Ódor *et al. Applied Network Science* (2023) 8:53
https://doi.org/10.1007/s41109-023-00566-3

Applied Network Science

RESEARCH Open Access

# Source identification via contact tracing in the presence of asymptomatic patients

Gergely Ódor[1,2*], Jana Vuckovic[1], Miguel-Angel Sanchez Ndoye[1] and Patrick Thiran[1*]

*Correspondence:
odog@ceu.edu;
patrick.thiran@epfl.ch

[1] EPFL, Lausanne, Switzerland
[2] CEU, Vienna, Austria

## Abstract

Inferring the source of a diffusion in a large network of agents is a difficult but feasible task, if a few agents act as sensors revealing the time at which they got hit by the diffusion. One of the main limitations of current source identification algorithms is that they assume full knowledge of the contact network, which is rarely the case, especially for epidemics, where the source is called patient zero. Inspired by recent implementations of contact tracing algorithms, we propose a new framework, which we call Source Identification via Contact Tracing Framework (SICTF). In the SICTF, the source identification task starts at the time of the first hospitalization, and initially we have no knowledge about the contact network other than the identity of the first hospitalized agent. We may then explore the network by contact queries, and obtain symptom onset times by test queries in an adaptive way, i.e., both contact and test queries can depend on the outcome of previous queries. We also assume that some of the agents may be asymptomatic, and therefore cannot reveal their symptom onset time. Our goal is to find patient zero with as few contact and test queries as possible. We implement two local search algorithms for the SICTF: the LS algorithm, which has recently been proposed by Waniek et al. in a similar framework, is more data-efficient, but can fail to find the true source if many asymptomatic agents are present, whereas the LS+ algorithm is more robust to asymptomatic agents. By simulations we show that both LS and LS+ outperform previously proposed adaptive and non-adaptive source identification algorithms adapted to the SICTF, even though these baseline algorithms have full access to the contact network. Extending the theory of random exponential trees, we analytically approximate the source identification probability of the LS/ LS+ algorithms, and we show that our analytic results match the simulations. Finally, we benchmark our algorithms on the Data-driven COVID-19 Simulator (DCS) developed by Lorch et al., which is the first time source identification algorithms are tested on such a complex dataset.

**Keywords:** Adaptive source identification, Contact tracing, Sensor selection, Epidemics

## Introduction

During the COVID-19 pandemic, we have seen a revolution of the contact tracing technology, which helped track and contain the epidemic (Braithwaite et al. 2020; Kretzschmar et al. 2020). Some contact tracing programs were conducted by governmental/ health agencies (Park et al. 2020), while others relied on decentralized approaches

(Troncoso et al. 2020). Most contact tracing approaches work by notifying people who could have received the infection from known infectious patients, i.e., they trace "forward" in time. However, some advocate that a "bidirectional" tracing, where the past history of the infection is also tracked, can be more effective (Bradshaw et al. 2021; Endo et al. 2020; Kojaku et al. 2021; Raymenants et al. 2022). In this paper we focus on the "backward" direction of the problem; the task of identifying the first patient who carried the disease, also called patient zero, or the source of the epidemic. The identification of patient zero can either be limited to a smaller population cluster, in which case it can be a first step towards "bidirectional" tracing, or it can be more ambitious; finding the first patient who developed the mutation of a certain disease. The identification of the source of an epidemic can be useful while planning our response as a society, since any information on the disease is crucial in uncertain times (Ingraham and Ingbar 2021). For example, source identification can aid contact tracing efforts (Carinci 2020; Russo et al. 2020; Feng et al. 2021), and identify superspreading events (Chen et al. 2022), moreover, understanding how the mutation occurred can give information on how dangerous the outbreak is (Kandeel et al. 2021; Kupferschmidt 2021; Zhang et al. 2020).

Until the COVID-19 epidemic, source identification algorithms had rarely been been applied beyond proof of principle studies. Our goal in this paper is to examine the applicability of the source identification models in the literature (which we call frameworks from now on), and then propose a new framework, which improves them in several aspects. See Fig. 1 and Table 1 for a visual and a table representation of our literature review of the previous frameworks. Originally, source identification was introduced in the context of rumor spreading instead of epidemics by Zaman and Shah in their pioneering paper (Shah and Zaman 2010, 2011). Translating to the language of epidemics for clarity, in the framework of (Shah and Zaman 2011), an epidemic spreads over a network of agents that is completely known to us, and we observe a *snapshot* of the network, which means that every agent reveals if they are infected or not at some given time (not too early, because then the problem is trivial, nor too late, because then the problem is impossible). Shortly after (Shah and Zaman 2011), Pinto et al. proposed a different framework, in which agents (also called *sensors*) reveal, in addition to their state, the

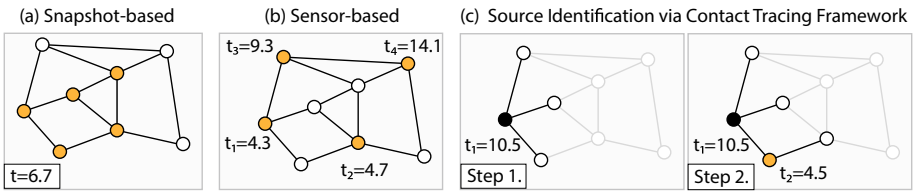


**Fig. 1** Illustration of three source identification frameworks. In all three cases, the epidemic process starts from a single source, which needs to be estimated based on different observations. **a** In the snapshot framework, every node reveals whether it had been infected (marked in orange) by some time $t$, when the snapshot is taken (in this case at time $t = 6.7$, where time is measured in days, relative to some arbitrary initial time, since the infection time of the source is unknown in all cases). **b** In the sensor-based framework, each node is assumed to have already been infected, and a few sensor nodes (marked in orange) report their infection time $t_i$. **c** In the Source Identification via Contact Tracking Framework (SICTF), the source identification task starts when the first node is hospitalized (marked in solid black), and initially most of the network is unknown to the algorithm (marked in grey). Then, in each step, the algorithm proceeds to explore the network and query the infection state of the nodes in an adaptive way, and queried nodes reveal their symptom onset time

**Table 1** Classification of cited papers based on the framework they operate in

| Data | Snapshot-based (binary information) | Sensor-based (time information) |
|---|---|---|
| Full network | Shah and Zaman (2010, 2011), and subsequent works | Pinto et al. (2012); Hu et al. (2018); Li et al. (2019); Paluch et al. (2018, 2020b); Shen et al. (2016); Tang et al. (2018); Xu et al. (2019); Zhu et al. (2016); Lecomte et al. (2020); Zejnilovic et al. (2013); Spinelli et al. (2017c, 2018); Paluch et al. (2020a); Shelke and Attar (2019); Odor (2022) |
| Contact tracing | Yu et al. (2022) | Waniek et al. (2022), and current paper (SICTF) |

The SICTF is a sensor-based framework with contact tracing data. The differences between SICTF and Waniek et al. (2022), are explained in "The source identification via contact tracing framework" section

time when they became infected, but where only a few of them do so and act as sensors (Pinto et al. 2012); indeed, the problem is trivial if all agents are sensors. This framework is better tailored to epidemics, as it is reasonable that obtaining any information from all the agents is much harder than asking one more question about the starting time of the symptoms of the disease to only some of them. Pinto et al. found that in their framework, if the sensors are already selected, the maximum likelihood estimator of the source has a closed form solution when the underlying network is a tree, and the time it takes for an agent to infect one of its susceptible contacts follows a Gaussian distribution. For general graphs, it is difficult to find an algorithm with any theoretical guarantees, although we note that many heuristics have been developed (Hu et al. 2018; Li et al. 2019; Paluch et al. 2018, 2020b; Shen et al. 2016; Tang et al. 2018; Xu et al. 2019; Zhu et al. 2016). The only exception is on very simple contact networks (Lecomte et al. 2020), or when the epidemic spreads deterministically between the agents (Zejnilovic et al. 2013), which is not a realistic assumption for epidemics, but at least the estimation algorithm is trivial, and more emphasis can be put on the question of how the sensors should be selected for good performance (Spinelli et al. 2017c, 2018), which again is studied by heuristics in the general case (Paluch et al. 2020a). For recent reviews of source identification algorithms, see (Shelke and Attar 2019; Odor 2022). Besides epidemics, related applications of source identification are rumor spreading between individual humans (Shah and Zaman 2011), a virus spreading in a computer network (Xie et al. 2005), train delay propagation (Manitz et al. 2014a), and food-borne disease outbreaks (Manitz et al. 2014b).

One of the main criticisms of the original framework of Pinto et al. is that, even though the contact network is fully known, it is very difficult to find the source exactly unless a large fraction (20–50%) of the population act as sensors, which is unrealistic in the case of an epidemics, when the source is searched in a large population. An alternative recently proposed is to compute confidence sets for the source instead of finding it (Dawkins et al. 2021). But if our goal is to locate the source exactly, a promising approach is to allow the sensors to be selected adaptively to previous observations (Zejnilović et al. 2015, 2017), which we call *adaptive sensor placement*. When the contact network is known, adaptive strategies have been studied by simulations (Spinelli et al. 2017a, b) and by theoretical analysis (Lecomte et al. 2020), and they show a large reduction in the number of required sensors in real networks. In this paper, we will also allow the sensors to be placed adaptively.

We believe that the most problematic assumption that is still present in source identification papers, is the full knowledge of the contact network of agents, which is unrealistic (let alone because of privacy concerns). Due to this lack of data-availability, algorithms in the source identification literature have not been tested on realistic epidemic real large-scale contact networks. Moreover, while governmental/health agencies might have access to private datasets, such as cellular location data, from which a contact network may be estimated, these networks may be very noisy, and are potentially unfit for the source identification task. We only know of a few papers that study the effect of imperfections in the network data on the source identification task (Mashkaria et al. 2020; Zejnilović et al. 2016), but these papers study epidemics that spread deterministically between the agents. Recently, a few papers suggested to use contact tracing data instead of assuming full knowledge about the network both in the snapshot-based (Yu et al. 2022) and in the sensor-based frameworks (Waniek et al. 2022). In this paper, we propose the *Source Identification via Contact Tracing Framework* (SICTF), which shares some similarities with the sensor framework of (Waniek et al. 2022), but while the latter framework is focused on the tradeoff between the contact tracing and source identification, our framework is focused only on the feasibility of source identification (see "The source identification via contact tracing framework" section for a more detailed comparison). In SICTF, algorithms can have two types of queries: contact queries, which can be used to explore the network, and sensor (test) queries, after which agents reveal their symptom onset time as before. The goal of the algorithm is to find the source as accurately as possible, while minimizing the number of contact and sensor queries. The SICTF is a way to formalize the source identification task; it determines the goal of the algorithm and how information can be gained about the epidemic, but it does not specify the underlying epidemic and mobility data models (simulated or real). In this paper, we analyse different algorithms in the SICTF with various epidemic and mobility models.

Besides specifying the possible queries that algorithms can make, the SICTF also determines the way the outbreak is detected, which marks the starting time of the source identification task. In sensor-based source identification, the source identification task often starts long after the outbreak, when essentially all agents in the network are infected (Pinto et al. 2012), which can be seen as a limitation of source identification frameworks. The SICTF is also closely related to contact tracing frameworks, where it is standard to assign a probability that each node spontaneously self-reports after developing symptoms, which triggers the activation of contact tracing algorithms (Kretzschmar et al. 2020; Bradshaw et al. 2021). In the SICTF, we adopt the idea of self-reporting with a slight modification. We believe that the most interesting time to perform the source identification task is when a new disease (or a new mutation of the disease) appears, and therefore we tie these self-reporting events to hospitalizations, where infections are properly diagnosed by healthcare professionals. This means that the SICTF can only be applied to epidemic data where hospitalizations are well-defined, which makes finding an appropriate dataset very challenging. Indeed, due to privacy concerns, there are no publicly available mobility and epidemiological datasets at the individual level (Ahn et al. 2020). We must note that given the appropriate resources, it is possible to collect individual level datasets for academic purposes. In Raymenants et al. (2022), backward

contact tracing was shown to be more efficient compared to forward contact tracing in a real COVID-19 dataset collected from university students residing in the city of Leuven. However, the dataset in Raymenants et al. (2022) was collected when the disease was present throughout the entire population with contact tracing as the primary goal, whereas we focus on early phase of the epidemic, and we aim to identify the source as accurately as possible. Collecing a dataset similar to Raymenants et al. (2022) with source identification in mind would be ideal, but it is outside the scope of the current paper. Instead, inspired by recent advances in the COVID-19 modelling literature, we restricted ourselves to agent-based epidemiological simulators that use aggregate datasets to generate individual synthetic mobility traces and simulate epidemic processes on them (Chang et al. 2021; Lorch et al. 2022; Müller et al. 2021). We emphasize that we can only use agent-based epidemic models in our study; in mean-field or meta-population models such as GLEaM (Balcan et al. 2010), it is not possible to identify patient zero due to the lack of fine-grained network structure. Among the mentioned agent-based models, we chose to work with the epidemiological simulator implemented by Lorch et al. (2022), which we refer to as the Data-driven COVID Simulator (DCS) from now on, because we found that the program code of the DCS strikes a good balance between complexity and extensibility. Because of its complexity, we only use the DCS to motivate the definition of simplified models and to validate our findings, and we perform our theoretical analysis on the simplified models, which we introduce in "The DDE model and Household network model" sections. In other words, our goal in this paper is to complete a full mathematical modeling cycle (Perrenet and Zwaneveld 2012), with the caveat that since we do not have access to real epidemiological datasets, we use one of the most complex and realistic epidemiological simulators (Lorch et al. 2022), which is based on real aggregate datasets.

In the SICTF, we propose a simple algorithm called LocalSearch (LS), which adaptively traces back the transmission path from the first hospitalized patient to the source. The LS algorithm is a natural candidate in the context of source identification with contact tracing data; the recently and independently proposed source identification algorithm of Waniek et al. (2022) is almost identical to the LS algorithm with a few minor differences arising from the differences between the two frameworks (see "The LocalSearch Algorithms LS and LS+" section). The LS algorithm is quite efficient at finding the source; the number of contact and sensor queries that it uses does not depend on the size of the network, but only on the local neighborhood of the source. Moreover, the LS algorithm provably finds the source with 100% accuracy, because of our assumption that every contact and sensor query is answered without noise. However, it is well known that data-availability is a major issue in contact tracing (BeidasRinad et al. 2020), either because the agents do not comply with contact tracing efforts, or possibly (and in particular in the current COVID-19 epidemic) because they do not develop symptoms, and are unaware that they have the disease. In this paper, we model the effect of asymptomatic agents. When queried and tested, these agents do not reveal their time of infection, only whether they have or had the disease at some point. We show that the accuracy of the LS algorithm drops in the presence of asymptomatic agents, because the algorithm can get stuck while tracing back the transmission path from the first hospitalized patient to the source. Therefore, we propose an improved version of LS called LS+, which accounts for

the presence of asymptomatic agents by placing more sensors. We are not aware of any previous work in the source identification literature that models the effect of asymptomatic patients, but the resulting model can be seen as a mix between the snapshot and the sensor-based models. We mention that non-complying agents or agents who provide noisy observations have been studied by Altarelli et al. (2014); Hernando et al. (2008); Louni et al. (2015). Non-complying agents could also be included in our framework by treating them as asymptomatic agents (even though in this case we have no information about whether the agent had the disease or not), without jeopardizing the correctness of our algorithms.

We benchmark the LS and LS+ algorithms in both our data-driven and our synthetic epidemic and mobility models, and we compare them to state-of-the-art adaptive (Spinelli et al. 2017b) and non-adaptive (Jiang et al. 2016; Lokhov et al. 2014) algorithms tailored to the SICTF, whenever possible. We find that both LS and LS+ outperform these baseline algorithms in accuracy (probability of finding the correct source).

While the LS/LS+ are designed to be simple algorithms, their theoretical analysis is quite challenging. Nevertheless, we are able to provide rigorous results about the source identification probability of both algorithms after a series of simplifications to the epidemic and mobility models, by extending some recent results on the theory of exponential random trees (Feng and Mahmoud 2018; Mahmoud 2021), which have previously not been connected to the source identification literature. We present these theoretical results in "Theoretical results" section, after formally introducing the SICTF, our models and the LS/LS+ algorithms in "Models, methods, algorithms" section. By simulations, we show that our analytic results approximate the accuracy of the algorithms well, even in the most realistic setting in "Simulation results" section. Our analytic results provide additional insight into how the parameters of the epidemic and mobility models affect the performance of the algorithms. We discuss these insights along with some non-rigorous computations that mirror our main proof ideas in Appendix A. Reading Appendix A before Sects. Models, methods, algorithms–Simulation results is useful to build intuition, but is not necessary to understand the paper.

## Models, methods, algorithms

### Epidemic models

#### *The DCS model*

We call DCS the model implemented by Lorch et al. (2022). We only use this model for validation, but we find it informative to introduce it first, because it inspired many of our modelling choices. Since the DCS model is fairly complex, we only give a brief overview.

Each agent in the agent set $V$ can be in one of 8 states: susceptible, exposed, asymptomatic infectious, pre-symptomatic infectious, symptomatic infectious, hospitalized, recovered or dead. Transitions between different states are characterized by counting processes described by stochastic differential equations with jumps. The most important, and also most complicated of these counting processes is the exposure counting process $N_i(t)$, which is modeled by a Hawkes process for each agent $i$. Hawkes processes are point processes with a time-dependent, self-exciting conditional intensity function $\lambda_i^*(t)$

$$\lambda_i^*(t) = \beta \sum_{j \in V \setminus \{i\}} \int_{t-\delta}^{t} K_{i,j}(\tau)\, \gamma e^{-\gamma(t-\tau)}\, d\tau \tag{1}$$

where the kernel $K_{i,j}(\tau)$ indicates whether $j$ has been at time $\tau$ at the same site where $i$ is at time $t$, and whether $j$ is in the infectious state. Parameters $\gamma$ and $\delta$ are the decay of infectiousness at sites and the non-contact contamination window, respectively, and they account for the fact that $j$ can infect $i$ even if they are never at the same site, as $j$ can leave some pathogens behind (airborne for instance). Parameter $\beta$ is the transmission rate for symptomatic and asymptomatic individuals, and it comes in two versions: $\beta_c$ accounts for infections outside the household and $\beta_h$ accounts for infection in the household. Parameters $\beta_c$ and $\beta_h$ are fitted to the COVID-19 infection data of Tubingen from 12/03/2020 to 03/05/2020 using Bayesian Optimization. The model also has a parameter for the relative asymptomatic transmission rate built into the function $K_{i,j}(\tau)$, which scales down the infectiousness of asymptomatic agents (to 55% of the infectiousness of symptomatic agents by default).

Once a susceptible agent becomes infected, the disease can take three possible courses (see Fig. 2a). With probability $p_a$, the agent becomes asymptomatic infectious after time $T_E$, and then recovers after time $T_I$. With probability $1 - p_a$, the agent becomes pre-symptomatic infectious after time $T_E$, next symptomatic infectious after time $T_P$, and then recovers with probability $1 - p_h$ after time $T_I - T_P$, or becomes hospitalized with probability $p_h$ after time $T_H$. Agents in the DCS are also assigned age values based on demographic data, and the hospitalization probability $p_h$ of each agent is determined based on its age (following COVID-19 infection data). The times $T_E$, $T_P$, $T_I$ and $T_H$ are drawn from an appropriately parametrized (using values from the COVID-19 lature) lognormal distribution as shown in Table 2.

### *The DDE model*

Our main model of study is inspired by the DCS model (Lorch et al. 2022), but it is significantly simpler to make the theoretical analysis more feasible. In the Deterministically Developing Epidemic (DDE) model, continuous time (used in DCS) is replaced by discrete time-steps: we refer to one time-step in the DDE as one day. Instead of modelling the infection propagation as a Hawkes process, an infectious agent (symptomatic or

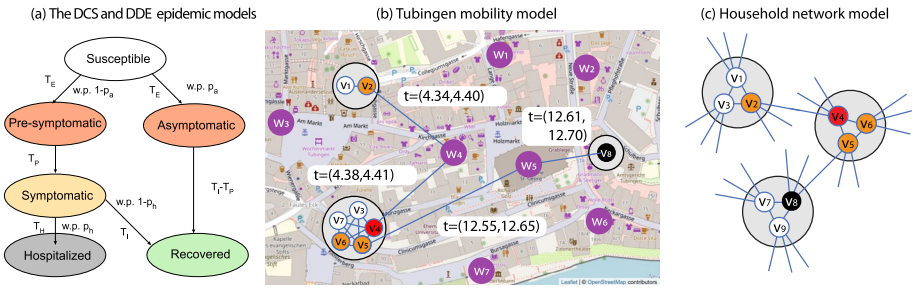


**Fig. 2** **a** The flow diagram of the DCS and DDE epidemic models. **b** A possible epidemic outbreak in the Tubingen mobility model, and **c** the Household network model. The large grey circles mark households, and the purple nodes mark places, otherwise we use the same coloring as in **a**. In both cases **b** and **c**, the transmission paths are ($v_2, v_4, v_5, v_8$). In subfigure **b**, time ($t$) is measured in days, relative to some arbitrary initial time, since the infection time of the source is unknown

**Table 2** Default values for the infection parameters in the DCS+TU and the DDE+HNM models

| Interpretation | Parameter | DCS+TU | DDE+HNM |
|---|---|---|---|
| Exposed time | $T_E$ | Lognormal distribution with $\mu = 3.22, \sigma = 2.3$ | 3 |
| Pre-symptomatic time | $T_P$ | Lognormal distribution with $\mu = 2.3$ and $\sigma = 1$ | 2 |
| Infectious time | $T_I$ | Lognormal distribution with $\mu = 14.0$ and $\sigma = 1$ | 14 |
| Hospitalization time | $T_H$ | Lognormal distribution with $\mu = 7.0$ and $\sigma = 1$ | 7 |
| Probability of asymptomatic | $p_a$ | 0.4 | 0.4 |
| Probability of hospitalization | $p_h$ | Age dependent (mean is 0.0817) | 0.083 |
| Probability of infection | $p_i$ | Hawkes process with various parameters on average $\approx$ 0.02 for a contact | 0.1 |
| External contacts | $d_c$ | From mobility simulation, on average 15 each day | 3 |
| Number of external infections caused by a single agent each day | $d_c p_i$ | On average around 0.3 | 0.3 |
| Household contacts | $d_h$ | From data, on average 1.51 | 2 |
| Number of nodes (agents) | $N$ | 9054 | 400 or 1000 |

asymptomatic) can infect its susceptible neighbor with probability $p_i$ each day. Thereafter, the disease progresses the same way as in the DCS, except that in the DDE model the transition times are deterministic (the infection events and the severity of the disease (i.e., the (a)symptomatic and hospitalized states) are still determined randomly), and we have a single parameter $p_h$ for the hospitalization probability (agents in this model do not have an age parameter). We discuss how we set the parameters of the DDE model in "Parameters" section.

## Simulating mobility

### *Tubingen mobility model*

We briefly review the mobility model introduced in Lorch et al. (2022), and illustrated in Fig. 2b. The population is partitioned into households of possibly varying size (usually between 1 and 5). The households are assigned a location, and we also place some external sites (shops, offices, schools, transport stations, recreating sites) on the map, which the agents may visit. The location of the households and the number of agents in them is sampled randomly based on demographic datasets. Initially, each agent is assigned a few favorite sites (randomly based on distance), and will only visit these throughout the simulation. Each agent decides to leave home after some exponentially distributed time, visits one of its (randomly chosen) favorite sites, and comes back home after another (usually much shorter) exponentially distributed time. If two agents visit the same site at the same time, or within some time $\delta$, we record them as a contact, which gives an opportunity for the infection to propagate. We denote the Tubingen mobility model as TU, and the DCS epidemic model that runs on the TU mobility model as DCS+TU.

### *Household network model*

The Household network model (HNM) was inspired by Lorch et al. (2022), however we note that similar models have been studied in the theoretical community by Ball et al. (2009). As in the Tubingen mobility model, in HNM $N$ nodes are assigned into households, but of constant size $d_h + 1$. Every pair of nodes in the same household are

connected by an edge, forming therefore cliques of size $d_h + 1$. Additionally, each node is assigned $d_c$ half edges, which are paired uniformly at random with other half-edges in the beginning. Some half-edge pairings can result in self-loops or multi-edges, which are discarded. This construction defines a random graph generated by a configuration model, which shares a lot of similarities with Random Regular Graphs (RRG) (Wormald et al. 1999). In fact, if we join nodes in the same household into a single node in the HNM (which we refer to as the *network of households* of the HNM), then the resulting graph is equivalent to the *pairing model* of RRGs with degree $d_c(d_h + 1)$. It is well-known that in the pairing model of RGGs of degree $d$, the local neighborhood (of constant radius, as the number of nodes tends to infinity) of a uniformly randomly chosen vertex is a $d$-regular tree (with probability tending to 1), which implies that locally there are asymptotically almost surely no self-loops, multi-edges or any cycles in the graph. This result has various names; in random graph theory the result is usually proved by subgraph counting (Wormald et al. 1999), in probability theory it is the basis of branching process approximations (Ball et al. 2009), and in graph limit theory it is called the local convergence to the infinite $d$-regular tree (Benjamini and Schramm 2011). In our theoretical analysis, this result motivates the approximation of the neighborhood of the source in the network of households of the HNM by an infinite $d_c(d_h + 1)$-regular tree. The HNM itself is then approximated by replacing each (household) node of the infinite $d_c(d_h + 1)$ -regular tree of households by a $(d_h + 1)$-clique, and by setting the edges so that each (individual) node has degree exactly $d_c + d_h$, while keeping the connection between cliques unchanged (see Fig. 2c for a visualization).

Since the HNM is a time-independent graph, we adopt the standard notations from graph theory. Formally, the HNM is given by the set of nodes and edges $G = (V, E)$. Let us denote by $H(v)$ the set of nodes that are in the same household as node $v$. The distance between two nodes $u, v \in V$ (denoted by $d(u, v)$) is defined as a number of edges of the shortest path between $u$ and $v$. We denote the DDE epidemic model that runs on the HNM network as DDE+HNM.

#### The source identification via contact tracing framework

We present the Source Identification via Contact Tracing Framework (SICTF), which can be applied to both epidemic and mobility models presented so far. The framework determines how the government/health agency, which conducts the source identification task, learns about the outbreak, and how it can gather further information to locate the source. In the SICTF, as in "A simple network and epidemic model and a simple algorithm" section, the agency learns about the outbreak when the first hospitalization occurs, and it also learns the identity of nodes when they become hospitalized (including the identity of the first hospitalized node).

After the outbreak is detected, the agency can make three types of queries.

(1) The *household query* with parameter $v$ reveals the agents that live in the same household as $v$. The household query works the same way in both the TU and the HNM models, and we do not limit the number of times it can be called (these queries are considered as cheap in the SICTF).

(2) The *contact query* reveals the agents who are direct contacts of $v$. It works differently in the TU and the HNM models. For the TU model, a contact query has two parameters: an agent $v$ and a time window $[t_1, t_2]$. As a result, all agents that have been in contact with $v$ (and therefore could have infected $v$ or could have been infected by $v$) at an external site between $t_1$ and $t_2$ are revealed. In the HNM, no time window is needed for the contact query (which we also call edge query), and all neighbors of $v$ in graph $G$ are revealed. Contact (and edge) queries are considered expensive in the SICTF. While in this paper we do not limit the number of available queries, we track the number of contacts and edges that are revealed as the algorithm runs. Note that in the TU model if two agents $v_1$ and $v_2$ have been in contact during the time window $[t_1, t_2]$ and also during a different time window $[t_3, t_4]$, then those are counted as separate contacts, whereas in the HNM an edge between $v_1$ and $v_2$ is only counted once. Although contact queries are considered expensive, both household and contact queries are answered instantly in the SICTF.

(3) The third kind of query is the *test query* with parameter $v$, which reveals information about the course of the disease in the queried agent (see Fig. 2a). Symptomatic patients reveal the time of their symptom onset (which exactly determines their time of infection in the DDE due to the deterministic transition times) if they are past the pre-symptomatic state (i.e., if they are either infectious or recovered). Asymptomatic and pre-symptomatic patients do not reveal any information about their infection time; they just reveal that they have the disease or had the disease at some point and have recovered (this assumption is based on the finding that antibody tests can detect asymptomatic patients (Lei et al. 2021; Kopel et al. 2021)). For all algorithms we assume that asymptomatic patients do not reveal whether they have the infection at the time they are queried. Finally, agents who have not been exposed, or are still in their exposed state, give a negative test result. Test queries are again considered expensive in the SICTF, we even limit the population that can be tested on any given day to at most 1% of the total population, due to the capacity of testing facilities. However, since in this paper we do not limit the number of days that the algorithm can use to locate the source, the limit on the number of tests does not play an important role. As opposed to household and contact queries (and the model in "A simple network and epidemic model and a simple algorithm" section), test results are only answered the next day in the SICTF, which means that the algorithms must operate in "real-time", while the epidemic keeps propagating.

We note that the SICTF shares many similarities with the recently proposed framework of Waniek et al. (2022), with a few notable exceptions: (i) there are no hospitalizations in Waniek et al. (2022), and source detection starts from 10 random nodes after 28 days, (ii) in Waniek et al. (2022) asymptomatic patients reveal their infection time, (iii) there is no household structure in Waniek et al. (2022), (iv) there is only one type of query in Waniek et al. (2022), which is equivalent to our test query, and instead of contact queries, each node reveals 10 of its uniformly random neighbors.

**Parameters**

The DCS+TU model has many parameters, most of which are fitted to COVID-19 datasets of Tubingen from 12/03/2020 to 03/05/2020 by Lorch et al. (2022) (we show the most relevant parameters in Table 2). We determined the parameters of the DDE+HNM model so that they fit the parameters of the DCS+TU as closely as possible (see the precise values in Table 2). We determine the values of $T_E, T_P, T_I$ in the DDE+HNM by rounding the expected value of the corresponding distribution in the DCS+TU to the nearest integer. Since $p_a$ is simply a constant in both models, we keep the same numerical value in the DDE+HNM. The parameter $p_h$ is more complicated, because in the DCS+TU model there is a different hospitalization probability for each age group. We take the average hospitalization probability across the population to be $p_h$. The most complicated parameter to fit is $p_i$, because in the DCS+TU model, infections are modelled by a Hawkes process, which depends on many parameters, including whether the infectious agent is symptomatic or asymptomatic, the length of the visit, the site where the infection happens, etc (see Eq. (1)). We empirically observe the probability of infection in every contact in several simulations, and we find that an agent has on average 15 contacts outside the household each day, and that the average probability of infection during such a contact is around 0.02. However, since we use smaller networks for the DDE+HNM ($N = 400$ or 1000, because running the baselines on larger networks is not feasible) than the DCS ($N = 9054$), setting $d_c$ to be as high as 15 would violate the assumption that the network of households of the HNM can be locally approximated by a tree (see "Household network model" section). Therefore we chose $d_c = 3$ for the HNM and we scale $p_i$ so that $d_c p_i$ (the expected number of external infections caused by a single agent each day) is the same in the DCS+TU and the DDE+HNM models. Finally, we choose $d_h$ in the DDE+HNM by rounding the average household connections in the DCS+TU. Note that the average number of household connections is not the same as the average number of household members, because the number of connections grows quadratically in the size of the households, and thus fitting to the number of connections results in a higher $d_c$ (due to the Quadratic Mean-Arithmetic Mean inequality).

In this paper, we always work with fixed parameters. We believe this is an acceptable assumption in our scenario, since we focus on identifying the source within a few days after the outbreak is detected, and we do not expect the disease parameters to change so quickly (see also Appendix B.1).

Finding the default values for the parameters is useful to create a realistic model. However, we are also interested in the effect of each of the parameters on the performance of our algorithms. Therefore, in the DDE+HNM, we vary the parameters $p_a, p_h, p_i, d_h$ and $d_c$, while keeping the other ones unchanged. For the DCS+TU model, we also keep the mobility model fixed and we focus on varying the parameters $p_a, p_h$ and $p_i$. As noted above, there is no single parameter $p_h$ or $p_i$ in the DCS+TU model, therefore we change all hospitalization probabilities and all intensities of the Hawkes processes so that the hospitalization probability averaged across the population and the infection probability averaged across contacts equal the desired values.

### The LocalSearch algorithms LS and LS+

The LS algorithm finds patient zero by local greedy search. It keeps track of a candidate node, which is always the node with the earliest reported symptom onset time. We denote the candidate of the algorithm at iteration $i > 0$ by $s_{c,i}$. We think of $s_c$ as a list, which is updated in each iteration of the algorithm, and we use the notation $s_{c,-1}$ for the last element of the list (i.e., the current candidate). In each iteration of the algorithm, we compute a new candidate denoted by $s'_c$, and we append it at the end of the list $s_c$ at the beginning of the next iteration, unless $s'_c = s_{c,-1}$, in which case the algorithm terminates.

Since we consider the SICTF, the outbreak is detected when the first hospitalized case is reported. At that time, $s'_c$ is initialized to be the hospitalized patient, the test queue is initialized to be empty, and the algorithm is started. In the beginning of an iteration, if the test queue is empty, the household members and the "backward" contacts of the current candidate $s_{c,-1}$ are queried and are added to the test queue (see Fig. 3a). We define "backward" contacts as the set of nodes that have been in contact with $s_{c,-1}$ in the interval $[t_{s_{c,-1}} - (T_E + T_P) - (\sigma_E + \sigma_P), t_{s_{c,-1}} - (T_E + T_P) + (\sigma_E + \sigma_P)]$, where $t_{s_{c,-1}}$ is the symptom onset time of current candidate $s_{c,-1}$. The terms $\sigma_E$ and $\sigma_P$ model the standard deviation of the transition times, and they are set to zero for the DDE and to $\sigma_E = 2$ and $\sigma_P = 1$ for the DCS based on Table 2. We note that the notion of "backward" contacts is only meaningful in the case of time-dependent network models; for the HNM, all neighbors are counted as backward contacts.

After the test queue is initialized, the agents inside the queue are tested (see Fig. 3b). Not all nodes can be tested on the same day because of the limitation on the number of tests available per day in the SICTF, however, this has little effect because we do not proceed to the next iteration until the test queue becomes empty. Once the test results come back to the agency, if any of the (symptomatic) nodes $v$ reports an earlier symptom onset time than the current candidate $s_{c,-1}$, then we update our next candidate $s'_c$ to be $v$ (see Fig. 3c). We note that the iteration does not stop immediately after $s'_c$ is first updated; the iteration runs until the test queue becomes empty, and until then, $s'_c$ can be updated multiple times. This is important in the theoretical results to prevent the algorithm from

**Algorithm 1.1:** The LS algorithm

```
s_c ← [];
s'_c ← first hospitalized node ;
while s_{c,-1} ≠ s'_c do
    s_c.append(s'_c);
    (a): Add household members and backwards contacts of s_{c,-1} to the
     test queue;
    while the test queue is non-empty do
        (b): Test nodes of the test queue, which were untested for the
         current s_{c,-1};
        for v in test results do
            if v is symptomatic and t_v < t_{s'_c} then
                (c): s'_c ← v;
return s_{c,-1};
```

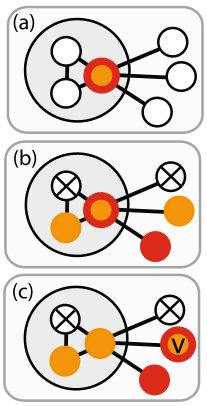


**Fig. 3** Pseudocode and graphical explanation for the LS algorithm. We use the same coloring as in Fig. 2a. Black edges show the queried edges, a node with black cross (X) marks a negative test result, and red stroked node marks the node currently maintained as source candidate by the LS algorithm. We denote by $t_v$ the symptom onset time of symptomatic node $v$ and by $H(v)$ the household of a node $v$ similarly to the main text

getting sidetracked (see Fig. 10). We also experimented with a version of the LS and LS+ algorithms where the iteration stops immediately once $s'_c$ is updated; we call these algorithms LSv2 and LS+v2.

We note that if we tried to adapt the source identification algorithm of Waniek et al. (2022) to the SICTF, we would get back the LS algorithm. Indeed, Waniek et al. (2022) keeps track of a testing queue, and for parameters $\beta_{tr}$ and $\omega_{tr}$, they test the neighbors of the $\beta_{tr}$ earliest nodes $v$ of the testing queue, in the time window $[t_v + \omega_{tr} - 6, t_v + \omega_{tr}]$. They find that for the best source identification results, one should choose both $\beta_{tr}$ and $\omega_{tr}$ as low as possible. In our case, these minimal values amount to $\beta_{tr} = 1$ and $\omega_{tr} = -(T_P + T_E)$, and for these parameters, the algorithm of Waniek et al. (2022) is essentially the same as the LS algorithm.

The main drawback of the LS algorithm is that is gets stuck very easily if there is even one asymptomatic node on the transmission path. For this reason, we introduce the LS+ algorithm, in which we enter the backward contacts of the asymptomatic household members of $s_{c,-1}$, and the household members of any asymptomatic node into the testing queue (see Fig. 4d–f). Since the symptom onset times of asymptomatic nodes $v$ are not revealed, we define backward contact in this case as any contact in the time window $[t_{s_{c,-1}} - (T_P + 2T_E + T_I), t_{s_{c,-1}} - (T_P + 2T_E)]$, where $t_{s_{c,-1}}$ is still the symptom onset time of the current candidate $s_{c,-1}$. Indeed, in the DDE model, since $s_{c,-1}$ was infected at $t_{s_{c,-1}} - (T_P + T_E)$, if $v$ infected $s_{c,-1}$, agent $v$ must have been infectious at that time, which implies that $v$ could not have been infected later than $t_{s_{c,-1}} - (T_P + 2T_E)$ or earlier than $t_{s_{c,-1}} - (T_P + 2T_E + T_I)$. In the DCS model, the terms $\sigma_E$ and $\sigma_P$ can be subtracted and added to the two ends of the queried time window to account for the randomness in the transition times.

## Theoretical results

In this section we present theoretical results for the LS and LS+ algorithms described in "The LocalSearch Algorithms LS and LS+" section. We follow a similar approach as in the non-rigorous computation in "Back of the envelope calculation" section, which is useful but not necessary for understanding this section. All the statements are rigorously

**Algorithm 1.2:** The LS+ algorithm

```
s_c ← [];
s'_c ← first hospitalized node ;
while s_{c,-1} ≠ s'_c do
    s_c.append(s'_c);
    (a): Add household members and backwards contacts of s_{c,-1} to the
     test queue;
    while the test queue is non-empty do
        (b): Test nodes of the test queue, which were untested for the
         current s_{c,-1};
        for v in test results do
            if v is symptomatic and t_v < t_{s'_c} then
                (c): s'_c ← v;
            if v is asymptomatic then
                if v ∈ H(s_{c,-1}) then
                    (d): Add backwards contacts of v to test queue;
                else
                    (e) or (f) Add household members of v to test queue;
return s_{c,-1};
```

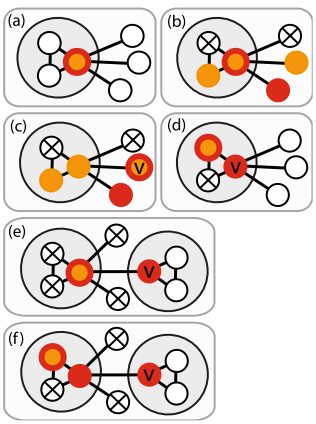


**Fig. 4** Pseudocode and graphical explanation for the LS+ algorithm, similarly to the LS algorihtm in Fig. 3. The difference between the two algorithms is only in the innermost for loop, after the first if statement

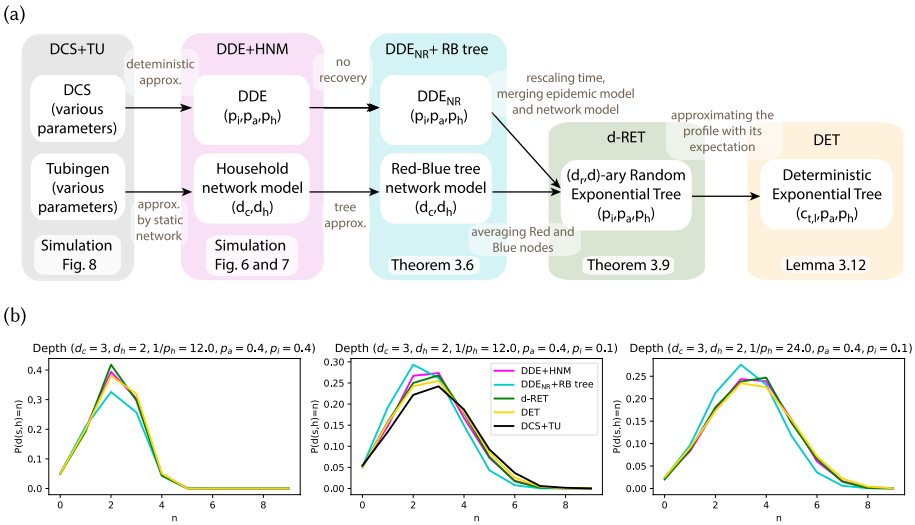


**Fig. 5** The different approximation methods (**a**) and the distribution of the length of transmission path in the different models (**b**) proposed in "Theoretical results" section. Panel **b** also shows the length of the transmission path in the DCS model on the TU dynamics, to highlight the fit of our model

established, and whenever we reach a point where the computations would become intractable, we propose a simpler approximate model to study. One of the main contributions of this paper is to identify which computations can be done on more general models, and which computations need more simplified ones (see Fig. 5 for an overview of the different models used for the computations in this section). In fact, none of our theoretical results hold for the HNM and the DDE models. Already in a first step, we need to approximate the HNM model with a tree graph, because finding the exact infection probabilities of nodes in epidemic models on loopy networks is notoriously difficult (Auffinger et al. 2015).

We compute the source identification probability of the LS and LS+ algorithms in two steps. First, we consider a tree approximation of the HNM called the Red-Blue (RB) tree (defined in "Red-Blue tree models" section), and a slightly modified version of the DDE model called $DDE_{NR}$, and we compute the source identification probabilities conditioned on the length of the transmission path in "Source identification probability of LS and LS+ algorithms on the RB tree" section. This simplification allows us to provide analytical results, while still preserving most of the properties of the original models.

For the second step, we would need to compute the distribution of the transmission path on the RB tree. However, finding a closed form expression is still intractable. Instead, we combine the network and epidemic models into a growing random tree model, and we consider a $d$-ary Random Exponential Tree (RET). The $d$-ary RET model has only been studied for $d = 2$ (Feng and Mahmoud 2018); we extend the results on their expected profile for general $d$ in "$d_r$, $d$-ary random exponential tree" section. Nevertheless, working on $d$-ary RETs still remains difficult, and therefore, in our last modeling step, we introduce a Deterministic Exponential Tree (DET) model,

whose profile is close to the expected profile of the RET, and we compute the distribution of the transmission path on this model in "Deterministic exponential tree with parameters $p_a$,$p_h$ and $(c_{t,I})_{t,I\in\mathbb{N}}$" section.

To summarize all models considered in this paper, we have a data-driven and a synthetic model for simulations (DCS+TU and HNM+DDE), an analytically tractable model (RB-tree+ $\text{tDDE}_{\text{NR}}$) where we can compute the source identification probability if the length of the transmission path is known. In a second stage, we compute the distribution of a transmission path on a deterministic tree (DET), which has a similar profile as a random tree (RET) that approximates our analytically tractable model. We visualize these five different models in Fig. 5a, and we show by simulations in Fig. 5b that the distribution of the transmission path is similar in all of the considered models with appropriately scaled parameters. We compare our analytic results on the source identification probabilities of the LS and LS+ algorithms with our simulation results in "Comparison of simulations and theoretical results" section in Fig. 7.

#### Source identification probability of LS and LS+ algorithms on the RB tree

In this section we introduce the Red-Blue (RB) tree model (which is a tree approximation to the HNM), and we calculate the exact probability that the LS and LS+ algorithms find the source, if the length of the transmission path is known.

##### *Red–blue tree models*

In short, a RB tree is a two-type branching process with a deterministic offspring distribution that depends on $d_h$ and $d_c$. The lack of randomness in this distribution makes us adopt the formalism of deterministic rooted trees.

**Definition 3.1** Let a rooted tree, denoted by $G(s)$, be a tree graph with a distinguished node root node $s$. Let $u$ and $v$ be two nodes connected by an edge in $G(s)$. If $d(u,s) < d(v,s)$, we say that $u$ is a parent of $v$, otherwise $u$ is a child of $v$. Moreover, if $d(s,v) = l$ we say that $v$ is on level $l$. An RB tree with parameters $(d_c, d_h)$ is an infinite rooted tree, such that the nodes also have an additional color property. The root is always colored red and the rest of the nodes are colored red or blue. The root has $d_c$ red and $d_h$ blue children. Every other red node has $d_c - 1$ red and $d_h$ blue children, and every blue node has $d_c$ red children and no blue child. Red nodes and their $d_h$ blue children partition the nodes of the RB tree $G(s)$ into subsets of size $d_h + 1$, which we call households.

***Remark 3.1*** In the RB tree, each blue node has degree $d_c + 1$, and each red node has degree $d_c + d_h$, including the root of the tree $s$ (which is the source of the epidemic, when the RB tree is combined with an epidemic model).

The RB tree can be seen as a local tree approximation of the HNM. Let $G = (V, E)$ be an HNM with parameters $(d_c, d_h)$, and let $s \in V$ be the distinguished source node. In "Household network model" section we noted that the HNM can be approximated

locally around the source node by replacing each node of an infinite $d_c(d_h+1)$-regular tree by a $(d_h+1)$-clique, and setting the edges so that each node has degree exactly $d_c+d_h$, while keeping the connection between cliques unchanged. Let us call this infinite graph $G^*$. Although $G^*$ is not a tree, all cycles in $G^*$ must be contained entirely inside the households, which implies that in each household there exists exactly one node that has the minimal distance to the source. We will refer to these nodes with minimal distance to the source as the red nodes, and we color the rest of the nodes blue. In other words, the red nodes will be the first ones in their households to be infected. Let us now delete the edges between the blue nodes in $G^*$ to obtain graph $G'$. We claim that $G'$ is isomorphic to the RB tree $G(s)$ rooted at the source $s$. Indeed, since the edges between blue nodes have been deleted in $G^*$ to form $G'$, each blue node has $d_c+1$ red neighbors and no blue neighbor, and since the edges incident to red nodes have been unchanged, each red node has $d_c$ red and $d_h$ blue neighbors, exactly as in the definition of RB tree above.

Note that a household in $G^*$ is completely characterized by only specifying the colors of the nodes: a household always consists of one red node and of its $d_h$ blue children. We use this characterization as a definition for households in the RB tree $G'$, because it does not depend on the edges from $G$ that are deleted in $G^*$, whereas this deletion makes the original definition of a household as a clique in $G$ unusable.

Next, we make some important observations the behavior of the LS and the LS+ algorithms on RB trees, which we prove in Appendix C.1. We start by formalizing the notion of transmission path.

**Definition 3.2** Let $h$ be the first hospitalized node and $s$ be the source. We call the path $(s=v_0, v_1, ...v_l=h)$, where $v_i$ is the infector of $v_{i+1}$ for $0 \le i < l$, the *transmission path*. Also we call the path $(v_l, v_{l-1}, ...v_1)$ the *reverse transmission path*.

***Remark 3.2*** Note that in an RB tree, each household traversed by a transmission path shares one (the red node in the household) or two (the red node of the household and one of its $d_h$ children in the household) nodes with this path. Moreover, the red node of a household traversed by a transmission path is followed by another red node on the path (in another household) if it is the only node of that household on the transmission path, whereas it is followed by a blue node (in the same household) if two nodes of that household are on the transmission path.

**Lemma 3.3** *In the RB tree network, the LS algorithm finds the source if and only if all nodes on the transmission path are symptomatic, and the LS+ algorithm finds the source if among the nodes of the transmission path, there exists a symptomatic node in each household, and the source is symptomatic.*

***Remark 3.3*** We note that the statement for LS+ in Lemma 3.3 cannot be reversed, i.e., it is possible that LS+ finds the source even if among the nodes of the transmission path, there is a household with no symptomatic node (see Fig. 10a). Also, the proof of Lemma 3.3 does not hold if the LS+ algorithm proceeds to the next iteration at the first time $s'_c$ is updated (see Fig. 10b). Finally, in the proof of Lemma 3.3, we do not make any assumptions about asymptomatic patients having had the disease previously or not, which implies that we could treat non-complying agents as asymptomatic patients without jeopardizing the correctness of the algorithms.

#### *The* $\mathrm{DDE}_{\mathrm{NR}}$ *model*

Focusing on tree networks is an important step towards making our models tractable for theoretical analysis, but it will not be enough; we will make two minor simplifications to the DDE model as well: we eliminate (i) the pre-symptomatic state and (ii) the recovered state, and we call the new model $\mathrm{DDE}_{\mathrm{NR}}$ (where NR stands for No Recovery). (i) The first assumption can be made without loss of generality, because the pre-symptomatic state does not have any effect on the disease propagation, nor on the success of the LS and LS+ algorithms. Indeed, according to Lemma 3.3, the success of the LS and LS+ algorithms depends only on the information gained about the transmission path, and by the time of the first hospitalization, every node on the transmission path must have left the pre-symptomatic state (since we always have $T_P < T_E + T_H$), even if we include it in the model. (ii) The second assumption on the absence of recovery states amounts to take $T_I \to \infty$, which does have a small effect on the disease propagation, however, this effect is minimal because $T_I = 14$ is already quite large, and because only the very early phase of the infection is interesting for computing the source identification probabilities of the algorithms. Finally, this last assumption has no effect on the information gained by the algorithm since we assumed that recovered patients (who were symptomatic) can remember and reveal their symptom onset time in the same way as symptomatic infectious patients.

#### *Source identification probability of LS*

Assuming that the distribution of length of the transmission path is provided for us (we give an approximation in "Approximating the depth of the path to the first hospitalized node" section), the source identification probability of LS can be computed succinctly. We need a short definition before stating our result.

**Definition 3.4** Let $p$ be the probability that a node is asymptomatic conditioned on the event that it is not hospitalized.

A simple computation shows that

$$p = \mathbf{P}(v \text{ is asy} \mid v \text{ is not hosp}) = \frac{p_a}{p_a + (1-p_a)(1-p_h)}. \tag{2}$$

**Lemma 3.5** *For the* $\mathrm{DDE_{NR}}$ *epidemic model with parameters* $(p_i, p_a, p_h)$ *on the RB tree with parameters* $(d_c, d_h)$*, and with p computed in Eq.* (2)*, we have*

$$\mathbf{P}(LS \text{ finds the source}) = \sum_{n=0}^{\infty} (1-p)^n \mathbf{P}(d(s,h) = n). \tag{3}$$

We defer the proof of the lemma to Appendix C.2.

***Remark 3.4*** Since we approximate the HNM with an infinite tree, we have an infinite sum in Eq. (3). However, since in a real network $d(s, h)$ is upper bounded by the diameter of the network, and since the contributions of the terms drop very fast with $n$, for all practical purposes the sum can be taken until a small finite number instead.

#### *Source identification probability of LS+*

Computing the source identification probability of the LS+ algorithm is far more challenging compared to the LS algorithm, even if the distribution of the length of the transmission path is provided to us. Indeed, since the LS+ algorithm does further testing on the contacts and household members of asymptomatic nodes, it is essential to have additional information about the number of households on the transmission path. We give our main result on the LS+ in the next theorem, which we prove in Appendix C.3.

**Theorem 3.6** *Let p be as in* (2) *and let* $\mathcal{S}(n, \alpha, \beta)$ *be the set of k integer values such that k and n have different parity and* $n + 1 - 2(\alpha + \beta) \geq k \geq 2 - (\alpha + \beta)$*. Then, for the* $\mathrm{DDE_{NR}}$ *epidemic model with parameters* $(p_i, p_a, p_h)$ *on the RB tree with parameters* $(d_c, d_h)$*, we have*

$$\begin{aligned}\mathbf{P}(LS+ \text{ finds the source}) &\geq \mathbf{P}(d(s,h)=0) + (1-p)\mathbf{P}(d(s,h)=1) \\ &+ \sum_{n=2}^{\infty} \sum_{\substack{\alpha,\beta \in \{0,1\} \\ k \in \mathcal{S}(n,\alpha,\beta)}} \binom{\frac{n+k-3}{2}}{k-2+\alpha+\beta} \frac{(d_h(1-p))^{\frac{n+k-1}{2}} (d_c(1+p))^{\frac{n-k+1}{2}-\alpha-\beta} d_c (d_c-1)^{k+\alpha+\beta-2}}{\lambda_1 \left(\frac{d_c-1+D}{2}\right)^n + \lambda_2 \left(\frac{d_c-1-D}{2}\right)^n} \mathbf{P}(d(s,h)=n),\end{aligned} \tag{4}$$

*where*

$$D = \sqrt{(d_c-1)^2 + 4 d_c d_h} \tag{5}$$

$$\lambda_1 = \frac{(d_c + 1 + D)(2d_h + d_c - 1 + D)}{2D(d_c - 1 + D)} \tag{6}$$

$$\lambda_2 = \frac{(D - d_c - 1)(2d_h + d_c - 1 - D)}{2D(d_c - 1 - D)}. \tag{7}$$

The formula in Theorem 3.6 still assumes that the distribution of the length of the transmission path $\mathbf{P}(d(s, h) = n)$ is provided for us. We approximate this distribution analytically in the next section, and we give an interpretation of Theorem 3.6

afterwards in Remark 3.6, "Comparison of simulations and theoretical results" section and Appendix A.

The additional complexity in (4) compared to (3) comes from the fact that unlike LS, the LS+ algorithm makes use of the household structure of the network, and we need to count of the number of paths of a given length $n$ in the RB tree separately depending on how the path is embedded in the household structure. This requires us to introduce some binary parameters $\alpha$ (resp, $\beta$) that indicate whether the source node (resp, the last node of the path) is in the same household as another node on the path, and an integer $k$ counting the number of nodes not sharing a household with another node on the path. The precise definition of these parameters can be found in Definition C.5 in Appendix C.3.

#### Approximating the depth of the path to the first hospitalized node

##### ($d_r, d$)-*ary random exponential tree*

When we introduced the $\mathrm{DDE}_{\mathrm{NR}}$ model in "The $\mathrm{DDE}_{\mathrm{NR}}$ model" section, we removed both parameters $T_P$ and $T_I$ from the DDE model (by removing the presymptomatic and the recovered states, respectively), but we kept the parameter $T_E$. In this step we will rescale the time parameter to make $T'_E = 1$ by changing $p'_i$ to be $1-(1-p_i)^{T_E}$. Since we had $T_E = 3$ by default, using $T'_E$ and $p'_i$ instead of $T_E$ and $p_i$ means that we choose 3 days to be our time unit, and the probability of infection is scaled to be the probability that the infection is passed in at least one of three days (since the RB tree is time-independent, if two nodes are connected, the infection can spread on it every day). We drop the prime from $p'_i$ and $T'_E$ for ease of notation. As a second approximation, instead of keeping track of two types of nodes (red and blue) as it is done in the RB tree, we propose to change our network model to an infinite $d$-regular tree, where $d$ is set to be the average degree of an RB tree.

By making these two changes (tracking time at a coarser scale and simplifying the network topology to a $d$-regular tree), the growth of the epidemic becomes equivalent to a known model, the $d$-ary Random Exponential Tree ($d$-RET). Binary RETs have been introduced in Feng and Mahmoud (2018). We give the definition below for completeness.

**Definition 3.7** A $d$-ary Random Exponential Tree ($d$-RET) with parameters $d, p_i$ at time day $t$, denoted by $G_t(s)$, is a random tree rooted at node $s$. At day 0, the tree $G_t(s)$ only has its root node $s$. Let $\bar{G}_t(s)$ be the closure of $G_t(s)$, which is obtained by attaching external nodes to $G_t(s)$ until every internal node (a node that was already present in $G_t(s)$) has degree exactly $d$ in the graph $\bar{G}_t(s)$. Then, $G_{t+1}(s)$ is obtained from $\bar{G}_t(s)$ by retaining each external node with probability $p_i$, and dropping the remaining external nodes.

Indeed, each node of a $d$-RET infects a new node with probability $p_i$ each day, and after a sufficiently long time, the $d$-RET becomes close to a large $d$-ary tree. Of course, we do not want to let the $d$-RET grow for a very long time, we only want it to grow

until the first hospitalization occurs. So far we have not talked about the course of the disease of the nodes in the $d$-RET model because we could define the spread of the infection without it. Since we still need to do one final simplification to compute the distribution of the transmission path, we defer the discussion about hospitalizations, and how the parameters $p_a$ and $p_h$ are part of the model, to "Deterministic exponential tree with parameters $p_a$,$p_h$ and $(c_{t,I})_{t,I\in\mathrm{N}}$" section. Note that by considering the $d$-RET, we deviate from the idea of separating the epidemic and the network models; we only have a randomly growing tree, which is stopped at some time, when the tree is still almost surely finite.

So far we only did simplifications to the model, which resulted in further and further deviations from the original version. Now we will make a small modification that brings our model back closer to the RB tree, without complicating the computations too much. We still make almost all maximum degrees of the RET uniform $d$, but we make an exception with the root, which will have maximum degree $d_r = d_c + d_h$. This makes the maximum degree of the root the same as the degree of the root of the RB tree. We call the resulting model a $(d_r, d)$-RET with parameter $p_i$. Since the close neighborhood of the source has a high impact on the source identification probability, we found that this solution gives the best results while keeping the computations tractable.

In our computations, only the profile the infection tree will be important, which motivates the next definition.

**Definition 3.8** In the $(d_r, d)$-RET model with parameter $p_i$, let $A_{t,l}$ be the number of nodes during day $t$ at level $l$, and let $a_{t,l} = \mathbb{E}[A_{t,l}]$. Moreover, we define the random variable

$$A_t = \sum_{t=0}^{+\infty} A_{t,l} \tag{8}$$

with $A_{-1,l} = 0$ for all $l$, and its expectation $a_t = \mathbb{E}[A_t]$.

As noted earlier, the $d$-RET model has only been analyzed for $d = 2$ to this date. We provide the expected number $a_{t,l}$ of nodes at level $l$ in day $t$ for the general case in the next theorem and corollary, which we prove in Appendices C.4 and C.5.

**Theorem 3.9** *In the $(d_r, d)$-RET with parameter $p_i$, let $a_{t,l}$ be as in Definition 3.8. Then*

$$a_{t,0} = 1 \tag{9}$$

$$a_{t,l} = d_r p_i \sum_{m=l-1}^{t-1} \binom{m}{l-1} (1-p_i)^{m-l+1} d^{l-1} p_i^{l-1}, \text{ for } t \geq l \geq 1 \tag{10}$$

$$a_{t,l} = 0, \text{ for } l > t. \tag{11}$$

**Corollary 3.10** *In the RET*$(p_i, d_r, d)$*, let* $a_t$ *be the expectation of* (8)*, as in Definition* 3.8*. For* $t \geq 0$*,*

$$a_t = 1 + d_r \frac{(1 - p_i + dp_i)^t - 1}{d - 1}. \tag{12}$$

***Deterministic exponential tree with parameters*** $p_a, p_h$ ***and*** $(c_{t,l})_{t,l\in\mathbb{N}}$

In the $(d_r, d)$-RET model it is still complicated to calculate the distribution of the depth of the first hospitalized node. For this reason, we approximate the RET model by a deterministic time-dependent tree with a prescribed profile.

**Definition 3.11** Let $(c_{t,l})_{t\in\mathbb{N}\bigcup\{-1\},l\in\mathbb{N}}$ be a two-dimensional array with $c_{t,l} = 0$ for $t \in \{-1, 0\}$ and $l \in \mathbb{N}$, except for $c_{0,0} = 1$, and with $c_{t,l} \geq c_{t,l-1}$ for any $t$ and any $l \geq 1$. Additionally, if we define $c_t = \sum_l c_{t,l}$, then the array $(c_{t,l})$ must satisfy $c_t > c_{t-1}$ for $t \geq 0$. Then, we define the Deterministic Exponential Tree (DET) with parameter $(c_{t,l})_{t\in\mathbb{N}\bigcup\{-1\},l\in\mathbb{N}}$, as a time-dependent rooted tree, that has exactly $c_{t,l}$ nodes on level $l$ at time $t$. The edges between the adjacent levels are drawn arbitrarily so that the tree structure is preserved.

The formal assumptions on the array $(c_{t,l})$ are simply made to ensure that the DET starts with a single node at $t = 0$, that it never shrinks on any level $(c_{t,l} \geq c_{t,l-1})$, and that it grows by at least one node in each time step $(c_t > c_{t-1})$.

We have defined the DET at any given time $t$, however, to determine the length of the transmission path, we are not interested in the DET at any given time, but only when the first hospitalization occurs.

To compute the distribution of the first hospitalized node, we would like to have an absolute order on the times when the nodes are added, which we do by randomization. We say that on day $t$, nodes are added one by one to the DET, their order given by a uniformly random permutation, and each node is hospitalized with probability $(1 - p_a)p_h$ (as in the original DDE model). When the first hospitalization occurs, we stop growing the tree, and we call the resulting (now random) model a stopped DET with parameters $(c_{t,l}), p_a, p_h$. We find the transmission path length distribution on the stopped DET in the next lemma, which we prove in Appendix C.6.

**Lemma 3.12** *Let us consider the stopped DET model with parameters* $(c_{t,l}), p_a, p_h$*, and let h denote the first hospitalized node. Then*

$$\mathbf{P}(d(s,h) = l) = \sum_{t=0}^{+\infty} \frac{c_{t,l} - c_{t-1,l}}{c_t - c_{t-1}} (1 - (1 - p_a)p_h)^{c_{t-1}} \left(1 - (1 - (1 - p_a)p_h)^{c_t - c_{t-1}}\right). \tag{13}$$

We would like to set $c_{t,l}$ so that the DET is close to the RET described in "$d_r$,$d$-ary random exponential tree" section. For Eq. (13) to make sense, we should substitute integer values for $c_{t,l}$, however, for an approximation, the equation can also be evaluated for fractional values as well.

***Remark 3.5*** If we substitute $c_{t,l} = a_{t,l}$ and $c_t = a_t$ in Eq. (13), where $a_{t,l}$ is given in Theorem 3.9 and $a_t$ is computed in Corollary 3.10, then we get the expression

$$d_r p_i^{l-1} d^{l-1} \sum_{t=l}^{+\infty} \frac{\binom{t-1}{l-1}(1-p_i)^{t-l}}{(1-p_i+dp_i)^{t-1}} (1-(1-p_a)p_h)^{1+d_r \frac{(1-p_i+dp_i)^{t-1}-1}{d-1}} \left(1-(1-(1-p_a)p_h)^{d_r(1-p_i+dp_i)^{t-1}}\right), \tag{14}$$

which approximates the distribution of the transmission path length in the $(d_r, d)$-ary RET stopped at the first hospitalization.

***Remark 3.6*** To arrive to the source identification probabilities of LS and LS+, Eq. (14) needs to be substituted into Lemma 3.5 and Theorem 3.6. We perform this task numerically in Fig. 7 to see how close our analytic approximations are to the original HNM+DDE model. However, the main value of these theoretical tools is that by studying the analytical expressions, we can gain insight into the mathematical properties of the source identification probability function (see Sects. Comparison of simulations and theoretical results and Appendix A).

## Simulation results

### Baseline algorithms

#### *Non-adaptive baseline: dynamic message passing*

Besides Waniek et al. (2022), which is essentially equivalent to the LS algorithm (see "The LocalSearch Algorithms LS and LS+" section), there are few source identification algorithms that are compatible with time-varying networks in the literature (Huang 2017; Jiang et al. 2016; Fan et al. 2020; Chai et al. 2021). The most promising one among these algorithms (Jiang et al. 2016) has a close resemblance to the a previous work of Lokhov et al. (2014) on Dynamic Message Passing (DMP) algorithms. Given the initial conditions on the identity of the source node and its time of infection, the DMP algorithm approximates the marginal distribution of the outcome of an epidemic at some later time $t$. The algorithm is exact on tree networks, and it computes a good approximation when there are not too many short cycles in the network. Therefore, the DMP algorithm can be used to approximate the likelihood of the observed symptom onset times for any (source,time) pair. Due to its flexibility, we were able to adapt the DMP algorithm to the SICTF (see Appendix D for more details).

Originally, the DMP was applied to the source identification problem by computing the likelihood values for all possible (source,time) pairs, and then choosing the source node from the most likely pair as the estimate (Lokhov et al. 2014). However, testing all (source,time) pairs increases the time complexity of the algorithms potentially by a factor

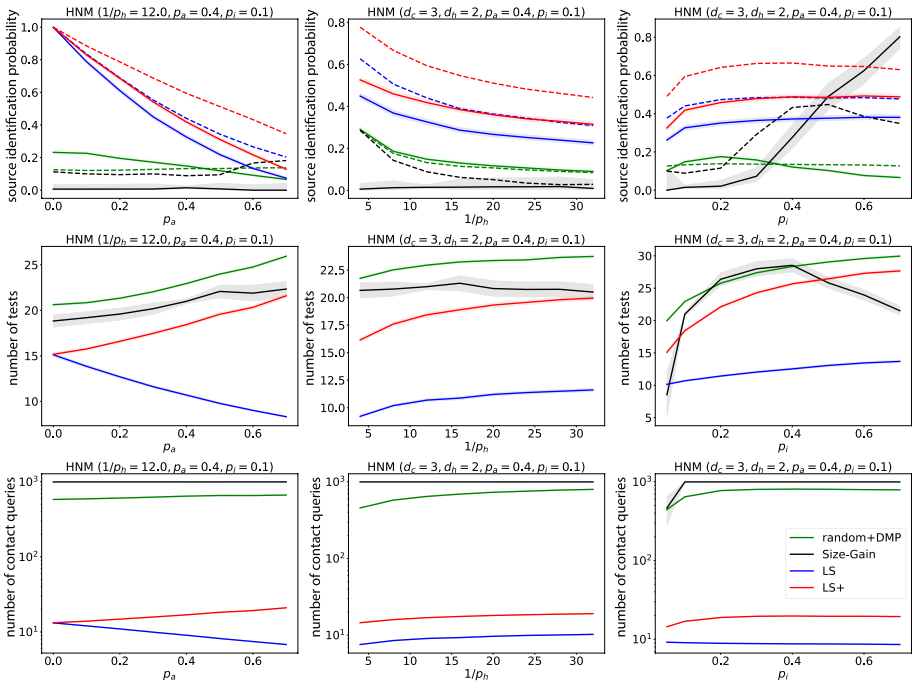


**Fig. 6** The performance of the algorithms LS, LS+, R and SG if the metric is the probability of finding the source (solid curves) or the first symptomatic patient (dashed curves). The simulations were computed on a population of $n = 400$ individuals in the DDE model on the HNM, and each datapoint is the average of 4800 independent realizations except for the SG algorithm, which was run with 192 independent realizations. The confidence intervals for the source identification probabilities are computed using the Wilson score interval method, and for the tests and the queries using the Student's t-distribution

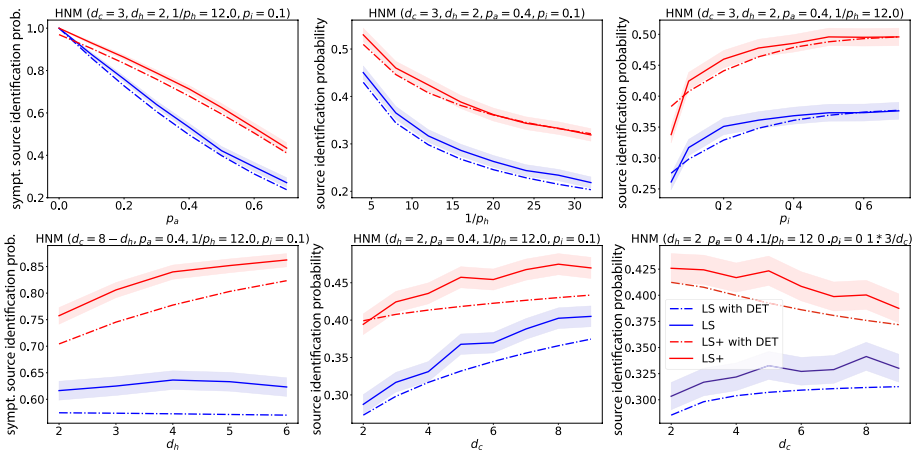


**Fig. 7** The source identification probability of the LS and LS+ algorithms (solid curves) and their theoretical estimate (dash-dotted curves) with the source identification probabilities computed in Lemma 3.5 and Theorem 3.6, while the transmission path distribution computed in Eq. (14) until path length $d(s, h) = 20$ (which is larger than the diameter of the simulated networks, see Remark 3.4). The simulation results were generated using the DDE model on HNM networks of size $n = 1000$ with 4800 independent samples. The 95% confidence intervals are computed using the Wilson score interval method

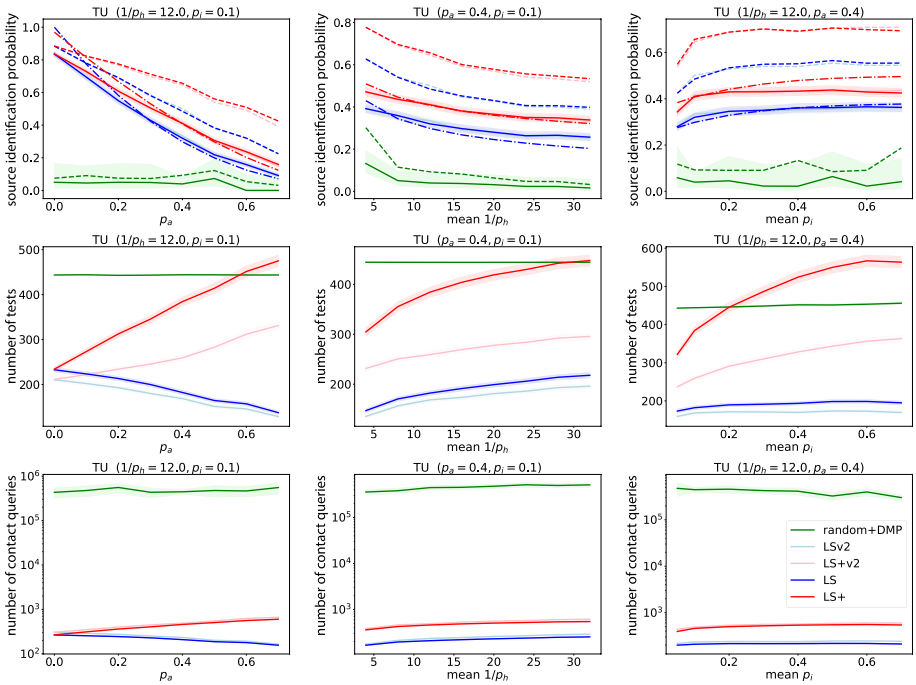


**Fig. 8** The performance of the algorithms LS, LS+ and random+DMP on the DCS model with the Tubingen dynamics if the metric is the probability of finding the source (solid curves) or the first symptomatic patient (dashed curves), together with the theoretical results (dash-dotted lines), as shown in Fig. 7. The simulations were computed on a population of $n = 9054$ individuals, and each datapoint is the average of 2400 independent realizations for the LS/LS+/LSv2/LS+v2 algorithms, and 48 independent realizations for the random+DMP algorithm. The default population and infection parameters were selected to match the population and COVID-19 infection datasets of Tubingen. As in all of the experiments, the algorithms were allowed to test 1% of the population (in this case 90 individuals) each day, and the algorithms finished (successfully or not) after 3–8 days after the first hospitalization occured

of $N^2$, which makes the algorithm intractable in many applications. Jiang et al. (2016) proposed a very similar algorithm to the DMP equations (which is unfortunately not exact even on trees), and solved the issue of intractability by a heuristic preprocessing step to the DMP algorithm. This preprocessing step identifies a few candidate (source,time) pairs, by spreading the disease backward from the observations in a deterministic way (called reverse dissemination). Since we already approximate our data-driven model (DCS) by an epidemic model with deterministic transition times (DDE), it is natural for us to also implement the deterministic preprocessing step proposed by Jiang et al. (2016). We produce 5 (source,time) pairs which are feasible for the 5 earliest symptom onset time observations (see Appendix D.3 for more details). It would have been ideal to run the algorithms for more than 5 pairs, but this was made impossible by the runtimes becoming very high. We run therefore our implementation of the DMP algorithm with the previously computed feasible (source,time) pairs as initial conditions to find the most likely source candidate.

The source estimation algorithms developed using the DMP algorithm do not specify how the sensors should be selected, and therefore place these non-adaptive sensors randomly. We refer to the resulting algorithm as random+DMP. The number of sensors is set so that it always exceeds the number that LS/LS+ would use. The simulation results are shown in Fig. 6 for the DDE+HNM model. Importantly, the deterministic preprocessing

step of Jiang et al. (2016) is compatible with time-varying networks, which allows us to run the algorithm for the DCS+TU model as well (see Fig. 8).

#### *Adaptive baseline: size-gain*

The Size-Gain (SG) algorithm was developed for epidemics which spread deterministically (Zejnilović et al. 2015), and has been later extended to stochastic epidemics (Spinelli et al. 2017b). It works by narrowing a candidate set based on a deterministic constraint. If $\nu_1, \nu_2$ are symptomatic observations, then $s_c$ is in the candidate set of SG if and only if

$$|(t_{\nu_2} - t_{\nu_1}) - (d(\nu_2, s_c) - d(\nu_1, s_c))| < \sigma\,(d(\nu_2, s_c) + d(\nu_1, s_c)), \tag{15}$$

where $\sigma$ is the standard deviation of the infection time of a susceptible contact. If one of the observations, say $\nu_2$, is negative, then SG uses a condition almost identical to Eq. (15), except that the absolute value is dropped, since a negative observation at time $t_{\nu_2}$ is only a lower bound on the true symptom onset time of $\nu_2$. These deterministic conditions are checked for every symptomatic-symptomatic or symptomatic-negative pair $(\nu_1, \nu_2)$ to determine if $s_c$ can be part of the candidate set. Next, SG places the next sensor adaptively at the node which reduces the candidate set by the largest amount in expectation (assuming a uniform prior on the source and its infection time), and it terminates when the candidate set shrinks to a single node. Note that the SG algorithm can fail if at least one of the deterministic conditions in Eq. (15) is violated for some $(\nu_1, \nu_2)$ because of the randomness of the epidemic.

We use the existing implementation of the SG algorithm by Spinelli et al. (2017b), and adapt it to the SICTF. We incorporate asymptomatic-symptomatic and asymptomatic-negative observations $(\nu_1, \nu_2)$ the same way as symptomatic-negative are incorporated; we drop the absolute value sign in Eq. (15), because an asymptomatic observation at time $t_{\nu_1}$ is only an upper bound on the true symptom onset time of $\nu_1$. We impose the same daily limit to the number of sensors that can be placed by the SG algorithm in a single day as for the LS/LS+ algorithm, and if the candidate set size does not shrink to one on the day when both LS and LS+ have already provided their estimates, then the SG algorithm must make a uniformly random choice from the current candidate set as its source estimate. The simulation results are shown in Fig. 6 for the DDE+HNM model. We do not implement the SG algorithm for the DCS+TU model, because its runtime is too high, and because it is not clear how it should be implemented for time-varying networks.

#### Comparison with baselines

We show our simulation results comparing the random+DMP, SG, LS and LS+ algorithms in Fig. 6. In the first row of Fig. 6, we show the accuracy of the algorithms with solid curves. Since the LS/LS+ algorithms cannot identify an asymptomatic source, we also show what the accuracy would look like if the goal of the SICTF was to identify the first symptomatic agent with dashed lines. It is clear that in both metrics and across a wide range of parameters, the LS+ algorithm performs best, followed by LS, next random+DMP, and finally SG. The only exception is for high values of $p_i$, where SG performs best. The good performance of SG for these

parameters is expected, because SG was originally developed for deterministically spreading epidemics (i.e., $p_i = 1$). For the other parameter ranges, one could argue that the comparison is not fair, since the baseline algorithms were developed for different frameworks. Nevertheless, as any with new framework, we find it important to quantify the performance of the previously proposed algorithms in the SICTF, to motivate the need for the LS and the LS+ algorithm.

In the second row of Fig. 6, we show the number of test/sensor queries used by the algorithms. LS uses the fewest tests, followed by LS+ (except for large values of $p_i$). Finally, in the last row of Fig. 6 we show the number of contact (or in this case edge) queries used by the algorithms. Again, LS uses fewer queries than LS+, while both the random+DMP and SG algorithms query essentially the entire network.

Figure 6 shows that the LS/LS+ algorithms are fairly robust to changes in the parameters of the model, except for the parameter $p_a$. Indeed, if there are many asymptomatic nodes in the network, then source identification becomes very challenging. It may be surprising that as $p_a$ grows, the number of tests that LS uses decreases, contrary to LS+. This is because as $p_a$ grows, the LS algorithm gets stuck more rapidly, while the LS+ algorithm compensates for the presence of asymptomatic nodes by using more test/sensor queries.

#### Comparison of simulations and theoretical results

The analytic results from "Theoretical results" section are in good agreement with the simulation results in Fig. 7. We also experiment with changing the parameters $d_h$, $d_c$ while keeping all the parameters fixed, and with changing $d_c$ while keeping the product $d_c p_i$ fixed. We observe that LS is not affected by the parameter $d_h$, whereas LS+ performs better with a higher $d_c$, which is expected because LS+ leverages the household structure of the network to improve over LS. Surprisingly, we also observe that a higher $d_c$ also improves the performance of both algorithms. This can be explained by the fact that a larger $d_c$ implies that there are more nodes in the close neighborhood of the source, which results in shorter transmission paths, making source identification less challenging. Finally, if we increase $d_c$ but keep $d_c p_i$ fixed, the performance of the algorithms does not change as much, which confirms the intuition that it is the number of infections caused by an infectious node in a single day that matters the most. These qualitative results are in agreement with our naive approximations in Appendix A and by the simulation results of Waniek et al. (2022) in a similar source identification framework.

#### Simulations on the DCS model

As validation, we show our simulation results on our most realistic DCS+TU model in Fig. 8. We make very similar observations on this model as the ones that we have made on the DDE+HNM model in "Comparison with baselines and Comparison of simulations and theoretical results" sections, which shows that the LS/LS+ algorithms and our analysis of their performance is robust to changes in the epidemic and network models.

In the DCS+TU model, we used a fixed limit on the number of sensors that the random+DMP model selects, instead of setting the limit based on the LS+ algorithm. As

a result, for a few parameters the LS+ algorithm used more tests than the random+DMP model. However, we note that by updating the candidate node immediately after an earlier symptom onset time is revealed (see "The LocalSearch Algorithms LS and LS+" section), we can essentially cut the number of required tests for the LS+ algorithm by half (LSv2 and LS+v2), without sacrificing the performance of the algorithms.

## Discussion

We introduced a new source identification framework, the SICTF, and two source identification algorithms, the LS and the LS+. We find that both LS and LS+ outperform baseline algorithms, even though the baselines essentially query all contacts on a transmission path between agents, while LS and LS+ query only a small neighborhood of the source.

We showed that the LS/LS+ algorithms are robust to changes in the the parameters, and also changes in the epidemic and the mobility models. We supported our arguments by theoretical analysis, and by simulation results on a state of the art COVID-19 simulator developed in Lorch et al. (2022).

Theoretical analysis of source identification is a challenging mathematical task on loopy graphs. In this paper, we provide results on a series of tree approximations of the original graph. It is an interesting future direction to quantify how tight these approximations are, or to provide tight theoretical results without tree approximations.

Another interesting question is whether the LS/LS+ can be improved to also be robust to imprecise reporting, or the presence of multiple variants of the same disease. On the more applied side, we believe that the "low-tech" approach in the design of the LS/LS+ algorithms increases their potential to be implemented in real-world scenarios, similarly to contact tracing applications (Kendall et al. 2023; Troncoso et al. 2020).

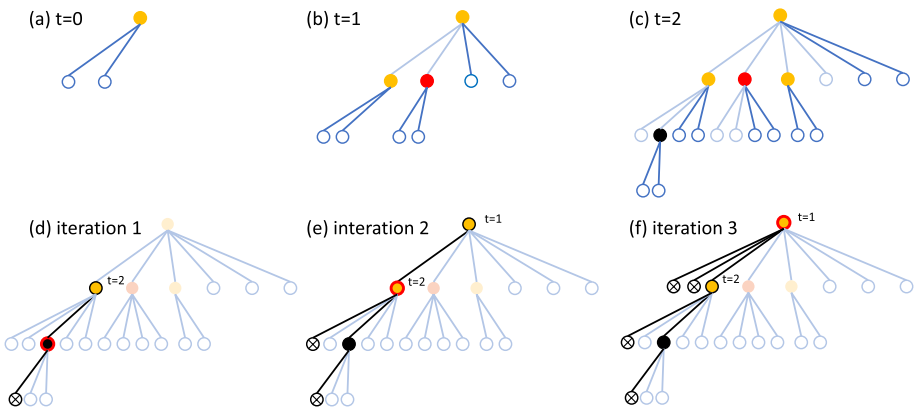


**Fig. 9** **a**–**c** shows the spread of the infection in the model considered in "A simple network and epidemic model and a simple algorithm" section, which is equivalent to the growth of the RERT, with $d = 2$. Dark blue edges show the contacts on day $t$, and light blue edges show contacts present on previous days (and thus subfigures). Orange (resp., red; black) nodes mark symptomatic non-hospitalized (resp, asymptomatic; symptomatic hospitalized) nodes. **d**–**f** shows the LS source identification algorithm introduced in "Back of the envelope calculation" section, which finds the source in this example because there are no asymptomatic nodes on the transmission path between the first hospitalized node and the source. Black edges show the queried edges, and black stroke marks nodes already discovered by the algorithm. A node with black X marks a negative test result, and red stroked node marks the node currently maintained as source candidate by the LS algorithm

## Appendix A: Warmup results

### A.1: A simple network and epidemic model and a simple algorithm

Let us consider a time-dependent network model, where each agent meets $d$ new agents each day in such a way that the contact network is an infinite tree (ignoring the label of the edges giving the propagation time along the edge). This network models homogeneous mixing in a very large population; we consider more realistic network models in "Models, methods, algorithms" section. On this network, we consider an epidemic model that starts at $t = 0$ with one infected agent, and then progresses as infected agents infect their $d$ susceptible contacts each independently with probability $p_i$ each day. Since our goal is to study the epidemic process, it is sufficient to track only the agents who are already infectious (also called *internal nodes*), and the agents who are in contact with infectious agents at time $t$ (also called *external nodes*), as shown in Fig. 9a–c. For $d = 1$, the spread of the infection is then equivalent to the growth a random tree $\mathcal{T}_t$ rooted at the source of the infection, known under the name of Random Exponential Recursive Tree (RERT) and recently introduced in Mahmoud (2021). Because of the similarities of the models, we refer to the model with general $d$ as RERT in the remaining of this section. We point out that the standard literature on elementary branching processes such as Galton-Watson trees or random recursive trees (Drmota 2009) is not applicable in our scenario, because these branching processes have no notion of global time (i.e., a node in such processes becomes infectious immediately after receiving the infection), whereas nodes in diseases commonly go through an exposed, non-infectious period before becoming infectious, which is well captured by the RERT model. We mention that there is literature on more advanced branching processes that do have a notion of global time, e.g. Crump-Mode-Jagers trees (Jagers and Nerman 1984), however we opt for the RERT because of its simple definition.

After a node (patient) becomes infected, the disease can take three courses (which for now do not affect $\mathcal{T}_t$): with probability $p_a$ the patient is asymptomatic, with probability $(1 - p_a)p_h$ the patient is hospitalized, and with probability $(1 - p_a)(1 - p_h)$ the patient recovers without hospitalization. The governmental/health agency learns about the outbreak when the first hospitalization occurs (see Fig. 9c) and starts the source identification process right away. It can inquire about the contacts of each agent and it can test the agents. From patients that were symptomatic (at any point in time in the past), the agency learns about their symptom onset time (which, in this simple model, is always one day after the infection time), but from asymptomatic patients it only learns that they had (or have) the disease at some point when they are tested. The framework introduced in this paragraph (including both the identification of the outbreak through the first hospitalization, and the possible actions the agency can take) is a simplified version of the SICTF (Source Identification via Contact Tracing Framework), introduced in "The source identification via contact tracing framework" section.

The network and epidemic models introduced in this section have four parameters: $d, p_i, p_a, p_h$, and it is important to understand how each of them affects the difficulty of source identification in the SICTF. We distinguish two important factors. First, if the

outbreak is not detected rapidly enough, the length of the transmission path to the first hospitalized agent is long, and source identification becomes then difficult, because a lot of information needs to be recovered. Therefore, a low $p_i$, a low $p_h$ and/or a high $p_a$ parameter can hinder source identification (recall that the probability of hospitalization was $p_h(1-p_a)$). The second factor is related to the difficulty of recovering information about the transmission path. If $p_a$ is high, then there are a lot of nodes who are asymptotic and therefore do not reveal their symptom onset time, making source identification very difficult. Since $p_a$ affects both the length of the transmission path and the amount of collected information, it is safe to expect that, of all parameters, $p_a$ has the largest effect on the difficulty of source identification. The parameter $d$ is interesting, because a large $d$ can reduce the length of the transmission path, but it also makes the information about the transmission path less accessible as more agents need to be tested. Since in this paper we do not set a hard constraint on the total number of available tests, the advantage of a shorter path takes over the drawback of additional tests and a large $d$ increases the source identification probability. These qualitative results are in agreement with the results of Waniek et al. (2022), and we also confirm them in the SICTF by simulation in "Simulation results" section.

To say anything quantitative about source identification in the SICTF, we must discuss specific algorithms that solve the source identification task. In this paper we propose a simple algorithm called LocalSearch (LS), shown in Fig. 9d–f. The LS algorithm maintains one candidate node $s_c$ at each iteration (initially, the first hospitalized node), which is always symptomatic, and it updates it in a greedy way: at the time of the infection of $s_c$, all its $d$ incident edges are queried, and all its $d$ neighbors are tested. Then the agent with the lowest reported infection time will be the new candidate $s_c$. The algorithm stops when $s_c$ does not change anymore between two consecutive iterations. For simplicity, we assume that the infection does not spread any further during these iterations, however, this assumption does not affect the ability of the algorithm to find the source or not. Indeed, it is not difficult to see that on tree networks, LS finds the source if and only if there are no asymptomatic nodes on the transmission path from the source to the first hospitalized agent. This observation leads us to enhance the LS algorithm by also searching within the neighbors of asymptomatic nodes; we explore this idea in the LS+ algorithm introduced in "The LocalSearch Algorithms LS and LS+" section. We are not aware of this simple greedy algorithm being studied in the context of source identification, although similar ideas were implemented for non-adaptive source identification to lower the runtime of the algorithms (Paluch et al. 2018).

#### A.2: Back of the envelope calculation

Now, we have all the tools to estimate the source identification probability of the LS algorithm. First we condition on the course of the disease in the source. With probability $p_a$, the source is asymptomatic and LS can never find the source. With probability $(1-p_a)p_h$, the source itself becomes hospitalized, and LS always finds the source. Finally, with probability $(1-p_a)(1-p_h)$ the source is symptomatic but not hospitalized,

which we call event $\mathcal{A}$. If event $\mathcal{A}$ happens, then LS may or may not find the source depending on whether there are any asymptomatic nodes on the transmission path. More precisely, conditioned on event $\mathcal{A}$ and on the transmission path having length $l$, the source identification probability is $(1-p_a)^{l-1}$ (since there are $l-1$ nodes on the path which can be asymptomatic), which implies

$$\begin{aligned}&\mathbf{P}(\text{LS finds source})\\&= (1-p_a)p_h + (1-p_a)(1-p_h)\left(\sum_{l=1}^{t}\mathbf{P}\big(\text{transmission path has length } l \mid \mathcal{A}\big)(1-p_a)^{l-1}\right).\end{aligned} \tag{16}$$

The difficult part is to compute the distribution of the transmission path conditioned on event $\mathcal{A}$; indeed we already saw that all four parameters $d, p_i, p_a, p_h$ affect this distribution in a non-trivial way. Let us perform a back of the envelope computation to get more insight into the effect of these parameters. The exact structure of the infection tree will not matter for this computation, only its *profile* does. It is denoted by $\mathcal{T}_t(l)$ and defined as the number of (internal) nodes at level $l$ (i.e., at distance $l$ from the source of the infection). Remember that by definition the RERT has $d\cdot\mathcal{T}_{t-1}(l-1)$ external nodes on level $l$, and that at time $t$ each external node is promoted to be internal with probability $p_i$ to form $\mathcal{T}_t$. Consequently, the level of a node $h$ added at time $t>0$ has the same distribution (conditioned on the tree $\mathcal{T}_{t-1}$ at the previous step) as the size (number of internal nodes) of the profile $\mathcal{T}_{t-1}(l-1)$, that is,

$$\mathbf{P}(\text{level(h)} = \text{l} \mid \mathcal{T}_{\text{t}-1} = \text{T}_{\text{t}-1}) = \frac{\text{T}_{\text{t}-1}(\text{l}-1)}{|\text{T}_{\text{t}-1}|}. \tag{17}$$

Working on the RERT directly can be a daunting task, therefore we propose to approximate the numerator and the denominator of Eq. (17) by $\mathbf{E}[\mathcal{T}_{t-1}(l-1)]$ and $\mathbf{E}|[\mathcal{T}_{t-1}|]$, respectively. It can be shown by a simple inductive argument, or by generating functions as in Mahmoud (2021), that for RERTs we have $\mathbf{E}[\mathcal{T}_t(l)] = \binom{t}{l}(dp_i)^l$ and $\mathbf{E}[|\mathcal{T}_t|] = (1+dp_i)^t$, which suggests a binomial distribution for the level of $h$. And indeed, we can approximate the distribution of the level of a node $h$ added at time $t$ as

$$\begin{aligned}\mathbf{P}(\text{level(h)} = \text{l}) &\approx \frac{\mathbf{E}[\mathcal{T}_{t-1}(l-1)]}{\mathbf{E}[|\mathcal{T}_{t-1}|]}\\&= \frac{\binom{t-1}{l-1}(dp_i)^{l-1}}{(1+dp_i)^{t-1}}\\&= \binom{t-1}{l-1}\left(\frac{dp_i}{1+dp_i}\right)^{l-1}\left(1-\frac{dp_i}{1+dp_i}\right)^{t-l}\\&= \mathbf{P}(\text{Bin}(\text{t}-1,\text{q}) = \text{l}-1),\end{aligned}$$

with $q = dp_i/(1+dp_i)$.

One of the main challenges of this calculation is that we do not know the day of the first hospitalization $t$ conditioned on event $\mathcal{A}$, we only know that each node

is hospitalized with probability $(1-p_a)p_h$, which means that the index of the first hospitalized node follows a geometric distribution with mean $1/((1-p_a)p_h)$. We approximate $t-1$ by the first time that the expected size of the infection tree (excluding the source since we condition on event $\mathcal{A}$) exceeds the expected index of the first hospitalized node. Therefore we solve

$$\mathbf{E}[|\mathcal{T}_{t-1}|-1] = (1+dp_i)^{t-1} - 1 = \frac{1}{(1-p_a)p_h} = \mathbf{E}[\text{index of the first hospitalized node}]$$

for $t$ (relaxing the constraint that $t$ is an integer), which gives

$$t-1 = \frac{\log\left(1+\frac{1}{(1-p_a)p_h}\right)}{\log(1+dp_i)}.$$

Consequently, we approximate $\mathbf{P}(\text{transmission path has length l} \mid \mathcal{A})$ by $\mathbf{P}(\text{Bin(t}-1,\text{q}) = \text{l}-1)$. Continuing Eq. (16), and using the well-known expression of the probability generating function of the binomial distribution, we get

$$\begin{aligned}\mathbf{P}(\text{LS finds source}) &\approx (1-p_a)p_h + (1-p_a)(1-p_h)\left(\sum_{l=1}^{t}\mathbf{P}(\text{Bin(t}-1,\text{q}) = \text{l}-1)(1-p_a)^{l-1}\right)\\ &= (1-p_a)\left(p_h + (1-p_h)\left((1-p_a)\frac{dp_i}{1+dp_i} + 1 - \frac{dp_i}{1+dp_i}\right)^{\frac{\log\left(1+\frac{1}{(1-p_a)p_h}\right)}{\log(1+dp_i)}}\right).\end{aligned} \tag{18}$$

One can check that this expression agrees with our qualitative intuition. However, it is not at all clear whether it is valid because of the strong approximations made in some steps of the above computation. In "Theoretical results" section, we prove a rigorous upper bound on the source identification probability, and we also provide much more careful approximations by proving exact theorems about the simplified models that we use. Then, in "Simulation results" section we compare our results with simulation results on synthetic data, as well as with data generated by the DCS model.

## Appendix B: Further remarks

### B.1: Remarks on changing mobility parameters

In this paper, we always worked with fixed parameters. We believe this is an acceptable assumption in our scenario, since we focus on identifying the source within a few days after the outbreak is detected, and we do not expect the disease parameters to change so quickly. In a real scenario, a quickly reacting government could impose public health interventions, which could change the way the epidemic spreads after the first hospitalization, but this will not change the infection path from the source to the first hospitalized patient, which is the main factor that decides the success of our proposed algorithms. Moreover, if the epidemic spreads more slowly after the public health interventions, then we record

less asymptomatic cases, which makes contact tracing easier (the LS and LS+ algorithms require fewer tests).

### B.2: Remarks on the scalability of the LS and LS+ algorithms

Since both algorithms work locally by tracing back the source from the first hospitalized patient, as long as the degree distribution stays the same, neither the source identification probability, nor the sample complexity, nor the runtime of the algorithms depend on the size of population ($n$). In other words, the complexity of the LS and the LS+ algorithms is constant in $n$. We do not have precise results about the dependence of the sample complexity or the runtime on the degree distribution, however, we note that despite the relatively small population size ($n = 9054$), the degree distribution (contact dynamics) of the DCS+TU model presented in Fig. 8 is chosen to match the degree distribution (contact dynamics) of a real-world scenario (with no public health interventions). Therefore, even though real mobility networks consist of much more than 9054 individuals, we expect the performance of the algorithms to be similar to Fig. 8 in a real-world scenario as well.

## Appendix C: Additional proofs

See Fig. 10.

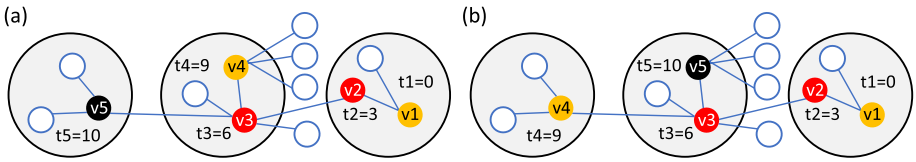


**Fig. 10** Illustration for Lemma 3.3 using the same coloring as Fig. 2a. **a** An example for an epidemic where among the nodes of the transmission path ($v_1, v_2, v_3, v_5$), the middle household contains no symptomatic node (only the asymptomatic node $v_3$), but the LS+ algorithm still finds the source. Indeed, at iteration 0 we set $s_{c,0} = v_5$, after which we find that $v_3$ is asymptomatic, and next that $v_2$ is asymptomatic and $v_4$ is symptomatic, with a lower symptom onset time then $v_5$. Hence, in iteration 1 we set $s_{c,1} = v_4$, and we find that $v_3, v_2$ are asymptomatic and $v_1$ is is symptomatic, with a lower symptom onset time then $v_4$. Finally, in iteration 2 we set $s_{c,2} = v_1$, and we find $s'_c = v_1 = s_{c,2}$, which implies that the algorithm stops, and returns the correct source $v_1$. **b** An example for an epidemic where the LS+ algorithm would fail if we would update the candidate before the test queue becomes empty. Similarly to subfigure **a**, in iteration 0 of the algorithm first learns about asymptomatic node $v_3$ and next about asymptomatic node $v_2$ and symptomatic node $v_4$. If the algorithm updates the candidate to $v_4$ and continues further, instead of scheduling the tests of the household members of $v_2$, then it is not hard to check that $v_4$ will be the final estimate and the algorithm fails. However, if the algorithm waits until the test queue becomes empty and tests the household members of $v_2$, then $v_1$ becomes the next candidate and the algorithm finds the source

### C.1: Proof of Lemma 3.3

We start by restating the lemma here for convenience.

**Lemma C.1** *In the RB tree network, the LS algorithm finds the source if and only if all nodes on the transmission path are symptomatic, and the LS+ algorithm finds the source if among the nodes of the transmission path, there exists a symptomatic node in each household, and the source is symptomatic.*

***Proof*** Throughout the proof we assume that there is no limitation on the number available tests. We can make this assumption because in the SICTF there is only a daily limit on the number tests, there is no limitation on the number of days, and neither the LS nor the LS+ algorithms proceed in an iteration until the test queue becomes empty, which implies that all nodes that enter the test queue get eventually tested.

Suppose that the LS algorithm finds the source. Then the list of candidate nodes $s_c$ at different iterations forms a path that consists entirely of symptomatic nodes between the source and the first hospitalized node. In tree networks, the transmission path is the only path between the source and the first hospitalized node, which yields the "only if" part of the statement on the LS algorithm.

Next, suppose that all nodes on the transmission path are symptomatic. Then, we claim that the candidate node $s_{c,i}$ computed in the $i$th iteration of the LS algorithm is $v_{l-i}$, the $i$th node of the reverse transmission path. Our claim is definitely true for $i = 0$, because $s_{c,0}$ is initialized to be the first hospitalized node $v_l$. Then, the proof proceeds by induction. By the induction hypothesis, in the $i$th step, $s_{c,i} = v_{l-i}$, and since we are on a tree, the symptom onset time of $v_{l-(i+1)}$ (which is revealed because all nodes on the transmission path are symptomatic by assumption) is the only symptom onset time among the neighbors of $s_{c,i}$ that have a lower symptom onset time than $s_{c,i}$ itself. Therefore $s'_c = v_{l-(i+1)}$, and $s_{c,i+1}$ is updated to be $v_{l-(i+1)}$ in the beginning of the next iteration, which proves that the induction hypothesis holds until the source is reached.

Finally, suppose that among the nodes of the transmission path, there exists a symptomatic node in each household, and the source is symptomatic. Let us denote by $w_i$ the $i$th symptomatic node of the *reverse* transmission path. Then, we claim that the candidate list $s_{c,i}$ computed in the $i$th iteration of the LS+ algorithm equals $w_i$. Similarly to the case of the LS algorithm, the $i = 0$ case holds by definition, and we proceed by induction. Suppose that $s_{c,i} = w_i$. It will also be useful to define the index of $w_i$ on the *forward* transmission path (without skipping asymptomatic nodes). Let $j$ be this index, for which therefore $w_i = v_j$. Now we distinguish 3 cases: (i) $v_{j-1} = w_{i+1}$ is symptomatic, (ii) $v_{j-1}$ is asymptomatic and $v_{j-2} = w_{i+1}$ is symptomatic, and (iii) $v_{j-1}$ and $v_{j-2}$ are asymptomatic and $v_{j-3} = w_{i+1}$ is symptomatic. We claim that there are no more cases, and that in all three cases $w_{i+1}$ is tested in the $i$th iteration of the LS+ algorithm. Case (i) is immediate because all neighbors of $s_{c,i}$ are tested. Case (ii) is only possible if either $v_{j-1} \in H(s_{c,i})$ or $v_{j-2} \in H(v_{j-1})$, otherwise $v_{j-1}$ would be a lone asymptomatic node in a household,

which contradicts the assumption that there is a symptomatic node in each household. Since all the contacts of asymptomatic nodes in $H(s_{c,i})$ (see Fig. 3d) and all nodes in the household of asymptomatic nodes are tested in the LS+ algorithm (see Fig. 3e), $v_{j-2}$ must be tested too. Finally, case (iii) is possible only if $v_{j-1} \in H(s_{c,i})$ and $v_{j-3} \in H(v_{j-2})$ both hold, otherwise $v_{j-1}$ or $v_{j-2}$ would be a lone asymptomatic node in a household. Similarly to the previous case, $v_{j-3}$ must be tested (see Fig. 3f). There are no more cases because, by Remark 3.2, on the RB tree a transmission path can only have two nodes in each household, and we assumed that there exists a symptomatic node in each household among the nodes of the transmission path.

After we proved that $w_{i+1}$ is tested in the $i$th iteration of the LS+ algorithm, we must still show that it will be the next candidate $s_{c,i+1}$ for the induction hypothesis to hold. This is true because once the symptom onset time of $w_{i+1}$ is revealed, none of its neighbors are scheduled for testing, and therefore all tested nodes have $w_{i+1}$ on their path to the source, which means that $w_{i+1}$ must have the lowest revealed symptom onset time, and therefore that it will be the next candidate $s_{c,i+1}$. □

### C.2: Proof of Lemma 3.5

We start by restating the lemma here for convenience.

**Lemma C.2** *For the* $\mathrm{DDE_{NR}}$ *epidemic model with parameters* $(p_i, p_a, p_h)$ *on the RB tree with parameters* $(d_c, d_h)$*, and with* $p$ *computed in Eq.* (2)*, we have*

$$\mathbf{P}(LS \text{ finds the source}) = \sum_{n=0}^{\infty} (1-p)^n \mathbf{P}(d(s,h) = n). \tag{19}$$

***Proof*** Let us reveal the randomness that generates the epidemic in a slightly modified way than in the definition ("The DDE model and The $\mathrm{DDE_{NR}}$ model" sections). As before, at the beginning only the source is infectious, and depending on course of the disease, the source can be symptomatic and hospitalized, symptomatic but not hospitalized, or asymptomatic with probabilities $(1-p_a)p_h$, $(1-p_a)(1-p_h)$, $p_a$, respectively. In each moment, each infectious node infects each of its susceptible neighbors with probability $p_i$. If a node is infected, we reveal the information whether it will become hospitalized (which happens with the probability $(1-p_a)p_h$), but if it does not become hospitalized, we do not reveal whether the node is asymptomatic or symptomatic yet. Indeed, this information is not necessary for continuing the simulation of the epidemic since we assumed that there is no difference between the infection probabilities of symptomatic and asymptomatic nodes. Thereafter, when the first hospitalized case occurs, we reveal for each infected node $v$ on the transmission path (except the last node, which we know is hospitalised; see Definition 3.2) whether it is asymptomatic or not. The only information we have about these nodes is that they are not hospitalized, which implies that the probability that a node is revealed to be asymptomatic on the transmission path is exactly the probability $p$ from Definition 3.4 computed in (2).

By Lemma 3.3, LS finds the source if and only if each node on the transmission path is symptomatic. Conditioning on the length of the transmission path, we can compute the probability of each node being symptomatic by Eq. (2) as

$$\mathbf{P}(LS \text{ suceeds}|d(s,h)=n) = \big(1-\mathbf{P}(\nu \text{ is asy}|\nu \text{ is not hosp})\big)^n = (1-p)^n, \tag{20}$$

from which (19) follows immediately. □

### C.3: Proof of Theorem 3.6

We are going to need prove a few intermediate results before proving Theorem 3.6. A first step is to count all the possible paths from the source with a given length.

**Definition C.3** Let $G(s)$ be the RB tree with parameters $(d_c, d_h)$, and let $s$ be the source. A Red-Blue (RB) path of length $n$ is any path of nodes in $(s=v_0, v_1, ...v_n)$ such that $(v_i, v_{i+1}) \in E'$ for $0 \le i < n$. Let $\mathcal{C}_n$ be the set of RB paths of length $n$.

**Lemma C.4** *In the RB tree with parameters* $(d_c, d_h)$, $|C_0| = 1$, *while for* $n \ge 1$,

$$|\mathcal{C}_n| = \lambda_1 \left(\frac{d_c - 1 + D}{2}\right)^n + \lambda_2 \left(\frac{d_c - 1 - D}{2}\right)^n \tag{21}$$

*where*

$$D = \sqrt{(d_c-1)^2 + 4d_c d_h} \tag{22}$$

$$\lambda_1 = \frac{(d_c + 1 + D)(2d_h + d_c - 1 + D)}{2D(d_c - 1 + D)} \tag{23}$$

$$\lambda_2 = \frac{(D - d_c - 1)(2d_h + d_c - 1 - D)}{2D(d_c - 1 - D)}. \tag{24}$$

***Proof*** Let us keep track of the number of RB paths of length $n$ depending on the color of the last node in the path. Let $r_n$ and $b_n$ be the numbers of RB paths of length $n$ such that the last node is red and blue, respectively. A RB path of length 0 consists only of the source, which implies that $r_0 = 1$ and $b_0 = 0$. The source has $d_c$ red and $d_h$ blue neighbours, which implies that $r_1 = d_c$ and $b_1 = d_h$.

Suppose that $P$ is an RB path of length $n \ge 2$. If the last node of $P$ is red, then the node before the last node can be both blue or red. Red nodes other than the source have $d_c - 1$ red children, while blue nodes have $d_c$ red children, yielding

$$r_n = (d_c - 1)r_{n-1} + d_c b_{n-1}, \text{ for } n \ge 2. \tag{25}$$

If the last node of $P$ is blue, then the node before has to be red. Since every red node, including the source, has $d_h$ blue children, we have

$$b_n = d_h r_{n-1}, \text{ for } n \geq 1. \tag{26}$$

By substituting Eq. (26) into Eq. (25), we obtain the recurrence

$$r_n = (d_c - 1)r_{n-1} + d_c d_h r_{n-2}, \text{ for } n \geq 2. \tag{27}$$

We solve this recurrence equation by calculating the characteristic equation

$$t^2 - (d_c - 1)t - d_c d_h = 0, \tag{28}$$

whose roots are

$$t_1 = \frac{d_c - 1 + \sqrt{(d_c-1)^2 + 4d_c d_h}}{2} = \frac{d_c - 1 + D}{2} \tag{29}$$

$$t_2 = \frac{d_c - 1 - \sqrt{(d_c-1)^2 + 4d_c d_h}}{2} = \frac{d_c - 1 - D}{2} \tag{30}$$

yielding the the general solution

$$r_n = c_1 t_1^n + c_2 t_2^n, \tag{31}$$

where $c_1, c_2$ are given by the initial conditions for $n = 0, 1$

$$c_1 + c_2 = r_0 = 1 \tag{32}$$

$$c_1 t_1 + c_2 t_2 = r_1 = d_c, \tag{33}$$

which are

$$c_1 = \frac{1}{2} + \frac{d_c + 1}{2\sqrt{(d_c-1)^2 + 4d_c d_h}} = \frac{1}{2} + \frac{d_c + 1}{2D} \tag{34}$$

$$c_2 = \frac{1}{2} - \frac{d_c + 1}{2\sqrt{(d_c-1)^2 + 4d_c d_h}} = \frac{1}{2} - \frac{d_c + 1}{2D}. \tag{35}$$

From Eqs. (25) and (26) we conclude that for $n \geq 1$,

$$b_n = d_h(c_1 t_1^{n-1} + c_2 t_2^{n-1}) \tag{36}$$

and therefore

$$|C_n| = r_n + b_n = \lambda_1 t_1^n + \lambda_2 t_2^n, \tag{37}$$

where

$$\lambda_1 = c_1\left(1 + \frac{d_h}{t_1}\right) \tag{38}$$

$$\lambda_2 = c_2\left(1 + \frac{d_h}{t_2}\right). \tag{39}$$

Inserting the values for $t_1, t_2, c_1, c_2$ we obtain the desired result. □

Since LS+ improves on LS by making use of the household structure of the network, we need further information about the household structure of the transmission paths. Recall that by Remark 3.2, households on transmission paths on an RB tree were characterized either by a single red node (that is followed by a red node), or a pair of consecutive red and blue nodes. The following definition and lemma refine our previous result on counting the number of RB paths by taking the household structure into account.

**Definition C.5** Let $P = \{s = v_0, v_1, \ldots, v_n = h\}$ be a RB path of length $n$. We say that a node $v$ on the path $P$ is in a $P$-single-household if no other node from $P$ is in the same household as $v$. Otherwise, we say $v$ is in a $P$-multi-household. Given a path $P$, let $M_s : \mathcal{C}_n \to \{0, 1\}$ be the indicator function that the source is in a $P$-multi-household. Similarly, let $M_l : \mathcal{C}_n \to \{0, 1\}$ be the indicator function that the last node of path $P$ is in a $P$-multi-household. Finally, for $0 \le k \le n + 1$ and $\alpha, \beta \in \{0, 1\}$, let

$$C_{n,k,\alpha,\beta} = \{P \in \mathcal{C}_n : (\text{there are exactly } k \text{ nodes in } P - \text{single-households}) \wedge (M_s(P) = \alpha) \wedge (M_l(P) = \beta)\}. \tag{40}$$

The set $C_{n,k,\alpha,\beta}$ depends on 4 parameters, but only some combinations of these parameters make it non-empty. The following definition will be useful in this regard.

*Condition 1* Let $\alpha, \beta \in \{0, 1\}$ and $n \ge 2$. We say $k \in \mathbb{N}$ satisfies Condition 1 if and only if $k$ and $n$ have different parity and $n + 1 - 2(\alpha + \beta) \ge k \ge 2 - (\alpha + \beta)$.

**Lemma C.6** *It holds that* $|C_{0,1,0,0}| = 1, |C_{1,0,1,1}| = d_h$ *and* $|C_{1,2,0,0}| = d_c$. *Let* $\alpha, \beta \in \{0, 1\}$, *let* $n \ge 2$ *and let* $k \in \mathbb{N}$ *satisfy Condition* 1. *Then*

$$|\mathcal{C}_{n,k,\alpha,\beta}| = \binom{\frac{n+k-3}{2}}{k-2+\alpha+\beta} d_h^{\frac{n-k+1}{2}} d_c^{\frac{n-k+3}{2}-\beta-\alpha} (d_c - 1)^{k+\alpha+\beta-2}. \tag{41}$$

*In all other cases* $|C_{n,k,\alpha,\beta}| = 0$.

***Proof*** Since there are $n + 1$ nodes on path $P$, with $k$ in $P$-single households and thus $n + 1 - k$ of them in $P$-multi-households, we must have

$$k + \frac{n+1-k}{2} = \frac{n+k+1}{2}$$

households along path $P$ in total. Clearly, the numbers $n$ and $k$ cannot be of the same parity for any RB path $P$, which is thus assumed for the rest of the proof (this assumption is also part of Condition 1).

If $n = 0$, then the source is also the first hospitalized node, and it is in a $P$-single-household, which implies that $|C_{0,1,0,0}| = 1$. If $n = 1$, then there are two cases: either the source is in the same $P$-multi-household with the first hospitalized node, or both of them are in $P$-single-households. The former case is possible via $d_h$ edges from the source, which gives $|C_{1,0,1,1}| = d_h$, while the latter case is possible via $d_c$ edges, and gives $|C_{1,0,1,1}| = d_c$. Since these are the only possible RB paths of length $n \leq 1$, we must have $|C_{0,k,\alpha,\beta}| = |C_{1,k,\alpha,\beta}| = 0$ for any other choice of parameters $k, \alpha$ and $\beta$.

Let us assume that $n \geq 2$. Then, the source and the first hospitalized node are not in the same household. Let us denote the household of the source by $H_s$ and the household of the first hospitalized node by $H_h$. Note that $(1-\alpha)$ and $(1-\beta)$ are the indicators of $H_s$ and $H_h$ being $P$-single-households, and therefore $k \geq (1-\alpha)+(1-\beta)$. If this inequality (which is also part of Condition 1) does not hold, then clearly $|\mathcal{C}_{n,k,\alpha,\beta}| = 0$. Similarly, the number of $P$-multi-households is $\frac{n-k+1}{2}$ and we must have $\frac{n-k+1}{2} \geq \alpha + \beta$ for $|\mathcal{C}_{n,k,\alpha,\beta}| > 0$, which implies the inequality $n + 1 - 2\alpha - 2\beta \geq k$. Therefore $\mathcal{C}_{n,k,\alpha,\beta}$ is empty if Condition 1 does not hold. For the rest of the proof we assume that Condition 1 does hold.

There are $\frac{n+k-3}{2}$ households along path $P$, excluding $H_s$ and $H_h$. Among them, there are $k-(1-\alpha)-(1-\beta)$ $P$-single-households, which can be chosen in $\binom{\frac{n+k-3}{2}}{k-2+\alpha+\beta}$ ways. Once we know the color of each node along the path, the number of RB paths can be computed by multiplying the numbers of children with the appropriate color of each node. $P$-single-households have no blue nodes, and $P$-multi-households have exactly one, which implies that there are $\frac{n-k+1}{2}$ blue nodes. Since blue nodes are preceded by red nodes that have $d_h$ blue children, they give the multiplicative factor $d_h^{\frac{n-k+1}{2}}$. Blue nodes, except from the first hospitalized node (if it is blue), have $d_c$ red children. So far we have accounted for all of the nodes in $P$-multi-households and none of the nodes in $P$-single-households. If the source is in a $P$-single-household, then we must count its red children, whose number is $d_c$. This implies that there exist $\frac{n-k+1}{2} - \beta + (1-\alpha)$ nodes with $d_c$ red children. Finally, each $P$-single-household, except $H_s$ and/or $H_h$ in case they are $P$-single households, has $d_c - 1$ red children. There are $k-(1-\alpha)-(1-\beta)$ such $P$-single-households, which gives the final term in Eq. (41). □

The sets $\mathcal{C}_{n,k,\alpha,\beta}$ define equivalence classes on the transmission paths based on their household structure. In the next lemma we show that once we know which equivalence class we are in, it is possible compute the source identification probability of the LS+ algorithm.

**Lemma C.7** *Let $P$ be the transmission path in the* $\mathrm{DDE}_{\mathrm{NR}}$ *epidemic model with parameters $(p_i, p_a, p_h)$ on the RB tree with parameters $(d_c, d_h)$, and let $p$ be as computed in* (2). *Then, it holds that*

$$\mathbf{P}(LS+ \text{ finds the source}) | P \in \mathcal{C}_{0,1,0,0}) = 1$$

*and*

$$\mathbf{P}(LS+ \text{ finds the source})|P \in \mathcal{C}_{1,0,1,1}) = \mathbf{P}(LS+ \text{ finds the source})|\mathcal{C}_{1,2,0,0}) = 1-p.$$

*Let* $\alpha, \beta \in \{0,1\}$, *let* $n \geq 2$ *and let* $k \in \mathbb{N}$ *satisfy Condition 1. Then, it holds that*

$$\mathbf{P}(LS+ \text{ finds the source})|P \in \mathcal{C}_{n,k,\alpha,\beta}) \geq (1-p)^{\frac{n+k-1}{2}}(1+p)^{\frac{n-k+1}{2}-\alpha-\beta}. \tag{42}$$

*In all other cases* $\mathbf{P}(LS+ \text{ finds the source})|P \in \mathcal{C}_{n,k,\alpha,\beta})$ *is not defined.*

***Proof*** If $n = 0$, then $k = 1$ and $\alpha = \beta = 0$. In that case, the source is the first hospitalized node and LS+ always finds the source. If $n = 1$, then the first hospitalized node is in the neighbourhood of the source, and LS+ finds the source if and only if the source is symptomatic, which happens with probability $1 - p$.

Let us assume that $n \geq 2$ and that $k$ satisfies Condition 1 (otherwise $|\mathcal{C}_{n,k,\alpha,\beta}| = 0$ and

$$\mathbf{P}(LS+ \text{ finds the source})|P \in \mathcal{C}_{n,k,\alpha,\beta})$$

is not defined). By Lemma 3.3 the LS+ algorithm finds the source in the $\mathrm{DDE}_{\mathrm{NR}}$ model on the RB tree if, among the nodes of the transmission path, there exists a symptomatic node in each household, and the source is symptomatic, which means that we can prove a lower bound on the source identification probability of LS+. Let us assume that the source is indeed symptomatic. Since the first hospitalized node is symptomatic by definition, the households of the source and of the first hospitalized node cannot make the LS+ algorithm fail. Let us denote these two households by $H_s$ and $H_h$, respectively. Also, let $M$ and $S$ be the sets of all $P$-multi- and $P$-single-households, respectively, excluding $H_s$ and $H_h$. Then, LS+ finds the source if all nodes in the households of $S$ are symptomatic, and if at least one node in the households of $M$ is symptomatic, which has probability $1 - p$ and $1 - p^2$ for each type of household, respectively, by Eq. (2). These observations yield that

$$\begin{aligned}\mathbf{P}(LS+ \text{ finds the source})|P \in \mathcal{C}_{n,k,\alpha,\beta}) &\geq \mathbf{P}(\text{source is sym})(1-p)^{|S|}(1-p^2)^{|M|} \\ &= (1-p)(1-p)^{k-2+\alpha+\beta}(1-p^2)^{\frac{n-k+1}{2}-\alpha-\beta} \\ &= (1-p)^{k-1+\alpha+\beta}(1-p^2)^{\frac{n-k+1}{2}-\alpha-\beta}.\end{aligned} \tag{43}$$

□

Finally, we are ready to state and prove Theorem 3.6 on the source identification probability of LS+, which we restate here for convenience.

**Theorem C.8** *Let* $p$ *be as in* (2) *and let* $\mathcal{S}(n,\alpha,\beta)$ *be the set of* $k$ *values that satisfy Condition 1. Then, for the* $\mathrm{DDE}_{\mathrm{NR}}$ *epidemic model with parameters* $(p_i, p_a, p_h)$ *on the RB tree with parameters* $(d_c, d_h)$ *we have*

$$\mathbf{P}(LS+ \text{ finds the source}) \geq \mathbf{P}(d(s,h)=0) + (1-p)\mathbf{P}(d(s,h)=1)$$
$$+ \sum_{n=2}^{\infty} \sum_{\substack{\alpha,\beta \in \{0,1\} \\ k \in \mathcal{S}(n,\alpha,\beta)}} \binom{\frac{n+k-3}{2}}{k-2+\alpha+\beta} \frac{(d_h(1-p))^{\frac{n+k-1}{2}} (d_c(1+p))^{\frac{n-k+1}{2}-\alpha-\beta} d_c(d_c-1)^{k+\alpha+\beta-2}}{\lambda_1\left(\frac{d_c-1+D}{2}\right)^n + \lambda_2\left(\frac{d_c-1-D}{2}\right)^n} \mathbf{P}(d(s,h)=n), \tag{44}$$

*where $D, \lambda_1$ and $\lambda_2$ are terms depending on parameters $d_c$ and $d_h$ and are computed explicitly in Lemma C.4.*

***Proof*** Let us extend the domain of $\mathbf{P}(LS+ \text{ finds the source}) | P \in C_{n,k,\alpha,\beta})$ by function $g$ defined as $g : \mathbb{N} \times \mathbb{N} \times \{0,1\} \times \{0,1\} \to [0,1]$ such that

$$g(n,k,\alpha,\beta) = \begin{cases} \mathbf{P}(LS+ \text{ finds the source}) | P \in C_{n,k,\alpha,\beta}) & \text{if } k \in \mathcal{S}(n,\alpha,\beta) \\ 0 & \text{if } k \notin \mathcal{S}(n,\alpha,\beta). \end{cases} \tag{45}$$

Unlike $\mathbf{P}(LS+ \text{ finds the source}) | P \in C_{n,k,\alpha,\beta})$, $g$ is defined for every 4-tuple of parameters $(n,k,\alpha,\beta) \in \mathbb{N} \times \mathbb{N} \times \{0,1\} \times \{0,1\}$. By the law of total probability we expand the source identification probability by conditioning on the path $P$ being of length $n$ as

$$\begin{aligned} \mathbf{P}(LS+ \text{ finds the source}) &= \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} \sum_{\alpha,\beta \in \{0,1\}} g(n,k,\alpha,\beta) \mathbf{P}(P \in C_{n,k,\alpha,\beta}) \\ &= \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} \sum_{\alpha,\beta \in \{0,1\}} g(n,k,\alpha,\beta) \mathbf{P}(P \in C_{n,k,\alpha,\beta} | P \in C_n) \mathbf{P}(d(s,h)=n). \end{aligned} \tag{46}$$

Next, we exchange the sums over $\alpha, \beta$ and $k$. This allows us to sum over only those $k$ values that satisfy Condition 1, which implies that $\mathbf{P}(LS+ \text{ finds the source}) | P \in C_{n,k,\alpha,\beta})$ is well-defined. As in Lemma C.6, we need to treat the $n=0$ and $n=1$ cases separately. Continuing Eq. (46), we arrive to

$$\mathbf{P}(LS+ \text{ finds the source}) = \mathbf{P}(d(s,h)=0) + (1-p)\mathbf{P}(d(s,h)=1)$$
$$+ \sum_{n=2}^{\infty} \sum_{\alpha,\beta \in \{0,1\}} \sum_{k \in \mathcal{S}(n,\alpha,\beta)} \mathbf{P}(LS+ \text{ finds the source}) | P \in C_{n,k,\alpha,\beta}) \frac{|C_{n,k,\alpha,\beta}|}{|C_n|} \mathbf{P}(d(s,h)=n) \tag{47}$$

Substituting in the results from Lemmas C.4, C.6 and C.7 into Eq. (47) gives the desired result. □

### C.4: Proof of Theorem 3.9

We start by restating Theorem 3.9 for convenience.

**Theorem C.9** *In the $(d_r, d)$-RET with parameters $p_i, p_a, p_h$, let $a_{t,l}$ be as in Definition 3.8. Then*

$$a_{t,0} = 1 \tag{48}$$

$$a_{t,l} = d_r p_i \sum_{m=l-1}^{t-1} \binom{m}{l-1} (1-p_i)^{m-l+1} d^{l-1} p_i^{l-1}, \text{ for } t \geq l \geq 1 \tag{49}$$

$$a_{t,l} = 0, \text{ for } l > t. \tag{50}$$

***Proof*** Similarly to Feng and Mahmoud (2018); Mahmoud (2021), the proof relies on generating functions. We start by addressing the boundary cases. For all $t \geq 0$, it holds that $A_{t,0} = 1$, and therefore $a_{t,0} = 1$. Similarly, for all $l$, $t$ such that $l > t$, it holds that $A_{t,l} = 0$, and therefore $a_{t,l} = 0$. Suppose that $t \geq l = 1$. During day $t-1$, on the first level, there are $A_{t-1,1}$ infected (internal) nodes and $d_r - A_{t-1,1}$ (external) nodes that may be infected with probability $p_i$ during day $t$. Thus,

$$A_{t,1} = A_{t-1,1} + \mathrm{Bin}(\mathrm{d_r} - \mathrm{A_{t-1,1}}; \mathrm{p_i}). \tag{51}$$

Taking the expectation of both sides in Eq. (51) yields

$$a_{t,1} = a_{t-1,1}(1-p_i) + d_r p_i, \text{ for } t \geq 1. \tag{52}$$

By subtracting the appropriate recurrence equations for $a_{t,1}$ and $a_{t-1,1}$ for $t \geq 2$ we obtain the homogeneous recurrence equation

$$a_{t,1} - a_{t-1,1}(2-p_i) + (1-p_i)a_{t-2,1} = 0, \text{ for } t \geq 2 \tag{53}$$

and boundary conditions $a_{0,1} = 0$ and $a_{1,1} = d_r p_i$. We solve for $a_{t,1}$ using the same methods as in the proof of Lemma C.4 and obtain

$$a_{t,1} = d_r\left(1 - (1-p_i)^t\right), \text{ for } t \geq 0. \tag{54}$$

Next, let us consider the general case $t \geq l > 1$. On day $t-1$, there are $A_{t-1,l-1}$ nodes on level $l-1$. Since, each node on level $l-1$ has $d$ children, there are $dA_{t-1,l-1}$ nodes on level $l$ that have an infectious parent on level $l-1$. However, $A_{t-1,l}$ of them are already infected. Therefore $dA_{t-1,l-1} - A_{t-1,l}$ nodes of level $l$ may be infected on day $t$, each with probability $p_i$, which implies

$$A_{t,l} = A_{t-1,l} + \mathrm{Bin}(\mathrm{dA_{t-1,l-1}} - \mathrm{A_{t-1;l}}, \mathrm{p_i}), \text{ for } \mathrm{t} \geq \mathrm{l} \geq 2. \tag{55}$$

Taking the expectation of both sides in Eq. (55) yields

$$\begin{aligned} a_{t,l} &= a_{t-1,l} + (da_{t-1,l-1} - a_{t-1,l})p_i \\ &= a_{t-1,l}(1-p_i) + dp_i a_{t-1,l-1}, \text{ for } t \geq l \geq 2. \end{aligned} \tag{56}$$

For convenience, let us introduce $\lambda = 1 - p_i$ and $\mu = dp_i$, and also let

$$f(x,y) = \sum_{t=1}^{\infty}\sum_{l=1}^{\infty} a_{t,l} x^t y^l = \sum_{t=1}^{\infty}\sum_{l=1}^{t} a_{t,l} x^t y^l \tag{57}$$

be the generating function for $a_{t,l}$ with $t, l \geq 1$. By multiplying (56) by $x^t y^l$ and summing it over $t, l \geq 2$ we obtain

$$\begin{aligned}\sum_{t=2}^{\infty}\sum_{l=2}^{t} a_{t,l}x^t y^l &= \lambda\sum_{t=2}^{\infty}\sum_{l=2}^{t} a_{t-1,l}x^t y^l + \mu\sum_{t=2}^{\infty}\sum_{l=2}^{t} a_{t-1,l-1}x^t y^l \\ &= \lambda x\sum_{t=1}^{\infty}\sum_{l=2}^{t} a_{t,l}x^t y^l + \mu xy\sum_{t=1}^{\infty}\sum_{l=1}^{t} a_{t,l}x^t y^l.\end{aligned} \tag{58}$$

Since $a_{1,l} = 0$ for $l \geq 2$,

$$\sum_{t=1}^{\infty}\sum_{l=2}^{t} a_{t,l}x^t y^l = \sum_{t=2}^{\infty}\sum_{l=2}^{t} a_{t,l}x^t y^l, \tag{59}$$

and by inserting (59) into (58), we obtain

$$(1-\lambda x)\sum_{t=1}^{\infty}\sum_{l=2}^{t} a_{t,l}x^t y^l = \mu xy\sum_{t=1}^{\infty}\sum_{l=1}^{t} a_{t,l}x^t y^l \overset{(57)}{=} \mu xy f(x,y). \tag{60}$$

Now, we can also decompose the sum (59) using geometric series as

$$\begin{aligned}\sum_{t=1}^{\infty}\sum_{l=2}^{t} a_{t,l}x^t y^l &= \sum_{t=1}^{\infty}\sum_{l=1}^{t} a_{t,l}x^t y^l - \sum_{t=1}^{\infty} a_{t,1}x^t y \\ &\overset{(54)}{=} f(x,y) - d_r y\sum_{t=1}^{\infty}(1-\lambda^t)x^t \\ &= f(x,y) - d_r xy\left(\frac{1}{1-x} - \frac{\lambda}{1-\lambda x}\right).\end{aligned} \tag{61}$$

By plugging (61) into (60), we obtain the expression

$$f(x,y) = d_r(1-\lambda)xy\frac{1}{1-x}\frac{1}{1-\lambda x-\mu xy}. \tag{62}$$

Then, we expand the fractions in (62) into a power series and we next apply the binomial theorem, we arrive to

$$\begin{aligned}f(x,y) &= d_r(1-\lambda)xy\sum_{n=0}^{\infty}x^n\sum_{m=0}^{\infty}x^m(\lambda+\mu y)^m \\ &= d_r(1-\lambda)xy\sum_{n=0}^{\infty}x^n\sum_{m=0}^{\infty}x^m\sum_{j=0}^{m}\binom{m}{j}\lambda^{m-j}(\mu y)^j \\ &= d_r(1-\lambda)\sum_{n=0}^{\infty}\sum_{m=0}^{\infty}\sum_{j=0}^{m}\binom{m}{j}\lambda^{m-j}\mu^j x^{1+n+m}y^{j+1}.\end{aligned} \tag{63}$$

Let $t = 1+n+m$ and $l = j+1$. In order to obtain an expression for $a_{t,l}$, we must change the variables in the sums of Eq. (63) from $(n, m, k)$ to $(t, m, l)$. Changing the inner sum from variable $j$ to $l$ is simple. Changing the variables in the two outer sums is more

challenging because $t$, $n$ and $m$ depend on each other in a nontrivial way. More precisely, since $m, n \geq 0$ we have $t \geq 1$ and also $m \leq t-1$, which means that we have to set the lower limit of $t$ and the upper limit of $m$ accordingly. As for the remaining limits, variable $t$ can be arbitrary large, and $m$ can take any integer value starting from 0 independently of $t$, which yields the expression

$$f(x,y) = d_r(1-\lambda)\sum_{t=1}^{\infty}\sum_{m=0}^{t-1}\sum_{l=1}^{m+1}\binom{m}{l-1}\lambda^{m-l+1}\mu^{l-1}x^t y^l. \tag{64}$$

For the values of $l$ with $l \geq m+1$, the binomial coefficient $\binom{m}{l-1}$ is 0, which implies that we can increase the upper limit of the inner sum from $m+1$ to $t$ in Eq. (64). Then,

$$\begin{aligned} f(x,y) &= d_r(1-\lambda)\sum_{t=1}^{\infty}\sum_{m=0}^{t-1}\sum_{l=1}^{t}\binom{m}{l-1}\lambda^{m-l+1}\mu^{l-1}x^t y^l \\ &= \sum_{t=1}^{\infty}\sum_{l=1}^{t} d_r(1-\lambda)\sum_{m=0}^{t-1}\binom{m}{l-1}\lambda^{m-l+1}\mu^{l-1}x^t y^l. \end{aligned} \tag{65}$$

Finally we can read off the value of $a_{t,l}$ from Eq. (65) as

$$a_{t,l} = d_r(1-\lambda)\sum_{m=0}^{t-1}\binom{m}{l-1}\mu^{l-1}\lambda^{m-l+1} = d_r p_i \sum_{m=0}^{t-1}\binom{m}{l-1}(dp_i)^{l-1}(1-p_i)^{m-l+1}. \tag{66}$$

□

### C.5: Proof of Corollary 3.10

We start by restating Corollary 3.10 for convenience.

**Corollary C.10** *In the RET$(p_i, d_r, d)$, let $a_t$ be the expectation of (8), as in Definition 3.8. For $t \geq 0$,*

$$a_t = 1 + d_r\frac{(1-p_i+dp_i)^t - 1}{d-1}. \tag{67}$$

***Proof*** By using linearity of expectation, Eq. (8) and Theorem 3.9 we obtain:

$$\begin{aligned} a_t &= \sum_{l=0}^{+\infty} a_{t,l} \\ &= 1 + \sum_{l=1}^{+\infty} a_{t,l} \\ &= 1 + d_r p_i \sum_{l=1}^{t}\sum_{m=l-1}^{t-1}\binom{m}{l-1}(1-p_i)^{m-l+1}d^{l-1}p_i^{l-1} \end{aligned} \tag{68}$$

Before we use binomial theorem, we need to swap the sums. Boundaries from (68) are equivalent to $t-1 \geq m \geq l-1 \geq 0$, so we can rewrite this as 2 conditions: $m+1 \geq l \geq 1$ and $t \geq m \geq 0$.

$$\begin{aligned} a_{t,l} &= 1 + d_r p_i \sum_{m=0}^{t-1} \sum_{l=1}^{m+1} \binom{m}{l-1} (1-p_i)^{m-l+1} d^{l-1} p_i^{l-1} \\ &= 1 + d_r p_i \sum_{m=0}^{t-1} \sum_{l=0}^{m} \binom{m}{l} (1-p_i)^{m-l} d^{l} p_i^{l} \end{aligned} \tag{69}$$

Finally, by applying the binomial theorem and summing the geometric series, we obtain the desired equation:

$$\begin{aligned} a_{t,l} &= 1 + d_r p_i \sum_{m=0}^{t-1} (1 - p_i + d p_i)^m \\ &= 1 + d_r \frac{(1-p_i+dp_i)^t - 1}{d-1}. \end{aligned} \tag{70}$$

□

### C.6: Proof or Lemma 3.12

We restate Lemma 3.12 here for convenience.

**Lemma C.11** *Let us consider the stopped DET model with parameters* $(c_{t,l}), p_a, p_h$, *and let* $h$ *denote the first hospitalized node. Then*

$$\mathbf{P}(d(s,h) = l) = \sum_{t=0}^{+\infty} \frac{c_{t,l} - c_{t-1,l}}{c_t - c_{t-1}} (1 - (1-p_a)p_h)^{c_{t-1}} \left(1 - (1 - (1-p_a)p_h)^{c_t - c_{t-1}}\right). \tag{71}$$

***Proof*** Recall that a node added at day $t$ is uniformly distributed among the $c_t - c_{t-1} > 0$ nodes added that day, and that the number of nodes added to level $l$ is $c_{t,l} - c_{t-1,l}$ on day $t$. If we condition on the time of the first hospitalized case, denoted by $TI_h$, then

$$\begin{aligned} \mathbf{P}(d(s,h) = l) &= \sum_{t=0}^{+\infty} \mathbf{P}(d(s,h) = l | TI_h = t) \mathbf{P}(TI_h = t) \\ &= \sum_{t=0}^{+\infty} \frac{c_{t,l} - c_{t-1,l}}{c_t - c_{t-1}} \mathbf{P}(\text{node is not hosp})^{c_{t-1}} (1 - \mathbf{P}(\text{node is not hosp})^{c_t - c_{t-1}}) \\ &= \sum_{t=0}^{+\infty} \frac{c_{t,l} - c_{t-1,l}}{c_t - c_{t-1}} (1 - (1-p_a)p_h)^{c_{t-1}} \left(1 - (1 - (1-p_a)p_h)^{c_t - c_{t-1}}\right). \end{aligned} \tag{72}$$

□

## Appendix D: Dynamic message passing for the DDE model

In this section, we explain how we derived and implemented the DMP equations for the DDE+HNM model. We start by reviewing the previous work on the DMP equations for the SIR model in Appendix D.1, and then we proceed to our derivations in Appendix D.2. In Appendix D.3, we explain how we find candidate (node,time) pairs for the DMP equations, and in Appendix D.4 we conclude by combining Appendices D.2 and D.3 into a source identification algorithm.

### D.1: DMP equations for the SIR model

The DMP equations were first derived by Lokhov et al. (2014) for the SIR model in the context of source identification. Their goal is to compute the marginal probabilities that node $i$ is in a given state at time $t$ (denoted by $P_S^i(t), P_I^i(t)$ and $P_R^i(t)$ for the susceptible, infected and recovered states, respectively), given initial conditions $P_S^i(t_0), P_I^i(t_0)$ and $P_R^i(t_0)$ at some initial time $t_0$. To solve this problem in tree networks, we may consider a dynamic programming approach, where we delete a node $i$, we compute the marginal probabilities of $P_S^j(t-1)$ for all neighbors $j$ of $i$ in the remaining subtrees, and use this information to compute $P_S^i(t)$ (as the marginals are independent in each of the subtrees conditioned on the state of $i$). The DMP equations make the dynamic programming intuition explicit. Originally, the DMP equations were developed for static networks, but since the generalization to time-varying networks is straightforward, and has already been foreshadowed in a similar heuristic algorithm (Jiang et al. 2016), we include it in this preliminary section. For time-varying networks, we define $N_i(t)$ as the set of neighbors of node $i$ in the time-window $[t, t+1)$.

To formalize the dynamic programming approach, Lokhov et al. (2014) introduces some new notation. Let $\lambda$ be the probability that an infectious node infects a susceptible neighbor, and let $\mu$ be the probability that an infectious node recovers. Let $D_i$ be the auxiliary dynamics, where node $i$ receives infection signals, but ignores them, and thus remains in the $S$ state at all times. Let $P_S^{j\to i}(t)$ be the probability that node $j$ is in the state $S$ at time $t$ in the dynamics $D_i$, and let $\theta^{k\to i}(t)$ be the probability that the infection signal has not been passed from node $k$ to node $i$ up to time $t$ in the dynamics $D_i$. Finally, let $\phi^{k\to i}(t)$ be the probability that the infection signal has not been passed from node $k$ to node $i$ up to time $t$, and that node $k$ is in the state $I$ at time $t$, in the dynamics $D_i$. With these definitions, the dynamic programming approach is formalized by the following equations for $t \geq t_0$:

$$P_S^{i\to j}(t+1) = P_S^i(t_0) \prod_{k\in N_i(t)\setminus j} \theta^{k\to i}(t+1), \tag{73}$$

$$\theta^{k\to i}(t+1) - \theta^{k\to i}(t) = -\lambda\phi^{k\to i}(t), \tag{74}$$

$$\phi^{k\to i}(t) = (1-\lambda)(1-\mu)\phi^{k\to i}(t-1) + \left(P_S^{k\to i}(t-1) - P_S^{k\to i}(t)\right). \tag{75}$$

The marginal probabilities that node $i$ is in a given state at time $t$ are then given by

$$P_S^i(t+1) = P_S^i(t_0) \prod_{k\in N_i(t)} \theta^{k\to i}(t+1)\,, \tag{76}$$

$$P_R^i(t+1) = P_R^i(t) + \mu P_I^i(t)\,, \tag{77}$$

$$P_I^i(t+1) = 1 - P_S^i(t+1) - P_R^i(t+1)\,. \tag{78}$$

These equations are only exact on trees, but they can also be applied to networks with cycles as a heuristic approach. The heuristic gives good approximations to the true marginals if the network is at least locally tree-like (Karrer and Newman 2010).

#### D.2: DMP equations for the DDE+HNM model

There are several differences between the SIR model on locally tree-like networks and the DDE+HNM model (see Fig. 2a). First, the DDE model has additional compartments (exposed nodes, asymptomatic nodes), which motivates the introduction of several new variables. Let $\lambda_{(a)}$ (resp., $\lambda_{(s)}$) be the probability that an asymptomatic (resp., symptomatic) node infects a susceptible node. Let $\phi^{k\to i}(t)^{(a)}$ (resp., $\phi^{k\to i}(t)^{(s)}$) be the probability that the infection signal has not been passed from node $k$ to node $i$ up to time $t$, and that node $k$ is asymptomatic (resp., symptomatic) infectious at time $t$, in the dynamics $D_i$.

The second important difference is that in the DDE model, the transition times between different compartments are deterministic instead of following a geometric distribution as in the standard SIR model. While deterministic transition times sound simpler at first, it turns out that they make the DMP equations more complex, because the Markovian property that each marginal probability depends only on the previous timestep is lost if the transition times are larger than 1. Recall that the times for the transitions $E \to I$ and $I \to R$ (with their default values) are $T_E = 3$ and $T_I = 14$.

Let us incorporate these two differences into Eqs. (73)–(75) to derive the DMP equations for the DDE model. Equation (79) is essentially a copy of (73). Equation (80) follows Eq. (74), but we incorporate the two different variants of infected (asymptomatic and symptomatic) patients with their respective infection probabilities $\lambda_{(a)}$ and $\lambda_{(s)}$. Equation (81) is a new equation, which is necessary because recovery times are no longer geometric random variables; instead we need to check the probabilities of infection $T_E + T_I$ timesteps earlier than the current time $t$. Finally, Eq. (82) (resp., (83)) is the asymptomatic (resp., symptomatic) version of Eq. (75), while also incorporating the deterministic time for the transition $E \to I$. For $t \geq t_0$, this yields equations

$$P_S^{i\to j}(t+1) = P_S^i(t_0) \prod_{k\in N_i(t)\setminus j} \theta^{k\to i}(t+1) = \frac{P_S^i(t+1)}{\theta^{j\to i}(t+1)}, \tag{79}$$

$$\theta^{k\to i}(t+1) - \theta^{k\to i}(t) = -\lambda_{(a)}\phi_{(a)}^{k\to i}(t) - \lambda_{(s)}\phi_{(s)}^{k\to i}(t), \tag{80}$$

$$P_R^{k\to i}(t) = P_S^{k\to i}(t - T_E - T_I - 1) - P_S^{k\to i}(t - T_E - T_I) \tag{81}$$

$$\begin{aligned}\phi_{(a)}^{k\to i}(t) = (1-\lambda_{(a)})(1-P_R^{k\to i}(t))\phi_{(a)}^{k\to i}(t-1) \\ + p_a[P_S^{k\to i}(t - T_E - 1) - P_S^{k\to i}(t - T_E)].\end{aligned} \tag{82}$$

$$\begin{aligned}\phi_{(s)}^{k\to i}(t) = (1-\lambda_{(s)})(1-P_R^{k\to i}(t))\phi_{(s)}^{k\to i}(t-1) \\ + (1-p_a)[P_S^{k\to i}(t - T_E - 1) - P_S^{k\to i}(t - T_E)].\end{aligned} \tag{83}$$

We note that for early values of $t$, Eqs. (81)–(83) depend on $P_S^{k\to i}$ before $t_0$, which we initialize to be 1 (all nodes are susceptible before the first node develops the infection). The marginal probability that node $i$ is susceptible at time $t$ is still computed by Eq. (76) as before. Equations (77)–(78) do not apply anymore; we explain it in Appendix D.4 how to take into account observations for nodes in the infectious compartments.

The third difference between the the SIR model on locally tree-like networks and the DDE+HNM model is that the HNM model contains many short cycles inside the households. Short cycles can cause unwanted feedback loops in the DMP equations where, loosely speaking, nodes are treated as if they could reinfect themselves. We solve this issue by modifying the underlying graph to be locally tree-like (only for the computation of the DMP equations). Specifically, we introduce a new central household-node for each household, and we replace the cliques inside the households by a star graph centered at this new household-node node. Introducing such a central household-node does of course alter epidemic process, in particular it makes household infections less independent and slower (all household infections need to pass through an extra node). To mitigate this issue, we assume that central household-nodes have $T_E = 1$ and that they are infected with probability 1 by any node in the same household. We tested the validity of the resulting DMP equations against simulations of the epidemic progressions and we found the results to be quite accurate, in particular, more accurate than the version without the introduction of these central household-nodes.

Note that we derived the DMP equations for the DDE+HNM model, however, since (i) the compartments are the same, (ii) the equations support temporal networks, and (iii) we have separate infection probabilities $\lambda_{(a)}$ and $\lambda_{(s)}$ for asymptomatic and symptomatic nodes, our equations can also be applied to the DCS+TU model after a discretizing (rounding) the time observations.

Finally, we touch upon the computational complexity of computing the DMP equations. In principle, we need to update $O(dN)$ equations (for each edge) over $t_{\max}$ timesteps, where $t_{\max}$ is the maximum time during which the marginals can still change, which can be as large as $O(N)$. However, since we are only interested in computing the likelihood of the 5 earliest observations, $t_{\max}$ is typically quite low. Moreover, since we assume to be in an early stage of the epidemic, most of the equations remain unchanged. For better computational scalability, we only compute $P_S^i(t)$ and $\theta^{k\to i}(t)$ for nodes $k$, $i$ that have $P_I^{k\to i}(t) > 0.01$, i.e., we only update nodes that are at least somewhat likely to have received the infection. Otherwise, we set $P_I^{k\to i}(t) = P_I^{k\to i}(t-1)$, $\theta^{k\to i}(t) = \theta^{k\to i}(t-1)$, and in the implementation we can perform these assignments implicitly using appropriate data structures. With these adjustments, the time-complexity of the algorithm becomes independent of $N$, but remains dependent on the network parameters, the epidemic parameters and the number of sensors in a non-trivial way.

#### D.3: Feasible source-time pairs for source identification

In this section we explain how we implemented the feasible source identification algorithm, which was suggested as a preprocessing step for a method very similar to the DMP equations by Jiang et al. (2016). Let us define the directed graph $G_2$ on (node,infecton_time) pairs (we use "nodes" for the nodes of the original graph $G$ and "pairs" for the nodes of $G_2$), and draw an edge between two pairs $(v_1, t_1) \to (v_2, t_2)$ if $v_1$ and $v_2$ are in contact at $t_2$, and $t_2$ is in the interval $[t_1 + T_E, t_1 + T_E + T_I]$. Observe that in the DDE model there is an edge $(v_1, t_1) \to (v_2, t_2)$ if and only if $v_1$ becoming infected at time $t_1$ can infect $v_2$ at time $t_2$. The definition of $G_2$ is applicable to the DCS model as well after discretization (rounding), however, since the infection times are not deterministic anymore, not all possible infections $(v_1, t_1) \to (v_2, t_2)$ have a corresponding edge in $G_2$.

Then, we perform a breadth-first search backwards on the directed edges of $G_2$, starting from each pair $(v_i, t_i - T_E - T_P)$, where $v_i$ is a symptomatic sensor node, and $t_i$ is the symptom onset time of $v_i$ (for the DCS model, we start from integer times in the $t_i - T_E - T_P \pm (\sigma_E + \sigma_P)$ interval to account for the randomness of the transition times). To limit the time complexity of the algorithm, we only consider the $k_1$ earliest observations, which means that we start $k_1$ breadth-first searches. With this construction, each pair $(v, t)$ discovered by a breadth-first search started from $(v_i, t_i - T_E - T_P)$ could have caused the infection in $v_i$; we say that $(v, t)$ is an explanation for observation $i$. We perform the breath-first searches until we find $k_2$ pairs that explain all of the $k_1$ earliest observations. See the pseudocode in Algorithm D.1.

**Claim 1** *In the DDE model, Algorithm D.1 with $\sigma_E = \sigma_P = 0$ finds the $k_2$ feasible explanations with the latest starting time of the $k_1$ earliest symptomatic nodes.*

***Proof*** By construction, a source node $v$ that becomes infectious at time $t$ can cause an observation $(v_i, t_i)$ if and only if there is a directed path from $(v, t)$ to $(v_i, t_i - T_E - T_P)$. Therefore, the breadth-first search algorithm finds all of the closest feasible sources in time. □

**Algorithm D.1:** Feasible source identification (reverse dissemination [25])

**Input:**

- The mean exposed time $T_E$ the mean pre-infectious time $T_P$, the mean infectious time $T_I$, the std of the exposed time $\sigma_E$ and the std of the pre-infectious time $\sigma_P$
- $F(v)_{min}$ and $F(v)_{max}$ returns the minimum and maximum times when $v$ could have been exposed based on all of its (possibly asymptomatic or negative) test results
- $S(v)$ returns the time of symptom onset for a node $v$ tested positive symtomatic.
- $N(v, [t_{min}, t_{max}])$ returns the set of neighbors of node $v$ in the interval $[t_{min}, t_{max}]$
- A lower estimate of the time the source became infectious $t_{min}$
- Integers $k_1, k_2$

**Output:** A list of at most $k_2$ tuples of node and time pairs that can explain the first $k_1$ symptomatic nodes

$l \leftarrow \{\}$; `// if the list` $l[t]$ `contains the tuple` $(v, w)$, `then the infection started at` $w$ `at time` $t$ `can explain` $v$

$D \leftarrow \{\}$; `// if the list` $D[w, t]$ `contains the node` $v$, `then the infection started at` $w$ `at time` $t$ `can explain` $v$

$doneList \leftarrow []$;

**for** $v \in SortIncreasingByValues(S)[0 : k_1]$ **do**

  $t'_{min} \leftarrow S(v) - (T_E + T_P) - (\sigma_E + \sigma_P)$;

  $t'_{max} \leftarrow S(v) - (T_E + T_P) + (\sigma_E + \sigma_P)$;

  **for** $t' \leftarrow t'_{min}$ *to* $t'_{max}$ **do**

    $Append((v, v), l[t'])$;

    $Append(v, D[v, t'])$;

    **if** $Length(D[v, t']) = k_1$ **then**

      $Append((v, t'), doneList)$

$t \leftarrow SortIncreasingByValues(S)[k_1 - 1]$;

$stopCondition \leftarrow False$ ;

**while** *not* $stopCondition$ *and* $t > t_{min}$ **do**

  **for** $v, w \in l[t]$ **do**

    **for** $u \in N(w, [t, t-1])$ **do**

      $t'_{min} \leftarrow \max(F(u)_{min}, t - T_E - T_I)$;

      $t'_{max} \leftarrow \min(F(u)_{max}, t - T_E)$;

      **for** $t' \leftarrow t'_{max}$ *to* $t'_{min}$ **do**

        $Append((v, u), l[t'])$;

        $Append(v, D[(u, t')])$;

        **if** $Length(D[u, t']) = k_1$ **then**

          $Append((u, t'), doneList)$

  $doneList \leftarrow SortBySecondElement(doneList)$;

  **if** $Length(doneList \geq k2)$ *and* $t - T_E \leq doneList[k2][1]$ **then**

    $stopCondition \leftarrow True$;

  **else**

    $t \leftarrow t - 1$;

**return** $doneList$

### D.4: Source identification via feasible source identification and DMP

In this section we explain how to combine Algorithm D.1 with the DMP equations derived in Appendix D.2. See the pseudocode in Algorithm D.2.

We start by computing the DMP Eqs. (79)–(83) and (73) for the $k_2$ tuples of node and time pairs that can explain the first $k_1$ symptomatic observations returned by Algorithm D.1. Next, our goal is to use these DMP equations to compute the likelihood of each of the $k_2$ tuples using the $k_1$ observations. Similarly to Lokhov et al. (2014), we make the assumption that the first $k_1$ observations are independent, and we can compute the likelihood by multiplying their respective marginals together.

For symptomatic observed nodes $v$, we know the time of symptom onset, which we denote by $S(v)$. Then, the marginal probability of $v$ developing symptoms exactly at time $t$ can be computed by taking the difference of $P_S^v(S(v) - T_P - T_E - 1)$ and $P_S^v(S(v) - T_P - T_E)$ and multiplying the difference by $(1 - p_a)$. In Algorithm D.2 we drop the multiplicative factor $(1 - p_a)$ because it is present for all of the tuples, and it does not change the final order of their scores. For asymptomatic (resp., negative) observations, we only know that at the time of testing, denoted by $A(v)$ (resp., $NE(v)$), at least a time interval of length $T_E$ has passed (resp., $T_E$ has not passed) since the time of infection. Therefore, dropping the $p_a$ factor similarly to the symptomatic case, we compute the marginal of asymptomatic observations as $1 - P_S^v(A(v) - T_E)$, and we compute the marginal of negative observations as $P_S^v(NE(v) - T_E)$. Finally, the contributions of the observations are multiplied together for each of the $k_2$ tuples returned by Algorithm D.1, and the scores approximating the likelihoods are returned.

**Algorithm D.2:** Source Identification via DMP

**Input:**

- The mean exposed time $T_E$, the mean pre-infectious time $T_P$, the mean infectious time $T_I$
- $S(v)$ returns the time of symptom onset for a node $v$ tested positive symtomatic.
- $A(v)$ and $NE(v)$ return the time of asymptomatic and negative test results, respectively

**Output:** A dictionary $L$ of $k_2$ elements, which contains a score for each $(v, t)$ pair that explains the first $k_1$ observations. Higher scores signify higher confidence of being the source.

$L \leftarrow \{\}$;
$doneList \leftarrow$ Algorihtm D.1$(k_1, k_2)$;
**for** $v, t_0 \in doneList$ **do**
  $P_S \leftarrow$ eq. (73) based on DMP eq. (79)-(83) with $P_S^v(t_0) = 0$, and $P_S^w(t_0) = 1$ for all $w \neq v$;
  $L[v, t_0] \leftarrow 1$;
  **for** $w \in S$ **do**
    $L[v, t_0] \leftarrow L[v, t_0] \cdot (P_S^v(S(v) - T_P - T_E - 1) - P_S^v(S(v) - T_P - T_E))$;
  **if** $w \in A$ **then**
    $L[v, t_0] \leftarrow L[v, t_0] \cdot (1 - P_S^v(A(v) - T_E))$;
  **for** $w \in NE$ **do**
    $L[v, t_0] \leftarrow L[v, t_0] \cdot P_S^v(NE(v) - T_E)$;
**return** $L$

**Abbreviations**

| | |
|---|---|
| SICTF | Source identification via contact tracing framework |
| DDE | Deterministically developing epidemic |
| $DDE_{NR}$ | Deterministically developing epidemic with no recovery |
| HNM | Household network model |
| DCS | Data-driven COVID-19 simulator |
| TU | Tubingen mobility model |
| LS | Local search |

**Acknowledgements**
Not applicable.

**Author contributions**
GÓ, JV and PT designed the research and wrote the paper. GÓ and JV derived the analytic approximations. GÓ and M-ASN developed computational tools. All authors read and approved the final manuscript.

**Funding**
Open access funding provided by EPFL Lausanne. The work presented in this paper was supported in part by the Swiss National Science Foundation under grant numbers 200021-182407 and P500PT-211129.

**Availibility of data and materials**
The datasets used and/or analysed during the current study are available from the corresponding author on reasonable request.

## Declarations

**Competing interests**
The authors declare that they have no competing interests.

Received: 1 February 2023 Accepted: 26 June 2023
Published online: 21 August 2023

**References**

Ahn NY, Park JE, Lee DH, Hong PC (2020) Balancing personal privacy and public safety during COVID-19: the case of South Korea. Ieee Access 8(2020):171325–171333
Altarelli F, Braunstein A, Dall'Asta L, Ingrosso A, Zecchina R (2014) The patient-zero problem with noisy observations. J Stat Mech Theory Exp 10(2014):P10016
Auffinger A, Damron M, Hanson J (2015) 50 years of first passage percolation. Preprint arXiv:1511.03262
Balcan D, Gonçalves B, Hao H, Ramasco JJ, Colizza V, Vespignani A (2010) Modeling the spatial spread of infectious diseases: the GLobal Epidemic and Mobility computational model. J Comput Sci 1(3):132–145
Ball F, Sirl D, Trapman P (2009) Threshold behaviour and final outcome of an epidemic on a random network with household structure. Adv Appl Probab 41(3):765–796
BeidasRinad S, ButtenheimAlison M, KilaruAustin S, AschDavid A, VolppKevin G, LawmanHannah G, CannuscioCarolyn C, et al. (2020) Optimizing and implementing contact tracing through behavioral economics. NEJM Catalyst Innovations in Care Delivery
Benjamini I, Schramm O (2011) Recurrence of distributional limits of finite planar graphs. In: Selected works of oded schramm. Springer, pp 533–545
Bradshaw WJ, Alley EC, Huggins JH, Lloyd AL, Esvelt KM (2021) Bidirectional contact tracing could dramatically improve COVID-19 control. Nat Commun 12(1):1–9
Braithwaite I, Callender T, Bullock M, Aldridge RW (2020) Automated and partly automated contact tracing: a systematic review to inform the control of COVID-19. The Lancet Digital Health (2020)
Carinci F (2020) Covid-19: preparedness, decentralisation, and the hunt for patient zero. https://doi.org/10.1136/bmj.m799
Chai Y, Wang Y, Zhu L (2021) Information sources estimation in time-varying networks. IEEE Trans Inf For Secur 16(2021):2621–2636. https://doi.org/10.1109/TIFS.2021.3050604
Chang S, Pierson E, Koh PW, Gerardin J, Redbird B, Grusky D, Leskovec J (2021) Mobility network models of COVID-19 explain inequities and inform reopening. Nature 589(7840):82–87
Chen S, Yu P-D, Tan CW, Poor HV (2022) Identifying the superspreader in proactive backward contact tracing by deep learning. In: 2022 56th annual conference on information sciences and systems (CISS). IEEE, pp 43–48
Dawkins Q, Li T, Xu H (2021) Diffusion Source Identification on Networks with Statistical Confidence. In *Proceedings of the 38th international conference on machine learning* (Proceedings of machine learning research, Vol 139). PMLR, pp 2500–2509
Drmota M (2009) Random trees: an interplay between combinatorics and probability. Springer
Endo A et al (2020) Implication of backward contact tracing in the presence of overdispersed transmission in COVID-19 outbreaks. Wellcome Open Res 5
Fan L, Li B, Liu D, Dai H, Ru Y (2020) Identifying propagation source in temporal networks based on label propagation. In: International conference of pioneering computer scientists, engineers and educators. Springer, pp 72–88
Feng M, Ling Q, Xiong J, Manyande A, Weiguo X, Xiang B (2021) Occupational characteristics and management measures of sporadic COVID-19 outbreaks from June 2020 to January 2021 in China: the importance of tracking down "patient zero". Front Public Health 9(2021):670669
Feng Y, Mahmoud H (2018) Profile of random exponential binary trees. Methodol Comput Appl Probab 20(2):575–587
Hernando C, Mora M, Slater PJ, Wood DR (2008) Fault-tolerant metric dimension of graphs. Convexity Discrete Struct 5(2008):81–85
Huang Q (2017) Source locating of spreading dynamics in temporal networks. In: Proceedings of the 26th international conference on world wide web companion, pp 723–727
Ingraham NE, Ingbar DH (2021) The omicron variant of SARS-CoV-2: understanding the known and living with unknowns. Clin Transl Med 11(12):e685. https://doi.org/10.1002/ctm2.685
Jagers P, Nerman O (1984) The growth and composition of branching populations. Adv Appl Probab 16(2):221–259
Jiang J, Wen S, Shui Yu, Xiang Y, Zhou W (2016) Rumor source identification in social networks with time-varying topology. IEEE Trans Dependable Secure Comput 15(1):166–179
Kandeel M, Elsayed Mohamed Mohamed M, Abd El-Lateef HM, Venugopala N, El-Beltagi HS (2021) Omicron variant genome evolution and phylogenetics. J Med Virol
Karrer B, Newman MEJ (2010) Message passing approach for general epidemic models. Phys Rev E 82(1):016101. https://doi.org/10.1103/PhysRevE.82.016101
Kendall M, Tsallis D, Wymant C, Di Francia A, Balogun Y, Didelot X, Ferretti L, Fraser C (2023) Epidemiological impacts of the NHS COVID-19 app in England and Wales throughout its first year. Nat Commun 14(1):858

Kojaku S, Hébert-Dufresne L, Mones E, Lehmann S, Ahn Y-Y (2021) The effectiveness of backward contact tracing in networks. Nat Phys 2021:1–7
Kopel J, Goyal H, Perisetti A (2021) Antibody tests for COVID-19. In: Baylor University medical center proceedings, Vol 34. Taylor & Francis, pp 63–72
Kretzschmar ME, Rozhnova G, Bootsma MCJ, van Boven M, van de Wijgert JHHM, Bonten MJM (2020) Impact of delays on effectiveness of contact tracing strategies for COVID-19: a modelling study. Lancet Public Health 5(8):e452–e459
Kupferschmidt K (2021) Where did 'weird'Omicron come from? American Association for the Advancement of Science
Lecomte V, Ódor G, Thiran P (2020) The power of adaptivity in source location on the path. Theoretical Computer Science, Vol. 911, pp 92-123, 2022. Preprint arXiv:2002.07336
Lei Q, Li Y, Hou H, Wang F, Ouyang Z, Zhang Y, Lai D, Ndzouboukou J-LB, Zhao-wei X, Zhang B et al (2021) Antibody dynamics to SARS-CoV-2 in asymptomatic COVID-19 infections. Allergy 76(2):551–561
Li X, Wang X, Zhao C, Zhang X, Yi D (2019) Locating the source of diffusion in complex networks via Gaussian-based localization and deduction. Appl Sci 9(18):3758
Lokhov AY, Mézard M, Ohta H, Zdeborová L (2014) Inferring the origin of an epidemic with a dynamic message-passing algorithm. Phys Rev E 90(1):012801
Lorch L, Kremer H, Trouleau W, Tsirtsis S, Szanto A, Schölkopf B, Gomez-Rodriguez M (2022) Quantifying the effects of contact tracing, testing, and containment measures in the presence of infection hotspots. ACM Trans Spat Algorithms Syst 8(4):1–28
Louni A, Santhanakrishnan A, Subbalakshmi KP (2015) Identification of source of rumors in social networks with incomplete information. Preprint arXiv:1509.00557
Mahmoud H (2021) Profile of random exponential recursive trees. Methodol Comput Appl Probab 2021:1–17
Manitz J, Harbering J, Schmidt M, Kneib T, Schöbel A (2014a) Network-based source detection: from infectious disease spreading to train delay propagation. In: 29th international workshop on statistical modelling, vol 1, pp 201–205
Manitz J, Kneib T, Schlather M, Helbing D, Brockmann D (2014b) Origin detection during food-borne disease outbreaks-A case study of the 2011 EHEC/HUS outbreak in Germany. PLoS Curr 6
Mashkaria S, Ódor G, Thiran P (2020) On the robustness of the metric dimension to adding a single edge. Discrete Applied Mathematics, Vol 316, pp. 1-27, 2022. Preprint arXiv:2010.11023
Müller SA, Balmer M, Charlton W, Ewert R, Neumann A, Rakow C, Schlenther T, Nagel K (2021) Predicting the effects of COVID-19 related interventions in urban settings by combining activity-based modelling, agent-based simulation, and mobile phone data. PLoS ONE 16(10):e0259037
Odor G (2022) The Role of Adaptivity in Source Identification with Time Queries. Technical Report, EPFL
Paluch R, Gajewski ŁG, Hołyst JA, Szymanski BK (2020a) Optimizing sensors placement in complex networks for localization of hidden signal source: a review. Fut Gen Comput Syst 112(2020):1070–1092
Paluch R, Xiaoyan L, Suchecki K, Szymański BK, Hołyst JA (2018) Fast and accurate detection of spread source in large complex networks. Sci Rep 8(1):1–10
Paluch R, Suchecki K, Hołyst JA (2020b) Locating the source of interacting signal in complex networks. Preprint arXiv:2012.02039
Park YJ, Choe YJ, Park O, Park SY, Kim Y-M, Kim J, Kweon S, Woo Y, Gwack J, Kim SS et al (2020) Contact tracing during coronavirus disease outbreak, South Korea, 2020. Emerg Infect Dis 26(10):2465
Perrenet J, Zwaneveld B (2012) The many faces of the mathematical modeling cycle. J Math Model Appl 1(6):3–21
Pinto P, Thiran P, Vetterli M (2012) Locating the source of diffusion in large-scale networks. Phys Review Lett 109
Raymenants J, Geenen C, Thibaut J, Nelissen K, Gorissen S, Andre E (2022) Empirical evidence on the efficiency of backward contact tracing in COVID-19. Nat Commun 13(1):1–13
Russo L, Anastassopoulou C, Tsakris A, Bifulco GN, Campana EF, Toraldo G, Siettos C (2020) Tracing day-zero and forecasting the COVID-19 outbreak in Lombardy, Italy: a compartmental modelling and numerical optimization approach. PLoS ONE 15(10):e0240649. https://doi.org/10.1371/journal.pone.0240649
Shah D, Zaman T (2010) Detecting sources of computer viruses in networks: theory and experiment. In: Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems (New York, New York, USA) *(SIGMETRICS '10)*. ACM, New York, NY, USA, pp 203–214
Shah D, Zaman T (2011) Rumors in a network: Who's the culprit? IEEE Trans Inf Theory 57(8):5163–5181
Shelke S, Attar V (2019) Source detection of rumor in social network-a review. Online Soc Networks Media 9(2019):30–42
Shen Z, Cao S, Wang W-X, Di Z, Eugene HS (2016) Locating the source of diffusion in complex networks by time-reversal backward spreading. Phys Review E 93(3):032301
Shuaishuai X, Teng C, Zhou Y, Peng J, Zhang Y, Zhang Z-K (2019) Identifying the diffusion source in complex networks with limited observers. Phys A 527(2019):121267
Spinelli B, Celis E, Thiran P (2017a) A general framework for sensor placement in source localization. IEEE Trans Netw Sci Eng
Spinelli B, Celis LE, Thiran P (2017b) Back to the source: an online approach for sensor placement and source localization. In: Proceedings of the 26th international conference on world wide web, pp 1151–1160
Spinelli BL, Celis E, Thiran P (2017c) The effect of transmission variance on observer placement for source-localization. Appl Netw Sci 2(1):20
Spinelli B, Celis LE, Thiran P (2018) How many sensors to localize the source? The double metric dimension of random networks. In: 2018 56th annual allerton conference on communication, control, and computing (Allerton). IEEE, pp 1036–1043
Tang W, Ji F, Tay WP (2018) Estimating infection sources in networks using partial timestamps. IEEE Trans Inf For Secur 13(12):3035–3049
Troncoso C, Payer M, Hubaux J-P, Salathé M, Larus J, Bugnion E, Lueks W, Stadler T, Pyrgelis A, Antonioli D et al. (2020) Decentralized privacy-preserving proximity tracing. Preprint arXiv:2005.12273 (2020)
Waniek M, Holme P, Farrahi K, Emonet R, Cebrian M, Rahwan T (2022) Trading contact tracing efficiency for finding patient zero. Sci Rep 12(1):22582
Wormald NC et al (1999) Models of random regular graphs. Lond Math Soc Lect Note Ser 1999:239–298

Xie Y, Sekar V, Maltz DA, Reiter MK, Zhang H (2005) Worm origin identification using random moonwalks. In: 2005 IEEE symposium on security and privacy (S &P'05). IEEE, pp 242–256
Yu P-D, Tan CW, Hung-Lin F (2022) Epidemic source detection in contact tracing networks: epidemic centrality in graphs and message-passing algorithms. IEEE J Sel Top Signal Process 16(2):234–249
Zejnilovic S, Gomes J, Sinopoli B (2013) Network observability and localization of the source of diffusion based on a subset of nodes. In: 2013 51st annual allerton conference on communication, control, and computing (Allerton). IEEE, pp 847–852
Zejnilović S, Gomes J, Sinopoli B (2015) Sequential observer selection for source localization. In: 2015 IEEE Global Conference on signal and information processing (GlobalSIP). IEEE, pp 1220–1224
Zejnilović S, Gomes J, Sinopoli B (2017) Sequential source localization on graphs: a case study of cholera outbreak. In: 2017 IEEE global conference on signal and information processing (GlobalSIP). IEEE, pp 1010–1014
Zejnilović S, Mitsche D, Gomes J, Sinopoli B (2016) Extending the metric dimension to graphs with missing edges. Theoret Comput Sci 609(2016):384–394
Zhang X, Chen X, Zhang Z, Roy A, Shen Y (2020) Strategies to trace back the origin of COVID-19. J Infect 80(6):e39
Zhu K, Chen Z, Ying L (2016) Locating the contagion source in networks with partial timestamps. Data Min Knowl Disc 30(5):1217–1248
Zhao-Long H, Shen Z, Tang C-B, Xie B-B, Jian-Feng L (2018) Localization of diffusion sources in complex networks with sparse observations. Phys Lett A 382(14):931–937

## Publisher's Note