

Programação Inteira - 2IS

José Joaquim de Andrade Neto

February 2018

1 Conjunto Independente de Vértices

Dado um grafo simples $G(V,A)$ um **Conjunto Independente de Vértices** - *Independent Set* - é um subconjunto $X \subseteq V$ de vértices tal que para todo par de vértice $u, v \in X$ não há uma aresta (u, v) no grafo original. O problema do **Conjunto Máximo Independente de Vértices** - *Maximum Independent Set* (MIS) - é um problema que consiste em achar o conjunto máximo (isto é, que não dá para ser expandido) de vértices independentes. Esse problema é \mathcal{NP} -Difícil para grafos em geral, o que proporcionou diversos estudos na literatura sobre como achar tais conjuntos em um tempo viável. Tais estudos levam em consideração modelagens exatas ou heurísticas, na tentativa de geração de soluções.

Uma formulação inteira para o MIS é denotada por IP:MIS e é computada por

$$\begin{array}{ll} \max & \sum_{u \in V} u_i \\ \text{s.a.} & u_i + u_j \leq 1 \quad \forall (u, v) \in A \\ & u_i \in \{0, 1\} \quad \forall u \in V, \end{array}$$

onde u_i tem valor 1 se e somente se ele pertence ao conjunto independente.

O problema de achar *dois* conjuntos independentes de vértices (2IS) é definido de forma similar: a solução gerada deve retornar dois conjuntos independentes e disjuntos. Essa variante possui modelagem similar, uma vez que o método usado para gerar a solução é bastante semelhante. Uma vez que um conjunto foi gerado, exclui-se os vértices que o compõe, e tenta-se achar um novo conjunto com o novo grafo, gerado a partir da exclusão dos vértices. Ao final,

se o grafo for viável, haverão dois conjuntos independentes.

2 Experimentos

Os experimentos consistiram na realização de quatro tarefas diferentes. Todos os experimentos realizados nesse trabalho usou o resolvidor Gurobi para todas as modelagens lineares aqui propostas, sendo que a complexidade do modelo aumentava à medida que as tarefas evoluíam. Primeiramente, construiu-se o modelo do problema *2IS* apresentado anteriormente e foram realizados testes com a finalidade de observar as respostas e o tempo de execução consumido para a geração da solução. Depois, a modelagem foi adaptada para a rotina CLQ1, na tentativa de achar soluções viáveis e que fossem ótimas.

Com o objetivo de verificar o impacto (no modelo e na otimização) da inserção de rotinas de corte, um algoritmo de *Branch-and-Cut* (B&C) foi gerado e comparado com os resultados já obtidos anteriormente. Por ultimo, foram implementadas as heurísticas primais na tentativa de achar boas soluções viáveis ao problema do *2IS*.

2.1 Modelagem Básica

2.2 Rotina CLQ1

2.3 Branch-and-Cut

2.4 Heurísticas

3 Conclusões