Algorithmic Problems in Epidemiology

V.S. Anil Kumar

Dept. of Computer Science and
Virginia Bioinformatics Institute
Virginia Tech

Joint work with Stephen Eubank and Madhav Marathe (Virginia Tech),
Aravind Srinivasan and Nan Wang (U. Maryland, College Park)

Email: akumar@vbi.vt.edu
Web: http://ndssl.vbi.vt.edu/

The first Network Approach to Epidemiology

- Made an implicit network between people and water pumps
- People who drank water from Broad St. pump died
- also proved that cholera spreads through water

"Riddle of the Cholera Outbreak" solved by Dr. John Snow

EpiSims/Simdemics[1] - a network approach to epidemiology

Main steps:

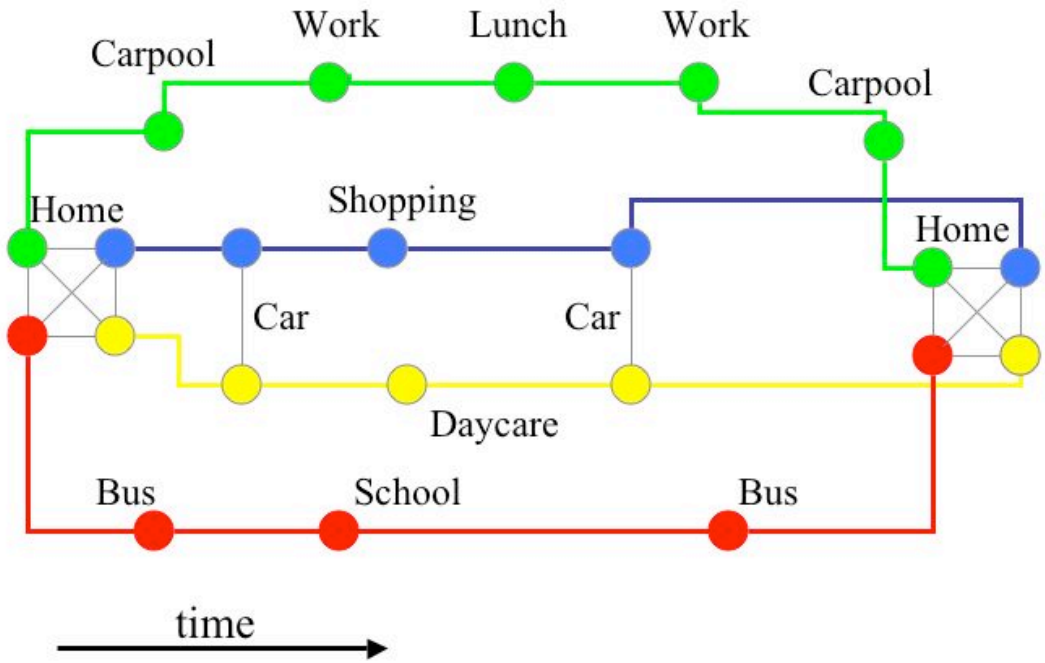
1. Construct a realistic social network
   - Not readily available
2. Develop an efficient disease simulation
   - Must scale to millions of people
   - Static analysis to guide simulation
3. Help in policy questions
   - Abstract out a simpler combinatorial model for the disease dynamics
   - whom to vaccinate/quarantine
   - where to place sensors

[1]S. Eubank et al., http://ndssl.vbi.vt.edu/

Constructing a Social Network

Main steps:

- Synthetic population from census data
- Location Information from census, land use, GIS, etc.
- Individual activities from surveys, traffic data, behavioral models, etc.

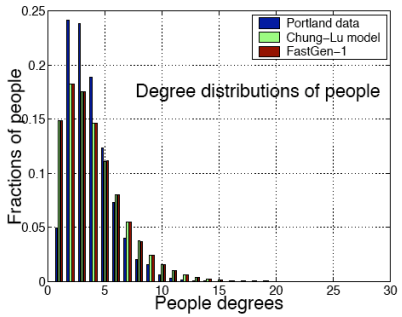


Main Difficulty: Slow and expensive

Can we construct graphs that look "like" real social networks?

Common approach: generative random graph models

The Social Network

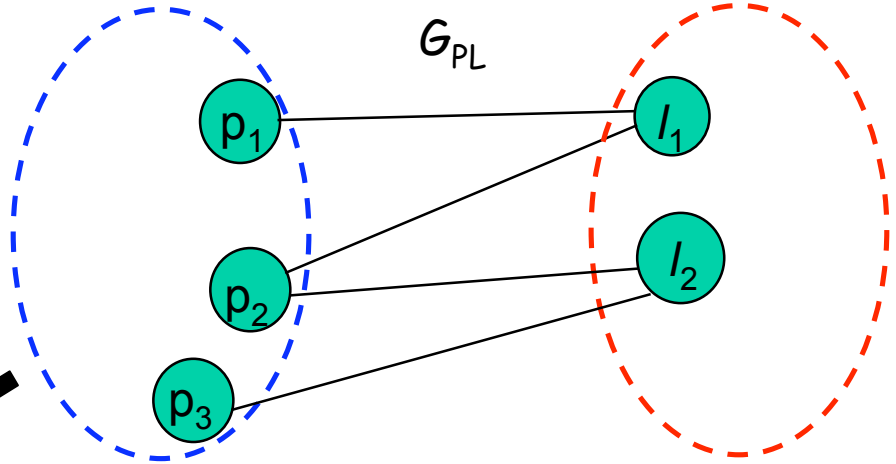


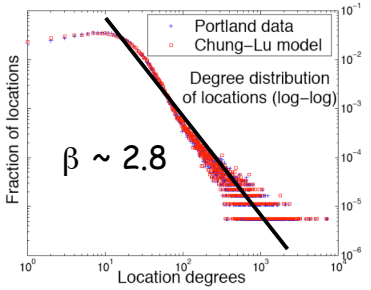
Portland: 1615860 people

people

locations

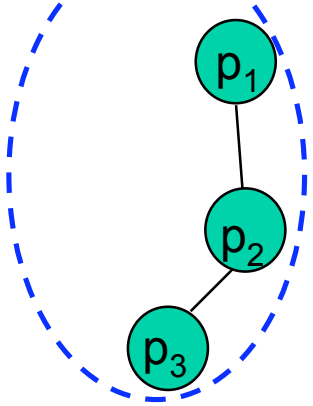
G_{PL}

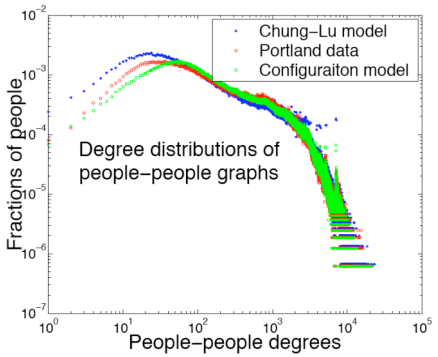




Portland: 181230 locations

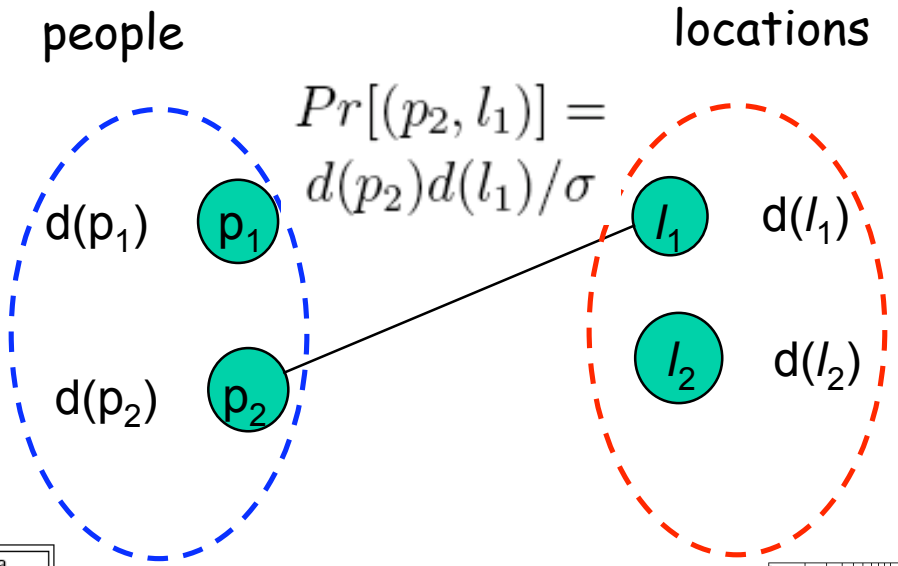
G_P



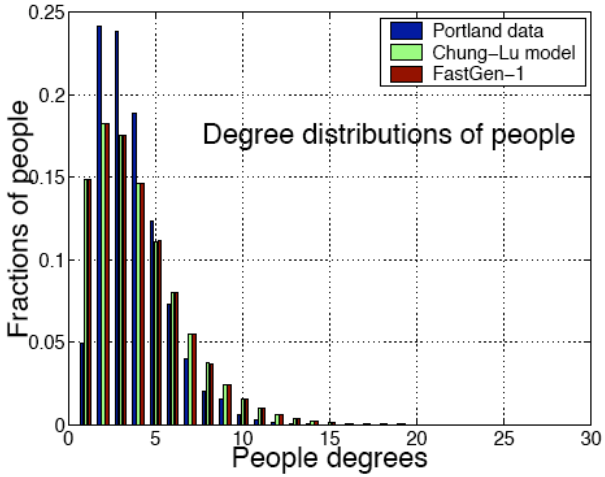


Giant component with 1615813 people

Generative Graph Models: Chung-Lu model



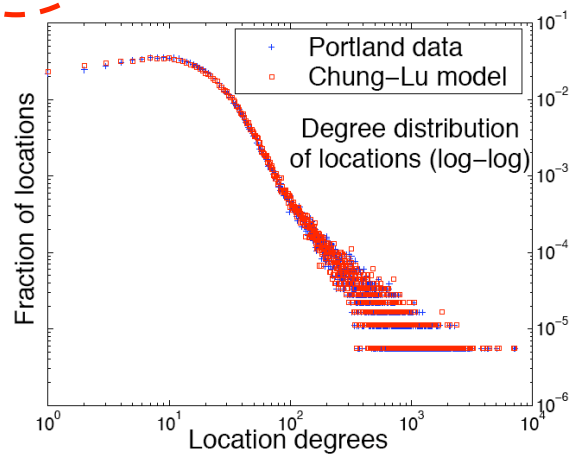
Degree distribution of people



$$\sigma = \sum_{p \in P} d(p) = \sum_{l \in L} d(l)$$

$$\max_{p \in P, l \in L} d(p)d(l) \leq \sigma$$

Degree distribution of locations

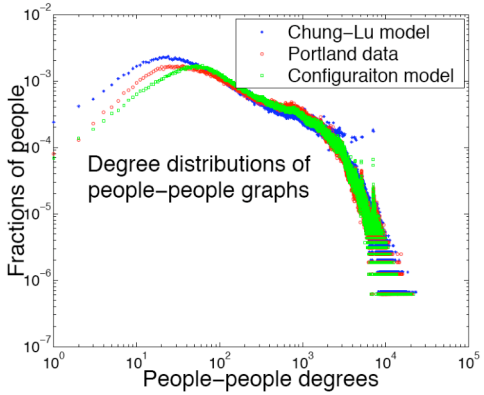


Connectivity in the people-people graph

Assume: d(p)=c, for each p∈ P

$d_0 \leq d(l) \leq d_1$, for each d(l) ∈ L

n_i: #locations with degree i ~ $c'|L|/i^{\beta}$



$$For\ p_1, p_2 \in P:$$

$$\Pr[(p_1,p_2) \notin G_P] = \Pi_{\ell \in L}(1 - \frac{d(p_1)d(p_2)d(\ell)^2}{\sigma^2})$$

$$\leq exp(-\sum_{\ell \in L} \frac{c^2 d(\ell)^2}{\sigma^2})$$

$$= exp(-\frac{1}{|P|^2}\sum_i \frac{i^2 n_i}{i^\beta})$$

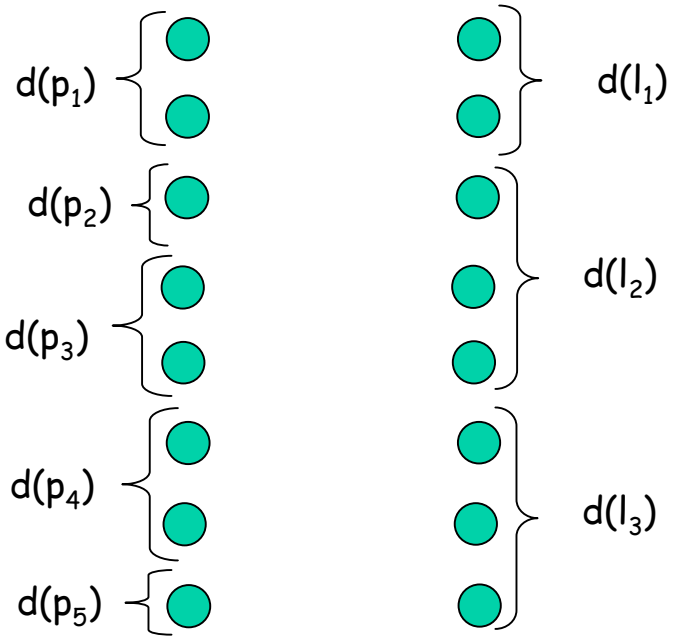
$$\simeq exp(-\frac{c'|L|}{|P|^2}\sum_{i=d_0}^{d_1} \frac{1}{i^{\beta-2}})$$

$$\simeq exp(-\frac{1}{|P|}\frac{c(\beta-2)d_0^{\beta-2}d_1^{3-\beta}}{3-\beta})$$

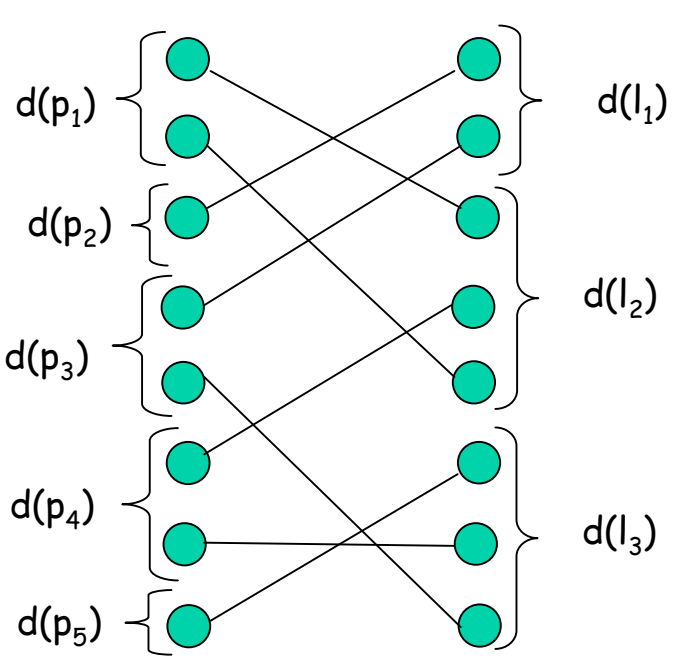
$$\ll exp(-\frac{2}{|P|}) \leq 1 - \frac{1}{|P|}$$

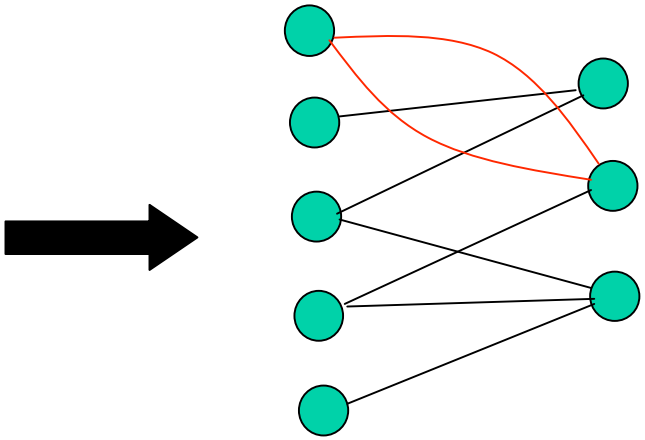
⇒ G_P of Bipartite Chung-Lu has a giant component

The Configuration Model

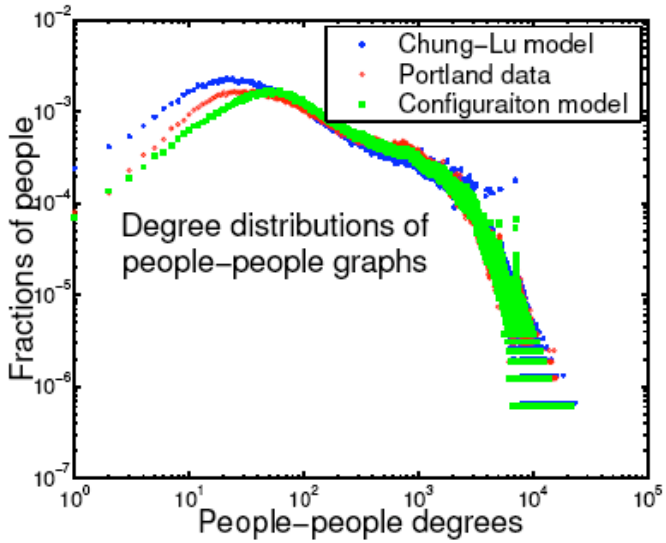


The Configuration Model

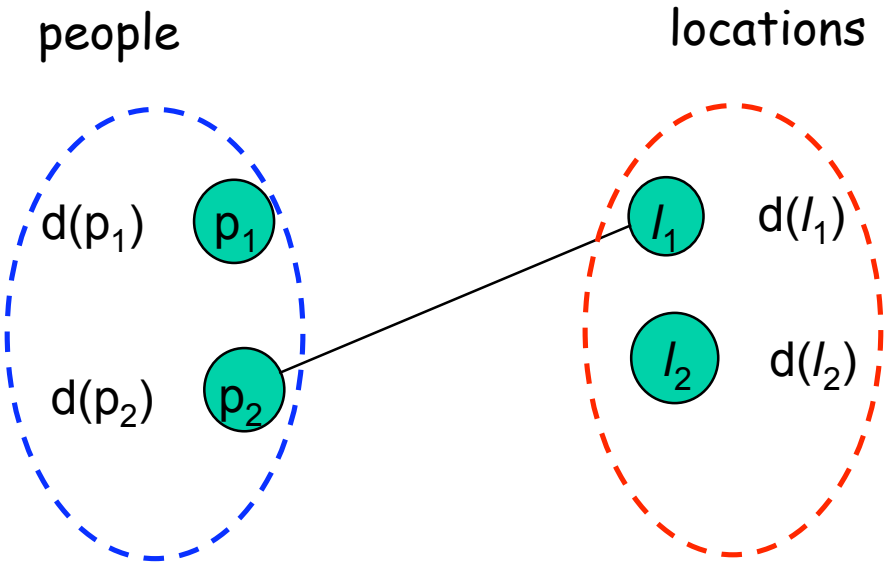




Choose a random matching



Chung-Lu vs Configuration



$$Pr[(p_2, l_1)] \propto d(p_2)d(l_1)/f(dist(p_2, l_1))$$

Candidates

$$f(dist(p_2, l_1)) = dist(p_2, l_1)^{\beta}$$

$$f(dist(p_2, l_1)) = e^{a \cdot dist(p_2, l_1)}$$

Fast Generation of CL

- Speed up (naive CL takes ~ 10 hours) random generation
- Preserve degrees
- Be "similar" to CL

Fast Generation of CL

- Speed up (naive CL takes ~ 10 hours) random generation
- Preserve degrees
- Be "similar" to CL

$$\forall p \in P, \forall \ell \in L, \Pr[X_{p,\ell} = 1] = \frac{d(p) \cdot d(\ell)}{\sigma}$$

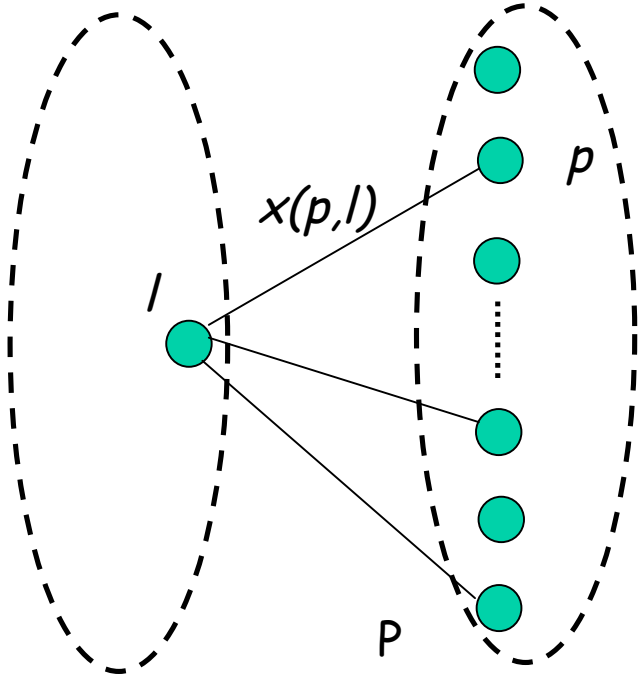
$$\forall \ell \in L, \Pr[|\{p : X_{p,\ell} = 1\}| = d(\ell)] = 1$$

$$\forall \ell \in L, P' \subseteq P :$$

Negative correlation constraints

$$\begin{cases} \Pr[\bigwedge_{p \in P'} (X_{p,\ell} = 0)] \leq \prod_{p \in P'} \Pr[X_{p,\ell} = 0], \\ \Pr[\bigwedge_{p \in P'} (X_{p,\ell} = 1)] \leq \prod_{p \in P'} \Pr[X_{p,\ell} = 1]. \end{cases}$$

Dependent Rounding[1]



Given:

Fractional $x(p,l) \in [0,1]$

Round $x(p,l)$ to $X(p,l) \in \{0,1\}$ s.t.:

- $\Pr[X(p,l)=1] = x(p,l)$
- $\forall\, l \in L$: $\Sigma_p X(p,l) \in \{\lfloor d(l) \rfloor, \lceil d(l) \rceil\}$
- $\forall\, p \in P$: $\Sigma_l X(p,l) \in \{\lfloor d(p) \rfloor, \lceil d(p) \rceil\}$
- $\forall\, P' \subset P$, $\forall\, b(p) \in \{0,1\}$

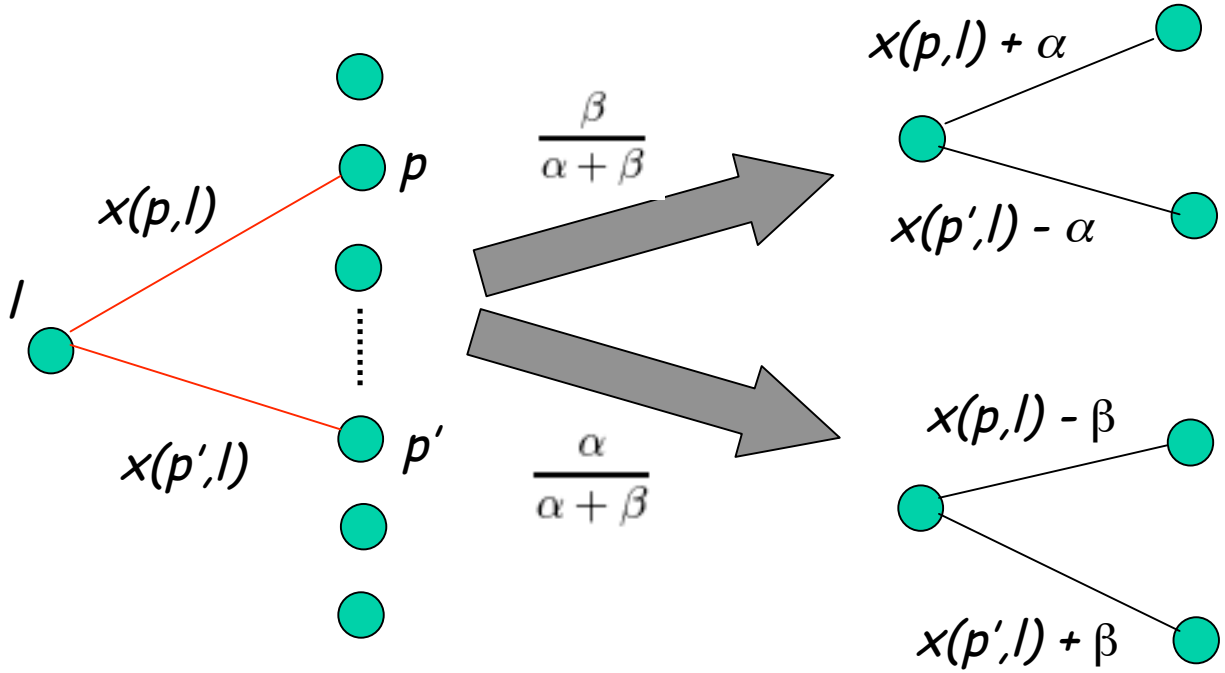
$$\Pr[\wedge_{p \in P'} X(p,l)=b(p)] \leq \Pi_{p \in P'} \Pr[X(p,l)=b(p)]$$

Time taken: *O(# fractional variables x(p,l))*

$\Rightarrow O(|P|\ |L|)$ time overall

[1] R. Gandhi, S. Khuller, S. Parthasarathy, A. Srinivasan, Dependent Rounding, FOCS 2004

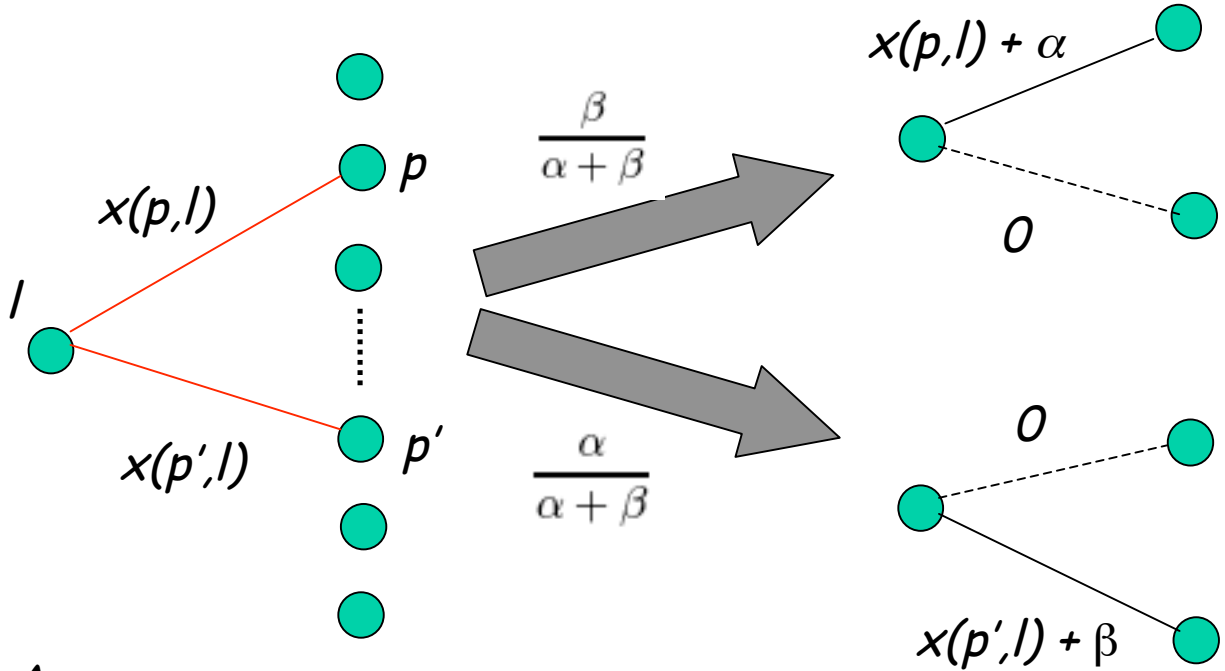
Graph generation by Dependent Rounding



α = min{ 1- x(p,l), x(p',l)}

β = min{ x(p,l), 1- x(p',l)}

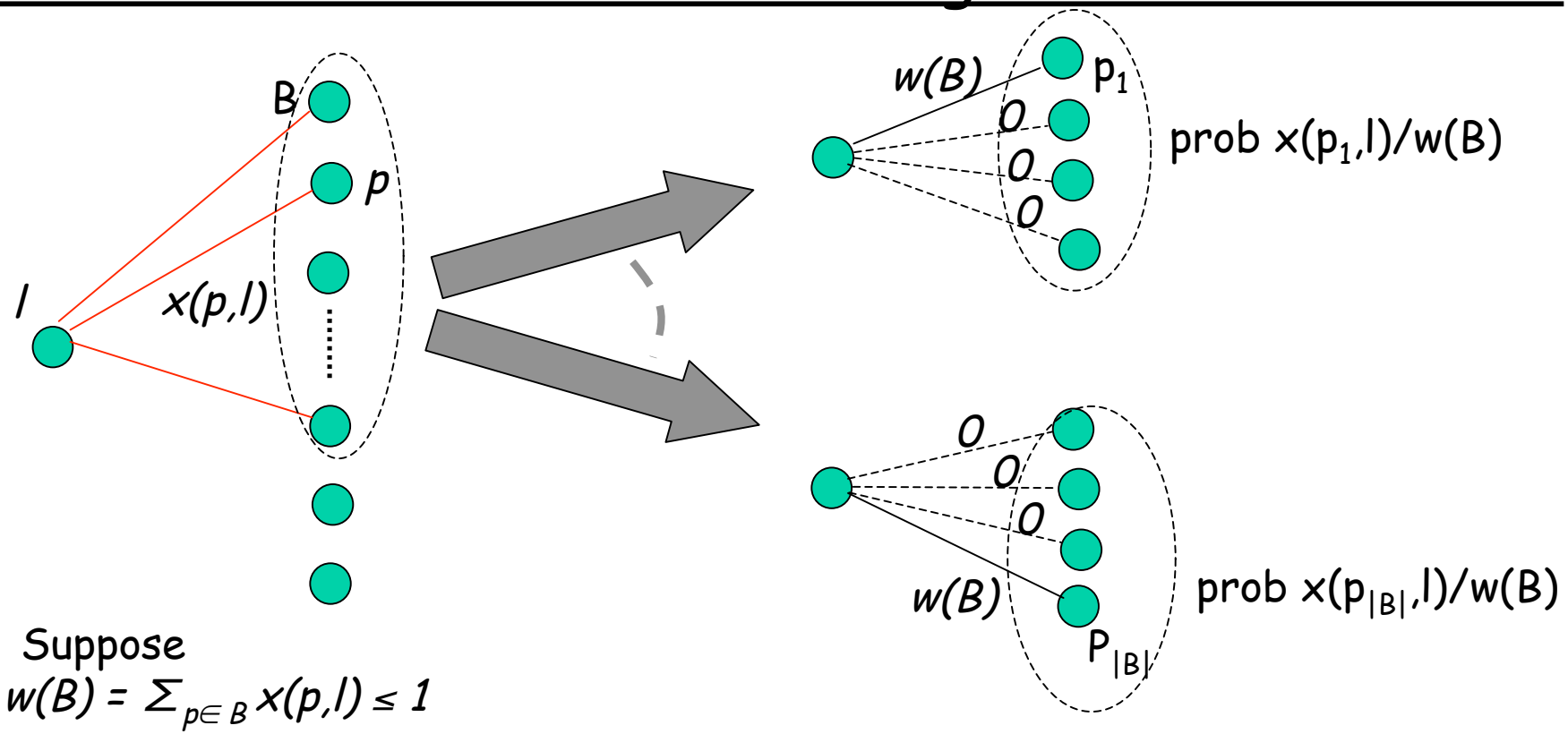
Graph generation by Dependent Rounding



Assume
$x(p,l) + x(p',l) \leq 1$

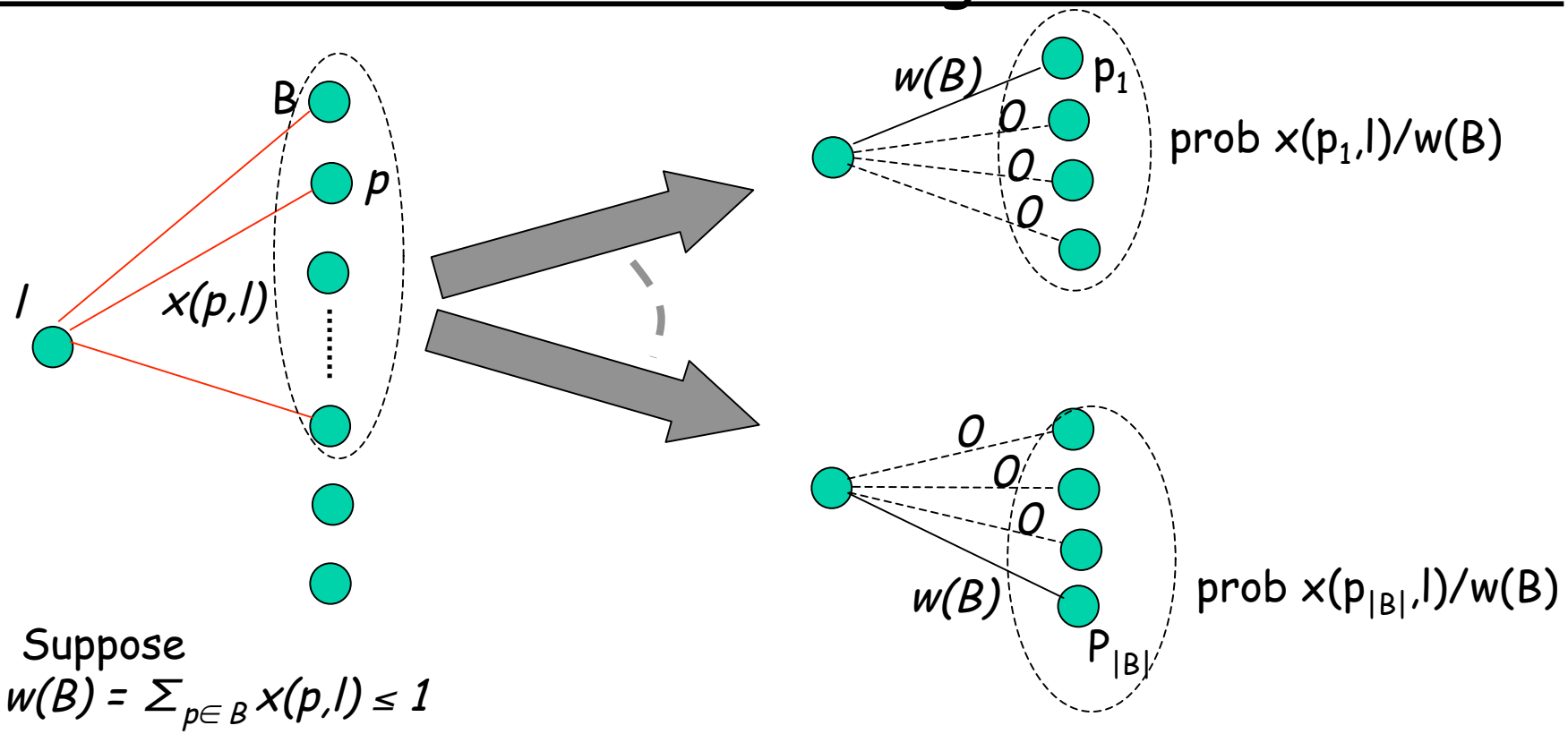
$\alpha = \min\{ 1- x(p,l), x(p',l)\} = x(p',l)$
$\beta = \min\{ x(p,l), 1- x(p',l)\} = x(p,l)$

Graph generation by Dependent Rounding



Suppose
$w(B) = \Sigma_{p \in B} x(p,l) \leq 1$

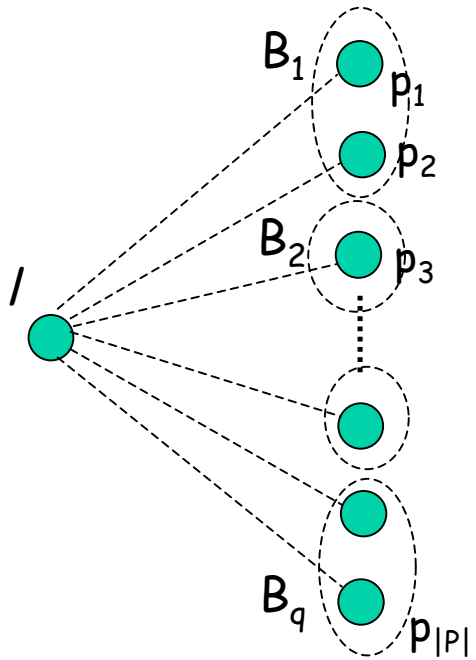
Graph generation by Dependent Rounding



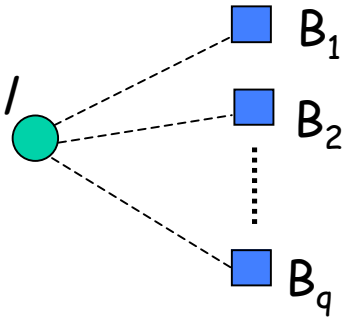
Suppose
$w(B) = \Sigma_{p \in B} x(p,l) \leq 1$

$w(B) \leq 1 \Rightarrow$ dependent rounding equivalent to sampling $p_i \in B$ with probability $x(p_i,l)/w(B)$

Fast Generation of CL



1. Partition into maximal blocks $B_1, B_2, \ldots, B_q$
   - $B_k = \{p(j_k), \ldots, p(j_{k+1})-1\}$
   - $w(B_k) = \Sigma_{j \in Bk}\, d(p_j)\, d(l)/\sigma$
   - $1 - d(p(j_{k+1}))d(l)/\sigma < w(B_k) \leq 1$
2. Reduce to smaller graph H on $\{l, B_1, B_2, \ldots, B_q\}$
3. Run Dependent rounding on H and choose subset $B' \subseteq \{B_1, B_2, \ldots, B_q\}$
4. For each block $B_i \in B'$, choose $p \in B_i$ with prob. $x(p,l) / w(B_i)$



$x(B_i, l) = \Sigma_{p \in Bi}\, x(p,l)$

Fast Generation of CL: Analysis

Observation: q <= 2(d(l) + 1)

Proof:

$\forall k$:

$$w(B_k) + w(B_{k+1}) > 1 - d(p(j_{k+1}))d(\ell)/\sigma + w(B_{k+1})$$
$$= 1 + w(B_{k+1}) - d(p(j_{k+1}))d(\ell)/\sigma$$
$$\geq 1$$

$$\Rightarrow \lfloor \frac{q}{2} \rfloor < \sum_{k=1}^{q} w(B_k)$$

$$\Rightarrow q \leq 2(d(\ell) + 1)$$

Fast Generation of CL: Running Time

$O(|P|\text{polylog})$ time overall

0. Construct prefix sums

$O(q \log |P|)$ time

1. Partition into maximal blocks $B_1, B_2, \ldots, B_q$
   - $B_k = \{p(j_k), \ldots, p(j_{k+1})-1\}$
   - $1 - d(p(j_{k+1}))d(l)/\sigma < w(B_k) \leq 1$

$O(q)$ time

2. Reduce to smaller graph H on $\{l, B_1, B_2, \ldots, B_q\}$

$O(q)$ time

3. Run Dependent rounding on H and choose subset $B' \subseteq \{B_1, B_2, \ldots, B_q\}$

$O(\log |P|)$ time

4. For each block $B_i \in B'$, choose $p \in B_i$ with prob. $x(p,l) / w(B_i)$

$\Rightarrow O(\Sigma\, q(l) \log |P|)$ time overall

$= O(\Sigma d(l) \log |P|) = O(|E| \log |P|)$

Fast Generation Properties

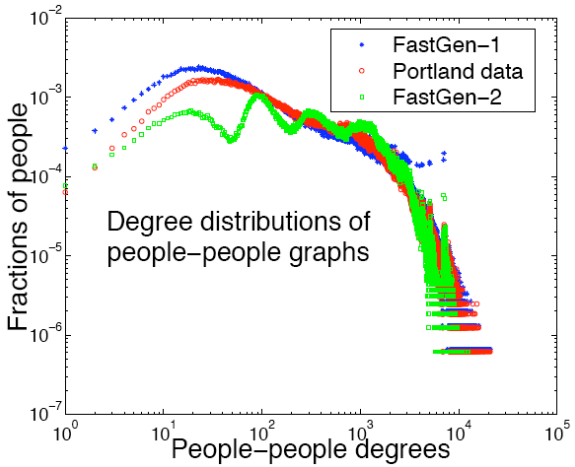
Equivalent to running Dependent Rounding on $G(\{l\}, P, \{(p,l): p \in P\})$ with fractional solution: $x(p,l) = d(p)d(l)/\sigma$

1. Partition into maximal blocks $B_1, B_2, \ldots, B_q$
   - $B_k = \{p(j_k), \ldots, p(j_{k+1})-1\}$
   - $1 - d(p(j_{k+1}))d(l)/\sigma < w(B_k) \leq 1$
2. Reduce to smaller graph H on $\{l, B_1, B_2, \ldots, B_q\}$
3. Run Dependent rounding on H and choose subset $B' \subseteq \{B_1, B_2, \ldots, B_q\}$
4. For each block $B_i \in B$, choose $p \in B_i$ with prob. $x(p,l) / w(B_i)$

$\Rightarrow$ 1. $\Pr[\text{edge } (p,l)] = x(p,l) = d(p)d(l)/\sigma$
2. Rounded-degree(l) = $\Sigma_{p \in P} x(p,l) = d(l)$
3. Negative correlation property

FastGen

	Portland data	CL	FastGen-1	FastGen-2
number of locations	181230	178746 (98.63%)	181230 (100%)	178668 (98.59%)
size of giant(locs.)	181192	178571	181088	178611
number of people	1615860	1507234 (93.28%)	1507291 (93.28%)	1615860 (100%)
size of giant(ppl.)	1615813	1507054	1507148	1615803
number of edges	6060679	6065637 (100.08%)	6060679 (100%)	6060679 (100%)
average deg. (ppl.)	3.7507	4.0227	4.0209	3.7507
time of generating a graph		$>$ 10 hours	$<$ 40 seconds	$<$ 30 seconds



	Portland data	CL	FastGen-1	FastGen-2
c.c. mean	0.6376	0.6161	0.6235	0.7021
c.c. std.	0.0167	0.0315	0.0236	0.0201
c.c l.b.	0.6286	0.6220	0.6241	0.7061
c.c. a.l.b.	0.6262	0.6220	0.6240	0.7061

A network approach to epidemiology

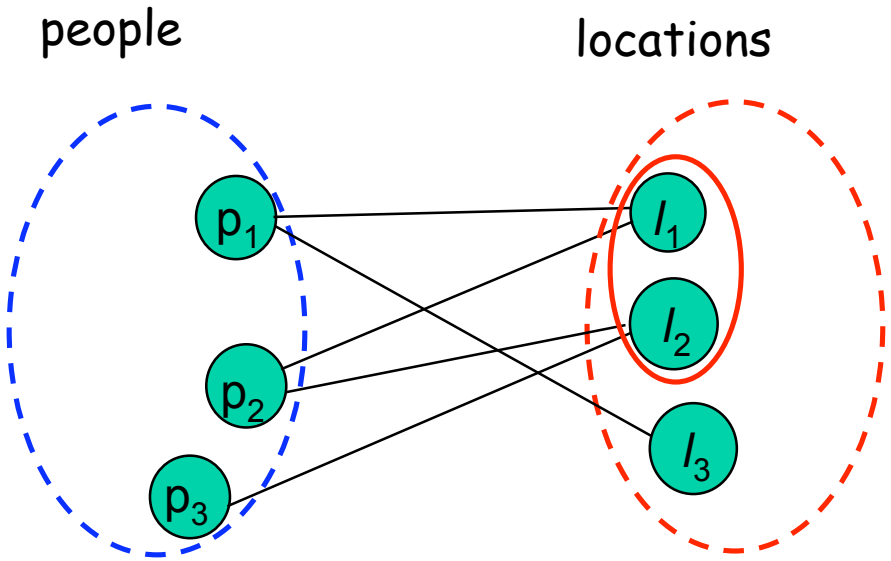
Main steps:

1. Construct a realistic social network
   - Not readily available
2. Develop an efficient disease simulation
   - Must scale to millions of people
   - Static analysis to guide simulation
3. Help in policy questions
   - Abstract out a simpler combinatorial model for the disease dynamics
   - whom to vaccinate/quarantine
   - where to place sensors

Policy Planning problems

- Disease detection
  - Fast detection + response, instead of mass vaccination
  - Sensor Location: dominating set problem
- Quarantining
  - Remove some edges so that disease is contained
- Vaccination
  - Remove some nodes so that disease is contained

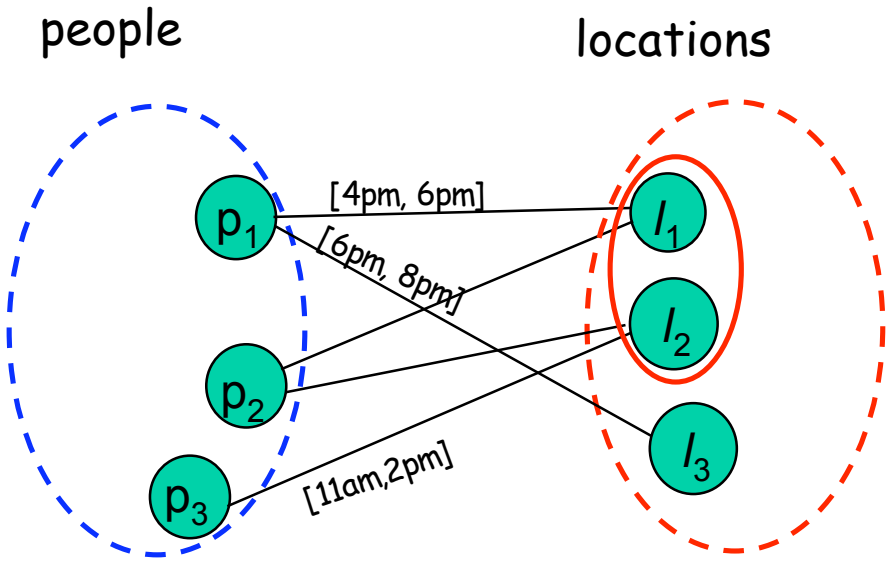
The Sensor Placement Problem



Dominating set problem: choose $L' \subseteq L$ s.t. $N(L') = P$

(1-ε)-Dominating set problem: choose $L' \subseteq L$ s.t. $|N(L')| \geq (1-\varepsilon)|P|$

Temporal Dominating Set



I(p, l): interval during which p visits l

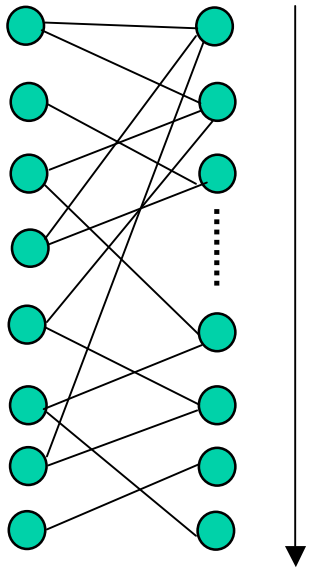
The Temporal Dominating set problem: choose $L' \subseteq L$, and time interval $[t_1, t_2]$ for each $l \in L'$ s.t. $\forall\, p \in P, \exists\, l \in L'$ s.t. $I(p,l) \cap [t_1, t_2] \neq \emptyset$

Sensor Placement: Results

- FastGreedy: choose large locations in non-increasing order of degrees sum of degrees is $(1-\varepsilon')|P|$
- FastGreedy gives (1+o(1))-approximation to optimum dominating set in Chung-Lu model
  - Same approximation in FastGeneration model
- FastGreedy works well in practice
  - 10% of the locations can dominate ~90% of people in Portland network
  - Very close to Greedy
  - Takes ~15 sec
- Temporal version hard to approximate within $\Omega(n^{\varepsilon})$

Dominating Set in the Chung-Lu model

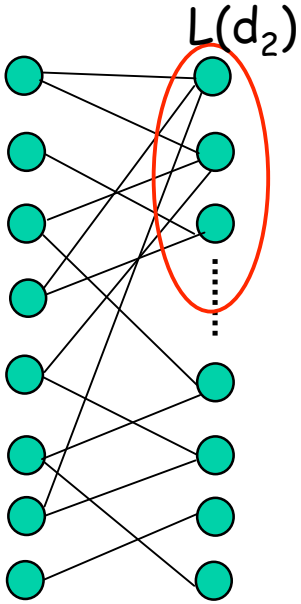
Theorem: FastGreedy is a (1+o(1))-approximation to the (1-ε)-Dominating Set in the Chung-Lu and Fast Generation models



Non-increasing
order of degrees

Dominating Set in the Chung-Lu model

Theorem: FastGreedy is a (1+o(1))-approximation to the (1-ε)-Dominating Set in the Chung-Lu and Fast Generation models



$L(d) = \{\text{locations } l \text{ with } d(l) \geq d\}$

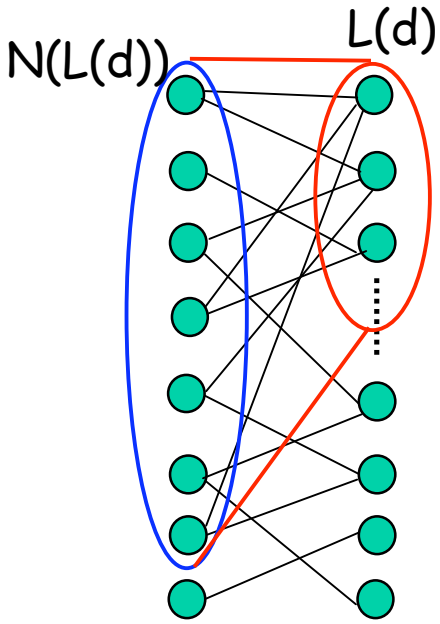
$S(L') = \Sigma_{l \in L'} d(l)$

d_2: largest degree s.t. $S(L(d_2)) \geq (1-\varepsilon')|P|$

To prove:

1. $|N(L(d_2))| \geq (1-\varepsilon)|P|$
2. Approximation ratio = $|L(d_2)|/|OPT| = 1+o(1)$

Property 1



Lemma: $S(L') \geq |P| \ln \frac{1-\delta_1}{\epsilon_1} \Rightarrow |N(L')| \geq (1-\epsilon_1-\delta_1)|P|, w.h.p.$

Proof:

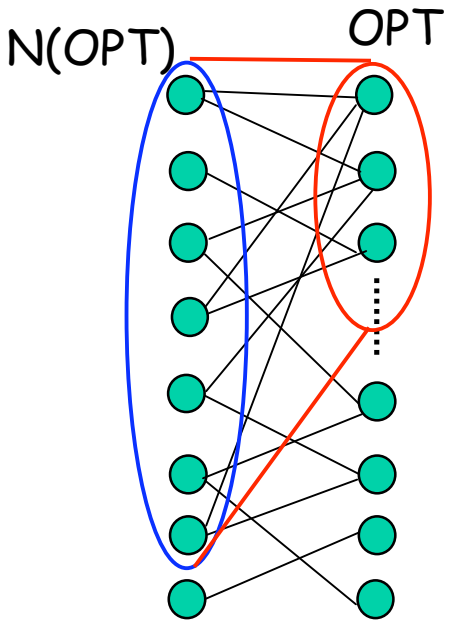
$$\forall p \in P, \Pr[p \notin N(L')] = \Pi_{\ell \in L'}(1 - \frac{d(p)d(\ell)}{\sigma})$$
$$< e^{-S(L')/|P|}$$
$$\Rightarrow E[|P \setminus N(L')|] < |P| e^{-S(L')/|P|}$$

$$S(L') \geq |P| \ln \frac{1-\delta_1}{\epsilon_1} \Rightarrow E[|P \setminus N(L')|] < |P| \frac{\epsilon_1}{1-\delta_1}$$
$$\Rightarrow E[|N(L')|] > |P|(1 - \frac{\epsilon_1}{1-\delta_1})$$
$$\Rightarrow |N(L')| > (1-\epsilon_1-\delta_1)|P| \quad w.h.p.$$

Chernoff bound, also holds for Fast Generation

⇒ Fast Greedy gives a (1-ε)-Dominating Set

Property 2



Lemma: $N(L') \geq (1-\epsilon)|P| \;\Rightarrow\; S(L') \geq |P|\frac{\ln \frac{2-\epsilon}{2\epsilon}}{1+\gamma}, \quad w.h.p.$

$\therefore$ S(OPT) is large

d_3 = largest degree s.t. $S(L(d_3)) \leq |P|\frac{\ln \frac{2-\epsilon}{2\epsilon}}{1+\gamma}$

$\Rightarrow |OPT| \geq |L(d_3)|$

Approximation factor of FastGreedy

Approximation factor $\leq \frac{|L(d_2)|}{|L(d_3)|}$

Power law $\Rightarrow$ $S(L(d)) \simeq \frac{c'|L|}{(\beta-2)d^{\beta-2}}$

$\therefore$ Approximation factor $\leq (\frac{S(L(d_2))}{S(L(d_3))})^{\frac{\beta-1}{\beta-2}}$

$\approx$ 1+o(1), w.h.p.

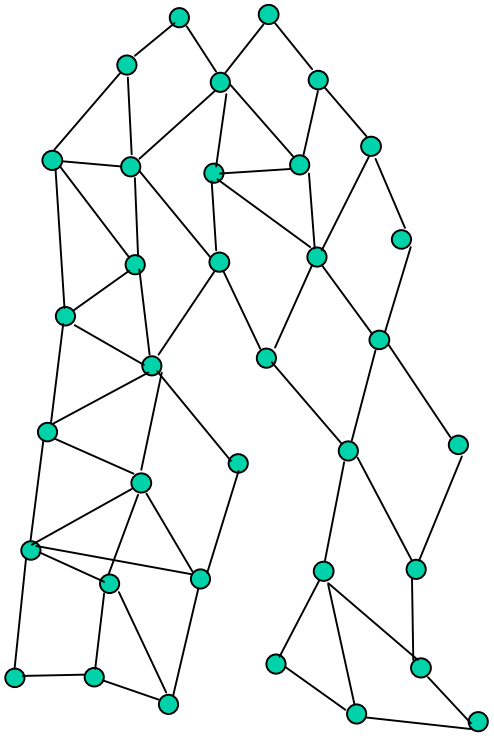
Policy Planning problems

- Disease detection
  - Fast detection + response, instead of mass vaccination
  - Sensor Location: dominating set problem
- Quarantining
  - Remove some edges so that disease is contained
- Vaccination
  - Remove some nodes so that disease is contained

Policy problems: whom to vaccinate

Given

- a social network G(P,E)
- initial infected set A
- budget B on # vaccinations



Policy problems: whom to vaccinate

Given
- a social network G(P,E)
- initial infected set A
- budget B on # vaccinations

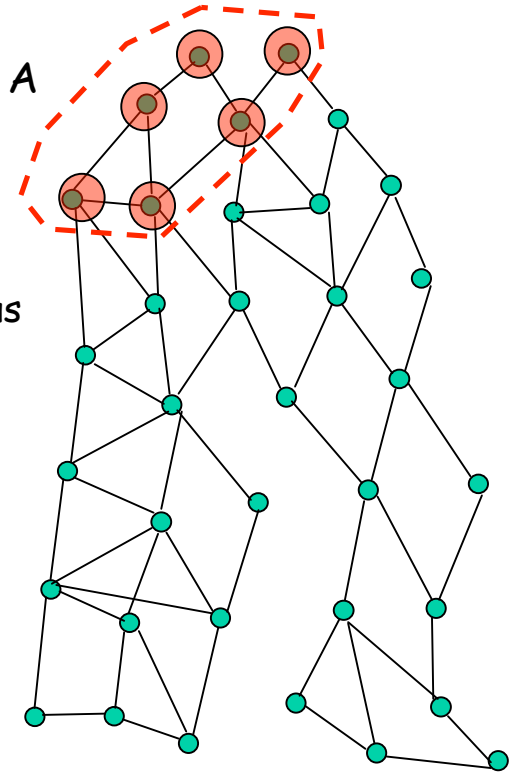
Goal

choose $S \subseteq P$ to vaccinate so that $|S| \leq B$, and # nodes reachable from A in G[P\S] is minimized

B=4

Assume: highly infectious disease

⇒ any node with a path to A gets infected

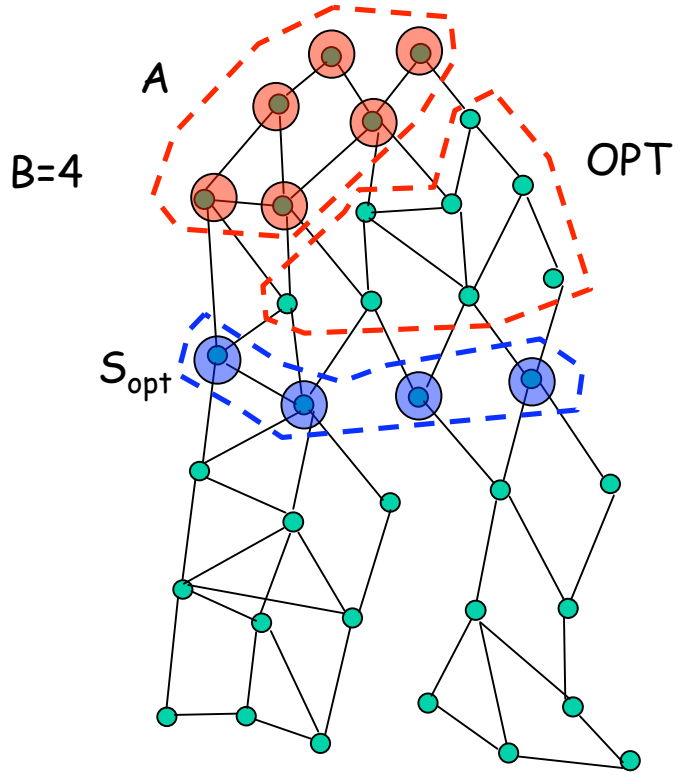


Policy problems: whom to vaccinate

Given
- a social network G(P,E)
- initial infected set *A*
- budget B on # vaccinations

Goal
choose $S \subseteq P$ to vaccinate so that $|S| \leq B$, and # nodes reachable from *A* in G[P\S] is minimized

Result
Bicriteria approximation: Vaccinate $(1+\varepsilon)B$ nodes so that at most $(1+1/\varepsilon)$ OPT infected people[1,2]



[1]S. Eubank, V.S. Anil Kumar, M. Marathe, A. Srinivasan and N. Wang, AMS-DIMACS special volume, 2005
[2]A. Hyrapetyan, D. Kempe, M. Pal and Z. Svitkina, ESA 2005

The Vaccination Problem

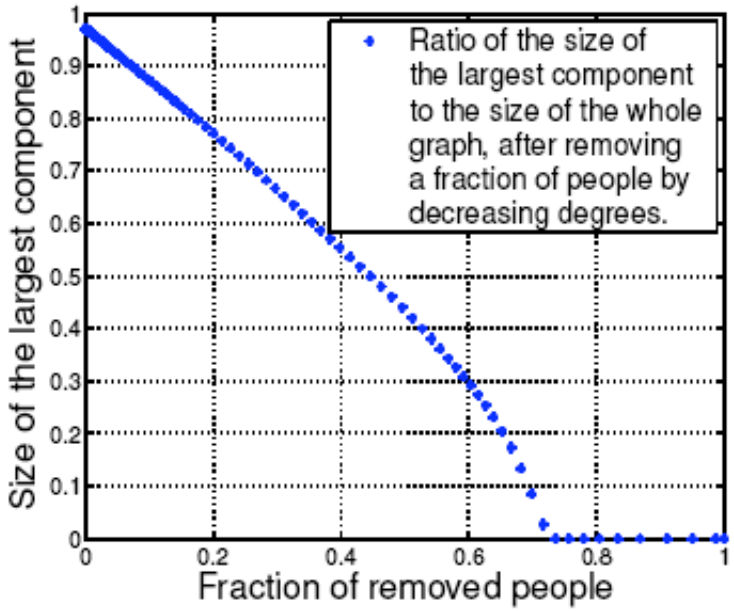
In reality: initial infected set is not known

Goal: Choose $S \subseteq P$, s.t. every component in $G[P \setminus S]$ is small

ρ-separator problem: polylog approximations known, but algorithms are not scalable

The "High Degree" Vaccination Policy

Vaccinate people of high degrees



Giant component remains after deleting a large fraction of nodes

Also true for G_P in the Chung-Lu and FastGen models

Open Problems

- Social Networks
  - Models for temporal and demographic aspects
  - Distance function
- Other network design problems for policy planning
  - Remove nodes/edges so that expansion is reduced
  - Remove nodes/edges so that average distance increases
  - Faster algorithms on large graphs
- More realistic disease models (SIR, SIS)
  - Dynamics on social networks
  - Policy planning problems
- Problems on temporal graphs

Thank You