2015 European Control Conference (ECC)
July 15-17, 2015. Linz, Austria



# Distributed Resource Allocation for Control of Spreading Processes

Chinwendu Enyioha, Ali Jadbabaie, Victor Preciado and George Pappas

***Abstract*—We present a distributed resource allocation strategy to control an epidemic outbreak in a networked population based on a Distributed Alternating Direction Method of Multipliers (D-ADMM) algorithm. We consider a linearized Susceptible-Infected-Susceptible (SIS) epidemic spreading model and present a distributed solution in which agents in the network are able to locally compute their optimal allocation of vaccination resources (for prevention) and antidotes (for treatment) to control the contagion. We express our epidemic control condition as a spectral constraint involving the Perron-Frobenius eigenvalue, and formulate the resource allocation problem as a Geometric Program (GP). Next, we separate the network-wide optimization problem into subproblems optimally solved by each agent in a fully distributed way. We conclude the paper by illustrating performance of our solution framework with numerical simulations.**

## I. Introduction

In this paper, we consider the problem of controlling the outbreak of an epidemic process in a networked population in the absence of a social planner. We propose a distributed solution that enables local computation of optimal investment in vaccines and/or antidotes at each node to contain the spread of an outbreak. Our distributed solution is based on a *Distributed Alternating Direction Method of Multipliers* (D-ADMM) algorithm. The distributed solution framework presented here is based on a Geometric Programming formulation of the epidemic control problem, and its convex characterization first proposed in [1].

The fairly recent outbreaks of the Middle Eastern Respiratory Syndrome (MERS) virus [2] and Ebola virus [3] have rekindled interest in analysis and control of epidemic outbreaks in networked populations. The analysis of an SIS epidemic process in arbitrary (undirected) networks was first studied by Wang et al. [4] using a discrete-time model. Key in [4] was the establishment of an epidemic threshold based on the spectral radius of the network adjacency matrix. Ganesh et al. in [5] studied a continuous-time SIS epidemic model where the relation between the spectral radius of the network adjacency matrix and the speed of spreading was further corroborated. A similar result was proposed in [6].

In the event of an epidemic outbreak, of interest is how to optimally allocate vaccines and/or antidotes across the the population to control the spread of the epidemic. Sometimes, the focus is to hasten the rate at which the contagion is contained; or to minimize the associated cost of containing the outbreak. Significant research attention has been given to this problem in the last decade; for instance, see [7, 8, 9] where different heuristics were presented. A convex framework to compute the optimal allocation of vaccines in undirected networks using Semidefinite programming (SDP) techniques was discussed in [10]; later generalized via Geometric Programming techniques to general weighted, directed, connected networks in [1].

In the absence of a social planner (or central coordinator), centralized computation of optimal investments in vaccines and/or antidotes as was done in [1] poses a challenge. Furthermore, the network might be too large for a centralized optimization scheme. These motivate the need for an optimal, decentralized framework to achieve the network-wide objective of controlling an outbreak. In this paper, we propose a fully distributed ADMM algorithm that enables local computation of optimal investment in vaccines and/or antidotes at each node to control the spread of an outbreak in a directed, network comprising heterogeneous agents. While the discussion in this paper focuses on strongly connected networks, the results are easily amenable to networks that are not necessarily strongly connected and other epidemic models.

Distributed optimization methods exist in the literature and have been applied to several classes of problems; see, for example [11, 12]. Our work is similar in spirit to [13] where tests for distributed control of positive systems were presented. However, we employ a D-ADMM – a dual-based method. In contrast to [13, 14], our problem in its original form is a nonconvex, constrained optimization problem on a directed network, where agents must cooperate to complete a global task. A game-theoretic formulation has also been studied in [15] where different equilibria were analyzed for undirected networks, contrasting our distributed solution that applies to directed networks with non-identical agents.

The rest of the paper is organized as follows – In section II, we present the notation, spreading model, as well as state the resource allocation problem. Section III comprises a GP formulation of the problem, alongside its convex characterization. The proposed distributed solution is presented in Section IV; with results from our solution illustrated in Section V. A summary and concluding remarks follow in Section VI.

## II. Preliminaries, Model and Problem Statement

### A. Graph Theory

We define a graph as $\mathcal{G} \triangleq (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} \triangleq \{v_1, \ldots, v_n\}$ is a set of $n$ nodes, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is a set of ordered pairs of nodes called edges. The node pair $(v_j, v_i)$ form an edge in $\mathcal{G}$. If $\mathcal{G}$ is a directed graph, the pair $(v_j, v_i)$ is an oriented edge from node $v_j$ to node $v_i$. For a directed graph $\mathcal{G}$, we define the in-neighborhood of node $v_i$ (that is, the set of nodes with edges pointing towards $v_i$), as $N_i^{in} \triangleq \{j : (v_j, v_i) \in \mathcal{E}\}$. A directed path from $v_{i_1}$ to $v_{i_l}$ in $\mathcal{G}$ is an ordered set of vertices $(v_{i_1}, v_{i_2}, \ldots, v_{i_{l+1}})$ such that $(v_{i_s}, v_{i_{s+1}}) \in \mathcal{E}$ for $s = 1, \ldots, l$.

*This research was supported by ARO MURI W911NF-12-1-0509 and AFOSR Complex Networks Program

All authors are with the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia PA, USA 19104 {cenyioha, jadbabai, preciado, pappasg} @seas.upenn.edu

978-3-9524269-3-7 ©2015 EUCA



Authorized licensed use limited to: University of Pennsylvania. Downloaded on October 29,2026 at 15:24:04 UTC from IEEE Xplore. Restrictions apply.

A directed graph $\mathcal{G}$ is *strongly connected* if, for every pair of nodes $v_i, v_j \in \mathcal{V}$, there is a directed path from $v_i$ to $v_j$.

We denote the adjacency matrix of a directed graph $\mathcal{G}$ by $A = [a_{ij}]$ if the edge $(v_j, v_i) \in \mathcal{E}$ points from $v_j$ to $v_i$, and $a_{ij} = 0$ otherwise. Given an $n \times n$ matrix $M$, we denote by $\mathbf{v}_1(M), \ldots, \mathbf{v}_n(M)$ and $\lambda_1(M), \ldots, \lambda_n(M)$ the set of eigenvectors and corresponding eigenvalues of $M$, respectively. We denote $\rho(M)$ the spectral radius of $M$, which is the maximum modulus across all eigenvalues of $M$. Vectors are denoted using boldface letters and matrices using capital letters. The letter $I$ denotes the identity matrix and $\mathbb{R}(z)$ denotes the real part of $z \in \mathbb{C}$.

### B. Spreading Model

The spreading model considered in this paper is a continuous-time heterogeneous (SIS) epidemic model and closely follows the development in [16]. In the heterogeneous model, we assume a networked Markov process in which nodes in the network can be in one of two states – susceptible or infected. Over time, based on each agent's infection rate $\beta_i$, recovery rate $\delta_i$ and interactions with neighboring agents and their states, the (infected or susceptible) state of each agent evolves. We assume the agents are non-identical in the sense that each agent has distinct infection rate $\beta_i$ and recovery rate $\delta_i$; and that these rates can be controlled by either injecting vaccines or antidotes.

The evolution of the spreading model is as follows. At each time-step, the state of node $v_i$ is a binary random variable $X_i(t) \in \{0, 1\}$. The states $X_i(t) = 0$ and $X_i(t) = 1$ respectively indicate that node $v_i$ is susceptible and infected. Let the vector of states be $X(t) = (X_1(t), \ldots, X_n(t))^T$, and let node $v_i$ be susceptible at time $t$. Its probability of switching to the infected state depends on its infection rate $\beta_i$, the state of its neighbors $\{X_j(t), \text{ for } j \in \mathcal{N}_i^{in}\}$. Mathematically, the probability of node $v_i$ switching from susceptible to infected state can be expressed as

$$\Pr\left(X_i(t+\Delta t) = 1 | X_i(t) = 0, X(t)\right) = \sum_{j \in \mathcal{N}_i^{in}} a_{ij}\beta_i X_j(t)\,\Delta t + o(\Delta t), \quad (1)$$

where $\Delta t > 0$ is an asymptotically small time interval. Similarly, suppose node $v_i$ is infected at time $t$; the probability of node $v_i$ returning to the susceptible state depends on its recovery rate $\delta_i$, as well as the states of its in-neighbors. This is given by

$$\Pr(X_i(t+\Delta t) = 0 | X_i(t) = 1, X(t)) = \delta_i \Delta t + o(\Delta t). \quad (2)$$

In an $n-$agent network, the $2-$state Markov process above has $2^n$ states. The exponentially increasing state space of this model poses computational challenges and makes this model difficult to analyze for large networks. To overcome this challenge, we use a mean-field approximation of its dynamics [6]. By applying mean-field theory to the Markov chain, the dynamics of infection spread can be approximated using a system of differential equations. Suppose we define $p_i(t) \triangleq \Pr(X_i(t) = 1) = \mathbb{E}(X_i(t))$, i.e., the marginal probability of node $v_i$ being infected at time $t$. Then, the Markov differential equation [6] for the state $X_i(t) = 1$ is:

$$\frac{dp_i(t)}{dt} = (1 - p_i(t))\,\beta_i \sum_{j=1}^{n} a_{ij} p_j(t) - \delta_i p_i(t). \quad (3)$$

Since $i = 1, \ldots, n$, we can represent (3) more compactly by a system of nonlinear differential equation with dynamics

$$\frac{d\mathbf{p}(t)}{dt} = (BA - D)\mathbf{p}(t) - P(t)BA\mathbf{p}(t), \quad (4)$$

where $\mathbf{p}(t) \triangleq (p_1(t), \ldots, p_n(t))^T$, $B \triangleq \text{diag}(\beta_i)$, $D \triangleq \text{diag}(\delta_i)$, and $P(t) \triangleq \text{diag}(p_i(t))$. This ODE presents an equilibrium point at $\mathbf{p}^* = 0$, called the disease-free equilibrium. Readers are referred to [6, 17] for a detailed treatment and presentation of the mean-field approximation used above.

**Proposition 1.** *Consider the nonlinear dynamical system in* (4)*, where the network adjacency matrix $A \geq 0$, $B = diag(\beta_i) \geq 0$ and $D = diag(\delta_i) > 0$. Suppose the real part of the largest eigenvalue of $BA - D$ satisfies*

$$\mathbb{R}\left(\lambda_1(BA - D)\right) \leq -\varepsilon, \quad (5)$$

*for $\varepsilon > 0$, the disease-free equilibrium is globally exponentially stable; that is, $\|\mathbf{p}(t)\| \leq \|\mathbf{p}(0)\| K e^{-\varepsilon t}$, for $K > 0$.*

*Proof:* See [10, 16] for proof; where we showed that the linear dynamical system $\dot{\mathbf{p}}(t) = (BA - D)\,\mathbf{p}(t)$ upper-bounds the dynamics in (3); thus, the spectral result in (5) is a sufficient condition for the mean-field approximation of (4) to be globally exponentially stable. ■

### C. The Distributed Resource Allocation Problem

We present the epidemic control problem as a resource allocation problem in which vaccines and antidotes are respectively applied at each node to reduce their infection rates and increase their recovery rates within feasible intervals $0 < \underline{\beta}_i \leq \beta_i \leq \overline{\beta}_i$ and $0 < \underline{\delta}_i \leq \delta_i \leq \overline{\delta}_i$. The values of $\beta_i$ and $\delta_i$ depend on the amount of vaccines and antidotes allocated at node $v_i$. We assume that vaccines applied at node $v_i$ have a convex cost $f_i(\beta_i)$ which is monotonically decreasing in $\beta_i$. Similarly, antidotes applied at node $v_i$ have a cost $g_i(\delta_i)$ which is monotonically nondecreasing in $\delta_i$, and represents the cost of modifying $\delta_i$ within the feasible interval to control the spread of the epidemic outbreak.

In this paper, we focus on the *rate-constrained* resource allocation problem, in which the objective is to find the cost-optimal, *local* allocation of vaccines and antidotes to contain the spread of an outbreak at a specified decay rate. The problem is formally stated below:

**Problem 1.** *Given a directed network with associated adjacency matrix A, node cost functions $\{f_i(\beta_i), g_i(\delta_i)\}_{i=1}^n$ and bounds on the infection and curing rates $0 < \underline{\beta}_i \leq \beta_i \leq \bar{\beta}_i$, and $0 < \underline{\delta}_i \leq \delta_i \leq \bar{\delta}_i$ respectively, and an exponential decay rate $\bar{\varepsilon} > 0$. Locally (via interaction between neighboring nodes), determine the cost-optimal distribution of vaccines and treatment resources to attain the desired decay rate.*



Authorized licensed use limited to: University of Pennsylvania. Downloaded on October 29,2026 at 15:24:04 UTC from IEEE Xplore. Restrictions apply.

Mathematically, the objective is to:

$$\underset{\{\beta_i,\delta_i\}_{i=1}^n}{\text{minimize}} \sum_{i=1}^{n} f_i(\beta_i) + g_i(\delta_i) \tag{6}$$

$$\text{subject to } \mathbb{R}(\lambda_1(BA - D)) \leq -\bar{\varepsilon}, \tag{7}$$

$$\underline{\beta}_i \leq \beta_i \leq \overline{\beta}_i, \tag{8}$$

$$\underline{\delta}_i \leq \delta_i \leq \overline{\delta}_i,\ i = 1, \ldots, n, \tag{9}$$

where (6) is the total investment across all agents, (7) constrains the decay rate to $\bar{\varepsilon}$, and (8)-(9) maintain the infection and recovery rates in their feasible limits. Our goal is to solve (6)-(9) in a fully distributed manner, with the computation of investment per node $f_i(\beta_i) + g_i(\delta_i)$, carried out locally. After presenting a convex characterization of the problem, we illustrate how it decomposes nicely for a distributed solution.

## III. The Resource Allocation Problem as a GP

In this section, we formulate the resource allocation problem in (6)-(9) as a Geometric Program and present its convex characterization. Geometric Programs (GPs) are a class of nonlinear, nonconvex optimization problems that can be transformed into convex optimization problems and efficiently solved [18]. Building blocks of a GP are monomial and posynomial functions. Let the vector $\mathbf{x} \triangleq (x_1, \ldots, x_n) \in \mathbb{R}^n_{++}$ denote $n$ decision variables. A monomial functions (of the variables $x_1, \ldots, x_n$) is a real-valued function $f(\mathbf{x}) = c x_1^{a_1} x_2^{a_2} \ldots x_n^{a_n}$, where $c > 0$ is a coefficient and $a_i \in \mathbb{R}$ are exponents. Posynomial functions are sums of monomials; that is, a function of the form $f(\mathbf{x}) = \sum_{k=1}^{K} c_k x_1^{a_{1k}} x_2^{a_{2k}} \ldots x_n^{a_{nk}}$, where $c_k > 0$, for $k = 1, 2, \ldots, K$ and $a_{jk} \in \mathbb{R}$ for $j = 1, \ldots, n$ and $k = 1, \ldots, K$. For example, while $x_1 + x_2^4$ and $x_1 x_2^{0.3} \pi$ are posynomials, $x_1 - x_2^2$ is not a posynomial. A GP in standard form is one in which a posynomial function is minimized subject to posynomial upper bound inequality constraints and monomial equality constraints. More formally, it is of the form

$$\begin{aligned} \text{minimize} \quad & f_0(\mathbf{x}) \\ \text{subject to} \quad & f_i(\mathbf{x}) \leq 1, \quad i = 1, \ldots, m, \\ & g_i(\mathbf{x}) = 1, \quad i = 1, \ldots, p, \end{aligned} \tag{10}$$

where $f_i$ are posynomial functions and $g_i$ are monomials. (10) is not convex; however, with a logarithmic change of variables and multiplicative constants: $y_i = \log x_i$, $b_l = \log c_l$, $b_{ik} = \log c_{ik}$ and a logarithmic change of the functions' values, we can transform (10) to the following equivalent problem in the variable $\mathbf{y}$:

$$\begin{aligned} \text{minimize} \quad & h_0(\mathbf{y}) = \log \sum_{k=1}^{K_0} \exp(\mathbf{a}_{0k}^T \mathbf{y} + b_{0k}) \\ \text{subject to} \quad & h_i(\mathbf{y}) = \log \sum_{k=1}^{K_i} \exp(\mathbf{a}_{ik}^T \mathbf{y} + b_{ik}) \leq 0, \quad \forall\ i \\ & q_i(\mathbf{y}) = \mathbf{a}_l^T \mathbf{y} + b_l = 0, \quad l = 1, \ldots, M. \end{aligned} \tag{11}$$

Problem (11) is convex and can be efficiently solved in polynomial time. (See [18] for a more detailed exposition on GPs). Hence, if our cost function is a posynomial (or more generally, convex in log scale), and if the constraints can be represented as monomial and posynomial functions, (6)-(9) can be formulated as a GP in convex form. Via the Perron-Frobenius lemma, from the theory of nonnegative matrices, we express the spectral constraint (5) in an equivalent form to transform it to a set of posynomial constraints in the decision variables.

**Lemma 2.** *(Perron-Frobenius) Let $M$ be a nonnegative, irreducible matrix. Then, the following statements about its spectral radius, $\rho(M)$, hold: $(i)$. $\rho(M) > 0$ is a simple eigenvalue of $M$, $(ii)$. $M\mathbf{u} = \rho(M)\mathbf{u}$, for some $\mathbf{u} \in \mathbb{R}^n_{++}$; and $(iii)$. $\rho(M) = \inf\{\lambda \in \mathbb{R} : M\mathbf{u} \leq \lambda\mathbf{u}$ for $\mathbf{u} \in \mathbb{R}^n_{++}\}$.*

Recall that our model assumes the contact network is directed, strongly connected. Since the adjacency matrix associated with a strongly connected, directed graph is irreducible, Lemma 2 holds for the spectral radius of the adjacency matrix of any positively weighted, strongly connected digraph. A corollary of Lemma 2 is that if a matrix $M$ is nonnegative and irreducible, its eigenvalue with the largest real part, $\lambda_1(M)$, is real, simple, and equal to its spectral radius $\rho(M) > 0$. In [1], based on Propositions 1 and Lemma 2, the following result established the formulation of (6)-(9) as a GP comprising monomial and posynomial functions.

**Theorem 3.** *([1]) Given a strongly connected graph $\mathcal{G}$ with adjacency matrix $A = [A_{ij}]$, posynomial cost functions $\{f_i(\beta_i), g_i(\delta_i)\}_{i=1}^n$, bounds on the infection and recovery rates $0 < \underline{\beta}_i \leq \beta_i \leq \overline{\beta}_i$ and $0 < \underline{\delta}_i \leq \delta_i \leq \overline{\delta}_i$, $i = 1, \ldots, n$, and a desired exponential decay rate $\bar{\varepsilon}$. Then, the optimal investment on vaccines and antidotes for node $v_i$ to solve Problem* (6)-(9) *are $f_i(\beta_i^*)$ and $g_i\left(\widetilde{\Delta} + 1 - \widetilde{\delta}_i^*\right)$, where $\widetilde{\Delta} \triangleq \max\{\bar{\varepsilon}, \overline{\delta}_i$ for $i = 1, \ldots, n\}$ and $\beta_i^*, \widetilde{\delta}_i^*$ are the optimal solution for $\beta_i$ and $\widetilde{\delta}_i$ in the following GP:*

$$\underset{\{u_i,\beta_i,\widetilde{\delta}_i,t_i\}_{i=1}^n}{\textit{minimize}} \sum_{k=1}^{n} f_k(\beta_k) + g_k(t_k) \tag{12}$$

$$\textit{subject to } \frac{\beta_i \sum_{j=1}^n A_{ij} u_j + \widetilde{\delta}_i u_i}{\left(\widetilde{\Delta} + 1 - \bar{\varepsilon}\right) u_i} \leq 1, \tag{13}$$

$$\left(t_i + \widetilde{\delta}_i\right) \Big/ \left(\widetilde{\Delta} + 1\right) \leq 1, \tag{14}$$

$$\widetilde{\Delta} + 1 - \overline{\delta}_i \leq \widehat{\delta}_i \leq \widetilde{\Delta} + 1 - \underline{\delta}_i, \tag{15}$$

$$\underline{\beta}_i \leq \beta_i \leq \overline{\beta}_i,\ i = 1, \ldots, n. \tag{16}$$

*Proof:* See Theorem 12 in [1] for proof. ∎

For simplicity of notation and ease of reading, we re-express (12)-(16) as

$$\underset{u_i,\beta_i,\delta_i,}{\text{minimize}} \quad \sum_{i=1}^{n} f_i(\beta_i) + g_i(\delta_i) \tag{17}$$

$$\text{subject to } \frac{\beta_i \sum_{j=1}^n A_{ij} u_j + \delta_i u_i}{u_i} \leq 1, \tag{18}$$

$$\underline{\delta} \leq \delta_i \leq \overline{\delta} \quad \text{and} \quad \underline{\beta} \leq \beta_i \leq \overline{\beta}, \quad i = 1, \ldots, n; \tag{19}$$

where the auxiliary variables $\widetilde{\Delta}, t_i, \widetilde{\delta}_i$ introduced in Theorem



Authorized licensed use limited to: University of Pennsylvania. Downloaded on October 29,2026 at 15:24:04 UTC from IEEE Xplore. Restrictions apply.

3 have been factored into the upper and lower bounds on $\delta_i$ in (19), and $\tilde{\Delta} + 1 - \bar{\varepsilon} = 1$. The rest of the paper will focus on a fully distributed solution for (17)-(19).

### A. Separability of (17)-(19)

To implement a distributed solution, first note that the cost function and constraint functions of the GP in (17)-(19) are separable per agent. Each agent is able to locally solve the following problem:

$$\underset{u_i,\beta_i,\delta_i,}{\text{minimize}} \quad f_i(\beta_i) + g_i(\delta_i) \tag{20}$$

$$\text{subject to } \frac{\beta_i \sum_{j=1}^{n} A_{ij} u_j + \delta_i u_i}{u_i} \leq 1, \tag{21}$$

$$\underline{\delta} \leq \delta_i \leq \overline{\delta} \quad \text{and} \quad \underline{\beta} \leq \beta_i \leq \overline{\beta}. \tag{22}$$

Though separable per agent, not all the decision variables are local. For agent $v_i$ to solve (20)-(22), it needs the value $u_j$ from nodes in its neighborhood set (see (21)). To address this problem, we employ the Alternating Direction Method of Multipliers algorithm, well suited for such distributed optimization problems.

## IV. Distributed solution

### A. Alternating Direction Method of Multipliers (ADMM)

The ADMM algorithm is a dual-based method for solving constrained optimization problems in which an augmented Lagrangian function is minimized with respect to the primal variables, and the dual variables are updated accordingly. Surveys can be found in [19, 20, 21]. The ADMM algorithm works by decomposing the original optimization problem into subproblems that can be sequentially solved in parallel by each agent, allowing for distributed solutions to large-scale optimization problems. The standard ADMM solves the following problem

$$\begin{aligned} \underset{x,z}{\text{minimize}} \quad & f(x) + g(z) \\ \text{subject to} \quad & Ax + Dz = c, \end{aligned} \tag{23}$$

where the variables $x \in \mathbb{R}^n$, $z \in \mathbb{R}^m$, matrices $A \in \mathbb{R}^{p \times n}$, $D \in \mathbb{R}^{p \times m}$, and $c \in \mathbb{R}^p$. The Augmented Lagrangian function for (23) is given by

$$L_\rho(x, z, \lambda) = f(x) + g(z) - \lambda^T (Ax + Dz - c) + \frac{\rho}{2} \|Ax + Dz - c\|^2, \tag{24}$$

where $\lambda$ is the Lagrange multiplier associated with the constraint $Ax + Dz = c$ and $\rho$ is a positive scalar. The update rules for the variables $x$, $z$ and $\lambda$ in the ADMM implementation is given by

$$x(k+1) = \arg\min_x L_\rho(x, z(k), \lambda(k)) \tag{25}$$

$$z(k+1) = \arg\min_z L_\rho(x(k+1), z, \lambda(k)) \tag{26}$$

$$\lambda(k+1) = \lambda(k) - \rho\left((Ax(k+1) + Dz(k+1) - c\right) \tag{27}$$

The updates in (25) - (27) is similar to those of dual descent algorithms, except that augmented Lagrangian is used and penalty parameter $\rho$ is used as the step size in the dual updates.

### B. Resource Allocation via D-ADMM

To make (20)-(22) amenable to a distributed solution via a D-ADMM algorithm, we introduce variables $\mathbf{u}_i \in \mathbb{R}^n$ representing a local copy of the global variable $\mathbf{u} = (u_1, \ldots, u_n)^T$ in (21) at each node $v_i$. We also introduce an auxiliary variable $\mathbf{z}_{ij}$, that enables communication and enforces consensus in the values of $\mathbf{u}_i$ and $\mathbf{u}_j$ for all neighboring nodes $(v_i, v_j) \in \mathcal{E}$. With these new variables, and consensus constraint, we can reformulate (20)-(22) as

$$\underset{\mathbf{u}_i,\beta_i,\delta_i,}{\text{minimize}} \quad f_i(\beta_i) + g_i(\delta_i)$$

$$\text{subject to } \frac{\beta_i \sum_{j=1}^{n} A_{ij} \mathbf{u}_i^j + \delta_i \mathbf{u}_i^i}{\mathbf{u}_i^i} \leq 1, \tag{28}$$

$$\mathbf{u}_i = \mathbf{z}_{ij}, \quad \mathbf{u}_j = \mathbf{z}_{ij}, \quad (v_i, v_j) \in \mathcal{E},$$

$$\underline{\delta} \leq \delta_i \leq \overline{\delta}, \quad \underline{\beta} \leq \beta_i \leq \overline{\beta}, \quad \prod_{j=1}^{n} \mathbf{u}_i^j = 1, \tag{29}$$

where the scalar $\mathbf{u}_i^j$ is the $j$th entry of the local estimate $\mathbf{u}_i$ at node $v_i$. The constraints $\mathbf{u}_i = \mathbf{z}_{ij}$ and $\mathbf{u}_j = \mathbf{z}_{ij}$ ensures that for all pairs of adjacent agents $(v_i, v_j)$, the feasible set of (28)-(29) is such that $\mathbf{u}_i = \mathbf{u}_j$. If the network is strongly connected, these local consensus constraints imply that feasible solutions must satisfy $\mathbf{u}_i = \mathbf{u}_j$ for all, not necessarily neighboring, pairs of agents $v_i$ and $v_j$.[1] Normalization of the vectors $\mathbf{u}_i$ for $i = 1, \ldots, n$ is to ensure that the local estimates $\mathbf{u}_i$ have the same direction.

To compute the augmented Lagrange of (28), let the dual variable $\alpha_{ij}$ and $\gamma_{ij}$ be associated with equality constraints $\mathbf{u}_i = \mathbf{z}_{ij}$, and $\mathbf{u}_j = \mathbf{z}_{ij}$, $\forall\ (v_i, v_j) \in \mathcal{E}$ respectively. In the iterations of the D-ADMM algorithm, the dual variable updates are:

$$\alpha_{ij}(k+1) = \alpha_{ij}(k) + \frac{\rho}{2}(\mathbf{u}_i(k) - \mathbf{u}_j(k)) \ \forall\ j \in N(i) \tag{30}$$

$$\gamma_{ij}(k+1) = \gamma_{ij}(k) + \frac{\rho}{2}(\mathbf{u}_j(k) - \mathbf{u}_i(k)) \ \forall\ j \in N(i). \tag{31}$$

Let $\Gamma_i(k+1) \triangleq \beta_i(k+1), \delta_i(k+1), \mathbf{u}_i(k+1)$. Further, let $\phi_i(k) \triangleq \sum_{j \in N(i)} (\alpha_{ij}(k) + \gamma_{ji}(k)) \ \forall\ v_i \in V$. Given $\mathbf{u}_i(0) \in \mathbb{R}^n$ and $\phi_i(0) = \mathbf{0}$, the iterative computations and updates are summarized in Algorithm 1.

## V. Numerical Simulations

In this section, we illustrate performance of the D-ADMM Algorithm 1 on two strongly connected directed networks, and briefly discuss its convergence. As a proof of concept, and to show that functional correctness of Algorithm 1, we will show that the investment in vaccines and antidotes for all agents in the network converge to the solution obtained in the centralized case. Further, we will show convergence of the of the dual variables $\phi_i$ for all agents $v_i$.

The D-ADMM algorithm was used to solve Problem 1 on a 8-node network with the following parameters: epidemic threshold $\tau_c = (1 - \overline{\delta})/\rho(A)$; $\overline{\beta} = 2\tau_c$, $\underline{\beta} = 0.2\overline{\beta}$ and

[1] In contrast to existing literature on Distributed ADMM algorithm [14, 21, 22], where only consensus of local estimates is required, the distributed resource allocation problem we consider in addition to being constrained locally, requires consensus in local estimates.



Authorized licensed use limited to: University of Pennsylvania. Downloaded on October 29,2026 at 15:24:04 UTC from IEEE Xplore. Restrictions apply.

**Algorithm 1** Distributed ADMM

1: **Given** initial variables $\beta_i(0), \delta_i(0) \in \mathbb{R}$, $\mathbf{u}_i(0), \phi_i(0) \in \mathbb{R}^n$ for each agent $v_i \in \mathcal{V}$.
2: **Set** $k = 1$
3: **repeat**
4: For all $v_i \in \mathcal{V}$

$$\phi_i(k+1) = \phi_i(k) + \rho \sum_{j \in N(i)} (\mathbf{u}_i(k) - \mathbf{u}_j(k)) \tag{32}$$

$$\Gamma_i(k+1) = \arg \min_{\beta_i, \delta_i, \mathbf{u}_i} \; f_i(\beta_i) + g_i(\delta_i) + \phi_i^T \mathbf{u}_i + \rho \sum_{j \in N(i)} \|\mathbf{u}_i - \frac{\mathbf{u}_i(k) + \mathbf{u}_j(k)}{2}\|_2^2 \tag{33}$$

subject to (28) − (29)

5: **Set** $k = k + 1$
6: **until** $\sum_{i=1}^{n} \sum_{j \in N(i)} \|\mathbf{u}_i(k) - \mathbf{u}_j(k)\| \leq \eta$, for $\eta$ arbitrarily small.

$\overline{\delta} = 0.9$, $\underline{\delta} = 0.5$. These parameters were chosen in such a way as to ensure the DFE is unstable in the absence of any investment in vaccines and/or antidotes, and stable otherwise. We normalize the investment in vaccines and antidotes using the following quasi-convex functions

$$f_i(\beta_i) = \frac{\beta_i^{-1} - \overline{\beta}_i^{-1}}{\underline{\beta}_i^{-1} - \overline{\beta}_i^{-1}}, \quad g_i(\delta_i) = \frac{(1-\delta_i)^{-1} - (1-\underline{\delta}_i)^{-1}}{(1-\overline{\delta}_i)^{-1} - (1-\underline{\delta}_i)^{-1}}. \tag{34}$$

The functions are such that for $\beta_i = \overline{\beta}_i$, $f_i(\overline{\beta}_i) = 0$ and for $\beta_i = \underline{\beta}_i$, $f_i(\underline{\beta}_i) = 1$. Similarly, for $\delta_i = \overline{\delta}_i$, $g_i(\overline{\delta}_i) = 1$ and for $\delta_i = \underline{\delta}_i$, $g_i(\underline{\delta}_i) = 0$. The Lagrangian penalty parameter $\rho$ was chosen to be 4.

We illustrate the distributed solution on an 8-node directed network, where the probability of a directed edge between two points is 0.32. The values of $\underline{\delta} = 0.025$ and $\overline{\delta} = 0.750$ and $\underline{\beta} = 0.1142$ and $\overline{\beta} = 0.4393$. Since a feasibility constraint is $\rho(BA - D) < 1$, the upper and lower bounds of $\beta_i$ and $\delta_i$ are such that[2] when $\beta_i = \underline{\beta}$ and $\delta_i = \underline{\delta}$, across all agents in the network, $\rho(BA - D) = 0.4600$. Further, when $\beta_i = \overline{\beta}$ and $\delta_i = \overline{\delta}$, $\rho(BA - D) = 1.6$. And when $\beta_i = \underline{\beta}$ and $\delta_i = \overline{\delta}$, $\rho(BA - D) = 1.04$. Finally, when $\beta_i = \overline{\beta}$ and $\delta_i = \underline{\delta}$, $\rho(BA - D) = 1.02$. With the above parameters, the optimal solution to the centralized problem using the cost functions in (34) was 1.9731. As can be seen in Figure 1, we observe convergence of the total investment in the distributed solution to that obtained in the centralized solution. Figure 2 illustrates the convergence of errors in the local estimates $\mathbf{u}_i$ across all agents in the network. The plot shows that the local estimates $\mathbf{u}_i$ of the global variable at node each node $v_i$ converges to those of their neighbors and the agents reach consensus in their estimates. As illustrated in Figure 3, the dual variables $\phi_i$ for all nodes also converge.

We illustrate the solution on a fairly larger strongly connected network comprising 20 nodes. As was done earlier, the values $\underline{\delta} = 0.025$, $\overline{\delta} = 0.750$, $\underline{\beta} = 0.0641$ and $\overline{\beta} = 0.2464$ are chosen such that when $\beta_i = \underline{\beta}$ and $\delta_i = \underline{\delta}$, across all agents in the network, $\rho(BA - D) = 0.3500$. Further, when $\beta_i = \overline{\beta}$ and $\delta_i = \overline{\delta}$, $\rho(BA - D) = 2.000$. And when $\beta_i = \underline{\beta}$ and $\delta_i = \overline{\delta}$, $\rho(BA - D) = 1.0750$. Finally, when $\beta_i = \overline{\beta}$ and $\delta_i = \underline{\delta}$, $\rho(BA - D) = 1.2750$. The total investment obtained in the centralized solution for the 20-node network was 4.8868. In Figure 4, we find that the total investment obtained from the distributed solution converges to that obtained in the centralized solution. Further, the local estimates $\mathbf{u}_i$ of the nodes reach consensus as illustrated in Figure 5, where the errors in the estimates converge to zero. Finally, the dual variables, do indeed, converge as illustrated in Figure 6.

### A. On convergence of the D-ADMM Algorithm 1

The ADMM algorithm applied to distributed optimization problems have typically considered distributed problems where the only constraint is consensus in the local estimates of the agents; for example [21, 22]. With just a consensus constraint and smoooth, differentiable convex cost, explicit, analytical computation of the local decision variables is possible. This allows for analytical expression of the optimal iterates, which enables convergence rate analysis of the algorithm. The problem considered in this paper, however, is a constrained optimization problem where, in addition to the consensus constraint, each agent also has local constraints to satisfy to achieve a feasible solution at each iteration of the algorithm as specified in (28). This informed the use of numerical solvers for implementing Algorithm 1.

**Remark 1.** *It is known that ADMM algorithm converges when applied to convex problems [21, 22]. The convex characterization of our resource allocation problem via GP presented in Section III guarantees that the D-ADMM solution in Algorithm 1 converges. The use of numerical solvers in computing the local optimal solution at each node (in line 4 of Algorithm 1) makes it difficult carrying out a convergence rate analysis.*

## VI. Summary

In this paper, we proposed a fully distributed solution to the problem of optimally allocating vaccine and antidote investment to control an epidemic outbreak in a networked population at a desired rate. The proposed solution was a D-ADMM algorithm, enabling each node to locally compute it's optimum investment in vaccine and antidotes needed to globally contain the spread of an outbreak, via local exchange of information with its neighbors. In contrast to existing literature, our problem is a constrained optimization problem associated with a directed network comprising non-identical agents. Since numerical solvers are used to solve convex subproblems at each node, a convergence rate analysis of the D-ADMM algorithm is not presented. However, it is known that the ADMM algorithm converges for convex problems [21]; further, illustration of our results via simulations in Figures 1–6 show that that the D-ADMM algorithm converges. The proposed distributed solution to the vaccine and antidote

[2]Specifically, the upper and lower bounds of $\beta_i$ and $\delta_i$ were chosen as follows and based on the spectrum of $A$. We have $\overline{\delta} = 0.8$, $\underline{\delta} = 3.9 \times \frac{2}{10}$, $\tau_c = (2/10)/\lambda_1(A)$, $\overline{\beta} = 4\tau_c$, $\underline{\beta} = 30\%\overline{\beta}$, all chosen in a way that ensures infeasibility of a solution when all agents are assigned the maximum or minimum possible infection or recovery rates.



Authorized licensed use limited to: University of Pennsylvania. Downloaded on October 29,2026 at 15:24:04 UTC from IEEE Xplore. Restrictions apply.

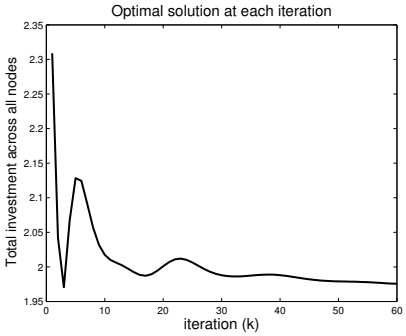



Fig. 1. Convergence of Optimal solution

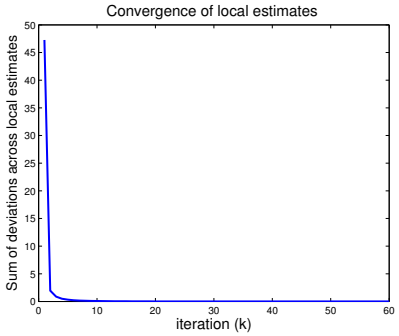



Fig. 2. Error convergence of global estimates

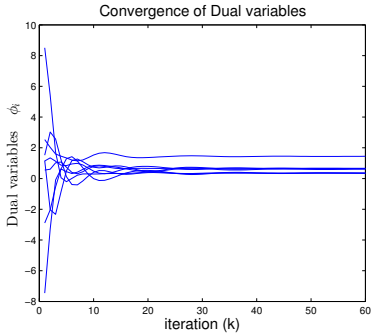



Fig. 3. Convergence of dual variable

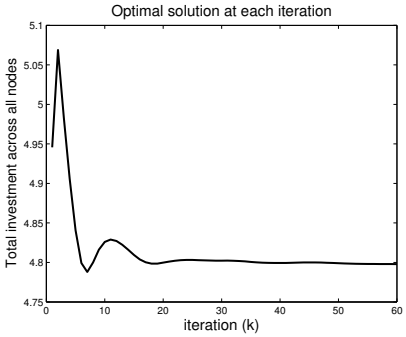



Fig. 4. Convergence of Optimal solution

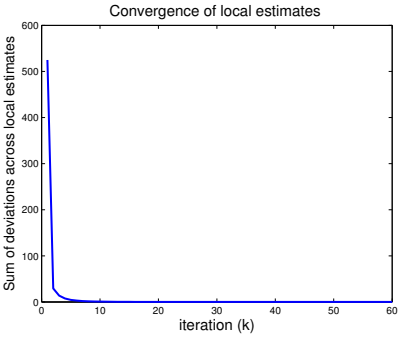



Fig. 5. Error convergence of global estimates

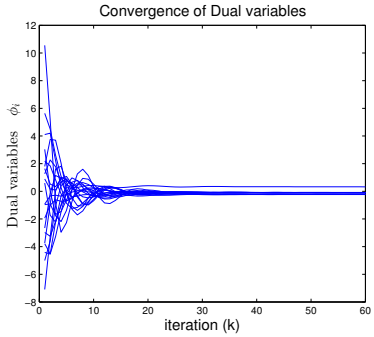



Fig. 6. Convergence of dual variable

allocation problem for epidemic control presents a framework to contain outbreak in the absence of a central social planner.

## REFERENCES


[1] Victor M Preciado, Michael Zargham, Chinwendu Enyioha, Ali Jadbabaie, and George Pappas. Optimal resource allocation for network protection: A geometric programming approach. *IEEE Transactions on Control of Network Systems*, 1(1):99–108, March 2014.

[2] Raoul J de Groot, Susan C Baker, Ralph S Baric, Caroline S Brown, Christian Drosten, Luis Enjuanes, Ron AM Fouchier, Monica Galiano, Alexander E Gorbalenya, Ziad A Memish, et al. Middle east respiratory syndrome coronavirus (mers-cov): announcement of the coronavirus study group. *Journal of virology*, 87(14):7790–7792, 2013.

[3] Andrea Du Toit. Ebola virus in west africa. *Nature Reviews Microbiology*, 12(5):312–312, 2014.

[4] Y. Wang, D. Chakrabarti, C. Wang, and C. Faloutsos. Epidemic spreading in real networks: An eigenvalue viewpoint. In *Proc. 22nd Int. Symp. Reliable Distributed Systems*, pages 25–34, 2003.

[5] A.J. Ganesh, L. Massoulie, and D.F. Towsley. The effect of network topology on the spread of epidemics. In *IEEE INFOCOM 2005*, volume 2, pages 1455–1466, 2005.

[6] Piet Van Mieghem, Jasmina Omic, and Robert Kooij. Virus spread in networks. *Networking, IEEE/ACM Transactions on*, 17(1):1–14, 2009.

[7] Reuven Cohen, Shlomo Havlin, and Daniel Ben-Avraham. Efficient immunization strategies for computer networks and populations. *Physical Review Letters*, 91(24):247901, 2003.

[8] Fan Chung, Paul Horn, and Alexander Tsiatas. Distributing antidote using pagerank vectors. *Internet Mathematics*, 6(2):237–254, 2009.

[9] Yan Wan, Sandip Roy, and Ali Saberi. Designing spatially heterogeneous strategies for control of virus spread. *IET Systems Biology*, 2(4):184–201, 2008.

[10] Victor Preciado, Michael Zargham, Chinwendu Enyioha, Ali Jadbabaie, and George Pappas. Optimal vaccine allocation to control epidemic outbreaks in arbitrary networks. In *IEEE Conference on Decision and Control*, 2013.

[11] Angelia Nedic and Asuman Ozdaglar. Distributed subgradient methods for multi-agent optimization. *Automatic Control, IEEE Transactions on*, 54(1):48–61, 2009.

[12] John Nikolas Tsitsiklis. Problems in decentralized decision making and computation. Technical report, DTIC Document, 1984.

[13] Anders Rantzer. Distributed control of positive systems. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pages 6608–6611. IEEE, 2011.

[14] Ermin Wei and Asuman E Ozdaglar. Distributed alternating direction method of multipliers. In *CDC*, pages 5445–5450, 2012.

[15] Stojan Trajanovski, Yezekael Hayel, Eitan Altman, Huijuan Wang, and Piet Van Mieghem. Decentralized protection strategies against sis epidemics in networks. *arXiv preprint arXiv:1409.1730*, 2014.

[16] Chinwendu Enyioha. *A Convex framework for Epidemic Control in Networks*. PhD thesis, University of Pennsylvania, Philadelphia, PA, USA, 2014.

[17] Subhonmesh Bose, Elizabeth Bodine-Baron, Babak Hassibi, and Adam Wierman. The cost of an epidemic over a complex network: A random matrix approach. *arXiv preprint arXiv:1309.2236*, 2013.

[18] Stephen Boyd, Seung-Jean Kim, Lieven Vandenberghe, and Arash Hassibi. A tutorial on geometric programming. *Optimization and engineering*, 8(1):67–127, 2007.

[19] Ioannis D Schizas, Alejandro Ribeiro, and Georgios B Giannakis. Consensus in ad hoc wsns with noisy links part i: Distributed estimation of deterministic signals. *Signal Processing, IEEE Transactions on*, 56(1):350–364, 2008.

[20] Dimitri P Bertsekas. Constrained optimization and lagrange multiplier methods. *Computer Science and Applied Mathematics, Boston: Academic Press, 1982*, 1, 1982.

[21] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[22] Qing Ling and Alejandro Ribeiro. Decentralized dynamic optimization through the alternating direction method of multipliers. In *Signal Processing Advances in Wireless Communications (SPAWC), 2013 IEEE 14th Workshop on*, pages 170–174. IEEE, 2013.




Authorized licensed use limited to: University of Pennsylvania. Downloaded on October 29,2026 at 15:24:04 UTC from IEEE Xplore. Restrictions apply.