Análise Empírica da Utilização de Algoritmos Genéticos ao Problema da Mochila

Sabrina Eloise Nawcki

[1]Bacharelado de Ciência da Computação - Universidade Tuiuti do Paraná (UTP)
Curitiba - PR - Brasil

snawcki@gmail.com

Abstract. *This article aims to conduct an empirical study of the application of genetic algorithms in the Knapsack Problem, which is a NP-Hard problem. The parameters studied population size, population initialization strategy, parent selection strategy, crossover cutoff points, mutation rate, survivor selection strategy. To validate this proposal, experiments were carried out in different instances of the Knapsack Problem, given by the professor of Computational Intelligence. The results obtained were compared in relation to the quality of the final solution and at the time of execution.*

Resumo. *Este artigo tem por objetivo realizar um estudo empírico da aplicação de algoritmos genéticos no Problema da Mochila, que faz parte da classe NP-completo. Os parâmetros estudados tamanho da população, estratégia de inicialização da população, estratégia de seleção dos pais, pontos de corte no crossover, taxa de mutação, estratégia de seleção dos sobreviventes. Para validar esta proposta, foram realizados experimentos em diferentes instancias do Problema da Mochila, dados pelo docente da disciplina de Inteligência Computacional. Os resultados obtidos foram comparados com relação à qualidade da solução final e no tempo de execução.*

1. Introdução

O Problema da Mochila (Knapsack Problem) é um problema de otimização combinatória e faz parte da classe de problemas NP-completo [Richard Karp 1972]. O problema é modelado em uma situação em que é necessário preencher uma mochila com objetos de diferentes pesos e valores obtendo o maior valor possível sem ultrapassar o peso máximo [Yu and Ge 2010].

Neste trabalho foi estudada a metaheurística de Algoritmos Genéticos (GA), para ser aplicada ao problema da mochila. Os GAs são um ramo dos algoritmos evolucionários e podem ser definidos como uma técnica de otimização global [Linden 2012].

A abordagem a ser estudada é as diferenças entre as seguintes técnicas de seleção de pais: Roleta Viciada, Torneio e Aleatória. Para validar esta proposta foram realizadas comparações de resultados de experimentos utilizando diferentes instâncias do problema da mochila.

Na seção 2 é apresentado o modelo de representação do problema. Na seção 3 é apresentada a fundamentação teória dos GAs. A metodologia, os experimentos realizados e os resultados obtidos estão na seção 4, seguido de considerações finais na seção 5.

2. Representação do Problema

A modelagem do problema é uma etapa importante para a sua resolução. Neste trabalho, a solução é representada por uma classe Mochila que contém 3 vetores de n posições, onde n é a quantidade de itens: o primeiro vetor armazena os

números dos itens, o segundo vetor armazena os pesos dos itens e o terceiro vetor armazena os valores dos itens e uma variável decimal que armazena o peso máximo que pode ser carregado na mochila.

3. Algoritmos Genéticos

Algoritmos Genéticos (GA) são um ramo dos algoritmos evolucionários e podem ser definidos como uma técnica heurística de otimização global baseada numa metáfora o processo biológico de evolução natural. Os GAs criam populações de indivíduos que são avaliados e submetidos aos operadores genéticos: seleção, recombinação e mutação, ao realizar essas operações a população passa por um processo de evolução natual, que eventualmente deverá gerar um indivíduo que caracterizará uma boa solução [Linden 2012].

3.1. Descrição do Algoritmo

A Figura 1 ilustra o bloco princial do GA [Linden 2012, adaptado].

1. Inicializar a população de cromossomos
2. Avaliar cada cromossomo na população
3. Selecionar os pais para gerar novos cromossomos
4. Aplicar operador de recombinação nos pais para gerar descendentes
5. Aplicar operador de mutação nos descendentes
6. Selecionar os sobreviventes
7. Se o tempo acabor, ou o melhor cromossomo satisfazer os requerimentos e desempenho, retorná-lo, caso contrário, voltar para o passo 3

Figura 1. Pseudo-código para GAs

3.2. Representação Cromossomial

A representação cromossomial consiste em traduzir a informação de um problema de forma que o computador consiga tratá-la. Essa representação é completamente arbitrária e é definida pelo desenvolvedor do GA junto a adequação ao problema [Linden 2012]. Apesar disso, Linden (2012) indica que algumas regras sejam seguidas para a obtenção da representação, apresentadas na Figura 2.

1. A representação deve ser o mais simples possível;
2. Se houver soluções proibidas ao problema, é preferível que elas não sejam representadas;
3. Se o problema impuser condições de algum tipo, estas devem estar implícitas dentro da representação

Figura 2. Regras para obtenção da representação cromossomial

3.3. Escolha da População Inicial

A inicialização da população de acordo com Linden (2012) normalmente é feita da forma mais simples possível através de uma escolha aleatória independente para cada indivíduo da população aleatória. A inicialização aleatória da população gera uma boa distribuição das soluções no espaço de busca e o uso do operador de mutação garante uma boa exploração de todo o espaço de busca [Linden 2012].

3.4. Função de Aptidão

A função de aptidão é utilizada pelos GAs para determinar a qualidade de um indivíduo como solução do problema através de um cálculo de valor numérico [Linden 2012].

3.5. Seleção dos Pais

O método de seleção de pais faz a simulação do mecanismo de seleção natural que atua na natureza, dando preferência à seleção dos pais com alta performace [Linden 2012].

3.5.1. Roleta Viciada

No método de seleção por Roleta Viciada, cada indivíduo da população é representado na roleta proporcionalmente ao seu índice de aptidão. Assim, para indivíduos com alta aptidão é dada uma porção maior da roleta, enquanto aos indivíduos de aptidão mais baixa, é dada uma porção relativamente menor [Pozo *et al.*].

3.5.2. Torneio

Para a implementação do método de torneio, é preciso selecionar *x* indivíduos randomicamente e comparar os valores de suas aptidões, o elemento com a melhor aptidão vence o torneio e é selecionado, o processo é repetido até que a quantidade de pais determinada seja alcançado [Yu and Ge 2010].

3.6. Recombinação

A recombinação é o processo de criação de novos indivíduos atrvés da mistura das características de dois indivíduos pais, esta mistura é feita tentando imitar a reprodução de genes em células onde trechos das características de um indivíduo são trocados pelo trecho equivalente do outro. O resultado desta operação é um indivíduo que potencialmente combine as melhores características dos indivíduos usados como base [Pozo *et al.*].

3.6.1. Um Ponto de Corte

O operador de recombinação de um ponto de corte é mais simples que outros operadores, funcionando da seguinte forma: após a seleção de dois pais, um ponto de corte é selecionado pelo operador, esse ponto de corte nada mais é que a separação entre dois genes que compoem o material genético do pai, contudo, cada indivíduo possui *n* genes e *n-1* pontos de corte. Depois de sortear o ponto de corte pelo operador, o material genético é separado em duas partes, à esquerda do ponto de corte e à direita do ponto de corte, o primeiro filho será composto pela concatenação da parte do primeiro pai à esquerda do ponto de corte com a parte à direita do ponto de corte do segundo pai, e o segundo filho com as partes que sobraram [Linden 2012].

3.7. Mutação

Após a combinação dos filhos é aplicado o operador de mutação. Normalmente o operador de mutação contém uma probabilidade extremamente baixa de 0.5%, o valor da probabilidade deve ser baixo pois, caso o valor fosse muito alto o algoritmo genético funcionaria de forma aleatória, nao sendo esse seu proposito.Se o valor sorteado for menor que a probabilidade predeterminada, então, o operador atua sobre o gene em questão, alterndo-lhe o valor aleatoriamente. O processo se repete para todos os genes dos dois filhos [Linden 2012].

3.8. Seleção dos Sobreviventes

A selecão dos sobreviventes consiste em controlar a população assumindo que a mesma não pode crescer. Para isso, é preciso substituir os elementos conforme são criados novos filhos nas gerações [Linden 2012].

4. Metodologia e Resultados Experimentais

O problema da mochila foi escolhido como base para os experimentos. Foram utilizadas como validação do algoritmo oito instâncias dadas pelo docente da disciplina de Inteligência Computacional, cada uma tendo um peso limite e quantidade de itens diferentes. Ao todo foram utilizados 24 conjuntos de testes: três testes, mudando o metódo de seleção de pais do algo-

ritmo genético entre roleta viciada, torneio e aleatória, para cada instância.

Para cada um destes problemas os algoritmos estudados foram executados 3 vezes. Destas execuções 1 foi realizada em um ambiente controlado para a análise do tempo computacional. Para todas as execuções os parâmetros do Algoritmo Genético foram: 20 elementos por população, a condição de parada é quando o algoritmo repetiu o mesmo resultado das aptidões nas últimas 4 gerações ou quando o algoritmo já realizou n gerações, sendo n o valor proporcional ao dobro da quantidade de itens disponíveis, quantidade de pais e descendentes proporcional à metade da quantidade de elementos, recombinação com um ponto de corte, 0.4% de taxa de mutação, e seleção de sobreviventes pelo método de elitismo.

Na Tabela 1 são apresentados os resultados obtidos. A primeira coluna é o nome do arquivo utilizado, seguido pela quantidade de itens e peso limite contidas nesta instância. A quarta coluna apresenta o valor de referência da melhor solução encontrada para a instância. A quinta coluna apresenta a melhor solução encontrada com a seleção de pais utilizando a Roleta Viciada e na sexta coluna está o desvio padrão, $\sigma 1$ das soluções encontradas, considerando as 3 execuções. A sétima coluna apresenta a melhor solução encontrada com a seleção de pais utilizando Torneio e na oitava coluna está o desvio padrão, $\sigma 2$ das soluções encontradas, considerando as 3 execuções. A nona coluna apresenta a melhor solução encontrada com a seleção de pais Aleatória e na décima coluna está o desvio padrão, $\sigma 3$ das soluções encontradas, considerando as 3 execuções.

Analisando os resultados obtidos com a seleção de pais utilizando o método de Roleta Viciada é possível observar que o algoritmo apresenta um bom comportamento na resolução do problema. Observando o desvio padrão, $\sigma 1$, em 7 instâncias de teste de 8 este método obteve a melhor solução encontrada.

Arquivo	Qtde. Itens	Peso Limite	Ref.	Roleta Viciada	$\sigma 1$	Torneio	$\sigma 2$	Aleatória	$\sigma 3$
ksi0012	5	12	35.0	35.0	0	35.0	0	35.0	0
ksi0032	12	32	259.0	259.0	0	235.0	24.0	259.0	0
ksi0100	20	100	986.0	986.0	0	817.0	169.0	986.0	0
ksi0322	80	322	5660.0	5660.0	0	3558.0	2102.0	5128.5	532.0
ksi0450	120	450	5453.8	4733.4	720.4	2819.7	2634.1	5453.8	0
ksi0890	220	890	12270.6	12270.6	0	8689.4	3581.2	12226.5	44.1
ksi1200	500	1200	22676.4	22676.4	0	11947.9	10728.5	20880.3	1796.1
ksi8000	1200	8000	188525.5	188525.5	0	46325.3	142200.2	N/A	N/A

Tabela 1. Resultados obtidos nos experimentos, com as abordagens estudadas: seleção de pais por roleta viciada, torneio e aleatória

Os resultados obtidos com a seleção de pais utilizando o método de Torneio, não apresentou resultados melhores que os métodos Roleta Viciada e Aleatória, exceto na instância de teste com 8000 itens, onde o método Aleatório não encontrou uma solução. Porém, esse método foi o que apresentou melhor tempo computacional (ver Tablea 2).

Os resultados obtidos com o método de seleção de pais Aleatória apresentou bons resultados em relação, porém na instância de teste com 8000 itens não encontrou uma solução para o problema. Observando o desvio padrão, $\sigma 3$, em 4 instâncias de teste de 8 este método obteve a melhor solução encontrada.

A Tabela 2 apresenta o tempo de execução em milisegundos da execução em ambiente controlado, do algoritmo genético para cada tipo de seleção de pais. É possível observar que o tempo aumenta consideravelmente a medida que a complexidade dos problemas é aumentada.

Analisando os resultados da Tabela 2, podemos observar que o metódo de seleção de pais Torneio tem um desempenho maior que os métodos Roleta Viciada e Aleatória em relação ao tempo de execução. Observando o método Aleatório é possível analisar que na maioria das vezes o seu tempo de execução é maior, pois o método

Arquivo	Qtde. Itens	Peso Limite	Roleta Viciada	Torneio	Aleatória
ksi0012	5	12	32.2	48.3	35.5
ksi0032	12	32	62.1	34.5	60.8
ksi0100	20	100	71.3	64.2	72.7
ksi0322	80	322	322.0	204.0	305.0
ksi0450	120	450	464.0	405.0	623.0
ksi0890	220	890	1140.0	863.0	1330.0
ksi1200	500	1200	4670.0	3030.0	5520.0
ksi8000	1200	8000	25100	20000	25700

Tabela 2. Comparação com relação ao custo computacional dos experimentos, com as abordagens estudadas: seleção de pais por roleta viciada, torneio e aleatória

aleatório nem sempre seleciona os melhores pais e por este motivo precisa de mais gerações até obter uma solução. Para todos os métodos quanto maior a quantidade de itens maior o tempo de processamento, isso ocorre pela condição de parada do algoritmo que está diretamente relacionada à quantidade de itens.

5. Considerações Finais

Neste trabalho foi apresentado o Algoritmo Genético com os métodos de seleção de pais Roleta Viciada, Torneio e Aleatória, para a resolução do problema da mochila. O algoritmo apresentou um bom comportamento nos experimentos realizados utilizando as instâncias de testes dada pelo docente da disciplina de Inteligência Computacional. Além disso, dada a complexidade do problema, foi possível comparar os resultados e tempo de execução das soluções para cada método de seleção de pais. Uma melhoria a ser adicionadas às instâncias de teste é a adição da referência global da melhor solução para cada problema, visto que nos testes realizados a melhor solução utilizada como referência é local, obtida através da execução dos testes.

Referências

Linden, Ricardo (2012). Algoritmos Genéticos.

Yu, Xinjie and Ge, Mitsuo (2010). Introduction to Evolutionary Algorithms, p. 270-271.

Pozo *et. al*. Computação Evolutica.