Preventing Infectious Disease in
Dynamic Populations Under Uncertainty

Bryan Wilder,[13] **Sze-Chuan Suen,**[23] **Milind Tambe**[13]

[1]Department of Computer Science [2]Department of Industrial and Systems Engineering
[3]Center for Artificial Intelligence in Society
University of Southern California
{bwilder, ssuen, tambe}@usc.edu

Abstract

Treatable infectious diseases are a critical challenge for public health. Outreach campaigns can encourage undiagnosed patients to seek treatment but must be carefully targeted to make the most efficient use of limited resources. We present an algorithm to optimally allocate limited outreach resources among demographic groups in the population. The algorithm uses a novel multiagent model of disease spread which both captures the underlying population dynamics and is amenable to optimization. Our algorithm extends, with provable guarantees, to a stochastic setting where we have only a distribution over parameters such as the contact pattern between agents. We evaluate our algorithm on two instances where this distribution is inferred from real world data: tuberculosis in India and gonorrhea in the United States. Our algorithm produces a policy which is predicted to avert an average of least 8,000 person-years of tuberculosis and 20,000 person-years of gonorrhea annually compared to current policy.

Introduction

Treatable infectious diseases cause hundreds of thousands of cases of disability and death worldwide. Often, this burden is caused by long-term diseases which are continuously present in the population, as opposed to short-term epidemics like influenza. For instance, tuberculosis (TB) deaths in India numbered over 480,000 in 2014 (WHO 2015b), and even developed nations like the U.S. have observed over 395,000 cases of gonorrhea in 2015 (CDC 2015). In both cases, many individuals remain undiagnosed although treatment is available. Outreach efforts to increase screening can lower disease burden; e.g., the Indian government conducts advertising campaigns for TB awareness. Limited resources require these campaigns to be carefully targeted at the most effective groups for reducing disease. Targeting is complicated by changing population dynamics, as individuals age and migrate over time, as well as by uncertainty around disease transmission rates. Officials currently make such decisions by hand as no algorithmic assistance is available.

To remedy this situation, we design an algorithm to divide a limited outreach budget between demographic groups in order to minimize long term disease prevalence under uncertain population dynamics. Our approach contrasts with

existing algorithms for disease control, which often consider disease spread between nodes on a static graph (Saha et al. 2015; Borgs et al. 2010). This is a sensible model of short term disease spread but is less suitable for long-term planning in diseases such as TB or gonorrhea, where people are born, die, age, and move (Luke and Stamatakis 2012). Accounting for changes in the underlying agents is particularly salient for a policymaker who must divide resources between demographic groups over many years to maximize societal long-term health. For instance, India produces 5 year plans to combat TB (RNTCP 2016). Our approach also contrasts with previous work on agent-based disease models (Jindal and Rao 2017; Lee et al. 2010). Such models may include realistic behaviors, but their complexity usually precludes algorithmic approaches to finding the optimal policy in an entire feasible set.

An additional challenge, largely unexplored in previous algorithmic work, is that of uncertainty. Data is always limited; policymakers are never sure of exactly how many people are infected in each group, or of the contact patterns between them. In order to impact real world policy, algorithms for resource allocation must account for such uncertainties.

We introduce a model which both captures underlying agent dynamics and can be solved using an algorithmic approach in a stochastic setting. We make four main contributions. *First*, we present the MCF-SIS model (Multiagent Continuous Flow-SIS) where disease spreads in a multiagent system with birth, death, and movement. The system evolves according to SIS (susceptible-infected-susceptible) dynamics and is stratified across age groups. This introduces a new problem in multiagent systems: computing the optimal resource allocation under MFS-SIS, as in the case where an outreach campaign must decide how to divide limited advertising dollars (or rupees) between the groups.

MCF-SIS introduces a continuous, nonconvex, highly nonlinear optimization problem which cannot be solved by existing methods. Many factors must be accounted for. E.g., between-group disease transmission makes focusing on the groups with the most infected agents suboptimal. Moreover, agents in a targeted group are not cured instantaneously, so, e.g., to reduce prevalence in age group 30, we may need to start targeting resources at age 27. Lastly, we consider a stochastic setting where parts of the model (contact patterns between agents, the number of infected agents in each group,
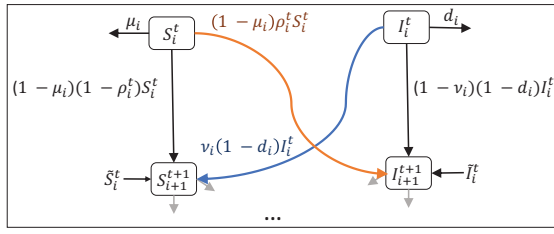
$$\beta = \begin{bmatrix} 0.4 & 0.1 \\ 0.1 & 0.3 \end{bmatrix} \quad \nu = \begin{bmatrix} 0.3 \\ 0.4 \end{bmatrix} \quad d = \begin{bmatrix} 0.2 \\ 0.2 \end{bmatrix} \quad \mu = \begin{bmatrix} 0.05 \\ 1 \end{bmatrix}$$

$$I^0 = \begin{bmatrix} 10 \\ 10 \end{bmatrix} \quad S^0 = \begin{bmatrix} 10,000 \\ 10,000 \end{bmatrix} \quad N^0 = \begin{bmatrix} 10,010 \\ 10,010 \end{bmatrix} \quad M = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \quad \tilde{I}^0 = \begin{bmatrix} 5 \\ 2 \end{bmatrix}$$

$$(1) \; M^{\mathrm{T}} \Big(diag(S^0) \, diag(1-\mu) \, \beta \, diag\Big(\frac{1}{N^0}\Big) + diag(1-\nu)diag(1-d) \Big) = \begin{bmatrix} 0 & 0 \\ 0.94 & 0.09 \end{bmatrix}$$

$$(2) \; I^1 = \begin{bmatrix} 0 & 0 \\ 0.94 & 0.09 \end{bmatrix} I^0 + \tilde{I}^0 = \begin{bmatrix} 5 \\ 12.3 \end{bmatrix}$$

Figure 1: Top: Illustration of the MCF-SIS model. Bottom: a single step in the model with 2 age groups.

etc.) are not known exactly but are drawn from a distribution.

Our *second* contribution shows that optimal allocation in MCF-SIS is a *continuous submodular* problem. This opens up a novel set of optimization techniques which have not previously been used in disease prevention. Continuous submodularity generalizes submodular set functions to continuous domains. Intuitively, infections averted by spending one unit of treatment resources can no longer be averted by additional spending, creating diminishing returns. *Continuous submodularity is deliberately enabled by our modeling choices, in particular our shift from the discrete, graph-based setting common in previous work (Saha et al. 2015; Borgs et al. 2010) to a continuous, population-based model.*

Our *third* contribution is a new algorithm called DOMO (Disease Outreach via Multiagent Optimization), which obtains an efficient $(1 - 1/e)$-approximation to the optimal allocation. Our algorithm builds on a recent theoretical framework for submodular optimization (Bian et al. 2017). DOMO's generalization of this framework to the stochastic setting may be of independent interest.

Our *fourth* contribution is to instantiate MCF-SIS in two domains using empirical data which takes into account behavioral, demographic, and epidemic trends: first, TB spread in India, and second, gonorrhea in the United States. DOMO averts 8,000 annual person-years of TB and 20,000 person-years of gonorrhea compared to current policy.

MCF-SIS: a new modeling approach

The MCF-SIS model has two goals: to enable both realistic population dynamics and efficient optimization. In MCF-SIS, a finite population evolves in discrete time. Each agent has two possible states. In the susceptible (S) state, an agent has not contracted the disease. In the infected (I) state, an agent can transmit the disease to others. They can also be cured and return to the susceptible state.

The population is segmented into n groups. Our running example is where each group is an age range because transmission patterns for infection vary with age (Suen et al. 2015). Figure 1 shows this instantiation of the model. However, our techniques generalize to any segmentation (e.g.,

geographic location or occupation). We denote the number of susceptible agents in each group at time t as the vector S^t where S_i^t is the number of susceptible agents of group i. Likewise, I^t gives the number of infected agents. The total population is $N^t = S^t + I^t$. At each time step, agents move between groups according to a movement matrix M, where M_{ij} is the fraction of agents in group i who move to group j. For instance, when the groups represent age, agents advance from age i to $i + 1$. So, we have $M_{i,i+1} = 1$, $i = 1...n - 1$, and all other entries of M are zero. Agents die of natural causes at rate μ_i. New agents enter the population through birth or migration, given by the vector $\tilde{S}^t$. We also allow for an exogenous inflow of infected agents $\tilde{I}^t$.

Disease spreads through contact with infected agents, described by the matrix β: agents in group i interact with group j with frequency β_{ij}. A fraction $\rho_i^t = \sum_{j=1}^n \beta_{ij} \frac{I_j^t}{N_j^t}$ of group i encounters an infected agent and becomes infected themselves. At each time step, a fraction d_i of infected agents in group i die. Of those who do not die, a fraction ν_i are cured and become susceptible again. ν_i is referred to as the clearance rate, and captures the total rate at which infected agents are diagnosed, enter treatment, and are successfully cured.

While compartmental models like ours do not simulate the micro-level details of individual agents, they can be realistic enough to capture long term trends. Similar models are commonly used in health policy analyses (White et al. 2005; Chan, McCabe, and Fisman 2011; Dowdy et al. 2012).

Interventions: We consider optimal resource allocation modeled through the clearance vector ν. Suppose a policymaker can conduct outreach to selected groups. Since some percentage of people who see an advertisement will enter treatment, we can model the policymaker's decisions as increasing ν_i for the targeted groups. We suppose the algorithm has a budget K for new advertising to split among the groups. ν starts at a lower bound L, reflecting pre-campaign treatment rates. The algorithm may select any post-campaign ν with $||\nu - L||_1 \leq K$ and $L_i \leq \nu_i \leq U_i \; \forall i$, where $U < 1$ is an upper bound. Note that U is strictly less than 1 because we can never realistically treat 100% of any given group. Denote the set of ν satisfying these constraints as the feasible polytope $\mathcal{P}$. While we focus on the above $\mathcal{P}$ for concreteness, our approach works for any downward-closed polytope. The goal is to select a $\nu \in \mathcal{P}$ which minimizes the total infected agents over a time horizon T.

Optimization formulation: We assume that the number of infected agents (and hence deaths) is small compared to the total population. For instance, less than 1% of the total population is infected with TB in India or gonorrhea in the U.S. (WHO 2015b; CDC 2015). Therefore, we consider the total population size (the vector N^t) as fixed independently of ν. Thus, the state of the system is captured just by the infected vector I^t. A single group i evolves as

$$I_i^{t+1} = \sum_{j=1}^n M_{ji} \Big(S_j^t (1 - \mu_j) \sum_{k=1}^n \beta_{jk} \frac{I_k^t}{N_k^t} + (1 - \nu_j)(1 - d_j) I_j^t \Big) + \tilde{I}_i^t.$$

The expression in parentheses is the number of infected agents in group j. The first term is the number who are newly infected and the second is the number from previous steps who are not cured and do not die. The outer summation accounts for the number of these infected agents who transition from group j to group i. Lastly, we add the new arrivals $\tilde{I}_i^t$.

We can iterate the equation for each group forward from $t = 1...T$ in order to obtain the total number of infected agents at time T. Instead though, we will work with an equivalent matrix formulation of the system for ease of notation. For convenience, we will use the augmented state vector $x^t = [I^t \; 1]$. That is, x^t is the number of infected people appended with a single one. The one is just for mathematical convenience. We formulate a time-varying linear operator $B^t(\nu)$ such that $x^{t+1} = B^t(\nu)x^t$ via the block form

$$B^t(\nu) = \begin{bmatrix} A^t(\nu) & \tilde{I}^t \\ \vec{0} & 1 \end{bmatrix}$$

where the block $A^t(\nu)$ is defined as

$$A^t(\nu) = M^\top \Big(diag(S^t) diag(1-\mu) \; \beta \; diag\Big(\frac{1}{N^t}\Big) \\ + diag(1-\nu) diag(1-d) \Big).$$

where $diag(v)$ is the matrix with the entries of v on the diagonal. $A^t(\nu)I^t$ gives the number of infected agents given only the internal dynamics of the population, resulting in a total of $A^t(\nu)I^t + \tilde{I}^t$ infected agents. Figure 1 shows an example of a simple case of the model with two age groups. Example parameter values are given, along with the initial prevalence I^0. The first equation computes the matrix $A^0(\nu)$. The second applies $A^0(\nu)$ to the initial prevalence I^0 and then adds the exogenous inflow $\tilde{I}^0$. The number of infected agents in group 1 increases because more agents are infected than cured. These agents then transition to group 2. Because there are only two groups in our modeled population for this example, all agents in group 2 exit the modeled population.

We aim to minimize the total infected agents over T steps:

$$\min \sum_{t=1}^{T} c^\top \left[\prod_{j=t}^{1} B^j(\nu) \right] x^0$$
$$1^\top \nu \le 1^\top L + K \qquad (1)$$
$$L_i \le \nu_i \le U_i \quad \forall i = 1...n$$

c can be any nonnegative cost vector, e.g., $c = [\vec{1} \; 0]$ (n ones and a zero) sums over the number of infected agents in each group. x^0 is the initial state (number of infected agents). We use the notation $\prod_{j=t}^{1} B^t(\nu)$ as shorthand for $B^t(\nu)B^{t-1}(\nu)...B^1(\nu)$.

Algorithmic approach

We now turn to computing a (near) optimal solution to Problem 1. This is a continuous optimization problem since each ν_i may take any value in $[L_i, U_i]$. Unfortunately, the objective function is nonconvex, which rules out standard methods for efficiently obtaining good solutions. It is also highly nonlinear since the decision variables ν are raised to the power T, which may be large (e.g., a time horizon of 10 or 20 years). This suggests that many local optima could be present and renders optimization more complicated.

However, MCF-SIS's definition contains useful structure. Intuitively, resources have diminishing returns: infections averted by increasing one ν_i can no longer be treated by increasing some other ν_j. Diminishing returns suggests submodularity. However, since our optimization problem is not discrete, standard submodularity and the greedy algorithm do not apply. Instead, we show that our objective is *continuous submodular*, a generalization of submodularity to continuous domains (Bach 2015; Bian et al. 2017). Continuous submodularity enables efficient optimization and allows us to handle the stochastic case in a natural manner. This framework is crucially enabled by the modeling choice to shift from the discrete, graph-based setting common in previous work (Saha et al. 2015; Borgs et al. 2010; Chung, Horn, and Tsiatas 2009) to a continuous, population-based model. Not only does our model account for population dynamics, but it is also more amenable to optimization.

We now define continuous submodularity[1]. Let $\wedge$ and $\vee$ denote coordinatewise minimum and maximum respectively. A function $F : R^n \to R$ is continuous submodular if $F(x) + F(y) \ge F(x \vee y) + F(x \wedge y)$ for all feasible x, y. This is reminiscent of submodular set functions, but extended to the continuous domain. F is called continuous supermodular if the inequality is reversed. If F is continuous submodular, $-F$ is continuous supermodular. Note that continuous submodularity is not convexity or concavity; it is a distinct class of functions with distinct optimization techniques.

We will draw on these techniques to solve Problem 1. Bian et al. (2017) define a theoretical framework for optimizing continuous submodular functions. In order to make use of this framework, we need to first show that our problem falls into it. Then, we need to fill in the algorithmic components required to instantiate the approach that the framework suggests (two oracles explained below). Lastly, we need to prove that our objective is sufficiently smooth for the resulting algorithm to converge in a reasonable number of iterations. None of these pieces are covered by previous work; they are algorithmic contributions specific to our domain.

We start out by showing that the continuous submodularity framework applies. Denote the objective of Problem 1 as $F(\nu)$. We will show that F is continuous *supermodular* which in turn implies that $-F$ is continuous *submodular*. Since minimizing F is equivalent to maximizing $-F$, this will allow us to design an efficient algorithm based on continuous submodularity. Our proof that F is supermodular has two steps. First, we show that F is a *posynomial* in the variables $1 - \nu_i$. A posynomial is a polynomial with

---

[1]Technically, we use the stronger condition of *DR-submodularity*. Details related to showing our objective is DR-submodular can be found in the supplement.

entirely nonnegative coefficients[2]. Then, we will show that any function which is a posynomial in $1 - \nu$ is continuous supermodular in ν. We start by showing the following:

Lemma 1. *F is a posynomial in the variables $1 - \nu_i$.*

Proof. First, note that F depends on ν only through the term $diag(1 - \nu)$. Note also that every term in the expression for the block $A^t(\nu)$ is nonnegative. Since matrix multiplication is just a series of multiplications and additions, it follows that $c^\top \prod_{j=t}^1 \left[B(\nu)^j \right] x_0$ (and hence the sum over time $t = 1...T$) is a polynomial in $1 - \nu$ and all of the coefficients of this polynomial are nonnegative. This can also be seen through the expression for the evolution of a single group, which contains only terms of the form $(1 - \nu_i)$ multiplied by nonnegative coefficients. $\square$

Note that this step hinged on MCF-SIS's continuous, population-based nature. Since F is a posynomial in $1 - \nu_i$, it can be written in the form $F(\nu) = \sum_{j=1}^\ell a_j \prod_{i=1}^n (1 - \nu_i)^{p_{ij}}$ where a_j is a nonnegative coefficient for term j and p_{ij} is a nonnegative integer. This representation does not have to be computed; its existence is just useful for the proofs.

We now turn to showing that any function that is a posynomial in $1 - \nu$ is continuous supermodular in ν. Our result builds on the following lemma:

Lemma 2 (Staib and Jegelka (2017)). *Let $f_1...f_n : R \to R_+$ be nonnegative, differentiable functions which are either all nonincreasing or all nondecreasing. Then, $F(x) = \prod_{i=1}^n f_i(x)$ is continuous supermodular.*

Using this, we show the following:

Lemma 3. *Whenever F is a posynomial in $1 - \nu$, it is also continuous supermodular in ν.*

Proof. First, note that continuous supermodularity is preserved under nonnegative linear combinations. Hence, we focus on an individual term $\prod_{i=1}^n (1 - \nu_i)^{p_{ij}}$ in the posynomial representation of F. For each $i = 1...n$, define $f_i(\nu) = (1 - \nu_i)^{p_{ij}}$. Note that each f_i is nonincreasing in ν_i since $0 \le \nu_i < 1$. Further, $f_i(\nu_i) \ge 0$ always holds. The conclusion now follows from Lemma 2. $\square$

To sum up: we want to minimize F, which via Lemma 1 is a posynomial in $1 - \nu$. Via Lemma 3, this implies that F is continuous supermodular in ν. Hence, maximizing $-F$ is a continuous submodular optimization problem. We will actually maximize the objective $G(\nu) = -F(\nu) + M$, where M is any constant large enough to ensure that G is nonnegative. Clearly, this is also equivalent to minimizing F.

The DOMO algorithm

We now introduce the DOMO algorithm (Disease Outreach via Multiagent Optimization, Algorithm 1) to exploit continuous submodularity. We start out with the deterministic setting where model parameters are fully known. Here, DOMO builds on the Frank-Wolfe approach (Bian et al.

[2]Under some circumstances, posynomials can be optimized via geometric programming. Unfortunately, this does not work for our problem since the feasible set is not convex in $1 - \nu$.

2017) (though new techniques are needed in the stochastic setting). DOMO generates a series of feasible solutions $\nu^0...\nu^R$, where R is the total number of iterations. More iterations imply greater accuracy (Theorem 1 bounds the number needed). The algorithm starts at $\nu^0 = L$, the lower bound. Each iteration alternates between two steps (lines 4-5). First, it computes the gradient of the objective at the current point. Second, it takes a step in the direction of the point which optimizes the gradient over the feasible set $\mathcal{P}$. Essentially, at each iteration the algorithm spends a fraction $\frac{1}{R}$ of the budget according to the current gradient. Higher R allows finer control over the solution.

It is known that this strategy gives a $(1 - 1/e)$-approximation for continuous submodular functions (Bian et al. 2017). *However, it is not an out-of-the-box approach (even in the deterministic setting).* DOMO requires two oracles (specific to our problem) to instantiate the algorithm – a gradient oracle which supplies the gradient of G at any given point, and a linear optimization oracle which maximizes a given linear function over the feasible set $\mathcal{P}$. Additionally, the number of iterations (and hence runtime) required is potentially unbounded. We prove (in Theorem 1) that our objective is sufficiently smooth for the algorithm to converge efficiently. We first supply appropriate oracles.

Gradient oracle: Instead of tediously computing the posynomial representation, we directly compute the gradient using the block matrix representation of MCF-SIS. Denote $Y^t(\nu) = \prod_{j=1}^t B(\nu)^j$. We can concisely express the gradient via matrix calculus (Petersen and Pedersen 2008):

$$\frac{\partial G(\nu)}{\nu_i} = -\sum_{t=1}^T \text{Tr} \left[\left(\frac{\partial c^\top Y^t(\nu) x_0}{Y^t(\nu)} \right)^\top \frac{\partial Y^t(\nu)}{\nu_i} \right] =$$
$$-\sum_{t=1}^T \text{Tr} \left[cx_0^\top \sum_{\ell=1}^t \left[\prod_{i=t}^{\ell+1} B^i(\nu) \right] \frac{\partial B^\ell(\nu)}{\partial \nu_i} \left[\prod_{i=\ell-1}^1 B^i(\nu) \right] \right]$$

where the first step is the chain rule and the second follows from the product rule and induction on n. Tr denotes the matrix trace. This reduces gradient evaluation to computing $\frac{\partial}{\partial \nu_i} B^j(\nu)$ for each i and j. B^j depends on ν_i only through the block $A^j(\nu)$, so we have $\frac{\partial}{\partial \nu_i} B^j(\nu) = (1 - d_i) M^\top J_{i,i}$, where $J_{i,i}$ is the matrix with a one in entry (i, i) and zeros elsewhere. By appropriately ordering multiplications, the entire procedure uses T matrix multiplications.

Linear optimization oracle: Since $\mathcal{P}$ is a polytope, linear optimization could be performed by solving a linear program. However, exploiting the special structure of $\mathcal{P}$ lets us perform linear optimization in time $O(n \log n)$ via a simple greedy algorithm (function GREEDYLINEAR in Algorithm 1). This algorithm simply orders the group $i = 1...n$ according to $\nabla_i G(\nu)$ (Line 10). It then proceeds through the groups in this order, spending as much of the budget as possible (Line 15) before moving on to the next.

Lemma 4. GREEDYLINEAR *finds an optimal solution to the linear optimization problem over $\mathcal{P}$.*

Proof. We recognize the linear optimization problem as a fractional knapsack problem where each item takes up the

Algorithm 1 DOMO

```
 1: function DOMO(R, K)
 2:     ν⁰ ← L
 3:     for k = 1...R do
 4:         ∇ᵏ ← GRADIENTORACLE(νᵏ⁻¹)
 5:         yᵏ ← LINEARORACLE(∇ᵏ, K)
 6:         νᵏ ← νᵏ⁻¹ + 1/R yᵏ
 7:     return νᴿ
 8: function GREEDYLINEAR(∇, K)
 9:     y ← L
10:     π ← ordering of 1...n such that ∇_π(i) ≥ ∇_π(i+1) ∀i
11:     i ← 0
12:     while ||y − L||₁ < K do
13:         y_π(i) += min(U_π(i) − L_π(i), K − ||y − L||₁)
14:         i += 1
15:     return y
```

same amount of space. The value of each item is the corresponding entry of the gradient. Hence, greedily taking as much as possible of the highest-value items is optimal. □

Convergence analysis: The runtime of Algorithm 1 depends on the number of iterations R, which Bian et al. (2017) show must be proportional to the Lipschitz constant of the gradient of G. Essentially, functions with Lipschitz continuous gradients are smooth in a sense that allows the algorithm to quickly converge. Let $U_{max} = \max_i U_i$. We bound the Lipschitz constant for our model and show that

Theorem 1. *For any $\epsilon > 0$, using $R = \frac{K}{\epsilon} \left(\frac{T}{1 - U_{max}} \right)^2$ iterations, the ν output by Algorithm 1 satisfies $G(\nu) \geq \left(1 - \frac{1}{e} - \epsilon \right) G(\nu^*)$, where ν^* is an optimal solution.*

The proof is given in the supplement due to space constraints. We remark that U_{max} is always bounded away from 1 since we can never realistically treat 100% of any single group. Thus, the number of iterations is $O\left(\frac{KT^2}{\epsilon} \right)$. Each iteration requires one linear optimization over $\mathcal{P}$ (which takes time $O(n \log n)$ using GREEDYLINEAR) and one gradient evaluation (which takes time $O(Tn^\omega)$, where ω is the matrix multiplication constant). The final runtime is $O\left(\frac{KT^3 n^\omega}{\epsilon} \right)$.

Stochastic optimization

In reality, some parameters of the multiagent system will not be known exactly. For instance, the contact matrix β is almost never precisely known in practice. Additionally, for many diseases, there is considerable uncertainty about the initial prevalence I^0 (Suen et al. 2015). We now extend DOMO to the stochastic case where model parameters are drawn from a distribution instead of known exactly. Hence, we can infer an appropriate prior distribution from whatever data is available and optimize the expected value over this distribution. Our formulation is very general, and will allow any of the parameters (M, β, I^0, $\tilde{I}$, etc.) to be unknowns. Suppose that we have an uncertainty set Ξ for the joint values of the unknowns and Ξ is equipped with a distribution

D. Let $G(\cdot, \xi), \xi \in \Xi$ denote the objective for any fixed set of parameters. We wish to solve the stochastic problem

$$\max_{\nu \in \mathcal{P}} \mathbb{E}_{\xi \sim D} \left[G(\nu, \xi) \right] \qquad (2)$$

Such stochastic problems are typically difficult computationally. For instance, Zhang et al. (2014) study vaccination on a graph when the initially infected nodes (I^0 in our model) are uncertain. To design a scalable algorithm, they must assume that D is an independent distribution. However, I^0 for different groups will clearly be correlated because of the underlying multiagent dynamics. A common approach to accounting for uncertainty without such strong assumptions is robust optimization, which solves the worst case problem $\max_{\nu \in \mathcal{P}} \min_{\xi \in \Xi} G(\nu, \xi)$. Han et al. (2015) take this approach for a vaccination problem when the graph β is unknown. However, robust optimization introduces a computationally challenging bilevel optimization problem which requires specialized techniques. This makes it difficult to incorporate uncertainty in multiple parts of the model.

We resolve these difficulties through an alternate approach which efficiently handles uncertainty over *any* of the model parameters, expressed through an *arbitrary* distribution D. Moreover, we obtain provable guarantees just as in the deterministic case. We start out by noting that the objective of Problem 2 is continuous submodular since it is a nonnegative linear combination of continuous submodular functions. Also note that Algorithm 1 accesses the objective only through GRADIENTORACLE. While we can no longer access the gradient in closed form, the key idea is to instead use a stochastic approximation. At each iteration, we draw r samples $\xi_1...\xi_r$ i.i.d. from D. Our estimate of the gradient is $\widehat{\nabla} = \frac{1}{r} \sum_{i=1}^{r} \nabla G(\nu, \xi_i)$. We then modify Line 5 of Algorithm 1 to call LINEARORACLE($\widehat{\nabla}$).

To our knowledge, no previous work has analyzed stochastic continuous submodular optimization. We give a new analysis which draws on tools for analyzing stochastic concave problems (Hazan and Luo 2016). We extend these techniques to (nonconcave) continuous submodular functions and prove the following guarantee:

Theorem 2. *Using $r = \left(\frac{4KT}{1 - U_{max}} \right)^2$ samples, DOMO outputs a ν satisfying $\mathbb{E}\left[G(\nu, \xi) \right] \geq \left(1 - \frac{1}{e} - \epsilon \right) \mathbb{E}\left[G(\nu^*, \xi) \right]$ where ν^* is an optimal solution to Problem 2. The number of iterations is the same as Theorem 1.*

Note that the guarantee for $\mathbb{E}[G(\nu, \xi)]$ exactly matches that for $G(\nu)$ in the deterministic case. Further, our analysis generalizes to *any* smooth continuous submodular function and may be of interest in other domains.

Experiments

We now present experimental results on two real-world problem instances: TB prevention in India and gonorrhea prevention in the United States. In both, we produce a highly realistic evaluation by instantiating MCF-SIS using demographic and epidemiological data drawn from a variety of governmental and NGO sources. Since prevalence numbers

Table 1: Infected people per 100,000 according to the National Family Health Survey. Reported in (Suen et al. 2015).

Year/Age	30	35	40	...
1993	555.499	680.426	1059.359	...
1998	781.136	940.218	827.718	...
2003	329.045	453.052	522.364	...
2005	539.154	625.982	722.140	...

Table 2: Example parameters. E.g., $d = 0.544$ indicates that 54.4% of people with active TB die each year. Ranges indicate variance over age and/or year.

| Parameter	Value	Source
Starting pop.	355,692,752	(UN 2015)
Total infected	2,949,057	(Suen et al. 2015)
Status quo ν	0.07 - 0.13	(RNTCP 2016)
μ	0.003 - 0.02	(WHO 2015a)
d	0.544	(Tiemersma 2011)

Figure 2: Top: Improvement in mean case-years averted by DOMO over each other algorithm. Bottom: Fraction of 100 sampled instances in which DOMO's averted case-years is at least as high as each baseline.

are highly uncertain, and the contact matrix β not explicitly known, we estimate a distribution over both using this data and apply DOMO as described above. MCF-SIS is stratified by age. We account for migration and births by comparing the number of individuals in each age group at each year to the next, after accounting for non-disease deaths.

Tuberculosis: True TB prevalence in India is subject to great uncertainty, as many patients do not report to approved treatment facilities (RNTCP 2010). We estimate prevalence (the initial infected vector I^0 and new arrivals $\tilde{I}$) using age-stratified data provided by the Indian government for the years 1993-2005 (IIPS 2014), see Table 1. These figures are reported with 95% confidence intervals; we sample values of $(I^0, \tilde{I})$ within these assuming a Gaussian distribution. Table 2 shows example parameter values. For each sample, we find the $\hat{\beta}$ that minimizes the mean squared error between the observed I and that predicted by MCF-SIS. Figure 3a shows an example $\hat{\beta}$; darker cells represent more interaction. The matrix is sparse, with most entries along the diagonal (representing within-group interaction) and a few groups who interact more with others. Population statistics and disease parameters (e.g., d) are taken from World Health Organization lifetables, the Indian government Revised National Tuberculosis Control Program reports, and United Nations Statistics Division demographic reports (see supplement). Our model includes 30 age groups representing ages 30-60.

Gonorrhea: We infer the initial prevalence I^0 and new arrivals $\tilde{I}$ from reported disease cases. However, up to 80% of cases are asymptomatic and may remain undetected (CDC 2015). We assume a uniform distribution for the true prevalence at every age (I) with an upper limit of 4 times the reported values and a lower bound equal to the value reported by the U.S. Centers for Disease Control. We generate a set of sampled $(I^0, \tilde{I})$ from this uniform distribution. For each sample, we infer the β matrix which best matches the age-stratified prevalence rate in the same manner as for TB.
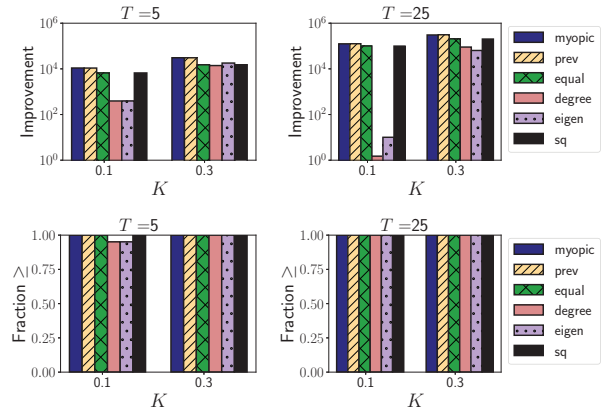
Data on population demographics is taken from the WHO and the U.S. Census (details in the supplement). Our model includes 46 age groups, representing ages 15-60.

Baselines: No previous work directly addresses our setting, so we define several baselines. First, *degree*, which greedily spends the budget on the groups with highest weighted degree with respect the contact matrix β. This captures the intuition that groups which are in contact with many others are important targets for treatment. Second, *eigen*, which greedily spends the budget on groups according to their eigenvector centrality in β. Degree and eigen test whether it is necessary to consider population dynamics, or if just the contact matrix is sufficient. Third, *myopic*, which selects the ν that gives the largest reduction in infections after a single timestep in the MCF-SIS. This can solved exactly via linear programming. Myopic tests if DOMO's long-term reasoning is needed. Fourth, *prevalence*, which allocates greedily to the groups with the most infected individuals. This is common practice in epidemiology. Fifth, *equal*, which splits the budget equally over all groups. Sixth, *SQ* which allocates the budget proportional to the status-quo ν produced by current government policies. This models spending the budget according to the currently used strategy.

Results: For each domain, we obtain the status quo treatment rate ν_{SQ} from existing data (which is the lower bound L). Then, we assume that a policymaker may distribute an additional budget K via an outreach campaign. We set $U = 1.05 \cdot \nu_{SQ}$. We do not plot runtime because all algorithms, implemented in Python, run in under 10 minutes on all datasets and parameter combinations. DOMO is run with $R = 100$ iterations and $r = 100$ samples.

We start with TB. The top row of Figure 2 shows the improvement in objective value of DOMO over each baseline. Improvement is in terms of disease burden: the total person-years of disease summed over time $1...T$ (the objective function). Each plot shows two values of K on the x axis corresponding to small and large budgets. The y axis plots the difference between the disease burden under each baseline
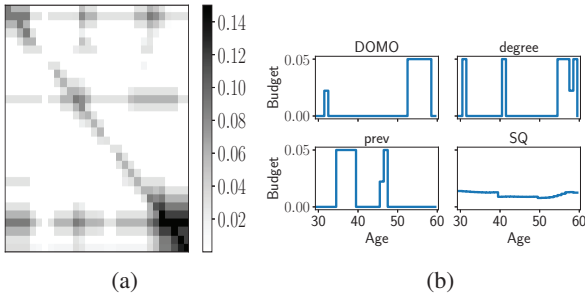
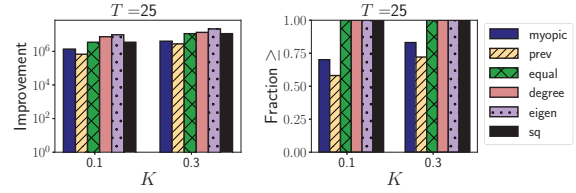Figure 3: (a) A sampled β matrix. (b) Illustrated allocation

Figure 4: Results for gonorrhea instance. Left: improvement in case-years averted by DOMO over each other algorithm. Right: fraction of instances in which DOMO performs at least as well as each baseline.

versus DOMO (note the log scale), averaging over 100 samples for the unknown parameters. One plot shows the short horizon $T = 5$, and one shows the long horizon $T = 25$.

DOMO outperforms all baselines (has positive improvement) under each configuration. The difference is larger for $K = 0.3$ than $K = 0.1$, indicating that DOMO makes more strategic use of the additional resources. Most differences also grow as T increases; the longer time horizon presents a more challenging planning problem. The two closest competitors are degree and eigen, which obtain relatively close values for $K = 0.1$. However, their gap with DOMO increases substantially for $K = 0.3$. The performance gap is very significant in policy terms: for $K = 0.3, T = 25$, DOMO averts (approximately) between 64,000 to 300,000 person-years of disease more than each baseline. All differences are statistically significant (t-test, $p < 0.001$). Given the annual death rate $d = 0.544$, DOMO averts over 6,500 TB deaths per year compared to status quo policy.

Further, DOMO performs optimally out of all considered algorithms in at least 90% of specific realizations of the parameters. In Figure 2, the y axis shows the fraction out of 100 randomly sampled parameter combinations in which DOMO performed at least as well as every other algorithm. We again plot results for $T = 5, 25$, and $K = 0.1, 0.3$. The values are all fairly high, ranging from 0.9 to 1. We conclude that our stochastic optimization approach successfully captures uncertainty in this domain because it has high performance almost all circumstances, not just in expectation.

Figure 3b contrasts the allocation made by DOMO and other policies. We focus on $K = 0.3, T = 10$. Each line shows the amount of budget the corresponding algorithm allocates to each group (shown on the x axis). To avoid crowding the plot, we show DOMO, degree, prevalence, and SQ. Myopic's allocation was very close to prevalence while eigen's was similar to degree. We see that SQ allocates the budget relatively uniformly, while DOMO concentrates heavily on particular groups. Moreover, DOMO does not simply allocate to high-prevalence groups. This indicates that DOMO exploits long term dynamics beyond which agents are immediately infected. DOMO also does not simply allocate to high degree groups. Its allocation overlaps with the high degree elderly groups, but places little budget on the high degree groups near ages 30 and 40. We conclude that DOMO leverages non-obvious patterns in the multi-agent system's dynamics to outperform simpler heuristics.

We find that DOMO also performs better than the baselines in our gonorrhea example (Figure 4). Generally, results are similar to TB, so due to space limitations, we show results for $T = 25$. The left hand figure shows the improvement in disease burden that DOMO makes over each baseline; DOMO substantially outperforms all of the baselines for both values of K. For $K = 0.3, T = 25$, DOMO results in at least 500,000 fewer person-years of disease than any other algorithm. The right hand figure plots the fraction of sampled instances in which DOMO performs at least as well as each algorithm. DOMO outperforms equal, degree, eigen, and SQ in 100% of instances. It outperforms myopic in approximately 75% of instances, and prevalence in 60-70%. DOMO's expected performance is much higher than prevalence because in those sampled instances where DOMO outperforms prevalence, it does so by a large margin. When DOMO outperforms prevalence, it does so by 3.9 million person-years on average. Conversely, when prevalence outperforms DOMO, it does so by approximately 200,000 person-years on average.

Conclusion and additional related work

We develop an algorithmic approach to targeting disease outreach campaigns which synthesizes agent-based modeling and algorithmic disease control. A large body of work in health policy and agent based modeling develops realistic disease models (Jindal and Rao 2017; Perez and Dragicevic 2009; Swarup, Eubank, and Marathe 2014; Beheshti and Sukthankar 2014; Picault et al. 2017). None use an algorithmic approach for disease control, instead examining a limited set of policies that can be exhaustively searched. In contrast, we consider the challenge of algorithmically optimizing over the entire feasible set.

Much algorithmic work focuses on immunizing the nodes of a graph to limit disease spread (Chen et al. 2016; Saha et al. 2015; Song, Hsu, and Lee 2015; Borgs et al. 2010; Drakopoulos, Ozdaglar, and Tsitsiklis 2014). None of this work considers the challenges of population dynamics. While others may examine subgroup dynamics (Zhang et al. 2016) or time trends (Prakash et al. 2010), our work presents a novel approach to optimizing resource allocation for infectious diseases in a stochastic setting.

References

Bach, F. 2015. Submodular functions: from discrete to continous domains. *arXiv preprint arXiv:1511.00394*.

Beheshti, R., and Sukthankar, G. 2014. A normative agent-based model for predicting smoking cessation trends. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, 557–564. International Foundation for Autonomous Agents and Multiagent Systems.

Bian, A. A.; Mirzasoleiman, B.; Buhmann, J. M.; and Krause, A. 2017. Guaranteed non-convex optimization: Submodular maximization over continuous domains. In *AISTATS*.

Borgs, C.; Chayes, J.; Ganesh, A.; and Saberi, A. 2010. How to distribute antidote to control epidemics. *Random Structures & Algorithms* 37(2):204–222.

CDC. 2015. Reported STDs in the United States.

Chan, C. H.; McCabe, C. J.; and Fisman, D. N. 2011. Core groups, antimicrobial resistance and rebound in gonorrhoea in north america. *Sex Transm Infect* sextrans–2011.

Chen, C.; Tong, H.; Prakash, B. A.; Tsourakakis, C. E.; Eliassi-Rad, T.; Faloutsos, C.; and Chau, D. H. 2016. Node immunization on large graphs: Theory and algorithms. *IEEE Transactions on Knowledge and Data Engineering* 28(1):113–126.

Chung, F.; Horn, P.; and Tsiatas, A. 2009. Distributing antidote using pagerank vectors. *Internet Mathematics* 6(2):237–254.

Dowdy, D. W.; Golub, J. E.; Chaisson, R. E.; and Saraceni, V. 2012. Heterogeneity in tuberculosis transmission and the role of geographic hotspots in propagating epidemics. *Proceedings of the National Academy of Sciences* 109(24):9557–9562.

Drakopoulos, K.; Ozdaglar, A.; and Tsitsiklis, J. N. 2014. An efficient curing policy for epidemics on graphs. *IEEE Transactions on Network Science and Engineering* 1(2):67–75.

Han, S.; Preciado, V. M.; Nowzari, C.; and Pappas, G. J. 2015. Data-driven network resource allocation for controlling spreading processes. *IEEE Transactions on Network Science and Engineering* 2(4):127–138.

Hazan, E., and Luo, H. 2016. Variance-reduced and projection-free stochastic optimization. In *International Conference on Machine Learning*, 1263–1271.

IIPS. 2014. International Institute for Population Sciences. District level household and facility survey. New Delhi, India. http://www.rchiips.org/obtainingdata.html.

Jindal, A., and Rao, S. 2017. Agent-based modeling and simulation of mosquito-borne disease transmission. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, 426–435. International Foundation for Autonomous Agents and Multiagent Systems.

Lee, B. Y.; Brown, S. T.; Korch, G. W.; Cooley, P. C.; Zimmerman, R. K.; Wheaton, W. D.; Zimmer, S. M.; Grefenstette, J. J.; Bailey, R. R.; Assi, T.-M.; et al. 2010. A computer simulation of vaccine prioritization, allocation, and rationing during the 2009 h1n1 influenza pandemic. *Vaccine* 28(31):4875–4879.

Luke, D. A., and Stamatakis, K. A. 2012. Systems science methods in public health: dynamics, networks, and agents. *Annual review of public health* 33:357–376.

Perez, L., and Dragicevic, S. 2009. An agent-based approach for modeling dynamics of contagious disease spread. *International journal of health geographics* 8(1):50.

Petersen, K. B., and Pedersen, M. S. 2008. The matrix cookbook. *Technical University of Denmark* 7:15.

Picault, S.; Huang, Y.-L.; Sicard, V.; and Ezanno, P. 2017. Enhancing sustainability of complex epidemiological models through a generic multilevel agent-based approach. In *International Joint Conference on Artificial Intelligence (IJCAI'2017)*, 374–380.

Prakash, B. A.; Tong, H.; Valler, N.; Faloutsos, M.; and Faloutsos, C. 2010. Virus propagation on time-varying networks: Theory and immunization algorithms. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 99–114. Springer.

RNTCP. 2010. Training module for medical practitioners. New Delhi, India: Revised National Tuberculosis Control Programme, Ministry of Health and Family Welfare.

RNTCP. 2016. Revised national tuberculosis control programme annual status report. New Delhi, India: Ministry of Health and Family Welfare. http://tbcindia.nic.in/showfile.php?lid=3180.

Saha, S.; Adiga, A.; Prakash, B. A.; and Vullikanti, A. K. S. 2015. Approximation algorithms for reducing the spectral radius to control epidemic spread. In *Proceedings of the SIAM International Conference on Data Mining*, 568–576. SIAM.

Song, C.; Hsu, W.; and Lee, M. L. 2015. Node immunization over infectious period. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 831–840. ACM.

Suen, S.; Bendavid, E.; and Goldhaber-Fiebert, J. 2015. Cost-effectiveness of improvements in diagnosis and treatment accessibility for tuberculosis control in india. *The International Journal of Tuberculosis and Lung Disease* 19(9):1115–1124.

Swarup, S.; Eubank, S. G.; and Marathe, M. V. 2014. Computational epidemiology as a challenge domain for multiagent systems. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, 1173–1176. International Foundation for Autonomous Agents and Multiagent Systems.

UN. 2015. United Nations Statistics Division Demographic Statistics. http://data.un.org/Data.aspx?d=POP&f=tableCode%3A22.

White, P. J.; Ward, H.; Cassell, J. A.; Mercer, C. H.; and Garnett, G. P. 2005. Vicious and virtuous circles in the dynamics of infectious disease and the provision of health care: gonorrhea in britain as an example. *The Journal of infectious diseases* 192(5):824–836.

WHO. 2015a. World Health Organization. Life Tables. http://www.who.int/gho/mortality_burden_disease/life_tables/life_tables/en/.

WHO. 2015b. World Health Organization. Tuberculosis country profiles. http://www.who.int/tb/country/data/profiles/en/.

Zhang, Y., and Prakash, B. A. 2014. Scalable vaccine distribution in large graphs given uncertain data. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, 1719–1728. ACM.

Zhang, Y.; Adiga, A.; Saha, S.; Vullikanti, A.; and Prakash, B. A. 2016. Near-optimal algorithms for controlling propagation at group scale on networks. *IEEE Transactions on Knowledge and Data Engineering* 28(12):3339–3352.