Maximum Clique Problem

In computer science, the clique problem is the computational problem of finding cliques (subsets of vertices, all adjacent to each other, also called complete subgraphs). Common formulations of the clique problem include finding a maximum clique.

The clique decision problem is NP-complete (one of Karp’s 21 NP-complete problems).

We chosed to implement two solutions for this problem: an

Often, it is important to find not just one large clique, but all maximal cliques. Many algorithms are now known for this problem.

We chosed to implement two solutions for this problem:

* A simple recursive solution (that it’s not feasible for big graphs)
* A random search algorithm

**The recursive solution:**

One of the most successful in practice is the **Bron-Kerbosch algorithm**, a simple backtracking procedure that recursively solves subproblems specified by three sets of vertices: the vertices that are required to be included in a partial clique, the vertices that are to be excluded from the clique, and some remaining vertices whose status still needs to be determined. This algorithm performs the following steps:

BronKerbosch1(R, P, X):

if P and X are both empty:

report R as a maximal clique

for each vertex v in P:

BronKerbosch1(R ⋃ {v}, P ⋂ N(v), X ⋂ N(v))

P := P \ {v}

X := X ⋃ {v}

**The random search algorithm:**

We implemented a simple random search algorithm for the maximum clique problem.

This algorithm performs the following steps:

1) Create an initial clique using a greedy algorithm based on non-increasing degrees of the nodes

and call it gBest

2) Randomly remove two vertics from the clique created in step one.

3) Add vertices to the incomplete clique returned by step two in order of non-increasing degrees.

4) If the complete clique formed in step 3 is better than gBest, gBest = (3).

5) Continue from step 2 till some termination criteria (Number of Iterations)

Results:

1. The recursive solution works only on small graphs, it’s not feasible for large problems.
2. The random solution obtained following results:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Instance | Best Known | Nodes | Edges | Obtained Result | Time |
| C125.9 | 34 | 125 | 6963 | 34 | 240 ms |
| C250.9 | 44 | 250 | 27984 | 43 | 245 ms |
| C500.9 | 57 | 500 | 112332 | 54 | 288 ms |
| C100.9 | 68 | 1000 | 450079 | 62 | 564 ms |
| MANN\_a45 | 345 | 1035 | 533115 | 342 | 3600 ms |

Bibliography:

1. Listing All Maximal Cliques in Sparse Graphs in Near-Optimal Time: David Eppstein, Maarten Loffler and Darren Strash.
2. A review on algorithms for maximum clique problems
3. Solving the maximum clique problem using a tabu search approach