");
165 }
166 fclose(file);
167
168 /*liberação da memória*/
169 gsl_rng_free(w);
170 free(phi);
171 free(dst);
172 free(c_r);
173
174 /* fim do programa*/
175 return 0;
176 }

```

```

1 #include "../rps.h"
2

```

```

3 void ic(const gsl_rng *w, int *phi){
4     int i;
5     /*sorteio da espécie para o sítio*/
6     for(i= 0; i< NxNyNz; i++){
7         phi[i]= gsl_rng_uniform(w)*(NS+1);
8     }
9 }

1 #include "../rps.h"
2
3 void op(int NI, int *phi){
4     int i, j, k, l;
5     char name[100];
6     /*abertura e impressão de arquivos de saída*/
7     sprintf(name, "dat/phi-%d.dat", NI);
8     FILE *file= fopen(name, "w");
9
10    #if Nz == 1
11        for(j= 0; j< Ny; j++){
12            for(i= 0; i< Nx; i++){
13                fprintf(file, "%d ", phi[j*Nx+i]);
14            }
15            fprintf(file, "\n");
16        }
17    #else
18        l= 0;
19        for(i= 0; i< Nx; i++){
20            for(j= 0; j< Ny; j++){
21                for(k= 0; k< Nz; k++){
22                    fprintf(file, "%d\n", phi[l]);
23                    l++;
24                }
25            }
26        }
27    #endif
28    fclose(file);
29 }

1 #include "../rps.h"
2
3 void cr(int n, int t, int *phi, double *c_r){
4     int i, j, k, N;
5     int *nor;
6     double *i2d;
7
8     /*alocação de memória*/
9     fftw_complex *inp= (fftw_complex *) fftw_malloc(sizeof(fftw_complex)*(NxNyNz));
10    fftw_complex *out= (fftw_complex *) fftw_malloc(sizeof(fftw_complex)*(NxNyNz));
11

```

```

12  nor= (int      *) calloc((Nx+Ny+Nz-2), sizeof(int  ));
13  i2d= (double *) calloc(NxNyNz      , sizeof(double));
14
15
16  for(N= 0; N< NxNyNz; N++){
17      i2d[N]= phi[N];
18  }
19
20  double mu= gsl_stats_mean(i2d, 1, NxNyNz);
21  double sd= gsl_stats_sd  (i2d, 1, NxNyNz);
22
23  for(N= 0; N< NxNyNz; N++){
24      i2d[N]-= mu;
25      inp[N]= i2d[N] + 0.0*I;
26  }
27  free(i2d);
28
29  /*Transformada de Fourier*/
30  fftw_plan FTF= fftw_plan_dft_3d(Nz, Ny, Nx, inp, out, FFTW_FORWARD, FFTW_ESTIMATE);
31  fftw_execute(FTF);
32
33  /*Autocorrelação*/
34  double sum= 0.0;
35  double k1= 0.0;
36  double k2= 0.0;
37  double kx, ky, kz;
38  N= 0;
39  for(k= 0; k< Nz; k++){
40      if(k< (Nz/2+1)){kz= (double)k/(double)Nz;}
41      else{kz= (double)(k-Nz)/(double)Nz;}
42      for(j= 0; j< Ny; j++){
43          if(j< (Ny/2+1)){ky= (double)j/(double)Ny;}
44          else{ky= (double)(j-Ny)/(double)Ny;}
45          for(i= 0; i< Nx; i++){
46              if(i< (Nx/2+1)){kx= (double)i/(double)Nx;}
47              else{kx= (double)(i-Nx)/(double)Nx;}
48              out[N]= out[N]*conj(out[N]);
49              sum+= out[N];
50              k1+= sqrt(kz*kz+ky*ky+kx*kx)*out[N];
51              k2+= (kz*kz+ky*ky+kx*kx)*out[N];
52              N++;
53          }
54      }
55  }
56
57  FILE *file= fopen("dat/l_k.dat", "a");
58  fprintf(file, "%7d %+e %+e\n", t, sum/k1, 1.0/sqrt(k2/sum - (k1*k1)/(sum*sum)));
59  fclose(file);

```

```

60
61  /* Transformada inversa*/
62  fftw_plan FTB= fftw_plan_dft_3d(Nz, Ny, Nx, out, inp, FFTW_BACKWARD, FFTW_ESTIMATE)
63  fftw_execute(FTB);
64  for(k= 0; k< Nz; k++){
65      for(j= 0; j< Ny; j++){
66          for(i= 0; i< Nx; i++){
67              N= i+j+k;
68              c_r[N*100+n] += creal(inp[(k*Ny+j)*Nx+i])/(Nx*Ny*Nz*sd*sd*Nx*Ny*Nz);
69              nor[N]++;
70          }
71      }
72  }
73  for(N= 0; N< (Nx+Ny+Nz-2); N++){
74      c_r[N*100+n]/= nor[N];
75  }
76  free(nor);
77  fftw_destroy_plan(FTF);
78  fftw_destroy_plan(FTB);
79  fftw_free(inp);
80  fftw_free(out);
81  fftw_cleanup();
82  }

```


Referências Bibliográficas

- [1] B. Sinervo and C. M. Lively, “The rock-paper-scissors game and the evolution of alternative male strategies,” *Nature*, vol. 380, p. 240, Março 21 1996.
- [2] M. W. F. B. Kerr, M. A. Riley and B. J. M. Bohannan, “Local dispersal promotes biodiversity in a real-life game of rock-paper-scissors,” *Nature*, vol. 418, p. 171, (2002).
- [3] M. M. T. Reichenbach and E. Frey, “Mobility promotes and jeopardizes biodiversity in rock-paper-scissors games,” *Nature*, vol. 448, p. 1046, (2007).
- [4] C. Scherer, *Métodos Computacionais da Física*. Editora Livraria da Física, (2005).
- [5] J. Walker, *Fast Fourier transforms*. Studies in Advanced Mathematics, CRC Press, 1991.
- [6] P. P. Avelino, D. Bazeia, L. Losano, J. Menezes, and B. F. Oliveira, “Junctions and spiral patterns in generalized rock-paper-scissors models,” *Phys. Rev. E*, vol. 86, p. 036112, 2012.
- [7] M. Mobilia, “Oscillatory dynamics in rock-paper-scissors games with mutations,” *Journal of Theoretical Biology*, vol. 264, pp. 1–10, março 2010.
- [8] “Rock paper scissors,” 2015. Disponível em:<http://www.usa.dottinghill.com/_uploads/skin_accessories/1192/200x150thumb.gif.jpg> Acesso em 11 Out. 2017.
- [9] A. Roman, D. Dasgupta, and M. Pleimling, “Interplay between partnership formation and competition in generalized may-leonard games,” *Phys. Rev. E*, vol. 87, p. 032148, Mar 2013.
- [10] “Rock paper scissors lizard spock,” 2015. Disponível em:<<http://docmadhattan.fieldofscience.com/2015/02/rock-paper-scissors-lizard-spock.htm>> Acesso em 11 Out. 2017.
- [11] Wikipédia, “Tabuleiro (xadrez) — wikipédia, a enciclopédia livre,” 2017. Disponível em:<[https://pt.wikipedia.org/wiki/Tabuleiro_\(xadrez\)#/media/File:Ausgeschnitten.jpg](https://pt.wikipedia.org/wiki/Tabuleiro_(xadrez)#/media/File:Ausgeschnitten.jpg)>i Acesso em 15 Abr. 2017.
- [12] P. Avelino, D. Bazeia, J. Menezes, and B. de Oliveira, “String networks in lotka–volterra competition models,” *Physics Letters A*, vol. 378, no. 4, pp. 393 – 397, 2014.
- [13] P. Avelino, D. Bazeia, L. Losano, J. Menezes, and B. de Oliveira, “String networks with junctions in competition models,” *Physics Letters A*, vol. 381, no. 11, pp. 1014 – 1020, 2017.

- [14] Wikipédia, “Tabuleiro toroidal xadrez — wikipédia, a enciclopédia livre,” 2017. Disponível em:<[https://pt.wikipedia.org/wiki/Tabuleiro_\(xadrez\)#/media/File:Toroidal_chessboard.jpg](https://pt.wikipedia.org/wiki/Tabuleiro_(xadrez)#/media/File:Toroidal_chessboard.jpg)> Acesso em 06 Fev. 2018.
- [15] J. Wilson and G. Ritter, *Handbook of Computer Vision Algorithms in Image Algebra*. CRC Press, 2000.
- [16] Wikipédia, “Vizinhança de von neumann,” 2015. Disponível em:<<http://upload.wikimedia.org/wikipedia/commons/b/ba/Vierer-Nachbarschaft.png>> Acesso em 6 Fev. 2018.
- [17] G. Melotti, *Aplicação de automatos celulares em sistemas complexos: Um estudo de caso em espalhamento de epidemias*. Dissertação de Mestrado, Universidade Federal de Minas Gerais, Belo Horizonte, Minas Gerais, 2009.
- [18] J. V. Neumann, *Theory of Self-Reproducing Automata*. Champaign, IL, USA: University of Illinois Press, 1966.
- [19] H. L. Anderson, “Metropolis, monte carlo, and the maniac,” 1986. Disponível em:<<http://library.lanl.gov/cgi-bin/getfile?00326886.pdf>> Acesso em 06 Fev. 2018.
- [20] M. Hjorth-Jensen, *Lecture Notes on Computational Physics*. University of Oslo, (2008). Disponível em: <http://www.physics.ohio-state.edu/~ntg/780/readings/hjorth-jensen_notes2008.pdf> Acesso em: 29 Nov. 2011.
- [21] J. V. de Oliveira Silva, *Estudando a função de autocorrelação no jogo pedra, tesoura e papel*. Trabalho de Conclusão de Curso, Universidade Estadual de Maringá, Maringá, Parana, 2016.
- [22] M. Peltomäki and M. Alava, “Three- and four-state rock-paper-scissors games with diffusion,” *Phys. Rev. E*, vol. 78, p. 031906, Setembro 2008.
- [23] N. Marwan, M. C. Romano, M. Thiel, and J. Kurths, “Recurrence plots for the analysis of complex systems,” *Physics Reports*, vol. 438, no. 5–6, pp. 237 – 329, 2007.
- [24] J.-P. Eckmann, S. O. Kamphorst, and D. Ruelle, “Recurrence plots of dynamical systems,” *EPL (Europhysics Letters)*, vol. 4, no. 9, p. 973, 1987.
- [25] C. Webber, C. Ioana, and N. Marwan, *Recurrence Plots and Their Quantifications: Expanding Horizons: Proceedings of the 6th International Symposium on Recurrence Plots, Grenoble, France, 17-19 June 2015*. Springer Proceedings in Physics, Springer International Publishing, 2016.
- [26] M. Fukino, Y. Hirata, and K. Aihara, “Coarse-graining time series data: Recurrence plot of recurrence plots and its application for music,” *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 26, no. 2, p. 023116, 2016.
- [27] C. Ahlstrom, P. Hult, and P. Ask, *Thresholding distance plots using true recurrence points*, vol. 3. 2006.
- [28] U. Rajendra Acharya, K. Paul Joseph, N. Kannathal, C. M. Lim, and J. S. Suri, “Heart rate variability: a review,” *Medical and Biological Engineering and Computing*, vol. 44, no. 12, pp. 1031–1051, 2006.

- [29] Z.-B. Wu, “Recurrence plot analysis of dna sequences,” *Physics Letters A*, vol. 332, no. 3–4, pp. 250 – 255, 2004.
- [30] K. Karandikar, T. Q. Le, A. Sa-ngasoongsong, W. Wongdhamma, and S. T. S. Bukapatnam, “Detection of sleep apnea events via tracking nonlinear dynamic cardio-respiratory coupling from electrocardiogram signals,” in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 7088–7091, July 2013.
- [31] “Nonlinear dynamics and recurrence plots for detecting financial crisis,” *The North American Journal of Economics and Finance*, vol. 26, pp. 416 – 435, 2013.
- [32] H. Xiong, P. Shang, and S. Bian, “Detecting intrinsic dynamics of traffic flow with recurrence analysis and empirical mode decomposition,” *Physica A: Statistical Mechanics and its Applications*, vol. 474, pp. 70 – 84, 2017.
- [33] X. Zeng, R. Eykholt, and R. A. Pielke, “Estimating the lyapunov-exponent spectrum from short time series of low precision,” *Phys. Rev. Lett.*, vol. 66, pp. 3229–3232, Junho 1991.
- [34] H. Rabarimanantsoa, L. Achour, C. Letellier, A. Cuvelier, and J.-F. Muir, “Recurrence plots and shannon entropy for a dynamical analysis of asynchronisms in noninvasive mechanical ventilation,” *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 17, no. 1, p. 013115, 2007.
- [35] P. Dunn and M. Davis, *Measurement and Data Analysis for Engineering and Science, Fourth Edition*. CRC Press, 2017.
- [36] Wikipedia, “Autocorrelation — wikipedia, the free encyclopedia,” 2017. [Online; accessed 26-July-2017].
- [37] L. Barroso, F. Filho, M. De Araujo Barroso, M. De Carvalho, and M. Maia, *Calculo Numerico - Com Aplicações*. HARBRA EDITORA.
- [38] E. Butkov, *Física Matemática*. LTC - Livros Técnicos e Científicos Editora S.A., 1988.
- [39] R. Brown and P. Hwang, *Introduction to random signals and applied Kalman filtering: with MATLAB exercises and solutions*. No. 1 in Introduction to Random Signals and Applied Kalman Filtering, Wiley, 1997.
- [40] W. Davenport and W. Root, *An introduction to the theory of random signals and noise*. Lincoln Laboratory publications, McGraw-Hill, 1958.
- [41] A. Roman, D. Dasgupta, and M. Pleimling, “Interplay between partnership formation and competition in generalized may-leonard games,” *Phys. Rev. E*, vol. 87, p. 032148, Março 2013.