Curing with the Network Polya Contagion Model

Mikhail Hayhoe* Fady Alajaji† Bahman Gharesifard†

***Abstract*— We investigate the curing of epidemics using a model based on the classical Polya urn scheme that takes into account spatial contagion among neighboring nodes. We define several quantities to measure the infection in the network, and use them to formulate an optimal control problem. We prove that this problem is feasible under high curing budgets by deriving conservative lower bounds that turn our measures of network infection into supermartingales. To handle the allocation of curing resources under fixed budget constraints, we provide a provably convergent gradient descent algorithm. Motivated by the computational complexity of this algorithm, we design a heuristic method that is locally implementable and nearly as effective. A suite of simulation results executed on a large-scale real-world social network demonstrate the performance of all proposed strategies.**

I. Introduction

In this paper we examine the problem of curing an epidemic using a network contagion model adapted from the Polya process [1], [2]. Here an epidemic can represent a disease [3], a computer virus [4], the spread of an innovation, rumour or idea [5], or the dynamics of competing opinions in a social network [6].

Epidemics on networks have been intensively studied in recent years [7], [8]. Our model is similar to the well-known susceptible-infected-susceptible (SIS) compartmental infection model [9], and is motivated by the classical Polya contagion process [10], [11], [12]. The classical process has been used to study a variety of epidemics such as the bubonic plague in Peru [13] and the spread of chlamydia in a closed population [14], and hence it is natural to extend this model to a network setting. In the network Polya contagion model, introduced in [1], each node of the underlying network is equipped with an individual urn; however, instead of sampling from these urns when generating its contagion process, each node has a "super urn", created by combining the contents of its own urn with those of its neighbors' urns. This adaptation captures the concept of spatial infection, since having infected neighbors increases the chance that an individual is infected in the future.

The stochastic properties of this model, along with a comparison to the traditional discrete-time SIS model, were examined in [2]. Comparatively, the network Polya contagion process runs at a microscopic level to better model the behavior of the contagion process mathematically. Indeed with finite memory our process becomes Markovian and exhibits similar behavior to the discrete-time SIS model, but in the case of infinite memory our model exhibits non-Markovian characteristics which may be truer to some real-life epidemics. In both cases, the microscopic viewpoint allows us to capture the behavior of the process, and the stochastic analysis of this behavior, in real time. With the explicit formulae provided for the network Polya contagion process, the joint and marginal probabilities of infection can be calculated exactly for any time without the need for mean-field approximations or other moment closure techniques, as is typical for SIS models [15].

In this work, we study the problem of controlling a contagion spread under this model. More specifically, we propose various natural ways to measure the total infection in the network Polya contagion model, and examine conditions under which these measures have limits as time grows without bound. Using these measures, we pose an optimal control problem within the context of the network Polya contagion model. As our first contribution, we characterize lower bounds on the allocation of curing to individual nodes which turn these infection measures into supermartingales. Our result hence provides a conservative strategy for curing network epidemics. We next focus on realistic scenarios, where the curing budget is constrained. As our next contribution, we prove that the constrained gradient flow method is convergent for this problem and hence can be employed to find near-optimal strategies under a fixed curing budget at each time step. In spite of its effectiveness, as we demonstrate, the gradient flow strategy is computationally expensive and only implementable in a centralized manner. These shortcomings motivates us to look into alternative strategies, which take advantage of notions of node centrality in the underlying network along with the composition of super urns at each time step. These strategies are less expensive computationally and can be implemented in a decentralized manner. Through extensive simulation results, we show that our proposed heuristic strategies perform well in curing epidemics.

II. Preliminaries

For a sequence $v_i = (v_{i,1}, \ldots, v_{i,n})$, we use the notation $v_{i,s}^t$ with $1 \leq s < t \leq n$ to denote the vector $(v_{i,s}, v_{i,s+1}, \ldots, v_{i,t})$, with $v_{i,0}^t = v_i^t$. Our technical results rely on notions from stochastic processes, some of which we recall here. Throughout, we assume that the reader is familiar with basic notions of probability theory.

Let $(\Omega, \mathcal{F}, P)$ be a probability space, and consider the stochastic process $\{Z_n\}_{n=1}^{\infty}$, where each Z_n is a random variable on Ω. We often refer to the indices of the process as "time" indices. We recall that the process $\{Z_n\}_{n=1}^{\infty}$ is *stationary* if for any $n \in \mathbb{Z}_{\geq 1}$, its n-fold joint probability distribution (i.e., the distribution of $(Z_1, \ldots, Z_n)$) is invari-

*Department of Electrical and Systems Engineering at the University of Pennsylvania, mhayhoe@seas.upenn.edu.

†Department of Mathematics & Statistics at Queen's University, Kingston, Ontario, Canada, {fady, bahman}@mast.queensu.ca.

This work was partially supported by the Natural Sciences and Engineering Research Council of Canada.

ant to time shifts. Lastly, the process $\{Z_n\}_{n=1}^{\infty}$ is called a *martingale* (resp. *supermartingale*, *submartingale*) with respect to the filtration $\{\mathcal{F}_n\}_{n=1}^{\infty}$ if $E[|Z_n|] < \infty$ and $E[Z_{n+1}|\mathcal{F}_n] = Z_n$ almost surely (resp. less than or equal to, greater than or equal to), for all n. If the inequality is strict, we call the process a *strict supermartingale* or *strict submartingale*. Doob's martingale convergence theorem [16] can then be used to show that $\{Z_n\}_{n=1}^{\infty}$ will have a limit as n grows without bound. Precise definitions of all notions can be found in standard texts (e.g., [16], [17]).

We now recall the classical version of the Polya contagion process [10], [12], which has been applied in many different contexts, including the modelling of contagious diseases [13], [14], communication channels with memory [18], image segmentation [19], as well as biology, statistics and other areas (see [20]). Consider an urn with $R \in \mathbb{Z}_{>0}$ red balls and $B \in \mathbb{Z}_{>0}$ black balls. We denote the total number of balls by T, i.e., $T = R + B$. At each time step a ball is drawn from the urn, and then returned along with $\Delta > 0$ balls of the same color. To describe this process, we use a replacement matrix M_R:

$$M_R = \begin{bmatrix} \Delta_{\text{red} \mid \text{red drawn}} & \Delta_{\text{red} \mid \text{black drawn}} \\ \Delta_{\text{black} \mid \text{red drawn}} & \Delta_{\text{black} \mid \text{black drawn}} \end{bmatrix} = \begin{bmatrix} \Delta & 0 \\ 0 & \Delta \end{bmatrix}.$$

We use an indicator Z_n to denote the color of ball in the nth draw (see Figure 1), so $Z_n = 1$ if we drew a red ball at time n, and 0 otherwise.

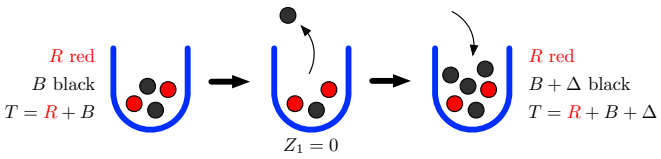


Fig. 1. Illustration of the first draw for a classical Polya process. We drew a black ball and hence $Z_1 = 0$. Here $R = 2$, $B = 2$, and $\Delta = 2$.

Let U_n denote the proportion of red balls in the urn after the nth draw. Then

$$U_n := \frac{R + \Delta \sum_{t=1}^{n} Z_t}{T + n\Delta} = \frac{\rho_c + \delta_c \sum_{t=1}^{n} Z_t}{1 + n\delta_c}$$

where $\rho_c = \frac{R}{T}$ is the initial proportion of red balls in the urn and $\delta_c = \frac{\Delta}{T}$ is a correlation parameter. It can be easily shown that $\{U_n\}_{n=1}^{\infty}$ is a martingale. Lastly, since we draw from this urn at each time step, the conditional probability of drawing red at time n, given $Z^{n-1} = (Z_1, \ldots, Z_{n-1})$, is

$$P(Z_n = 1 \mid Z^{n-1}) = \frac{R + \Delta \sum_{t=1}^{n-1} Z_t}{T + (n-1)\Delta} = U_{n-1}.$$

III. Model Description and Problem Statement

A. Network Polya Contagion Process

In this section we briefly recall the Polya network contagion process, fully described in [2]. Consider an undirected graph $\mathcal{G} = (V, \mathcal{E})$, where $V = \{1, \ldots, N\}$ is the set of $N \in \mathbb{Z}_{\geq 1}$ nodes and $\mathcal{E} \subset V \times V$ is the set of edges. We assume that $\mathcal{G}$ is connected, i.e. there is a path between any two nodes in $\mathcal{G}$. We use $\mathcal{N}_i$ to denote the set of nodes that are neighbors to node i, that is $\mathcal{N}_i = \{v \in V : (i, v) \in \mathcal{E}\}$, and $\mathcal{N}_i' = \{i\} \cup \mathcal{N}_i$. Each node $i \in V$ is equipped with an urn, initially with $R_i \in \mathbb{Z}_{>0}$ red balls and $B_i \in \mathbb{Z}_{>0}$ black balls (we do not let $R_i = 0$ or $B_i = 0$ to avoid any degenerate cases). In the context of epidemics, the red and black balls in an urn, respectively, represent "infection" and "healthiness"; a complete description of this relationship can be found in [2]. We let $T_i = R_i + B_i$ be the total number of balls in the ith urn, $i \in \{1, \ldots, N\}$. We use $Z_{i,n}$ as an indicator for the ball drawn for node i at time n, so $Z_{i,n} = 1$ if the nth draw for node i is red, and 0 otherwise. Thus we define the network contagion process as $\{Z_n\}_{n=1}^{\infty}$, where $Z_n = (Z_{i,n}, \ldots, Z_{N,n})$. However, instead of drawing solely from its own urn, each node has a "super urn" created by combining all the balls in its own urn with those in its neighbors' urns; see Figure 2. This allows the spatial relationships between nodes to influence their state. Thus $Z_{i,n}$ is the indicator for a ball drawn from node i's super urn, and not its individual urn. Hence, the super urn of node i initially has $\bar{R}_i = \sum_{j \in \mathcal{N}_i'} R_j$ red balls, $\bar{B}_i = \sum_{j \in \mathcal{N}_i'} B_j$ black balls, $\bar{T}_i = \sum_{j \in \mathcal{N}_i'} T_j$ balls in total, and the network-wide initial proportion of red balls is $\rho = \frac{\sum_{i=1}^{N} R_i}{\sum_{i=1}^{N} T_i}$.

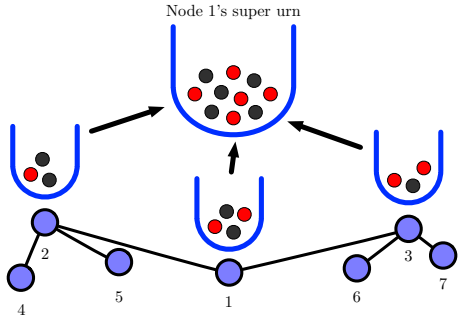


Fig. 2. Illustration of a super urn in a network.

We allow the number of added balls to vary based on the colour drawn, the time index, and the node for which it was drawn; hence, the replacement matrix for node i at time t is

$$M_{R,i}(t) = \begin{bmatrix} \Delta_{r,i}(t) & 0 \\ 0 & \Delta_{b,i}(t) \end{bmatrix}.$$

We assume that $\Delta_{r,i}(t) \geq 0$ and $\Delta_{b,i}(t) \geq 0$ for all $t \in \mathbb{Z}_{\geq 1}$, and that there exists $i \in V$ and t such that $\Delta_{r,i}(t) + \Delta_{b,i}(t) \neq 0$; otherwise we are simply sampling with replacement.

To express the proportion of red balls in the individual urns of the nodes, we define the random vector $U_n = (U_{1,n}, \ldots, U_{N,n})$, where $U_{i,n}$ is the proportion of red balls in node i's urn after the nth draw, $i \in V$. For node i,

$$U_{i,n} = \frac{R_i + \sum_{t=1}^{n} \Delta_{r,i}(t) Z_{i,t}}{X_{i,n}},$$

where

$$X_{i,n} = T_i + \sum_{t=1}^{n} \Delta_{r,i}(t) Z_{i,t} + \Delta_{b,i}(t)(1 - Z_{i,t}) \quad (1)$$

is the total number of balls in node i's urn after the nth draw, and the numerator represents the total number of red balls in the same urn. We define the random vector $S_n =$

$(S_{1,n}, \ldots, S_{N,n})$ as the proportion of red balls in the nodes' super urns after the nth draw, i.e., $S_{i,n}$ is the proportion of red balls in node i's super urn after n draws. So, for node i,

$$S_{i,n} = \frac{\bar{R}_i + \sum_{j\in\mathcal{N}_i'}\sum_{t=1}^{n}\Delta_{r,j}(t)Z_{j,t}}{\sum_{j\in\mathcal{N}_i'}X_{j,n}} = \frac{\sum_{j\in\mathcal{N}_i'}U_{j,n}X_{j,n}}{\sum_{j\in\mathcal{N}_i'}X_{j,n}}, \quad (2)$$

where $S_{i,0} = \frac{\bar{R}_i}{T_i}$. In fact, $S_{i,n}$ is a function of the random draw variables of the network, and in particular of $\{Z_j^n\}_{j\in\mathcal{N}_i'}$, but for ease of notation, when the arguments are clear, we write $S_{i,n}(Z_1^n, \cdots, Z_N^n) = S_{i,n}$. Then the conditional probability of drawing a red ball from the super urn of node i at time n given the complete network history, i.e. given all the past $n-1$ draw variables for each node in the network $\{Z_j^{n-1}\}_{j=1}^N = \{(Z_{1,1}, \ldots, Z_{1,n-1}), \ldots, (Z_{N,1}, \ldots, Z_{N,n-1})\}$, satisfies

$$P\left(Z_{i,n}=1|\{Z_j^{n-1}\}_{j=1}^N\right) = \frac{\bar{R}_i + \sum_{j\in\mathcal{N}_i'}\sum_{t=1}^{n-1}\Delta_{r,j}(t)Z_{j,t}}{\sum_{j\in\mathcal{N}_i'}X_{j,n-1}} = S_{i,n-1}. \quad (3)$$

That is, the conditional probability of drawing a red ball for node i given the entire past $\{Z_j^{n-1}\}_{j=1}^N$ is the proportion of red balls in its super urn, $S_{i,n-1}$. Since these random variables fully describe the evolution of the process, we say $\{\mathcal{F}_n\}_{n=1}^\infty$ is the natural filtration on $\{Z_i^{n-1}\}_{i=1}^N$ and by extension $\{U_{i,n}\}_{n=1}^\infty$ and $\{S_{i,n}\}_{n=1}^\infty$, for all $i \in V$. Thus note that in (3) we could have instead conditioned on $\mathcal{F}_{n-1}$.

Using the conditional probability established above, we next determine the n-fold joint probability of the entire network $\mathcal{G}$. Let $a_i^n \in \{0,1\}^n$, $i \in \{1, \ldots, N\}$, and we have

$$\begin{aligned} &P_{\mathcal{G}}^{(n)}(a_1^n, \cdots, a_N^n) \\ &:= P\left(\{Z_i^n = a_i^n\}_{i=1}^N\right) \\ &= \prod_{t=1}^{n}\prod_{i=1}^{N}\left(S_{i,t-1}\right)^{a_{i,t}}\left(1-S_{i,t-1}\right)^{1-a_{i,t}}, \end{aligned} \quad (4)$$

where $S_{i,t} = S_{i,t}(a_1^t, \cdots, a_N^t)$ is defined in (3). The study of the asymptotic behavior of each node's contagion process $\{Z_{i,n}\}_{n=1}^\infty, i \in V$ is established in [1]. Our objective in this work is to demonstrate the implications of these results in designing curing strategies. With the above explicit joint distribution, it is possible to determine the distributions of each node's process. More specifically, using (4), the n-fold distribution of node i's process at time $t \geq n$ is

$$P_{i,t}^{(n)}(a_{i,t-n+1}, \cdots, a_{i,t}) := \sum_{\substack{a_i^{t-n}\in\{0,1\}^{t-n} \\ a_j^t\in\{0,1\}^t, j\neq i}} P_{\mathcal{G}}^{(n)}(a_1^t, \cdots, a_N^t).$$

It can be shown that the draw contagion process $\{Z_{i,n}\}_{n=1}^\infty$ of each individual node i is not stationary in general [1], [2], and hence $\{Z_n\}_{n=1}^\infty$ is not stationary.

In order to measure the spread of contagion in the network at any given time, we wish to see how likely it is, on average, for a node to be infected at that instant. We thus define the *average infection rate* in the network at time n as

$$\tilde{I}_n := \frac{1}{N}\sum_{i=1}^{N}P(Z_{i,n}=1) = \frac{1}{N}\sum_{i=1}^{N}P_{i,n}^{(1)}(1).$$

Note that $\tilde{I}_n$ is a function of the network topology $(V, \mathcal{E})$, the initial placement of balls R_i and B_i, the draw processes $\{Z_{i,t}\}_{t=1}^n$, and number of balls added $\{\Delta_{r,i}(t)\}_{t=1}^n$ and $\{\Delta_{b,i}(t)\}_{t=1}^n$ for each node $i \in V$. Unfortunately for an arbitrary network, the above quantity does not yield an exact analytical formula (except in the simple case of complete networks). As such, in general it is hard to mathematically analyze the asymptotic behavior of $\tilde{I}_n$, which we wish to minimize when attempting to cure an epidemic. Instead we examine the asymptotic stochastic behavior of two closely related variables given by the average individual proportion of red balls at time n, namely

$$\tilde{U}_n := \frac{1}{N}\sum_{i=1}^{N}U_{i,n},$$

which we call the *network susceptibility*, and the average neighborhood proportion of red balls at time n,

$$\tilde{S}_n := \frac{1}{N}\sum_{i=1}^{N}S_{i,n},$$

which we call the *network exposure*. Through (2) we see that if $U_{i,n}$ increases then this node-specific value causes $S_{j,n}$ to increase for every neighbor j of node i, and hence by (3) their conditional probabilities of drawing red balls increase. More specifically,

$$\begin{aligned} \uparrow U_{i,n} &\overset{(2)}{\Longrightarrow} \uparrow S_{j,n} \text{ for all } j \in \mathcal{N}_i' \\ &\overset{(3)}{\Longrightarrow} \uparrow P\left(Z_{i,n+1}=1|\{Z_j^n\}_{j=1}^N\right) \text{ for all } j \in \mathcal{N}_i'. \end{aligned} \quad (5)$$

Thus if $\tilde{U}_n$ is high, then this average measure of individual nodes implies that the conditional probability of a node being infected is higher on average. Hence $\tilde{U}_n$ can be understood as the average node prevalence of infection. The effect of the network exposure here is more direct, since (3) shows that $\tilde{S}_n$ is in fact the network-wide average of the conditional probabilities of infection, which is a quantity that is intimately related to the state of infection in the neighborhood of node i. Thus $\tilde{S}_n$ represents the average neighborhood prevalence of infection. Note that similarly to $\tilde{I}_n$, both $\tilde{U}_n$ and $\tilde{S}_n$ are functions of the network variables.

With the model in hand, we turn to the exploration of a curing problem. Our objective is to control the average infection rate $\tilde{I}_n$ as n grows without bound; but when seeking analytic results, it might be more amenable to observe the asymptotic behavior of the network exposure $\tilde{S}_n$ (note that when $\tilde{S}_n$ decreases, $\tilde{I}_n$ tends to do the same, as seen in (5)).

B. Establishing a Control Problem

The quantities $\{\Delta_{b,i}(n)\}_{n=1}^\infty$, which denote the net number of "healthy" balls added to node i's urn after each draw, can play the role of "healing or curing parameters". Our

objective is to show that when these parameters are appropriately selected, one can steer the average infection rate towards a desirable level; the selection of curing parameters is, however, subject to an allowable budget on the maximal number of healthy balls that can be added in the network. Let us state this problem formally.

Problem 3.1. (Average Infection Rate Budget Constraint): *Minimize the limiting average infection rate $\tilde{I}_t$ subject to a budget $\mathcal{B}$ on the total healing at each time step:*

$$\min_{\substack{\sum_{i=1}^{N} \Delta_{b,i}(t) \leq \mathcal{B} \\ \forall t}} \limsup_{t \to \infty} \tilde{I}_t$$

Such optimal curing problems have been studied in many different contexts [21], [22]. For our model, the solution to Problem 3.1 would be an infinite horizon optimal control policy that would yield the best possible level of epidemic elimination, given the initial data. Finding such a policy is in general difficult. Nevertheless, as we demonstrate in the upcoming sections, one can obtain interesting analytical results regarding the feasibility of this problem, and design algorithmic strategies to curtail the average infection rate.

IV. Controlling Epidemics: Analytical Results

In order to determine when Problem 3.1 makes sense, we wish to examine when a limit exists. As stated earlier, working with $\tilde{I}_n$ can be difficult; we instead focus on the related measures of the network susceptibility $\tilde{U}_n$ and network exposure $\tilde{S}_n$. Our results will show how one can force these measures to form supermartingales by appropriately selecting the curing policies $\{\Delta_{b,i}(n)\}_{n=1}^{\infty}$, for all $i \in V$. In conjunction with Doob's martingale convergence theorem [16], these results show that $\{U_{i,n}\}_{n=1}^{\infty}$, $\{S_{i,n}\}_{n=1}^{\infty}$, and hence both $\{\tilde{U}_n\}_{n=1}^{\infty}$ and $\{\tilde{S}_n\}_{n=1}^{\infty}$, have limits. While the results presented do not obey the per-step budget constraint on the curing, these results in conjunction with the simulations presented later show that strategies that fit within the framework of Problem 3.1 exist that reduce $\tilde{I}_n$ on average.

An important assumption used herein is that the number of red balls to be added $\Delta_{r,i}(n)$ is known at least one step ahead of time, so that in particular $\Delta_{r,i}(n)$ is almost surely constant given $\mathcal{F}_{n-1}$. A sufficient, but not necessary, condition to satisfy this assumption is for $\{\Delta_{r,i}(n)\}_{n=1}^{\infty}$ to be set, for all $i \in V$, before the process begins. Using [2, Theorem 4.6], we have the following result. Throughout this section, proofs are omitted due to space limitations.

Theorem 4.1. (Individual Urn Proportion Categories): *In a general network $\mathcal{G} = (V, \mathcal{E})$, if we choose $\{\Delta_{b,i}(n)\}_{n=1}^{\infty}$ so that*

$$\Delta_{b,i}(n) \geq \frac{\Delta_{r,i}(n)(1 - U_{i,n-1})S_{i,n-1}}{U_{i,n-1}(1 - S_{i,n-1})}$$

almost surely for all $n \in \mathbb{Z}_{\geq 1}$ and $i \in V$ (resp. equal to, less than or equal to) then $\{U_{i,n}\}_{n=1}^{\infty}$ is a supermartingale (resp. martingale, submartingale) with respect to the natural filtration $\{\mathcal{F}_n\}_{n=1}^{\infty}$, i.e.,

$$E[U_{i,n}|\mathcal{F}_{n-1}] \leq U_{i,n-1} \qquad \text{almost surely } \forall n \in \mathbb{Z}_{\geq 1}.$$

Corollary 4.2. (Network Susceptibility Supermartingale): *In a general network $\mathcal{G} = (V, \mathcal{E})$, if the curing policies $\{\Delta_{b,i}(t)\}_{t=1}^{\infty}$ obey the bound*

$$\Delta_{b,i}(n) \geq \frac{\Delta_{r,i}(n)(1 - U_{i,n-1})S_{i,n-1}}{U_{i,n-1}(1 - S_{i,n-1})}$$

almost surely for all nodes $i \in V$, then the network susceptibility $\{\tilde{U}_n\}_{n=1}^{\infty}$, where $\tilde{U}_n = \frac{1}{N}\sum_{i=1}^{N} U_{i,n}$, is a supermartingale with respect to the natural filtration $\{\mathcal{F}_n\}_{n=1}^{\infty}$.

While Corollary 4.2 is useful, by (5) we know the network exposure $\tilde{S}_n$ is more closely related to the average infection rate $\tilde{I}_n$ than the network susceptibility $\tilde{U}_n$. It is with this in mind that we show the next results, which give us sufficient conditions for $\{S_{i,n}\}_{n=1}^{\infty}$ and $\{\tilde{S}_n\}_{n=1}^{\infty}$ to be supermartingales.

Theorem 4.3. (Super Urn Proportion Supermartingale): *In a general network $\mathcal{G} = (V, \mathcal{E})$, if the curing policy $\{\Delta_{b,i}(t)\}_{t=1}^{\infty}$ obeys the bound*

$$\Delta_{b,i}(n) > \Delta_{r,i}(n)\frac{S_{i,n-1}}{1 - S_{i,n-1}} \max_{k \text{ s.t. } i \in \mathcal{N}_k'} \frac{1 - S_{k,n-1}}{S_{k,n-1}}$$

almost surely for all nodes $i \in V$, then the neighborhood proportions of red balls $\{S_{i,n}\}_{n=1}^{\infty}$ are strict supermartingales with respect to the natural filtration $\{\mathcal{F}_n\}_{n=1}^{\infty}$, i.e.

$$E[S_{i,n}|\mathcal{F}_{n-1}] < S_{i,n-1} \qquad \text{almost surely } \forall i \in V, n \in \mathbb{Z}_{\geq 1}.$$

Corollary 4.4. (Network Exposure Supermartingale): *In a general network $\mathcal{G} = (V, \mathcal{E})$, if the curing policies $\{\Delta_{b,i}(t)\}_{t=1}^{\infty}$ obey the bound*

$$\Delta_{b,i}(n) > \Delta_{r,i}(n)\frac{S_{i,n-1}}{1 - S_{i,n-1}} \max_{k \text{ s.t. } i \in \mathcal{N}_k'} \frac{1 - S_{k,n-1}}{S_{k,n-1}}$$

almost surely for all nodes $i \in V$, then the network exposure $\{\tilde{S}_n\}_{n=1}^{\infty}$, where $\tilde{S}_n = \frac{1}{N}\sum_{i=1}^{N} S_{i,n}$, is a strict supermartingale with respect to the natural filtration $\{\mathcal{F}_n\}_{n=1}^{\infty}$.

It is important to note that the policy for $\{\Delta_{b,i}(t)\}_{t=1}^{\infty}$ used in Theorem 4.3 is not a tight lower bound on the curing resources which guarantee that the processes $\{S_{i,n}\}_{n=1}^{\infty}$ are supermartingales, and hence less costly policies may exist that still guarantee this property. In particular, strategies may exist which obey the fixed budget $\mathcal{B}$ on the amount of curing resources used. However, these results motivate the fact that the search for better policies makes sense, since policies exist that will fight the infection and reduce it on average.

V. Controlling Epidemics: Algorithmic Strategies

The supermartingale results established in the previous section demonstrate the feasibility of a relaxed version of Problem 3.1, with no budget limitation. In this section, we establish numerical methods to find control policies that find efficient sub-optimal policies for Problem 3.1, under budget constraints and with having computational complexity in mind. We compare these strategies with the ones obtained from our supermartingale results. A summary of all strategies that will be discussed in this section is given in Table I.

TABLE I

CURING STRATEGIES

(i)	Forcing all $U_{i,n}$ to be supermartingales (Theorem 4.1): $\Delta_{bi,}(t) = \frac{\Delta_{r,i}(n)(1-U_{i,n-1})S_{i,n-1}}{U_{i,n-1}(1-S_{i,n-1})}$
(ii)	Forcing all $S_{i,n}$ to be supermartingales (Theorem 4.3): $\Delta_{b,i}(t) = \Delta_{r,i}(n)\frac{S_{i,n-1}}{1-S_{i,n-1}} \max_{k \text{ s.t. } i\in\mathcal{N}_k'} \frac{1-S_{k,n-1}}{S_{k,n-1}}$
(iii)	Constrained gradient descent algorithm on a simplex: Find $\Delta_{b,i}(t)$ using Algorithm in [23, Chapter 2]
(iv)	Ratio of degree, closeness centrality and super urn proportion: $\Delta_{b,i}(t) = \mathcal{B}\frac{\lvert\mathcal{N}_i\rvert C_i S_{i,t-1}}{\sum_{j=1}^{N}\lvert\mathcal{N}_j\rvert C_j S_{j,t-1}}$
(v)	Uniformly allocate the budget to all nodes in the network: $\Delta_{b,i}(t) = \frac{\mathcal{B}}{N}$

A. Supermartingale Strategies

The supermartingales results in Section IV, specifically, Theorems 4.1 and 4.3, naturally lead to a class of curing strategies. In particular, these strategies guarantee that $\tilde{U}_n$ and $\tilde{S}_n$, respectively, are supermartingales; however, our theoretical results do not necessarily imply that the average infection rate $\tilde{I}_n$ forms a supermartingale. In spite of this, these strategies are still viable options for curing, as far as enough resources are available. We next describe the differences between the strategy given by individual urn proportions, and the one given by super urn proportions.

By Corollary 4.2, we know that strategy (i) guarantees that the network susceptibility $\tilde{U}_n$ will be a supermartingale. Hence we set the curing strategy for each node to force the individual urn proportions of red balls to be supermartingales. As shown in (5), the relationship between the reduction of $\tilde{U}_n$ and $\tilde{I}_n$ is not a strong one and our simulations suggests that this strategy does not appear to offer a large reduction in the average infection rate in general. In contrast, the curing strategy given by Corollary 4.4, where we force the super urn proportions of red balls to be supermartingales for all nodes, performs reasonably well.

B. Gradient Flow Method

In this section, we employ the well-known gradient descent algorithm [23] for Problem 3.1. As discussed earlier, using $\tilde{I}_n$ as a measure of infection is computationally expensive, and hence we focus on the network exposure $\tilde{S}_n$. While our suggested gradient descent algorithm will not provide the exact answer to Problem 3.1 for reducing $\tilde{I}_n$, we will show that it is guaranteed to provide the optimal policy to reduce the closely related measure $E[\tilde{S}_n|\mathcal{F}_{n-1}]$. We hence refer to the gradient descent strategy as a benchmark for optimality.

In Problem 3.1, our curing policy is constrained by a budget $\mathcal{B}$ at each time step and so the feasible set $\mathcal{X}$, or set of valid curing policies, for our gradient descent is all policies which do not exceed $\mathcal{B}$. However, any optimal policy will make use of the whole budget, and so we consider $\mathcal{X} = \left\{\{\Delta_{b,i}(n)\}_{i=1}^N \in \mathbb{R}^N_{\geq 0} \mid \sum_{i=1}^N \Delta_{b,i}(n) = \mathcal{B}\right\}$. Proposition 5.1 shows that for arbitrary initial conditions and network topologies, this problem for $E[\tilde{S}_n|\mathcal{F}_{n-1}]$ is convex.

Proposition 5.1. (Gradient descent conditions are met): *In a general network $\mathcal{G} = (V, \mathcal{E})$ with arbitrary initial conditions, the expected network exposure $E[\tilde{S}_n|\mathcal{F}_{n-1}]$ is convex with respect to the curing parameters $\{\Delta_{b,i}(n)\}_{i=1}^N$ for all n. Furthermore, the feasible set $\mathcal{X} = \left\{\{\Delta_{b,i}(n)\}_{i=1}^N \in \mathbb{R}^N_{\geq 0} \mid \sum_{i=1}^N \Delta_{b,i}(n) = \mathcal{B}\right\}$ is convex and compact.*

The structure of the feasible set $\mathcal{X}$ allows us to employ the simplex constrained gradient descent method, see [23, Chapter 2]. The time complexity of this algorithm is of the order $O(sa)$ at each time step, where s is the stopping time of the gradient descent and $\frac{1}{a}$ is the granularity used to find the limit-minimized step size. While Proposition 5.1 guarantees that the curing policy that this algorithm finds will be optimal for each individual step, it does not guarantee optimality over the entire time horizon. In spite of this, as the simulation results in Figure 3 show, this curing strategy still outperforms all other curing strategies studied in this paper. The downside of the gradient method is that it is computationally expensive to execute and requires intimate knowledge of the state of all nodes, including the entire history of draws and values of the curing parameters. This motivates us to seek other methods which are computationally easier to execute, although they do not perform as well as the gradient descent strategy.

C. Heuristic Strategies

Both sets of strategies identified above come with challenges: the gradient descent is computationally costly, and the supermatingale strategies do not significantly reduce the average infection rate. As a compromise, we design a heuristic strategy to split the budget between all nodes in the network whose time complexity will be of the order $O(1)$, which we call the centrality-infection ratio:

$$\Delta_{b,i}(t) = \mathcal{B}\frac{\lvert\mathcal{N}_i\rvert C_i S_{i,t-1}}{\sum_{j=1}^{N}\lvert\mathcal{N}_j\rvert C_j S_{j,t-1}}.$$

We consider three factors when determining how much curing a node should receive: local impact, topological position, and level of infection. Nodes with higher local impact have more neighbors, and hence any healing they receive is immediately distributed to a larger number of nodes; for this we use the degree, $|\mathcal{N}_i|$. Those with a better topological position are more centrally located within the network, in the sense that the distance from them to all other nodes is smaller. We hence use closeness centrality to measure topological position[1] [24], defined as $C_i := \left(\sum_{j\in V} d(i,j)\right)^{-1}$ for node i, where $d(i,j)$ is the length of the shortest path from node i to node j. Lastly, nodes with a higher level of infection will need more curing resources to become healthy, and so we capture this with the super urn proportion of red balls $S_{i,n}$. From (3), we know that this quantity captures how likely it is for node i to be infected at this time given the history of the process. Thus we give more importance to nodes who are more likely to be infected in order to make them less likely to be infected in the future. This allocation ratio is a generalization of the best heuristic strategy presented in [1], which only used the degree to measure centrality.

[1]While numerous centrality measures were tested, including eigenvalue and Bonacich, closeness provided the best empirical performance.

The advantage of this heuristic strategy is twofold. Not only does it reduce computational time complexity from $O(sa)$ to $O(1)$, it is also somewhat distributed in the sense that it does not require constant information from the entire network. Unlike the gradient descent algorithm, strategy (iv) simply needs to know information about the network topology and the state of infection of each node. Since we assume that our network's graph is constant in time, this topological information is only required initially and can be used thereafter. The only other information required from the network at large is the sum of the super urn ratios $\sum_{i=1}^{N} S_{i,n}$, and hence much less information needs to be communicated through the network for the implementation of this strategy.

Lastly, for comparison reasons we present the uniform curing strategy (v), which splits the budget $\mathcal{B}$ equally to all nodes in the network. We use this strategy as a baseline with no interaction to show the improvement achieved by intelligently assigning the curing resources.

VI. Simulation Results and Discussion

In order to confirm the results of Theorems 4.1 and 4.3, a number of simulations were performed. While these simulations allowed $\Delta_{r,i}$ to vary between nodes, they were constant in time. This was done to simplify the choice of the per-step budget, and does not affect the execution of the simulations themselves. All initial conditions used in the simulations herein, as well as videos displaying the average performance of the curing strategies, are available online[2].

The network shown in Figure 3(a) was generated by using a tool [25] to crawl through 500 posts in a Facebook group. Individuals who created posts or interacted with others' content are represented by nodes, while edges are created if individuals interacted with the post or comment of another (by commenting on the post, or liking the post or comment). The resulting graph has 1,363 nodes and 2,425 edges, and by design represents the topology of a real social network.

We now provide a detailed description of the simulation. The values of R_i, B_i and $\Delta_{r,i}$ were uniformly randomly assigned for each node as integers between 1 and 10, and remained consistent throughout all strategies used and trials performed. Since the values for $\Delta_{r,i}$ were fixed over time, the per-step budget was set at $\mathcal{B} = \sum_{i=1}^{N} \Delta_{r,i}$. With the initial conditions set, a number of trials were performed for each strategy. Each trial was performed by successively drawing balls from super urns for a fixed number of time steps. At time t, we first assigned the curing $\Delta_{b,i}(t)$ based on the strategy selected. Then balls were drawn from each super urn, and $Z_{i,t}$ was set accordingly. Based on what was drawn, we added $\Delta_{r,i}$ red or $\Delta_{b,i}(t)$ black balls into node i's urn, and hence its super urn and those of its neighbors. At the end of each trial the draw variables were saved, and then averaged over all trials to produce the empirical performance of the curing strategy.

Figure 3(b) compares the performance of all strategies described in Section V on a Facebook network. The time horizon considered is short due to the computational complexity of the gradient flow algorithm, but results for a much longer time horizon show that trends between all other strategies are unchanged. The baseline uniform strategy (v) performs the worst, which is to be expected. Although (iii) is only proven to be optimal for the expected network exposure $E[\tilde{S}_n|\mathcal{F}_{n-1}]$, these results are seen to be effective for the average infection rate $\tilde{I}_n$ as well, outperforming all other curing strategies. However, the heuristic strategy (iv) performs similarly while being far less computationally difficult. The supermartingale strategies (i) and (ii) both reduce $\tilde{I}_n$ below the initial average infection rate in the network ρ, but are less effective in doing so than the gradient flow and centrality-infection ratio methods. Strategy (i) sees only an immediate small reduction in $\tilde{I}_n$, while strategy (ii) continuously decreases $\tilde{I}_n$. Hence forcing $\tilde{U}_n$ to be a supermartingale is not enough to guarantee a large reduction in the average infection rate $\tilde{I}_n$, while guaranteeing this property for $\tilde{S}_n$ is enough.

In Figure 3(c) we examine the amount of curing resources used by each strategy. Since strategies (iii), (iv) and (v) all obey a per-step budget constraint their usages are fixed. Both supermartingale strategies, which may use arbitrary amounts of curing resources, initially use a larger amount of curing resources and then reduce their usage. Strategy (i) appears to reduce curing consumption at first but then steadily increase, while strategy (ii) continues to decrease its usage in time. Further, strategy (i) uses much more curing resources than the budget $\mathcal{B}$ initially compared to strategy (ii).

The amount of curing resources wasted by each strategy is displayed in Figure 3(d). Waste is defined as curing resources which were assigned to nodes that did not use them since they displayed "infected" behavior at that time, and is measured as $\sum_{i=1}^{N} \sum_{t=1}^{n} \Delta_{b,i}(t) Z_{i,t}$. We observe a correlation between the amount of resources wasted and curing performance. However, this does not tell the full story. The gradient flow algorithm has several spikes where it wastes more than the centrality-infection ratio (iv), but still achieves superior curing performance. Furthermore, strategy (i) initially wastes less than strategy (ii) even though it uses more curing resources, and it still performs worse with respect to reduction in $\tilde{I}_n$. This suggests that optimal curing strategies do not simply waste less, but also intelligently allocate their curing resources to make the best use of them.

[2]See: http://bit.ly/2szl8PY

VII. Acknowledgements

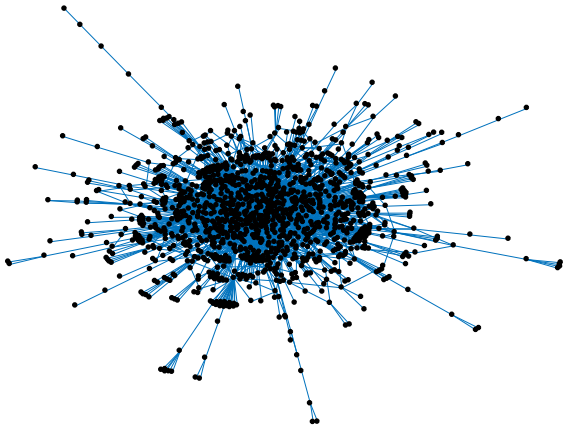
The authors wish to acknowledge the Centre for Advanced Computing at Queen's University, whose computing cluster allowed the simulations presented herein to be performed.

References

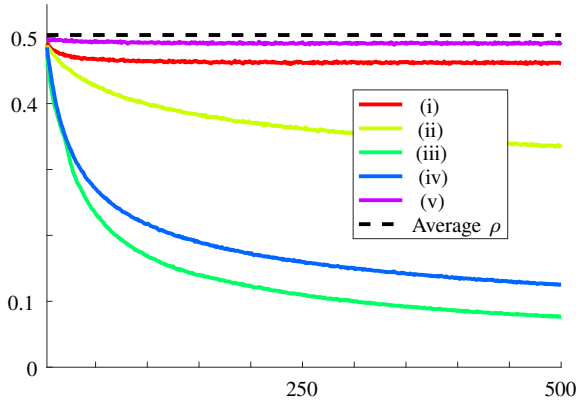
[1] M. Hayhoe, F. Alajaji, and B. Gharesifard, "A Polya urn-based model for epidemics on networks," *Proc. 2017 American Cont. Conf.*, 2017.

[2] M. Hayhoe, F. Alajaji, and B. Gharesifard, "A Polya contagion model for networks," *IEEE Trans. Cont. Netw. Sys., to appear*, 2017.

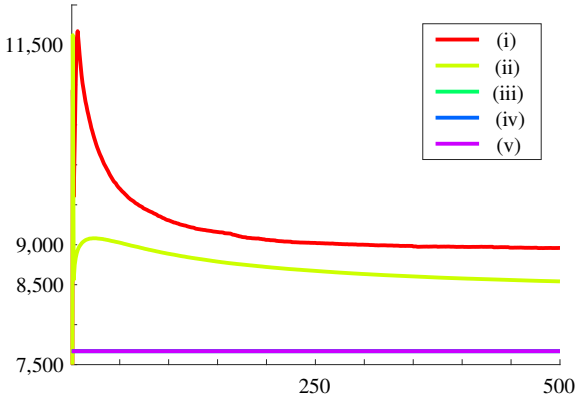
[3] L. Kim, M. Abramson, K. Drakopoulos, S. Kolitz, and A. Ozdaglar, "Estimating social network structure and propagation dynamics for an infectious disease," in *Proc. Int. Conf. Social Computing, Behavioral-Cultural Modeling, and Prediction*, pp. 85–93, Springer, 2014.



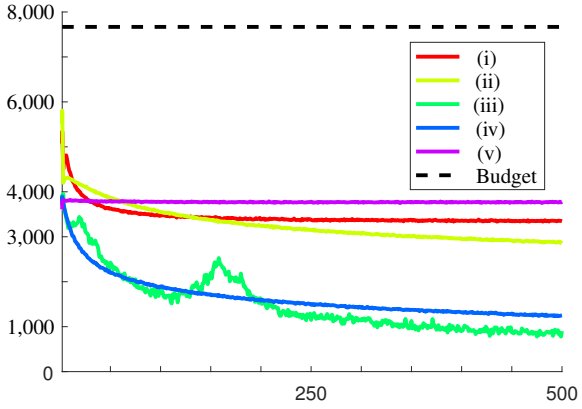
(a) Facebook group network with 1,363 nodes and 2,425 edges.



(b) Plot of empirical average infection rate $\tilde{I}_n$ compared to the initial level of infection $\rho = \frac{\sum_{i=1}^{N} R_i}{\sum_{i=1}^{N} T_i}$ (lower means less infection).



(c) Plot of average usage of curing resources, $\sum_{i=1}^{N} \Delta_{b,i}(t)$ (lower means less curing resources used). Note that strategies (iii)-(v)'s usages are fixed at the per-step budget $\mathcal{B}$, and are overlaid.



(d) Plot of average wasted curing, $\sum_{i=1}^{N} \sum_{t=1}^{n} \Delta_{b,i}(t) Z_{i,t}$ (lower means less curing resources assigned to nodes that did not use them).

Fig. 3. Comparison of all curing strategies presented in Table I. Simulation results were averaged over 250 trials for each strategy, and altogether took approximately 49 hours on 10 cores of an Intel Xeon processor at 2.20GHz. Initial numbers of balls R_i and B_i, and numbers of red balls added $\Delta_{r,i}$ (which remained constant in time), were uniformly randomly assigned for each node but stayed consistent throughout all trials and strategies, while the assignments of $\{\Delta_{b,i}(t)\}_{t=1}^{\infty}$ were different for each strategy. Since the $\Delta_{r,i}$ are all constant, the budget was set as $\mathcal{B} = \sum_{i=1}^{N} \Delta_{r,i}$.

[4] M. Garetto, W. Gong, and D. Towsley, "Modeling malware spreading dynamics," in *Proc. Conf. Comp. Comm.*, vol. 3, pp. 1869–1879, 2003.

[5] E. M. Rogers, *Diffusion of Innovations*. 5 ed., 2003.

[6] E. Adar and L. A. Adamic, "Tracking information epidemics in blogspace," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intelligence*, pp. 207–214, 2005.

[7] P. V. Mieghem, J. Omic, and R. Kooij, "Virus spread in networks," *IEEE/ACM Trans. Netw.*, vol. 17, no. 1, pp. 1–14, 2009.

[8] C. Nowzari, V. M. Preciado, and G. J. Pappas, "Analysis and control of epidemics: A survey of spreading processes on complex networks," *IEEE Control Systems Magazine*, vol. 36, pp. 26–46, 2016.

[9] D. Easley and J. Kleinberg, *Networks, Crowds and Markets: Reasoning about a Highly Connected World*. Cambridge Univ. Press, 2010.

[10] F. Eggenberger and G. Polya, "Über die statistik verketteter vorgänge," *Z. Angew. Math. Mech.*, vol. 3, no. 4, pp. 279–289, 1923.

[11] G. Polya and F. Eggenberger, "Sur l'interprétation de certaines courbes de fréquences," *Comptes Rendus C. R.*, vol. 187, pp. 870–872, 1928.

[12] G. Polya, "Sur quelques points de la théorie des probabilités," *Annales de l'institut Henri Poincaré*, vol. 1, no. 2, pp. 117–161, 1930.

[13] A. Rosenblatt, "Sur le concept de contagion de M. G. Polya dans le calcul des probabilites," *Acad. Nac. de Cien. exactas, Fis. Nat.*, vol. 3, pp. 186–204, 1940.

[14] C. F. Martin, L. J. S. Allen, and M. S. Stamp, "An analysis of the transmission of chlamydia in a closed population," *Jour. Diff. Eqns. Appl.*, vol. 2, pp. 1–29, 1996.

[15] E. Cator and P. V. Mieghem, "Second-order mean-field susceptible-infected-susceptible epidemic threshold," *Physical Review E*, vol. 85, no. 5, 2012.

[16] R. Ash and C. Doléans-Dade, *Probability and Measure Theory*. 2000.

[17] G. R. Grimmett and D. R. Stirzaker, *Probability and Random Processes*. Oxford Univ. Press, 3 ed., 2001.

[18] F. Alajaji and T. Fuja, "A communication channel modeled on contagion," *IEEE Trans. Inf. Theory*, vol. 40, no. 6, pp. 2035–2041, 1994.

[19] A. Banerjee, P. Burlina, and F. Alajaji, "Image segmentation and labeling using the Polya urn model," *IEEE Trans. Image Proc.*, vol. 8, no. 9, pp. 1243–1253, 1999.

[20] R. Pemantle, "A survey of random processes with reinforcement," *Probab. Surveys*, vol. 4, no. 0, pp. 1–79, 2007.

[21] E. Ramírez-Llanos and S. Martínez, "Distributed and robust fair optimization applied to virus diffusion control," *IEEE Trans. Netw. Sci. and Eng.*, vol. 4, pp. 41–54, 2017.

[22] C. Nowzari, V. M. Preciado, and G. J. Pappas, "Optimal resource allocation for control of networked epidemic models," *IEEE Trans. Cont. Netw. Sys.*, vol. 4, no. 2, pp. 159–169, 2017.

[23] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1 ed., 1995.

[24] A. Bavelas, "Communication patterns in task-oriented groups," *Journal of Acoustical Soc. of America*, vol. 22, pp. 725–730, 1950.

[25] B. Rieder, "Studying Facebook via data extraction: the Netvizz application," *Proc. 5th ACM Web Science Conf.*, pp. 346–355, 2013.