KFPS e Heurísticas Ingênuas

Rodrigo Kenji Asato Kobayashi[1]

[1]Faculdade de Computação (FACOM) - Universidade Federal de Mato Grosso do Sul (UFMS) - Campo Grande - MS - Brasil

rodrigo.kobayashi@ufms.br

Abstract. *This article presents an implementation of the knapsack problem with forfeit sets (KFPS) and 2 naive heuristics. The main objective is to evaluate the quality of the solutions generated by such heuristics.*

Resumo. *Este artigo apresenta uma implementação do problema da mochila com conjuntos de penalidades e 2 heurística ingênuas. O principal objetivo é avaliar a qualidade das soluções geradas por tais heurísticas.*

1. Introdução

O trabalho consiste na implementação de um algoritmo exato *branch-and-bound* utilizando o SCIP do problema da mochila com conjuntos de penalidades. O modelo PLI é composto pela função objetivo trata-se da maximização do somatório dos valores dos itens escolhidos descontando-se o somatório das penalidades, garantia que a soma dos pesos não ultrapasse a capacidade máxima da mochila, o número de violações não exceda k, a contabilização correta das violações e as restrições de integralidade.

Além disso há a implementação de 2 heurísticas ingênuas. Uma delas é uma heurística aleatória, onde é sorteado o item a ser inserido a partir da lista de candidatos. A segunda é uma heurística gulosa onde sua decisão de inclusão é escolher o item que possui maior razão entre valor e peso, no entanto não toma em conta possível penalidade.

2. Metodologia de Desenvolvimento

A partir do código parcial foi necessário implementar a garantia que o total de violações não exceda um k fornecido. Como o y_j de cada conjunto j de penalidades já havia sido criado, foi necessário apenas fazer o somatório e obrigá-lo a ser menor que I->k.

Na heurística aleatória, a partir de seu código parcial, também foi necessário apenas contabilizar o número de violações. Na heurística gulosa, além das condições implementadas como na aleatória, para a decisão de inclusão na solução criou-se uma estrutura que guardou o índice do item e sua "densidade monetária", seu valor monetário dividido pelo seu peso. A seguir apenas verifica-se se este está na lista de candidatos para prosseguir, caso contrário pulamos neste caso.

A seguir foram realizados 3 tipos de testes:

1. Execução sem nenhuma heurística, limitado a 2 minutos;
2. Execução com heurísticas aleatória e gulosa, aplicada apenas no nó raiz;
3. Execução da melhor heurística de 2, sem limitação do nó e limitado a 2 minutos.

3. Resultados

No primeiro teste, utilizando apenas a implementação básica sem nenhuma heurística habilitada, obtemos 36 soluções ótimas e 84 onde atingiram o limite de tempo, de um total de 120 instâncias diferentes.

Destas 36 soluções com gap zero, o tempo médio de resolução foi de 30,76 segundos, enquanto das 84 que atingiram o limite de tempo, foi obtido um gap de dualidade médio de 19,96%.

No segundo teste, com heurísticas habilitadas e limitando apenas ao nó raiz, a heurística aleatória ganhou da heurística gulosa 90 vezes, onde ganhar significa um gap de dualidade menor.

Das 120 instâncias, a heurística aleatória obteve uma média de 57,48% de gap de dualidade e tempo de execução médio de 0,345669 segundos, enquanto a heurística gulosa obteve uma média de 67,43% de gap de dualidade e tempo de execução médio de 0,346726 segundos. Na figura 1, após ordenar em relação ao gap dual da Heurística Aleatória e plotá-lo juntamente com a Heurística Gulosa, podemos observar que em média seu nível está abaixo em relação ao segundo, com qualidade levemente melhor. Nota-se também que cada instância corresponde a outra na figura 1.

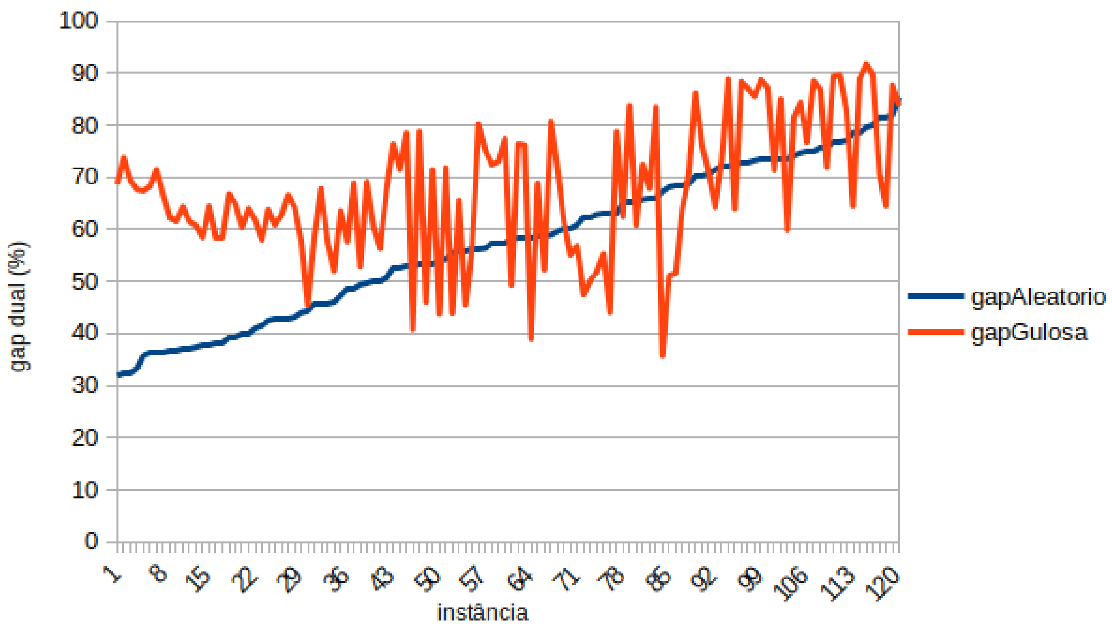

Figura 1. Gap Dual do teste 2

Após observarmos uma qualidade melhor da heurística aleatória no segundo teste, realizou-se o mesmo procedimento do teste 1 com sua heurística habilitada no teste 3. Foram então obtidas 34 soluções ótimas e 86 atingiram o limite de tempo, obtendo um tempo médio de solução com gap zero de 26,71 segundos e um gap de dualidade médio de 24,85% nas instâncias não resolvidas na otimalidade.

Nota-se que utilizando a heurística aleatória foram obtidas 2 soluções a menos que a execução sem heurísticas, ou seja, uma piora de qualidade. No mesmo sentido o gap dual também foi maior em 4,89 pontos percentuais, reforçando seu pior desempenho. E apesar do tempo médio de execução das soluções ótimas ser 4,05 segundos menor que do teste 1, tal ocorrência muito provavelmente não necessariamente sugere um melhor desempenho, é muito mais provável que as 2 soluções que não foram encontradas na otimalidade influenciaram tal estatística de forma ambígua. Na figura 2 podemos observar a diferença de gap dual de forma média após ordená-los ambos de forma crescente, neste caso cada instância não corresponde a outra como feito anteriormente.

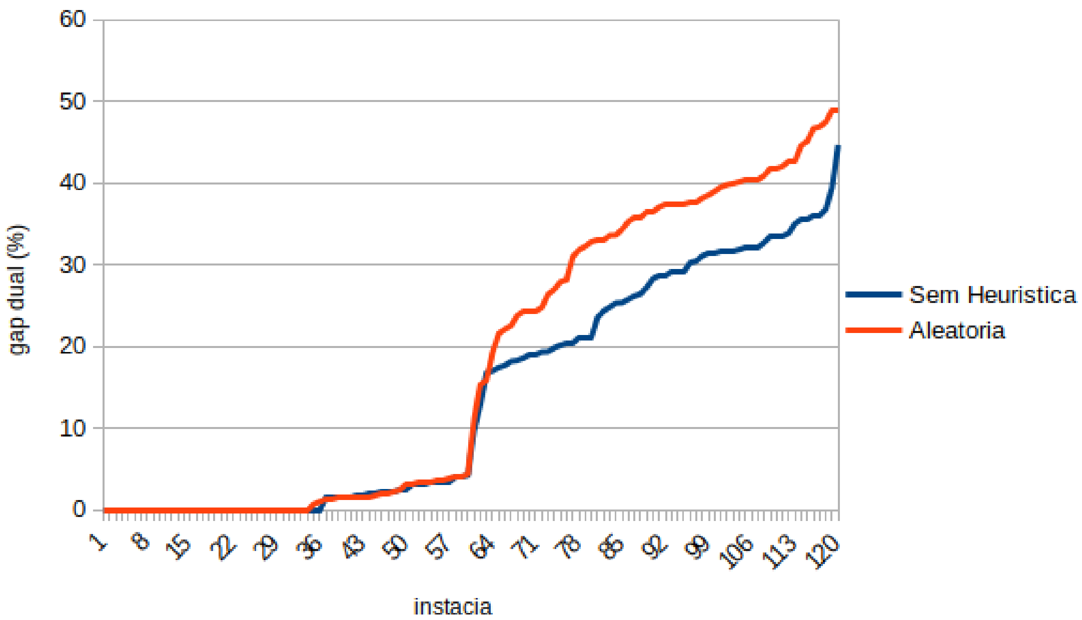

Figura 2. Gap Dual em % ordenando de forma crescente

Nas figuras 3 e 4, tentamos observar a diferença de tempo gasto de execução, na figura 3 como a maioria não foi resolvida na otimalidade não é possível grandes comentários. E na figura 4, observando apenas aquelas com gap dual zero, não há grandes diferenças, apenas uma leve maior demora da Heurística Aleatória quando observada graficamente, que pode reforçar o ponto levantado anteriormente da significância da estatística observada com quantidades diferentes. Nota-se que não foi feita a correspondência por instância entre execuções, apenas ordenados de forma crescente.

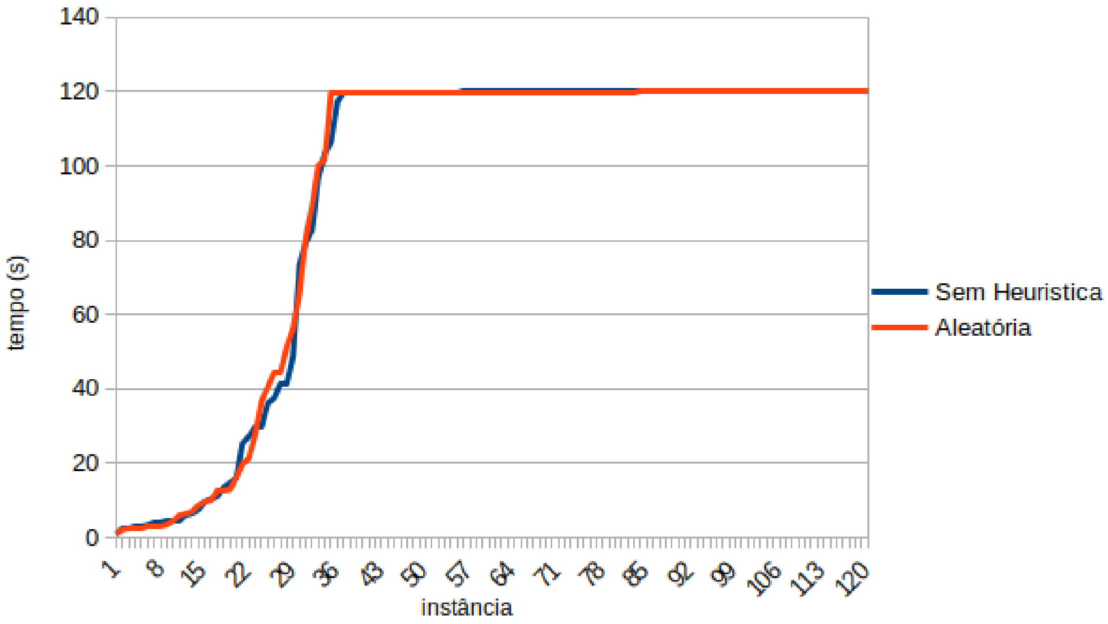

Figura 3. Tempo gasto de todas as instâncias

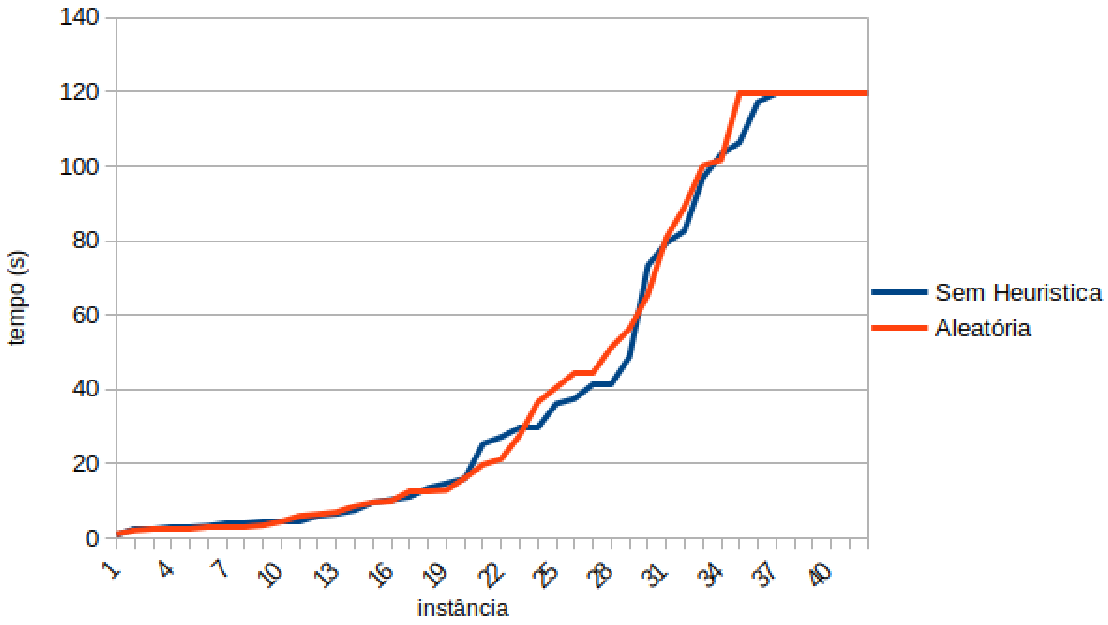

Figura 4. Tempo gasto nos resultados ótimos

4. Conclusões

Com base nessas observações e comparações com e sem utilização de Heurísticas Ingênuas não foi observado melhoria na qualidade dos resultados obtidos nestas condições e tempos de execução determinados.

5. Referências

Hoshino, Edna Ayako. 1º Trabalho Prático de Tópicos em Computação I - Heurísticas baseadas em modelos matemáticos. Disponível em: https://ava.ufms.br/pluginfile.php/1395186/mod_resource/content/4/trabalho-1-corrigido.pdf Acesso em: 22/05/2024.