Dense sub-graphs

Introduction to Network Science
Carlos Castillo
Topic 22

Universitat Pompeu Fabra Barcelona

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]

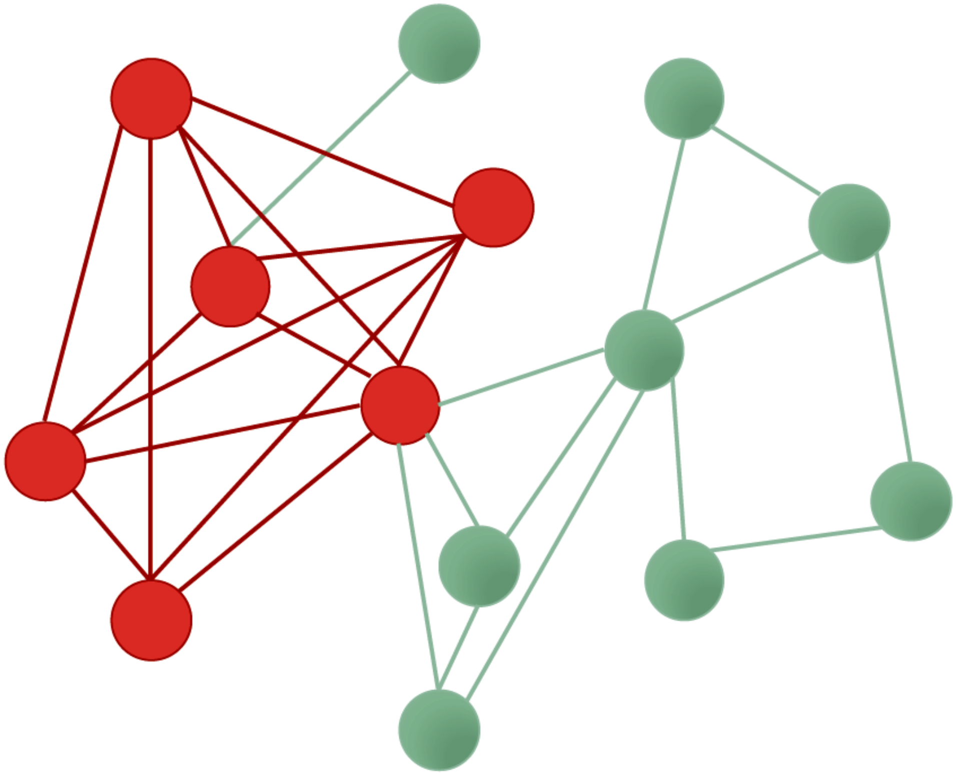
Density-based methods

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

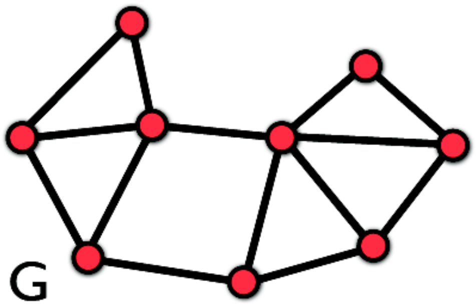


Goldberg’s algorithm

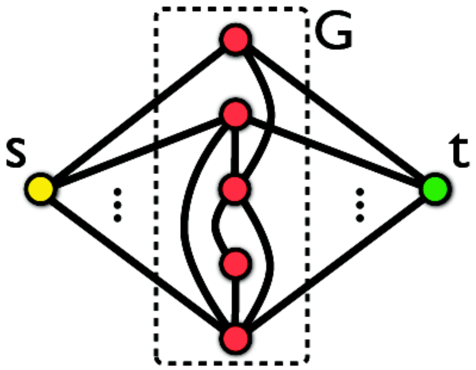
(exact and deterministic)

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?

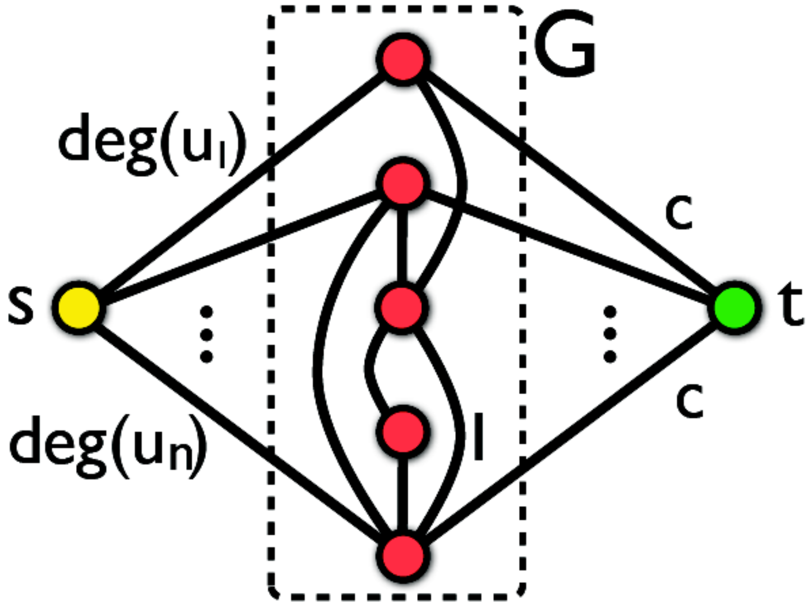
Goldberg's algorithm (2)

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

$$\begin{aligned}
\frac{2\,|E(S,S)|}{|S|} &\geq c \\
2\,|E(S,S)| &\geq c|S| \\
\sum_{u \in S} \deg(u) - |E(S,\bar{S})| &\geq c|S| \\
\sum_{u \in S} \deg(u) + \sum_{u \in \bar{S}} \deg(u) - \sum_{u \in \bar{S}} \deg(u) - |E(S,\bar{S})| &\geq c|S| \\
\sum_{u \in \bar{S}} \deg(u) + |E(S,\bar{S})| + c|S| &\leq 2\,|E|
\end{aligned}$$

Goldberg's algorithm (3)

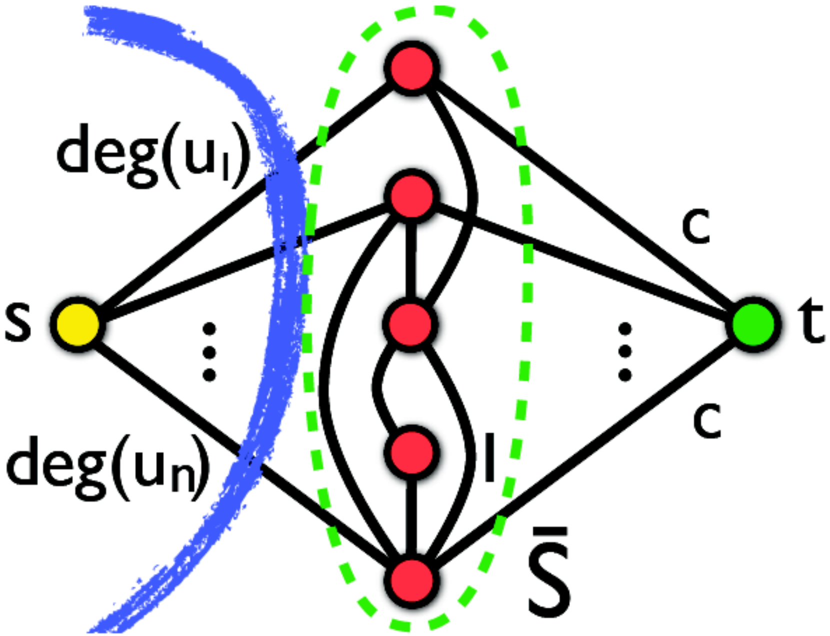
- transformation to min-cut instance



- 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)

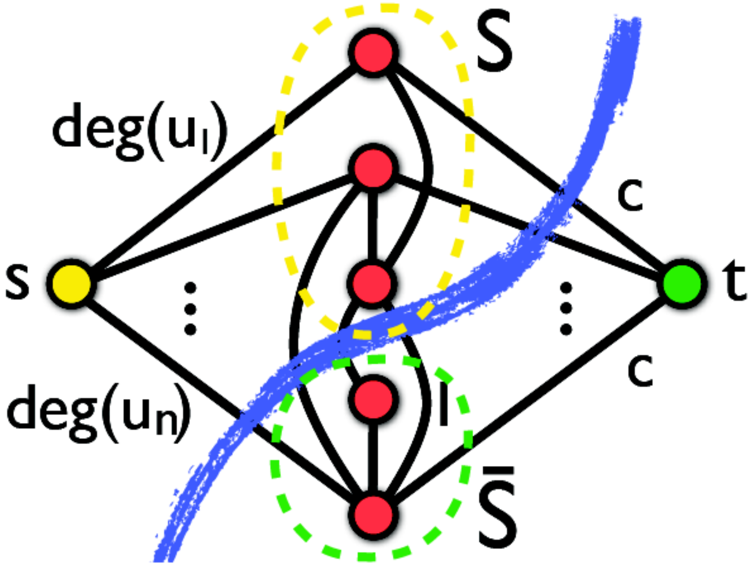
- transform to a min-cut instance



- 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)

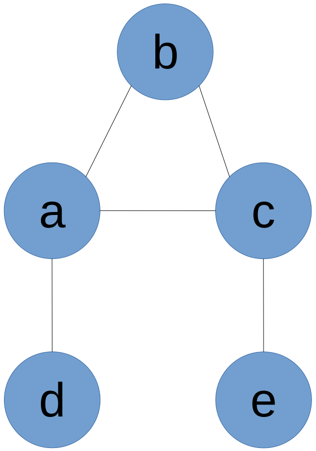
- transform to a min-cut instance



- 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

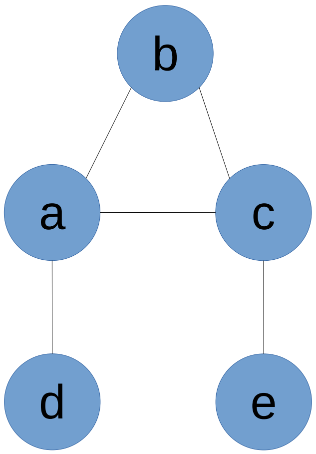
Example

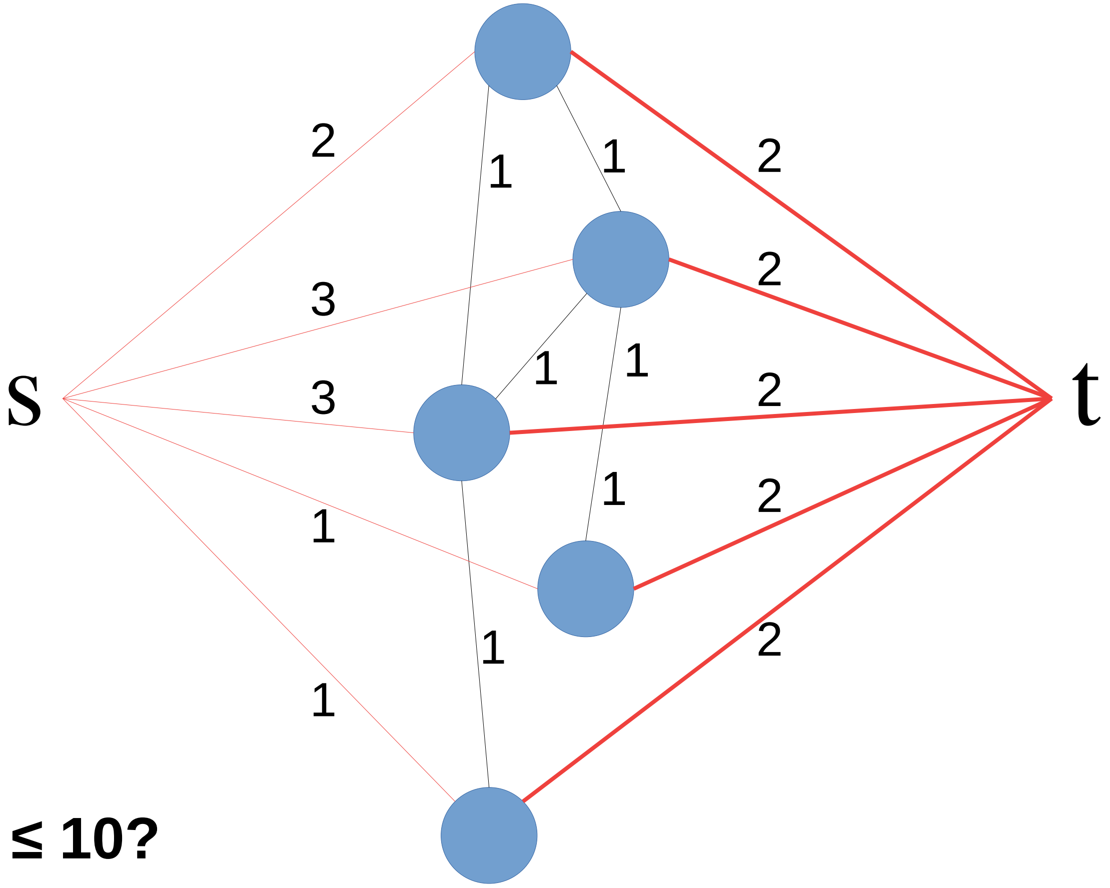


Is there S with $d(S) \geq 2$?

$d(S) = 2\,|E(S,S)|\,/\,|S|$

Example (cont.)

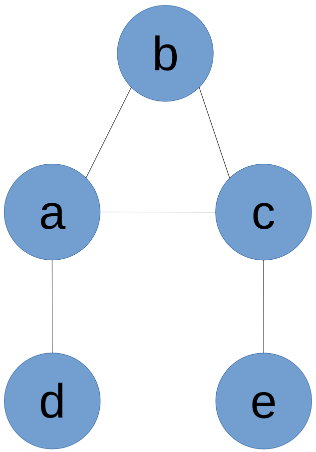


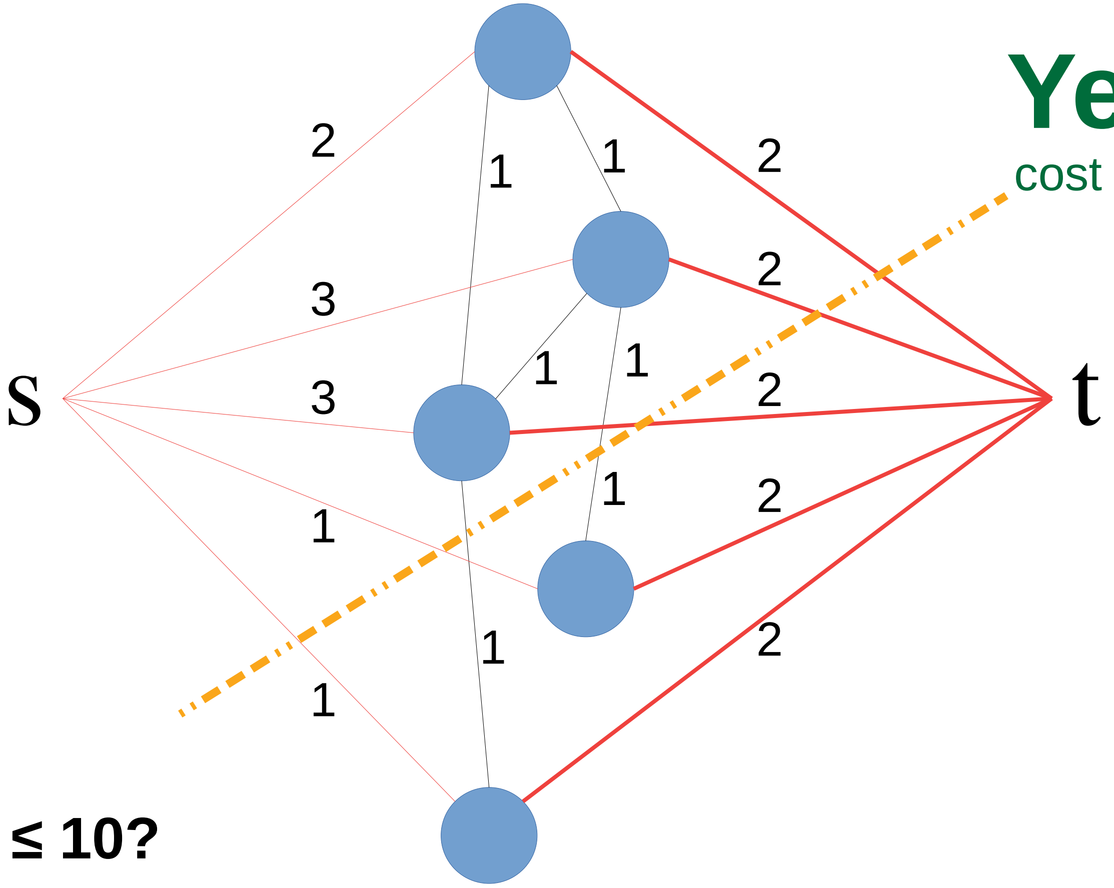


Is there S with d(S) ≥ 2?
d(S) = 2 |E(S,S)| / |S|

Is there an s-t cut with cost ≤ 10?
(2|E| = 10)

Example (cont.)





Yes!

cost = 10

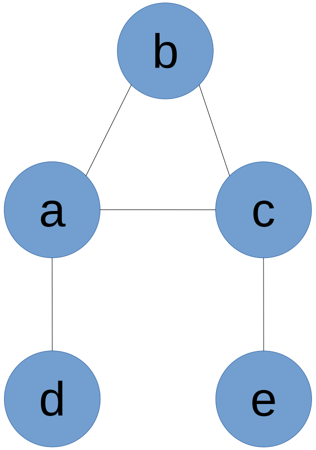
Is there S with d(S) ≥ 2?

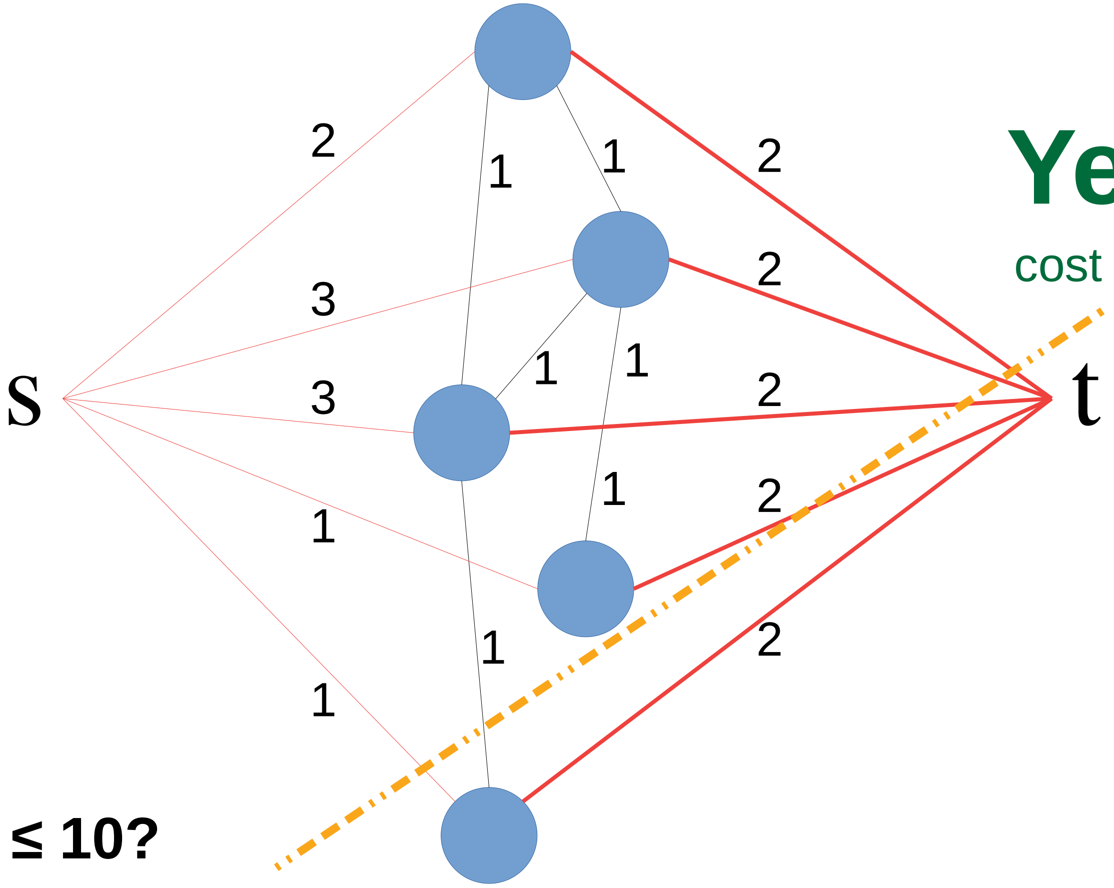
d(S) = 2 |E(S,S)| / |S|

Is there an s-t cut with cost ≤ 10?

(2|E| = 10)

Example (cont.)





Yes!

cost = 10

Is there S with d(S) ≥ 2?
d(S) = 2 |E(S,S)| / |S|

Is there an s-t cut with cost ≤ 10?
(2|E| = 10)

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

Charikar's algorithm
(approximate and randomized)

Charikar’s algorithm

- *Charikar, M. (2000). Greedy approximation algorithms for finding dense components in a graph. In APPROX.*
- **Approximate algorithm** (by a factor of 2)
  - If the optimal density is λ, in the worst case (if you’re very unlucky!) you will get density $\lambda/2$

Greedily remove nodes (break ties randomly)

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 $G_n, G_{n-1}, \ldots, G_1$

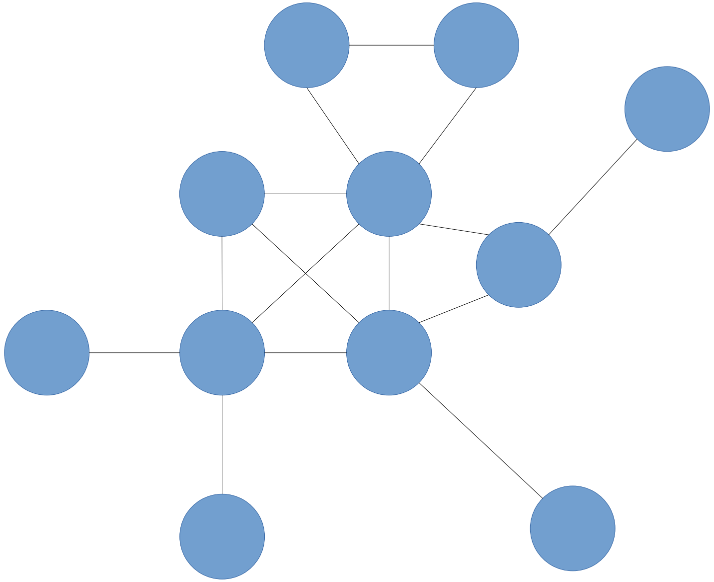
Compute density as |E|/|V|

Exercise

Density = $\dfrac{|E|}{|V|}$

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 $G_n, G_{n-1}, \ldots, G_1$



Draw in Nearpod Draw-it
https://nearpod.com/student/
Code to be given during class

Advanced materials
(not included in the exam)

Approximation guarantee

- S* = optimal sub-graph (highest density)
- density(S^*) = λ = $|e(S^*)|$ / $|S^*|$
- For all v in S*, deg(v) >= λ, because

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

Because of optimality of S*

https://people.cs.umass.edu/~barna/paper/dense-subgraph-full.pdf

Approximation guarantee (cont)

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

Hence,

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

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 >= λ, because v has degree >= λ
- 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

Summary

Things to remember

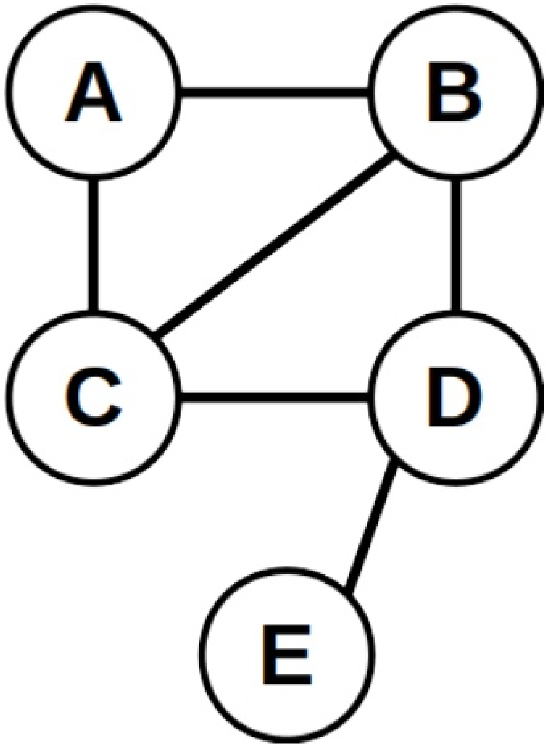
- Goldberg’s algorithm
  - How to write the construction for a given graph
- Charikar’s algorithm
  - How to run it on a given graph
- Practice executing these algorithms in small graphs
- If useful for you, write code for these algorithms

Practice on your own

Consider the graph on the right, which contains a subgraph with density $d(S) = 2|E(S, S)|/|S|$ equal to 5/2.

Draw the graph of **Goldberg's construction**, and in that graph, draw the s – t cut that crosses some of the original edges and proves that a subgraph of density 5/2 exists.

Indicate clearly (1) the cost of each edge in the construction, (2) the desired target cost as a function of $|E|$, (3) the cost of the cut you found, and (4) the sub-graph the method finds.



Practice on your own (cont.)

- Consider the graph on the right.
- Run **Charikar's** randomized algorithm for densest subgraph, indicating all intermediate graphs and their density, and marking clearly the graph with the largest density.
- For density use |E|/|V| where |E| is the number of edges in the subgraph and |V| the number of nodes in the subgraph.