

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
INSTITUTO DE FÍSICA  
TRABALHO DE CONCLUSÃO DE CURSO

## **Efeito de memória e reputação em teoria de jogos**

**João Vitor Oliveski Mesquita**

Trabalho de Conclusão de curso apresentado como requisito parcial para obtenção do título de Bacharel em Física.

**Orientador:**  
**Prof. Dr. Heitor Carpes Marques Fernandes.**

**Porto Alegre, RS**  
**Outubro de 2023**

Oliveski, João Vitor

Efeito de memória e reputação em teoria de jogos / João Vitor Oliveski Mesquita. -- 2023.

42

Orientador: Dr. Heitor Carpes Marques Fernandes.

Trabalho (Conclusão de curso) - Universidade Federal do Rio Grande do Sul, Instituto de Física, Porto Alegre, BR-RS, 2023.

Teoria de Jogos, Dilema do Prisioneiro, Monte Carlo, Cooperação, Memória, Percolação. I. Fernandes, Heitor C.M., orient.

# Agradecimentos

Aos meus pais, que sempre fizeram questão de dizer que estudar é fundamental. Devo tudo a eles e em especial à minha mãe que me deu tudo que eu sempre quis e precisei e nunca questionou minhas escolhas.

Aos meus dindos, que sempre me disseram que eu era como um filho. Hoje entendo o poder dessas palavras.

Ao meu primo Luan, com quem passei incontáveis noites incomodando a vó.

Ao meu primo Lucas e à Nina, que sempre me apoiaram e confiaram em mim para ser dindo do Noah.

Ao meu irmão Rodrigo, que sempre admirei pela dedicação aos estudos e comprometimento com a família.

Ao meu padrasto, que sempre fez tudo ao alcance e além.

Aos meus amigos, que tenho a sorte e o orgulho de dizer que são lendários, mas tantos que não poderei citar todos por nome aqui. Ainda assim, alguns me acompanharam mais de perto ao longo da graduação. Pedro Henrique Mendes e Matheus Demarchi, que entraram comigo no curso e me apoiaram em incontáveis momentos. Leonardo Evaristo e Matheus Tissot, que conheci no Laboratório de Altas Pressões durante minha primeira IC e são pesquisadores que admiro muito por suas capacidades, assim como pelo carisma e companheirismo. Isadora Marinho, que é minha amiga mais antiga e mesmo a milhares de quilômetros, separados até por um oceano, sempre esteve comigo inclusive nos momentos mais difíceis que enfrentei. João Pedro Dias e Bárbara Isabel de Moura, que sempre topam o caos. Ricardo Collicchio, que me mantém intacto já há 10 anos e sempre esteve disponível. Eric Naiber, que rapidamente fez toda a diferença do mundo. Jonas Germano, que apadrinhei dias antes de uma pandemia e que desde então se tornou um grande amigo.

Aos meus orientadores, que me apoiariam até se eu decidisse seguir uma carreira como *pole dancer*.

À Naira Balzaretto e ao Silvio Buchner, que me orientaram na minha primeira IC e abriram as portas do mundo da pesquisa para mim.

Obrigado.



# Resumo

O Dilema do Prisioneiro é um modelo clássico em teoria de jogos para o estudo sobre a cooperação entre agentes a princípio egoístas. Este trabalho busca investigar o efeito da memória das interações prévias na dinâmica de um sistema de jogadores espacialmente distribuídos em uma rede quadrada, com ligações entre os jogadores que podem ser acionadas e desligadas. A implementação da memória se dá através de um histórico de jogadas para cada jogador, acessível a todos até o limite da memória individual de cada jogador. Essa implementação visa simular a noção de reputação dos jogadores. Simulações numéricas extensivas foram realizadas, explorando grandes intervalos dos parâmetros, para melhor entender o comportamento do modelo proposto e como a cooperação se mantém (ou não) neste espaço de parâmetros. Os resultados mostram que a memória é um ingrediente importante para a manutenção da cooperação, visto que a inclusão da memória faz com que a cooperação se mantenha para valores de tentação até 100 vezes maiores do que no modelo sem memória, além de ter uma nova fase dominada por cooperadores que não ocorria no modelo sem memória. Além disso, se conclui que a distribuição da memória individual dos jogadores afeta drasticamente o comportamento do modelo para o intervalo de tentação estudado.

**Palavras-chave:** Teoria de Jogos, Dilema do Prisioneiro, Monte Carlo, Cooperação, Memória, Percolação.



# Abstract

The Prisoner's Dilemma is a classic model in game theory for studying cooperation among initially selfish agents. This work aims to investigate the effect of memory of previous interactions on the dynamics of a system of players spatially distributed in a square lattice, with connections between the players that can be activated and deactivated. The implementation of memory is achieved through a play history for each player, accessible to all up to the limit of each player's individual memory. This implementation aims to simulate the notion of player reputation. Extensive numerical simulations were conducted, exploring large parameter ranges, to better understand the behavior of the proposed model and how cooperation is maintained (or not) in this parameter space. The results show that memory is an important ingredient for the maintenance of cooperation, as the inclusion of memory enables cooperation to be maintained for temptation values up to 100 times greater than in the memoryless model, in addition to having a new phase dominated by cooperators that did not occur in the model without memory. Furthermore, it is concluded that the distribution of individual player memory drastically affects the model's behavior within the studied temptation range.

**Keywords:** Game Theory, Prisoner's Dilemma, Monte Carlo, Cooperation, Memory, Percolation.



# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Metodologia</b>	<b>3</b>
2.1	Monte Carlo com Cadeia de Markov . . . . .	3
2.2	Análise de Percolação . . . . .	6
2.2.1	Algoritmo de Hoshen-Kopelman . . . . .	7
<b>3</b>	<b>Revisão do modelo de partida</b>	<b>11</b>
<b>4</b>	<b>Modelo proposto</b>	<b>17</b>
<b>5</b>	<b>Resultados</b>	<b>21</b>
5.1	Definindo as condições iniciais do modelo . . . . .	21
5.2	Probabilidades de Percolação e Fração de Cooperadores vs Tentação . . . . .	23
5.3	Diferentes distribuições da capacidade de memória . . . . .	25
5.3.1	Distribuição uniforme . . . . .	25
5.3.2	Distribuição de lei de potência . . . . .	27
<b>6</b>	<b>Conclusão e perspectivas</b>	<b>31</b>
	<b>Referências Bibliográficas</b>	<b>33</b>



# Capítulo 1

## Introdução

Teoria de jogos é uma atividade multidisciplinar que envolve diferentes áreas da ciência como Biologia [1], Economia [2], Psicologia [3], Computação [4] e Física [5–11]. Neste trabalho damos conta da face da teoria de jogos cujos problemas são abordados através de métodos conhecidos da mecânica estatística e interpretados pelo ponto de vista físico.

A cooperação é uma das características que define a natureza humana. Sem os altos níveis de cooperação de que somos capazes, a vida em sociedade jamais seria possível [12]. Na literatura se encontram trabalhos que investigam quais são os possíveis mecanismos que levam ao surgimento e sobretudo à manutenção da cooperação em diversas populações [13] apesar de que inicialmente parece contraditório que o altruísmo seja evolutivamente selecionado. Ainda assim, o problema não está completamente resolvido.

Um importante modelo de teoria de jogos para o estudo da cooperação é o dilema do prisioneiro. O dilema do prisioneiro é um jogo que consiste em dois jogadores escolherem entre cooperar (C) e desertar (D), recebendo benefícios diferentes de acordo com a escolha de ambos jogadores. O jogador que coopera com um desertor recebe o *payoff* (ou ganho) do Sonso ( $S$ ) enquanto o desertor recebe a Tentação ( $T$ ). Ambos jogadores recebem uma Recompensa ( $R$ ) quando os dois cooperam ou uma Punição ( $P$ ) quando desertam. A regra que define esse jogo como um dilema do prisioneiro é a condição de que os *payoffs* estão relacionados de acordo com  $T > R > P > S$ , de maneira que a melhor decisão, independente da escolha do outro jogador, sempre é desertar. Além dessa relação, para jogos iterados repetidos ao longo de várias rodadas também se faz necessária a condição de que  $2R > T + S$ .

Nesse trabalho inicialmente fazemos uma revisão de um modelo estruturado apresentado por Hauert e Szabó em [14] e acrescentamos novos “ingredientes” a este modelo inspirados pelos resultados apresentados por Cuesta *et al* em [15]. Em [15] é demonstrado, utilizando os dados observados em um experimento com pessoas reais, que a grandeza que promove a cooperação entre os jogadores é a reputação. No experimento executado por Cuesta, uma das principais características é a rede de jogadores ser dinâmica, com os jogadores podendo escolher entre rodadas com quais jogadores eles ainda querem jogar se baseando no histórico de jogadas de todos participantes. Dessa característica do experimento surge o interesse de permitir uma dinâmica para as conexões na rede do modelo de partida, tentando aproximar um modelo simulado em computador a um modelo experimental com pessoas reais, o que traz a possibilidade de mais variações na modelagem além de poder simular os efeitos de ter muitos jogadores na rede<sup>1</sup>.

Para a execução desse trabalho utilizaremos de simulações de Monte Carlo com cadeia de Markov e análise de percolação como forma de avaliar efeitos geométricos no sistema. Assim, começamos com uma breve revisão do ferramental necessário (Capítulo 2), tratando brevemente de Monte Carlo (Seção 2.1), percolação (Seção 2.2) e o algoritmo de Hoshen-Kopelman (Seção 2.2.1), para então seguirmos com uma revisão do clássico dilema do prisioneiro estruturado numa rede quadrada com alguns resultados de [14] (Capítulo 3). Depois disso, apresento o modelo proposto pontuando as diferenças entre esse e o clássico modelo de partida (Capítulo 4) e finalmente os resultados desse modelo proposto para distribuições de probabilidade diferentes para a memória individual dos jogadores (Capítulo 5). Para o modelo proposto fazemos a medida da probabilidade de percolação da rede em função do valor da tentação utilizando o algoritmo de Hoshen-Kopelman.

---

<sup>1</sup>No caso do experimento de Cuesta *et al*, a título de comparação, foram feitas 24 experimentos com 243 participantes diferentes distribuídos em grupos de 17 a 25 pessoas. Em contraste, as simulações nesse trabalho contavam com redes de  $10^4$  a  $1,6 \times 10^4$  agentes.

# Capítulo 2

## Metodologia

### 2.1 Monte Carlo com Cadeia de Markov

De maneira generalista e simplificada, ainda que mantendo certo rigor, uma simulação de Monte Carlo em mecânica estatística consiste na seguinte ideia:

“Simular as flutuações térmicas de um sistema de estado em estado ao longo de um experimento. Consideramos o cálculo de um valor esperado como a média temporal sobre os estados pelos quais um sistema passa. Em uma simulação de Monte Carlo, simulamos diretamente esse processo, criando um modelo para o sistema no nosso computador e fazendo esse sistema passar por uma variedade de estados de maneira que a probabilidade de ele estar em qualquer estado particular  $i$  em um dado tempo  $t$  seja igual ao seu peso em um sistema real  $\omega_i(t)$ . Para fazermos isso, precisamos escolher uma dinâmica para nossa simulação - uma regra para trocarmos de um estado para outro durante a simulação - que resulta em cada estado sendo observado exatamente com a probabilidade adequada a ele.”

Traduzido livremente, esse trecho do primeiro capítulo do livro *Monte Carlo Methods in Statistical Physics* [16] resume muito bem no que consistem os métodos de Monte Carlo. Contudo, Newman e Barkema investem as primeiras 18 páginas do livro dando a base para que alguma coisa desse trecho faça sentido para o leitor. Como exemplificado, métodos de simulação de Monte Carlo são conteúdo o bastante para escrever livros inteiros e não é tarefa simples resumir no que estes consistem. Ainda assim, apesar

da vasta literatura tratando de métodos de Monte Carlo no contexto da mecânica estatística, apresentarei de maneira breve os principais pontos necessários para a execução desse trabalho.

O poder de um método de Monte Carlo vem da possibilidade de amostrar os estados de um sistema com a devida distribuição de probabilidade mesmo sem conhecer a função partição do sistema [16]. É utilizar do conceito de amostragem por importância junto com a ideia de cadeia de Markov para evoluir as configurações de um sistema de maneira que todos os estados pelos quais o sistema passa sejam representativos do comportamento do sistema no limite termodinâmico, ou seja, esses estados aparecem com probabilidade descrita pela distribuição de Boltzmann.

Provavelmente o algoritmo mais famoso, e também mais simples, para simulações de Monte Carlo é o algoritmo de Metropolis [17], que consiste em tomar a taxa de transição de um estado  $\mu$  para um estado  $\nu$  como 1 no caso em que a diferença de energia entre esses estados é negativa e  $e^{-\Delta E/kT}$  no caso contrário. A taxa de transição quando a diferença de energia entre os estados é positiva tem esse valor para garantir que os pesos de Boltzmann para os dois estados em questão sejam respeitados ao longo de uma longa sequência de transições de estados e a distribuição de Boltzmann para esse sistema seja reproduzida.

O algoritmo de Metropolis é muitas vezes apresentado tendo como o exemplo a simulação do modelo de Ising [18], que é uma representação de um ferromagneto com spins  $\pm 1$  e interação aos pares descrita pelo hamiltoniano  $\mathcal{H} = -J \sum_{\langle ij \rangle} s_i s_j - B \sum_i s_i$ . Nesse hamiltoniano,  $J$  é uma constante que dita a intensidade da interação entre os spins,  $s_i$  é o valor do spin  $i$  ( $+1$  ou  $-1$ ) e  $B$  é a intensidade de um campo magnético externo, muitas vezes tomado como zero para simplificar. Dessa maneira, uma simulação de Monte Carlo do modelo de Ising utilizando o algoritmo de Metropolis se daria da seguinte forma:

- Selecionar um spin qualquer e inverter seu valor;
- Calcular a diferença de energia no estado inicial e no estado proposto dada pelo hamiltoniano;
- Sortear um número entre 0 e 1;
- Comparar o número sorteado com  $e^{-\Delta E/kT}$ ;

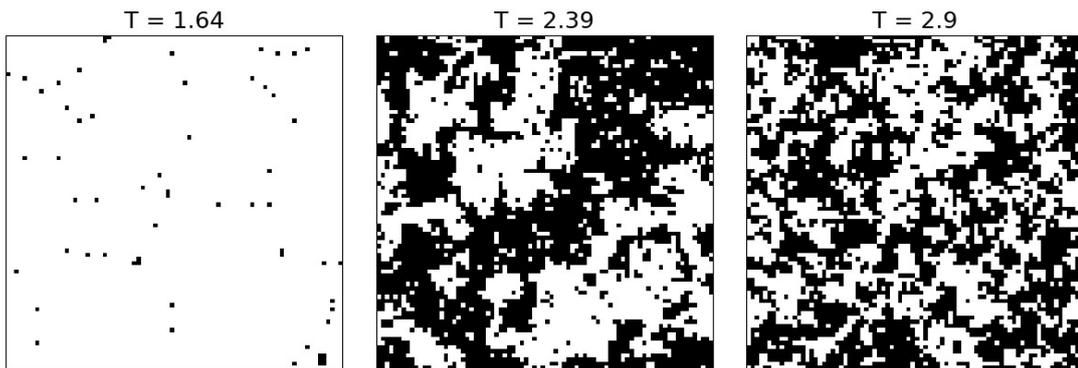


FIGURA 2.1. *Snapshots* típicos de uma simulação do modelo de Ising com três temperaturas diferentes. Cada sítio da rede é um spin. Essa imagem foi retirada de [19].

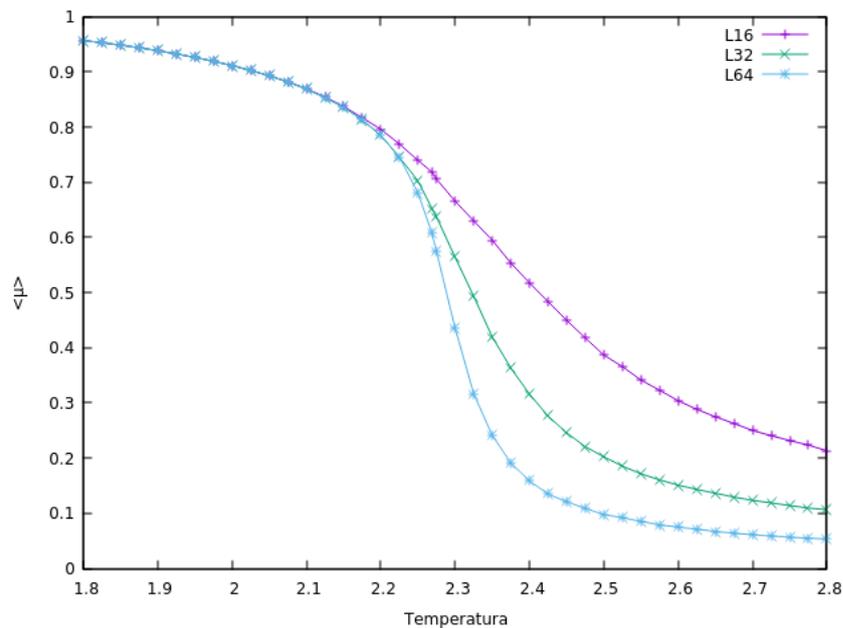


FIGURA 2.2. Magnetização no modelo de Ising em função da temperatura. A magnetização é um exemplo de medida macroscópica que vem da contribuição microscópica de cada spin da rede. Esse tipo de fenômeno é a ideia central da mecânica estatística e é um aspecto importante do por que usamos simulações de Monte Carlo para estudarmos sistemas complexos. Essa imagem foi retirada de [20].

- Se o número for maior que a exponencial, recusamos o estado proposto e permanecemos no estado inicial<sup>1</sup>. Caso contrário aceitamos a nova configuração;
- Repetimos esse processo diversas vezes.

Esse exemplo serve para ilustrar como uma simulação de Monte Carlo corre de

<sup>1</sup>Esse ponto é extremamente importante para que a condição de balanço detalhado seja respeitada [16].

maneira geral, mas a expressão para o hamiltoniano pode mudar completamente e um novo algoritmo adequado pode ser definido. A maneira como sorteamos novas configurações para o sistema é o que define a *dinâmica* da simulação e no caso do algoritmo de Metropolis temos uma dinâmica muito simples de sortear um sítio por vez e testar a troca do valor desse sítio. Outras dinâmicas, inclusive várias muito mais complexas, existem e são utilizadas para diferentes propósitos a depender do sistema estudado [16].

Dessa maneira, os modelos apresentados nesse trabalho foram todos simulados seguindo uma dinâmica muito parecida com a descrita aqui para o modelo de Ising, com a única diferença sendo que no lugar da energia do sistema estaremos avaliando os *payoffs* dos jogadores nos sítios sorteados e usaremos uma distribuição de Fermi para a probabilidade de transição como descreveremos nos próximos capítulos.

## 2.2 Análise de Percolação

Entender o estudo da teoria de percolação [21] é mais fácil através de um exemplo da análise do modelo mais tradicional da percolação: a ocupação aleatória.

Começamos com uma rede quadrada de  $N = L \times L$  sítios, onde  $L$  é o tamanho do lado da rede. Os vizinhos de um dado sítio são os sítios imediatamente acima, abaixo e dos lados de tal sítio (vizinhança de Von Neumann [22]<sup>2</sup>). Para cada um dos sítios da rede, iremos decidir ocupar ou não o sítio com probabilidade  $p$ . Após esse processo, verificamos quais são os conjuntos de sítios que fazem parte do mesmo *cluster* (aglomerado traduzido do inglês, ver Figura 2.3), isto é, quais são os sítios vizinhos entre si. Em um *cluster*, todos sítios fazem parte de uma mesma cadeia contínua de sítios vizinhos. Dizemos que um *cluster percola* se este toca todos os lados da rede. De maneira ilustrativa, se fossemos caminhar por cima desses sítios ocupados, quando um *cluster* percola quer dizer que poderíamos atravessar de um lado até o outro da rede sempre em cima do mesmo *cluster*. As propriedades desses *clusters* são de maneira resumida o tema de estudo da teoria de percolação.

Do exemplo da ocupação aleatória, a pergunta que surge inicialmente é: para qual

---

<sup>2</sup>Os vizinhos definidos assim muitas vezes são chamados de primeiros vizinhos.

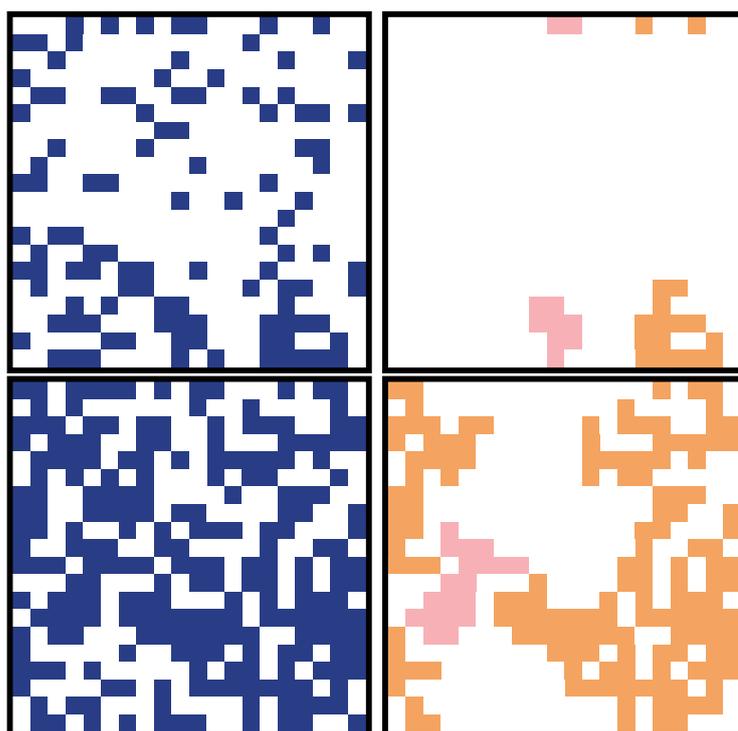


FIGURA 2.3. Do lado esquerdo da figura temos duas redes quadradas com  $L = 20$  ocupadas aleatoriamente com  $p = 0,3$  e  $p = 0,6$  respectivamente. Ao lado dessas temos indicados os dois maiores *clusters* de cada rede, sendo o maior pintado de laranja e o segundo maior pintado de rosa. Observe que para nossas análises utilizamos condições de contorno periódicas, por isso podemos notar que alguns sítios dos *clusters* indicados parecem desconectados, mas na verdade estes estão ligados pelas bordas da rede. Note também que o maior *cluster* da rede de baixo (ocupada com  $p = 0,6$ ) é um *cluster* percolante.

valor de  $p$  podemos afirmar que sempre encontraremos um *cluster* percolante? Para responder essa pergunta veremos uma breve apresentação do algoritmo de Hoshen-Kopelman, que é utilizado para encontrar os *clusters* de uma dada rede.

### 2.2.1 Algoritmo de Hoshen-Kopelman

O algoritmo de Hoshen-Kopelman [23–25] é utilizado para identificar *clusters* em uma dada rede. A ideia básica desse algoritmo é etiquetar cada sítio com um número que designa de qual *cluster* aquele sítio faz parte. Se um sítio tem um número que não aparece em mais nenhum outro sítio, este é um sítio isolado, ou seja, não faz parte de um *cluster*. Para formarmos os *clusters* passamos duas vezes pela rede toda. A primeira para etiquetarmos cada sítio com um número de *cluster* preliminar, olhando quais sítios

vizinhos estão ocupados e garantindo que o sítio receba um número que faça parte do *cluster* de seus vizinhos ou que esse número seja único; e a segunda para reduzirmos esses números do conjunto mínimo de *clusters*. Com o passo a passo ficará mais claro.

Começamos passando em cada um dos sítios a partir do primeiro sítio na primeira linha da rede e indo linha por linha até o último sítio. Inicialmente o número do *cluster* de cada sítio é sua própria coordenada. Para cada sítio iremos aplicar duas funções:  $union(x, y)$  e  $find(x)$ . A função  $union(x, y)$  faz com que a etiqueta do sítio  $x$  seja entendida como equivalente à etiqueta do sítio  $y$ . Caso passássemos por cada sítio da rede aplicando esta função comparando com o primeiro sítio, por exemplo, estaríamos indicando que todos os sítios da rede fazem parte do mesmo *cluster*. Já a função  $find(x)$  percorre a cadeia de etiquetas que foram indicadas como equivalentes pela função  $union$  e retorna a primeira dessas etiquetas. Tomemos as seguintes operações, assumindo que  $x < y < z$ .

- $union(x, y)$ : etiqueta[ $y$ ] = etiqueta[ $x$ ]
- $union(y, z)$ : etiqueta[ $z$ ] = etiqueta[ $y$ ]
- $find(z)$ : retorna  $x$

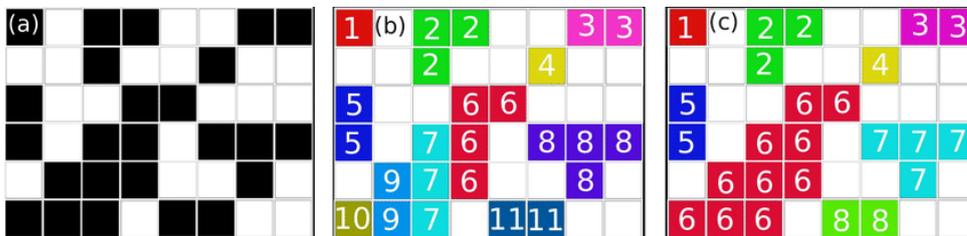


FIGURA 2.4. Ilustração da aplicação do algoritmo de Hoshen-Kopelman em uma rede quadrada com sítios binários (ocupados ou desocupados) (a). Em (b) vemos a mesma rede etiquetada após a primeira parte do algoritmo e em (c) vemos o resultado da segunda passada sobre a rede, que reduz o número de *clusters* encontrados ao mínimo. Figura retirada de [26].

Como cada sítio tem sua própria coordenada como etiqueta inicial, a menor dessas etiquetas é justamente  $x$ . As funções  $union$  nesse caso fizeram com que as etiquetas dos sítios  $x$  e  $y$  fossem equivalentes e depois que  $y$  e  $z$  também, de forma que a etiqueta “raíz” de  $z$  seja  $x$ , então  $find(z)$  retorna  $x$  e esses três sítios fazem parte do mesmo *cluster* designado pela etiqueta  $x$ . A Figura 2.4 ilustra a aplicação do algoritmo de Hoshen-Kopelman numa rede quadrada.

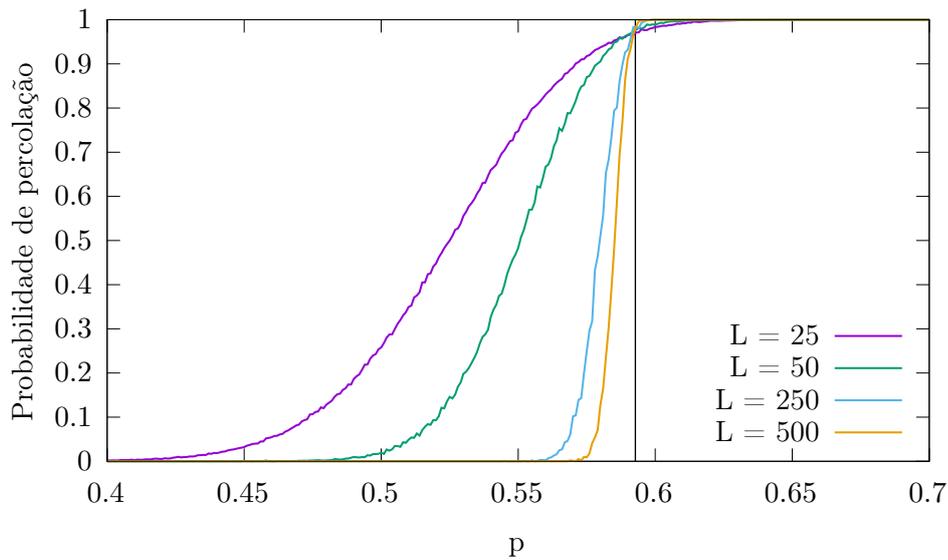


FIGURA 2.5. Probabilidade de percolação em função da probabilidade de ocupação  $p$  para a ocupação aleatória na rede quadrada. O limiar de percolação da rede quadrada é  $p_c = 0,592746$  [27] e está indicado pela linha vertical preta no gráfico.

Utilizando o passo a passo descrito acima podemos por exemplo chegar nas imagens da Figura 2.3. Para identificar quais são os maiores *clusters* podemos simplesmente contar a quantidade de sítios de cada etiqueta após a aplicação do algoritmo de Hoshen-Kopelman. Para verificar se um dado *cluster* percola a rede, podemos verificar se esse cluster tem pelo menos um sítio em cada linha da rede e depois em cada coluna. Dessa forma confirmamos que ele toca os quatro lados da rede.

Voltando ao caso da ocupação aleatória podemos agora numericamente determinar qual é o limiar de percolação de qualquer rede. Para isso podemos gerar várias redes, cada uma ocupada de acordo com uma probabilidade  $p$  e, então, utilizando o algoritmo de Hoshen-Kopelman verificar quantas dessas possuem *clusters* percolantes. A razão entre o número de redes com *clusters* percolantes e o número de redes geradas é uma estimativa da probabilidade de percolação para aquela probabilidade de ocupação  $p$ . Fazendo isso para diferentes tamanhos de rede se chega no resultado da Figura 2.5. Observe como a curva de probabilidade de percolação vai ficando cada vez mais íngreme conforme o tamanho da rede aumenta. Um algoritmo mais eficiente pensado para determinar o limiar de percolação da rede quadrada, que até o momento não tem solução analítica, é descrito por Newmann e Ziff em [27].

Na execução desse trabalho será utilizado o algoritmo de Hoshen-Kopelman para

estimar a probabilidade de percolação das redes do modelo de teoria de jogos que propomos no capítulo 4.

# Capítulo 3

## Revisão do modelo de partida

O modelo base que utilizamos nesse trabalho é o dilema do prisioneiro espacialmente distribuído em uma rede quadrada apresentado por Hauert e Szabó na revisão [14]. Esse modelo foi originalmente proposto por Martin Nowak e Robert May em um artigo para a revista Nature em 1992 [28], a menos de alguns detalhes, sobretudo quanto à estocasticidade do modelo.

Os autores partem da definição do dilema do prisioneiro, que pode ser resumido na matriz de *payoff* 3.1 e a relação entre os *payoffs*:

$$T > R > P > S \tag{3.1}$$

Então apenas como um exemplo supomos que o jogador 1 escolhe jogar C (cooperar) enquanto o jogador 2 escolhe D (desertar). Nesse caso o *payoff* do jogador 1 será  $S$  e o *payoff* do jogador 2 será  $T$ . Outro aspecto importante sobre o dilema do prisioneiro é se o jogo será repetido ou não. No caso do modelo de Hauert, e conseqüentemente do nosso na sequência, o jogo é iterado e se repete por várias rodadas. Nesse cenário, outra relação entre os *payoffs* se faz necessária:  $2R > T + S$ , de maneira que os *payoffs* somados dos jogadores, ou seja, da comunidade, sejam maiores ao cooperarem.

		Co-jogador	
		C	D
Jogador	C	R	S
	D	T	P

TABELA 3.1. Matriz de *payoffs* do dilema do prisioneiro.

Outra característica do modelo clássico é a distribuição espacial dos jogadores, que permite a formação de aglomerados de jogadores com a mesma estratégia. Cada jogador é posicionado em um sítio de uma rede quadrada  $L \times L$ , com condições de contorno periódicas de maneira a reduzir efeitos de tamanho finito, e terá uma estratégia inicial sorteada entre “All C” e “All D”. Assim, os jogadores jogam um dilema do prisioneiro com cada um de seus vizinhos, acumulando os 4 *payoffs* dessas interações até o final da rodada. Após a rodada, cada jogador tem um vizinho sorteado com o qual irá comparar os *payoffs*, copiando a estratégia do seu vizinho se o *payoff* deste for maior que seu próprio. A decisão de copiar ou não a estratégia vizinha se dá utilizando a distribuição de probabilidade da equação 3.2.

$$W(x \leftarrow y) = \frac{1}{1 + e^{-(P_y - P_x)/\kappa}} \quad (3.2)$$

onde  $W(x \leftarrow y)$  denota a probabilidade do jogador  $x$  copiar a estratégia do jogador  $y$ , dados os *payoffs* respectivos  $P_x$  e  $P_y$ .

A título de esclarecimento, as estratégias possíveis para o dilema do prisioneiro são diversas e normalmente são descritas como vetores de 4 componentes [29], sendo cada um deles a probabilidade do jogador cooperar dado sua última jogada e a última jogada de seu co-jogador  $(P_{CC}, P_{CD}, P_{DC}, P_{DD})$ . A estratégia “*Tit-for-Tat*” (“olho por olho” numa tradução livre) por exemplo é famosa por ganhar dois torneios computacionais [30] seguidos e pode ser expressa como  $(1, 0, 1, 0)$ , ou seja, o jogador que escolhe a estratégia *Tit-for-tat* sempre coopera caso o co-jogador tenha cooperado e sempre deserta caso o co-jogador tenha desertado. As estratégias “*All C*” e “*All D*” nesse caso são os vetores  $(1, 1, 1, 1)$  e  $(0, 0, 0, 0)$ , pois os jogadores escolhem C ou D independentemente de jogadas anteriores, e são essas que os jogadores dos nossos sistemas usam já que uma vez que escolheram C(D) para a rodada, jogarão C(D) com todos seus vizinhos.

A constante  $\kappa$  na distribuição de probabilidade de transição 3.2 é o ruído associado à regra de atualização da estratégia, análogo à temperatura na mecânica estatística, sendo uma maneira de incluir no modelo uma medida para a “irracionalidade” dos agentes. Essa “irracionalidade” vem da ideia de que os jogadores adotam automaticamente a estratégia de seu vizinho se ele está com melhor desempenho em termos de *payoff* ( $P_y - P_x > 0$ ), mas também pode com probabilidade não nula adotar a estratégia de um jogador vizinho

com *payoff* menor que o seu. Isso serve principalmente para evitar que o sistema fique preso em mínimos locais, mas é assumidamente uma de muitas abordagens possíveis pra atualizar o estado dos jogadores. Outras abordagens, inclusive que levam em consideração outras variáveis do sistema para a regra de atualização ou até que tratam a evolução do sistema como um automata celular se encontram na literatura [31, 32]. Ainda assim, a distribuição de Fermi é utilizada frequentemente na literatura por evitar os mínimos locais e por facilmente sair do limite estocástico ( $\kappa > 0$ ) e voltar ao que se costuma pensar como um limite determinístico ( $\kappa \rightarrow 0$ ).

Uma consequência direta dessa escolha para a dinâmica é que as únicas mudanças de estratégia que podem ocorrer são nas bordas entre um aglomerado de cooperadores e um de desertores já que um jogador só pode trocar de estratégia baseado na estratégia de seu vizinho [22]. Ora, se um jogador está na parte interna de um aglomerado, ele só poderá copiar a estratégia de seus vizinhos, que como estão no mesmo aglomerado tem todos a mesma estratégia. Se sabe que para modelos espacialmente distribuídos que seguem essa regra de atualização sempre haverá pelo menos dois estados absorventes: todos cooperam ou todos desertam. Em outros trabalhos [31] se fala desse tipo de regra de atualização como não inovativas já que um jogador só tem acesso às estratégias ao seu redor e não à quaisquer estratégias presentes na rede como um todo. A Figura 3.1 ilustra a dinâmica do modelo.

Para estudar de maneira mais sistemática esse modelo utilizamos uma parametrização dos *payoffs* de maneira que tenhamos

$$\begin{cases} T = 1 + r \\ R = 1 \\ P = 0 \\ S = -r \end{cases}$$

onde agora temos o parâmetro  $r$  como uma grandeza que é análoga à tentação, ou seja, quanto maior  $r$  maior a tentação dos jogadores em desertar e portanto se espera que a fração de jogadores que são cooperadores ( $\rho_C$ ) diminua. Outras parametrizações, como a do chamado dilema do prisioneiro fraco, são encontradas na literatura [9].

Para as simulações desse modelo, iniciamos o sistema com uma rede quadrada  $L \times L$

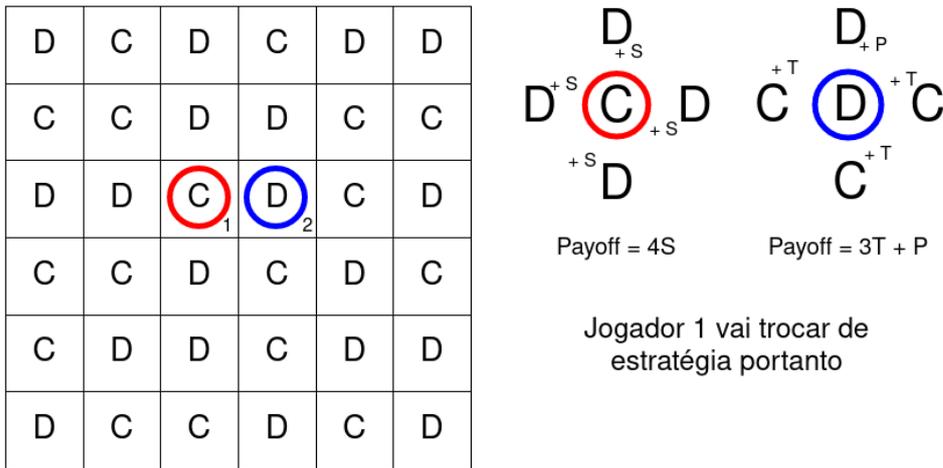


FIGURA 3.1. Ilustração da dinâmica adotada para a simulação do modelo. O jogador sorteado para avaliar sua estratégia está circulado em vermelho e o vizinho sorteado para a comparação está circulado em azul. Como o *payoff* do jogador 2 é claramente maior que o *payoff* do jogador 1, esse vai trocar de estratégia e jogar D na próxima rodada. Repetindo  $L^2$  vezes esse processo completamos 1 MCS.

sorteando cada sítio como um cooperador (C) ou desertor (D) com probabilidade 50%. Então os passos de Monte Carlo se iniciam: sorteamos um jogador aleatoriamente na rede, calculamos seu *payoff* baseado no estado dos seus vizinhos, sorteamos um vizinho, calculamos seu *payoff* também, comparamos esses *payoffs* e por fim atualizamos o estado do jogador. Repetimos esse processo  $L^2$  vezes para que em média cada jogador tenha seu estado atualizado uma vez a cada passo de Monte Carlo (Monte Carlo *step*, MCS). Tomando os primeiros  $10^4$  MCS como um período de transiente para que o sistema chegue no estado estacionário, medimos a fração média de cooperadores  $\rho_C$  utilizando a média temporal dos próximos  $10^5$  MCS. Na Figura 3.2 vemos o gráfico de uma série temporal típica no regime estacionário.

Podemos entender então qual o comportamento do modelo conforme o valor de  $r$  cresce. Para cada simulação fixamos um valor para  $r$  e evoluímos o sistema conforme descrito acima, tomando várias amostras para cada  $r$ . Com isso chegamos no gráfico da Figura 3.3.

O resultado encontrado é uma transição de fase em  $r = r_c$ , de maneira que o sistema passa de uma fase heterogênea com cooperadores e desertores num equilíbrio dinâmico para uma fase em que os desertores sempre dominam a rede inteira. Pelas simulações feitas por Hauert e Szabó se sabe que o valor para  $r_c$  nesse modelo é  $r_c = 0,02112(2)$ .

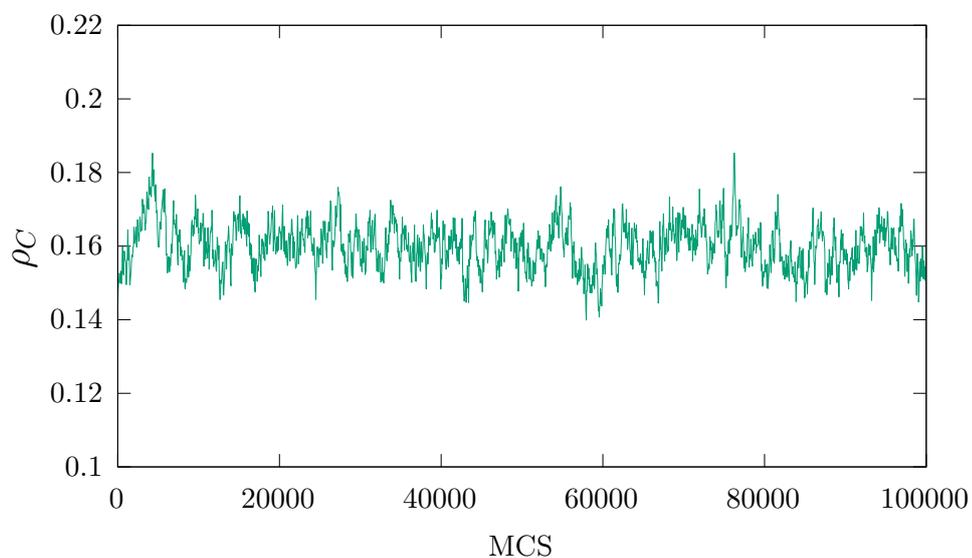


FIGURA 3.2. Série temporal típica de uma simulação do modelo clássico do dilema do prisioneiro espacialmente distribuído. Essa série temporal tem  $L = 400$  e  $r = 0,0198$ .

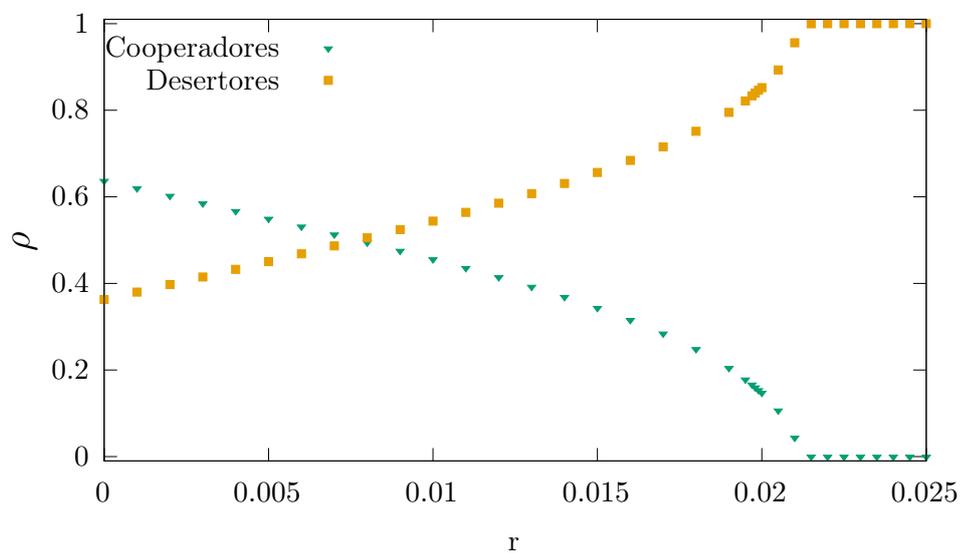


FIGURA 3.3. Fração de cooperadores (e desertores) em função do parâmetro  $r$  no modelo estudado por Hauert e Szabó em [14]. Para todas as simulações foram utilizadas redes com  $L = 400$  e o desvio padrão de cada ponto é menor que o tamanho do símbolo utilizado no gráfico.



# Capítulo 4

## Modelo proposto

O modelo clássico apresentado no capítulo anterior é o modelo de partida que escolhemos para acrescentar novos “ingredientes” na tentativa de aproximar um modelo computacional a um mais representativo do comportamento humano, inspirados no experimento de Cuesta [15]. Então, falemos das diferenças entre nosso modelo e o modelo clássico:

A principal diferença entre o modelo proposto e o modelo clássico em termos espaciais é a rigidez da rede. No nosso modelo, os jogadores estão dispostos em uma rede quadrada onde todas as ligações entre primeiros vizinhos estão ativas e posteriormente poderão ser ativadas e desativadas. Note que a rede não evolui para uma rede irregular. Apenas as ligações entre os quatro vizinhos de um jogador podem ser ativadas e desativadas.

Outro ingrediente que adicionamos ao modelo clássico é a noção de memória dos jogadores. Essa memória consiste em um registro acessível parcialmente a todos jogadores. Quando um agente assume um estado  $C$  ou  $D$ , fica registrado em seu histórico a estratégia tomada por ele e ao fim de cada rodada do dilema, os vizinhos desse jogador vão poder avaliar se querem continuar jogando com ele ou não de acordo com suas memórias individuais.

A memória individual por sua vez é um valor inteiro  $n_i$  definido inicialmente para o  $i$ -ésimo jogador no início da simulação. Esse valor  $n$ , que varia entre 1 e  $M$  é um parâmetro do modelo que determina o tamanho máximo do histórico dos jogadores. Nesse trabalho usamos  $M = 10$ , de maneira que as 10 últimas jogadas de cada jogador são registradas.

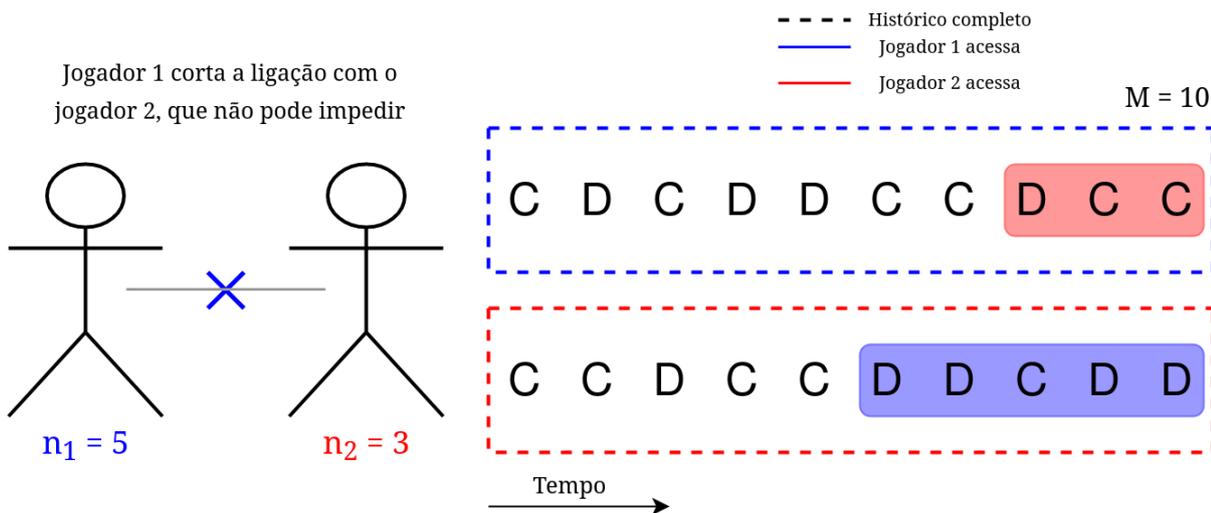


FIGURA 4.1. Ilustração do processo de avaliação das ligações entre dois jogadores. Observe que como o jogador 1 desse exemplo *se lembra* das 5 últimas jogadas do jogador 2 e estas contém 4 deserções, esse jogador escolhe cortar a ligação com seu vizinho. Importante notar também que o jogador 2 gostaria de manter a ligação ativa, já que este se lembra de 2 cooperações por parte do jogador 1 de um total de 3 jogadas.

O jogador  $i$  utiliza as últimas  $n_i$  jogadas do seu vizinho para definir se quer jogar com ele ou não na próxima rodada. Essa decisão é feita de maneira muito simples: se o vizinho cooperou mais da metade das vezes avaliadas então a ligação se mantém e os jogadores vão interagir de novo na próxima rodada. Essa razão entre as jogadas cooperativas e as jogadas observadas é pensada como uma espécie de reputação daquele jogador, uma impressão que o avaliador tem do seu vizinho.

Um ponto muito importante é que as ligações da rede não são direcionadas, ou seja, se um jogador decide cortar a ligação com seu vizinho, este não pode fazer nada para evitar.

Para melhor visualizar os novos ingredientes do modelo, observe a Figura 4.1, que ilustra o processo de avaliação das ligações de dois jogadores.

Nesse trabalho iremos nos limitar ao caso em que  $M = 10$ , porém outro aspecto interessante é a distribuição escolhida para a memória individual dos jogadores, ou seja: de que maneira são escolhidas as memórias individuais no começo da simulação. Inicialmente trataremos do caso mais simples, que é a distribuição delta para as memórias individuais, em que  $n_i = n = M$  para todo  $i$ . Nesse caso todos jogadores tem acesso ao histórico

de jogadas completo. Posteriormente, usaremos uma distribuição de lei de potência para inserir no modelo uma noção de diversidade entre os jogadores, inspirado por [33].



# Capítulo 5

## Resultados

### 5.1 Definindo as condições iniciais do modelo

Discutidas as características do modelo que propomos, é necessário então definir qual é um bom conjunto de condições iniciais do sistema para começarmos as simulações. Algumas das variáveis são iniciadas de acordo com simples restrições como a proporção entre cooperadores e desertores: iniciamos com 50% da rede cooperando, assim como Hauert e em tantos outros casos na literatura [5, 14, 22, 28, 31]. A rede é inicializada completamente conectada, assim a dinâmica começa pelos próprios mecanismos do modelo. De todas as variáveis do modelo, a parte mais sensível para a evolução do sistema é a maneira como se inicializa a memória dos agentes. Como falado na apresentação do modelo proposto, todos jogadores da rede têm acesso ao histórico de jogadas dos seus vizinhos, e esse histórico deve ser inicializado de alguma forma. Por simplicidade, se escolheu que o histórico já inicia na simulação “preenchido”, mas ainda devemos tratar de como será preenchido. Daqui, surgiram algumas abordagens a serem testadas:

- O Homem é mau
- Boa fé
- Memória aleatória

Na inicialização “O Homem é Mau”, o histórico de todos jogadores começa como se tivessem desertado o tempo todo até sua última jogada, que daí é registrada como sua estratégia atual (como definida na proporção inicial de cooperadores e desertores).

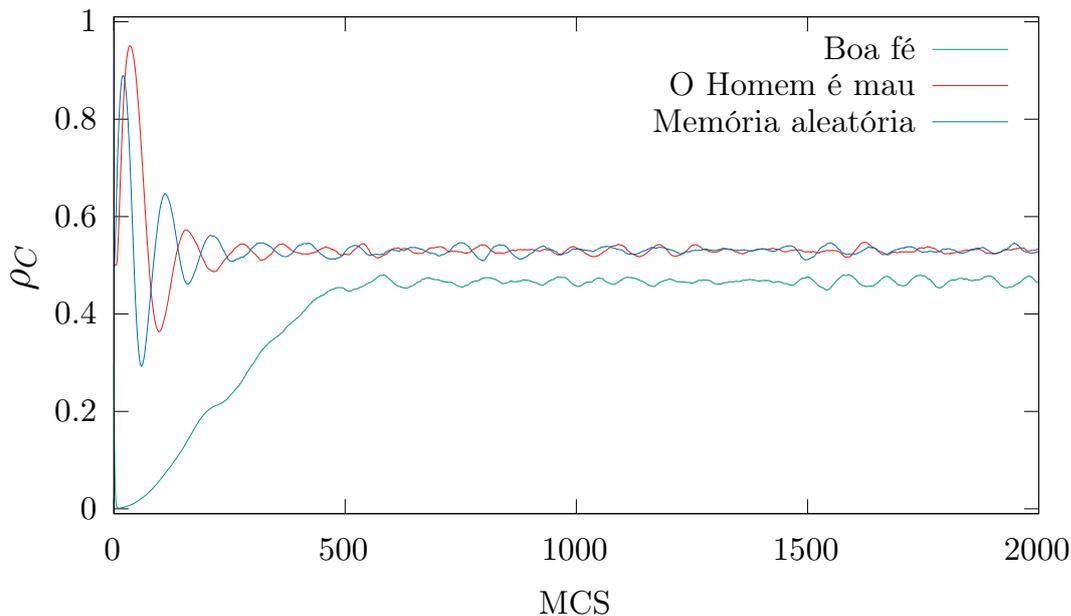


FIGURA 5.1. Fração de cooperadores média no tempo de Monte Carlo para  $r = 0,5$  com diferentes estratégias para a inicialização da memória dos jogadores. Perceba como a inicialização da Boa Fé não só parte praticamente do zero como também fica abaixo da média das outras curvas. Isso ocorre devido à quantidade de amostras extintas logo no começo da simulação.

No caso em que se tem “Boa Fé”, o contrário acontece e se supõe que todos jogadores cooperaram até o começo da simulação, exceto mais uma vez pela última jogada que é igual a estratégia daquele jogador. Por fim, temos a inicialização de Memória Aleatória, em que o arranjo de jogadas de cada jogador é escolhido ao acaso simplesmente e a última jogada do histórico é igual à estratégia do jogador como no início da simulação. Perceba que existe aqui um quê de interpretação social acerca do modelo. Se nota pelos resultados dos transientes desses modelos que a inicialização por “Boa Fé” leva a uma quantidade consideravelmente maior de extinções dos cooperadores logo no começo da simulação, quando os desertores têm tempo de explorar os cooperadores ainda conectados. Esse efeito é similar ao caso do modelo clássico sem memória, onde os desertores invadem os cooperadores que ainda não tiveram tempo de formar um *cluster*. Assumir que os agentes são desertores desde o princípio também leva a uma quantidade grande de extinções de amostras, mas ainda assim permite um transiente um tanto mais robusto, provavelmente porque os cooperadores cortam rapidamente suas conexões com desertores e assim tem tempo de formarem *clusters*. Apesar disso, a inicialização mais robusta dentre as que exploramos é a de memória aleatória, que pode ser pensada como juntar vários jogadores

de origens diferentes, cada um com seu histórico de jogadas em outro contexto qualquer, e distribuir eles na rede para que agora passem a jogar entre si. Mesmo com a quantidade muito maior de extinções para o caso da “Boa Fé”, ainda é de se supor que o comportamento do modelo não esteja afetado pela estratégia de inicialização do sistema e portanto não há perda de generalidade para o estado estacionário do sistema em comparação às outras estratégias e a diferença na média iria a zero com amostras suficientes<sup>1</sup>. É por esses motivos que decidimos seguir com as simulações sendo feitas com a estratégia de memória aleatória.

## 5.2 Probabilidades de Percolação e Fração de Cooperadores vs Tentação

Como mencionado previamente, um dos ingredientes inseridos no modelo clássico é a memória individual dos jogadores. Essa memória pode ter diferentes valores para diferentes jogadores, representada pela variável  $n_i$ , sendo  $n$  a quantidade de jogadas passadas que o jogador  $i$  lembra. Podemos, então, escolher diferentes distribuições para essa memória individual e é pensando nisso que vamos dividir as próximas seções dos resultados. Começamos tratando de uma distribuição  $\delta$ , ou seja,  $n_i = M$  para todo  $i$ , onde  $M$  como descrito no capítulo 4 é o máximo de memória no sistema e para todas as simulações feitas<sup>2</sup>  $M = 10$ .

Foram feitas 100 amostras para cada valor de  $r$  simulado, com o tamanho da rede  $L = 100$ , cada uma com  $10^3$  MCS de tempo de transiente, ou seja, usados para que o sistema alcance o estado estacionário, e  $10^4$  MCS para as medidas. Foi verificado que o tempo de transiente escolhido era suficiente para que o sistema atingisse o estado estacionário e além disso é importante mencionar que para o cálculo das médias foram utilizados medidas a cada 50 MCS para diminuir os efeitos de correlação da série temporal, mas um estudo mais cuidadoso das correlações temporais seria necessário. Com estas simulações se chega no gráfico da Figura 5.2

---

<sup>1</sup>Essa suposição não foi testada.

<sup>2</sup>A escolha do valor de  $M$  foi arbitrária, mas tomar um valor muito maior faz com que as simulações levem muito mais tempo e tornam a amostragem apropriada pouco prática. Ainda assim, esse é um dos parâmetros do modelo que resta ser explorado.

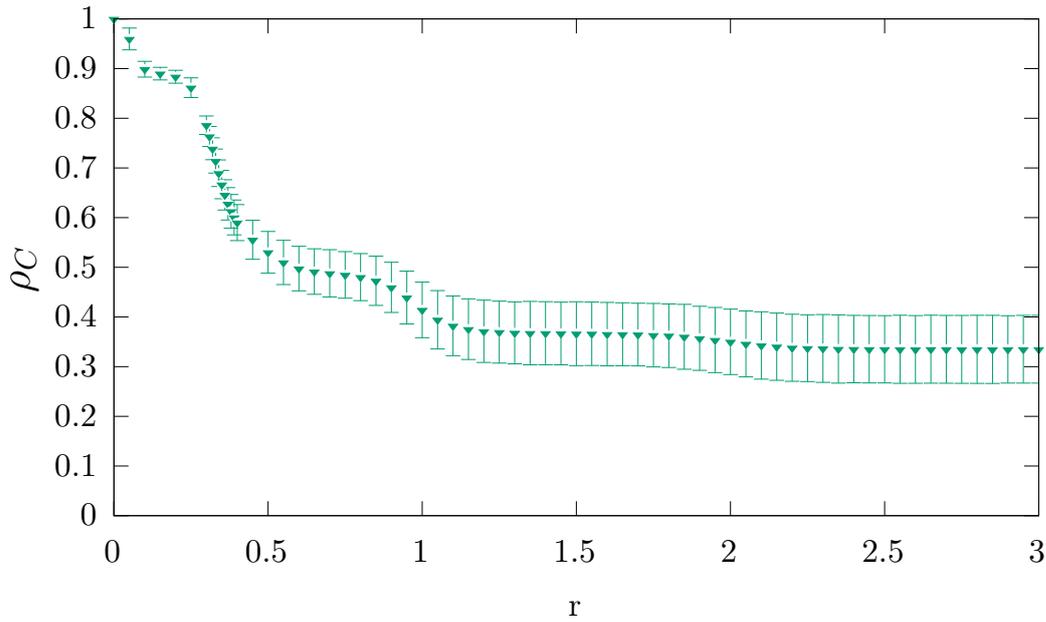


FIGURA 5.2. Fração de cooperadores em função do parâmetro  $r$  usando a distribuição de memória individual  $\delta$ , que representa a tentação. Os desvios nas medidas são calculados pelo desvio padrão sobre todas amostras para o valor de  $r$ .

A primeira e provavelmente mais importante diferença entre esse resultado e o resultado apresentado por Hauert e Szabó em [14] é a diferença do intervalo de valores de  $r$  para o qual ainda se encontram cooperadores na rede. No modelo clássico a fração de cooperadores na rede cai monotonicamente com  $r$ , chegando a zero no valor crítico  $r_c = 0,02112$ , enquanto que no modelo proposto observamos uma primeira fase dominada por cooperadores até  $r_c = 0,05$ , ou seja, já além do valor crítico do modelo clássico. O fato de a inclusão de memória favorecer a manutenção da cooperação é esperado de acordo com resultados da literatura [34].

Além dessa evidente diferença, ainda podemos notar que o comportamento de  $\rho_C$  é mais complexo que no modelo clássico, tendo inicialmente uma queda abrupta em  $r \approx 0,05$  e depois outra em  $r \approx 0,25$ . Essas quedas são seguidas por uma terceira muito mais suave por volta de  $r \approx 0,95$  e então  $\rho_C$  parece se manter relativamente constante.

Em conjunto com a fração de cooperadores, podemos observar a probabilidade de percolação da rede em função de  $r$ . Na Figura 5.3 temos as duas curvas juntas. Observe como a probabilidade de percolação corrobora a existência de uma fase intermediária no intervalo  $0,5 < r < 1,0$  já que essa também tem um patamar no mesmo intervalo. Esse

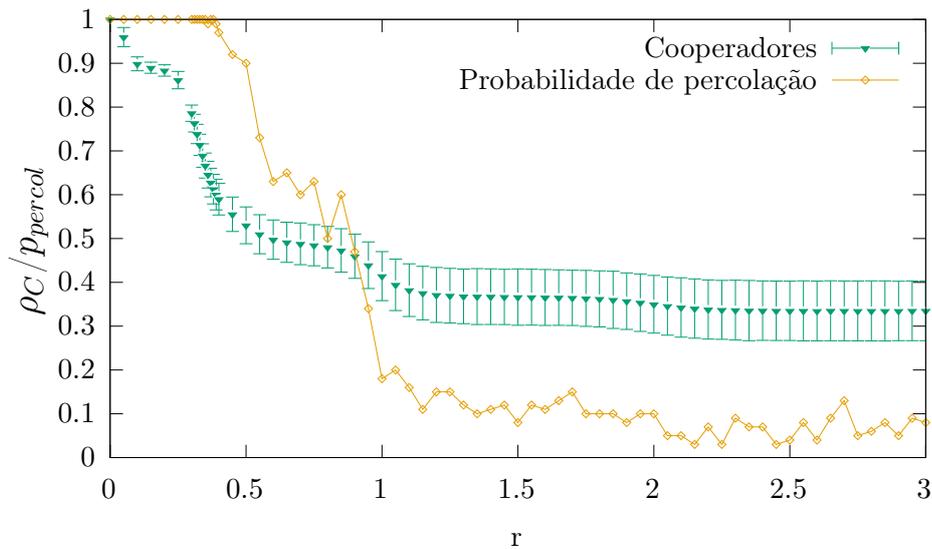


FIGURA 5.3. Fração de cooperadores e probabilidade de percolação em função de  $r$  usando a distribuição de memória individual  $\delta$ .

resultado mostra que o modelo proposto tem uma riqueza maior de comportamentos no espaço de parâmetros.

## 5.3 Diferentes distribuições da capacidade de memória

### 5.3.1 Distribuição uniforme

A distribuição de memória individual mais simples que podemos estudar sem ser a  $\delta$  é a distribuição uniforme. Nesse caso, a memória dos jogadores tem probabilidade igual de assumir qualquer um dos possíveis valores entre 1 e  $M = 10$ . Repetindo o processo feito para a distribuição  $\delta$  chegamos no resultado da Figura 5.4.

Observe como a curva de  $\rho_C$  no caso da distribuição uniforme segue aproximadamente o mesmo comportamento da curva no caso da distribuição  $\delta$ . Há, porém, uma diminuição considerável no desvio padrão das medidas e principalmente o primeiro patamar da distribuição  $\delta$ , o que ocorre entre  $r \approx 0,05$  e  $r \approx 0,25$  não aparece mais. Na Figura 5.5 temos a curva de  $\rho_C$  junto da curva da probabilidade de percolação medida. A probabilidade de percolação nesse caso não aparenta ter o mesmo comportamento do caso da  $\delta$ . Note como o patamar de  $0,5 < r < 1$  não está mais presente no gráfico. Isso

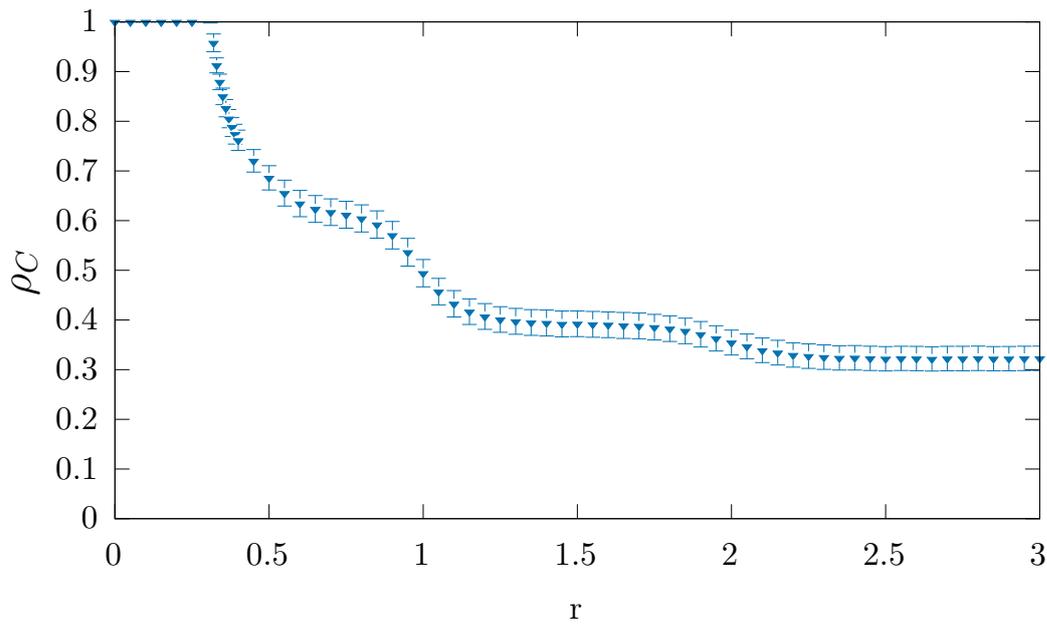


FIGURA 5.4. Fração de cooperadores em função de  $r$  usando a distribuição de memória individual uniforme.

sugere que de fato a inclusão de jogadores com memórias individuais diferentes traz pro sistema diferentes comportamentos. Pensando justamente nesse aspecto de diversidade dos agentes que foi explorado o caso da distribuição de lei de potência.

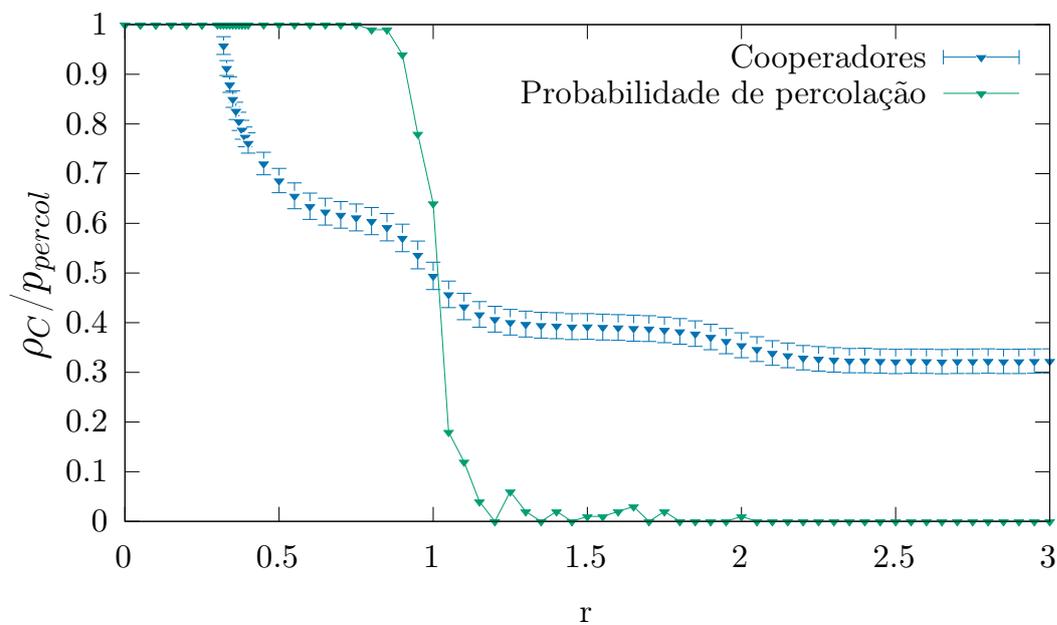


FIGURA 5.5. Fração de cooperadores e probabilidade de percolação em função de  $r$  usando a distribuição de memória individual uniforme.

### 5.3.2 Distribuição de lei de potência

A distribuição de lei de potência segue  $x^{-\tau}$ , onde  $\tau$  é um parâmetro da distribuição que quanto maior for, mais frequentes os primeiros valores de memória serão. Assim, para  $\tau$  pequeno (ainda que sempre positivo), a distribuição tende a se aproximar da distribuição uniforme, e para  $\tau$  suficientemente grande nos aproximariamos da distribuição  $\delta$ , mas ao invés de essa  $\delta$  estar centrada em  $M$  ela estaria centrada em  $n = 1$ . Temos então os resultados de  $\rho_C$  para dois valores de  $\tau$ . A Figura 5.6 mostra a curva de  $\rho_C$  para  $\tau = 2$ .

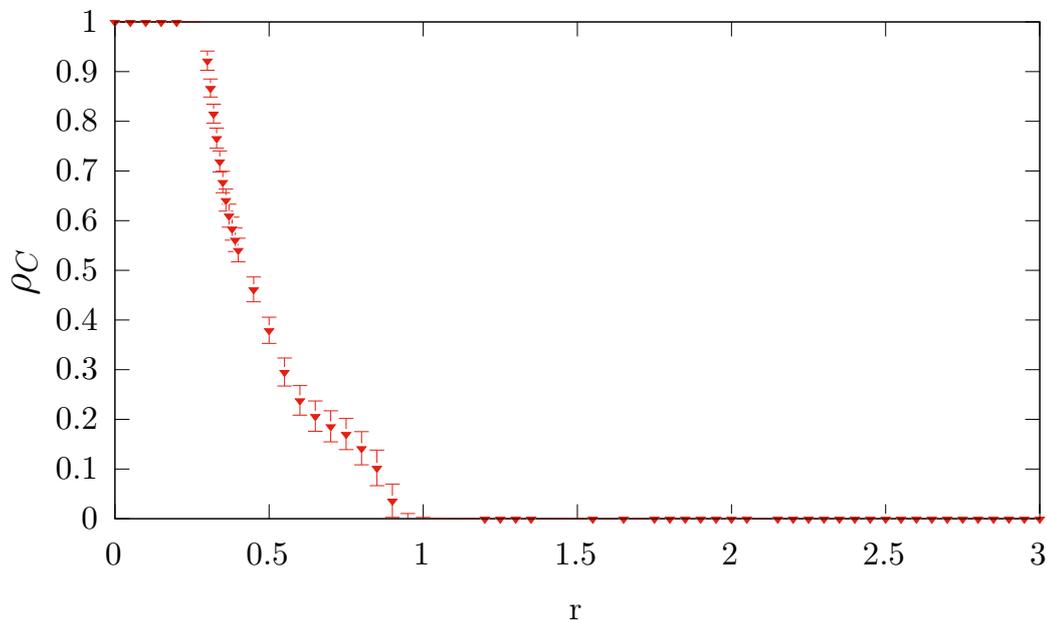


FIGURA 5.6. Fração de cooperadores em função de  $r$  usando a distribuição de memória individual de lei de potência  $x^{-\tau}$  com  $\tau = 2$ .

Observe como a fração de cooperadores média do sistema cai vertiginosamente e vai a zero por volta de  $r \approx 1$ . Os patamares que foram encontrados nas duas primeiras distribuições de memória individual agora não aparecem mais, com exceção talvez do patamar observado em  $0,5 < r < 1$ , já que a curva apresenta uma mudança na inclinação antes de ir a zero. Essa mudança de comportamento é interessante porque mostra que a distribuição da memória individual tem papel importante no sistema e que não é trivial supor qual comportamento este apresentará baseado apenas na distribuição de memória. Na Figura 5.7 temos novamente a curva de  $\rho_C$  junto da curva da probabilidade de percolação e mais uma vez encontramos um resultado interessante: o patamar de  $0,5 <$

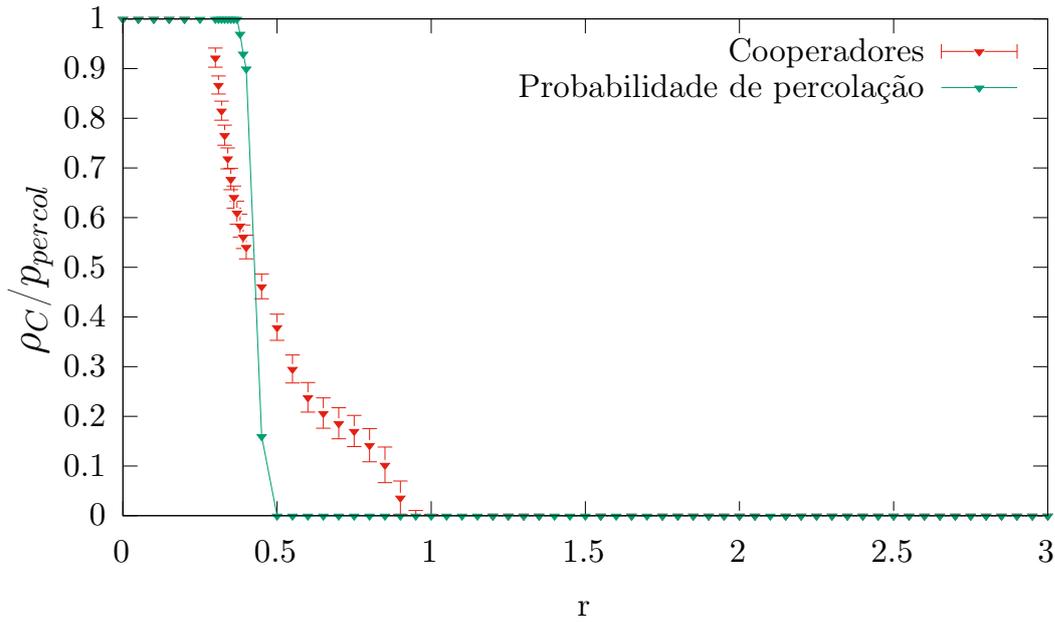


FIGURA 5.7. Fração de cooperadores e probabilidade de percolação em função de  $r$  usando a distribuição de memória individual de lei de potência  $x^{-\tau}$  com  $\tau = 2$ .

$r < 1$  não aparece novamente. Mais que isso, a probabilidade de percolação vai a zero rapidamente conforme a fração de cooperadores começa a cair também. Isso e o fato de a probabilidade de percolação é muito menor que 1 apenas quando  $\rho_C \approx 0,59$ , sugere que aqui os mecanismos envolvidos na dinâmica entre cooperadores e desertores é diferente do que se tem no sistema com as outras distribuições de probabilidade que vimos. Como a única mudança do sistema entre esses casos é a distribuição de memória individual, tomemos então outro valor de  $\tau$ . Como falado antes, a distribuição de lei de potência favorece as memórias individuais mais curtas quando  $\tau$  é grande, vamos diminuir o valor de  $\tau$  para aumentar a frequência de jogadores com memórias mais próximas de  $M$ .

A Figura 5.8 mostra a curva de  $\rho_C$  para a distribuição de lei de potência, mas dessa vez com  $\tau = 1,1$ , ou seja, agora com o sistema tendo mais jogadores com memórias próximas de  $M$ . Agora temos novamente o comportamento mais rico da curva de  $\rho_C$ , com o patamar de  $0,5 < r < 1,0$  presente e um novo patamar facilmente identificado em  $1,0 < r < 1,9$ , que nas outras distribuições poderia ser entendido como uma flutuação das médias do intervalo já que a diferença entre  $\rho_C$  nesse intervalo e  $\rho_C$  para  $r \geq 2,0$  estava dentro do erro da medida. Com a curva de probabilidade de percolação da Figura 5.8 vemos novamente que o patamar de  $0,5 < r < 1,0$  coincidindo com um patamar na probabilidade de percolação.

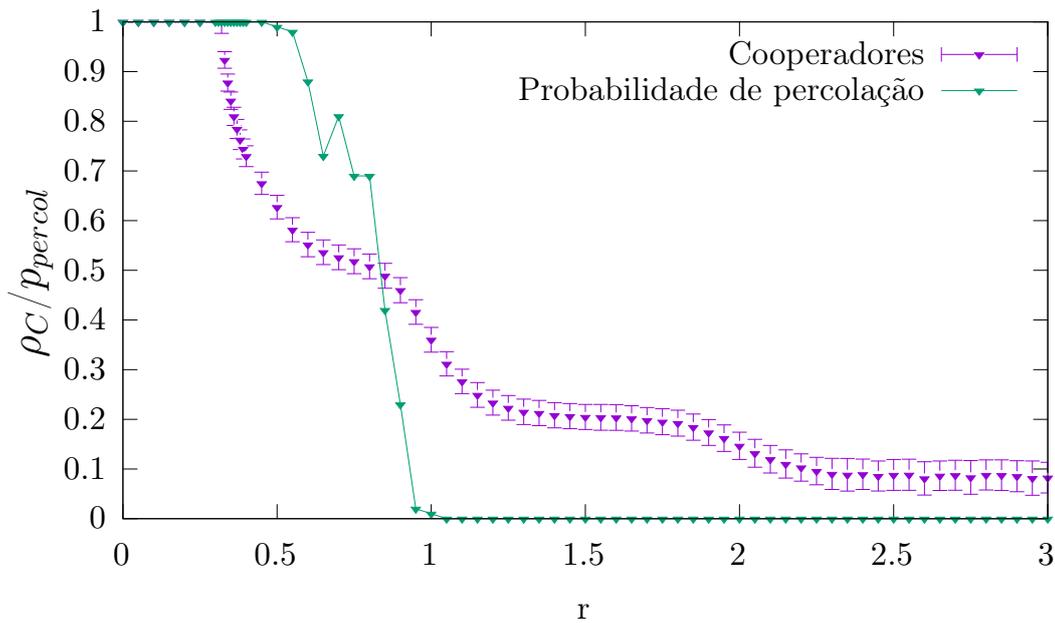


FIGURA 5.8. Fração de cooperadores e probabilidade de percolação em função de  $r$  usando a distribuição de memória individual de lei de potência  $x^{-\tau}$  com  $\tau = 1, 1$ .

Esse último resultado com a distribuição de lei de potência mostra que memórias mais longas trazem uma riqueza de comportamentos maior que o sistema composto por jogadores de pouca memória. Isso está de acordo com os resultados de [34]. Além disso, vimos que a maior diversidade de jogadores com memórias mais curtas e mais longas promove o surgimento de novos comportamentos, esse fato sendo evidenciado pela diferença de  $\rho_C$  entre  $1, 0 < r < 2, 0$  e  $2, 0 < r < 3$  na curva da Figura 5.8. A Figura 5.9 agrega todas curvas no mesmo gráfico para que possam mais facilmente ser comparadas diferenças qualitativas entre essas curvas.

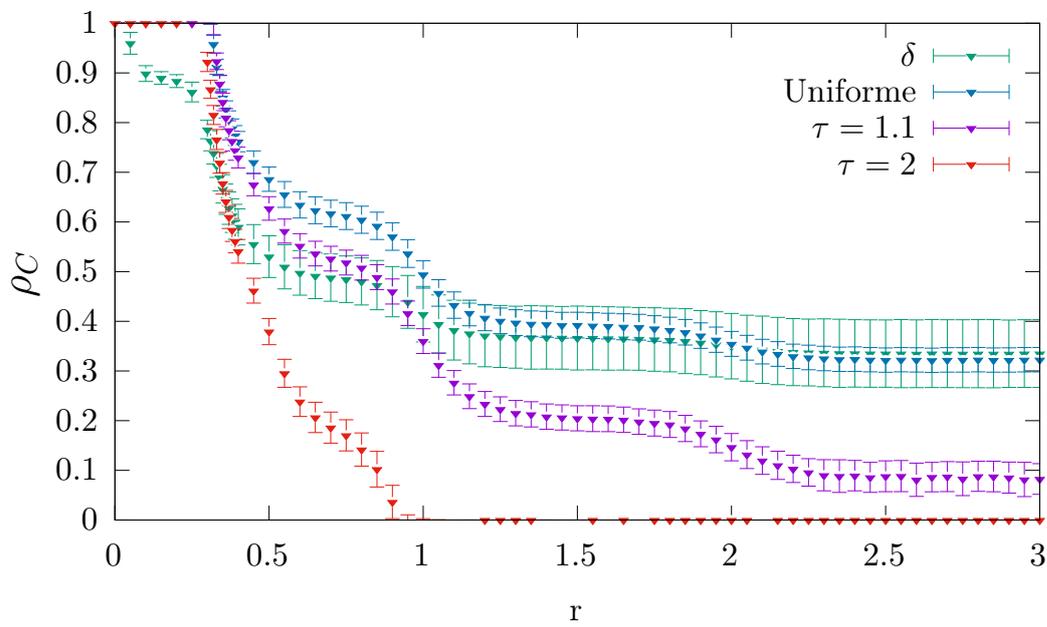


FIGURA 5.9. Fração de cooperadores em função de  $r$  para todas as distribuições estudadas:  $\delta$ , uniforme e lei de potência com  $\tau = 2$  e  $\tau = 1, 1$ .

# Capítulo 6

## Conclusão e perspectivas

Como foi tratado no Capítulo 5, a memória favorece a manutenção da cooperação. Esse dado corrobora resultados da literatura [34]. A inclusão da diversidade dentre os jogadores, representada pela diferença nas capacidades de memória individual, traz comportamentos mais complexos acima de  $r \approx 1,0$ , fato evidenciado pela diferença na curva de  $\rho_C$  entre as distribuições de memória individual uniforme e de lei de potência.

Ainda assim, faz-se necessário investigar mais a fundo quais são os mecanismos presentes no sistema que levam a esses resultados. É possível identificar que a percolação da rede por parte dos cooperadores tem um papel relevante, pelo menos com relação ao patamar da curva de  $\rho_C$  no intervalo  $0,5 < r < 1,0$ . Porém, pelo menos à primeira vista, a probabilidade de percolação não parece trazer informações diretas para entender o que ocorre por exemplo no intervalo de  $1,0 < r < 2,0$  na Figura 5.8, ou mesmo o que mantém a cooperação para valores de tentação tão grandes quanto  $r = 3,0$ . É necessário avaliar quais são as configurações geométricas que o sistema assume no estado estacionário para cada intervalo do parâmetro  $r$ . De maneira geral, o que encontramos foram pelo menos 3 fases nesse intervalo de parâmetros, em oposição às duas fases do modelo clássico que tratamos no capítulo 3: Uma fase dominada por cooperadores, que sem memória e o dinamismo das conexões da rede não existia, uma fase dominada por desertores e pelo menos uma fase onde cooperadores e desertores coexistem.

Para cada distribuição de memória individual restaria investigar quais são as características da(s) fase(s) em que os cooperadores e os desertores coexistem. Como exemplo de análise possível, poderíamos tomar as médias das séries temporais para um dado valor de  $r$  e verificar a distribuição dessas médias para, por exemplo, verificar se a região

$2,0 < r < 3,0$ , presente para todas as distribuições menos a de lei de potência com  $\tau = 2$ , é na verdade uma fase biestável onde o sistema as vezes evolui para um estado com  $\rho_C > 0$  e as vezes para um estado dominado por desertores. Outra verificação possível seria repetir as medidas, mas dessa vez com uma rede maior. Dessa maneira efeitos de tamanho finito seriam atenuados. Como exemplo, podemos olhar novamente a região dos maiores  $r$  dos gráficos de  $\rho_C$ . Com uma rede maior, extinções dos cooperadores naquela região seriam menos numerosas caso a única fase estável fosse de coexistência, enquanto que se a única fase estável nesse intervalo de  $r$  é a dominada por desertores seria de se esperar que  $\rho_C$  tendesse a zero.

Outra característica do modelo proposto que poderia ser estudada extensivamente é a quantidade de ligações ativas na rede. Aqui não foi tratado, mas a noção de *cluster* que vimos na seção 2.2 também pode ser entendida baseada nas ligações entre sítios de uma rede e não somente baseado no estado (cooperador ou desertor no nosso caso) do sítio, e daí as medidas de percolação também podem ser feitas e analisadas, inclusive em conjunto com o número médio de ligações ativas no lugar da fração média de cooperadores como fizemos aqui.

Por fim, evidentemente poderiam ser estudados novos valores que aqui foram escolhidos arbitrariamente. A saber, o valor de  $M = 10$  foi escolhido arbitrariamente de maneira a não causar grande lentidão para gerar as amostras, mas mantendo memórias suficientemente longas no sistema de maneira que observamos os resultados encontrados aqui; Outros valores de  $\tau$ , tanto maiores quanto menores poderiam ser explorados para potencialmente identificar um  $\tau$  “ótimo”, que maximiza a cooperação; e a reputação mínima para que uma ligação entre dois jogadores se mantenha também poderia ter valores diferentes de 0,5, que também foi escolhido arbitrariamente.

# Referências Bibliográficas

- [1] M. A. Nowak, *Evolutionary Dynamics: Exploring the Equations of Life*. Belknap Press, 2006.
- [2] J. von Neumann, O. Morgenstern, and A. Rubinstein, *Theory of Games and Economic Behavior (60th Anniversary Commemorative Edition)*. Princeton University Press, 1944.
- [3] J. Grujić, B. Eke, A. Cabrales, J. A. Cuesta, and A. Sánchez, “Three is a crowd in iterated prisoner’s dilemmas: experimental evidence on reciprocal behavior,” *Scientific Reports*, vol. 2, p. 638, Sep 2012.
- [4] J. Y. Halpern, *Computer Science and Game Theory*, pp. 1–14. London: Palgrave Macmillan UK, 2016.
- [5] L. S. Flores, H. C. Fernandes, M. A. Amaral, and M. H. Vainstein, “Symbiotic behaviour in the public goods game with altruistic punishment,” *Journal of Theoretical Biology*, vol. 524, p. 110737, 2021.
- [6] M. H. Vainstein and J. J. Arenzon, “Spatial social dilemmas: Dilution, mobility and grouping effects with imitation dynamics,” *Physica A: Statistical Mechanics and its Applications*, vol. 394, no. 0, pp. 145 – 157, 2014.
- [7] M. H. Vainstein, C. Brito, and J. J. Arenzon, “Percolation and cooperation with mobile agents: Geometric and strategy clusters,” *Phys. Rev. E*, vol. 90, p. 022132, Aug 2014.
- [8] E. A. Sicardi, H. Fort, M. H. Vainstein, and J. J. Arenzon, “Random mobility and spatial structure often enhance cooperation,” *J. Theor. Biol.*, vol. 256, no. 2, pp. 240–246, 2009.
- [9] M. H. Vainstein and J. J. Arenzon, “Disordered environments in spatial games,” *Phys. Rev. E*, vol. 64, p. 051905, 2001.
- [10] M. H. Vainstein, A. T. C. Silva, and J. J. Arenzon, “Cooperation in diffusive spatial games,” in *Nonequilibrium Statistical Mechanics and Nonlinear Physics* (O. Descalzi, O. A. Rosso, and H. A. Larrondo, eds.), vol. CP913, AIP, 2007.
- [11] M. H. Vainstein, A. T. C. Silva, and J. J. Arenzon, “Does mobility decrease cooperation?,” *J. Theor. Biol.*, vol. 244, pp. 722–728, 2007.
- [12] P. Seabright, *The Company of Strangers: A Natural History of Economic Life (Revised Edition)*. Princeton University Press, stu - student edition ed., 2010.
- [13] M. A. Nowak, “Five rules for the evolution of cooperation,” *Science*, vol. 314, pp. 1560–1563, 2006.

- [14] C. Hauert and G. Szabó, “Game theory and physics,” *American Journal of Physics*, vol. 73, no. 5, pp. 405–414, 2005.
- [15] J. A. Cuesta, C. Gracia-Lázaro, A. Ferrer, Y. Moreno, and A. Sánchez, “Reputation drives cooperative behaviour and network formation in human groups,” *Scientific Reports*, vol. 5, no. 0, p. 7843, 2015.
- [16] M. E. J. Newman and G. T. Barkema, *Monte Carlo Methods in Statistical Physics*. New York: Oxford University Press, 1999.
- [17] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, “Equation of State Calculations by Fast Computing Machines,” *The Journal of Chemical Physics*, vol. 21, pp. 1087–1092, 6 1953.
- [18] S. R. A. Salinas, *Introduction to statistical physics*. Springer, 2001.
- [19] P. H. Mendes, “Aprendizado não supervisionado em sistemas de spins,” 2022.
- [20] Repositório no GitHub de pedhmendes: [github.com/pedhmendes](https://github.com/pedhmendes).
- [21] D. Stauffer and A. Aharony, *Introduction To Percolation Theory*. 1992.
- [22] L. S. Flores, M. A. Amaral, M. H. Vainstein, and H. C. Fernandes, “Cooperation in regular lattices,” *Chaos, Solitons & Fractals*, vol. 164, p. 112744, 2022.
- [23] T. Fricke, “The hoshen-kopelman algorithm,” tech. rep., 2004.
- [24] D. W. H. Kurt Binder, *Monte Carlo Simulation in Statistical Physics*. Springer Cham, 6 ed., 2019.
- [25] J. Hoshen and R. Kopelman, “Percolation and cluster distribution. i. cluster multiple labeling technique and critical concentration algorithm,” *Phys. Rev. B*, vol. 14, pp. 3438–3445, Oct 1976.
- [26] S. Weis and M. Schröter, “Analyzing x-ray tomographies of granular packings,” *Review of Scientific Instruments*, vol. 88, p. 051809, 05 2017.
- [27] M. E. J. Newman and R. M. Ziff, “Efficient monte carlo algorithm and high-precision results for percolation,” *Physical Review Letters*, vol. 85, pp. 4104–4107, nov 2000.
- [28] M. A. Nowak and R. M. May, “Evolutionary games and spatial chaos,” *Nature*, vol. 246, pp. 15–18, 1992.
- [29] S. K. Baek, H.-C. Jeong, C. Hilbe, and M. A. Nowak, “Comparing reactive and memory-one strategies of direct reciprocity,” *Scientific Reports*, vol. 6, p. 25676, May 2016.
- [30] R. Axelrod, *The Evolution of Cooperation*. New York: BasicBooks, 1984.
- [31] M. A. Amaral, L. Wardil, M. Perc, and J. K. L. da Silva, “Stochastic win-stay-lose-shift strategy with dynamic aspirations in evolutionary social dilemmas,” *Physical Review E*, vol. 94, sep 2016.
- [32] B. A. Huberman and N. S. Glance, “Evolutionary games and computer simulations.,” *Proceedings of the National Academy of Sciences*, vol. 90, no. 16, pp. 7716–7718, 1993.
- [33] O. Mazzarisi, A. de Azevedo-Lopes, J. J. Arenzon, and F. Corberi, “Maximal diversity and zipf’s law,” *Phys. Rev. Lett.*, vol. 127, p. 128301, Sep 2021.
- [34] A. J. Stewart and J. B. Plotkin, “Small groups and long memories promote cooperation,” *Scientific Reports*, vol. 6, p. 26889, Jun 2016.