Netw Spat Econ (2013) 13:399–426
DOI 10.1007/s11067-013-9186-6

# Inferring Contagion Patterns in Social Contact Networks with Limited Infection Data

**David Fajardo · Lauren M. Gardner**

Published online: 19 June 2013

**Abstract** The spread of infectious disease is an inherently stochastic process. As such, real time control and prediction methods present a significant challenge. For diseases which spread through direct human interaction, (e.g., transferred from infected to susceptible individuals) the contagion process can be modeled on a social-contact network where individuals are represented as nodes, and contacts between individuals are represented as links. The model presented in this paper seeks to identify the infection pattern which depicts the current state of an ongoing outbreak. This is accomplished by inferring the most likely paths of infection through a contact network under the assumption of partially available infection data. The problem is formulated as a bi-linear integer program, and heuristic solution methods are developed based on sub-problems which can be solved much more efficiently. The heuristic performance is presented for a range of randomly generated networks and different levels of information. The model results, which include the most likely set of infection spreading contacts, can be used to provide insight into future epidemic outbreak patterns, and aid in the development of intervention strategies.

**Keywords** Contagion · Social-contact networks · Optimization

## 1 Introduction

Many factors contribute to the contagion process of infectious disease, such as demographic characteristics, population density, infection prevention practices (e.g.,

D. Fajardo
CE 113 School of Civil and Environmental Engineering, The University of New South Wales,
Sydney, NSW 2052, Australia
e-mail: d.fajardo@unsw.edu.au

L. M. Gardner (✉)
CE 112 School of Civil and Environmental Engineering, The University of New South Wales,
Sydney, NSW 2052, Australia
e-mail: l.gardner@unsw.edu.au

vaccination), local programs (e.g., health, emergency response), and also critically significant, the interaction patterns among individuals. Today, a large proportion of the population lives in increasingly dense conditions, an ideal environment for rapid disease transmission.

The stochastic nature of the contagion process (i.e., contact between an infectious and susceptible person may or may not result in a new infection) makes it difficult to identify the path of infection or predict the impact that a new disease might have on a region. Over the last 100 years, significant research efforts have focused on predicting the expected spreading behavior of contact-based infectious diseases, exploiting characteristics of the population and the disease itself. However, there have been limited research efforts focusing on the use of future social network data: while current social network models are abstract constructs where people are anonymously represented as nodes, it is not unreasonable to expect developments in data collection (through Facebook, Twitter, Foursquare, etc.) which will allow accurate mappings between known individuals.

Spatial analysis of networks, such as transport and communication networks, is a growing area of research (Gastner and Newman 2006; Schintler et al. 2007; Erath et al. 2009), and has recently been expanded to include social network modeling, specifically the ability to reproduce spatial structure and interaction between individuals for large-scale social networks (Illenberger et al. 2012). Furthermore, the ongoing development of activity-based travel models, which examine why, where and when various activities are engaged in by individuals (Lam and Huang 2003; Roorda et al. 2009; Ramadurai and Ukkusuri 2010), as well as innovations in pedestrian modeling (Hoogendoorn and Bovy 2005) present additional promising alternatives to generate social contact networks in the future. As such, it is critical to develop methods which can exploit this data in aiding the prevention and mitigation of contagion episodes.

The objective of the model proposed in this paper is to infer the spatiotemporal path of infection through a social-contact network for an ongoing outbreak scenario under the assumption that limited infection information is available. This work specifically considers contact-based diseases, which refer to the family of infectious diseases that are transmitted from an infected to susceptible individual via direct contact. This category includes sexually transmitted diseases, various strands of the flu, SARS and the common cold, among others. In turn the social contact network is representative of the social interactions (e.g. through school, work or home) which occur among a group of individuals in a given time period (e.g. a day).

The problem approached in this paper considers the case in which the structure of the network is deterministically known (set of nodes and links), but time-of-infection data is available only for a fraction of the population. We further assume that no information is known about the infection tree (i.e., the set of social contacts through which the disease spread). We refer to this set of assumptions as the *partial information* version of the problem, in contrast to the *full information* case in which time-of-infection information is available for all infected nodes. In previous work by Gardner et al. (2012), an application of the *full information* problem was addressed, where the objective was to infer the most likely air travel routes responsible for spreading the Swine Flu to unexposed geographic regions. In Gardner's paper social contact networks were not considered, and the network structure was defined by the air traffic system.

Generalization of the full information case to the partial information case introduces a significant increase in computational complexity. The partial information problem can be modeled as an integer program, and represents a considerably more difficult problem to solve than the full information case. Heuristic solution methods are therefore developed based on sub-problems which can be solved much more efficiently. The model performance is based on how accurately it predicts the paths of infection for a given contagion episode (which are extracted from simulation outputs). The outcome of the model can provide insight into future epidemic outbreak behavior and aid in the evaluation and recommendation of intervention strategies. In addition, the proposed solution methodology can be extended to alternative contagion processes which occur atop known network structures (e.g. tracking food borne outbreaks which propagate though a distribution network).

In the following section a literature review of relevant network models is provided. Section 3 defines the problem and section 4 presents the mathematical problem formulation and solution methodology. Section 5 describes the evaluation procedure and numerical results. Section 6 concludes the paper with discussion of the results and future research directions.

## 2 Background

Dynamic contagion processes impact copious network systems, and are therefore the focus of various studies within the emerging field of network science. In addition to the transmission of infectious disease through communities and biological systems (Murray 2002; Anderson and May 1991), the spread of information, ideas and opinions via social networks can also be modeled as a contagion process (Coleman et al. 1966; Hasan and Ukkusuri 2011); as well as the global spread of computer viruses on the Internet network (Newman et al. 2002; Balthrop et al. 2004); power grid failures in electricity markets (Kinney et al. 2005; Sachtjen et al. 2000); and the collapse of financial systems (Sornette 2003). Of interest to this study is the propagation of disease through a social contact network, and therefore will be focus of the remainder of the section 2.

The infection rate and pattern of the disease spreading process through a network is dependent on both the parameters of the disease (infectious period, level of contagiousness, etc.) and the fundamental structure of the network. In efforts to predict expected disease spreading behavior and characteristics, epidemiological models span from extremely generalized and simplified analytical models to increasingly in-depth stochastic agent based simulation tools. Analytical models are used to quantify the statistical properties of epidemic patterns (Colizza et al. 2006; Balcan et al. 2009); however, they are unable to capture certain behavioral aspects of the dynamics of disease spreading, and often lack detailed information about the network structure. In contrast, agent based simulation models can be used to replicate possible spreading scenarios, predict average spreading behavior, and analyze various intervention strategies for a given network and disease while capturing a greater degree of detail, but in turn require a highly detailed set of input data (see Rvachev and Longini (1985), Epstein and Cummings (2002), Eubank et al. (2004), Hufnagel et al. (2004), Dibble and Feldman (2004), Cahill et al. (2005), Dunham (2005), Meyers et al.

(2005), Small and Tse (2005), Carley et al. (2006), Ferguson et al. (2006), Germann et al. (2006), Ekici et al. (2008), Roche et al. (2011), and Haydon et al. (2003)). The most recent and comprehensive models provide a greater degree of realism, but are difficult to implement within the short time frames in which real time control decisions must be made. Large scale simulation models can also be computationally taxing because multiple runs are required to accurately predict expected outcomes.

There currently exists a gap in the literature which calls for scenario specific disease prediction models. Most contagion models predict future potential outbreak scenarios based on system-wide information; however, they are not able to reconstruct the contagion process of an ongoing outbreak to reveal information about the current state of the network. Recent advances in disease modeling have begun addressing this issue. For example, there are models which use genetic sequencing data to analytically infer the geographic history of a given virus's migration (Drummond and Rambaut 2007; Lemey et al. 2009; Wallace et al. 2007; Cottam et al. 2008; Haydon et al.; 2003). Often this approach involves first enumerating all possible evolutionary trees, then assigning posterior probabilities based on specifics of the respective virus' mutation rates. Additionally the infection trees only include locations where samples were available. Jombart et al. (2009) proposed a novel approach to reconstruct the spatiotemporal dynamics of outbreaks from sequence data by inferring ancestries directly between strains of an outbreak using their genotype and collection date. The "infectious" links were selected such that the number of mutations between nodes is minimized. The idea of using infection data to construct the most likely path of transmission is the highlighted goal of this paper.

This study is motivated by the need to track viruses through space and time in order to aid in the implementation of real-time containment strategies. Often the required genetic data and mutation based statistical properties are unavailable, or impossible to gather within the required time-frame. The proposed approach relies instead on available infection reports, contact network structure and disease properties to infer the spatiotemporal path of infection through a contact network, data which can be more realistically gathered during an epidemic. The proposed methodology accounts for missing infection information, enabling previously over-looked infection sources to be included.

## 3 Problem Definition

Using infection reports, contact network structure and disease properties, the methodology described in this section makes inferences about infection spreading patterns in a population. The problem assumes an underlying contagion process which can be represented on a network by a discrete-time, stochastic process. The following terminology is used for the remainder of this paper:

i. $t_i$, *time stamps*: the time period at which a node was reportedly infected, or predicted to be infected
ii. $p_{ij}$, *link transmission probability*: the probability that an infected node $i$ will infect a susceptible (and adjacent) node $j$ in a single time step.

iii. $\lambda$ *infectious period*: the number of time steps an infected node remains infectious (i.e., is able to infect others) following its own infection. $\lambda$ can also represent the amount of time before recovery, hospitalization or some other type of removal from the network.

The problem objective is defined as follow: assume we are given a social contact network which has been exposed to infection, such as that shown in Fig. 1(a), in which a contagion process occurs resulting in a set of infected nodes (and corresponding time stamps for each) such as shown in Fig. 1(b). Assuming we are only given information on a subset of the infected nodes, such as the scenario shown in Fig. 1(c), we seek an infection tree such as that shown in Fig. 1(d) that branches to all known infected nodes, which maximizes the likelihood of the infection event.

The social contact network $G \in (V, A)$ is formally defined by a set of nodes, $V$, which represent a population of individuals, and links, $A$, which represent physical daily contacts between individuals. The set $N$ represents the set of individuals that became infected during the time period when population $V$ was exposed to infection. The set $I$ represents the set of information nodes: a subset of the infected individuals $N$, which were identified as infected (i.e., they visited a doctor, hospital, pharmacy, etc.). The problem can be further broken down into two information-based cases:

I. Full information: The ***complete set*** of infected nodes and the time stamp, $t_i$, for each infected node is available, i.e., $I=N$.
II. Partial information: Information on ***a subset*** of the infected node set, $I \subseteq N$, is available. This problem serves as the more general version of the problem and is the focus of this paper.

Relaxing the full information assumption results in a more realistic setting where only a fraction of infected individuals consult a physician, visit a hospital, etc., resulting in partial information. The objective of the partial information case is again to determine the most likely set of infection spreading contacts when only a subset of the infected nodes are identified. One highlight of this study explores the performance of the proposed model under different levels of available information.

### 3.1 Link-Based Infection Process

The relationship between the underlying contagion process and the mathematical programming formulation (presented in section 4) are of specific interest in regards to the problem definition. This section introduces the link-based infection process. The network-based contagion process is introduced in section 3.2. The link-based infection process consists of a set of link trials which are the basic building blocks of the network level contagion process. In other words, a given infection scenario at the network level is the result of many individual link-based trials.

Each link trial consists of the following evaluation: At a discrete time step $t$, assume node $i$ is in an infectious state, node $j$ is in a susceptible state, and the two nodes are connected by link $(i,j)$ with a link transmission probability $p_{ij}$. A successful link trial is defined as when node $i$ infects node $j$ in time step $t$, and occurs with probability $p_{ij}$. The

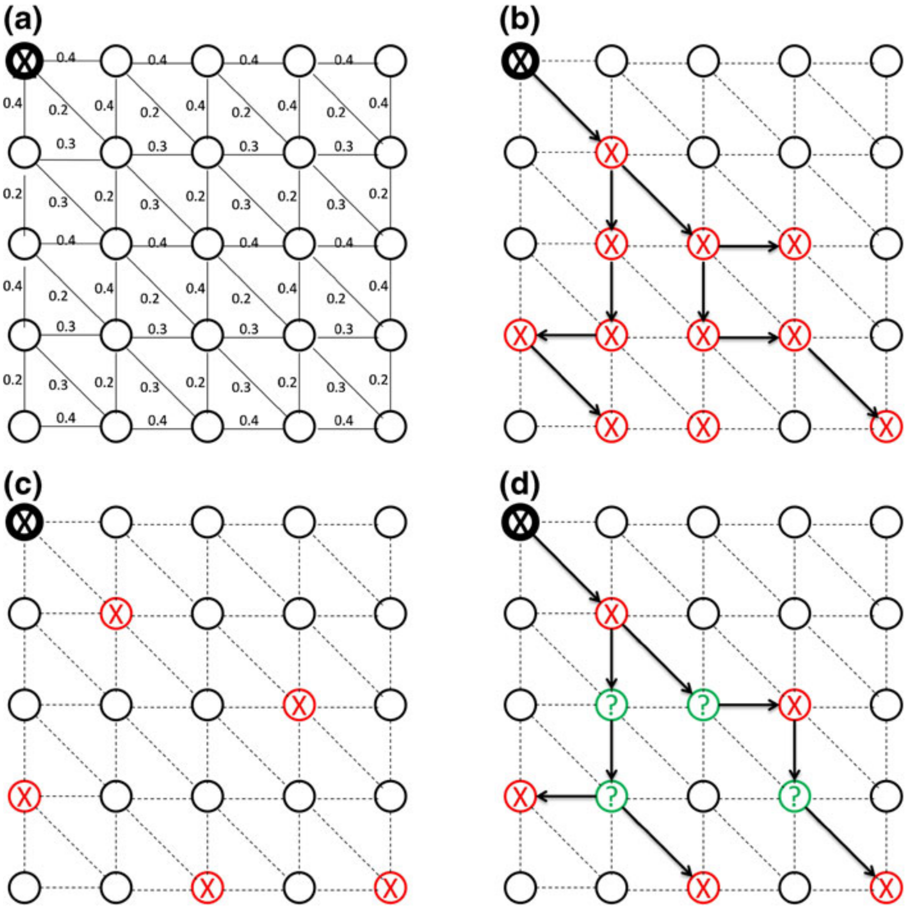

**Fig. 1** Illustration of the problem definition (**a**) Sample contact network structure $\boldsymbol{G} \in (V, A)$, and link transmission probabilities, $\boldsymbol{p}_{ij}$ (**b**) Example of outbreak scenario, (**c**) Example of the node level information provided (after an outbreak); set of information nodes, $\boldsymbol{I}$ are highlighted with "x" (**d**) Example of model output (*arrows*) predicting the infection pattern. The *upper left* hand corner node is the source

probability a link trial is unsuccessful is therefore $(1-p_{ij})$. A simulation time step $t$ is representative of the latent period, or the amount of time between when an individual contracts the disease and becomes infectious. The timestamp of node $i$, denoted by $t_i$, represents the time (e.g., day) at which individual $i$ was infected.

We now consider two connected nodes, $i$ and $j$. In calculating the probability associated with the inclusion or exclusion of link $(i,j)$ in the infection tree, we must account for two events: either no infection trials are successful, or exactly one infection trial is successful.

We denote the probability of no successful trials on $(i,j)$ by $\gamma_{ij}$. More explicitly, $\gamma_{ij}$ represents the probability that the correct number of trials were unsuccessful so as to ensure that node $i$ did not infect node $j$. The number of necessary unsuccessful trials is represented by:

$$\Delta t_{ij} = \min\left\{\max\left\{t_j - t_i, 0\right\}, T - t_i, \lambda\right\} \tag{1}$$

This expression accounts for situations where node $j$ was infected after $i$ (corresponding to $t_j-t_i$), infected before $i$ (corresponding to 0), or not infected (corresponding to $T-t_i$) and λ. The value of $\gamma_{ij}$ is given by expression (1):

$$\gamma_{ij} = \left(1-p_{ij}\right)^{\Delta t_{ij}\left(1-x_{ij}\right)\sum_{(k,i)\in A}\left[x_{ki}\right]} \tag{2}$$

The decision variable $x_{ij}$ is included so as to account for this term only if the link is excluded from the network (i.e., $x_{ij}=0$) and node $i$ has been infected (i.e., $\sum\limits_{(k,i)\in A}[x_{ki}] = 1$). If either condition is not satisfied, the term will evaluate to 1.

Similarly, the probability of exactly one successful trial on link $(i,j)$, which we denote by $\alpha_{ij}$, can be calculated for nodes $i$ and $j$ such that $t_j>t_i$ as the probability of $\Delta t_{ij}-1$ unsuccessful trials, and a single successful trial:

$$\alpha_{ij} = \left(1-p_{ij}\right)^{x_{ij}\left(\Delta t_{ij}-1\right)} p_{ij}^{x_{ij}} \tag{3}$$

The decision variable $x_{ij}$ allows the expression to take on the correct probability expression if the link is included in the tree (i.e., $x_{ij}=1$), and 1 otherwise. Combining expressions (2) and (3), we can develop an expression which represents the probability associated with both the inclusion and exclusion of a link:

$$\left(1-p_{ij}\right)^{x_{ij}\left(\Delta t_{ij}-1\right)} p_{ij}^{x_{ij}} \left(1-p_{ij}\right)^{\Delta t_{ij}\left(1-x_{ij}\right)\sum_{(k,i)\in A}\left[x_{ki}\right]} \forall(ij)\in A \tag{4}$$

When $x_{ij}=\mathbf{0}$ and $\exists\ k{:}x_{ki}=\mathbf{1}$, the link is not included in the infection tree and the probability is equal to $(1-p_{ij})^{\Delta t_{ij}}$. Then the term evaluates to 1. If $x_{ij}=\mathbf{1}$ the link is included in the infection tree and the associated probability is equal to $\left(1-p_{ij}\right)^{\Delta t_{ij}-1} p_{ij}$. In the next section we extend this result to the network level.

## 3.2 Network-Based Infection Process

This work treats the network-based infection process as an iterative aggregation of individual link trials. We begin the simulation model by initializing all nodes to a susceptible state, and randomly choose a set of nodes to be infected ($0 \in$ V). Then we simulate transmission of the disease over multiple time steps, $t$, for a predetermined simulation period, $T$. During each time step, we identify all links that connect infectious nodes and susceptible nodes, and perform an infection trial for each such link. If the link infection is successful, then we change the newly infected node status to "infected" in the following time step. The node remains infected for λ time steps. After a node is infected for λ time steps, its status is changed to "recovered". Once a node is recovered it can no long transmit the disease or become infected again (the equivalent of gaining immunity or being removed from the network).

This process is representative of a discrete-time network Susceptible-Infectious-Removed (SIR) contagion process. The simulation model described in words above forms the basis for the mathematical formulation and evaluation presented in the remainder of this paper. It follows that the aim of this solution methodology is to replicate the actual infection tree for a specific outbreak scenario by exploiting node level infection information and the network structure.

### 3.3 Assumptions

Multiple simplifying assumptions are necessary to solve the proposed problem. This work assumes:

i. a priori knowledge of the underlying social contact network, $G \in (V, A)$
ii. The contagion process can be approximated as discrete-time network SIR contagion process with known transmission probabilities, $p_{ij}$
iii. An individual can be infected at most once, and thus only those diseases for which immunity is acquired after recovery are considered.
iv. Known timestamps $t_i$ for the set of information nodes, $I$

The first assumption is the most debatable of the four. Social networks are difficult to characterize, as they are not directly observable, and also highly unstable. However, the increase of social networking information available online, and improvements in activity-based travel modeling both contribute towards the possibility of access to more detailed social contact information in the future. The second assumption is also present in many previous epidemiological models and ongoing research is focused on accurately quantifying these parameters. The third assumption restricts the set of applications to those diseases for which acquiring immunity restricts an individual from being infected more than once over the entire course of an outbreak. Assumption 4 is based on the premise that some infected individuals report to a medical authority (i.e., hospital, private clinic, pharmacy), and that information is made available.

## 4 Method

In this section the mathematical formulation and proposed solution methodology for the partial information case (as defined in section 3) are presented. A non-linear integer program formulation for the partial information case is given in section 4.1and the solution method for the partial information case is described in section 4.2.

### 4.1 Mathematical Formulation

The partial information case represents the case where not all infections are reported. In this scenario, the fraction of missing information is unknown, i.e., the nodes that are unreported may or may not have been infected. These nodes are referred to as *zero-information nodes, $i \in N \backslash I$*. The nodes with known timestamps are referred to as *information nodes,* $i \in I$. In the case of partial information, the objective is to determine the set of links which spread the infection, while simultaneously determining the time at which zero-information nodes were infected, if at all. What follows is a non-linear integer programming formulation:

$$\prod_{\forall(i,j)\in A}\left(1-p_{ij}\right)^{x_{ij}\left(\Delta t_{ij}-1\right)} p_{ij}^{x_{ij}}\left(1-p_{ij}\right)^{\Delta t_{ij}\left(1-x_{ij}\right)\sum_{(k,i)\in A}\left[x_{ki}\right]} \tag{5}$$

s.t.

$$\Delta t_{ij}=\min\left\{\begin{matrix}\max\left\{t_j-t_i,0\right\},\\ T-t_i,\\ \lambda\end{matrix}\right\}\forall(i,j)\in A \tag{6 – 8}$$

$$t_{\max}\left(x_{ij}-1\right) < t_j-t_i\forall(i,j)\in A \tag{9}$$

$$t_{\max}\left(1-x_{ij}\right)+\lambda\geq t_j-t_i\forall(i,j)\in A \tag{10}$$

$$t_i=t_i^f\forall i\in I \tag{11}$$

$$\sum_{j\in N}x_{ji}=1\forall i\in I\setminus O \tag{12}$$

$$\sum_{j\in N}x_{ji}\leq 1\forall i\in N\setminus I \tag{13}$$

$$x_{ij}\leq\sum_{j\in N}x_{ji}\forall i\in N\setminus I \tag{14}$$

$$x_{ij}=\{0,1\}\forall(ij)\in A \tag{15}$$

$$t_i\in\mathbb{Z}\forall i\in N\setminus I \tag{16}$$

The two decision variables are i) $x_{ij}$, which is set to 1 if link $(i,j)$ is included in the infection tree and 0 otherwise, and ii) $t_i$ which is the timestamp assigned to node $i$. The objective (5) enforces that the set of links included in the final spanning tree maximizes the likelihood of the tree. The first constraint (6-8) provides the definition of $\Delta t$. The next two constraints enforce consistency between the $x$ and $t$ variables: If $x_{ij}$=1, meaning $i$ is the predecessor of $j$ in the infection tree, then (9) guarantees that the infection time of $j$ will be later than that of $i$. Constraint (10) ensures that that the infection time of $j$ will be within the time period when $i$ is infectious, i.e., within λ time units of the infection time of $i$. Constraint (11) fixes the timestamp variable $t_i$ for all information-nodes. Constraints (12)-(14) enforce the spanning tree structure of the solution. Constraint (12) ensures that every known infected node, except the source node, is infected exactly once, i.e., has exactly one incoming link. Constraint (13) allows zero-information nodes to be part of the infection tree, but restricts them to have at most one predecessor. Constraint (14) ensures that only zero information nodes that have been previously infected will be able to in turn infect other nodes. Constraints (15) and (16) force the decision variables $x_{ij}$ to be binary and $t_i$ to be integer.

The objective function (5) can be transformed from a product of terms to an equivalent summation of terms by maximizing the natural logarithm of the objective function.

$$\sum_{\forall(i,j)\in A} x_{ij}\left(\Delta t_{ij}-1\right)\ln q_{ij}+x_{ij}\ln p_{ij}-x_{ij}\Delta t_{ij}\ln q_{ij}\sum_{(k,i)\in A} x_{ki}+\Delta t_{ij}\ln q_{ij}\sum_{(k,i)\in A} x_{ki}$$

The last term of this expression represents the penalty associated with link ($i$,$j$) resulting in no infections based on the infection of node $i$. The individual terms of this summation can be redistributed among the expressions corresponding to each incoming link to $i$. Redistributing the terms, and expanding the first term, we obtain the following expression:

$$\sum_{\forall(i,j)\in A} x_{ij}\Delta t_{ij}\ln q_{ij}-x_{ij}\ln q_{ij}+x_{ij}\ln p_{ij}-x_{ij}\Delta t_{ij}\ln q_{ij}\sum_{(k,i)\in A} x_{ki}+x_{ij}\sum_{(j,l)\in A}\Delta t_{jl}\ln q_{jl}$$

We can further simplify the above expression by examining the behavior of the term $x_{ij}\Delta t_{ij}\ln q_{ij}\sum_{(k,i)\in A} x_{ki}$.

**Lemma 1** If the $x$ variables are integer, the behavior of the expression $x_{ij}\sum_{(k,i)\in A} x_{ki}$ is equivalent to the behavior of $x_{ij}$.

Proof. If $x_{ij}$=1, because of constraints (13) and (14), $1=x_{ij}\leq\sum_{(k,i)\in A} x_{ki}=1$. If $x_{ij}=$ 0, then $x_{ij}\sum_{(k,i)\in A} x_{ki}=0$.

Using Lemma 1, we can simplify the formulation, cancel similar terms, arriving at the objective function shown in Eq. 17. The new formulation features additive terms rather than multiplicative terms, although it remains nonlinear due to interaction terms between the $x$ and $t$ variables. The set of constraints remains the same, resulting in the formulation below:

$$\max \sum_{\forall (i,j) \in A} x_{ij} \left[ \ln p_{ij} - \ln q_{ij} + \sum_{(j,l) \in A} \Delta t_{jl} \ln q_{jl} \right] \tag{17}$$

s.t.

$$\Delta t_{ij} = \min \left\{ \begin{array}{c} \max\{t_j - t_i, 0\}, \\ T - t_i, \\ \lambda \end{array} \right\} \forall (i,j) \in A \tag{18}$$

$$t_{max}(x_{ij} - 1) < t_j - t_i \forall (i,j) \in A \tag{19}$$

$$t_{max}(1 - x_{ij}) + \lambda \geq t_j - t_i \forall (i,j) \in A \tag{20}$$

$$t_i = t_i^f \forall i \in I \tag{21}$$

$$\sum_{j \in N} x_{ji} = 1 \forall i \in I \setminus O \tag{22}$$

$$\sum_{j \in N} x_{ji} \leq 1 \forall i \in N \setminus I \tag{23}$$

$$x_{ij} \leq \sum_{j \in N} x_{ji} \forall i \in N \setminus I \tag{24}$$

$$x_{ij} = \{0, 1\} \forall (ij) \in A \tag{25}$$

$$t_i \in \mathbb{Z} \forall i \in N \setminus I \tag{26}$$

We will refer to constraints (18) - (26) as the *IP constraints.*

## 4.2 Solution Method Under Full Information

The full information case is a special, more tractable version of the partial information case presented above. Under the assumption that $I$=$N$, i.e., we know the full set of infected nodes, we need not optimize over the variables $t$, since they are all fixed, which allows us to make the problem linear. The simplified formulation allows us to exploit specific properties to develop a much more efficient solution method than solving this linear program directly.

The properties of the full information case result in a spanning tree that branches to every node $i \in I$. The general problem of finding a directed maximum branching tree can be solved using a simplified version of the algorithm developed by Edmonds (1967). Edmonds' algorithm consists of maintaining an optimal sub-network that reaches every node, and works towards feasibility by replacing links that form a cycle in that sub-network. As such, the cycle finding subroutine of the algorithm is the most computationally taxing part of the algorithm. A significantly more efficient algorithm can be developed for the full information version of the problem by pruning the set of links to be considered based on constraints (19) and (20). The resulting network is acyclic, which greatly simplifies the maximum branching procedure.

Let the set of feasible links $(i,j) \in L$ be such that $t_i < t_j < (t_i + \lambda)$. It is trivial to show that in all feasible solutions, $x_{kl} = 0$ for all links $(k,l)$ in $A \backslash L$. Therefore, we can limit our focus to the link set $L$. Because of constraint (19), which requires feasible links in $L$ to connect nodes with increasing timestamps, the resulting sub-network has a topological ordering, and as such, cannot contain any directed cycles. We further note that constraints (23) and (25) represent the requirement that exactly one incoming link is chosen for every infected node. As such, any solution that only chooses links in $L$ so that every node has exactly one incoming link will be feasible. The following algorithm can be implemented to solve the full information case, and the resulting set of selected links $S$ forms the maximum likelihood tree.

i. Define the set of feasible links, $L = \{(i,j): t_i < t_j < (t_i + \lambda)\}$
ii. Calculate link costs, $P_{ij}$ for links $(i,j)$ in feasible set $L$
iii. For each infected node, $j \in I$, select the incoming link $(i,j)$ with the highest cost, $P_{ij}$, from the set of feasible links $L$ and add it to the solution tree $S$.

However under the case of partial information, the nonlinear nature of the objective function, i.e., the multiplication of $x$ and $t$, makes the formulation for the partial information case very difficult to solve. The introduction of the $\boldsymbol{t}$ vector as a variable rather than a set of inputs restricts the set $S$ from being solved directly and instead must be solved for simultaneously with the set of timestamps. Because the link costs are no longer constant, rather a function of the timestamps, the proposed full information algorithm can no longer be used to find an optimal solution. The resulting problem is in fact a non-linear, non-convex mixed-integer program.

### 4.3 Partial Information Heuristic Solution Method

Due to the complexity of the mixed-integer bi-linear formulation presented in section 4.1, the focus of this research is to develop a heuristic which exploits the bilinearity of the problem in finding good solutions. While approaches exist in the literature to solve bilinear programs with mixed integer variables or jointly constrained feasible regions, no such approach exists which can accommodate both complicating factors concurrently. The non-convexity of the feasible region due to the integrality constraints, and the changing nature of the feasible region, in particular the extreme points of such a region, make the problem intractable. As such, the solution algorithm presented in this paper involves identifying a set of feasible solutions from which improvements can be made through the solving of simpler problems. The general outline of the algorithm is as follows:

i. Find a feasible solution by solving a ***linear integer program***, which is an approximation of the original bilinear objective.
ii. Fix the infection tree, and solve a ***linear program*** to find the optimal set of time stamps.
iii. Fix the set of variables corresponding to the time stamps, and use a ***branching procedure*** to find the optimal infection tree given the chosen time of infection variables.
iv. Continue to iterate between steps ii and iii until solution converges.
v. Refine the optimal tree from step iv by systematically adding nodes to the tree which increase its likelihood.

Each step is discussed in detail in this section.

### *4.3.1 Finding a Feasible Solution*

Due to the complexity of the problem's feasible region, finding a feasible solution to the problem represents a non-trivial effort, and is in fact the most difficult and most crucial step of the procedure. As the size of the problem can grow quickly with respect to the network size, different pre-processing procedures were developed so as to improve the performance of the algorithm.

In order to find a feasible solution, we replace the bilinear objective function (17) with the *linear approximation* objective function shown below:

$$\sum_{\forall(i,j)\in A} x_{ij}\left[\ln p_{ij} - \ln q_{ij} + \lambda \sum_{(j,l)\in A} \ln q_{jl}\right] \tag{27}$$

The objective function (27) represents a linear approximation of the bilinear objective function (although it retains integer constraints), where the difference in time stamps is replaced by the infectious period. This model will hereby be referred to as *A_ILP*. The formulation is further reduced because $\Delta t_{ij}$ is no longer a part of the problem, which removes the complicating constraint related to it. Because of the size of the IP, and the fact that a linear objective function will not in general approximate the bilinear objective function globally, preprocessing techniques were used to find feasible solutions quicker. In order to accomplish this step Lemma 2 was established.

**Lemma 2.** Assume a feasible solution exists, i.e., one which satisfies the IP constraints. If such a solution exists, then at least one solution exists where $x_{ij}=\mathbf{1}$ for any $t_j>t_i, t_j<t_i+\lambda$, and $ij \in I$.

*Proof.* Consider a feasible solution where $x_{ij}=\mathbf{0}$, i.e., link $(i,j)$ is not included. Because $j \in I$, then $\exists(k,j) \in A{:}x_{kj}=\mathbf{1}$. We can construct a new feasible solution by setting $x_{kj}=\mathbf{0}$ and $x_{ij}=\mathbf{1}$. Changing the value of this variables will only affect constraints related to nodes $i,j,k$ and links $(i,j)$ and $(k,j)$. Removing link $(k,j)$ only affects the constraint requiring node $j$ to be infected once, which is satisfied by including link $(i,j)$. The infection timing requirements given by constraints (19) and (20) are satisfied based on the conditions provided in the lemma. And because both nodes $i$

and *j* belong to *I*, no further constraints need to be satisfied. This implies the new solution is feasible, and therefore the lemma is proven.

The implications of this lemma are important in developing a more efficient method to construct an initial feasible solution. We can exploit Lemma 2 to more easily construct a feasible solution by adding any links that satisfy the conditions of the lemma a priori. This procedure is defined in Algorithm 1. The objective of the algorithm is to identify a set of link variables, $x_{ji}$, to fix a priori based on their contribution to the objective function (27). To do this the variable **bestPredecessorValue** (initialized to$-\infty$) is created to identify the incoming link for each node which maximizes the contribution to the objective function (27), such that both nodes belong to *I*. The **bestPredecessorValue** variable is updated by evaluating each incoming link (*j,i*) for node *i* such that $i,j \in I$. The best incoming link will be added to the solution, and its corresponding $x_{ji}$ variable fixed to 1. Fixing variables a priori will decrease the number of decision variables in *A_ILP*. Additionally, the way in which they are fixed guarantees a feasible solution can be obtained. The outline of the algorithm is outlined below.

**Algorithm 1**: Find Initial Feasible Solution

```
Inputs:
Infected nodes, $I^*$;
for all $i \in I^*$
    $j' \leftarrow NULL$
    bestPredecessorValue$\leftarrow -\infty$
    for all $j \in I^*: (j,i) \in A, t_j < t_i$
        if $(t_i - t_j) \ln(1 - p_{ji}) >$ bestPredecessorValue
            $j' \leftarrow j$
            bestPredecessorValue $\leftarrow (t_i - t_j) \ln(1 - p_{ji})$
        end if
    end for
    set $x_{j'i} = 1$
end for
```

The next step of the heuristic is implemented to find a set of initial feasible solutions, which is accomplished by solving *A_ILP*. Steps (ii)-(v) of the heuristic exploit not only the optimal solution to *A_ILP*, but also intermediate solutions found throughout the resolution of the problem. We solve *A_ILP* using a branch and cut procedure which is implemented using a commercial IP solver. More precisely, after fixing a sub-set of the links to be included in the tree (based on the output from Algorithm 1), we solve *A_ILP* to optimality. While solving *A_ILP* we store a subset of the feasible solutions found, which will be used as starting solutions in the subsequent steps of the heuristic. Because we do not want to consider all feasible solutions found while solving *A_ILP*, only feasible solutions which improve upon the current best known solution in terms of the objective value (27) are stored.

The outcome from this procedure is a set of initial feasible trees, and corresponding timestamps. Each tree/timestamps combination represents a starting point for the remainder of the heuristic which is computationally efficient to solve, and only benefits from a more diverse set of initial starting solutions. Given an initial starting solution (i.e., a fixed infection tree), the next step of the heuristic involves exploiting the bi-linearity of the problem to obtain two much simpler problems by fixing one set of variables at a time, and then iterating between the sub-problems to increase the likelihood of the tree. In one sub-problem the set of links ***x*** included in the initial solution are fixed and the optimal timestamps for the zero-infection nodes included in the tree are sought. In the second sub-problem the timestamps, ***t***, are fixed for the same set of nodes, and the optimal set of links are sought.

### *4.3.2 Finding Optimal Time Stamps*

Fixing the ***x*** (infection pattern variable) vector linearizes the objective function (17), transforming the problem into a linear integer program with decision variable ***t*** (time of infection). However, we are left with the complicating constraint needed to determine the appropriate value of $\Delta t$. The formulation is shown below.

$$\max \sum_{\forall (i,j) \in A, i,j \in I*} x_{ij} \left[ \ln p_{ij} - \ln q_{ij} + \sum_{(j,l) \in A} \Delta t_{jl} \ln q_{jl} \right] \tag{28}$$

s.t.

$$\Delta t_{ij} = t_j - t_i \, \forall \, (i,j) \in A, i, j \in I^* \tag{29}$$

$$\Delta t_{ij} = \min\{\lambda, T - t_i\} \forall (i,j) \in A, i \in I*, j \notin I^* \tag{30}$$

$$t_{\max}(x_{ij} - 1) < t_j - t_i \forall (i,j) \in A, i, j, \in I^* \tag{31}$$

$$t_{\max}(1 - x_{ij}) + \lambda \geq t_j - t_i \forall (i,j) \in A, i, j \in I^* \tag{32}$$

$$t_i = t_i^f \forall i \in I^* \tag{33}$$

$$0 \leq t_i \leq T \forall i \in I^* \backslash I \tag{34}$$

We can remove the complicating constraint by introducing a new integer variable $s_i$ which indicates the region the timestamp of node $i$ is in, and as such whether min $\{\lambda, T - t_i\}$ is equivalent to $\lambda$ or $T - t_i$. If $s_i = \mathbf{0}$, then $0 \leq t_i \leq T - \lambda$. Alternatively, if $s_i = \mathbf{1}$, then $T - \lambda \leq t_i \leq T$. We also simpify the notation by denoting $A^*$ to be the set of links included in the initial feasible solution.

$$\max \sum_{\forall (i,j) \in A^*, i,j \in I^*} \left\{ \sum_{(j,l) \in A} \Delta t_{jl} \ln q_{jl} \right\} \tag{35}$$

s.t.

$$\Delta t_{ij} = t_j - t_i \forall (i,j) \in A, i, j \in I^* \tag{36}$$

$$\Delta t_{ij} \geq \lambda - s_i M \forall (i,j) \in A, i \in I^*, j \notin I^* \tag{37}$$

$$\Delta t_{ij} \geq T - t_i - (1 - s_i) M \forall (i,j) \in A, i \in I^*, j \notin I^* \tag{38}$$

$$t_i \geq s_i (T - \lambda) \forall i \in I^* \tag{39}$$

$$t_i \leq T - (1 - s_i) \lambda \forall i \in I^* \tag{40}$$

$$t_{\max} \left( x_{ij} - 1 \right) < t_j - t_i \forall (i,j) \in A, i, j \in I^* \tag{41}$$

$$t_{\max} \left( 1 - x_{ij} \right) + \lambda \geq t_j - t_i \forall (i,j) \in A, i, j \in I^* \tag{42}$$

$$t_i = t_i^f \forall i \in I^* \tag{43}$$

$$0 \leq t_i \leq T \forall i \in I^* \backslash I \tag{44}$$

While the program above is still an integer program, the timestamp variables can now be relaxed to be continuous and will maintain integrality. As such, rather than having to consider general integer variables, the problem is simplified to include only binary variables.

### *4.3.3 Finding Optimal Infection Tree*

Given a set of time stamps (i.e. ***t*** is fixed), the next step of the heuristic is to identify the optimal set of links, ***x***, branching to each node with a fixed $t_i$ which maximizes the likelihood of the tree. Once again exploiting the structure of the master problem, the objective function becomes linear when ***t*** is fixed. Furthermore, the $\Delta t$ constraint can be simplified because we only consider links which connect nodes with known timestamps (i.e. $t_i$ is no longer a decision variable). The formulation is shown below.

$$\max \sum_{\forall (i,j) \in A} x_{ij} \left[ \ln p_{ij} - \ln q_{ij} + \sum_{(j,l) \in A} \Delta t_{jl} \ln q_{jl} \right] \tag{45}$$

s.t.

$$\Delta t_{ij} = \min \left\{ \begin{array}{c} \max\{t_j - t_i, 0\} \\ T - t_i, \\ \lambda \end{array} \right\} \forall (i,j) \in A \tag{46}$$

$$t_{\max}\left(x_{ij}-1\right) < t_j-t_i\forall(i,j)\in A, i, j\in I^* \tag{47}$$

$$t_{max}\left(1-x_{ij}\right) + \lambda \geq t_j - t_i\forall(i,j)\in A, i, j\in I^* \tag{48}$$

$$\sum_{j\in N} x_{ji} = 1\forall i\in I^*\backslash 0 \tag{49}$$

$$x_{ij} = \{0,1\}\forall(i,j)\in A, i, j\in I^* \tag{50}$$

The reduced problem has a very specific structure, as it is the integer programming representation of a single root branching problem in an acyclic graph. This step of the heuristic is equivalent to the full information version of the problem, and can be solved using the method described in section 4.2. The solution method hinges on the acyclic nature of the graph: unlike general branching problems, there is no need to check for cycles because the time stamp constraints on links will ensure that no feasible links will produce a cycle.

The two steps of the heuristic, ii) fixing $\boldsymbol{x}$ and solving for $\boldsymbol{t}$, and iii) fixing $\boldsymbol{t}$ and solving for $\boldsymbol{x}$, are iterated until convergence is reached (i.e. the set of links in the tree, $x_{ij}$, does not change between iterations).

### 4.3.4 Refining Optimal Tree

The performance bottleneck in the heuristic is the fixed set of nodes included in the initial feasible tree, which remain constant for steps ii and iii of the heuristic. More specifically, the procedure used to find an initial feasible solution has two shortcomings:

i. The objective function is an approximation of the true objective function
ii. The set of nodes to be infected is fixed with the initial feasible solution.

While the steps of the heuristic following the identification of a feasible solution do account for the true objective function, their inability to add nodes to the infection tree, i.e., a fixed set $I^*$, represents a limitation of the model. Therefore the final step of the heuristic is a procedure to improve the current solution by adding links/nodes to the infection tree. In particular a random greedy insertion method is implemented, which randomly selects nodes excluded from the tree, calculates the potential improvement associated with adding the node to the tree, and decides whether or not to include it. The improvement from adding a node is the difference between the probability of no successful trial reaching the node (current case), and the probability of one successful trial (the node gets infected) plus the probability of no successful trials of the newly infected node (the node doesn't infect any others).

In order to efficiently identify nodes which can potentially improve our solution, we first calculate a node-based metric which represents the potential improvement associated with adding any node to the current tree. The following three factors determine the potential contributions of a new node $j$ to the network:

i. Assigning a timestamp to node *j* will affect the number of unsuccessful trials between node *j* and all adjacent infected nodes.
ii. Infecting node *j* will require accounting for the appropriate number of unsuccessful trials between node *j* and all adjacent non-infected nodes.
iii. Assuming a node *i* infects node *j*, the probability associated with excluding link (*i,j*) will be replaced with the probability of including it.

Figure 2a and b illustrate the calculations necessary for evaluating the benefit of excluding and including a specific node *i*, respectively, given nodes *j,k* and *l* are already included in the infection tree. As shown in Fig. 2a, if node *i* is *excluded* from the infection tree, the adjacent infected nodes could not have been infected by node *i*, therefore we must account for λ unsuccessful infection trials between node *i* and infected nodes *j,k,l*. If node *i* is to be *included* in the infection tree and if node *i* was infected by node *l* with a timestamp $t_i$=3, as illustrated in Fig. 2b, the number of unsuccessful trials between *l* and *i* is now 0, and 2 between *k* and *i*. Furthermore, because the timestamp assigned to *i* is less than that of *j*, we must now account for 2 unsuccessful trials from node *i* to node *j*. Finally, because nodes *a* and *b* were not included in the infection tree, we must also account for the λ unsuccessful trials between node *i* and nodes *a* and *b*. In general, the benefit of including link (*l,i*) in the infection tree will depend on the sum of terms associated with it.

The algorithm for evaluating the inclusion and exclusion benefit of each node is presented below as Algorithm 2. The algorithm is performed for a fixed number of iterations (denoted as maxIterations). During each iteration, a given node *i* which is not currently in the infected set $I^*$ is selected, and the benefit associated with allowing this node to be infected, ln *P* (*inc*), is computed and compared with the benefit associated with keeping it excluded from the infection tree.

First the benefit of inclusion is computed. In order to allow a new node to be infected, we must also identify the node's timestamp and predecessor, denoted by $\delta^*$ and ξ, respectively. The benefit of selecting each possible predecessor *j* is computed by enumerating the possible timestamps for node *i*, indexed by δ, (based on a given

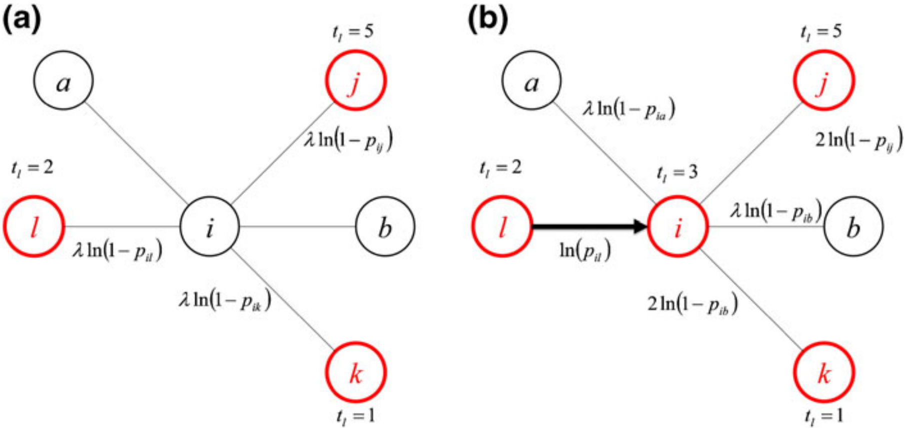

**Fig. 2** Representation of the probabilities considered when calculating the exclusion (**a**) or inclusion (**b**) of node ***i*** from the infection tree

predecessor *j*) and calculating the benefit that infecting node *i* will have on every adjacent node, indexed by *k*. This calculation varies depending on whether node *k* belongs to the infected set $I^*$, the timestamp of *k,* and the potential timestamp of *i*. The benefit that infecting node *i* will have on an adjacent node *k* is equal to the expression $(t_i - t_k)\ln(1 - p_{ki})$. Next the link-based benefits are summed to obtain the node-level benefit for each timestamp and predecessor combination. For each node *i* evaluated the best combination of predecessor $\xi$ and timestamp $\delta^*$ is selected.

After the benefit of inclusion is complete, the benefit of excluding the link, *P*(*exc*), is computed, and equal to the probability that no adjacent infected node successfully infected *i*. If the benefit associated with including the node *i* is greater than that of excluding it, node *i* is added to the infected set, $I^*$ and the relevant link (based on the optimal predecessor) with corresponding timestamp is added to infection tree. Otherwise, the node remains excluded from the infection tree.

**Algorithm 2**: Node Addition

```
Inputs: Infected nodes, $I^*$; Timestamps, $\boldsymbol{t}$; Infection Tree, $\boldsymbol{x}$.
iteration ← 0
while   iteration < maxIterations
    Randomly choose $i \notin I^*$
    ln(Probability of Inclusion), $\ln P(inc) \leftarrow -\infty$
    Predecesor $\xi \leftarrow NULL$
    Timestamp $\delta^* \leftarrow NULL$
    for all $j \in J^*: (j,i) \in A$
        for $\delta = t_j + 1$ to $t_j + \lambda$
            $lnP(temp) = \ln p_{ji}$
            for all $k \in N \backslash \{j\}: (i,k) \in A$
                if $k \in I^*$
                    if $t_k \leq t_i$
                        $\ln P(temp) \leftarrow \ln P(temp) + (t_i - t_k)\ln(1 - p_{ki})$
                    else
                        $\ln P(temp) \leftarrow \ln P(temp) + (t_k - t_i)\ln(1 - p_{ik})$
                    end if
                else
                    $lnProb \leftarrow lnProb + \min\{\lambda, T - t_i\}\ln(1 - p_{ik})$
                end if
            end for
            if $\ln P(temp) > P(inc)$
                $P(inc) \leftarrow \ln P(temp)$
                $\xi \leftarrow$ j
                $\delta^* \leftarrow \delta$
            end if
        end for
    end for
    $P(exc) \leftarrow 0$
    for all $j \in N: (i,j) \in A$
        if $j \in I^*$ $P(exc) \leftarrow P(exc) + \min\{\lambda, T - t_j\}\ln(1 - p_{ji})$
        end if
    end for
    if $P(inc) \geq P(exc)$
        $x_{\xi,i} \leftarrow 1$
        $t_i \leftarrow \delta^*$
        $I^* \leftarrow I^* \cup \{i\}$
    end if
end while
```

## 5 Numerical Testing

This section details the procedure used to evaluate the performance of the heuristic solution method proposed, and presents numerical results.

### 5.1 Measure of Performance

Although the solution method itself does not require the use of a stochastic simulation model, one as defined in section 2.2 is used in to evaluate the performance of the proposed methodology. The model performance is measured by comparing the set of links (nodes) infected in the simulation-based scenario, *K*, with those predicted to be infected by the model, *S*. We take the following steps to evaluate the performance of the proposed methodology:

1. Randomly generate a network of the specified degree distribution, $G \in (V, A)$,
2. Set the infectious period λ, specify link transmission probabilities, $p_{ij}$, and specify level of information, *p*
3. Randomly introduce an infected individual into the network, *0*
4. Simulate an outbreak for some preset time period, *T*
5. Extract the following information from the simulation to use for evaluating the heuristic performance

   a) Full set of links in the infection tree, *K*
   b) The full set of infected nodes, *N*
6. Identify the set of information nodes, *I*, by randomly selecting *p* percentage of nodes in *N*
7. Extract the following (required) information from the simulation to use as input for the heuristic

   a) The set of infected nodes in the information set, *I*
   b) Timestamps for each known infected node, $t_i \ \forall i \in I$.
8. Implement the heuristic (as described in section 4.3)
9. Identify the percentage of correctly *infected* links, *q*, and percentage of correctly *infected* nodes, *b* in the model output tree, *S*.

   a) **q**=|M|/|K|, where M is the set of infection spreading links identified by the heuristic and included in the output tree S.
   b) **b**=|J|/|N|, where J is the set of infected nodes identified by the heuristic, and included in the output tree S.
10. Repeat steps (1)-(9) X **times** for a specified information level and degree distribution, and average *q* and *b* (step 9.a and 9.b) over all X iterations.

The procedure outlined above returns the *expected* performance of the solution methodology; ***Q*** (and ***B***) which is how accurately *S* represents the actual spreading scenario, on average for a specified network structure and level of information.

In addition to excluding infected nodes and links, the model also has the potential to include additional links in the tree, *S*, which did not actually spread infection, as well as incorrectly label uninfected nodes as infected. Therefore two additional performance measures were computed which reflect the ability of the model to appropriately exclude uninfected nodes and links: *i)* the percentage of correctly *labeled* links and *ii)* the percentage of correctly *labeled* nodes. These measures are computed similarly to step 9.a and 9.b above, but account for all links (nodes) in the network, rather than just the set of links (nodes) included in the infection tree, therefore penalizing the model for over-infecting the network. The full set of performance measures used to evaluate the model performance are listed below:

i. Expected percentage of correctly *infected* links: The percentage of infected links which are correctly identified as infected by the model on average.
ii. Expected percentage of correctly *labeled* links: The percentage of all links in the network which are correctly identified, as either infected or non-infected by the model on average.
iii. Expected percentage of correctly *infected* nodes: The percentage of infected nodes which are correctly identified as infected by the model on average.
iv. Expected percentage of correctly *labeled* nodes: The percentage of all nodes in the network which are correctly identified, as either infected or non-infected by the model on average.

Steps 1–9 were implemented to evaluate the heuristic performance for various network structures and information levels. Given the stochastic nature of the contagion process and the fact that the model performance will vary based on the size of the outbreak and specific contagion process which occurs, each performance measure was computed and averaged over 1,000 iterations to generate an expected performance.

Lastly, the heuristic performance was evaluated as a function of outbreak size. Specifically the expected percentage of correctly labeled links and nodes were evaluated as the percentage of infected nodes in the population increases. The results are presented for each network structure in Figs. 4, 6, and 8 under the assumption of 70 % information availability for the uniform and power law networks, and 75 % information availability for the Poisson network (the information level is higher for Poisson networks because of the time required to solve *A_ILP* to optimality 1,000 times). The results reveal behavioral patterns of the contagion process as a function of network structure, as well as illustrate the model performance as the percentage of the population infected grows. Numerical results and discussion are presented in the following section.

### 5.2 Numerical Results and Analysis

The model was evaluated for three different network structures with varying degree distributions: power law, Poisson, and uniform. The parameters for the power law and Poisson networks were set such that the average degree was the same for all networks, and equal to 3. The set of networks used for numerical testing were developed randomly so as to evaluate the performance of the algorithm over a wide range of possible network structures and levels of connectivity. All networks evaluated have 70 nodes, but the number of links varies depending on the degree distribution and random construct.

The model performance for each network structure was evaluated under increasing levels of information ranging from (0.7, 1), which represents the range from 70 % to

100 % of infections being reported. For each criterion the expected performance of the heuristic described in section 4.3 was compared with the performance of the solution from *A_ILP*. Because *A_ILP* increases rapidly in size as the network size grows and as the information level decreases, solving *A_ILP* to optimality proved to be the computational bottleneck in the evaluation process (due to the large number of iterations). However it should be noted that *A_ILP* does not have to be solved to optimality in order to implement the proposed heuristic solution; it is the feasible solutions generated while solving *A_ILP* which serve as the required starting points for the heuristic, which are computationally much easier to obtain.

### *5.2.1 Power Law Network*

The first network structure evaluated has a power law node degree distribution. Various studies have found that power law networks are representative of many real world networks, including social contact networks (Barabási and Albert 1999; Gonzalez et al. 2008). Power law networks have a hub and spoke type structure with few highly connected nodes, (known as super spreaders in the context of contagion problems), while most nodes have a very low degree. Due to the extremely heterogeneous structure of power law networks contagion processes behave differently compared with more homogenous network structures such as Poisson and uniform networks. In addition a power law network structure is much more subjective to the transmission properties of the disease; a low transmission probability translates to a low probability of infecting a super spreader node, therefore an outbreak is less likely to spread to a large proportion of the population. As the transmission probability increases the probability of infecting a super spreader increases, significantly

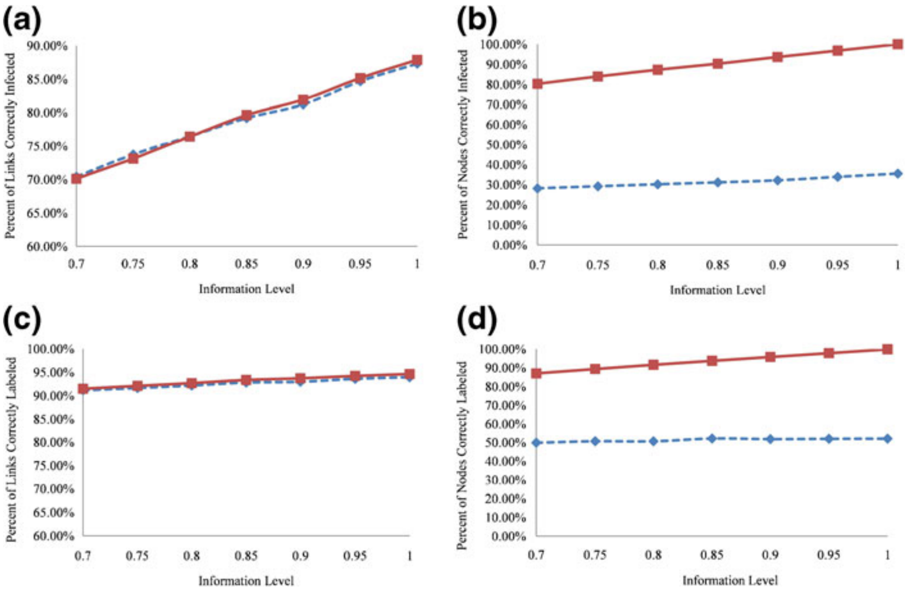

**Fig. 3** Percentage of correctly *infected* (**a**) links and (**b**) nodes, and percentage of correctly *labeled* (**c**) links and (**d**) nodes for power law network structure. The *dotted line* represents the *A_ILP* performance, and the *solid line* represents the heuristic performance

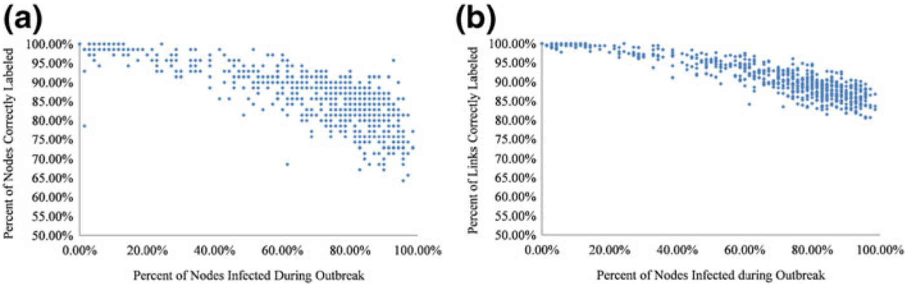

**Fig. 4** (**a**) Percent of correctly labelled nodes and (**b**) Percent of correctly labelled links by heuristic as a function of outbreak size for power law network structure under 70 % information availability

increasing the likelihood of infecting a large portion of the population. This behavior is illustrated by comparing Figs. 4 and 6 which display the outbreak size (as the percentage of nodes infected) for 1,000 independent simulations of an outbreak on power law and Poisson network structures, respectively. For the Poisson network many more of the simulations resulted in a higher number of infections than in the power law network.

Regarding the model performance, in Figs. 3, 5 and 7 the dotted line represents the performance of the solution from *A_ILP*, and the solid line represents the performance of the heuristic solution. Each of these figures represents *expected* performance for each of the four criteria, averaged over 1,000 iterations for each information level and network structure combination. From all three figures it is apparent that both the *A_ILP* and heuristic performance improves as the information availability increases for all performance measures, and that the heuristic substantially outperforms *A_ILP* in node-level prediction, but not link-level prediction.

The performance improvement with information is expected, and is more dramatic for the performance measures $\boldsymbol{Q}$ and $\boldsymbol{B}$, the percentage of correctly infected nodes and links, which only consider the accuracy of the model predictions across the set of infected nodes and links. The reason the performance improvement is greater for $\boldsymbol{Q}$ and $\boldsymbol{B}$ relative to the percentage of correctly *labeled* nodes and links is because on average only 20 % of links and 65 % of the nodes are included in the simulated infection tree for the power law network, therefore each correctly infected node and link will have a larger marginal impact for the performance measures $\boldsymbol{Q}$ and $\boldsymbol{B}$ in comparison to the performance measures which account for the entire network structure (i.e. percentage of correctly labeled nodes and links). The same behavior applies to all network structures; therefore the performance measures are not included for the Poisson and uniform networks. The correct labeling of the nodes and links (above 90 % for both measures) suggests the uninfected nodes (and links) are being appropriately excluded by the model.

Figure 3 also reveals the dominating performance of the heuristic relative to *A_ILP* at the node level, but not the link level. The overall node-level performance of the heuristic exceeds the link-level performance because it is possible for the heuristic to correctly predict the set of infected nodes without correctly predicting every infected link. Each network and contagion scenario will result in a set of infected nodes which may exhibit many feasible (link-level) infection patterns. As such, it is possible for the heuristic to correctly predict a higher percentage of infected nodes than infected links. The difference in node level and link level performance suggest that *A_ILP* does a poor job at identifying the set of infected nodes, but a better job at identifying the set of infected links; additionally, the heuristic is able to significantly improve upon the set of infected nodes identified by *A_ILP*, but is unable to significantly improve upon the set of links identified in the same solution. The node level performance gap between *A_ILP* and heuristic is due to the accuracy of the *node addition* step of the heuristic, until which the set of nodes is constrained to those included in the initial feasible tree. The set of nodes added in this step can represent a substantial percentage of the total infected nodes, often improving upon the *A_ILP* solution by 60 %. However the set of links selected to branch to the newly added nodes are not as accurately identified. This behavior is observed across network structures.

In Fig. 4 the performance of the heuristic is illustrated as a function of outbreak size, which is measured by the percentage of nodes infected during the outbreak. It is apparent from the figure that the performance of the heuristic decreases as outbreak size increases, which is an expected outcome of the model. For large outbreaks there are more possible spreading scenarios, therefore the actual scenario is more difficult to accurately predict. None the less, over 65 % of links and 80 % of nodes are correctly labeled for all cases, and for nearly all outbreaks which reach less than half the population, more than 85 % of nodes and more than 90 % of links are correctly labeled.

### *5.2.2 Poisson Network*

The second network structure evaluated was a Poisson network, which has a more homogenous structure, with most nodes having close to the average degree distribution. For this network structure an outbreak is likely to spread to more nodes quicker than in the power law network structure. In a Poisson network each infected node has more links on average than in a power law network, therefore the link level prediction is more difficult, and the performance is slightly lower for the Poisson network (Fig. 5a) relative to the power law network (Fig. 3a). However for the Poisson network the node level performance is slightly better for the heuristic and lower for *A_ILP*, relative to the power law network (Figs. 3b versus 5b). In Fig. 3 the same

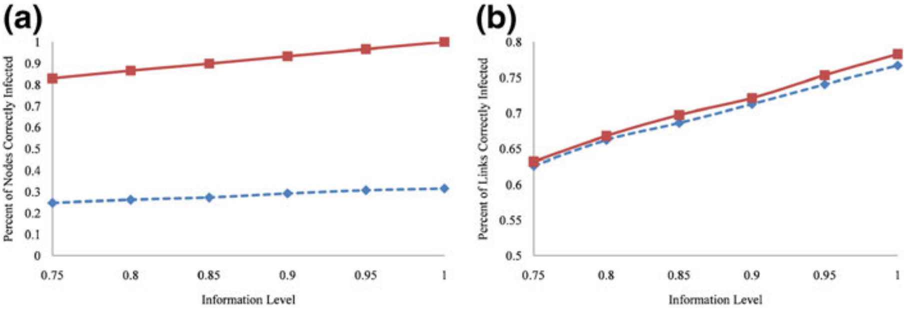

**Fig. 5** Percentage of correctly *infected* (**a**) nodes and (**b**) links for poisson network structure. The *dotted line* represents *A_ILP* performance, and the *solid line* represents the heuristic performance

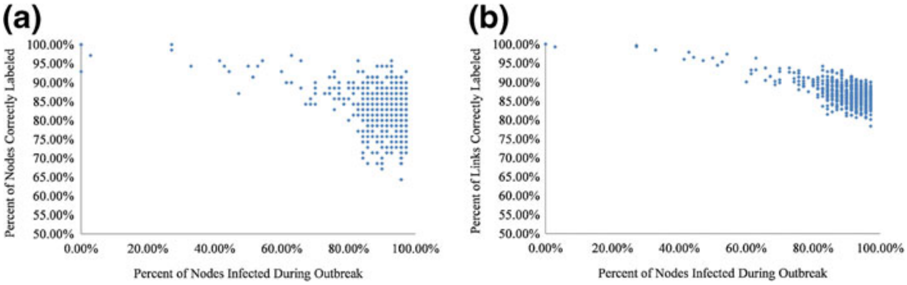

**Fig. 6** (**a**) Percent of correctly labeled nodes and (**b**) Percent of correctly labeled links by heuristic as a function of outbreak size for poisson network structure under 75 % information availability

trends can be observed for the Poisson network as with the power law network in terms of the performance improvement relative to information availability. In addition the same trends regarding the heuristic performance as a function of outbreak size are illustrated in Fig. 6. The main difference for the Poisson network is the larger number of scenarios which reached a larger percentage of the population, which are harder to accurately identify, thus the decreased performance.

### *5.2.3 Uniform Network*

The last network structure evaluated was a uniform network, where all nodes have the same average degree of 3. For the uniform network structure the percentage of population infected in each scenario is more evenly distributed relative to the Poisson and power law networks, resulting in smaller outbreaks which can be predicted with greater accuracy (see Fig. 8). It is however interesting to note that the larger outbreaks (greater than 80 % of the nodes infected) are also predicted more accurately for the uniform network structure, with 80 % of the nodes correctly labeled, relative to 70 % for Poisson networks. As the most homogenous of the networks evaluated, the uniform network resulted in improved heuristic performance at both the node and link level relative to the other network structures. In contrast *A_ILP* performance was poorest for the uniform network node prediction. Figure 7 illustrates a similar trend in model performance relative to information availability.

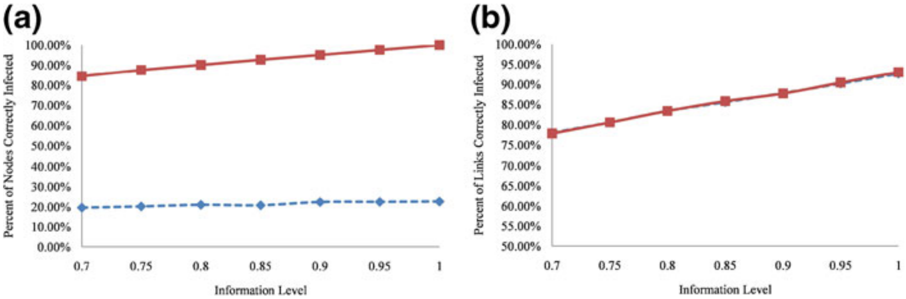

**Fig. 7** Percentage of correctly *infected* (**a**) nodes and (**b**) links for uniform network structure. The *dotted line* represents *A_ILP* performance, and the *solid line* represents the heuristic performance

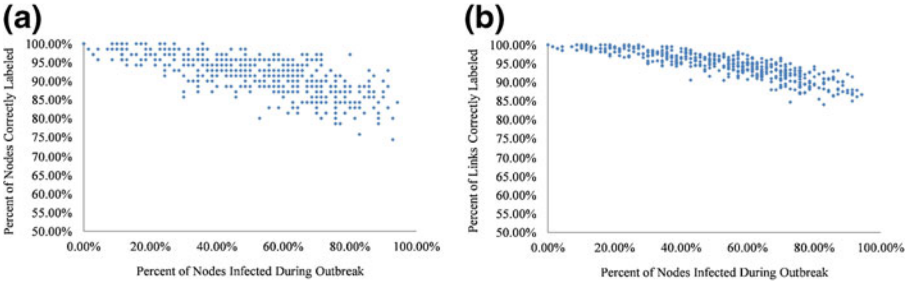

**Fig. 8** (**a**) Percent of correctly labeled nodes and (**b**) Percent of correctly labeled links by heuristic as a function of outbreak size for uniform network structure under 70 % information availability

## 6 Conclusions

The main focus of this research was to develop a formulation and solution method for inferring a contagion process in a social contact network where only a selection of the infections are known of. A heuristic solution method was proposed which exploits known properties of the problem, and reduces to solving various simpler subproblems. The proposed methodology provides a novel procedure for evaluating a region that has been exposed to infection (compared with the traditional methods of enumeration followed by a posteriori analysis). The performance of the proposed methodology was evaluated as a function of information availability and network structure, and the heuristic was shown to accurately identify infection spreading links and unreported infected nodes.

As expected, the performance of the heuristic decreased as the level of available information decreased. With 80 % of the infections reported, the heuristic correctly identified over 76 % of the infection spreading links and 86 % of the infected nodes (for the power law network, which is known to be the most representative of social contact network structures). Additionally, the heuristic continually outperformed *A_ILP* by over 60 % in node-level prediction.

The novelty of the model lies in the use of network optimization techniques, infection and contact data to infer spatiotemporal outbreak patterns, aiding in the development of real-time analysis and decision support for outbreak scenarios. The largest weakness with the proposed methodology is the lack of verifiability due to limited data availability. Without accurate social contact networks and link-level infection data to validate the model's performance, it is not possible to evaluate certain model characteristics. As such, one major motivation for this work is to incentivize better data collection efforts. For the proposed model, collecting contact-level infection data would be the most valuable. Contact level information requires data from infected individuals on their recent social interactions. Such link-level infection data is difficult to collect, but would permit quantitative analysis of the models' performance. Additionally research in the development of social networks which more accurately depict the set of social contacts in a region will be integral in the implementability of this research. One such approach is to exploit regional travel patterns to define a social network. By using regional travel patterns (such as origin–destination tables and activity-based travel patterns), individuals' daily trips, specific types of interaction, and length of interaction can be accounted for. With an accurate depiction of individuals daily contact patterns, the proposed model can be implemented to provide information on the set of contacts most likely responsible for spreading infection during an outbreak, aiding in the development of regional mitigation strategies.

## References

Anderson RM, May RM (1991) Infectious diseases of humans: dynamics and control. Oxford University Press, Oxford

Balcan D, Colizza V, Goncalves B, Hu H, Ramasco JJ et al (2009) Multiscale mobility networks and the spatial spreading of infectious diseases. Proc Natl Acad Sci USA 106:21484–21489

Balthrop J, Forrest S, Newman MEJ, Williamson MM (2004) Technological networks and the spread of computer viruses. Science 304(5670):527–529

Barabási A-L, Albert R (1999) Emergence of scaling in random networks. Science 286:509–512

Cahill E, Crandall R, Rude L, Sullivan A (2005) Space-time influenza model with demographic, mobility, and vaccine parameters. Proc. 5th Annual Hawaii Internat. Conf. Math., Statist., and related fields

Carley KM, Fridsma DB, Casman E, Yahja A, Altman N, Chen LC, Kaminsky B, Nave D (2006) BioWar: scalable agent-based model of bioattacks. Syst Man Cybern A: Syst Hum IEEE Trans 36(2):252–265

Coleman J, Menzel H, Katz E (1966) Medical innovations: a diffusion study. Bobbs Merrill, New York

Colizza V, Barrat A, Barthélemy M, Vespignani A (2006) The modeling of global epidemics: Stochastic dynamics and predictability. Bull Math Biol 68:1893–1921

Cottam et al (2008) Integrating genetic and epidemiological data to determine transmission pathways of foot-and-mouth disease virus. Proc R Soc B 275:887–895

Dibble C, Feldman PG (2004) The GeoGraph 3D computational laboratory: network and terrain landscapes for RePast. J Artif Soc Soc Simul 7(1)

Drummond AJ, Rambaut A (2007) Beast: Bayesian evolutionary analysis by sampling trees. BMC Evol Biol 7:214

Dunham JB (2005) An agent-based spatially explicit epidemiological model in MASON. J Artif Soc Soc Simul 9(1):3

Edmonds J (1967) Optimum branchings. J Res Nat Bur Stand 9:233–240

Ekici A, Keskinocak P, Swann JL (2008) Pandemic influenza response, Simulation Conference, 2008. WSC 2008. Winter, pp 1592–1600, doi:10.1109/WSC.2008.4736242

Epstein J, Cummings DAT et al. (2002) Toward a containment strategy for smallpox bioterror: an individual-based computational approach. Brookings Institute Press 2004 c. 55 pp

Erath A, Löchl M, Axhausen K (2009) Graph-theoretical analysis of the Swiss road and railway networks over time. Netw Spat Econ 9(3):379–400. doi:10.1007/s11067-008-9074-7

Eubank S, Guclu H et al (2004) Modeling disease outbreaks in realistic urban social networks. Nature 429:180–184

Ferguson NM, Cummings DAT, Fraser C, Cajka JC, Cooley PC, Burke DS (2006) Strategies for mitigating an influenza pandemic. Nature 442:448–452

Gardner L, Fajardo D, Waller ST (2012) Inferring infection spreading links in an air traffic network. In press for the J Transp Res Board

Gastner MT, Newman MEJ (2006) The spatial structure of networks. Eur Phys J B 49(2):247–252

Germann TC, Kadau K, Longini IM, Macken CA (2006) Mitigation strategies for pandemic influenza in the United States. Proc Natl Acad Sci 103(15):5935–5940

Gonzalez MC, Hidalgo CA, Barabasi A-L (2008) Understanding individual human mobility patterns. Nature 453:479–482

Hasan S, Ukkusuri SV (2011) A contagion model for understanding the propagation of hurricane warning information. Transp Res B (Methodological) 45(10):1590–1605

Haydon DT, Chase-Topping M, Shaw DJ, Matthews L, Friar JK, Wilesmith J, Woolhouse MEJ (2003) The construction and analysis of epidemic trees with reference to the 2001 UK foot-and-mouth outbreak. Proc R Soc B 270:121–127

Hoogendoorn SP, Bovy PHL (2005) Pedestrian travel behavior modeling. Netw Spat Econ 5(2):193–216

Hufnagel L, Brockmann D, Geisel T (2004) Forecast and control of epidemics in a globalized world. Proc Natl Acad Sci USA 101:15124

Illenberger J, Nagel K, Flötteröd G (2012) The role of spatial interaction in social networks. Netw Spat Econ. pp 1–28. doi:10.1007/s11067-012-9180-4

Jombart T, Eggo RM, Dodd P, Balloux F (2009) Spatiotemporal dynamics in the early stages of the 2009 A/H1N1 influenza pandemic. PLoS Curr Influenza. RRN1026

Kinney R, Crucitti P, Albert R, Latora V (2005) Modeling cascading failures in the North American power grid. Eur Phys J B 46(1):101–107

Lam WHK, Huang H (2003) Combined activity/travel choice models: time-dependent and dynamic versions. Netw Spat Econ 3(3):323–347

Lemey P, Suchard M, Rambaut A (2009) Reconstructing the initial global spread of a human influenza pandemic: a Bayesian spatial-temporal model for the global spread of H1N1pdm. PLoS Curr Influenza. doi:10.1371/currents.RRN1031

Meyers L, Pourbohloul B, Newman MEJ, Skowronski D, Brunham R (2005) Network theory and SARS: predicting outbreak diversity. J Theor Biol 232:71–81

Murray JD (2002) Mathematical biology, 3rd edn. Springer, New York

Newman MEJ, Forrest S, Balthrop J (2002) Email networks and the spread of computer viruses. Phys Rev E 66(3):035101

Ramadurai G, Ukkusuri S (2010) Dynamic user equilibrium model for combined activity-travel choices using activity-travel supernetwork representation. Netw Spat Econ 10(2):273–292. doi:10.1007/s11067-008-9078-3

Roche B, Drake J, Rohani P (2011) An agent-based model to study the epidemiological and evolutionary dynamics of influenza viruses. BMC Bioinforma 12(1):87, BioMed Central Ltd

Roorda MJ, Carrasco JA, Miller EJ (2009) An integrated model of vehicle transactions, activity scheduling and mode choice. Transp Res B 43(2):217–229. doi:10.1016/j.trb.2008.05.003

Rvachev L, Longini I (1985) A mathematical model for the global spread of influenza. Math Biosci 75:3–22

Sachtjen ML, Carreras BA, Lynch VE (2000) Disturbances in a power transmission system. Phys Rev E 61(5):4877–4882

Schintler L, Kulkarni R, Gorman S, Stough R (2007) Using raster-based GIS and graph theory to analyze complex networks. Netw Spat Econ 7(4):301–313. doi:10.1007/s11067-007-9029-4

Small M, Tse CK (2005) Small world and scale free model of transmission of SARS. Int J Bifurcations Chaos Appl Sci Eng 15:1745. doi:10.1142/S0218127405012776

Sornette D (2003) Why stock markets crash: critical events in complex financial systems. Princeton University Press, Princeton

Wallace RG, HoDac H, Lathrop RH, Fitch WM (2007) A statistical phylogeography of influenza A H5N1. PNAS 104(11):4473–4478