



Universidade do Minho
Escola de Engenharia

Carlos Manuel da Silva

**Um estudo sobre emergência de linguagens
para a comunicação de agentes em sistemas
de vida artificial**



Universidade do Minho
Escola de Engenharia

Carlos Manuel da Silva

**Um estudo sobre emergência de linguagens
para a comunicação de agentes em sistemas
de vida artificial**

Mestrado em Informática
Área de Especialização em Sistemas Distribuídos, Comunicações
por Computador e Arquitectura de Computadores

Trabalho efectuado sob a orientação do
Doutor Miguel Rocha

É AUTORIZADA A REPRODUÇÃO PARCIAL DESTA TESE, APENAS PARA EFEITOS DE INVESTIGAÇÃO, MEDIANTE DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE COMPROMETE

Universidade do Minho, ___/___/_____

Assinatura: _____

Ào meu pai ...

onde estiver.

"Na sobrevivência dos indivíduos e raças favorecidas, durante a luta constante e recorrente pela existência, vemos uma forma poderosa e incessante de selecção."

(Charles Darwin)

Agradecimentos

Ao meu professor orientador, Dr. Miguel Rocha, pela apoio incondicional que sempre me concedeu, pelos conselhos, ensinamentos, paciência e compreensão dos meus momentos mais difíceis.

Aos meus professores da componente lectiva do curso de mestrado pelo modo como contribuíram para a minha formação e actualização nas diversas áreas da Engenharia Informática.

Aos meus colegas de escola, que sempre me incentivaram no prosseguimento desta tarefa.

UM ESTUDO SOBRE EMERGÊNCIA DE LINGUAGENS PARA A COMUNICAÇÃO DE AGENTES EM SISTEMAS DE VIDA ARTIFICIAL

Resumo:

O objectivo deste trabalho é investigar os processos que uma comunidade de agentes artificiais poderá utilizar de forma a criar a sua interpretação da sua realidade e chegar a um consenso quanto à simbologia decorrente dessa interpretação. Esse consenso permitirá a emergência de linguagens que suportem uma forma básica de comunicação, sem interferência de nenhum agente exterior. Desta forma, estes agentes poderão adquirir ganhos de desempenho na resolução das suas tarefas no determinado contexto. A investigação é suportada por uma abordagem de aprendizagem por reforço, com a utilização de Redes Neurais Artificiais num contexto evolucionário.

Para o suporte experimental da investigação realizada foi utilizada uma plataforma de Vida Artificial, baseada no paradigma presa-predador, chamada *getAlife*, desenvolvida na Universidade do Minho, e que inclui módulos de manipulação de Redes Neurais.

Os resultados obtidos comprovaram as expectativas da tese, ou seja, demonstraram a evolução de uma linguagem rudimentar que resulta da emergência da representação de factos da realidade dos agentes por símbolos comuns. Além disso, verificou-se que as espécies dotadas desta capacidade de comunicação ganham com esse facto vantagens competitivas.

Palavras chave: Algoritmos Evolucionários, Comunicação Artificial, Emergência de Linguagens, Vida Artificial, Redes Neurais Artificiais, Sistemas Multi-agente.

A STUDY ON EMERGENCE OF LANGUAGES FOR AGENTS COMMUNICATING IN AN ARTIFICIAL LIFE SYSTEM

Abstract:

The aim of this study is to investigate the processes used by a community of artificial agents to create their own interpretation of the reality and reach a consensus about the symbols used to represent such reality. This consensus will allow the emergence of languages that support a basic form of communication, without interference from the outside world. In this way, agents may acquire performance gains in solving their problems within a given framework. The research is supported by an approach based on reinforcement machine learning, using Artificial Neural Networks in an evolutionary context.

To support the experimental research, an Artificial Life platform entitled *getAlife* was used, based on the predator-prey paradigm and developed at the University of Minho. This platform includes modules that enable the manipulation of Neural Networks.

The results obtained confirmed the expectations of the thesis, i.e. they have shown the evolution of a shared language brought by the emergence of symbols representing facts in the agent's reality. Furthermore, the species with this new communication capability outperform others without this trait, thus gaining competitive advantages.

Key words: Evolutionary Algorithms, Artificial Communication, Speech Emergence, Artificial Life, Artificial Neural Networks, Multi-Agent Systems.

Índice

| | Pág. |
|--|-----------|
| 1 Introdução | 1 |
| 1.1. Motivação | 1 |
| 1.2. Objectivos | 3 |
| 1.3. Organização da Tese | 5 |
| 2 Vida Artificial | 9 |
| 2.1. Conceito de “vida” | 9 |
| 2.2. Conceito de “vida artificial” | 11 |
| 2.3. Vida Artificial: enquadramento científico | 12 |
| 3 Sistemas Multi-agente | 17 |
| 3.1. Agentes | 17 |
| 3.2. Sistemas Multi-agente | 20 |
| 3.3. Reconhecimento | 22 |
| 3.4. Interação | 22 |
| 3.5. Cooperação | 23 |
| 3.6. Organização vs Auto-organização | 25 |
| 3.7. Sistemas cooperativos | 26 |
| 3.8. Sistemas reactivos | 28 |
| 4 Comunicação Artificial | 29 |
| 4.1. Comunicação | 29 |
| 4.2. Conversação | 30 |
| 4.3. Aquisição de símbolos por agentes | 31 |
| 4.4. Emergência de Linguagens | 32 |
| 5 Redes Neurais Artificiais e Algoritmos Evolucionários | 33 |
| 5.1. Redes Neurais Artificiais | 33 |

| | |
|--|-----------|
| 5.1.1. Definições | 33 |
| 5.2. Modelos de Aprendizagem | 40 |
| 5.2.1. Aprendizagem Supervisionada | 40 |
| 5.2.2. Aprendizagem por Reforço | 41 |
| 5.2.3. Aprendizagem não Supervisionada | 41 |
| 5.3. Algoritmos Evolucionários | 41 |
| 5.3.1. Definições | 42 |
| 5.3.2. Modelos de Programação | 47 |
| 5.3.3. Algoritmos Genéticos | 47 |
| 5.3.4. Programação Evolucionária | 49 |
| 5.3.5. Estratégias Evolutivas | 50 |
| 5.3.6. Programação Genética | 50 |
| 5.3.7. Algoritmos Evolucionários | 52 |
| 5.3.8. Algoritmos Evolucionários Híbridos | 54 |
| 5.4. Modelos Lamarkianos | 54 |
| 5.5. Modelos Baldwinianos | 55 |
| 6 Plataformas de Vida Artificial | 57 |
| 6.1. Definições | 57 |
| 6.2. Exemplos Ilustrativos | 60 |
| 6.3. getAlife | 63 |
| 7 Emergência de linguagens via Computação Evolucionária | 69 |
| 7.1. Justificação da estratégia | 69 |
| 7.2. Definição da plataforma | 71 |
| 7.3. Explicação das novas classes | 72 |
| 7.4. Definição das novas espécies | 74 |
| 7.5. Planeamento do modelo | 76 |
| 7.5.1. Justificação do método de aprendizagem | 76 |
| 7.5.2. Escolha da topologia da RNA | 77 |

| | |
|--|------------|
| 7.5.3. Escolha do Algoritmo Supervisionado | 77 |
| 7.5.4. Definição do método de treino das RNA | 78 |
| 7.5.5. Explicação das técnicas usadas | 79 |
| 7.6. Parametrização do modelo | 81 |
| 7.6.1. Valores da topologia da RNA | 81 |
| 7.6.2. Parâmetros estatísticos | 82 |
| 7.6.3. Parâmetros de cálculo | 82 |
| 7.6.4. Parâmetros medidos | 84 |
| 8 Análise de Resultados | 87 |
| 8.1. Resultados de emergência de linguagens. | 87 |
| 8.1.1. Configuração do sistema em análise | 88 |
| 8.1.2. Resultados obtidos | 89 |
| 8.2. Resultados de melhoria de desempenho da espécie | 95 |
| 9 Conclusões e trabalho futuro | 103 |
| Referências | 107 |

ABREVIATURAS

| | |
|------|---|
| AG | Algoritmo Genético |
| AGE | Algoritmo Genético e Evolucionário |
| AM | Algoritmo Memético |
| AR | Aprendizagem por Reforço |
| ANS | Aprendizagem Não Supervisionada |
| AS | Aprendizagem Supervisionada |
| BP | (algoritmo de) Backpropagação |
| BDI | (sistemas) Believe, Desire, Intention |
| CC | Ciência Cognitiva |
| EE | Estratégias Evolutivas |
| IA | Inteligência Artificial |
| IAD | Inteligência Artificial Distribuída |
| PE | Programação Evolucionária |
| PG | Programação Genética |
| PVA | Plataformas de Vida Artificial |
| QP | (algoritmo de) QuickPropagação |
| RNA | Rede Neuronal Artificial |
| RNAU | Rede Neuronal Artificial Unidireccional |
| RNAR | Rede Neuronal Artificial Recorrente |
| RNAC | Rede Neuronal Artificial Construtiva |
| SD | Sistemas Distribuídos |
| SMA | Sistemas Multi-Agente |
| SVA | Sistemas de Vida Artificial |
| RP | (algoritmo de) Retropropagação |
| VA | Vida Artificial |

| Índice de imagens | Pág. |
|---|-------------|
| Fig.3.1: Evolução de estados em SMA | 18 |
| Fig.3.2: Módulo de decisão | 19 |
| Fig.3.3: Tipologia baseada em Franklin | 21 |
| Fig.3.4: Sistema BDI genérico | 27 |
| Fig.5.1: Representação de um neurónio | 34 |
| Fig.5.2: Esquema geral de um neurónio artificial | 34 |
| Fig.5.3: Modelo Funcional de Neurónio Artificial | 35 |
| Fig.5.4: Função Sigmoid e sua derivada, com parametrização | 36 |
| Fig.5.5: Estrutura de RNA com duas camadas intermédias..... | 37 |
| Fig.5.6: Operação de cruzamento com ponto único | 44 |
| Fig.5.7: Operação de cruzamento uniforme | 44 |
| Fig.5.8: Algoritmo Genético de Holland | 48 |
| Fig.5.9: Exemplo de Koza para demonstrar a programação genética | 51 |
| Fig.5.10: Exemplo de Koza - ramos escolhidos para cruzamento | 51 |
| Fig.5.11: Exemplo de Koza – descendentes | 52 |
| Fig.5.12: Fluxograma típico de um AE | 52 |
| Fig.6.1: Modelação do comportamento de um vírus, por K.Schulten | 58 |
| Fig.6.2: Criatura Ero a aprender a andar e em fase adiantada | 60 |
| Fig.6.3: Criaturas EvoRunners, em concorrência | 61 |
| Fig.6.4: <i>Screenshot</i> da plataforma SCS | 62 |
| Fig.6.5: Ambiente da Plataforma Biots | 62 |
| Fig.6.6: Ecrã da plataforma Species | 64 |
| Fig.6.7. <i>Screenshot</i> do ecrã principal do <i>getAlife</i> | 65 |
| Fig.6.8: Evolução de uma espécie | 66 |
| Fig.6.9: Comparação de duas espécies | 66 |
| Fig.6.10: <i>Screenshot</i> de “Control Speech” (menu 2). | 67 |

| | |
|---|-----|
| Fig.7.1: Hierarquia dos elementos de segunda geração no <i>getAlife</i> | 71 |
| Fig.7.2: Rede Neuronal Artificial com uma camada intermédia | 77 |
| Fig.8.1: Gráfico comparativo para a primeira geração | 89 |
| Fig.8.2: Gráfico comparativo do desvio padrão para a primeira geração | 90 |
| Fig.8.3: Gráfico comparativo para entre as primeiras gerações | 92 |
| Fig.8.4: Gráfico comparativo para médio prazo | 93 |
| Fig.8.5: Gráfico comparativo para um longo prazo, 100 000 iterações. | 94 |
| Fig.8.6: Gráfico comparativo para uma geração e 10 presas | 96 |
| Fig.8.7: Gráfico comparativo para as primeiras gerações e 10 presas | 97 |
| Fig.8.8: Gráfico para 10 000 iterações, 10 presas | 97 |
| Fig.8.9: Gráfico para 100 000 iterações e 10 presas | 99 |
| Fig.8.10: 100 iterações, 30 presas | 99 |
| Fig.8.11: 1000 iterações, 30 presas | 99 |
| Fig.8.12: 10 000 iterações, 30 presas | 99 |
| Fig.8.13: 100 000 iterações, 30 presas | 100 |

Índice de tabelas

| | |
|--|----|
| Tabela 8.1: Parâmetros de teste | 88 |
| Tabela 8.2: Comparação de totais de presas | 97 |

Capítulo 1

Introdução

1.1. Motivação

A evolução económica dos países, a concorrência originada pelo Mundo Global, a produção de ferramentas de apoio à investigação, cada vez mais sofisticadas, o investimento na investigação como necessidade de obter produção de mais valias em sociedades modernas, a facilidade de comunicação, consulta e troca de ideias suportada pela Internet, levou a uma proliferação de conhecimento científico sem paralelo.

A comunidade científica, apoiada num período de paz e estabilidade política, como consequência da integração de nações em blocos poderosos, estáveis e cooperantes, desenvolveu um comportamento de cooperação, partilha e crítica construtiva, que se revela pelas trocas de conhecimento originadas nos inúmeros colóquios, palestras, grupos de trabalho transnacionais e sociedades científicas que proliferam por todo o mundo.

Esta multiplicação de conhecimento concretiza-se na produção em grande escala de tecnologias de ponta, assim como na criação e aperfeiçoamento, em espiral, de técnicas e teorias de suporte às mesmas. Estas tecnologias, devido a pressões comerciais e políticas, têm um tempo de vida útil muito curto, sendo constantemente substituídas por novos modelos mais capazes, atractivos, inteligentes e até mais económicos.

As consequências são uma produção tecnológica com uma grande capacidade de parametrização e de cada vez mais difícil controlo de configuração e programação, já que, associado à evolução do hardware, também se verifica uma alteração constante dos algoritmos e programas necessários.

Tempos de vida muito curtos tornam os sistemas obsoletos muito antes de a sua capacidade limite ser atingida. Além disso, torna-se humanamente impossível que se formem técnicos que acompanhem constantemente essa evolução tecnológica com o fim de manter a produtividade desses sistemas.

É neste ponto que se perspectiva uma solução, com visão de futuro, de tornar a tecnologia mais autónoma relativamente à intervenção e monitorização humana. Tornar uma tecnologia mais autónoma significa torná-la mais inteligente e com capacidade de decisão própria para resolver problemas com que é confrontada. Significa também dotar essa tecnologia de uma capacidade de aprendizagem que lhe permita grandes capacidades interpretativas e adaptativas à evolução constante do seu meio exterior. E, finalmente, grande capacidade de aprendizagem implica, certamente, o desenvolvimento de boas técnicas de comunicação com o mundo envolvente.

Dentro destas tecnologias, têm especial interesse aquelas que se relacionam em grupo. De notar que, tal como a sociedade responde ao aumento de complexidade com a partilha de informação e trabalho em grupo, as tecnologias respondem com o desenvolvimento de Sistemas Distribuídos, onde predominam também essa partilha e distribuição. Predomina também uma boa capacidade de auto-organização, que exige sem dúvida, uma bem organizada capacidade de comunicação e decisão dos sistemas.

Nestas tecnologias, a autonomia relativamente à interferência humana, deverá dotá-la de uma enorme versatilidade; exigirá, no entanto, a capacidade de essas entidades tecnológicas terem a possibilidade de decidir e criar as suas próprias formas de comunicação, com regras por si criadas e com uma plasticidade que possibilite que essa comunicação seja reinventada constantemente entre si e os seus pares.

É neste contexto que, nos tempos que decorrem, tem emergido um interesse reforçado pelas teorias associadas às áreas da Inteligência Artificial e Ciências Cognitivas. Têm-se verificado a recuperação de modelos desenvolvidos teoricamente nas últimas décadas, a criação de versões alteradas desses modelos e o surgimento de novas propostas, heurísticas e teorias que vêm confrontar ou complementar essas áreas.

Uma comunicação autónoma entre sistemas tecnológicos só pode ser sustentada com uma linguagem autónoma, e por isso a importância do estudo dos processos de Emergência de Linguagens, que conduziu à escolha deste tema de investigação para dissertação de mestrado.

Se na natureza emergem sistemas de comunicação para resolver problemas de optimização nos seus subsistemas, porque não recorrer à mesma estratégia para resolver problemas de optimização em sistemas artificiais? Como dotar os sistemas artificiais dessa capacidade? Como é que se processa a emergência completamente autónoma? Tornam-se os sistemas artificiais, de facto, mais capazes, se forem dotados dessa possibilidade? É tentativa de resposta a estas e outras perguntas semelhantes que

estiveram na base da motivação deste trabalho, e que procuraremos investigar ao longo dos capítulos seguintes.

1.2. Objectivos

A emergência da comunicação em sistemas Multi-Agente, parece ser um nicho de investigação que desperta actualmente um interesse inesperado, embora consequente com as ideias referidas na secção anterior. Mesmo não se vislumbrando, ainda, uma grande proliferação de desenvolvimento teórico nesta área, nota-se uma certa urgência na sua exploração. É comumente aceite que a comunicação entre agentes é de vital importância para aumentar a capacidade e desempenho na produção, análise ou pesquisa, dos sistemas artificiais.

Embora se procure autonomizar a comunicação entre agentes, muitos investigadores e instituições continuam a utilizar plataformas, por si desenvolvidas, para essa comunicação (Wooldridge, 2002). Chega-se, inclusive, a propor gramáticas de comunicação, para agentes, com linguagens pré-estabelecidas, embora abertas, que seriam o suporte técnico, formal ou padronizado, à emergência de linguagens posteriormente assumidas pelos agentes. Estão neste caso, como exemplo, as linguagens do grupo ACL (Agent Communication Languages) de que faz parte o KQML (Knowledge Query and Manipulation Language), o KIF (Knowledge Interchange Format) ou, mais recentemente, o FIPA (Foundation for Intelligent Physical Agents).

O propósito deste trabalho é analisar os processos de uma emergência de linguagem entre seres artificiais em que não foram definidos quaisquer suportes de estado inicial. Os agentes, seres artificiais, deverão desenvolver entre si uma linguagem, sem qualquer ajuda do exterior ao sistema artificial em que estão inseridos. O nosso objectivo será dotar os seres artificiais de uma memória e de uma possibilidade de aprendizagem através da incorporação, na sua estrutura, de uma Rede Neuronal Artificial (RNA). Esses seres serão dotados também da capacidade de exprimir para o exterior o resultado da sua interpretação pessoal dos estados que lhe são transmitidos pelos seus sensores, naquilo que chamaremos de “tendência”, e que será explicado no capítulo sobre Sistemas Multi-Agente.

Note-se que a exteriorização da percepção de novos estados pelos agentes provocarão uma alteração no mundo exterior (no caso real os sons). Esta alteração, por sua vez, será detectável pelos sensores dos outros agentes como um novo estado do ambiente, que implicará uma nova interpretação e que estará na base da comunicação entre os agentes. As RNA's serão o suporte para a capacidade de armazenamento de informação e aprendizagem; no nosso caso, através de um processo muito especial de Aprendizagem Supervisionada.

O processo de emergência decorrerá da aplicação de Algoritmos Evolucionários (AE's) sobre a população dos agentes, onde se espera que a comunicação possa aumentar a probabilidade de maior tempo de vivência, logo maior capacidade reprodutiva e passagem de genes, dos elementos mais capazes. Se esta capacidade de comunicação tornar os elementos mais capazes, então estes desenvolverão automaticamente uma competência cultural, que, pelo mesmo processo evolucionário, conduzirá a uma convergência cultural, logo a uma emergência de uma linguagem comunicativa, própria, dinâmica, adaptável e útil. Aquilo a que poderia ser chamada de "cultura artificial", já que seria totalmente criada, evoluída e compreendida apenas pelos elementos artificiais de um sistema fechado.

O objectivo do nosso trabalho, para além da análise dos processos de emergência, dentro das condições descritas, passa também pelo estudo das consequências que essa nova aptidão provocará nas relações dos agentes com o meio exterior. Para isso serão postas em concorrência duas espécies, uma herdeira da outra, com a diferença da incorporação, na primeira espécie, de uma potencial de capacidade comunicativa implementada por uma RNA e pela Tendência Expressiva.

Finalmente, pretende-se verificar a adequação e eficácia das RNA's e AE's a processos de emergência de linguagens.

Esta verificação será feita utilizando uma ferramenta já existente, o *getAlife*, que será descrita e explicada num capítulo posterior. Nesta ferramenta, que evolui agentes artificiais num processo competitivo do tipo presa-predador, serão feitas algumas alterações de forma a criar e controlar uma segunda geração de agentes, com as propriedades acima indicadas (uma RNA de aprendizagem de linguagem e uma Capacidade Expressiva). Serão feitas também algumas alterações de forma a permitir a visualização gráfica e analítica da evolução de parâmetros de comunicação e competência dos agentes.

Um dos objectivos do trabalho passa também por dotar essa ferramenta, o *getAlife*, de mais uma possibilidade: a comunicação entre agentes. Esta poderá, assim, permitir a base de desenvolvimento para um estudo da emergência de comunicação entre agentes e a comparação dos desempenhos que essa comunicação implica.

1.3. Organização da tese

Esta tese está organizada em duas partes. Na primeira parte, são apresentados os resultados de uma investigação ao estudo da arte, sendo feito um levantamento de todos os elementos, conceitos e teorias ligados ao objectivo do trabalho. Embora o tema seja bastante específico, o estudo das emergências de linguagens utilizando AE's, de um modo geral, os conceitos associados abordam vastas áreas da ciência tais como a Inteligência Artificial, Ciências Cognitivas, Biologia, Filosofia, Linguística ou Antropologia. Foi decidido, assim, compilar um conjunto de conhecimentos teóricos básicos para o enquadramento e compreensão dos processos de análise que decorrerão ao longo do trabalho de investigação e análise prática.

A investigação teórica feita, deixou bem claro que a comunidade científica não é unânime nas definições, interpretações ou métodos, em praticamente todas as áreas investigadas. Bem pelo contrário, existem bastantes discordâncias ou até antagonismos na abordagem dessas teorias. Isto torna bastante difícil a definição exacta de conceitos para a sustentação teórica correcta do trabalho. Assim, foi necessário fazer uma referência a várias perspectivas mais marcantes e de seguida dar preferência às que melhor se adaptavam às necessidades práticas do nosso projecto. Obviamente, não se pretende concordar ou não com uma ou outra teoria, mas sim apoiar-nos naquela que pareceu ser a ferramenta mais indicada, neste instante, para sustentar a nossa metodologia de investigação.

No capítulo 2, começamos por desenvolver os conceitos de “Vida” e Vida Artificial”, e qual é o posicionamento relativo da disciplina científica do estudo da Vida Artificial (VA) no edifício científico clássico.

O capítulo 3 surge como uma extensão do capítulo 2. De facto, a VA sugere de imediato, também, o estudo de seres artificiais, que normalmente evoluem em grupo. Isto implica que é necessária a definição de conceitos associados a estes seres e que nos

meios académicos e científicos surge com o nome de Sistemas Multi-Agente (SMA). Neste capítulo 3, serão decididas as definições adoptadas para “agente”, “sistemas multi-agente”, “reconhecimento”, “interacção”, “cooperação”, “organização e auto-organização”, “sistemas multi-agente cooperativos” e “sistemas multi-agente reactivos”. Serão ainda assumidos os modelos de Sistemas Multi-Agente adoptados, de entre uma vasta gama disponível.

O capítulo 4 alarga o estudo dos sistemas multi-agente às técnicas de comunicação. Assim, serão consideradas algumas definições que sustentarão as nossas descrições posteriores: comunicação, conversação, aquisição simbólica e emergência de linguagem. Este capítulo foi bastante difícil de compilar devido ao pouco acordo dos investigadores sobre conceitos como signo, símbolo, aquisição e armazenamento simbólico, fala, emergência, etc., que aparece bem representada na controvérsia originada pelo *Symbol Grounding Problem*. As divergências são muitas e mantêm-se com vigor redobrado. Neste capítulo, abordaremos algumas dessas ideias e definiremos aquelas que iremos adoptar.

O capítulo 5 descreverá algumas das definições e técnicas de Computação Genética e Evolucionária, que estão disponíveis através da plataforma *getAlife*. Estas técnicas estiveram na base de variados testes durante o trabalho de investigação. A teoria exposta neste capítulo, baseia-se no trabalho de Miguel Rocha, na sua tese de doutoramento de 2004, “*Optimização dos Modelos Conexionistas de Aprendizagem via Computação Genética Evolucionária*” (Rocha, 2004). Abordará os conceitos associados às Redes Neurais Artificiais, com os correspondentes algoritmos de treino. Descreverá ainda, resumidamente, os modelos de aprendizagem, Supervisionada, por Reforço e Não Supervisionada, assim como os AE’s limitando-nos aos Modelos de Programação mais comuns. Conclui, finalmente, com o Modelo Lamarkiano e o Modelo Baldwiniano, que poderão ser uma boa ajuda para a compreensão da plasticidade cultural transmitida pelos agentes entre gerações.

O capítulo 6, fará uma retrospectiva de alguns conceitos de Plataformas de Vida Artificial, PVA. Mostrará uma organização dos vários tipos destas plataformas, e descreverá algumas plataformas mais características, como exemplo desses tipos. Esta abordagem pretenderá enquadrar a plataforma que irá ser usada, o *getAlife*. Este será desenvolvido com maior pormenor na secção 6.3.

Depois destas exposições teóricas foram feitas as exposições mais consequentes com o trabalho prático.

Assim, o capítulo 7 aborda e justifica as opções e valores assumidos na investigação em causa, para além de explicar o desenvolvimento dos procedimentos de análise. Procura-se justificar as decisões de opção pelos métodos utilizados, as topologias e métodos de treinos das RNA's, e a explicação das novas classes criadas na plataforma *getAlife*, associadas às novas potencialidades de comunicação dos agentes.

No capítulo 8 foi feita uma análise crítica aos resultados do trabalho em duas perspectivas: a emergência das linguagens e as consequências que essa emergência provoca sobre o sistema multi-agente, no tocante à variação do seu comportamento e eficácia.

No capítulo 9 é sintetizada a conclusão final deste estudo. São também projectadas para trabalho futuro, algumas ideias que foram surgindo durante a investigação, que poderão complementar ou incrementar esta ideia base da emergência das linguagens com o recurso às RNA's e AE's.

Capítulo 2

Vida Artificial

A Vida Artificial é uma das áreas de investigação que nos últimos tempos tem revelado uma enorme vitalidade, já que parece ser uma solução muito prática e fiável para a criação de modelos que se apresentam cada vez mais com um grau de complexidade muito elevado. Embora alguns cientistas mostrem reservas ao recurso às características humanas para implementar em sistemas artificiais, com o fim de resolver sistemas muito complexos (Winograd & Flores, 1986), outros defendem esta técnica como a mais adequada (Stefik & Bobrow, 1987). Ao longo deste estudo iremos abordar esses pontos de vista.

2.1 Conceito de “vida”

A concepção que iremos assumir para sustentar a terminologia de “Vida Artificial”, neste trabalho, obriga a que, primeiro, definamos o que iremos interpretar como “vida” e como “artificial”. A definição de “vida” tem muitas facetas que vão de uma consideração mais filosófica incorporando o próprio conceito de alma (um impulso vital não observável segundo a corrente vitalista (Bergson, 1911) (Driesch, 1908)), até uma concepção mais pragmática e muito simplista que apenas assume a dinâmica de uma qualquer entidade, segundo a visão mecanicista (Cannon, 1939) (Pribean, 1968). Pelo meio existe um conjunto de teorias que se diferenciam pelo modo como cada uma delas interpreta o conceito de entidades orgânicas ou não orgânicas (Miller, 1978) e (Bunge, 1979). Uma consequência da interpretação mais simplista, é a ideia de que tudo é vida uma vez que tudo o que existe no Universo evolui e se movimenta, deixando de existir qualquer separação entre a física e a biologia. No outro extremo, a consideração de “alma” (impulso vital inobservável) impõe sérias limitações a uma visão científica, que exige, necessariamente, mensurabilidade e precisão. No meio destas definições, discute-se a vida como um arranjo complexo de elementos físicos com certas características evolutivas; mas mesmo aqui, verificam-se desencontros teóricos que se centram agora

no grau de complexidade desses arranjos. Na actualidade, este debate ainda se mantém bem aceso e estes limites de definição resistem bem sólidos, pelo que a assumpção de uma ideia de vida apenas se poderá considerar dentro de uma perspectiva específica. Iremos adoptar para o desenvolvimento do nosso trabalho uma definição baseada nos trabalhos de J. Farmer e A. Belin (1992), já que mostra servir melhor as características do modelo de estudo a que nos propomos. Consideraremos que é suficiente e necessário, para uma entidade pertencer à categoria dos “elementos vivos”, que verifique a seguinte propriedade: capacidade para identificar e utilizar as propriedades na sua vizinhança de forma a utilizá-las na sua própria existência. Esta característica implicará um conjunto de propriedades mais ou menos vincadas como:

- Metabolismo;
- Adaptabilidade;
- Autonomia;
- Auto-reparação;
- Crescimento;
- Replicabilidade;
- Evolução;
- Reprodução.

É de notar, no entanto, que os próprios autores usam esta lista apenas como referência, que está longe de ser absoluta. Com efeito, podem-se considerar seres vivos, entidades que apenas adoptam algumas destas propriedades. Por exemplo, as mulas não têm a capacidade de reprodução e não deixam de ser elementos vivos.

Esta definição de sistema vivo é muito próxima da introduzida por H. Maturana e F. Varela (1973), quando definiram o seu modelo de *autopoiesis*. Segundo estes autores, um sistema é vivo se verificar as propriedades *autopoiéticas*, isto é, se o sistema fluir em circuito fechado, produzindo componentes à saída que serão a base para as próprias matérias-primas para a entrada: o objectivo do processo é o seu próprio desenvolvimento e replicação e as decisões de processamento serão definidas também pelo próprio sistema. Esta teoria é muito importante para o desenvolvimento de sistemas artificiais de vida (para além de ser a base de investigações noutras áreas como a Biologia ou Economia) e utilizaremos bastantes dos seus pressupostos no decorrer do nosso trabalho. Não nos esqueçamos que um dos objectivos do nosso trabalho é investigar a emergência de linguagens em sistemas de vida artificial, num sistema

autopoiético, já que pretendemos garantir que não será aplicada qualquer interferência do exterior para esse processo.

2.2 Conceito de “vida artificial”

O conceito de “vida artificial” surgiu, introduzido pela primeira vez, através de um trabalho desenvolvido por Christopher C. Langton (1995). Para Charles Taylor e David Jefferson (1995) a vida artificial seria qualquer sistema produzido pelo homem que replicasse as propriedades da vida natural. Embora a tradução literal do conceito assuma que será “vida artificial” todo o tipo de vida criado pelo Homem em vez de ser criado pela Natureza, Langton concorda que é difícil de especificar uma definição exacta para esse conceito, embora possa definir com clareza o objectivo da sua aplicação prática. Assim, este autor questiona-se sobre a rigidez das actuais leis, que orientam certas áreas científicas de estudo da natureza, como a Física, Química ou Biologia. Lembra que essas leis não incluem a imprevisibilidade ou mutações, pelo que, mais do que a aplicação directa de leis para a definição dos fenómenos da Natureza, faria todo o sentido a aplicação de modelos que representassem o seu comportamento nessas áreas. Para isso, seria necessário que o Homem simulasse ambientes controláveis e mensuráveis para essa análise, onde já seriam incluídos os fenómenos imprevisíveis, irregularidades, mutações, acidentes e contingências históricas, de que a natureza é pródiga. Aliás, será pelo teste a estas mutações que a natureza faz a sua evolução positiva, pelo que a sua inclusão nos processos de estudo é de fulcral importância. A vida artificial surge, assim, como uma resposta para a elaboração desses modelos, que abarcam para lá das regularidades.

A Vida Artificial permite a captação das principais características de complexidade dos sistemas de vida, assim como a sua repetibilidade, mensurabilidade e controlo o que torna esta técnica adequada para o estudo desse tipo de sistemas que podem abranger áreas tão díspares como a origem da vida, a evolução cultural, a evolução das espécies, e o comportamento da mente, entre bastantes outros (Taylor & Jefferson, 1995).

A criação pelos cientistas de ambientes simulados, que reflectem comportamentos naturais, permitirá uma abordagem mais abrangente e realista. No entanto, enquanto as leis assumidas como representativas do comportamento da Natureza, pretendem ser

gerais e intemporais, embora não abordem as suas mutações, a vida artificial é mais sintética e localizada temporalmente. Mas, como diz Langton, “para ter uma teoria actualizada, é necessário compreender aquilo que é possível”. Nenhuma lei estará completa até que todos os fenómenos relacionados sejam incorporados, o que em termos de uma Natureza constantemente em evolução, parece ser tarefa impossível, pelo que uma abordagem sintética, mais localizada em termos de espaço e tempo, apoiada num modelo de vida artificial, parece ser uma boa solução. Realça deste ponto de vista de análise que, tão importante como compreender o comportamento da Natureza, será crucial a compreensão, também, da evolução desse comportamento.

2.3 A Vida Artificial: enquadramento científico

A consideração de sistemas utilizando vida artificial, muitos deles com a evolução de agentes, introduziu alguma perturbação, entre os cientistas das várias áreas de investigação, relacionadas com o fenómeno da resolução de problemas complexos com o recurso a modelos apoiados em Sistemas de Vida Artificial (SVA).

Desta tensão surgiu a dúvida e debate sobre o enquadramento das novas áreas de Vida Artificial e Sistemas Multi-Agente, relativamente às disciplinas clássicas como: Sistemas da Informação, Ciência Cognitiva, Inteligência Artificial ou Sistemas Distribuídos.

Os Sistemas de Informação emergiram devido a um trabalho de T. Winograd e F. Flores (1987) que contestaram o objectivismo, o *design* de sistemas de computadores baseados no racionalismo e a metodologia da Inteligência Artificial. Eles defenderam que os sistemas devem ser desenvolvidos baseados num processo de linguagem/acção. De entre as várias diferenças referenciadas, destacar-se-á, apenas, a forma como passa a ser encarado o processo do conhecimento e da linguagem; o conhecimento deixa de ser uma representação do mundo no bloco cognitivo dos agentes para passar a ser uma representação que o agente vai construindo do mundo, baseado no estímulo do exterior (sensores e aprendizagem) (Mingers, 1995). Este conhecimento não se implementa rígida e definitivamente, podendo variar, uma vez que o mundo também vai sofrer mudanças contínuas.

A linguagem, para Winograd e Flores, também tem um processo idêntico; a linguagem não pode consistir numa representação simbólica pré-estabelecida dos elementos do exterior (objectos, agentes, relações, acções,..) mas sim de uma negociação das representações simbólicas de cada agente e a aceitação do símbolo depois de um acordo tácito ou imposto, para passar a ser partilhado. Neste pressuposto, o mesmo objecto poderia ser partilhado com símbolos representativos diferentes para grupos diferentes e inclusive o mesmo agente poderia utilizar um símbolo, quando em comunicação com um grupo de agentes, e outro símbolo quando em comunicação com outro grupo de agentes, uma vez que esses grupos concordassem em representações simbólicas diferentes para o objecto em causa (um grupo chega a um consenso de um símbolo para um objecto, e, outro grupo chega a um consenso de outro símbolo para o mesmo objecto, se estes grupos estiverem mais ou menos separados). Este conceito leva a uma dinâmica evolucionária para a linguagem. Estas ideias entram em confronto com as estratégias objectivistas normalmente assumidas pela Inteligência Artificial. Estes pontos de vista, de Winograd e Flores, foram vigorosamente criticados por defensores de outras áreas como M. Stefik e D. Bobrow (1987), A.Vellino (1987), L. Suchman (1987) e W. Clancey (1987).

A Ciência Cognitiva (CC) começou por ser uma disciplina transversal que englobava todas as áreas que abordassem o estudo dos processos cognitivos e que incluía desde a Filosofia e a Psicologia até à Inteligência Artificial (IA), passando pela Neurociência. Devido às críticas, cada vez em tom mais elevado, que alguns investigadores começaram a tecer aos métodos usados pela IA, que consideram utilizar métodos de decisão inteligentes mas não reflectindo os processos de pensamento reais do cérebro, a CC, começou por se afastar desta área e criou, segundo alguns autores, o seu próprio espaço no âmbito das Ciências da Computação.

Recentemente, para alguns, a Ciência Cognitiva é a área que estuda a fisiologia do cérebro para compreender o seu funcionamento real e exportar para os sistemas artificiais as bases desse funcionamento (Eysenk & Keane, 1994). Esta posição, criou os seus próprios críticos, que referem a extrema rigidez desta metodologia, e sustentam que um sistema artificial inteligente, para processar decisões que se consideram inteligentes, não necessita de se inspirar no funcionamento do cérebro, mas, pelo contrário, deveria libertar-se desses constrições, e desenvolver os seus próprios processos artificiais de inteligência que se tornariam mais maleáveis e específicos, logo mais eficazes.

A Inteligência Artificial (IA) é uma área científica, herdeira da Cibernética, que, na sua essência, procura desenvolver algoritmos racionalistas com o objectivo de realizar processamentos intencionalmente inteligentes. Os algoritmos, embora utilizem estratégias inteligentes, processam de uma forma sequencial seguindo o processo de Turing (Hodges, 1983). Ao contrário da actual CC, nesta área assume-se que o cérebro funciona numa lógica semelhante à de um computador (e não que o computador deve funcionar como um cérebro). Os algoritmos inteligentes são desenvolvidos com base numa analogia; como o cérebro solucionaria um problema se o processasse como um computador, em modo sequencial.

Certos estudiosos acham que não é suficiente, apenas, usar estratégias inteligentes, mas é necessário também replicar a forma como o cérebro humano constrói essas estratégias. Sabe-se que o cérebro não funciona de uma forma sequencial, mas sim de uma forma paralela e distribuída (conexionismo). É assim que surge outra área científica que evolui dentro deste conceito e que é chamada de Sistemas Distribuídos (SD) (Dreyfus, 1988).

É obvio que todas estas áreas partilham vastos sectores de investigação, e nalguns segmentos se confundem, sendo, por vezes, a fronteira de diferenciação bastante abstracta. E isso torna difícil a tarefa de inserir a Vida Artificial dentro desta topologia. Vejamos algumas características relativas para compreendermos melhor o nosso propósito: enquanto a Vida Artificial potencia uma análise do tipo bottom-up, a Inteligência Artificial opta pela análise top-down, tal com a Ciência Cognitiva. Enquanto a Vida Artificial privilegia a evolução da sua própria estrutura, a IA e a CC, privilegiam a procura de uma solução para um determinado problema.

Quase todos os investigadores, neste campo, reivindicam a VA para as suas disciplinas de investigação. Existem ainda definições mais redutoras; a VA, na sua componente de representação do comportamento de elementos em grupo (sistema multi-agente) que podem cooperar ou concorrer, é estudada como o suporte de uma área da Inteligência Artificial que investiga esse tipo de sistemas, e que se chama, Inteligência Artificial Distribuída (IAD). Segundo G. Weiss (2000), “IAD é o estudo, construção e aplicação de sistemas multi-agente, ou seja, sistemas nos quais diversos agentes inteligentes, interactivos, perseguem algum conjunto de objectivos ou desempenham algum conjunto de tarefas”.

Existem, finalmente, os que definem a VA como uma área transversal com pontes de ligação entre as disciplinas clássicas, e os que defendem que se trata de uma área nova e

independente das áreas atrás mencionadas, mas que se relaciona com elas de uma forma autónoma.

As características do sistema de vida artificial que vamos estudar, associada aos SMA, parece enquadrar-se muito bem na definição de Sistemas de Informação durante o seu processo evolutivo (já que funciona segundo uma lógica de estímulo - resposta) mas os processos de aprendizagem dos agentes reflectem um sistema conexionista, uma vez que tem por base as Redes Neurais Artificiais (Sistemas Distribuídos / Inteligência Artificial Distribuída).

Portanto, para o nosso trabalho, assumiremos que a Vida Artificial é uma área de investigação que tem características próprias e que faz a ligação com outras disciplinas nesses objectos de investigação, com quem partilha métodos e processos.

Capítulo 3

Sistemas Multi-Agente

Este capítulo definirá qual o significado científico que vamos adotar para os termos “agente” e “multi-agente”. Esta definição será importante na medida em que existem variantes significativas do sentido em que estes termos são usados pelos vários autores. Uma vez que esta área científica da organização de sistemas multi-agente é muito extensa, apenas referirei os aspectos essenciais necessários ao suporte teórico deste estudo. Para maiores detalhes consultar J. Ferber (1999).

3.1 Agentes

O termo *agente* será assumido neste trabalho como uma entidade física (entidade que evolui num mundo real: robô, computador, *router*...) ou virtual (entidade sem existência física, representada por software: módulo de software, ser representado por modelo de software, ...) que obedece a diversas propriedades (Ferber, 1999):

- Interage com o ambiente onde está inserido (real ou virtual);
- **Pode** comunicar com outros agentes;
- O seu comportamento é orientado por uma tendência (otimização, sobrevivência, competição, prestar um serviço...);
- Compreende pelo menos uma área parcial do ambiente em que está inserido;
- **Pode** ter a capacidade de se reproduzir;
- É autónomo.

Os agentes são, então, um sub-conjunto específico dos sistemas de vida artificial, e, portanto, a sua definição aparece englobada pela definição de vida artificial.

As interações com o ambiente definirão a forma como os agentes interferem com o exterior (suporte de vida ou outros agentes). Esta interferência, obviamente, irá alterar o

estado dessa estrutura exterior que por seu lado interferirá com os sensores do próprio agente, alterando, por sua vez, o seu próprio estado num ciclo contínuo. Este processo dinâmico será o responsável pela evolução do sistema artificial multi-agente (Fig.3.1).

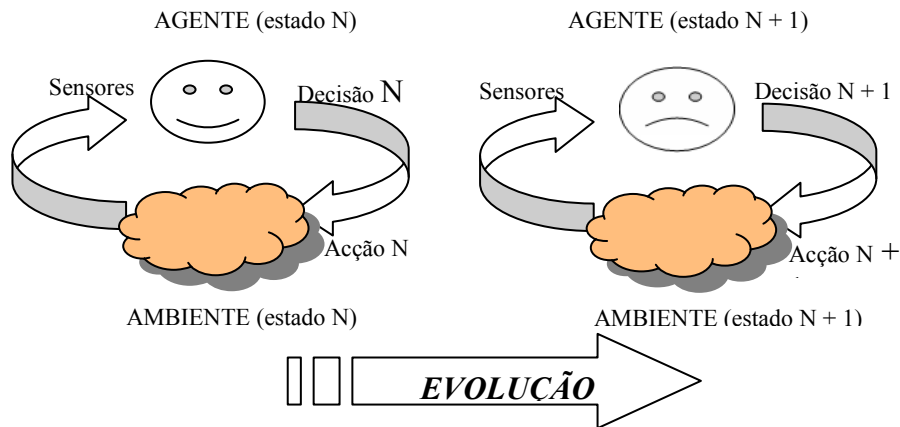


Fig.3.1: Evolução de estados em SMA

Num SMA, os agentes podem ou não comunicar uns com os outros. Os agentes podem evoluir de uma forma completamente independente ignorando a existência dos outros agentes como tal. Isto significa que um agente não interfere no estado de outro agente por qualquer processo à distância, logo por qualquer processo de transferência de informação.

Alguns autores referem-se à tendência que motiva o comportamento do agente, como “propriedade de pró-atividade” (Franklin & Grasser, 1997). Existem ainda autores que preferem o termo “iniciativa” (Wooldridge & Jennings, 1995). Preferimos o termo “tendência” porque é mais abrangente e engloba as acções que fazem alterar o “mundo” de uma forma evolutiva com sentido, mesmo sem estar explícita na intenção dos agentes. Por exemplo, a emissão de um sinal sonoro de uma presa no momento de detectar a presença de um predador é uma tendência da definição do agente (instinto), mesmo que não haja a intenção de avisar outro agente, embora esse sinal possa ser um estímulo útil para outros agentes na proximidade. É, de igual modo, essa tendência que faz o agente evoluir numa procura constante de aumentar a sua energia e impedir a perda dessa energia. A tendência englobará também decisões intencionais como as decorrentes do seu módulo de decisão, por exemplo parar, fugir, perseguir, comer, etc.

De acordo com as várias perspectivas que analisaremos em SMA, os agentes podem ter uma visão parcial do mundo que os rodeia, ou até imediatista se for um agente puramente reactivo, ou podem ter uma compreensão global de todo o sistema, em

sistemas cooperativos, onde todo o sistema está planejado com antecedência e é do conhecimento do agente. Pelo meio, é óbvio, teremos situações que evoluem entre estes extremos, de acordo com o interesse e especificidade do sistema implementado.

Os agentes podem ter a capacidade de se reproduzir. É esta possibilidade que permite que o sistema adquira capacidades plásticas, vantajosas para a adaptabilidade às variações do ambiente, já que os novos agentes poderão ser portadores de características que reequilibram todo o SMA.

A propriedade de autonomia dos agentes é muito importante para o processo de modelação em VA, já que implica que os agentes não serão controlados por qualquer operador externo, ou por outro agente, mas o seu comportamento é apenas uma consequência de um módulo de decisão interno de cada agente. Este módulo funciona como uma caixa preta em que as várias entradas (estados dos sensores) provocam um valor de saída (decisão), dependendo esta das alterações operadas no seu interior (regras definidoras do ambiente em causa, tendência de comportamento, elementos de aprendizagem), como é reportado na Figura 3.2. Muitos autores usam as Redes Neurais para implementar este módulo com excelentes resultados.

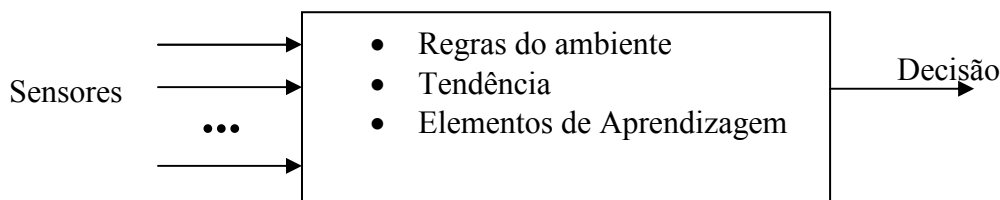


Fig.3.2: Módulo de decisão

As regras do ambiente representam todo o conjunto de regras que definem a existência de todos os elementos no mundo em que evolui o agente (limites físicos do mundo, velocidade, género do ser, aceleração, obstáculos, nascimento, morte, ataque, fuga, aprender, etc,..).

O ambiente é o espaço físico ou virtual onde decorre a vivência dos agentes. Muitos defendem que o ambiente é sempre uma representação do mundo real (Brooks, 1999), enquanto outros rebatem essa tese e defendem que ambientes virtuais (sistemas operativos, Web, ...) também são válidos (Etzioni, 1993).

A tendência representa a motivação que gere a evolução dos seres. Se esta propriedade não existisse teríamos uma evolução completamente casual e aleatória, o que vai contra

o modelo da natureza que como se sabe procura sempre uma optimização (viver mais, ganhar energia, deixar descendência, organizar, comunicar, etc).

Os elementos de aprendizagem, são os componentes responsáveis por permitir que um agente tenha a capacidade de decidir, ou de um modo reactivo, ou de um modo ponderado (processos de aprendizagem, regras de decisão, redes neuronais artificiais, etc). Segundo Michael Huns e Munindar Singh (Huns & Singh, 1997), os processos de aprendizagem podem dividir-se em cinco tipos diferentes:

- Implantação, introdução exterior do conteúdo de conhecimento, sem a iniciativa do agente;
- Transformação, incorporação da representação dos conceitos em signos internos;
- Empirismo, resultado de experiências com resultados positivos ou negativos;
- Analogia, tendência a copiar, ou repetir situações com graus elevados de semelhança;
- Descoberta, conhecimento obtido a partir da geração de respostas baseadas em heurísticas.

As representações dos vários modelos que interpretam estas realidades podem ir desde implementações de agentes simples, até agentes muito complexos que combinam cada uma destas propriedades originando um elevado potencial de aprendizagem.

Existem ainda representações de agentes em que lhes são imputadas características humanóides (crença, obrigação, emoção, veracidade, benevolência, etc) para modelar situações mais específicas (Wooldridge & Jennings, 1995).

3.2 Sistemas Multi-agente

Existem variadas tipologias de SMA. Estas tipologias distinguem-se pelo realce que os investigadores pretendem dar a cada característica dos agentes. Assim, se o modelo necessitar de uma maior visibilidade das características baseadas no tipo de acção (reflexo, análise de modelo do mundo, resposta a objectivos, selecção de respostas úteis) obteremos uma tipologia (Russel & Norving, 1995). Se para além destas características também for importante considerar os atributos do agente, assim como a definição das funções que este executa teremos outra tipologia (Nwana, 1996).

Uma tipologia que servirá de base à organização do nosso modelo é a definida por S. Franklin e A. Grasser (1997). Nesta tipologia, os SMA são divididos em dois grandes grupos: independentes ou cooperativos (Fig.3.3).

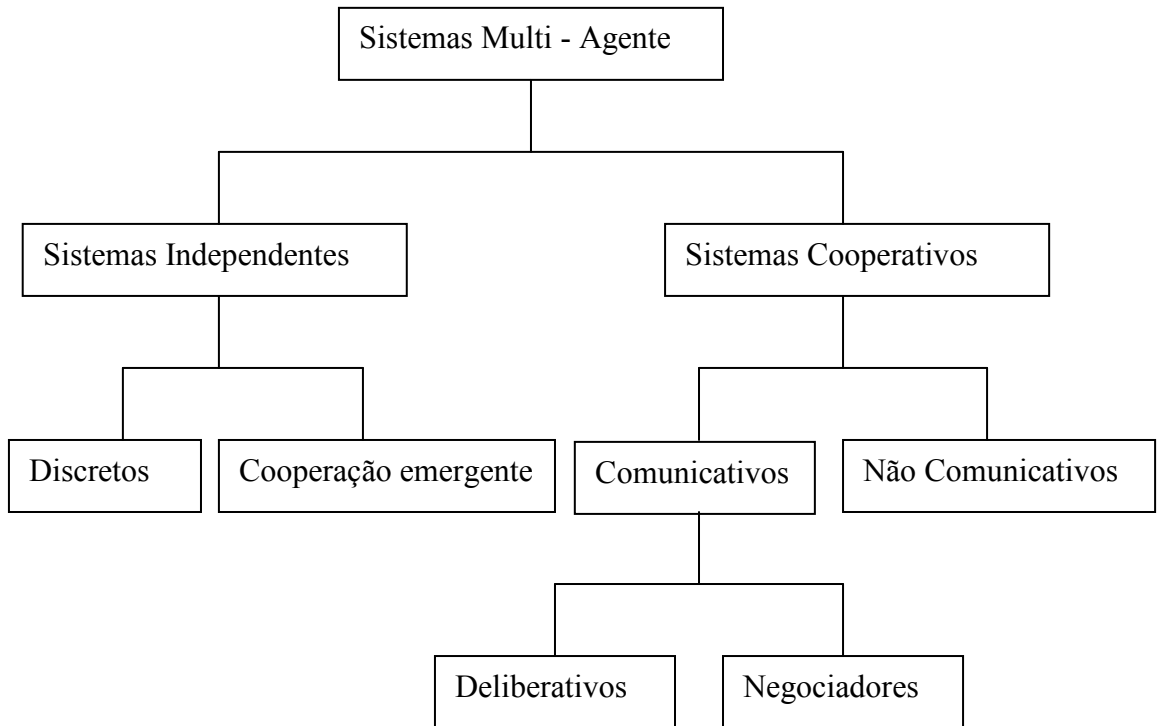


Fig.3.3: Tipologia baseada em Franklin

Para o nosso estudo precisamos de potenciar a propriedade de o agente poder comunicar, referida atrás. Devido ao objecto de estudo implicar uma emergência de linguagem, que permita essa possibilidade de comunicação, então na tipologia de Franklin daremos especial atenção aos sistemas independentes de cooperação emergente. Ainda, para o nosso modelo, não é suficiente ter no mesmo sistema mais do que um agente para obtermos um sistema multi-agente. É necessário também que eles verifiquem a sua propriedade de comunicação com, pelo menos, outro agente. Isto implica que tem que existir, no mínimo, uma relação de interacção entre os agentes que implique uma interferência nos seus estados.

Se considerarmos um sistema multi-agente definido por Ferber, ele consistirá de uma reunião de subconjuntos:

E U O U A U R

em que **E** representa o espaço (ambiente) representativo do sistema, **O** representa um conjunto de objectos (elementos passivos cujo estado pode ser percebido e manipulado pelos agentes presentes), **A** será um conjunto não vazio de agentes e **R** um conjunto de relações possíveis entre os agentes (Ferber, 1999).

Assim sendo, no sistema a que nos propomos, o sub-conjunto **R** terá que ser não vazio. As relações possíveis entre os agentes podem ser: reconhecimento, interação, cooperação, e auto-organização.

3.3 Reconhecimento

Para o nosso modelo, será importante a sub-propriedade de reconhecimento, que está contida nas propriedades gerais dos agentes definidas na secção 3.1. Esta sub-propriedade é decisiva para modelos que evoluam numa lógica de presa-predador, ou em modelos de análise de concorrência entre espécies. De facto, os agentes nestes modelos têm que conter um módulo de decisão, por mais simples que seja. Estes módulos de decisão têm como entrada os estados dos seus sensores, e a decisão só é possível se o agente tiver a capacidade de reconhecimento de objectos ou de situações extraídas desses estados.

Em agentes de tipo independente, essa capacidade de reconhecimento, deverá ser conseguida por aprendizagem. A forma de os agentes conseguirem comunicar esse reconhecimento deverá também ser conseguido por um processo de aprendizagem. Isto garantirá a propriedade de autonomia dos agentes referida antes.

3.4 Interação

Existe interação entre dois agentes quando as decisões de cada um são influenciadas por entradas nos seus sensores que reflectam estados do outro agente. Isto é, se o módulo de decisão de um agente produzir um resultado que reflecte uma reacção à influência de outro agente então este sistema contém relações de interação. A acção de cada agente é condicionada pela acção de outros agentes.

No modelo proposto na secção 3.1, a forma como cada agente reage à interacção com outros agentes será incluído nas regras que definem o ambiente (o mundo artificial em estudo no modelo).

O termo “interacção” é usado por outros autores numa perspectiva mais abrangente (Wortmann & Szirbik, 2001), em que a relação de reacção não se fecha apenas com os outros agentes, mas também com todos os componentes no modelo do mundo artificial em que o agente se movimenta, assim como com elementos exteriores ao sistemas (operadores do sistema).

A primeira definição, mais autónoma e fechada, é a que vai ser usada no nosso modelo, já que pretendemos analisar a evolução de um mundo sem qualquer interferências exteriores.

3.5 Cooperação

O conceito de cooperação entre agentes é outra fonte de controvérsia. Qual a fronteira que separa a interacção da cooperação? Como é que a cooperação se poderá relacionar com a comunicação e auto-organização (Franklin & Grasser, 1997)? Em sistemas independentes, como é que a cooperação se mantém e até que ponto coloca em causa a propriedade de autonomia dos agentes? Estas perguntas são algumas em discussão nos meios científicos, que ainda se mantêm bastante acesa.

Na tipologia de Franklin, os sistemas discretos contêm agentes que têm a sua própria agenda independentemente da acção dos outros agentes. Por exemplo, dois robôs que se movimentam num recinto sem qualquer possibilidade de comunicação. No entanto, se os robôs tiverem capacidade de aprendizagem, embora não possam comunicar, acabam por aprender a movimentar-se sem chocar um com o outro, iniciando uma forma muito rudimentar de cooperação sem comunicação. Os agentes cooperam sem a intenção de o fazer como tal.

Segundo J. Doran e M. Palmer (1995), dois agentes cooperam se:

- Os agentes tendem a comportar-se de forma a conseguirem um objectivo comum;

- Cada agente tende a comportar-se de forma a colaborar com outro agente na obtenção de um objectivo que não é o seu.

Esta definição de cooperação adequa-se a sistemas cooperativos mas também a sistemas independentes com cooperação emergente. A forma como essa capacidade de cooperar é decidida depende do tipo de agente. Do ponto de vista da cooperação, os agentes podem dividir-se em três tipos:

- Agentes reactivos (independentes). A cooperação surge por ser, nesse instante, a melhor escolha de acção, independentemente da intenção de cooperar (Steels, 1990).
- Agentes deliberativos. Os agentes planificam a acção, tendo em conta as suas possibilidades de decisão assim como as dos outros agentes, e depois de um processo negocial.
- Agentes conceptuais. Os agentes são criados de forma a terem comportamentos pré-definidos (crença), entre os quais o comportamento de entreatajuda desinteressada.

Assumiremos, então, que dois agentes cooperam quando a acção de um deles contribui para que o outro evolua para um estado mais favorável, independentemente do facto de ter havido comunicação ou intenção. Essa contribuição pode ser activa ou passiva (informação).

Embora existam várias teorias, iremos debruçar-nos sobre o modelo de C. Zhang (1992) para definir os modos de cooperação quanto à forma de interacção dos agentes envolvidos. Segundo este autor, existem quatro tipos de cooperação:

- Horizontal – Não existe dependência entre os agentes. A cooperação passa pela informação propiciada por outro agente;
- Em árvore – Os agentes só conseguem resolver os seus próprios problemas com a ajuda de outros agentes;
- Recursivas – Os agentes resolvem problemas comuns com entreatajuda;
- Híbrida – Se a cooperação horizontal se verifica enquadrada numa cooperação em árvore ou recursiva.

Para processos de emergência, o tipo de cooperação horizontal parece ser o mais indicado, para além de manter a situação de autonomia dos agentes.

3.6 Organização vs Auto-organização

Entendemos por “organização” em SMA, a forma como as regras de interacção são definidas. Uma sociedade multi-agente é organizada se obedecer a um conjunto de regras que regulamentam a forma como se processam as interacções, as relações de cooperação, regras de negociação e de resolução de conflitos. Durante a investigação tornou-se claro que estas regras são, de uma forma geral, do tipo centralizado. Isto é, na definição do modelo, as regras são definidas à partida, através de um sistema (ou módulo) controlador, que intervém em todos os processos interactivos, inclusive nos processos de decisão de agentes reactivos.

Este tipo de organização, no entanto, está mais adequado a sistemas cooperativos. É fácil de implementar e por isso é o utilizado nos sistemas industriais e aplicações concretas dos Sistemas Multi-Agente. Embora a tendência da investigação se vá orientando no sentido de sistemas cada vez mais independentes e autónomos, também nos sistemas de VA com SMA, se tem optado por utilizar este tipo de organização centralizada. Nos sistemas de Vida Artificial que foram analisados, esta regra é aplicada igualmente, de forma mais ou menos visível. Para que os sistemas fossem independentes e autónomos, *autopoieticos* segundo Maturana e Varela (1973), como tem sido a nossa principal directiva neste trabalho, seria importante que esses sistemas organizassem autonomamente essas interacções. Isto é, seria importante que se promovesse uma **emergência de organização**, a partir de sistemas inicialmente caóticos.

Assim, um sistema com uma auto-organização teria a capacidade de ele próprio decidir as suas próprias regras de interacção entre agentes.

Isto significaria que os sistemas reactivos, totalmente independentes, deveriam desenvolver por si próprios as regras de decisão, interacção, cooperação, negociação e resolução de conflitos, baseados nas tendências inerentes aos próprios agentes (semelhante aos sistemas BDI descritos na secção seguinte) e na análise das

propriedades dos elementos exteriores indicadas pelos seus sensores, complementadas com processos de aprendizagem do próprio agente. Mas definir as regras básicas de interacção de uma forma reactiva, sem qualquer controlo centralizado, não é fácil. Os estudos nesta área não são muito abundantes.

Num SMA de Vida Artificial, por exemplo, os elementos constituintes são definidos centralmente (pelo sistema): quem são predadores e quem são a presa. São definidos também centralmente a forma como um predador se comporta perante uma presa ou perante outro predador. Num sistema auto-organizado puro, o agente (ou ser), deveria reagir de acordo com a informação dos seus sensores (por exemplo: o elemento detectado está a investir ou a fugir, é maior ou menor, tem mais energia ou não, é do mesmo género ou não, etc.) e de um processo de aprendizagem decidido por experiência própria ou por passagem de informação de outros agentes (herança cultural). O nosso trabalho, procurando analisar o processo de emergência de linguagens neste tipo de sistemas, procura também dar alguma contribuição para esta análise destes sistemas totalmente independentes.

Os defensores de sistemas cooperativos são muito críticos desta solução argumentando que os sistemas não podem perder tempo para criar as regras específicas (muito menos em implementações de agentes em sistemas industriais e produtivos), nem correr riscos de interacções erradas até que as decisões comecem a ser coerentes.

No entanto, sistemas auto-organizativos teriam vantagens em muitas situações em que agentes tenham que evoluir num sistema totalmente desconhecido à partida, ou em que operadores centralizados (homem, módulos de controlo) tenham dificuldades em acompanhar a sua evolução, pelo que a sua análise poderia ser de muita importância prática futura.

Uma coisa parece certa: a emergência de comunicação seria uma peça fulcral para a emergência da organização. E a emergência da comunicação só terá sentido com uma emergência de linguagem.

3.7 Sistemas cooperativos

Os sistemas multi-agente, podem ser ou não cooperativos. Normalmente, este tipo de sistemas é estático, e a sua evolução obedece a uma planificação em que as opções de

acção estão praticamente definidas na definição do sistema. Isto significa que estes agentes não estão particularmente preparados para processos de aprendizagem, embora possam possuir esta capacidade de uma forma não desenvolvida, para dar alguma autonomia a certos aspectos da sua evolução. Esta rigidez está adequada a certo tipo de problemas, mas inadequa-se a problemas que exijam bastante elasticidade de decisão, ou a problemas em que os parâmetros em causa sofram variações com elevado grau de imprevisibilidade.

Como a evolução nestes sistemas obedece normalmente a uma planificação pré-estabelecida, então os agentes que neles evoluem poderão ter a possibilidade de serem ou não comunicativos. Com efeito, existe uma certa centralização do processo de acção, que também regula as passagens de informação necessárias. Estes modelos procuram otimizar sistemas em que seja necessário uma grande colaboração dos agentes com o fim de obter características de resolução de problemas muito complexos. Isto impõe uma grande capacidade de cooperação aos agentes constituintes do sistema, com regras de participação e organização bastante elaboradas na planificação do sistema.

Dentro deste tipo de modelos, encontram-se os sistemas que reflectem o comportamento humano para a tomada de decisão, nomeadamente sistemas BDI (Believe, Desire, Intention) (Fig.3.4), que orientam a decisão do agente segundo tendências baseadas nos impulsos humanizados de Crença, Desejo e Intenção (Shoam, 1993), (Corrêa & Coelho, 1993), (D’Inverno et al, 1997). A reacção a um estímulo é ponderada com o recurso às três tendências BDI referidas.

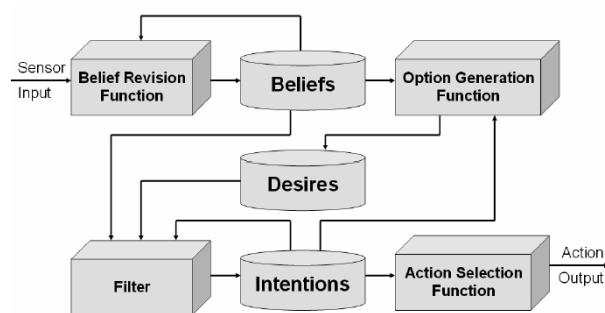


Fig.3.4: Sistema BDI genérico

3.8 Sistemas reactivos

Os sistemas multi-agente reactivos (independentes) são um paradigma diferente e oposto dos sistemas cooperativos. Estes sistemas sustentam que o comportamento dos agentes se define apenas como processos de reacção a estímulos. Esse comportamento é também intrínseco a cada agente, independente e autónomo. Nestes sistemas, o comportamento com características de inteligência (como a capacidade de cooperação) de um agente, não advém de modelos do seu mundo já predefinidos na sua estrutura, mas emerge de interacções básicas com o ambiente, produzidas por comportamentos simples (Brooks, 1986).

Para Franklin, um SMA é independente se cada agente tiver a sua própria agenda, sem tomar em consideração a agenda dos outros agentes (Franklin & Grasser, 1997).

Os agentes reactivos são um modelo muito útil para problemas onde seja necessária uma grande capacidade de adaptação, em que o comportamento dos agentes consiga reajustar-se constantemente a alterações do ambiente.

Muito interessantes são os agentes que na tipologia de Franklin, são referidos como Agentes Independentes de Cooperação Emergente, como vimos atrás. Inserem-se neste caso agentes que emulam a Vida Artificial. Será, portanto, um paradigma muito útil para a utilização de modelos que recorram a Algoritmos Evolucionários. O grande problema nestes sistemas é o facto de a teoria ainda estar pouco definida para regulamentar cientificamente este tipo de modelos, já que é uma área muito recente. Por exemplo, a definição de algoritmos que definam a emergência de comportamentos a partir das interacções reactivas ainda se mantém numa fase muito embrionária.

O nosso trabalho consiste, exactamente, na investigação, de emergência de linguagens em sistemas reactivos, que conduzam a uma decisão de comportamento baseado na informação daí resultante. Consiste também em investigar se essas decisões tornam o agente mais capaz, logo, com maior potencial para obter soluções vantajosas. As ferramentas, como veremos, são as Redes Neurais Artificiais e os Algoritmos Evolucionários.

Capítulo 4

Comunicação Artificial

Neste capítulo, procuraremos esclarecer os termos e ideias que estão associados ao processo de comunicação. Este processo é dos mais complexos e é abordado, mais uma vez, de pontos de vista bastante diferentes dependendo da área científica em causa. Para o nosso estudo, estamos particularmente interessados num conjunto de termos e ideias que possam caracterizar os métodos e a forma como dois seres artificiais possam passar uma informação de um para o outro, assim como o modo como essa informação pode ser interpretada.

4.1 Comunicação

Uma comunicação implica a passagem de informação entre um emissor e pelo menos um receptor, e a sua compreensão por parte destes. Assumimos que possa haver transmissão de informação sem comunicação se o conjunto de informações enviado não for compreendido ou integrado pelo receptor. Por exemplo, se o agente A enviar um sinal S1 identificador de um objecto O, mas este objecto é identificado pelo agente B com um sinal S2, diferente de S1, então, embora A transmita uma informação para B, não lhe comunica nada porque este não compreende a mensagem. B recebe um sinal de A que não processa porque não o compreende. Esta definição permite estabelecer uma diferença entre comunicar e transmitir.

Para além do emissor e do receptor, assim como do meio usado para a transmissão, será considerado também neste modelo a forma como a mensagem é codificada. Adoptaremos a proposta, para esta forma de codificação, apresentada por Ferber (1999), que se ajusta correctamente à análise a que nos propomos. Este modelo é fundamentado através de três termos relacionados entre si:

- Um **sinal** será uma alteração do ambiente, que se propaga entre dois pontos (emissor, receptor). O modo de propagação pode ser muito diversificado, mas

será apercebido pelo receptor (nos seres vivos através dos seus sentidos, nos seres virtuais por impulsos eléctricos). Estes sinais, se forem compreendidos, terão como consequência um estímulo ou uma alteração de comportamento por parte do receptor. O estímulo poderá ser um reforço de tendência, o que não implica necessariamente uma reacção imediata do comportamento do agente. Podemos considerar neste caso o processo de aprendizagem de linguagem.

- Um **signo** será a representação do significado de um sinal quando este é incorporado no processo cognitivo (Eco, 1986).
- Um **indício** será um tipo de sinal que permite aceder de uma forma indirecta a um determinado significado. Por exemplo, a visão de fumo (indício) pode permitir a representação de um fogo. Um indício é portanto um sinal ou conjunto de sinais que permitem o acesso a um significado que seria representado por um sinal diferente. Este significado será uma hipótese.

A forma como é feita a integração do signo no processo cognitivo, envolve o tipo de relação entre três entidades: o próprio Signo, o Objecto que o signo representa e o Interpretante (Peirce, 1958). De acordo com essa relação, o signo pode ser dividido em três sub-entidades:

- **Ícone**, quando o objecto é representado directamente (existem propriedades semelhantes, intimamente relacionadas entre o objecto e o signo),
- **Índice**, se o objecto é representado numa associação indirecta (o objecto é representado como uma consequência de uma relação directa, existindo por isso uma ligação contínua entre o objecto e a sua representação),
- **Símbolo**, se o objecto é representado por uma associação descontínua (não existem propriedades partilhadas mas a associação é feita por regra ou hábito).

4.2 Conversação

Uma conversação é um encadeamento de estados de comunicação entre um emissor e um receptor em que cada estado depende do estado anterior. Isto é, a comunicação associada a um estado é consequência do conteúdo da comunicação ocorrida no estado

anterior (Winograd & Flores, 1986). Uma conversação é caracterizada por algumas propriedades fundamentais:

- Tem um início, que normalmente é devido à intenção de um agente enviar uma comunicação para outro agente;
- Tem um fim, que pode ser induzido por um sinal de desejo de terminar o processo de trocas de comunicação;
- Cada estado da conversação pode dar origem a um conjunto reduzido de acções: recusa, aceitação, inquirição,...;
- Tem a capacidade de alterar não só o controlo e orientação da própria conversação, mas também de alterar o estado dos agentes envolvidos: contribui para o reforço das tendências.

Diversos estudiosos analisam estes processos e criam modelos usando diagramas de transição de estado, Redes de Petri, ou cadeias de Markov. Este desenvolvimento não faz parte deste nosso estudo, que apenas se vai reportar a um estado inicial da emergência de linguagem e não dos processos de conversação em si, pelo que não aprofundaremos este aspecto.

4.3 Aquisição de signos por agentes

Para um agente poder desenvolver uma possibilidade de comunicação ele deverá ter a capacidade de atribuir autonomamente um signo, novo ou já conhecido de experiências anteriores, a qualquer perturbação da sua vizinhança detectável. Isto permite a sua catalogação e armazenamento relacional na sua memória. A forma como este processo é desencadeado é um dos grandes problemas da Inteligência Artificial, responsável pela introdução de uma nova linha de investigação, controversa entre diferentes posições, conhecida como ‘Problema da Fundamentação do Símbolo’ (Harnad, 1990), que sustenta a validade das regras que permitam uma ligação correcta entre os signos e os objectos que eles representam. Para o nosso estudo interessa apenas considerar que a aquisição de signos por parte dos agentes, tem que verificar as seguintes propriedades:

- Autonomia: o signo tem que ser gerado ou associado pelo próprio agente;
- Adaptabilidade: a relação dos signos com o respectivo objecto, não deverá ser rígida, mas flexível permitindo um processo de aprendizagem, em que essa relação poderá fortalecer ou atenuar;
- Abertura: o agente deverá ser capaz de atribuir signos a todas as situações novas que sejam introduzidas no sistema.

4.4 Emergência de Linguagens

A principal *tendência* que define um *agente* é a sobrevivência. Esta consegue-se com uma *interacção* com o ambiente de forma a procurar uma optimização permanente de procura de energia (que nos seres vivos se traduz pela procura de alimento). Num sistema do tipo presa-predador também é considerada a *interacção* com o ambiente que traduz uma acção de fuga da presa relativamente ao predador. Em sistemas reais (sociais) verifica-se que os agentes envolvidos acabam por estabelecer uma forma mais ou menos complexa de comunicação. Agentes que interagem e comunicam com o “grupo” parecem ter mais chances de sobreviver (mais energia ou mais hipóteses de fuga). O modelo da natureza dotou portanto os indivíduos dessa possibilidade.

A emergência de uma linguagem terá como base, portanto, a consequência evolutiva da capacidade da aquisição de signos por parte de agentes. A aquisição de signos e a sua exteriorização para a vizinhança permitirá um processo de relacionamentos, signos-sinais-consequências, que associado a um sistema de aprendizagem, mais ou menos complexo, originará uma proto-linguagem. É a este processo de criação de uma proto-linguagem que chamaremos de emergência de linguagem. Obviamente que ela não suportará ainda as regras semânticas ou sintácticas, próprias de uma linguagem já estabelecida, mas apenas a atribuição e aprendizagem de signos a sinais obtidos da vizinhança.

Capítulo 5

Redes Neurais Artificiais e Algoritmos Evolucionários

Neste capítulo, procurar-se-á fazer uma breve referência aos aspectos que deveremos tomar em consideração para a elaboração e compreensão do trabalho, no que diz respeito à evolução dos agentes portadores dos signos. Serão assumidas algumas definições e conceitos que respeitarão aos temas das redes neuronais, dos modelos de aprendizagem e dos algoritmos evolucionários.

5.1 Redes Neurais Artificiais

Pretende-se realçar, nesta vasta área de investigação, os aspectos mais relevantes da teoria das Redes Neurais Artificiais, para este trabalho. Para uma mais profunda análise teórica destes temas remeto para o trabalho desenvolvido por M. Rocha (2004).

5.1.1 Definições

As Redes Neurais Artificiais, são uma tentativa de reprodução tecnológica de um dos maiores e complexos êxitos da natureza: o cérebro humano. É obvio que esta reprodução se apresenta, actualmente, impossível devido às limitações técnicas actuais, assim como aos conhecimentos ainda rudimentares sobre o funcionamento físico e lógico do cérebro. Outra dificuldade advém da influência de certas correntes de investigação em Ciências Neuronais que defendem que, melhor que imitar o cérebro, deveriam ser desenvolvidas tecnologias e algoritmos que fossem mais eficazes e adequados que o próprio cérebro para a resolução de problemas complexos específicos. No entanto, é pacífico que existem certos tipos de modelos de funcionamento e técnicas decorrentes da análise do cérebro humano que são possíveis e interessantes de aplicar.

Um desses tipos de modelos é a forma básica que o cérebro utiliza para guardar e processar informação, que se pode simplificar num modelo de Rede Neuronal.

Neurónios Artificiais

Os neurónios artificiais são um modelo de funcionamento de um neurónio biológico, e cuja representação deste se pode ver na figura 5.1:

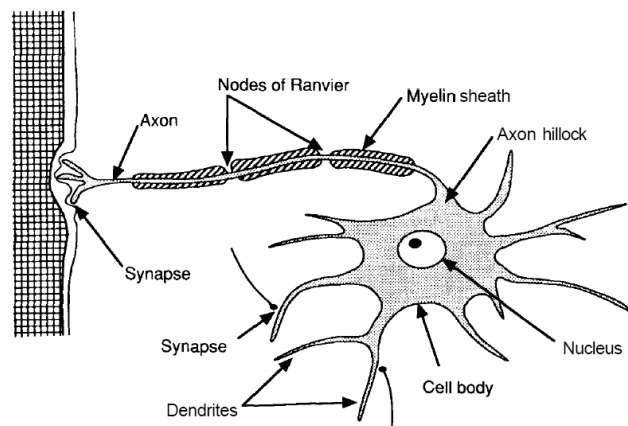


Fig.5.1: Representação de um neurónio

A representação artificial deste neurónio pode surgir com diversos formatos, sendo a sua forma esquemática mais geral, mostrada na figura 5.2:

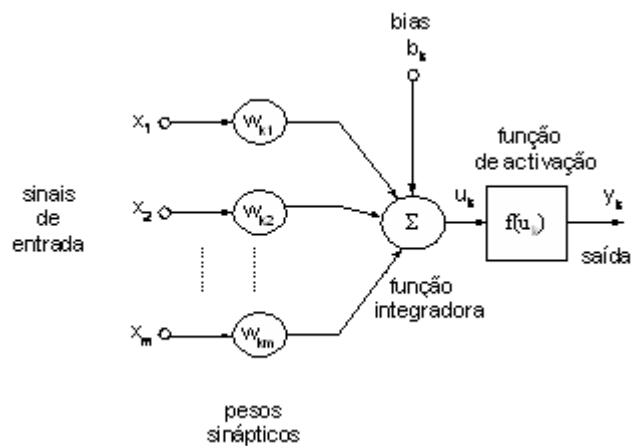


Fig.5.2: Esquema geral de um neurónio artificial

Os modelos funcionais deste esquema podem ser vários, (por exemplo, o modelo de McCulloch-Pitts), em que alguns se podem mesmo abstrair alguns dos componentes do esquema geral visto atrás. No nosso caso, iremos optar pelo modelo representado na seguinte estrutura da fig.5.3:

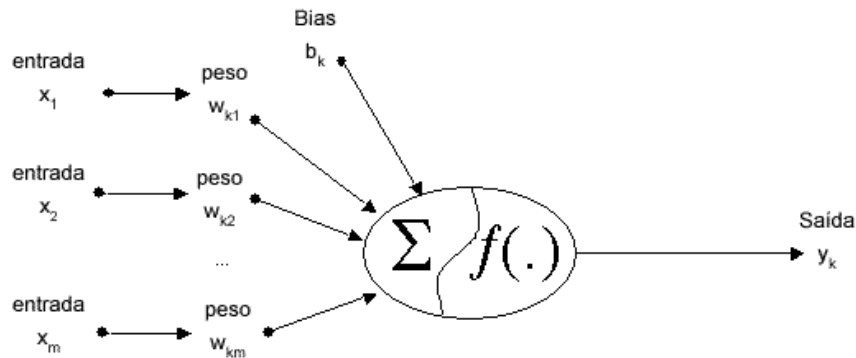


Fig.5.3: Modelo Funcional de Neurónio Artificial

Neste modelo, iremos considerar os seus elementos constituintes:

- Conexões de entrada (sinapses) – cada entrada é associada a um peso que produzirá um *efeito excitatório* se esse peso tiver sinal positivo, ou um *efeito inibitório* se tiver um sinal negativo.
- Bias – conexão de entrada especial que não recebe valores a jusante, mas apenas incrementa o peso da sua ligação na das conexões.
- Nó – será o local onde as conexões de entrada se unem através de uma função integradora, *combinador linear*, que neste modelo assumimos como somatório, Σ . É também neste local que se aplica a função de activação, para produzir um valor de saída.
- Combinador linear (Σ) – função que integra todas as entradas depois de multiplicadas pelos respectivos pesos
- Função de Activação - O valor resultante da junção das conexões de entrada é tratado pela aplicação de uma função, chamada *função de activação*. Normalmente esta função é não-linear, sendo muito popular a *função sigmoid*, ou função logística, com parametrização.

- Saída do neurónio – valor que resulta da aplicação da função de activação ao somatório das saídas das conexões, representado pela seguinte expressão matemática:

$$y_k = f(u_k) = f\left(\sum_{j=1}^m w_{kj} x_j + b_k\right)$$

em que:

y_k – saída do neurónio k

w_{kj} – peso da sinapse que liga o neurónio j ao neurónio k

x_j – valor da entrada na sinapse que provém do neurónio x

b_k – valor da Bias

m – valor total de sinapses do neurónio k que provém de outros neurónios

As sigmóides são uma das funções de activação mais populares (Fig.5.4).

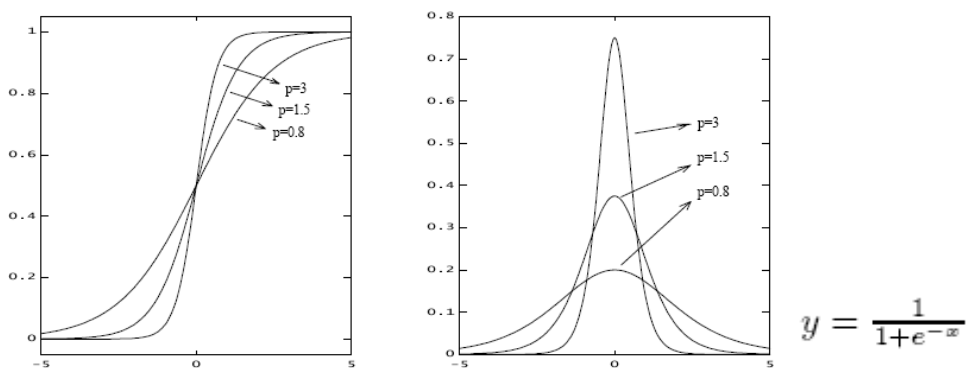


Fig.5.4: Função Sigmoid e sua derivada, com parametrização

O objectivo de uma rede neuronal é armazenar informação, e processar valores de saída para as excitações dos valores de entrada, o que, estabelecendo uma rede de neurónios artificiais conectados segundo diversas topologias, pode permitir uma grande capacidade de aprendizagem e processamento de informação. Por isso, as RNA's são muito utilizadas em sistemas que exigem grande maleabilidade de aprendizagem.

Topologias de RNA's

Entende-se por topologias das RNA's, a forma como os neurónios artificiais se interligam. Estas ligações apresentam uma estrutura típica de camadas, onde a primeira camada engloba os nós cujas conexões de entrada recebem os valores correspondentes às excitações externas à topologia. Podem apresentar uma camada de saída que agrupa os nós cujos valores de saída são disponibilizados para o exterior da topologia, e uma camada intermédia que agrupa os nós cujas entradas são alimentadas por nós de uma camada anterior e cujas saídas alimentam camadas posteriores. O número de camadas intermédias é variável, mas são bastante populares as estruturas que apresentam apenas uma camada intermédia, devido à sua simplicidade de tratamento, assim como a uma grande eficácia. Na figura 5.5 apresenta-se uma estrutura tipo, com duas camadas intermédias:

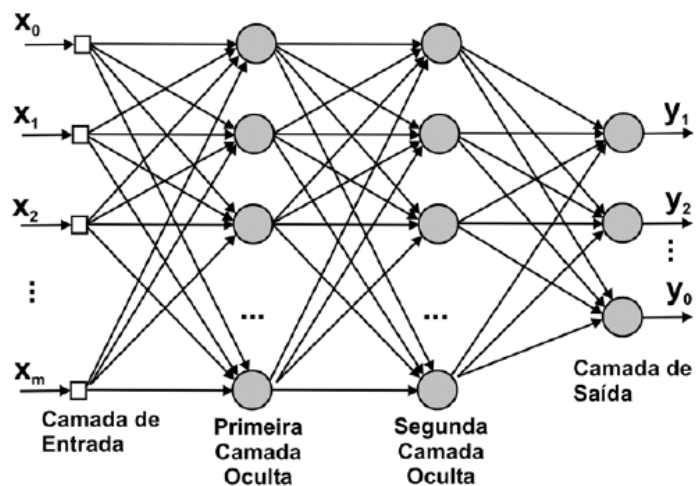


Fig.5.5: Estrutura de RNA com duas camadas intermédias.

Se todos os nós de cada camada ligarem a todos os nós da camada seguinte, então teremos uma estrutura chamada *completamente ligada*. Chamaremos ainda de *atalho*, as ligações entre nós que não estejam em camadas consecutivas.

Estas estruturas são representadas por um código simples. Por exemplo, o código

[2-3-1-2]

representa uma topologia com 2 nós de entrada, 3 nós na segunda camada, 1 nó na terceira camada, e 2 nós na camada de saída. Tem, portanto, 2 camadas intermédias.

Existem estruturas que não têm camada de saída, estando mais adaptadas para modelar sistemas de aprendizagem não supervisionada (Kohonen, 1982).

As topologias estão associadas aos diferentes tipos de aprendizagem que se pretende implementar. Podem ser consideradas diversas tipologias mais usuais:

- Topologias Unidireccionais (RNAU) – no grafo que constitui a rede, não existem nós em que as entradas sejam realimentadas com valores das suas saídas ou de saídas de nós posteriores;
- Topologias Recorrentes (RNAR) – permite que as entradas sejam alimentadas com as saídas de nós posteriores, ou da própria saída, criando ciclos.

Treinos de RNA's

As RNA's começam num estado em que, à partida, não sabem quais os problemas concretos com que vão ser confrontadas. Sabe-se, no entanto, que elas terão que aprender e preparar-se para responder a qualquer situação, dentro do ambiente do problema para a qual foram criadas. É assim que os pesos das RNA's, normalmente, são inicializados com valores aleatórios, que podem ser condicionados, ou não, se se souber alguma informação restritiva logo de início. A partir desse estado inicial, a rede terá que ter um treino específico. Entende-se por treino específico o treino da rede obtido pela sua confrontação com situações que evoluem dentro de determinadas características. Supõe-se que estas características serão semelhantes às que irão balizar os casos que mais tarde a rede terá que resolver.

No processo de treino, a RNA ajustará os seus pesos de forma a que a resposta para uma determinada entrada obtenha sucesso. O sucesso pode ser total ou parcial, de acordo com o modelo de aprendizagem em causa.

Serão considerados exemplos, elementos independentes que contenham a informação necessária à aprendizagem da rede, e que atribuem valores, contínuos ou discretos, inteiros, reais, binários ou lógicos, a atributos representativos das características dessa informação.

Os exemplos que irão servir de treino para a rede são chamados de casos de treino, e o processo de treino tem bastantes variantes. Estas variantes vão desde a execução sequencial simples, de uma lista de casos de treino até à escolha de grupos ou subgrupos

de elementos dessa lista, recombinação desses grupos, critérios de selecção mais ou menos sofisticados para esses agrupamentos, etc.

A avaliação do sucesso do treino de uma RNA é feita com *exemplos de teste*, que reflectem situações não previstas nos exemplos de treino e medem a capacidade que a RNA desenvolveu para responder a situações novas. Esta avaliação é feita com os chamados *métodos de estimação*, que aqui não iremos abordar, mas que podem ser consultados no trabalho de Rocha (2004).

Alguns dos algoritmos de treino mais conhecidos são: o Algoritmo de Retropropagação e o Algoritmo RProp.

Algoritmo de Retropropagação

Trata-se de um algoritmo da classe dos supervisionados desenvolvido nos trabalhos de Rumelhart, Hinton e Williams (1986). Este algoritmo actualiza os pesos de forma a minimizar a função de erro, através da análise do seu gradiente. O algoritmo desencadeia duas acções consecutivas: primeiro impõe uma propagação frontal, em que os valores fornecidos à entrada projectam as saídas dos neurónios num processo que se vai propagando até à saída. De seguida, aplica-se uma função de erro, que se vai projectando desde a saída até aos neurónios de entrada. Finalmente os pesos são recalculados utilizando a regra de Widrow-Hoff (1960), seguindo uma direcção de minimização desses erros.

Algoritmo RProp

O algoritmo de treino das RNA's conhecido por RProp, Resilient backPropagation, é um algoritmo muito usado na IA (Riedmiller & Braun, 1993). O algoritmo anterior, de Retropropagação, toma em consideração os valores das derivadas parciais¹ (Freeman, J.A., Shapura, D.M., 1991). No entanto, o valor deste diferencial não representa o valor realmente necessário para a alteração dos pesos. O algoritmo RProp, então, numa perspectiva mais simplificada, não calcula este valor desnecessário, e apenas considera

¹ Desenvolvimento matemático em “Neural Networks, Algorithms, Applications, and Programming Techniques” de J.A.Freeman e D.M. Shapura, capítulo 4.

o valor do sinal da derivada parcial, indicadora da direcção da alteração do peso. Neste algoritmo, se o sinal da derivada é positivo o peso incrementa de um valor de actualização, se é negativo decrementa esse valor e se inverte o sinal relativamente ao sinal anterior (o máximo foi ultrapassado), então o sinal de actualização do peso anterior é revertido.

5.2 Modelos de Aprendizagem

As RNA's têm enormes potencialidades devido à sua capacidade de reflectir duas propriedades muito importantes, que podem estar relacionadas: a memorização e a aprendizagem. O conceito que vamos assumir para aprendizagem pelos sistemas artificiais, será o de T. Mitchell (1997) :

“Um programa de computador aprende com a experiência E, numa classe de tarefas T, e dada uma medida de desempenho P, se o seu desempenho nas tarefas contidas em T, medida por P, melhora com a experiência E.”

A aprendizagem, por sistemas artificiais, pode ser de três tipos, segundo D. Patterson (1996): a Aprendizagem Supervisionada (AS), a Aprendizagem por Reforço (AR), e a Aprendizagem não Supervisionada (ANS), de que passaremos a fazer uma breve descrição.

5.2.1. Aprendizagem Supervisionada

O sistema generaliza o comportamento a partir de diversas situações em que sabe à partida as soluções, e que se chamam de “casos de treino”. A RNA, a partir deste conhecimento, fica preparada (treinada) para deduzir outras situações diferentes (casos de teste). Normalmente as redes são treinadas aplicando vários processos de treino em que, para cada vector de entrada, é dado um vector de saída correcto. A RNA terá que ajustar a sua estrutura e pesos, para que esteja preparada para reagir correctamente a

essas situações. Assim, já fora do processo de treino, a RNA poderá ter respostas que se podem considerar sustentadas, para novas situações reais. Estes modelos estão muito relacionados com os dados, e com as estruturas das RNA.

5.2.2. Aprendizagem por Reforço

Este tipo de aprendizagem segue o mesmo processo de supervisão, mas, neste caso, não é conhecida a resposta correcta. O sistema apenas é informado se a resposta que deu é benéfica ou prejudicial para o seu desempenho. Assim, funciona como um binário recompensa-punição, em que o sistema aprende a seguir as estratégias que levam à recompensa e a evitar as estratégias que levam à punição. No caso de aprendizagem do tipo “emergência” (de linguagens, comunicação, organização, etc) este modelo é bastante adequado já que nestes casos não se sabe a resposta correcta para cada acto, mas sabe-se em cada instante, se cada acto gerado foi ou não vantajoso para o agente, nas consequências com o exterior (que neste caso funciona como professor), provocando a longo prazo um processo de emergência.

5.2.3 Aprendizagem não supervisionada

O sistema artificial não é informado exteriormente sobre o problema ou consequências de uma qualquer resposta. O sistema é fechado, e a aprendizagem é feita por tendências internas (referidas no capítulo 3 sobre sistemas multi-agente) que evoluem a descoberta de características específicas nos dados de entrada, tais como, regularidades, agrupamentos, repetições, etc.

Nas RNA's este modelo coincide com as estruturas em que não existe camada de saída.

5.3 Algoritmos Evolucionários

A utilização do termo “Algoritmo Genético” teve o seu início com o trabalho de J. Holland (1975). Estes algoritmos surgem como uma solução de programação inspirados na evolução natural. No entanto, várias alternativas têm sido propostas, tomando o

nome mais genérico de Algoritmos Evolucionários (AE's). São utilizados, normalmente, em problemas não lineares, em que as variáveis em jogo mostram relações de dependência mais ou menos fortes entre si. Devido a esta dependência entre as variáveis, estes problemas tornam-se muito complexos, e os AE's têm mostrado ser uma solução muito capaz.

Embora existam várias estratégias, estas consistem, essencialmente, na criação de uma população, normalmente aleatória, de elementos portadores de cromossomas. Os cromossomas representam soluções, e os algoritmos são desenhados de forma a garantir que a evolução atribua mais possibilidades de reprodução aos elementos com melhores soluções, eliminando os que têm soluções (cromossomas) piores. Esta evolução é feita através de contínuos processos reprodutivos em que os elementos herdam as soluções dos seus progenitores numa contínua busca da melhor adaptação.

As heranças podem ser feitas de vários modos e tipicamente contêm. As mutações são pequenas alterações aleatórias introduzidas nos cromossomas herdados. Se estas alterações aproximarem o cromossoma de uma melhor solução diz-se que temos uma mutação positiva. De outro modo, gerou-se uma mutação negativa, que nos seres humanos se traduz por doenças genéticas que tornam o indivíduo menos capaz.

As mutações são uma forma de introduzir um factor de diversidade e abertura que normalmente potencia uma melhor pesquisa já que as mutações positivas podem orientar de uma forma mais rápida a convergência de uma linhagem de soluções, enquanto as mutações negativas podem acelerar o desaparecimento de elementos inadaptados ao sistema. Estas mutações têm ainda a vantagem de impedir que os elementos evoluam na proximidade de “óptimos locais”, permitindo criar soluções que saltam para outras zonas de pesquisa, que poderão ser potencialmente melhores. A diversidade de aplicações originou alguns modelos base de programação deste tipo de algoritmos.

5.3.1 Definições

Os termos usados na programação genética são os mesmos usados na Genética, e que usaremos neste trabalho com o seguinte significado:

- Genes – conjunto de parâmetros que caracterizam uma solução e representam os problemas a resolver;

- Cromossomas – cadeia de parâmetros que representa a sequência de genes;
- Genoma ou genótipo– conjunto de todos os cromossomas. Em muitos casos é usado como sinónimo de cromossoma dado que apenas um é usado.
- Fenótipo – resultado físico da evolução de um genótipo;
- Locus – localização de um gene dentro do cromossoma.

Os AE's incorporam alguns componentes importantes na sua definição e parametrização:

- Representação;
- Codificação e Descodificação do Problema;
- Função de Avaliação;
- Operadores Genéticos;
- Parâmetros de Evolução.

A Representação define a estrutura dos cromossomas, com a sua configuração de genes, de forma a descrever o espaço de busca do problema. Este espaço pode não ser o total possível, mas o mais relevante para o problema em causa. Esta representação pode ser feita com números binários, reais, inteiros, ou outros símbolos menos usuais.

A Codificação passa por transformar cada solução de um problema num conjunto de parâmetros (genes), que são agrupados num conjunto de valores (cromossomas), e, finalmente, estes são codificadas num formato controlável por um algoritmo (Goldberg, 1989), utilizando um alfabeto de codificação. Inicialmente, usou-se o alfabeto binário, mas mais recentemente dá-se particular importância aos números reais ou a outras codificações mais próximas do problema (Michalewicz, 1996). A Descodificação executa uma construção da solução a partir do cromossoma, usando um processo inverso, para que esta seja sujeita a uma avaliação pelo problema.

A Função de Avaliação, ou função de adaptação, avalia cada indivíduo (solução), na população corrente, para definir o seu grau de adaptabilidade ao sistema. Esta Função permite assim uma avaliação da proximidade do algoritmo à solução óptima o que garante uma orientação do processo de busca. Grande parte do processamento do algoritmo faz-se na avaliação desta função para cada indivíduo. É comum associar a cada cromossoma um valor, valor de adaptação, que é proporcional à adaptabilidade ou utilidade do indivíduo para solucionar o problema considerado.

Os Operadores Genéticos envolvem os métodos utilizados para passar a informação genética de uma geração para a geração seguinte. Os mais conhecidos são:

- Selecção e reinserção – agrupa os operadores responsáveis pelo processo de selecção e cópia de um cromossoma para a geração seguinte, tendo em conta o seu valor de aptidão, bem como os processos de inserção de novos indivíduos na população.
- Cruzamento (Crossover) – consiste no método de cruzamento de genes de dois progenitores com o fim de gerar uma combinação de um novo código no(s) descendente(s). O mais conhecido é o *cruzamento de ponto único* (fig.5.6).

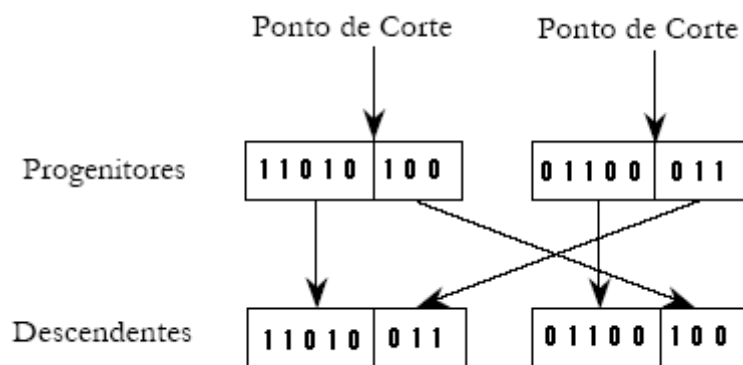


Fig.5.6: Operação de cruzamento com ponto único

Podem, no entanto, ser usadas outras técnicas como o *multi-point crossover* (Michalewicz, 1996), em que existem mais do que um ponto de corte para cruzamento. Existe ainda um método chamado de *cruzamento uniforme*, que utiliza uma máscara binária, gerada aleatoriamente, de comprimento igual ao dos cromossomas, e que decide quais são os genes que os descendentes herdam (Fig.5.7).

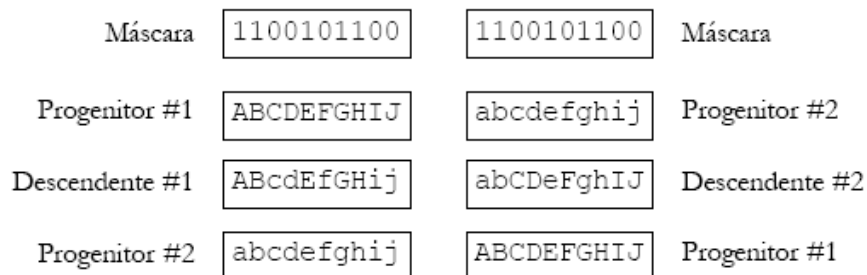


Fig.5.7: Operação de cruzamento uniforme

Dependendo das representações, podem ainda existir outras abordagens;

- Mutações - representa a acção de uma alteração ínfima e imposta, ao código genético de um indivíduo, que ocorre de forma aleatória. A mutação pode ser feita de várias formas, algumas delas dependentes dos tipos de dados que configuram a codificação do cromossoma. Por exemplo, se a codificação for binária, a mutação pode ser a alteração do valor de um bit, numa determinada posição. Esta mutação é conhecida por *mutação binária*. Chama-se *ponto de mutação*, à posição do bit que vai sofrer a alteração de valor. Outras mutações podem ser definidas para codificações por números reais, cadeias de caracteres, ou outras. No caso dos números reais, as mutações podem ocorrer pela adição de um valor mediante regras pré-definidas que dependem do algoritmo. É comum utilizar a distribuição de Gauss de média nula para gerar este valor.

Os Parâmetros de Evolução, representam os parâmetros que mais interferem no desempenho da maioria dos AE's, e são, nomeadamente:

- Tamanho da População – representa o número de elementos que constituem um determinado grupo de análise. Se este valor for muito baixo, o espaço de cobertura é reduzido e a convergência demorada ou até do tipo Finalização Lenta (definida à frente). Se este valor for elevado, a convergência é mais rápida e previne convergência prematura em soluções locais. No entanto, valores muito elevados podem pôr em causa o desempenho do sistema, devido a maior exigência dos recursos informáticos;
- Taxa de Substituição – define qual a percentagem de população que vai ser sujeita ao processo de substituição na geração seguinte;
- Taxa de Cruzamento – representa a probabilidade de um elemento ser recombinado com outro. Este valor deve ser alto (tipicamente à volta dos 90%). No entanto, isto pode em alguns problemas dar origem à perda de elementos com elevados valores de aptidão. Assim, alguns tipos específicos de problemas exigem taxas mais baixas, da ordem dos 60%. Taxas muito baixas podem originar um atraso decisivo no desenvolvimento do algoritmo;
- Taxa de Mutações - representa a percentagem de elementos que sofre processos de mutação. Um valor muito alto pode originar uma busca num processo aleatório sem interesse. Por isso opta-se por valores muito baixos neste parâmetro, tipicamente entre 0.5% e 1%;

- Número de Gerações – define o número de ciclos de evolução de um AE, constituindo, portanto um dos critérios de paragem do algoritmo. Um valor pequeno origina uma insuficiência de desempenho do problema, e um valor elevado, embora muito exigente em termos de recursos, produz uma cobertura maior do domínio do problema.

Os algoritmos genéticos podem ser confrontados com alguns problemas padrão:

- Convergência Prematura – um grupo de indivíduos, com elevado valor de adaptação, podem sobrepor-se ao resto da população, provocando uma convergência rápida para um máximo que pode ser local. O resultado é uma solução insuficiente;
- Finalização Lenta – a população converge muito lentamente, não encontrando uma solução satisfatória em tempo útil.

Podemos considerar três fases na execução de um algoritmo genético:

- Criação da população inicial – na maioria das aplicações a população inicial é obtida de forma aleatória. São criados indivíduos com os respectivos cromossomas e depois avalia-se cada um para lhe atribuir um valor de aptidão.
- Reprodução – durante as iterações são seleccionados e recombinados pares de indivíduos para gerar novos indivíduos. A selecção desses pares pode seguir várias estratégias:
 - Aleatória – é deixada à evolução a tarefa de reduzir o seu tempo disponível de reprodução, por vários motivos, um dos quais, prazos de vida mais curtos, devido a menor capacidade de impedir a perda de energia. Menor reprodução constante ao longo das gerações ir-se-á reflectindo na redução cada vez maior da sua prole, até perder toda a influência na população;
 - Escolha dos mais capazes – o algoritmo procura os pares mais capazes e selecciona-os para recombinar os seus genes nos seus descendentes, garantido uma prole mais adaptada que os seus progenitores. Um dos processos mais populares de fazer esta selecção é o Método da Roleta (roulette wheel);

- Reprodução elitista – para garantir que os melhores indivíduos contribuam sempre com os seus genes para a população, é reservada sempre uma “casta” de privilegiados que projectam sempre os seus genes para as populações vindouras. Esta situação impedirá o desaparecimento de indivíduos de grande valor adaptativo, devido ao processo de mutação ou de cruzamento.

Em qualquer dos casos, podem usar-se várias técnicas de geração de descendência, sendo as mais comuns, o cruzamento e a mutação, referidas anteriormente.

- Convergência – se não ocorrerem os problemas indicados atrás, a Convergência Prematura ou Finalização Lenta, as populações evoluirão até o algoritmo convergir num máximo global.

5.3.2 Modelos de Programação

Os trabalhos desenvolvidos sobre as teorias evolucionistas para a resolução de problemas específicos de alta complexidade, originaram o surgimento de variadas técnicas para a sua implementação em sistemas informáticos. Estas técnicas podem ser agrupadas em seis grandes famílias (Rocha, 2004). Nas secções seguintes serão analisadas estas famílias.

5.3.3 Algoritmos Genéticos

Os Algoritmos Genéticos (AG), fortemente divulgados a partir do trabalho de Holland (1975), traduzem-se num processamento algorítmico das fases referidas em 5.3.1. Este processamento assumiu o nome de Algoritmo Genético de Holland e pode ser visualizado no organigrama da figura 5.8:

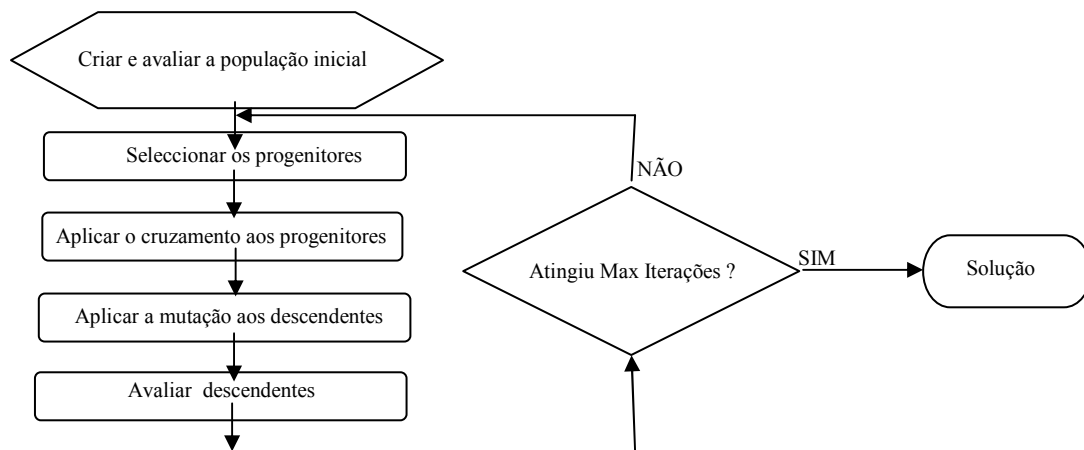


Fig.5.8: Algoritmo Genético de Holland

O organigrama refere 5 situações de processamento e uma condição de saída baseada na contagem de iterações;

Criar e avaliar a população inicial. É típico que esta população seja criada de uma forma aleatória, representando as possíveis soluções para a resolução do problema em causa. Cada elemento dessa população é avaliado mediante a função de avaliação estipulada, sendo-lhe atribuído um valor de aptidão.

Seleccionar os progenitores. Esta operação inicia o ciclo repetitivo do algoritmo. Consiste em decidir quais e quantos são os elementos da população que irão dar origem a novos elementos. A quantidade escolhida depende da parametrização do algoritmo. A decisão que selecciona os elementos progenitores depende da estratégia assumida: podem ser escolhidos aleatoriamente, os mais capazes, alguns mais capazes juntamente com outros aleatórios, os elementos de uma elite já reservada e os mais capazes, etc.

Aplicar o operador de cruzamento aos progenitores. O operador de cruzamento é inspirado na reprodução de seres sexuais. Consiste em definir os genes de um descendente como uma combinação dos genes dos progenitores, com o fim de passar às gerações seguintes a informação refinada das gerações anteriores. Os operadores de cruzamento foram já referidos na secção 5.3.1.

Aplicar mutação aos descendentes. No algoritmo de Holland é escolhida a mutação binária, mas qualquer outra mutação pode ser utilizada, dependendo do tipo de representação dos genes, assim como do algoritmo em causa.

Avaliar Descendentes. Esta operação utiliza o mesmo método de avaliação escolhido para a população inicial, se for escolhido um método de avaliação fixo. Poder-se-ia pensar que, em circunstâncias em que as condições externas sejam dinâmicas, os

factores de avaliação deverão também ser dinâmicos reflectindo essas mudanças. Então, a função de avaliação deveria ser reajustável. Fazemos um paralelismo com a vida real; durante um período de aquecimento global, a função de avaliação deverá escolher os indivíduos com características de maior adaptação a temperaturas mais elevadas. No entanto, se o exterior se alterar e passar a evoluir no sentido de uma era de glaciação, então a função de avaliação deverá seleccionar os indivíduos com características de adaptação a valores de temperatura baixas, para garantir a continuidade da espécie.

Seleccionar a população seguinte. Nesta operação, serão excluídos os elementos que verificaram certas condições de incapacidade (idade, inadaptabilidade, morte, ..) e serão introduzidos novos elementos a partir das operações definidas no parágrafo anterior. De acordo com várias estratégias, o algoritmo pode decidir forçar a manutenção de uma população estabilizada ou deixar que a selecção decorra sem restrições externas.

Número de iterações atingidas. Neste algoritmo, o fim do processamento depende de uma definição prévia de um número de iterações. No entanto, podem ainda surgir outras variantes do algoritmo de Holland que alarga esta condição de fim de iterações, para a verificação de **convergência prematura** ou de **finalização lenta** (ver definições na secção 5.3.1).

5.3.4. Programação Evolucionária

A Programação Evolucionária (PE) obedece aos mesmos princípios que os AG's, e surgiu como consequência do trabalho de L. Fogel, A. Owens e M. Walsh (1966). Nesta técnica, existe uma simplificação de procedimentos deixando a evolução apenas sob a responsabilidade do operador de mutação. Os operadores de cruzamento não são considerados relevantes para o processo.

Neste paradigma, os indivíduos são independentes, pelo que a sua adaptabilidade ou não ao sistema depende apenas do efeito do operador de mutação sobre a característica do respectivo elemento, melhorando ou prejudicando a adaptabilidade. Assim, os indivíduos são submetidos a mutações constantes que resultam numa alteração da solução que o indivíduo representa. Normalmente, estas mutações seguem uma distribuição estatística de acordo com a distância da solução ao óptimo global. Distâncias maiores implicam a aplicação de mutações com maior peso na solução, e distâncias menores implicam mutações com um menor peso na solução.

5.3.5. Estratégias Evolutivas

Estes algoritmos foram desenvolvidos a partir dos trabalhos de I. Rechemberg (1973), que se baseou em trabalhos da década de 60 que exploravam a resolução de problemas de optimização paramétrica, e que se alargaram para tratamento de problemas discretos. Neste método, na primeira versão, o algoritmo começa com um indivíduo, que representa um candidato à solução. Durante a evolução, é criado um novo indivíduo em cada ciclo, que é uma mutação do indivíduo original. A representação dos genes baseia-se em números reais, com a consequente mutação (ver definições na secção 5.3.1.).

Nestas EE, um indivíduo é representado por um par de vectores reais $(x, \bar{\sigma})$ em que x representa um ponto de busca e $\bar{\sigma}$ representa o vector do desvio padrão associado.

Algoritmos mais recentes aumentam as capacidades do algoritmo estabelecendo vários progenitores em vez de um, que geram um descendente; neste caso poderá o algoritmo recorrer ao operador de *crossover* para recombinar sub-conjuntos de genes.

Mais tarde ainda, surgiram variantes em que vários progenitores podem gerar vários indivíduos, que entrarão para a população. Uma destas variantes, muito popular, conhecida por EE- $(\mu+\lambda)$ utiliza a selecção elitista em que μ pais geram λ filhos. Desta população são escolhidos os μ melhores indivíduos para transitar para a geração seguinte.

Esta estratégia EE- $(\mu+\lambda)$, deu origem a outra chamada EE- $(\mu+\lambda)$ *modificada*. Nesta, introduz-se um parâmetro β que representa os indivíduos da população com melhor valor de adaptação. Estes indivíduos transitam para a população dos descendentes competindo com eles. Procura-se assim evitar a estagnação em óptimos locais assim como a perda de indivíduos com boas características de adaptação.

5.3.6. Programação Genética

A programação genética foi introduzida a partir da investigação de John Koza (1990). Estas técnicas evolucionárias centram-se na criação e evolução automática de programas que procuram resolver um determinado problema.

É muito usada em problemas que trabalham com grandes quantidades de informação (investigação do genoma, pesquisa de informação na Internet, mercados financeiros). Normalmente, os indivíduos são constituídos por instruções de uma determinada linguagem de programação (Koza usou o LISP) armazenados em árvores sintáticas. O algoritmo funciona aplicando os operadores de cruzamento a ramos dessas árvores, esperando que esse processo evolua para uma árvore sintática que represente o melhor programa para a resolução do problema em causa.

Por exemplo, no trabalho de Koza, consideremos os programas em sintaxe LISP dados por:

(OR (NOT D1) (AND D0 D1))
 e
 (OR (OR D1 (NOT D0)) (AND (NOT D0) (NOT D1)))

Estes programas têm uma representação em árvore, chamadas árvores progenitoras, como está representado na figura 5.9:

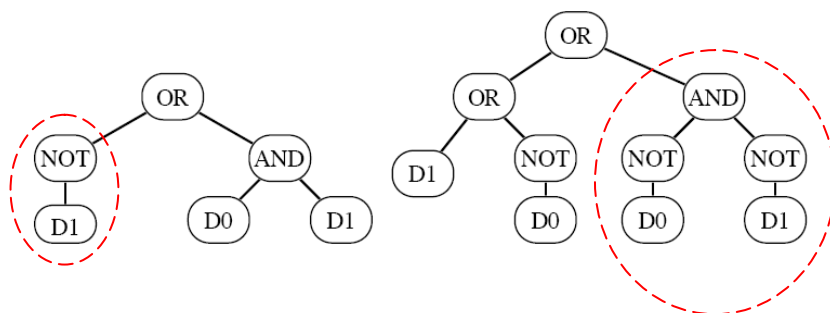


Fig.5.9: Exemplo de Koza para demonstrar a programação genética

Os ramos da árvore seleccionados para cruzamento estão referenciados pelos círculos na figura 5.9, e são os visualizados na figura 5.10:

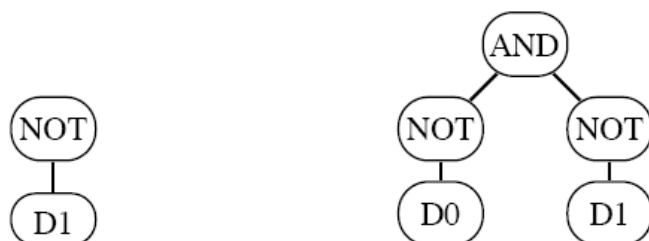


Fig.5.10: Exemplo de Koza - ramos escolhidos para cruzamento

A substituição destes ramos nas árvores anteriores produz uma nova recombinação de código, como se pode ver na figura 5.11:

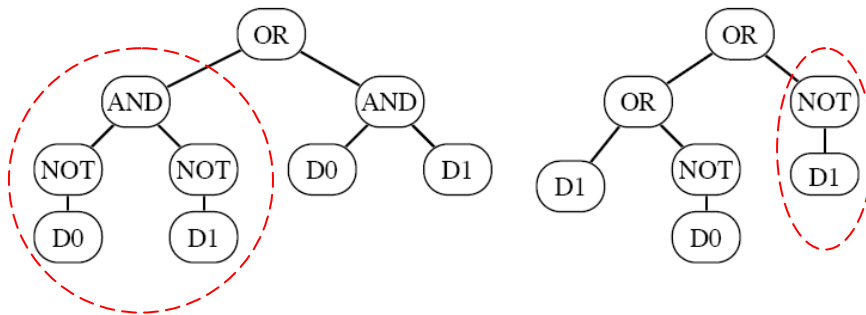


Fig.5.11: Exemplo de Koza - descendentes

Nestes algoritmos não é aplicada, normalmente, a operação de mutação.

Os trabalhos de W. Banzhaf, P. Nordin e R. Keller, (1998) deram um novo impulso para que esta técnica se tornasse muito popular na comunidade científica.

5.3.7. Algoritmos Evolucionários

Os Algoritmos Evolucionários (AE's) são uma forma de unificação e generalização dos modelos anteriores (Rocha, 2004). Eles são representados pelo fluxograma da figura 5.12:

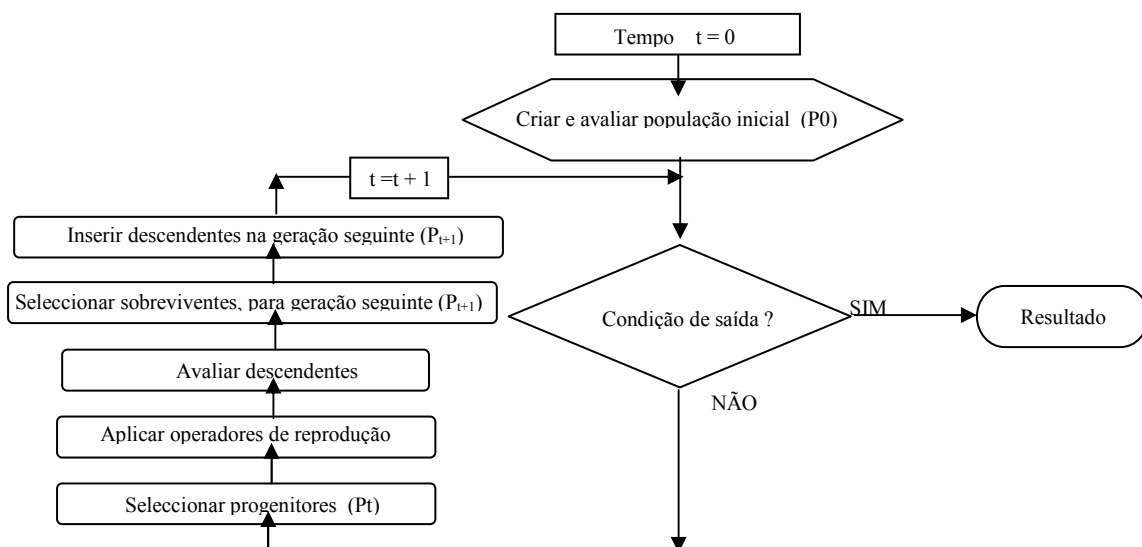


Fig.5.12: Fluxograma típico de um AE

Relativamente aos algoritmos anteriores, este novo paradigma apresenta algumas diferenças com cariz mais aberto e integrador:

- População inicial - permite criar as populações iniciais baseados em certos critérios específicos do problema, sem ser apenas o aleatório.
- Condição de saída - o ciclo acaba desde que se verifique uma condição, ou condições pré estabelecidas, sem ser apenas um contagem rígida dos tempos.
- Forma de representação das soluções – deixa de ser apenas possível a representação binária e passam a ser consideradas outros tipos de representação. Um exemplo é a representação real para variáveis ou parâmetros de domínio contínuo, com os prós e contras dessa representação (Rocha, 2004). O algoritmo permite ainda outras possibilidades de representação que se aproximam das variáveis de decisão do problema.
- Processo de renovação da população – a taxa de substituição de 100%, usada por Holland para substituir a população em cada geração, passa a ter um valor variável que se adapta às características de cada problema. Incorporará, assim, a situação de taxa mínima, de apenas um par substituído, proposta por D. Whitley (1989), a taxa máxima de 100% de Holland e soluções intermédias que permitam uma competição entre progenitores e descendentes, propostas por outros estudiosos.
- Processo de selecção – o processo de selecção é aberto permitindo a selecção baseada no escalonamento dos valores de aptidão de D.E. Goldberg (1989), nas técnicas de amostragem determinísticas ou nas técnicas de amostragem estocástica (*roulette wheel*) (Baker, 1987).
- Operadores de reprodução – permite a criação de uma tabela de operadores, em vez dos clássicos operadores de *cruzamento* e de *mutação*. Esta tabela incorpora diversos operadores de reprodução que são definidos de acordo com as características do problema. É utilizado o processo *roulette wheel* para seleccionar os operadores da respectiva tabela com a finalidade de os aplicar a uma lista seleccionada de progenitores, em cada geração.

5.3.8. Algoritmos Evolucionários Híbridos

Os AE's demonstram alguns problemas em aplicações com especificidades próprias em que um algoritmo mais personalizado poderia obter um melhor resultado e desempenho (Rocha, 2004). Assim, surgiram os AE's Híbridos que incorporam na estrutura dos AE's as especificidades do problema com o fim de criar "atalhos" de execução e cálculo. De entre as mais conhecidas destacam-se os métodos de otimização local, inspirando-se nas ideias da Evolução Lamarkiana, em que utilizam operadores de reprodução que têm em conta também informações da vizinhança da solução, e não só dos progenitores. No capítulo seguinte, debruçar-me-ei sobre esta Evolução Lamarkiana. Esta otimização local pode levar a modelos em que os indivíduos alterem o seu valor de aptidão durante o seu ciclo de vida, baseados em informações na vizinhança dos máximos locais, originando uma família de algoritmos chamados Algoritmos Meméticos (AM) (Moscato & Norman, 1989).

Outro grupo conhecido deste tipo de modelos, baseia-se na recombinação de duas ou mais soluções com o fim de retirar as melhores características para definir os operadores de cruzamento. Temos, neste caso, os cruzamentos miópicos (Grenfenstette et al, 1985), e os cruzamentos de Falkenauer (1998).

5.4. Modelos Lamarkianos

No entusiasmo decorrente das novas áreas de investigação abertas pela teoria da evolução de Darwin, evidenciou-se uma teoria que despertou muito interesse nos meios científicos. Lamark surgiu a defender que as capacidades adquiridas pela aprendizagem, durante o período de vida de um indivíduo, eram passadas directamente aos seus descendentes, através da uma rectificação genética. O conhecimento adquirido ocasionaria uma adaptação do indivíduo forçando alterações genéticas, que se propagariam aos seus descendentes, garantindo, assim, directamente, a passagem das bases desse conhecimento. Esta situação permitiria uma interpretação da selecção natural que se manifestaria através de mudanças bruscas ocasionais, em contraponto com o gradualismo lento, mas inexorável, defendido pelos Darwinianos.

Este paradigma inspirou muitos modelos científicos, e é muito utilizado em sistemas em que se necessita de uma convergência rápida, na procura de soluções, através de

técnicas evolutivas. Miguel Rocha (Rocha, 2002) desenvolveu estudos comparativos deste paradigma com outros modelos, onde confirmou a sua grande rapidez de convergência, o que o torna um bom candidato para a resolução de problemas que exijam essas características.

Tecnicamente, na aplicação deste paradigma a modelos utilizando AE's, incorpora-se no código genético de cada indivíduo (que codifica uma solução do problema) os factores de adaptação adquiridos pela experiência vivencial. Em cada ciclo evolutivo do sistema, o código genético será alterado, o que será uma garantia de que essas alterações serão passadas para os descendentes. A convergência será bastante mais rápida, e os factores aleatórios de mutação dificultarão que o sistema fique preso em mínimos locais.

Em 1893, Augusto Weismann defendeu que os organismos complexos têm apenas dois tipos de células: células que passam a informação genética (*germ cells*) associadas à reprodução, e células somáticas que não interferem no processo reprodutivo. Weismann provou que não existe qualquer forma de a informação adquirida pelas células somáticas poderem passar para as *germ cells*. Esta teoria praticamente pôs de lado a tese de Lamarck, mas isso não impediu que o modelo de Lamarck continue a ser uma poderosa fonte de inspiração para modelos informáticos.

5.3.4. Modelos Baldwinianos

Tendo a teoria de Lamarck sido fortemente contestada e posta em causa pelos estudos de Weismann, e parecendo certo que, de alguma forma, a aprendizagem tinha um efeito positivo na concorrência evolutiva, James Mark Baldwin (1896) propôs uma teoria mais consensual. Defendeu que a capacidade de aprendizagem de cada indivíduo pode condicionar a direcção do processo evolutivo. A evolução deixa de estar apenas dependente da selecção das características genéticas mais adaptadas, mas permite que a capacidade de aprendizagem de cada indivíduo reforce a sua vantagem evolutiva. Este factor externo de vantagem vai acabar, com a passagem de várias gerações, por se incorporar no comportamento instintivo dos descendentes, transformando esse processo de aprendizagem numa habilidade inata e instintiva, que passa a dispensar a aprendizagem, e garante vantagens muito importantes no decorrer de processos de selecção. Este fenómeno ficou conhecido por *efeito de Baldwin*.

Em termos de computação genética, o efeito de Baldwin é muito interessante, uma vez que permite passar alguma informação implícita de conhecimento de uma geração para a outra sem que isso implique uma alteração dos genes. Esperar-se-á que esta passagem de informação permita que os descendentes sejam mais capazes ainda, o que levaria a uma mais rápida convergência. Assim é, com efeito, como mostram os resultados obtidos no trabalho de Miguel Rocha na sua investigação (2002), onde demonstra grandes vantagens deste tipo de modelos, que garantem um equilíbrio muito interessante entre o aumento da velocidade de convergência e a capacidade para evitar os mínimos locais.

O efeito de Baldwin tem uma característica importante que é transposta para os sistemas informáticos: ele processa-se em duas fases (Turney et al, 1996). Na primeira fase, a aprendizagem interferirá na plasticidade do fenótipo garantindo uma mutação parcial que será utilizada para o sucesso adaptativo do indivíduo. Numa segunda fase, com o desenvolver do processo evolutivo, se o ambiente se mantiver estável, devido aos custos de aprendizagem, a natureza acabará por substituir esse conhecimento incorporado na plasticidade do agente, num mecanismo rígido equivalente através de uma assimilação genética: o instinto. Em ciências informáticas, este efeito é utilizado para resolver problemas com soluções representadas por populações. Na primeira fase, o sistema evolui rapidamente para os mínimos locais (adaptação da plasticidade do fenótipo). Na segunda fase, o sistema já tem os melhores génotipos e gera-se então um processo de selecção.

De notar que surgirão posteriormente, situações em que os indivíduos com grande capacidade de aprendizagem (desenvolvidos na primeira fase) entrarão em concorrência com indivíduos com grande capacidade instintiva (desenvolvidos numa segunda fase). Seria interessante modelar e estudar os resultados da competição de grupos de agentes com essas duas características, mas isso não faz parte do âmbito deste trabalho.

Capítulo 6

Plataformas de Vida Artificial

Neste capítulo, serão analisadas as ferramentas que servem de suporte ao estudo dos modelos de VA. Normalmente, estas ferramentas existem concentradas em plataformas que oferecem uma interface visual bastante prática e atractiva. Este visual tenta sempre permitir uma grande capacidade de parametrização dos componentes da plataforma. Dependendo do tema que se está a modelar, as plataformas podem mostrar desde simples símbolos representativos até imagens muito sofisticadas e detalhadas.

Depois de expor algumas definições, que iremos assumir dentro do âmbito destas plataformas na secção 6.1, iremos referenciar alguns exemplos na secção 6.2. Finalmente procuraremos detalhar a plataforma *getAlife*, na secção 6.3. Esta última plataforma será a que adoptaremos para o desenvolvimento deste trabalho, já que apresenta inúmeras vantagens que serão descritas ao longo deste capítulo.

6.1 Definições

Os programas de software que modelam sistemas de vida artificial, conhecidos nos meios científicos por plataformas *ALife*, são uma ferramenta muito importante para testar situações em que o objecto de análise seja complexo, ou fortemente imprevisível, ou ainda situações de análise de sistemas que dependem do comportamento de agentes. Estes sistemas têm uma grande capacidade de parametrização, de onde se podem tirar informações científicas muito construtivas. Este tipo de modelos veicula três propriedades características (Langton, 1989):

- Análise Bottom-Up - a representação do conhecimento é feita através de regras locais em vez de regras globais. Os processos de regulação devem ser orientados a partir do interior do próprio sistema;

- Emergência - as decisões e comportamentos das criaturas (agentes) devem emergir da interacção destes e não de imposições exteriores. O mesmo se verifica se o sistema tiver que tratar com um sistema de comunicação;
- Não-Objectividade - o objectivo fulcral para o desenvolvimento de um sistema *ALife* passa pela evolução estrutural do próprio sistema, em detrimento da resolução específica de um determinado problema. Neste tipo de sistemas *behavioristas* as entradas são sempre dependentes das saídas.

Assumiremos que as plataformas *ALife* apenas se reportam à evolução de entidades, individuais ou em grupo, com capacidade de adaptação e aprendizagem, mais ou menos desenvolvida. Poderão ser também consideradas as plataformas que analisam sistemas vegetativos, em que não existe uma aprendizagem no sentido lato do termo, como é o caso da plataforma *Magic Garden* ou *Nerve Garden*. Neste caso, dá-se uma especial importância aos processos evolutivos e de desenvolvimento físico de uma entidade mediante os vários factores que o envolvem no exterior. Podem ainda ser plataformas de grande capacidade, como é exemplo o programa desenvolvido com a supervisão de Klaus Schulten, na Universidade de Illinois, que permite o detalhe átomo a átomo da simulação do comportamento de uma estrutura de vida completa (neste caso um vírus)² (Fig.6.1).

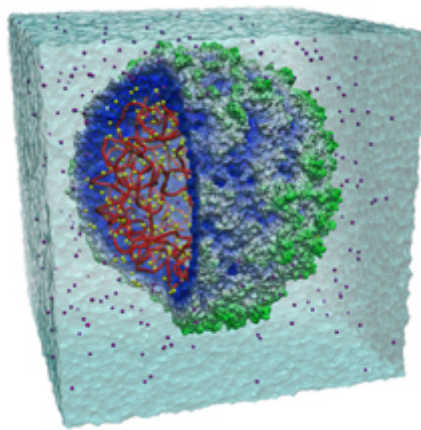


Fig.6.1: Modelação do comportamento de um vírus, por K.Schulten

² <http://www.ks.uiuc.edu/Publications/Brochures/symposium/index.html>

As plataformas *ALife* podem ser divididas em três grandes grupos:

- Plataformas *ALife* de evolução individual;
- Plataformas *ALife* de coordenação ou concorrência de grupo;
- Plataformas *ALife* do tipo presa-predador.

No primeiro grupo, temos a modelação da evolução de um indivíduo, onde tem especial interesse estudar a capacidade evolutiva de organismos individuais, focando essencialmente dois aspectos: o desenvolvimento físico e a evolução de comportamento adaptativo. As mais comuns são as que procuram analisar as leis do movimento; como é que um agente aprende a andar, a nadar, a saltar, etc. Nesta área destacam-se as plataformas que utilizam agentes cujo corpo, simplificado, é constituído por *cubóides*, de que são um exemplo as criaturas de Karl Sims (1994). Estes modelos têm tido um enorme sucesso devido à sua potencialidade na visualização das tentativas de evolução do agente. Para além do grande interesse académico, têm também interesse lúdico, surgindo no mercado com características comerciais devido à curiosidade que tem despertado em outros sectores da população.

No segundo grupo, incorporam-se as plataformas mais ligadas aos processos evolutivos com características de comportamento colectivo. É neste grupo que se enquadram as plataformas de análise da emergência da fala ou da organização. Normalmente, são plataformas que modelam relações de sistemas sociais, ou de aprendizagem cultural, isto é, por informação de outros elementos. Exemplo deste tipo é o programa *Creatures*³ que evolui agentes que, para além de aprenderem a interagir, aprendem a falar, através da introdução de texto pelo utilizador utilizando o teclado, pela utilização, pelo agente, de uma máquina de comunicação existente no próprio ambiente da plataforma, ou ainda pelo contacto com outros agentes. Inserem-se também neste grupo as plataformas do tipo *boids*⁴ que simulam o comportamento de enxames, cardumes, ou grupos de pássaros para estudar as leis da sua movimentação relativa.

No terceiro grupo, é dada uma maior ênfase à concorrência e selecção natural, onde predominam os algoritmos evolucionários e genéticos. Poderá abarcar também as capacidades de comunicação como uma habilidade adquirida, com o fim de obter

³ http://www.gamewaredevelopment.co.uk/creatures_index.php

⁴ <http://www.red3d.com/cwr/boids/>

vantagens na selecção natural. Normalmente, são sistemas que agregam várias espécies em competição e em que o desenvolvimento de umas se faz à custa da predação de outras. Pode ainda conter elementos passivos que servirão de obstáculos para filtrar, ainda mais, os elementos mais capazes, e introduzir dificuldades inesperadas que os agentes deverão aprender a contornar. Embora existam versões comerciais, este grupo de plataformas está mais disseminado em ambientes académicos para modelação de muitos sistemas, estando normalmente disponível à comunidade científica.

6.2 Exemplos Ilustrativos

Sendo esta área de investigação relativamente recente, verifica-se uma produção de qualidade ainda relativamente pobre, embora cada vez mais elevada nos últimos tempos o que é bastante elucidativo do interesse que começou a despertar. Vejamos alguns exemplos demonstrativos dos tipos de plataformas referidos:

Ero⁵

Plataforma *ALife* do grupo individual, com elemento cubóide. O objectivo é analisar a evolução de articulação dos componentes simples do corpo com o fim de obter movimentos correctos de equilíbrio. O objecto procura aprender a articulação correcta dos seus componentes através de uma aprendizagem por tentativa e erro, até aprender a movimentar-se. Neste programa, existem testes para andar, saltar, nadar, etc. (Fig. 6.2).

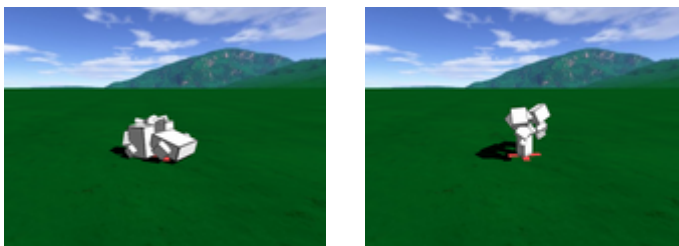


Fig.6.2: Criatura Ero a aprender a andar e em fase adiantada

⁵ <http://www.frams.alife.pl/common/>

EvoRunners⁶

Esta plataforma, pertence ao grupo de plataformas de concorrência de grupo, embora os agentes sejam inspirados nos cubóides. Ela cria agentes que têm que aprender a correr, e os que melhor o conseguirem ganham vantagem competitiva relativamente aos outros, permitindo passar para os descendentes esta capacidade, o que irá refinar o processo evolutivo ao longo das gerações (Fig.6.3). Esta plataforma utiliza também mutações.

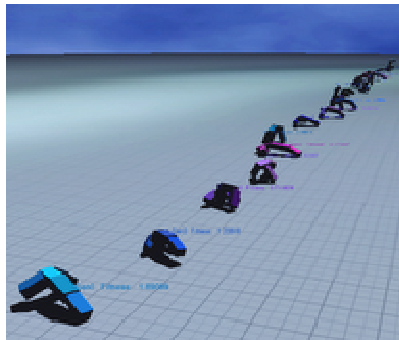


Fig.6.3: Criaturas EvoRunners, em concorrência

Symbolic Creatures Simulation, SCS⁷

Esta plataforma pertence ao terceiro grupo de plataformas, definida acima como plataformas do tipo presa-predador. Um dos principais objectivos é o estudo sobre a emergência das linguagens. Caracteriza-se por simular a interacção entre criaturas através de um mundo artificial do género presa-predador (Fig.6.4). Os predadores dividem-se em três tipos: terrestres, aéreos e rastejantes. As presas podem ser do tipo: instrutores, aprendizes e auto organizadores (Loula et al, 2004).

Neste modelo, os agentes têm capacidade de visão, audição e locomoção. As suas acções baseiam-se no modelo de comportamento de Brooks (1990). Um desses comportamentos, nas presas, é a capacidade de produzir e aprender relações simbólicas.

⁶ <http://www.frams.alife.pl/common/>

⁷ <http://www.dca.fee.unicamp.br/projects/artcog/symbcreatures/>

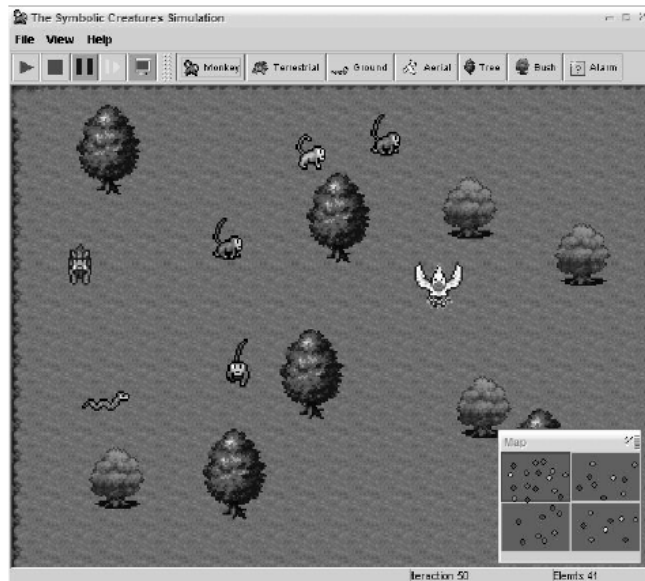


Fig.6.4: *Screenshot* da plataforma SCS

A integração cognitiva baseia-se na neuroetologia, e o processo de reforço da aprendizagem segue o modelo hebbiano (Hebb, 1949).

Biots⁸

Esta versão livre também pertence ao grupo de plataformas do tipo presa-predador. Tem um excelente visual e é muito educativo. É muito completo em termos de componentes, contendo presas, predadores, alimentos vegetais (frutas), reprodução sexuada e assexuada, emoções nos agentes (e.g. medo), alterações do exterior simulando as quatro estações do ano, diferenciação de comportamento etário, etc, (Fig.6.5).



Fig.6.5: Ambiente da Plataforma Biots

⁸ <http://www.scarybug.org/biots/>

6.3 Plataforma getAlife

Esta é a plataforma que iremos adoptar para o nosso trabalho. Pertence ao terceiro grupo de plataformas *ALife*, referenciada como do tipo presa-predador. Foi desenvolvida na Universidade do Minho, sob a supervisão de Miguel Rocha com uma colaboração muito destacada de Daniel Machado (Rocha e Machado, 2008).

A plataforma *getAlife* permite a escolha de diversificadas configurações, métodos e processos de análise. Tendo sido desenvolvida em linguagem Java, este simulador permite ainda uma capacidade multi-plataforma muito útil.

Esta plataforma é uma ferramenta *open-source* que permite, entre outras coisas, testar o comportamento e evolução de sistemas multi-agente. Permite, assim, que os próprios utilizadores possam re-programar e alargar as potencialidades do sistema, partilhando recursos cada vez mais desenvolvidos e completos.

No âmbito da ferramenta existem seres vivos, artificiais, distribuídos por espécies em concorrência, bem caracterizados por um genoma que transmitirão aos seus descendentes, e cujos comportamentos serão definidos por um módulo de decisão próprio de cada indivíduo. O módulo de decisão baseia-se na informação que cada ser recebe do ambiente, através dos seus sensores.

Existe um parâmetro na iniciação do módulo de decisão, que pode assumir quatro possibilidades: *stop*, *random*, *defense* e *attack*. Este parâmetro vai definir a intencionalidade de cada espécie.

A plataforma *getAlife* disponibiliza, entre outras, a aprendizagem por reforço, essencial para estratégias de processamentos de emergências. Como permite uma constituição de agrupamentos por espécies, ela garante a definição de comportamentos ou restrições gerais de espécie, tal como velocidade máxima, espécies de que se pode alimentar, raio de visão, distância máxima de caça, etc,

Os autores deste programa têm tido o particular cuidado de diversificar o seu objectivo de utilidade, procurando torná-lo atractivo para a utilização como ferramenta de ensino, e possível utilização para a incorporação em jogos onde seja importante a criação de situações de emergência de comportamento, decisão ou comunicação, o que poderá despertar o interesse e curiosidade em alunos de níveis básico ou secundário, pela facilidade com que expõem conceitos como complexidade, emergência, Inteligência Artificial, RNA's, etc.

É neste contexto que foi criada a plataforma *Species* (Fig.6.6), desenvolvida por estudantes da Universidade do Minho, a partir da plataforma *getAlife*, e que pretende ser um jogo educativo.



Fig.6.6: Ecrã da plataforma Species

Em termos de implementação, a plataforma *getAlife*, é constituída por quatro blocos principais;

- *Packages* de Redes Neurais e Algoritmos Evolucionários (*ann, gann*);
- *Graphic User Interface* (GUI), com o aspecto visual do mundo e seus componentes, assim como a gestão desse mundo (*gui, world, species*);
- *Packages* que incorporam os algoritmos de interface dos elementos do mundo com os *packages* RNA e AE. (*genome, learning*);
- *Package* de análise e visualização gráfica e estatística (*graphPanels, statistics*).

Este trabalho vai adicionar um *package, communication*, que deverá ser o embrião onde serão implementadas as classes necessárias ao desenvolvimento dos aspectos culturais da evolução, como a emergência de linguagens e mais tarde processos mais evoluídos de comunicação, organização por comunicação, negociação comunicativa, etc.

A ferramenta *getAlife*, na sua génese, tem uma interface muito fácil de usar. O seu ecrã principal está dividido em 5 áreas (Fig.6.7).

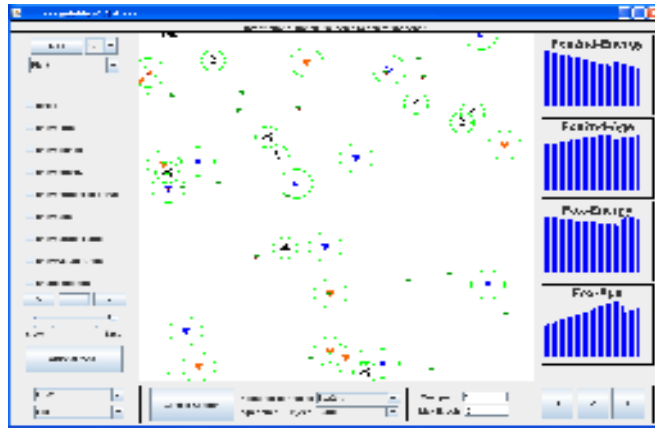


Fig.6.7. Screenshot do ecrã principal do *getAlife*

Na área superior, permite uma linha de avisos de erro, ou informações de controlo avulso.

Na área da esquerda, está implementado o *frame* de parametrização da aplicação. Neste *screenshot* já está inserida a opção de utilização ou não dos procedimentos de comunicação. Este *frame* permite a escolha das espécies que evolverão no mundo a testar, a sua quantidade e as características que deverão ser visualizadas no ecrã, nomeadamente, o raio de visão, o raio de ataque, o número de presas, a idade, etc. Este *frame* pode ainda seleccionar a velocidade de visualização, a paragem momentânea, ou o recomeço do sistema (apocalipse).

Na área central visualiza-se a evolução das espécies no seu mundo. Este pode ser limitado ou toroidal. Visualiza-se também um contador discreto do tempo.

Na área da direita, temos a visualização *on-line*, da evolução gráfica das características das espécies, como a sua energia média ou número médio de presas. Podem ser seleccionadas duas espécies, onde se poderão comparar as suas performances em concorrência (presa vs predador ou predador1 vs predador2).

Na área inferior serão seleccionadas as espécies que se pretendem analisar, as características que se pretendem ver nos gráficos *on-line*, ou a selecção de outros menus, como o menu de visualização gráfica de cada espécie (Fig.6.8), ou o da visualização gráfica comparativa de duas espécies (Fig.6.9).

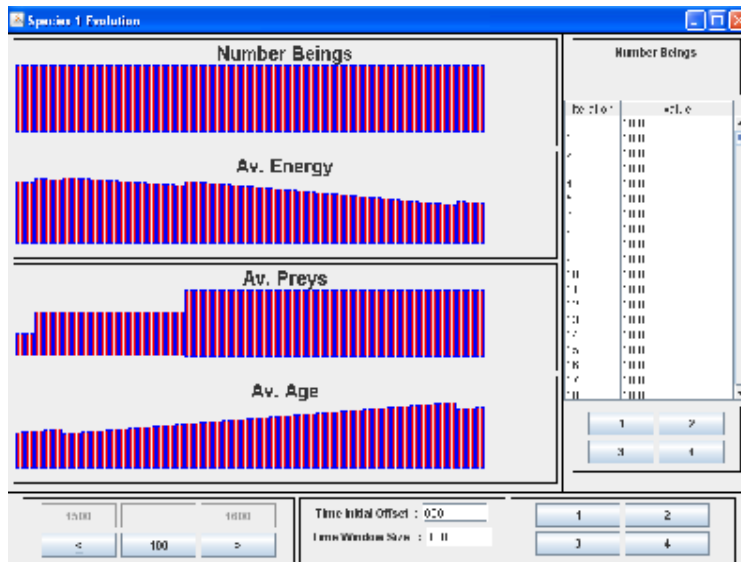


Fig.6.8: Evolução de uma espécie

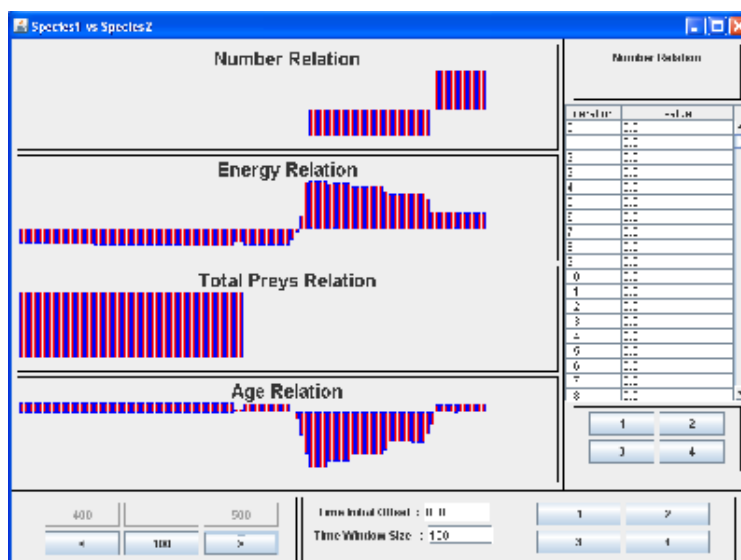


Fig.6.9: Comparação de duas espécies

Pode ser seleccionado, ainda, um gráfico de visualização da evolução dos símbolos de comunicação, que dá informações completas sobre a evolução do número de agentes comunicadores, o número de símbolos diferentes que referenciam o mesmo conceito, e a evolução da dispersão desses símbolos diferentes, medidos em termos de Desvio Padrão e Coeficiente de Pearson. Estes ecrãs de controlo de visualização de comunicação foram também inseridos na execução deste trabalho, para poder obter resultados de análise de uma forma directa e simples.

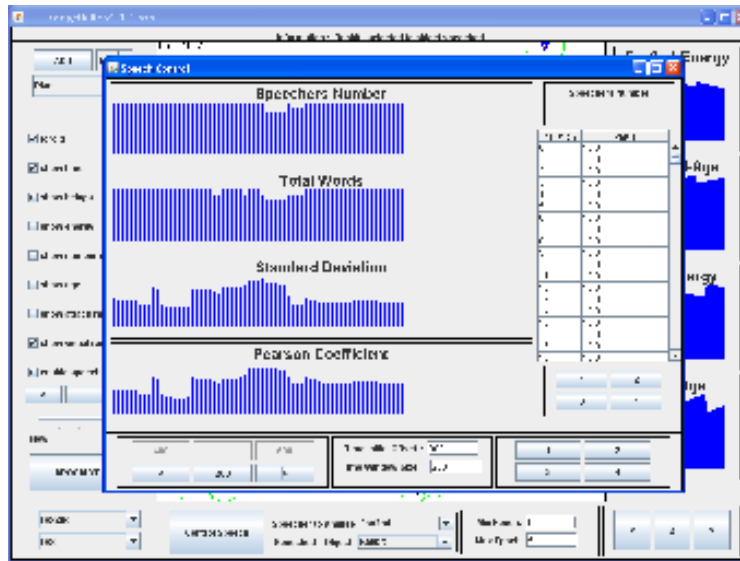


Fig.6.10: Screenshot de “Control Speech” (menu 2).

O menu inferior, com este sistema extremamente simples, permite ir incrementando constantemente novos menus para novas funcionalidades, como foi exemplo a introdução deste menu de controlo de comunicação; dá ao *getAlife* uma grande maleabilidade e abertura. Estes novos menus permitem também abrir novas zonas para parametrização dos algoritmos dando maior interactividade e flexibilidade ao programa.

Como base de trabalho, a plataforma disponibiliza dois tipos de agentes:

- Agentes passivos – elementos estáticos que servem de alimento para outros elementos. O elemento disponível é a espécie “Plant”;
- Agentes activos – elementos dinâmicos que evoluem segundo regras de subsistência definidas. Estes elementos agrupam-se em três tipos:
 - Presas – espécies “Bug”, “Mouse” e “Rabbit”;
 - Presas-Predadores – espécie “Snake”;
 - Predadores – espécies “Cat” e “Fox”.

As presas activas, no entanto, poderão ter uma intenção do tipo predador perante os elementos passivos, se estes forem definidos como seus alimentos possíveis. Por exemplo, a presa activa “Rabbit” pode ter acções do tipo predador sobre os elementos passivos “Plant”.

O nosso trabalho introduziu duas espécies novas: uma espécie predadora, “Fox2nd”, e uma espécie do tipo presa, “Rabbit2nd”. O sufixo “2nd” significa que esta espécie herda todas as características de uma espécie já existente, acrescentando uma nova

característica evolutiva que representará uma segunda geração dessa espécie. Mais à frente, neste texto, serão explicadas mais em pormenor estas novas espécies.

Capítulo 7

Emergência de Linguagens via Computação Evolucionária

Neste capítulo, iremos explicar e justificar o processo de emergência das linguagens utilizando como técnica a Computação Evolucionária.

Serão explicadas as escolhas dos métodos, processos e técnicas de entre as descritas nos capítulos anteriores assim como a estratégia usada para, a partir delas, se obter uma emergência de linguagem. O objectivo desta emergência será, como ficou claro no início do trabalho, a obtenção de agentes mais capazes de responder melhor a problemas. A estratégia será, pois, orientada nesse sentido, e, devido a isso, as decisões de parâmetros e sub-técnicas foram consideradas depois de variados testes práticos, com o fim de obter os melhores resultados.

7.1. Justificação da estratégia

Como já foi referido anteriormente, é prioritário, neste trabalho, dar uma autonomia de aprendizagem aos agentes com o objectivo de obter a máxima generalidade. Quer isto dizer que, a forma como os agentes irão procurar as soluções será independente do tipo de problemas, assim como do modo como estes se apresentarem dimensionados. Pretende-se também, com esta estratégia, garantir que os agentes se adaptem a quaisquer alterações das condições, que possam ocorrer dentro do sistema em evolução, sem necessidade de ajustes ou reactualizações por meio de intervenção exterior.

Assim, foi sempre tida em consideração a não interferência específica no processo, embora com a consciência da perda de eficácia em termos de resultados. Como consequência, preferimos não artificializar, nestas análises, o processo de decisão baseado na aprendizagem. Este processo será deixado para o próprio agente.

No sistema que pretendemos criar, pelo facto de alguns agentes passarem a ter mais um sensor, a “audição de sons”, será esperado um aumento de capacidade desses agentes de perceber o que se passa na sua vizinhança.

As novas classes resultantes destas alterações não modificarão o módulo de decisão já existente, embora este passe a contar com mais um factor de informação.

Na nossa plataforma existia um sistema onde, em cada interacção, cada agente verificava a sua vizinhança através de uma verificação “visual”, e construía um vector dos elementos na sua vizinhança, com as respectivas propriedades, que servia de base à sua tomada de decisão. Com a incorporação do processo de comunicação, o que passa a suceder é que, depois dessa lista ser criada, e antes do processo de decisão, essa lista é complementada com as informações provenientes da verificação “auditiva”.

Este processo de comunicação é considerado “**comunicação directa**”. Caracteriza-se por uma passagem directa da informação obtida pelos sensores de um agente para outro agente (Tan, 1993). Segundo este autor, a comunicação poderia ainda consistir da passagem de estados, ou experiências passadas, para outros agentes. O âmbito deste trabalho é apenas a análise da emergência e por isso optaremos por esta comunicação directa simples.

Será de esperar, assim, que as capacidades da espécie sejam melhoradas, uma vez que, se um agente não tiver uma presa no seu raio de visão, poderá ter uma presa referenciada pelo seu sensor de audição, próxima de um seu vizinho informador. O módulo de decisão deste agente, em vez de tomar uma decisão de movimento aleatório devida à lista vazia de presas na vizinhança, poderá agora decidir movimentar-se em direcção ao local da presa informada pelo seu vizinho. Este movimento potenciará um “cerco” à presa em questão beneficiando a espécie comunicadora.

É óbvio que este processo só passará a dar resultados depois da emergência da linguagem entre ambos os agentes. Isto é, só depois de a informação passada entre os agentes ter o mesmo significado para ambos. Para isso é necessário que os sinais transmitidos entre os intervenientes tenham um valor mínimo de semelhança quando pretendem representar o mesmo objecto.

Com efeito, deve notar-se que, neste processo em que os sinais são “inventados” pelos próprios agentes para referirem determinado objecto, esses sinais não precisam de ser exactamente iguais, mas basta que tenham um valor mínimo de semelhança. Posteriormente, explicaremos este aspecto.

7.2. Definição da plataforma

Para a criação do modelo prático de análise foi utilizada a plataforma *getAlife*, já referida na secção 6.3, que garantirá a componente evolucionária dos agentes constituintes do modelo. O processo de aprendizagem, por seu turno, constituirá a componente neuronal do processo, que será desenvolvida com o recurso ao *package ann*, já existente na referida plataforma. Este *package* foi desenvolvido por Paulo Cortez e Miguel Rocha, da Universidade do Minho, e disponibiliza, em Java, as classes e métodos para a geração e controlo genéricos de todos os processos relativos às RNA's.

O nosso trabalho consistiu em gerar novas espécies, descendentes directas das existentes (Fig.7.1), mas com um pequeno salto evolutivo que lhe incorpora uma memória (RNA de aprendizagem cultural), que permite a cada elemento dessa nova espécie, a capacidade de atribuir um signo para cada situação referenciada pelos seus sensores.

Será interessante colocar em concorrência esta nova espécie, com capacidade comunicativa, com espécies, suas ancestrais, sem essa componente (por exemplo, colocar em concorrência as espécies *Fox2nd* e *Fox*, referidas no capítulo anterior), e comparar o desenvolvimento dos correspondentes níveis de sucesso.

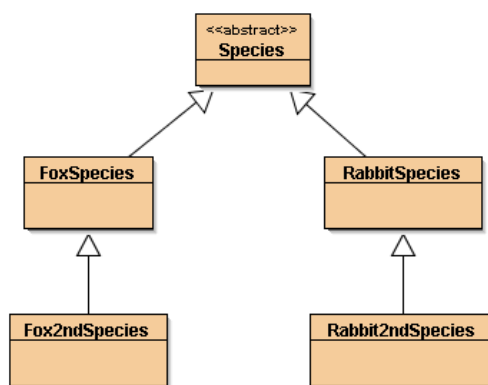


Fig.7.1: Hierarquia dos elementos de segunda geração no *getAlife*.

Para uma maior maleabilidade do programa, a estrutura da RNA de aprendizagem cultural é inserida na definição de cada espécie, o que permite que espécies diferentes possam ter diferentes topologias. Esta opção garantirá uma futura análise comparativa entre essas topologias, confrontando os correspondentes agentes (por exemplo introduzindo uma nova espécie, *human2ndSpecie*, com uma RNA mais complexa).

7.3. Explicação das novas classes

Relativamente às alterações respeitantes à plataforma *getAlife*, com o objectivo de possibilitar a análise dos resultados e os testes, foram introduzidos dois *packages*:

- **GraphPanels**: este é o *package* responsável pela análise e monitorização do mundo do *getAlife*. Foi aumentada a capacidade visual da plataforma, podendo agora o operador ver também a evolução gráfica dos parâmetros, assim como uma tabela com todos os valores históricos dos parâmetros em causa. Estas alterações foram introduzidas nas classes *FrameGraphics* e *WorldAnalysis*. Foi criada uma nova classe neste *package*, *GraphicValues*, que generaliza a utilização do *frame* a qualquer situação, desde que na chamada sejam convertidos os valores que se pretendam visualizar, para um formato previsto e disponibilizado neste tipo. Isto permitiu monitorar através da mesma classe *JFrame*, tanto a evolução da espécie (número de seres, energia média, número médio de presas, ou idade média) como o seu controlo estatístico (número de falantes, variabilidade de signos para o mesmo objecto, desvio padrão e Coeficiente de Pearson);
- **Communication**: este novo *package* foi criado com a intenção de aglomerar todas as classes associadas ao processo de desenvolvimento da comunicação dos seres da plataforma *getAlife*. Neste *package*, para este estudo, foram desenvolvidas as seguintes classes:
 - **Cerebrus** : classe que passa a ser incluída nos agentes de segunda geração, <nome_da_especie>2ndSpecies. Nesta classe existe uma rede neuronal, inicializada aleatoriamente, que responde a estímulos exteriores originados pelos sensores, com a geração de um som, e, ao longo da vida do agente, esta RNA vai sendo treinada de forma a

aproximar estes sons dos sons dos vizinhos, com ou sem condicionantes, e acabando no limite por gerar um som igual ao som da sua prole para o mesmo estímulo. Esta classe tem os métodos necessários a este processamento;

- InputANN: esta classe apenas serve como filtro dos valores de entrada. Com efeito, no caso em causa, são detectadas nos sensores a espécie, velocidade e energia dos agentes na proximidade. Esta entrada considera que no parâmetro velocidade, por exemplo, só tem interesse distinguir a velocidade nula ($=0$), de aproximação (<0) ou de afastamento (>0). O objectivo da classe é evitar a atribuição de símbolos diferentes a estados diferentes com um diferencial pouco significativo em termos de consequências; é o caso, por exemplo, da detecção de um agente com velocidade 20 ou com uma velocidade 21. Neste caso não faz sentido que o agente atribua signos diferentes para esses estados próximos. É, aliás, uma situação igual à que ocorre nos seres reais;
- Sound: guarda o tipo da variável som, a entidade que serve de comunicação. Guarda o tempo em que o som em causa foi gerado, a localização desse som (para permitir uma decisão), o código desse som (como as nossas RNA's funcionam com valores numéricos do tipo double, todos os estados de entrada provocam um valor do tipo double na saída, que é a representação do som). O facto de este valor ser do tipo double torna quase impossível o processo de convergência e por isso, o programa funciona com um limite de similaridade. Nos nossos testes, configuramos o programa de forma a considerar que dois sons referem-se ao mesmo objecto se forem iguais até à terceira casa decimal, o que permite proteger o desempenho e ao mesmo tempo a possibilidade de os agentes distinguirem várias centenas de situações diferentes. De notar que, de igual modo, o mesmo se passa na natureza já que a mesma palavra dita por duas pessoas é sempre diferente devido às especificidades da tonalidade, timbre, sotaque, etc. O indivíduo que ouve assume o sentido do som, relacionando-o, por aproximação, a um som que ele já tenha catalogado. No entanto, nesta classe, o grau de similaridade é parametrizado, para permitir outro tipo de análises;

- *SpeechStatus*: esta classe serve para criar o tipo de dados que será usado no processo estatístico da evolução da emergência dos símbolos;
- *Symbol*: esta classe pretende ser o tipo de dados que representa um objecto em si. Contém o nome do objecto (palavra que define o objecto no programa, por exemplo, “Rabbit”, “Fox”, “Bug”, etc) e as propriedades que o caracterizam (no caso dos seres definidos na plataforma *getAlife*, estas propriedades são a espécie, a velocidade e a energia). A definição das propriedades, permite a emergência de símbolos não só para objectos mas também para situações como: fugir, atacar, aproximar, etc. É esta classe que define os valores de entrada na RNA de aprendizagem, através das suas propriedades. Ela também converte o nome do objecto num código numérico do tipo *double*, já que a RNA, funciona com variáveis deste tipo;
- *WorldSpeech*: é a classe que controla todo o processo de interacção dos agentes, relacionados com a comunicação, e faz a ligação com o resto do programa. É nesta classe que são especificados os parâmetros de teste e de cálculo, assim como a escolha dos algoritmos e restrições, e é também ela que faz a ligação às classes gráficas para a monitorização gráfica do processo.

Estas classes encontram-se disponíveis na plataforma *getAlife*, e podem ser acedidas, já que são de código aberto.

7.4. Definição das novas espécies

Já referimos na secção 6.3. que, no nosso trabalho, foram introduzidas espécies de segunda geração. Neste capítulo iremos dar uma breve explicação sobre essas espécies e as nossas opções sobre o seu modo de decisão. As novas espécies introduzidas foram:

- *Fox2ndSpecies* – espécie do tipo predador que herda da espécie já existente, *FoxSpecies*, todas as suas características. O objectivo é comparar as respectivas eficácias num ambiente de concorrência com presença de presas.

- Rabbit2ndSpecies – espécie do tipo presa que herda da espécie já existente, RabbitSpecies, todas as suas características. O objectivo é comparar as respectivas eficácias num ambiente de concorrência sem a presença de predadores.

O módulo de decisão das espécies de segunda geração é exactamente o mesmo da espécie ancestral, uma vez que herda totalmente as suas características. O código Java deste módulo de decisão já existe na plataforma *getAlife*. Segundo as indicações do seu autor, este módulo não tem capacidade de evolução (número de genes=0), e permite que o agente tenha quatro opções de decisão pré-programadas: não fazer nada, mover-se aleatoriamente, fugir dos vizinhos e atacar os vizinhos. A escolha de cada uma destas decisões é definida pela informação sensorial do agente, que, entre outras coisas, informa a quantidade e características dos elementos na sua vizinhança.

Esperamos que, embora o módulo de decisão não seja alterado, as informações que chegam aos sensores dos agentes, sejam mais completas nas espécies de segunda geração, devido à previsível possibilidade de comunicação, o que permitirá a estas espécies a escolha de uma decisão mais eficaz.

As espécies escolhidas para este trabalho, utilizam este módulo de decisão simples, como é exemplificado no código extraído da classe da espécie FoxSpecies:

```
public DecisionModule createDecisionModule()
{
    return new (SimpleDecisionModule getDimensions(),
                SimpleDecisionModule, random);
}
```

A vantagem competitiva das espécies de segunda geração traduz-se na incorporação da componente “Cerebrus”, que é uma classe que manipula uma RNA com capacidade de memorização de signos. Esta classe constrói uma estrutura neuronal fixa e disponibiliza um método de inicialização, cujos parâmetros foram escolhidos depois de algumas experiências, que tiveram em conta o equilíbrio entre a complexidade necessária e a rapidez de resposta da RNA. No caso da espécie FoxSpecies, a estrutura é inicializada conforme o método indicado:

```
setCerebrus ([3, 6, 1], SIGMOID, false, false);
```


Esta parametrização, então, inicializa a estrutura da RNA com uma topologia de 3 entradas, 6 neurónios na camada intermédia e uma saída, uma função de activação, *Sigmoid*, e, para simplificar a observação dos primeiros resultados, não são consideradas as hipóteses de *bias* nem de atalhos (*shortcuts*).

7.5 Planeamento do Modelo

Nesta secção, serão explicadas as opções pelos elementos que serviram de base à estruturação do modelo final, como o método de aprendizagem, a topologia, algoritmo, e métodos de treino.

7.5.1. Justificação do Método de Aprendizagem

O fenómeno da emergência das linguagens decorre, fundamentalmente, de um processo de aprendizagem. A questão que se põe, de acordo com as teorias da Inteligência Artificial, é se essa aprendizagem é supervisionada, por reforço ou não supervisionada. Em qualquer das situações, o desenvolvimento de um modelo de aprendizagem tem levado à cada vez mais frequente utilização do paradigma das RNA's, devido ao seu processamento massivamente paralelo, enorme flexibilidade, potencial de aprendizagem devido à alteração dos pesos das conexões, e robustez (Rocha, 2004).

A vertente evolucionária é muito importante, já que permite obter duas propriedades determinantes no surgimento e evolução da linguagem:

- **adaptabilidade autónoma** do sistema a qualquer alteração do seu estado (desaparecimento de elementos ou de espécies, introdução de novas espécies);
- **dinâmica interpretativa do sistema** (com a introdução, pelos descendentes, de novas soluções para a interpretação simbólica do mundo exterior).

7.5.2. Escolha da topologia da RNA

Para a criação do nosso modelo iremos adoptar uma estrutura de redes neuronais sujeita a uma aprendizagem supervisionada, uma vez que esse modelo implica um vector de valores de entrada, V_e , de comprimento variável para cada espécie, que guardará as possíveis propriedades de cada estímulo exterior, e uma saída resultante da activação de cada uma dessas variações de entrada (decisão da RNA).

A RNA unidireccional (RNAU), foi a topologia implementada e analisada, que procura representar um modelo muito concreto. Cada agente analisa, isoladamente, as suas informações sensoriais, e gera uma saída que servirá de treino para a sua RNA. Em simultâneo, servirá de aferição para as RNA's dos agentes na sua proximidade. Por isso, a RNA só tem uma entrada, definida por um vector com um tamanho relacionado com o número de sensores dessa espécie, e uma saída, bem como apenas terá uma camada de neurónios intermédios (Fig.7.2).

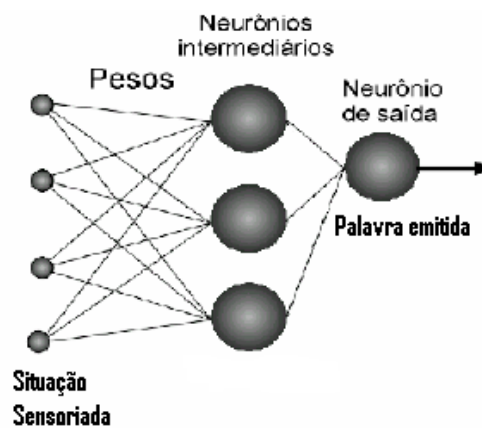


Fig.7.2: Topologia da Rede Neuronal Artificial utilizada

7.5.3. Escolha do Algoritmo supervisionado

No processo de treino da RNA foi seleccionado o algoritmo Retropropagação (*Backprop*). Foram também testados os algoritmos Retropropagação-momentum (*Momentum*), e QuickPropagação (*Quickprop*), mas não foram notadas melhorias nos resultados.

Uma vez que a situação em estudo implica uma aprendizagem ao longo da vida do agente, como já foi referido, a escolha do controlo da lista de treino obedeceu a critérios de eficácia. Assim, não parece ser aconselhável definir em cada iteração um valor muito elevado para o número máximo de épocas, uma vez que cada época será executada em cada iteração que despolete um processo de aprendizagem (alteração de estado nos sensores). Para o algoritmo foi considerada a função de activação *sigmoid*, com domínio $[-1,1]$, e com os valores predefinidos na classe *Function*, do *package ann*.

7.5.4. Definição do método de treino das RNA

Tradicionalmente, o processo de treino implica a utilização de um algoritmo que actua sobre exemplos de treino pré-definidos, em que as entradas e as saídas esperadas correspondentes, são utilizadas com o fim de ajustar os pesos das conexões, até obter um valor de erro aceitável (Rocha, 2004). Depois deste treino, poder-se-ão aplicar, então, os exemplos de teste, onde as saídas não serão conhecidas.

Na nossa situação, iremos introduzir uma pequena *nuance*, necessária à característica especial do objectivo do trabalho. Repare-se que o modelo pretende analisar situações de aprendizagem permanente ao longo do ciclo de vida do agente e da sua civilização. Com efeito, estamos a estudar a resolução de problemas dinâmicos em que as variáveis dos problemas vão sofrendo pequenas alterações e as soluções terão que acompanhar essas alterações ao longo do tempo.

Assim sendo, o processo de treino, terá que coexistir com o processo de teste, durante todo o ciclo de existência do agente. Estes modelos já foram analisados e são previstos nos *packages* desenvolvidos por Rocha e Cortez, que servirão de suporte a este trabalho. A aprendizagem é feita de dois modos:

- Se a situação sensorizada ainda não tem um equivalente de saída, então o agente assume um valor de saída oriundo de duas situações possíveis.
 - se não existir na sua vizinhança outro agente da mesma espécie, então o agente gerará um signo aleatório, que vai servir para a sua própria referência, com o fim de classificar o fenómeno responsável pela entrada no sensor, e que será o caso responsável pelo treino da sua RNA;

- se existir na sua vizinhança um ou mais agentes da mesma espécie, então o agente criará uma lista de treino (casos de treino) com as respostas que os seus vizinhos produzem para o mesmo fenómeno de entrada. O agente gerará a sua primeira interpretação deste fenómeno, com a resposta da sua RNA, treinada pela lista de treino obtida da forma referida atrás.
- Se a situação sensorizada já tem um equivalente de saída, o agente responderá de acordo com duas situações:
 - se não existir na sua vizinhança outro agente da mesma espécie, então o agente manterá a mesma RNA sem nenhum treino;
 - se existir na sua vizinhança um ou mais agentes da mesma espécie, então o agente anexará à sua lista de treino as respostas que os seus vizinhos produzem para o fenómeno de entrada em causa. O agente de seguida treinará novamente a sua RNA, considerando agora a lista de treino ampliada com os novos casos.

De notar que, neste modelo de aprendizagem, a informação dos vizinhos complementa a própria informação do agente, já que ela reforça sempre a lista dos casos de treino, mesmo em casos de situações repetidas.

As respostas (signos) serão representadas por um valor numérico, que codifica a sua representação lexical.

7.5.5. Explicação das técnicas usadas

Como se tem vindo a observar, existem bastantes alternativas para a criação de modelos que implementem este tipo de estudos, envolvendo aprendizagem e optimização. No nosso caso, escolhemos uma estratégia própria cujas opções serão justificadas.

Embora os agentes, através dos seus sensores, detectem da mesma forma as propriedades dos elementos externos, activos ou passivos, do mundo em que evoluem, assumimos que a transmissão de informação entre eles implicará a geração de um símbolo (emergência da palavra) para representar a “ideia” a ser transmitida. Isto faz

todo o sentido, já que na Natureza, a comunicação também é feita através de sons simbólicos, e não pelas descrições das propriedades dos elementos em causa.

De facto, todos os seres humanos, por exemplo, apreendem um objecto pela captação das propriedades físicas desse objecto nos seus sensores (cor, forma, cheiro, textura, movimento, etc.) mas comunicam esse objecto pela transmissão de um símbolo com significado consensualizado, e não pela transmissão dessas propriedades.

Será esta a técnica que também adoptaremos, embora com a consciência da perda de alguma eficácia na comunicação entre agentes. Na verdade, a geração dos símbolos e a sua consensualização são um processo bastante complexo e “pesado” em termos de implementação e execução algorítmica. No entanto, a obtenção de um modelo que represente o processo real de emergência da linguagem, assim como as vantagens resultantes de uma grande autonomia interpretativa, compensam a longo prazo essa falta de eficácia, e é, afinal, um dos objectivos deste estudo.

Como o mundo está em constante evolução, os “sons” simbólicos não podem ser fixos, mas sim dinâmicos, o que evidencia a importância da opção da emergência da linguagem em detrimento da definição prévia dos símbolos a usar na comunicação. Além disso, esta opção permite que cada espécie comunicadora adapte autonomamente os processos e símbolos de comunicação às características da sua própria espécie, o que teremos, em estado mais avançado, no mesmo mundo virtual, as espécies a comunicarem entre si com símbolos (sons) diferentes de outras espécies, para a representação do mesmo objecto.

As classes que desenvolvemos em Java, no âmbito deste trabalho, anexadas à plataforma *getAlife*, têm isso em consideração; se for introduzida uma espécie com um comportamento sensorial com características diferentes (por exemplo, mais componentes sensoriais), ela gerará, certamente, uma comunicação diferente, própria, para essa espécie (a topologia da RNA, alterar-se-á, obviamente, com uma variação na componente de entrada). Se, no mundo virtual que estamos a usar, for introduzido um obstáculo físico (um rio virtual, por exemplo) que potencie a separação dos elementos de uma espécie em dois grupos com dificuldades de ligação, esses grupos, embora da mesma espécie, irão evoluir duas formas diferentes de linguagem, devido ao processo de aprendizagem cultural, apoiado na informação do vizinho, que adoptámos. Esta hipótese abre opções de análise muito interessantes, de que falaremos no fim deste texto, em termos de trabalho futuro.

7.6. Parametrização do modelo

Antes de todo o processo de estudo prático válido para esta tese, foi feito um trabalho de pesquisa e selecção prévio para encontrar as melhores opções de ferramentas para o objectivo do trabalho. Neste texto, e neste capítulo em particular, apenas serão referidas as configurações e resultados relevantes para as conclusões e desenvolvimento assertivo do trabalho. Evidentemente, foram testados muito outros cenários, como por exemplo, outros modelos, assim como variadas combinações de parâmetros e condições. Apenas aproveitamos aquelas que nos deram sinais de resultados interessantes e com garantia de conclusões concretas e fiáveis (positivas ou negativas).

7.6.1. Valores da topologia da RNA

Na RNA, depois de várias experiências com variações de parâmetros, optou-se pela estrutura [n,6,1]. Estes valores são explicados do seguinte modo:

- n – as entradas dependem da espécie em causa. No nosso caso, as espécies escolhidas foram, *fox2ndSpecies* e *rabbit2ndSpecies*, herdeiras das espécies *foxSpecies* e *rabbitSpecies*, respectivamente, que por sua vez têm três parâmetros sensoriais: espécie, velocidade e energia. Assim, assumimos um vector de três entradas para estas espécies, donde $n = 3$;
- 6 – número de neurónios na camada intermediária. Depois de alguns testes, começando com 20 neurónios intermédios, verificou-se que a RNA ficava muito mais pesada em termos de ocupação de memória e de velocidade de cálculo. Os resultados não pioravam se fosse reduzido esse valor, e, depois de sucessivas reduções, este foi finalmente fixado em 6. Este valor garante a complexidade suficiente da RNA sem colocar em causa a eficácia do cálculo. No entanto este valor será, na versão seguinte, colocado à interactividade do utilizador, como parâmetro no ecrã de arranque, que permitirá analisar algumas situações diferentes;
- 1 – apenas há um resultado para cada combinação dos valores de entrada. Este resultado representa o som (palavra) que o agente gera, para cada entrada em causa. Este som serve para o agente gerir a sua aprendizagem e para exteriorização, que assumimos como “tendência” dos agentes de segunda geração.

7.6.2. Parâmetros estatísticos

Para a análise dos resultados foi conveniente a consideração de dois parâmetros muito importantes:

- **Desvio Padrão.** Mede a dispersão das palavras que os agentes da mesma espécie, usam para referenciar a mesma situação. Existirá uma evolução positiva se o processo de aprendizagem fizer com que as iterações aproximem a representação dos símbolos até se tornarem semelhantes. Portanto, espera-se uma diminuição gradual deste desvio. A fórmula desta medida é:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{X})^2}{n}}$$

onde, x_i representa cada valor dos símbolos, comunicados pela espécie em análise, para uma interpretação da entrada, \bar{X} representa o valor médio dos símbolos e n representa o número de agentes comunicadores dessa espécie.

- **Coefficiente de Pearson.** Como se sabe, representa a razão do desvio padrão pelo valor médio, o que permite obter valores mais independentes. Note-se, que um desvio padrão de 4 para uma situação com valor médio 60, não tem o mesmo significado do mesmo valor de desvio padrão se o valor médio for 10, pelo que o Coeficiente de Pearson permite uma perspectiva diferente e complementar para a evolução da experiência. A fórmula usada foi:

$$CV = \frac{\sigma}{\bar{X}}$$

onde σ representa o desvio padrão, e \bar{X} representa o valor médio da série.

7.6.3. Parâmetros do cálculo

Para testar as melhores respostas foram consideradas diversas condicionantes assim como variados parâmetros:

- Parâmetro de idade de cada signo no agente; de cada vez que um agente actualiza um signo, ele associa-o à data de actualização. Isto permite dotar os agentes de opções de optimização nas suas listas de treino.

- Opção de lista de caso FIFO: dotou-se o sistema da possibilidade de permitir aos agentes a função de “esquecimento”. Pensou-se que, para permitir uma melhor convergência, os casos de treino mais antigos não seriam interessantes, pelo que a lista de treino teria um comprimento máximo, e, quando este fosse atingido, o agente “esqueceria” os mais antigos num processo de First In First Out. Esta condicionante teria ainda a vantagem de dar mais celeridade ao sistema já que cada agente lidaria com listas de treino reduzidas e fixas. No entanto, os resultados não comprovaram estas expectativas. A convergência não melhorou, e a velocidade piorou. Este facto deve-se a que o tempo e memória ganho com a limitação das listas de treino, é descompensado pelo processo de controlo da lista FIFO. No entanto, o sistema mantém esta possibilidade para testes.
- Opção de credibilidade dos vizinhos educadores de acordo com:
 - Mais **velho**: testou-se a hipótese de considerar a possibilidade de aumentar a convergência se cada agente aprendesse apenas com os vizinhos que fossem mais velhos que ele próprio. A justificação é que se poderá pensar que vizinhos mais velhos têm uma resposta mais próxima da generalidade da espécie pelo que o seu ensinamento aproximará a resposta do aluno. Se pelo contrário o vizinho fosse mais novo poderia ocorrer o oposto, afastando a solução. Os testes não mostraram melhorias. De notar que os mais velhos podem não ter as soluções mais actuais, o que irá afastar a solução nesses casos.
 - Mais **actual** (resposta mais recente para o caso em causa). Variante optimizada do raciocínio anterior. Aqui o agente aceita ensinamento apenas se os vizinhos tiverem uma resposta mais actual.
 - Mais **experiente** (lista de treino maior). Esta condição considera que os vizinhos são fiáveis se a sua lista de treino tiver um comprimento maior.
- Opção de competência dos vizinhos (estudo em aberto). Usa as condições descritas em cima. A diferença é que o agente apenas aceita um único professor de cada grupo de vizinhos escolhendo o que tem a melhor característica, dentro das citadas no parágrafo anterior. Esta opção, embora considerada, não foi testada no âmbito deste trabalho, já que iria introduzir uma complexidade que se quer evitar nesta fase.

7.6.4. Parâmetros medidos

Uma vez que o estudo aborda o processo de emergência de linguagem num sistema artificial, e o efeito da comunicação dessa linguagem sobre o desempenho desse sistema, tivemos que considerar a medição de dois tipos de parâmetros:

- Número de elementos comunicadores: parâmetro para medir emergência da linguagem. Note-se que a plataforma representa um mundo evolutivo pelo que, embora a configuração utilizada imponha que as populações sejam estáveis, existem alturas em que se operam processos de refrescamento das gerações, ou por “morte natural” dos agentes no processo evolutivo, ou porque as gerações vão perdendo energia média e provocam no sistema uma reposição brusca de população. Assim, nessas situações, os elementos comunicadores podem ter perturbações momentâneas nos seus valores.
- Número total de palavras: parâmetro para medir emergência da linguagem. Este parâmetro mede a quantidade de palavras diferentes que uma espécie comunicadora usa para referir o mesmo símbolo. O sucesso da emergência, exige que o valor deste parâmetro, embora começando com um valor próximo ou igual ao número de elementos da espécie, se vá reduzindo com a aprendizagem. Quanto mais baixo este valor, mais agentes usam a mesma palavra para o mesmo símbolo. No trabalho optamos sempre pelo símbolo “rabbit” como objecto de monitorização. Analisámos a evolução do número de seres da espécie “fox2ndSpecies” que acabavam por se referir ao objecto *rabbit*, na sua proximidade, de maneira semelhante.
- Relação de população: parâmetro para medir a competência da espécie. Mede a razão entre duas espécies em concorrência.
- Relação de energia: parâmetro para medir a competência da espécie. Mede a razão entre a energia entre duas espécies. Se for positiva, a primeira espécie está com uma média superior de energia, e se for negativa, então a segunda espécie está com mais energia.
- Relação total de presas: parâmetro para medir a competência da espécie. Uma vez que este parâmetro é muito importante para comparar a competência de duas espécies, este valor não é dado pela razão da média das presas em cada instante, mas pela razão do total do acumulado de presas dessa geração até esse instante.

O objectivo é limitar as situações em que estes valores médios ficam sem significado aquando das alterações bruscas do número de elementos da espécie. Se assim não fosse, uma espécie com um agente que conseguisse uma presa teria melhor resultado do que uma espécie em que os elementos tivessem várias presas, mas em média tivesse um valor inferior a um. O que não faria sentido. O sistema soma, em cada iteração, o total de presas acumuladas por cada agente, em vida, até esse instante, e obterá o valor médio de presas caçadas até ao momento.

- Relação das idades: parâmetro para medir a competência da espécie. Procura dar uma indicação sobre a comparação média das idades das espécies em concurso. De notar que o sucesso da espécie aumenta a sua energia, o que impõe uma maior tempo de vida dos seus elementos. A visualização deste parâmetro permite estimar a evolução desse sucesso.

Capítulo 8

Análise de Resultados

A análise dos resultados deste trabalho dividiu-se em dois grandes grupos:

- Análise da emergência da linguagem. O objectivo foi a constatação de que os agentes integrantes no modelo definido no capítulo anterior, tinham capacidade de criar entre si uma capacidade comunicativa. Foram escolhidos as espécies *foxSpecies*, *fox2ndSpecies*, *rabbitSpecies* e *plantSpecies*;
- Análise da melhoria de desempenho da espécie. Foram colocadas em concorrência duas espécies da mesma família, diferenciadas apenas pela capacidade de emergência autónoma de comunicação: *rabbit2ndSpecies* e *rabbitSpecies*. O objectivo foi verificar se a espécie com capacidade comunicativa, *rabbit2ndSpecies*, tinha melhores resultados, em termos de procura e obtenção do seu alimento, *plantSpecies*, que a outra espécie, *rabbitSpecies*, sem essa capacidade comunicativa.

Será explicado mais à frente, neste texto, a opção pela escolha destas espécies, e correspondente configuração do sistema.

8.1. Resultados de Emergência de Linguagens

Os resultados obtidos tiveram origem em várias configurações, com o fim de comparar e obter conclusões o mais genéricas possíveis, i.e., foram escolhidas as configurações que permitiram que as conclusões consequentes não reflectissem situações específicas ou particulares, dentro das limitações impostas pelo tempo disponível e pelas próprias condições impostas pelo sistema utilizado.

8.1.1. Configuração do sistema em análise

As espécies seleccionadas para esta análise foram:

- *fox2ndSpecies*. É a espécie comunicadora. Consegue criar e memorizar símbolos para todos os agentes que encontra no seu mundo. Vai adaptando esses símbolos durante as interacções com elementos da mesma espécie, com o objectivo de consensualizar esse símbolo para o mesmo significado. No entanto, embora a sua RNA memorize e evolua todos os símbolos das espécies e acções presentes, por simplicidade de análise, apenas iremos monitorizar a convergência dos símbolos evoluídos para a sua presa, *rabbit*;
- *rabbitSpecies*. É a espécie que serve de presa à espécie comunicadora. Servirá de referência para a monitorização da convergência da comunicação, como o objecto representativo do símbolo em estudo;
- *plantSpecies*. É uma espécie que servirá de alimento à espécie *rabbit*, com o fim de impedir variações muito rápidas das gerações desta espécie devido a quedas abruptas de energia. Estas substituições muito rápidas provocam perturbações nas medidas de convergência como se verificou nos testes prévios.

Os parâmetros utilizados para as medidas finais foram os constantes na Tabela 8.1:

| Nº Iterações | fox2nd | rabbit | plant |
|--------------|--------|--------|-------|
| 100 | 10 | 50 | 50 |
| 1 000 | 30 | | |
| 10 000 | 50 | | |
| 100 000 | 100 | | |

Tabela 8.1: Parâmetros de teste

Para cada número de iterações, referidos na tabela 8.1, foram testados alguns valores possíveis de *fox2nd*, com o objectivo de testar a relação entre os valores de

convergência e o número de elementos comunicadores. Os valores escolhidos para estes foram 10, 30, 50 e 100 (Tab.8.1).

O número de presas, *rabbit*, nesta medida, foi fixado em 50. Este valor será variável nas medidas da concorrência entre espécies que será feita na secção 8.2.

O número de elementos da espécie *plant* é fixo, uma vez que não interfere directamente na convergência. Depois de algumas experiências fixou-se este valor em 50.

8.1.2. Resultados obtidos

Iremos analisar os resultados em termos de evolução e tendência gráfica assim como em termos de valores absolutos resultantes das medições, para as várias situações referidas na secção 8.1.1

Situação 1. Número de iterações = 100.

Este valor de iterações permite tirar algumas conclusões relativamente ao efeito do nosso modelo de aprendizagem sobre uma geração de agentes comunicadores.

A visualização gráfica permite-nos concluir que existe uma grande capacidade de convergência do algoritmo de aprendizagem, logo no início do processo. Esta convergência está bem representada no seguinte gráfico comparativo (Fig.8.1):

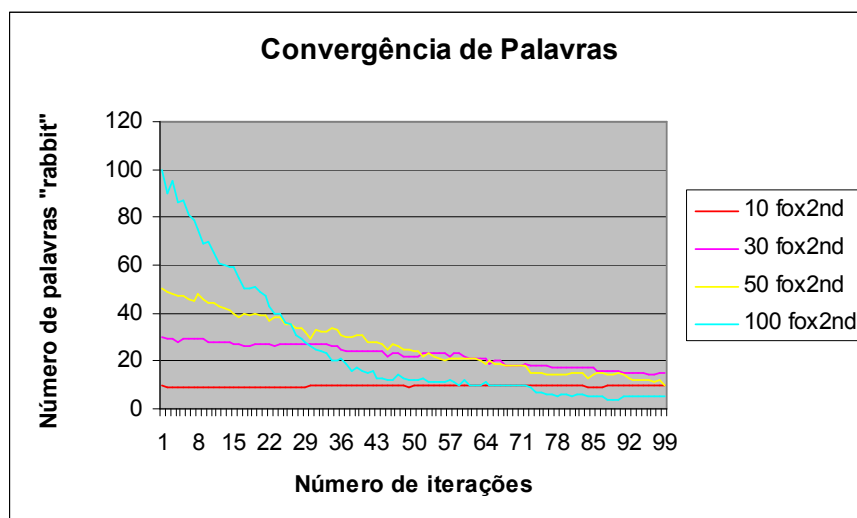


Fig.8.1: Gráfico comparativo para a primeira geração

Este gráfico representa o número de símbolos diferentes que a população da espécie comunicadora utiliza para representar a presa “*rabbit*”. Nota-se que, quanto maior for o número de comunicadores, mais acentuada é a convergência, como seria de esperar, uma vez que uma maior quantidade de trocas de informação permite mais rápidas correções dos símbolos. Nota-se, no entanto, outro fenómeno interessante: quanto maior for o número de agentes comunicadores menos símbolos são utilizados pelo grupo para representar o mesmo objecto. Repare-se que no fim da geração, no conjunto dos 100 agentes predadores apenas existem 5 símbolos diferentes para representar a presa. Quando o número de predadores desce para 30, o número de símbolos sobe para cerca de 15, no fim da geração, o que reduz bastante a eficácia da comunicação. Para 10 predadores, verifica-se a inexistência de convergência devido a insuficiência de processos de comunicação. Há demasiada dispersão para pequenas quantidades de agentes comunicadores.

Embora durante as primeiras iterações os símbolos se possam manter diferentes, podemos no entanto visualizar o sentido da convergência utilizando a medida de dispersão do desvio padrão. Esta medida, neste caso de análise, informa-nos da distância dos símbolos que representam o objecto “*rabbit*”. Espera-se que, com a aprendizagem, estes símbolos se vão aproximando, até irem assumindo o mesmo valor representativo consensualizado. Isto implica que esta medida de dispersão se vá reduzindo ao longo do tempo. A figura 8.2., confirma esta expectativa.

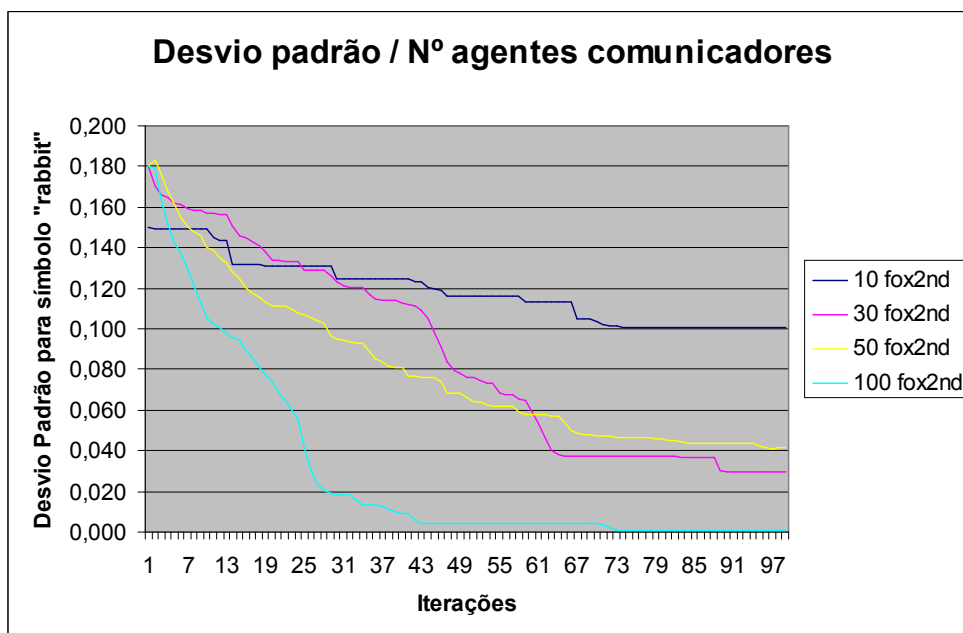


Fig.8.2: Gráfico comparativo do desvio padrão para a primeira geração

O gráfico da figura 8.2. mostra a redução acentuada da dispersão, confirmando que, conforme esperado, a dispersão reduz tanto mais quanto maior o número de agentes comunicadores. Mostra ainda que, para poucos agentes comunicadores, 10 no caso em análise, embora os símbolos nunca se assumam como iguais devido à constante dispersão introduzida pelo nascimento de novos elementos, existe, mesmo assim, uma aproximação dos símbolos representativos bem visível no declínio do desvio padrão.

Situação 2. Número de iterações = 1 000.

O objectivo destas medições é analisar o comportamento da passagem da componente “cultural”, introduzida pela comunicação, entre as primeiras gerações.

A substituição das gerações tem dois efeitos determinantes, mas opostos, sobre este processo. Um efeito negativo sobre a convergência, já que a introdução de novos agentes comunicadores vem inserir novos símbolos de interpretação para os objectos cuja convergência tinha sido conseguida pelos agentes da geração anterior. Esta consequência provoca uma pequena mutação nesses símbolos que se alastram a toda a população dos agentes comunicadores. As consequências serão pequenas variações dos símbolos de geração para geração responsável por um aumento das medidas de dispersão. O que aliás é interessante já que reflecte o que se passa na evolução linguística dos seres humanos. Basta lembrar a evolução de uma palavra desde o latim até ao português actual, em que as variações lentas entre gerações podem provocar diferenças substanciais a longo prazo, para o mesmo termo.

Em contraponto, a introdução de novas gerações tem um efeito positivo, já que permite que o processo de aprendizagem seja dinâmico. Ele introduz uma espécie de “selecção natural” para os símbolos. Novas interpretações podem permitir uma procura de símbolos sempre constante e impedir que a convergência estagne em soluções pouco interessantes. Pode impedir, também, a criação de grupos linguísticos isolados que se concentrem numa dada vizinhança. Além disso permite que objectos ou situações que deixem de fazer parte do sistema deixem também de ter uma representação simbólica, de uma forma totalmente gerida internamente pelo sistema, sem qualquer ajuste ou correcção externa.

Este comportamento foi traduzido na representação da análise gráfica da figura 8.3.

Por motivos de simplificação de tratamento, foi considerado um intervalo de 1000 iterações, onde cada medida representa a média de cada série de 10 iterações. Este critério não põe em causa qualquer tendência gráfica e permite uma concentração de valores mais fácil de tratar. As alterações feitas ao programa *getAlife*, permitiram usar estas concentrações.

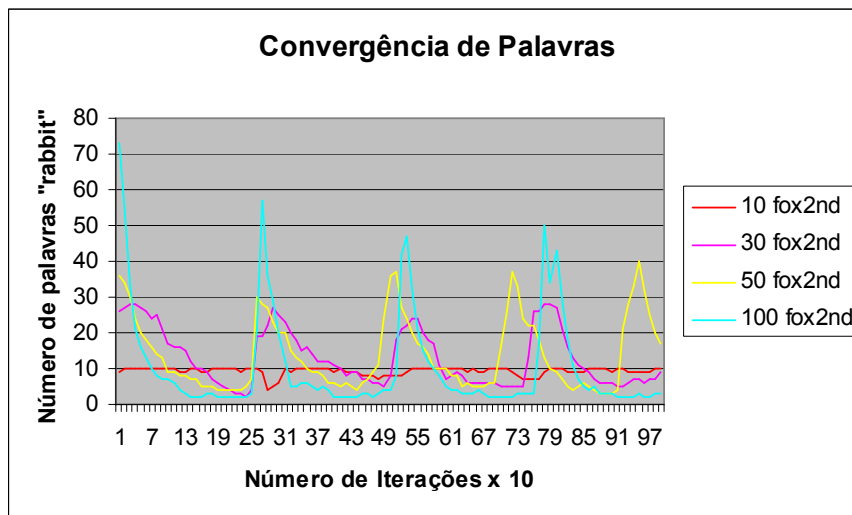


Fig.8.3: Gráfico comparativo para entre as primeiras gerações

É claramente visível, na figura 8.3., a periodicidade introduzida pelo refrescamento das gerações, nos primeiros tempos do processo. A eficácia do algoritmo de aprendizagem, em termos de convergência, mantém-se de geração para geração. Nota-se que a periodicidade é independente do número de agentes, pelo menos nas primeiras gerações. Isto leva-nos à conclusão de que, nas primeiras gerações, a passagem de conhecimento para as gerações seguintes não é muito relevante, de contrário seria notada uma atenuação nos picos apontados no gráfico da figura 8.3. A explicação deste fenómeno será a introdução muito abrupta de novos agentes, que criam referências novas, muito fortes, para novos símbolos de interpretação do objecto “*rabbit*”. Será então interessante ver o que sucede a mais longo prazo.

Situação 3. Número de iterações = 10 000.

Neste caso procuramos analisar a evolução da convergência para um prazo médio. A ideia é verificar se os ciclos de gerações continuam a causar dispersão ou se no médio prazo essas dispersões se vão diluindo.

O gráfico obtido foi o indicado na fig. 8.4:

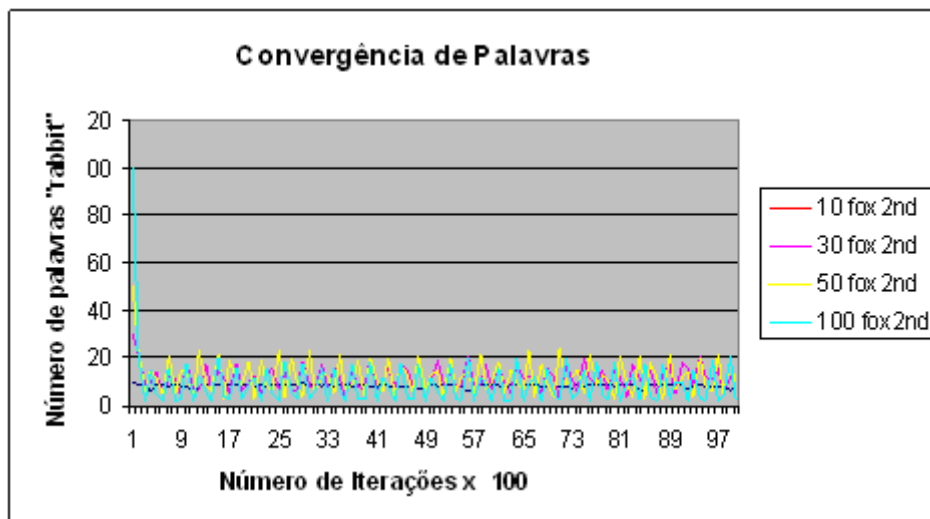


Fig.8.4: Gráfico comparativo para médio prazo

A análise de médio prazo, permite concluir, sem qualquer dúvida, duas situações bem claras. As oscilações entre gerações são uma constante ao longo das iterações. Essas oscilações mantêm-se dentro de uma curta faixa de valores, entre 0 e 20, pelo que podemos dizer que a convergência existe e é tanto mais significativa quanto maior for o número de elementos comunicadores. No entanto, para um número de 10 elementos, não podemos assumir essa convergência. Do mesmo modo, fica fora dessa consideração a situação para um número de elementos igual a 30, já que as oscilações de aproximam deste valor o suficiente para não assumir a situação como decisivamente convergente, em termos de médio prazo.

Para leituras de concorrência entre espécies, no médio prazo, este número de 30 elementos é, por esse motivo, desaconselhado, já que não verifica a premissa de essa espécie ter uma capacidade suficientemente capaz de comunicação.

Situação 4. Número de iterações = 100 000.

Neste caso procuramos analisar a evolução da convergência para um longo prazo. De notar que os AE's são adequados e orientados a aplicações de longo prazo, pelo que a análise destes valores é muito importante para concluir sobre situações de estabilização ou não do modelo.

O gráfico obtido foi o indicado na figura 8.5:

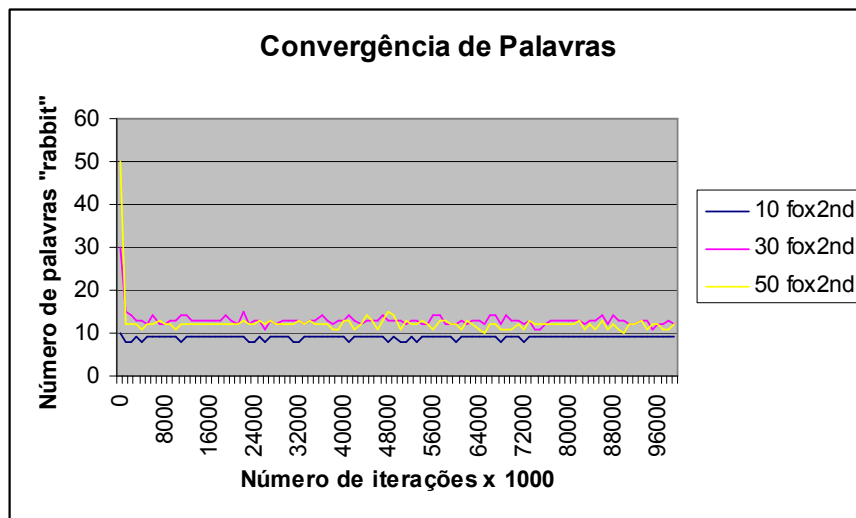


Fig.8.5: Gráfico comparativo para um longo prazo, 100 000 iterações.

È visível que os valores mantêm uma oscilação inter-geracional, o que evidencia uma dinâmica de renovação constante dos símbolos para as interpretações dos objectos.

A faixa de valores, entre 11 e 14, numa população de 30 e 50 elementos, permite uma assumpção de que existem canais de comunicação significativos que poderão conduzir a uma optimização no comportamento da espécie, a longo prazo.

Podemos assim concluir que, no longo prazo, o modelo permite obter valores de convergência aceitável para o considerarmos como um sistema comunicacional. Em simultâneo, o modelo garante uma dinâmica renovadora que o impede de se cristalizar num sistema estático e com dificuldade de reacção adaptativa. Para grupos de 10 elementos comunicadores, continua a ser descartável essa assumpção, como se vê no gráfico 8.5.

8.2. Resultados de melhoria de desempenho da espécie

Na análise da melhoria de desempenho da espécie, foi considerado primordial perceber se as espécies com capacidade comunicadora podem obter melhor desempenho do que as espécies que não desenvolveram essa capacidade. Por questões de simplificação de análise, uma vez que alguns destes cálculos demoram muitos dias e até semanas, foi criado um ambiente de três espécies, com duas dessas espécies em concorrência na predação de uma terceira espécie do tipo presa. Neste caso considerou-se a presa fixa, *plant*, e os predadores em concorrência as espécies *rabbit2nd* e *rabbit*, tendo a espécie de segunda geração, *rabbit2nd*, a capacidade de desenvolver uma forma de comunicação de acordo com o que foi exposto na secção 8.1.

Do ponto anterior, concluímos que as comunicações entre agentes, para sistemas com características evolutivas inter-gerações, apenas se tornam significativas no médio e no longo prazo, se as populações garantirem um valor mínimo de elementos. Para valores inferiores de agentes verificamos que os surgimentos constantes de novas gerações causam perturbações na comunicação entre gerações que não conseguem ser estabilizadas pelo grupo. É obvio que este valor mínimo dependerá de múltiplos factores intrínsecos a cada sistema. No nosso modelo comprovamos que grupos de 50 elementos nos dão garantias de uma estabilização cultural aceitável.

Assim sendo, analisámos a concorrência entre espécies com populações de 50 elementos, que nos permitiriam garantir valores correctos de análise, como explicado atrás.

Foram testados dois tipos de características; o comportamento dos valores de concorrência em relação ao tempo, e a relação existente entre o número de presas e a eficácia da comunicação, em fenómenos concorrenciais.

Optámos assim por analisar duas situações; uma com poucas presas e outra com um número médio de presas. Para cada uma destas situações foram testados os resultados para os tempos, 100, 1000, 10 000, e 100 000 iterações.

Situação 1. Número de presas = 10

Esta situação pretendeu verificar qual a influência da comunicação inter-agentes nos resultados de predação, em sistemas com um número reduzido de presas (soluções).

Os valores usados nos testes foram, 50 predadores comunicadores, 50 predadores não comunicadores, e 10 presas. Estas opções foram todas justificadas nos pontos anteriores. Os tempos de análise abordaram a primeira geração, 100 iterações, a transição entre gerações, 1000 iterações, o médio prazo, 10 000 iterações, e o longo prazo, 100 000 iterações.

Os predadores foram parametrizados com o modo “*atack*” na criação do seu módulo de decisão, de forma a garantir que a informação recebida pelos seus vizinhos, seja considerada nas suas decisões predatórias.

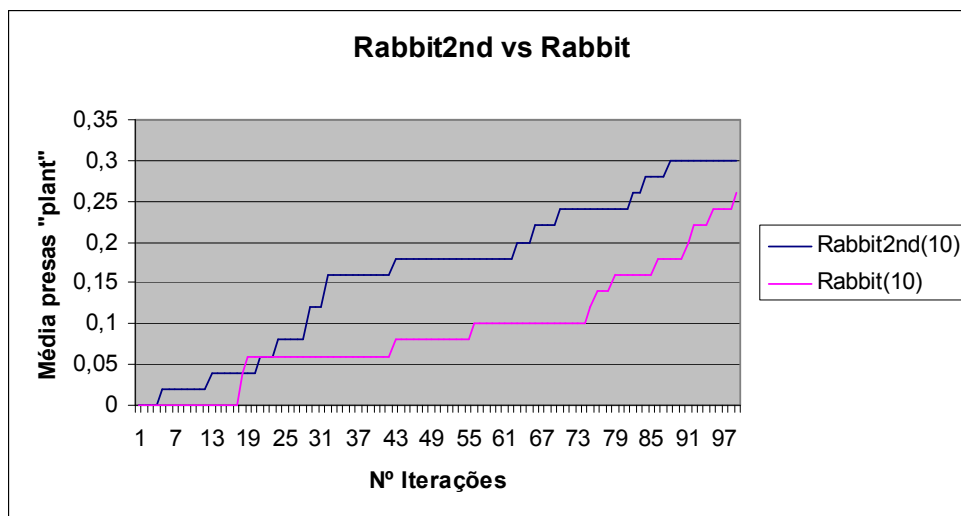


Fig.8.6: Gráfico comparativo para uma geração e 10 presas

O gráfico da figura 8.6. mostra que, em situação de apenas 10 presas, em média, constantes, os agentes comunicadores acabam por ganhar vantagem competitiva sobre os agentes não comunicadores durante uma geração.

Com a mudança de gerações introduz-se alguma perturbação na comunicação que poderá afectar a eficácia na predação. Os resultados para 1000 iterações, com o mesmo número de agentes nas espécies seleccionadas, permitiram a elaboração da figura 8.7:

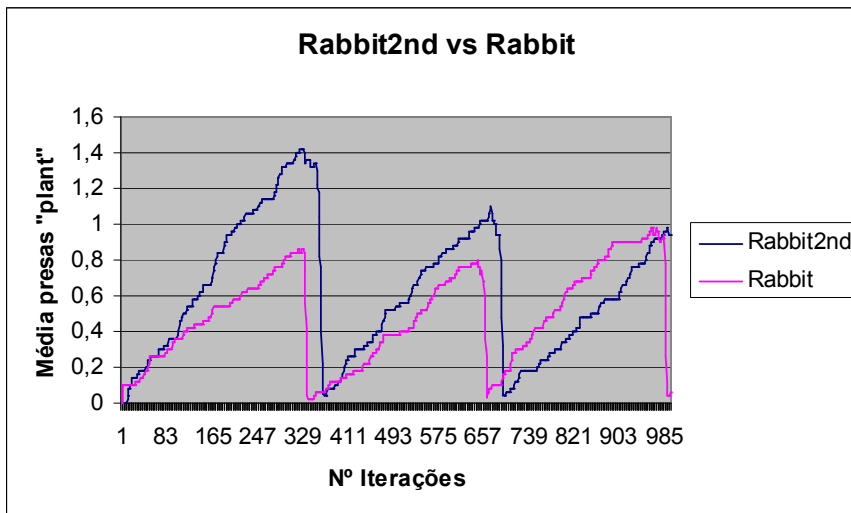


Fig.8.7: Gráfico comparativo para as primeiras gerações e 10 presas

Para uma evolução a curto prazo, teremos o seguinte comportamento para um baixo número de presas: verifica-se que a substituição das primeiras gerações, que ocorre de uma forma inicialmente abrupta, prejudica a eficácia da espécie porque ainda não tornou consistente a sua “cultura”. No entanto, ainda existe vantagem competitiva, como se vê na figura 8.7.

Para medidas de médio e longo prazo, utilizamos medidas médias para verificar com maior clareza a evolução das predações em concorrência. Assim, para 10 000 iterações usamos médias de blocos de 2000 iterações, e para 100 000 iterações iremos usar médias de blocos de 20 000 iterações. Os resultados são visualizados nas figuras 8.8, e 8.9.

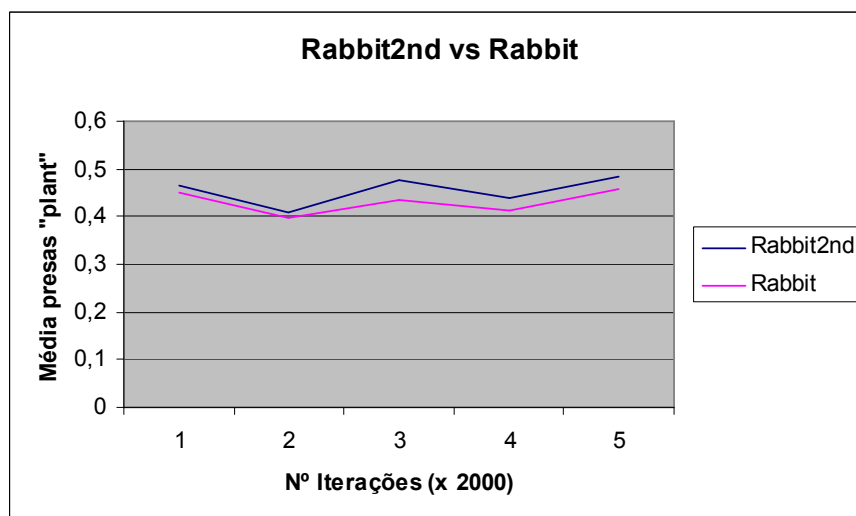


Fig.8.8: Gráfico para 10 000 iterações, 10 presas

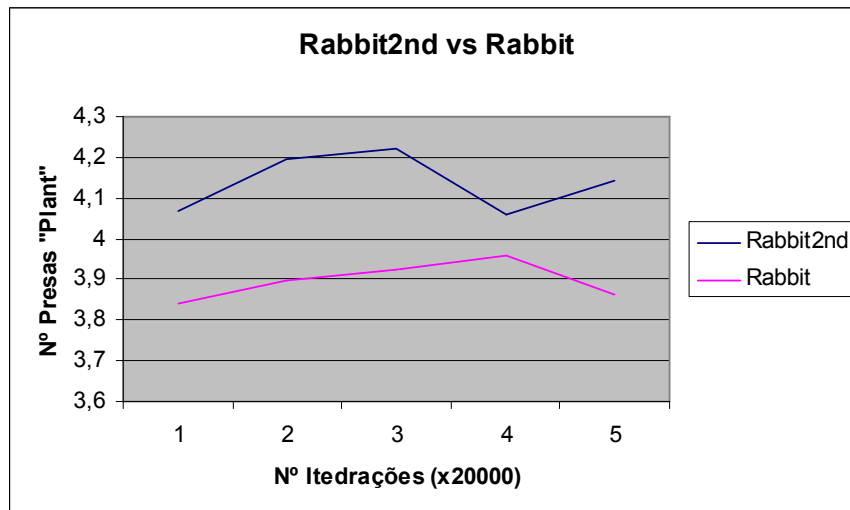


Fig.8.9: Gráfico para 100 000 iterações e 10 presas

É muito clara a vantagem dos agentes comunicadores para situações de poucas presas, como se pode ver nas figuras acima, pese embora o facto de um número reduzido de presas não permitir uma convergência muito rápida na comunicação, como explicado anteriormente.

Situação 2. Número de presas = 30

Esta situação pretende verificar qual a influência da comunicação inter-agentes em sistemas com um número médio de soluções (presas).

Os valores usados nos testes foram, 50 predadores comunicadores, 50 predadores não comunicadores, e 30 presas. Estas opções foram todas justificadas nos pontos anteriores. Os tempos de análise abordam a primeira geração, representada por 100 iterações, transição entre gerações de 1000 iterações, tempo médio de 10 000 iterações, e longo prazo de 100 000 iterações.

Para esta situação, obtivemos os resultados para a primeira geração e primeiras transições entre gerações, visualizados nas figuras 8.10. e 8.11, que continuam a mostrar uma predominância da espécie comunicadora.

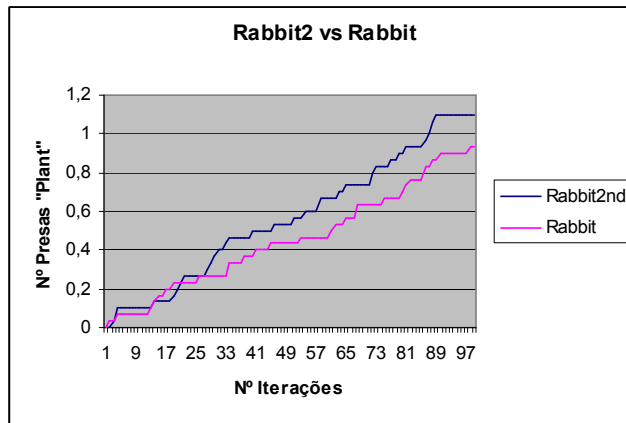


Fig.8.10: 100 iterações, 30 presas

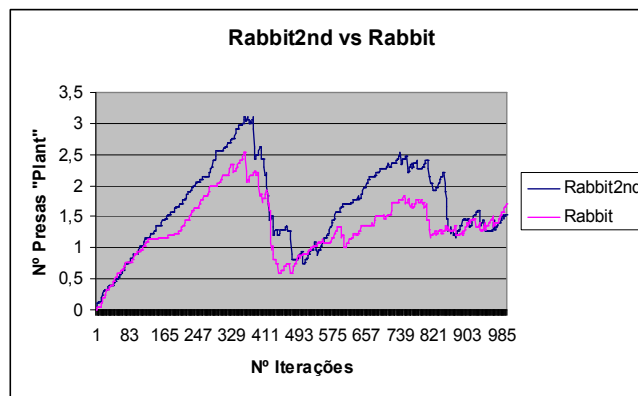


Fig.8.11: 1 000 iterações, 30 presas

Para valores mais elevados de iterações obtivemos, usando também as médias para facilitar a visualização, os resultados explicitados nas imagens 8.12 e 8.13.

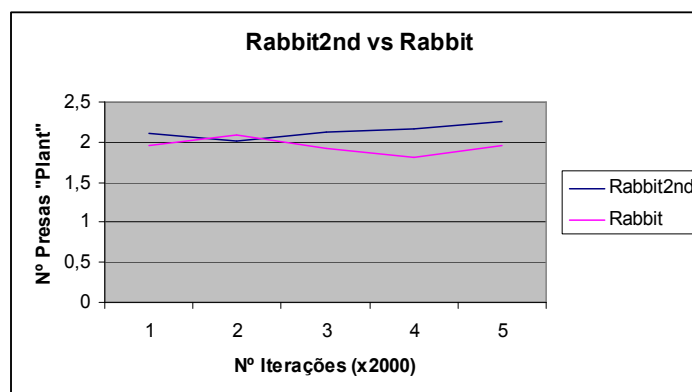


Fig.8.12: 10 000 iterações, 30 presas

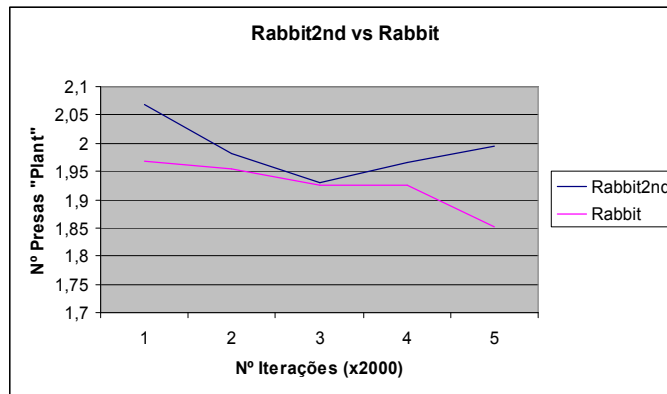


Fig.8.13: 100 000 iterações, 30 presas

Para longos prazos, continua a verificar-se a predominância dos elementos comunicadores. Poder-se-ia pensar que quanto maior for o número de presas, tanto maior será a predominância dos elementos comunicadores, uma vez que mais rapidamente estes conseguirão atingir a convergência de comunicação e mantê-la entre as gerações seguintes. No entanto, os diversos testes feitos mostram que esta conclusão não é assim tão linear. De facto, os resultados mostram que, para uma grande abundância de presas, os factores de vantagem se tornam mais irrelevantes, já que as movimentações aleatórias dos elementos não comunicadores garantem sempre a aproximação a uma das inúmeras presas.

A vantagem da comunicação é, assim, importante em ambientes de concorrência, para situações de carência de presas, com a condicionante de existir um limiar de presas mínimo que garanta o desenvolvimento do “processo cultural”, como aliás foi visto na secção anterior.

Finalmente, é de lembrar que estamos a tratar com ambientes com uma grande quantidade de elementos que se vão substituindo ao longo de gerações, pelo que pequenas diferenças que impliquem pequenas vantagens podem ser determinantes com o acumular do tempo do processo (Tab.8.2).

| | | 10 Presas permanentes | | | |
|------------------------------|----------------------|-----------------------|-------|--------|----------|
| | | Iterações | | | |
| | | 100 | 1000 | 10000 | 100000 |
| Total Presas | 50 Comunicadores | 819 | 31380 | 227081 | 20685101 |
| | 50 Não Comunicadores | 453 | 24585 | 215129 | 19483664 |
| Diferença do total de presas | | 366 | 6795 | 11952 | 1201437 |
| % Vantagem dos Comunicadores | | 181% | 128% | 106% | 106% |

Tabela 8.2: Comparação de totais de presas

Como se pode ver na Tabela 8.2, embora para 100 000 iterações a vantagem não pareça decisiva (106%), o acumulado, no entanto, envolve mais 1 201 437 soluções resolvidas, o que não é desprezável para muitos contextos.

Capítulo 9

Conclusões e trabalho futuro

A sociedade evoluiu de uma forma tão acentuada que viu os problemas com que se confronta, aumentarem de um modo dramático o seu grau de complexidade. O futuro ameaça com um aumento exponencial dessa complexidade. As soluções para a resolução desses problemas deixaram de ser possíveis através das técnicas tradicionais, recorrendo-se cada vez mais a soluções que envolvem paradigmas relacionados com a computação evolucionária.

Dentro da computação evolucionária ganha particular destaque o recurso a sistemas multi-agente. Estes sistemas poderão ter ou não uma capacidade de comunicação que poderá ser ou não determinante para a eficácia do sistema.

O nosso trabalho debruçou-se sobre um tipo particular de sistema: sistema dinâmico em que as condições e componentes envolvidos, podem mudar de forma inconstante e imprevisível.

Este factor orientou a nossa pesquisa para o estudo daquilo a que chamámos “cultura artificial”; se o sistema desconhece, num determinado instante, o estado que vai encontrar em instantes futuros, então ele tem que desenvolver uma cultura adaptativa, que lhe permita recriar novos símbolos de comunicação para novas situações ou novas espécies que surjam, assim como deverá ter a capacidade de ir extinguindo as representações de situações que forem desaparecendo definitivamente do sistema. Assim, o sistema tem que ser totalmente autónomo (*autopoietico*, segundo Varela) na criação das representações dos seus estados, com o objectivo de conseguir uma possibilidade de comunicação, sempre dinâmica.

Neste contexto, este estudo procurou responder a duas perguntas: podem sistemas artificiais, totalmente autónomos, criar, desenvolver e adaptar uma forma própria de comunicação entre si? Pode essa forma de comunicação, emergente, contribuir para a melhoria do desempenho dos elementos que a adquirirem?

Para responder a estas perguntas utilizámos como ferramentas o programa *getAlife*, desenvolvido na Universidade do Minho, e, como base para as estruturas de aprendizagem, aplicámos os módulos de tratamento e manipulação de Redes Neurais existentes na mesma. Foi necessário introduzir novos módulos para incorporar nessa

base de suporte, que é disponibilizada na modalidade *Open Source* (<http://darwin.di.uminho.pt/getalife>), incorporando ferramentas de comunicação e de visualização gráfica da evolução do sistema.

Os resultados do trabalho mostraram que foi possível verificar a emergência de símbolos próprios nos agentes artificiais, e a consensualização, entre si, sobre o processo de interpretação dos diversos estados, permitindo uma comunicação básica.

Este efeito foi conseguido, apenas dotando estes agentes de uma RNA, para armazenar e tratar os símbolos representativos da sua “realidade”, e de uma intencionalidade básica que passa por exteriorizar as suas alterações de estado. Esta exteriorização vai funcionar como uma perturbação do estado do estado do sistema que será, obviamente, apreendido como uma alteração de estado, e tratado como tal pelos agentes na sua proximidade.

Para verificar as vantagens de uma comunicação entre agentes na melhoria do seu desempenho, criámos um sistema com duas espécies semelhantes em concorrência na predação de uma terceira espécie. A diferença entre as espécies é, unicamente, a dotação ou não dos elementos descritos no parágrafo anterior, o que lhes confere, como ficou provado, uma capacidade comunicativa.

Os resultados analisados mostraram a predominância da espécie comunicadora, o que torna esta espécie com mais capacidade para a procura de soluções em problemas de grande grau de complexidade.

Durante o estudo foram observados alguns fenómenos, inicialmente desprezáveis ou imperceptíveis, mas que mostraram ter importância neste tipo de processos. As espécies comunicadoras provocam um efeito de aproximação entre si, enquanto existe uma presa na proximidade de um deles, o que poderá ter duas consequências antagónicas no que respeita à sua eficiência; se as presas surgirem aleatoriamente espalhadas ao longo do espaço do ambiente artificial, esta concentração poderá não ser benéfica para a eficácia dos elementos, uma vez que depois do “cerco” à presa, esta apenas é assimilada por um dos agentes, e a nova presa surgirá, algures, num local, distante do grupo que terá que reiniciar individualmente a sua procura. No entanto, este factor de concentração pode ser muito importante em situações em que as presas (soluções) surjam aglomeradas. Neste caso, o factor de concentração dos predadores decorrente da comunicação, é muito beneficiado com a proximidade de uma aglomeração de presas.

Para trabalho futuro, seria interessante um estudo da medição destes factores de concentração consequentes da comunicação, assim como o estudo de parâmetros que

relacionem esta concentração de predadores com aglomerados de presas, verificando as medidas de desempenho nestes casos.

Seria também interessante a introdução no sistema de obstáculos separadores de culturas, embora parcialmente, i.e., seria interessante analisar situações que forçassem a aglomeração das espécies em zonas, de forma a originarem a convergência de duas “culturas artificiais” diferentes. Este tipo de sistema, permitindo ligações físicas entre as zonas definidas, deixará que elementos de um grupo possam transitar para o outro grupo introduzindo factores de renovação, que ajudarão a impedir uma cristalização cultural e genética, nociva, em cada um desses grupos. Seria, pois, interessante medir os efeitos e valores que esta separação e “miscigenação artificial” provocam nas medidas de eficácia competitiva dos agentes que evoluem no sistema.

As situações analisadas centraram-se na melhoria do desempenho dos predadores. Obviamente, a situação inversa também obedece às mesmas conclusões e, por isso, poderíamos tratar um sistema em que o objectivo fosse melhorar o desempenho da presa. Esta situação dar-nos-ia algumas respostas sobre um cenário concreto em que o sistema fosse a Internet, os servidores *web* fossem as presas e os elementos hostis como os vírus, troianos, ou *spywares*, fossem os elementos predadores. Se um modelo teórico como o nosso pudesse ser aplicado a este sistema seria de todo o interesse verificar se os servidores *web* poderiam emergir uma forma de linguagem de forma a trocarem informações para melhorarem a sua eficácia defensiva contra estes atacantes. A escolha deste cenário não foi por acaso. Note-se que, tal como no nosso modelo, o ambiente é altamente dinâmico, com espécies e características novas constantemente a surgir e extinguir-se do sistema.

Em cenários concretos, poderíamos também aplicar o modelo em programas de pesquisas de soluções ou de elementos, em espaços tipicamente intratáveis e complexos, em que os agentes predadores poderiam ser gerações de *threads* com as características dos agentes do nosso modelo.

A robótica, e outras áreas concretas em ascensão, poderiam igualmente ser alguns dos destinatários destes estudos.

Noutra área não menos importante, como os próprios modelos de análise do surgimento das linguagens, este modelo poderia ser uma boa ferramenta para testar os fenómenos de emergência, evolução e dispersão das linguagens.

Referências

Baker, J., *Reducing Bias and Inefficiency in the Selection Algorithm*. In Grenfenstette, J. editor, *Proceedings of the Second International Conference on Genetic Algorithms and their Applications*. Lawrence Erlbaum Associates, 1987.

Bergson, H., *Creative Evolution*. MacMillan, London, 1911.

Brooks, R., *Elephants don't play chess*. *Robotics and Autonomous Systems*, 6:3–15. 1990.

Brooks, R., *Intelligence without reason*. Technical report AI. Memo n.1293. MIT. 1991.

Brooks, R., *Cambrian Intelligence: the early story of the new AI*. Bradfore Book. MitPress. 1999.

Bunge, M., *Ontology II : a world of systems*, Reidel Dordrecht, 1979.

Cangelosi, A., Parisi, D., *Computer simulation: A new scientific approach to the study of language evolution*. In Cangelosi, A. and Parisi, D., editors, *Simulating the Evolution of Language*, chapter 1, pages 3–28. Springer Verlag, London, 2001.

Cannon, W. *The wisdom of the body*, Norton, New York, 1939.

Cheney, D., Seyfarth, R., *How Monkeys See the World*. University of Chicago, Press, Chicago and London, 1990.

Clancey, W., *Understanding Computers and Cognition: a New Foundation for Design*, a review. *Artificial Intelligence*, 232- 250, 1987.

Corrêa, M., Coelho, H., *Around the architectural agent approach to model conversations*. *Proceedings of the Modelling Autonomous Agents in a MultiAgent World Workshop (MAAMAW-93)*, Nêuchatel, Switzerland. Springer-Verlag, 1993.

D’Inverno, M. et al, *Formalisms for multiagent systems*. The Knowledge Engineering Review, 3, 1997.

Doran, J., Palmer, M., *The EOS Project: Integrating Two Models of Palaeolithic Social Change*. In “Artificial Societies: the Computer Simulation of Social Life” (eds. N Gilbert and R Conte) London: UCL Press Pps 103-125, 1995.

Dreyfus, H., Dreyfus, S., *Making a mind versus modelling the brain: Artificial intelligence back at branchpoint*. Artificial Intelligence Debate, MIT Press, Cambridge, MA, 1988.

Driesch, H., *The Science and Philosophy of the Organism*, Black, London, 1908.

Eco, U., *Semiotics and the Philosophy of language*. Indiana University Press, 1986.

Etzioni, O., *Intelligence without robots (a reply to Brooks)*. Artificial Intelligence. 1993.

Fahlman, S., *The Recurrent Cascade-Correlation Architecture*. *Advances in Neural Information Processing Systems 3*, Lippmann, R.P., Moody, J.E., Touretzky, D.S., eds. Morgan Kaufmann, San Mateo CA, 190-196, 1991.

Falkenauer, E., *Genetic Algorithms and Grouping Problems*. John Wiley and Sons, Chichester, United Kingdom, 1998.

Farmer, J., Belin, A., *Artificial Life: the coming evolution*. Langton et al., 1992.

Ferber, J., *Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*. Addison-Wesley, 1999.

Fogel, L., Owens, A., Walsh, M., *Artificial Intelligence, Through Simulated Evolution*, John Wiley, New York, 1966.

Fonseca, J., *Protocolos de negociação com coligações em sistemas multi-agente*, tese de doutoramento, Universidade Nova de Lisboa, Lisboa, 2000.

Franklin, Stan and Graesser, *Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents.*- Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages, Springer-Verlag, 1997.

Frean, M., *The Upstart Algorithm: A method for constructing and training feedforward neural networks.* Neural Computation, 2:198-209,1990.

Freeman, J.A., Shapura, D.M., *Neural Networks, Algorithms, Applications, and Programming Techniques*, Addison-Wesley Publishing Company, pp.89-114 , 1991.

Gallant, S., *Neural Network Learning and Expert Systems.* MIT Press, Cambridge, USA, 1993.

Goldberg, D.E., *Genetic Algorithm in Search, Optimization and Machine Learning*, Addison-Wesley, 1989.

Grenfenstette, J., Gopal, R., Rosmaita, B., Gucht, D., *Genetic Algorithms for the Traveling Salesman Problem.* In Grenfenstette, J. editor, Proceedings of the Second International Conference on Genetic Algorithms and their Applications. Hillsdale, New Jerse. MIT, Lawrence Erlbaum Associates. 1985.

Harnad, S.,1990 *The symbol grounding problem.* Physica D, 42:335–346.

Hebb, D., *The Organization of Behavior.* John Wiley, New York, 1949.

Hodges, A., *Alan Turing – the Enigma*, Burnetts Books, London, 1983.

Holland, J., *Adaptation in Natural and Artificial Systems*, Ann.Arbor, University of Michigan Presss, (MIT Press, 1992), 1975.

Huns,M.N., Singh, M., *Ontologies of Agents.* IEEE Internet Computing, 1997.

Kohonen, T., *Self-organized Formation of Topologically Correct Feature Maps.* Biological Cybernetics, 43:59-69, 1982.

Koza, J.R., *Genetic Programming: A Paradigm for Genetical Breeding Populations of Computer Programs to Solve Problems*, Stanford University Computer Science Department, technical report STAN-CS-90-1314, 1990.

Langton, C.G., *Artificial Life*. C. Langton (Ed). Addison Wesley, 1989.

Langton, C.G., *Artificial Life: An Overview*. A Bradford Book, The MIT Press, Cambridge, Mass. E London, England, 1995.

Loula, A., Gudwin, R., Queiroz, J., *Comunicação Simbólica entre Criaturas Artificiais: um experimento em vida artificial*. Tese de mestrado. 2004.

Maturana, H., Varela, F., *Autopoiesis: The organization of the living*, Reidel, Dordrecht, 1973.

Michalewicz, Z., *Genetic Algorithms + data structures=evolution programs*, Springer-Verlag, 1996.

Miller, J., *Living Systems*, McGraw Hill, New York, 1978.

Mingers, J., *Self-Producing Systems, Implications and Applications of Autopoiesis*, Plenum Press, New York, 1995.

Mitchel, T., *Machine Learning*. McGraw Hill, 1997.

Moscato, P., Norman, M., *A 'Memetic' Approach for the Travelling Salesman Problem. Implementation of a Computational Ecology for Combinational Optimization of Message-Passing Systems*. In Valero, M. Onate, E., Jane, M., Larriba, J.L., and Suarez, B., editors *Parallel Computing and Transputer Applications*, 187-194, Amsterdam. IOS Press. 1992.

Nwana, H., *Software Agents: an overview*. Knowledge Engineering Review. 205-244. 1996.

Parekh, R., Yang, J., Honovar, V., *Constructive Neural Network Learning Algorithms for Multi-Category Pattern Classification*, TR95-15, Iowa State University, Ames, IA 50011 USA, 1995.

Patterson, D., *Artificial Neural Networks – Theory and Applications*. Prentice Hall, Singapore, 1996.

Peirce, C.S., *Signs and their objects*, Meaning, CP2, 230-232, 1910.

Peirce, C.S., *Collected Papers of Charles Sanders Peirce*. Cambridge: Harvard University Press, 1960.

Priban, I., *Models in Medicine, Systems Behaviour*, Open U. Press, London, 1968.

Rechemberg, I., *Evolutionstrategie: Optimierung Technischer Systeme nach Prinzipien der Biologischen Evolution*. Holzboog Verlag, Stuttgart, 1973.

Riedmiller, M., Braun, H., *A direct adaptive method for faster backpropagation learning: The RPROP algorithm*, Proc. Int. Conf. Neural Networks, San Francisco, 586–591, 1993.

Riedmiller, M., *Supervised Learning in Multilayer Perceptrons – from Backpropagation to Adaptive Learning Techniques*. Computer Standards and Interfaces, 16, 1994.

Rocha, M., *Optimização dos Modelos Conexionistas de Aprendizagem via Computação Genética Evolucionária*, tese de doutoramento, 2004.

Rocha, M, Machado, D., *getAlife - an Artificial Life Environment for the Evaluation of Agent-based Systems and Reinforced Learning Algorithms*, 2004.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., *Learning Representations by Backpropagation Error*, Nature, 323, 533-536, 1986.

Russel, S., Norving, P., *Artificial Intelligence – A Modern Approach*. Prentice Hall. Inc. 1995.

Shoam, Y., *Agent Oriented Programming*, Artificial Intelligence, 60, 51-568, 1993.

Sims, K., *Evolving 3D Morphology and Behaviour by Competition*, Artificial Life IV Proceedings, Brooks & Maes, MIT Press, 28-39, 1994.

Steels, L., *Cooperation Between Distributed Agents through Self-Organisation In "Decentralized AI"*, Elsevier Science Publishers B.V. 175-196, 1990.

Stefik, M., Bobrow, D., *Understanding Computers and Cognition – a new foundation for design*, a review, Artificial Intelligence, 220-226, 1987.

Suchman, L., *Understanding Computers and Cognition – a new foundation for design*, a review, Artificial Intelligence, 227-232, 1987.

Tan, M., *Multi-agent reinforcement learning: Independent vs. cooperative learning*. In M. N. Huhns and M. P. Singh, editors, *Readings in Agents*, pages 487–494. Morgan Kaufmann, San Francisco, CA, USA, 1993.

Taylor, C., Jefferson, D., *Artificial Life as a Tool for Biological Inquiry (Artificial Life : An Overview)*. A Bradford Book, The MIT Press, Cambridge, Mass. E London, England, 1995.

Vellino, A., *Understanding Computers and Cognition: a New Foundation for Design* a review. Artificial Intelligence, 213- 220, 1987.

Von Neumann, J., *Computers and Brain*, Yale U. Press, New Haven, 1958.

Weiss, G., *Multiagent Systems: a modern approach to Distributed Artificial Intelligence*. Ed. MIT Press. Massachusetts, 2000.

Whitley, D., *GENITOR Algorithm and Selection Pressure: why Rank-based Allocation of Reproductive Trials is Best*. Schafer, J.D., editor, Proceedings of the Third International Conference on Genetic Algorithms. George-Mason University, Morgan Kaufman, 1989.

Winograd, T., Flores, F., *Understanding Computers and Cognition*, Addison-Wesley, New York, 1987.

Winograd, T., Flores, F., *Understanding Computers and Cognition: a New Foundation for Design*. Ablex Publishing Corp., 1986.

Widrow, B., Hoff, M. E., *Adaptive switching circuits*, WESCON Convention Record, New York, 1960.

Wooldridge, M., *An Introduction to Multiagent Systems*, John Wiley and Sons, 2002.

Wortmann, H., Szirbik, N., *ICT Issues among Collaborative Enterprises: from Rigid to Adaptive Agent-Based Technologies*, International Journal of Production Planning and Control 12 (5) 452-465, 2001.

Zhang, C., *Cooperation under uncertainty in distributed expert systems*. Artificial Intelligence, 56, 21-69, 1992.