SIAM J. Control Optim.
Vol. 60, No. 2, pp. S274–S293

ADAPTIVE TEST ALLOCATION FOR OUTBREAK DETECTION AND TRACKING IN SOCIAL CONTACT NETWORKS[*]

PAU BATLLE[†], JOAN BRUNA[†], CARLOS FERNANDEZ-GRANDA[†], AND VICTOR M. PRECIADO[‡]

Abstract. We present a general framework for adaptive allocation of viral tests in social contact networks and arbitrary epidemic models. We pose and solve several complementary problems. First, we consider the design of a social sensing system whose objective is the early detection of a novel epidemic outbreak. In particular, we propose an algorithm to select a subset of individuals to be tested in order to detect the onset of an epidemic outbreak as fast as possible. We pose this problem as a hitting time probability maximization problem and use submodularity optimization and Monte Carlo techniques to obtain solutions with explicit quality guarantees. Second, once an epidemic outbreak has been detected, we consider the problem of using the data from the sensing system to obtain estimates of the initial patient and the current status of the epidemic. Finally, we consider the problem of adaptively distributing viral tests over time in order to maximize the information gained about the current state of the epidemic. We formalize this problem in terms of mutual information and propose an adaptive allocation strategy with quality guarantees. For these problems, we derive analytical solutions for any stochastic compartmental epidemic model with Markovian dynamics, as well as efficient Monte Carlo–based algorithms for non-Markovian dynamics or large networks. We illustrate the performance of the proposed framework in numerical experiments involving a model of COVID-19 applied to a real human contact network.

Key words. epidemiology, social networks, Markov chains, submodular optimization, entropy-based sampling

AMS subject classifications. 92D30, 91D30, 60J20, 90C35, 94A17

DOI. 10.1137/20M1377874

1. Introduction.

On December 31, 2019, The Municipal Health Commission of Wuhan (China) reported a cluster of cases of pneumonia caused by a novel coronavirus [3]. This new virus rapidly propagated worldwide through the air transportation network, and many countries decided to implement severe mobility restrictions and social distancing policies to "flatten the curve" of the pandemic. Through 2020 and 2021, the evolution in the pandemic followed a pattern of waves in which periods of exponential growth alternated with periods of decreasing cases, both due to new circulating variants [4] and varying strength of social distancing measures.

Therefore, in this situation, it is of utmost societal importance to develop efficient strategies for early detection and tracking of new epidemic waves in order to implement social distancing measures as fast as possible. Furthermore, information about where the epidemic outbreak started and the current state of the network is valuable to enact localized and effective measures.

[*] Received by the editors November 2, 2020; accepted for publication (in revised form) December 14, 2021; published electronically April 4, 2022.
https://doi.org/10.1137/20M1377874

Funding: The first author's research was partially supported by Capital One Inc. and the Cellex Foundation. The second author's research was partially supported by the Alfred P. Sloan Foundation, and by NSF RI-1816753 and NSF CAREER CIF 1845360. The third author's research was partially supported by NSF DMS 2009752. The fourth author's research was partially supported by NSF CAREER ECCS-1651433 and NSF-III-200884556, and by the Rockefeller Foundation.

[†] Courant Institute of Mathematical Sciences, Center for Data Science, New York University, New York, NY 10012 USA (pau.batlle.franch@gmail.com, bruna@cims.nyu.edu, cfgranda@cims.nyu.edu).

[‡] Department of Electrical and Systems Engineering, Applied Mathematics and Computational Science, University of Pennsylvania, Philadelphia, PA 19104 USA (victormpreciado@gmail.com).

In this paper, we study the problem of allocating viral tests [2] for an *arbitrary* continuous time epidemic model in order to (i) detect a novel epidemic outbreak as early as possible, as well as to (ii) retrieve as much information as possible about the evolution of the epidemic given the obtained data. In our work, we consider a social contact network over which a disease is spreading according to a stochastic compartmental model [26]. The main questions explored in this article are the following:

1. *Early detection of epidemic outbreaks with limited viral tests*: What nodes should we test in a contact social graph to maximize the probability of early detection? We will pose this problem in terms of hitting times of a stochastic process associated to the social graph and propose an algorithm to solve it with quality guarantees based on submodular optimization.

2. *Estimation of past and current state of the disease*: Given the results of a collection of viral tests, what are the probabilities of infection for each individual in the social network? Furthermore, we analyze the past evolution of the epidemic to estimate where and when the infection is most likely to have started.

3. *Adaptive test allocation for epidemic tracking*: Once an epidemic outbreak has been detected, how should we dynamically allocate viral tests to gain as much information as possible about the current state of the epidemic?

General work on stochastic compartmental models in networks include [22], [25], [18]. Additionally, [26], [6] provide a general survey of problems involving spreading processes in networks.

In [23], van Mieghen, Omic, and Kooij study the spread of malware in computer networks using Markov chains; however, their focus is on mean-field approximations derived from continuous-time Markov chains, while our work focuses on the exact stochastic model of the epidemic process.

The first question of which nodes to test for early detection has been explored in [21], where Leskovec et al. propose a sensor placement framework to detect outbreaks of water contaminants and other spreads. In this work, we use a different objective function that applies to any continuous-time epidemic process and that is both interpretable and reliable to estimate via Monte Carlo samples. We prove that our objective function and its Monte Carlo approximation have the same submodularity properties as their family of functions, and therefore the same approximation quality guarantees and optimization techniques they developed follow. Moreover, in this paper we analyze the impact of optimizing with an approximation to the true function and answer the question of how to use the placed tests to obtain information from the ongoing epidemic *once* the tests have been decided and detect the epidemic for the first time.

In [28], Shah and Zaman study the culprit detection problem, developing results for the popular SI epidemic model on a network using the novel concept of rumor centrality. Our work in the second question differs from theirs in two fundamental aspects: first, we do not consider a fixed model and our methodology is applicable to any compartmental epidemic model, and second, we focus on the information obtained by the tested nodes only. In [29], Spinelli, Celis, and Thiran also study the culprit detection problem for a specific family of spreading models without considering early detection. In [32], Yan et al. consider the independent cascade model and study the problem of immunizing edges in order to minimize the expected number of infected nodes at the conclusion of the spreading process.

As far as testing is concerned, there is literature that gives results on different network monitoring techniques [8], [12], [27], [30], [13]; however, these works do not aim to find an optimal solution according to any metric, but they analyze the performance of particular heuristics. Finally, the works in [5], [9], [19] analyze several heuristics for epidemic detection based on different network centrality measures.

The third question on how to dynamically allocate tests can be seen as a particular case of the general sequential Bayesian experiment design problem introduced in [14]. In our work, we focus on an application of the well-known submodularity property of mutual information to propose an algorithm for dynamic testing, similarly as in [20] but with Monte Carlo samples of the epidemic model instead of having access to the analytic likelihood of a Gaussian process.

The article is organized as follows. In section 2, we formalize our theoretical setup and discuss the models to which this framework is applicable. The three questions described above are explored in sections 3, 4, and 5, respectively. Finally, section 6 presents experiments in a real dataset of human interactions where we apply our framework using a realistic model of the spread of COVID-19 [10].

2. Notation and preliminaries.

For a given $n \in \mathbb{N}$, we let $[n]$ be the set $\{1, \dots, n\}$. We consider a given network $G = (V, E)$ where $V = [n]$, and a continuous-time stochastic compartmental epidemic process, denoted by $\{X(t)\}_{t\geq 0}$, running over G. At every $t \in \mathbb{R}_+$, each of the n nodes in the network is in one out of s possible states, where each state represents a compartment in the epidemic model. Since we have n nodes, the networked stochastic process $\{X(t)\}_{t\geq 0}$ has a finite state space $\mathcal{S}$ with $|\mathcal{S}| = s^n$. One of the simplest networked compartmental models is the SIR model [17], which presents three compartments: Susceptible, Infectious, and Removed. In this model, infectious nodes may infect healthy neighbors with probability rate β and may transition into the removed compartment (i.e., no longer infectious) with probability rate γ.

In the rest of the paper, we assume that the initial state $X(0)$ of the epidemic process is randomly chosen from a known probability distribution D supported in $\mathcal{S}$ and that all subsequent probabilities are conditioned on the realization of $X(0)$. If the epidemic process $\{X(t)\}_{t\geq 0}$ is Markovian, we can derive analytical solutions to the problems under consideration (shown in Appendix A). However, these analytical solutions are usable in practice only for relatively small graphs. In the following sections, we will provide efficient computational tools to analyze non-Markovian epidemic processes running over large graphs.

3. Early detection of epidemic outbreaks with limited viral tests.

The first key question we address is how to optimally monitor a contact network for early detection of a new outbreak with limited resources. We assume that we are able to continuously monitor the health of k nodes of the network before the onset of an outbreak. We aim to answer two optimization questions.

Q1A (test placement with monitoring constraint). *For a given $k \in \mathbb{N}$ and $\tau > 0$, which k nodes should we continuously monitor to detect a novel outbreak before a certain time τ (counting from the onset of the outbreak) with the highest possible probability?*

Q1B (test placement with probability constraint). *Given a threshold time $\tau > 0$ and a probability P, what is the minimum number k of nodes we need to monitor to detect the epidemic outbreak before time τ with a probability P? Where should we place them?*

To analyze these questions, we assume that those nodes being monitored are frequently tested. We assume that the available tests provide partial information about the state of the node. In particular, we consider a partition of the set of s possible states into two nonempty subsets, G_+ and G_-, and assume that the test is able to determine in what subset the state of the monitored node is. In practice, the set G_+ (resp., G_-) represents node states that would result in a positive (resp., negative) viral test result. We say that the node is *detectable* if its state is in G_+ and that the epidemic is detected when one of the monitored nodes becomes detectable for the first time.

Given a network stochastic process $\{X(t)\}_{t\geq 0}$ and a subset $A \subset \mathcal{S}$, its *stopping time* T_A is defined as the random variable $\min\{t \geq 0 : X(t) \in A\}$, where $X(t) \in \mathcal{S}$ is the state of the stochastic process at time t. If the process never reaches A, we set $T_A = \infty$. Given a subset of nodes $W \subset V$, we additionally define the *detection set* of W, denoted by D_W, as the subset of $\mathcal{S}$ consisting of those network states in which at least one of the nodes in W is in a detectable state (i.e., one of the nodes in W would test positive). This means that if we monitor the nodes in W, the epidemic outbreak is detected when the network process $\{X(t)\}_{t\geq 0}$ reaches one of the states in D_W.

Now, Question **Q1A** can be formalized as follows: Given a time horizon $\tau > 0$, we want to monitor k nodes of the network in order to maximize the probability that the process reaches the detection set D_W before time $t = \tau$ (counting from the onset of the epidemic outbreak). Hence, the optimal set of nodes to be monitored can be found as the solution of the following optimization problem:

$$\text{(Q1A)} \qquad \underset{W\subset V, |W|=k}{\operatorname{argmax}} \ \mathbb{P}\left(T_{D_W} \leq \tau\right).$$

Similarly, the answer to Question **Q1B** is the solution to the following optimization:

$$\text{(Q1B)} \qquad \underset{W\subset V \text{ s.t. } \mathbb{P}\left(T_{D_W}\leq\tau\right)\geq P}{\operatorname{argmin}} \quad |W|,$$

i.e, the smallest set of nodes that we need to monitor such that the probability of detecting the epidemic outbreak before time $t = \tau$ is greater than P. We conveniently define the optimization objective function over subsets of nodes for a given τ as $f_\tau \colon W \mapsto \mathbb{P}\left(T_{D_W} \leq \tau\right)$, so that (**Q1A**) and (**Q1B**) can be written, respectively, as

$$\text{(3.1)} \qquad \underset{W\subset V, |W|=k}{\operatorname{argmax}} \ f_\tau(W) \quad \text{and} \quad \underset{W\subset V, f_\tau(W)\geq P}{\operatorname{argmin}} \ |W| \,.$$

Notice that these are combinatorial optimization problems and that finding the optimal solutions is exponentially hard. In the rest of the paper, we focus on finding approximate solutions with quality guarantees. In this direction, there are two separate subproblems we need to address: First, the function f_τ can only be computed for Markovian epidemic processes taking place in small networks (see Appendix A); hence, we need to approximate this objective function for non-Markovian processes over large networks. Second, we also need an optimization scheme to find an approximate solution with quality guarantees (without evaluating f_τ an exponential number of times).

3.1. Function evaluation. The function f_τ can be approximated using Monte Carlo samples, as described below. First, we simulate the stochastic epidemic process N_R times, where each simulation will be stopped when one of two things happen: Either we reach an absorbing state, or all the nodes have already reached a detectable state at least once. Then, f_τ can be approximated as follows: Let L be an $N_R \times n$

matrix such that, for every run of the process $r \in [N_R]$,

$$L[r,j] = \min\{T \in \mathbb{R}_+ \colon X_j(T) \in G_+ \text{ in run } r\}\ , \tag{3.2}$$

where $X_j(t)$ is the state of node j at time t and $L[r,j] = \infty$ if node j is never detectable in run r. Given the matrix L and any time τ, an estimator for f_τ, denoted by $\hat{f}_\tau$, can be calculated as follows: $\hat{f}_\tau(W) = \frac{1}{N_R}|\{r \in [N_R] \colon \min_{i\in W} L[r,i] \leq \tau\}|$. As $N_R \to \infty$, we have that $\hat{f}_\tau \to f_\tau$ uniformly because of the law of large numbers. Using standard Chernoff bounds, one can show that the probability of $\hat{f}_\tau(W)$ not being in an interval $((1-\delta)f_\tau(W), (1+\delta)f_\tau(W))$ for any $\delta \in [0,1]$ is at most $2\exp(-\frac{\delta^2}{2+\delta}f_\tau(W)N_R)$, so to obtain a probability of at least $1-\alpha$ of $(1-\delta)f_\tau(W) \leq \hat{f}_\tau(W) \leq (1+\delta)f_\tau(W)$, $N_R = \Omega(\frac{\log 1/\alpha}{\delta^2 f_\tau(W)})$ samples suffice.

After the matrix L is precomputed, a query of the approximate function $\hat{f}$ has a worst case complexity of $\mathcal{O}(|W|N_R)$, $\mathcal{O}(nN_R)$ when $|W| = \mathcal{O}(n)$. One can reduce the run-time complexity even more at the expense of memory complexity by storing suitable function values.

3.2. Optimization of f_τ via submodularity. The combinatorial structure of the problem requires not only a way to rapidly evaluate the objective function but an optimization scheme that avoids evaluating an exponential number of possible node monitorizations. In order to do that, we prove and exploit the submodularity properties of f_τ and $\hat{f}_\tau$ combined with fundamental results about submodular optimization from the literature.

If Ω is a finite set, a function $h \colon \mathcal{P}(\Omega) \to \mathbb{R}$ is called *submodular* if it satisfies one of these three equivalent conditions.

Condition 1. For all $X, Y \subseteq \Omega$ with $X \subseteq Y$ and every $x \in \Omega \setminus Y$, we have that $h(X \cup \{x\}) - h(X) \geq h(Y \cup \{x\}) - h(Y)$.

Condition 2. For all $S, T \subseteq \Omega$ we have that $h(S) + h(T) \geq h(S \cup T) + h(S \cap T)$.

Condition 3. For all $X \subseteq \Omega$ and $x_1, x_2 \in \Omega\backslash X$ such that $x_1 \neq x_2$, $h(X \cup \{x_1\}) + h(X \cup \{x_2\}) \geq h(X \cup \{x_1, x_2\}) + h(X)$.

We aim to prove that f_τ is a nonnegative, monotone (i.e., $f_\tau(X) \leq f_\tau(Y)$ for $X \subset Y$), and submodular function. The non-negativity is trivial from the definition of probability, and monotonicity comes from the fact that for $A \subset B$, $D_A \subset D_B$, and so the event $T_{D_A} \leq \tau$ implies that $T_{D_B} \leq \tau$, and hence $f_\tau(A) \leq f_\tau(B)$. Furthermore, f_τ is submodular (as proved in Appendix B).

THEOREM 3.1. *The set-function $f_\tau \colon W \mapsto \mathbb{P}(T_{D_W} \leq \tau)$ is submodular.*

We can now invoke two well-known results in submodular optimization theory to derive quality guarantees of greedy-like optimization schemes aiming to solve problems (**Q1A**) and (**Q1B**), using Algorithms 3.1 and 3.2, described below. Both algorithms run in polynomial time and only require $\mathcal{O}(nk)$ evaluations of the objective function, or $\mathcal{O}(n^2)$ when $k = \mathcal{O}(n)$.

THEOREM 3.2 (see [24]). *If a set-function f is monotone, submodular, and nonnegative, the greedy scheme in Algorithm* 3.1 *applied to problem* (**Q1A**) *returns a solution S' for which $f(S') \geq (1-\frac{1}{e})f(S^*)$, where S^* is the optimal set.*

THEOREM 3.3 (see [31]). *If f is monotone and submodular, the greedy scheme in Algorithm* 3.2 *applied to problem* (**Q1B**) *returns a solution S' for which $\frac{|S'|}{|S^*|} \leq 1 + \log\frac{f(V)-f(\emptyset)}{f(S')-f(S_{-1})}$, where S^* is the optimal set and S_{-1} is the solution set at the iteration prior to the termination of Algorithm* 3.2.

Algorithm 3.1. Greedy scheme applicable to (**Q1A**) when $f = f_\tau$.

```
Input: $k \in \mathbb{N}, f$ function over subsets of $V$
Output: $S \subset V$ with $|S| = k$, an approximate solution to (Q1A)
$S \leftarrow \emptyset$, $i \leftarrow 0$
while $i \leq k$ do
    $S \leftarrow S \cup \operatorname*{argmax}_{v \in V \setminus S} f(S \cup \{v\})$
    $i \leftarrow i + 1$
return $S$
```

Algorithm 3.2. Greedy scheme applicable to (**Q1B**) when $f = f_\tau$.

```
Input: $P \in [0, 1], f$ set-function over subsets of $V$
Output: $S \subset V$ with $f(S) \geq P$, an approximate solution to (Q1B)
$S \leftarrow \emptyset$
while $f(S) < P$ do
    $S \leftarrow S \cup \operatorname*{argmax}_{v \in V \setminus S} f(S \cup \{v\})$
return $S$
```

Note that in the case of non-Markovian epidemic models and/or large networks, we cannot directly evaluate f_τ, but we can with the previously defined approximation $\hat{f}_\tau$. A natural question is whether $\hat{f}_\tau$ has properties similar to those of f_τ, so that we can guarantee quality of the optimization. The answer to this question is positive and summarized in the following theorem.

THEOREM 3.4. *The approximation function $\hat{f}_\tau$, defined in subsection* 3.1*, is nonnegative, monotone, and submodular for all $N_R \in \mathbb{N}$.*

Using this result (proved in Appendix B), we conclude that the quality guarantees in Theorems 3.2 and 3.3 are also applicable to the approximation function $\hat{f}_\tau$. Furthermore, if we have chosen N_R so that $(1 - \delta) f_\tau \leq \hat{f}_\tau \leq (1 + \delta) f_\tau$ with sufficient probability, then one can derive guarantees similar to those of Theorems 3.2 and 3.3 about the performance of the greedily selected subset using evaluations of $\hat{f}_\tau$, relating it to the true optimum of f_τ as a function of δ.

THEOREM 3.5. *Let S^* be the optimal set for f_τ. Let $\hat{f}_\tau$ be a nonnegative, monotone, and submodular function such that $(1 - \delta) f_\tau \leq \hat{f}_\tau \leq (1 + \delta) f_\tau$. Then, using Algorithm* 3.1 *with evaluations of $\hat{f}_\tau$ yields a solution $\hat{S}'$ such that $f_\tau(\hat{S}') \geq \frac{1-\delta}{1+\delta} \left(1 - \frac{1}{e}\right) f_\tau(S^*)$.*

THEOREM 3.6. *Let S^* be the optimal set for f_τ. Let $\hat{f}_\tau$ be a nonnegative, monotone, and submodular function such that $(1 - \delta) f_\tau \leq \hat{f}_\tau \leq (1 + \delta) f_\tau$. Then, using Algorithm* 3.2 *with evaluations of $\hat{f}_\tau$ until finding a set with $\hat{f}_\tau(W) \geq P(1 - \delta)$ yields a solution $\hat{S}'$ such that $\frac{|\hat{S}'|}{|S^*|} \leq 1 + \log \frac{(1+\delta) f_\tau(V) - (1-\delta) f_\tau(\emptyset)}{(1-\delta) f_\tau(\hat{S}') - (1+\delta) f_\tau(\hat{S}_{-1})}$, where $\hat{S}_{-1}$ is the solution set at the iteration prior to the termination of Algorithm* 3.2 *with $\hat{f}_\tau$.*

The proofs of Theorems 3.5 and 3.6 are included in Appendix B.

Combining the complexity of the greedy schemes and the complexity of evaluating $\hat{f}_\tau$, the overall complexity of running Algorithms 3.1 and 3.2 in their most naive version is $\mathcal{O}(nk^2 N_R)$, and $\mathcal{O}(n^3 N_R)$ in the worst case of $k = \mathcal{O}(n)$. In practice, one can significantly decrease the number of function evaluations by further exploiting submodularity of the objective function to perform lazy evaluation, as explained in [21].

3.3. Toy example in a small network. We illustrate our procedures using the graph in Figure 1. We use an SIR model with $\beta = 0.5$, $\delta = 0.25$, and a single initially infected node chosen uniformly at random. Setting $\tau = 0.5$, the set of $k = 2$ nodes to be monitored such that f_τ is maximized is $\{1, 5\}$ (circled in black in the figure) with a value of $f_\tau(\{1, 5\}) = 0.442$; in other words, monitoring these two nodes, we are able to detect the epidemic outbreak before 0.5 time units with a probability equal to 0.442. This solution is obtained via an exhaustive combinatorial search. If, in contrast, we use the greedy scheme in Algorithm 3.1, we obtain $\{3, 0\}$ as our approximate solution and $f_\tau(\{3, 0\}) = 0.438$. Theorem 3.4 ensures that the greedy solution (i.e., 0.438) is not worse than $(1 - 1/e) \times 0.442 = 0.279$ (notice that the greedy solution is much better than that worst case value).

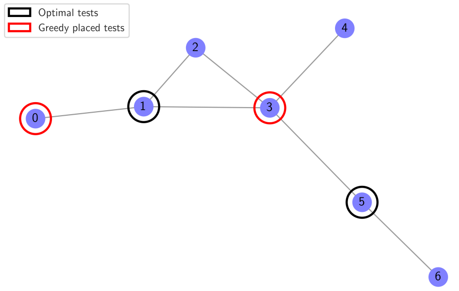

FIG. 1. *Toy example used in subsection* 3.3 *with* $n = 7$ *nodes. For* $\tau = 0.5$, *nodes* 1 *and* 5 *are the optimal set, but a greedy approach selects nodes* 3 *and* 0.

4. Estimation of past and current state of the disease.

Once an epidemic outbreak has been detected, it is of practical interest to use the results of the tests used during the monitoring phase to estimate the network state of the disease. In this section, we estimate the global state of the network using the information obtained from the viral test results retrieved from a subset of nodes. In this direction, given test results for a subset of nodes, we formulate three different subquestions.

Q2A (patient zero detection). *What is the probability of each node being patient zero?*

Q2B (outbreak time estimation). *How much time has passed since the outbreak started?*

Q2C (current network status assessment). *What is the probability of each individual node being infected?*

Assuming that nodes $v_1, \ldots, v_k \in V$ are our k monitoring nodes, we define the k dimensional observation vector O such that $O_i = 1$ when v_i have tested positive during the monitoring phase and $O_i = 0$ otherwise. Hence, assuming that $x_i \in \mathcal{S}$ is the network state in which only node i is infected, **Q2A** asks us to estimate

$$\mathbb{P}(X(0) = x_i | O) \propto \mathbb{P}(O | X(0) = x_i)\mathbb{P}(X(0) = x_i) \quad \forall i, \tag{Q2A}$$

while **Q2B** asks us about the distribution of the time t since the beginning of the epidemic outbreak conditioned to our observation O, i.e., $\mathbb{P}(t \leq u | O)$ for $u \in \mathbb{R}_+$.

In subproblem **Q2C**, we aim to estimate $\mathbb{P}(X = x|O)$, where X is the state of the stochastic process at the present time. However, since the number of possible network states grows exponentially with the number of nodes, it is computationally intractable to solve **Q2C**. Alternatively, we will aim to estimate the $n \times s$ marginal probabilities $\{\mathbb{P}(S_i = s_j|O)\}_{i=1:n,j=1:s}$, where S_i is the current state of node i and $s_j \in [s]$ is one of the possible s states or compartments. We can also collect reliable information about correlations between the status of a particular node and other nodes in the network.

To estimate solutions to questions **Q2A**, **Q2B**, and **Q2C**, we are using the Monte Carlo estimator for conditioned probability, as described in the previous section. Here, $Obs_r \in \{0,1\}^k$ refers to the observation vector O at detection time t_r obtained in the rth run of the Monte Carlo iteration with initial state $X^r(0)$. Hence, the approximate solution for **Q2A** (patient zero detection) is the following distribution over initial states:

$$\hat{\mathbb{P}}(X(0) = x_i|O) = \frac{|\{r \in [N_R]\colon X^r(0) = x_i \cap Obs_r = O\}|}{|\{r \in [N_R]\colon Obs_r = O\}|}. \tag{4.1}$$

For **Q2B** (outbreak time estimation), our empirical distribution of times depends on how long it took to detect the outbreak in those runs producing an observation $Obs_r = O$, as stated in the following equation:

$$\hat{\mathbb{P}}(t \le k|O) = \frac{|\{r \in [N_R]\colon t_r \le k \cap Obs_r = O\}|}{|\{r \in [N_R]\colon Obs_r = O\}|}. \tag{4.2}$$

Defining S_i^r as the status of node i at the time of detection of run r, we have the following approximation for **Q2C** (current status assessment):

$$\hat{\mathbb{P}}(S_i = s_j|O) = \frac{|\{r \in [N_R]\colon S_i^r = s_j \cap Obs_r = O\}|}{|\{r \in [N_R]\colon Obs_r = O\}|} . \tag{4.3}$$

Finally, we can obtain information about correlations in a similar way:

$$\hat{\mathbb{P}}(S_i = s_j|S_k = s_l, O) = \frac{|\{r \in [N_R]\colon S_i^r = s_j \cap S_k^r = s_l \cap Obs_r = O\}|}{|\{r \in [N_R]\colon Obs_r = O \cap S_k^r = s_l\}|} . \tag{4.4}$$

All the estimators converge to the true probabilities as $N_R \to \infty$, with the same convergence rates of $\hat{f}$ in the previous section. In particular, the same bounds on the number of runs required so that the estimators are within a factor of $1 \pm \delta$ of the true values with high probability apply.

4.1. Toy example in a small network.

Consider an SIR epidemic model on the network in Figure 1 using the same settings as in subsection 3.3. Let us assume that we are continuously monitoring nodes 1 and 5. Suppose that the first time a test detects the epidemic, node 5 is infectious. Using our results, we can calculate the posterior distribution of patient-zero probabilities and the time-since-outbreak t, which are plotted in Figures 2 and 3, respectively. The probability of $t = 0$ (detection immediately after outbreak) is 0.416, in agreement with the posterior distribution of patient zero in Figure 2. The expected value of the distribution is 0.760. Finally, in Figure 4 the estimated marginal distributions for each node and state can be observed.

5. Dynamic allocation for epidemic tracking.

In a practical scenario, once an epidemic is detected we may be interested in retrieving as much information as possible about its state sequentially in different points in time. In this section, we

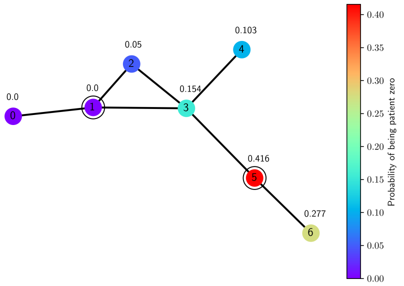

FIG. 2. *Posterior distribution of patient-zero probabilities for the example in subsection* 4.1.

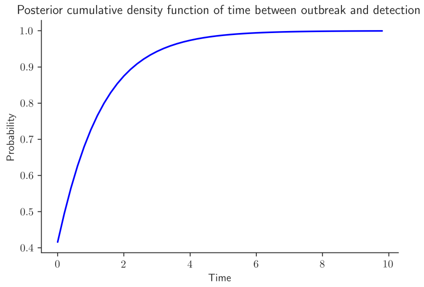

FIG. 3. *Posterior distribution of time-since-outbreak for the example in subsection* 4.1.

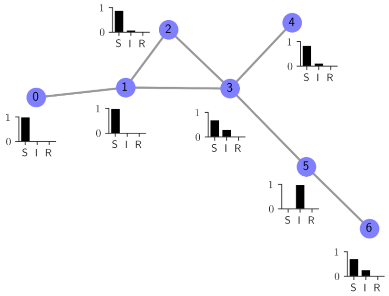

FIG. 4. *Marginal distributions of the current state of the network after detection, given the observation of node* 5 *as the first infectious between nodes* 1 *and* 5.

address this problem by providing quality guarantees of a strategy that assigns nodes to be tested over time.

We let t_0 be the time at which an epidemic outbreak is first detected and consider that, afterwards, we are able to perform a number of new tests $n_1, n_2, \dots, n_l$ at times $t_1 < t_2 < \cdots < t_l$. Question **Q3**, stated below, is concerned with the design of a testing strategy aiming to maximize the amount of information extracted about the state of the disease using this series of tests, as well as providing an algorithm to use Monte Carlo runs to go from the results from testing at time t_i to the tests to perform at time t_{i+1} for $1 \leq i < l$.

Q3 (optimal dynamic test allocation after detection). *Assuming that we are free to sequentially allocate n_i tests at each time t_i, which nodes should we test at each time to maximize the information about the state of the disease at each time?*

We let X be the random variable of the state of the epidemic at detection time conditioned on the observation of the sensors, from which we know the marginals using **Q2C**. This will act as a prior. We let A be the set of tested nodes tests and X_A the random variable including the status of the nodes in A. Similarly as in the Bayesian experiment design literature [7], we aim to maximize the mutual information between X and X_A, i.e., the information about X provided by knowing the result of testing the nodes in A. Equivalently, we aim to choose A so that the expected KL divergence between the prior p_X and the posterior $p_{X|X_A}$ is maximized. Here, the expectation is taken with respect to the test results.

We assume an online scenario without future rewards: this means that at time t_i

we receive the state of the network (coming from the initial marginals and updates from testing at all times before t_i) and the number of tests n_i; we then aim to choose nodes to test that maximize the information we obtain *immediately* after testing. This also corresponds to a case in which it is unknown how many tests (if any) will be available in the future.

The high-level overview of the proposed procedure is as follows. After tests at time t_i are taken, we define D_i as the distribution over nodes conditioned on the test results. If the process is Markovian, the conditions needed to start the model are just the status of all nodes. For non-Markovian processes, other elements, such as how much time a node has been in its status, need to be considered. We include these under D_i, understanding that we sample all the values needed to uniquely determine the system current status and evolution. We use the Monte Carlo simulator with initial states sampled from D_i until $t = t_{i+1} - t_i$. After we have enough runs, one can estimate the marginal distributions and correlations using (4.3) and (4.4), then decide where the new n_{i+1} optimal tests are allocated, and then calculate D_{i+1} using the results from the testing. This procedure is conceptually similar to particle filtering or sequential Monte Carlo methods, in which measurements of reality are combined sequentially with a simulator of the associated dynamics.

In the next subsections, we explain the mathematical details of each part of the process.

5.1. Choosing the optimal tests.

At any fixed time, we assume knowledge of (an approximation to) the marginals $\mathbb{P}(S_i = s_j|O)$ from (4.3) and the correlations $\mathbb{P}(S_i = s_j|S_k = s_l, O)$ from (4.4). We henceforth omit the dependence on O for notation simplicity. Our problem at time t_i then takes the form

$$\underset{A\subset V, |A|=n_i}{\operatorname{argmax}}\ I(X; X_A) = H(X) - H(X|X_A), \tag{5.1}$$

where $I(X; X_A)$ is the mutual information between X once we test the nodes in A, which can be expressed as the difference in entropies between X and $X|X_A$. Here, $H(X|X_A)$ is the expectation of the entropy of the random variable $X|(X_A = x_a)$ with respect to the test results x_a. This function is known to be submodular [11], [16], positive (by Jensen's inequality), and monotone; therefore, we can directly apply Theorem 3.2 to obtain the same $(1 - 1/e)$ approximation guarantee result as in Question **Q1A** when using the greedy algorithm. To use Algorithm 3.1, it remains to be seen how to evaluate $H(X|X_A)$ for subsets $A \subset V$.

5.2. Updating the marginals from the test results and evaluating the objective function.

Throughout, we assume conditional independence in test results, assuming that

$$\mathbb{P}(S_i = s_j|S_1 = s_1, \ldots, S_n = s_n) = \prod_{k=1}^{n} \mathbb{P}(S_i = s_j|S_k = s_k). \tag{5.2}$$

This is due to the fact that we can estimate the correlations $\mathbb{P}(S_i = s_j|S_k = s_k)$ with Monte Carlo runs but not higher order correlations, as there are $n^2 s^2$ first order correlations but $n^{k+1}s^{k+1}$ order k correlations, which is prohibitively large even for moderate k.

Let T_A be the set of possible test results; then $H(X|X_A) = \sum_{x_a \in T_A} H(X|X_A = x_a)\mathbb{P}(X_A = x_a)$. The elements in T_A and the values of $X_A = x_a$ depend on the tests available, and $\mathbb{P}(X_A = x_a)$ depends on the test parameters (sensitivity, specificity, etc.) and the input marginals. Since $H(X|X_A) = \mathbb{E}_{x_a}[H(X|X_A = x_a)]$, we can approximate $H(X|X_A)$ as the average of the entropy of the conditional distribution once the test results are obtained from the different Monte Carlo runs. In particular, for a single run in which the results x_a include each node α in A being in state $s(\alpha)$, we estimate the conditional entropy from the marginals as follows:

(5.3)

$$\begin{aligned}\hat{H}(X|X_A = x_a) &= -\sum_{i=1}^{n}\sum_{j=1}^{s}\mathbb{P}(S_i = s_j|X_A = x_a)\log\mathbb{P}(S_i = s_j|X_A = x_a)\\ &= -\sum_{i=1}^{n}\sum_{j=1}^{s}\prod_{\alpha\in A}\mathbb{P}(S_i = s_j|S_\alpha = s(\alpha))\log\mathbb{P}(S_i = s_j|S_\alpha = s(\alpha)),\end{aligned}$$

where in the last step we have used the conditional independence assumption and the values are known approximately from (4.4). Finally, we approximate the objective function $H(X|X_A)$ by $\hat{H}(X|X_A)$, the average of $\hat{H}(X|X_A = x_a)$ across Monte Carlo runs with different testing results x_a.

Once we get to know the true result of the tests x_{true}, all that remains is using the Monte Carlo simulator with the initial distribution being the new marginals $\{\mathbb{P}(S_i = s_j|X_A = x_{\text{true}})\}_{i=1:n;j=1:s}$ to obtain the marginals at t_{i+1}, a process that can be applied repeatedly.

5.3. Toy example in a small network. Figure 5 shows the process of adaptive testing in the case of the toy example in Figure 1. We start with the marginals in Figure 4, which are used to calculate their entropies and decide the optimal nodes to test next. After the test is taken, one uses the information to update the marginals and use them to sample the initial condition for subsequent Monte Carlo runs, in which we simulate the stochastic process until the time in which we are allowed to allocate more tests (e.g., every week). After that, we can update the marginals, decide on the next tests to take, and repeat the cycle.

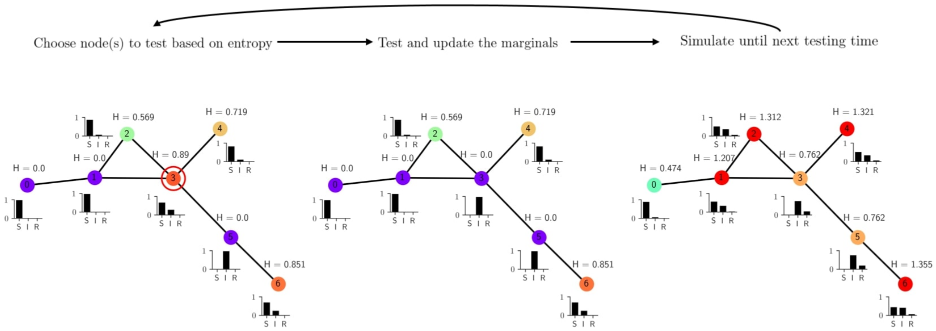

Fig. 5. *Dynamic testing process, consisting of alternating testing and simulation, applied to the toy example network.*

6. Experiments. In this section, we illustrate the proposed procedures to a non-Markovian model of COVID-19 in a real human interaction network. In order to simulate the stochastic processes, we use an event-driven simulation algorithm in which the next events (infections, recoveries, etc.) are stored in a priority queue and processed in order of time [18]. This method can be used to efficiently simulate many stochastic models, such as the model studied here. We use the Hypertext 2009 network [1], a network of human-to-human interactions, in our simulations. The ACM Conference on Hypertext and Hypermedia 2009 was held in Turin, Italy, in 2009 and, during the conference, the conference badges included radio-frequency identification (RFI) devices able to mine face-to-face proximity relations [15]. The exchange of radio packets between badges implies a proximity of less than 1–1.5m, a distance in which contagious diseases could spread. In this network, a node represents a conference visitor and an edge represents a face-to-face contact that was active for at least 20 seconds. The network has $n = 100$ nodes and $m = 946$ edges once we aggregate edges over time during the first day of the conference.

For this network, we use an adaptation to networks of a realistic non-Markovian model of COVID-19 proposed in [10]. In this work, the authors infer that, for COVID-19, the incubation period (time between contracting the disease and showing symptoms) follows a lognormal distribution with meanlog 1.644 and sdlog 0.363, and the generation time (time between infection of the source and infection of the target) follows a Weibull distribution with shape parameter 2.826 and scale parameter 5.665. Additionally, the authors infer that the proportion of infectious individuals who are asymptomatic is 0.4 and that the asymptomatic transmission rate is 10 times lower than for symptomatic patients. We use this data to create a non-Markovian model with susceptible, presymptomatic, symptomatic, asymptomatic, and removed compartments in which each node can independently infect its neighbors as long as it is not susceptible or removed. The times for that infection to occur and symptoms to appear is sampled from the distributions in [10]. We draw the random time to full recovery from first infection to removal from a normal distribution of mean 14 and standard deviation 2, both for symptomatic and asymptomatic carriers. The model is summarized in Figure 6.

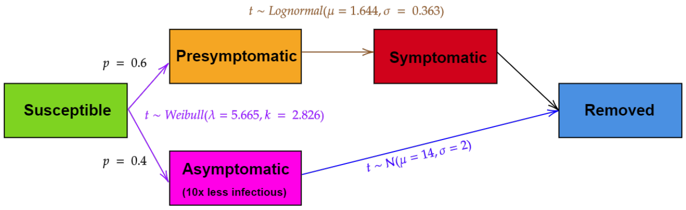
$t \sim Lognormal(\mu = 1.644, \sigma = 0.363)$
$p = 0.6$
Presymptomatic
Symptomatic
Susceptible
$t \sim Weibull(\lambda = 5.665, k = 2.826)$
Removed
$p = 0.4$
Asymptomatic
(10x less infectious)
$t \sim N(\mu = 14, \sigma = 2)$

FIG. 6. *Summary of the non-Markovian model of COVID-*19 *spreading in networks based in* [10].

We assume that the outbreak is started by a single infectious node chosen uniformly at random. We then monitor $k = 10$ nodes decided according to the greedy scheme in Algorithm 3.1, which aims to maximize the probability of detection during the first $\tau = 3$ days of the outbreak. We test the greedy algorithm against three simple baselines: uniformly random node subset selection, random node subset selection weighted by node degree, and random node selection eliminating neighbors from chosen nodes iteratively. Figure 7 summarizes the results, where the greedy algorithm outperforms all the strategies by around 3% in probability. In order to understand how this translates to real scenarios in practice, we run 10^5 simulations for the greedy test placement and 10^5 for the best randomly found placement in which we set lockdown measures as soon as the epidemic is detected in each case. The curves of infectious (asymptomatic + symptomatic + presymptomatic) and recovered nodes for each case can be seen in Figure 8. On average, four out of the 100 nodes do not contract the disease by using the greedy placement instead of the best random placement.

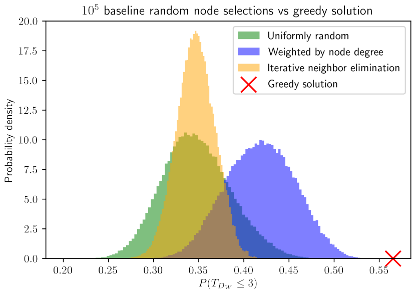

FIG. 7. *Histogram of results after using the three baseline random placement strategies in comparison to the greedy algorithm in order to place* $k = 10$ *tests. The greedy algorithm scores a detection probability of* 0.57*, while the best random solution scores* 0.542*.*

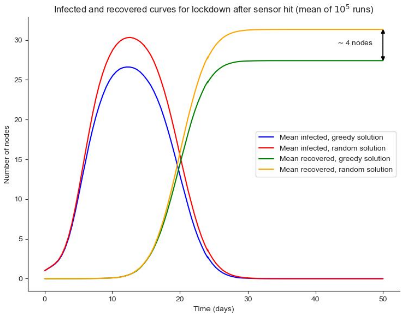

FIG. 8. *Infectious (asymptomatic + symptomatic + presymptomatic) and recovered mean curves for* 10^5 *runs in which lockdown is imposed once detecting the epidemic. The small probability gain of using the set of sensors found with the greedy strategy translates to four fewer nodes being infectious on average.*

A similar analysis as the one performed in the toy example in subsection 4.1 can be performed to estimate patient zero. The most likely node is in this case the node in which the epidemic was detected with a probability of 0.08 of being patient zero assuming a uniform prior. Similarly as before, we can estimate the probability density function of time-since-outbreak at the time of detection. Here, we do it for three different kinds of tests: tests that detect antibodies (meaning all kind of non-susceptible nodes), "tests" that detect symptoms only, and "tests" that detect removed people only. These two last cases correspond to the cases in which epidemic outbreaks are detected late instead of using actual viral tests, simulating scenarios in which countries or populations are unprepared for an outbreak and can only detect it after the first death (or person with symptoms) is detected. The results can be seen in Figure 9, in which we can observe that the difference is of the order of several days in each case.

Finally, we illustrate the proposed adaptive testing algorithm with a fixed amount of tests at $t = t_0$ and every 3 days, up to four times. We compare it with a baseline algorithm of randomly selecting which nodes to choose at each iteration. In this comparison, testing is only used to monitor the epidemic (i.e., no lockdown measures are imposed regardless of test results). In Figure 10, the mean entropy of the predicted marginal distributions are plotted over time, averaging over 10^4 different real runs. As expected, more tests translate distributions with less entropy. By comparing testing strategies, it can be observed how the proposed strategy improves on the uncertainty that the distributions convey. The time of higher uncertainty is around 6 days after detection, as there are more possible scenarios of the current state of the pandemic than later on. The reason for this is that, in this non-lockdown scenario, after a certain point most of the nodes will most likely have been infected.

7. Conclusions. We have introduced a flexible framework to analyze problems concerning the early detection of epidemic outbreaks, as well as the dynamic allo-

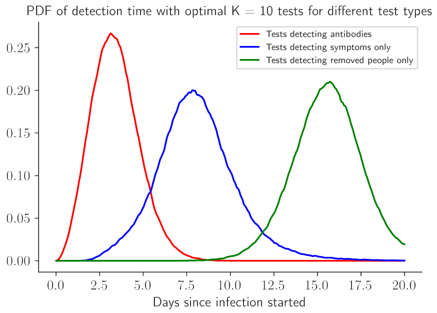

Fig. 9. *Probability density functions of times between disease outbreak and detection for optimal monitorization of $k = 10$ nodes using tests of different types, corresponding to being able to test people (left), detecting just the symptoms (center), or just the death of patients (right).*

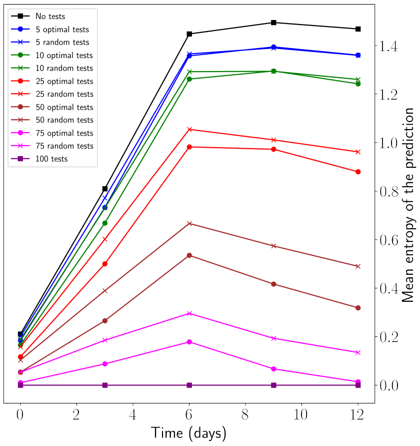

Fig. 10. *Mean entropy time evolution in the dynamic testing scenario in which a fixed number of tests are used every 3 days, up to 12 days after the outbreak is detected. Averages of 10^4 runs.*

cation of tests to maximize the information retrieved about the state of the infection. We have stated and solved several problems of practical interest by analyzing continuous-time stochastic compartmental models over complex networks. First, we have considered the problem of designing a monitoring system whose objective is to detect a novel epidemic outbreak as soon as possible. In particular, we have developed an algorithm able to select a subset of individuals to be continuously monitored in order to detect the onset of an epidemic as fast as possible. We have mathematically described this problem as a hitting-time probability maximization and use submodularity optimization techniques to derive explicit quality guarantees for the proposed solution. Second, assuming that an epidemic outbreak has been detected, we have also considered the problem of dynamically allocating viral tests over time in order to maximize the amount of information gained about the state of the epidemic. We have proposed an adaptive allocation strategy with quality guarantees based on the concepts of information entropy and mutual information. For all these problems, we have derived analytical solutions for Markovian stochastic compartmental models, as well as efficient Monte Carlo–based algorithms for non-Markovian dynamics and large-scale networks. We have illustrated the performance of the proposed algorithms using numerical experiments involving a model of COVID-19 applied to a real human contact network.

Appendix A. Analytical solution for Markov chains. In this section, we provide analytical solutions for the questions from section 1 in the case where the Markov property holds and the epidemic model defines a continuous-time Markov chain. Similarly as in the general case, the epidemic model can be formulated as a continuous-time Markov chain of state space $\mathcal{S}$ with $|\mathcal{S}| = s^n$ if the Markov property holds. An initial probability distribution D such that $X(0) \sim D$ is also assumed. The continuous-time Markov chain is characterized by a transition rate matrix Q of dimensions $s^n \times s^n$.

A.1. Early detection of epidemic outbreaks with limited viral tests. We use the same notation as in the general case: T_{D_W} is the minimum time in which the Markov chain reaches the detection set of $W \subset V$. We can analytically calculate $\mathbb{P}(T_{D_W} \leq \tau) = \mathbb{P}(T_{D_W} \leq \tau | T_{D_W} < \infty)\mathbb{P}(T_{D_W} < \infty)$, where $\mathbb{P}(T_{D_W} < \infty)$ refers to the probability of the Markov process ending in an absorbing state in D_W. Some models, such as SIR, have absorbing states which represent the epidemic dying before any detection is done (for example, the first person gets cured before transmitting it to anyone), which might therefore not be part of D_W. To treat that, we use the jump matrix of the Markov chain. If Q is the transition rate matrix of a continuous-time and discrete space-Markov chain and we are at state i, the probability of the next jump of the Markov chain being to state $j \neq i$ is $\frac{Q_{ij}}{-Q_{ii}}$. This lets us define the jump matrix M, where $M_{ij} = \frac{Q_{ij}}{-Q_{ii}}$ if $i \neq j$ and $M_{ii} = 1 - \sum_{i \neq j} M_{ij}$, which corresponds to the matrix of a discrete-time Markov chain describing the jumps the Markov chain makes without taking into account how long it takes to do such jumps. For calculations such as the probability of being absorbed in a particular state, we can use the jump matrix and theory from discrete-time absorbing Markov chains. In particular, we can calculate $\mathbb{P}(T_{D_W} < \infty)$ from the fundamental matrix of the jump matrix created from Q.

Conditional absorption Markov chain theory. To calculate $\mathbb{P}(T_{D_W} \leq \tau | T_{D_W} < \infty)$, we need to eliminate the absorbing states not in D_W from the chain, which we denote by S^-. The dynamics of those runs that do not end in S^- are Markovian, and its rate transition matrix can be found according to the following proposition.

PROPOSITION 1. *Given a Markov chain with two sets of absorbing states S^+ and S^- and a rate transition matrix Q, one can construct a matrix Q^+ corresponding to the Markovian dynamics of those processes that get absorbed at S^+ (which we can write as $X_\infty \in S^+$) as follows:*

$$Q^+_{ij} = \begin{cases} Q_{ij} \dfrac{\mathbb{P}(X_\infty \in S^+ | X(0) = j)}{\mathbb{P}(X_\infty \in S^+ | X(0) = i)}, & \mathbb{P}(X_\infty \in S^+ | X(0) = i) \neq 0, \\ 0, & \mathbb{P}(X_\infty \in S^+ | X(0) = i) = 0. \end{cases} \tag{A.1}$$

Furthermore, the initial probability distribution $\{\mathbb{P}(X(0) = i)\}_i$ also gets modified:

$$\mathbb{P}(X(0) = i | X_\infty \in S^+) = \frac{\mathbb{P}(X_\infty \in S^+ | X(0) = i)\mathbb{P}(X(0) = i)}{\sum_j \mathbb{P}(X_\infty \in S^+ | X(0) = j)\mathbb{P}(X(0) = j)}. \tag{A.2}$$

Once we have a Markov chain with the only set of absorbing states S^+ (and therefore $\mathbb{P}(X_\infty \in S^+) = 1$), the distribution of stopping times (usually called hitting times in the context of Markov chains) to S^+ follows a phase-type distribution. If we collapse all the states of S^+ into one (by adding the probability rates that reach it), the hitting times are unchanged and the rate matrix of the Markov chain with N_t transient states (N_t depends on the choice of compartment model and n) takes the form

$$Q' = \begin{pmatrix} 0 & 0 \\ S^0 & S \end{pmatrix},$$

where S is an $N_t \times N_t$ matrix and S_0 is equal to $-S\vec{1}$, where $\vec{1}$ is the column vector of all ones. The time it takes to be absorbed in state 0 starting from a vector of initial probabilities $\vec{\alpha}$ is distributed according to the distribution function $F(t) = \mathbb{P}(T_{S^+} \leq t) = 1 - \vec{\alpha}\exp(St)\vec{1}$, where $\exp(\cdot)$ denotes the matrix exponential. The expected value is $-\vec{\alpha}S^{-1}\vec{1}$.

The full equation reads as $\mathbb{P}(T_{D_W} \leq \tau) = \mathbb{P}(T_{D_W} < \infty)\mathbb{P}(T_{D_W} \leq \tau | T_{D_W} < \infty) = \mathbb{P}(T_{D_W} < \infty)(1 - \vec{\alpha}\exp(St)\vec{1})$, where S is obtained by the process of first conditioning and then collapsing the absorbing states and α also comes from conditioning D and then collapsing the states in D_W.

However, this is infeasible in practice, as S scales roughly as Q, which is $s^n \times s^n$.

A.2. Estimation of the state of the disease. We now continue to solve analytically the rest of the tasks. **Q2A** and **Q2B** ask about patient-zero posterior probabilities and outbreak-time estimation.

We define a state of the Markov chain to be *compatible* with our observation if in that state the tested nodes are in a state which agrees with the tests. These include all possibilities for nontested nodes but may include some variations in tested nodes if the tests do not perfectly distinguish all states. We let $\mathcal{C} \subset D_W \subset \mathcal{S}$ be the set of compatible states and O denote our observation.

For **Q2A**, we can apply Bayes' theorem,

$$\mathbb{P}(X_0 = x|O) = \frac{\mathbb{P}(O|X_0 = x)\mathbb{P}(X_0 = x)}{\mathbb{P}(O)} \propto \mathbb{P}(O|X_0 = x)\mathbb{P}(X_0 = x), \tag{A.3}$$

as $\mathbb{P}(O)$ is just a constant that ensures $\sum_{i\in\mathcal{S}} \mathbb{P}(X_0 = i|O) = 1$. We know $\mathbb{P}(X_0 = x)$, as the initial distribution D is known. For $\mathbb{P}(O|X_0 = x)$, we again consider all states of the detection set of the placed tests as absorbing, and we need to sum the probabilities of getting absorbed to exactly those states in the detection set which are compatible with our observation. Therefore,

$$\mathbb{P}(O|X_0 = x) = \sum_{\alpha\in\mathcal{C}} \mathbb{P}(X_\infty = \alpha|X_0 = x). \tag{A.4}$$

The probability of ending in a specific absorbing state starting from a specific transient state can be found with the fundamental matrix of the jump Markov chain matrix. **Q2B** asks about the distribution of time since the epidemic began. To calculate the distribution function $\mathbb{P}(t \le k|O)$ conditioned on the absorption happening on a compatible state, we can do exactly the same as we have done to calculate $\mathbb{P}\left(T_{D_W} \le \tau \middle| T_{D_W} < \infty\right)$, except for replacing D_W for its subset C. **Q2C** asks about the probability distributions of the current state over the states in D_W. This is calculated using the same idea as **Q2A**. We denote by X the actual state

$$\mathbb{P}(X = x|O) = \frac{\mathbb{P}(O|X = x)\mathbb{P}(X = x)}{\mathbb{P}(O)} \propto \mathbb{P}(O|X = x)\mathbb{P}(X = x) = \mathbb{I}(x \in \mathcal{C})\mathbb{P}(X = x). \tag{A.5}$$

$\mathbb{P}(X = x)$ is the probability of being absorbed at state x, which is known a priori with the fundamental matrix. Therefore, we see that the observation just restricts the probability distribution from D_W to its subset $\mathcal{C}$. We can now solve for the probability of node i being in state s_l by summing over the posterior probabilities of all states in which i is in s_l.

A.3. Dynamic allocation for epidemic tracking. Note that if we perfectly know $P(X = x|O)$, then we are able to evaluate the expression for $H(X|O, T_{W'})$ since if $\mathcal{C}_i \subset \mathcal{C}$ are the compatible states with test output t_i, $\mathbb{P}(T_{W'} = t_i) = \sum_{\alpha\in\mathcal{C}_i} \mathbb{P}(\alpha|O)$, and similarly as before,

$$\mathbb{P}(X = x|O, T_{W'} = t_i) \propto \mathbb{P}(T_{W'} = t_i|O, X = x)\mathbb{P}(X = x|O) = \mathbb{I}(x \in \mathcal{C}_i)\mathbb{P}(X = x|O). \tag{A.6}$$

We can now evaluate the mutual information for all tests to obtain the best one. Therefore, in this case we can perform the following iterative procedure:

- At $t = t_i$, we test nodes according to the entropy criterion. After the tests, we update the distributions conditioned on test results using Bayes' theorem. Let D_i be the distribution over nodes conditioned on the test results.
- Run the Markov chain analytically until $t = t_{i+1} - t_i$ with initial distribution D_i.
- Calculate the state distributions at time t_{i+1}.
- Decide which nodes to test at time t_{i+1} according to the criteria in Q5, and calculate D_{i+1} with the obtained results.

Appendix B. Proofs of Theorem 3.1, Theorem 3.4, Theorem 3.5, and Theorem 3.6.

B.1. Proof of Theorem 3.1.

THEOREM B.1 (submodularity of f_τ). *$f_\tau : W \mapsto \mathbb{P}\left(T_{D_W} \leq \tau\right)$ is a submodular function.*

LEMMA B.2. *Let S be a finite set, and consider M a continuous time stochastic process over the states of S. Then, for $\tau \in \mathbb{R}_+$ the function $h(W) : W \mapsto \mathbb{P}\left(T_W \leq \tau\right)$, where $W \in \mathcal{P}(S)$, is submodular.*

Proof. We want to see that for $X \subset Y$ and $x \in S \setminus Y$, $\mathbb{P}(T_{X\cup\{x\}} \leq \tau) - \mathbb{P}(T_X \leq \tau) \geq \mathbb{P}(T_{Y\cup\{x\}} \leq \tau) - \mathbb{P}(T_Y \leq \tau)$. For $Z \subset S$, $\mathbb{P}(T_{Z\cup\{x\}} \leq \tau) - \mathbb{P}(T_Z \leq \tau) = \mathbb{P}(T_{\{x\}} \leq \tau \cap T_Z > \tau)$. But since $X \subset Y$, $T_Y \geq \tau \implies T_X \geq \tau$, and so $\mathbb{P}(T_{\{x\}} \leq \tau \cap T_X \geq \tau) \geq \mathbb{P}(T_{\{x\}} \leq \tau \cap T_Y \geq \tau)$, and we are done. □

The second part of the submodularity proof for f_τ concerns being able to conserve the submodularity of h under composition with functions of certain properties.

LEMMA B.3 (conservation of submodularity under pullback). *Let V, S be finite sets, and let $h : \mathcal{P}(S) \to \mathbb{R}$ be monotone submodular. Let $g : \mathcal{P}(V) \to \mathcal{P}(S)$ be a function satisfying $g(A \cup B) = g(A) \cup g(B)$ and $A \subset B \implies g(A) \subset g(B)$ for A, $B \subset V$. Then, $h \circ g : \mathcal{P}(V) \to \mathbb{R}$ is monotone submodular.*

(Note that by $g(A)$ we do not mean $\{g(x) | x \in A\}$ but rather the image under g of A as an element $g(\{A\})$, but we omit the brackets. A is a subset of V but an element of $\mathcal{P}(\mathcal{V})$, and therefore g sends it to a subset of S, an element of $\mathcal{P}(S)$.)

Figure 11 provides a scheme of the situation, in which we have used the explicit notation.

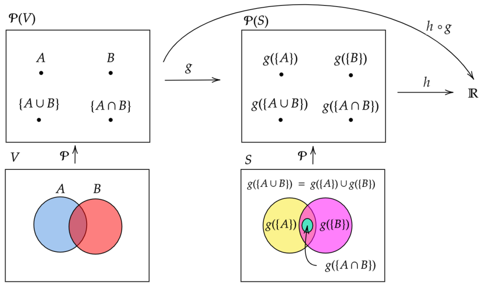

FIG. 11. *Scheme of Lemma B.3. We want to see that given that h is monotone submodular and g satisfies certain conditions, the submodularity is conserved under the composition $h \circ g$.*

Proof. The monotonicity of $h \circ g$ comes from the monotonicity of h and g. For the submodularity, we want to see that for all $S, T \subset V$,

$$\begin{aligned}(h \circ g)(S) + (h \circ g)(T) &\geq (h \circ g)(S \cup T) + (h \circ g)(S \cap T),\\ h(g(S)) + h(g(T)) &\geq h(g(S \cup T)) + h(g(S \cap T)),\\ h(g(S)) + h(g(T)) &\geq h(g(S) \cup g(T))) + h(g(S \cap T)).\end{aligned}$$

Since h is submodular and $g(S)$ and $g(T)$ are subsets of S, we know that

$$h(g(S)) + h(g(T)) \geq h(g(S) \cup g(T))) + h(g(S) \cap g(T)).$$

But since g is monotonous we have $g(S \cap T) \subset (g(S) \cap g(T))$ and since h is monotonous we have $h(g(S \cap T)) \leq h(g(S) \cap g(T))$, and so we have $h(g(S)) + h(g(T)) \geq h(g(S) \cup g(T))) + h(g(S \cap T))$. □

Proof of Theorem 3.1. We apply Lemma B.3 to the composition $h \circ D$, where D is the detection set function $D\colon \mathcal{P}(\mathcal{V}) \to \mathcal{P}(\mathcal{S})$ mapping W to D_W, where V is the set of vertices in the graph and S the set of states of the stochastic process. By Lemma B.2, h is submodular and it is also clearly monotonous by the same argument that we have shown that f_τ is monotonous. By the definition of the detection set function D, $D_A \subset D_B$ if $A \subset B$ and $D_{A \cup B} = D_A \cup D_B$. Therefore, $f = h \circ D$ is submodular. □

B.2. Proof of Theorem 3.4.

THEOREM B.4. *The sample approximation of f_τ, $\hat{f}_\tau$ defined in subsection* 3.1*, is nonnegative, monotone, and submodular for all $N_R \in \mathbb{N}$.*

Proof. Nonnegativity is true by definition, and monotonicity comes from the fact that if $A \subset B$, then $\min_{k \in A} L[i,k] \geq \min_{k \in B} L[i,k]$, and so for a lesser or equal number of runs the minimum over A will be less than or equal to the minimum over B, so $f(A) \leq f(B)$. For submodularity, we want to prove that for $X \subset Y$ and $x \in V \setminus Y$,

$$\hat{f}(X \cup \{x\}) - \hat{f}(X) \geq \hat{f}(Y) - \hat{f}(Y \cup \{x\})\ . \tag{B.1}$$

The left-hand side is

$$|i \in [N_R] \text{ s.t. } \min_{k \in X \cup \{x\}} L[i,k] \leq \tau| - |i \in [N_R] \text{ s.t. } \min_{k \in X} L[i,k] \leq \tau|,$$

which equals $|i \in [N_R] \text{ s.t. } (\min_{k \in X} L[i,k] > \tau) \cap (L[i,x] \leq \tau)|$. Similarly, the right-hand side is $|i \in [N_R] \text{ s.t. } (\min_{k \in Y} L[i,k] > \tau) \cap (L[i,x] \leq \tau)|$. As $\min_{k \in Y} L[i,k] > \tau \implies \min_{k \in X} L[i,k] > \tau$, there are at least as many elements in the set of the left-hand side than in the set of the right-hand side, proving the inequality. □

Taking limits in the submodularity inequality for $\hat{f}_\tau$ provides an alternative proof that f_τ is submodular.

B.3. Proof of Theorem 3.5.

THEOREM B.5. *Let S^* be the optimal set for f_τ. Let $\hat{f}_\tau$ be a nonnegative, monotone, and submodular function such that $(1-\delta) f_\tau \leq \hat{f}_\tau \leq (1+\delta) f_\tau$. Then, using Algorithm* 3.1 *with evaluations of $\hat{f}_\tau$ yields a solution $\hat{S}'$ such that $f_\tau(\hat{S}') \geq \frac{1-\delta}{1+\delta}\left(1 - \frac{1}{e}\right) f_\tau(S^*)$.*

Proof. Let $\hat{S}^*$ be the optimal solution of the optimization of $\hat{f}_\tau$. Then, we have

$$\begin{aligned} f_\tau(\hat{S}') &\geq \frac{1}{1+\delta}\hat{f}_\tau(\hat{S}') \geq \frac{1}{1+\delta}\left(1-\frac{1}{e}\right)\hat{f}_\tau(\hat{S}^*) \\ &\geq \frac{1}{1+\delta}\left(1-\frac{1}{e}\right)\hat{f}_\tau(S^*) \geq \frac{1-\delta}{1+\delta}\left(1-\frac{1}{e}\right)f_\tau(S^*). \end{aligned} \tag{B.2}$$ □

B.4. Proof of Theorem 3.6.

Theorem B.6. *Let S^* be the optimal set for f_τ. Let $\hat{f}_\tau$ be a nonnegative, monotone, and submodular function such that $(1-\delta)f_\tau \leq \hat{f}_\tau \leq (1+\delta)f_\tau$. Then, using Algorithm* 3.2 *with evaluations of $\hat{f}_\tau$ until finding a set with $\hat{f}_\tau(W) \geq P(1-\delta)$ yields a solution $\hat{S}'$ such that $\frac{|\hat{S}'|}{|S^*|} \leq 1 + \log\frac{(1+\delta)f_\tau(V)-(1-\delta)f_\tau(\emptyset)}{(1-\delta)f_\tau(\hat{S}')-(1+\delta)f_\tau(\hat{S}_{-1})}$, where $\hat{S}_{-1}$ is the solution set at the iteration prior to the termination of Algorithm* 3.2 *with $\hat{f}_\tau$.*

Proof. Let $\hat{S}^*$ be the optimal solution of the optimization of $\hat{f}_\tau$. First, we have $\frac{|\hat{S}'|}{|S^*|} = \frac{|\hat{S}'|}{|\hat{S}^*|}\frac{|\hat{S}^*|}{|S^*|}$. Since every W such that $f(W) \geq P$ also satisfies $\hat{f}(W) \geq P(1-\delta)$, we have that $\frac{|\hat{S}^*|}{|S^*|} \leq 1$. Finally, from Theorem 3.3 we have

$$\frac{|\hat{S}'|}{|\hat{S}^*|} \leq 1 + \log\frac{\hat{f}_\tau(V)-\hat{f}_\tau(\emptyset)}{\hat{f}_\tau(\hat{S}')-\hat{f}_\tau(\hat{S_{-1}})} \leq 1 + \log\frac{(1+\delta)f_\tau(V)-(1-\delta)f_\tau(\emptyset)}{(1-\delta)f_\tau(\hat{S}')-(1+\delta)f_\tau(\hat{S}_{-1})}. \tag{B.3}$$ □

REFERENCES

[1] *Hypertext* 2009 *Network Dataset – KONECT*, 2016, http://konect.cc/networks/sociopatterns-hypertext/.

[2] *Test for Current Infection — CDC*; available online from https://www.cdc.gov/coronavirus/2019-ncov/testing/diagnostic-testing.html (accessed on 07/17/2020).

[3] *Timeline of WHO's Response to COVID*-19; available online from https://www.who.int/news-room/detail/29-06-2020-covidtimeline (accessed on 08/12/2021).

[4] *Tracking SARS-CoV*-2 *Variants*; available online from https://www.who.int/en/activities/tracking-SARS-CoV-2-variants/ (accessed on 08/13/2021).

[5] S. P. Borgatti, *Centrality and network flow*, Soc. Netw., 27 (2005), pp. 55–71, https://doi.org/10.1016/j.socnet.2004.11.008.

[6] T. Britton, *Epidemic models on social networks—with inference*, Stat. Neerl., 74 (2020), pp. 222–241.

[7] K. Chaloner and I. Verdinelli, *Bayesian experimental design: A review*, Statist. Sci., 10 (1995), pp. 273–304.

[8] N. A. Christakis and J. H. Fowler, *Social network sensors for early detection of contagious outbreaks*, PLoS ONE, 5 (2010), e12948, https://doi.org/10.1371/journal.pone.0012948.

[9] R. M. Christley, G. L. Pinchbeck, R. G. Bowers, D. Clancy, N. P. French, R. Bennett, and J. Turner, *Infection in social networks: Using network analysis to identify high-risk individuals*, Am. J. Epidemiol., 162 (2005), pp. 1024–1031, https://doi.org/10.1093/aje/kwi308.

[10] L. Ferretti, C. Wymant, M. Kendall, L. Zhao, A. Nurtay, L. Abeler-Dörner, M. Parker, D. Bonsall, and C. Fraser, *Quantifying SARS-CoV*-2 *transmission suggests epidemic control with digital contact tracing*, Science, 368 (2020), eabb6936, https://doi.org/10.1126/science.abb6936.

[11] S. Fujishige, *Submodular Functions and Optimization*, Elsevier, Amsterdam, 2005.

[12] M. Garcia-Herranz, E. Moro, M. Cebrian, N. A. Christakis, and J. H. Fowler, *Using friends as sensors to detect global-scale contagious outbreaks*, PLoS ONE, 9 (2014), e92413, https://doi.org/10.1371/journal.pone.0092413.

[13] J. L. Herrera, R. Srinivasan, J. S. Brownstein, A. P. Galvani, and L. A. Meyers, *Disease surveillance on complex social networks*, PLoS Comput. Biol., 12 (2016), e1004928, https://doi.org/10.1371/journal.pcbi.1004928.

[14] X. Huan and Y. M. Marzouk, *Sequential Bayesian Optimal Experimental Design via Approximate Dynamic Programming*, preprint, https://arxiv.org/abs/1604.08320, 2016.

[15] L. Isella, J. Stehlé, A. Barrat, C. Cattuto, J.-F. Pinton, and W. V. den Broeck, *What's in a crowd? Analysis of face-to-face behavioral networks*, J. Theor. Biol., 271 (2011), pp. 166–180, https://doi.org/10.1016/j.jtbi.2010.11.033.

[16] R. Iyer, N. Khargonkar, J. Bilmes, and H. Asnani, *Submodular Combinatorial Information Measures with Applications in Machine Learning*, preprint, https://arxiv.org/abs/2006.15412, 2020.

[17] W. O. Kermack and A. G. McKendrick, *A contribution to the mathematical theory of epidemics*, Proc. Roy. Soc. London Ser. A, 115 (1927), pp. 700–721, https://doi.org/10.1098/rspa.1927.0118.

[18] I. Z. Kiss, J. C. Miller, and P. L. Simon, *Mathematics of Epidemics on Networks*, Springer, Cham, 2017, https://doi.org/10.1007/978-3-319-50806-1.

[19] M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley, and H. A. Makse, *Identification of influential spreaders in complex networks*, Nat. Phys., 6 (2010), pp. 888–893, https://doi.org/10.1038/nphys1746.

[20] A. Krause, A. Singh, and C. Guestrin, *Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies*, J. Mach. Learn. Res., 9 (2008), pp. 235–284.

[21] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance, *Cost-effective outbreak detection in networks*, in Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '07), 2007, https://doi.org/10.1145/1281192.1281239.

[22] L. A. Meyers, *Contact network epidemiology: Bond percolation applied to infectious disease prediction and control*, Bull. Amer. Math. Soc. (N.S.), 44 (2006), pp. 63–87, https://doi.org/10.1090/s0273-0979-06-01148-7.

[23] P. V. Mieghem, J. Omic, and R. Kooij, *Virus spread in networks*, IEEE ACM Trans. Netw., 17 (2009), pp. 1–14, https://doi.org/10.1109/tnet.2008.925623.

[24] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, *An analysis of approximations for maximizing submodular set functions*—I, Math. Programming, 14 (1978), pp. 265–294, https://doi.org/10.1007/bf01588971.

[25] M. E. J. Newman, *Exact Solutions of Epidemic Models on Networks*, Working Papers 01-12-073, Santa Fe Institute, Santa Fe, NM, 2001, https://ideas.repec.org/p/wop/safiwp/01-12-073.html.

[26] C. Nowzari, V. M. Preciado, and G. J. Pappas, *Analysis and Control of Epidemics: A Survey of Spreading Processes on Complex Networks*, preprint, https://arxiv.org/abs/1505.00768, 2015.

[27] M. Salathe, M. Kazandjieva, J. W. Lee, P. Levis, M. W. Feldman, and J. H. Jones, *A high-resolution human contact network for infectious disease transmission*, Proc. Natl. Acad. Sci. USA, 107 (2010), pp. 22020–22025, https://doi.org/10.1073/pnas.1009094108.

[28] D. Shah and T. Zaman, *Rumors in a network: Who's the culprit?*, IEEE Trans. Inform. Theory, 57 (2011), pp. 5163–5181, https://doi.org/10.1109/tit.2011.2158885.

[29] B. Spinelli, L. E. Celis, and P. Thiran, *A general framework for sensor placement in source localization*, IEEE Trans. Netw. Sci. Eng., 6 (2019), pp. 86–102, https://doi.org/10.1109/tnse.2017.2787551.

[30] L. Sun, K. W. Axhausen, D.-H. Lee, and M. Cebrian, *Efficient detection of contagious outbreaks in massive metropolitan encounter networks*, Sci. Rep., 4 (2014), 5099, https://doi.org/10.1038/srep05099.

[31] L. A. Wolsey, *An analysis of the greedy algorithm for the submodular set covering problem*, Combinatorica, 2 (1982), pp. 385–393, https://doi.org/10.1007/bf02579435.

[32] W. Yan, P.-L. Loh, C. Li, Y. Huang, and L. Yang, *Conquering the worst case of infections in networks*, IEEE Access, 8 (2020), pp. 2835–2846, https://doi.org/10.1109/access.2019.2962197.