

PROGRAMAÇÃO GENÉTICA APLICADA A RESOLUÇÃO DO PROBLEMA PRESA-PREDADOR¹

Marcelo Wuttig Friske², Tales Ruan Scheffler Panke³, Mateus Reizes Coimbra Da Silva⁴, Rogério Samuel De Moura Martins⁵, Juliano Gomes Weber⁶, Sandro Sawicki⁷.

¹ Parte integrante do Trabalho Conclusão de Curso

² Aluno do curso de Ciência da Computação da Unijui, marcelo8118@gmail.com

³ Aluno do curso de Ciência da Computação da Unijui, talesruan@gmail.com

⁴ Aluno do curso de Ciência da Computação da Unijui, mateus_reizes@hotmail.com

⁵ Professor Orientador, Mestre em Computação Aplicada, Curso de Ciência da Computação, rogerio.martins@unijui.edu.br

⁶ Professor Mestre do Departamento de Ciências Exatas e Engenharias, jgw@unijui.edu.br

⁷ Professor Doutor do Departamento de Ciências Exatas e Engenharias, sawicki@unijui.edu.br

Introdução

No decorrer da evolução da computação, podemos observar sua requisição para resolução de problemas com complexidades cada vez maiores. Busca-se a partir disso, métodos que vão além de aumento do poder computacional para solucionar tais problemas. Conforme Russel (2004), existem algoritmos de buscas que não consideram o caminho percorrido para chegar ao resultado, pois não fará parte dele (como por exemplo, escalonamento de horários). Algoritmos assim são denominados de busca local. Além deles, encontramos outra área que busca soluções para problemas complexos: A inteligência artificial (AI – Artificial Intelligence), capaz de sistematizar e automatizar tarefas intelectuais através de agentes inteligentes. Ou seja, capazes de perceber e atuar sob o ambiente através de sensores - como os olhos, e de atuadores - como as mãos.

Existe um problema conhecido como presa-predador, proposto por Miroslav Benda (Korf, 1992), onde há um plano de coordenadas x e y finito, um agente chamado presa e quatro agentes denominados predadores. Os agentes podem mover-se vertical e horizontalmente, não podendo ocupar a mesma posição. O plano (ou “mundo”) é limitado ou toroidal, este último define que quando um agente atinge a última célula do plano, ele passa para a primeira célula de posição contrária. O objetivo é que os predadores cerquem a presa, de modo que ela não possa se mover.

Será utilizado esse problema para testar a eficiência de agentes inteligentes que serão criados através da Programação Genética (PG). Ela é baseada na teoria da Evolução das Espécies, de Darwin, a qual explica que as espécies competem entre si por recursos, sendo que a mais adaptada ao habitat é a que terá maior probabilidade de sobreviver - pois conseguirá se reproduzir mais facilmente. A PG foi desenvolvida por John Koza em 1989, e é considerada uma das técnicas de Computação Evolucionária na qual os indivíduos são programas computacionais que utilizam



Modalidade do trabalho: Relatório técnico-científico

Evento: XXI Seminário de Iniciação Científica

árvores sintáticas, variando de forma e tamanho (Souza, 2006). Ainda existem conjuntos de estados denominados populações, que serão submetidos a operadores genéticos tais como seleção (fitness), crossover e mutação. Eles irão simular um processo de evolução natural da população, gerando novos estados (ou indivíduos) que serão uma boa solução para o problema. Além de buscar resolver o problema da presa-predador atuando sobre os predadores com PG e apresentar os resultados através de um conjunto de regras, esse trabalho propõe a comparação entre um estudo semelhante, que busca resolver o mesmo problema de forma semelhante.

Metodologia

A proposta do presente trabalho foi realizar testes em uma implementação própria, onde é simulado o ambiente presa-predador e evoluída a inteligência dos predadores através de PG. Para isso, foi utilizada a linguagem de programação Java.

Foram criadas classes para a implementação da árvore sintática. Elas representam os operadores de controle de fluxo, lógicos, de relação e aritméticos. Também foram criadas classes de operandos, que são representados por variáveis denotadas pelas coordenadas de cada predador ($x_1, y_1 \dots x_4, y_4$) e da presa (x_p e y_p). Houve a criação de classes que implementam os personagens presa e predadores, a classe Mapa que define as características do ambiente e a classe "Simulacao", a qual é executada em virtude da classe "Evolucao".

A classe "Evolucao" é a responsável pela Programação Genética, e recebe os parâmetros para execução da simulação, tais como: máximo de gerações, tamanho da população, taxa de mutação, tamanho do mapa e se o mesmo é toroidal. No decorrer da simulação, a evolução avalia o desempenho de cada indivíduo na simulação e lhe atribui um fitness. Logo após realiza a mutação dos indivíduos conforme a probabilidade desta acontecer. Ao final da simulação, é retornada à tela de interface gráfica a expressão definida pelo indivíduo de melhor avaliação.

Resultados e discussão

Côrtes (Deise da Silva, 2005) expõe em sua dissertação de mestrado a utilização de redes neurais que sejam evoluídas através de algoritmos genéticos para resolver o problema presa-predador. Em sua implementação, os predadores passam primeiramente por um treino, onde o grau de dificuldade do problema aumenta conforme os predadores executam a tarefa de captura com sucesso. Para a função fitness foi utilizado um cálculo onde a qualidade do indivíduo é inversamente proporcional à distância final dos predadores em relação à presa.

Contudo, os resultados obtidos pela autora não foram de todo satisfatórios. Quando o mundo era definido como limitado, os predadores não conseguiam capturar a presa, embora fosse tecnicamente mais fácil. No ambiente de evolução aqui descrito essa tarefa foi muito fácil quando a presa era estacionária - tanto quando o objetivo era tocar quanto cercar a presa.





Modalidade do trabalho: Relatório técnico-científico

Evento: XXI Seminário de Iniciação Científica

Na implementação foram criados dois níveis iniciais para a evolução dos personagens, semelhante ao trabalho de Côrtes, os indivíduos são submetidos a níveis de avaliação, onde o primeiro objetivo é tocar a presa estacionária, posteriormente cerca-la. Já as informações que o predador recebe para evoluir são as próprias coordenadas, dos demais predadores e da presa. O fitness é calculado diferentemente de acordo com o nível, sendo que para o primeiro nível é definido o número de passos necessários para tocar-se a presa, e o segundo utiliza uma média de distância entre os predadores e a presa. A simulação inicia com o primeiro nível, ao chegar à sexta geração de indivíduos, foi definido que os indivíduos teriam de enfrentar o segundo nível. Em ambos os casos os predadores conseguiram atingir o seu objetivo

Os parâmetros utilizados na classe "Evolucao" para testes com bons resultados foram: máximo de gerações: 250; tamanho da população: 50; taxa de mutação: 5%; largura e largura do mapa: 10 e mapa limitado. Todos esses parâmetros não podem ser definidos como ótimos, pois a cada execução novas expressões são geradas, podendo ser de melhor ou menor qualidade que a anterior. Contudo, esses parâmetros foram suficientes para que os indivíduos conseguissem tocar e capturar a presa. Realizaram-se testes de 10 simulações, onde a aptidão da população se estabilizou a partir da 163ª geração.

Como não foi implementada a função de crossover, somente ocorreu a seleção e mutação de indivíduos, esta última com uma probabilidade de 5% em cada nó da árvore sintática, a fim de haver diversidade de indivíduos e maior probabilidade de alcance do objetivo. O mapa foi definido como limitado, pois é parte do primeiro e segundo nível da evolução não podendo defini-lo como toroidal.

Conclusão

Nota-se que a proposta de Côrtes não foi tão satisfatória quanto o desejado. Seus agentes possuíam o mesmo conhecimento se comparado ao da implementação deste trabalho (coordenadas dos personagens), contudo para casos em que o mundo era limitado, seus predadores não conseguiram capturar a presa, enquanto isso foi uma tarefa muito simples nesse trabalho, quando a presa era estacionária.

O princípio da navalha de Ockham, o qual define que se existem diversas soluções para um problema, a mais simples é a melhor (Patriquin, 2007), nota-se que dentro da PG, dependendo do problema a ser resolvido, nem todos os operadores genéticos precisam ser implementados. Não foi utilizado o crossover neste trabalho, pois ao cruzar duas soluções consideradas boas na função fitness, acabara-se por criar uma expressão mais complexa e não necessariamente melhor.

Como sugestão a trabalhos futuros, será buscado implementar mais níveis de complexidade para o problema presa-predador. O objetivo final do trabalho é adquirir conhecimento e experiência em





Modalidade do trabalho: Relatório técnico-científico

Evento: XXI Seminário de Iniciação Científica

programação genética, onde se possam implementar soluções para outros tipos de problemas utilizando a PG.

Palavras-Chave: Programação genética, inteligência artificial, agentes inteligentes.

Referências Bibliográficas

CORTÊS, Deise da Silva. Modelo Neuro-Evolutivo de Coordenação Adaptativa em Ambientes Dinâmicos. Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, 2005.

KORF, Richard E. A simple solution to pursuit games. International Workshop on Distributed Artificial Intelligence, 1992.

PATRIQUIN, Travis. O Princípio da Navalha de Occam para Vincular Fatos: O Partido Ba'ath e a Insurgência em Tal Afar. Military review, 2007.

RUSSEL, Stuart J. Inteligência Artificial: tradução da segunda edição. Elsevier Editora Ltda, 2004.

SOUZA, Luiza Vidal de. Programação Genética e Combinação de Preditores para Previsão de Séries Temporais. Universidade Federal do Paraná, 2004. Disponível em <<http://www.ppgmne.ufpr.br/arquivos/teses/4.pdf>> Acessado em 14/03/2013.

