4
Modelo Reinforcement Learning Neuro-Fuzzy Hierárquico para Múltiplos Agentes – RL-NFHP-MA

4.1
Introdução

A idéia central da versão multi-agentes dos algoritmos da família RL-NFH é fazer com que cada agente possa explorar diferentes tarefas ou diferentes pares de estado-ação de forma simultânea, acelerando assim o aprendizado e a convergência para uma política ótima.

Conforme descrito no capítulo 2, existem vários tipos de abordagens envolvendo multi-agentes, do ponto de vista do objetivo a ser aprendido, da coordenação entre os agentes e da homogeneidade do aprendizado. No desenvolvimento da proposta do modelo RL-NFHP-MA, buscou-se estender o modelo RL-NFHP de forma flexível para contemplar a maior gama possível de aplicações.

Neste capítulo, serão explorados os diferentes tipos de sistemas RL-NFHP-MA criados a partir da versão original. Com este objetivo, os SMA propostos serão apresentados em detalhes, descrevendo seu funcionamento, particularidades e objetivos. Em seguida, para cada modelo proposto, serão exploradas as diferenças no modelo de aprendizado, na coordenação e na dinâmica do aprendizado de maneira detalhada.

Entretanto, antes de apresentar os novos SMA da família RL-NFH, este capítulo apresenta, na próxima seção, os princípios de *Satisfatoriedade* e *Não-Dominação* propostos por Goodrich e Quigley (2004), os quais foram incorporados ao modelo RL-NFHP (Figueiredo, 2003), com o objetivo de melhorar o desempenho do modelo.

PUC-Rio - Certificação Digital Nº 0610800/CA

4.2
Princípios de *Satisfatoriedade* e *Não-Dominação*

Conforme descrito na seção *3.4.2-Conseqüentes das Regras do Modelo RL-NFHP*, os conseqüentes das regras fuzzy nas células RL-NFHP são as ações que devem ser identificadas através de aprendizado por reforço. Quando as células se tornam ativas, as ações são selecionadas, cada uma relativa à combinação das partições *baixo* e *alto* de cada uma das entradas da célula. A seleção ocorre em função de valores atribuídos a cada uma das ações que pertencem ao conjunto de ações disponíveis para cada polipartição baixa e alta.

A atualização destes valores é realizada através do algoritmo SARSA (Sutton e Barto, 1998), conforme descrito na seção 3.4.2. Uma das principais limitações do algoritmo é a necessidade de explorar um grande número de vezes todos os possíveis pares de estado-ação para garantir a convergência para uma política ótima. Na prática, muitas vezes isso não é viável. Logo, o algoritmo se restringe a convergir com maior probabilidade para valores-Q que correspondam a políticas aceitáveis. Além disso, o método convencional do *Q-learning* também necessita que os valores de recompensa e punição sejam escolhidos de forma que um não mascare o outro, e vice-versa. Geralmente, isso demanda um longo processo de ajustes (Goodrich e Quigley, 2004).

Visando superar estas limitações e dificuldades do algoritmo, este trabalho propõe uma modificação nos conseqüentes da estrutura RL-NFHP, através do conceito de *satisfatoriedade*. O termo foi usado primeiramente por Simon (1996) no contexto de tomada de decisões para definir ações que sejam "boas o suficiente". De uma maneira geral, são ações em que a recompensa esperada supera o custo. Obviamente, é necessário que as medidas de recompensa e custo esperados sejam avaliadas através de uma mesma métrica, para que possam ser comparadas.

Um conjunto de ações *satisfatórias* em um estado θ é dado por $S(\theta) = \{a: \mu_A(a, \theta) > \mu_R(a, \theta)\}$, onde μ_A indica o que é aceitável, ou o grau que uma ação tem de alcançar recompensas, e μ_R indica o que rejeitável, ou o grau que uma ação tem de alcançar penalidades. Supondo que seja possível aprender μ_A e μ_R de maneira independente e que ambos sejam comparáveis, o conjunto de ações

PUC-Rio - Certificação Digital Nº 0610800/CA

satisfatórias S(θ) pode ser definido e usado como base para exploração do ambiente, evitando que escolhas ruins de ações sejam feitas.

Em paralelo, com o objetivo de maximizar o valor de utilidade na seleção das ações, o princípio da **não-dominação** pode ser usado. Uma ação é *dominada* se existir outra ação com uma maior *aceitabilidade* e menor *rejeitabilidade*. Os princípios de *satisfatoriedade* e *não-dominação* podem ser combinados para refinar o conjunto de ações justificáveis. Ou seja, as ações satisfatórias (recompensa esperada maior que penalidade) e não-dominadas (nenhuma outra ação possui recompensa maior e penalidade menor) compõem o conjunto das ações fortemente satisfatórias, e que devem ser exploradas (Goodrich e Quigley, 2004).

Para viabilizar os princípios acima descritos, a estrutura de reforços deve ser definida de forma que r(a, θ) > 0 indique que o agente atingiu algum objetivo, e r(a, θ) < 0 indique que o agente realizou alguma ação indesejável, onde r(a, θ) é o reforço da ação a no estado θ. O algoritmo de atualização dos valores-Q deve ser alterado de forma que seja criada uma estrutura separada para as recompensas e penalidades. A estrutura das recompensas, denominada função-G (*Goal*), utiliza a mesma equação do *Q-learning* para atualização, porém, apenas as recompensas afetam sua estimativa. Já, a estrutura das penalidades, função-L (*Loss*), é sensibilizada apenas pelas penalidades, mas também usa a mesma estrutura do *Q-learning*.

$$G(s_t, a_t) = (1 - \alpha_t)\, G(s_t, a_t) + \alpha_t [\, r_{t+1} + \gamma\, G(s_{t+1,}\, a_{t+1})] \tag{30}$$

$$L(s_t, a_t) = (1 - \alpha_t)\, L(s_t, a_t) + \alpha_t [\, - r_{t+1} + \gamma\, L(s_{t+1,}\, a_{t+1})] \tag{31}$$

Para calcular μ_A e μ_R, divide-se o valor de G(a, θ) ou L(a, θ) de uma determinada ação em um determinado estado pela soma de todos os G e L de todas as ações no mesmo estado θ.

$$\begin{aligned} \mu_A(a,\theta) &= \frac{G(a,\theta)}{\sum_a G(a,\theta)} \\ \mu_R(a,\theta) &= \frac{L(a,\theta)}{\sum_a L(a,\theta)} \end{aligned} \tag{32}$$

Para um melhor entendimento, a Tabela 3 mostra um exemplo numérico, onde, em um determinado estado, existem 4 ações disponíveis.

PUC-Rio - Certificação Digital Nº 0610800/CA

Tabela 3: Exemplo da forma de cálculo de μ_a e μ_r.

Ações	a_1	a_2	a_3	a_4	$\sum_a$
L	0,4	0,6	0,5	0,3	**1,8**
G	0,8	0,4	0,5	0,7	**2,4**
μ_a	0,33	0,17	0,21	0,29	-
μ_r	0,22	0,33	0,28	0,17	-

Dado o exemplo, as ações que podem ser consideradas *satisfatórias* são aquelas em que a o grau de *aceitabilidade* é maior que o grau de *rejeitabilidade* ou $\mu_A > \mu_R$. Neste caso, ações satisfatórias são 1, 3 e 4. Já, as ações *não-dominadas* são aquelas em que nenhuma outra ação possui, ao mesmo tempo, *aceitabilidade* maior e *rejeitabilidade* menor. Neste caso, seriam as ações 1 e 4. As ações 2 e 3 são dominadas tanto pela ação 1, como pela 4.

Com a introdução dos princípios da *satisfatoriedade* e *não-dominação*, os conseqüentes do modelo RL-NFHP passam a possuir uma nova estrutura, conforme mostrado na Figura 21.

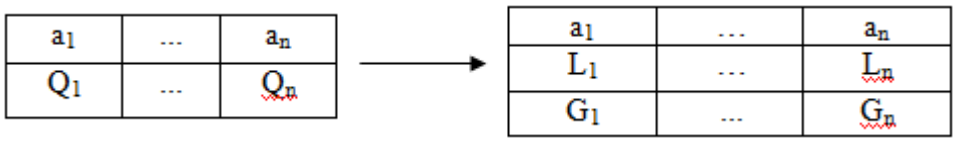

Figura 21: Subdivisão da Célula RL-NFP.

Devido aos benefícios sugeridos nesta seção, os princípios propostos por Goodrich e Quigley (2004) foram incorporados ao modelo RL-NFHP original e estendidos para os modelos RL-NFHP-MA, descritos a seguir.

4.3
Sistemas RL-NFHP-MA

Os sistemas RL-NFHP-MA propostos diferem entre si em dois aspectos principais: a *dinâmica do aprendizado* e o *mecanismo de coordenação*.

PUC-Rio - Certificação Digital Nº 0610800/CA

4.3.1
Dinâmica do Aprendizado

Em relação à dinâmica de aprendizado dos múltiplos agentes, ela pode ocorrer basicamente de duas maneiras: com ou sem o compartilhamento da estrutura de aprendizado RL-NFH.

Havendo o compartilhando, a estrutura de aprendizado é única e igual para todos os agentes. Ou seja, cada agente pode executar uma determinada ação em um estado distinto do problema e alimentar a estrutura com o conhecimento adquirido. Os agentes seguintes acessam a informação "depositada" pelos agentes anteriores na estrutura única, selecionam a ação a ser tomada e atualizam os valores de Q de acordo com o retorno recebido.

Vale ressaltar que este tipo de modelo, onde múltiplos agentes utilizam a mesma estrutura, é indicado para problemas em que os agentes precisam aprender uma mesma tarefa e buscam a cooperação. Se todos competissem utilizando exatamente a mesma "inteligência" não haveria perdedores ou vencedores. Além disso, em um ambiente de competição, não faz sentido um agente fornecer o conhecimento obtido aos demais competidores. A figura 22 ilustra o funcionamento do aprendizado coletivo dos agentes através do modelo RL-NFHP para multi-agentes.

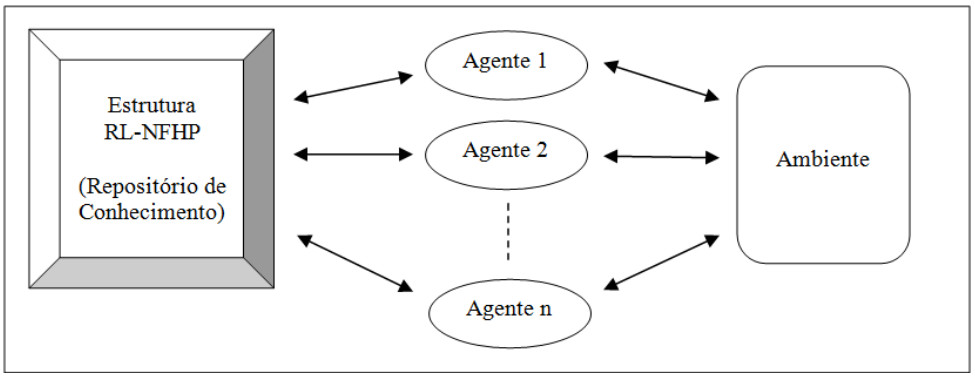

Figura 22: Compartilhamento do conhecimento e exploração do ambiente pelos multi-agentes.

A outra forma de aprendizado é quando cada agente mantém sua própria estrutura, independente dos demais agentes presentes no ambiente. Neste caso, cada agente se especializa em uma determinada tarefa, com o objetivo de cooperar

PUC-Rio - Certificação Digital Nº 0610800/CA

ou competir. Esta abordagem visa contemplar problemas que envolvam tanto a cooperação, quanto a competição entre múltiplos agentes.

Para que ocorra a cooperação entre os agentes, deve existir um mecanismo de coordenação, ou até mesmo um agente central, recebendo e combinando as informações de cada estrutura RL-NFHP. A figura 23 mostra um esquema de sistema onde cada agente mantém sua própria estrutura de aprendizado e um agente central recebe e combina as informações aprendidas.

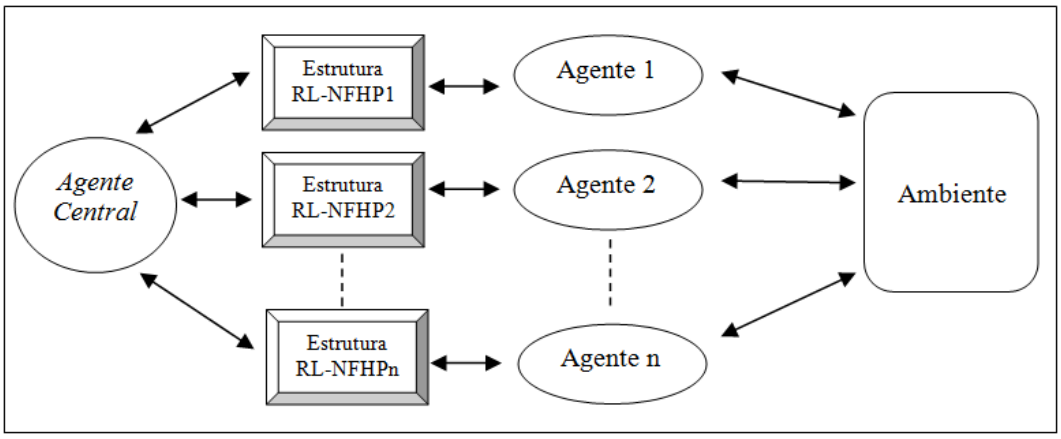

Figura 23: Sistema RL-NFHP-MA em que os agentes possuem diferentes estruturas de aprendizado.

A representação da arquitetura de um sistema RL-NFHP-MA com diferentes estruturas está ilustrada na Figura 24. No exemplo, um "agente central" recebe os valores de saída de dois agentes RL-NFHP com *n* entradas e as combina de acordo com as necessidades da aplicação. O "agente central" pode ser desde uma mera combinação das saídas, como no caso abaixo, até uma outra estrutura RL-NFHP, que use como entradas as saídas dos demais agentes.

PUC-Rio - Certificação Digital Nº 0610800/CA

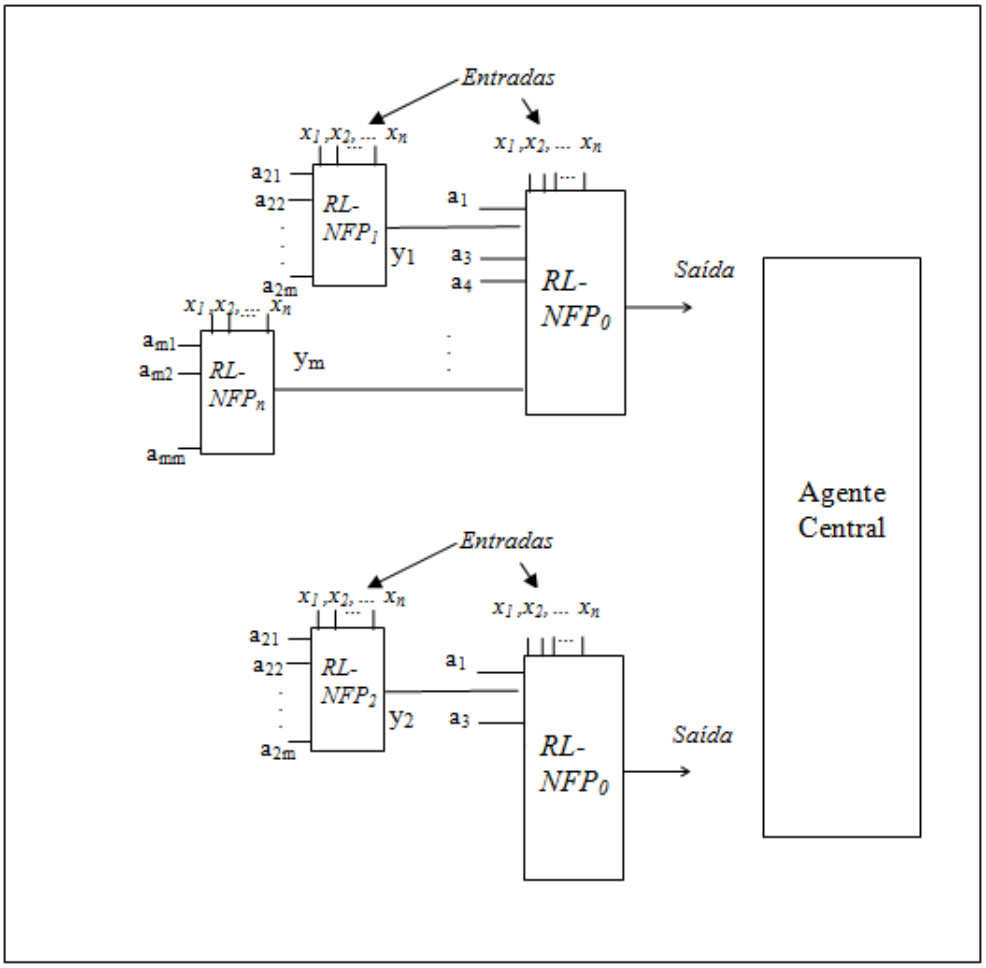

Figura 24: Arquitetura RL-NFHP-MA com diferentes estruturas e um Agente Central.

Nos casos de problemas competitivos, os agentes podem utilizar suas próprias estruturas de forma independente, com o objetivo de aprender uma tarefa específica, e competir uns contra os outros, sendo que cada agente está igualmente "equipado" com uma estrutura de aprendizado RL-NFHP.

O modelo permite também a competição entre um grupo de múltiplos agentes RL-NFHP-MA, coordenados por um agente central, e agentes individuais, ou outros grupos de múltiplos agentes.

4.3.2
Mecanismo de Coordenação entre os agentes

A coordenação dos objetivos individuais dos agentes pode ocorrer basicamente de duas formas:

PUC-Rio - Certificação Digital Nº 0610800/CA

- **Coordenação explícita:** um mecanismo de coordenação pode ser incluído, por exemplo, para definir qual o objetivo do agente selecionado naquele passo. A figura 25 mostra a coordenação sendo realizada através de um agente central.

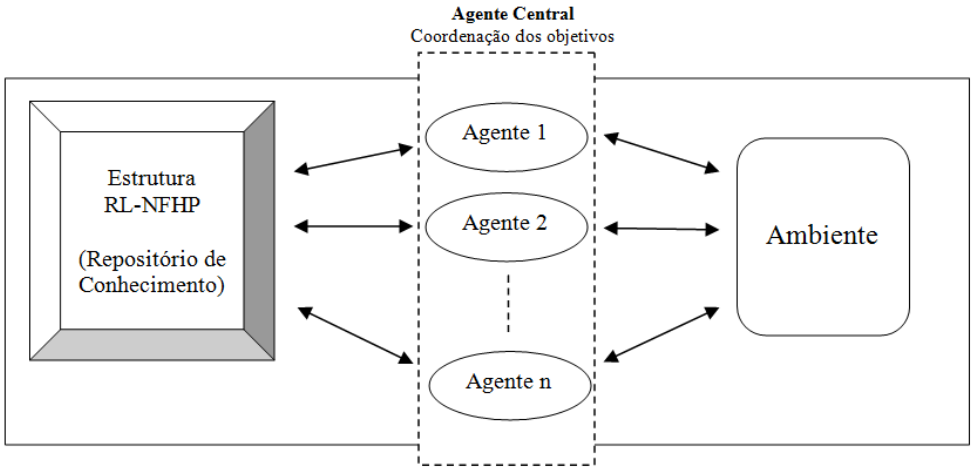

Figura 25: Sistema RL-NFHP-MA em que os agentes usam a mesma estrutura de aprendizado.

- **Coordenação implícita**: não havendo um mecanismo de coordenação global, a segunda possibilidade é fazer com que os agentes redefinam suas estratégias a cada rodada, sem comunicação com os demais, de acordo com a sua percepção do ambiente.

Para melhor entendimento e análise das diferenças na forma de coordenação, será usado como exemplo o jogo da Presa/Predador (Benda et al., 1986). Em sua forma mais tradicional, participam do jogo 4 predadores e 1 presa, em um grid ortogonal, dividido por linhas e colunas, onde cada espaço no grid não pode ser ocupado por dois participantes do jogo simultaneamente. O objetivo do jogo é fazer com que os predadores capturem a presa da forma mais rápida possível. A figura 26 mostra o funcionamento do jogo e como uma captura ocorre.

PUC-Rio - Certificação Digital Nº 0610800/CA

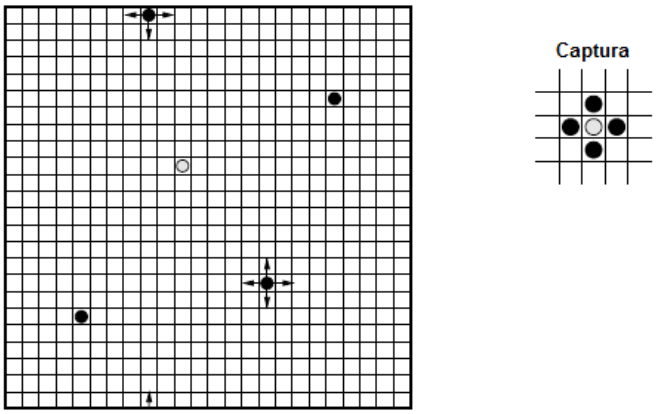

Figura 26: Grid ortogonal mostrando o esquema do Jogo da Presa/Predador e a captura da presa.

Para concretizar a captura, é importante que cada agente faça o cerco à presa em uma posição distinta (acima, abaixo, à direita e à esquerda). Sendo a coordenação realizada por um agente central, a cada rodada, este agente informará aos agentes filhos ou "escravos" a posição alvo que cada um deve chegar, com base nas informações gerais do ambiente, como a distância de cada agente em relação à presa. Esse tipo de coordenação está ilustrado na figura 27. Os pontos pretos representam os predadores e o ponto claro a presa.

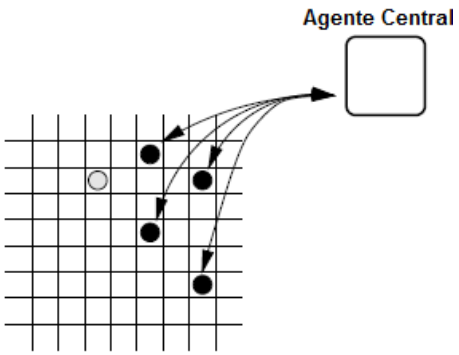

Figura 27: Representação da coordenação realizada por um agente central no jogo da presa-predador.

No caso da ausência de um agente central, cada agente determina seu ponto de chegada baseado nas informações que chegam através de seus próprios sensores, conforme mostra a figura seguinte. Ou seja, de acordo com sua percepção da localização da presa e dos demais agentes, ele define seu ponto de chegada que seria a melhor forma de contribuir para a captura. Como a presa e os

PUC-Rio - Certificação Digital Nº 0610800/CA

demais agentes se movimentam ao longo do jogo, o objetivo do agente pode ser redefinido a cada rodada.

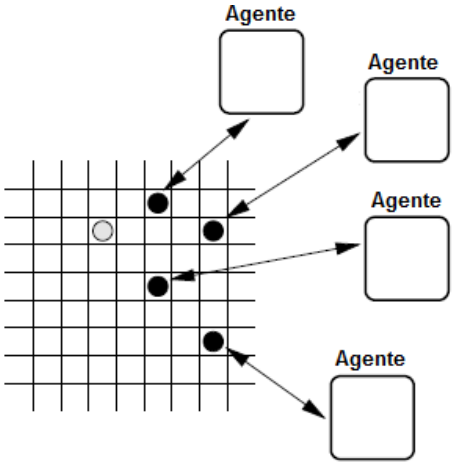

Figura 28: Representação do jogo da presa-predador sem a figura de um agente central.

O objetivo do agente em um determinado passo do processo de aprendizado afeta diretamente a função de avaliação/retorno. Uma mesma ação executada em um determinado estado, por um mesmo agente, pode ter diferentes valores de reforço, de acordo com o seu objetivo naquele passo do aprendizado. Por exemplo, dependendo da posição e, consequentemente, dos objetivos dos demais agentes, ele pode ter como alvo capturar a presa pela esquerda, pela direita, por cima ou por baixo. Ou seja, a função de avaliação deverá retornar um valor diferente de acordo com o objetivo do agente, que pode mudar a cada rodada. Por isso, a definição de um bom mecanismo de coordenação é extremamente importante nos SMA.

Vale ressaltar ainda que, durante o jogo, a Presa pode se mover de maneira aleatória, ou pode possuir um mecanismo de aprendizado, sendo mais um agente inteligente no jogo. Neste caso, o objetivo da presa é aprender a buscar sempre movimentos que a deixem o mais longe possível dos agentes predadores, caracterizando um problema de competição entre um único agente, a Presa, e múltiplos agentes, os Predadores. Ambos podem utilizar os modelos da família RL-NFHP-MA ou não, para adaptar seu comportamento no jogo. A figura 29 ilustra a representação do jogo com a presa sendo também controlada por um "agente inteligente".

PUC-Rio - Certificação Digital Nº 0610800/CA

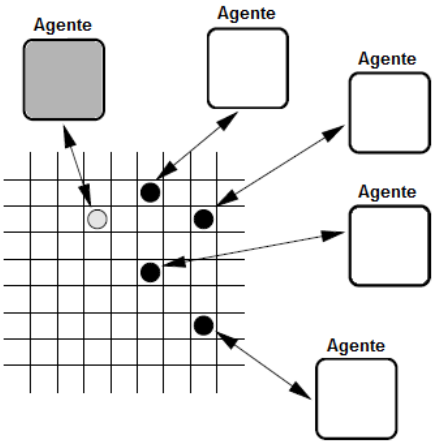

Figura 29: Representação do jogo da presa-predador em que a Presa também é um Agente Inteligente.

O jogo da presa e predador será explorado com mais detalhes no próximo capítulo, Estudo de Casos.

PUC-Rio - Certificação Digital Nº 0610800/CA

4.3.3
Tipos de Sistemas RL-NFHP-MA e Objetivos

Resumidamente, pode-se dizer que os sistemas RL-NFHP-MA projetados com uma única estrutura de aprendizado para todos os agentes têm como principal objetivo a exploração paralela de diferentes estados e ações do ambiente, de maneira cooperativa..

Já, nos sistemas construídos com uma estrutura distinta RL-NFHP para cada agente, a grande finalidade é a especialização dos agentes no aprendizado de determinadas tarefas, e posterior combinação do conhecimento obtido por todos, para a realização de uma tarefa maior. Nestes casos, as aplicações podem ser de natureza cooperativa ou competitiva.

Nos dois casos acima, pode haver ou não um mecanismo central explícito de coordenação entre os agentes.

A Tabela 4 fornece um resumo geral dos diferentes sistemas RL-NFHP-MA propostos nesta tese, comparando seus mecanismos de aprendizado e coordenação, tipos de interação e objetivos.

Tabela 4: Quadro com uma síntese dos diferentes modelos RL-NFHP-MA.

Aprendizado	Coordenação	Tipo de Interação	Objetivo
Estrutura de conhecimento compartilhada	Implícita	Cooperação	Exploração paralela do ambiente
	Explícita (com Agente Central)		
Estruturas de aprendizado independentes	Implícita	Cooperação	Especialização em tarefas distintas
	Explícita (com Agente Central)	Competição	

4.4
Algoritmo de Aprendizado

Nos casos em que os agentes utilizam estruturas independentes, o algoritmo de aprendizado funciona de maneira semelhante à versão voltada para um agente único. A grande diferença é que a saída obtida de cada modelo pode ser combinada de diversas maneiras, dependendo do problema em questão. Ou ainda, podem ser utilizadas como entradas para uma nova estrutura RL-NFHP.

Já, nas situações em que os agentes compartilham uma mesma estrutura como repositório de conhecimento, a grande diferença é a alternância dos agentes, que ocorre de forma cíclica na exploração do ambiente. Inicialmente, os agentes acessam a estrutura para tomar uma decisão, exploram os pares de estado-ação do ambiente, recebem o retorno e repassam "o que aprenderam" à estrutura de conhecimento, que processa as informações do par estado-ação explorado e o retorno recebido. Em seguida, um novo ciclo é realizado para outro agente, que pode estar explorando um espaço completamente distinto do ambiente, em relação ao agente anterior.

O algoritmo de aprendizado da estrutura utiliza os mesmos mecanismos do modelo RL-NFHP para a seleção das ações, a atualização dos valores de Q, o particionamento da estrutura, etc. O processo completo de treinamento é dividido em sete etapas, exibidas na figura abaixo.

PUC-Rio - Certificação Digital Nº 0610800/CA

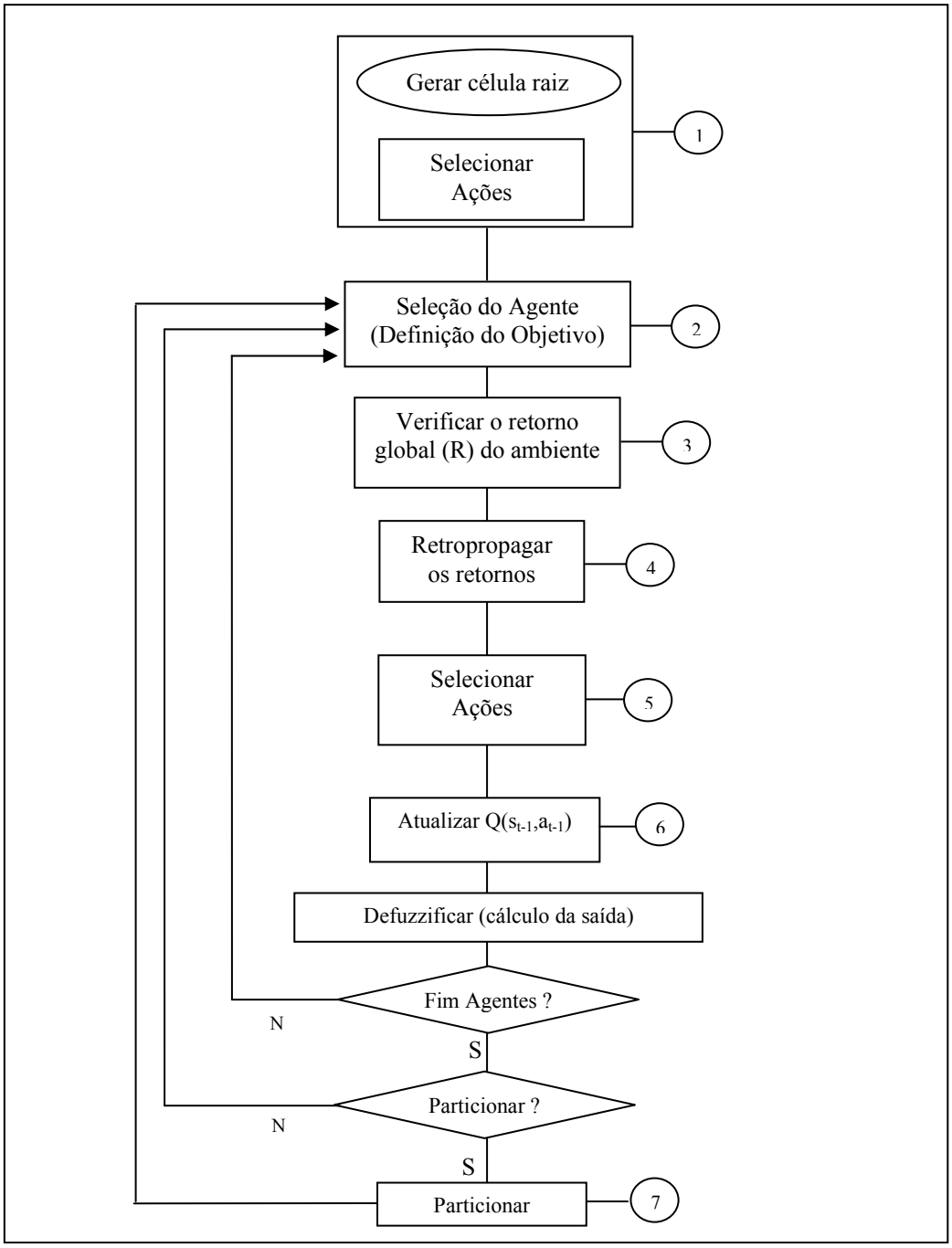

Figura 30: Algoritmo de aprendizado do modelo RL-NFHP-MA.

1- Inicialização – ocorre basicamente da mesma forma que na versão para um único agente. Uma célula raiz é criada, tendo como domínios dos seus conjuntos fuzzy, relativos a cada uma das entradas, os valores mínimo e máximo destas entradas (*Limite Inferior* e *Limite Superior*). Com o objetivo de generalidade, os valores das variáveis de entrada são normalizados. Os valores correspondentes às variáveis de entrada da célula são lidos do ambiente,

PUC-Rio - Certificação Digital Nº 0610800/CA

normalizados e podem ser aplicados diretamente às entradas da célula ou ser modificados segundo uma função que os torne adequados às variáveis das entradas da célula. Estes valores são avaliados nos conjuntos fuzzy *baixo* e *alto*, resultando em dois graus de pertinência, $\rho(x_1),\mu(x_1)$; ... $\rho(x_n),\mu(x_n)$; respectivamente, para cada variável de entrada. Cada uma das polipartições escolhe uma das ações de seu conjunto de ações baseando-se nos métodos descritos na seção 4.5 (passo 5 do algoritmo – Seleção das Ações). A saída da célula é calculada pelo processo de defuzzificação e representa a ação que será executada pelos *atuadores* do agente.

2- Seleção do Agente – nesta etapa, um dos agentes no ambiente é selecionado para tomar uma ação e ter seu retorno avaliado. Os agentes são numerados de forma aleatória. A cada ciclo, a seleção é realizada de forma seqüencial, sempre do agente 1 ao agente n, para que todos possam interagir igualmente com o ambiente. Na grande maioria dos problemas em que os SMA são aplicados, os agentes possuem um objetivo a ser realizado. Quando o objetivo é alcançado ao longo do processo de aprendizado, os agentes podem ser reinicializados, voltando para um determinado estado de origem, ou para algum estado selecionado de forma aleatória. O aprendizado prossegue normalmente com a movimentação do próximo agente.

Além disso, nesta etapa, o agente selecionado também poderá definir/redefinir seu objetivo durante o aprendizado. Por exemplo, em caso de problemas em que cada agente possui um objetivo distinto, estes objetivos podem ser alterados constantemente ao longo da dinâmica do aprendizado, devido à movimentação dos agentes. Não há uma regra única para a determinação dos objetivos. Ela varia de acordo com o problema ou aplicação que está sendo abordada. Alguns exemplos serão vistos no capítulo 5.

3- Função de Avaliação/Retorno - Após a execução de uma ação por um determinado agente, uma nova leitura do ambiente é realizada. Esta leitura permite que seja calculado o valor de reforço do ambiente e se avalie a ação tomada pelo agente. Este valor deve ser calculado através de uma função de

PUC-Rio - Certificação Digital Nº 0610800/CA

avaliação definida segundo os objetivos do agente, ou do grupo de agentes, sendo fundamental para a orientação do agente ao longo do processo de aprendizado.

A grande diferença em relação à versão original do RL-NFHP é que o reforço de uma ação de um agente passa a depender também da situação dos demais agentes no ambiente (ou do seu objetivo naquele momento do jogo). Uma determinada ação em um estado pode ser boa ou ruim, de acordo com o posicionamento dos demais agentes no jogo. Logo, a função de avaliação, além de fundamental no aprendizado, passa a ter também extrema importância na coordenação dos agentes.

4- Retropropagar o retorno – ocorre da mesma forma que na versão para um único agente. Porém, ao invés da retropropagação do retorno ocorrer somente após o final de cada passo do aprendizado, ela ocorre ao final de cada interação de um agente com o ambiente. O retorno é calculado para cada partição de todas as células ativas, mediante a sua participação na ação resultante. Dessa forma, o retorno do ambiente é retropropagado a partir da célula raiz até as células folhas.

5- Seleção das ações – é realizada através dos mesmos métodos usados pelo algoritmo RL-NFHP. O primeiro, o ϵ*-greedy* (Sutton, 1998), seleciona a ação associada à maior função de valor-Q esperada com probabilidade (1-ϵ); e com probabilidade ϵ seleciona aleatoriamente uma ação qualquer.

No segundo método, a ação escolhida com probabilidade (1-ϵ) é também a que apresentar a maior função de valor-Q associada; e com um número de vezes proporcional a ϵ, a seleção é baseada na distribuição de probabilidade dada pelas funções de valores-Q. Esta forma de seleção é mais conservadora, uma vez que as ações que apresentarem as maiores funções de valores-Q terão maiores chances de serem escolhidas. Este tipo de método de seleção de ações é conhecido como *softmax* (Sutton & Barto, 1998).

6- Atualização dos valores Q – no caso de múltiplos agentes, para a atualização dos valores de Q devem ser considerados o retorno global atual e o retorno global anterior sempre do mesmo agente. Só assim é possível avaliar sua ação. Para que isso seja viável, os retornos dos agentes devem permanecer armazenados até que o algoritmo selecione novamente o agente em questão,

PUC-Rio - Certificação Digital Nº 0610800/CA

calcule seu novo retorno, e compare com o seu retorno anterior. Isso evita que retornos de diferentes agentes sejam misturados, causando problemas na atualização dos valores de Q.

Individualmente, cada atualização de valores-Q também ocorre de duas formas distintas: para o caso do valor do retorno global atual ser maior que o retorno global anterior ($R_{t+1} > R_t$) e para o caso do retorno global atual ser menor ou igual ao retorno global anterior ($R_t \geq R_{t+1}$). No primeiro caso, é utilizada a equação do algoritmo *SARSA* (Sutton, 1996), com o objetivo de premiar a ação escolhida. Na segunda situação, as ações escolhidas são punidas, para que se tornem menos propensas a serem selecionadas nas próximas vezes em que as células, às quais elas pertencem, estiverem ativas. A punição ocorre através da mesma equação utilizada pelo modelo RL-NFHP. Além deste método, também foram aplicados os princípios de Não-Dominação e Satisfatoriedade descritos na seção 4.2.

7- Particionamento das células – A decisão de particionar ou não, e o particionamento em si, também ocorre exatamente da mesma forma que na versão para um único agente. Porém, este passo ocorre sempre após uma rodada dos passos 2, 3, 4, 5 e 6 para todos os agentes, e não após cada passo de "Atualização de Valores de Q". Ou seja, para cada agente selecionado, o retorno da ação anterior escolhida é avaliado (passo 3), retropropagado (passo 4), uma nova ação é selecionada (passo 5), e os valores de Q são atualizados (passo 6). Somente após a realização destes cinco passos para todos os agentes é que o passo 7 é executado.

4.5
Outras Considerações sobre os Sistemas RL-NFHP-MA

4.5.1
Homogeneidade do Aprendizado

Nos sistemas RL-NFHP-MA, pode-se dizer que a dinâmica do aprendizado é homogênea, já que os agentes aprendem através do mesmo algoritmo. Mas vale reforçar que, dependendo do tipo de problema, mesmo

PUC-Rio - Certificação Digital Nº 0610800/CA

compartilhando a estrutura de conhecimento, nada impede que os agentes explorem diferentes pares de estados e ações. O potencial da utilização de múltiplos agentes neste tipo de estrutura é justamente a exploração paralela do ambiente, através de diferentes comportamentos.

Por exemplo, supondo um problema com 3 agentes investidores, com diferentes perfis, que utilizem a mesma estrutura de aprendizado RL-NFHP-MA. Um dos agentes possui perfil conservador e opta sempre por explorar ações (investimentos) de baixo risco. O agente com perfil agressivo, por sua vez, busca investimentos de alto risco, atrás da possibilidade de altos retornos. E finalmente, o agente com perfil moderado explora em paralelo estratégias intermediárias.

O ambiente (ou o "mercado"), definido por um grupo de variáveis e indicadores financeiros, é o mesmo para todos os agentes. Porém, dado um conjunto de possíveis ações disponíveis, mesmo utilizando a mesma estrutura RL-NFHP-MA, os agentes irão explorar aquelas que estão de acordo com o seu perfil, e contribuir para a mesma base de conhecimento, informando que estratégias foram boas e ruins em determinados estados. A figura 31 mostra a representação de como seria o SMA descrito acima.

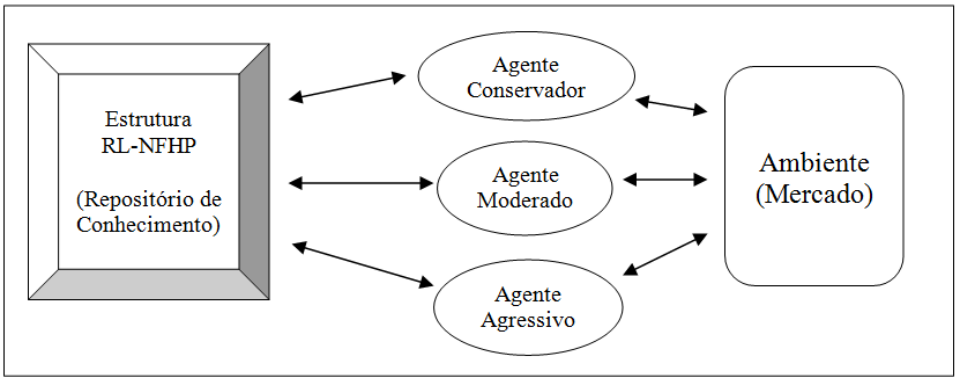

Figura 31: Exemplo de ambiente multi-agente em que agentes possuem comportamentos distintos.

Para definir que estratégias cada agente irá explorar (ou que ações serão escolhidas dado um determinado estado do ambiente), pode haver um agente central controlando os demais agentes e indicando que ações cada um deve realizar. Por exemplo, no jogo da presa/predador descrito anteriormente, um agente central pode determinar que ação cada agente deve tomar para chegar ao seu alvo. No caso dos "agentes investidores", um agente central pode determinar

PUC-Rio - Certificação Digital Nº 0610800/CA

que ação cada agente deve realizar para alcançar um determinado nível de risco/retorno, que equivaleria ao 'alvo do agente'.

Portanto, mesmo sendo um algoritmo com uma dinâmica de aprendizado homogênea, o RL-NFHP-MA permite que o comportamento dos agentes seja distinto, de acordo com as definições originais do problema.

4.5.2
Percepção do Agente

Na ausência de um agente central, que tem domínio e percepção total do ambiente, também é importante a definição do "tamanho da percepção" dos agentes para o modelo RL-NFHP-MA. Os agentes podem possuir desde uma percepção total do ambiente, incluindo a situação dos demais agentes, até uma visão bastante restrita, sendo capaz de reconhecer apenas os espaços de estado mais próximos ao seu no ambiente. Quanto maior e preciso o volume de informações recebido pelo agente, mais fácil será a avaliação do reforço para o par de estado-ação explorado.

No caso do jogo da presa/predador, o agente pode enxergar o grid inteiro e a posição de todos os participantes no jogo, ou apenas posições próximas à sua posição. Na figura abaixo, os círculos ao redor do agente cinza ilustram diferentes possibilidades de percepção do ambiente. Caso o agente fosse capaz apenas de "visualizar" o que ocorre dentro do círculo menor à sua volta, seria impossível avaliar a eficiência de uma ação, já que ele não saberia em que posições estão a presa e os demais agentes. Logo, ele não daria nenhuma contribuição para a estrutura de conhecimento.

PUC-Rio - Certificação Digital Nº 0610800/CA

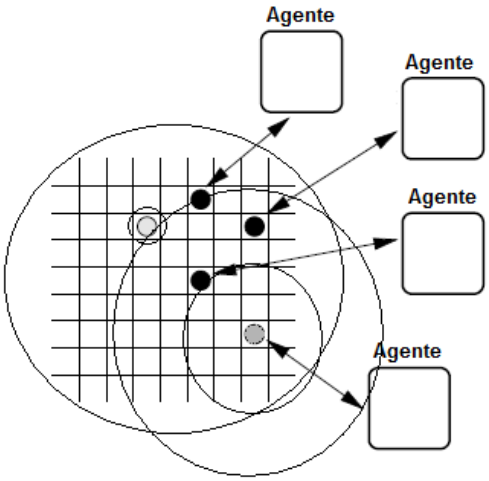

Figura 32: Representação de diferentes níveis de percepção do agente no jogo da presa-predador.

No capítulo seguinte, serão apresentados os estudos de casos de Sistemas Multi-Agentes, que tiveram como objetivo a validação do modelo proposto ao longo do capítulo 4. Três aplicações foram desenvolvidas: o jogo da presa e predador (*pursuit game*), um leilão de energia elétrica e uma aplicação na área de Gerenciamento de Projetos. Os problemas possuem número de variáveis de entradas e graus de complexidade diferentes. Com isso, o desempenho do modelo proposto pode ser comparado com suas versões anteriores para apenas um agente, e com outros sistemas *benchmark*. Além disso, aspectos como a coordenação, a percepção e processo de aprendizado podem ser melhor estudados, conforme será visto no capítulo a seguir.

PUC-Rio - Certificação Digital Nº 0610800/CA