

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
INSTITUTO DE INFORMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

RICARDO MATSUMURA DE ARAÚJO

**Aprendizado de máquina em sistemas  
complexos multiagentes: estudo de caso em  
um ambiente sob racionalidade limitada**

Dissertação apresentada como requisito parcial  
para a obtenção do grau de  
Mestre em Ciência da Computação

Dr. Luís C. Lamb  
Orientador

Porto Alegre, dezembro de 2004

## CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Araújo, Ricardo Matsumura de

Aprendizado de máquina em sistemas complexos multiagentes: estudo de caso em um ambiente sob racionalidade limitada / Ricardo Matsumura de Araújo. – Porto Alegre: PPGC da UFRGS, 2004.

83 f.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR-RS, 2004. Orientador: Luís C. Lamb.

1. Aprendizado de máquina. 2. Minority game. 3. Sistemas complexos. I. Lamb, Luís C.. II. Título.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. José Carlos Ferraz Hennemann

Vice-Reitor: Prof. Pedro Cezar Dutra Fonseca

Pró-Reitora Adjunta de Pós-Graduação: Prof<sup>a</sup>. Valquiria Linck Bassani

Diretor do Instituto de Informática: Prof. Philippe Olivier Alexandre Navaux

Coordenador do PPGC: Prof. Flávio Rech Wagner

Bibliotecária-Chefe do Instituto de Informática: Beatriz Regina Bastos Haro

*“A Economia é mesmo uma grande tentativa. Trata das realidades mais reais do mundo, que são os apetites humanos, e das mais abstratas, como o valor do trabalho, do tempo e das coisas. Tenta tratar o impoderável com precisão, mas sempre deixa claro que a sua ciência só vai até onde começa o mistério do comportamento humano.”*

— LUIS FERNANDO VERISSIMO  
*Comédias da Vida Pública*

# SUMÁRIO

<b>LISTA DE ABREVIATURAS E SIGLAS</b> . . . . .	6
<b>LISTA DE FIGURAS</b> . . . . .	7
<b>LISTA DE TABELAS</b> . . . . .	9
<b>RESUMO</b> . . . . .	10
<b>ABSTRACT</b> . . . . .	11
<b>1 INTRODUÇÃO</b> . . . . .	12
<b>2 APRENDIZADO DE MÁQUINA</b> . . . . .	14
<b>2.1 Introdução</b> . . . . .	14
<b>2.2 Paradigmas de aprendizado</b> . . . . .	15
2.2.1 Aprendizado por Indução . . . . .	15
2.2.2 Aprendizado por reforço . . . . .	16
2.2.3 Aprendizado baseado em instâncias . . . . .	16
2.2.4 Aprendizado evolutivo . . . . .	17
<b>2.3 Estruturas de aprendizado</b> . . . . .	20
2.3.1 Regras de produção . . . . .	20
2.3.2 Redes Neurais . . . . .	21
2.3.3 Árvores de decisão . . . . .	23
<b>3 SISTEMAS COMPLEXOS</b> . . . . .	25
<b>3.1 Sistemas, emergência e auto-organização</b> . . . . .	25
<b>3.2 Sistemas complexos e computação</b> . . . . .	27
<b>4 RACIONALIDADE E APRENDIZADO EM SISTEMAS ECONÔMICOS</b> . . . . .	30
<b>4.1 Introdução</b> . . . . .	30
<b>4.2 Racionalidade Limitada</b> . . . . .	34
<b>4.3 Aprendizado e economia</b> . . . . .	36
<b>4.4 O problema do bar El Farol</b> . . . . .	38
<b>5 MINORITY GAME</b> . . . . .	40
<b>5.1 Introdução</b> . . . . .	40
<b>5.2 Definição do MG</b> . . . . .	41
<b>5.3 Dinâmica do MG</b> . . . . .	43

<b>6</b>	<b>ANÁLISE DO APRENDIZADO NO MINORITY GAME</b>	47
6.1	Um modelo básico de aprendizado criativo	50
6.2	Metodologia	51
6.3	Experimentos	52
6.3.1	Efeitos sobre $A(t)$	52
6.3.2	Efeitos de $\tau$	53
6.3.3	Efeitos de $p_m$	54
6.3.4	Efeitos da atribuição de <i>fitness</i>	56
6.3.5	Efeitos da seleção	56
6.3.6	Comportamento das estratégias	57
6.4	Discussão dos resultados	62
<b>7</b>	<b>ANÁLISE DE AGENTES NO MINORITY GAME</b>	65
7.1	Introdução	65
7.2	Metodologia	66
7.3	Experimentos	66
7.3.1	Influência da memória	66
7.3.2	Influência de $p_m$	72
7.3.3	Influência de $\tau$	73
7.4	Discussão dos resultados	74
<b>8</b>	<b>CONCLUSÃO E TRABALHOS FUTUROS</b>	76
	REFERÊNCIAS	79

## LISTA DE ABREVIATURAS E SIGLAS

MG	Minority Game
ACO	Ant Colony Optimizarion
PSO	Particle Swarm Optimization
COIN	Collective Intelligence
AG	Algoritmo Genético
AE	Algoritmo Evolutivo
M	Tamanho da memória dos agentes em um jogo do MG
N	Número de agentes em um jogo do MG
S	Número de estratégias possuídas por cada agente no MG
$\sigma^2$	Variância

## LISTA DE FIGURAS

Figura 2.1:	Diagrama esquemático de uma operação de <i>crossover</i> . . . . .	19
Figura 2.2:	Exemplo de uma rede neural . . . . .	22
Figura 2.3:	Exemplo de uma árvore de decisão . . . . .	23
Figura 5.1:	Diagrama esquemático do MG . . . . .	41
Figura 5.2:	$A(t)$ para diversos valores de $M$ . $M = 2$ (gráfico superior); $M = 6$ (gráfico central); $M = 10$ (gráfico inferior) . . . . .	44
Figura 5.3:	Variância em função de $M$ para o MG tradicional com $S = 2$ e $N = 101$ . . . . .	46
Figura 6.1:	$A(t)$ no MG tradicional (esquerda) e $A(t)$ utilizando-se o algoritmo de aprendizado criativo com $\tau = 1.0$ e $p_m = 0.3$ (direita). . . . .	53
Figura 6.2:	Variância em função de $M$ para o MG utilizando aprendizado criativo. Círculos são para $\tau = 0$ e quadrados para $\tau = 1$ e $p_m = 0.5$ . . . . .	54
Figura 6.3:	Variância em função de $M$ para $\tau = 0.01$ , $\tau = 0.1$ , $\tau = 0.3$ e $\tau = 0.5$ (respectivamente: quadrados pretos, círculos pretos, quadrados brancos e círculos brancos). $p_m = 0.5$ . . . . .	55
Figura 6.4:	Variância em função de $p_m$ . $M = 7$ , $S = 2$ , $\tau = 1.0$ . . . . .	55
Figura 6.5:	Variância em função de $M$ utilizando o algoritmo modificado onde novas estratégias iniciam com <i>fitness</i> zero, $p_m = 0.5$ e $\tau = 1$ . . . . .	56
Figura 6.6:	Variância em função de $M$ utilizando algoritmos modificados de aprendizagem. Quadrados representam o algoritmo modificando as melhores estratégias enquanto círculos representam escolhas aleatórias de estratégias para serem modificadas. . . . .	57
Figura 6.7:	Espaço de estratégias na primeira rodada do jogo (gráfico superior) e espaço de estratégias na última rodada do jogo (gráfico inferior). Cada marca representa a melhor estratégia de cada agente, convertida para uma representação decimal. . . . .	58
Figura 6.8:	Estratégia optada por um agente em cada rodada de uma simulação. . . . .	59
Figura 6.9:	Variância do sistema <i>versus</i> o número de agentes congelados para diversas simulações com diferentes inicializações. . . . .	60
Figura 6.10:	Variância do sistema (círculos) e a fração de agentes congelados (quadrados) em função de $M$ . . . . .	60
Figura 6.11:	Evolução da fração de agentes congelados. $M = 3$ , $\tau = 1.0$ . . . . .	61
Figura 6.12:	<i>Fitness</i> médio dos agentes congelados (círculos) e dos agentes não-congelados (quadrados) para vários valores de $M$ . . . . .	61

Figura 6.13: Evolução da distância entre as estratégias utilizadas em um jogo, quando todas estratégias iniciam no mesmo ponto (i.e. idênticas). $M = 3$ . . . . .	63
Figura 7.1: Regimes a serem estudados com ambiente tradicional. . . . .	67
Figura 7.2: Regimes a serem estudados com ambiente criativo. . . . .	69
Figura 7.3: Performance do agente-alvo em função de $pm_a$ . Para o ambiente, $p_m = 0$ e $M = 2$ . Linha tracejada indica a média de performance dos agentes no ambiente. . . . .	73
Figura 7.4: Performance do agente-alvo em função de $\tau$ (linha contínua). Para o ambiente, $\tau = 0$ e $M = 2$ . Linha tracejada indica a média de performance dos agentes no ambiente. . . . .	74

## LISTA DE TABELAS

Tabela 4.1:	Exemplo de jogo na sua forma normal. . . . .	33
Tabela 4.2:	Jogo pedra-papel-tesoura na forma normal. . . . .	33
Tabela 4.3:	O Dilema do Prisioneiro em sua forma normal. . . . .	34
Tabela 5.1:	Exemplo de estratégia com $M = 3$ . . . . .	43
Tabela 6.1:	Tempo de execução de $T = 5000$ rodadas em função de $M$ para a implementação do algoritmo utilizada. $N = 101$ e $S = 2$ . . . . .	52
Tabela 7.1:	Resultados com ambiente e agente-alvo tradicionais. . . . .	67
Tabela 7.2:	Resultados com ambiente e agente-alvo criativos. . . . .	69
Tabela 7.3:	Resultados com ambiente tradicional e agente-alvo criativo. . . . .	71
Tabela 7.4:	Resultados com ambiente criativo e agente-alvo tradicional. . . . .	72

## RESUMO

O presente trabalho investiga a relação entre aprendizado e dinâmica em sistemas complexos multiagentes. Fazemos isso através de estudos experimentais em um cenário de racionalidade limitada que situa-se na interseção entre Inteligência Artificial, Economia e Física Estatística, conhecido como “Minority Game”. Apresentamos resultados experimentais sobre o jogo focando o estudo do cenário sob uma perspectiva de Aprendizado de Máquina. Introduzimos um novo algoritmo de aprendizado para os agentes no jogo, que chamamos de aprendizado criativo, e mostramos que este algoritmo induz uma distribuição mais eficiente de recursos entre os agentes. Este aumento de eficiência mostra-se resultante de uma busca irrestrita no espaço de estratégias que permitem uma maximização mais eficiente das distâncias entre estratégias. Analisamos então os efeitos dos parâmetros deste algoritmo no desempenho de um agente, comparando os resultados com o algoritmo tradicional de aprendizado e mostramos que o algoritmo proposto é mais eficiente que o tradicional na maioria das situações. Finalmente, investigamos como o tamanho de memória afeta o desempenho de agentes utilizando ambos algoritmos e concluímos que agentes individuais com tamanhos de memória maiores apenas obtém um aumento no desempenho se o sistema se encontrar em uma região ineficiente, enquanto que nas demais fases tais aumentos são irrelevantes - e mesmo danosos - à performance desses agentes.

**Palavras-chave:** Aprendizado de máquina, minority game, sistemas complexos.

## **Machine Learning in multi-agent complex systems: a case study in a bounded rationality scenario**

### **ABSTRACT**

This work investigates the relationship between learning and dynamics in complex multiagent systems. We do so by means of experimental studies in a bounded rationality scenario which lays at the intersection of Artificial Intelligence, Economics and Statistical Physics known as the “Minority Game”. We present experimental results about the game aiming at studying such scenario under a Machine Learning perspective. We introduce a new learning algorithm for agents in the Minority Game, namely creative learning, and show that the algorithm renders a more efficient distribution of resources among agents. This increase in efficiency is shown to be a result of an unrestricted search in the strategies space which allows for an efficient maximisation of the distance between agents strategies. We then analyse the effects of this algorithm’s parameters in the performance of an agent, comparing the results with the traditional learning algorithm, concluding that the proposed algorithm is more efficient than the traditional one in most situations. Finally, we investigate how memory size affects agent’s performance using both algorithms, showing that individual agents with larger memory sizes achieve better performances only when the system as a whole is at an inefficient phase, while in other phases such increases are irrelevant - and even harmful - to the performance of the agents.

**Keywords:** minority game,machine learning,complex systems.

# 1 INTRODUÇÃO

Que dinâmica surge da interação de múltiplos indivíduos? Como capacidades individuais de aprendizado influenciam a dinâmica dos sistemas que estes compõem? Tais perguntas não são novas, tampouco foram respondidas definitivamente. O estudo dos problemas envolvidos na resposta destas questões tem ganho renovado interesse nos últimos anos, em parte devido a crescente necessidade em analisar-se sistemas sociais, principalmente econômicos, de forma mais correta ou precisa e em parte devido ao relativamente novo ramo de pesquisa em inteligência de enxame <sup>1</sup>.

Uma das abordagens mais comuns para o estudo de sistemas complexos é partir de um modelo bastante simples e avaliar seu comportamento para diversos parâmetros ou variações do mesmo, na busca de algum padrão de comportamento que expliquem o fenômeno real (GILBERT; CONTE, 1995). O *Minority Game* (ZHANG; CHALLET, 1997) é um desses modelos, onde iterativamente múltiplos agentes independentemente optam por estar em um entre dois grupos e, ao final das escolhas, todos que optaram por estar no grupo que contém o menor número de agentes são recompensados. Os agentes podem unicamente utilizar um limitado número de resultados de rodadas anteriores para decidir na próxima rodada e, para tanto, devem *aprender* estratégias que garantam recompensas durante as jogadas.

Este modelo é tido como um proto-modelo para alguns tipos de mercados (ZHANG, 1998), mas a mais simples analogia é a de múltiplos carros que devem optar por uma dentre duas rodovias que levam a um mesmo lugar (BAZZAN et al., 2000). Todos desejam estar na menos movimentada e, efetivamente, aqueles que estiverem na rodovia com menor número de carros terá seu trajeto facilitado. A questão é como devem os agentes envolvidos decidir que rodovia tomar? Se essas decisões devem ser tomadas periodicamente, todo dia digamos, e se os agentes podem levar em conta resultados passados, que padrão emerge em cada rodovia?

O *Minority Game* tem recebido grande atenção de físicos e economistas, tendo um papel de destaque na crescente área de *econofísica* (FARMER, 1999). O presente trabalho tem a intenção de estudar tal jogo do ponto de vista de aprendizado de máquina. O principal motivo para se analisar algoritmos de aprendizagem em um ambiente como o *Minority Game* é o fato de este ser um ambiente com características bastante diferentes

---

<sup>1</sup>Swarm intelligence

daquelas onde usualmente esses algoritmos são aplicados: enquanto a maior parte da literatura de aprendizado de máquina lida com o aprendizado de padrões em ambientes externos, onde o próprio algoritmo não tem influência, em ambientes como o *Minority Game* o algoritmo e sua interação com outros é o próprio gerador dos padrões que se tenta capturar. Para estudar tanto os padrões gerados como os algoritmos que os geram, realizamos dois estudos utilizando o *Minority Game* como base.

Inicialmente, dotamos agentes no *Minority Game* com um algoritmo de aprendizagem ligeiramente mais complexo que o originalmente oferecido e comparamos a influência desta alteração no comportamento do sistema. Este algoritmo de aprendizado mantém um fluxo constante de novas estratégias no jogo, divergindo do modelo tradicional onde os agentes possuem um conjunto fixo de estratégias. É nossa intenção traçar paralelos entre os parâmetros do algoritmo de aprendizagem proposto e as propriedades ditas *emergentes* do sistema. Mostramos que para uma ampla faixa de parâmetros o algoritmo mostra-se mais eficiente que o tradicionalmente utilizado.

Em um segundo momento, alteramos o alvo de estudo e passamos a tentar *especificar* um agente que seja vencedor no jogo. Para tanto, realizamos diversas simulações utilizando diferentes tipos de agentes em jogos com diferentes parâmetros e extraímos características desejáveis ao se especificar um agente para jogar o *Minority Game*. Em particular, veremos que a crença de que agentes com mais memória se saem melhor no jogo é apenas parcialmente verdade e dependente do *regime* em que se encontra o sistema como um todo.

Esses estudos têm ligação com o crescente interesse em inteligência de enxame, que faz uso de sistemas massivamente multiagente para lidar com problemas diversos. Particularmente, há forte conexão com a recente área de estudos em *Inteligência Coletiva* (COIN - Collective Intelligence) (WOLPERT; TUMER, 2001), onde a otimização de determinada função global é realizada distribuindo-a em funções locais a serem otimizadas por múltiplos agentes em interação. Apesar de aqui não haver o interesse em resolver problemas específicos, o estudo que segue é de interesse para essas áreas, podendo inclusive tornar-se central em algumas aplicações.

Neste trabalho fazemos conexões entre inteligência de enxames, sistemas complexos e sistemas econômicos, sendo o aprendizado de máquina o elo entre essas áreas. Estudamos as propriedades e comportamentos emergentes do *Minority Game*, mas focando o que acontece no nível do sistema de aprendizado e, de forma inversa, estudamos alterações no sistema de aprendizado dos agentes e suas influências no sistema como um todo.

Os capítulos 1 e 2 são introdutórios, realizando uma revisão, respectivamente, sobre aprendizado de máquina e sistemas complexos. É nossa intenção definir conceitos e termos que serão utilizados nos demais capítulos. No capítulo 3 estabelecemos o vínculo entre aprendizado, economia e racionalidade em jogos econômicos, de forma a contextualizar o problema que está sendo abordado. O capítulo 4 define o *Minority Game* e revisa a literatura sobre este jogo. Nos capítulos 5 e 6 descrevemos os estudos propostos para, no capítulo 7, concluirmos.

## 2 APRENDIZADO DE MÁQUINA

Uma área de conhecimento central no presente trabalho é a de Aprendizado de Máquina. Neste capítulo, fazemos uma coletânea de alguns tópicos relevantes para o trabalho desta área. É nossa intenção definir vocabulário e conceitos para serem utilizados nas seções seguintes.

### 2.1 Introdução

Aprendizado de Máquina é o ramo da Inteligência Artificial em que estuda-se e constrói-se sistemas capazes de *aprender* com a experiência. Isto é, um sistema que realiza determinada tarefa deve poder obter informações que permitam que realizações futuras desta tarefa tenham melhor performance. Mais formalmente, Mitchell (MITCHELL, 1997) define a tarefa de aprendizado por um algoritmo como:

Um programa de computador *aprende* a partir da experiência  $E$  com respeito a alguma classe de tarefas  $T$  e medida de performance  $P$ , se sua performance em tarefas de  $T$ , medidas por  $P$ , melhora com a experiência  $E$ .

Ter sistemas capazes de aprender e melhorar sua própria performance é central na Inteligência Artificial e de grande utilidade prática em sistemas cotidianos e futuros. Em particular, são especialmente úteis pelos seguintes motivos (NILSSON, 1996):

- Algumas tarefas não podem ser bem definidas exceto através de exemplos. Desejamos criar sistemas capazes de aprender as relações contidas nos exemplos e generalizar para outras instâncias do problema;
- Em grandes quantidades de dados podem haver relações desconhecidas que se deseja explicitar. Desejamos criar sistemas capazes de revelar tais relações (*data mining*);
- Certas características do ambiente em que o sistema será utilizado podem não ser conhecidas durante o projeto e implementação do sistema. Um sistema capaz de adaptar-se ao ambiente tem potencialmente maior capacidade de obter melhor eficiência;

- A quantidade de conhecimentos disponível para determinada tarefa pode ser excessivamente grande para serem codificadas explicitamente no sistema. Sistemas que aprendam estes conhecimentos de forma automática tornam-se necessários;
- Ambientes se alteram com o tempo. Sistemas que se readaptam às mudanças exigem menor esforço de manutenção;

Podemos adicionar a esta lista, ainda, um novo item: a necessidade de *simulação* de sistemas de aprendizado. Neste caso, o Aprendizado de Máquina tem a função não de resolver um problema específico de forma eficiente, mas sim de fornecer ferramentas para simulação de um outro sistema de aprendizado já existente como, por exemplo, o aprendizado humano. As questões envolvidas para desenvolver algoritmos para ambas abordagens são semelhantes, porém ao simularmos sistemas de aprendizagem estamos interessados não somente em eficiência e eficácia, mas em coerência e similaridade com o sistema sendo simulado. Apesar de serem as mesmas técnicas utilizadas nos dois casos, as formas de avaliação são bastante distintas. Este é um ramo de pesquisa relacionado ao proposto por Valiant (VALIANT, 2003): o estudo de algoritmos para computar as tarefas mais básicas de comportamento humano.

Três questões estão no centro do estudo de técnicas de aprendizado de máquina: (i) formas de representação do conhecimento, (ii) algoritmos para manipulação de conhecimento e (iii) formas de aquisição de novo conhecimento. Classificações de técnicas podem ser feitas a partir destas questões. Em particular, (MICHALSKI; CARBONELL; MITCHELL, 1983) propõe uma taxonomia baseada nas estruturas utilizadas para representação do conhecimento e uma sub-classe desta será utilizada para categorizar e melhor apresentar os algoritmos. Não há o interesse em exaurir as possíveis classes, mas sim apresentar de forma contextualizada as que são de interesse para introduzir conceitos ao nosso estudo.

## 2.2 Paradigmas de aprendizado

Dentro de cada uma das classes acima podemos separar as técnicas pela forma como essas estruturas são manipuladas. Porém, aqui é importante distinguir entre alguns *paradigmas de aprendizado*, isto é, formas com que novo conhecimento é adquirido. Três paradigmas serão mencionados aqui: *aprendizado por indução*, *aprendizado baseado em instâncias* e *aprendizado por reforço*. Vale notar que um paradigma, exceto em sua forma extrema, não constitui uma abordagem única a um problema e frequentemente temos problemas que englobam dois ou mais paradigmas simultaneamente.

### 2.2.1 Aprendizado por Indução

O aprendizado por indução é também conhecido como aprendizado através de exemplos (CARBONELL, 1990)(MICHALSKI; CARBONELL; MITCHELL, 1983). O objetivo deste tipo de aprendizado é aprender um conceito geral a partir de uma seqüência de instâncias do conceito (exemplos) e, possivelmente, alguns contra-exemplos. Nesta

forma, o algoritmo deve ser capaz de gerar uma hipótese de conceito que possibilite derivar todos os exemplos e nenhum contra-exemplo. Os exemplos são chamados “instâncias positivas”, enquanto os contra-exemplos são “instâncias negativas”. Mais formalmente, um algoritmo de aprendizado por indução deve ser capaz de aprender um conceito  $c$  a partir da apresentação de instâncias na forma  $(x, c(x))$  e  $(x, \neg c(x))$ , onde  $x$  é a instância em si e  $c(x)$  a aplicação do conceito sobre essa instância. Por exemplo,  $c$  pode ser uma função booleana que classifique instâncias segundo algum critério.

O aprendizado por indução se baseia na seguinte hipótese (MITCHELL, 1997):

Qualquer hipótese que aproxime a função alvo bem o suficiente para um grande conjunto de exemplos de treinamento irá também aproximar a função alvo bem o suficiente para instâncias não observadas.

Em (MITCHELL, 1997) também se salienta que o aprendizado por indução é basicamente uma busca por hipóteses que se encaixem nos exemplos fornecidos. Visto dessa forma, o espaço de busca é bastante grande mesmo para conceitos simples e heurísticas para encontrar hipóteses tornam-se essenciais.

### 2.2.2 Aprendizado por reforço

Métodos de aprendizado por reforço estão interessados em resolver problemas de *decisão seqüencial* através de experimentação com o ambiente (MORIARTY; SCHULTZ; GREFFENSTETTE, 1999). Neste tipo de problema, o método deve construir uma seqüência de ações que maximize um determinado ganho fornecido pelo ambiente. Este ganho pode vir em qualquer momento, comumente apenas após uma longa seqüência de ações. O grande problema deste tipo de aprendizado é, portanto, como creditar o ganho às ações realizadas. Problemas deste tipo são chamados “ganho atrasado<sup>1</sup>”. Como exemplo, considere um jogo de damas. Não é possível creditar ganhos individuais a cada jogada efetuada, pois é praticamente impossível saber se uma certa seqüência de jogadas levará à vitória ou não até que o jogo efetivamente termine. Assim, o crédito só é efetuado ao fim do jogo e é tarefa do algoritmo de aprendizagem aprender quais jogadas foram efetivamente úteis (ou ruins) para o resultado final e desenvolver uma seqüência ótima de jogadas que levem à maximização dos ganhos.

O fato de o ambiente fornecer o resultado das ações torna imprescindível que o agente possa *experimentar* o ambiente. Estes experimentos podem ser tanto físicos, onde o agente efetivamente atua sobre o ambiente real, como simulados, onde uma simulação do ambiente é executada exclusivamente para experimentações.

### 2.2.3 Aprendizado baseado em instâncias

A técnica de aprendizado baseado em instâncias pode ser vista como uma técnica indutiva onde a indução ocorre na *aplicação* do sistema e não no seu *treinamento*. Não há o interesse em formular o modelo para então aplicá-lo, mas sim em construir um modelo a cada aplicação. Desta forma, não há armazenagem do modelo, mas sim dos exemplos

---

<sup>1</sup>*Delayed reward*

como eles são apresentados.

Esta técnica, ao ser aplicada para tomar uma decisão, utiliza os exemplos armazenados para classificar o novo objeto. Isto permite que um modelo diferente seja construído a cada apresentação de um novo objeto, diferindo do aprendizado por indução onde, após o modelo ter sido induzido, este é aplicado para todas novas instâncias apresentadas. Como consequência, é possível classificar novas instâncias apenas baseando-se em exemplos “próximos” a este, sem haver a preocupação em generalização para futuras instâncias. Há vantagens neste método para casos onde o conceito a ser aprendido é bastante complexo, mas pode ser descrito como uma coleção de aproximações locais (MITCHELL, 1997). Outra característica é a transferência do custo de processamento do estágio de treinamento para o de classificação.

Usualmente, a classificação de uma nova instância é feita por um método de vizinhança. Assim, realiza-se a comparação do novo objeto com os armazenados de modo a determinar a distância entre eles, realizando a classificação de acordo com a menor distância encontrada. Para alguns problemas, o conceito a ser aprendido pode depender de apenas alguns atributos representados nos exemplos mas, como o cálculo da distância leva em conta *todos* os atributos, podem haver classificações errôneas. Formas de contornar estes problemas tornam-se necessárias para uma aplicação mais geral do método.

#### **2.2.4 Aprendizado evolutivo**

Apesar de alguns autores classificarem aprendizado evolutivo como um paradigma de aprendizado (e.g. (CARBONELL, 1990)), no mesmo nível dos demais paradigmas citados anteriormente, é difícil classificá-lo desta forma uma vez que o aprendizado evolutivo pode assumir diversas formas e fazer uso de indução, instâncias ou reforço. É, portanto, mais um modelo geral de algoritmo do que um paradigma efetivamente. De qualquer forma, nesta seção apresentamos alguns conceitos relevantes ligados ao aprendizado evolutivo, uma vez que este será importante nas seções que seguem.

Aprendizado evolutivo é uma forma de aprendizado baseada em algoritmos evolutivos que, por sua vez, constituem uma classe de algoritmos utilizados para otimização. Algoritmos evolutivos fazem parte de uma classe mais geral de algoritmos de otimização, os chamados algoritmos estocásticos adaptativos e o termo “Algoritmo Evolutivo” surgiu para abranger uma série de propostas de algoritmos desenvolvidas de forma independente, mas com conceitos semelhantes como a analogia com evolução natural ou genética natural, verificação de múltiplos pontos paralelamente e utilização de operadores adaptativos.

Os Algoritmos Evolutivos (AE) possuem quatro ramos principais: Algoritmos Genéticos, Estratégias Evolutivas, Programação Genética e Programação Evolutiva. Hoje, cada ramo é visto como tentativas de explorar aspectos diferentes dos Algoritmos Evolutivos, porém atualmente a distinção entre estes é mais histórica do que prática, principalmente devido ao grande intercâmbio de informações entre as áreas, o que tornou os limites difusos (FOGEL, 2000). Todos possuem uma base comum de operação e exploram aspectos específicos, como inclusão de operadores, formas de codificação, método de seleção, etc. Apesar do uso das nomenclaturas individuais para cada classe ser cada vez

menos útil, no meio científico é comum se valer dessa classificação, mesmo quando há clara intersecção entre as classes. A classe dos Algoritmos Genéticos, particularmente, parece estar englobando as demais e se tornando o termo cada vez mais comum para designar um AE.

A idéia por trás de um AE é a de utilizar conceitos da evolução natural e genética para manipular codificações de possíveis soluções para um problema de otimização. Em um AE, pontos no espaço de soluções são explorados por *indivíduos* e cada indivíduo é composto por um vetor de componentes no domínio do problema, constituindo uma possível solução, e uma respectiva avaliação desta solução. Esta avaliação é chamada comumente de *fitness* e é feita por uma função externa, composta da função que se deseja otimizar (podendo haver restrições) e de possíveis funções auxiliares (heurísticas) que auxiliem na busca. Indivíduos integram uma *população* que nada mais é, portanto, que um conjunto de possíveis soluções para o problema.

Indivíduos podem ser representações diretas de soluções para o problema ou podem codificar de forma indireta essas soluções. Quando há codificação indireta, diz-se que o indivíduo representa um *genótipo* que é decodificado para tornar-se um *fenótipo*, que é então avaliado. Por exemplo, números reais podem ser codificados como *strings* de bits. Os bits são o genótipo do indivíduo e os números reais representados pelos bits são seu fenótipo.

Podemos resumir um algoritmo evolutivo geral nas seguintes etapas (ARAÚJO, 2004; FOGEL, 2000):

- Geração de uma *população* de soluções;
- *Seleção* de bons indivíduos;
- *Alteração* dos selecionados;

Inicializa-se a população, usualmente, com indivíduos gerados aleatoriamente e cobrindo de forma uniforme o espaço de soluções. As iterações do algoritmo tem por objetivo modificar a posição destes indivíduos de forma que se encontre um ponto ótimo neste espaço. Para tanto, utiliza-se o conceito da evolução: indivíduos ruins tem mais chances de serem descartados e darem lugar a variações dos melhores indivíduos. Essas variações são efetuadas por *operadores genéticos*, que modificam o genótipo dos indivíduos na população. A forma de seleção, o tipo de operador utilizado e outros parâmetros específicos variam para cada ramo dos AE.

Em um Algoritmo Genético (AG) (GOLDBERG, 1989), por exemplo, utilizam-se dois operadores básicos: *crossover* e mutação. O operador de *crossover* toma dois indivíduos (chamado *pais*) e “mistura” o código genético de ambos, isto é, constrói novos indivíduos compostos por partes da solução contida nos pais. Por exemplo, se indivíduos são compostos por *strings* de números reais (que poderiam ser coeficientes em uma função), o operador de *crossover* poderia quebrar em duas partes cada uma das *strings* dos pais e formar um novo indivíduo tomando a primeira metade da *string* de um dos pais e a

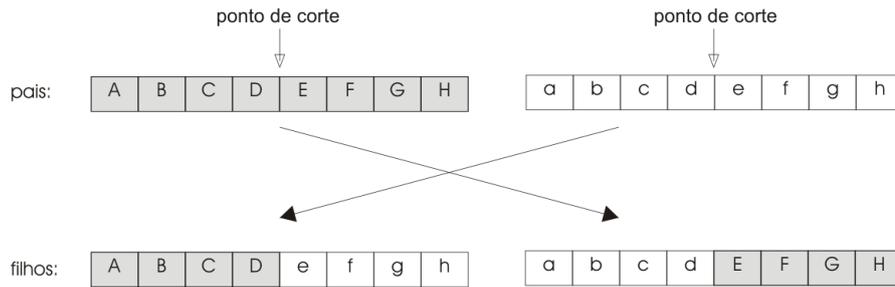


Figura 2.1: Diagrama esquemático de uma operação de *crossover*

segunda metade da *string* do outro. De fato, a operação descrita é o mais comum tipo de *crossover* e é chamada de “*crossover* de um ponto” (ver Fig. 2.1)

O operador de mutação, por outro lado, não necessita dois indivíduos. Ele toma um único indivíduo e realiza algum tipo de modificação aleatória em seu código genético. Se o indivíduo for composto por *bits*, a mutação pode inverter algum de seus bits. No caso de uma *string* de valores reais, pode-se substituir alguns valores por outros ou adicionar algum tipo de ruído gaussiano em todos valores.

É o objetivo da mutação assegurar que todo o espaço de soluções tenha chance de ser explorado, enquanto o *crossover* tem o objetivo de transmitir informações entre indivíduos (GOLDBERG, 1989). Porém, esta distinção é freqüentemente questionada e alguns estudos mostram que o *crossover* é nada mais que um operador de mutação em larga escala (FOGEL, 2000).

Um AG típico utiliza estes operadores para gerar uma nova população a partir da atual. Tomando-se uma população, o AG faz sucessivas escolhas de pais e aplica sobre eles o operador de *crossover*, gerando um ou mais filhos. Estes filhos são colocados na nova população e o processo é repetido até que esta seja totalmente preenchida (i.e. possua o mesmo número de indivíduos que a população original). Após a criação da nova população, aplica-se a cada indivíduo nesta, com uma probabilidade bastante baixa, o operador de mutação.

A escolha dos pais em um AG usualmente se faz utilizando o conceito de “roleta”, onde faz-se uma escolha probabilística sobre toda a população. A probabilidade de um indivíduo ser escolhido é proporcional ao seu *fitness*, de forma que melhores indivíduos possuem mais chances de serem escolhidos mas, ao mesmo tempo, dá-se possibilidade para que indivíduos piores também o sejam, garantindo certa diversidade nas populações. É comum que o melhor indivíduo de uma população seja transmitido para a nova população sem modificações, de forma a garantir que a melhor solução encontrada até o momento não seja perdida.

Algoritmos Evolutivos, em sua forma geral, são amplamente utilizados para resolver problemas de otimização e mesmo aprendizado (MICHALEWICZ, 1996). São reconhecidos por serem robustos e de fácil aplicação em uma ampla gama de problemas, em especial àqueles onde se sabe pouco sobre sua estrutura (FOGEL, 2000).

## 2.3 Estruturas de aprendizado

Os paradigmas apresentados anteriormente tratam de formas de manipulação de informação, mas estas informações devem estar devidamente codificadas para que possam ser manipuladas. Para tanto, utilizamos estruturas de codificação. A seguir apresentaremos brevemente algumas das estruturas mais comuns utilizadas em aprendizado de máquina.

### 2.3.1 Regras de produção

Regras de produção são pares “condição-ação” na forma “if C then A”. Se todas condições em C forem satisfeitas então um conjunto de ações, A, é executado. Devido a sua simplicidade e grande capacidade de expressão, regras de produção são amplamente utilizadas em aprendizado de máquina (MICHALSKI; CARBONELL; MITCHELL, 1983). Também são consideradas uma das formas mais *legíveis* para os humanos, isto é, que fornecem resultados finais que podem ser facilmente interpretados.

A capacidade de expressão de regras está limitada ao tipo de *lógica* utilizada por estas, uma vez que é esta lógica que determinará que tipos de conceitos podem ser codificados nas condições. Duas lógicas se destacam na literatura de aprendizado de máquina: lógica proposicional e lógica de primeira ordem.

A lógica proposicional é a mais simples das lógicas e também é chamada de “baseada em atributos”. De forma simples, ela permite verificar a presença de determinados atributos em exemplos. Por exemplo, em lógica proposicional pode-se escrever:

$$\text{if Formato=redondo} \wedge \text{Cor=vermelho then Bal\~{a}o = true}$$

Assim, a lógica proposicional permite que as condições sejam *conjunções* de proposições. Disjunções são lidadas através de subconjuntos de regras que tenham as mesmas ações.

A lógica de primeira ordem (FOL - *first order logic*) estende a lógica proposicional permitindo que sejam definidas *relações* entre objetos. Isto permite que se generalize conceitos de forma muito mais abrangente. Por exemplo, a seguinte cláusula é possível em FOL, representando o conceito de “Avô”:

$$\text{if Pai}(x,y) \wedge \text{Pai}(y,z) \text{ then Av\~{o}}(x,z)$$

Para quaisquer instâncias que possuam tais relações esta regra será aplicável. Em lógica proposicional isso não é possível e, ao apresentar exemplos do conceito “Avô”, poderíamos chegar na seguinte regra (dados determinados exemplos):

$$\text{if Pai\_Jose = Joao} \wedge \text{Pai\_Joao = Antonio then Av\~{o}\_Antonio\_Jose = True}$$

Esta regra é, naturalmente, demasiadamente restrita, aplicando-se especificamente para as instâncias envolvidas. A lógica proposicional é incapaz de extrair as relações, pois esta apenas analisa os atributos como são apresentados. Assim, a grande expressividade

da FOL é de grande interesse para o aprendizado de máquina, mesmo que para diversas situações a lógica proposicional seja perfeitamente adequada. É comum, ao trabalhar-se com FOL, limitar-se a descrição de cláusulas na *forma de Horn* por questões de computabilidade e decidibilidade. O aprendizado indutivo de regras em FOL restritas às cláusulas de Horn é também chamado de Programação em Lógica Indutiva ou ILP (*Inductive Logic Programming*), pois o processo pode ser visto como a construção por indução de programas em *Prolog*.

Seja qual for a lógica por trás do processo, um algoritmo de aprendizagem que trabalhe com este tipo de estrutura deve se preocupar em encontrar um conjunto de regras que represente o conceito a ser aprendido. O espaço de hipóteses é muito grande, mesmo para lógicas proposicionais. Para aprendizado indutivo é comum utilizar-se do chamado *método de cobertura*, também conhecido como método de divisão-e-conquista. Este método consiste em quebrar o conceito em uma disjunção de diversas cláusulas, cada uma cobrindo parte dos exemplos. Assim, uma disjunção é aprendida por vez. O algoritmo deste método é como segue (MITCHELL, 1997):

1. Regras\_aprendidas  $\leftarrow$  []
2. Regra  $\leftarrow$  Gera\_Regra(Exemplos, Atributos, Conceito)
3. *while* performance(Regra, Exemplos) > Limite *do*
  - (a) Regras\_aprendidas  $\leftarrow$  Regras\_aprendidas + Regra
  - (b) Exemplos  $\leftarrow$  Exemplos - [exemplos corretamente classificados por Regra]
  - (c) Regra  $\leftarrow$  Gera\_Regra(Exemplos, Atributos, Conceito)
4. *return* Regras\_aprendidas

A idéia deste algoritmo é realizar sucessivas reduções no espaço de busca. Uma disjunção é aprendida por vez e os exemplos cobertos por esta são removidos, passando-se para a próxima disjunção, até que o conjunto de exemplos torne-se vazio (i.e. totalmente coberto).

### 2.3.2 Redes Neurais

Redes Neurais, ou Sistemas Conexionistas, fornecem um método prático e geral com inspiração biológica para aprendizado de diversos tipos de funções (MITCHELL, 1997). Uma rede neural pode ser vista como um processador massivamente paralelo e distribuído, constituído de unidades de processamento simples, que têm propensão natural para armazenar conhecimento experimental e torná-lo disponível para o uso (HAYKIN, 2001). Uma rede neural é composta por diversas unidades básicas de processamento (neurônios) que são interligadas segundo uma topologia. Essas conexões entre neurônios possuem *peso* variável, isto é, podem ser alteradas de forma a modular os sinais que por elas se propagam. A Figura 2.2 apresenta uma rede neural típica. Neurônios de entrada capturam informações externas (e.g. instâncias do conceito a ser aprendido) que são

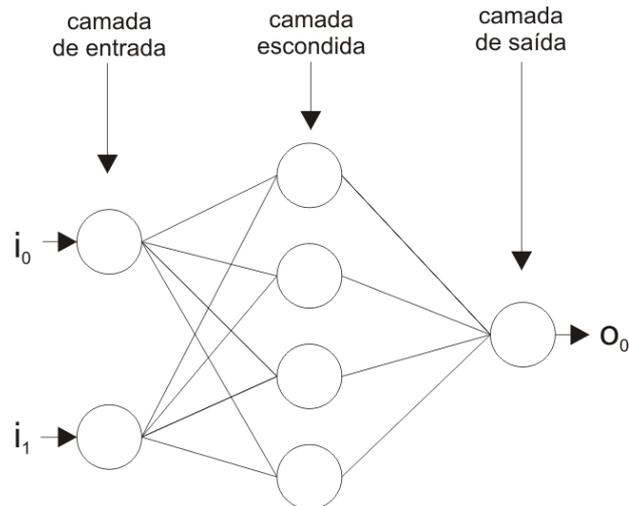


Figura 2.2: Exemplo de uma rede neural

processadas por camadas internas (escondidas) e o resultado é extraído por neurônios em uma camada de saída. Cada neurônio realiza uma operação simples, usualmente uma soma de suas entradas, ativando sua saída apenas se esta soma atinge um determinado limiar, representado por uma *função de ativação*. O padrão de ativação por toda a rede é, então, responsável pelo processamento da informação. As informações em uma rede neural são armazenadas de forma distribuída, principalmente nos pesos das conexões.

Pode-se dividir as redes neurais em duas classes segundo o tipo de conexões existentes: *feedforwards* e recorrentes (YAO, 1999). Em redes do tipo *feedforward* não há laços de realimentação, isto é, neurônios em camadas à frente na topologia não ligam suas saídas em entradas de neurônios anteriores. Em redes recorrentes, por outro lado, não há tal restrição. Pode-se, ainda, diferenciar-se entre redes de múltiplas camadas e redes de camada única (HAYKIN, 2001).

Algoritmos de aprendizagem para redes neurais se preocupam principalmente em encontrar os pesos corretos para cada conexão de forma que a rede passe a representar a função sendo aprendida. Há algoritmos que também procuram encontrar topologias adequadas para o aprendizado, como por exemplo (STANLEY; MIIKKULAINEN, 2001). Redes neurais são extensivamente utilizadas para aprendizado supervisionado, não supervisionado e por reforço (YAO, 1999).

Um algoritmo amplamente utilizado é o chamado *backpropagation* (RUMELHART; HINTON; WILLIAMS, 1986), com uso principalmente em aprendizagem supervisionada por indução. Este algoritmo realiza uma busca local por gradiente alterando os pesos das conexões de modo a reduzir o erro nas saídas. O processo inclui o seguinte algoritmo (MITCHELL, 1997):

1. Apresenta-se o exemplo  $\vec{x}$  e computa-se o valor de saída  $o_u$  de cada unidade  $u$  na rede;

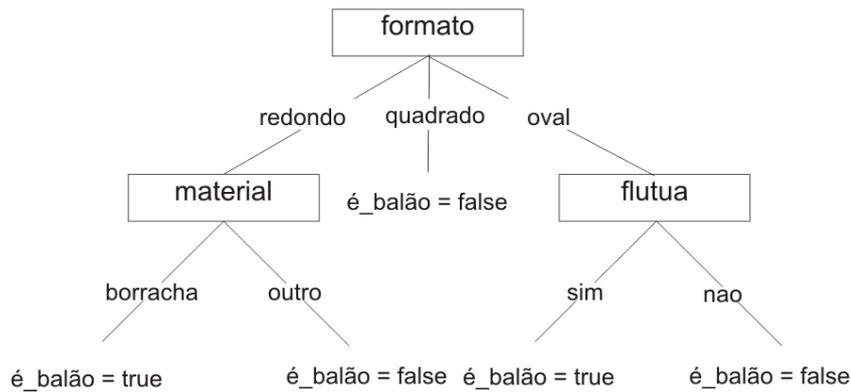


Figura 2.3: Exemplo de uma árvore de decisão

2. Para cada unidade de saída  $k$ , calcula-se o termo de erro:

$$\delta_k = o_k(1 - o_k)(t_k - o_k)$$

3. Para cada unidade escondida  $h$ , calcula-se o termo de erro:

$$\delta_h = o_h(1 - o_h) \sum_{k \in \text{saídas}} w_{kh} \delta_k$$

4. Atualiza-se cada peso de conexão  $w_{ji}$ :

$$w_{ji} = w_{ji} + \eta \delta_j x_{ji}$$

Este algoritmo é iterado para todos exemplos disponíveis até que o erro da rede se torne aceitável. A inicialização dos pesos é usualmente aleatória, dentro de intervalos como  $[-0.5, 0.5]$ . O valor  $\eta$  é chamado de “taxa de aprendizagem” e define a taxa de descida do gradiente. Problemas com este tipo de estratégia incluem a despreocupação com a topologia, que deve ser definida *a priori*, e a possibilidade de atingir mínimos locais que não satisfaçam os requisitos de performance.

### 2.3.3 Árvores de decisão

Árvores de decisão são um dos métodos mais utilizados no aprendizado indutivo (MITCHELL, 1997). São utilizadas principalmente para representar funções com valores discretos e possuem certa equivalência com regras de produção. Uma árvore de decisão está representada na Figura 2.3. Cada nó na árvore representa um teste de algum atributo da instância sendo analisada e cada ramo a partir daquele nó corresponde a um possível valor para aquele atributo. Este processo de teste e ramificação prossegue até as folhas, onde o conceito da instância é determinado.

Nota-se que uma árvore de decisão representa uma disjunção de conjunções de testes em atributos de instâncias, onde cada ramo completo representa uma conjunção de atributos, enquanto os nós provêm disjunções. Um algoritmo de aprendizado para uma árvore de decisão deve descobrir quais atributos devem ser testados em quais nós de modo

a aprender o conceito. Assim, uma árvore de decisão é construída quebrando um problema complexo de decisão em diversas decisões mais simples (SAFAVIAN; LANDGREBE, 1991), realizadas nos nós.

O algoritmo mais conhecido para aprendizado sobre árvores de decisão é o chamado ID3 (MITCHELL, 1997) e este é um bom exemplo das técnicas usualmente utilizadas. A construção é feita de cima para baixo e em cada nó é tomada uma decisão (gulosa) de qual atributo deve-se testar. Esta decisão é tomada verificando-se cada atributo disponível e analisando a sua capacidade isolada em classificar os exemplos fornecidos. A medida utilizada para avaliação de cada atributo é a chamada *entropia*, conceito derivado da teoria da informação. De forma simplificada, a entropia é uma medida da capacidade do atributo sendo testado de classificar os exemplos. Assim, do nó até as folhas, o algoritmo escolhe com base na entropia os atributos para serem testados. Como a escolha é gulosa, não sendo possível rever decisões já tomadas, há a chance de obter-se árvores sub-ótimas.

## 3 SISTEMAS COMPLEXOS

No presente capítulo, realizamos uma revisão dos principais conceitos envolvendo sistemas complexos e suas propriedades. É nosso interesse estabelecer termos que serão utilizados nos capítulos subsequentes. Iremos definir o que são sistemas, sistemas complexos, emergência e auto-organização e revisaremos usos destes sistemas na computação, incluindo inteligência de enxame e inteligência coletiva.

### 3.1 Sistemas, emergência e auto-organização

*Sistemas* são definidos como “complexos de elementos em interação” (BERTALANFFY, 1975) e representam um grupo de elementos independentes mas interrelacionados que formam um todo distinguível do ambiente em que reside.

O estudo de sistemas possui duas correntes científicas identificáveis que se intercalam em períodos da História. Uma corrente, que teve seu auge no século XVI e XVII com as idéias e estudos de Descartes, Newton e outros, procura compreender um sistema através da decomposição e estudo isolado de cada uma de suas partes (CAPRA, 1998). Procura-se nas propriedades das partes as causas dos fenômenos observados no todo. Essa corrente é chamada *mecanicista*, *reducionista* ou, ainda, *cartesianista*.

Uma outra corrente, denominada *holística*, *organísmica* ou *sistêmica*, vê um sistema como um fenômeno completo que não pode ser decomposto sem que as propriedades que lhe são únicas sejam perdidas (CAPRA, 1998). Nenhuma propriedade do sistema é presente nas partes e o estudo destas é de pouca utilidade para a compreensão do todo. Credita-se estas propriedades às interrelações e conexões das partes mais do que à função isolada de cada uma. Assim, pode-se conhecer tudo sobre cada parte individualmente e ainda assim compreender muito pouco sobre a totalidade.

A visão sistêmica, em geral, parece ser mais aplicável quando o número de partes de um sistema torna-se maior e suas interrelações, mais complexas. Em sistemas simples a visão reducionista é suficiente para explicar a maioria dos fenômenos observados. O “problema dos três corpos” ilustra bem esta questão. Um sistema de dois corpos livres sob efeito apenas de suas forças gravitacionais pode ser analisado sobre a ótica reducionista: compreendendo as propriedades de cada corpo é possível prever como o sistema se comportará. Porém, ao adicionarmos um terceiro corpo o sistema subitamente passa a se comportar de forma complexa e, neste caso, caótico e qualquer previsão a partir unica-

mente das propriedades dos corpos falha para o sistema como um todo. O sistema com três corpos passa a ser melhor descrito pela corrente sistêmica, onde o sistema e suas propriedades (como o movimento caótico dos corpos) só existe na sua concepção como um todo e não em suas partes isoladas.

Outro exemplo de transição entre visões sistêmicas e reducionistas pode ser observado no estudo de colônias de formigas. Ao estudarmos algumas poucas formigas, seus comportamentos são perfeitamente explicáveis pelo estudo das formigas em si (modo de se locomover, sentidos, formas de comunicação). Porém, ao lidarmos com uma colônia composta por milhares de formigas, o comportamento apresentado por esta dificilmente é explicado pelo estudo das formigas individuais (GORDON, 2002). A colônia não é uma extrapolação da formiga, mas algo que é diferente, uma entidade única. A visão sistêmica, novamente, torna-se útil.

Na década de 40, Ludwing von Bertalanffy desenvolveu a chamada *Teoria Geral dos Sistemas* (BERTALANFFY, 1975), uma teoria fundamentalmente preocupada em substituir a visão reducionista por uma sistêmica, com fundamentação e aplicação baseada em sistemas biológicos. Bertalanffy se preocupava em discernir a *organização e padrões* de sistemas, de modo a permitir a análise e aplicação de semelhanças e isomorfismos estruturais em áreas diversas. Seu principal interesse estava nos sistemas ditos de *complexidade organizada* ou, simplesmente, *sistemas complexos*.

Johnson (JOHNSON, 2001) define um *sistema complexo* como um sistema onde o número de elementos em interação é nem tão pequeno que possa ser analisado diretamente através de relações causa-efeito, nem tão grande que se possa utilizar análises estatísticas ou probabilísticas. Sistemas complexos são interessantes devido a principalmente uma característica: a partir de interações de partes com comportamentos simples *emerge* um padrão complexo. De certa forma, o meio-termo quantitativo de Johnson é também um meio-termo entre a visão reducionista e a visão sistêmica. Enquanto se mantém a questão do todo e ênfase nas relações, torna-se também necessário compreender as partes ou, ao menos, seus aspectos relevantes à formação do coletivo.

Uma propriedade central em sistemas complexos é a *emergência*. Como foi visto, a perspectiva sistêmica dá importância à interrelação das partes e as propriedades no sistema como um todo não se faz presente nestas. Por exemplo, o conceito de temperatura é central no estudo da termodinâmica, mas inexistente se tratarmos de átomos individuais. Diz-se que tais propriedades *emergem* da interação das partes.

Sabe-se que diversos sistemas compostos por múltiplas partes possuem propriedades *macroscópicas* que só fazem sentido ou são percebidas no sistema composto, mas em nenhuma das partes individualmente. Da mesma forma, diversos outros sistemas são compostos por partes menores, que existem e são funcionais individualmente mas que dão origem a comportamentos e propriedades únicas quando agregadas. Em sistemas complexos, tais comportamentos e propriedades freqüentemente são ditas *emergentes*.

Um comportamento ou propriedade é dito emergente quando é resultado de partes executando processos relativamente simples em uma escala inferior àquela em que o comportamento é observado (JOHNSON, 2001). São os fenômenos também chamados

de *bottom-up*, de baixo para cima. Assim, formigas são as partes simples que compõem os comportamentos emergentes de uma colônia. Neste trabalho utilizaremos o termo *emergente* nesse sentido, para designar propriedades ou comportamentos macroscópicos que são fruto da interação entre partes no nível microscópico. Gilbert (GILBERT; CONTE, 1995) destaca a importância da emergência, dentro de um contexto de sistemas sociais:

[...] the importance of emergence is that interacting agents in a common world produce phenomena that are not intended by the agent themselves but are none the less relevant to their later behaviour and achievements.<sup>1</sup>

Sistemas complexos são também conhecidos como sistemas de *complexidade organizada*, de forma a enfatizar outra de suas principais propriedades: a auto-organização. Capra (CAPRA, 1998) faz a seguinte definição de auto-organização:

[...] a auto-organização é a emergência espontânea de novas estruturas e de novas formas de comportamento em sistemas abertos, afastados do equilíbrio, caracterizados por laços de realimentação internos e descritos matematicamente por meio de equações não-lineares.

Auto-organização é, portanto, o processo através do qual um sistema tem sua organização aumentada ou mantida sem intervenção direta externa. É a emergência da organização. Esta propriedade, em uma primeira leitura, parece divergir da segunda lei da termodinâmica, que dita que um sistema isolado sempre tende a ter sua entropia (i.e. medida de desorganização) aumentada. Porém, a visão sistêmica praticamente impede a existência de sistemas isolados (i.e. fechados), uma vez que seu contexto e ambiente sempre é levado em conta. Portanto, sistemas auto-organizáveis são sistemas abertos, como foi observado em (BERTALANFFY, 1975), não havendo violação estrita da segunda lei da termodinâmica.

Sistemas auto-organizáveis são responsáveis por muitos fenômenos observados na natureza ou gerados artificialmente. Exemplos vão desde a sincronização do pulsar de milhares de vagalumes em uma floresta (STROGATZ, 2003) até a emissão coerente de luz em um LASER (CAPRA, 1998), passando por ninhos de cupins (GORDON, 2002) e redes e sistemas sociais (WATTS, 2003a; GILBERT; CONTE, 1995).

A característica não-linear de sistemas complexos os tornam extremamente difíceis de serem analisados e frequentemente utilizam-se modelos computacionais para não só modelar e observar tais sistemas, mas também como ferramenta para resolver problemas diversos.

### 3.2 Sistemas complexos e computação

Em computação, o estudo da emergência tem como exemplo típico os *autômatos celulares*. Um autômato celular é uma estrutura bastante simples, composta de células

---

<sup>1</sup>[...] a importância da emergência é que agentes interagindo em um mundo comum produzem fenômenos que não são intencionados pelos próprios agentes mas que ainda assim são relevantes para seus comportamentos e ganhos posteriores.

justapostas que obedecem regras usualmente simples (WOLFRAM, 2002). Cada célula é, efetivamente, uma máquina de estado finito que modifica seu estado de acordo com seu estado atual e o estado de células vizinhas, mantendo interações em nível local. Esta simples composição é capaz de gerar comportamentos bastante complexos quando as regras e os estados iniciais são ajustados apropriadamente, tendo servido para a modelagem de sistemas naturais e até mesmo estudos de vida artificial, entre outras aplicações.

Mais recentemente, o estudo de sistemas multiagentes (MAS - *Multi-Agent Systems*) tem abraçado a área de sistemas complexos, necessária quando se trabalha com um grande número de agentes. Em (RESNICK, 1997), inclusive, o estudo da emergência em MAS é proposto como ferramenta de educação da visão sistêmica. Uma extrapolação desses sistemas atualmente possui uma área de estudos própria, a chamada inteligência de enxame<sup>2</sup>. Nestes sistemas procura-se resolver problemas através de comportamentos que emergem da interação de múltiplos, freqüentemente idênticos, agentes.

Fazem parte do estudo de inteligência de enxame os chamados Otimização por Colônia de Formigas (ACO - *Ant Colony Optimization*) e Otimização por Enxame de Partículas (PSO - *Particle Swarm Optimization*). Nos ACO, utiliza-se a metáfora de uma colônia de insetos sociais (formigas, cupins etc.) para especificar agentes que se comportam como tal e soluciona-se problemas através da interação entre estes. Em sua forma mais simples, cada agente realiza uma busca aleatória no espaço de soluções do problema e, ao fazer isso, deixa um “rastro” indicando seu caminho. Esse rastro (cuja função é análoga ao uso de feromônios por formigas) tem sua intensidade proporcional à qualidade das soluções visitadas e outros agentes são atraídos por estes rastros, de forma a gerar uma tendência a iniciar trajetos que derivem de boas soluções encontradas até então. Diversos problemas podem ser resolvidos através desse simples modelo como, por exemplo, otimizações no problema do caixeiro viajante (STÜTZEL; DORIGO, 1999).

Já nos PSO utiliza-se uma metáfora “social-psicológica” para controlar indivíduos (partículas) que se locomovem no espaço de soluções de um problema e interagem umas com as outras (KENNEDY; SPEARS, 1998). Em um algoritmo PSO típico, um grupo de partículas é distribuído no espaço de soluções e se locomovem neste de forma inicialmente aleatória. Uma partícula pode, porém, “lembrar-se” dos melhores pontos que já visitou e retornar a estes após certo tempo. Adicionalmente, o movimento de uma partícula é influenciado pelo sucesso de partículas vizinhas. Desta forma, forma-se uma rede de conexões entre partículas vizinhas que influencia o movimento global destas.

Em ambos os casos acima, um problema é resolvido através de comportamentos emergentes de um sistema complexo. Por exemplo, na otimização por ACO do problema do caixeiro viajante (STÜTZEL; DORIGO, 1999), nenhum agente tem a intenção ou tarefa explícita de encontrar um caminho ótimo global, mas a descoberta deste é feita no nível do sistema como um todo, composto por centenas ou milhares de agentes, caracterizando um comportamento emergente. É importante ressaltar que não é um simples problema de divisão de tarefas. Se assim o fosse, um único agente seria capaz de resolver o problema mesmo que levando um tempo maior. Em contraste, é a interação de agentes

---

<sup>2</sup>*Swarm intelligence*

com outros agentes que fornece a solução para o problema e nenhum individualmente é capaz de resolvê-lo.

Também nota-se, nos casos citados, que os agentes são extremamente simples. Basicamente são agentes reativos, guiados por poucas regras que indicam como devem se comportar frente a situações. Uma formiga no ACO segue regras tão simples como “siga o feromônio” e “ande aleatoriamente”. Em particular, um agente em geral não é capaz de *aprender* apesar de ser possível e comum que o sistema como um todo apresente características de aprendizado. Frente a uma mesma situação os agentes respondem sempre da mesma maneira ou, no máximo, utilizam algum algoritmo estocástico para aleatorizar decisões.

Dentro dos estudos de inteligência de enxame, uma exceção a isto é a chamada Inteligência Coletiva (COIN - *Collective Intelligence*, desenvolvida por Wolpert e Tumer (WOLPERT; TUMER, 2001). Uma COIN é um sistema massivamente multiagente onde uma função de utilidade global é otimizada a partir da interação entre agentes capazes de aprendizado por reforço. Esta abordagem foi desenvolvida com a intenção de desenvolver controladores distribuídos onde diversos agentes, dotados de aprendizagem, cooperam para atingir um objetivo comum quantificável pela função de utilidade (e.g. qualidade de conexões em uma rede, área de abrangência de satélites etc.). Essa função é quebrada em funções menores e distribuída entre os agentes, que a otimizam através da interação com outros e da sua própria capacidade de aprendizado.

A abordagem COIN difere de algoritmos ACO e PSO pela complexidade das partes: enquanto partículas e formigas são agentes incapazes de aprender, o aprendizado é parte fundamental em sistemas COIN. A grande questão nesta abordagem é como especificar os agentes envolvidos para que um problema em específico seja resolvido globalmente. Isto é, como fazer um sistema se comportar da forma desejada quando este é composto por agentes capazes de aprendizado?

Assim, o modelo COIN provê um forte argumento pragmático para estender a compreensão da relação entre sistemas complexos e a capacidade de aprendizagem de elementos constituintes. Uma maior compreensão pode advir do estudo de modelos já existentes e há, na natureza, sistemas exatamente assim, cujos agentes estão entre os mais complexos de todos: os sistemas sociais.

## 4 RACIONALIDADE E APRENDIZADO EM SISTEMAS ECONÔMICOS

Neste capítulo, revisamos a literatura relevante sobre racionalidade limitada e aprendizado em sistemas econômicos. Fazemos a definição do que consideramos ser “racionalidade” e diferenciamos a racionalidade perfeita da racionalidade limitada, fornecendo exemplos de casos onde essa distinção se faz necessária na análise de sistemas sociais.

Uma pequena introdução à Teoria dos Jogos é feita, de forma a introduzir conceitos e termos que serão necessários em capítulos seguintes. Por fim, revisamos trabalhos e paradigmas da aplicação de aprendizado em sistemas econômicos.

### 4.1 Introdução

Na literatura recente, sistemas sociais, i.e. compostos por pessoas, são frequentemente caracterizados como sistemas complexos. Trabalhos como (WATTS, 2003a) e (WATTS, 2003b) apresentam modelos de estruturas sociais (mais especificamente relações entre indivíduos) que procuram modelar sistemas reais e, a partir destes, extrair propriedades emergentes. De fato, sistemas sociais têm diversas características que apóiam sua classificação como sistemas complexos: há múltiplos elementos em interação (pessoas) e propriedades e comportamentos emergentes em um nível superior, o da sociedade. Duncan Watts (WATTS, 2003a) explora como regras simples para conexão entre pessoas dão origem a redes sociais complexas, as chamadas redes de mundo pequeno<sup>1</sup>.

Um problema de se transpor as teorias de sistemas complexos para sistemas sociais é que estes são compostos por agentes capazes de análise e tomada de decisões conscientes, gerando uma variedade de comportamento e capacidade de adaptação que supera em muito a de qualquer outro sistema conhecido. O fascínio em relação a sistemas complexos com comportamentos emergentes é o fato de “tanto ser obtido de tão pouco” (HOLLAND, 1999) e, inclusive, a complexidade dos comportamentos emergentes é creditada por Johnson exatamente às limitações, ou à simplicidade, das partes que interagem (JOHNSON, 2001).

Se não somos exatamente simples, como lidar com sistemas formados por pes-

---

<sup>1</sup>*Small world networks.*

soas? Apesar de haver um grande interesse em diversos sistemas de interação humana, um em particular tem uma ciência dedicada à ele, por razões bastante óbvias: a Economia. A Economia como ciência se preocupa em estudar as relações de troca de bens e serviços em uma sociedade. Assim, interações entre agentes se dão através da compra e venda de valores, uma especialização bastante grande das possibilidades de interação entre pessoas. A Economia como um sistema complexo é um ramo recente de estudos por economistas (ARTHUR; DURLAUF; LANE, 1997; ARTHUR, 1999).

Mesmo com as restrições nas relações entre agentes criadas pela Economia, persiste o problema de como decisões são tomadas em ambientes econômicos. O paradigma mais amplamente utilizado para tratar este problema, o do *homem racional*, impõe limitações às escolhas possíveis pelos agentes envolvidos de modo a torná-los partes mais simples, tratáveis, em um todo complexo. O modelo do homem racional dita que um agente econômico (i.e. envolvido em uma transação econômica), frente a múltiplas escolhas sempre decidirá por aquela que lhe é mais benéfica. Na Economia, o benefício é caracterizado pela chamada *função de utilidade*,  $\mu$ , inerente a cada agente e que lhe fornece um valor associado para cada opção envolvida na escolha. Um agente é então dito racional, ou *substancialmente racional*, quando sua escolha é coerente e tal que maximize  $\mu$ . Uma escolha coerente é usualmente definida como uma que preenche as seguintes condições (COLMAN, 2003):

- R-1. Completude. Para cada par de alternativas  $(a_i, a_j)$  do conjunto total  $A$  então, para o agente, apenas uma dessas opções vale:  $\mu(a_i) > \mu(a_j)$ ,  $\mu(a_i) < \mu(a_j)$  ou  $\mu(a_i) = \mu(a_j)$ ;
- R-2. Transitividade. Se  $\mu(a_i) \geq \mu(a_j)$  e  $\mu(a_j) \geq \mu(a_k)$  então  $\mu(a_i) \geq \mu(a_k)$ ;
- R-3. Ordenação livre de contexto. Se  $\mu(a_i) \geq \mu(a_j)$  em  $A$ , então  $\mu(a_i) \geq \mu(a_j)$  também é válida em um conjunto  $A'$  tal que  $A \subset A'$ .

Simon (SIMON, 1980, p. 130) define de forma mais geral um comportamento substancialmente racional:

[...] behavior is substantively rational when it is appropriate to the achievement of given goals within the limits imposed by given conditions and constraints.<sup>2</sup>

Desta forma, é importante notar que o conceito de racionalidade substancial não exige que um agente econômico tenha *informações ilimitadas* nem que as informações a que tem acesso sejam verdadeiras, mas impõe que, dadas as informações disponíveis, o agente tomará a melhor decisão possível para atingir seus objetivos. Outros paradigmas de racionalidade existem, impondo maiores ou menores condições sobre as possibilidade de decisão (COLMAN, 2003).

<sup>2</sup>[...] um comportamento é substancialmente racional quando ele é apropriado para alcançar dados objetivos dentro de limites impostos por condições e restrições dadas.

Freqüentemente o paradigma do homem racional é aliado a outro paradigma, o do *homem egoísta*<sup>3</sup>, que data desde o início dos estudos da economia, fazendo-se presente na obra “*The Wealth of Nations*” de Adam Smith. Neste paradigma, considera-se que os agentes envolvidos no sistema preocupam-se unicamente consigo mesmos, isto é, jamais sacrificam seu bem estar em prol de outro agente. Adam Smith acreditava que pessoas e insituições egoístas trariam o bem comum à sociedade como um todo, garantindo eficiência e possibilitando a auto-regulagem sem órgãos centralizadores, guiados pela “mão invisível” do mercado (BUCHANAN, 2003).

O paradigma do homem racional e egoísta é amplamente utilizado e constitui a base de duas importantes teorias sobre tomada de decisões: a *Teoria das Decisões* e a *Teoria dos Jogos*. A Teoria das Decisões engloba o estudo de decisões em ambientes onde apenas um único agente está envolvido, ou onde outros agentes são indiferentes às escolhas do agente em questão. Encaixam-se nesta teoria problemas como decisão de rotas para viagens, escolha de itens em um mercado ou compra de ações em pequenas quantidades.

A *Teoria dos Jogos*, formalizada por von Neumann e Morgenstern, se propõe a modelar, analisar e prever o resultado de interações entre agentes racionais egoístas em jogos bem definidos (WOOLDRIDGE, 2002). Pode ser vista como uma extensão multiagente da Teoria das Decisões e, para tanto, os agentes devem considerar em suas decisões também a reação de outros agentes às suas escolhas, tornando necessário *pensamento estratégico* (KRAUS, 2000).

Um *jogo*, na Teoria dos Jogos, é o processo de interação entre dois ou mais jogadores. Cada jogador possui *ações* que são tomadas segundo *estratégias* que levam em conta *informações* percebidas. Cada combinação das ações de todos jogadores dá a cada um uma *recompensa*<sup>4</sup>. Sendo os agentes egoístas, é do interesse deles maximizar suas recompensas, que são medidas segundo uma função de utilidade, intrínseca a cada jogador mas conhecida por todos participantes.

Sendo os agentes racionais e egoístas, é possível saber exatamente qual será sua escolha em qualquer momento de um jogo bem definido. A maneira com que os agentes chegam a tais decisões é totalmente abstraída nesta metodologia, toda a teoria é desenvolvida sobre a simples concepção de que um agente econômico é capaz de maximizar sua função de utilidade. Assim, pessoas são tomadas como elementos simples como átomos e moléculas, que seguem leis bastante específicas (BUCHANAN, 2003). Mesmo com estas simplificações, a análise de jogos não é nada simples e muitas teorias foram desenvolvidas de modo a prever e analisar resultados.

A forma mais simples de representar as interações em um jogo é a chamada *forma normal*. Um exemplo de jogo em sua forma normal, com dois jogadores, está representado na Fig. 4.1. Cada linha representa uma ação possível de ser tomada pelo jogador 1 (ação 1 e ação 2), enquanto as colunas representam as ações do jogador 2 (ação 3 e ação 4). A intersecção das ações fornece o resultado, sendo a recompensa para o jogador

---

<sup>3</sup>*Self-interested*

<sup>4</sup>*Payoff*.

2 o valor no canto inferior esquerdo e a recompensa para o jogador 1 o valor no canto superior direito de cada intersecção. Duas outras formas de representação são amplamente utilizadas: forma extensiva ou em árvore, e forma característica ou de coalizão (WOOLDRIDGE, 2002).

Tabela 4.1: Exemplo de jogo na sua forma normal.

	ação 1	ação 2
ação 3	4 4	2 -2
ação 4	1 1	0 0

Um conceito central na Teoria dos Jogos é o de *equilíbrio*, que define uma *solução* de um jogo. Em um jogo na forma normal, um equilíbrio é visto como um ponto no espaço de combinação de estratégias que satisfaz certas propriedades. O equilíbrio mais geral é conhecido como *Equilíbrio de Nash* e, de forma intuitiva, é definido como sendo um ponto no espaço de combinações de estratégias no qual os agentes não tem incentivo para optar por outra estratégia que não a tomada naquele ponto. Isto é, para todo agente não há uma estratégia que forneça maiores retornos do que aquela utilizada no equilíbrio. No jogo representado na Fig. 4.1 há um equilíbrio de Nash onde o jogador 1 opta pela ação ação 1 e o jogador 2 pela ação ação 3. Pode-se ver que não há outra ação, tanto para o jogador 1 quanto para o jogador 2, que lhes forneça um maior retorno e, portanto, ambos devem optar por jogar as ações naquele equilíbrio de Nash.

O equilíbrio descrito é chamado *puro*, pois é definido por uma ação específica por parte de cada agente envolvido. Porém, nem todo jogo possui um equilíbrio puro mas todos os jogos possuem ao menos um equilíbrio *misto* (WOOLDRIDGE, 2002). Um equilíbrio misto envolve ações com probabilidades associadas a cada uma. A avaliação de cada ação é feita levando em conta estas probabilidades e é chamada de *utilidade esperada*<sup>5</sup>. A Fig. 4.2 traz o conhecido jogo Pedra-Papel-Tesoura na sua forma normal. A melhor estratégia neste jogo é jogar cada ação com probabilidade iguais, isto é, 1/3.

Tabela 4.2: Jogo pedra-papel-tesoura na forma normal.

	Pedra	Papel	Tesoura
Pedra	0 0	1 -1	-1 1
Papel	-1 1	0 0	1 -1
Tesoura	1 -1	-1 1	0 0

A Teoria de Jogos encontra ampla utilização em grande diversidade de áreas mas

<sup>5</sup>Expected utility

em alguns cenários sua adequação é questionada. Talvez o mais aclamado jogo onde as deficiências chamam a atenção é o *Dilema do Prisioneiro* (AXELROD, 1985). Neste jogo, dois criminosos são presos, colocados em prisões diferentes e interrogados sobre a participação do *outro* prisioneiro no crime. Cada prisioneiro tem a opção de cooperar com seu parceiro, respondendo que o outro não teve participação, ou traí-lo, dizendo que o outro teve participação no crime. Se um prisioneiro trair e o outro cooperar, o que cooperou recebe toda a culpa e tem a maior punição, enquanto o que traiu é libertado. Se ambos traem, recebem punições pesadas. Se ambos cooperam, recebem punições leves. A escolha de cada prisioneiro é feita sem comunicação com o outro.

A Fig. 4.3 traz o jogo em sua forma normal. Pode-se observar que, não importando o que o outro jogador escolhe, trair sempre é a melhor opção para o jogador racional e egoísta: sabendo-se que o outro jogador irá trair, o melhor a fazer é também trair; sabendo-se que o outro irá cooperar, novamente é melhor trair e obter o benefício maior. O dilema advém do fato de que quando ambos traem, recebem benefícios menores do que se ambos tivessem cooperado. Porém, não é simples obter cooperação mútua, pois se um tomar a iniciativa de cooperar na esperança de obter os ganhos de cooperação mútua, o outro jogador pode explorar tal jogada e trair, obtendo um valor maior para si do que obteria se tivesse cooperado. Sabendo disso, nenhum jogador tem incentivo para cooperar. Efetivamente, o ponto onde ambos traem é um equilíbrio de Nash. A questão é que, tomando decisões racionais e traindo, os agentes obtêm ganhos inferiores ao que poderiam obter se, por algum outro método, decidissem pela cooperação.

Tabela 4.3: O Dilema do Prisioneiro em sua forma normal.

	trair	cooperar
trair	2, 1	2, 5
cooperar	5, 1	3, 3

De fato, em experimentos realizados com voluntários é freqüente a cooperação mútua (COLMAN, 2003), algo que a teoria não prescreve. Pessoas parecem agir fora do paradigma de racionalidade substancial. Diversas explicações já foram propostas para levar em conta os desvios da previsão feita pela Teoria dos Jogos (WOOLDRIDGE, 2002). Seja qual for o motivo, prevalece o fato de que as pessoas não se comportam como o modelo prevê, evidenciando que a teoria deve ser estendida ou modificada para que também englobe tais casos.

## 4.2 Racionalidade Limitada

Na década de 50, Herbert Simon já se preocupava com modelos que não se encaixavam bem aos fatos observados na economia, se perguntando se o paradigma do homem racional era adequado para modelar agentes econômicos. Citando evidências de experimentos na psicologia, Simon criticou a racionalidade imposta nos modelos por ser

incompatível com “[...] o acesso a informações e a capacidade computacional que de fato são possuídas por organismos, incluindo o homem, nos tipos de ambientes em que tais organismos existem”<sup>6</sup> (SIMON, 1955). Simon propôs que os limites na racionalidade dos agentes deveria estar presente nos modelos, dando origem aos estudos da *racionalidade limitada*<sup>7</sup>.

Os estudos na psicologia sobre como escolhas são feitas sugerem que tomar decisões consome recursos cognitivos e que pessoas procuram poupar tais recursos, utilizando sempre que possível regras pré-estabelecidas por experiência<sup>8</sup> para decidir, ao invés de realizar uma análise rigorosa sobre as múltiplas opções. A utilização sistemática dessas heurísticas garante uma resposta rápida a eventos, mas em determinadas situações leva também a erros sistemáticos (TVERSKY; KAHNEMAN, 1974). Diversos experimentos são encontrados na psicologia e um típico é descrito a seguir.

Os chamados “efeitos de formulação”<sup>9</sup> ocorrem quando a maneira com que o problema é formulado tem influência sobre a decisão do agente, mesmo que entre as formulações a informação efetiva seja a mesma. Por exemplo, em experimento realizados (TVERSKY; KAHNEMAN, 1981), voluntários foram informados que uma doença causaria seiscentas mortes no país. Dois programas mutuamente exclusivos para amenizar o problema eram considerados pelos seus resultados e o voluntário deveria decidir por um deles. Para um grupo de pessoas, os seguintes programas foram apresentados:

- A. Duzentas pessoa serão salvas;
- B. Com probabilidade de 1/3, seiscentas pessoas serão salvas; com probabilidade de 2/3, nenhuma será salva.

Para outro grupo distinto, as seguintes opções foram oferecidas:

- C. Quatrocentas pessoas irão morrer;
- D. Com probabilidade de 1/3 ninguém irá morrer; com probabilidade de 2/3 todas seiscentas morrerão.

De posse de ambos conjuntos de alternativas, não é difícil ver que *A,C* e *B,D* são alternativas idênticas. Os dois grupos estavam escolhendo opções idênticas e, esperar-se-ia, escolheriam opções idênticas. Porém, 72 % das pessoas no primeiro grupo escolheram *A* enquanto 78 % do segundo grupo optaram por *D*, em clara violação da condição R-1 de racionalidade apresentada na seção 4.1. Um das explicações para este resultado é que o primeiro caso, colocado na forma de ganhos, invoca uma regra de aversão a riscos enquanto, no segundo caso, a formulação na forma de perdas induz predileção por riscos (RUBINSTEIN, 1998).

---

<sup>6</sup> “[...] with the access to information and the computational capacities that are actually possessed by organisms, including man, in the kind of environment that such organisms exist.”

<sup>7</sup> *Bounded rationality.*

<sup>8</sup> *Rules of thumb*

<sup>9</sup> *Framing effects.*

Casos como esse são abundantes na literatura. Em (RUBINSTEIN, 1998) são descritos dois outros experimentos que contradizem os padrões de racionalidade e fornecem pistas sobre como pessoas raciocinam em determinadas situações. Em (KAHNEMAN, 2003) uma revisão dos trabalhos dos psicólogos Daniel Kahneman e Amos Tversky, relacionados à racionalidade limitada, é feita e fornece uma boa referência para outros experimentos e conclusões.

A grande quantidade de indícios da psicologia sobre a quebra do paradigma de racionalidade levou Simon a sugerir, portanto, que modelos de sistemas econômicos também devem especificar tais erros sistemáticos. Mais especificamente, Simon propunha que os *procedimentos* de raciocínio fossem introduzidos nos modelos (SIMON, 1986). Como foi visto, a Teoria dos Jogos, por exemplo, nunca se preocupa com a maneira com que decisões são tomadas e isto é também verdade para grande parte dos modelos econômicos vigentes, mas Simon desejava incluir esses processos em uma teoria mais abrangente, na forma de uma *racionalidade procedural*. Nesta forma de racionalidade, a modelagem dos agentes inclui os procedimentos de decisão, isto é, a maneira com que os agentes observam as informações disponíveis sobre as opções e fazem o mapeamento para uma decisão.

Apesar da idéia de que agentes econômicos são racionalmente limitados ser bem aceita, não há uma única abordagem para sua modelagem. Como observado em (ARTHUR, 1994): “A questão não é se racionalidade perfeita funciona, mas sim do que colocar em seu lugar.<sup>10</sup>”. Porém, em diversas abordagens a questão do aprendizado torna-se essencial.

### 4.3 Aprendizado e economia

Seria de se esperar que a mudança na forma com que os agentes se comportam, através da racionalidade limitada, ocasionaria uma grande mudança no sistema como um todo, afastando ainda mais os modelos tradicionais dos fatos reais, e seria o caso de se perguntar como a racionalidade substancial sobreviveu como conceito até os dias de hoje. A explicação para tanto é que, em diversos casos, adicionar racionalidade limitada aos agentes não altera o comportamento do sistema. Alguns estudos mostram que mesmo sistemas racionalmente limitados podem ter um macro-comportamento indistinguível de um modelo com racionalidade substancial. Uma das explicações mais amplamente aceitas para tanto é que há uma seleção natural aplicada constantemente sobre os agentes em sistemas econômicos: os que não lidam bem com as informações, cometendo erros sistemáticos, são explorados por melhores agentes e acabam saindo do sistema restando, ao final, agentes especializados que agem racionalmente ao menos para uma situação específica, levando ao comportamento e equilíbrios previstos por modelos econômicos tradicionais (RUBINSTEIN, 1998).

Usualmente, “sair do sistema” significa que o agente faliu ou não mais se arrisca a jogar para evitar perdas (RUBINSTEIN, 1998). Mas há outra alternativa: o agente em si

---

<sup>10</sup>“The question is not whether perfect rationality works, but rather what to put in its place.”

pode adaptar-se para evitar ser explorado, evitando sua própria saída. Do ponto de vista do sistema, não há diferença entre um agente que sai para dar lugar a outro melhor e um que se adapta e melhora a si próprio. Considerando que os agentes podem *aprender* e se adaptar, a seleção natural passa a ser não sobre as estratégias utilizadas, mas sobre a capacidade de adaptação e aprendizado de cada agente. Aqueles capazes de aprender a evitar serem explorados permanecem, enquanto os incapazes são efetivamente removidos do sistema. Nesse contexto, há dois níveis de adaptação ocorrendo. O primeiro é no nível dos agentes, que descartam e criam novas hipóteses e estratégias para permanecerem no jogo. O segundo é no nível do sistema, que descarta agentes e permite a entrada de novos. O comportamento adaptativo dos agentes leva a um macro-comportamento que, em alguns casos, é previsto por modelos tradicionais, mas sem a carga cognitiva imposta pela racionalidade substantiva.

No argumento acima é fácil ver paralelos com os sistemas complexos, apresentados anteriormente. Comportamentos em uma escala (dos agentes econômicos) produzem comportamentos macroscópicos em uma escala superior (do sistema econômico), sendo o comportamento emergente a convergência para um determinado equilíbrio. Não há, por parte dos agentes, intenção em levar o sistema ao equilíbrio, eles apenas desejam permanecer jogando e utilizam aprendizagem para tanto.

No modelo tradicional de racionalidade limitada pouco é dito sobre a capacidade de aprendizado dos agentes. O foco se encontra em substituir a regra simples de maximização de utilidade, imposta pela racionalidade substantiva, por outras regras que sejam mais condizentes com as capacidades verificadas em pessoas. Agentes são, então, meramente reativos, no sentido de que reagem a estímulos do ambiente segundo regras bem definidas e fixas. Porém, o conceito de aprendizado está intimamente ligado aos limites na racionalidade, como é observado em (GARROUSTE, 2001):

Learning in economics can be linked with the fact that individuals are perfectly rational in the sense that they optimally change their behavior because information coming from outside justify such a change. [...]. If however, individuals are bounded in terms of rationality, it is possible to assume that they try to modify or improve their behavior. In other words, they can learn.<sup>11</sup>

Apesar de economistas considerarem o aprendizado na economia, até recentemente isto era feito unicamente com a intenção de explicar a inovação e evolução em mercados e não como uma explicação para mudança de comportamento por parte do agentes envolvidos (GARROUSTE, 2001).

Recentemente, porém, estudos têm relacionado aprendizado e equilíbrio no sentido da Teoria dos Jogos. Sob esta nova perspectiva, consideram-se jogos onde um equi-

---

<sup>11</sup>“Aprendizado em economia pode ser ligado ao fato de que indivíduos que são perfeitamente racionais no sentido de que eles mudam seu comportamento de forma ótima por que informação vindo do exterior justificam tais mudanças. [...]. Se no entanto, indivíduos são limitados em termos de racionalidade, é possível assumir que eles tentam modificar ou melhorar seu comportamento. Em outras palavras, eles podem aprender.”

líbrio existe mas não é deduzível diretamente pelos agentes envolvidos, o que aconteceria se os considerássemos substancialmente racionais. Os agentes devem, então, aprender o equilíbrio durante o jogo (VIDAL, 2003).

Neste contexto, diversos algoritmos de aprendizagem foram propostos que possibilitam aos agentes encontrarem e se coordenarem em um equilíbrio. Tal convergência nem sempre é possível, podendo ser robusta ou não conforme o tipo de jogo e algoritmos de aprendizagem sendo utilizados (CHEN; GAZZALE, 2004). Entre os algoritmos mais estudados encontram-se: Jogada Fictícia, Dinâmica de Replicação e Melhor Resposta de Cournot<sup>12</sup>.

Diversas classes de jogos têm sido identificadas que garantem uma convergência robusta para uma grande variedade de mecanismos de aprendizado (CHEN; GAZZALE, 2004), porém para quase todos admite-se um número pequeno de agentes jogando o jogo, freqüentemente apenas dois. Isso é devido à característica do aprendizado: cada agente tenta modelar a estratégia utilizada pelos oponentes o que, para um grande número de jogadores, torna-se inviável. Também é o caso que nem sempre se tem acesso às ações dos outros jogadores, mas apenas ao resultado das ações coletivas, tornando impossível manter registro de estratégias individuais.

Ao levar-se em conta o aprendizado na Teoria dos Jogos, adiciona-se uma nova camada ao conceito de equilíbrio. Enquanto a Teoria dos Jogos tradicional assume que os jogadores tomam a ação no equilíbrio diretamente, ao considerar-se o aprendizado temos uma transição até o equilíbrio. Ainda assim, o conceito de equilíbrio é fundamental e procura-se um padrão consistente que induz os agentes envolvidos a não mais agirem (ARTHUR, 1999). No entanto, há jogos onde um equilíbrio estático, consistente, é inexistente. A adaptação nestes jogos não se torna um meio de atingir um equilíbrio, mas um fim em si. Um desses jogos é o chamado problema do bar *El Farol*<sup>13</sup>.

#### 4.4 O problema do bar El Farol

W. Brian Arthur, no Instituto Santa Fe, foi um dos pioneiros em tratar o problema do aprendizado e adaptação em cenários com múltiplos agentes racionalmente limitados. Arthur (ARTHUR, 1994) valeu-se da simulação em computador para criar um modelo simples de mercado e obter introspecções sobre a natureza do problema. O modelo é como segue:  $N$  pessoas desejam ir, em uma certa noite, para um bar (chamado El Farol); porém, a noite só será agradável se no máximo um determinado número de pessoas comparecer, caso contrário o bar estará superlotado e a noite será desagradável. É um jogo bastante simples: para cada agente envolvido, ficar em casa tem um benefício menor do que ir e o bar não estar lotado; porém, ficar em casa é melhor do que ir ao bar se este estiver lotado. Os agentes não podem se comunicar com outros agentes e devem tomar simultaneamente a decisão de ir ou não. A cada semana o processo é repetido e os agentes tem conhecimento de quantas pessoas compareceram ao bar nas semanas passa-

<sup>12</sup>*Fictitious Play, Replicator Dynamics, Cournot's Best Reply*

<sup>13</sup>*El Farol Bar Problem*

das. Note-se que um agente não têm conhecimento das ações tomadas por cada um dos demais agentes, mas têm acesso apenas ao resultado das ações coletivas (o número de pessoas que comparecem).

A intenção de Arthur ao criar tal jogo era o de estudar os efeitos do aprendizado na dinâmica desse sistema. Para tanto, ele forneceu a cada agente um simples sistema de aprendizado. Cada agente possui uma certa quantidade fixa de *hipóteses* que podem ser utilizadas para tentar prever o número de pessoas que comparecerão na semana atual. Um agente utiliza sua melhor hipótese para criar uma expectativa do número de pessoas no bar na semana atual. Se o número esperado for maior do que a lotação do local, então o agente decide não ir; caso contrário, ele vai. Estas hipóteses são codificadas na forma de regras que fornecem um valor previsto de pessoas no bar. Regras típicas são como segue.

*Prever o próximo número de agentes no bar como sendo:*

- o mesmo da última rodada;
- um determinado valor fixo;
- a média das últimas quatro rodadas;
- o mesmo que há seis rodadas;

Hipóteses nessa forma são distribuídas entre os agentes, podendo ou não serem compartilhadas por mais de um agente. O interesse está na heterogeneidade dos agentes, pois há baixa probabilidade de um grande número compartilhar as mesmas hipóteses (o que tornaria a dinâmica essencialmente trivial). Cada agente atribui pontos às suas hipóteses, de acordo com a performance observada no passado e a hipótese utilizada na hora da decisão é aquela que possui o maior número de pontos.

Devemos observar que o interesse nesse jogo não é em *como* os agentes aprendem, nem se eles podem aprender em absoluto. O foco do jogo está em gerar e permitir a observação de comportamentos macroscópicos (i.e. emergentes) a partir da dinâmica de um ambiente multiagente com agentes heterogêneos. Assim, a pergunta não é tanto “como devem os agentes decidir nessa situação?”, mas “supondo que os agentes decidem de alguma forma não trivial, o que acontece com o sistema?”.

Ao rodar simulações utilizando este modelo, Arthur percebeu que a média de pessoas no bar em cada rodada era próxima do previsto pela Teoria dos Jogos. Sendo  $N = 100$  e se os agentes consideram superlotado o bar se 60 ou mais pessoas comparecem, então há um equilíbrio misto de Nash onde cada agente opta por ir com probabilidade de 0.6 e não ir com probabilidade de 0.4, o que garante que, em média, 60 pessoas comparecem ao bar (ARTHUR, 1994). Sem conhecimento algum de Teoria dos Jogos e utilizando apenas um simples sistema de aprendizado, os agentes foram capazes de encontrar e se coordenarem em um equilíbrio de Nash. Isto é, o equilíbrio emerge do sistema dinâmico formado pelos agentes, caracterizando um sistema auto-organizável. Este resultado mostra que, pelo menos para alguns casos, a Teoria dos Jogos tradicional é útil para explicar ou prever interações de múltiplos agentes.

## 5 MINORITY GAME

Neste capítulo introduzimos o jogo que é o foco de nosso trabalho, o *Minority Game*. Introduzimos a definição mais utilizada na literatura e realizamos uma revisão dos trabalhos relevantes.

### 5.1 Introdução

O modelo do El Farol (ARTHUR, 1994) foi desenvolvido por um economista, sendo o interesse principalmente em assuntos econômicos. Como foi visto, o modelo é de fato um modelo de uma economia bastante simples, mesmo que conceitos característicos de sistemas econômicos estejam ausentes, como valores ou preços. Os atributos de sistemas dinâmicos presentes no modelo despertaram o interesse de físicos interessados na crescente área da *econofísica* (FARMER, 1999), que visa aplicar ferramentas da física em sistemas econômicos.

O *Minority Game* (MG) foi desenvolvido originalmente por Yi-Cheng Zhang e Damien Challet (ZHANG; CHALLET, 1997) como uma simplificação do modelo do El Farol de Arthur, em uma tentativa de reduzir o problema às suas partes mais básicas, permitindo uma análise mais rigorosa do modelo. A simplicidade do modelo contrasta com a variedade e riqueza de comportamentos gerados por ele, uma clara representação de um sistema complexo, o que aumentou ainda mais seu apelo. Desde sua criação, o modelo gerou diversas publicações, principalmente por físicos, com a intenção de distinguir propriedades e comportamentos e obter soluções analíticas para alguns casos.

Apesar de haverem diversas tentativas de realizar uma conexão mais forte entre o MG e situações reais (e.g. (BAZZAN et al., 2000; ZHANG, 1998)), grande interesse surgiu no modelo como uma entidade em si. Interesse em modelos mesmo sem validação no mundo real são comuns na área de simulação, como é mencionado em (GILBERT; CONTE, 1995):

Once the process of modeling has been accomplished, the model achieves a substantial degree of autonomy. It is an entity in the world and, as much as any other entity, it is worthy of investigation. Models are not only necessary instruments for research, they are themselves also

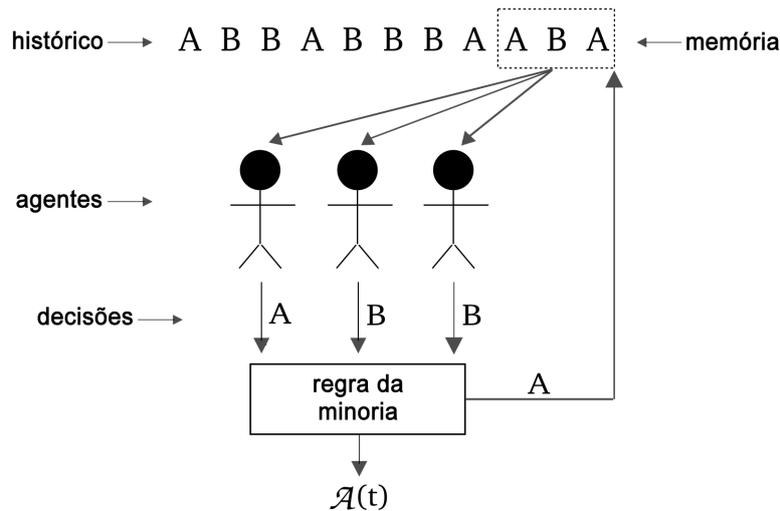


Figura 5.1: Diagrama esquemático do MG

legitimate objects of enquiry.<sup>1</sup>

Assim, nenhuma tentativa de validação dos modelos propostos será apresentada explicitamente, mesmo que possíveis vínculos com situações reais sejam mencionadas onde apropriado. Nosso interesse é no jogo em si.

Nas próximas seções faremos a definição formal do MG e revisaremos as principais contribuições já feitas sobre o jogo. A definição é uma adaptação da feita em (MORO, 2004) e (ZHANG, 1998), enquanto as diversas contribuições revistas serão citadas quando apresentadas.

## 5.2 Definição do MG

O MG é um jogo iterativo composto por  $N$  agentes, dois grupos e um vetor binário de histórico. Cada agente deve optar a cada rodada do jogo por um dos grupos e os agentes que pertencem ao grupo com o menor número de agentes (grupo da minoria) são recompensados. As escolhas são realizadas simultaneamente e sem comunicação entre os agentes. Para tomar uma decisão, cada agente tem acesso ao vetor de histórico que registra qual grupo foi o grupo da minoria em  $M$  iterações passadas. Assim,  $M$  define o tamanho da memória dos agentes no jogo. A Fig. 5.1 mostra o diagrama esquemático básico do MG.

Para melhor formalizar o problema, definimos que cada agente pode optar por duas ações: uma com valor “+1” e outra com valor “-1”, sendo cada ação associada a um grupo. Assim, definimos  $a_i(t)$  como sendo a ação tomada pelo agente  $i$  na rodada  $t$ . Ao final de uma rodada, após todos agentes terem tomado sua decisão, é possível calcular o

<sup>1</sup>“Uma vez que o processo de modelagem foi completada, o modelo atinge um grau substancial de autonomia. Ele é uma entidade no mundo e, tanto quanto qualquer outro modelo, ele é digno de investigação. Modelos não são apenas instrumentos necessários à pesquisa, são eles mesmos também objetos de investigação legítimos.”

valor total das ações:

$$A(t) = \sum_{i=1}^N a_i(t)$$

Cada agente, então, recebe como recompensa:

$$r_i(t) = -a_i(t)g(A(t))$$

A função  $g$  deve ser uma função ímpar de  $A(t)$  e no artigo original (ZHANG; CHALLET, 1997) é tida como  $g(x) = \text{sign}(x)$ . Com esta opção, os agentes que estão no grupo da minoria recebem um ponto de recompensa, enquanto um ponto é subtraído dos demais. Outras formas para  $g$  são possíveis, mas essencialmente as propriedades do jogo se mantêm para outras opções (LI; VANDEEMEN; SAVIT, 2000) e, portanto, utilizaremos aqui a função  $g$  como originalmente proposta.

O vetor de histórico não armazena o valor  $A(t)$ , mas apenas a informação sobre qual grupo conteve a minoria. Definimos, então,  $H(t) = -\text{sign}(A(t))$  como sendo a função que fornece o grupo da minoria na rodada  $t$ . Assim, um vetor de histórico  $H$  é composto unicamente por valores pertencentes a  $+1, -1$  sendo, portanto, um vetor binário. Este é um dos principais diferenciais do MG sobre o modelo do El Farol, uma vez que o El Farol registrava não somente se o bar estava superlotado ou não, mas também quantas pessoas compareceram.

Do ponto de vista de Teoria dos Jogos, observa-se que não há qualquer equilíbrio de Nash puro (WOLPERT; TUMER, 2001); tentativas de dedução levam a regressões infinitas sobre o raciocínio dos demais agentes (ARTHUR, 1999) e expectativas são sempre frustradas. Se muitos agentes preverem que um determinado grupo será o da minoria, então este terá a maioria, enquanto se poucos preverem que um grupo será o da minoria este de fato o será. A regra da minoria sempre frustra as expectativas da maioria. Seguindo o raciocínio aplicado no El Farol (ARTHUR, 1994), há um equilíbrio misto de Nash onde cada agente sempre decide por um grupo com probabilidade de 0.5. Mas, como no El Farol, estamos trabalhando com agentes que não são substancialmente racionais, o que permite que desvios na racionalidade gerem dinâmicas mais interessantes.

Para tomarem suas decisões, a cada agente são dadas  $S$  estratégias<sup>2</sup>. Uma estratégia é basicamente uma tabela que informa que decisão o agente deve tomar para cada combinação possível de eventos nas últimas  $M$  rodadas, isto é, é um conjunto de regras conjuntivas que cobre todas possibilidades combinatórias em  $M$ . A Fig. 5.1 mostra uma possível estratégia para  $M = 3$  onde, por exemplo, a primeira linha informa que se nas três últimas rodadas o grupo “-1” conteve a minoria, então o agente deve optar por “+1”. Assim, efetivamente uma estratégia pode ser representada por um vetor cujos componentes pertencem a  $\{-1,+1\}$  havendo, portanto,  $2^{2^M}$  possíveis estratégias.

As  $S$  estratégias são escolhidas para cada agente de forma aleatória, isto é, cada agente recebe  $S$  estratégias totalmente aleatórias e permanece com estas até o fim da

<sup>2</sup>O termo “estratégia” não é exatamente adequado, como foi argumentado em (BAZZAN et al., 2000), porém é o termo mais amplamente utilizado na literatura e será usado aqui.

Tabela 5.1: Exemplo de estratégia com  $M = 3$ .

$A(-3)$	$A(-2)$	$A(-1)$	Decisão
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

simulação. Não é permitido a um agente modificar ou descartar suas estratégias. Sobre estas, cada agente aplica um algoritmo de aprendizagem, de forma a escolher uma entre as  $S$  estratégias para ser utilizada a cada rodada. Este algoritmo de aprendizagem é bastante simples: a cada rodada, os agentes aplicam todas suas estratégias ao histórico da rodada anterior e aquelas que fornecem o resultado correto para a rodada anterior recebem um ponto de recompensa. Note-se que estes pontos de recompensa são dadas às estratégias e não ao agente em si e, portanto, são chamados “pontos virtuais”. Um agente utiliza sua estratégia com maior número de pontos virtuais para tomar a decisão final na rodada presente. Em caso de empate, opta-se por uma aleatoriamente.

Esta forma de aprendizado é praticamente idêntica à proposta para o El Farol original em (ARTHUR, 1994). As estratégias no MG equivalem às regras no El Farol, porém neste as regras eram criadas para representar raciocínios “razoáveis”, de certa forma antropomórficos, enquanto no MG estratégias são criadas de forma totalmente aleatória.

### 5.3 Dinâmica do MG

A definição do MG fornece o modelo para cada agente decidir entre um grupo ou outro. A dinâmica do MG é o resultado das decisões de cada agente, isto é, a quantidade de indivíduos em cada grupo com o passar do tempo, calculada através de  $A(t)$ . O MG, como foi visto, possui uma definição bastante simples. É exatamente sua simplicidade que torna interessante a dinâmica complexa gerada pelo modelo.

A maior parte dos estudos neste jogo utilizam simulações em computador para analisar sua dinâmica. A Fig. 5.2 mostra o resultado de simulações típicas. Pode-se notar que  $A(t)$  não se estabiliza em nenhum momento, comportamento atribuído à regra da minoria, e a média de  $A(t)$  é sempre zero (MORO, 2004).

Enquanto a média de  $A(t)$  é previsível e constante no MG, de maior interesse são as flutuações em torno desta média. É simples de ver que quanto mais distante está  $A(t)$  de zero, menor é o número de agentes vencedores: valores afastados de zero em  $A(t)$  indicam uma diferenciação na quantidade de agentes em cada grupo e, de acordo com a regra da minoria, o maior número de agentes que o grupo da minoria pode conter é

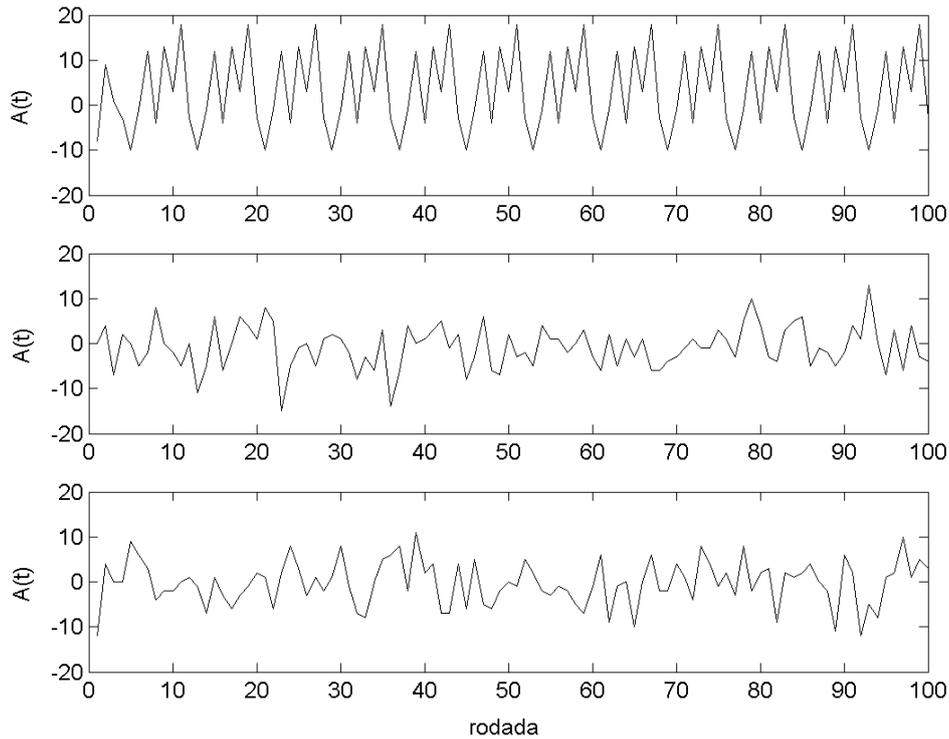


Figura 5.2:  $A(t)$  para diversos valores de  $M$ .  $M = 2$  (gráfico superior);  $M = 6$  (gráfico central);  $M = 10$  (gráfico inferior)

$(N - 1)/2$ , logo para  $A(t) \neq 0$  temos que o grupo da minoria é necessariamente menor que o máximo possível. Uma vez que apenas o grupo da minoria acumula pontos, há o interesse em que este seja tão grande quanto possível. Números pequenos de ganhadores indicam uma *ineficiência* no sistema, pois beneficia-se um número pequeno de agentes (ZHANG, 1998). Para  $A(t) = 0$  temos o maior número possível de agentes no grupo vencedor.

Para levar estas flutuações em torno de  $A(t)$  em conta introduz-se em (ZHANG; CHALLET, 1997) o conceito de *volatilidade*, fornecida pela *variância* ( $\sigma^2$ ) de  $A(t)$ . Calculamos a variância como sendo (ARAÚJO; LAMB, 2004a):

$$\sigma^2 = \frac{1}{(T - t_0)} \sum_{t=t_0}^T (\mathcal{A}(t) - \langle \mathcal{A} \rangle)^2$$

A variância é uma medida do desvio da média e, portanto, uma forma de quantificar as flutuações. Assim,  $\sigma^2$  passa a ser uma medida da *ineficiência global do sistema*. Valores altos de variância indicam que menos agentes estão vencendo no jogo e, portanto, mais ineficiente é o sistema. Em  $\sigma^2 = 1$  temos o caso ótimo, onde a flutuação é mínima e o maior número possível de agentes é vencedor. Chamaremos aqui de *caso aleatório* o caso onde cada um dos agentes escolhe aleatoriamente a cada rodada um dos grupos. Para este caso, a variância esperada é de  $\sigma^2 = N/4$  (ANDRECUT; ALI, 2001) e utilizaremos este valor como referência para as demais medições.

Através de extensivas simulações, foi identificado que a variância no MG depende

unicamente de um parâmetro  $\alpha$  para  $S \geq 2$  (MANUCA et al., 2000) que tem a seguinte forma:

$$\alpha = \frac{2^M}{N}$$

Nota-se que  $2^M$  é a medida do número de estratégias possíveis e, portanto,  $\alpha$  é uma razão entre a quantidade de estratégias disponíveis e o número de agentes no jogo. Assim,  $\alpha$  pode ser visto como uma medida do quanto o espaço de estratégias está coberto por agentes. Por exemplo, com  $N$  pequeno em relação a  $2^M$  o espaço de estratégias é pobremente coberto, uma vez que não há agentes suficientes para conter uma parte significativa das estratégias possíveis.

Do ponto de vista de aprendizado, poderíamos talvez esperar que quanto maior  $\alpha$  menor seria a variância, uma vez que o sistema e seus agentes tem à disposição uma maior área do espaço de estratégias para ser explorada. Mas não é o que acontece. Apesar de se saber que  $\sigma^2$  varia unicamente com  $\alpha$ , a forma com que esta varia não é nada simples. Para simplificar a análise, fixaremos  $N = 101$  e utilizaremos  $M$  como o parâmetro principal do sistema. A Fig. 5.3 mostra um gráfico da variância em função de  $M$ . Observa-se três estágios distintos:

- Para valores pequenos de  $M$  há variâncias muito acima do esperado para o caso aleatório, sendo grande o número de agentes perdedores;
- Para valores grandes de  $M$ , o modelo apresenta variância igual ao esperado para o caso aleatório. Chamaremos de  $M_{al}$  o valor de  $M$  a partir do qual o sistema apresenta este comportamento;
- Em valores intermediários de  $M$ , a variância atinge um mínimo, abaixo do esperado para o caso aleatório.

Este último caso é de particular interesse, pois mostra que os agentes, para alguns valores de  $M$ , conseguem se organizar e tornar o sistema mais eficiente do que se cada agente estivesse decidindo aleatoriamente, sugerindo que o sistema é capaz de se auto-organizar para reduzir as perdas de recurso. Tal organização seria emergente, uma vez que nenhum agente se preocupa com a eficiência do sistema como um todo mas apenas com seu próprio bem estar. É essa capacidade de auto-organização e sua dependência de  $\alpha$  as principais responsáveis pelo grande interesse no modelo do MG.

Pela definição vista anteriormente para um sistema auto-organizável, vemos que o MG encaixa-se bem nesta classe: há uma ordem emergente, representada pela capacidade do sistema de atingir variâncias inferiores ao caso aleatório; esta ordem é fruto de um sistema fora do equilíbrio, pois o equilíbrio é impossível devido à regra da minoria; o retorno fornecido pelo histórico aos agentes forma um laço de realimentação interno ao sistema.

Desde sua apresentação, diversas variações e usos foram dados ao MG. Em (BAZZAN et al., 2000) o MG é utilizado como modelo de tráfego urbano e modificado para

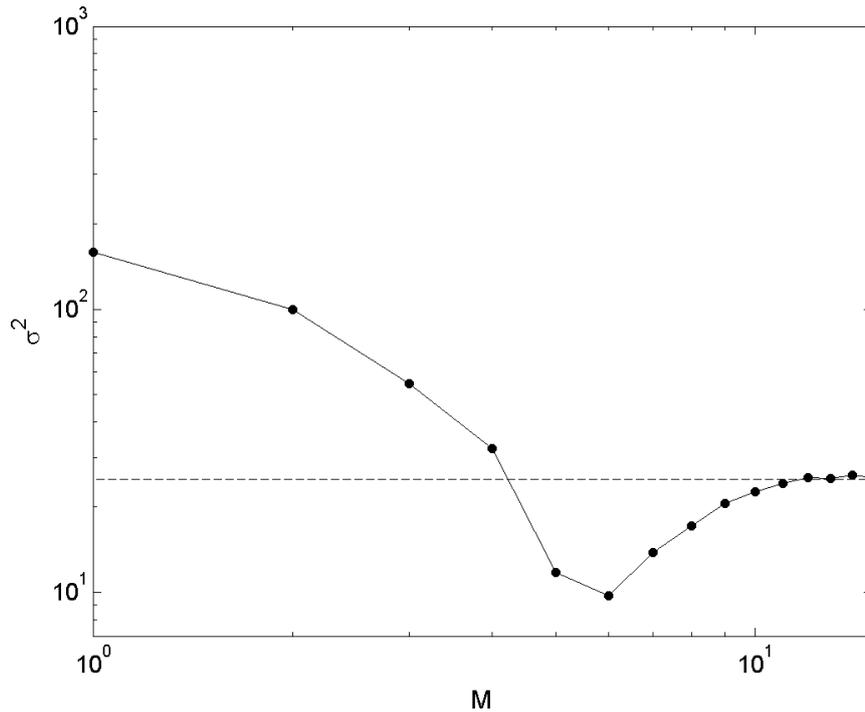


Figura 5.3: Variância em função de  $M$  para o MG tradicional com  $S = 2$  e  $N = 101$ .

conter estratégias que replicam personalidades de motoristas reais, sobre as quais é feito um estudo de quais se revelam mais eficientes. Os agentes envolvidos possuem capacidade restrita de aprendizado, equivalente ao MG tradicional, sendo o foco efetivamente o conceito de personalidade.

Em (CHALLET, 2003) a capacidade de auto-organização do MG é utilizada de forma reversa: procura-se utilizar o MG não como modelo mas como ferramenta de otimização. Propõe-se uma forma de quebrar uma determinada função em funções de recompensa no jogo, de forma que ao executar-se o MG a função desejada seja otimizada. Sob esta perspectiva, há o interesse em descobrir-se como permitir que os agentes atinjam melhores níveis de auto-organização e eficiência.

Outras variações do MG tem a intenção de explorar aspectos diversos do jogo ou estendê-lo para representarem outros problemas como, por exemplo: ligação espacial onde agentes apenas jogam o jogo com alguns vizinhos (MOELBERT; RIOS, 2002; SLANINA, 2000) e generalização para que o jogo permita um número indefinido de grupos (CHOW; CHAU, 2003).

## 6 ANÁLISE DO APRENDIZADO NO MINORITY GAME

Como foi visto nas seções anteriores, o principal foco de estudo no MG é na sua dinâmica em grupo. Isto é, que propriedades emergem no sistema pela interação de múltiplos agentes homogêneos. Homogêneos, aqui, denota a igualdade de condições entre os agentes: todos possuem a mesma capacidade de aprendizado e tamanho de memória, mesmo que destoe nas estratégias específicas possuídas por cada um.

Apesar do artigo original de Arthur (ARTHUR, 1994) enfatizar a questão do aprendizado em seu modelo, até recentemente pouca ou nenhuma atenção foi dada a este aspecto. O modelo de aprendizado indutivo com estratégias fixas, como proposto originalmente tanto para o El Farol como para o MG, foi adotado sem modificações pelos pesquisadores posteriores, interessados na solução analítica do modelo (e.g. (CHALLET; ZHANG, 1998; CHALLET; MARSILI, 2000)) ou extração de outras propriedades do sistema (e.g. (CARA; PLA; GUINEA, 1999; WAKELING; BAK, 2001)), com pouca ou nenhuma ênfase no sistema de aprendizado (em (WAKELING; BAK, 2001), porém, estuda-se os efeitos da *estrutura* de aprendizado ao utilizar-se uma rede neural).

Jogos como o MG fornecem um ambiente de interesse também para o estudo de algoritmos de aprendizado. Na maior parte da literatura atual, algoritmos de aprendizado buscam aprender padrões onde o algoritmo em si não tem qualquer tipo de influência. Os dados sobre os quais, por exemplo, uma rede neural aprende a categorizar tipos de objetos não se alteram com o processo de categorização. No MG e jogos relacionados, um algoritmo de aprendizagem tem uma tarefa substancialmente diferente, que é o de aprender padrões gerados pela interação com outros algoritmos e sobre os quais o próprio algoritmo tem influência. Adicionalmente, não há, nestes jogos, interesse direto em gerar um modelo a partir do aprendizado, como é o caso em diversas tarefas convencionais, mas unicamente permitir a adaptação do agente ao ambiente em que está inserido. É, no entanto, o processo emergente das interações entre múltiplos agentes capazes de aprendizado que são de interesse.

Neste sentido, o estudo dos efeitos de aprendizado na dinâmica desses jogos é relacionado à crescente área de inteligência de enxames, que se preocupa em utilizar sistemas massivamente multiagente para resolver problemas diversos como roteamento (ZHANG; KUHN; FROMHERZ, 2004) e traçado de caminhos (STÜTZEL; DORIGO, 1999). Usualmente agentes em sistemas utilizando inteligência de enxames, quase por definição, não são capazes de aprendizado sofisticado, ou o aprendizado se dá pela população e não nos

agentes individualmente, mas sistemas mais complexos com agentes capazes de aprendizado (como o modelo COIN (WOLPERT; TUMER, 2001)) estão surgindo e estudos levando em conta esta habilidade devem ser realizados e, para isso, o MG pode contribuir.

Recentemente o MG tem recebido alguma atenção da comunidade de aprendizado de máquina, interessada em aplicar outras técnicas de aprendizado nos agentes do modelo, apesar do desestímulo feito por Arthur (ARTHUR, 1994) para o El Farol:

It might be objected that I lumbered the agents in these experiments with fixed sets of clunky predictive models. If they could form more open-ended, intelligent predictions, different behavior might emerge. [...]. But I would be surprised if this changes the results in any qualitative way.<sup>1</sup>

Em (FOGEL; CHELLAPILLA; ANGELINE, 1999) é feita exatamente esta objeção e experimentos são realizados com agentes capazes de adaptarem suas hipóteses por meio de um algoritmo evolutivo. Cada agente possui  $k$  estratégias para predição, porém cada hipótese é composta de coeficientes para uma equação auto-regressiva da seguinte forma (FOGEL; CHELLAPILLA; ANGELINE, 1999):

$$x_j^i(n) = \text{round} \left( \left( a_j^i(0) + \sum_{t=1}^{l_j^i} a_j^i(t)x(n-t) \right) \right)$$

onde  $x(t)$  é o número de pessoas que foram ao bar na semana  $t$ ,  $l_j^i$  é o tamanho da memória para a hipótese  $j$  do agente  $i$ ,  $a_j^i(t)$  são coeficientes de modulação para cada semana  $t$  na memória e  $a_j^i(0)$  é uma constante.

A cada rodada, as  $S$  hipóteses de cada agente dão origem a outras  $S$  novas hipóteses através da adição de uma variável aleatória gaussiana com média zero a cada um dos coeficientes presentes nas hipóteses. A seguir, as  $2S$  hipóteses são testadas contra o histórico de resultados passados e as  $S$  melhores são retidas e as demais, descartadas.

Os resultados utilizando esta forma de aprendizagem mostravam, em (FOGEL; CHELLAPILLA; ANGELINE, 1999), baixa convergência, indicando que a observada por Arthur era inerente ao uso de estratégias fixas. Porém, os experimentos foram realizados utilizando parâmetros bastante específicos e, em especial, nenhuma referência à possibilidade de dependência da convergência ao tamanho de memória dos agentes é feita.

Estes resultados, aliados ao intenso interesse no MG, levou pesquisadores a experimentar com outros modelos de aprendizagem diretamente no MG. Exceto em alguns casos (e.g. (ARAÚJO; LAMB, 2004a)), o MG concentrou a atenção também da comunidade de aprendizado de máquina, reduzindo o interesse no El Farol especificamente. A seguir, algumas tentativas de aplicação de outros modelos de aprendizagem no MG são apresentadas.

---

<sup>1</sup>Pode ser objetado que embuti os agentes nesses experimentos com conjuntos fixos de modelos preditivos rudimentares. Se eles pudessem formar previsões mais abertas, inteligentes, comportamento diferente poderia emergir. [...]. Mas eu me surpreenderia se essas mudanças alterassem os resultados em qualquer forma qualitativa.

No artigo original que introduz o MG (ZHANG; CHALLET, 1997) já é feita uma proposta de algoritmo evolutivo para permitir mudança nas estratégias. Neste, o pior jogador é removido do jogo e substituído por um *clone* do melhor jogador, isto é, um novo agente exatamente com as mesmas estratégias do melhor jogador do momento, mas com os pontos virtuais zerados. Para garantir diversidade, durante a cópia é introduzida uma probabilidade de mutação. Esta mutação é implementada ao permitir, com certa (baixa) probabilidade, que uma das estratégias do novo jogador seja descartada e uma nova, aleatória, tome seu lugar. Com isso garante-se que todo espaço de busca esteja disponível para o jogo aprender novas estratégias. O artigo conclui que este algoritmo permite um aprendizado durante o jogo, uma vez que a variância total diminui com o passar das rodadas até estabilizar-se em um patamar. Adicionalmente, mostra-se que se aos agentes é permitido alterar seu tamanho de memória, então uma “corrida armamentista” ocorre causando um crescente aumento no tamanho médio de memória. Porém, nenhuma comparação com o MG original é feita e, particularmente, nenhuma relação da variância com o tamanho de memória dos agentes utilizando este algoritmo de aprendizagem é argumentada. É também interessante notar que, em artigos subsequentes dos mesmos autores, este tipo de algoritmo é incrementalmente negligenciado em favor de maiores estudos utilizando o algoritmo tradicional com estratégias fixas.

Em (SYSI-AHO; CHAKRABORTI; KASTI, 2003) a idéia de “evoluir” as estratégias retorna e propõe-se um *algoritmo genético* para lidar com o aprendizado nos agentes. No algoritmo proposto, a cada determinado número de rodadas os  $n$  piores agentes invocam o algoritmo genético para modificar suas estratégias. O algoritmo genético, por sua vez, escolhe aleatoriamente duas estratégias e realiza um *crossover* de um ponto nestas, gerando dois novos filhos que substituem os pais. Assim, os piores jogadores têm periodicamente suas estratégias modificadas. Nos experimentos utilizando este algoritmo observa-se, como em (ZHANG; CHALLET, 1997), que a variância se reduz com o passar das rodadas em uma simulação. Na data de publicação deste artigo, a relação da variância e o tamanho da memória dos agentes já era bem conhecida e o artigo traz dois gráficos relacionando estas duas quantias. Surpreendentemente, observa-se que apesar do algoritmo genético atingir variâncias inferiores ao algoritmo tradicional para baixos valores de  $M$ , o tradicional ainda obtém o menor valor absoluto no ponto de mínima variância.

O principal interesse dos casos relatados estava em verificar se e como a dinâmica do sistema se modifica com a utilização de um tipo de aprendizado diferente do originalmente proposto. Porém, ambos possuem uma característica fundamental: são capazes de gerar novas estratégias durante o jogo. Esta é uma forma de aprendizado que aqui chamaremos de *aprendizado criativo*, em contraste com o modelo estático original onde estratégias novas não podem entrar no jogo. Esta característica importante é deixada de lado em favor da análise dos resultados gerados e pouco ou nenhum estudo é feito sobre a relação exata entre este tipo de aprendizado e a dinâmica que surge.

Adicionalmente, os casos acima exigem uma intervenção global para funcionarem. Em (ZHANG; CHALLET, 1997) é o *pior jogador* que altera suas estratégias enquanto em (SYSI-AHO; CHAKRABORTI; KASTI, 2003) são os  $n$  *piores jogadores* que

as alteram. Isto não só dá aos jogadores uma informação extra não prevista no MG, sua eficácia em relação aos demais jogadores, como também exige que haja um mecanismo global que matenha um *ranking* global de agentes, criando uma centralização inconveniente.

De forma a resolver estes problemas, apresentamos um novo algoritmo de aprendizado criativo para, então, estudar sua influência na dinâmica do MG. O objetivo é, novamente, estudar a dinâmica do sistema na presença de algoritmos de aprendizado criativos. O interesse em aprendizado criativo vem da crença de que é um modelo mais realista: pessoas são capazes de descartar hipóteses que não funcionaram no passado e formular novas para utilizar no futuro.

Nossa intenção principal, porém, é ter um algoritmo bastante simples que permita estudar seus parâmetros e relacioná-los à dinâmica do sistema, permitindo uma maior introspecção na interrelação entre aprendizagem em geral e a dinâmica no MG. Da mesma forma que o MG reduziu o El Farol à sua essência, é nosso objetivo fazer o mesmo para algoritmos de aprendizagem criativos.

## 6.1 Um modelo básico de aprendizado criativo

O modelo que estamos interessados em construir deve ser *mínimo*, isto é, deve adicionar ao modelo convencional o mínimo possível para implementar um aprendizado criativo. Para tanto, mantemos exatamente a mesma estrutura do MG, apenas adicionando um estágio extra de aprendizagem.

*Para cada agente, com probabilidade  $\tau$ :*

1. Ordene as estratégias em ordem de *fitness*;
2. Para as  $R$  piores estratégias e para cada bit destas, com probabilidade  $p_m$ , inverta o bit.

Trata-se de um procedimento bastante simples mas que, ainda assim, permite que sistematicamente estratégias com baixa performance sejam modificadas. Pode-se observar que trata-se de um algoritmo *evolutivo*, o que é natural para a estrutura do problema, uma vez que o elemento de competição entre as estratégias no modelo original já se faz presente.

O algoritmo aqui apresentado se destaca dos apresentados anteriormente não só pela sua maior simplicidade, mas também: não necessita uma ordenação global dos agentes como em (SYSI-AHO; CHAKRABORTI; KASTI, 2003), cada agente manipula suas estratégias de forma independente dos demais; ao contrário do proposto em (ZHANG; CHALLET, 1997), pode-se controlar, através de  $p_m$ , o quão diferente das predecessoras são as novas estratégias entrando no jogo.

Três parâmetros são facilmente identificáveis neste algoritmo: a probabilidade de aplicação das regras ( $\tau$ ), o número de estratégias descartadas a cada rodada ( $R$ ) e a probabilidade de mutação  $p_m$ . É interessante observar que o algoritmo não altera o *fitness*

de nenhuma estratégia, incluindo a que é modificada. O argumento para tanto é que, para  $p_m$  baixos, a estratégia modificada permanece bastante semelhante à original e, do ponto de vista do agente, pode-se esperar que tenha o mesmo desempenho desta. Em (ZHANG; CHALLET, 1997) estratégias novas iniciam com *fitness* zero pois as novas estratégias entrando no jogo são totalmente aleatórias. Veremos adiante que este conceito de *herança de fitness* é crucial para o bom desempenho do algoritmo.

Com este algoritmo, desejamos responder às seguintes perguntas: como se comporta a dinâmica do sistema quando alteramos os diversos parâmetros de aprendizado? O quão dependente do algoritmo de aprendizagem são as propriedades do MG? Como se compara os resultados obtidos com outros algoritmos de aprendizado da literatura?

Não estamos interessados porém, neste momento, em avaliar a *performance* do sistema ou de seus agentes, sendo a análise feita com o intuito de observar comportamentos qualitativos. Porém, utilizaremos o MG tradicional como descrito em (ZHANG, 1998) para comparações.

## 6.2 Metodologia

Soluções ou predições analíticas acerca da dinâmica do MG são consideradas não-triviais, exigindo simplificações que potencialmente causam danos à generalidade da solução. Frequentemente são utilizadas experimentações (simulações) em computador para gerar dados, posteriormente sendo feita uma análise sobre estes. Essa é a abordagem adotada aqui. Iniciaremos com uma série de experimentos e analisaremos os dados gerados por estes, comparando-os a resultados previamente evidenciados na literatura. Nosso interesse é na principal característica do MG, a capacidade de auto-organização, mas outras propriedades serão também mencionadas e analisadas quando necessário.

Para todos experimentos realizados, fixamos os seguintes parâmetros para o modelo:

- $N = 101$ . Este é o valor mais amplamente utilizado na literatura e fornece uma boa performance nas simulações.
- $T = 5000$ . Valores próximos a esse são comumente adotados na literatura.
- $t_0 = 1000$ . O parâmetro  $t_0$  indica a rodada em que o sistema é considerado estar fora do período de transiência inicial, isto é, a partir do qual os efeitos devido à inicialização aleatória desaparecem e apenas informações endógenas ao sistema são consideradas. Uma vez que a inicialização afeta principalmente o conteúdo inicial do histórico, adotamos  $t_0 \gg M$ .
- $S = 2$ . Diversas fontes na literatura (e.g. (MANUCA et al., 2000)) indicam que as propriedades qualitativas verificadas no MG se mantém de forma independente de  $S$  desde que  $S \geq 2$ . Assim, adotamos o caso mais simples.

A fixação de  $S = 2$  impõe também limites para o número de estratégias modificadas quando da execução do algoritmo de aprendizagem. Assim,  $R = S/2 = 1$  ou  $R = S$ .

No primeiro caso, temos sempre duas estratégias e a pior entre estas é sempre modificadas. No segundo caso, todas estratégias são modificadas. O caso onde  $R = 0$  não é de interesse, pois ao não modificarmos estratégias reduzimos o modelo ao MG tradicional.

Os dados apresentados em cada experimentos são resultados de uma média sobre 20 diferentes execuções com diferentes inicializações, exceto onde explicitamente indicado.

A implementação do sistema foi realizada em linguagem C++ e executada em arquiteturas x86. O algoritmo, como implementado, possui complexidade exponencial com  $M$  e, portanto, evitou-se realizar experimentos com valores muito altos desta variável. A complexidade exponencial foi verificada empiricamente e a Tabela 6.1 traz valores de tempo de execução do algoritmo para diversos valores de  $M$  ao ser executado em um processador Intel Celeron de 2GHz.

Tabela 6.1: Tempo de execução de  $T = 5000$  rodadas em função de  $M$  para a implementação do algoritmo utilizada.  $N = 101$  e  $S = 2$ .

$M$	tempo de execução (segundos)
2	0
4	2
6	22
8	321
10	4639

## 6.3 Experimentos

### 6.3.1 Efeitos sobre $A(t)$

Iniciamos nosso estudo verificando qual o efeito de nosso algoritmo de aprendizado criativo no MG. A Fig. 6.1 mostra a evolução da variância com o passar das rodadas em uma típica simulação com  $M = 2$ ,  $\tau = 1$  e  $p_m = 0.5$ , mostrando também a mesma evolução para o MG tradicional. Observa-se que ao utilizar-se o algoritmo de aprendizado proposto a variância mostra uma redução efetiva com o passar das rodadas, indicando que os agentes estão se adaptando e melhor organizando-se de forma a reduzir a variância total do sistema e, conseqüentemente, aumentando a eficiência deste. Este é um resultado interessante, pois mostra que agentes no MG capazes de adaptação unicamente em benefício próprio podem levar a uma adaptação global do sistema e a uma melhora relacionada na eficiência do sistema.

Porém, esse experimento apenas mostra que há uma redução *relativa* na variância. Isto é, a variância apresenta uma redução em relação aos valores em que a simulação se inicia. É interessante verificar como se comparam esses valores em relação ao MG tradicional. Para realizar esta verificação é necessário estudarmos a influência dos diversos parâmetros do algoritmo na dinâmica do sistema e a comparação com o MG tradicional será feita paralelamente nas subseções seguintes.

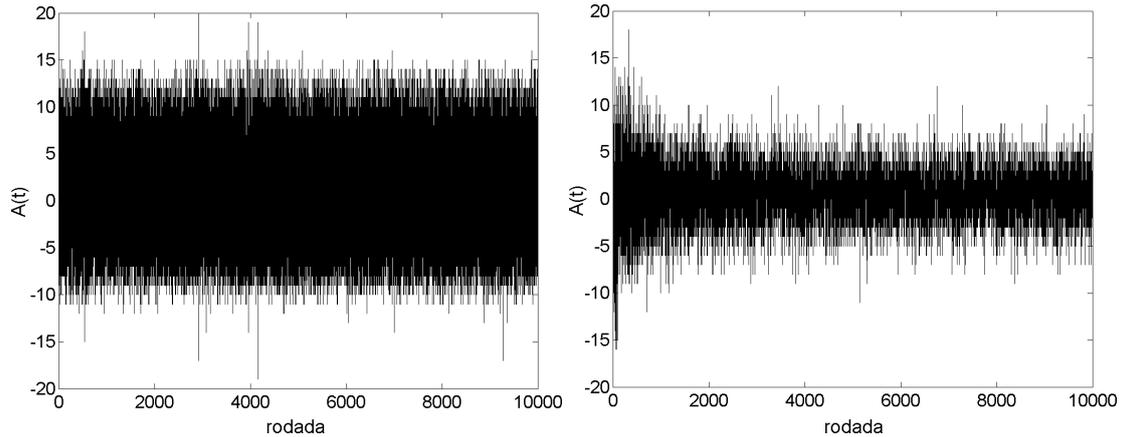


Figura 6.1:  $A(t)$  no MG tradicional (esquerda) e  $A(t)$  utilizando-se o algoritmo de aprendizado criativo com  $\tau = 1.0$  e  $p_m = 0.3$  (direita).

### 6.3.2 Efeitos de $\tau$

O primeiro parâmetro do algoritmo proposto que desejamos estudar é a probabilidade de aplicação do algoritmo de aprendizagem ( $\tau$ ). Este parâmetro é equivalente a uma taxa de aplicação do aprendizado criativo, isto é, indica a frequência com que o algoritmo é invocado. Antes de analisar os dados gerados pelos experimentos, é interessante compreender a influência de  $\tau$  sobre o MG.

Para  $\tau = 0$ , o algoritmo de aprendizado criativo nunca é invocado, nenhuma estratégia é modificada durante todo o experimento. O modelo torna-se, então, idêntico ao MG tradicional: cada agente possui um conjunto fixo de estratégias e as mantém para sempre. No outro extremo, para  $\tau = 1$ , o algoritmo é invocado com máxima frequência. Como consequência, a escolha de quais poderão ser modificadas é feita com base na performance acumulada apenas durante a rodada anterior. Para  $0 < \tau < 1$ , temos casos intermediários e estratégias são modificadas com base em sua performance acumulada em *pelo menos* uma rodada, sendo o número exato inversamente proporcional a  $\tau$ .

Para os experimentos abaixo, fixamos  $p_m = 0.5$ . Nas medições realizadas, uma vez que o algoritmo permite a adaptação e conseqüente redução da variância com o passar das rodadas, o cálculo da variância para um determinado valor de  $M$  é feito a partir de  $t = t_0 = 1000$ , onde o sistema já se encontra com variância estável, e não a partir de  $t = 0$ .

A Fig. 6.2 mostra os resultados relacionando a variância com o tamanho de memória dos agentes para os casos  $\tau = 1$  e  $\tau = 0$ . O gráfico para  $\tau = 0$  tem a forma característica do MG tradicional, o que era esperado e está representado para fins de comparação. Já para  $\tau = 1$  observa-se uma mudança drástica na dinâmica. O ponto de mínima variância passa a coincidir com o menor tamanho de memória apresentando neste ponto, ainda, um valor inferior a qualquer valor atingido pelo MG tradicional. Notavelmente, para nenhum valor de  $M$  o sistema se comporta de forma *pior* que o esperado para o caso aleatório, sendo este o limite para valores grandes de  $M$ .

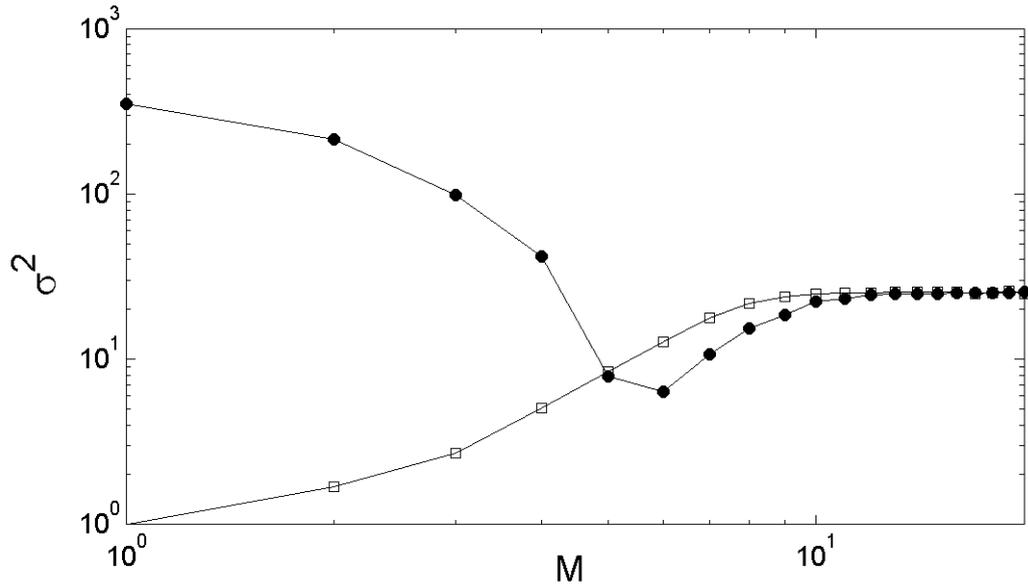


Figura 6.2: Variância em função de  $M$  para o MG utilizando aprendizado criativo. Círculos são para  $\tau = 0$  e quadrados para  $\tau = 1$  e  $p_m = 0.5$ .

Na Fig. 6.3 podemos ver o comportamento do sistema para vários valores de  $\tau$ . O que se observa é que a dinâmica transita suavemente entre a tradicional e a observada nos experimentos anteriores. Quanto maior a frequência de aplicação do algoritmo de aprendizado criativo (i.e. quanto maior  $\tau$ ), mais o sistema destoa do MG tradicional, indicando que a mudança no comportamento é de fato fruto do algoritmo. Observa-se também que as menores variâncias são obtidas para maiores valores de  $\tau$ .

### 6.3.3 Efeitos de $p_m$

Outro parâmetro importante no algoritmo é  $p_m$ , a probabilidade de alteração da pior estratégia de cada agente. Este parâmetro permite controlar o quanto a pior estratégia será alterada, sendo que com  $p_m = 0$  a estratégia permanece sempre inalterada e com  $p_m = 1$  todos os bits são alterados, isto é, a estratégia é *invertida*. O primeiro caso novamente reduz o sistema ao MG tradicional já que nunca há alterações em quaisquer estratégias. A Fig. 6.4 mostra a relação obtida entre  $p_m$  e a variância do sistema quando mantemos os demais parâmetros fixos. Observa-se uma decréscimo na variância conforme o aumento de  $p_m$  até  $p_m \approx 0.3$ . A partir deste ponto não há ganhos na eficiência com maiores aumentos no parâmetro, inclusive para valores próximos de  $p_m = 1$ , onde esperar-se-ia que a simples inversão de estratégias não traria benefícios muito superiores ao uso de estratégias fixas por ser uma exploração pobre do espaço de estratégias possíveis. Porém, esta característica fornece uma importante evidência para compreender como se dá a auto-organização do sistema, como será visto na Seção 6.3.6.

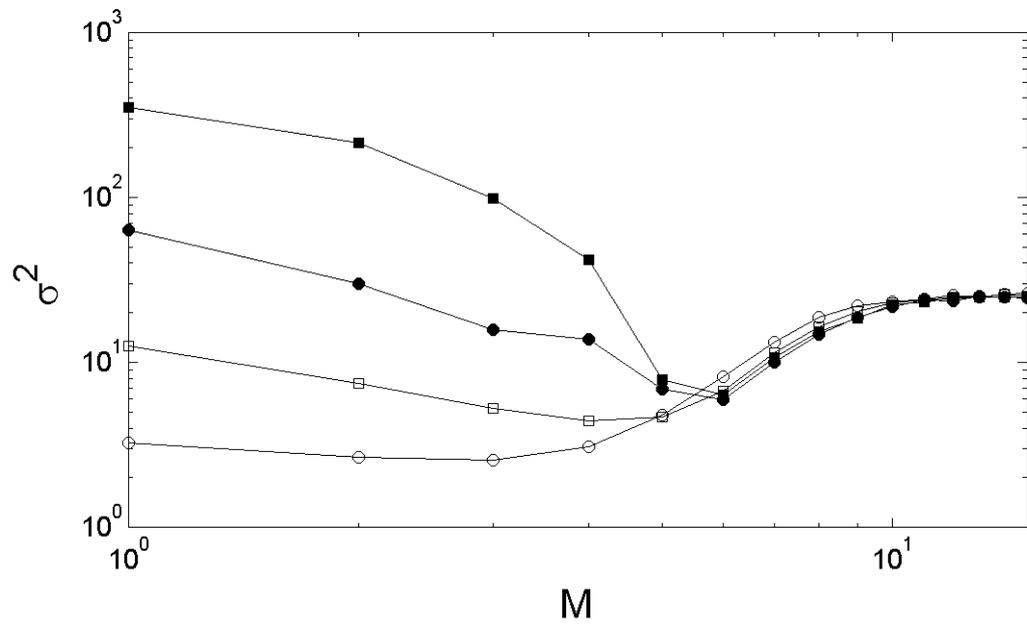


Figura 6.3: Variância em função de  $M$  para  $\tau = 0.01$ ,  $\tau = 0.1$ ,  $\tau = 0.3$  e  $\tau = 0.5$  (respectivamente: quadrados pretos, círculos pretos, quadrados brancos e círculos brancos).  $p_m = 0.5$ .

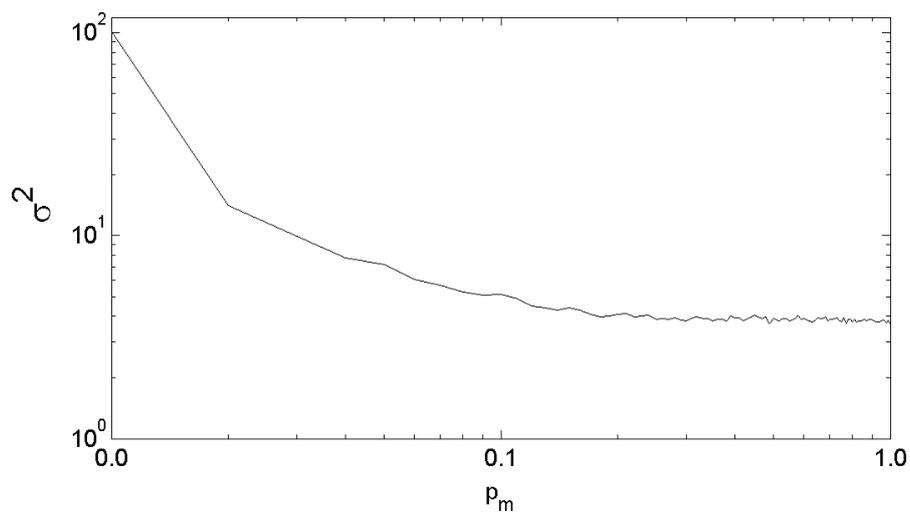


Figura 6.4: Variância em função de  $p_m$ .  $M = 7$ ,  $S = 2$ ,  $\tau = 1.0$ .

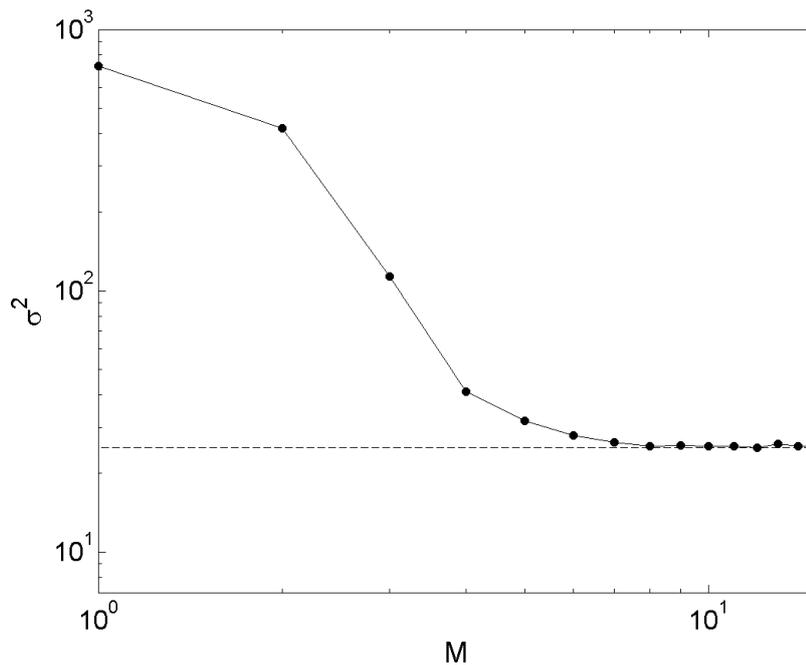


Figura 6.5: Variância em função de  $M$  utilizando o algoritmo modificado onde novas estratégias iniciam com *fitness* zero,  $p_m = 0.5$  e  $\tau = 1$ .

### 6.3.4 Efeitos da atribuição de *fitness*

Ao apresentarmos o algoritmo, foi dito que o *fitness* de uma estratégia modificada se mantém inalterado, contrastando com outros algoritmos na literatura onde comumente novas estratégias iniciam com *fitness* zero (e.g. (ZHANG; CHALLET, 1997)). Para compreender a importância deste fator na dinâmica induzida pelo algoritmo a Fig. 6.5 mostra os resultados de experimentos onde o algoritmo de aprendizado é alterado para atribuir *fitness* zero às estratégias modificadas. O que se vê é que a dinâmica novamente se altera, apresentando apenas variâncias acima do esperado para o caso aleatório para qualquer tamanho de memória. Uma vez que a auto-organização do MG é atribuída exatamente à sua capacidade de apresentar variâncias inferiores ao caso aleatório, conclui-se que, ao atribuímos *fitness* zero às estratégias modificadas, o sistema é incapaz de se auto-organizar. Assim, confirma-se que a “herança” de *fitness* tem papel fundamental na eficiência do algoritmo de aprendizado proposto.

### 6.3.5 Efeitos da seleção

É interessante neste momento também verificarmos a influência de outros aspectos do algoritmo no comportamento observado. Um ponto a ser verificado é se, no algoritmo proposto, é necessário modificar apenas as *piores* estratégias. Pode-se questionar se o efeito alcançado não é fruto unicamente da inserção de novas estratégias no jogo, não importando se modificamos as melhores, piores ou quaisquer estratégias. Para resolver tal questão, novos experimentos foram realizados com versões alteradas do algoritmo

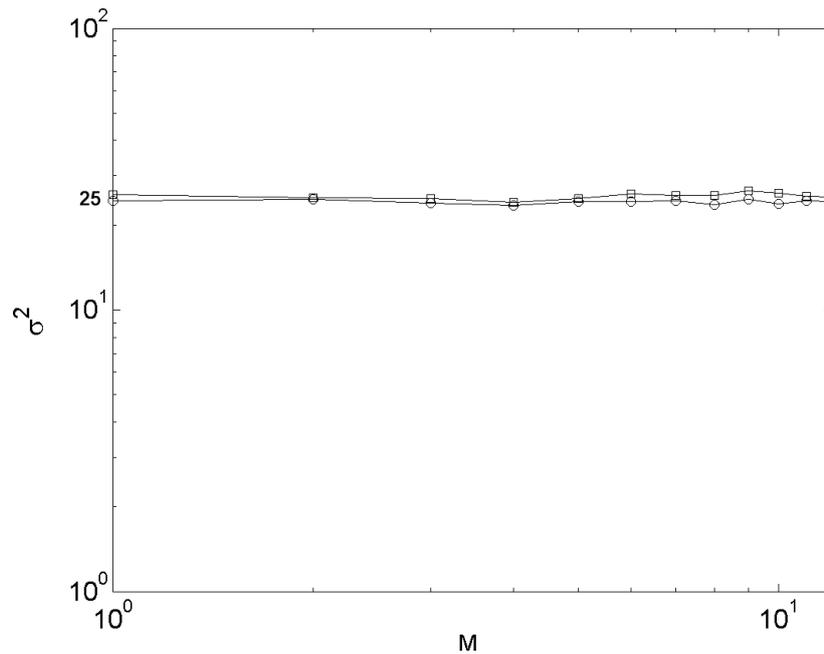


Figura 6.6: Variância em função de  $M$  utilizando algoritmos modificados de aprendizagem. Quadrados representam o algoritmo modificando as melhores estratégias enquanto círculos representam escolhas aleatórias de estratégias para serem modificadas.

original proposto.

Em um primeiro caso, alteramos o algoritmo para modificar as  $R$  melhores estratégias. Assim, executamos uma seleção inversa onde as piores estratégias permanecem intactas (i.e. sobrevivem). Em um segundo caso, alteramos o algoritmo para que escolha aleatoriamente  $R$  estratégias para serem modificadas. A Fig. 6.6 traz os resultados. Para ambos os casos observamos que a variância atingida é, para todos valores de  $M$ , muito próxima a esperada para o caso aleatório, isto é, o algoritmo passa a fazer com que os agentes optem aleatoriamente pelos grupos. Assim, confirma-se que a seleção aplicada de modo a manter as melhores estratégias tem papel fundamental na melhor eficiência apresentada.

### 6.3.6 Comportamento das estratégias

Passamos agora a tentar compreender a questão de como a melhor eficiência observada é alcançada pelos agentes utilizando o algoritmo de aprendizado criativo proposto. O algoritmo tem como suas estruturas de aprendizado as estratégias e, portanto, desejamos estudar com maior detalhe como estas são modificadas durante o jogo, sob diversos parâmetros.

Começamos questionando se há uma convergência nas estratégias, isto é, se com o passar das rodadas as estratégias convergem para regiões específicas no espaço de estratégias.

Podemos analisar esta questão através da distribuição de estratégias nesse espaço. Como foi visto, cada estratégia pode ser representada por um vetor binário. Portanto,

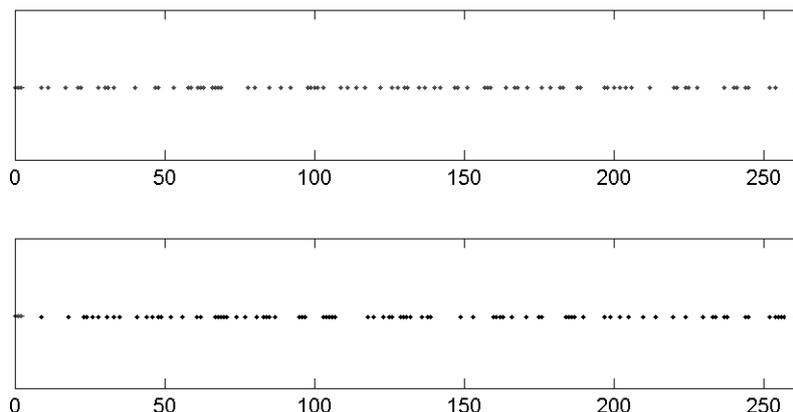


Figura 6.7: Espaço de estratégias na primeira rodada do jogo (gráfico superior) e espaço de estratégias na última rodada do jogo (gráfico inferior). Cada marca representa a melhor estratégia de cada agente, convertida para uma representação decimal.

podemos facilmente representar cada estratégia possível por um número decimal único e definimos um espaço de estratégias unidimensional com valores entre 0 e  $2^{2^M} - 1$ . A inicialização aleatória das estratégias no MG garante que o espaço seja preenchido de forma aproximadamente uniforme e estamos interessados nesta distribuição ao final de uma simulação. Por exemplo, poderia ser o caso de as estratégias concentrarem-se em alguns pontos, indicando serem áreas que garantem o sucesso de agentes. A Fig. 6.7 mostra a evolução da melhor estratégia de cada agente em uma típica simulação do algoritmo proposto. Nenhuma convergência no espaço de estratégias pode ser observada e a preferência é bastante uniforme em toda amplitude de possibilidades, indicando que o aprendizado não advém da descoberta de estratégias específicas por parte dos agentes.

A mesma distribuição aproximada é observada ao final de simulações se alteramos o algoritmo para que, em sua inicialização, todas estratégias partam de um mesmo ponto, isto é, sejam idênticas entre si. Partindo-se de um único ponto, o algoritmo de aprendizado distribui estratégias de forma aproximadamente uniforme por todo o espaço de estratégias. Se pensarmos na principal característica do MG, a regra da minoria, estes resultados são esperados. No MG, regras de sucesso são aquelas utilizadas por poucos agentes e uma regra que garantiu sucesso até determinado momento pode provar-se ruim se muitos outros agentes passarem a adotá-la também. Isso gera um incentivo à *diferenciação* das estratégias, onde cada agente procura ter uma estratégia possuída pelo menor número de outros agentes possíveis, o que implica em uma distribuição uniforme no espaço de estratégias, o que é efetivamente observado.

Apesar de não haver convergência para estratégias específicas, a maior parte dos agentes parece optar por *alguma* estratégia e não mais alterá-la a partir de determinado momento. A Fig. 6.8 mostra a alternância entre estratégias de um agente típico escolhido aleatoriamente. Observa-se que após um período de ajuste, o agente não mais troca de estratégia, preferindo utilizar uma até o fim da simulação. É importante salientar que o uso de uma única estratégia não implica em o agente sempre optar apenas por um dos grupos, mas implica que ele responderá sempre da mesma maneira para um mesmo padrão de

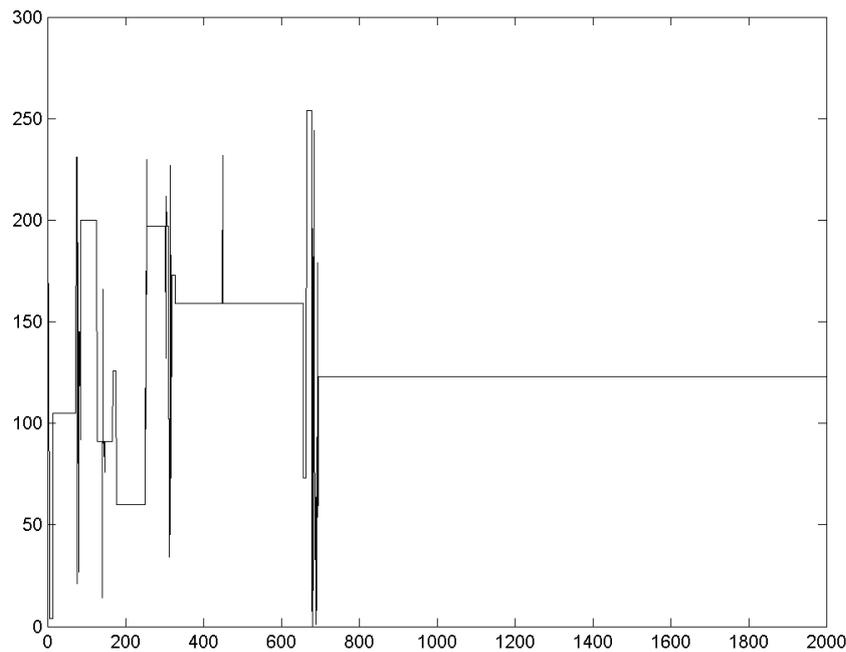


Figura 6.8: Estratégia optada por um agente em cada rodada de uma simulação.

estímulo presente no histórico. Na Fig. 6.9 pode-se observar que menores variâncias são mais observadas quando um maior número desses agentes congelados estão presentes no sistema. Esta relação é ainda reforçada na Fig. 6.10 e já foi observada em (ZHANG; CHALLET, 1997) para o MG tradicional. Aqui observamos a validade também para o MG com aprendizado criativo.

Também é de interesse reparar em como evoluem os agentes utilizando estratégias fixas. A Fig. 6.11 mostra exatamente isto, relacionando a fração de agentes utilizando estratégias fixas com o passar das rodadas. Observa-se que o número de agentes congelados aumenta progressivamente com o passar das rodadas, saturando próximo a  $N$ .

O incremento no número de agentes utilizando uma única estratégia indica que esta é um comportamento vantajoso. De forma a comprovar esta hipótese, na Fig. 6.12 mostramos a evolução do *fitness* médio dos agentes congelados e não congelados. De fato, para valores abaixo de  $M_{al}$ , fixar uma estratégia mostra-se vantajoso. Porém, a partir do momento que a variância total do sistema passa a se igualar ao caso aleatório (para  $M > M_{al}$ ), manter uma estratégia fixa torna-se subitamente ruim. Isto acontece devido a, na região  $M > M_{al}$ , os agentes estarem decidindo por grupos de forma aleatória e, neste caso, o melhor que um agente qualquer pode fazer é também decidir aleatoriamente. Manter uma estratégia fixa torna impossível para um agente aleatorizar sua decisão e, portanto, os que o fazem obtêm um menor *fitness* em média. Este é também o motivo de, na Fig. 6.10, existir uma súbita queda na fração de agentes congelados ao nos aproximarmos de  $M_{al}$ .

O congelamento de estratégias pode ser entendido como uma espécie de acordo entre os agentes, uma sincronização. A estratégia escolhida por cada agente é aquela que

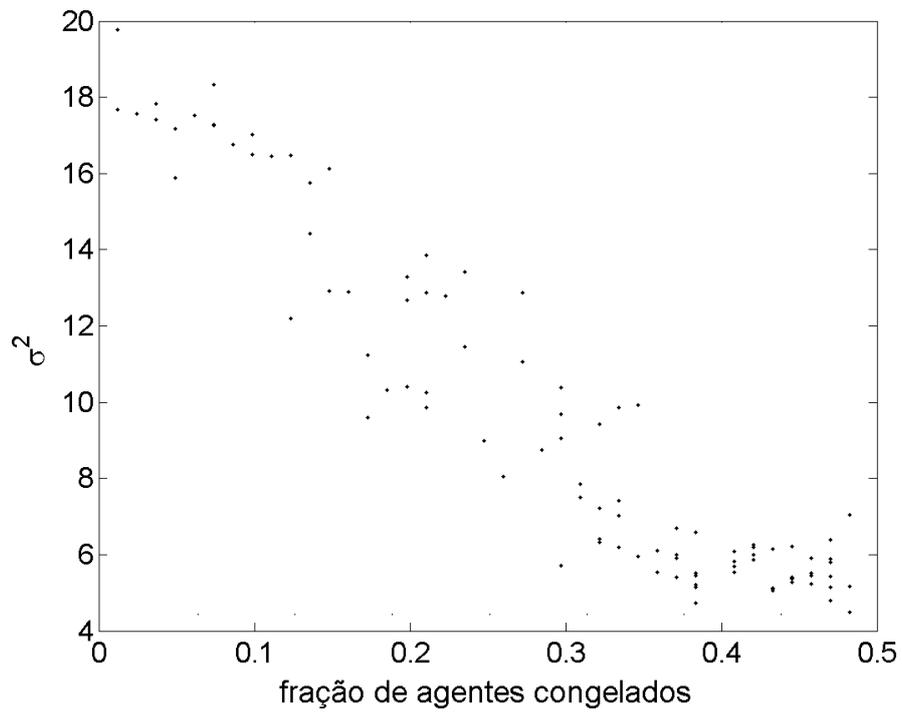


Figura 6.9: Variância do sistema *versus* o número de agentes congelados para diversas simulações com diferentes inicializações.

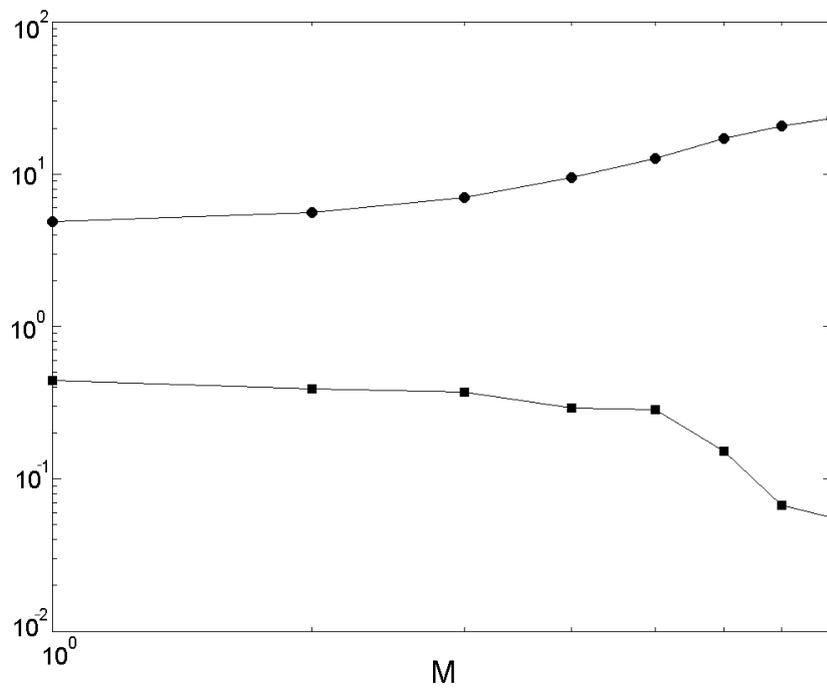


Figura 6.10: Variância do sistema (círculos) e a fração de agentes congelados (quadrados) em função de  $M$ .

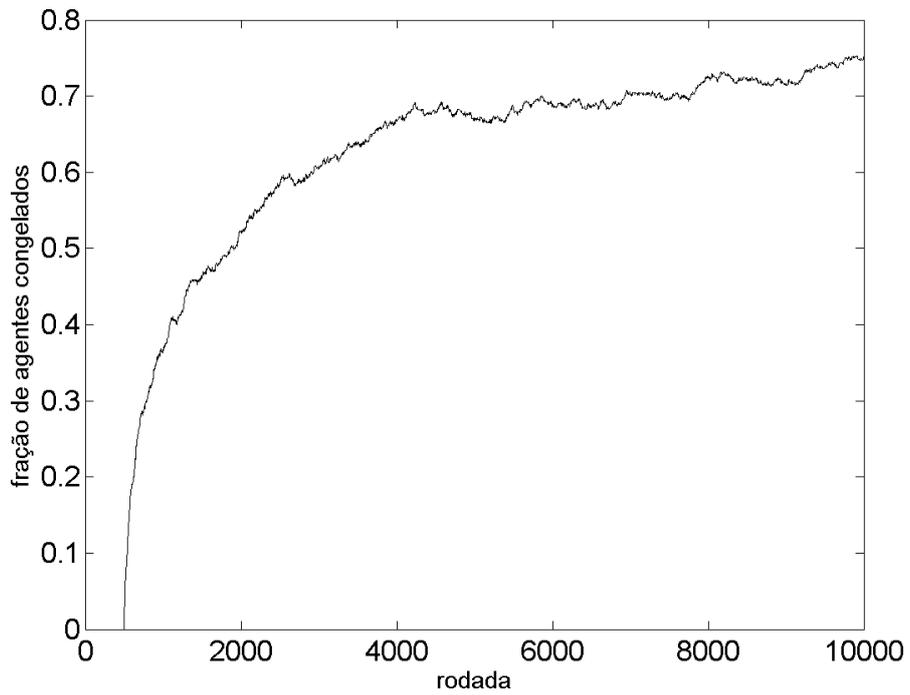


Figura 6.11: Evolução da fração de agentes congelados.  $M = 3$ ,  $\tau = 1.0$ .

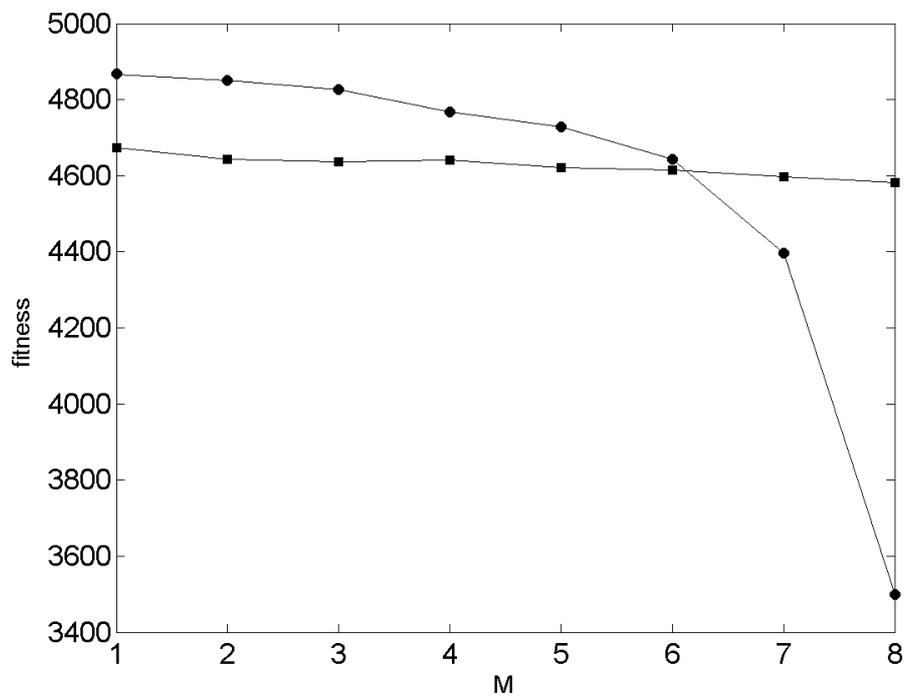


Figura 6.12: *Fitness* médio dos agentes congelados (círculos) e dos agentes não-congelados (quadrados) para vários valores de  $M$ .

melhor lhe fornece resultados no contexto das estratégias utilizadas pelos outros agentes. Isto é, as estratégias fixas são de certa forma respostas a um equilíbrio de Nash, no sentido de que não há outra estratégia que o agente pode utilizar que lhe daria resultados melhores. Diferentemente da Teoria dos Jogos, porém, os agentes aprendem este fato efetivamente tentando outras estratégias e não através de dedução. Após algumas rodadas testando estratégias alternativas, o agente finalmente encontra uma que é a melhor resposta às estratégias de outros agentes até então e passa a utilizar somente esta.

Uma explicação sobre que característica especial possuem as estratégias utilizadas pelos agentes congelados é evidenciada se considerarmos a evolução da “distância” entre as melhores estratégias de todos agentes. Esta distância pode ser calculada valendo-se do fato de que as estratégias são vetores binários o que permite o cálculo da distância de Hamming entre estas. A distância entre todas melhores estratégias é definida como (CHALLET; ZHANG, 1998):

$$d = \frac{1}{2^m(N-1)^2} \sum_{i \neq j} \langle \|\vec{st}_i - \vec{st}_j\| \rangle$$

onde  $\|\cdot\|$  define a distância de Hamming no espaço  $2^m$  de estratégias e  $\vec{st}_i$  o vetor característico da melhor estratégia do agente  $i$ .

A Fig. 6.13 mostra evolução de  $d$  com o passar das rodadas. Observa-se que há a tendência à diferenciação das estratégias, como já foi observado quando analisamos a evolução do espaço de estratégias. Os agentes optam por utilizar estratégias que são o mais diferentes possíveis das estratégias utilizadas pelos demais agentes. Assim, conclui-se que as estratégias fixadas por agentes congelados são aquelas que maximizam a sua distância em relação às demais. O algoritmo de aprendizado não faz a busca por estratégias específicas, mas sim por uma estratégia que maximize esta distância.

A maior eficiência observada utilizando o algoritmo criativo aqui proposto, portanto, parece advir da maior habilidade do algoritmo em maximizar a distância entre as estratégias utilizadas.

## 6.4 Discussão dos resultados

Os experimentos realizados neste capítulo visaram explorar a relação entre aprendizado e dinâmica no MG. Enquanto propriedades e comportamentos do sistema como um todo foram o principal foco, também foram realizados experimentos com o intuito de explorar como os agentes aprendem utilizando um algoritmo de aprendizado criativo, isto é, capaz de gerar novas estratégias com o passar do tempo. Dos experimentos realizados é possível retirar algumas conclusões.

### 1. Propriedades do MG são dependentes do algoritmo de aprendizado

Utilizando nosso algoritmo de aprendizado criativo obtivemos propriedades macroscópicas bastante diferentes do MG tradicional. Vimos que a curva de eficiência contraposta ao tamanho de memória dos agentes tem um comportamento distinto e variável

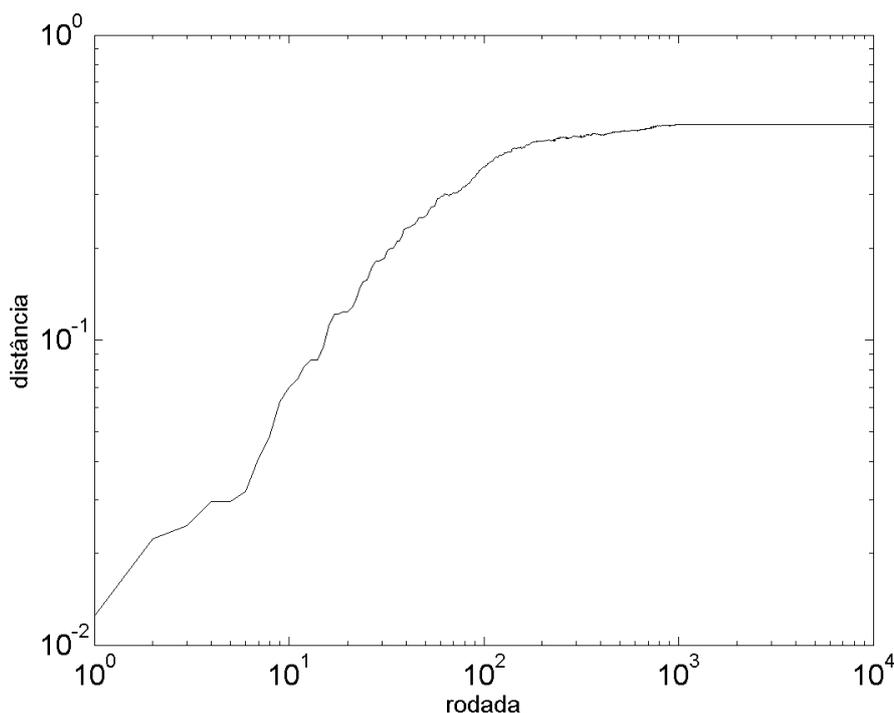


Figura 6.13: Evolução da distância entre as estratégias utilizadas em um jogo, quando todas estratégias iniciam no mesmo ponto (i.e. idênticas).  $M = 3$ .

de acordo com os parâmetros do algoritmo, mostrando-se particularmente sensível à atribuição de *fitness* às estratégias novas. Ao considerarmos a atribuição de *fitness* zero às novas estratégias, procedimento comumente adotado na literatura revista, passa-se a ter um sistema onde não há indícios de auto-organização, uma vez que em nenhum momento são obtidas variâncias melhores do que o caso aleatório. Estes resultados são importantes por mostrar a criticalidade do algoritmo de aprendizagem utilizado no comportamento macroscópico do sistema, uma vez que é razoável assumir que isto também é válido para outros sistemas semelhantes, inclusive os de inteligência de enxame.

## 2. Maior eficiência é possível ao utilizar-se aprendizado “criativo”

Ao permitirmos que nosso algoritmo herdasse o *fitness* das estratégias predecessoras, obtivemos variâncias bastante inferiores às do MG tradicional e, portanto, maior eficiência. Adicionalmente o sistema foi capaz de se auto-organizar (i.e. obter variâncias melhores do que o caso aleatório) para uma faixa maior de tamanhos de memória, com pico de eficiência em tamanho de memória inferior ao tradicional, demonstrando uma maior eficácia do algoritmo no uso da informação disponível. Analisamos a causa dessa melhor eficiência, descartando a existência de estratégias “ótimas” e atribuindo a melhora ao aumento de agentes “congelados”, isto é, que após certo tempo passam a utilizar uma única estratégia. Foi observado que as estratégias congeladas não reduzem o desempenho dos agentes, indicando uma possível coordenação de estratégias onde dois grandes grupos de agentes se alternam no grupo da minoria. Verificamos que o algoritmo permite que as estratégias utilizadas diferenciem-se durante uma simulação, com a intenção de

maximizar a distância entre as estratégias em jogo.

### **3. Algoritmo proposto é eficaz para estudos do aprendizado no MG**

O algoritmo simples de aprendizado que apresentamos mostrou-se eficaz para as tarefas de interesse. Com um único algoritmo, através da alteração de seus parâmetros, obtivemos comportamentos qualitativamente semelhantes as de outros algoritmos mais complexos descritos na literatura (e.g. (ZHANG; CHALLET, 1997; SYSI-AHO; CHAKRABORTI; KASTI, 2003)), indicando que a simplicidade obtida levou também a uma maior generalização do processo de aprendizado criativo, tornando possível analisar a influência específica de fatores e parâmetros do algoritmo na dinâmica do MG. Adicionalmente, com o controle de um único parâmetro, pode-se alternar o regime entre o MG tradicional e um onde o aprendizado criativo é predominante.

Resultados parciais dos experimentos apresentados neste capítulo foram publicados em (ARAÚJO; LAMB, 2004b).

## 7 ANÁLISE DE AGENTES NO MINORITY GAME

Este capítulo passa a analisar o *Minority Game* (MG) do ponto de vista dos agentes, estudando como os parâmetros individuais alteram a performance de um agente em comparação com outros. Damos particular atenção ao estudo do tamanho de memória dos agente, verificando em que circunstâncias é realmente vantajoso ter uma memória maior como é usualmente argumentado (e.g. Zhang (ZHANG; CHALLET, 1997)). Fazemos a análise de agentes utilizando tanto o aprendizado tradicional como o aprendizado criativo, como proposto no Capítulo 7.

### 7.1 Introdução

No capítulo anterior estudamos como agentes homogêneos, capazes de aprendizado criativo, alteram a dinâmica do MG. O estudo foi feito de um ponto de vista coletivo, analisando o sistema como um todo ou propriedades aplicáveis a todos agentes. Era nosso interesse traçar relações entre a influência do tipo de aprendizado de agentes e a dinâmica que surge da interação destes.

No presente capítulo, desejamos mudar a perspectiva e passar a estudar um único agente específico em um ambiente de agentes heterogêneos. Assim, não mais nos interessamos tão somente pela dinâmica do sistema, mas também, e principalmente, em como o agente escolhido se comporta na interação com os demais. Mais especificamente, estamos interessados em responder perguntas do tipo: que condições e características são favoráveis para um agente *explorar* o sistema? Que características devem ter algoritmos de aprendizado para que tenham melhor performance nesses sistemas?

Para responder tal pergunta passamos a considerar os sistemas vistos no capítulo anterior como um *ambiente* no qual inserimos um agente que receberá especial atenção. O sistema como um todo passa a ser heterogêneo, podendo possuir agentes dotados de diferentes tipos de aprendizado ou diferentes parâmetros para seus algoritmos. Então, ainda no contexto do MG, analisamos a performance efetiva de cada tipo de agente em diferentes ambientes (i.e. entre diferentes tipos de agentes) de modo a compreender que características tornam um agente melhor e em que circunstâncias. Nesse sentido, o estudo que realizamos é inspirado nos trabalhos de Robert Axelrod sobre o Dilema do Prisioneiro (AXELROD, 1985).

Ao retirarmos o foco da dinâmica de sistemas para nos concentrarmos nos agentes

em si estamos nos afastando tanto da área de Física Estatística quanto de Economia para nos aproximar de áreas como Aprendizado de Máquina e Ciência Cognitiva. Assim, também nos afastamos de uma agenda *descritiva*, onde nos interessava compreender o porquê de sistemas se comportarem como se comportam, para uma agenda *prescritiva* na qual queremos prescrever como agentes *devem* se comportar (ou serem especificados) no contexto proposto. Esta mudança de foco é defendida de uma forma geral, entre outras agendas possíveis, por Shoham e Powers (SHOHAM; POWERS; GRENAGER, 2004).

## 7.2 Metodologia

De forma a cumprir os objetivos propostos para este capítulo, definimos um *ambiente* como sendo um MG composto de agentes homogêneos e um *agente-alvo* como sendo um agente no estilo do MG, mas com propriedades alteradas, isto é, diferente dos agentes do ambiente. Este tipo de configuração nos permite simular diferentes agentes, com diversos parâmetros, em diferentes contextos.

Desejamos estudar dois tipos diferentes de ambientes: o primeiro formado por agentes tradicionais (“ambiente tradicional”) e o segundo por agentes que utilizam o algoritmo de aprendizado criativo apresentado no capítulo anterior (“ambiente criativo”).

Para cada ambiente, estudaremos como se comportam agentes dos dois tipos (tradicional e criativo), sob diversos parâmetros. Utilizaremos, adicionalmente, um *agente aleatório*, que sempre toma decisões aleatórias, de forma a melhor comparar a performance dos agentes estudados. O agente aleatório permite uma referência em termos de performance, pois representa, como foi argumentado na Seção 5.1, a decisão prevista pela Teoria dos Jogos se o jogo fosse composto de apenas uma rodada.

Como medida da eficiência de um agente utilizaremos sua porcentagem de acertos, isto é, a porcentagem de rodadas que o agente esteve no grupo da minoria. Os resultados apresentados representam médias da execução de 20 simulações independentes com 5000 rodadas cada uma.

Devido ao crescimento exponencial com  $M$  do tempo de execução do algoritmo, como exposto na Seção 6.2, optou-se por realizar experimentos para apenas alguns valores de  $M$  e não para toda uma faixa de valores como foi feito no Capítulo 6.

## 7.3 Experimentos

### 7.3.1 Influência da memória

Iniciamos a análise considerando a seguinte pergunta: qual o papel do tamanho da memória de um agente na sua eficiência? No artigo original sobre o MG (ZHANG; CHALLET, 1997) já são relatados experimentos nesse sentido e levando à conclusão que maior memória é sempre benéfico. Porém, em (ZHANG; CHALLET, 1997) o estudo é realizado utilizando um ambiente heterogêneo, isto é, com agentes dotados de vários tamanhos de memória, o que significativamente pode alterar a dinâmica do sistema, o que não é considerado. Aqui estudaremos um único agente imerso em um ambiente

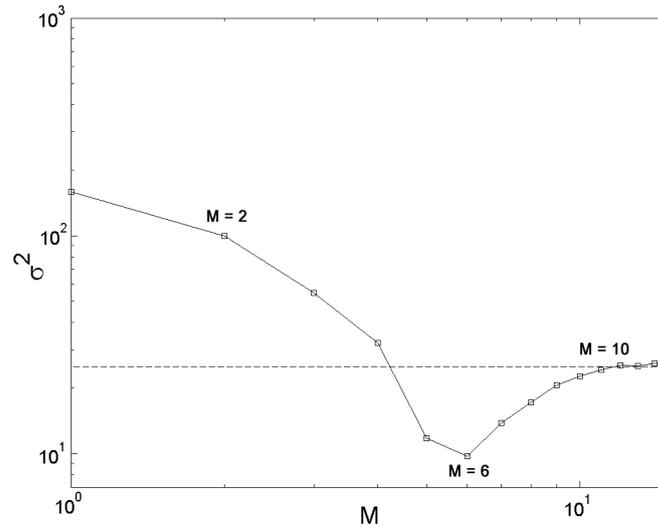


Figura 7.1: Regimes a serem estudados com ambiente tradicional.

homogêneo de modo a tornar mais evidente o papel de seu tamanho de memória. Este único agente-alvo, para valores altos de  $N$  (número total de agentes), não tem influência sobre a dinâmica do sistema sendo esta determinada totalmente pelo “ambiente” (i.e. os demais agentes).

### 7.3.1.1 Ambiente tradicional, agente tradicional

Especificamos o ambiente, inicialmente, como sendo um MG tradicional, isto é, composto por agentes que seguem o algoritmo de aprendizado tradicional com estratégias fixas. Estudaremos esse ambiente em três fases distinguíveis: o de alta ineficiência, o de máxima eficiência e o de eficiência esperada para o caso aleatório. Como no capítulo anterior, fixaremos  $N = 101$ , por ser este o valor mais utilizado na literatura, e tomaremos  $\alpha$  como sendo definido unicamente por  $M$ . A Fig. 7.1 mostra os pontos em que estamos interessados fornecendo, portanto, os valores para  $M$ : 2, 6 e 10. Em cada um dessas fases incluiremos um único agente-alvo com propriedades alteradas. Definimos  $m_a$  como sendo o tamanho da memória desse agente alvo e estudamos o seu desempenho para valores acima de, iguais a e abaixo de  $M$ .

Tabela 7.1: Resultados com ambiente e agente-alvo tradicionais.

$M$	média	$ac_{best}$	$ac_{rand}$	$m_a = 2$	$m_a = 4$	$m_a = 6$	$m_a = 10$	$m_a = 12$
2	41.62	50.02	49.52	42.08	46.16	50.04	55.34	60.82
6	47.28	53.94	43.53	45.30	46.60	47.30	45.21	44.46
10	46.08	50.56	45.15	46.40	45.80	45.78	46.04	46.60

A Tabela 7.1 mostra os resultados dos experimentos para a configuração descrita, onde:

- “ $M$ ” é o tamanho de memória dos agentes no ambiente;

- “média” é a média de acertos dos agentes no ambiente;
- “ $ac_{best}$ ” é a percentagem de acertos do *melhor* agente no ambiente;
- “ $ac_{rand}$ ” é a percentagem de acertos do agente aleatório;
- “ $m_a$ ” indica o tamanho da memória do agente-alvo;

Analisamos a seguir os resultados obtidos para cada valor de  $M$ .

## **M = 2**

Com  $M = 2$ , o sistema encontra-se em uma fase altamente ineficiente, com amplas variâncias muito acima do esperado para o caso aleatório. Pode-se verificar que neste caso, para  $m_a > M$ , o agente-alvo obtém acertos progressivamente superiores à média do sistema e se sai melhor que o agente aleatório. Como esperado, para  $m_a = M$  sua performance é semelhante à média. Assim, com o ambiente em sua fase ineficiente, pode-se concluir que um agente possuindo uma maior memória ( $m_a > M$ ) possui, em média, vantagens em relação aos demais agentes. É interessante também notar que, nesta configuração, o agente aleatório se sai melhor do que a média.

Este resultado pode ser colocado na forma de conceitos da *Teoria Evolucionária de Jogos* (SAMUELSON, 2003). Diz-se, nesta teoria, que uma população é *evolucionariamente estável* se esta não pode ser invadida por um indivíduo mutante (SAMUELSON, 2003), isto é, uma alteração em alguma característica de algum indivíduo não é vantajosa e não se propaga pela população, uma vez que a sobrevivência e reprodução depende do sucesso do indivíduo. De acordo com esta definição, observa-se que para  $M = 2$  o MG não é evolucionariamente estável, já que observa-se uma clara vantagem em se possuir um tamanho de memória maior.

## **M = 6**

Em  $M = 6$ , o sistema é caracterizado por baixas variâncias, abaixo do esperado para o caso aleatório. De fato, em  $M = 6$  temos a mínima variância do sistema sendo estudado. Para  $m_a \leq M$  obtém-se valores comparáveis à média e para  $m_a > M$  há uma clara queda na performance. Nesta fase, diferentemente do caso anterior, a performance do agente aleatório é inferior à média do ambiente e abaixo de 50%.

Este regime evidencia um fenômeno interessante: possuir tamanho de memória superior a  $M$  é danoso, mas é indiferente possuir tamanho de memória inferiores a  $M$ . Podemos concluir, assim, que  $M = 6$  é evolucionariamente estável: um mutante com  $m_a > M$  obviamente não se perpetua, por ter uma performance inferior aos demais agentes; um mutante com  $m_a < M$  poderia-se perpetuar, uma vez que tem performance similar ao caso com  $m_a = M$ , porém a medida que a população adota tamanhos menores de memória (por algum incentivo diverso) o sistema sai de sua fase eficiente e passa a se comportar como no regime anterior, onde maiores tamanhos de memória são vantajosos e se propagam, levando novamente o sistema à fase eficiente. Assim,  $M = 6$  é um ponto de equilíbrio e a população pode ser dita evolucionariamente estável.

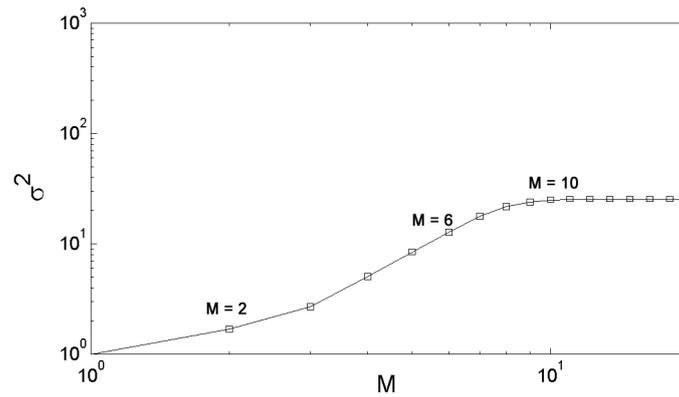


Figura 7.2: Regimes a serem estudados com ambiente criativo.

### **M = 10**

O último caso nesta configuração tem  $M = 12$  e o sistema encontra-se em sua fase aleatória, isto é, apresentando variâncias idênticas às que se esperaria se os agentes estivessem jogando de forma aleatória ( $\sigma^2 = N/4$ ). Pode-se observar que o agente-alvo apresenta performance relativamente constante, independente do valor de  $m_a$  e próxima da média. O agente aleatório também obtém performances semelhantes à média, indicando que de fato os agentes agem como se estivessem decidindo aleatoriamente. Decisões aleatórias não são dependentes do tamanho da memória e, portanto, este parâmetro passa a ser irrelevante para a performance do agente-alvo, o que se reflete na independência entre performance e  $m_a$  observado.

#### 7.3.1.2 Ambiente criativo, agente criativo

Passamos agora a estudar uma configuração onde tanto os agentes compondo o ambiente como o agente-alvo utilizam o algoritmo de aprendizado criativo proposto no capítulo anterior. Utilizaremos os mesmos parâmetros, exceto o tamanho de memória, para o ambiente e para o agente-alvo, sendo eles:  $p_m = 0.3$  e  $\tau = 0.9$ . Novamente, utilizaremos  $N = 101$  e  $S = 2$ .

A Fig. 7.2 aponta os regimes que estamos interessados. Tomamos os mesmos valores de  $M$  do experimento anterior, de modo a facilitar a comparação, mas adicionamos  $M = 1$  ao conjunto, por este ser o ponto de mínima variância no sistema. Como pode ser visto, porém, estes valores representam regimes distintos daqueles no sistema tradicional. A Tabela 7.2 traz os resultados.

Tabela 7.2: Resultados com ambiente e agente-alvo criativos.

$M$	média	$ac_{best}$	$ac_{rand}$	$m_a = 2$	$m_a = 4$	$m_a = 6$	$m_a = 10$	$m_a = 12$
2	48.27	50.30	40.15	48.47	43.80	41.05	41.61	41.31
6	46.80	50.71	45.15	46.67	46.00	46.53	45.28	44.20
10	46.16	48.95	45.16	45.90	46.03	46.93	46.20	45.97

**M = 2**

Observando os valores de  $m_a$  para  $M = 2$ , nesta configuração, poderia-se argumentar que o comportamento observado é distinto daquele na configuração anterior, onde valores maiores de  $m_a$  eram acompanhados de ganhos progressivos na performance. Agora, aumentos em  $m_a$  fazem com que o agente-alvo perca performance. Porém, observamos que para  $M = 2$ , utilizando ambiente com agentes criativos, temos uma fase *eficiente*, ao contrário do modelo tradicional onde este ponto é caracterizado por alta ineficiência.

De fato, o comportamento apresentado é idêntico ao observado na configuração anterior quando na sua fase eficiente ( $M = 6$ ): valores maiores de memória tornam o agente menos eficaz. Este experimento comprova que o argumento é válido também para agentes utilizando aprendizado criativo.

**M = 6**

Em  $M = 6$  o sistema está transitando entre sua fase altamente eficiente e sua fase aleatória, mas as variâncias ainda encontram-se abaixo do caso aleatório e, portanto, esta fase ainda se caracteriza como eficiente. Observando o comportamento de  $m_a$  nota-se que é o mesmo comportamento característico da fase eficiente: para  $m_a > M$  o agente-alvo apresenta queda de performance, enquanto que para  $m_a \leq M$  a performance é próxima à média.

Apesar da queda de performance em  $m_a > M$ , esta é muito menos acentuada do que em  $M = 2$ . Como  $M = 6$  caracteriza uma transição, a eficiência apresentada é inferior ao caso anterior e, assim, a estabilidade do sistema é afetada. Analisando as médias na Tabela 7.2, constata-se que a média para  $M = 6$  é inferior a de  $M = 2$  e, portanto, esse regime não é evolucionariamente estável pois, em média, valores inferiores de memória provam-se mais eficazes.

**M = 10**

Para  $M = 10$  temos o sistema em sua fase aleatória, da mesma maneira que acontece com o caso tradicional. Novamente, a performance mostra-se independente do tamanho da memória, sendo comparável à média para toda a amplitude de valores de  $m_a$ . Com agentes decidindo de forma aleatória mais uma vez não há vantagem em ter-se qualquer valor de tamanho de memória.

**7.3.1.3 Ambiente tradicional, agente criativo**

Utilizando um ambiente composto por agentes tradicionais, dotamos agora nosso agente-alvo com o algoritmo de aprendizado criativo apresentado no capítulo anterior. Já sabemos que o algoritmo criativo pode tornar o sistema como um todo mais eficiente, porém pode-se questionar se, do ponto de vista do agente, esta é uma melhor abordagem quando consideramos seus acertos em relação a agentes tradicionais. Novamente, estudamos os casos onde  $M = 2$ ,  $M = 6$  e  $M = 12$ . A Tabela 7.3 informa os resultados.

Tabela 7.3: Resultados com ambiente tradicional e agente-alvo criativo.

$M$	média	$ac_{best}$	$ac_{rand}$	$m_a = 2$	$m_a = 4$	$m_a = 6$	$m_a = 10$	$m_a = 12$
2	41.62	50.02	49.52	49.18	70.32	66.86	68.33	70.49
6	47.28	53.94	43.53	47.75	47.27	47.51	46.03	43.30
10	46.08	50.56	45.15	45.31	45.95	45.95	46.76	46.61

**M = 2**

O que se observa agora, para  $M = 2$ , é que o agente-alvo, dotado de aprendizado criativo, obtém performances superiores à média para todos valores de  $m_a$ . Um aumento considerável na performance ocorre para  $m_a > M$ .

Enquanto o aumento de performance com o aumento de memória neste regime já foi observado para o agente-alvo utilizando o algoritmo tradicional de aprendizagem, os valores absolutos obtidos pela aprendizagem criativa são claramente superiores. Adicionalmente, para  $m_a = M = 2$  o agente-alvo já obtém um resultado acima da média, o que não acontecia no outro caso, demonstrando a eficiência do algoritmo proposto também no nível do agente neste regime de baixa eficiência.

Assim, novamente conclui-se que, no regime de baixa eficiência do MG tradicional, aumentos no tamanho da memória são vantajosos. Agentes com aprendizado criativo tiram particular vantagem do aumento da memória, obtendo performances bastante superiores à média dos demais agentes tradicionais.

**M = 6**

Com o sistema em seu regime de máxima eficiência, em  $M = 6$ , obtemos com o agente criativo o mesmo comportamento obtido com o agente tradicional, tanto qualitativa como quantitativamente. Pode-se constatar que a performance do agente-alvo é aproximadamente constante para  $m_a \leq M$ , mantendo-se próxima da média, e para  $m_a > M$  há uma queda acentuada.

Constata-se que quando o MG está em sua fase eficiente mesmo um agente utilizando aprendizado criativo não se beneficia de um aumento do tamanho da memória. Porém, o argumento utilizado com o agente-alvo tradicional para estabelecer  $M = 6$  como um ponto de equilíbrio não é válido aqui devido à diferença na forma de aprendizado. Como foi visto no capítulo anterior, um sistema composto unicamente de agentes criativos é capaz de obter variâncias muito baixas para pequenos tamanhos de memória e, assim, um agente mutante com menor memória mas dotado de aprendizado criativo poderia vir a dominar a população. Portanto, ao levarmos em conta o tipo de algoritmo de aprendizado sendo utilizado, não podemos considerar o ponto  $M = 6$  como evolutivamente estável.

**M = 10**

Para  $M = 10$  resultados idênticos aos obtidos para o agente-alvo utilizando aprendizagem convencional foram obtidos. A performance torna-se independente de  $m_a$ , sendo

constante e semelhante à média. A explicação também é a mesma do caso anterior: com agentes decidindo aleatoriamente não há informação para ser explorada e o tamanho da memória torna-se irrelevante.

#### 7.3.1.4 Ambiente criativo, agente tradicional

Tendo estudado como se comportam agentes tradicionais e criativos em um ambiente tradicional, é hora de darmos atenção ao caso onde o ambiente é criativo, isto é, composto por agentes dotados do algoritmo de aprendizado criativo proposto e o agente-alvo é tradicional. Com isto, podemos analisar o quão bem o algoritmo tradicional de aprendizado se sai em um ambiente composto de outra forma de aprendizado. A Tabela 7.4 traz os resultados.

Tabela 7.4: Resultados com ambiente criativo e agente-alvo tradicional.

$M$	média	$ac_{best}$	$ac_{rand}$	$m_a = 2$	$m_a = 4$	$m_a = 6$	$m_a = 10$	$m_a = 12$
2	48.27	50.30	40.15	47.05	43.35	39.06	39.14	38.16
6	46.80	50.71	45.15	47.96	47.97	47.67	45.22	44.15
10	46.16	48.95	45.16	46.01	46.23	46.21	46.54	46.00

Observa-se que os comportamentos obtidos anteriormente, para a configuração com ambiente e agente-alvo criativos, são mantidas, apenas com performances menores em termos absolutos, sendo tal comportamento, portanto, devido ao ambiente e não ao agente-alvo.

#### 7.3.2 Influência de $p_m$

Passamos agora a estudar a influência de  $p_m$  na performance dos agentes. Este parâmetro estabelece o quanto as estratégias selecionadas pelo algoritmo criativo proposto são modificadas. Assim, valores baixos de  $p_m$  indicam que as estratégias permanecem basicamente as mesmas, sofrendo pouca variação, enquanto valores altos de  $p_m$  indicam que as estratégias são amplamente modificadas.

Como foi discutido na Seção 6.1, o algoritmo de aprendizagem pressupõe valores baixos para  $p_m$ . Vimos também, na Seção 6.3, que quando tratamos a dinâmica como um todo o sistema deixa de apresentar ganhos para, aproximadamente,  $p_m > 0.1$ . Assim, estamos interessados em analisar a performance de agentes em valores abaixo deste valor crítico.

Para tanto, definimos o ambiente como composto por agentes com  $p_m = 0$  e introduzimos o agente-alvo com valores diversos de  $p_m$ , mantendo os demais parâmetros idênticos ao do ambiente. Utilizaremos a notação  $pm_a$  para diferenciar o valor de  $p_m$  do agente-alvo. A Fig. 7.3 mostra os resultados obtidos. Observa-se que a performance do agente-alvo cresce com o aumento de  $pm_a$ , até estabilizar-se para valores acima de  $pm_a \approx 0.05$ .

O mesmo comportamento foi observado ao utilizar ambientes com valores diferentes de  $p_m$  e para diferentes valores de  $M$ , apenas alterando-se os valores absolutos das

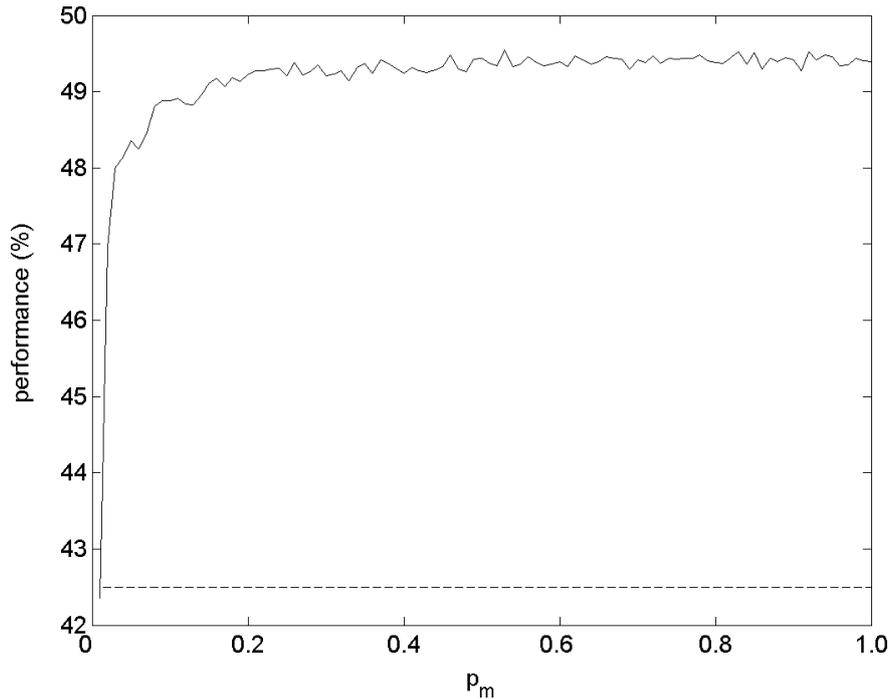


Figura 7.3: Performance do agente-alvo em função de  $p_{m_a}$ . Para o ambiente,  $p_m = 0$  e  $M = 2$ . Linha tracejada indica a média de performance dos agentes no ambiente.

performances.

Este resultado mostra que não há um limite superior no quanto se pode alterar uma estratégia ruim para ganhar performance. Isto é, ao contrário de  $M$  nos experimentos anteriores, não há um valor a partir do qual torna-se uma desvantagem ter maiores valores de  $p_{m_a}$ . Portanto, em geral, quanto mais um agente modifica suas estratégias ruins melhor será sua performance.

### 7.3.3 Influência de $\tau$

O último parâmetro que desejamos analisar é  $\tau$ . Este parâmetro define a probabilidade de aplicação do algoritmo de aprendizagem criativo e, como foi visto na Seção 6.1, estabelece em média o número de turnos em que cada estratégia é avaliada antes de se efetuar a escolha de qual será modificada.

Como em nossa análise de  $p_m$ , iniciamos com um ambiente onde o parâmetro é nulo,  $\tau = 0$ , e inserimos um agente-alvo utilizando diversos valores diferentes de  $\tau$ , mantendo os demais parâmetros idênticos aos do ambiente. Utilizaremos  $\tau_a$  para indicar o valor desse parâmetro para o agente-alvo.

A Fig. 7.4 traz os resultados. Observa-se que maiores valores de  $\tau$  trazem sempre melhor performance, para toda extensão de valores permitidos ( $0 \leq \tau \leq 1$ ). De fato, a melhor performance é obtida quando  $\tau = 1$  e o agente-alvo modifica a cada turno sua pior estratégia, com base na sua performance unicamente no turno anterior.

Novamente, o mesmo comportamento foi observado ao utilizar ambientes com valores diferentes de  $\tau$  e diferentes valores de  $M$ , apenas alterando-se os valores absolutos

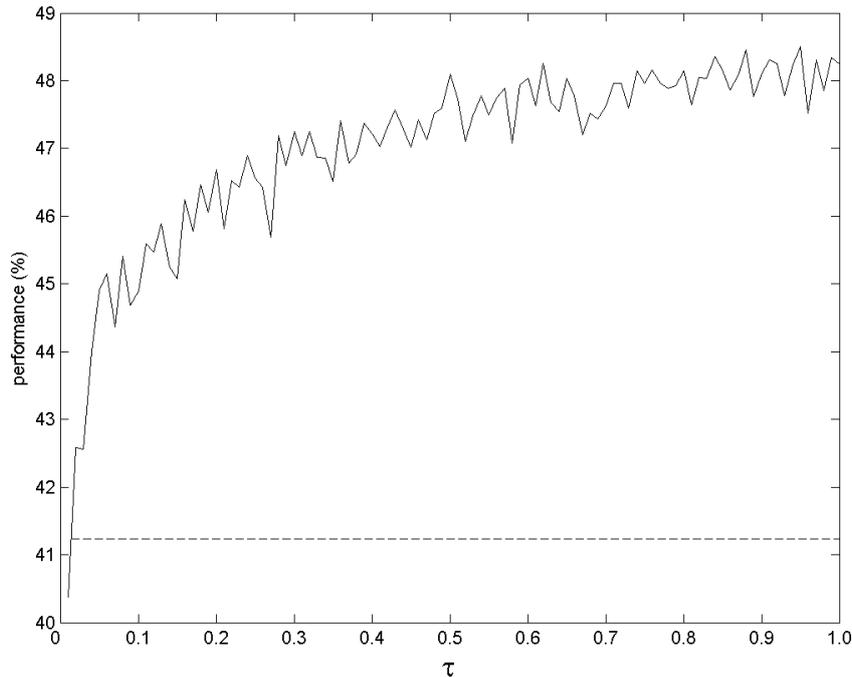


Figura 7.4: Performance do agente-alvo em função de  $\tau$  (linha contínua). Para o ambiente,  $\tau = 0$  e  $M = 2$ . Linha tracejada indica a média de performance dos agentes no ambiente.

das performances obtidas.

Com este resultado, conclui-se que não é vantajoso avaliar durante muitas rodadas as estratégias antes de decidir por modificar as piores, sendo uma melhor tática modificar uma estratégia tão logo ela mostre-se ruim.

## 7.4 Discussão dos resultados

Os experimentos realizados neste capítulo tiveram por objetivo compreender o efeito de diversos parâmetros dos algoritmos de aprendizado, estudados no MG (tradicional e criativo), na performance dos agentes.

Ao contrário da crença geral sobre os benefícios de uma maior memória no MG, verificamos que maiores memórias podem ser uma desvantagem se o sistema se encontrar em sua fase eficiente. Porém, em sua fase ineficiente, de fato agentes com maiores valores de memória se beneficiam, obtendo maiores performances. Verificamos também que durante a fase aleatória um agente tem sua performance independente do seu tamanho de memória.

Ao permitirmos que os dois sistemas de aprendizado atuassem em um mesmo jogo, observamos que agentes utilizando o algoritmo de aprendizado criativo proposto obtiveram em média maiores performances. Um estudo dos parâmetros deste algoritmo ( $\tau$  e  $p_m$ ) levou à conclusão de que agentes que selecionam estratégias para serem modificadas com base em um pequeno número de rodadas e que alteram estas estratégias de forma mais acentuada obtêm maiores performances.

Assim um agente ideal, dentro do contexto dos algoritmos aqui apresentados e analisados, utiliza aprendizado criativo, descarta estratégias tão logo se mostrem ineficazes e produz novas estratégias radicalmente diferentes das descartadas. Não é possível indicar o tamanho da memória para esse agente, uma vez que este valor depende do ambiente em que se encontra: se há grande volatilidade no sistema, o agente pode ter uma memória maior do que a de seus companheiros; no entanto, se o sistema é caracterizado por baixa volatilidade, então o agente se sairá melhor com tamanho de memória idêntico aos demais.

## 8 CONCLUSÃO E TRABALHOS FUTUROS

Neste trabalho, estudamos tarefas de aprendizado em um jogo multiagente conhecido como *Minority Game* (MG). O MG é amplamente estudado tanto na Economia e Física quanto em sistemas multiagentes, sendo alvo de estudo da crescente área de *econofísica*, que procura aplicar métodos da Física na Economia e que se aproveita de métodos computacionais.

As regras do MG são bem simples:  $N$  agentes devem optar em uma rodada por estar em um de dois grupos e, após as decisões, aqueles no grupo com menor número de agentes são recompensados; várias rodadas são executadas e os agentes têm acesso a um histórico que contém informações sobre que grupo conteve o menor número de agentes nas últimas  $M$  rodadas; para decidir, cada agente possui um algoritmo de aprendizado que procura explorar padrões contidos no histórico e este algoritmo trabalha manipulando estratégias que são mapeamentos de padrões no histórico para decisões. No MG tradicional, cada agente possui um conjunto fixo de estratégias e não pode descartar ou modificá-las durante todo o jogo. Isto é, um agente não pode criar novas estratégias para tentar se adaptar ao jogo. O limite na informação disponível para os agentes, bem como na sua capacidade de aprendizado, caracteriza o sistema como sob racionalidade limitada.

Estas regras, apesar de simples, dão origem a comportamentos complexos quando analisamos o número de agentes em cada grupo com o passar do tempo. De fato, o jogo é caracterizado neste trabalho como um sistema complexo, onde partes simples (agentes) dão origem a comportamentos e propriedades emergentes em um nível superior. A principal propriedade do jogo é a sua volatilidade, medida pela variância do número de agentes em um dos grupos. A volatilidade representa a eficiência do sistema como um todo, uma vez que é relacionado ao número total de agentes que é recompensado (e que deseja-se maximizar).

Foram apresentados diversos estudos realizados por outros pesquisadores, a maioria relacionando e analisando propriedades do jogo. Em um dos estudos apresentados, foi observado que a variância do sistema é unicamente função de duas variáveis:  $M$  (o tamanho da memória dos agentes) e  $N$  (o número de agentes do sistema). O comportamento em função dessas variáveis não é trivial, apresentando um máximo para  $M$  pequenos e estabilizando-se para valores grandes de  $M$ , atingindo um mínimo entre estes dois estágios. De forma interessante, o valor mínimo situa-se abaixo do que se esperaria se os agentes estivesse decidindo aleatoriamente, isto é, os agentes organizam-se para aumentar

a eficiência do sistema.

Partindo do modelo tradicional do MG, propusemos um novo algoritmo de aprendizagem para os agentes, chamado de “aprendizado criativo”. Neste algoritmo, baseado em algoritmos evolutivos, é permitido que os agentes modifiquem suas estratégias, criando um fluxo constante de novas estratégias no jogo. Com agentes dotados desse algoritmo, analisamos como se altera a dinâmica do jogo quando comparada com o tradicional, através de extensivas simulações.

Observamos que a dinâmica se altera de forma significativa ao utilizarmos o modelo de aprendizagem proposto. A mínima variância passa a situar-se no menor valor de  $M$  e apresenta um valor muito menor do que o menor valor obtido utilizando-se aprendizado tradicional. Para nenhum valor de  $M$  o sistema apresenta variâncias acima do esperado para o caso aleatório (i.e. o caso onde todos agentes decidem de forma aleatória).

De posse dessa nova dinâmica, passamos a tentar compreender o porquê deste comportamento. Para tanto, estudamos o que acontece com as estratégias dos agentes em jogo. Foi verificado que grande parte dos agentes, mesmo podendo criar novas estratégias, optam por utilizar uma única a partir de certa rodada. Diz-se que um agente que utiliza uma única estratégia está *congelado*. Observamos que a eficiência do sistema está relacionado com o número de agentes congelados, sendo que maiores eficiências são obtidas onde um maior número de agentes está congelado. Outras experiências realizadas mostram que não há convergência para estratégias específicas, mas que há um aumento na distância entre elas com o passar das rodadas, sendo a distância definida como a distância de Hamming. Portanto, observou-se que o algoritmo de aprendizado utilizado permite uma maior diferenciação entre as estratégias, responsável pelo aumento na eficiência do sistema.

Em um segundo momento, trocamos o estudo da dinâmica para a análise dos agentes envolvidos no jogo, tendo sido nossa intenção avaliar a influência dos parâmetros dos agentes em sua performance, isto é, na taxa de acertos em simulações. Existe uma crença generalizada nos estudos do MG de que agentes com maior tamanho de memória podem explorar agentes menos dotados, obtendo maior performance. Em nossos experimentos, verificamos que isso é apenas verdade quando o sistema encontra-se em sua fase ineficiente, caracterizado por altas variâncias. Quando o sistema está na sua fase eficiente, um agente com maior memória tem sua performance decrescida. Observamos que essa característica é válida para ambos modelos de aprendizado estudados e atribuímos a causa à maior organização dos agentes na fase eficiente, gerando um efeito de “filtragem” que impede que agentes operando com memórias maiores se beneficiem no jogo.

Também analisamos outros parâmetros do novo algoritmo proposto e a influência na performance dos agentes. Verificamos que agentes com maior taxa de aplicação do algoritmo (maior  $\tau$ ) possuem em média performances maiores e, da mesma forma, agentes que introduzem maiores inovações nas estratégias (maior  $p_m$ ), realizando modificações mais drásticas nas estratégias que provam-se ineficazes, obtém em média melhores resultados.

Assim, o novo modelo de aprendizado mostrou-se mais eficiente que o tradicional tanto no que diz respeito à dinâmica macroscópica do sistema como também no nível inferior dos agentes.

Os estudos aqui apresentados, na sua totalidade, tiveram por objetivo expandir a compreensão dos efeitos do aprendizado no *Minority Game*, um jogo central na recente área de econofísica. Com isso, esperamos ter também acrescentado à área de aprendizado em sistemas multiagentes em geral, e em especial em sistemas massivamente multiagentes como os de inteligência de enxames e inteligência coletiva.

De forma a ampliar os estudos aqui realizados, seria interessante investigar os seguintes tópicos:

- Formalização do conceito de estabilidade evolucionária verificada empiricamente no regime eficiente do jogo;
- Solução analítica da dinâmica do sistema com o novo algoritmo de aprendizado, ao menos para os casos limítrofes;
- Estudo do comportamento do sistema para uma faixa maior de valores de  $\alpha$ , e
- Comparação com outras formas de aprendizado, além dos aqui apresentados.

## REFERÊNCIAS

ANDRECUT, M.; ALI, M. Q Learning in the Minority Game. **Physical Review E**, [S.l.], v.64, n.067103, 2001.

ARAÚJO, R. M. **Computação Evolutiva aplicada ao Aprendizado de Máquinas**. 2004. Trabalho Individual (Mestrado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.

ARAÚJO, R. M.; LAMB, L. C. Neural-Evolutionary Learning in a Bounded Rationality Scenario. In: INTERNATIONAL CONFERENCE ON NEURAL INFORMATION PROCESSING, 11., 2004. **Proceedings...** Berlin: Springer-Verlag, 2004. p.996–1001. (Lecture Notes in Computer Science, v. 3316).

ARAÚJO, R. M.; LAMB, L. C. Towards Understanding the Role of Learning Models in the Dynamics of the Minority Game. In: IEEE INTERNATIONAL CONFERENCE ON TOOLS WITH ARTIFICIAL INTELLIGENCE 2004, 16., 2004. **Proceedings...** [S.l.: s.n.], 2004. v.16, p.727–731.

ARTHUR, W. B. Inductive Reasoning and Bounded Rationality. **American Economic Review**, [S.l.], v.84, p.406–411, 1994.

ARTHUR, W. B. Complexity and the Economy. **Science**, [S.l.], v.284, p.107–109, 1999.

ARTHUR, W. B.; DURLAUF, S.; LANE, D. **The Economy as and Evolving Complex System**. [S.l.]: Addison-Wesley, 1997.

AXELROD, R. **The Evolution of Cooperation**. [S.l.]: Basic Books, 1985.

BAZZAN, A. L. C.; BORDINI, R. H.; VICARI, R. M.; WAHLE, J. Evolving Populations of Agents with Personalities in the Minority Game. In: IBERO-AMERICAN CONFERENCE ON ARTIFICIAL INTELLIGENCE, 2000. **Proceedings...** Berlin: Springer-Verlag, 2000. p.166–175. (Advances in Artificial Intelligence, v.7).

BERTALANFFY, L. von. **Teoria Geral dos Sistemas**. 2.ed. Petropolis: Vozes, 1975.

BUCHANAN, M. **Nexus**. New York: W.W. Norton and Company, 2003.

CAPRA, F. **A Teia da Vida**. São Paulo: Cultrix, 1998.

CARA, A. R. de; PLA, O.; GUINEA, F. Competition, efficiency and collective behavior in the "El Farol" bar model. **Eur. Phys. J. B**, [S.l.], v.10, p.187–191, 1999.

CARBONELL, J. **Machine Learning: paradigms and methods**. Cambridge: MIT Press, 1990.

CHALLET, D. **Competition between adaptive agents: from learning to collective efficiency and back**. Berlin: Springer-Verlag, 2003.

CHALLET, D.; MARSILI, M. Relevance of memory in the minority game. **Phys. Rev. E**, New York, n.62, 2000.

CHALLET, D.; ZHANG, Y.-C. On the minority game: analytical and numerical studies. **Physica A**, [S.l.], v.256, p.514–532, 1998.

CHEN, Y.; GAZZALE, R. S. When Does Learning in Games Generate Convergence to Nash Equilibria? The Role of Supermodularity in an Experimental Setting. **American Economic Review**, Nashville, 2004.

CHOW, F.; CHAU, H. Multiple choice minority game. **Physica A**, Amsterdam, v.319, p.601–615, 2003.

COLMAN, A. M. Cooperation, psychological game theory, and limitations of rationality in social interaction. **The Behavioral and Brain Sciences**, [S.l.], n.26, p.139–153, 2003.

FARMER, J. D. Physicists Attempt to Scale the Ivory Towers of Finance. **Computing in Science and Engineering**, New York, Nov./Dec. 1999.

FOGEL, D. **Evolutionary Computation: toward a new philosophy of machine intelligence**. 2<sup>nd</sup>.ed. New York: IEEE Press, 2000.

FOGEL, D.; CHELLAPILLA, K.; ANGELINE, P. Inductive Reasoning and Bounded Rationality Reconsidered. **IEEE Transactions on Evolutionary Computation**, New York, v.3, n.2, p.142–146, July 1999.

GARROUSTE, P. **Learning in economics: the austrian insights**. Torino: ICER, 2001. (ICER Working Papers Series, n.25).

GILBERT, N.; CONTE, R. **Artificial Societies: the computer simulation of social life**. London: UCL Press, 1995.

GOLDBERG, D. E. **Genetic Algorithms in Search, Optimization and Machine Learning**. Boston: Addison-Wesley, 1989.

GORDON, D. **Formigas em Ação**. Rio de Janeiro: J. Zahar, 2002.

HAYKIN, S. **Redes Neurais: princípios e prática**. 2.ed. Porto Alegre: Bookman, 2001.

HOLLAND, J. H. **Emergence: from chaos to order**. [S.l.]: Perseus Books Group, 1999.

JOHNSON, S. **Emergence**: the connected lives of ants, brains, cities, and software. [S.l.]: Scribner, 2001.

KAHNEMAN, D. Maps of Bounded Rationality: psychology for behavioral economics. **The American Economic Review**, Nashville, v.93, n.5, p.1449–1475, 2003.

KENNEDY, J.; SPEARS, W. M. Matching Algorithms to Problems: an experimental test of the particle swarm and some genetic algorithms on the multimodal problem generator. In: IEEE WORLD CONGRESS ON COMPUTATIONAL INTELLIGENCE, 1998. **Proceedings...** [S.l.: s.n.], 1998. p.74–77.

KRAUS, S. **Strategic Negotiation in Multiagent Environments**. Cambridge: The MIT Press, 2000.

LI, Y.; VANDEEMEN, A.; SAVIT, R. The minority game with variable payoffs. **Physica A**, Amsterdam, v.284, p.501–503, 2000.

MANUCA, R.; LI, Y.; RIOLO, R.; SAVIT, R. The Structure of Adaptive Competition in Minority Games. **Physica A**, Amsterdam, v.282, p.559, 2000.

MICHALEWICZ, Z. **Genetic Algorithms + Data Structures = Evolution Programs**. 3rd.ed. Berlin: Springer, 1996.

MICHALSKI, R. S.; CARBONELL, J. G.; MITCHELL, T. M. **Machine Learning**: an artificial intelligence approach. Palo Alto: Morgan Kaufmann, 1983.

MITCHELL, T. **Machine Learning**. Boston: McGraw-Hill, 1997.

MOELBERT, S.; RIOS, P. D. L. The Local Minority Game. **Physica A**, Amsterdam, v.303, p.217–225, 2002.

MORIARTY, D. E.; SCHULTZ, A. C.; GREFENSTETTE, J. J. Evolutionary Algorithms for Reinforcement Learning. **Journal of Artificial Intelligence Research**, San Francisco, n.11, p.241–276, 1999.

MORO, E. The Minority Game: an introductory guide. In: KORUTCHEVA, E.; CUPERNO, R. (Ed.). **Advances in Condensed Matter and Statistical Physics**. [S.l.]: Nova Science Publishers, 2004.

NILSSON, N. J. **Introduction to Machine Learning**. [S.l.: s.n.], 1996.

RESNICK, M. **Turtles, Termites, and Traffic Jams**: explorations in massively parallel microworlds. Cambridge: The MIT Press, 1997.

RUBINSTEIN, A. **Modeling Bounded Rationality**. Cambridge, Massachusetts: The MIT Press, 1998. (Zeuthen Lecture Book Series).

RUMELHART, D.; HINTON, G.; WILLIAMS, R. Learning internal representations by error propagation. In: RUMELHART, D.; MCCLELLAND, J. (Ed.). **Parallel Distributed Processing: explorations in the microstructure of cognition**. Cambridge: MIT Press, 1986. v.1, p.318–362.

SAFAVIAN, S. R.; LANDGREBE, D. A Survey of Decision Tree Classifier Methodology. **IEEE Transactions on Systems, Man, and Cybernetics**, New York, v.21, n.3, p.660–674, May 1991.

SAMUELSON, L. Evolution and Game Theory. **Journal of Economic Perspectives**, [S.l.], v.16, n.2, p.47–66, 2003.

SHOHAM, Y.; POWERS, R.; GRENAGER, T. On the Agenda(s) of Research on Multi-Agent Learning. In: AAAI FALL SYMPOSIUM ON ARTIFICIAL MULTI-AGENT LEARNING, 2004. **Proceedings...** [S.l.: s.n.], 2004.

SIMON, H. A. From substantive to procedural rationality. In: LATSIS, S. (Ed.). **Method and appraisal in economics**. [S.l.]: Cambridge University Press, 1980.

SIMON, H. A Behavioral Model of Rational Choice. **The Quarterly Journal of Economics**, [S.l.], v.69, Feb. 1955.

SIMON, H. Rationality in Psychology and Economics. **Journal of Business**, [S.l.], v.59, n.4, p.S209–S224, 1986.

SLANINA, F. Social organization in the Minority Game model. **Physica A**, Amsterdam, n.286, p.367–376, 2000.

STANLEY, K.; MIIKKULAINEN, R. **Evolving Neural Networks through Augmenting Topologies**. [S.l.]: The University of Texas at Austin, 2001. (TR-AI-01-290).

STROGATZ, S. **Sync: the emerging science of spontaneous order**. New York: Theia, 2003.

STÜTZEL, T.; DORIGO, M. ACO Algorithms for the Traveling Salesman Problem. In: MIETTINEN, K.; MÄAKELÄ, M.; NEITTAANMÄÄKI, P.; PERIAUX, J. (Ed.). **Evolutionary Algorithms in Engineering and Computer Science**. [S.l.]: John Wiley and Sons, 1999.

SYSI-AHO, M.; CHAKRABORTI, A.; KASTI, K. Intelligent Minority Game with genetic-crossover strategies. **Eur. Phys. J. B**, [S.l.], v.34, n.00234, p.373–377, 2003.

TVERSKY, A.; KAHNEMAN, D. Judgment under uncertainty: heuristics and biases. **Science**, [S.l.], v.185, n.4157, p.1124–1131, 1974.

TVERSKY, A.; KAHNEMAN, D. The Framing of Decisions and the Psychology of Choice. **Science**, [S.l.], v.211, 1981.

VALIANT, L. G. Three Problems in Computer Science. **Journal of the ACM**, New York, v.50, n.1, p.96–99, 2003.

VIDAL, J. M. Learning in Multiagent Systems: an introduction from a game-theoretic perspective. In: ALONSO, E. (Ed.). **Adaptive Agents**. Berlin: Springer-Verlag, 2003. (Lecture Notes in Artificial Intelligence, v.2636).

WAKELING, J.; BAK, P. Intelligent systems in the context of surrounding environment. **Physical Review E**, New York, v.64, p.051920, 2001.

WATTS, D. **Small Worlds** : the dynamics of networks between order and randomness. Princeton University Press, 2003.

WATTS, D. **Six Degrees**: the science of a connected age. [S.l.]: W. W. Norton and Company, 2003.

WOLFRAM, S. **A New Kind of Science**. Champaign, IL: Wolfram Media, 2002.

WOLPERT, D. H.; TUMER, K. **An Introduction to Collective Intelligence**. [S.l.]: NASA, 2001. (NASA-ARC-IC-99-63).

WOOLDRIDGE, M. **Introduction to MultiAgent Systems**. New York: John Wiley and Sons, 2002.

YAO, X. Evolving Artificial Neural Networks. **Proceedings of the IEEE**, [S.l.], v.87, 1999.

ZHANG, Y.-C. Modeling Market Mechanism with Evolutionary Games. **Europhysics News**, Amsterdam, March/April 1998.

ZHANG, Y.-C.; CHALLET, D. Emergence of Cooperation and Organization in an Evolutionary Game. **Physica A**, Amsterdam, v.246, p.407, 1997.

ZHANG, Y.; KUHN, L. D.; FROMHERZ, M. P. Improvements on Ant Routing for Sensor Networks. In: ANTS, 2004. **Proceedings...** [S.l.: s.n.], 2004.