Analysis of Charikar's Greedy Approximation Algorithm for Densest Subgraph

Aris Anagnostopoulos

We are given a simple undirected graph $G = (V, E)$. In graph community detection, the goal is to discover subsets of nodes that are connected highly connected with each other.

An extreme example of a community is a *clique*: A subset of nodes $S \subseteq V$ in which each of the $\binom{|S|}{2}$ edges among the nodes of S exist.

Yet, trying to find cliques in a graph is not very useful for community detection, as it is very restrictive. Furthermore, finding a clique of maximum size is not only NP-hard, but also very hard to approximate: essentially it cannot be approximated better than a factor of $\sqrt{|V|}$.

Therefore we will relax the definition of a community. There are many ways to do this. Here we will choose one of them: Given a set of nodes $S \subseteq V$, we define its *sparsity* to be:

$$f(S) = \frac{|E(S)|}{|S|},$$

where $E(S) = \{\{u, v\} \in E : u \in S, v \in S\}$, is the set of edges of G with both endpoints in S.

We also define $\deg_S(v) = |\{\{v, u\} \in E : u \in S\}|$, to be the degree of node v restricted to the nodes in S.

Then notice that $f(S)$ is also half the average degree among the nodes in S:

$$\frac{\sum_{v \in S} \deg_S(v)}{|S|} = \frac{2\,|E(S)|}{|S|}.$$

We can now define the problem of finding the *densest subgraph*: Find a set $S \subseteq V$ that maximizes $f(S)$.

It turns out that there exists an algorithm based on linear programming that solves optimally this problem in polynomial time. Here instead we will see a simple and elegant greedy algorithm, which provides a 2-approximation to the optimal solution.

1. **Function** GreedyDensestSubgraph(G)
2. **Input**: $G = (V, E)$: Simple undirected graph
3. **Output**: A set of nodes $S \subseteq V$
4. $S \leftarrow V$
5. $S^G \leftarrow V$
6. **while** $|S| > 1$
7. $v = \arg\min_{v \in S} \deg_S(v)$
8. $S \leftarrow S \setminus \{v\}$
9. **if** $f(S) \geq f(S^G)$
10. $S^G \leftarrow S$
11. **end if**
12. **end while**
13. **return** S^G

In words, the algorithm starts with S being the entire set V and it keeps removing from the graph induced by S a node with minimum degree, until S remains with one node. It then returns the set S that during the execution had the highest density.

Note that S^G is the returned solution, and let S^* be optimal solution to the problem, and $\text{OPT} = f(S^*)$. Then we have:

Theorem 1. *Algorithm* GreedyDensestSubgraph *is a 2-approximation algorithm, that is,*

$$f(S^G) \geq \frac{1}{2}\text{OPT}.$$

Proof. The problem when we analyze approximation algorithms is that we do not know what is the optimal solution. Therefore, typically what we do is to provide an upper bound on the value of the optimal solution. To this end we first show the following.

Claim 2. *For each $v \in S^*$ we have that* $\deg_{S^*}(v) \geq \text{OPT}$.

Let's try to prove the claim. From the fact that S^* is optimal, we have that

$$\frac{|E(S^*)|}{|S^*|} \geq \frac{|E(S^* \setminus \{v\})|}{|S^* \setminus \{v\}|}.$$

If we remove v from S^*, then we remove $\deg_{S^*}(v)$ edges from $E(S^*)$. Therefore we obtain that

$$\frac{|E(S^*)|}{|S^*|} \geq \frac{|E(S^*)| - \deg_{S^*}(v)}{|S^*| - 1}.$$

If we simplify this expression we obtain Claim 2.

The next step is to show a lower bound for the output of our solution. Let v_0 be the first node of S^* that we remove from S during the execution of the algorithm and let S_0 be the set S just before we removed v_0 from S. In particular, we have that $S^* \subseteq S_0$.

We will now compare the value $f(S_0)$ with OPT. We have

$$f(S_0) = \frac{|E(S_0)|}{|S_0|} = \frac{\frac{1}{2}\sum_{v \in S_0} \deg_{S_0}(v)}{|S_0|}.$$

But the next node that we removed from S_0 was v_0, which means that for each $v \in S_0$ we have that $\deg_{S_0}(v) \geq \deg_{S_0}(v_0)$. Therefore, we obtain

$$f(S_0) \geq \frac{\frac{1}{2}\sum_{v \in S_0} \deg_{S_0}(v_0)}{|S_0|} = \frac{\frac{1}{2}|S_0| \cdot \deg_{S_0}(v_0)}{|S_0|} = \frac{1}{2}\deg_{S_0}(v_0) \geq \frac{1}{2}\deg_{S^*}(v_0) \geq \frac{1}{2}\mathrm{OPT},$$

where the second-to-last inequality follows from the fact that $S^* \subseteq S_0$ and the last inequality follows from Claim 2, after noticing that $v_0 \in S^*$.

Now, we notice that by the definition of our solution S^G, we have that $f(S^G) \geq f(S_0)$: S^G is the best among all the intermediate solutions including S_0. This conclude the proof. □