Netw Spat Econ (2016) 16:687–721
DOI 10.1007/s11067-015-9294-6

Finding Outbreak Trees in Networks with Limited Information

David Rey[1] · Lauren Gardner[1] · S. Travis Waller[2]

Published online: 17 April 2015

Abstract Real-time control of infectious disease outbreaks represents one of the greatest epidemiological challenges currently faced. In this paper we address the problem of identifying contagion patterns responsible for the spread of a disease in a network, which can be applied in real-time to evaluate an ongoing outbreak. We focus on the scenario where limited information, *i.e.* infection reports which may or may not include the actual source, is available during an ongoing outbreak and we seek the most likely infection tree that spans at least a set of known infected nodes. This problem can be represented using a maximum likelihood constrained Steiner tree model where the objective is to find a spanning tree with an assignment of integer nodes weights. We propose a novel formulation and solution method based on a two-step heuristic which (1) reduces the initial graph using a polynomial time algorithm designed to find feasible infection paths and (2) solves an exact mixed integer linear programming reformulation of the maximum likelihood model on the resulting subgraph. The proposed methodology can be applied to outbreaks which may evolve from multiple sources. Simulated contagion episodes are used to evaluate the

✉ David Rey
d.rey@unsw.edu.au

Lauren Gardner
l.gardner@unsw.edu.au

S. Travis Waller
s.waller@unsw.edu.au

[1] School of Civil and Environmental Engineering, UNSW Australia, Sydney, Australia

[2] School of Civil and Environmental Engineering, UNSW Australia and NICTA, Sydney, Australia

performance of our solution method. Our results show that the approach is computationally efficient and is able to reconstruct a significant proportion of the outbreak, even in the context of low levels of information availability.

Keywords Contagion patterns · Social contact networks · Network optimization · Integer programming · Shortest path

1 Introduction

Infectious diseases pose an increasing risk to humans due to a growing world population, increasingly dense urban environments, and highly connected global transport systems which together provide the necessary groundwork for a global epidemic. Over the past decades copious research efforts have contributed to the development of contagion models for predicting the expected spreading behavior of infectious diseases by exploiting population demographics, human travel patterns, social interactions and properties of the disease itself. These models are used to assess various disease prevention, intervention and response strategies. The same models however, are unable to reconstruct the contagion process of an ongoing outbreak in order to reveal the spatiotemporal transmission patterns within a given population. We address this gap in the literature with the development of an optimization based solution method designed to identify the most likely infection path of a disease in a network. In particular, we focus on the scenario in which only a limited amount of the infection-related information is available, *i.e.* we assume that the infection status of only a subset of the population is known, which is often the case in real-world outbreaks.

In this paper, we propose a new combinatorial approach to quantify the likelihood of possible disease transmission patterns in social contact networks. Our solution method utilizes the network topology (*e.g.* nodes, links), estimated disease parameters (*e.g.* transmission probabilities, infectious period) and available infection reports (*e.g.* node status information, time of infection) to evaluate a region that has been exposed to infection. Our objective is to find the set of links spanning to all known infected nodes which maximizes the likelihood of the infection pattern. We formulate this maximum likelihood problem as an Integer Program (IP) and introduce a novel model which improves upon previous works by relaxing the assumption that the source(s) of the infection are known; and propose a tailored heuristic algorithm to reduce the solution search space. From a mathematical perspective, the problem considered in this study is closely related to a constrained Steiner tree problem, where a directed tree with an assignment of integer node weights is sought. The solution method developed is based on a k-shortest path algorithm which attempts to find feasible infection paths between known infected nodes and then solve an exact reformulation of the IP into a Mixed-Integer Linear Program (MILP) on the subgraph induced by the collection of such paths. The performance of the solution method is measured by quantifying its ability to accurately identify the observed infection pattern. The solution method is shown to be able to reveal a significant portion of the set of links and nodes responsible for having spread the

disease. Furthermore, the methodology can account for missing infection information, enabling epidemiologists to better understand and anticipate disease transmission patterns during an ongoing outbreak and offer insights into the success of outbreak control measures.

We start by reviewing the literature on the modelling of contagion processes and on the optimization problems that unfold in our approach (Section 2). We then introduce the epidemiological context of this research as well as the mathematical formulation of the maximum likelihood model (Section 3). The solution method is presented in two steps (Section 4): (1) the mathematical properties of feasible infection paths are assessed and a polynomial time graph reduction algorithm is presented and (2) an exact MILP reformulation of the initial IP is proposed to improve computational performance. Network topology and disease parameters are introduced in the validation framework (Section 5) and the results obtained after the implementation of our solution method are then presented (Section 6). Finally, the contributions of this research is discussed and summarized (Section 7).

2 State of the Art

In this section we review the literature on epidemic modelling and discuss the position of this work with regards to network optimization problems.

2.1 Contagion Processes Modelling

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 (Anderson and May 1991; Murray 2002), the spread of information, ideas and opinions via social networks can also be modelled 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 (Sachtjen et al. 2000; Kinney et al. 2005); and the collapse of financial systems (Sornette 2003).

Another emerging area of research is the spatial analysis of networks, such as transport and communication networks (Gastner and Newman 2006; Schintler et al. 2007; Erath et al. 2009), including social network modelling, specifically the ability to reproduce spatial structure and interaction between individuals for large-scale social networks (Illenberger et al. 2013). 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. Wesolowski et al. (2014) highlight the value of mobile network data for modelling human mobility patterns in real time, the results of which can be used in designing surveillance and containment strategies during

an outbreak. The authors constructed intra and international mobility patterns in the set of African countries affected by and surrounding the Ebola outbreak. Their work exemplifies the role of increasingly available real-time connectivity data in public health response. However, in order to exploit these data sets to their full potential in disease mitigation and control efforts, real-time outbreak prediction models, such as that proposed in this paper, must be developed.

Published contagion prediction 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 (V C and M B 2006; D B et al. 2009; Broeck et al. 2011); 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 (Cummings et al. 2002; DT et al. 2003; Eubank et al. 2004; Dunham 2005; Ferguson et al. 2006; Roche et al. 2011). 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 modelling 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' migration (DT et al. 2003; AJ D 2007; Wallace et al. 2007; P et al. 2009). 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. Luo et al. (2013) developed a solution method to identify infection sources and regions in networks based on probabilistic estimators. They approximate an infection source estimator for the class of geometric trees and derive an algorithm to estimate the actual number of sources of infection and their identities. Jombart et al. (2009) proposed an innovative 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. 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 timeframe. The proposed approach relies instead on available infection reports, network topology and disease properties to infer the spatiotemporal path of infection through a network.

2.2 Related Network Optimization Problems

Using infection data to reconstruct the infection tree of a contagion process can be approached with mathematical optimization techniques. Indeed, identifying the most likely infection links responsible for the spread of a disease in a network is closely related to finding the maximum cost spanning tree (Graham and Hell 1985) to all infected individuals, where the cost on the links represents the likelihood of the link to have spread the disease. Gardner L M et al. (2014) proposed an efficient solution method for this problem when all the infected individuals are known which is based on an optimum branching strategy. In Gardner et al. (2012), a novel 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, and the network structure was defined by the air traffic system. In the context of epidemic modelling, it is seldom that the status of all infected individuals is known; hence it is critical to address the scenario where only partial information on the population is available. This more general problem can be represented as a Steiner Tree Problem (STP) (Hwang and Richards 1992), where Steiner nodes represent individuals whose infection status is unknown, which is NP-Hard (Garey and Johnson 1977).

Infection patterns of disease spreading processes are dependent on the parameters of the disease (exposed and infectious periods, transmission probabilities), therefore feasibility challenges arise in the search for valid infection paths. As such, the reconstruction of contagion processes with limited information is structurally similar to a constrained STP. The resource constrained STP has been introduced by Rosenwein and Wong (1995); in their formulation, the authors attribute a resource for each link in the network and ask for the minimum cost Steiner tree such that the total amount of resources used in the tree does not exceed a given threshold. The authors discuss the efficiency a Lagrangian relaxation versus a Lagrangian decomposition of the problem. Voss (1999) focused on the hop-constrained STP, which can be seen as special case of the resource constrained STP where every resource is set to a unit cost, and presented a dynamic tabu search heuristic. Several new techniques have emerged to address the resource-constrained and the diameter-constrained STP (Gouveia and Magnanti 2003; Santos et al. 2010; Gouveia et al. 2011), in these formulations the resource constraint is imposed on the entire tree. In contrast, we require a Steiner tree and an assignment of integer nodes weights to Steiner nodes such that each path from a root node to a leaf of the tree respects some feasibility constraints.

In this study, we address the problem of finding the Most Likely Infection Tree (MLIT) that spans to all known infected nodes in a network where the available information (*e.g.* node status and time of infection) enforces timestamp constraints on the infection tree. Fajardo and Gardner (2013) designed a heuristic approach to solve a relaxed version of the MLIT. We extend that line of research by introducing a new solution method that is based on a polynomial-time graph reduction algorithm and an exact MILP reformulation of the initial model. In the next section we present the epidemiological context of this study and the mathematical formulation of the maximum likelihood model.

3 Problem Formulation

In this section, we first define the mathematical problem of interest using graph theoretical concepts. We then describe how this combinatorial problem can be matched to the epidemiological problem of finding the most likely infection tree responsible for the spread of a disease in a network with limited information and introduce a mathematical programming model to represent this problem.

Table 1 provides a summary of the acronyms and the mathematical notation used throughout the paper. Note that acronyms and symbols are defined in the paper when necessary. Note that throughout the paper, we use the word "source(s)" to refer to the actual individual(s) responsible for the initial introduction of the disease within the population (*i.e.* first case(s)) and the word "root(s)" to refer to origin nodes of infection paths which are used in the proposed solution method.

3.1 Mathematical Definition

We assume a directed weighted graph and seek a directed tree in this graph which spans at least a subset of the nodes, and an assignment of integer weights to the nodes in the tree. This directed tree should maximize a link-based likelihood function and respect link-feasibility constraints. The combinatorial problem of interest is therefore similar to a constrained STP and is formally defined as follows.

Definition 1 (Maximum Likelihood Constrained Steiner Tree Problem) Given a graph $G = (N, A)$ with a subset of nodes $I \subseteq N$, we seek to find a directed Steiner tree spanning the nodes in I and an assignment of integer node weights $t_i, \forall i \in N$ such that the relative weight of each link in the tree $\Delta t_{ij} = t_j - t_i$ is feasible, *i.e.* appropriately lower and upper bounded. The objective function of this problem is to maximize the likelihood function $\prod_{(i,j)\in A} \lambda_{ij}$ where for each link $(i, j) \in A$ λ_{ij} is a function of the relative weight Δt_{ij}.

We next show how this maximum likelihood constrained STP can be adapted to represent the problem of identifying the most likely infection tree in an epidemiological context.

3.2 Epidemiological Model

We are concerned with the problem of reconstructing the path of infection of a disease in a network given that its source(s) is unknown and its spread can be represented by a stochastic process. To reproduce the spread of infectious diseases in networks, we use a generic compartmental model. The Susceptible-Exposed-Infectious-Recovered (SEIR) model (Anderson and May 1991) is a well-established stochastic simulation model used in the epidemiological literature to model the progress of an epidemic in a large population. The SEIR model considers a fixed population of individuals which can be broken into four compartments:

Table 1 Acronyms and Mathematical Notation

MLIT	Most Likely Infection Tree
SP	Shortest Path
FIP	Feasible Infection Path
SFIP	Shortest Feasible Infection Path
STP	Steiner Tree Problem
SEIR	Susceptible-Exposed-Infected-Recovered
OD	Origin-Destination
IP	Integer Program
MILP	Mixed Integer Linear Program
L	Disease exposition period (in timesteps)
D	Disease infection period (in timesteps)
N	Set of nodes
A	Set of arcs
G	Graph $G = (N, A)$
I	Set of information nodes: $I = I_i \cup I_n$
I_i	Set of infected information nodes
I_e	Set of earliest infected information nodes: $I_e \subseteq I_i$
I_n	Set of non-infected information nodes
R_s	Set of possible roots for leaf node $s \in I_i$
K	Maximum number of FIP per OD pair
$\mathscr{P}_{rs}$	Path from node r to node s
$\underline{H}_{rs}$	Minimum number of hops for OD pair (r, s)
$\overline{H}_{rs}$	Maximum number of hops for OD pair (r, s)
$\mathscr{T}^{\star}$	Optimal oubreak tree
$\mathscr{T}^{obs}$	Observed oubreak tree
p_{ij}	Probability of node i infecting node j
λ_{ij}	Likelihood of node i infecting node j
T_i	Known timestamps of information node $i \in I$
t_i	Integer variable representing the timestamp of node i
Δt_{ij}	Relative timestamp variable for arc (i, j)
x_{ij}	Binary variable equal to 1 if arc $(i, j) \in \mathscr{T}^{\star}$ and 0 otherwise

- S (Susceptible): the individuals are susceptible to the disease and have never been infected.
- E (Exposed): the individuals have been infected but are not yet contagious. We assume that individuals remain in this compartment for a period of L time steps.
- I (Infectious): the individuals have been infected and are contagious. We assume that individuals remain in this compartment for a period of D time steps.
- R (Recovered): the individuals have been infected and are now recovered or removed. These individuals are not capable of spreading the disease.

The flow of the SEIR model can then be represented as

$$\mathrm{S} \rightarrow \mathrm{E} \rightarrow \mathrm{I} \rightarrow \mathrm{R} \tag{1}$$

Both L and D are parameters in the model expressed in units of time and are assumed to be disease-specific. L represents the incubation period, or the minimum number of time steps before a newly infected individual may infect a susceptible individual. At each time step, every infectious individual attempts to infect its neighbors in the network. The maximum number of infection trials between any two individuals is bounded by parameter D which represents the infectious period. The SEIR model assumes that individuals can only be infected by a single other individual, hence the topology of the infection induced by this model is a tree. Furthermore, we assume that an individual cannot be infected and become infectious at the same time step.

Our objective is to reconstruct the spreading pattern of an outbreak in a network where only limited information is available, *i.e.* where only a subset of the population's status is known. This approach is motivated by the more realistic setting in which only a subset of infected individuals report to public health authorities, whether due to limited medical accessibility or asymptomatic cases.

3.3 Problem Notation and Information Availability

To match this epidemiological problem to the maximum likelihood constrained STP, we assume that every node $i \in N$ represents an individual; every link $(i, j) \in A$ is a relationship among two individuals and that each link is weighted by a disease transmission probability. Namely, $p_{ij}, \forall (i, j) \in A$ is the probability that individual i infects j at each time period. Assuming that time can be discretized, the integer node weights $t_i, \forall i \in N$ represent the time of infection (timestamp) of each node in the network.

Using the terminology of the SEIR compartmental model, the state (e.g. infected, recovered) of a subset of individuals $I \subseteq N$ is assumed known, and for the known infected individuals their timestamp is also assumed known. These nodes are hereby referred to as *information* nodes. If an individual $i \in N$ has been infected by the disease (e.g. i is in state E,I or R), its timestamp is represented by a fixed integer weight $T_i \in \mathbb{Z}$, else if i has never been infected, we set $T_i \rightarrow \infty$. In turn, any node in $N \setminus I$ may or may not have been infected, this subset is referred to as the set of *zero-information* nodes and is equivalent to the set of Steiner nodes.

Let $I_i \subseteq I$ be the set of infected information nodes and let $I_n \subseteq I$ be the set of non-infected information nodes ($I_i \cup I_n = I$). Let $t_i \in \mathbb{Z}$ be a decision variable representing the timestamp of individual $i \in N$; the value of t_i depends on the available information, namely:

$$\forall i \in N, \qquad t_i \equiv \begin{cases} T_i \in \mathbb{Z} & \text{if } i \in I_i \\ T_i \rightarrow \infty & \text{if } i \in I_n \\ \in \mathbb{Z} & \text{otherwise} \end{cases} \tag{2}$$

Hence if $i \in N \setminus I$, t_i is represented by an integer decision variable. The proposed model works from a set of information nodes and seeks to reconstruct the spread of the disease from the earliest known infected nodes to the latest known infected

nodes. We define the set of earliest infected information nodes $I_e \subseteq I_i$ as: $I_e \equiv \text{argmin}_{i \in T_i}\{T_i\}$. Without any loss of generality we can set $T_i = 0, \forall i \in I_e$ and define $T = \max_{i \in I_i}\{T_i\}$ as the last known timestamp. We next introduce the probabilistic inference model used to estimate the likelihood of an infection tree.

3.4 Maximum Likelihood Model

To find the MLIT of an outbreak in a network, we seek to determine the probability that a node has been infected by its neighbor during the spread of the disease. We can extend the definition of feasible link to the epidemiological context presented in Section 3.2; we define a feasible infection link as follows:

Definition 2 (Feasible Infection Link) Let $(i, j) \in A$, be a link of the network. (i, j) is a feasible infection link if and only if

$$L \leqslant \Delta t_{ij} \leqslant L + D - 1 \tag{3}$$

Equation 3 states that node j may have been infected by i only if their timestamp difference is greater than L or if it is lower than $L + D - 1$, which corresponds to an interval of $D - 1$ time steps. Recall that we assume that that a node cannot be infected and infect an adjacent node at the same time step, which is equivalent to assume that $L \geqslant 1$. Using the link transmission probability p_{ij}; the probability α_{ij} that j is infected by i is then

$$\alpha_{ij} = p_{ij}(1 - p_{ij})^{(\Delta t_{ij} - L)^+} \tag{4}$$

where $(X)^+ \equiv \max\{X, 0\}$. $(\Delta t_{ij} - L)^+$ is the number of unsuccessful trials between nodes i and j given that eventually one trial is successful. To account for the event that a node is not infected during the outbreak, we introduce the associated probability γ_{ij} as

$$\gamma_{ij} = (1 - p_{ij})^{\min\{D, (\Delta t_{ij} - L + 1)^+\}} \tag{5}$$

where $\min\{D, (\Delta t_{ij} - L + 1)^+\}$ is the maximum number of unsuccessful infection trials between nodes i and j. In order to account for both probabilities in the model, we combine α_{ij} and γ_{ij} in a single expression. As α_{ij} and γ_{ij} are complementary, that is, only one of these events can occur; we introduce a binary decision variable x_{ij} to model this relationship. Let $\mathscr{T}^\star$ be the optimal infection tree, we define

$$\forall (i, j) \in A, \qquad x_{ij} \equiv \begin{cases} 1 & \text{if } (i, j) \in \mathscr{T}^\star \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

x_{ij} are the main decision variables in the model as they define the resulting infection tree $\mathscr{T}^\star$. We define the likelihood of the infection tree as the product of probabilities α_{ij} and γ_{ij} over all the links in the network. Our objective is to maximize this likelihood, hence we introduce the likelihood function $\lambda_{ij}(x_{ij}, \Delta t_{ij})$ defined as

$$\lambda_{ij}(x_{ij}, \Delta t_{ij}) \equiv \alpha_{ij}^{x_{ij}} \gamma_{ij}^{(1 - x_{ij})} \tag{7}$$

Let $\Gamma^+(i)$ and $\Gamma^-(i)$ be the sets of successors and predecessors of node $i \in N$ in graph G, respectively. To find the MLIT of an outbreak, we propose the IP summarized in Model 1. Objective function (8) maximizes the likelihood that the set of links (i, j) for which $x_{ij} = 1$ infers the correct infection tree. Constraints (9), (10) and (11) are the *feasibility constraints* and require that each link in the tree is a feasible infection link as defined by Eq. 3. Constraints (12) to (15), hereby referred to as the *tree constraints* ensure that $\mathscr{T}^\star$ is a directed tree spanning at least all infected information nodes. Constraints (16) and (17) enforce that the timestamps of information nodes are fixed; whereas constraints (18) and (19) ensure that the timestamp of zero-information nodes is in the range $[0, T]$. These constraints are hereby referred to as the *timestamps constraints*.

Model 1 (IP for the MLIT)

$$\max \prod_{(i,j)\in A} p_{ij}^{x_{ij}} (1 - p_{ij})^{x_{ij}(\Delta t_{ij} - L)^+} (1 - p_{ij})^{(1-x_{ij}) \min\{D, (\Delta t_{ij} - L + 1)^+\}} \tag{8}$$

subject to

$$x_{ij}(\Delta t_{ij} - L) \geqslant 0 \qquad \forall (i, j) \in A \tag{9}$$

$$x_{ij}(\Delta t_{ij} - D - L + 1) \leqslant 0 \qquad \forall (i, j) \in A \tag{10}$$

$$\Delta t_{ij} = t_j - t_i \qquad \forall (i, j) \in A \tag{11}$$

$$\sum_{i \in \Gamma^-(j)} x_{ij} = 1 \qquad \forall j \in I_i \setminus R \tag{12}$$

$$\sum_{i \in \Gamma^-(j)} x_{ij} = 0 \qquad \forall j \in I_n \tag{13}$$

$$\sum_{i \in \Gamma^-(j)} x_{ij} \leqslant 1 \qquad \forall j \in N \setminus I \tag{14}$$

$$\sum_{i \in \Gamma^-(j)} x_{ij} \geqslant x_{jk} \qquad \forall j \in N \setminus R, k \in \Gamma^+(j) \setminus R \tag{15}$$

$$t_i = T_i \qquad \forall i \in I_i \tag{16}$$

$$t_i = T \qquad \forall i \in I_n \tag{17}$$

$$t_i \geqslant T \left(1 - \sum_{j \in \Gamma^-(i)} x_{ji}\right) \qquad \forall i \in N \setminus I \tag{18}$$

$$t_i \leqslant T \qquad \forall i \in N \setminus I \tag{19}$$

$$t_i \in \mathbb{Z}^+ \qquad \forall i \in N \tag{20}$$

$$\Delta t_{ij} \in \mathbb{Z} \qquad \forall (i, j) \in A \tag{21}$$

$$x_{ij} \in \{0, 1\} \qquad \forall (i, j) \in A \tag{22}$$

Proposition 1 (Subtour Elimination) *If $L \geqslant 1$, then Model 1 produces an optimal directed tree without any subtours.*

Proof The proof follows from the definition of feasible infection link. Assume by contradiction that the links (1, 2), (2, 3) and (3, 1) form a subtour in the optimal tree, *i.e.* $x_{12} = x_{23} = x_{31} = 1$. Constraints (9) and (10) impose that:

$$L \leqslant \Delta t_{12} \leqslant D + L - 1 \tag{23}$$

$$L \leqslant \Delta t_{23} \leqslant D + L - 1 \tag{24}$$

$$L \leqslant \Delta t_{31} \leqslant D + L - 1 \tag{25}$$

Since $\Delta t_{ij} = t_j - t_i$, $t_2 \geqslant t_1 + L$ and $t_3 \geqslant t_2 + L$, hence $t_3 \geqslant t_1 + 2L$. Similarly $t_1 \geqslant t_3 + L$, which leads to a contradiction if $L \geqslant 1$. □

Model 1 can be used to infer the MLIT for outbreaks with any number of sources of infection since no information on the source(s) of the infection is assumed. If every node of the network is an information node, that is $I = N$, solving Model 1 is equivalent to finding the most likely spanning tree to all infected individuals. In this scenario, all variables t_i are fixed, hence x_{ij} are the only decision variables. It should be noted that instances can be pre-processed: non-infected information nodes $i \in I_n$ as well as infeasible links between two infected information nodes $i, j \in I_i$ can be removed from the network to improve computational performance. Nevertheless, in the general case, *i.e.* in the presence of zero-information nodes, the problem represented by Model 1becomes intractable on large instances. Furthermore, this maximum likelihood can rapidly lead to scenarios where multiple optimal trees with the same likelihood coexist, as illustrated in Example 1.

Example 1 Consider the network depicted by Fig. 1, which represents a partial information case study. In this example, each link has a specified transmission probability: nodes A and D are known to be infected with corresponding timestamps of 0 and 3, respectively, and nodes B and C are zero-information nodes. The epidemiological parameters are $L = 1$ and $D = 2$. For this simple network structure and information scenario there are twenty feasible infection trees. More importantly, for a range of specified transmission probabilities there are multiple trees which have equal and maximum likelihoods. For instance, for the set of transmission probabilities $(p_1, p_2, p_3, p_4, p_5) = (0.2, 0.3, 0.5, 0.2, 0.3)$, there are four equally likely and optimal infection trees with a likelihood of 0.05136. These four trees and their likelihood are detailed in Fig. 2. Recall that for link l, p_l represents a successful infection trial whereas $(1 - p_l)$ represents an unsuccessful trial. This example shows that even on simple networks several optimal solutions may coexist.

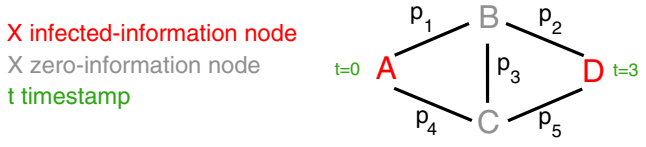


Fig. 1 An example network and information scenario

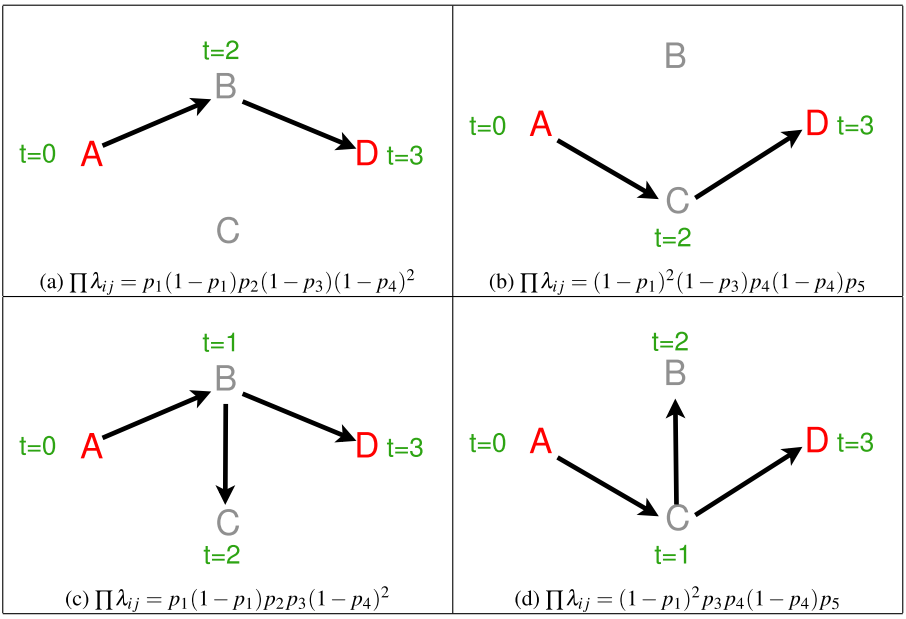


Fig. 2 Multiple maximum likelihood trees for the example network and information scenario depicted in Fig. 1

The presence of multiple optimal trees may significantly burden the resolution of Model 1 as global optimality can be computationally expensive to prove. To solve this maximum likelihood problem on large networks we propose an efficient heuristic to reduce the size of the initial graph and present an exact mixed integer linear reformulation of Model 1. The resulting solution method is then implemented and evaluated on multiple outbreak scenarios.

4 Solution Method

In this section, we present a new solution method to solve Model 1. Due to the potentially large number of Feasible Infection Path (FIP) between any two infected information nodes in networks with limited information, we propose to restrict the search to a set of good candidates. To do so, we stress the impact of the hop-distance on the likelihood of a FIP in the objective function. The hop-distance between two nodes in an unweighted graph can be defined as the number of links in the Shortest Path (SP) connecting them. While the length of a path is not the only criterion to evaluate its likelihood, it is reasonable to assume that the longest FIP is most certainly not the most likely. This hypothesis is driven by the fact that the contribution of each link in the objective function is determined by a product of probabilistic quantities. Hence an increase in the hop-distance of a FIP will most likely reduce the likelihood of the path in objective function (8). To find a FIP, we proceed from the set of information infected nodes I_i and for each leaf $s \in I_i$ we identify a set of possible

roots $R_s \subseteq I_i$ which may have been responsible for the spread of the infection to this leaf. Our solution method is based on this heuristic approach and can be decomposed into two main steps:

1. for each leaf node and for each possible root node for this leaf, find (at most) K shortest FIP for this Origin-Destination (OD) pair using a polynomial time constrained shortest path algorithm;
2. solve an exact linearization of Model 1 on the subgraph formed by all the FIPs found at Step 1 using a generic Branch & Bound & Cut algorithm for MILPs.

We next present the details of each step of the proposed solution method.

4.1 Graph Reduction Algorithm

We first formally extend the concept of link-feasibility to path-feasibility.

Definition 3 (Feasible Infection Path) Let $\mathscr{P}_{rs}$ be a path from node r to node s, path $\mathscr{P}_{rs}$ is a Feasible Infection Path (FIP) if

$$\forall (i,j) \in \mathscr{P}_{rs}, \qquad L \leqslant t_j - t_i \leqslant L + D - 1 \tag{26}$$

In the context of outbreaks with limited available information, a link from or to a zero-information node is potentially feasible, hence it is not straightforward to evaluate the feasibility of an infection path from or to a zero-information node. However the feasibility of an infection path between an OD pair, *i.e.* infected information nodes, can be decided efficiently. We first consider the elementary case where a subpath between two infected information nodes is composed of zero-information nodes only.

Proposition 2 (Subpath Feasibility) *Let* $\mathscr{P}_{rs} = \{(r, z_1), (z_1, z_2) \ldots, (z_l, s)\}$ *be a subpath between* $r, s \in I_i$ *composed of zero-information nodes* $z_i \in N \setminus I$ *only, as depicted by Fig. 3.* $\mathscr{P}_{rs}$ *is a FIP if and only if*

$$L|\mathscr{P}_{rs}| \leqslant T_s - T_r \leqslant (L + D - 1)|\mathscr{P}_{rs}| \tag{27}$$

where $|\mathscr{P}_{rs}|$ *is the number of links in path* $\mathscr{P}_{rs}$.

Proof If $\mathscr{P}_{rs}$ is a FIP, then by construction $\forall (i,j) \in \mathscr{P}_{rs}, L \leqslant t_j - t_i \leqslant L + D - 1$. Hence

$$\underbrace{L + \cdots + L}_{|\mathscr{P}_{rs}| \text{ times}} \leqslant T_s - t_{z_l} + t_{z_l} + \cdots - t_{z_1} + t_{z_1} - T_r \leqslant \underbrace{(L + D - 1) + \cdots + (L + D - 1)}_{|\mathscr{P}_{rs}| \text{ times}} \tag{28}$$

Fig. 3 Infection path between a pair of infected information nodes containing only zero-information nodes

which is equivalent to inequality (27). Reciprocally, since any link $(i, j) \in \mathcal{P}_{rs}$ is potentially feasible, inequality (27) can be broken into $|\mathscr{P}_{rs}|$ valid inequalities

$$L \leqslant t_{z_1} - T_r \leqslant L + D - 1 \tag{29}$$

$$\cdots$$

$$L \leqslant T_s - t_{z_l} \leqslant L + D - 1 \tag{30}$$

Hence every link in $\mathscr{P}_{rs}$ is a feasible infection link and $\mathscr{P}_{rs}$ is a FIP. □

To evaluate the feasibility of an infection occurring between any two infected information nodes, infection paths can be decomposed into subpaths containing zero-information nodes only.

Proposition 3 (Path Feasibility) *Let $\mathscr{P}_{rs}$ be a path with $r, s \in I_i$ as depicted by Fig. 4. $\mathscr{P}_{rs}$ is a FIP if and only if every subpath between any two infected information nodes in $\mathscr{P}_{rs}$ is a FIP.*

Proof If $\mathscr{P}_{rs}$ is a FIP, then by construction any link in $\mathscr{P}_{rs}$ is feasible, hence any subpath $\mathscr{P}_{i_j i_k} \subseteq \mathscr{P}_{rs}$ where $i_j, i_k \in I_i$ is a FIP. Reciprocally, let $\{r, i_1, \ldots, i_l, s\}$ be the set of infected information nodes contained in $\mathscr{P}_{rs}$. If the subpaths $\mathscr{P}_{ri_1}, \ldots, \mathscr{P}_{i_l s}$ are FIP, then by Proposition 2 we have

$$L|\mathscr{P}_{ri_1}| \leqslant T_{i_1} - T_r \leqslant (L + D - 1)|\mathscr{P}_{ri_1}| \tag{31}$$

$$\cdots$$

$$L|\mathscr{P}_{i_l s}| \leqslant T_s - T_{i_l} \leqslant (L + D - 1)|\mathscr{P}_{i_l s}| \tag{32}$$

Summing these inequalities yields

$$L(|\mathscr{P}_{ri_1}| + \cdots + |\mathscr{P}_{i_l s}|) \leqslant T_s - T_r \leqslant (L + D - 1)(|\mathscr{P}_{ri_1}| + \cdots + |\mathscr{P}_{i_l s}|) \tag{33}$$

where by construction $|\mathscr{P}_{ri_1}| + \cdots + |\mathscr{P}_{i_l s}| = |\mathscr{P}_{rs}|$. □

To reduce the size of graph G, we propose to find at most K FIP per OD pair and regroup these paths into a subgraph. This graph reduction heuristic is inspired by the efficient algorithms available to solve the length (hop-distance) constrained SP (Saigal 1968) and the k loopless SP problem (Yen 1971). Namely, our approach works in two stages; for each OD pair (leaf, root): (1) find the shortest FIP from the leaf node to the root node if it exists, and (2) find the next $K - 1$ shortest FIP for this OD pair if they exist. Note that due the limited information available, it is possible that no FIP exists for an OD pair: in this case, the leaf node cannot be connected to any other known infected node and will remain disconnected in the resulting subgraph. Isolated infected information nodes will lead Model 1 to produce partial

Fig. 4 Infection path between a pair of infected information nodes

outbreak trees in which not all known infected nodes are covered. The first stage can be achieved using a recursive procedure which re-labels nodes until they meet the feasibility constraint. The second stage requires to search for feasible deviations from the shortest FIP and can be done using a k-SP like procedure.

From Proposition 3, we know that an infection path between $r, s \in I_i$ is feasible if and only if $L|\mathscr{P}_{rs}| \leqslant T_s - T_r \leqslant (L + D - 1)|\mathscr{P}_{rs}|$, hence the length of the candidate paths is bounded by

$$\underline{H}_{rs} = \left\lfloor \frac{T_s - T_r}{L + D - 1} \right\rfloor \leqslant |\mathscr{P}_{rs}| \leqslant \left\lceil \frac{T_s - T_r}{L} \right\rceil = \overline{H}_{rs} \tag{34}$$

$\underline{H}_{rs}$ and $\overline{H}_{rs}$ are the minimum and maximum feasible number of hops for path $\mathscr{P}_{rs}$, respectively. Hence to find the shortest FIP from r to s, we need to find the SP such that $|\mathscr{P}_{rs}| \geqslant \underline{H}_{rs}$. Saigal (1968) introduced a dynamic programming algorithm – later revised by Rosseel (1968) – to find the SP of a given length. Using his notation, for each pair of nodes, let $c(i, j) = 1$ if $(i, j) \in A$ and let $c(i, j) = \infty$ otherwise. Let $C_{rs}(h)$ be the length of the SP from r to s with h hops and let $\mathscr{P}_{rs}(h)$ be this path. The procedure is initialized as follows:

$$C_{rs}(1) = \begin{cases} 1 & \text{if } (r, s) \in A \\ \infty & \text{otherwise} \end{cases} \tag{35}$$

$$\mathscr{P}_{rs}(1) = \begin{cases} \{(r, s)\} & \text{if } (r, s) \in A \\ \emptyset & \text{otherwise} \end{cases} \tag{36}$$

and the recursive formulation (with unit link weights) for every $h \geqslant 2$ can be summarized as

$$C_{rj}(h + 1) = \min_{i \in \Gamma^+(r)} \{C_{ri}(h) + c(i, s)\} \quad \forall j \in N \tag{37}$$

$$\mathscr{P}_{rs}(h + 1) = \mathscr{P}_{rj}(h) \cup \{(j, s)\} \quad \text{where} \quad j = \operatorname*{argmin}_{j \in \Gamma^+(r)} \{C_{rj}(h + 1)\} \tag{38}$$

This algorithm stops when h is equal to the desired number of hops and takes $\mathscr{O}(h|N|^2)$ time. In our case, for each OD pair, this recursive formulation needs to be evaluated at most $\overline{H}_{rs}$ times before the existence of a (shortest) FIP from r to s can be established. To find the next shortest FIP – if it exists – we use a procedure inspired by k-SP algorithms. Most k-SP algorithms start by finding the SP and then look among the possible deviations from this path to find next SP. This procedure is then iteratively applied until k paths have been found. Yen (1971) algorithm uses two containers to store the candidate SPs found: we denote A_{rs}^{SP} the container containing the k SPs and B_{rs}^{SP} the one containing all the deviations from r to s that have been identified so far. We denote A_{rs}^{SFIP} the container containing the K Shortest FIPs from r to s.

To maximize the chances that every infected information node can be connected by a FIP to a root node, we consider every possible OD pair. Specifically, let $R_s \equiv \{r \in I_i : T_s \geqslant T_r + L\}$, R_s represents the set of possible root nodes for each leaf node $s \in I_i$. The set of possible OD pairs is then composed of all the pairs (s, r) where $S \in I_i$ and $r \in R_s$. The pseudo-code of the graph reduction heuristic

is summarized in Algorithm 1. The K-SFIP algorithm first searches for the shortest FIP using the recursive procedure defined by Eqs. 37 and 38. If the a FIP has been found, then this is the shortest FIP (denoted 1-SFIP) and the algorithm restricts the search for deviations to the paths contained in A_{rs}^{SFIP} and uses a k-SP like procedure to enumerate candidate paths for the current OD pair. This algorithm stops when the K shortest FIP are found or if the length of the current SP is strictly greater than the maximum feasible number of hops for this OD pair.

Algorithm 1 K-SFIP Algorithm

Data: A graph $G = (N, A)$, a constant K
Result: A subgraph $G_K = (N_K, A_K)$
$R \leftarrow \arg\min_{i \in I_i} \{t_i\}$;
for $s \in I_i$ **do**
  for $r \in R_s$ **do**
    Determine the existence of 1-SFIP using the recursive procedure (37) and (38);
    if *1-SFIP exists* **then**
      Store 1-SFIP in A_{rs}^{SFIP};
      while $|A_{rs}^{SFIP}| < K \wedge T_s - T_r \geqslant |\mathscr{P}_{rs}|L$ **do**
        for *all deviations from the last path stored in* A_{rs}^{SFIP} **do**
          Find the SP and store it in B_{rs}^{SP};
        end
        Move the shortest deviation $\mathscr{P}_{rs}$ from B_{rs}^{SP} to A_{rs}^{SP};
        if $\underline{H}_{rs} \leqslant |\mathscr{P}_{rs}| \leqslant \overline{H}_{rs}$ **then** store $\mathscr{P}_{rs}$ in A_{rs}^{SFIP};
      end
    end
  end
end
$G_K \leftarrow \bigcup_{s \in I_i} \bigcup_{r \in R_s} (A_{rs}^{SFIP})$;

Theorem 1 (Correctness and time complexity of the K-SFIP algorithm) *The K-SFIP algorithm finds the K shortest FIP to each leaf from every possible root node for this leaf node if they exist in $\mathscr{O}(|I_i|^2(\overline{H}|N|^2 + K|N|^3))$ time, where $\overline{H} = \max_{r,s \in I_i}\{\overline{H}_{rs}\}$.*

Proof For each leaf node $s \in I_i$ and for each possible root node $r \in R_s$, the K-SFIP algorithm starts by determining the existence of a FIP from r to s using the recursive formulation defined by Eqs. 37 and 38. For each OD pair, if a FIP exists, then this is the shortest FIP. If there does not exist any FIP from r to s, the algorithm moves to the next OD pair. Otherwise, to find the next shortest FIP, the algorithm searches among deviations from the last shortest FIP stored in A_{rs}^{SFIP}. The shortest deviation is moved from B_{rs}^{SP} to A_{rs}^{SP} and, if this path is a FIP, it is the next shortest FIP and this path is moved to A_{rs}^{SFIP}. The search for K shortest FIP terminates when K (shortest) FIP from r to s have been found or if the length of the current k-SP times the exposed period is greater than the timestamp difference for the current OD pair, that is, if $T_s - T_r < |\mathscr{P}_{rs}^k|L$. In this case, all the remaining paths for the current OD pair are infeasible. Hence K-SFIP algorithm finds the K shortest FIP to each leaf node from each possible root node for this leaf node if they exist. Since the time complexity of the recursive algorithm of Saigal (1968) is $\mathscr{O}(h|N|^2)$, the existence of the shortest FIP

for each OD pair can be determined in $\mathcal{O}(\overline{H}_{rs}|N|^2)$ time. Observe that all the paths found after a shortest FIP has been found are either feasible if $T_s - T_r \geqslant |\mathscr{P}^k_{rs}|L$ or infeasible. Hence the while loop is executed at most $K-1$ times. Since the time complexity of Yen (1971) k-SP algorithm is $\mathcal{O}(k|N|^3)$ and at most K shortest FIP per possible OD pair are sought, the time complexity of the K-SFIP algorithm is $\mathcal{O}(|I_i|^2(\overline{H}|N|^2 + K|N|^3))$. □

We can take advantage of the subpath feasibility property (Proposition 3) to speed-up the search procedure by strategically iterating over the set of possible OD pairs and pruning the search tree. Namely, if we sort the infected information nodes by their timestamps, we can iterate over the leaf nodes (first **for** loop) by increasing order of timestamps and iterate over the possible root nodes (second **for** loop) by decreasing order of timestamps. Then, for each leaf node $s \in I_i$ and for each root node $r \in R_s$, if at least one FIP has been found from r to s, we can then prune the set of possible root nodes R_s by removing all the nodes to which a FIP from the current root node r has been found. Namely, for each OD pair, we can re-define the set of possible roots as follows:

$$R_s \leftarrow R_s \setminus \{r' \in R_s : \exists \underline{H}_{r'r} \leqslant |\mathscr{P}_{r'r}| \leqslant \overline{H}_{r'r}\} \quad \forall s \in I_i, \forall r \in R_s : \exists \underline{H}_{rs} \leqslant |\mathscr{P}_{rs}| \leqslant \overline{H}_{rs} \tag{39}$$

This pruning procedure guarantees that for every root node r' removed from R_s, there exist at least one FIP, and at most K, from r' to s which pass through the current root node r. This helps in improving the computational performance of the K-SFIP algorithm. Algorithm 1 provide a method to significantly reduce the search space. This heuristic however, does not ensure that the obtained subgraph contains a feasible infection tree. This is illustrated in Example 2.

Example 2 Consider the network depicted by Fig. 5. In this outbreak scenario, only three of the individuals are information nodes (red nodes). If Algorithm 1 is executed with $K = 1$, the search for the shortest FIP from root A to destination node D returns path $\{A, E, D\}$ and the search for the shortest FIP to destination node F returns path $\{A, E, F\}$. Solving Model 1 on the subgraph composed by both paths produces an infeasible solution because the constraints on node E are conflicting. Indeed, path $\{A, E, D\}$ imposes that $t_E \in [1, 2]$ and path $\{A, E, F\}$ imposes that $t_E \in [4, 8]$.

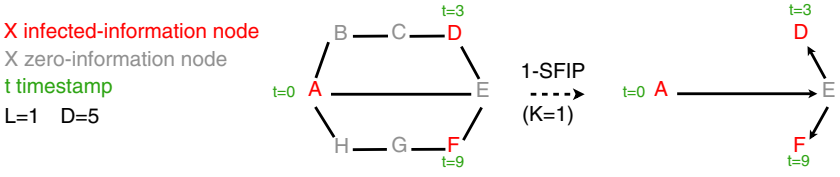


Fig. 5 An outbreak scenario where the execution of the K-SFIP algorithm with $K = 1$ does not produce a subgraph containing a feasible infection tree

To increase the chances that the subgraph produced by the K-SFIP algorithm contains a feasible infection tree, it is necessary to increase the value of K. Searching for a large number of K shortest FIP leads to the generation of larger subgraphs and increases the probability of obtaining a feasible infection tree but may also deteriorate the computational performance of the second step, *i.e.* finding the maximum likelihood tree in the resulting subgraph. To find the MLIT, we propose to solve Model 1 for every subgraph G_K generated. We next present an exact reformulation of Model 1 as a MILP that can be used to find the MLIT on every subgraph G_K.

4.2 Exact MILP Reformulation

In order to provide an efficient formulation to solve the MLIT, we introduce auxiliary decision variables and constraints to linearize objective function (8) and the feasibility constraints (9) and (10) in Model 1 with respect to decisions variables x_{ij} and t_i. The feasibility constraints can be linearized using T (resp. $-T$) as the upper (resp. lower) bound on Δt_{ij}:

$$\forall (i,j) \in A, \qquad \begin{cases} \Delta t_{ij} \leqslant x_{ij}(L+D-1) + (1-x_{ij})T & (40) \\ \Delta t_{ij} \geqslant x_{ij}L - (1-x_{ij})T & (41) \end{cases}$$

It is a common practice among optimization techniques to consider the logarithm of likelihood functions instead of its proper expression. Applying this technique to Eq. 8 we obtain the following objective function

$$\max \sum_{(i,j)\in A} x_{ij} \log(p_{ij}) + \log(1-p_{ij}) \left(x_{ij}(\Delta t_{ij} - L)^+ + (1-x_{ij}) \min\{D, (\Delta t_{ij} - L)^+\}\right) \quad (42)$$

which can be written as

$$\max \sum_{(i,j)\in A} x_{ij} \log(p_{ij}) + \log(1-p_{ij})(B_1 + B_2) \quad (43)$$

where

$$B_1 = x_{ij}(\Delta t_{ij} - L)^+ \quad (44)$$

$$B_2 = (1-x_{ij}) \min\left\{D, (\Delta t_{ij} - L + 1)^+\right\} \quad (45)$$

To linearize the objective function (42) we have to linearize B_1 and B_2. B_1 can be linearized by introducing a continuous decision variable $\rho_{ij}^{x,\Delta t}$ defined as

$$\forall (i,j) \in A, \qquad \rho_{ij}^{x,\Delta t} \equiv x_{ij}\, \Delta t_{ij} \quad (46)$$

and the following set of constraints

$$\forall (i,j) \in A, \qquad C(x_{ij}, \Delta t_{ij}) \equiv \begin{cases} \rho_{ij}^{x,\Delta t} \leqslant \Delta t_{ij} + (1-x_{ij})T & (47) \\ \rho_{ij}^{x,\Delta t} \geqslant \Delta t_{ij} - (1-x_{ij})T & (48) \\ \rho_{ij}^{x,\Delta t} \leqslant x_{ij}T & (49) \\ \rho_{ij}^{x,\Delta t} \geqslant -x_{ij}T & (50) \end{cases}$$

$C(x_{ij}, \Delta t_{ij})$ ensure that variable $\rho_{ij}^{x,\Delta t}$ behaves as the product $x_{ij} \cdot \Delta t_{ij}$ and is known as the Fortet linearization. If $x_{ij} = 1$, then the first two constraints ensure that

$\rho_{ij}^{x,\Delta t} = \Delta t_{ij}$. If $x_{ij} = 0$, then the first two constraints ensure that $\rho_{ij}^{x,\Delta t}$ is lower than a positive value and greater than a negative value and the last two constraints ensure that $\rho_{ij}^{x,\Delta t} = 0$. This exact linearization is possible mainly because Δt_{ij} is a bounded variable; note that this reformulation is also possible if Δt_{ij} is a continuous variable (Liberti et al. 2009). In the remainder of this paper, the notation $C(bin, int)$ is used to design the set of constraints used to reformulate the decision variable product $bin \cdot int$, where bin is a binary variable and int is a bounded integer variable.

B_1 can hence be expressed linearly as

$$B_1 = \rho_{ij}^{x,\Delta t} - x_{ij} L \tag{51}$$

To linearize B_2 first observe that since $D \leqslant 0$, we have:

$$\min\{D, (\Delta t_{ij} - L + 1)^+\} = (\min\{D, (\Delta t_{ij} - L + 1)\})^+ \tag{52}$$

In Eq. 43, B_1 and B_2 are weighted by logarithms of probabilities, therefore the objective function of this problem can be represented so as to maximize a sum of negative-weighted terms. Hence we can linearize the max and the min by introducing two variables $l_{ij} \geqslant 0$ and $m_{ij} \geqslant 0$ defined as:

$$\forall (i, j) \in A, \qquad l_{ij} \equiv \min\{D, (\Delta t_{ij} - L + 1)\} \tag{53}$$

$$\forall (i, j) \in A, \qquad m_{ij} \equiv (l_{ij})^+ \tag{54}$$

and using the following constraints for each link $(i, j) \in A$:

$$m_{ij} \geqslant 0 \tag{55}$$

$$m_{ij} \geqslant l_{ij} \tag{56}$$

$$l_{ij} \leqslant \Delta t_{ij} - L + 1 \tag{57}$$

$$l_{ij} \leqslant D \tag{58}$$

$$l_{ij} \geqslant D - (D + L + T - 1)(1 - z_{ij}) \tag{59}$$

$$l_{ij} \geqslant \Delta t_{ij} - L + 1 - 2T z_{ij} \tag{60}$$

where z_{ij} is a binary decision variable that takes value 1 if $D \leqslant \Delta t_{ij} - L + 1$ and 0 otherwise. Hence B_2 can be expressed linearly as

$$B_2 = m_{ij} - \rho_{ij}^{x,m} \tag{61}$$

where $\rho_{ij}^{x,m}$ is a continuous decision variable defined as

$$\forall (i, j) \in A, \qquad \rho_{ij}^{x,m} \equiv x_{ij} m_{ij} \tag{62}$$

through constraint set $C(x_{ij}, m_{ij})$. Note that due to the direction of the objective function, if $x_{ij} = 0$ then $B_2 = m_{ij} = l_{ij}$. Since l_{ij} is an exact linearization of $\min\{D, (\Delta t_{ij} - L + 1)\}$ and both D and $\Delta t_{ij} - L + 1$ are integers, then l_{ij} must also be integer. If $x_{ij} = 1$, then $B_2 = 0$ and variables l_{ij} and m_{ij} have no impact on the value of the objective function. The resulting model is presented below.

Model 2 (MILP for the MLIT)

$$\max \sum_{(i,j)\in A} x_{ij}\log(p_{ij}) + \log(1-p_{ij})\left(\rho_{ij}^{x,\Delta t} - x_{ij}L + m_{ij} - \rho_{ij}^{x,m}\right) \tag{63}$$

subject to

$$\left.\begin{array}{lll}
\rho_{ij}^{x,\Delta t} & \equiv & C(x_{ij}, \Delta t_{ij}) \\
\rho_{ij}^{x,m} & \equiv & C(x_{ij}, m_{ij}) \\
\Delta t_{ij} & = & t_j - t_i \\
\Delta t_{ij} & \leqslant & x_{ij}(L + D - 1) + (1 - x_{ij})T \\
\Delta t_{ij} & \geqslant & x_{ij}L - (1 - x_{ij})T \\
m_{ij} & \geqslant & 0 \\
m_{ij} & \geqslant & l_{ij} \\
l_{ij} & \leqslant & \Delta t_{ij} - L + 1 \\
l_{ij} & \leqslant & D \\
l_{ij} & \geqslant & D - (D + L + T - 1)(1 - z_{ij}) \\
l_{ij} & \geqslant & \Delta t_{ij} - L + 1 - 2T z_{ij}
\end{array}\right\} \quad \forall (i,j) \in A \tag{64}$$

Tree constraints (12) − (15)

Timestamps constraints (16) − (19)

Domain constraints (20) − (22) *and auxiliary variables*

Model 2 is an exact reformulation of Model 1 that can be solved using commercial MILP optimization software. We use Model 2 in our solution method to find the MLIT on the subgraphs induced by Algorithm 1. In the next section, we present the validation framework used to measure the performance of our approach.

5 Validation Framework

In this section, we present the methodology used to evaluate the performance of the proposed solution method. To validate our approach we (1) simulate outbreak scenarios on a randomly generated network using the SEIR compartmental model (presented in Section 3.2), (2) extract information on a subset of the nodes corresponding to a specified level of information availability which is to be used as input for the model, (3) implement the solution method, and (4) evaluate the performance of the solution method based on its ability to accurately identify the actual infection pattern. Only outbreaks evolving from a single source, *i.e.* a single infected individual, are considered in the analysis presented in this paper. Evaluation of the proposed methodology for outbreaks which evolve from multiple sources will be explored in future work.

The model performance is measured by comparing the set of nodes and links involved in the spread of the disease in the SEIR simulation-based scenario (hereby referred to as actual outbreak) with those identified by the MLIT obtained by the solution method. Of specific interest in this work is the performance of the solution

method for different levels of information availability *i.e.* size of the information subset. Recall that the status and the timestamps of zero-information nodes are unknown from the perspective of the solution method. For the evaluation of each sample we define a set of metrics to quantify the performance of the solution method. The metrics are determined by comparing the actual infection tree obtained at the chosen observation date, with the MLIT obtained by the solution method. We consider two types of metrics:

– Link metrics; which compare the observed network links that have spread the disease (extracted from the SEIR simulation) with the links identified by the MLIT.
– Node metrics; which compare the observed compartment of the nodes (extracted from the SEIR simulation) with the compartmental status of the nodes as specified by the MLIT.

For a given network G, disease parameters D and L, set of link transmission probabilities $[p_{ij}]$, and a specified level of information; the following steps are used to compute the performance metrics:

1. Randomly introduce an infected individual into the network (source node)
2. Simulate an outbreak for a specified number of time steps (up to the observation date) using the SEIR model
3. Extract the observed infection tree $\mathcal{T}^{obs}$ from the simulation to use for evaluating the performance of the solution method, namely:
   (a.) the full set of links in $\mathcal{T}^{obs}$
   (b.) the full set of infected nodes in $\mathcal{T}^{obs}$ and their timestamps
4. Randomly select a subset I of nodes according to the level of information
5. Extract the following (required) information from the simulation to use as input for the solution method:
   (a) the set of information nodes I
   (b) the timestamps of all information nodes $T_i, \forall \in I$
6. For each value of K tested:
   (a) Implement Algorithm K-SFIP on G and store the obtained paths in G_K
   (b) Solve Model 2 on the resulting subgraph G_K
7. Solve Model 2 on G
8. Repeat Steps 1 to 7 n times and compute statistics on the performance of the solution method

The procedure outlined above returns the expected performance of our solution method, which is how accurately the MLIT represents the actual spreading scenario, for a specified network structure, level of information and set of disease parameters. To create the network structures we use a random graph generator which relies on a preferential attachment rule, resulting in networks with 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

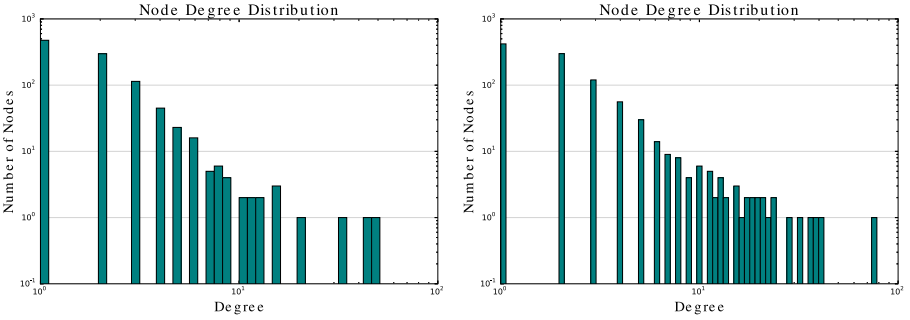
1999; Gonzales et al. 2008). Power law networks have a hub and spoke type structure with few highly connected nodes, (known as super spreaders in the context of contagion problems), while most nodes have a very low degree and are representative of many real world network structures (Clauset et al. 2009). The level of heterogeneity depends on the power law exponent; we present results from two power law networks:

– exponent of 3: 1,000 nodes and 2,184 directed links
– exponent of 2.5: 1,000 nodes and 2,752 directed links

The networks are generated using Networkx, a Python module for complex networks representation (Hagberg et al. 2008). A sequence of node degrees are sampled from a power-law distribution with a specified exponent (*i.e.* 3 or 2.5) and a graph is randomly generated by assigning edges to match the degree sequence (self edges and parallel edges are removed). This procedure is executed until a connected component with the desired number of nodes (*i.e.* 1,000) is found. The node degree distributions of the two networks used are given in Fig. 6. In this case study, we use the following parameters values:

- Exposed period $L = 1$
- Infectious period $D = 3$
- Observation date: 7 time steps
- 5 levels of information: 20 %, 40 %, 60 %, 80 % and 100 %
- 2 range of disease transmission probabilities:
  - low range: $\forall(i, j) \in A$, $p_{ij} \in [0.1, 0.5]$
  - high range: $\forall(i, j) \in A$, $p_{ij} \in [0.5, 0.9]$

Both exposed and infectious periods are expressed in units of time and the outbreak data is extracted after 7 simulation steps (recall that the contagion process herein is assumed to evolve with a discrete time step). Our choice of the observation period was made in accordance to the outbreak size (proportion of the population infected): we have chosen our observation date to ensure that the outbreaks observed



(a) Power law network with exponent of 3

(b) Power law network with exponent of 2.5

Fig. 6 Node Degree Distribution of the power law networks

spanned different outbreak sizes, for the range of disease transmission probabilities and network structures tested.

6 Results

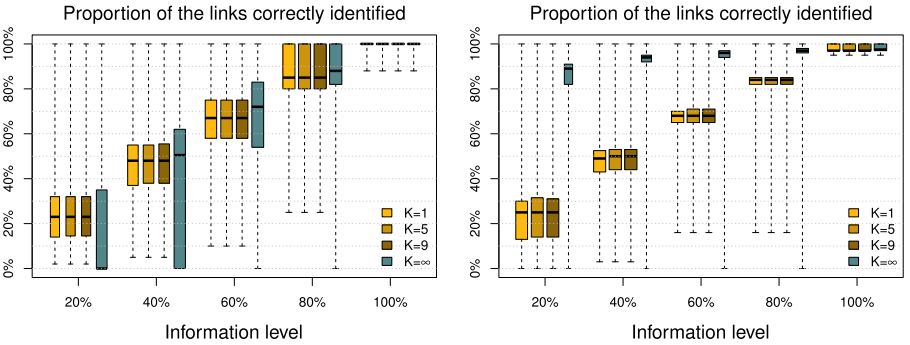
We first carry out a detailed evaluation of the proposed solution method before presenting a sensitivity analysis with regards to the network topology and the spread of the outbreak, and discussing its computational performance.

6.1 Solution Method Evaluation

For this evaluation, we focus on the power law network with an exponent of 3 and examine the performance of the solution method with regards to the range of transmission probabilities and the level of information. For each link of the network, the disease transmission probabilities are randomly and uniformly generated in the selected range (low or high). To reproduce the limited availability of information, we randomly select a subset of the nodes to be information nodes. Due to the heterogeneous structure of power law networks, contagion processes can result in a wide range of scenarios, even for the same set of disease parameters and source of infection. Given the stochastic nature of the contagion process and the fact that the model performance can vary drastically based on the specific contagion process which evolves, the performance of the model is evaluated by averaging over multiple samples (*i.e* simulated contagion episodes). Namely, for each combination of level of information and range of disease transmission probabilities, we evaluate $n = 500$ samples and for each sample the model performance is based on how accurately it predicts the actual paths of infection (which are extracted from simulation outputs).

The solution method is implemented for three values of K: $K = 1, 5$ and 9. Recall that the graph G_1 obtained for $K = 1$ is a subgraph of the subgraph obtained for higher values of K, that is $G_1 \subseteq G_5 \subseteq \ldots \subseteq G$. To measure the performance of the K-SFIP algorithm, we also implement Model 2 on graph G (data series $K = \infty$). The simulation of the SEIR compartmental model and the K-SFIP algorithm are implemented in C++ on a 64-bit machine with 188Gb of RAM and the optimization problems (MILP instances) are solved using the CPLEX v12.5 commercial package (C++ API) with a time limit of five minutes and an integrality gap of $1e-5$. The results are presented using boxplots in which each box represents the span of the values in the InterQuartile Range (IQR) and the whiskers extend to the minimal and the maximal values observed. Furthermore, the width of the box is proportional to the number of feasible solutions found by Model 2 for this combination of parameters.

Figure 7 shows the proportion of the links correctly identified by the solution method for each combination of level of information and value K tested as well as the performance of the MILP when no graph reduction algorithm is used; for a low disease transmission probabilities range (7a) and for a high range (7b). We report that the performance of the solution method increases quasi-linearly with the information level. In the case of a low range of probabilities, the performance of the solution method is competitive with the one observed when solving on the entire



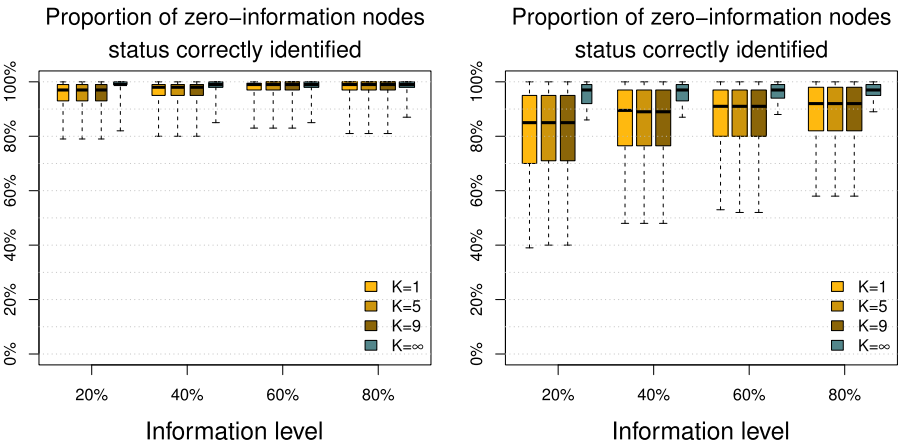
(a) Link identification with low range disease transmission probabilities

(b) Link identification with high range disease transmission probabilities

Fig. 7 Zero-information node status identification

graph G, i.e. $K = \infty$ and the variance of the number of links correctly identified decreases when more information is available. While solving Model 2 on G directly ($K = \infty$) yields more feasible solutions than using the K-SFIP algorithm, it may also results in extremely poor performance, in particular for low levels of information (20 % and 40 %). With a high range of probabilities, solving on G produces a better link identification than with G_K for all the values of K tested but the difference in performance decreases with the information level. Using a high range of probabilities, the outcome of the model on these instances is less volatile than with a low range. We also observe that, for an information level of 20 %, contrary to the tests with a low range of probabilities, solving on G directly yields less feasible solutions than using the graph reduction algorithm. Since for each sample in which a feasible solution has been found using the heuristic, this solution could have been found by solving on the whole graph, this outcome indicates that CPLEX has not been able to find this feasible solution within the allocated five minutes of runtime. For both ranges of probabilities, the performance of the solution method improves marginally with K.

Figure 8 demonstrates the performance of the proposed solution method by showing the average proportion of zero-information nodes which status, *i.e.* infected or not, has been correctly identified for a low range of probabilities (Fig. 8a) and for a high range of probabilities (Fig. 8b). For both transmission probability ranges, the solution method is relatively robust with regards to the level of information available, although its performance is slightly inferior for low levels of information (20 % and 40 %). For the low range of probabilities, on average, more than 90 % of the zero-information nodes status are correctly identified and the status of at least 80 % of these nodes is correctly identified, for each value of K tested. For the high range of probabilities, solving on the entire graph G outperforms the heuristic algorithm which is more volatile and only marginally improves with the value of K. Figure 9 demonstrates the ability of the solution method to correctly determine the timestamp of zero-information nodes for a low range of probabilities (Fig. 9a) and for a high range of probabilities (Fig. 9b); the trend observed is similar to the one depicted by



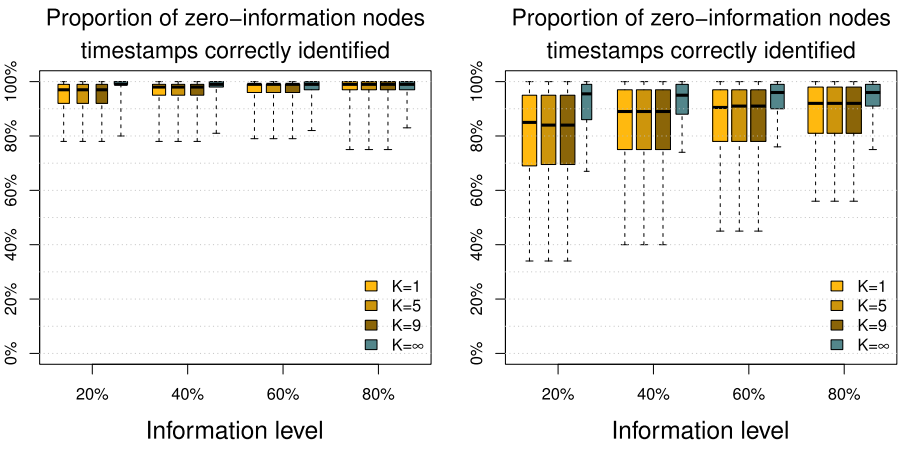
(a) Zero-node status with low range disease transmission probabilities

(b) Zero-node status with high range disease transmission probabilities

Fig. 8 Zero-information node status identification

the node status identification metric with a marginal loss of performance. On average, the solution method is able to correctly determine more than 90 % of the timestamps for the low range whereas this figure is decreased to 70 % for the high range and is more volatile.

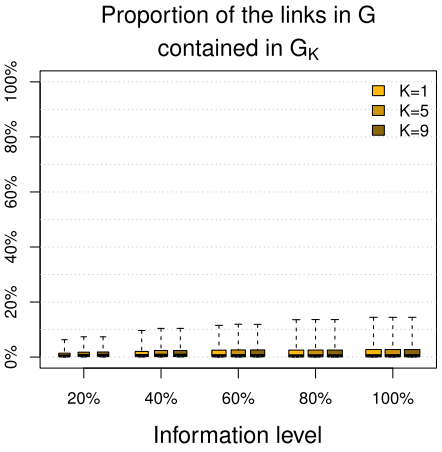
Figure 10 demonstrates the performance of the K-SFIP algorithm by showing the average proportion of links contained in G_K for each combination of level of information and value K for the low range (Fig. 10a) and for the high range (Fig. 10b); and the average proportion of links in the actual outbreak tree which are contained in G_K for the low range (Fig. 10c) and for the high range (Fig. 10d). For the low



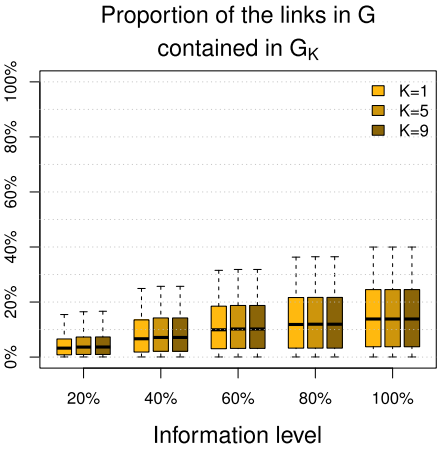
(a) Zero-node timestamp with low range disease transmission probabilities

(b) Zero-node timestamp with high range disease transmission probabilities

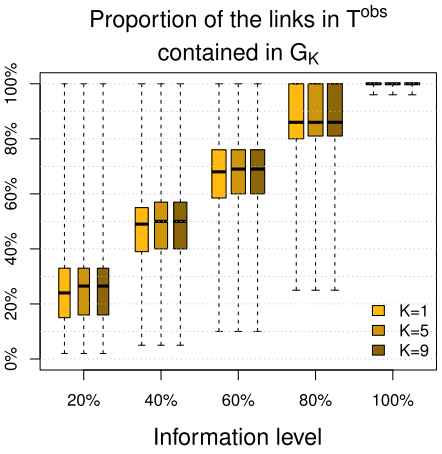
Fig. 9 Zero-information node timestamp identification



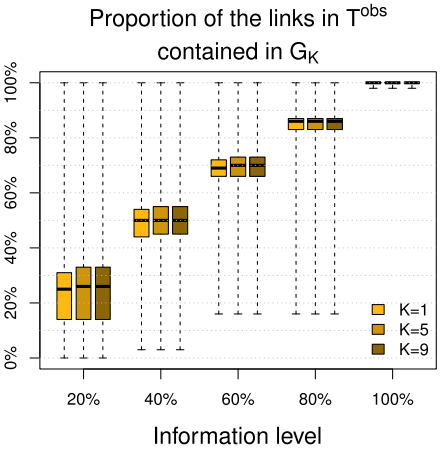
(a) % of Graph reduction with low range disease transmission probabilities



(b) % of Graph reduction with high range disease transmission probabilities



(c) % of Subgraph accuracy with low range disease transmission probabilities



(d) % of Subgraph accuracy with high range disease transmission probabilities

Fig. 10 Performance of the K-SFIP algorithm for different values of K and level of information

range case study, Graph G is always reduced by more than 85 %, and focusing on the IQR values, only 5 % of the links of G are contained in G_K, for each value of K tested and information level. The accuracy of the resulting subgraph G_K, *i.e.* the number of links in T^{obs} contained in G_K increases linearly with the level of information. We observe that, on average, the accuracy G_5 outperforms the one of G_1 but G_9 only marginally improves this performance measure with regards to G_5. The performance of the K-SFIP algorithm for a high range of transmission probabilities follows a similar trend but is slightly more sensitive to the level of information. The median (resp. maximal) graph reduction for $K = 5$ is 5 % (resp. 17 %) with a level of information of 20 % and 15 % (resp. 40 %) for the full information

scenario. The accuracy of the subgraph G_K is less volatile for a high range than for a low range of transmission probabilities but the average values for each combination of level of information and value of K is of the same order of magnitude for both ranges.

This analysis shows that the proposed solution method is efficient in the sense that it is able to significantly reduce the initial graph G and at the same time maintain a high level of accuracy. While solving Model 2 on the whole graph G may produce a better outcome than solving this model on a subgraph G_K, the heuristic algorithm is shown to provide a competitive performance for most of the scenarios tested. We next conduct a sensitivity analysis with regards to the link density and the outbreak size, *i.e.* the number of individuals infected in the actual outbreak.

6.2 Sensitivity Analysis

For this case study, we examine the performance of the solution method on the two aforementioned power law networks, *i.e.* with exponents of 3 and 2.5. The results are shown for $K = 5$ since no significant improvement has been observed for $K = 9$ and are plotted according to the outbreak size of each sample evaluated using scatter plots (each dot corresponds to a sample). Figure 11 shows the performance of the solution method with regards to the link identification metric and Fig. 12 shows the node status identification metric. For both figures, each column represents the results obtained for a combination of link-density (network) and range of disease transmission probabilities and each row shows the performance for a specified level of information. The outbreak size, *i.e.* proportion of the population infected, can change considerably but, as expected, we observe smaller outbreaks for a low range and the outbreak size increases with the number of links in the network.

The link identification metric is relatively robust with regards to the outbreak size and the link-density in contrast to the zero-information nodes identification metric which performance decreases with the outbreak size for partial information scenarios. This is a consequence of our model which is designed to search the graph based on the available information. When the size of the outbreak increases, the likelihood that the actual timestamp of a zero-information node is greater than the timestamp of the latest known infected individual (T) increases as well. Meanwhile, the model is highly likely to not include such zero-information nodes in the optimal tree. Hence, for outbreaks which spread to a large portion of the population, we expect the model to underestimate the spread of the infection. However, in the context of the infectious diseases, an outbreak which spreads to most or all members of the population is highly unlikely. More importantly, the same assumption that produces this weakness in the model performance under high transmission probabilities proves beneficial under lower transmission probabilities, which are more realistic in practice. The increase in link-density also affects the performance of the solution method with regards to the node identification metric, but the model remains able to correctly identify the links responsible for the spread of the disease.

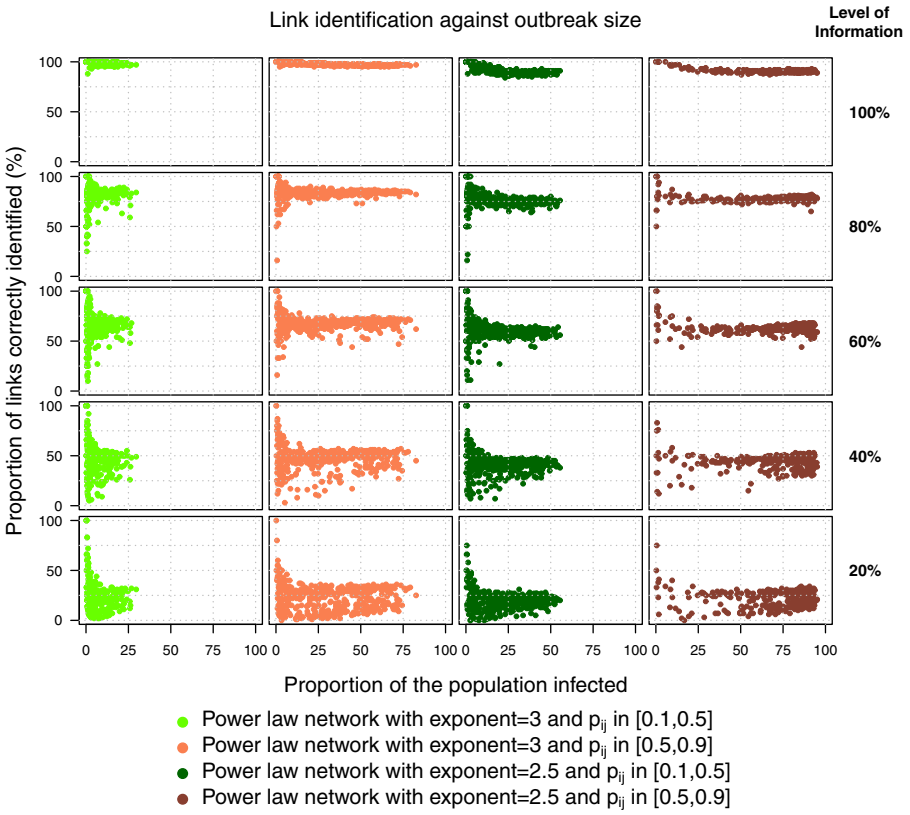


Fig. 11 Link identification sensitivity against network density, disease transmission probabilities range and outbreak size

6.3 Computational Performance

Table 2 describes the computational performance of the proposed solution method. The mean runtime of the K-SFIP algorithm is given for each network structure (power law exponent); range of disease transmission probabilities and information level. For each combination of input parameters, the number of feasible and optimal solutions obtained after solving Model 2 (MILP) using the CPLEX v12.5 solver with a five-minute time limit and the mean running time (averaged over the set of feasible instances) for each subgraph G_K as well as for graph G is detailed. The K-SFIP algorithm is faster for a low range of transmission probabilities than for a high range. This behavior is expected since using a high range results in more individuals infected and therefore more OD pairs (root, leaf) must be searched for FIPs. There is a decreasing marginal return on performance as K increases, with limited improvement occurring after $K = 5$. However, for the denser network structure, the performance increase with K was more noticeable. Using the K-SFIP algorithm and solving on the subgraph G_K is faster than solving directly on the whole graph G under low information levels but this trend is gradually reversed when the level of information increases.

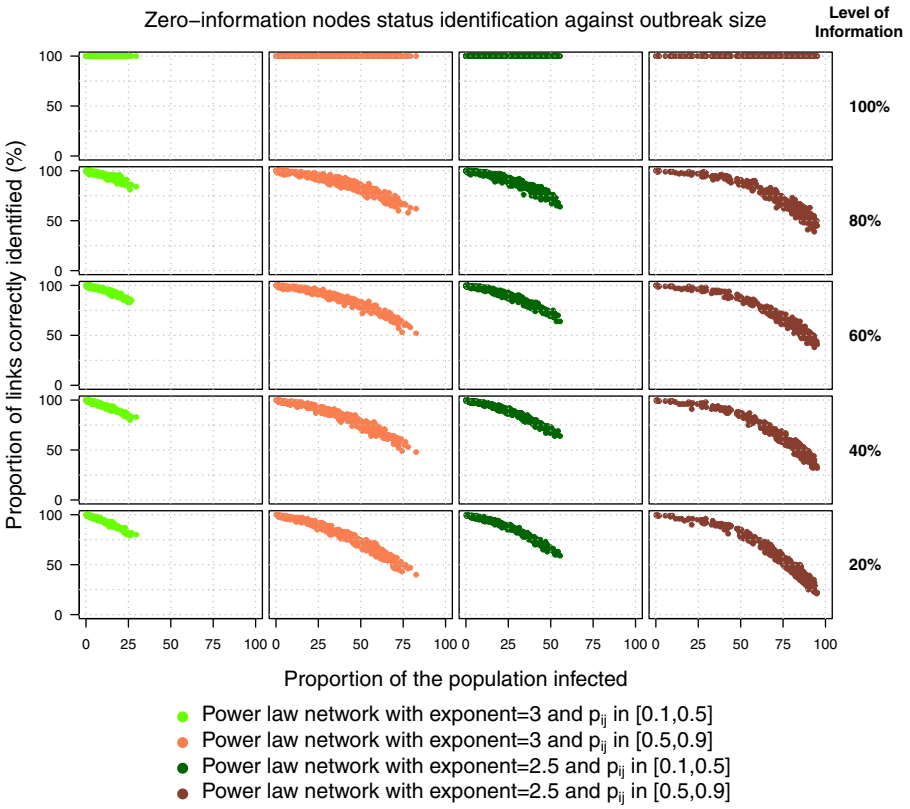


Fig. 12 Nodes status identification sensitivity against network density, disease transmission probabilities range and outbreak size

Furthermore, the K-SFIP algorithm is often able to identify optimal solutions when the MILP fails (within the allocated runtime), especially for the denser network structures. Comparing the runtime between the two different power law exponents, 3 and 2.5, which resulted in a 25 % increase in the number of links, it is clear that as the network density increases there is a significant increase in the required runtime for both the K-SFIP algorithm and the MILP, especially for higher probabilities. In order to reduce the runtime of the K-SFIP algorithm on large outbreaks, we have implemented a variant of our algorithm where the set of possible roots for each leaf node is reduced to the set of earliest infection information nodes, *i.e.* $R_s = I_e, \forall s \in I_i$. Using this heuristic to reduce the number of OD pairs to search considerably reduces, on average, the runtime of the graph reduction algorithm. However, this approach may occasionaly result in poor metric performance, in particular if the nodes in the set I_e are not responsible for the spread of the infection to most other infected information nodes. The results presented indicate that the proposed heuristic has the potential to significantly outperform the pure MILP approach, and that a fine tuning of K can result in balanced outcomes between the solution accuracy and the computational performance.

Table 2 Computational performance of the solution method for each combination of power law network exponent, information level, K and range of disease transmission probabilities, *i.e.* low ([0.1,0.5]) and high [0.5,0.9])

Instance			K-SFIP algorithm		MILP performance					
Network Exponent	Info. level	K	Low Runtime (s)	High	Low % Feasible	% Optimal	Runtime (s)	High % Feasible	% Optimal	Runtime (s)
	20 %	1	0.3	6.4	54	54	0.9	89	89	0.8
		5	0.3	6.4	55	55	0.9	91	91	0.9
		9	0.4	6.5	55	55	0.9	91	91	0.9
		∞	N/A	N/A	73	72	4.9	44	32	115
	40 %	1	0.7	19	71	71	0.8	94	94	0.8
		5	0.8	20	72	72	0.9	96	96	0.9
		9	0.8	20	72	72	0.9	96	96	0.9
		∞	N/A	N/A	82	82	1.4	64	63	4.8
3	60 %	1	1.1	34	84	84	0.8	97	97	0.9
		5	1.2	35	85	85	0.9	98	98	0.9
		9	1.2	35	85	85	0.9	98	98	0.9.
		∞	N/A	N/A	90	90	1.1	79	79	1.2
	80 %	1	1.2	38	94	94	0.9	99	99	0.8
		5	1.3	38	94	94	0.9	99	99	0.9
		9	1.3	39	94	94	0.9	99	99	0.9
		∞	N/A	N/A	96	96	1.0	92	92	1.1
	100 %	1	0.8	19	100	100	0.9	100	100	0.9
		5	0.9	20	100	100	0.9	100	100	0.9
		9	0.9	21	100	100	0.9	100	100	0.9
		∞	N/A	N/A	100	100	1.0	100	100	0.9

Table 2 (continued)

Instance			*K*-SFIP algorithm		MILP performance					
Network Exponent	Info. level	*K*	Low Runtime (s)	High	Low % Feasible	% Optimal	Runtime (s)	High % Feasible	% Optimal	Runtime (s)
	20 %	1	2.5	25	71	71	0.9	89	89	1.1
		5	3.0	26	78	77	5.3	95	94	4.1
		9	3.1	26	78	77	8.8	95	95	2.9
		∞	N/A	N/A	40	32	71	6.3	4.5	91.3
	40 %	1	7.4	83	77	77	0.9	89	89	1.1
		5	8.9	85	85	85	1.9	95	95	1.2
		9	9.0	86	88	88	2.4	95	95	1.3
		∞	N/A	N/A	68	57	70.4	57	15	237
2.5	60 %	1	13	157	84	84	0.9	92	92	0.9
		5	15	162	89	89	1.0	98	98	1.1
		9	16	163	89	89	1.0	98	98	1.1
		∞	N/A	N/A	83	83	4.9	73	72	8.9
	80 %	1	20	213	93	93	0.9	97	97	0.9
		5	21	215	95	95	0.9	100	100	1.0
		9	22	216	95	95	0.9	100	100	1.0
		∞	N/A	N/A	93	93	1.2	89	89	1.4
	100 %	1	9.1	74	100	100	0.9	100	100	0.9
		5	9.5	77	100	100	0.9	100	100	0.9
		9	9.7	77	100	100	0.9	100	100	0.9
		∞	N/A	N/A	100	100	1.0	100	100	1.0

The time limit for the MILP runs is 300s and any instance for which a feasible solution has not been found in the allocated solve time is considered not feasible. The mean runtimes reported is taken among the set of feasible instances

7 Conclusion

The problem addressed in this research is to infer a contagion tree in a network utilizing partially available node-level information. The underlying problem is similar to a constrained Steiner Tree problem in the sense that it requires an assignment of integer weights on the Steiner nodes. We introduced a novel IP model for identifying the MLIT, which improves on previous works by relaxing the assumption that the source(s) of the infection is known. We proposed a two-step solution method which relied on reducing the initial graph by finding FIP and solving an exact MILP reformulation of the initial IP on the obtained subgraph with a predetermined maximum number of FIP per OD pair. A polynomial time algorithm was developed for the graph reduction heuristic which was inspired by efficient procedures to solve the length constrained SP and the k-SP problems. The K-SFIP algorithm works from known infected nodes and attempts to find FIP in the graph by connecting root and leaf nodes.

The specific application of focus was disease outbreaks in social contact networks. The computed metrics reflect the ability of the solution method to appropriately identify infected nodes and infection spreading links, while also penalizing the model for over-infecting the network. As expected, the performance of the solution method was sensitive to the level of information in the network: the proportion of links correctly identified increased with information availability. With regards to the identification of the status and the timestamps of zero-information nodes, the solution method was shown to be robust to information availability. From a heuristic perspective, the K-SFIP algorithm was shown to be efficient as it was able to generate small subgraphs containing a significant share of the original graph which account for the majority of the actual infection tree. The implementation conducted in this paper shows that this epidemiological pattern inference problem can be solved efficiently on the power law networks considered in this study, whose node degree distribution is representative of realistic social contact networks. Further analysis is required to examine the performance of the proposed solution method on larger networks and on alternative network topologies. In particular, applying this model to real world contagion episodes may require the development of additional heuristics to cope with regional metropolitan areas.

While our approach relies on the assumption of available contact information and infection data which may seem limiting, it may not be an unrealistic assumption for the future. Information technology has evolved exponentially in the past decade alone, and is continually advancing in the ability to track individuals over time and space. While the specific means in which these contact networks are generated is beyond the scope of this paper, generating the required model input through the use of online social network data, cell phone data, and activity based travel models has been proposed and extensively studied by experts in these chosen fields. Such research efforts may potentially allow accurate mappings between known individuals, and there is currently a lack of modeling research which exploits the use of these data sets for modeling disease transmission in real time as a means to effectively manage outbreaks. Additionally, real-time infection data is becoming increasingly available from various online global health databases and real-time

reporting through the World Health Organization (WHO), Centers for Disease Control and Prevention (CDC), Centers for Infectious Disease Research and Policies (CIDRAP: www.cidrap.umn.edu) and the International Society for Infectious Diseases (ProMED: www.promedmail.org). These organizations and their respective data sets are becoming increasingly relied upon for tracking outbreaks, and also provide a means for public health and medical experts to gather the necessary information to estimate properties of emerging diseases which are required inputs for our model, such as transmission probability. This serves as the main motivation behind the proposed research. Furthermore, even today, there exist isolated communities and closed settings (schools, hospitals, etc) where the necessary inputs for our model (disease parameters, contacts and infection reports) can be made available, and the proposed model could be directly applied.

In conclusion, the proposed methodology provides a novel procedure for evaluating a group of individuals that has been exposed to infection. The performance of the solution method was shown to accurately identify a significant proportion of the nodes and links responsible for the spread of a disease in a network. Such information is valuable in aiding public health policy in the design of surveillance and outbreak intervention strategies. These types of models also serve to incentivize specific data collection efforts that would be the most valuable for modelling purposes, specifically validating models such as that proposed. Planned extensions of this research include further sensitivity analysis to network structure and epidemiological parameters, as well as exploring performance under outbreaks which evolve from multiple sources.

References

AJ D AR (2007) Beast: Bayesian evolutionary analysis by sampling trees. BMC Evol Biol 7:214

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

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

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

Broeck WV, Gioannini C, Gonċalves B, Quaggiotto M, Colizza V, Vespignani A (2011) The gleamviz computational tool, a publicly available software to explore realistic epidemic spreading scenarios at the global scale. BMC BMC Infect Dis 11(1):37

Clauset A, Shalizi C, Newman M (2009) Power-law distributions in empirical data. SIAM Rev 51(4):661–703. doi:10.1137/070710111

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

Cummings D, Burke D, Epstein JM, Singa R, Chakravarty S (2002) Toward a containment strategy for smallpox bioterror: an individual-based computational approach. Brookings Institute Press

D B VC, B G HH, JJ R AV (2009) The modelling of global epidemics: stochastic dynamics and predictability . Proc Natl Acad Sci USA 106:21484–21489

DT H, M CT, DJ S, L M, JK F, J W, MEJ W (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

Dunham J (2005) An agent-based spatially explicit epidemiological model in mason. J Artif Societies and Social Simulation 9(1):3

Erath A, Löchl M, Axhausen KW (2009) Graph-theoretical analysis of the swiss road and railway networks over time. Netw Spat Econ 9(3):379–400

Eubank S, Guclu H, Kumar V, Marathe M, Srinivasan A, Toroczkai Z, Wang N (2004) Modeling disease outbreaks in realistic urban social networks. Nature 429:180–184

Fajardo D, Gardner L (2013) Inferring contagion patterns in social contact networks with limited infection data. networks and spatial economics
Ferguson N, Cummings D, Fraser C, Cajka J, Cooley P, Burke D (2006) Strategies for mitigating an influenza pandemic. Nature 442:448–452
Gardner LM, Fajardo D, Waller ST (2012) Inferring infection-spreading links in an air traffic network. Transp Res Rec: J Transp Res Board 2300(1):13–21. doi:10.3141/2300-02
Gardner L M, Fajardo D, Travis W S (2014) Inferring contagion patterns in social contact networks using a maximum likelihood approach. ASCE, natural hazards review
Garey M, Johnson D (1977) The rectilinear Steiner tree problem is NP-complete. SIAM J Appl Math 32(4):826–834. doi:10.1137/0132071
Gastner MT, Newman ME (2006) The spatial structure of networks. Eur Phys J B-Condens Matter Complex Syst 49(2):247–252
Gonzales M, Hidalgo C, Barabási AL (2008) Understanding individual human mobility patterns. Nature 453:479–482
Gouveia L, Magnanti TL (2003) Network flow models for designing diameter-constrained minimum-spanning and steiner trees. Networks 41(3):159–173. doi:10.1002/net.10069
Gouveia L, Simonetti L, Uchoa E (2011) Modeling hop-constrained and diameter-constrained minimum spanning tree problems as steiner tree problems over layered graphs. Math Program 128(1–2):123–148. doi:10.1007/s10107-009-0297-2
Graham RL, Hell P (1985) On the history of the minimum spanning tree problem. Ann Hist Comput 7(1):43–57. doi:10.1109/MAHC.1985.10011
Hagberg AA, Schult DA, Swart PJ (2008) Exploring network structure, dynamics, and function using networkX. In: Proceedings of the 7th python in science conference (SciPy2008), Pasadena, pp 11–15
Hasan S, Ukkusuri S (2011) A contagion model for understanding the propagation of hurricane warning information. Transp Res B 45:1590–1605
Hoogendoorn SP, Bovy PH (2005) Pedestrian travel behavior modeling. Netw Spat Econ 5(2):193–216
Hwang FK, Richards DS (1992) Steiner tree problems. Networks 22(1):55–89. doi:10.1002/net.3230220105
Illenberger J, Nagel K, Flötteröd G (2013) The role of spatial interaction in social networks. Netw Spat Econ 13(3):255–282
Jombart T, Eggo RM, Dodd P, Balloux F (2009) Spatiotemporal dynamics in the early stages of the 2009 a/h1n1 influenza pandemic. PLoS currents influenza
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 WH, Huang HJ (2003) Combined activity/travel choice models: time-dependent and dynamic versions. Netw Spat Econ 3(3):323–347
Liberti L, Cafieri S, Tarissan F (2009) Reformulations in mathematical programming : a computational approach. In: Foundations of computational intelligence volume 3 - global optimization. Springer
Luo W, Tay WP, Leng M (2013) Identifying infection sources and regions in large networks. IEEE Trans Sigs Process 61(11):2850–2865
Murray J (2002) Mathematical biology, 3rd edn. Springer
Newman M, Forrest S, Balthrop J (2002) Email networks and the spread of computer viruses. Phys Rev E 66(3)
P L, M S, A R (2009) Reconstructing the initial global spread of a human influenza pandemic: a bayesian spatial-temporal model for the global spread of h1n1pdm. PLoS currents influenza
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
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
Roorda MJ, Carrasco JA, Miller EJ (2009) An integrated model of vehicle transactions, activity scheduling and mode choice. Transp Res B Methodol 43(2):217–229
Rosenwein MB, Wong RT (1995) A constrained steiner tree problem. European journal of operational research
Rosseel M (1968) Comments on a paper by romesh saigal: a constrained shortest route problem. Oper Res 16(6):1232–1234
Sachtjen M, Carreras B, Lynch V (2000) Disturbances in a power transmission system. Phys Rev E 61(5):4877–4882

Saigal R (1968) A constrained shortest route problem. Oper Res 16(1):205–209
Santos M, Drummond LM, Uchoa E (2010) A distributed dual ascent algorithm for the hop-constrained steiner tree problem. Oper Res Lett 38(1):57–62. doi:10.1016/j.orl.2009.09.008
Schintler LA, 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
Sornette D (2003) Why stock markets crash: critical events in complex financial systems. Princeton University Press
V C AB, M B AV (2006) The modelling of global epidemics: Stochastic dynamics and predictability. Bull Math Biol 68:1893–1921
Voss S (1999) The steiner tree problem with hop constraints. Annals of operations research
Wallace R, HoDac H, Lathrop R, Fitch W (2007) A statistical phylogeography of influenza a h5n1. Proc Natl Acad Sci USA 104(11):4473–4478
Wesolowski A, Buckee C, Bengtsson L, Wetter E, Lu X, Tatem A (2014) Commentary: containing the ebola outbreak–the potential and challenge of mobile network data. PLOS currents outbreaks
Yen JY (1971) Finding the k shortest loopless paths in a network. Management science