

Relatório - Segundo Trabalho Prático

Lucas Néia Torres
GRR20210570

Leonardo Becker de Oliveira
GRR20211779

I. INTRODUÇÃO

Este relatório apresenta a modelagem e implementação do problema de Separação de grupo minimizando conflitos, utilizando a abordagem do algoritmo Branch and Bound. O problema consiste em dividir um grupo de super-heróis em dois grupos, levando em consideração suas afinidades e conflitos, de forma a minimizar o número de pares de super-heróis com conflito dentro de um mesmo grupo.

O algoritmo Branch and Bound é uma técnica utilizada para resolver problemas de otimização combinatória. Ele utiliza uma abordagem de busca em árvore, ramificando em subproblemas, e faz uso de limites superiores e inferiores para descartar subárvores que não levarão à solução ótima.

O objetivo do algoritmo é encontrar a distribuição ótima de elementos em dois grupos, levando em consideração tanto as afinidades entre os elementos quanto os conflitos existentes.

II. MODELAGEM

O problema de Branch and Bound foi modelado utilizando duas principais estruturas, árvores e triângulos, onde a árvore é utilizada para armazenar todas as recursões percorridas no Branch and Bound e a estrutura triângulos para armazenar os triângulos fornecidos pelos heróis com conflitos ainda não escolhidos pela recursão, onde é usado um vetor dessa estrutura. Dentro de cada nó da árvore são armazenados, em um vetor, os super-heróis contidos no grupo A; um vetor para os super-heróis do grupo B; número de conflitos do nó; número de afinidades do nó; além dos filhos desse nó. Na estrutura de triângulos são armazenados, em diferentes atributos, os super-heróis que pertencem ao triângulo.

III. ANÁLISE DAS FUNÇÕES LIMITANTES

A análise das funções limitantes é um aspecto fundamental na resolução de problemas de otimização. Uma função limitante fornece uma estimativa superior do valor ótimo da solução, permitindo uma busca mais eficiente. A função limitante elaborada foi baseada em mE, que representa o número de conflitos entre elementos já atribuídos ao mesmo grupo, adicionado ao número de triângulos formados por conflitos não resolvidos.

$$Bdada(E) = (x, y) \quad CE - g(x) = g(y) - + tE + mE$$

A mE é calculada a partir do conjunto de conflitos envolvendo apenas elementos com grupos atribuídos, sendo uma medida significativa para avaliar a qualidade de uma solução parcial. Ao calcular a mE em diferentes estágios da busca, é possível realizar podas e direcionar o algoritmo para caminhos mais promissores, aumentando a eficiência e acelerando a

convergência para a solução ótima. A utilização da mE como função limitante oferece uma estratégia inteligente para guiar a busca em direção às melhores soluções, tornando-se uma ferramenta valiosa no contexto da otimização combinatorial.

IV. IMPLEMENTAÇÃO

O algoritmo recebe como entrada um conjunto de afinidades e conflitos representados por matrizes, onde cada linha representa um par de elementos. Inicialmente, todos os super-heróis são atribuídos a um vetor de atribuições e é criado um nó raiz para a árvore de soluções. É chamada a função de Branch and Bound com o nó raiz criado e com tal vetor de atribuições. Dentro da função, no caso base da recorrência, é feita uma verificação se o vetor de atribuições está vazio, ou seja, está na folha e pode retornar, também é verificado se os conflitos do nó atual são menores que a solução ótima do subproblema, todas as afinidades estão juntas e se os grupos estão preenchidos. No corte de viabilidade são retiradas as possibilidades de haver nós inviáveis, para esse caso, as afinidades não estarem juntas no mesmo grupo. No corte de otimalidade, é verificado se o número de conflitos atual é maior que o da solução ótima do subproblema, caso seja, é feita a poda. Então, é retirado o primeiro valor do vetor de atribuições e armazenado em uma variável. Com isso, são feitas duas chamadas recursivas, para o filho da direita e da esquerda. Para o filho da esquerda, esse elemento retirado do vetor de atribuições é adicionado ao grupo A, na direita esse elemento é adicionado ao grupo B.

V. TESTES

Após o desenvolvimento do programa, foram feitos alguns testes com relação ao número de nós da árvore resultante e o tempo de execução de cada exemplo. Segue os exemplos:

• Teste 1

Arquivo de entrada:

5 7 3

1 2

4 1

1 3

3 5

2 3

3 4

4 5

1 4

2 3

3 5

- Teste 2

Arquivo de entrada:

15 8 4
3 4
8 14
3 6
6 14
10 13
2 12
8 3
9 11
4 14
12 13
8 15
1 3

- Teste 3

Arquivo de entrada:

30 10 10
1 15
9 3
12 21
24 4
17 6
9 14
17 27
2 28
4 7
29 7
12 5
27 14
17 1
24 29
20 4
1 13
10 21
1 2
3 22
24 28

VI. CONCLUSÃO

Com isso, é evidente que o objetivo do trabalho desenvolvido foi alcançado. A modelagem e implementação de um programa utilizando Branch and Bound se mostraram eficazes para o problema de minimização de conflitos. Os testes realizados demonstraram a capacidade do programa em encontrar a solução ótima, minimizando o número de conflitos e respeitando as limitações impostas.

Tabela I
TABELA FUNÇÃO LIMITANTE ALUNO

	Exemplo 1	Exemplo 2	Exemplo 3
Nós	5	35	3345
Tempo	30us	349us	173ms

Tabela II
TABELA FUNÇÃO LIMITANTE PROFESSOR

	Exemplo 1	Exemplo 2	Exemplo 3
Nós	7	81	7115
Tempo	27us	1,52ms	346ms