

4 Sistemas Coevolutivos

4.1. Introdução

Após descrito o sistema NFHB e seus parâmetros, é necessário definir como o sistema criado otimiza esses parâmetros para torná-lo inteiramente automático.

A coevolução genética (CG) é uma técnica criada recentemente como uma extensão dos métodos evolucionários tradicionais, e utiliza vários AGs evoluindo em conjunto e trocando informações para atingir um objetivo final.

Essa técnica tem sido utilizada principalmente em problemas envolvendo otimizações complexas, como é o caso deste trabalho. [HERR99] e [JENS01] demonstram exemplos de CG aplicada em problemas de otimização do tipo minimax. Em [PAGI02] é realizado um estudo comparativo entre sistemas de busca evolutivos e coevolutivos, onde foi encontrada eficácia muito maior na solução de problemas utilizando a coevolução.

[POTT00] mostra aplicações da coevolução em problemas de complexidade crescente. Nesse exemplo, os AGs são criados em tempo de execução e sob demanda, sendo eliminados da mesma forma. Os autores demonstram que neste tipo de aplicação os AGs devem surgir e sumir naturalmente na coevolução conforme a sua contribuição para o ecossistema como um todo, ao invés de serem impostos desde o início.

Em [MIK97] foi criado um sistema coevolutivo simbiótico para a criação de redes neurais chamado SANE (*Symbiotic Adaptive Neuro-Evolution*). Esse trabalho serviu de base para diversos outros sistemas coevolutivos.

A maioria das abordagens coevolucionárias da literatura pode ser dividida em duas categorias: as competitivas e as cooperativas. As competitivas normalmente definem AGs hospedeiros e AGs parasitas. Esses AGs competem entre si ao longo da evolução e o sucesso dos hospedeiros implica no fracasso dos parasitas, como nos problemas de otimização do tipo minimax.

Por outro lado, nas abordagens cooperativas os indivíduos das diferentes populações geralmente representam soluções parciais, ou partes da solução completa. A junção dos melhores indivíduos de cada AG compõe a melhor solução global. Este trabalho propõe uma arquitetura que pode ser classificada como cooperativa, como será visto mais adiante.

Na próxima seção será descrita a arquitetura coevolutiva tradicional e, posteriormente, uma versão hierárquica dessa arquitetura.

4.2. O Modelo Coevolutivo Tradicional

Muitos problemas complexos podem ser divididos em subproblemas com características próprias. Dessa forma, é possível atacá-los separadamente seguindo a filosofia do “dividir para conquistar”. A coevolução cooperativa segue essa abordagem, atribuindo a subproblemas AGs criados especificamente para resolvê-los. Aparece, assim, o conceito de espécies que devem evoluir de forma independente, porém trocando informações, ou seja, cooperando entre si. Cada espécie, apesar de pertencer a um ecossistema único, possui características, domínios, parâmetros e comportamento próprios, devendo ser tratada e evoluída por diferentes AGs. Cada AG deve ser configurado de forma específica para explorar seu nicho correspondente.

Modelos coevolucionários apresentam como principais características:

- Aumento da **escalabilidade**, ou seja, a complexidade do problema deixa de ser um entrave ao sucesso da evolução, pois não existe um limite definido de AG's coevolucionários, sendo possível a adequação ao problema;
- Ganho de **desempenho**, pois muitas tarefas realizadas pelos AGs componentes podem ser realizadas em paralelo;
- Ganho em **eficácia**, principalmente em problemas complexos que podem ser divididos em subproblemas menores. AGs específicos podem ser definidos para cada um dos problemas, atacando nichos específicos e obtendo êxito na solução do problema como um todo.

A arquitetura coevolutiva tradicional utilizada na solução de diversos problemas é composta basicamente por diferentes AGs, que evoluem cada uma das espécies envolvidas no problema, e um controlador central que cuida da coordenação dos mesmos. Essa arquitetura pode ser vista na Figura 19.

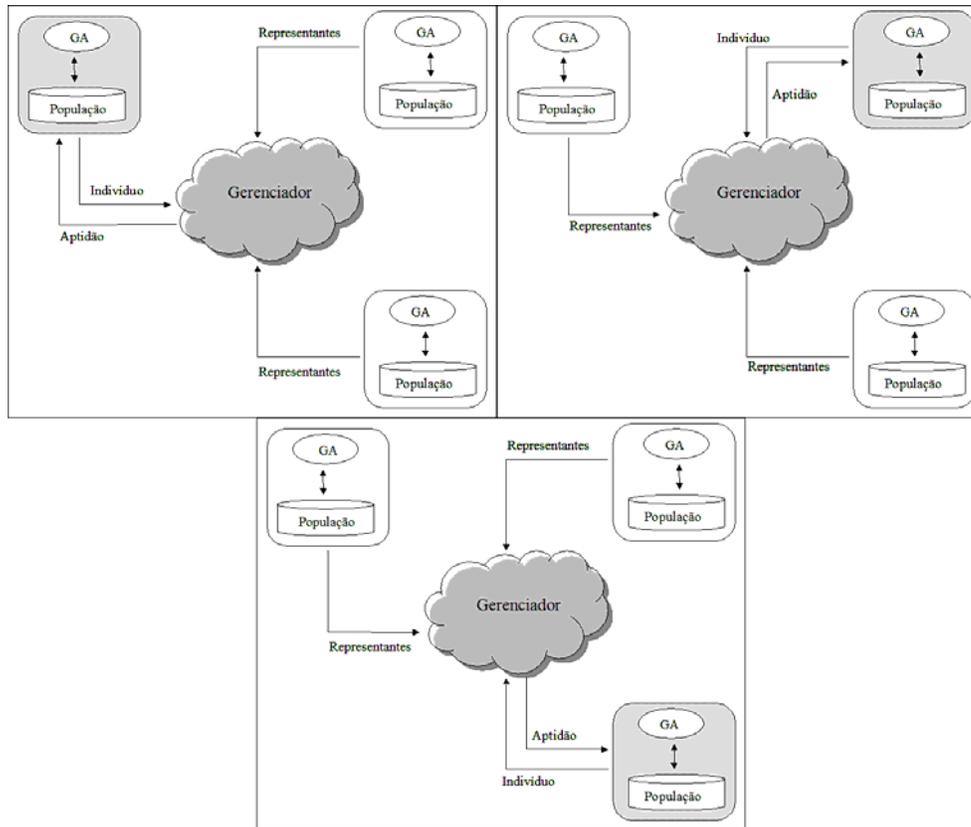


Figura 19: Exemplo de Arquitetura Coevolutiva Tradicional com 3 AG's

As espécies evoluem separadamente, porém a colaboração ocorre no momento em que são calculadas as avaliações dos indivíduos. Como pode ser visto na Figura 19 [POTT00], no momento em que os indivíduos de uma população têm as suas avaliações calculadas, as outras espécies dos outros GA's enviam alguns representantes de sua população para serem usados na avaliação. O cálculo da avaliação ocorre no controlador central para posteriormente ser atribuída ao indivíduo correspondente.

Como são utilizados representantes de outras espécies no cálculo das avaliações de uma população específica, é necessário existir uma política de seleção desses representantes.

Muitas são as estratégias existentes; normalmente os melhores indivíduos das outras populações são selecionados, mas também são usadas escolhas aleatórias, mecanismos de roleta, entre outras [POTT00].

O gerenciador central possui papel fundamental nessa arquitetura, pois, além de controlar o cálculo das aptidões, estabelece o sincronismo entre os AGs componentes, determinando o momento em que são executados os mecanismos de seleção, *crossover*, mutação e ordenação dos indivíduos das

populações. Através da independência existente entre os AGs, normalmente esses são atribuídos a unidades de processamento distintas, sendo papel do gerenciador controlar a concorrência, o sincronismo e, caso existam, recursos comuns aos AGs.

O diagrama de seqüência de um exemplo de iteração de coevolução pode ser visualizado na Figura 20.

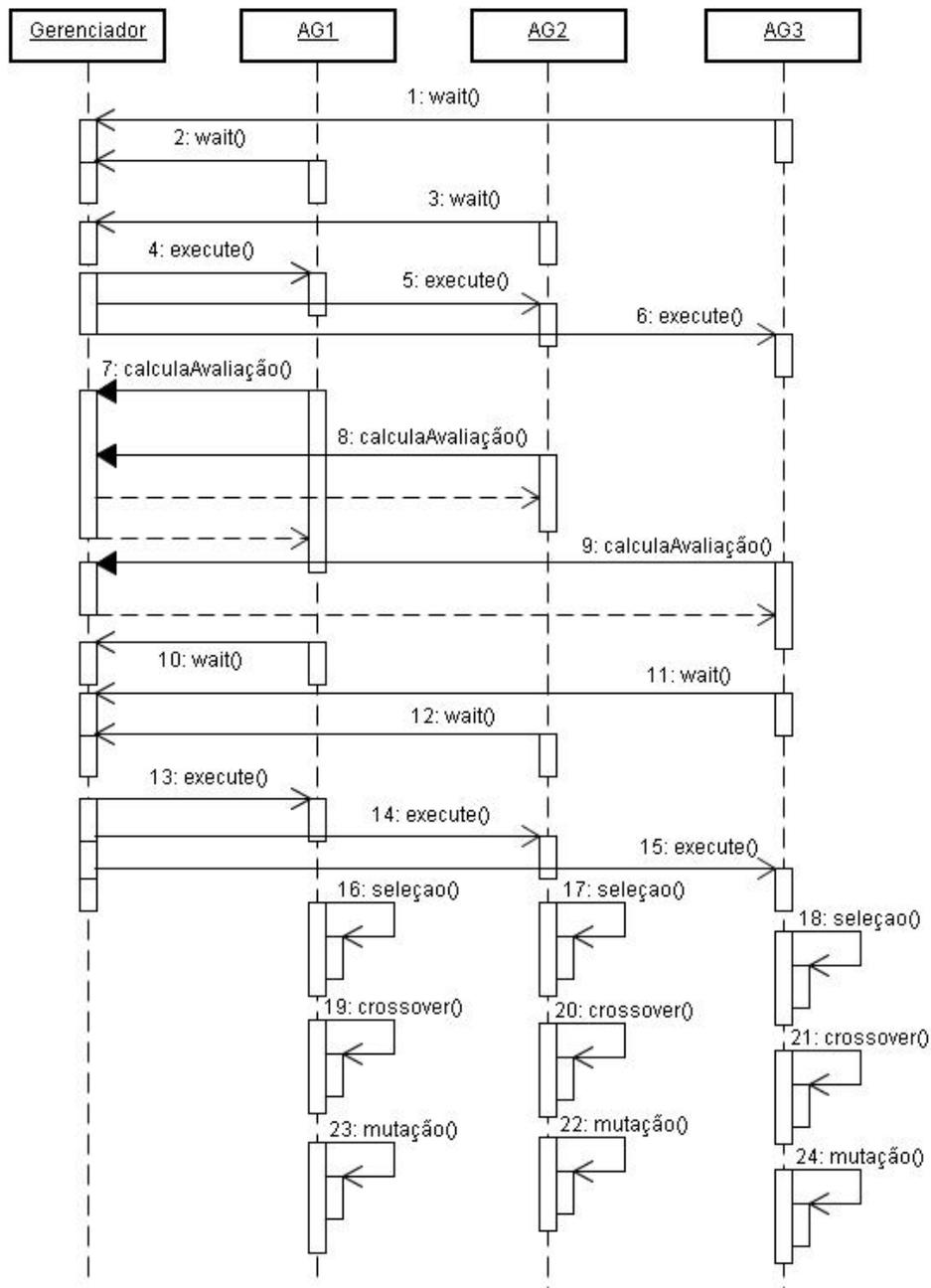


Figura 20: Diagrama de Seqüência de uma Arquitetura Coevolutiva Tradicional

O diagrama de seqüência da Figura 20 segue a especificação da linguagem unificada de modelagem UML (*Unified Modeling Language*). Os retângulos representam as entidades envolvidas, nesse caso, os AGs e o gerenciador. Cada um desses possui a sua linha de vida que define o seu processamento ao longo do tempo. As setas representam as trocas de mensagens entre as entidades, sendo que as tracejadas indicam que existe um retorno para a chamada correspondente.

No momento do cálculo das avaliações, o gerenciador cria uma “barreira” às unidades de processamento para que todos os AGs cheguem nesse ponto da evolução e fiquem aguardando em modo de espera (*wait*). Quando todos os AGs se encontram nesse estado, nenhuma população está sendo modificada e o gerenciador libera os AGs para o cálculo das avaliações de todos os indivíduos de todas as populações. Após o cálculo das avaliações, uma nova barreira é realizada para evitar novamente uma possível alteração em alguma população enquanto avaliações estão sendo calculadas. Quando todos os AGs estão novamente em modo de espera, o gerenciador os libera para o processamento paralelo das operações de seleção, *crossover*, mutação e ordenação. Todo esse sincronismo é realizado para evitar que populações evoluam mais rapidamente que outras, gerando resultados inconsistentes.

Essa abordagem que utiliza processamento com unidades independentes de processamento (*threads*) pode obter melhor performance em casos onde existem vários processadores e até mesmo quando são realizadas operações que independem de processamento para serem executadas (ex.: operações de entrada/saída como leitura de arquivo). Dessa forma, o uso de múltiplos *threads* na implementação da coevolução é indicada, pois ocorre no mínimo na mesma velocidade que a abordagem síncrona, além de ser semanticamente mais adequada em sua modelagem.

4.3. Arquitetura Coevolutiva Hierárquica

Variações no modelo coevolutivo tradicional, muitas vezes, são realizadas para adequação a problemas específicos. [DELG01] introduziu a idéia de estabelecer uma relação de importância entre os AGs coevolutivos, surgindo assim os sistemas coevolutivos hierárquicos.

Nos sistemas coevolutivos hierárquicos, normalmente os AGs de nível inferior na hierarquia dependem dos de nível superior. Dessa forma, os

indivíduos componentes das espécies do topo da hierarquia representam a solução global e devem ser avaliados em primeiro lugar. Os indivíduos das espécies inferiores têm as suas avaliações calculadas a partir das avaliações dos seus pais na hierarquia.

Normalmente, os indivíduos do topo da hierarquia fazem referência a alguns indivíduos das populações inferiores que, ao serem selecionados, compõem a solução global. Nada impede que existam vários níveis de hierarquia e que, conseqüentemente, indivíduos de níveis abaixo do topo possuam também referências a indivíduos mais abaixo na hierarquia, como pode ser visto na Figura 21.

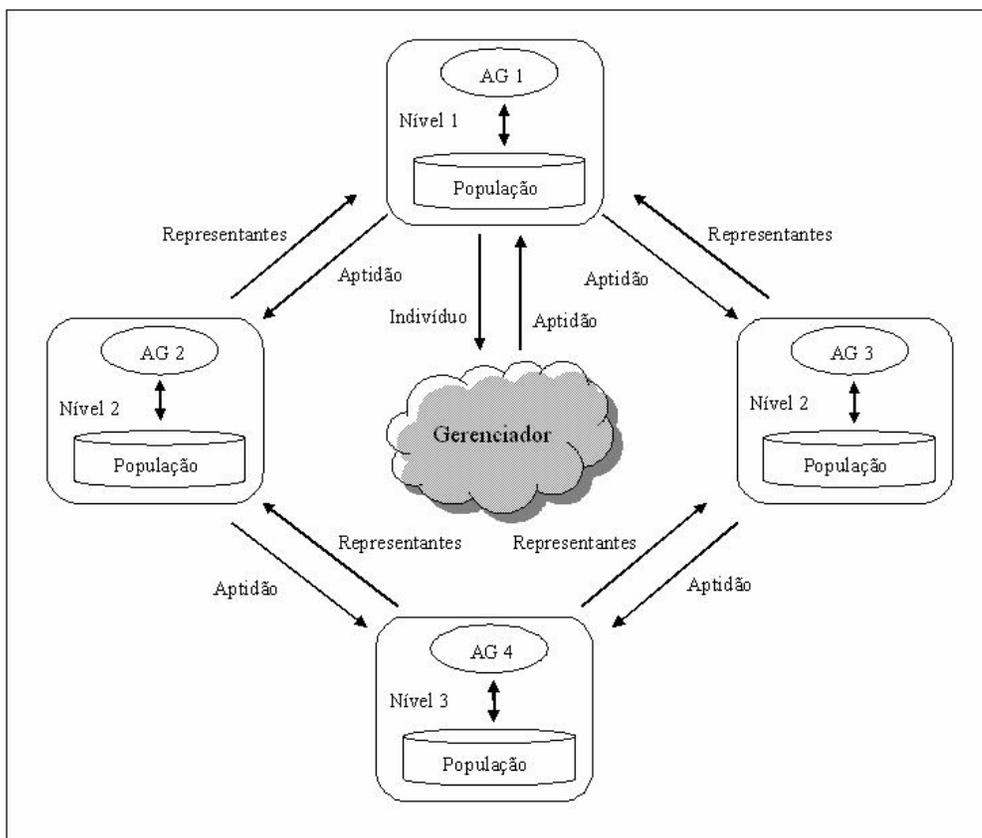


Figura 21: Exemplo de Arquitetura Coevolutiva Hierárquica com 4 AG's e 3 Níveis

No exemplo da figura, os AGs do nível 2 da hierarquia fornecem representantes para compor indivíduos do AG do nível 1 (nível principal). Os indivíduos do nível 1, juntamente com as referências aos representantes do nível inferior, estrategicamente selecionados, podem ser avaliados pelo gerenciador central. As aptidões desses indivíduos são repassadas aos indivíduos do nível 2 para que suas aptidões sejam também calculadas.

A mesma relação ocorre entre os níveis 2 e 3, onde representantes do nível 3 são referenciados por indivíduos do nível 2. As aptidões calculadas no nível 2 também são repassadas ao nível 3 para o cálculo das correspondentes aptidões dos representantes participativos. Dessa forma, percebe-se que todos os níveis são utilizados na formação de uma solução global para o problema, no momento em que os indivíduos são enviados ao gerenciador para avaliação.

O sincronismo na arquitetura coevolutiva hierárquica é um pouco diferente do caso tradicional. Neste, o gerenciador não pode apenas criar uma barreira para o cálculo das avaliações. Além disso, as avaliações das espécies de níveis inferiores dependem das avaliações de níveis superiores e, portanto, as avaliações dos níveis superiores devem ser calculadas em primeiro lugar.

A primeira barreira ocorre da mesma forma que no caso tradicional, interrompendo o processamento dos AGs no momento do cálculo das avaliações. Em seguida, as avaliações dos AGs são calculadas no sentido *top-down* da hierarquia. No exemplo demonstrado na Figura 21, primeiramente são calculadas as avaliações dos indivíduos da espécie de nível 1, posteriormente do nível 2 e, finalizando, as avaliações do nível 3. Após calculada a avaliação, o AG de nível superior envia suas avaliações aos representantes de nível inferior que participaram do cálculo. Todo AG, após ter suas avaliações calculadas, fica em espera por um sinal do gerenciador para recomeçar o seu processamento. Depois de calculadas as avaliações do último nível, todos AGs são liberados para executarem as operações de seleção, *crossover*, mutação e ordenação em paralelo gerando as novas populações da iteração seguinte.

A Figura 22 demonstra um diagrama de seqüência representando o momento do cálculo das avaliações dos AGs na coevolução hierárquica. No exemplo, existem três AGs, onde o AG1 está no topo da hierarquia e os outros dois se encontram em um nível inferior.

Ao final do processo coevolutivo hierárquico, pode-se obter a melhor solução global de duas formas distintas, dependendo do problema. Quando os cromossomas dos indivíduos da população principal possuem como genes referências aos indivíduos das populações inferiores, a melhor solução é representada pelo melhor indivíduo da população principal e seus respectivos indivíduos referenciados. Quando não existe relação implícita no cromossoma entre os níveis da hierarquia, simplesmente são utilizados os melhores indivíduos de todas as populações para formar a solução final.

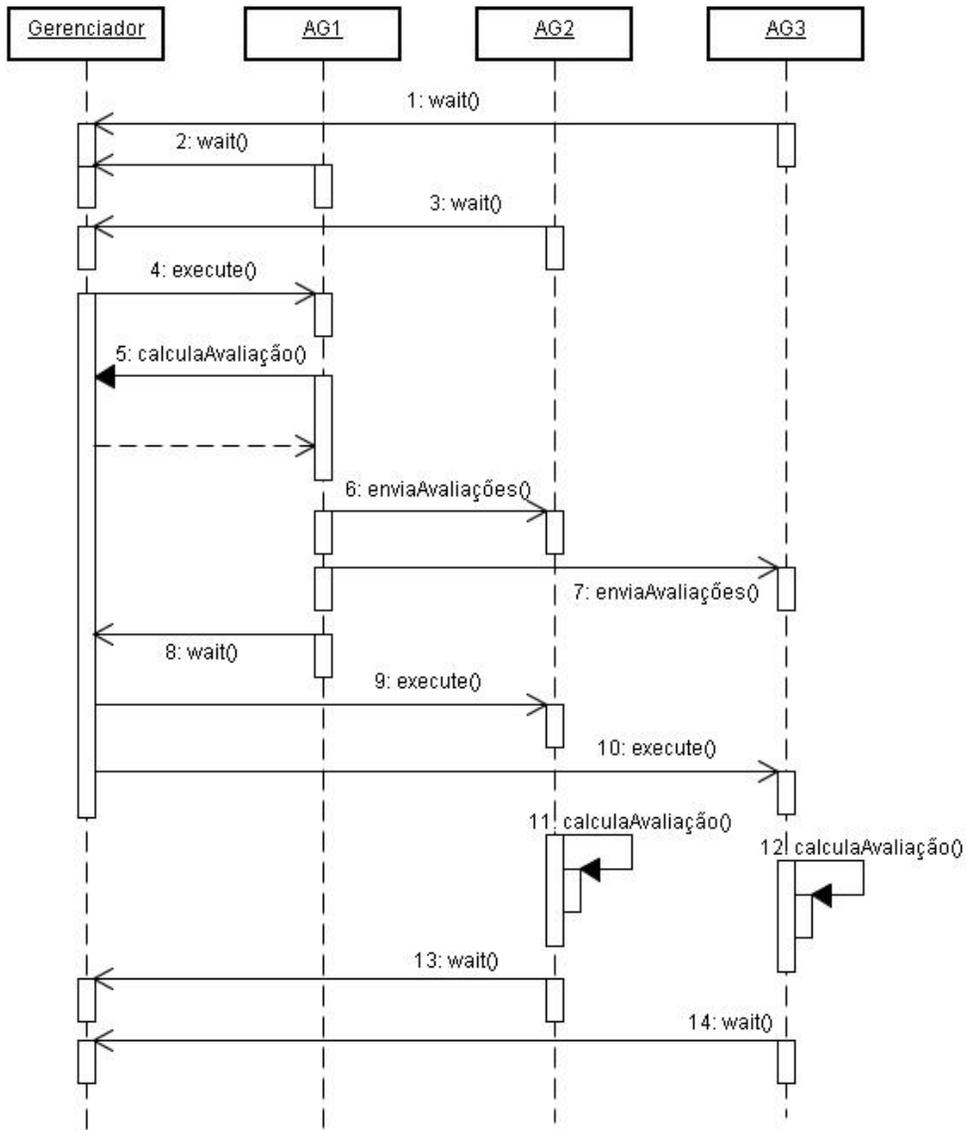


Figura 22: Diagrama de Seqüência de uma Arquitetura Coevolutiva Hierárquica