

Dense sub-graphs

Introduction to Network Science

Carlos Castillo

Topic 13

Sources

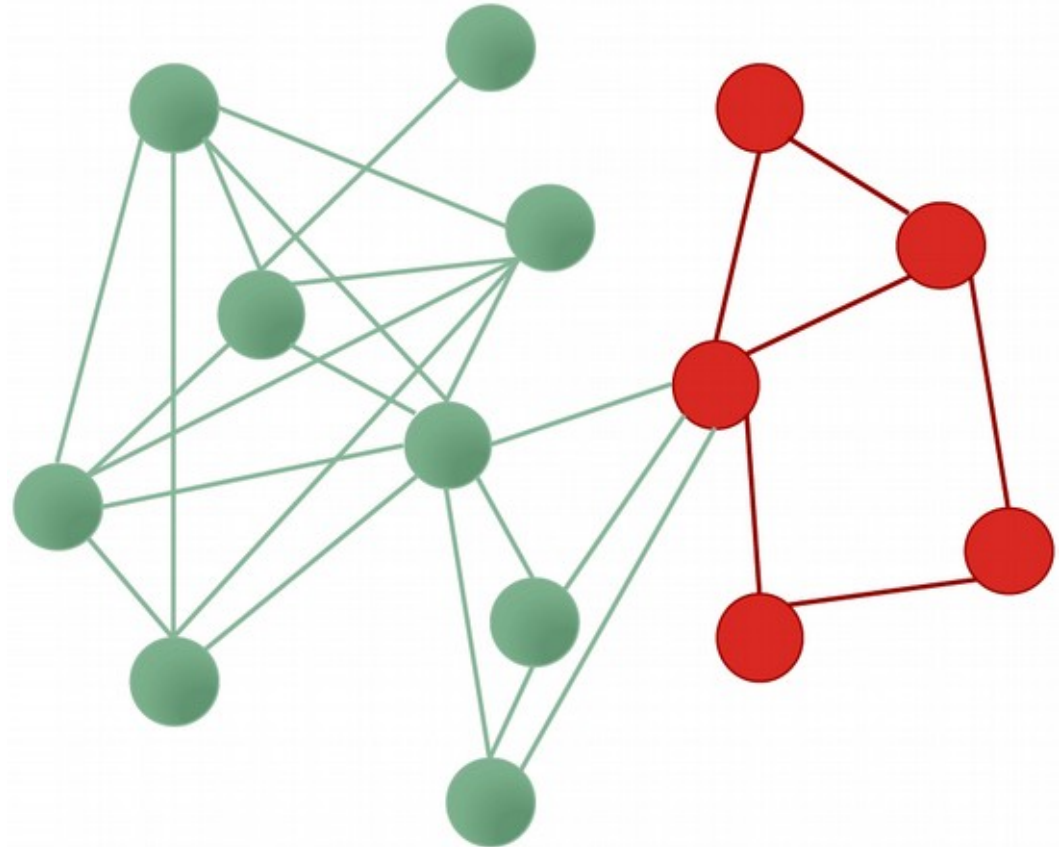
- Barabási 2016 Chapter 9
- [Networks, Crowds, and Markets](#) Ch 3
- C. Castillo (2017) [Dense Sub-Graphs](#)
- Tutorial by A. Beutel, L. Akoglu, C. Faloutsos [[Link](#)]
- Frieze, Gionis, Tsourakakis: “Algorithmic techniques for modeling and mining large graphs (AMAZING)” [[Tutorial](#)]
- A survey of algorithms for dense sub-graph discovery [[link](#)]

Communities

2 communities	[previous topic]
1 community	[this topic]
3+ communities	[next topic/course]

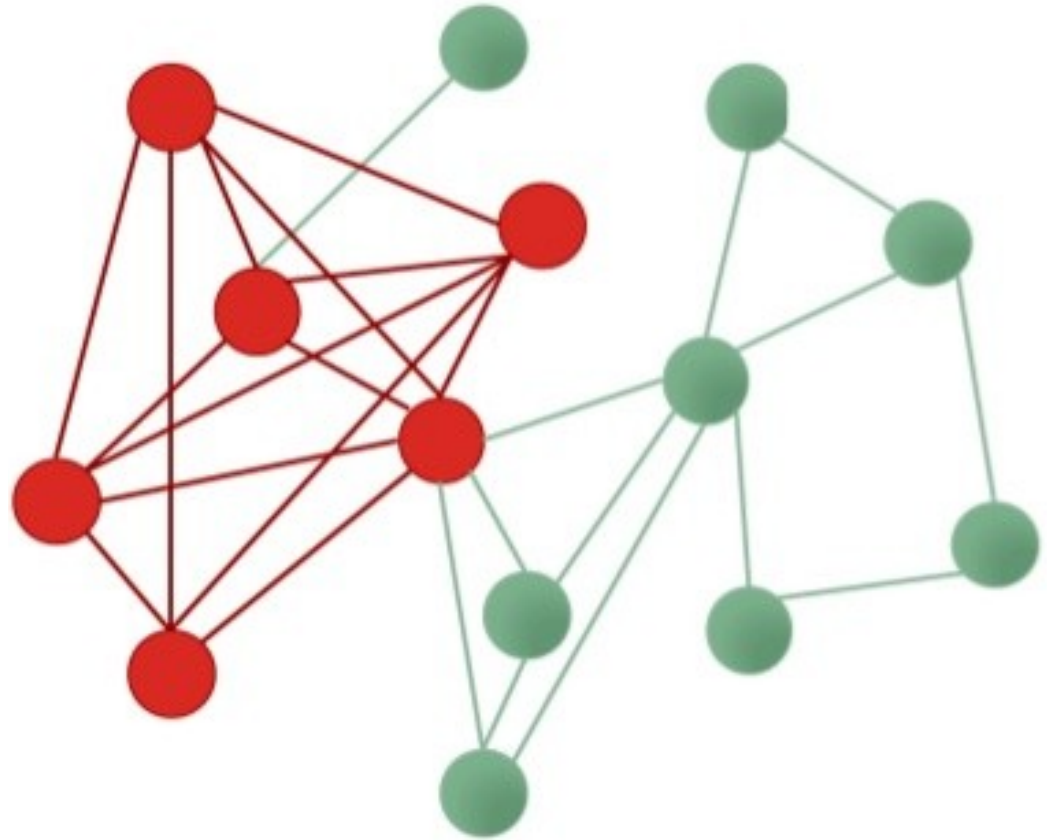
What is a sub-graph?

Subset of
nodes, and
edges among
those nodes



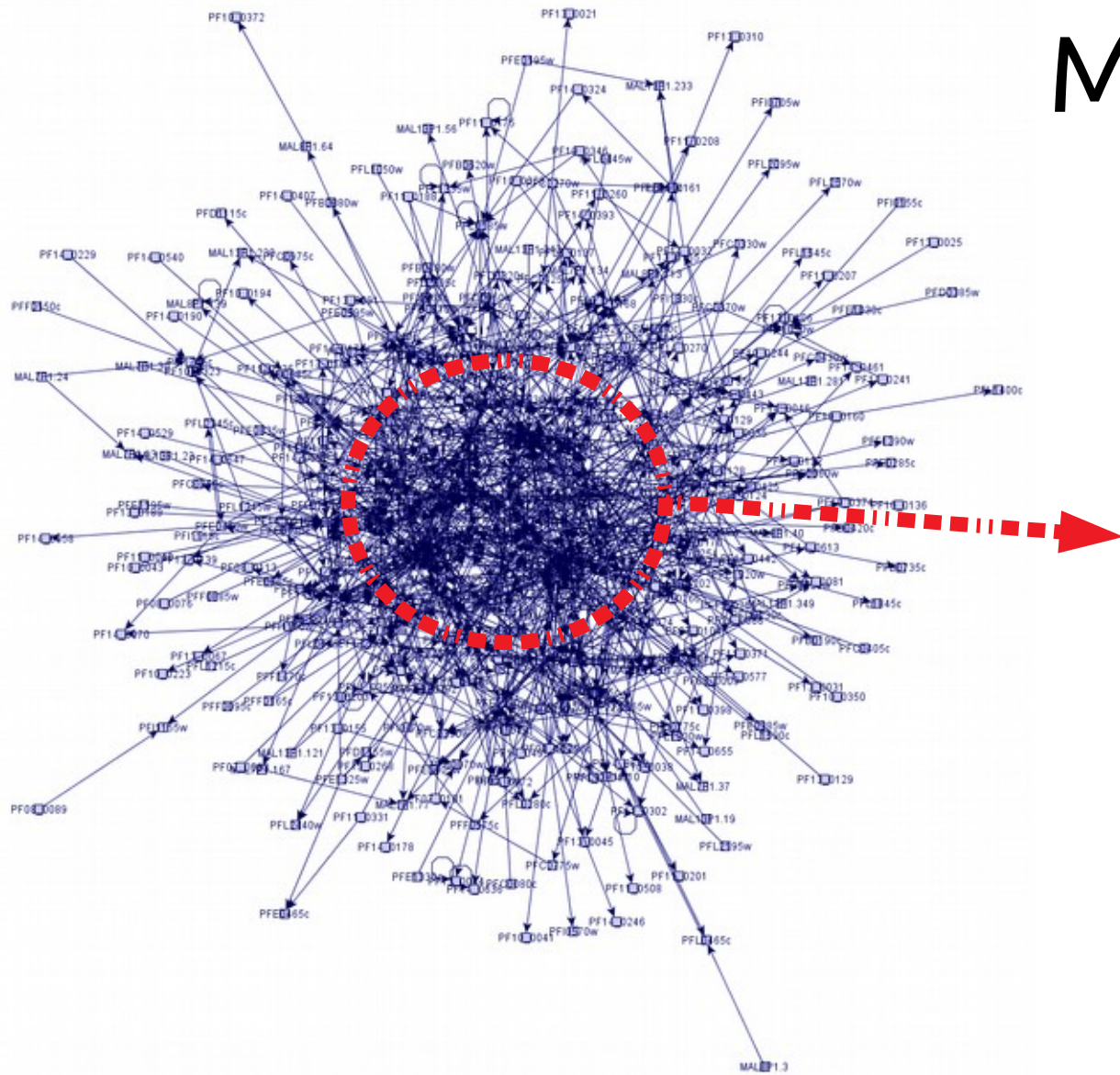
Densest sub-graph

Sub-graph
having the
maximum
density

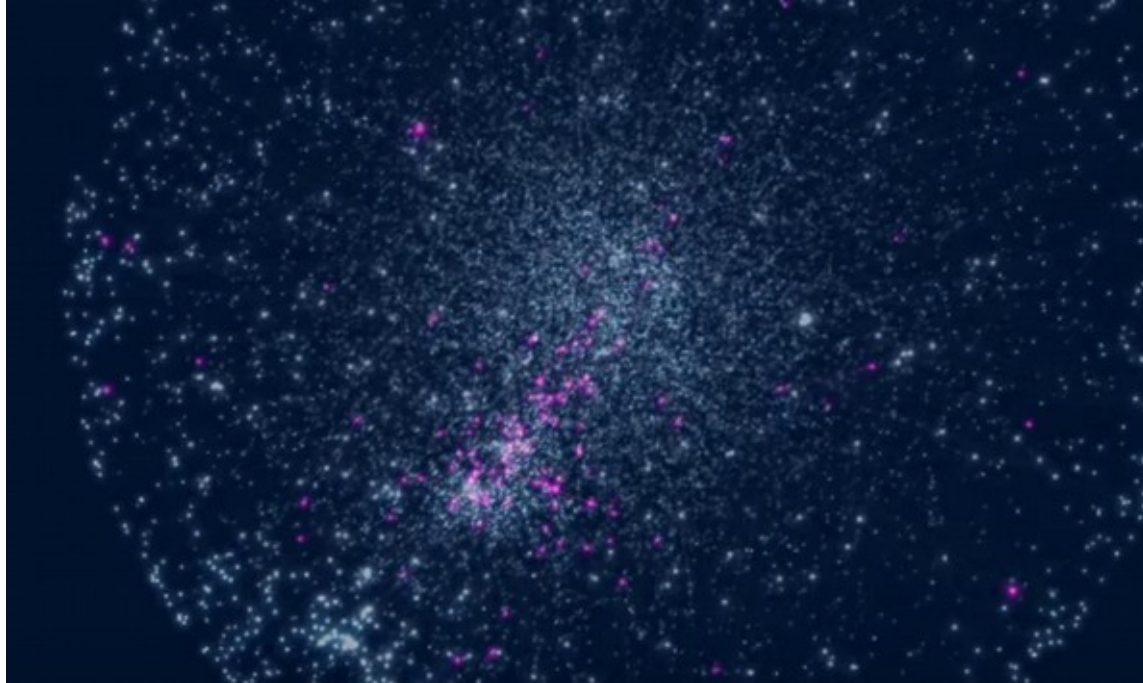


Many graphs look like “hairballs”

Sometimes, at the center these graphs may have an interesting dense sub-graph



Asthma-related genes



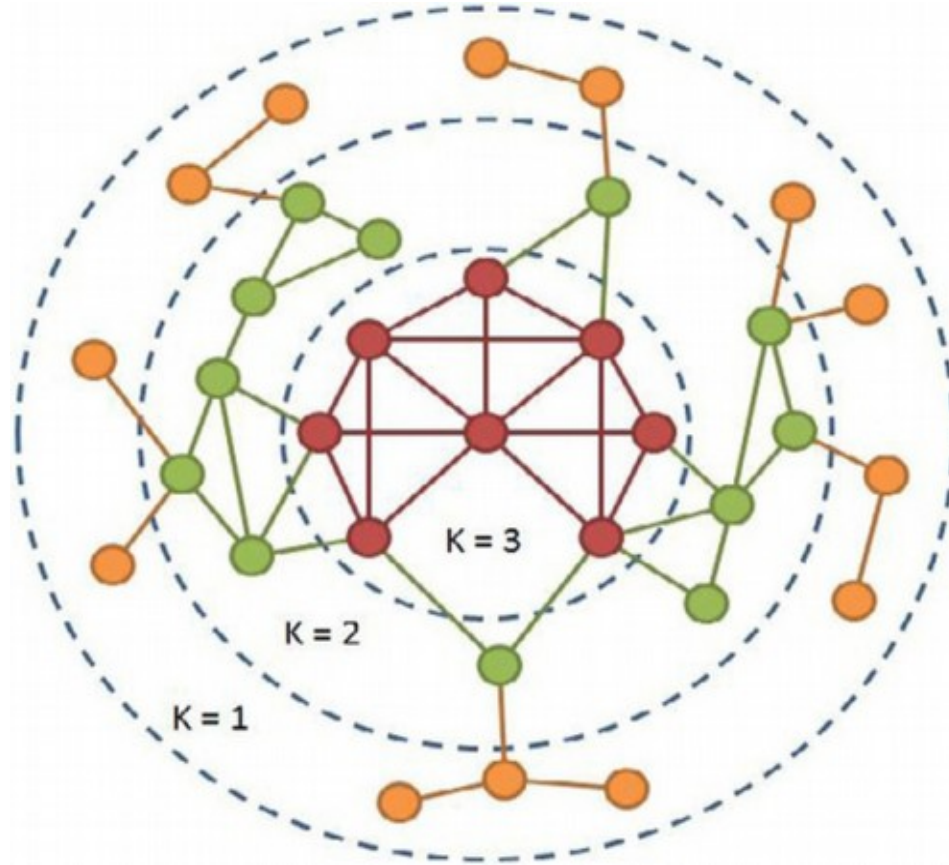
https://www.youtube.com/watch?v=VU_7FHAKMgA

k-core decomposition

k-core decomposition

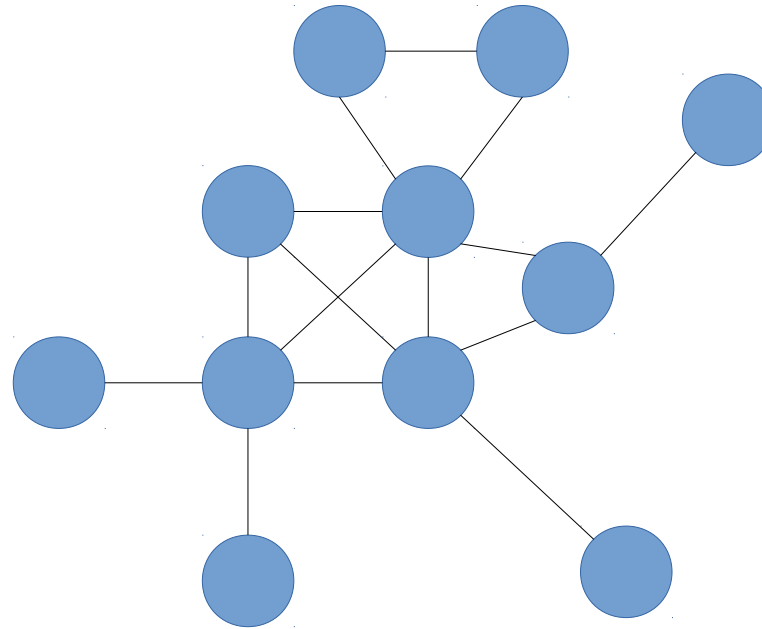
- Remove all nodes having degree 1
 - Those are in the 1-core
- Remove all nodes having degree 2 *in the remaining graph*
 - Those nodes are in the 2-core
- Remove all nodes having degree 3 *in the remaining graph*
 - Those nodes are in the 3-core
- Etc.

Example



Try it!

How many nodes are there in each core of this graph?



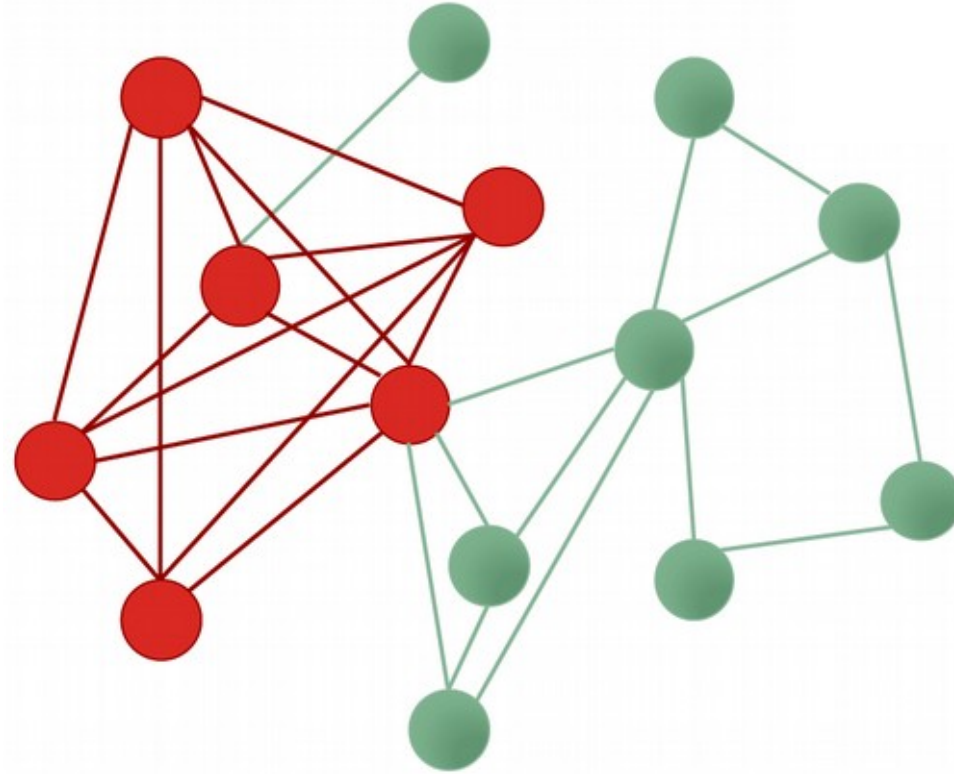
<http://www.cpt.univ-mrs.fr/~barrat/NHM.pdf>

Density-based methods

Density measures

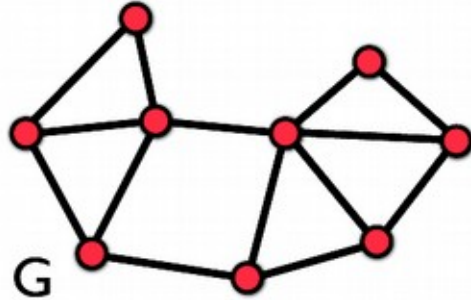
- Density = Average degree = $2|E|/|V|$
 - Sometimes just $|E|/|V|$
- Edge ratio =
$$\frac{2|E|}{|V|(|V| - 1)}$$
- What is $|V|(|V| - 1)/2$?

Densest sub-graph



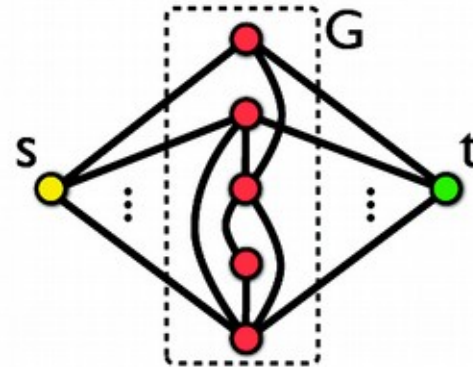
Goldberg's algorithm (1)

- consider first degree density d



- is there a subgraph S with $d(S) \geq c$?
- transform to a min-cut instance

- on the transformed instance:
- is there a cut smaller than a certain value?



Goldberg's algorithm (2)

is there S with $d(S) \geq c$?

$$\frac{2|E(S, S)|}{|S|} \geq c$$

$$2|E(S, S)| \geq c|S|$$

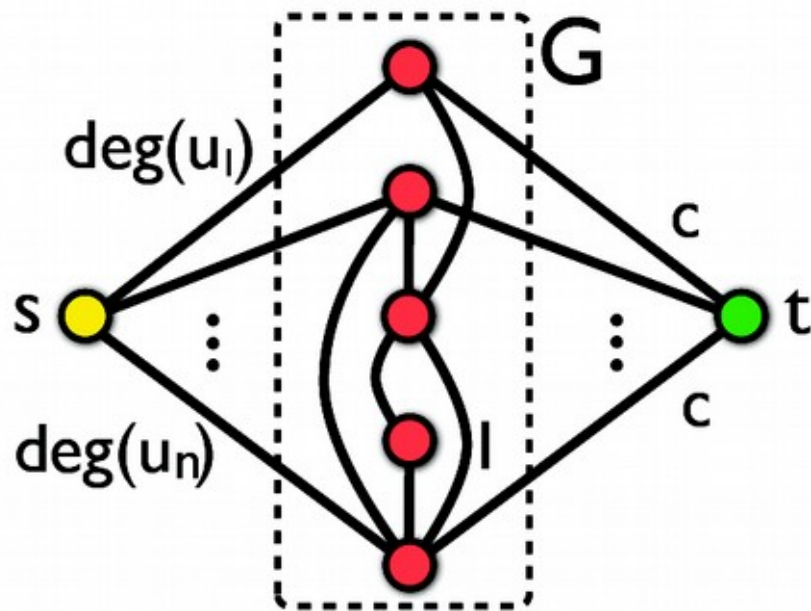
$$\sum_{u \in S} \deg(u) - |E(S, \bar{S})| \geq c|S|$$

$$\sum_{u \in S} \deg(u) + \sum_{u \in \bar{S}} \deg(u) - \sum_{u \in \bar{S}} \deg(u) - |E(S, \bar{S})| \geq c|S|$$

$$\sum_{u \in \bar{S}} \deg(u) + |E(S, \bar{S})| + c|S| \leq 2|E|$$

Goldberg's algorithm (3)

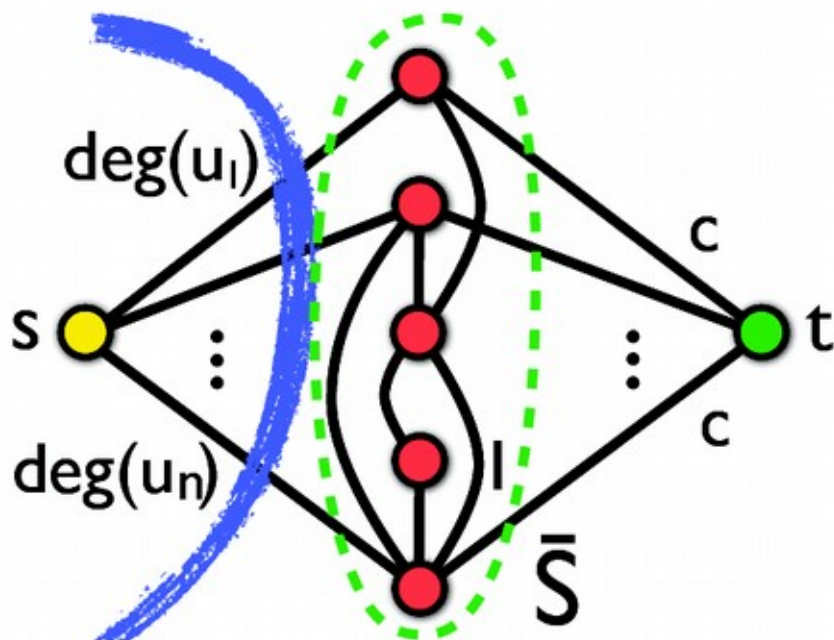
- transformation to min-cut instance



- is there S s.t. $\sum_{u \in \bar{S}} \deg(u) + |e(S, \bar{S})| + c|S| \leq 2|E|$?

Goldberg's algorithm (4)

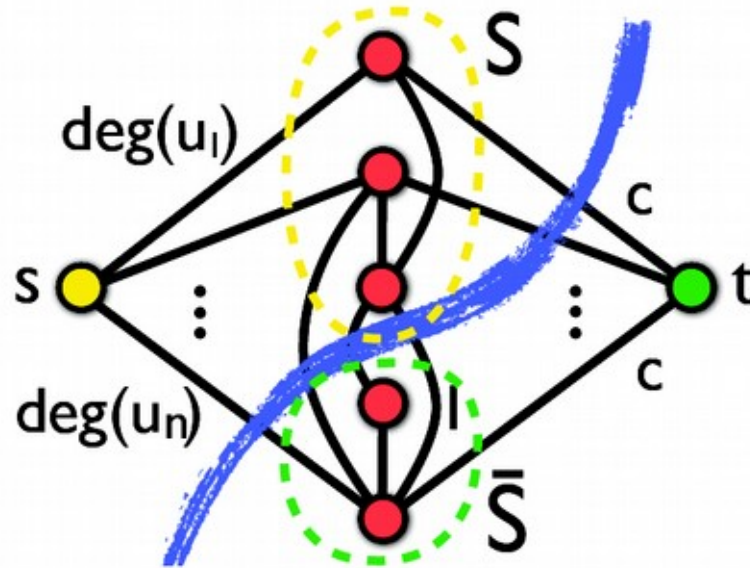
- transform to a min-cut instance



- is there S s.t. $\sum_{u \in \bar{S}} \deg(u) + |e(S, \bar{S})| + c|S| \leq 2|E|$?
- a cut of value $2|E|$ always exists, for $S = \emptyset$

Goldberg's algorithm (5)

- transform to a **min-cut** instance



- is there S s.t. $\sum_{u \in \bar{S}} \deg(u) + |e(S, \bar{S})| + c|S| \leq 2|E|$?
- $S \neq \emptyset$ gives cut of value $\sum_{u \in \bar{S}} \deg(u) + |e(S, \bar{S})| + c|S|$

If this exists for non-empty S , then S is a sub-graph of density c

Goldberg's algorithm (6)

- to find the densest subgraph perform binary search on c
 - logarithmic number of min-cut calls
 - each min-cut call requires $O(|V||E|)$ time
- problem can also be solved with one min-cut call using the **parametric max-flow algorithm**

A faster algorithm

- Charikar, M. (2000). Greedy approximation algorithms for finding dense components in a graph. In APPROX.
- Approximate algorithm (by a factor of 2)

Greedy algorithm

input: undirected graph $G = (V, E)$

output: S , a dense subgraph of G

- 1 set $G_n \leftarrow G$
- 2 for $k \leftarrow n$ downto 1
 - 2.1 let v be the smallest degree vertex in G_k
 - 2.2 $G_{k-1} \leftarrow G_k \setminus \{v\}$
- 3 output the densest subgraph among G_n, G_{n-1}, \dots, G_1

Compute density as $|V|/|E|$

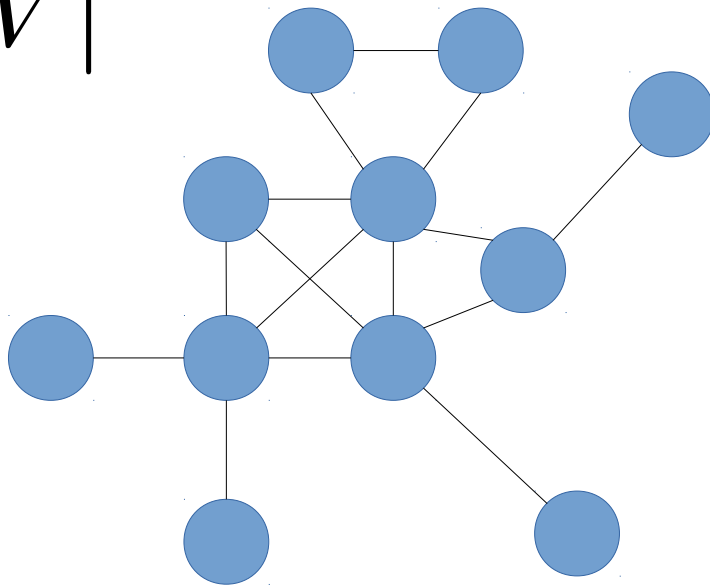
Try it!

$$\text{Density} = \frac{|E|}{|V|}$$

input: undirected graph $G = (V, E)$

output: S , a dense subgraph of G

- 1 set $G_n \leftarrow G$
- 2 for $k \leftarrow n$ downto 1
 - 2.1 let v be the smallest degree vertex in G_k
 - 2.2 $G_{k-1} \leftarrow G_k \setminus \{v\}$
- 3 output the densest subgraph among G_n, G_{n-1}, \dots, G_1



Advanced materials
(not included in the exam)

Approximation guarantee

- S^* = optimal sub-graph (highest density)
- $\text{density}(S^*) = \lambda = |e(S^*)| / |S^*|$
- For all v in S^* , $\deg(v) \geq \lambda$, because

$$\frac{|e(S^*)|}{|S^*|} \geq \frac{|e(S^* \setminus v)|}{|S^* \setminus v|} = \frac{|e(S^*)| - \deg_{S^*}(v)}{|S^*| - 1}$$



Because of optimality of S^*

Approximation guarantee (cont)

$$\frac{|e(S^*)|}{|S^*|} \geq \frac{|e(S^* \setminus v)|}{|S^* \setminus v|} = \frac{|e(S^*)| - \deg_{S^*}(v)}{|S^*| - 1}$$

Hence,

$$\deg_{S^*}(v) \geq \frac{|e(S^*)|}{|S^*|} = \text{density}(S^*) = \lambda$$

Approximation guarantee (cont.)

- Now, let's consider when greedy removes the **first** vertex of the optimal solution $v \in S^*$
- At that point, all the vertices of the remaining subgraph (S) have degree $\geq \lambda$, because v has degree $\geq \lambda$
- Hence, this subgraph has more than $\frac{\lambda|S|}{2}$ edges, and density more than $\frac{\frac{\lambda|S|}{2}}{|S|} = \frac{\lambda}{2}$

Hence this is a 2-approximate algorithm