Large-Scale Density-Friendly Decomposition via Convex Programming [1]

Mauro Sozio

Telecom ParisTech

January 16, 2020
LP formulation for Densest Subgraph

\[
\text{LP}(G) \quad \max \sum_{e \in E} w_e x_e \\
\text{s.t.} \quad x_e \leq y_u, \quad \forall u \in e \\
\sum_{u \in V} y_u = 1, \\
x_e, y_u \geq 0, \quad \forall u \in V, e \in E
\]
Dual of the LP

\[ \text{DP}(G) \]

\[
\begin{align*}
\text{min} & \quad \rho \\
\text{s.t.} & \quad \rho \geq \sum_{e:u\in e} \alpha_u^e, \quad \forall u \in V \\
& \quad \sum_{u\in e} \alpha_u^e \geq 1, \quad \forall e \in E \\
& \quad \alpha_u^e \geq 0, \quad \forall u \in e \in E
\end{align*}
\]
Convex Program $\text{CP}(G)$

Let $Q_G(\alpha) := \sum_{u \in V} r(u)^2$, where $(r, \alpha)$ is an invariant pair, i.e., $r(u) = \sum_{e \in E: u \in e} \alpha_e^u$ and $\alpha_e^u + \alpha_e^v = 1$ for every $e = uv \in E$.

The convex program $\text{CP}(G)$ is defined as follows:

$\text{CP}(G) := \min \{ Q_G(\alpha) : \alpha \text{ is feasible for } \text{DP}(G) \}$. 
The Frank-Wolfe Algorithm [2]

1: **Input:** function $f$ convex and continuously differentiable, a compact convex set $D$, integer $T$
2: Let $x^{(0)} \in D$
3: for $t = 1, \ldots, T$ do
4: \hspace{1em} $\gamma_t \leftarrow \frac{2}{t+2}$
5: \hspace{1em} Compute $s := \arg \min_{s \in D} \langle s, \nabla f(x^{(k)}) \rangle$
6: \hspace{1em} Update $x^{(k+1)} = (1 - \gamma_t)x^{(k)} + \gamma_s$
Frank-Wolfe-Based Algorithm for Densest Subgraph

1: for each $e = uv$ in $E$ in parallel do
2: \[ \alpha_u^e(0), \alpha_v^e(0) \leftarrow \frac{1}{2} \]
3: for each $u \in V$ in parallel do
4: \[ r^{(0)}(u) \leftarrow \sum_{e \in E: u \in e} \alpha_u^e(0) \]
5: for each iteration $t = 1, \ldots, T$ do
6: \[ \gamma_t \leftarrow \frac{2}{t+2} \]
7: for each $e$ in $E$ in parallel do
8: \[ x \leftarrow \arg \min_{v \in e} r^{(t-1)}(v) \]
9: for each $u \in e$ do
10: \[ \hat{\alpha}_u^e \leftarrow 1, \text{ if } u = x \text{ and } 0 \text{ otherwise.} \]
11: \[ \alpha^{(t)} \leftarrow (1 - \gamma_t) \cdot \alpha^{(t-1)} + \gamma_t \cdot \hat{\alpha} \]
12: for each $u \in V$ in parallel do
13: \[ r^{(t)}(u) \leftarrow \sum_{e \in E: u \in e} \alpha_u^e(t) \]
14: return $(\alpha^{(t)}, r^{(t)})$
Convergence

Theorem 1

(Convergence of the Frank-Wolfe-based Algorithm.) Let $G = (V, E)$ be an undirected graph with maximum degree $\Delta$. Let $(r^*, \alpha^*)$ be an invariant pair for $G$ where $\alpha^*$ is an optimal solution for $\mathcal{CP}(G)$. In Algorithm 2, for any $\epsilon > 0$ for any $t > \frac{4\Delta |E|}{\epsilon^2}$, we have $\|r(t) - r^*\|_2 \leq \epsilon$. 
Experiments: Settings

<table>
<thead>
<tr>
<th>networks</th>
<th>$n$</th>
<th>$m$</th>
</tr>
</thead>
<tbody>
<tr>
<td>LiveJournal</td>
<td>4,036,538</td>
<td>34,681,189</td>
</tr>
<tr>
<td>Wikipedia</td>
<td>2,080,370</td>
<td>42,336,692</td>
</tr>
<tr>
<td>Orkut</td>
<td>3,072,627</td>
<td>117,185,083</td>
</tr>
<tr>
<td>Twitter</td>
<td>52,579,683</td>
<td>1,614,106,500</td>
</tr>
<tr>
<td>Friendster</td>
<td>124,836,180</td>
<td>1,806,067,135</td>
</tr>
<tr>
<td>gsh-2015</td>
<td>988,490,691</td>
<td>25,690,705,119</td>
</tr>
</tbody>
</table>

Table: Our set of large graphs.

Used a linux machine with 2 processors Intel Xeon CPU E5-2660 @ 2.60 GHz with 10 cores split in 2 threads each, as well as 64G of RAM DDR4 2133 MHz. We employ 10 threads.
Convergence to the $r^*$ vector

where $r^t$ is the $r$ vector at step $t$ and $r^G = r^*$. 
Densest Subgraph

How to extract the densest subgraph from a sufficiently good \( r^t \)? Use the fact that for any graph \( H_1, H_2 \), \(|\rho(H_1) - \rho(H_2)| \geq \frac{1}{|V|^2}\).
Densest Subgraph

How to extract the densest subgraph from a sufficiently good $r^t$? Use the fact that for any graph $H_1, H_2$, $|\rho(H_1) - \rho(H_2)| \geq \frac{1}{|V|^2}$.

From Thm. 1, after some step the nodes in a densest subgraph will have max $r^t$ score. Sort the $r^t$'s non-increasingly: $r(v_1)^t \geq r(v_2)^t, \ldots, \geq r(v_n)^t$. Recall that $r(v_1)^t$ gives an upper bound on the max density. Let $G_k$ be the graph induced by $v_1, \ldots, v_k$. As soon as we find a graph $G_k$ such that $|\rho(G_k) - r(v_1)^t| < \frac{1}{n^2}$ we know that $G_k$ is densest.
Densest Subgraph

How to extract the densest subgraph from a sufficiently good $r^t$? Use the fact that for any graph $H_1, H_2$, $|\rho(H_1) - \rho(H_2)| \geq \frac{1}{|V|^2}$.

From Thm. 1, after some step the nodes in a densest subgraph will have max $r^t$ score. Sort the $r^t$'s non-increasingly: $r(v_1)^t \geq r(v_2)^t, \ldots, \geq r(v_n)^t$. Recall that $r(v_1)^t$ gives an upper bound on the max density. Let $G_k$ be the graph induced by $v_1, \ldots, v_k$. As soon as we find a graph $G_k$ such that $|\rho(G_k) - r(v_1)^t| < \frac{1}{n^2}$ we know that $G_k$ is densest.

In practice: as soon as we find a “sufficiently” small stable subset $H$ we compute a densest subgraph in $H$ via the LP-based algorithm or maximum flow. This works well...
Computation of the densest subgraph

where $r^t$ is the $r$ vector at step $t$ and $r^G = r^*$. 
K-Core and Density-Friendly Decomposition

- *k*-core decomposition: compute for each node \( v \) the largest integer \( c_v \) such that \( v \) is in an induced subgraph with minimum degree \( c_v \).
- *k*-core decomp. might reveal the structural organization of a graph.
- It has been applied to the analysis of the internet topology [5], social network analysis [8], bionformatics [7], analysis of the human brain [4], as well as influence analysis [3]. There is also a startup based in NYC using such an algorithm [http://www.kcore-analytics.com/](http://www.kcore-analytics.com/).
- In the *k*-core decomposition, outer cores might be denser than inner cores, which is not ideal.
- Here, we will show that the Frank-Wolfe based algorithm actually computes a so-called diminishingly-dense decomposition where inner cores are always denser than outer cores.
Quotient Graph

Definition 2 (Quotient Graph)

Given an undirected graph $G = (V, E)$, and a subset $B \subseteq V$, the quotient graph of $G$ with respect to $B$ is a weighted graph $G \setminus B = (\hat{V}, \hat{E}, \hat{w})$, which is defined as follows.

- $\hat{V} := V \setminus B$.
- $\hat{E} := \{e \cap \hat{V} : e \in E, e \cap \hat{V} \neq \emptyset\}$, i.e., every edge $e \in E$ not contained in $B$ contributes towards $\hat{E}$.
- For $e' \in \hat{E}$, $\hat{w}(e') := |\{e \in E : e' = e \cap \hat{V}\}|$. 
Definition 3 (Diminishingly-dense Decomposition)

Given an undirected graph $G = (V, E, w)$, we define the diminishment-dense decomposition $B$ of $G$ as the sequence

$\emptyset = B_0 \subsetneq B_1 \subsetneq B_2 \subsetneq \cdots \subsetneq B_k = V$ as follows:

Initially, we set $B_0 = \emptyset$ and $G_0 := G$. For $i \geq 1$, if $B_{i-1} = V$, the decomposition is fully defined. Otherwise, let $G_i := G_{i-1} \setminus B_{i-1}$ be the quotient graph of $G_{i-1}$ with respect to $B_{i-1}$. Let $S_i$ be the maximal densest subset in $G_i$ (with respect to $w_i$). We define $B_i := B_{i-1} \cup S_i$. For each $i = 1, \ldots, k$, we denote $r_i = \rho_i(S_i)$. Moreover, we define the maximal density vector $r_G \in \mathbb{R}^V$ such that if $u \in S_i$, then $r_G(u) := r_i$. 

Mauro Sozio (LTCI TPT)
Definition 3 (Diminishingly-dense Decomposition)

Given an undirected graph $G = (V, E, w)$, we define the diminishingly-dense decomposition $\mathcal{B}$ of $G$ as the sequence

$\emptyset = B_0 \subsetneq B_1 \subsetneq B_2 \subsetneq \cdots \subsetneq B_k = V$ as follows:

Initially, we set $B_0 := \emptyset$ and $G_0 := G$. For $i \geq 1$, if $B_{i-1} = V$, the decomposition is fully defined.
Definition 3 (Diminishingly-dense Decomposition)

Given an undirected graph $G = (V, E, w)$, we define the diminishly-dense decomposition $\mathcal{B}$ of $G$ as the sequence

$\emptyset = B_0 \subsetneq B_1 \subsetneq B_2 \subsetneq \cdots \subsetneq B_k = V$ as follows:

Initially, we set $B_0 := \emptyset$ and $G_0 := G$. For $i \geq 1$, if $B_{i-1} = V$, the decomposition is fully defined.

Otherwise, let $G_i := G_{i-1} \setminus B_{i-1} = (V_i, E_i, w_i)$ be the quotient graph of $G_{i-1}$ with respect to $B_{i-1}$. Let $S_i$ be the maximal densest subset in $G_i$ (with respect to $w_i$). We define $B_i := B_{i-1} \cup S_i$. For each $i = 1, \ldots, k$, we denote $r_i = \rho_i(S_i)$. Moreover, we define the maximal density vector $r^G \in \mathbb{R}^V$ such that if $u \in S_i$, then $r^G(u) := r_i$. 
Properties of the Decomposition

Lemma 4 (the decomposition is unique)

*Given a graph $G$, there is a unique diminishingly-dense decomposition.*
Properties of the Decomposition

Lemma 4 (the decomposition is unique)

Given a graph $G$, there is a unique diminishingly-dense decomposition.

Lemma 5 (Diminishing $r_i$’s)

In the diminishingly-dense decomposition in Definition 3, if $B_i \subset V$, then

$$r_i > r_{i+1} \quad i = 1, \ldots, k - 1.$$ 

Lemma 6 (Diminishing Densities)

In the diminishingly-dense decomposition in Definition 3, if $B_i \subset V$, then

$$\rho(B_i) > \rho(B_{i+1}) \quad i = 1, \ldots, k - 1.$$
References I

[1] Maximilien Danisch, Hubert Chan, Mauro Sozio
Large Scale Density-Friendly Decomposition via Convex Programming

ICML, 2013.

Makse.
Identification of influential spreaders in complex networks.

O. Sporns.
Mapping the structural core of human cerebral cortex.
References II

A model of internet topology using k-shell decomposition.

Density-friendly graph decomposition.

An automated method for finding molecular complexes in large protein interaction networks.

[8] Seidman, Stephen B.
Network structure and minimum degree.