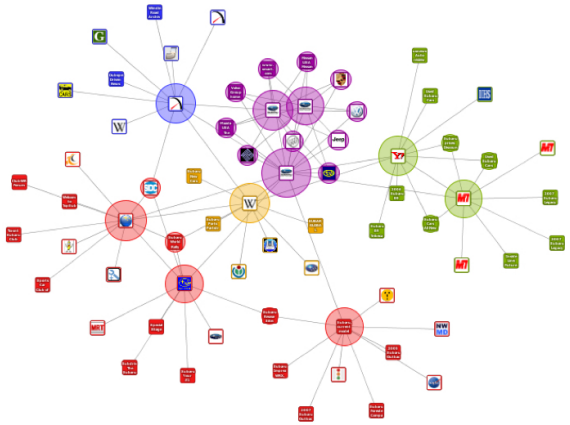
Risk-Averse Graph Mining

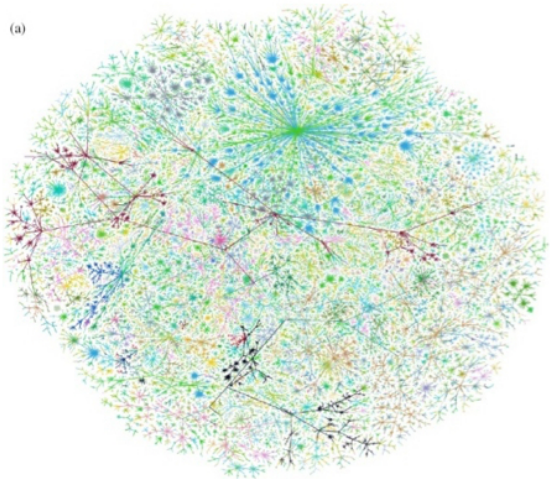
Charalampos E. Tsourakakis
Boston University

CS591 Graph Analytics

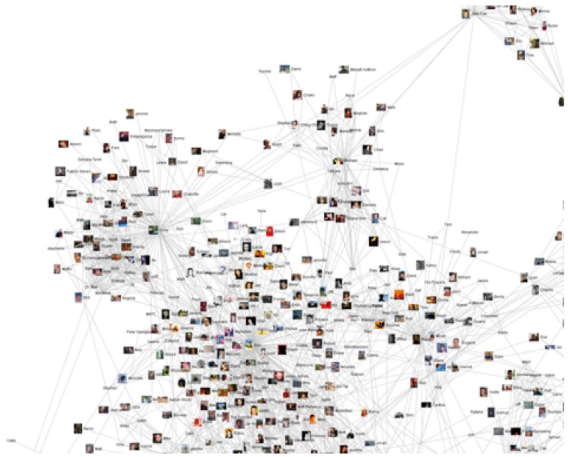
Graphs are ubiquitous...



Computer network



Internet



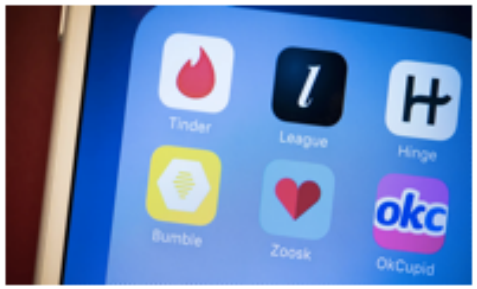
Social network

Connectome

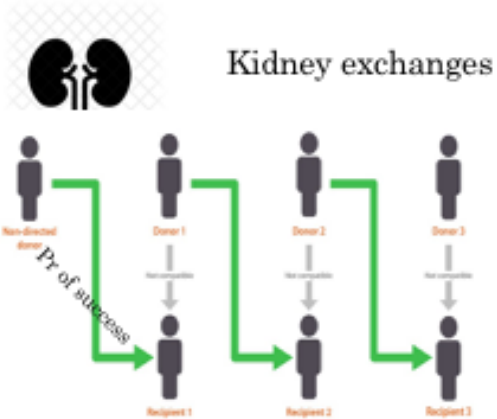
Airline network

Images

and many of them are uncertain!



Online dating



Uncertain (aka stochastic) graphs and hypergraphs are ubiquitous!

- PPI networks [Asthana et al., 2004, Krogan et al., 2006]
- Dating apps
- Kidney exchange [Roth et al., 2004]
- Sensor networks
- Influence maximization [Kempe et al., 2003]
- Injecting privacy [Boldi et al., 2012]
- ...

Risk aversion

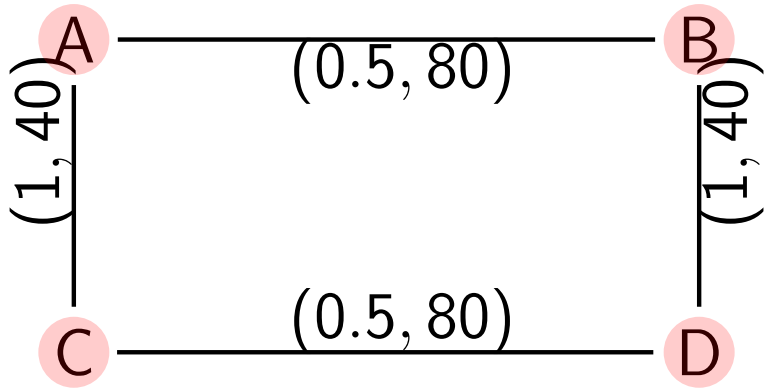
Suppose you have 100$, and two hedge funds with the same expected return. How should you invest them?

Risk averse optimization is a major topic in OR, control theory and finance.

- Suppose that $f(\omega, X)$ is a cost function of a random variable X, and a decision variable ω.
- The goal of risk-averse optimization is to choose ω such that both $\mathbb{E}[f]$ and $R(f)$ are small.
- Foundations of portfolio theory ([Markowitz, Nobel prize in Economics 1990])

Risk averse graph mining

Graph matchings:



- $(A, B), (C, D)$ expected reward 80
- $(A, C), (B, D)$ also expected reward 80

Our key question:

How can we find graph structures with high expected reward, and low risk?

Outline of today's talk

1. Introduction
2. Uncertain (hyper)graph model
3. Risk-averse (hyper)graph matchings
4. Risk-averse dense subgraphs (and a bonus extension)
5. Open problems

Uncertain graph model

Existing work has focused on the following model (e.g., [Bonchi et al., 2014, Kollios et al., 2013])

- Let $\mathcal{G} = (V, E, p)$ be an uncertain (hyper)graph where $p : E \rightarrow (0, 1]$.
- (Hyper)edge e exists with probability p_e independently from the rest of the edges
- Possible-world semantics interprets $\mathcal{G}$ as a set $\{G : (V, E_G)\}_{E_G \subseteq E}$ of $2^{|E|}$ possible deterministic graphs (worlds)

$$\mathbf{Pr}[G] = \prod_{e \in E_G} p(e) \prod_{e \in E \setminus E_G} (1 - p(e)).$$

- **Weighted case:** (Hyper)edge e reward equals w_e with probability p_e, reward 0 with probability $1 - p_e$

Uncertain graph model (general)

- Let $\mathcal{G}([n], E, \{f_e(\theta_e)\}_{e \in E})$ be an uncertain complete graph on n nodes, $E = \binom{[n]}{2}$.

- We assume that the weight of each edge is drawn independently from the rest; Let f_e be the probability distribution for edge e with parameters $\vec{\theta_e}$:

$$w(e) \sim f_e(x; \vec{\theta_e}) \forall e \in E.$$

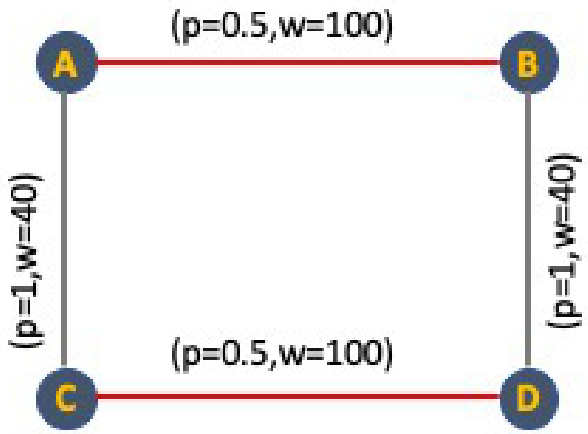
- Likelihood/probability of a given graph G:

$$\mathbf{Pr}\left[G; \{w(e)\}_{e \in E}\right] = \prod_{e \in E} f_e(w(e); \vec{\theta_e}). \tag{1}$$

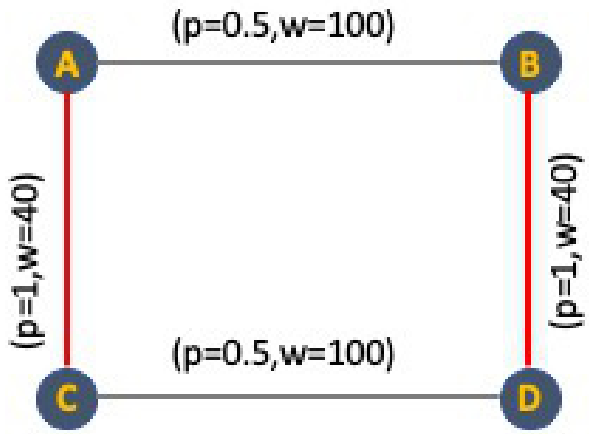
Outline of today's talk

1. Introduction
2. Uncertain (hyper)graph model
3. Risk-averse (hyper)graph matchings
4. Risk-averse dense subgraphs (and a bonus extension)
5. Open problems

Back to our example



- Maximum Expected Weight Matching
- Reward = zero with p = 0.25



- Risk-Averse Matching
- Reward = 80 with p = 1

Standard approach

- Maximize expected weight

- Solvable exactly in (strongly) polynomial time using Edmond's algorithm. (Originally in $O(n^4)$ time [Edmond's 1965], nowadays in $O(nm + n^2 \log n)$ time [Gabow, 1990])
- Greedy algorithm: $\frac{1}{2}$-approximation.
- NP-hard for hypergraphs. Greedy $\frac{1}{k}$-approximation where k is the maximum cardinality of an edge.

- Issue: Optimal matching in expectation may involve significant risk!
- Instead of just maximizing the expected reward, can we optimize efficiently over matchings with bounded risk?

Formulation

$$\begin{array}{ll} \max\limits_{M \in \mathcal{M}} & R(M) \qquad \text{[BR-MWM problem]} \\ \text{s.t} & \mathit{risk}(M) \leq B \end{array} \tag{2}$$

- $R(M)$ is the expected reward of a matching, i.e.,

$$R(M) = \sum_{e \in M} \mu_e.$$

- The risk *intuitively* is associated with the variance.

$$\mathit{risk}(M) = \sum_{e \in M} \text{risk}(e).$$

We use the following versions of edge risk: $\text{risk}(e) = \sigma_e^2, \sigma_e$.

Hardness

Theorem

The BR-MWM PROBLEM IS NP-HARD.

Remark: While finding maximum weight matchings in graphs is poly-time solvable, our problem becomes NP-hard even for graphs.

This naturally brings the following question:

- Can we design fast, efficient approximation algorithms?

Yes!

From now on let MATCH-ALG be a black-box algorithm we use to find maximum weight matchings on a (hyper)graph.

Proposed algorithm

The "heart" of our algorithm consists of the following steps.

1. Remove all hyperedges that can never appear in a maximum weight matching (i.e., $\mu_e \leq 0, \text{risk}(e) > B$).
2. Define $\alpha_e = \frac{\mu_e}{\text{risk}(e)}$. Sort the hyperedges in non-increasing order, i.e., $\alpha_{e_1} \geq \alpha_{e_2} \geq \ldots \geq \alpha_{e_m}$.
3. This creates a sequence of subgraphs $\emptyset = H^{(0)} \subset H^{(1)} \subset \ldots \subset H^{(m)} = H$, let $M^{(i)}$ be the matching returned by MATCH-ALG on $H^{(i)}$.
4. Find index ℓ^* for which $risk(M^{(\ell^*)}) \leq B < risk(M^{(\ell^*+1)})$.
5. Output $M^{(\ell^*)}$ or e_{ℓ^*+1} depending on which one achieves greater expected reward.

Proposed algorithm

Theorem

Let $T(m,n)$ be the running time of Match-Alg. *If* Match-Alg *achieves a c-approximation ($c \leq 1$), our algorithm achieves $\frac{c}{c+2}$ approximation. It can be implemented in $O(m \log m + \log m T(m,n))$ time using binary search for ℓ^*.*

- **Corollary 1**: For graphs, using an exact algorithm we get a $\frac{1}{3}$-approximation.
- **Corollary 2**: For graphs, using the greedy algorithm we get a $\frac{1}{5}$-approximation in $O(n \log m + m \log^2 m)$ time.
- **Corollary 3**: For hypergraphs, using the greedy we get a $\Omega(\frac{1}{k})$-approximation.

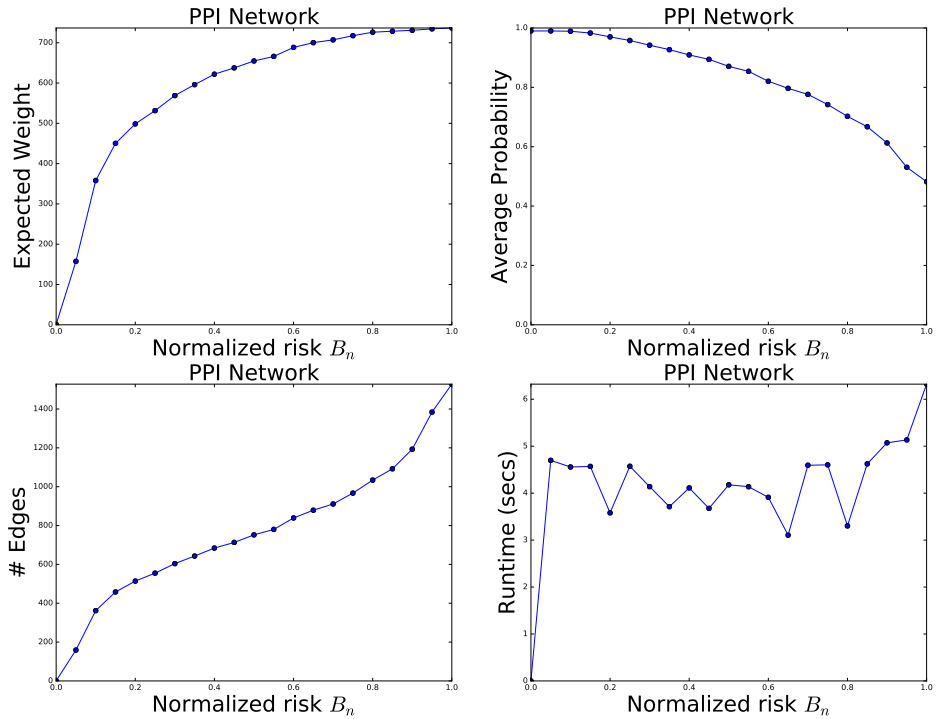
Experimental setup

- We normalize the allowed risk B by dividing by an upper bound $B_{\max}$ on the total risk.
- We range B according to the rule:

$$B = B_n \times B_{\max},$$

  where $B_n \in [0, 1]$ and is incremented in steps of 0.05. We refer to B_n as the *normalized risk* from now on.
- All experiments were performed on a laptop with 1.7 GHz Intel Core i7 processor and 8GB of main memory.
- The code is available at `https://github.com/tsourolampis/risk-averse-graph-matchings`.

Experiments–PPI network



(a) Expected reward, (b) average probability (over matching's edges), (c) number of edges in the matching, and (d) running time in seconds versus normalized risk B_n for the uncertain PPI network.

Experiments – Recommending impactful and probable recommendations

- Academic collaboration is an ideal playground to explore the effect of risk-averse team formation for research projects:
  - Research potential
  - Chances of collaboration
- Let P_i be the set of papers authored by researcher i.
- **Hypergraph construction:**
  - Nodes $\leftrightarrow$ DBLP authors
  - Hyperedge $\leftrightarrow$ Paper
  - Weight $\leftrightarrow$ Citations
  - Hyperedge probability $p(e)$ set to:

$$p_e = \frac{|P_1 \cap P_2 \cap \ldots \cap P_\ell|}{|P_1 \cup P_2 \cup \ldots \cup P_\ell|}.$$

Collaboration dataset description

- Our dataset consists of
- $n = 1,752,443$ nodes and,
- $m = 3,227,380$ hyperedges.

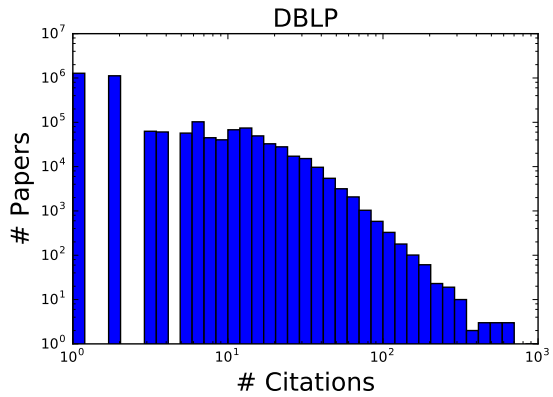
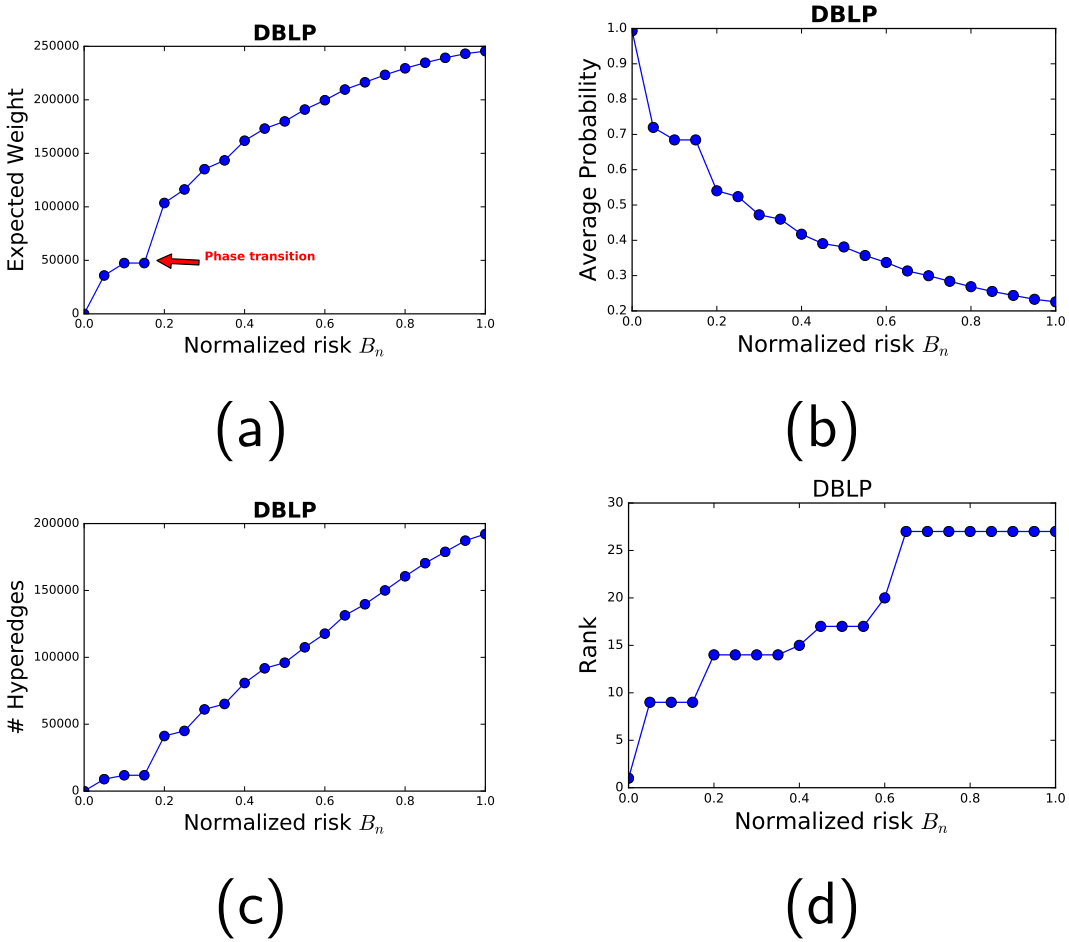


Figure: DBLP citation histogram.

Experimental findings

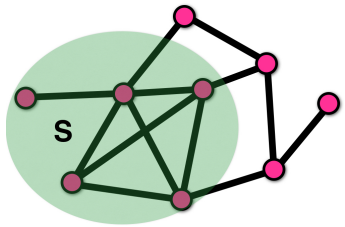


(a) Expected reward, (b) average probability, (c) number of edges in the hypermatching, and (d) hypergraph rank k versus normalized risk B_n.

Outline of today's talk

1. Introduction
2. Uncertain (hyper)graph model
3. Risk-averse (hyper)graph matchings
4. Risk-averse dense subgraphs (and a bonus extension)
5. Open problems

Densest subgraph problem (DSP)

Degree density: $\boxed{\rho(S) = \frac{e(S)}{|S|}}$. E.g., 

$\rho(S) = \frac{7}{5}$

- $\max_{S \subseteq V} \rho(S)$ **Poly-time solvable for non-negative weights!**
  (via max flows)
  [Goldberg, 1984, Gallo et al., 1989, Khuller et al., 2009]

  2-approximation algorithm which uses linear space $O(n+m)$ and runs in linear time $O(n+m)$ due to Charikar

  "The densest subgraph problem **(DSP)** *lies at the core of large scale data mining"* [Bahmani et al., 2012]
- DSP is not studied on uncertain graphs!

Risk-averse DSD formulation

Intuitively, our goal is to find a subgraph $G[S]$ induced by $S \subseteq V$ such that:

1. Its average expected reward $\frac{\sum_{e \in E(S)} w_e}{|S|}$ is large.
2. The associated average risk is low $\frac{\sum_{e \in E(S)} \sigma_e^2}{|S|}$.

We approach the problem as follows:

- For each edge we create two edges:
  1. A positive edge with weight equal to the expected reward, i.e., $w^+(e) = \mu_e$
  2. A negative edge with weight equal to the opposite of the risk of the edge, i.e., $w^-(e) = \sigma_e^2$.

Risk-averse DSD formulation

- Our goal is to find a subgraph $S \subseteq V$ such that:
  1. large positive average degree $\frac{w^+(S)}{|S|}$ (large reward)
  2. small negative average degree $\frac{w^-(S)}{|S|}$ (small risk)

We combine the two objectives into one objective $f : 2^V \to \mathbb{R}$ that we wish to maximize:

$$f(S) = \frac{w^+(S) + \lambda_1 |S|}{w^-(S) + \lambda_2 |S|}.$$

Questions: But can maximize this objective in polynomial time? Can we solve the DSP in poly-time when the weights are negative?

Insights

If we can answer the following query in polynomial time, then by binary search we can solve the problem:

> *Does there exist a subset of nodes $S \subseteq V$ such that $f(S) \geq q$, where q is a query value?*

$$\frac{w^+(S) + \lambda_1|S|}{w^-(S) + \lambda_2|S|} \geq q \rightarrow$$

$$w^+(S) + \lambda_1|S| \geq q(w^-(S) + \lambda_2|S|) \rightarrow$$

$$\sum_{e \in E(S)} \underbrace{\left(w^+(e) - qw^-(e)\right)}_{\tilde{w}(e)} \geq |S| \underbrace{(q\lambda_2 - \lambda_1)}_{q'} \rightarrow \sum_{e \in E(S)} \frac{\tilde{w}(e)}{|S|} \geq q'.$$

Hardness

Theorem

The DSP on graphs with negative weights is NP-hard.

Reduction from MAX-CUT.

However, by our insights from the previous slide, we observe the following:

Corollary: Assume that $w^+(e) \geq q_{max} w^-(e)$ for all $e \in E^+ \cup E^-$, where q_{max} is the maximum possible query value. Then, the densest subgraph problem is solvable in polynomial time.

Bounding risk: $f(S) = \frac{w^+(S)+\lambda_1|S|}{Bw^-(S)+\lambda_2|S|}$ by changing parameter B.

Algorithm - DSP with Negative Weights

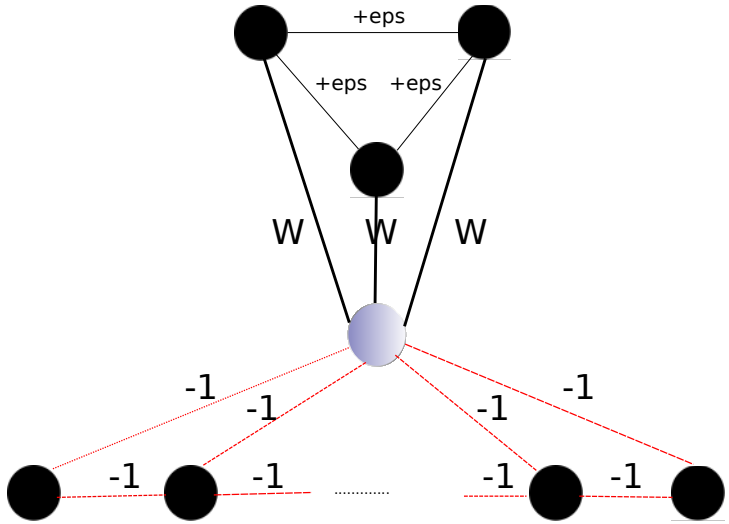
Algorithm 2 `Peeling`(G)

$n \leftarrow |V|, H_n \leftarrow G$
for $i \leftarrow n$ to 2 **do**
  Let v be the vertex of G_i of minimum degree, i.e., $d(v) = deg^+(v) - deg^-(v)$ (break ties arbitrarily)
  $H_{i-1} \leftarrow H_i \backslash v$
end for
Return H_j that achieves maximum average degree among H_is, $i = 1, \ldots, n$.

Theorem

Let $G(V, E, w)$, $w : E \rightarrow \mathbb{R}$ be an undirected weighted graph with possibly negative weights. If the negative degree $deg^-(u)$ of any node u is upper bounded by Δ, then our Algorithm outputs a set whose density is at least $\frac{\rho^*}{2} - \frac{\Delta}{2}$.

Bad instance



Let $W = \frac{n-4}{3}$. Then, $3W - n < -3$. The degrees of the $n+4$ nodes are as follows:

$$\underbrace{3W - n}_{\text{one node}} < \underbrace{-3}_{n-2 \text{ nodes}} < \underbrace{-2}_{\text{two nodes}} < 0 < \underbrace{2\epsilon + W}_{\text{three nodes}}.$$

Heuristic

Algorithm 3 `Heuristic-Peeling`(G, C)

Input: $C \in (0, +\infty)$

$n \leftarrow |V|, H_n \leftarrow G$

for $i \leftarrow n$ to 2 **do**

Let v be the vertex of G_i of minimum degree, i.e., $d(v) = Cdeg^+(v) - deg^-(v)$ (break ties arbitrarily)

$H_{i-1} \leftarrow H_i \backslash v$

end for

Return H_j that achieves maximum average degree among H_is, $i = 1, \ldots, n$.

Rule of thumb: Run the above heuristic for various values of C, and return the best possible subgraph!

Bonus extension – Exclusion queries

We can use our heuristic to develop a new algorithmic primitive!

Problem

Given a multigraph $G(V, E, \ell)$, where $\ell : E \to \{1, \ldots, L\} = [L]$ is the labeling function, and L is the number of types of interactions, and an input set $\mathcal{I} \subseteq [L]$ of interactions, how do we find a set of nodes S that (i) induces a dense subgraph, and (ii) does not induce any edge e such that $\ell(e) \in \mathcal{I}$?

Application: Given the daily Twitter interactions, find a dense subgraph in *follows* and *quotes* but with no *replies*.

Approach: Use $-\infty$ weights for the excluded edge types.

Datasets

Name	n	m
Biogrid	5 640	59 748
Collins	1 622	9 074
Gavin	1 855	7 669
Krogan core	2 708	7 123
Krogan extended	3 672	14 317
TMDB	160 784	883 842
Twitter (Feb. 1)	621 617	(902 834, 387 597, 222 253, 30 018, 63 062)
Twitter (Feb. 2)	706 104	(1 002 265, 388 669, 218 901, 29 621, 64 282)
Twitter (Feb. 3)	651 109	(1 010 002, 373 889, 218 717, 27 805, 59 503)
Twitter (Feb. 4)	528 594	(865 019, 435 536, 269 750, 32 584, 71 802)
Twitter (Feb. 5)	631 697	(999 961, 396 223, 233 464, 30 937, 66 968)
Twitter (Feb. 6)	732 852	(941 353, 407 834, 239 486, 31 853, 67 374)
Twitter (Feb. 7)	742 566	(1 129 011, 406 852, 236 121, 30 815, 68 093)

Experimental findings – Exploring B

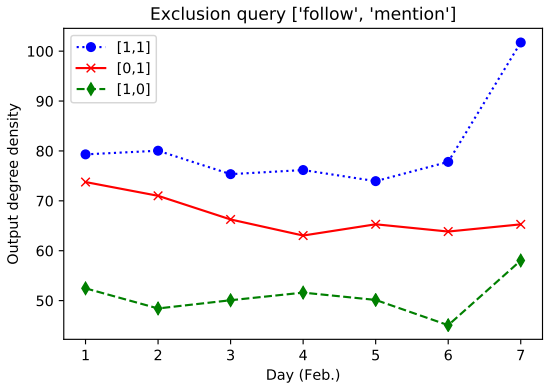
We test the trade-off between reward and risk by ranging B.

B	Average exp. reward	average risk
0.25	0.18	0.09
1	0.17	0.08
2	0.13	0.06

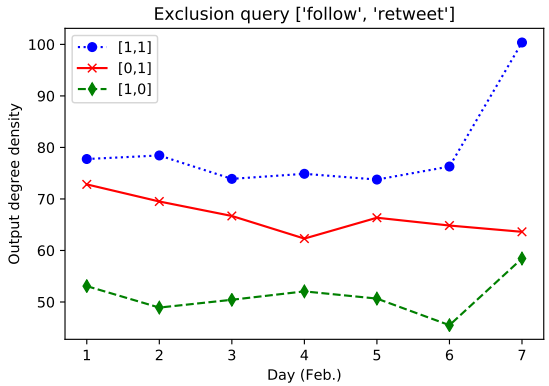
Gavin dataset ($n = 1\,855, m = 7\,669$).

Experimental findings – Exclusion queries on Twitter

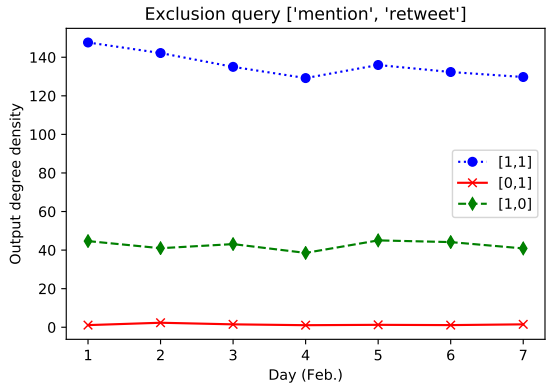
We set $C = 1, W = -\infty$:



(α)



(β)



(γ)

Degree density for three exclusion queries per each pair of interaction types over the period of the first week of February 2018. (α) Follow and mention. (β) Follow and retweet. (γ) Mention and retweet.

Experimental findings - Ranging W, C

C	W	$\lvert S^* \rvert$	$\rho_{\text{retweet}}(S^*)$	$\rho_{\text{reply}}(S^*)$
0.1	1	296	63.44	-0.75
	5	99	45.67	-0.01
	200 000	200	30.37	0
1	1	346	72.70	-2.75
	5	319	68.70	-1.29
	200 000	200	30.38	0
10	1	351	73.10	-3.31
	5	351	73.10	-3.31
	200 000	200	30.37	0

Exploring the effect of the negative weight $-W$ on the excluded edge types for various C values.

Outline of today's talk

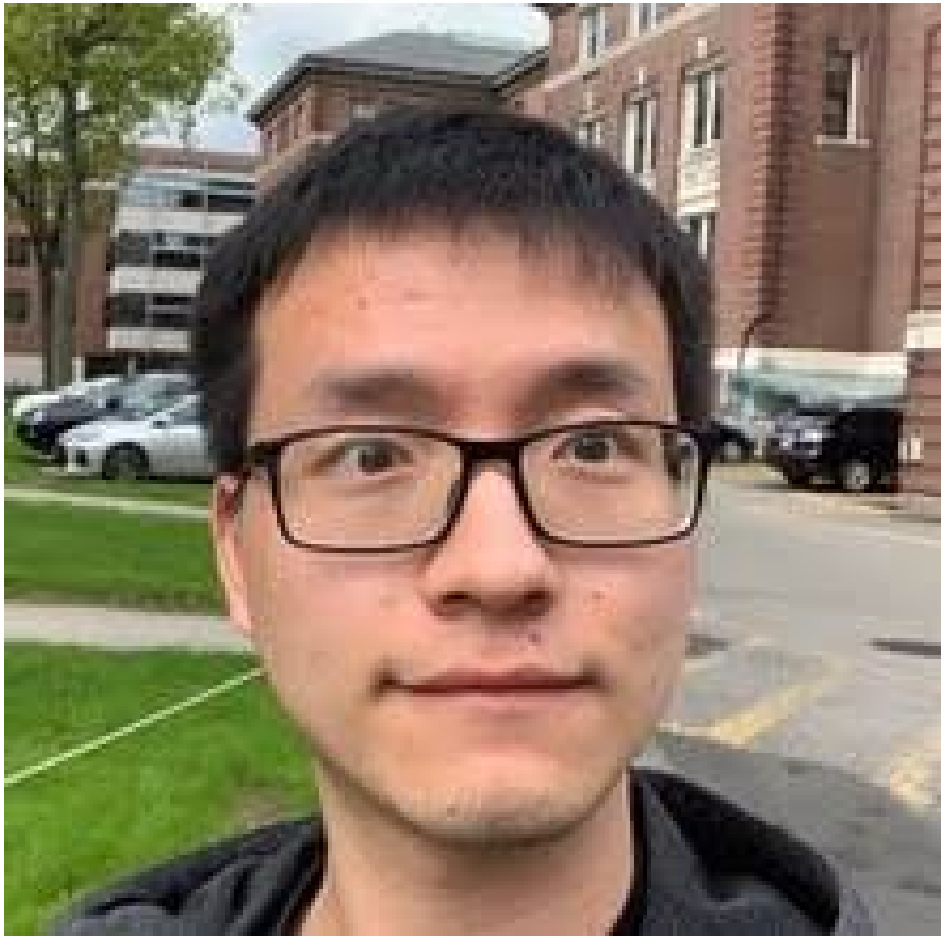
1. Introduction
2. Uncertain (hyper)graph model
3. Risk-averse (hyper)graph matchings
4. Risk-averse dense subgraphs (and a bonus extension)
5. Open problems

Open problems

- **Matchings:** Improve approximation guarantees for risk-averse graph matchings
- **Dense subgraphs:** Study in greater depth the computational complexity of DSD with negative weights
- **General direction:** Design risk-averse algorithms that combine efficiency, and solid theoretical guarantees

Readings

1. *Novel Dense Subgraph Discovery Primitives: Risk Aversion and Exclusion Queries* by Tsourakakis, Chen, Kakimura, Pachocki
2. *Risk-Averse Matchings over Uncertain Graph Databases* by Tsourakakis, Sekar, Lam (BU senior at the time, now at Amazon), Yang



Tianyi Chen (a classmate of yours)

references I

Asthana, S., King, O. D., Gibbons, F. D., and Roth, F. P. (2004).

Predicting protein complex membership using probabilistic network reliability.

Genome research, 14(6):1170–1175.

Bahmani, B., Kumar, R., and Vassilvitskii, S. (2012).

Densest subgraph in streaming and mapreduce.

Proc. VLDB Endow., 5(5):454–465.

Boldi, P., Bonchi, F., Gionis, A., and Tassa, T. (2012).

Injecting uncertainty in graphs for identity obfuscation.

Proceedings of the VLDB Endowment, 5(11):1376–1387.

references II

Bonchi, F., Gullo, F., Kaltenbrunner, A., and Volkovich, Y. (2014).

Core decomposition of uncertain graphs.

In *Proc. of the 20th ACM SIGKDD conference*, pages 1316–1325. ACM.

Kempe, D., Kleinberg, J., and Tardos, É. (2003).

Maximizing the spread of influence through a social network.

In *Proceedings of KDD 2003*, pages 137–146. ACM.

Kollios, G., Potamias, M., and Terzi, E. (2013).

Clustering large probabilistic graphs.

IEEE Transactions on Knowledge and Data Engineering, 25(2):325–336.

references III

Krogan, N. J., Cagney, G., Yu, H., Zhong, G., Guo, X., Ignatchenko, A., Li, J., Pu, S., Datta, N., Tikuisis, A. P., et al. (2006).

Global landscape of protein complexes in the yeast saccharomyces cerevisiae.

Nature, 440(7084):637.

Roth, A. E., Sönmez, T., and Ünver, M. U. (2004).

Kidney exchange.

The Quarterly Journal of Economics, 119(2):457–488.