

Trabalho Prático 2: Problema do Clique Máximo

Ivan Gomes da Cruz e Rachel Freire Biezuner

Introdução

O problema do clique máximo é um problema clássico de otimização que possui diversas aplicações. Ele consiste em encontrar o maior clique (conjunto de vértices em que quaisquer 2 pares estão conectados) máximo, ou seja, de maior tamanho, de um grafo e não deve ser confundido com o problema do clique maximal (achar um clique que só está contido em si mesmo). Porém, este problema é NP-difícil, o que implica que não conhecemos um algoritmo de tempo polinomial para solucioná-lo. Portanto, devemos utilizar heurísticas para resolvê-lo. Uma dessas heurísticas utiliza o ACO (Ant Colony Optimization) para aproximar uma solução. Ele se trata de um algoritmo estocástico que simula formigas em busca de formigas que utilizam feromônios para se comunicar por meio de estigmergia. A seguir, explicamos como nosso grupo utilizou esse algoritmo para solucionar o problema do Clique Máximo.

Implementação

Para a implementação foi utilizado como referência o algoritmo apresentado em [1]. Ele funciona da seguinte maneira: repetimos um ciclo em que mandamos todas as formigas de nosso formigueiro. As formigas são inicializadas em uma posição aleatória de nosso grafo e caminham nele sempre escolhendo vértices de uma coleção de candidatos. Esses candidatos fazem parte de uma lista de vértices que, se adicionados, faz o conjunto de vértices explorados permanecer um clique (um vértice único é um clique, por definição). As formigas escolhem os vértices entre os candidatos de acordo com a quantidade de feromônios depositada neles. Escolhemos depositar o feromônio nos vértices e não nas arestas, porque a implementação é mais simples e a complexidade menor (já que o número de arestas é bem maior que o de vértices). Após cada formiga percorrer seu caminho, escolhemos a melhor solução dentre esses caminhos para atualizar o feromônio, que depende do tamanho da melhor solução global e da melhor solução na iteração atual. Após esses passos, inicia-se o ciclo novamente.

Um detalhe de implementação importante de se notar é que escolhemos um τ_{\min} de 0.01 e um τ_{\max} de 6, para impedir que nossas trilhas de feromônios estourassem para um valor muito grande ou atingissem um valor ínfimo, que impedissem as próximas formigas de acessar certos vértices. Além disso, utilizamos

um fator de evaporação, definido experimentalmente, que favorece a exploration, principalmente de soluções novas

A probabilidade de se seguir um caminho foi calculada como:

$(\text{quantidade de feromônio neste caminho}) / (\text{soma dos feromônios dos caminhos possíveis})$

A fitness de um clique encontrado é dada pela diferença do seu tamanho para o melhor encontrado até o momento, portanto neste caso queremos minimizar a fitness.

$$fitness = len(largest\ clique) - len(clique)$$

Já a quantidade de feromônio depositada no clique é dada por:

$$pheromone = 1 / (1 + fitness)$$

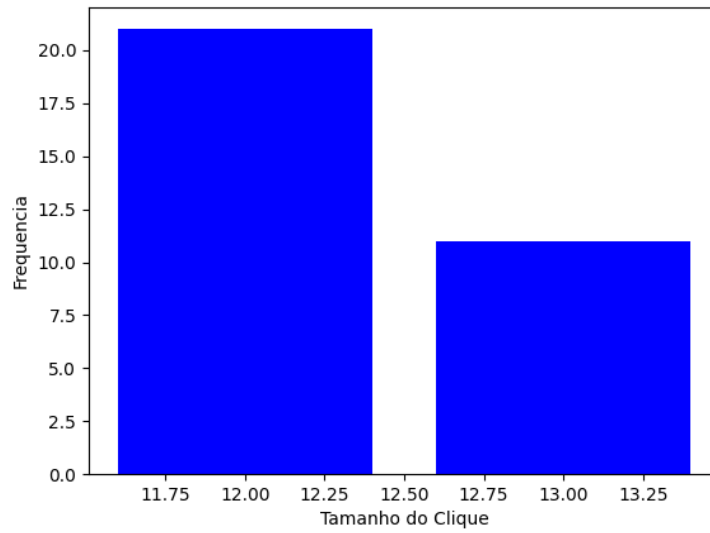
Experimentos

Como o ACO se trata de um algoritmo estocástico, precisamos rodá-los inúmeras vezes para poder analisar os resultados. Como regra do dedão, escolhemos 30 vezes.

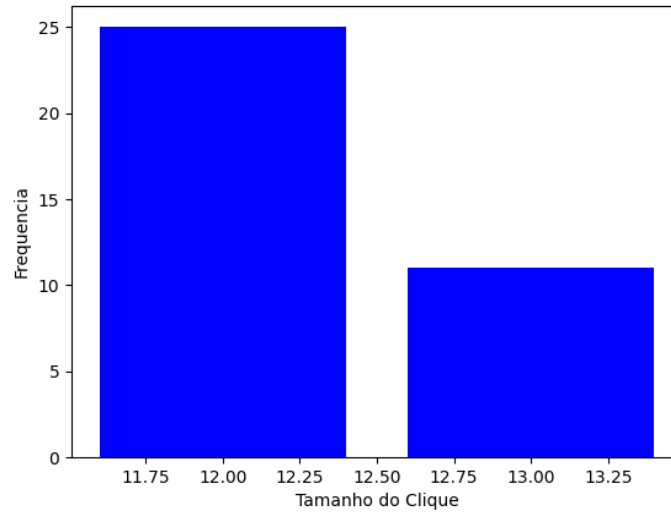
Dataset de 500 vértices e 62624 arestas

Inicialmente, rodamos experimentos sobre o dataset de 500 vértices. Mantivemos os valores de ciclos e formigas fixos em 100 e 30, respectivamente e variamos a taxa de evaporação em 0.01, 0.05 e 0.1.

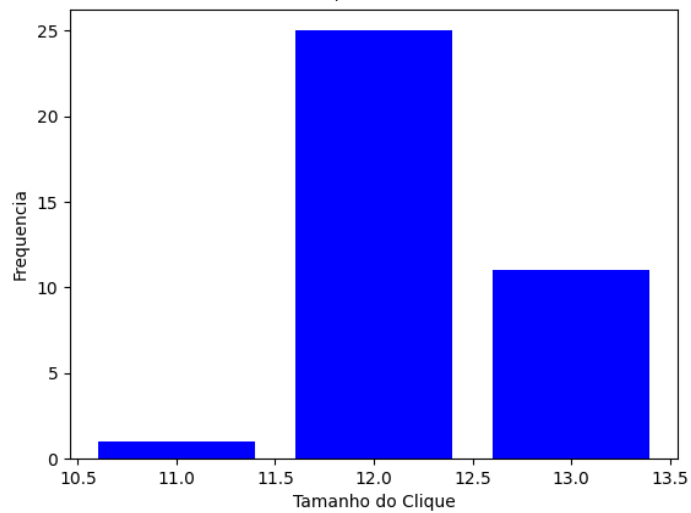
500_62624.txt, 100 Ciclos, 30 Formigas, Taxa de Evaporação: 0.01
Média: 12.25, Desvio Padrão: 0.43



500_62624.txt, 100 Ciclos, 30 Formigas, Taxa de Evaporação: 0.05
Média: 12.11, Desvio Padrão: 0.31

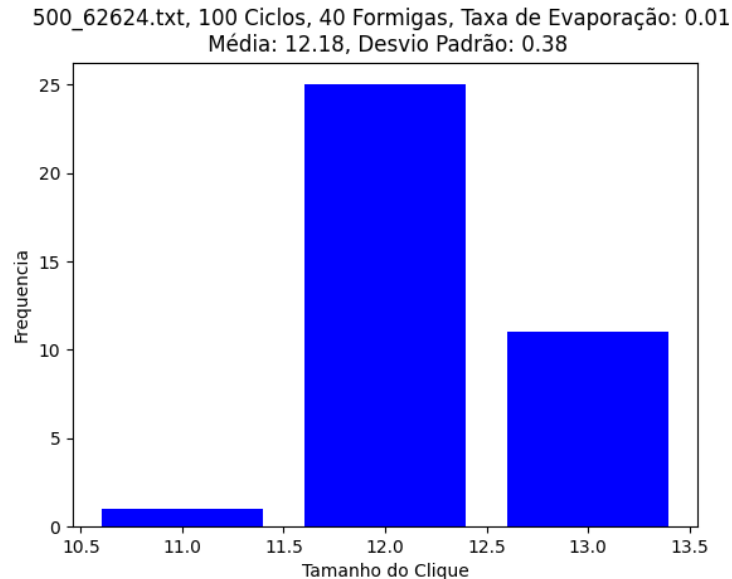
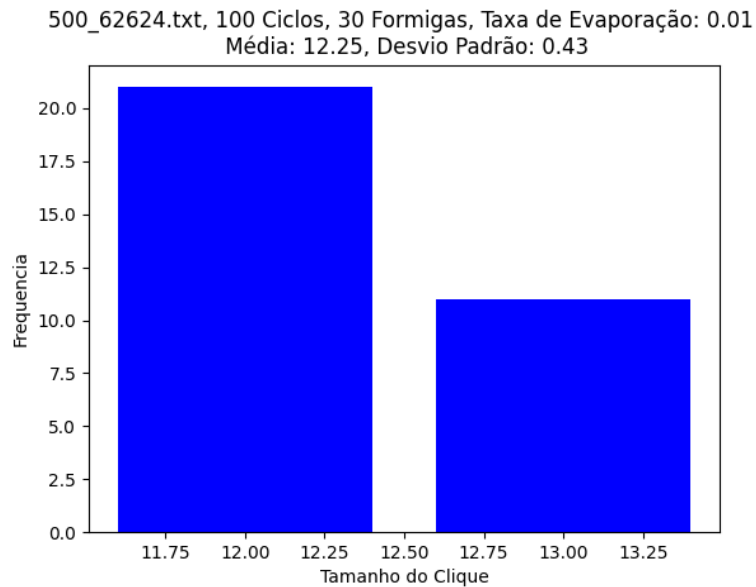


500_62624.txt, 100 Ciclos, 30 Formigas, Taxa de Evaporação: 0.1
Média: 12.07, Desvio Padrão: 0.37

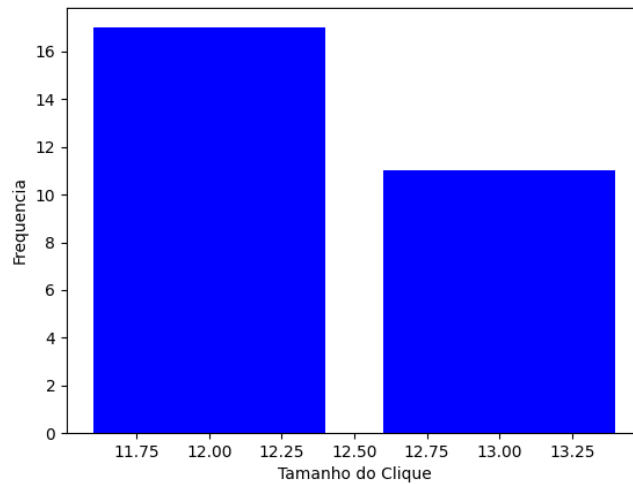


Como pôde ser observado, o melhor valor obtido foi com uma taxa de evaporação de 0.01, pois a média mais se aproximou de 13, que é o valor do clique máximo deste dataset. No entanto, a maioria dos valores ainda atinge apenas um clique de tamanho 12. Isso indica que devemos experimentar mais com os outros parâmetros e manter a taxa de evaporação em 0.01.

Em seguida, variamos o número de formigas de nosso algoritmo em 30, 40 e 100. Obtivemos os seguintes resultados

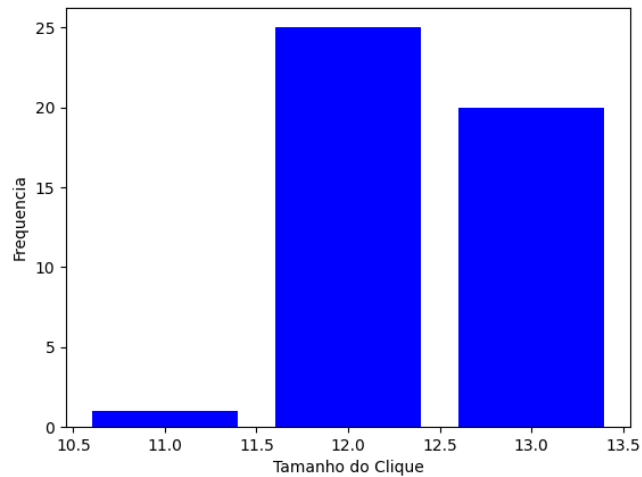


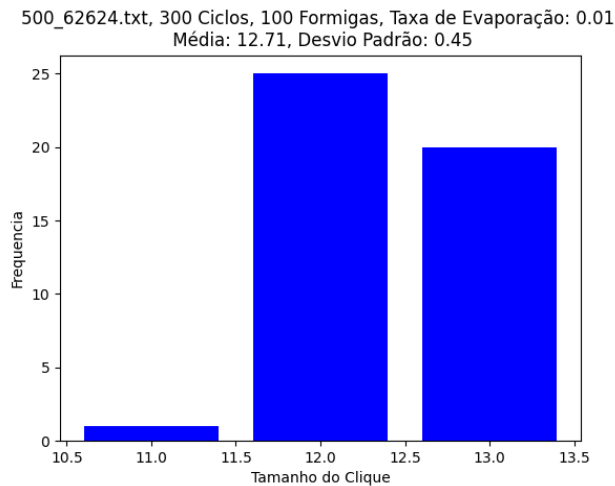
500_62624.txt, 100 Ciclos, 100 Formigas, Taxa de Evaporação: 0.01
Média: 12.39, Desvio Padrão: 0.49



Isso nos leva à conclusão que aumentar o número de formigas aumenta nossa média de valores. Vamos manter o número de formigas então fixo em 100 e variar agora o número de ciclos em 50 e 300. O melhor resultado obtido foi para 300 ciclos, embora ainda o algoritmo não tenha convergido totalmente. Esses dados experimentais indicam que aumentar o número de formigas e ciclo em geral é uma boa decisão, pois faz o ACO ser mais confiável e acertar mais vezes.

500_62624.txt, 50 Ciclos, 100 Formigas, Taxa de Evaporação: 0.01
Média: 12.29, Desvio Padrão: 0.45



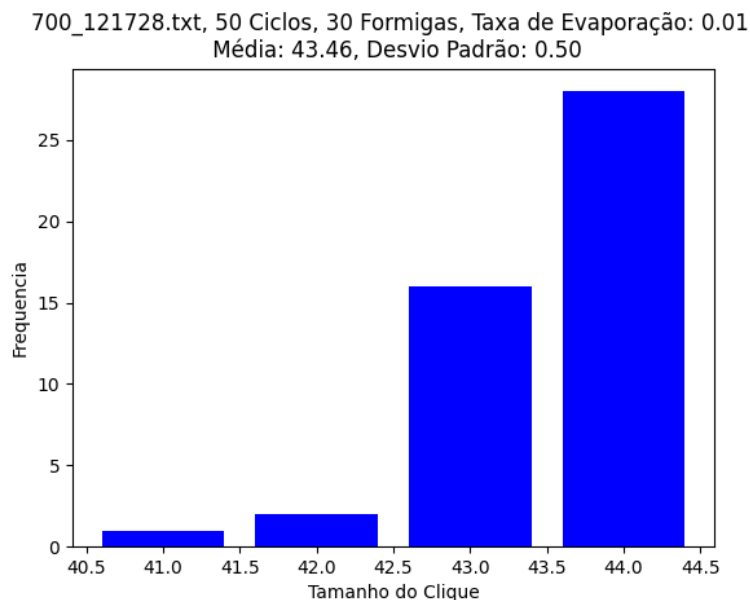


Após nossa análise definimos que o melhor conjunto de parâmetros para esse dataset seria:

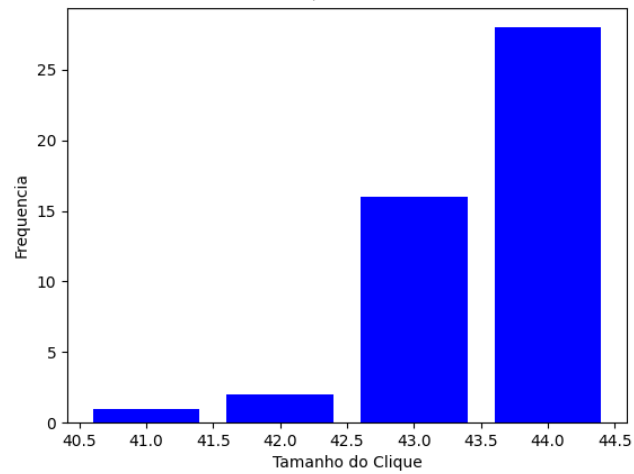
- Taxa de Evaporação: 0.1
- Número de formigas: 100
- Número de iterações: 300

Dataset de 700 vértices e 121728 arestas

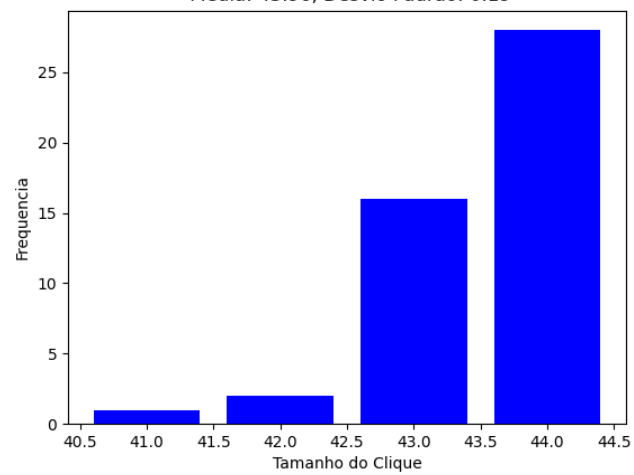
Vamos experimentar em um outro dataset para ver se nossos insights se traduzem para ele. Começamos testando a taxa de evaporação, com os valores 0.01, 0.05 e 0.1, assim como no outro exemplo. Os resultados foram idênticos nesse caso, então escolhemos 0.05.



700_121728.txt, 50 Ciclos, 30 Formigas, Taxa de Evaporação: 0.05
Média: 43.96, Desvio Padrão: 0.19

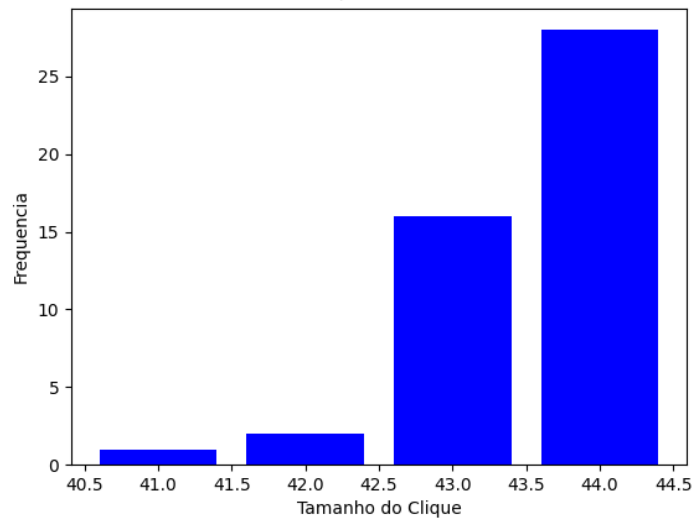


700_121728.txt, 50 Ciclos, 30 Formigas, Taxa de Evaporação: 0.1
Média: 43.96, Desvio Padrão: 0.19

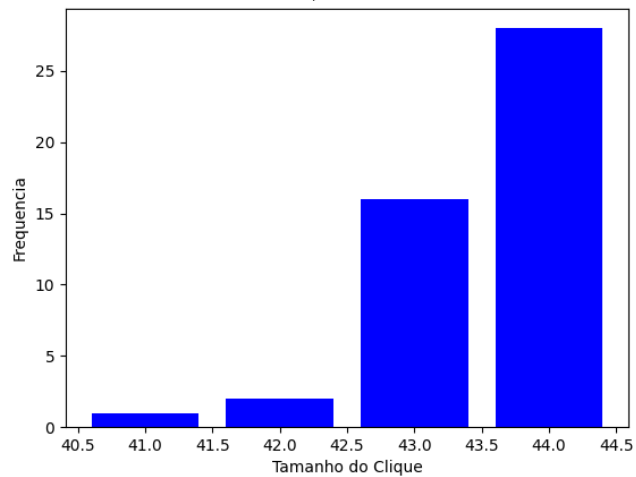


Após isso, passamos para o número de formigas. Novamente o resultado empatou. Então mantemos 30 formigas.

700_121728.txt, 50 Ciclos, 10 Formigas, Taxa de Evaporação: 0.05
Média: 43.18, Desvio Padrão: 0.71

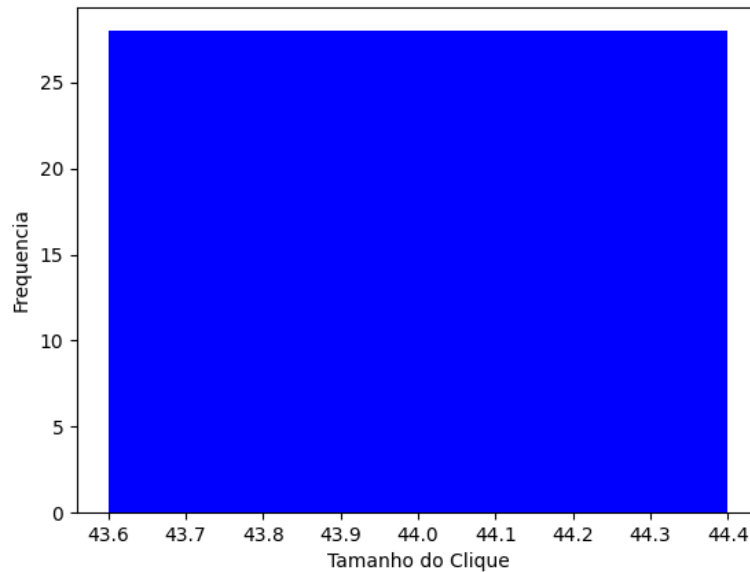


700_121728.txt, 50 Ciclos, 20 Formigas, Taxa de Evaporação: 0.05
Média: 43.79, Desvio Padrão: 0.41



Por fim, aumentamos o número de ciclos. Com 100 ciclos conseguimos um desvio padrão 0 e todas as rodadas acertam.

700_121728.txt, 100 Ciclos, 30 Formigas, Taxa de Evaporação: 0.05
Média: 44.00, Desvio Padrão: 0.00



Após nossa análise definimos que o melhor conjunto de parâmetros para esse dataset seria:

- Taxa de Evaporação: 0.05
- Número de formigas: 30
- Número de iterações: 100

Conclusão

Chegamos à conclusão que o problema do clique máximo depende bastante do grafo em mãos e certas instâncias convergem mais facilmente, enquanto em outras o algoritmo atinge apenas um máximo local. Por isso, foi atestada a necessidade de rodar o ACO com mais ciclos para evitar esses ótimos locais. Além disso, o ACO parece resolver muito bem esse problema NP-Difícil, o que o torna uma ferramenta super útil de se ter.

Bibliografia

[1] <https://liris.cnrs.fr/Documents/Liris-1847.pdf>

[2] <https://www.geeksforgeeks.org/cliques-in-graph/>

[3] <https://www.geeksforgeeks.org/introduction-to-ant-colony-optimization/>