



电子科技大学

University of Electronic Science and Technology of China

# A Brief Introduction To Graph Reconstruction

*Data Mining Lab*  
*Reporter: Xinzuo Wang*

# An Outline:

- Graph (network) reconstruction by using information diffusion
- Graph reconstruction by compressive sensing

# Network Reconstruction:

- Network Reconstruction Problem:

Given a **network with missing edges**, how is it possible to **uncover the network structure** based on certain **observable** quantities extracted from partial measurements ?

E.g.

- Biological systems : uncover the Protein interaction data;
- Social networks : the existence of missing data is almost inevitable.

# State-of-the-art Methods :

- Using structural information :

- Aaron Clauset, Cristopher Moore, and Mark Newman. Structural Inference of Hierarchies in Networks. 2006.
- Aaron Clauset, Cristopher Moore, and M E J Newman. Hierarchical structure and the prediction of missing links in networks. *Nature*, 453(7191):98–101, 2008.
- Myunghwan Kim and Jure Leskovec. The Network Completion Problem: Inferring Missing Nodes and Edges in Networks. SIAM International Conference on Data Mining, pages 47–58, 2011.
- Roger Guimer' and Marta Sales-Pardo. Missing and spurious interactions and the reconstruction of complex networks. *Proceedings of the National Academy of Sciences of the United States of America*, 104(51):20167–20172, 2007.
- Steve Hanneke and Eric P Xing. Network Completion and Survey Sampling. *Aistats*, 5:209–215, 2009.

- Using non-structural information :

- Kevin Bleakley, Gérard Biau, and Jean-Philippe Vert. Supervised reconstruction of biological networks with local models. *Bioinformatics (Oxford, England)*, 23(13):i57–i65, 2007.
- Manuel Gomez-Rodriguez, Jure Leskovec, and Andreas Krause. Inferring Networks of Diffusion and Influence. 5(4), 2010.
- Steve Hanneke and Eric P Xing. Network Completion and Survey Sampling. *Aistats*, 5:209–215, 2009.
- Payam Siyari, Hamid R Rabiee, Mostafa Salehi, and Motahareh Eslami Mehdiabadi. Network Reconstruction under Compressive Sensing. pages 130–143, 2012.

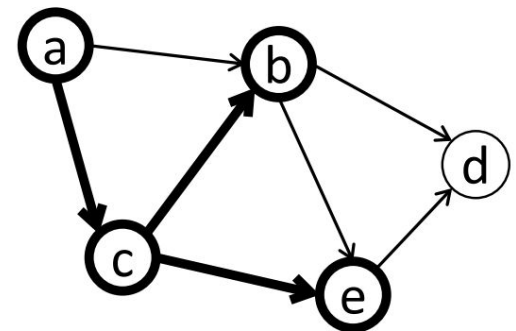
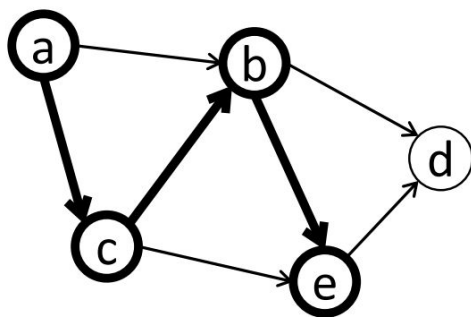
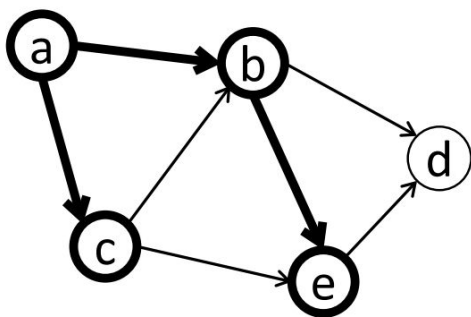
# Using diffusion information :

## Inferring Networks of Diffusion and Influence

Manuel Gomez-Rodriguez, Jure Leskovec, and Andreas Krause.

# General idea :

- A method for **tracing paths of diffusion** and influence through networks and **inferring** the networks over which contagions propagate.
  - Partially observed data
  - Tree structure assumption and probabilistic model
  - Find optimal network by maximize the likelihood
- A picture to illustrate



# Cascade Transmission Model :

- Probability of an individual transmission
  1. **Probability of infection** : When a new node  $u$  gets infected, it gets a chance to transmit the contagion to each of its currently uninfected neighbors  $w$ , **with some small probability  $\beta$** .
  2. **Incubation time** : If the contagion is transmitted, we then sample the incubation time --- how long after  $w$  got infected,  $w$  will get a chance to infect its (at that time uninfected) neighbors.

# Cascade Transmission Model :

- **Probability of an individual transmission**

Consider a pair of nodes  $u$  and  $v$ , connected by a directed edge  $(u, v)$ , and the corresponding hit times  $(t_u)_c$  and  $(t_v)_c$

If  $t_u < t_v$  then  $P_c(u, v) = 0$

If  $t_u > t_v$

$$P_c(u, v) = P_c(\Delta_{u,v}) \propto e^{-\frac{\Delta_{u,v}}{\alpha}}$$

$$\text{or } P_c(u, v) = P_c(\Delta_{u,v}) \propto \frac{1}{\Delta_{u,v}^\alpha}$$

1. When the cascade stops?

With probability  $(1 - \beta)$  the cascade stops, and never reaches  $v$ , thus  $t_v = \infty$ ;

2. With probability  $\beta$ , the cascade transmits over the edge  $(u, v)$ , and the hit c time  $t_v$  is set to  $t_u + \Delta_{u,v}$ , where  $\Delta_{u,v}$  is the incubation time that passed between hit times  $t_u$  and  $t_v$



# Cascade Transmission Model :

- Likelihood of a cascade spreading in a **given tree pattern T**

The likelihood  $P(c|T)$  that contagion  $c$  in a graph  $G$  **propagated in a particular tree pattern**  $T(V_T, E_T)$ .

Due to the assumption that **cascades are trees**, the likelihood is simply

$$P(c|T) = \prod_{(u,v) \in E_T} \beta P_c(u, v) \prod_{u \in V_T, (u,x) \in E \setminus E_T} (1 - \beta)$$

$$P(c|T) = \beta^q (1 - \beta)^r \prod_{(u,v) \in E_T} P_c(u, v)$$

---

The probability of the occurrence of a cascade is determined by the given tree structure.

Since the cascade spread in tree pattern  $T$ , the contagion successfully propagated along those edges. And, along the edges where the contagion did not spread, the cascade had to stop.

# Cascade Transmission Model :

- Cascade likelihood

A single contagion  $c$  propagates in a particular tree pattern  $T \in Tc(G)$  :

$$P(c|G) = \sum_{T \in \mathcal{T}_c(G)} P(c|T)P(T|G)$$

Assuming that all trees are a priori equally likely ( $P(T|G) = 1/|Tc(G)|$ ),

The probability of a set of cascades  $C$  occurring in  $G$

$$P(C|G) = \prod_{c \in C} P(c|G)$$

# Estimating the Network :

- Estimating the Network that Maximizes the Cascade Likelihood

Given a set of node infection times  $\mathbf{t}_c$  for a set of cascades  $\mathbf{c} \in \mathbf{C}$ , a propagation probability parameter  $\beta$  and an incubation time distribution  $P_c(\mathbf{u}, \mathbf{v})$ , find the network  $\hat{G}$  such that:

$$\hat{G} = \operatorname{argmax}_{|G| \leq k} P(C|G)$$

However, The time cost is too expensive !

# An Alternative Formulation :

- Compute an approximation of the likelihood of a single cascade by considering only the ***most likely tree*** instead of all possible propagation trees;
- Devising an algorithm that provably finds networks with ***near optimal approximate likelihood***.

# An Alternative Formulation :

## 1. Modeling external influence via $\epsilon$ -edges

--- *nodes may get infected for reasons **other than the network influence**.*

For example,

- In online media, not all the information propagates via the network, some is pushed onto the network by the **mass media**
- Similarly, in viral marketing, a person may purchase a product due to the influence of peers (network effect) or for some other reason (e.g., **seeing a commercial on TV**)

# An Alternative Formulation :

## 1. Modeling external influence via $\varepsilon$ -edges

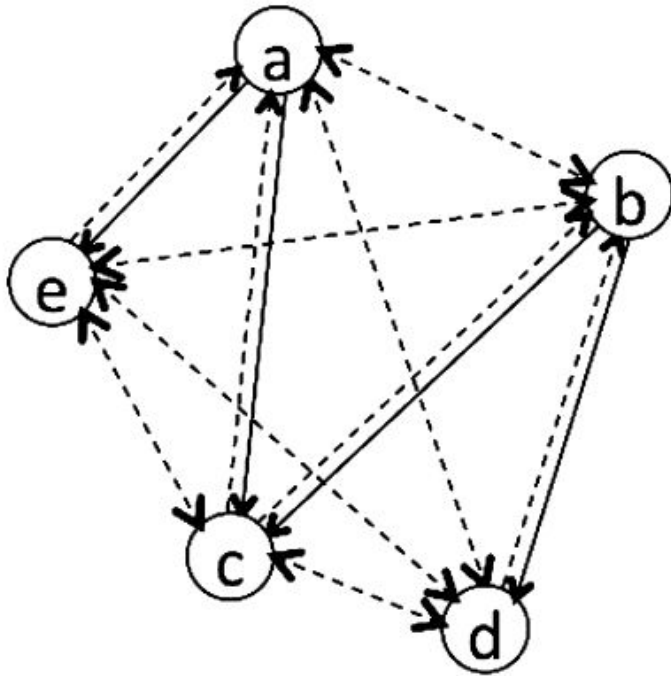
If there is **no network edge** between a node  $i$  and a node  $j$  in the network, we **add an  $\varepsilon$ -edge** and then **node  $i$  can infect node  $j$  with a small probability  $\varepsilon$** . In this way  $E \cap E_\varepsilon = \emptyset$ ,  $E \cup E_\varepsilon = V * V$

$$P(c|T) = \prod_{u \in V_T} \prod_{v \in V} P'_c(u, v)$$

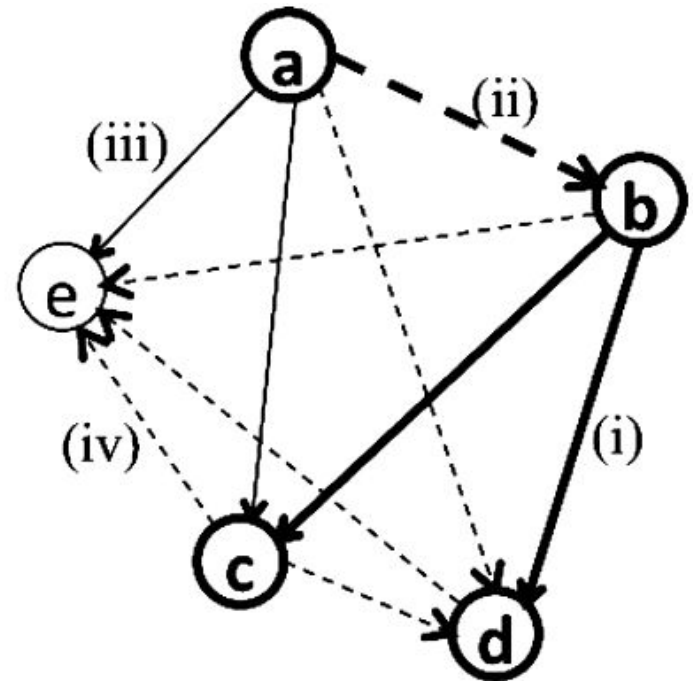
$$P'_c(u, v) = \begin{cases} \beta P_c(t_v - t_u) & \text{if } t_u < t_v \text{ and } (u, v) \in E_T \cap E & (u, v) \text{ is network edge} \\ \varepsilon P_c(t_v - t_u) & \text{if } t_u < t_v \text{ and } (u, v) \in E_T \cap E_\varepsilon & (u, v) \text{ is } \varepsilon\text{-edge} \\ 1 - \beta & \text{if } t_v = \infty \text{ and } (u, v) \in E \setminus E_T & v \text{ is not infected, network edge} \\ 1 - \varepsilon & \text{if } t_v = \infty \text{ and } (u, v) \in E_\varepsilon \setminus E_T & v \text{ is not infected, } \varepsilon\text{-edge} \\ 0 & \text{else (i.e., } t_u \geq t_v\text{).} \end{cases}$$

# An Alternative Formulation :

## 1. Modeling external influence via $\varepsilon$ -edges



(a) Graph  $G$  on five vertices and four network edges (solid edges).  $\varepsilon$ -edges shown as dashed lines



(b) Cascade propagation tree  $T = \{(a, b), (b, c), (b, d)\}$

# An Alternative Formulation :

2. Considering only the most likely propagation tree

$$P(C|G) = \prod_{c \in C} \sum_{T \in \mathcal{T}_c(G)} P(c|T) \approx \prod_{c \in C} \max_{T \in \mathcal{T}_c(G)} P(c|T).$$



# Experimental Evaluation :

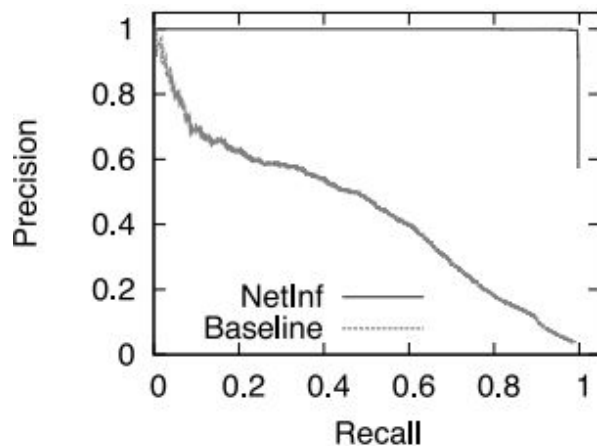
- Generating the minimum number of  $|C|$  cascades so that  $f$ -fraction of edges participated in at least one cascade  $|E_l| \geq f|E|$ . These  $|C|$  cascades generated the total of  $r$  edge transmissions, *i.e.*, average cascade size is  $r/|C|$ .

Table II. Performance of Synthetic Data

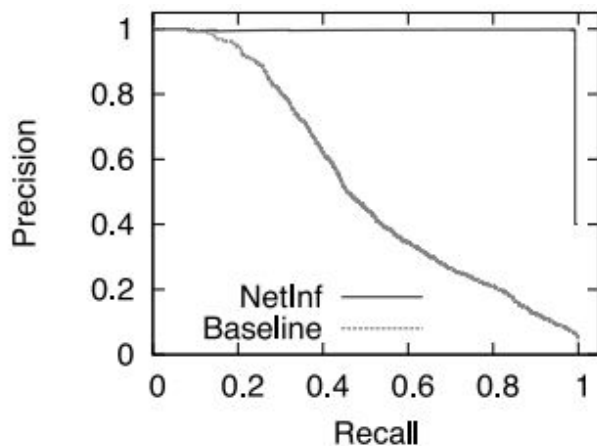
Type of network	$f$	$ C $	$r$	BEP	AUC
Forest Fire	0.5	388	2,898	0.393	0.29
	0.9	2,017	14,027	0.75	0.67
	0.95	2,717	19,418	0.82	0.74
	0.99	4,038	28,663	0.92	0.86
Hierarchical Kronecker	0.5	289	1,341	0.37	0.30
	0.9	1,209	5,502	0.81	0.80
	0.95	1,972	9,391	0.90	0.90
	0.99	5,078	25,643	0.98	0.98
Core-periphery Kronecker	0.5	140	1,392	0.31	0.23
	0.9	884	9,498	0.84	0.80
	0.95	1,506	14,125	0.93	0.91
	0.99	3,110	30,453	0.98	0.96
Flat Kronecker	0.5	200	1,324	0.34	0.26
	0.9	1,303	7,707	0.84	0.83
	0.95	1,704	9,749	0.89	0.88
	0.99	3,652	21,153	0.97	0.97

# Experimental Evaluation :

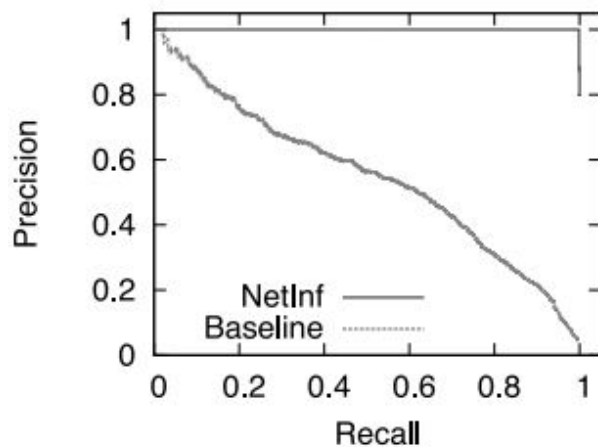
- Recall and Precision



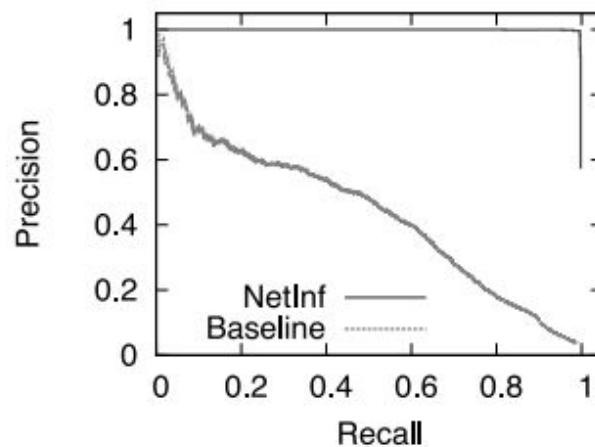
(a) Hier. Kronecker (Exp)



(b) Core-Periph. Kronecker (Exp)



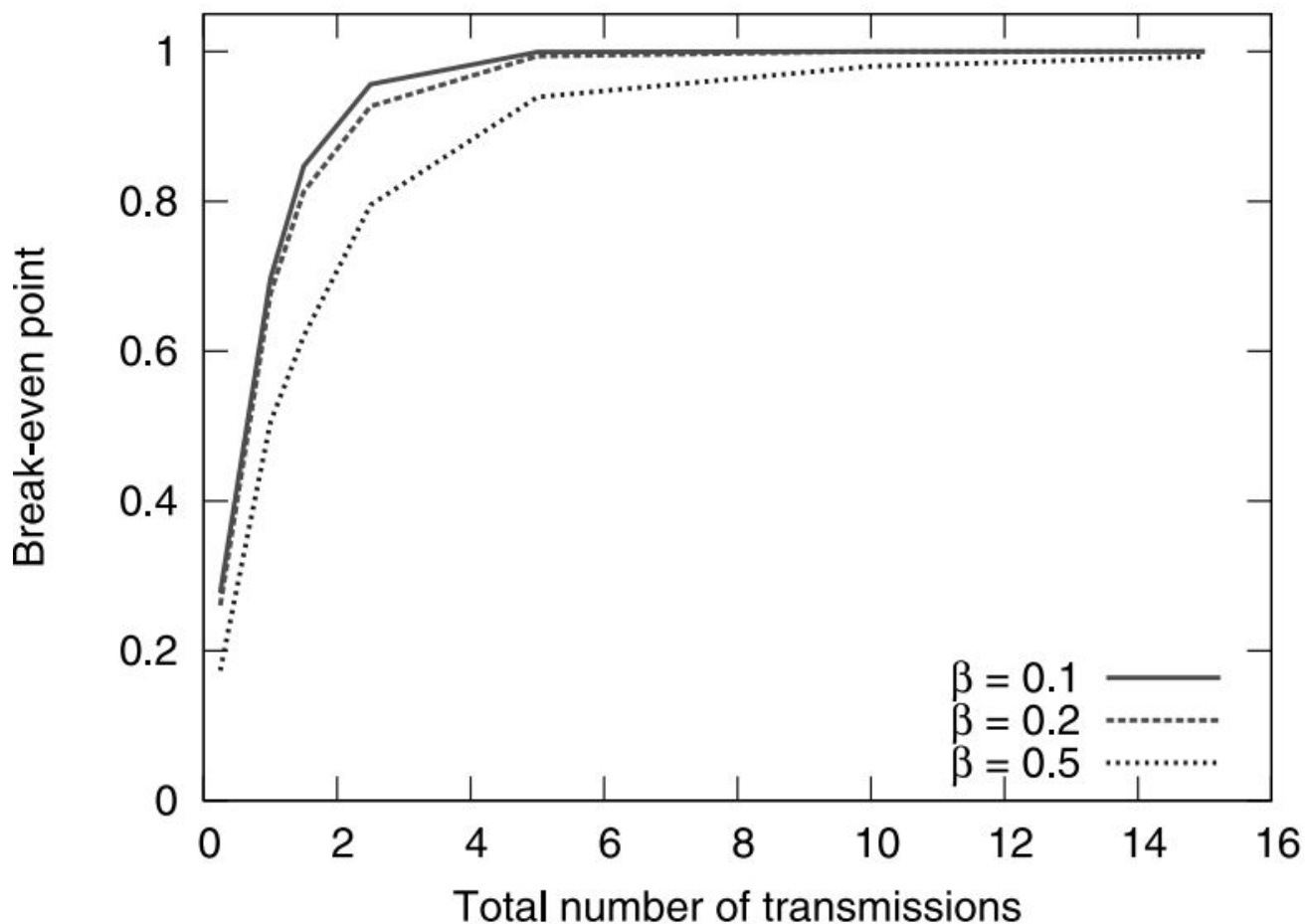
(c) Flat Kronecker (Exp)



(d) Hier. Kronecker (PL)

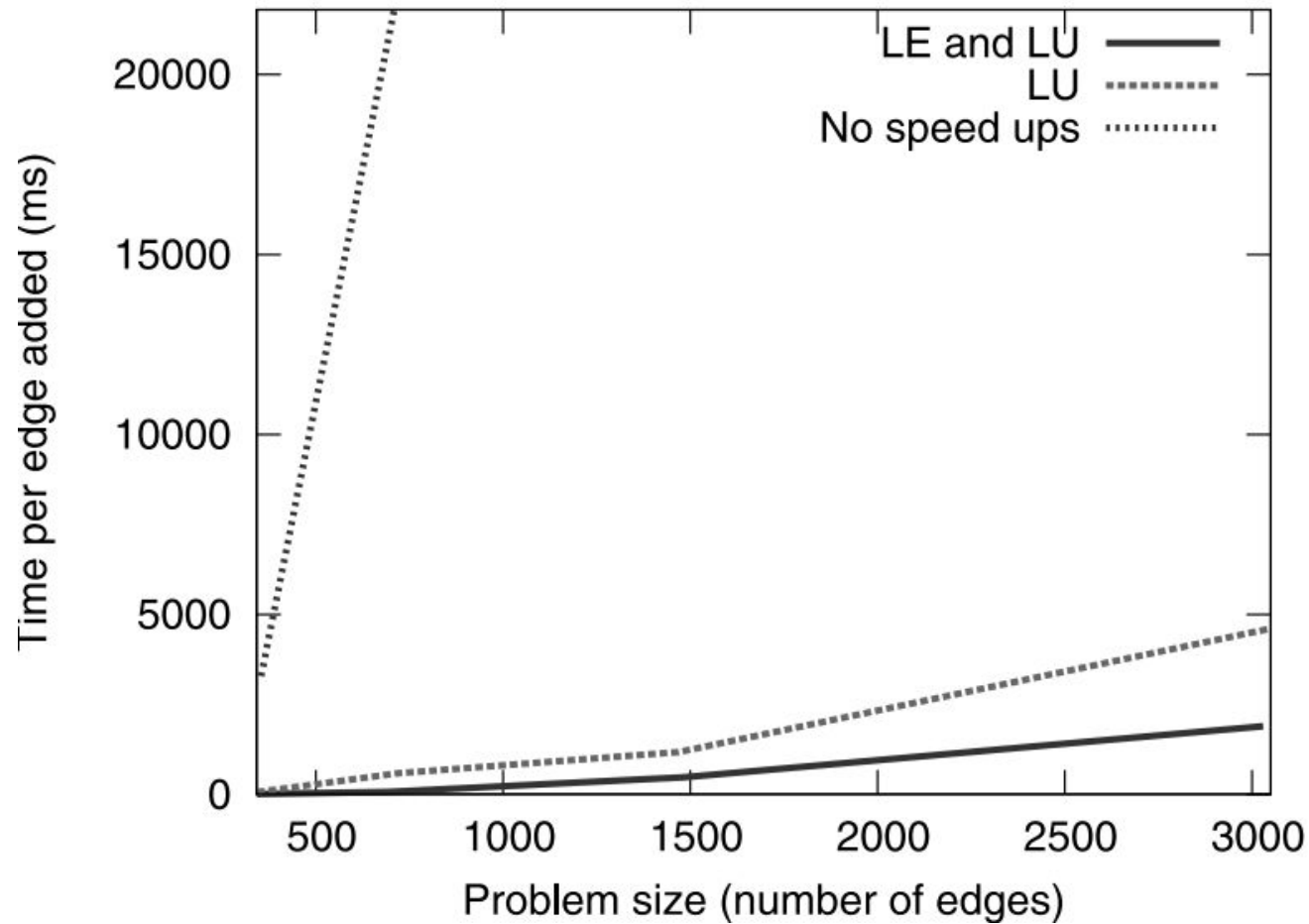
# Experimental Evaluation :

- Performance of NETINF as a function of the amount of cascade data



# Experimental Evaluation :

- Time Cost



# Network Reconstruction under Compressive Sensing

Siyari, P., Rabiee, H. R., Salehi, M., & Mehdiabadi, M. E

Siyari, P., Rabiee, H. R., Salehi, M., & Mehdiabadi, M. E. (2012).  
Network Reconstruction under Compressive Sensing, 130–143.  
<http://doi.org/10.1109/SocialInformatics.2012.84>

# General Idea:

1. Utilizing the **cascade probability data** from the diffusion of an arbitrary type of information throughout the desired networked data;
2. Then **estimate the probability** that a cascade can diffuse over the network;
3. Finally, by **formation of a linear system from the diffusion process**, we utilize the theory of CS in order to reconstruct the network of interest.

# Cascade Modeling :

- The conditional probability of observing cascade  $c$  spreading from  $u$  to  $v$

$$P_c(u, v) = P_c(\Delta_{u,v}) = e^{-\frac{\Delta_{u,v}}{\alpha}}$$

- The likelihood of a cascade spreading in a given tree pattern  $T$

$$P(c|T) = \prod_{u,v \in E_T} \beta P_c(u, v) \prod_{u \in V_T, (u,x) \in E \setminus E_T} (1 - \beta)$$

- The probability that a cascade  $c$  can occur in the graph  $G$

$$P(c|G) = \max_{T \in \tau_c(G)} P(c|T) P(T|G)$$

# Build Linear System :

$$P(c|G) = \max_{T \in \tau_c(G)} P(c|T) P(T|G)$$

taking log from both sides of this equation, we can **approximate** it as the inner product of two vectors:

$$LP(c|G) = vLP(c|T_c^*)^T \cdot \text{vec}(\text{Adj}(G))$$

$$\begin{bmatrix} LP(c_1|G) \\ LP(c_2|G) \\ LP(c_3|G) \\ \vdots \\ LP(c_m|G) \end{bmatrix} = \begin{bmatrix} LP_{c_1}(v_1, v_2) & \dots & LP_{c_1}(v_i, v_j) & \dots & LP_{c_1}(v_n, v_{n-1}) \\ LP_{c_2}(v_1, v_2) & \dots & LP_{c_2}(v_i, v_j) & \dots & LP_{c_2}(v_n, v_{n-1}) \\ LP_{c_3}(v_1, v_2) & \dots & LP_{c_3}(v_i, v_j) & \dots & LP_{c_3}(v_n, v_{n-1}) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ LP_{c_m}(v_1, v_2) & \dots & LP_{c_m}(v_i, v_j) & \dots & LP_{c_m}(v_n, v_{n-1}) \end{bmatrix} \begin{bmatrix} v_{1,2} \\ \vdots \\ v_{i,j} \\ \vdots \\ v_{n,n-1} \end{bmatrix}$$

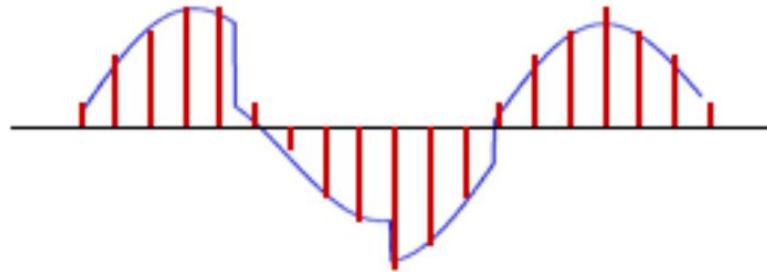


So, what is compressive sensing?

# Classical data acquisition :

- *Shannon-Nyquist sampling theorem* (Fundamental Theorem of DSP):

--- “if you sample at twice the bandwidth, you can perfectly reconstruct the data”



time

# The Sensing Problem :

- Consider sensing mechanisms in which information about a signal  $f(t)$  is obtained by linear functionals recording the values (e.g. MRI) :

$$y_k = \langle f, \varphi_k \rangle, \quad k = 1, \dots, m.$$

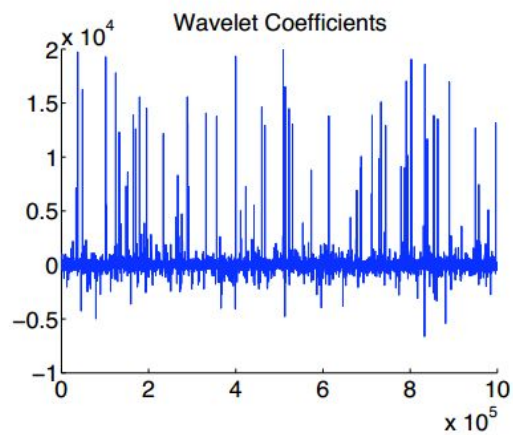
- In under-sampled situations where the number  $m$  of available measurements is much smaller than the dimension  $n$  of the signal  $f$ :
  - Is it possible to design  $m \ll n$  sensing waveforms to capture almost all the information about  $f$  ?
  - Is it possible to reconstruct nearly all the information of  $f$  from sampled data ?

# Two principles :

- **Sparsity** : Compressive Sensing exploits the fact that many natural signals are sparse or compressible in the sense that they have **concise representations** when expressed in the proper basis  $\Psi$ .
- **Incoherence** : Suppose we are given a pair  $(\Phi, \Psi)$  of orthobases of  $R^n$ , The first basis  $\Phi$  is used for sensing the object  $f$  and the second is used to represent  $f$ . A good result often requires a low coherence between  $\Phi$  *and*  $\Psi$ .

# Sparsity :

$$f(t) = \sum_{i=1}^n x_i \psi_i(t)$$



# Incoherent sampling :

- **Incoherence** : Suppose we are given a pair  $(\Phi, \Psi)$  of orthobases of  $R^n$ , The first basis  $\Phi$  is used for sensing the object  $f$  and the second is used to represent  $f$ . A good result often requires a low coherence between  $\Phi$  and  $\Psi$ .

$$\mu(\Phi, \Psi) = \sqrt{n} \cdot \max_{1 \leq k, j \leq n} |\langle \varphi_k, \psi_j \rangle|$$

$$\mu(\Phi, \Psi) \in [1, \sqrt{n}]$$

# Sparse Signal Recovery :

$$\min_{\tilde{x} \in \mathbb{R}^n} \|\tilde{x}\|_{\ell_1} \quad \text{subject to} \quad y_k = \langle \varphi_k, \Psi \tilde{x} \rangle, \quad \forall k \in M$$

When  $f$  is sufficiently sparse, the recovery via  $L1$ -minimization is provably exact.

*Theorem 1* ([E. Candes, 2007]): Fix  $f \in \mathbb{R}^n$  and suppose that the **coefficient sequence  $x$  of  $f$  in the basis  $\Psi$  is  $S$  – sparse**. Select  $m$  measurements in the  $\Phi$  domain uniformly at random. Then if

$$m \geq C \cdot \mu^2(\Phi, \Psi) \cdot S \cdot \log n$$

for some positive constant  $C$ , the solution to the above equation is exact with overwhelming probability.

# Sparse Signal Recovery :

1. The **role of the coherence** is completely transparent; the smaller the coherence, the fewer samples are needed;
2. One suffers no information loss by measuring just about any set of  $m$  coefficients which may be **far less than the signal size apparently demands**.
3. The signal  $f$  can be exactly recovered from our condensed dataset by **minimizing a convex functional**



# Sparse Signal Recovery :

- **Definition 4.1 Restricted Isometry Property (RIP):** For each integer  $s = 1, 2, \dots$ , define the isometry constant  $\delta_s$  of a matrix  $A$  as the smallest number such that :

$$(1 - \delta_s) \|x\|_{\ell_2}^2 \leq \|Ax\|_{\ell_2}^2 \leq (1 + \delta_s) \|x\|_{\ell_2}^2$$

holds for all ***s-sparse*** vectors  $x$ .

- If the RIP holds, then the reconstruction obtained by solving the linear program is accurate.

$$\min_{\tilde{x} \in \mathbb{R}^n} \|\tilde{x}\|_{\ell_1} \quad \text{subject to} \quad A\tilde{x} = y (= Ax)$$

# Robust signal recovery from noisy data:

- We are given noisy data (e.g.  $Y = Ax + z$ ) and use  $L1$  minimization with relaxed constraints for reconstruction:

$$\min \|\check{x}\|_{l_1} \quad \text{subject to} \quad \min \|A\check{x} - y\|_{l_2} \leq \varepsilon$$

where  $\varepsilon$  bounds the amount of **noise** in the data.

- *Theorem 2* ([E. Candes, 2006]): Assume that  $\delta_{2S} < \sqrt{2} - 1$ . Then the solution  $x^*$  obeys

$$\|x^* - x\|_{l_2} \leq C_0 \cdot \|x - x_S\|_{l_1} / \sqrt{S} + C_1 \cdot \varepsilon$$

for some constants  $C_0$  and  $C_1$ .