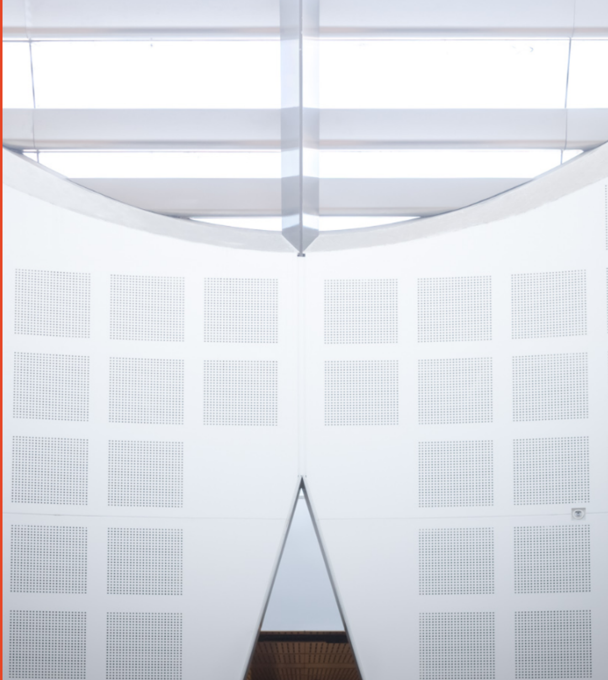
Deconstruct Densest Subgraphs

Lijun Chang

Joint work with Miao Qiao (The University of Aukland)

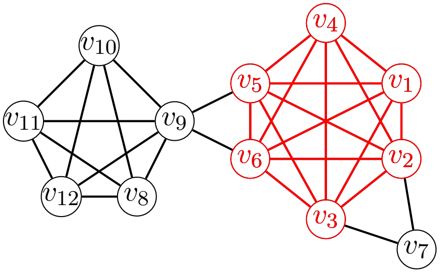
April 23, 2020

THE UNIVERSITY OF SYDNEY



Densest Subgraph

- Given an unweighted and undirected graph $G = (V, E)$, the densest subgraph problem aims to find the subgraph with the largest average degree.
  - Communities in social networks
  - Expert teams in co-authorship graphs
  - Spam links in web graphs



State of the Art: Goldberg's Algorithm[1]

- Use vertex subset S to denote node-induced subgraph
- Use $\bar{S}$ to denote $V \backslash S$
- Let $\rho(S)$ be the density of S: $\frac{|E(S)|}{|S|}$
- $\rho(S) > \lambda \iff 2|E| > 2\lambda|S| + |E(S,\bar{S})| + \sum_{u \in \bar{S}} d(u)$
  - $2|E| = 2|E(S)| + |E(S,\bar{S})| + |E(S,\bar{S})| + 2|E(\bar{S})|$
  - $2|E| = 2|E(S)| + |E(S,\bar{S})| + \sum_{u \in \bar{S}} d(u)$
  - $\rho(S) > \lambda \iff 2|E(S)| > 2\lambda|S|$

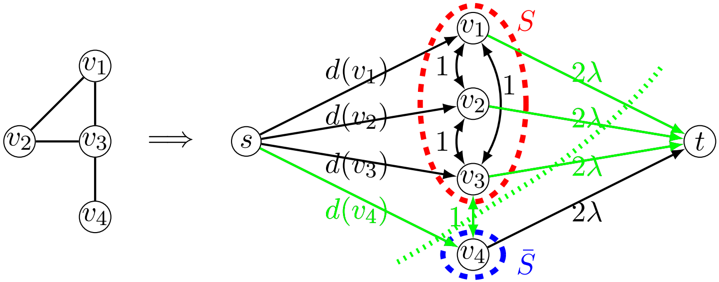
$$V = S \cup \bar{S}$$

$$E = E(S) \cup E(S,\bar{S}) \cup E(\bar{S})$$

[1] A. V. Goldberg. *Finding a Maximum Density Subgraph*. Tech. rep. Berkeley, CA, USA, 1984.

State of the Art: Goldberg's Algorithm[2]

- $\rho(S) > \lambda \iff 2\lambda|S| + |E(S,\bar{S})| + \sum_{u \in \bar{S}} d(u) < 2|E|$



λ-Graph G_λ

- The maximum density is larger than λ iff the minimum cut of G_λ has a value smaller than $2|E|$
- Conduct binary search on λ to find the densest subgraph

[2] A. V. Goldberg. *Finding a Maximum Density Subgraph*. Tech. rep. Berkeley, CA, USA, 1984.

Motivation: Deconstruct Densest Subgraphs

- Goldberg's algorithm identifies the maximal densest subgraph
- Most of the existing studies only care about maximal densest subgraphs, *e.g.*, they have been used in
  - Locally densest subgraph discovery[3]
  - Density-friendly graph decomposition[4]
- One exception: minimal densest subgraph is used in finding k subgraphs with maximum total density and limited overlap[5]
  - However, finding one minimal densest subgraph triggers $\mathcal{O}(n \log n)$ instances of finding a densest subgraph
- Moreover, the relationship among all densest subgraphs in a graph is unclear.

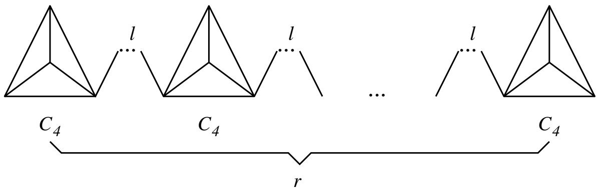
[3] L. Qin et al. "Locally Densest Subgraph Discovery". In: *Proc. of KDD'15*. 2015.

[4] M. Danisch, T.-H. H. Chan, and M. Sozio. "Large Scale Density-friendly Graph Decomposition via Convex Programming". In: *Proc. of WWW'17*. 2017.

[5] O. D. Balalau et al. "Finding Subgraphs with Maximum Total Density and Limited Overlap". In: *Proc. of WSDM'15*. 2015.

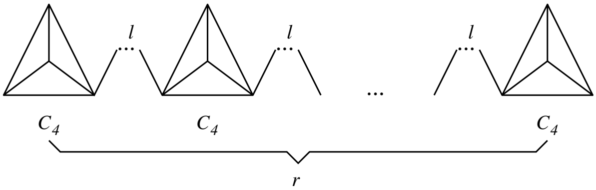
Fact 1: Number of Densest Subgraphs can be Exponential

- Let's consider the following graph
  - It contains r 4-cliques C_4
  - Each pair of consecutive 4-cliques is connected by a path with l intermediate nodes



- For $l > 2$
  - Each 4-clique is a minimal densest subgraph
  - Each densest subgraph is a union of a subset of the 4-cliques
  - There are $2^r - 1$ densest subgraphs in total

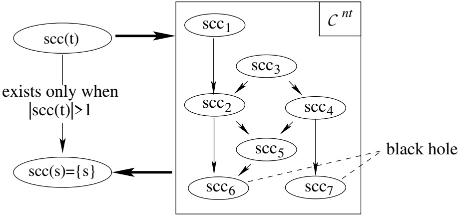
Fact 2: Densest Subgraph is NOT simply Union of Minimal Densest Subgraphs



- For $l = 2$
  - Minimal densest subgraphs are still the 4-cliques
  - A union of a subset of 4-cliques is still a densest subgraph
  - But now, densest subgraphs have more shapes
    - E.g., the union of the first two 4-cliques and the path that connects these two 4-cliques is a densest subgraph

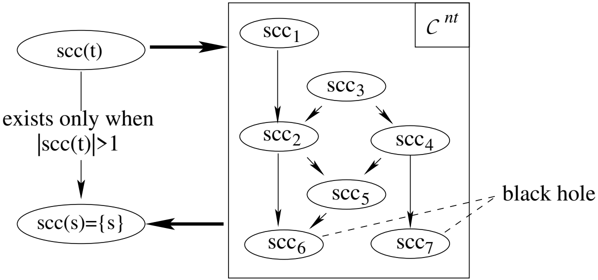
Deconstruct Densest Subgraphs

- The key to the structure of all densest subgraphs of G is hidden in the λ-graph with $\lambda = \rho^*(G)$ (the maximum density of G)
  1. Given G
  2. Compute $\lambda^* = \rho^*(G)$ by using Goldberg's algorithm
  3. Construct G_{λ^*}
  4. Run maximum-flow algorithm on G_{λ^*}, let $\mathcal{H}_{f^*}$ be the resulting residual graph which is a weighted directed graph
  5. Contract each strongly connected component (SCC) into a super-node, call the resulting graph critical component graph $\mathcal{H}^C$



Deconstruct Densest Subgraphs

- The maximal densest subgraph is the subgraph induced by vertices in $\mathcal{C}^{nt}$
  - $\mathcal{C}^{nt}$: the set of all SCCs excluding $scc(s)$ and $scc(t)$
- Let L be the size of the maximal densest subgraph
- All minimal densest subgraphs can be reported in $\mathcal{O}(L)$ time in total
- All densest subgraphs can be enumerated with a delay of $\mathcal{O}(L)$



The critical component graph $\mathcal{H}^C$

Scale the Computation in Practice

- The time complexity of Goldberg's algorithm is $\mathcal{O}(n \cdot m \cdot \log \frac{n^2}{m})$ by using parametric maximum flow[6]
  - This is the bottleneck of deconstructing densest subgraphs
- To scale the computation to large graphs, we first reduce the graph instance
  - All vertices whose degrees are smaller than $\rho^*(G)$ can be removed

Let S^* be the maximal densest subgraph of G, then the minimum degree of S^* is no smaller than $\rho^*(G)$, *i.e.*, $d_{min}(S^*) \geq \rho^*(G)$

- Use a lower bound of $\rho^*(G)$, obtained by the linear-time 2-approximation algorithm[7], for pruning
  - Iteratively remove from the graph the minimum-degree vertex
  - The densest one among the n subgraphs provides a 2-approximation result

[6] G. Gallo, M. D. Grigoriadis, and R. E. Tarjan. "A Fast Parametric Maximum Flow Algorithm and Applications". In: *SIAM J. of Comp.* 18.1 (1989).

[7] M. Charikar. "Greedy approximation algorithms for finding dense components in a graph". In: *Proc. of APPROX'00*. 2000.

Experimental Setting

- Machine: 3.4GHz CPU, 16GB main memory
- All algorithms are implemented in `C++` and run in single-thread mode
- Report the overall running time, excluding only the I/O time

Graph Statistics and Reduced Graph Sizes

Graph G	$\lvert V(G)\rvert$	$\lvert E(G)\rvert$	$\rho^*(G)$	$\lceil\tilde{\rho}\rceil$	Reduced graph G^-		
					$\lvert V(G^-)\rvert$	$\lvert E(G^-)\rvert$	$\frac{\lvert E(G^-)\rvert}{\lvert E(G)\rvert}$
dblp	317,080	1,049,866	56.57	57	280	13,609	1.30%
web-Stanford	281,903	1,992,636	59.39	60	1,370	78,797	3.95%
com-youtube	1,134,890	2,987,624	45.60	46	2,269	103,342	3.46%
web-Google	875,713	4,322,051	28.04	28	787	16,641	0.39%
WikiTalk	2,388,953	4,656,682	114.14	115	1,384	157,968	3.39%
youtube-growth	3,223,585	9,375,374	77.47	78	1,219	94,427	1.01%
as-skitter	1,694,616	11,094,209	89.40	90	915	73,480	0.66%
soc-flickr-und	1,715,255	15,555,041	468.83	469	3,135	1,469,797	9.45%
patent	3,774,768	16,518,947	40.13	41	730	25,697	0.16%
soc-pokec	1,632,803	22,301,964	41.13	42	8,974	368,613	1.65%
LiveJournal	4,843,953	42,845,684	229.85	228	3,639	661,891	1.54%
twitter-mpi	9,862,152	99,940,317	602.44	603	8,448	5,089,428	5.09%
tech-p2p	5,792,297	147,829,887	750.18	751	7,641	5,732,158	3.88%
uk-2002	18,459,128	261,556,721	471.50	472	3,429	1,231,751	0.47%
uk-2005	39,252,879	781,439,892	485.75	429	51,784	15,037,470	1.92%
webbase	115,554,441	854,809,761	816.92	804	9,990	6,631,895	0.78%
it-2004	41,290,577	1,027,474,895	2008.19	2,009	4,279	8,593,024	0.84%
twitter-2010	41,652,230	1,202,513,046	1643.30	1,644	11,619	17,996,107	1.50%

Time for Computing All Minimal Densest Subgraphs

Graph G	$\lvert V(G)\rvert$	$\lvert E(G)\rvert$	Processing time (seconds)		
			Reduction	Flow	Total
dblp	317,080	1,049,866	0.049	0.021	0.07
web-Stanford	281,903	1,992,636	0.083	0.15	0.233
com-youtube	1,134,890	2,987,624	0.193	0.807	1
web-Google	875,713	4,322,051	0.336	0.11	0.446
WikiTalk	2,388,953	4,656,682	0.195	0.935	1.13
youtube-growth	3,223,585	9,375,374	0.982	0.768	1.75
as-skitter	1,694,616	11,094,209	0.644	0.286	0.93
soc-flickr-und	1,715,255	15,555,041	0.566	19.434	20
patent	3,774,768	16,518,947	2.88	0.6	3.48
soc-pokec	1,632,803	22,301,964	1.87	6.53	8.4
LiveJournal	4,843,953	42,845,684	4.65	1.72	6.37
twitter-mpi	9,862,152	99,940,317	5.36	164.64	170
tech-p2p	5,792,297	147,829,887	19	283	302
uk-2002	18,459,128	261,556,721	9.6	2.4	12
uk-2005	39,252,879	781,439,892	26	77	103
webbase	115,554,441	854,809,761	61	36	97
it-2004	41,290,577	1,027,474,895	30	53	83
twitter-2010	41,652,230	1,202,513,046	143	320	463

Plugin Our Algorithm for Top-k Densest Subgraph Computation[8]

Graph G	TopkDS (seconds)	The existing algorithm
dblp	0.635	
web-Stanford	1.9	0.3 hours
com-youtube	5.1	0.54 hours
web-Google	6.2	2.15 hours
WikiTalk	4.5	
youtube-growth	18	1.5 hours
as-skitter	12	1.29 hours
soc-flickr-und	80	
patent	35	
soc-pokec	480	
LiveJournal	67	
twitter-mpi	624	
tech-p2p	1,671	
uk-2002	142	
uk-2005	515	
webbase	813	
it-2004	1,523	
twitter-2010	10,205	

[8] O. D. Balalau et al. "Finding Subgraphs with Maximum Total Density and Limited Overlap". In: *Proc. of WSDM'15*. 2015.

Conclusions

- We studied the relationship among all densest subgraphs of a graph
- By conducting a precomputation of $\mathcal{O}(n \cdot m \cdot \log \frac{n^2}{m})$, the same as the time complexity of Goldberg's algorithm
  - All minimal densest subgraphs can be reported in $\mathcal{O}(L)$ time in total
  - All densest subgraphs can be enumerated with a delay of $\mathcal{O}(L)$
  - L is the size of the maximal densest subgraph
- The source code of our algorithm will be available at https://github.com/LijunChang/Cohesive_subgraph_book/densest_subgraph