

# Iterated Local Search no Problema da Clique Máxima

Gabriel Cardoso de Carvalho

**Resumo:** Esse artigo mostra os resultados de uma implementação do Iterated Local Search no problema da Clique Máxima, comparando com os resultados de outras implementações desse mesmo método e de outros métodos como o genético.

## 1 Introdução

O problema de encontrar a Clique Máxima (CM) é extremamente conhecido e estudado, pois inúmeros problemas práticos de diversas áreas diferentes, como biologia computacional, economia e análise de redes sociais podem ser modelados como CM. Além disso, a sua versão de decisão foi um dos primeiros problemas a serem provados NP-Completo.

Ele pode ser definido da seguinte maneira, seja o grafo  $G = (V, E)$  onde  $V = 1, 2, \dots, n$  é o conjunto de vértices e  $E \subseteq V \times V$  é o conjunto de arestas, uma Clique  $C \subseteq V$  é tal que  $\forall i, j \in C, (i, j) \in E$ , ou seja, todos os vértices em  $C$  são adjacentes entre si. Ou ainda,  $C$  é um subgrafo completo de  $G$ . O problema da clique máxima é o problema de encontrar a clique de cardinalidade máxima do grafo  $G$ .

Diversas soluções foram propostas, tanto métodos exatos quanto heurísticas e metaheurísticas [2, 3, 4]. As propostas no geral tendem a utilizar o sistema *breach and bound* nos métodos exatos e heurísticas gulosas em buscas locais, preferindo vértices de maior grau. De maneira geral, os algoritmos gulosos partem de uma clique  $C$  inicial que contém apenas um vértice e um conjunto  $N_C$  de vértices  $v \in V$  que são os vértices vizinhos à  $C$ , ou seja,  $\forall u \in C, v$  é adjacente à  $u$ . Daí o algoritmo adiciona vértices de  $N_C$  em  $C$ , escolhendo sempre o vértice de  $N_C$  que tem o maior grau no subgrafo  $G(N_C)$ , até que  $C$  seja *maximal*, ou seja, até que não exista uma clique  $C'$  maior que  $C$  tal que  $C \subseteq C'$ . Em outros trabalhos esse método é chamado de *Busca Local 1-opt* [5].

A metaheurística implementada nesse artigo é a *Iterated Local Search (ILS)*, que pode ser resumida como uma metaheurística que cria, de maneira iterativa, uma sequência de soluções geradas por uma heurística interna (ou busca local) [6]. É esperado que as soluções providas pelo ILS sejam melhores do que uma simples repetição da heurística de maneira aleatória.

Nesse artigo é proposta uma implementação do ILS focada na aleatoriedade,

de modo a comparar seu desempenho com métodos gulosos, como a implementação do IKLS de *Katayama* [1], que utiliza a *Busca Local k-opt (KLS)* [5] como busca local, que é uma generalização da busca local 1-opt, onde adiciona-se  $k$  vértices à clique por vez, permitindo retirar vértices da clique para isso, de maneira dinâmica, ou seja, o  $k$  não é fixo, e uma perturbação baseada na *menor conectividade por arestas (LEC-KLS)*, onde escolhe-se um vértice  $v$  que não pertence à  $C$ , de maneira que  $v$  seja adjacente à menor quantidade de vértices de  $C$ . Então, adiciona-se  $v$  à  $C$  e remove de  $C$  todos os vértices que não são vizinhos à  $v$ .

A seção 2 apresenta como foi feita a implementação do ILS, enquanto a seção 3 cobre toda a implementação e os resultados experimentais. A seção 4 apresenta as conclusões e os trabalhos futuros.

## 2 ILS

```

Data: Grafo  $G$ , inteiro  $n_{iter}$ 
 $s_0 = \text{GeraSolucaoInicial}(G)$ ;
 $s^* = \text{BuscaLocal}(s_0)$ ;
 $k = 0$ ;
while  $k$  for menor que  $n_{iter}$  do
     $s' = \text{Perturbacao}(s^*)$ ;
     $s^{*'} = \text{BuscaLocal}(s')$ ;
     $s^* = \text{CritérioAceitacao}(s^*, s^{*'})$ ;
     $k_{++}$ ;
end

```

### Algorithm 1: Estrutura do ILS

Esta implementação do ILS segue o padrão de Glover [6] utilizando o algoritmo 1, e os detalhes de cada função são descritos nas subseções seguintes.

### 2.1 Geração da Solução Inicial

### 2.2 Busca Local

### 2.3 Perturbação

### 2.4 Critério de Aceitação

## 3 Resultados Experimentais

## 4 Conclusão

## References

- [1] Katayama, Kengo, Masashi Sadamatsu, and Hiroyuki Narihisa. "Iterated k-opt local search for the maximum clique problem." *Lecture Notes in Com-*

*puter Science* 4446 (2007): 84.

- [2] Wu, Qinghua, and Jin-Kao Hao. "A review on algorithms for maximum clique problems." *European Journal of Operational Research* 242.3 (2015): 693-709.
- [3] I.M. Bomze, M. Budinich, P.M. Pardalos, and M. Pelillo. The maximum clique problem. In D.-Z. Du and P.M. Pardalos, editors, *Handbook of Combinatorial Optimization (suppl. Vol. A)*, pp. 1-74. Kluwer, 1999.
- [4] D.S. Johnson and M.A. Trick. *Cliques, Coloring, and Satisfiability*. Second DIMACS Implementation Challenge, DIMACS Series in Discrete Mathematics and Theoretical Computer Science. American Mathematical Society, 1996.
- [5] K. Katayama, A. Hamamoto, and H. Narihisa. An effective local search for the maximum clique problem. *Information Processing Letters*, Vol. 95, No. 5, pp. 503-511, 2005.
- [6] Glover, Fred W., and Gary A. Kochenberger, eds. *Handbook of metaheuristics*. Vol. 57. Springer Science & Business Media, 2006.