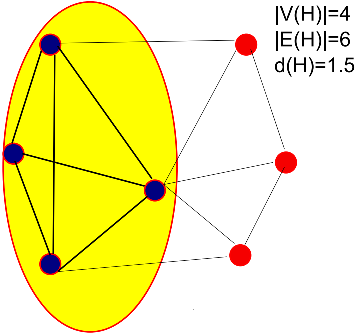
On Finding Dense Subgraphs

Barna Saha
(Joint work with Samir Khuller)

Department of Computer Science
University of Maryland, College Park, MD 20742

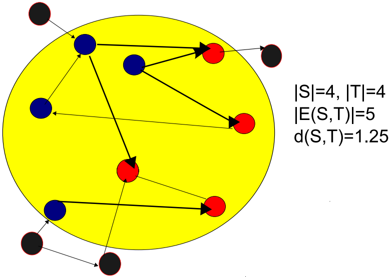
36th ICALP, 2009

Density for Undirected Graphs



- Given an undirected graph $G = (V, E)$, density of a subgraph $H \subseteq G$, is defined as $d_H = \frac{|E(H)|}{|V(H)|}$.

Density for Directed Graphs

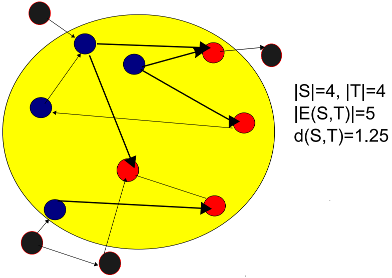


- Given two subsets of nodes S and T of a directed graph $G = (V, E)$, density is defined as

$$d(S,T) = \frac{|E(S,T)|}{\sqrt{|S||T|}}$$

- S and T may not be disjoint.

Density for Directed Graphs



- Proposed by Kannan and Vinay in 1999.
- Subsequently used in many other works [Charikar'00, Andersen'08].

Previous Results on Maximum Density Subgraphs

Undirected Graphs:

- Maximum density subgraph can be found in polynomial time for undirected graphs.
  - Combinatorial algorithms based on maxflow computations [Lawler'76, Goldberg'84].
  - Linear programming based algorithm [Charikar'00].
- Fast linear time algorithms for computing 2-approximate solutions [Kortsarz & Peleg'92, Charikar'00].

Previous Results on Maximum Density Subgraphs

Undirected Graphs:

- Maximum density subgraph can be found in polynomial time for undirected graphs.
  - Combinatorial algorithms based on maxflow computations [Lawler'76, Goldberg'84].
  - Linear programming based algorithm [Charikar'00].
- Fast linear time algorithms for computing 2-approximate solutions [Kortsarz & Peleg'92, Charikar'00].

Previous Results on Maximum Density Subgraphs

Undirected Graphs:

- Maximum density subgraph can be found in polynomial time for undirected graphs.
  - Combinatorial algorithms based on maxflow computations [Lawler'76, Goldberg'84].
  - Linear programming based algorithm [Charikar'00].
- Fast linear time algorithms for computing 2-approximate solutions [Kortsarz & Peleg'92, Charikar'00].

Previous Results on Maximum Density Subgraphs

Undirected Graphs:

- Maximum density subgraph can be found in polynomial time for undirected graphs.
  - Combinatorial algorithms based on maxflow computations [Lawler'76, Goldberg'84].
  - Linear programming based algorithm [Charikar'00].
- Fast linear time algorithms for computing 2-approximate solutions [Kortsarz & Peleg'92, Charikar'00].

Previous Results on Maximum Density Subgraphs

Directed graphs:

- Maximum density subgraph can be found in polynomial time.
  - Linear programming based algorithm by rounding the LP solution [Charikar'00].
    - Requires computations of $|V|^2$ linear programs.
  - No combinatorial algorithm known.
- $O(|V|^3 + |V|^2|E|)$ algorithm for computing 2-approximate solutions [Charikar'00].

Previous Results on Maximum Density Subgraphs

Directed graphs:

- Maximum density subgraph can be found in polynomial time.
  - Linear programming based algorithm by rounding the LP solution [Charikar'00].
    - Requires computations of $|V|^2$ linear programs.
  - No combinatorial algorithm known.
- $O(|V|^3 + |V|^2|E|)$ algorithm for computing 2-approximate solutions [Charikar'00].

Previous Results on Maximum Density Subgraphs

Directed graphs:

- Maximum density subgraph can be found in polynomial time.
  - Linear programming based algorithm by rounding the LP solution [Charikar'00].
    - Requires computations of $|V|^2$ linear programs.
  - No combinatorial algorithm known.
- $O(|V|^3 + |V|^2|E|)$ algorithm for computing 2-approximate solutions [Charikar'00].

Previous Results on Maximum Density Subgraphs

Directed graphs:

- Maximum density subgraph can be found in polynomial time.
    - Linear programming based algorithm by rounding the LP solution [Charikar'00].
        - Requires computations of $|V|^2$ linear programs.
    - No combinatorial algorithm known.
- $O(|V|^3 + |V|^2|E|)$ algorithm for computing 2-approximate solutions [Charikar'00].

Densest k Subgraph Problem

$|V(H)| = k$

- NP hard.
- Best approximation algorithm known: $|V|^{\frac{1}{3}-\epsilon}$ [Feige, Kortsarz, Peleg'93].
- Best hardness result known: No PTAS exists [Khot'04].

Relaxations of Densest k Subgraph Problem [Andersen, Chelapilla'08]

- Densest at least k Subgraph Problem.
  - $|V(H)| \geq k$
- Densest at most k Subgraph Problem.
  - $|V(H)| \leq k$

Previous Results on Densest At Least k Subgraph Problem

- 3-approximation linear time greedy algorithm [Andersen & Chelapilla'08]
- Polynomial time 2 approximation [Andersen]
  - Requires $|V|^2$ parametric flow computations.
- It was not known whether the problem is NP hard.

Previous Results on Densest At Least k Subgraph Problem

- 3-approximation linear time greedy algorithm [Andersen & Chelapilla'08]
- Polynomial time 2 approximation [Andersen]
    - Requires $|V|^2$ parametric flow computations.
- It was not known whether the problem is NP hard.

Previous Results on Densest At Least k Subgraph Problem

- 3-approximation linear time greedy algorithm [Andersen & Chelapilla'08]
- Polynomial time 2 approximation [Andersen]
  - Requires $|V|^2$ parametric flow computations.
- It was not known whether the problem is NP hard.

Previous Results on Densest At most k Subgraph Problem

- NP hard.
- A γ approximation to this problem implies a γ^2 approximation algorithm to the densest k subgraph problem.

CONTRIBUTIONS

Maximum Density Subgraph Problem

- **First combinatorial algorithm for maximum density subgraph problem on directed graphs.**
- A 2-approximation $O(|V| + |E|)$ time algorithm for computing maximum density subgraphs on directed graphs.
  - Improves the previous running time of $O(|V|^3 + |V|^2|E|)$.

Maximum Density Subgraph Problem

- First combinatorial algorithm for maximum density subgraph problem on directed graphs.
- **A** 2**-approximation** $O(|V| + |E|)$ **time algorithm for computing maximum density subgraphs on directed graphs.**
    - Improves the previous running time of $O(|V|^3 + |V|^2|E|)$.

Densest At least k Subgraph Problem

Undirected Graphs:

- **We show the problem is NP-complete.**
- We give a combinatorial algorithm that requires only $max(1, k - d_G)$ parametric flow computations and achieves 2-approximation.
  - Previous 2 approximation algorithm required n^2 parametric flow computations.
- We give a LP rounding based algorithm that also achieves an approximation factor of 2 and requires to solve the LP only once.

Densest At least k Subgraph Problem

Undirected Graphs:

- We show the problem is NP-complete.
- **We give a combinatorial algorithm that requires only** $max(1, k - d_G)$ **parametric flow computations and achieves** 2**-approximation.**
  - Previous 2 approximation algorithm required n^2 parametric flow computations.
- We give a LP rounding based algorithm that also achieves an approximation factor of 2 and requires to solve the LP only once.

Densest At least k Subgraph Problem

Undirected Graphs:

- We show the problem is NP-complete.
- We give a combinatorial algorithm that requires only $max(1, k - d_G)$ parametric flow computations and achieves 2-approximation.
  - Previous 2 approximation algorithm required n^2 parametric flow computations.
- **We give a LP rounding based algorithm that also achieves an approximation factor of 2 and requires to solve the LP only once.**

Densest At least k Subgraph Problem

Directed Graphs:

- **We define the densest at least k_1, k_2 subgraph problem for directed graphs.**
  - $|S| \geq k_1, |T| \geq k_2$
- We give a combinatorial 2 approximation algorithm for it.

Densest At least k Subgraph Problem

Directed Graphs:

- We define the densest at least k_1, k_2 subgraph problem for directed graphs.
  - $|S| \geq k_1, |T| \geq k_2$
- **We give a combinatorial 2 approximation algorithm for it.**

Densest At most k Subgraph Problem

- **We show a γ approximation algorithm for densest at most k subgraph problem implies a 4γ approximation algorithm for the densest k subgraph problem.**
  - Previously only a quadratic dependency on the approximation factors between at most k and exact k densest subgraph problem was known.

Today's Talk

	Maximum Density Subgraph:No Size Constraint
Complexity	-
Undirected	-
Directed	Combinatorial solution, linear time 2 approx

	Densest At least k Subgraph Problem
Complexity	NP hard
Undirected	Fast combinatorial and LP based algorithm 2 approx
Directed	Combinatorial 2 approx

	Densest At most k Subgraph Problem
Complexity	Linear dependency with exact k
Undirected	-
Directed	-

Today's Talk

	Maximum Density Subgraph:No Size Constraint
Complexity	-
Undirected	-
Directed	Combinatorial solution. Linear time 2 approx
	Densest At least *k* Subgraph Problem
Complexity	NP hard
Undirected	Fast combinatorial and LP based algorithm 2 approx
Directed	Combinatorial 2 approx
	Densest At most *k* Subgraph Problem
Complexity	Linear dependency with exact *k*
Undirected	-
Directed	-

Combinatorial Algorithm for Maximum Density Subgraph in Directed Graphs

Combinatorial Algorithm for Maximum Density Subgraph in Directed Graphs

Main Idea

- Suppose the optimum subgraph is (S, T).
- Let $g = d(S, T)$ and let $a = \frac{|S|}{|T|}$.
- Guess the value of g and a. For every possible guess, construct a flow network from G.

Combinatorial Algorithm for Maximum Density Subgraph in Directed Graphs

Main Idea

- Suppose the optimum subgraph is (S, T).
- Let $g = d(S, T)$ and let $a = \frac{|S|}{|T|}$.
- Guess the value of g and a. For every possible guess, construct a flow network from G.

Combinatorial Algorithm for Maximum Density Subgraph in Directed Graphs

Main Idea

- Suppose the optimum subgraph is (S, T).
- Let $g = d(S, T)$ and let $a = \frac{|S|}{|T|}$.
- Guess the value of g and a. For every possible guess, construct a flow network from G.

Combinatorial Algorithm for Maximum Density Subgraph in Directed Graphs

Main Idea

- The network satisfies the property:
  - For the correct guess of g and a, the densest subgraph is easy to detect.
- We will try all values of a and for each choice of a, we will do a binary search on the value of g.

Combinatorial Algorithm for Maximum Density Subgraph in Directed Graphs

Main Idea

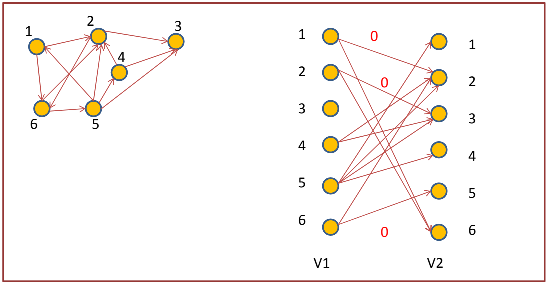
- The network satisfies the property:
  - For the correct guess of g and a, the densest subgraph is easy to detect.
- We will try all values of a and for each choice of a, we will do a binary search on the value of g.

Combinatorial Algorithm for Maximum Density Subgraph in Directed Graphs

Main Idea

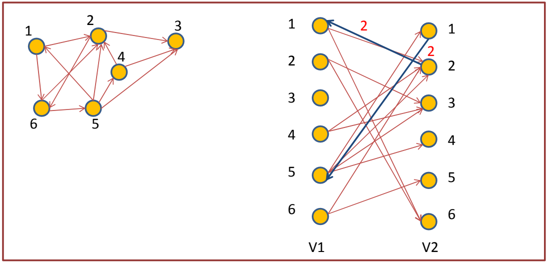
- The network satisfies the property:
  - For the correct guess of g and a, the densest subgraph is easy to detect.
- We will try all values of a and for each choice of a, we will do a binary search on the value of g.

Flow Network Construction for Maximum Density Subgraph in Directed Graphs



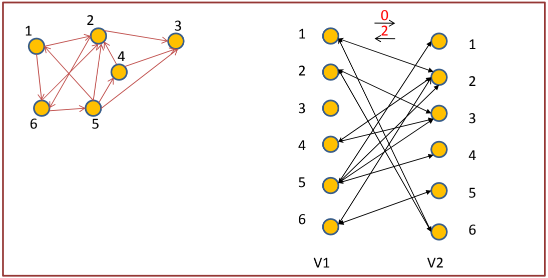
Replicate vertices on both sides and add forward edges of weight 0.

Flow Network Construction for Maximum Density Subgraph in Directed Graphs

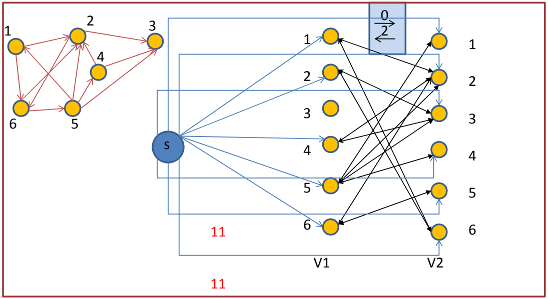


Add backward edges of weight 2.

Flow Network Construction for Maximum Density Subgraph in Directed Graphs

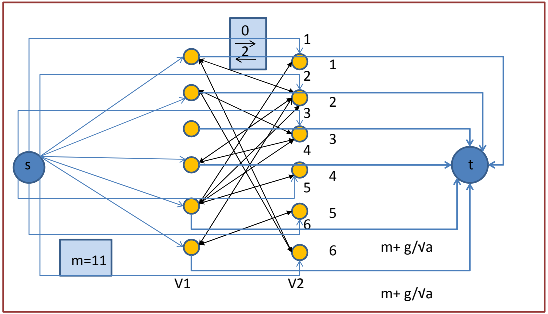


Flow Network Construction for Maximum Density Subgraph in Directed Graphs



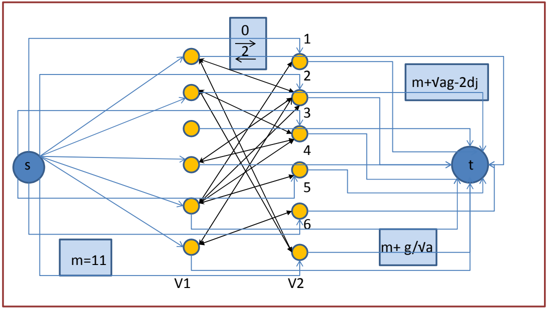
$\forall v \in V_1 \bigcup V_2$, add the edge (s, v) and set $w(s, v) = |E| = m$.

Flow Network Construction for Maximum Density Subgraph in Directed Graphs



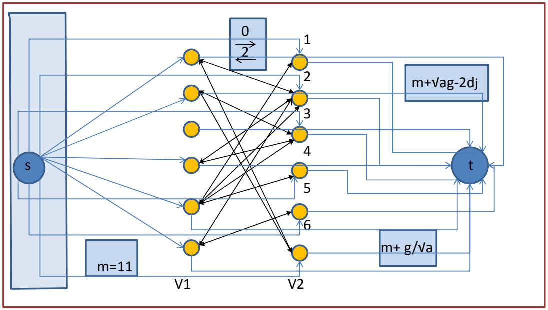
$\forall v \in V_1$, add the edge (v, t) with weight $w(v, t) = m + \frac{g}{\sqrt{a}}$.

Flow Network Construction for Maximum Density Subgraph in Directed Graphs



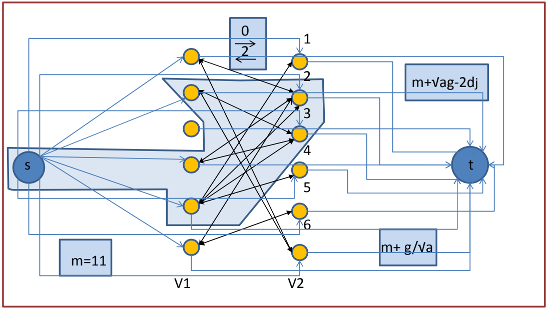
$\forall v \in V_2$, add the edge (v, t) with weight $w(v, t) = m + \sqrt{ag} - 2d_v$.

Flow Network Construction for Maximum Density Subgraph in Directed Graphs



Trivial cut has value $m(|V_1| + |V_2|)$.

Flow Network Construction for Maximum Density Subgraph in Directed Graphs



Cut-value=$m(|V_1| + |V_2|) + \frac{|S'|}{\sqrt{a}}\left(g - \frac{|E(S',T')|}{|S'|/\sqrt{a}}\right) +$
$|T'|\sqrt{a}\left(g - \frac{|E(S',T')|}{|T'|\sqrt{a}}\right)$

- Trivial cut=$m(|V_1| + |V_2|)$.
- Nontrivial cut=$m(|V_1| + |V_2|) + \frac{|S'|}{\sqrt{a}} \left(g - \frac{|E(S',T')|}{|S'|/\sqrt{a}}\right) + |T'|\sqrt{a} \left(g - \frac{|E(S',T')|}{|T'|\sqrt{a}}\right)$

- Trivial cut=$m(|V_1| + |V_2|)$.
- Nontrivial cut=$m(|V_1| + |V_2|) + \frac{|S'|}{\sqrt{a}} \left(g - \frac{|E(S',T')|}{|S'|/\sqrt{a}}\right) + |T'|\sqrt{a} \left(g - \frac{|E(S',T')|}{|T'|\sqrt{a}}\right)$

Case 1: $g < d(S,T)$,

- Argue that if the guessed a is correct, both $\left(g - \frac{|E(S,T)|}{|S|/\sqrt{a}}\right)$ and $\left(g - \frac{E(S,T)}{|T|\sqrt{a}}\right)$ are negative.
- Therefore mincut is formed by some nontrivial cut.

- Trivial cut=$m(|V_1| + |V_2|)$.
- Nontrivial cut=$m(|V_1| + |V_2|) + \frac{|S'|}{\sqrt{a}} \left(g - \frac{|E(S',T')|}{|S'|/\sqrt{a}}\right) + |T'|\sqrt{a} \left(g - \frac{|E(S',T')|}{|T'|\sqrt{a}}\right)$

Case 2: $g > d(S,T)$,

- Argue by contradiction that we always obtain a trivial cut.

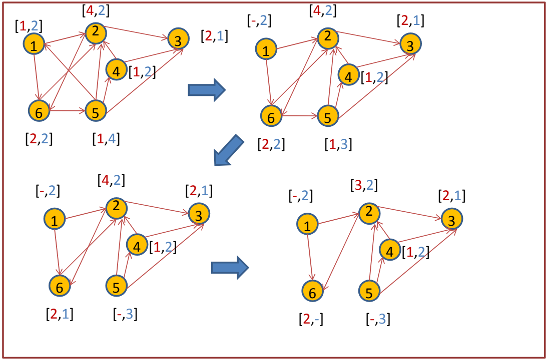
- Trivial cut=$m(|V_1| + |V_2|)$.
- Nontrivial cut=$m(|V_1| + |V_2|) + \frac{|S'|}{\sqrt{a}}\left(g - \frac{|E(S',T')|}{|S'|/\sqrt{a}}\right) + |T'|\sqrt{a}\left(g - \frac{|E(S',T')|}{|T'|\sqrt{a}}\right)$

Case 3: $g = d(S, T)$,

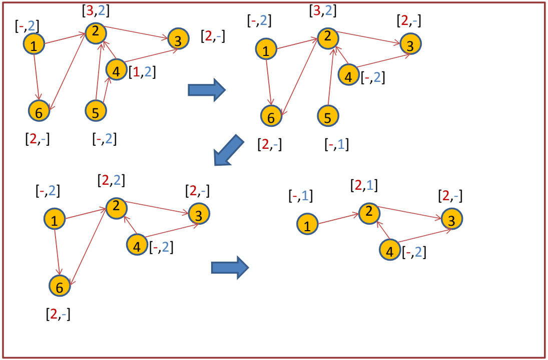
- If a is correct, argue that both the trivial cut and the cut $(\{s, S \subseteq V_1, T \subseteq V_2\}, \{t, (V_1 \setminus S) \subseteq V_1, (V_2 \setminus T) \subseteq V_2\}$ are min-cuts.
- If a is not correct, argue that the min cut is the trivial cut.

Linear Time 2 Approximation Algorithm for the Densest Subgraph in Directed Graphs

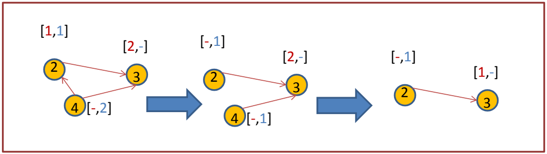
Linear Time 2 Approximation Algorithm for the Densest Subgraph in Directed Graphs



Linear Time 2 Approximation Algorithm for the Densest Subgraph in Directed Graphs



Linear Time 2 Approximation Algorithm for the Densest Subgraph in Directed Graphs



Algorithm

Algorithm 2.1: DENSEST-SUBGRAPH-DIRECTED($G = (V, E)$)

$n \leftarrow |V|, H_{2n} \leftarrow G, i \leftarrow 2n$
while $H_i \neq \emptyset$
do
- Let v be a vertex in H_i of minimum degree
- **if** category(v) $= IN$
  - **then** Delete all the incoming edges incident on v
  - **else** Delete all the outgoing edges incident on v
- **if** v has no edges incident on it **then** Delete v
- Call the new graph $H_{i-1}, i \leftarrow i - 1$

return (H_j which has the maximum density among H_i's)

Linear Time 2 Approximation Algorithm for the Densest Subgraph in Directed Graphs

Proof Sketch

- Detect two values λ_i and λ_o, such that in the optimum solution any vertex in S cannot have outdegree $< \lambda_o$ and any vertex in T cannot have indegree $< \lambda_i$.
    - Argue that $\lambda_i = |E(S^*, T^*)| \left(1 - \sqrt{1 - \frac{1}{|T^*|}}\right)$ and $\lambda_o = |E(S^*, T^*)| \left(1 - \sqrt{1 - \frac{1}{|S^*|}}\right)$ are appropriate choices.
- Consider the iteration of the algorithm when all the vertices have out-degree $\geq \lambda_o$ and indegree $\geq \lambda_i$ and argue that for the above choices of λ_i and λ_o, density is at least $\frac{1}{2}$ of the optimum.

Linear Time 2 Approximation Algorithm for the Densest Subgraph in Directed Graphs

Proof Sketch

- Detect two values λ_i and λ_o, such that in the optimum solution any vertex in S cannot have outdegree $< \lambda_o$ and any vertex in T cannot have indegree $< \lambda_i$.
  - Argue that $\lambda_i = |E(S^*, T^*)| \left(1 - \sqrt{1 - \frac{1}{|T^*|}}\right)$ and $\lambda_o = |E(S^*, T^*)| \left(1 - \sqrt{1 - \frac{1}{|S^*|}}\right)$ are appropriate choices.
- Consider the iteration of the algorithm when all the vertices have out-degree $\geq \lambda_o$ and indegree $\geq \lambda_i$ and argue that for the above choices of λ_i and λ_o, density is at least $\frac{1}{2}$ of the optimum.

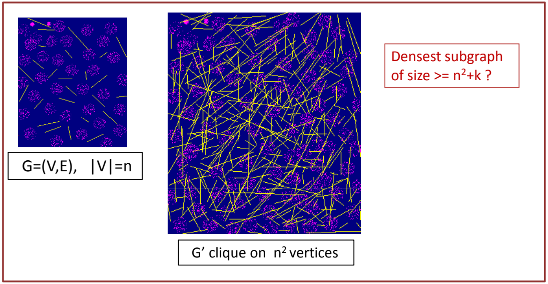
Linear Time 2 Approximation Algorithm for the Densest Subgraph in Directed Graphs

Proof Sketch

- Detect two values λ_i and λ_o, such that in the optimum solution any vertex in S cannot have outdegree $< \lambda_o$ and any vertex in T cannot have indegree $< \lambda_i$.
  - Argue that $\lambda_i = |E(S^*, T^*)| \left(1 - \sqrt{1 - \frac{1}{|T^*|}}\right)$ and $\lambda_o = |E(S^*, T^*)| \left(1 - \sqrt{1 - \frac{1}{|S^*|}}\right)$ are appropriate choices.
- Consider the iteration of the algorithm when all the vertices have out-degree $\geq \lambda_o$ and indegree $\geq \lambda_i$ and argue that for the above choices of λ_i and λ_o, density is at least $\frac{1}{2}$ of the optimum.

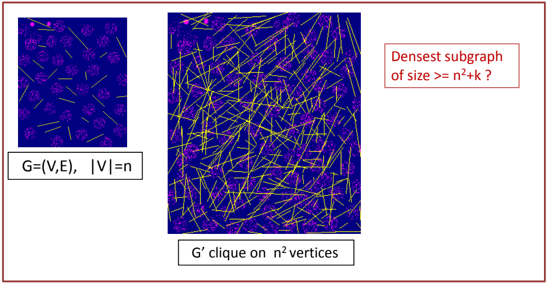
Densest At least k Subgraph Problem is NP Hard

Densest At least k Subgraph Problem is NP Hard



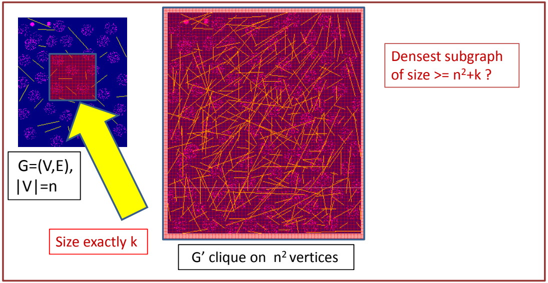
Want to know, whether there exists a subgraph of size k in $G = (V, E), |V| = n$ of density $\geq \lambda$.

Densest At least k Subgraph Problem is NP Hard



Add a clique G' of size n^2 and ask for the optimum densest at least $n^2 + k$ subgraph in $G \bigcup G'$.

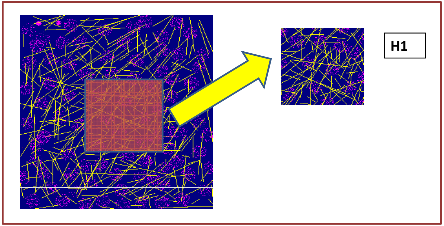
Densest At least k Subgraph Problem is NP Hard



Argue that the optimum solution consists of G' and the densest k subgraph of G.

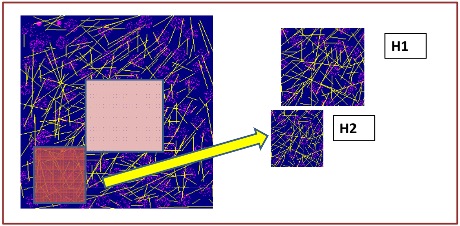
2-approximation Algorithm for Densest At least k subgraph

2-approximation Algorithm for Densest At least k subgraph



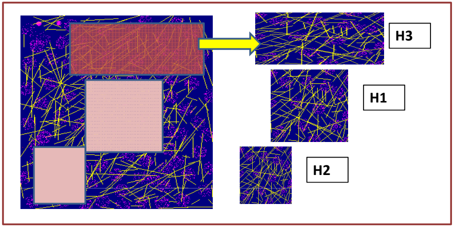
Obtain the maximum density subgraph H_1 of G. If $|V(H_1)| \geq k$ STOP.

2-approximation Algorithm for Densest At least k subgraph



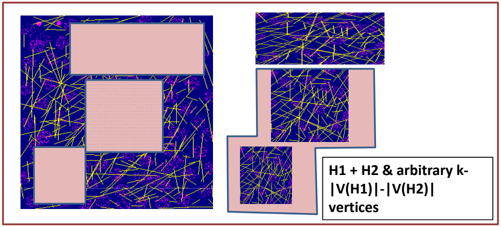
Otherwise, remove H_1. If $v \notin V(H_1)$ has x edges to $V(H_1)$, add a self-loop of weight x to it. Compute the densest subgraph H_2 in $G - H_1$.

2-approximation Algorithm for Densest At least k subgraph



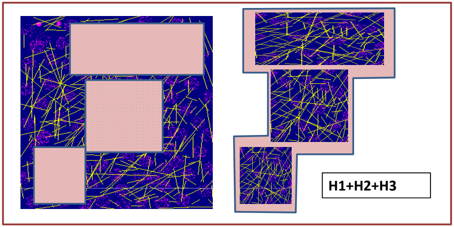
If $|V(H_1)| + |V(H_2)| \geq k$, STOP. Else remove H_2, adjust edge weights and compute H_3.

2-approximation Algorithm for Densest At least k subgraph



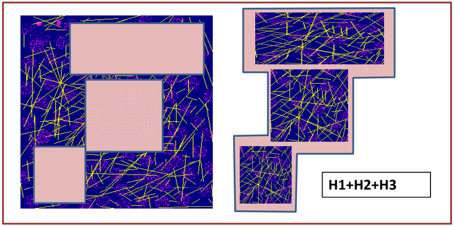
Suppose $|V(H_1)| + |V(H_2)| + |V(H_3)| \geq k$. Consider $H_1 \bigcup H_2$ and some arbitrary vertices to make up for size k.

2-approximation Algorithm for Densest At least k subgraph



Consider $H_1 \bigcup H_2 \bigcup H_3$.

2-approximation Algorithm for Densest At least k subgraph



Return the one which has higher density.

2-approximation Algorithm for Densest At least k subgraph

Proof Sketch.

- If H_1 and H_2 already covers half the edges of the optimum, then we get a 2 approximation from the first option.
- Otherwise, half the edges of the optimum still remains and therefore density of H_3 is at least half the density of the optimum. So the second option gives a 2 approximation in this case.

2-approximation Algorithm for Densest At least k subgraph

Proof Sketch.

- If H_1 and H_2 already covers half the edges of the optimum, then we get a 2 approximation from the first option.
- Otherwise, half the edges of the optimum still remains and therefore density of H_3 is at least half the density of the optimum. So the second option gives a 2 approximation in this case.

Open Problems

- Obtain linear time algorithm for maximum density subgraph problem for both directed and undirected cases, with approximation factor better than 2.
- Improve the running time of the combinatorial algorithm for computing maximum density subgraph in directed graphs.
  - How can we get rid off *trying all possible values of a* ?
- Improve the approximation factor of 2 for densest at least k subgraph problem for both undirected and directed graphs.

Open Problems

- Obtain linear time algorithm for maximum density subgraph problem for both directed and undirected cases, with approximation factor better than 2.
- Improve the running time of the combinatorial algorithm for computing maximum density subgraph in directed graphs.
  - How can we get rid off *trying all possible values of a* ?
- Improve the approximation factor of 2 for densest at least k subgraph problem for both undirected and directed graphs.

Open Problems

- Obtain linear time algorithm for maximum density subgraph problem for both directed and undirected cases, with approximation factor better than 2.
- Improve the running time of the combinatorial algorithm for computing maximum density subgraph in directed graphs.
  - How can we get rid off *trying all possible values of a* ?
- Improve the approximation factor of 2 for densest at least k subgraph problem for both undirected and directed graphs.

THANK YOU

ANY DENSE QUESTIONS !!