The Structure of Social Contact Graphs and their impact on Epidemics

Anil Vullikanti
Network Dynamics and Simulation Science Laboratory
Virginia Bioinformatics Institute, and Dept. of Computer Science, Virginia Tech

Joint work with: Chris Barrett, Madhav Marathe, Stephen Eubank, Dick Beckman, Jiangzhou Chen, Xizhou Feng, Maleq Khan, Keith Bisset, Zhao Zhao, Tridib Dutta
Understanding disease dynamics: key questions

- New outbreak
  - Characteristics of total outbreak size and peak
  - Will it become an epidemic?
- Who is likely to get infected?
- Design effective interventions to detect and control epidemics
Effect of graph structure

- Graph structure matters!
  - Disease dynamics depend on properties of graph structure

- Common *local* measures (e.g., degree, clustering distributions) or *global* but *static* measures (e.g., centrality) not very effective
  - Graphs with same properties w.r.t. these measures but varying disease dynamics
  - Individual epidemic characteristics not captured by these measures

- Main challenges
  - Lack of good social contact network models at large scale
  - Computational issues: efficient simulations
This talk

- **Vulnerability** measure for characterizing disease dynamics
  - Form of stochastic centrality
  - Better insights about disease spread
- Computationally difficult
  - Efficient sequential and parallel methods to compute it on large social contact graphs
- No correlations with *static* graph measures

- Applications:
  - Effective strategy for vaccination: use high vulnerability nodes
  - Understanding likelihood of an epidemic
Outline

- Synthetic social contact graphs
- SIR model for epidemics
- Vulnerability: definition and basic properties
- Fast algorithms for disease simulation and computing vulnerability
- Correlations with other graph measures
- Applications
Focus: social contact graphs

- Difficult to construct real contact graphs
  - Privacy/security issues
  - Dynamic networks
  - Data sets for small populations, e.g., [Meyers et al., 2006]

- Our focus: synthetic social contact graphs
  - Constructed by integrating a number of public and commercial data sets
  - Statistically similar to realistic contact networks
  - Significantly different from other complex networks
Synthetic Contact Graphs

More than 12 public and commercial data sets

Data sets, documentation at: ndssl.vbi.vt.edu
Some structural properties of these graphs

- Different from other complex networks
- Clique coefficient
Labeled graph properties

- Rich node and edge labeled structure
- Significant variation in individual properties with node/edge labels
Beyond degree distributions

Edge flip chain

Preserves degree distribution

Edge flips change disease dynamics
Outline

- Synthetic social contact graphs
- SIR model for epidemics
- Vulnerability: definition and basic properties
- Fast algorithms for disease simulation and computing vulnerability
- Correlations with other graph measures
- Applications
Epidemics on networks

Nodes: people
Properties: demographics, immunity

Edges: contacts between people
Properties: duration, nature of contact

Communicating FSM model
Transmission probabilities depend on states of neighbors

\[ p(u,v): \text{transmission prob. on edge } (u,v) \]
Example: SIR process on a network

\[ \text{Probability} = p(1,3)(1-p(1,2)) \]

\[ \text{Probability} = (1-p(1,2))p(3,2)(1-p(3,4)) \]
Outline

- Synthetic social contact graphs
- SIR model for epidemics
- Vulnerability: definition and basic properties
- Fast algorithms for disease simulation and computing vulnerability
- Correlations with other graph measures
- Applications
The *Vulnerability* measure

\[ V(i) = \text{Vulnerability of a node } i = \text{probability of getting infected, if the disease starts at a random node} \]

- Depends on
  - Initial conditions
  - Transmission probability
  - Network structure - not a first order property

Temporal version: probability of infection in specific duration

\[ V(i, t) = \text{Vulnerability of a node } i \text{ at time } t = \text{probability of getting infected during the first } t \text{ time steps} \]
Vulnerability based rank order: Dependence on transmission probability

Ordering can change in specific graphs
- small $p$: $V(a) < V(b)$
- large $p$: $V(a) > V(b)$
Vulnerability based rank order: Dependence on transmission probability

Ordering can change in specific graphs
- small $p$: $V(a)<V(b)$
- large $p$: $V(a)>V(b)$

Ordering relatively stable in Portland social contact network for different transmission probabilities
Vulnerability based rank order: Dependence on initial conditions

Low transmission probability
- few initial infections: $V(a) < V(b)$
- many initial infections: $V(a) > V(b)$
Vulnerability based rank order: Dependence on initial conditions

Low transmission probability
- few initial infections: $V(a) < V(b)$
- many initial infections: $V(a) > V(b)$

Ordering relatively stable in Portland contact graph with different initial infections
Outline

- Synthetic social contact graphs
- SIR model for epidemics
- Vulnerability: definition and basic properties
- Fast algorithms for disease simulation and computing vulnerability
- Correlations with other graph measures
- Applications
Computing vulnerability

- Monte-carlo samples: each sample by an epidemic simulation tool
- $V_k(i)$: probability node $i$ gets infected in $k$ iterations
- $R(\infty)$: top $n$ nodes in vulnerability order, $V(i)$
- $R(t)$: top $n$ nodes in temporal vulnerability order $V(i,t)$
EpiFast: sequential version

Graph $G$ and disease model

Choose random length $l(e)$ of each edge $e$ of $G$

Compute distances from $s$ in w.r.t. length function $l()$

One random run of simulation

random length for edge $e= (u,v)$:

\[
\ell(e) = \begin{cases} 
  i \in \{1, \ldots, S(u)\}, & \text{with probability } (1 - p(e))^{i-1}p(e); \\
  \infty, & \text{with probability } (1 - p(e))^{S(u)}. 
\end{cases}
\]

\[
V_t = \{v : \text{dist}_\ell(s, v) = t\}
\]

\[
I = \{e = (u,v) : \ell(e) = \text{dist}_\ell(s, v) - \text{dist}_\ell(s, u)\}
\]

$p(e) = \text{transmission prob. on edge } e$

$S(u) = \text{infectious duration of node } u$
Example

Theorem EpiFast produces each possible random disease trajectory $R$ with the same probability.

$$I = \{(1,3), (3,2)\}$$

$$P_s[R] = p^2 (1-p)^6$$
EpiFast: parallel version

- C++/MPI implementation, tested on commodity clusters and SGI Altix systems.
- Los Angeles population: 16 million people.
  - 180 days of epidemic duration.
  - With and without interventions.
  - 25 replicates for each configuration.
  - Each replicate takes < 15 minutes.

EpiFast: scaling

<table>
<thead>
<tr>
<th>Population</th>
<th>Population Size</th>
<th>CPU Number</th>
<th>Running Time (seconds) per simulation day</th>
</tr>
</thead>
<tbody>
<tr>
<td>Miami</td>
<td>2.09</td>
<td>32</td>
<td>0.47</td>
</tr>
<tr>
<td>Boston</td>
<td>4.15</td>
<td>64</td>
<td>0.54</td>
</tr>
<tr>
<td>Chicago</td>
<td>9.05</td>
<td>128</td>
<td>0.54</td>
</tr>
</tbody>
</table>
Outline

- Synthetic social contact graphs
- SIR model for epidemics
- Vulnerability: definition and basic properties
- Fast algorithms for disease simulation and computing vulnerability
- Correlations with other graph measures
- Applications
Correlation with static graph measures

Very little information from static graph measures
Correlations with labels

- Similar correlations at different transmission probabilities
- Need better models for individual activities and contact duration
Outline

- Synthetic social contact graphs
- SIR model for epidemics
- Vulnerability: definition and basic properties
- Fast algorithms for disease simulation and computing vulnerability
- Correlations with other graph measures
- Applications
Application: Vaccination based on vulnerability rank order

- Contact graph on Chicago, ~ 8 million people
- Highly vulnerable nodes are also most critical for this network
Application: which outbreaks take off?

Epidemic starting at higher vulnerability node is more likely to result in an outbreak
Conclusions

- **Vulnerability measure**
  - Useful for understanding disease propagation
  - Not first order properties
  - Not well correlated with other “standard” graph measures
  - Computationally intensive: efficient sequential and parallel algorithms
  - Need for good graph modeling