Explorando Paralelismo de Dados e Memória Compartilhada para Aumentar o Desempenho da Aplicação N-Rainhas

Vinícius Meyer[1], Clébio Dossa[1], Rodrigo da Rosa Righi[1]

[1]Prog. Interdisciplinar de Pós-Graduação em Computação Aplicada, Unisinos - Brasil

vinimeyer@hotmail.com, clebiodossa@gmail.com, rrrighi@unisinos.br

Resumo. *Este trabalho apresenta a resolução do problema das N-Rainhas, que consiste em encontrar todas as combinações possíveis em um tabuleiro de xadrez de dimensão N por N tal que nenhuma das rainhas ataque as demais. Para isso foi utilizado o algoritmo genético de recombinação e mutação com a biblioteca Pthreads. Os resultados apresentam que é possível obter 58,3% de redução no tempo de execução do algoritmo com a utilização de Pthreads.*

1. Introdução

Na área da computação existe uma classe de problemas no qual a tempo de busca é muito grande. Estes problemas não podem ser resolvidos por um algoritmo de busca clássico ou com programação linear (algortimo que garante a solução) porque o espaço de busca não é computável em tempo polinomial. Para resolver estes problemas é possível explorar outros tipos de algoritmos de busca que podem encontrar a solução. Estes algortimos procuram a melhor solução para o problema trilhando um caminho que melhora a função objetiva. Um exemplo desses algoritmos inclui aquele que imita o processo evolutivo em sua busca para a solução, o Algoritmo Genético [Man, K. F., Tang, K. S. and Kwong, S., 1996]. O problema das N-Rainhas é um problema NP-Difícil. O objetivo deste problema é colocar N rainhas em um tabuleiro de xadrez N por N, de tal maneira que elas não podem atacar umas as outras, ou seja, em cada linha, coluna ou diagonal, não pode existir mais de uma rainha. As rainhas podem se mover horizontalmente, verticalmente e diagonalmente, o que significa que só pode haver uma rainha por linha, uma por coluna, e não pode haver mais de uma rainha na mesma diagonal [Sosic, R. Gu, J., 1991]. Neste cenário, o presente trabalho propõe resolver o problema das N-Rainhas com o algoritmo genético de recombinação e mutação e após isso adicionar a biblioteca Pthreads para conseguir melhor desempenho na execução do algoritmo.

2. Implementação do Algoritmo Genético com Pthreads

Algoritmos Genéticos são algoritmos de otimização global, baseados nos mecanismos de seleção natural e da genética. Os mesmos utilizam a estratégia de busca paralela e estruturada, mas aleatória, que é voltada em direção ao reforço da busca de pontos de *fitness* elevado, ou seja, busca dos melhores resultados. Para isso o algoritmo segue as etapas a seguir: (1) criar a população, onde cada indivíduo representa uma distribuição aleatória de N posições no tabuleiro, (2) avaliação do fitness (avaliação da aptidão de cada indivíduo), (3) seleção dos indivíduos pais, (4) recombinação dos pais (gerando

novos filhos), (5) mutação (evolução dos indivíduos). Após isso volta até a etapa 2 e repete a quantidade de vezes que for desejada. Cada vez que é repetido esse processo é passada uma geração, ou seja, se for definido que o algoritmo deve evoluir x gerações, esse passo-a-passo deve ser executado x vezes. Esse código feito em linguagem C, foi executado em um computador com processador Intel Core I5-4210U 1,7GHz (2 núcleos físicos e 4 lógicos), com 4 GB de memória RAM. Nos testes realizados definiu-se a utilização de 256 rainhas, o que quer dizer que o tabuleiro é de 256 por 256 posições. Para fim de testes de desempenho foi utilizado o número de 10 gerações com uma população de 100 indivíduos. Após rodar o código de maneira sequencial percebeu-se que o tempo de execução foi de 34,67 segundos. Para otimizar este tempo foi adicionado o artifício do uso da biblioteca de Pthreads no trecho de código que diz respeito aos cálculo do fitness. Este pedaço do código é possível de ser paralelizado porque ele executa nos indivíduos já criados e o cálculo de aptidão de cada indivíduo não tem nenhuma dependência com o resto do código. Na tabela 1 é possível observar os tempos de execução do código com o uso de uma até dez Threads. Os tempos resultam de uma média de dez execuções a cada número de Threads.

Threads	Tempo(s)
1	34,67
2	19,43
3	16,25
4	14,46
5	14,96
6	14,92
7	14,94
8	15,00
9	15,06
10	15,15

Tabela 1 - Tempos de execução *versus* Nº de Threads

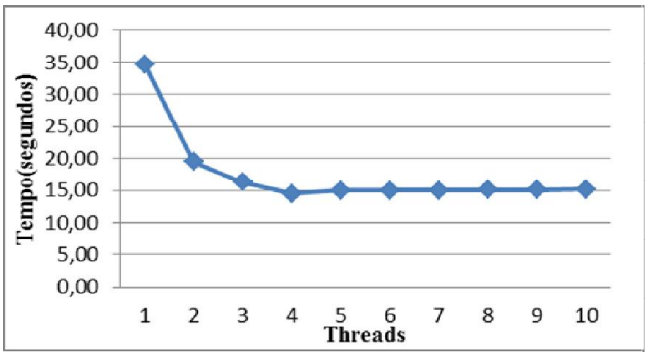

Figura 1 - Tempos de execução *versus* Nº de Threads

3. Conclusão

Este artigo investiga a eficácia do algoritmo genético na busca da melhor solução para o problema das N-Rainhas com a utilização da biblioteca Pthreads. Analizando os dados obtidos nas execuções do código, o melhor tempo foi com a utilização de quatro Threads (14,46 segundos). Com o uso de cinco Threads ou mais, o tempo de execução do código começa a aumentar, conforme mostra a figura 1. A partir dos resultados pode-se concluir que é possível obter 58,3% de economia de tempo na execução do algortitmo genético com o uso de Pthreads para resolver o problema das N-Rainhas.

Referências

Man, K. F., Tang, K. S. and Kwong, S. (1996) "Genetic algorithms: concepts and applications", in: Industrial Electronics, IEEE Transactions on (Volume: 43, Issue: 5), pages 519 – 534, IEEE.

Sosic, R. and Gu, J. (1991) "Fast search algorithms for the n-queens problem" in: Systems, Man and Cybernetics, IEEE Transactions on (Volume: 21, Issue: 6), pages 1572 – 1576, IEEE.