# Optimal Strategies for Dynamic Weight Selection in Consensus Protocols in the Presence of an Adversary

Mahmoud El Chamie[1] Tamer Başar[2]

***Abstract*— In this paper, we consider optimal design strategies in consensus protocols for networks vulnerable to adversarial attacks. First we study dynamic (multi-stage) weight selection optimal control for consensus protocols. For the general (multi-stage) case, the solution exists but can rarely be expressed in closed-form. In view of this, we apply optimization techniques to obtain a locally (and possibly globally) optimizing feasible control path. For the one-stage case, however, we obtain a closed-form solution for the optimal control and provide sufficient conditions for the existence of a control that makes the system reach consensus in only one iteration. We then consider a game theoretical model for the problem of a network with an adversary corrupting the control signal with noise. We derive the optimal strategies for both players (the adversary and the network designer) of the resulting game using a saddle point equilibrium (SPE) solution in mixed strategies.**

## I. Introduction

Consensus algorithms are gaining a lot of attention in recent years. These algorithms contribute, as a fundamental block, to the design of many applications such as formation control [1], load balancing [2], distributed state estimation in power systems [3], and data fusion in sensor networks [4]. Consensus in networks can be subject to changing network topology [5], quantization in communication [6], [7], communication delays [8], and adversarial intervention [9].

In consensus algorithms, nodes execute update rules to reach consensus based on neighbor to neighbor weighted average linear iterations. As in any protocol, some parameters (e.g., the weights) can be tuned for faster convergence. For instance, [10] formulates a semi-definite program (SDP) for a *fixed* weight selection algorithm to achieve fast convergence of consensus protocols independent of initial nodes' values, and a distributed implementation for an approximation of the SDP is given in [11]. Another approach is to design *time-varying* weights, for example [12], [13] study finite-time consensus by arbitrary time-varying weights chosen at the time of design using matrix factorization techniques. Reference [14] considers dynamic weights for least mean square design in correlated or uncorrelated initial node values. For a complete overview of consensus protocols, we refer the reader to [6], [15], and the references therein.

[1]INRIA Sophia Antipolis-Méditerranée, 2004 route des Lucioles - BP 93. 06902 Sophia Antipolis Cedex, France. Email: mahmoud.el_chamie@inria.fr. Research of this author was partially supported by the European Commission within the framework of the CONGAS project FP7-ICT-2011-8-317672.

[2]Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA. Email: basar1@illinois.edu. Research of this author was partially supported by the AFOSR MURI Grant FA9550-10-1-0573.

Further, networks can be susceptible to attacks from adversaries willing to drive the system away from consensus. There are different types of adversaries that can harm the network. For example compromised strategic nodes (like faulty nodes or stubborn ones [16], [17]) can harm the state of the network. Other types of strategic intervention include adversaries that cut communication links or insert noise signals in the agents' interaction protocol [9]. Yet another type of adversaries inject false data (collected by nodes) into the system, which bypass bad-data detection mechanisms. False data injections are known as stealth attacks and are widely studied for the security of state estimation in electric power networks [3], [18]. In order to mitigate the effect of an adversary, security procedures should be taken into account in the design of optimal strategies in consensus protocols.

Our present work shares with this set of references the same objectives of designing time-varying weights for faster consensus and studying optimal strategies for networks that are vulnerable to attacks. In the first part we study time-varying weights for consensus protocols within the framework an optimal control formulation. We apply optimization techniques to obtain a locally (and possibly globally) optimizing feasible control path and provide necessary and sufficient conditions for the existence of a control that makes the system reach consensus in only one iteration. The difference with previous related work is that in this paper we consider the initial values in our *dynamic* weight design. In the second part we study adversaries that can compromise these weights. We propose a game theoretical framework for an adversary that can add noise to the weights to drive the system away from consensus. We derive the optimal strategies using a saddle point equilibrium (SPE) solution in mixed strategies for both players (the adversary and the network designer) of the resulting game.

## II. Problem Formulation

A network is comprised of nodes (or agents) and communication links that allow these nodes to share information and resources. Consider a network where each of the $n$ nodes in a network has a scalar $x_i(k) \in \mathbb{R}$ called node's *state variable* that is located (and can be updated) in its local memory where $k$ is a discrete time index where $x_i(0)$ is its initial value. Average consensus protocols are algorithms where nodes, subject to some given communication constraints, reach consensus on the average of all initial values ($x_{ave} := \frac{1}{n}\sum_i x_i(0)$). We model the network as an *undirected connected* graph $G = (V, E)$ where $V = 1, \ldots, n$ is the set of vertices (nodes) and $E = 1, \ldots, m$ is the set

of edges (links). We use the notation $s \sim (ij)$ to indicate that the vertices $i$ and $j$ are incident to the link $s$. One class of algorithms to achieve consensus is obtained by nodes updating their values in a synchronized and iterative way as follows:

$$\mathbf{x}(k+1) = W(k)\mathbf{x}(k), \tag{1}$$

where $\mathbf{x}(k)$ is the state vector having $x_i(k)$ for $i = 1, \ldots, n$ as its elements and $W(k)$ is the weight matrix at iteration $k$ satisfying $w_{ij} = 0$ if $(ij) \notin E$.

Under some conditions on the weights $W(k)$, the values at the nodes are guaranteed to converge asymptotically to the average

$$\lim_{k\to\infty} \mathbf{x}(k) = \bar{\mathbf{x}},$$

where $\bar{\mathbf{x}} = x_{ave}\mathbf{1}$ and $\mathbf{1}$ is the vector of all ones. One such set of conditions is given in [10] with fixed weights (i.e., $W(k) = W\ \forall k$):

$$\mathbf{1}^T W = \mathbf{1}^T, \quad W\mathbf{1} = \mathbf{1}, \quad \rho(W - \frac{1}{n}\mathbf{1}\mathbf{1}^T) < 1,$$

where $\rho(.)$ is the largest eigenvalue in magnitude of a matrix. By the first condition, the average in the network is conserved, namely

$$\mathbf{1}^T\mathbf{x}(k) = \mathbf{1}^T\mathbf{x}(0) = nx_{ave}\ \forall k, \tag{2}$$

the second ensures stability, and the last condition guarantees contraction on the weight matrix. At any iteration $k$, we can define the squared error $L_k$ from consensus as follows:

$$L_k = ||\mathbf{x}(k) - \bar{\mathbf{x}}||_2^2 = \mathbf{y}_k^T\mathbf{y}_k, \tag{3}$$

where $\mathbf{y}_k = \mathbf{x}(k) - \bar{\mathbf{x}}$.

In this paper, we design time-varying weight matrices $W(k)$ such that consensus forms in the least number of iterations (achieving faster convergence) under the criterion of minimum squared error. Our work differs from the earlier work in the literature in that we design the weights depending on the initial values, i.e., $W(k) = W(k, \mathbf{x}(0))$.

## III. Optimal weight selection on undirected graphs

Toward the goal stated above, since we are dealing with an undirected graph, we consider the following properties for the weight matrix for all $k$:

$$W(k) = W(k)^T \text{ and } W(k)\mathbf{1} = \mathbf{1}. \tag{4}$$

Therefore, equation (2) is satisfied for all $k$ and the average is conserved. Moreover, we can consider a vector $\mathbf{u}_k \in \mathbb{R}^m$ as the control variable that represents the weights on the undirected links (each link $s \sim (ij)$ is given a control $u_s^{(k)}$). At stage $k$, the network designer will select a control $\mathbf{u}_k$. In particular, due to equation (4) we can write the weight matrix as a function of the control vector as follows:

$$W(k) = I_n - Q\text{diag}(\mathbf{u}_k)Q^T, \tag{5}$$

where $I_n$ is the $n$ by $n$ identity matrix, $Q$ is an $n \times m$ incidence matrix of the graph $G$ (each column corresponds to an edge such that if column $s \sim (ij) \in E$, then $Q_{is} = +1$ and $Q_{js} = -1$ while all other elements of the column are zeros).

For any iteration $k$, the square error $L_k$ metric measures the distance of the system from the average. Since the goal is to reach faster the consensus fast, cost is assigned only to the last stage. The optimal control problem is then given as follows:

$$\begin{aligned} &\underset{\mathbf{u}_0,\ldots,\mathbf{u}_{N-1}}{\text{argmin}} \quad L_N, \quad \text{subject to} \\ &\mathbf{y}_{k+1} = \mathbf{y}_k - Q\text{diag}(\mathbf{u}_k)Q^T\mathbf{y}_k, \text{ for } k = 0, \ldots, N-1, \end{aligned} \tag{6}$$

where $N$ is the number of stages in this optimization. We first show that an optimal control exists.

### A. Existence of a Solution

Let $J_N := \mathbf{x}(N)^T\mathbf{x}(N)$, and note that the cost function of the optimization problem can be written as

$$\begin{aligned} L_N = \mathbf{y}_N^T\mathbf{y}_N &= \mathbf{x}(N)^T\mathbf{x}(N) - 2\bar{\mathbf{x}}^T\mathbf{x}(N) + \bar{\mathbf{x}}^T\bar{\mathbf{x}} \\ &= J_N - 2x_{ave}\mathbf{1}^T\mathbf{x}(N) + nx_{ave}^2 = J_N - nx_{ave}^2. \end{aligned}$$

Then minimizing $L_N$ is equivalent to minimizing the function $J_N$ because the term $nx_{ave}^2$ depends only on the initial values. Let us define the product matrix $U_{(k_1:k_2)}$ as follows:

$$U_{(k_1:k_2)} = \begin{cases} W(k_1)W(k_1+1)\ldots W(k_2) & \text{if } k_1 < k2 \\ W(k_1)W(k_1-1)\ldots W(k_2) & \text{if } k_1 > k2 \\ W(k_1) & \text{if } k_1 = k2. \end{cases}$$

To show that an optimal control $(\mathbf{u}_k^*,\ k = 0, \ldots, N-1)$ exists, we write the optimization as an unconstrained one:

$$\underset{\mathbf{u}_0,\ldots,\mathbf{u}_{N-1}}{\text{argmin}}\ f(\mathbf{u}_0, \ldots, \mathbf{u}_{N-1}) \tag{7}$$

where

$$\begin{aligned} f(\mathbf{u}_0, \ldots, \mathbf{u}_{N-1}) &= J_N = \mathbf{x}(N)^T\mathbf{x}(N) \\ &= \mathbf{x}(0)^T U_{(N-1,0)}^T U_{(N-1,0)}\mathbf{x}(0). \end{aligned} \tag{8}$$

Since the elements of the matrix $U_{(N-1,0)}$ are linear in the control variables, and $U_{(N-1,0)}^T U_{(N-1,0)}$ is a positive semi-definite matrix, $f(.)$ is a quadratic function and bounded from below, and hence there exists at least one control $(\mathbf{u}_k^*,\ k = 0, \ldots, N-1)$ that globally minimizes $f$.

### B. Necessary Conditions

To find necessary conditions for the optimal control, we apply the maximum principle [19, p. 24] to problem (6). For $k = 0, \ldots, N-1$, the system equation, performance index, and Hamiltonian are given as:

- System equation: $\mathbf{y}_{k+1} = \mathbf{y}_k - Q\text{diag}(\mathbf{u}_k)Q^T\mathbf{y}_k$,
- Performance index: $L_N = \mathbf{y}_N^T\mathbf{y}_N$,
- Hamiltonian:

$$H^k = \lambda_{k+1}^T\left(\mathbf{y}_k - Q\text{diag}(\mathbf{u}_k)Q^T\mathbf{y}_k\right), \tag{9}$$

where $\lambda_{k+1}$ is the costate variable corresponding to iteration $k$.

Then, the costate equation and the associated boundary condition are:

- Costate: $\lambda_k = \frac{\partial H^k}{\partial \mathbf{y}_k} = \left(I_n - Q\text{diag}(\mathbf{u}_k)Q^T\right)\lambda_{k+1}$,
- Boundary condition: $\lambda_N = \mathbf{y}_N$.

Any optimal control should minimize the Hamiltonian [19]. Since the Hamiltonian is linear in the *unconstrained* control variables, if any coefficient of a control variable in (9) is nonzero, the optimal control would be unbounded. But an optimal control exists as we have already shown, so all the coefficients of the control variables in (9) are necessarily equal to zero, i.e.,

$$\frac{\partial H^k}{\partial \mathbf{u}_k} = \left(Q^T\mathbf{y}_k\right) \odot \left(Q^T\lambda_{k+1}\right) = \mathbf{0}, \quad \text{for } k = 0, \ldots, N-1, \tag{10}$$

where $\odot$ is the element-wise product of the vectors and $\mathbf{0}$ is the vector of all zeros. Equation (10) provides necessary conditions for a controller to minimize (8).

For example, when $N = 1$, the necessary conditions (10) reduce to, $\left(Q^T\mathbf{y}_0\right) \odot \left(Q^T\mathbf{y}_1\right) = \mathbf{0}$, i.e.,

$$(x_i(0) - x_j(0))(x_i(1) - x_j(1)) = 0 \quad \text{for all } (ij) \in E. \tag{11}$$

Let $G' = (V, E')$ be a sub-graph of $G$ defined on the same set of vertices, $V$, and with links $E' \subseteq E$ such that $(ij) \in E'$ if $(ij) \in E$ and $x_i(0) - x_j(0) \neq 0$. Then we have:

**Proposition 1.** *If $G' = (V, E')$ is connected, then any optimal control $\mathbf{u}^*$ drives the system to consensus in one iteration, i.e.,*

$$\bar{\mathbf{x}} = \left(I_n + Q\mathit{diag}(\mathbf{u}^*)Q^T\right)\mathbf{x}(0).$$

*Proof.* From (11), $x_i(1) = x_j(1) \quad \forall (ij) \in E'$. If $G'$ is connected, then there is a path in $E'$ between any two vertices, and thus $x_i(1) = x_j(1)\ \forall i, j \in V$. Using also the fact that the average is conserved (by (2)), we get $x_i(1) = x_{ave} \quad \forall i \in V$. □

### C. Locally Optimal Solution

In the general case, the optimization problem (7) is computationally hard because the function $f(\mathbf{u}_0, \ldots, \mathbf{u}_{N-1})$ is not convex (it is convex in the variables of each stage, $\mathbf{u}_k$, but not jointly convex). We therefore turn our attention to locally optimal solutions, and to obtain such a solution we apply the gradient method to (8).

**Proposition 2.** *Let $f(\mathbf{u}_0, \ldots, \mathbf{u}_{N-1})$ be given by* (8)*. Then, for $k = 0, \ldots, N-1$, the gradient $g_l^{(k)} = \frac{\partial f}{\partial u_l^{(k)}}$ of the function $f$ with respect to its variables $u_l^{(k)}$ where $u_l^{(k)}$ is the $l$-th element of the vector $\mathbf{u}_k$ corresponding to link $(ij)$ ($l \sim (ij)$) at stage $k$, is given as follows:*

$$\begin{aligned} g_l^{(k)} = 2[&(A_kW(k)B_k)_{ij} + (A_kW(k)B_k)_{ji} \\ &- (A_kW(k)B_k)_{ii} - (A_kW(k)B_k)_{jj}], \end{aligned} \tag{12}$$

*where $A_k$ and $B_k$ are as follows:*[1]

$$\begin{aligned} A_k &= U_{(N-1:k+1)}^T U_{(N-1:k+1)}, \text{ if } N-1 \geq k+1, \\ B_k &= \left(U_{(k-1:0)}\mathbf{x}(0)\right)\left(U_{(k-1:0)}\mathbf{x}(0)\right)^T, \text{ if } k-1 \geq 0. \end{aligned} \tag{13}$$

[1]Where $A_k = I_n$ if $N-1 < k+1$ and $B_k = \mathbf{x}(0)\mathbf{x}(0)^T$ if $k-1 < 0$.

*Proof.* By using the commutative property of the trace operator (i.e., $Tr(XY) = Tr(YX)$ for any conformable matrices $X$ and $Y$), $f(.)$ can be written for $k = 0, \ldots, N-1$:

$$\begin{aligned} f(\mathbf{u}_0, \ldots, \mathbf{u}_{N-1}) &= \mathbf{x}(0)^T U_{(N-1,0)}^T U_{(N-1,0)}\mathbf{x}(0) \\ &= Tr\left(W(k)^T A_k W(k) B_k\right), \end{aligned} \tag{14}$$

where $A_k$ and $B_k$ are given by (13) and are independent of the variables of stage $k$ (i.e., $\frac{\partial (A_k)_{st}}{\partial u_l^{(k)}} = \frac{\partial (B_k)_{st}}{\partial u_l^{(k)}} = 0 \quad \forall s, t \in V$, and $k = 0, \ldots, N-1$). From matrix calculus, if $h(W) = Tr(W^T AWB)$, then $\frac{\partial h}{\partial w_{ij}} = 2\left(AWB\right)_{ij}$, and since $W = I_n - Q\text{diag}(\mathbf{u})Q^T$, then for any $u_l$ such that $l \sim (ij)$ we have

$$\frac{\partial w_{st}}{\partial u_l} = \begin{cases} +1 & \text{if } (s = i \text{ and } t = j) \text{ or } (s = j \text{ and } t = i) \\ -1 & \text{if } (s = i \text{ and } t = i) \text{ or } (s = j \text{ and } t = j) \\ 0 & \text{else.} \end{cases} \tag{15}$$

Thus,

$$\begin{aligned} \frac{\partial h}{\partial u_l} &= \sum_{s,t}\left(\frac{\partial h}{\partial w_{st}}\right)\frac{\partial w_{st}}{\partial u_l} = 2\sum_{s,t}\left(AWB\right)_{st}\frac{\partial w_{st}}{\partial u_l} \\ &= 2\left[(AWB)_{ij} + (AWB)_{ji} - (AWB)_{ii} - (AWB)_{jj}\right]. \end{aligned} \tag{16}$$

We can apply equation (16) to every stage separately and this ends the proof. □

Let us stack up all the elements $u_l^{(k)}$ in one vector $\mathbf{w}$, and also stack up all the elements $g_l^{(k)}$ in one vector $\mathbf{g}$.

**Proposition 3.** *Consider the following gradient iterative procedure*

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \gamma_t \mathbf{g}^{(t)},$$

*where $\gamma_t = \frac{1}{(1+t)||\mathbf{g}||}$ is the stepsize and $\mathbf{w}^{(0)} = \mathbf{0}$. Then the elements $u_l^{(k)}$ of the vector $\mathbf{w}$ converge to a locally minimizing solution of the optimization problem* (6)*.*

*Proof.* The given procedure is a standard (sub-)gradient method for optimization and the convergence has been widely studied under the diminishing step-size rule: $\lim_{t\to\infty}\gamma_t = 0$ and $\sum_{t=1}^{\infty}\gamma_t = \infty$ (see [20]). □

### D. Closed-Form Solution for the One-Stage Problem

Consider now the case $N = 1$, that is with only one stage. Then the control would be a single vector $\mathbf{u}$ where each component is the weight for the corresponding edge. The optimization problem in this case is convex:

$$\mathbf{u}_S = \underset{\mathbf{u}}{\text{argmin}} \quad f(\mathbf{u}), \tag{17}$$

where $\mathbf{u}_S$ is the solution set (possibly an infinite set) and

$$\begin{aligned} f(\mathbf{u}) &= \mathbf{x}(0)^T(I_n - Q\text{diag}(\mathbf{u})Q^T)(I_n - Q\text{diag}(\mathbf{u})Q^T)\mathbf{x}(0) \\ &= ||\mathbf{x}(0) - Q\text{diag}(\mathbf{u})Q^T\mathbf{x}(0)||^2 \\ &= ||\mathbf{x}(0) - Q\text{diag}(Q^T\mathbf{x}(0))\mathbf{u}||^2 = ||D\mathbf{u} - \mathbf{b}||^2, \end{aligned}$$

where

$$D = Q\text{diag}(Q^T\mathbf{x}(0)), \text{ and } \mathbf{b} = \mathbf{x}(0). \tag{18}$$

The problem is then reduced to a least squares approximation problem, where any element in the solution set $\mathbf{u}_S$ satisfies what is known as the normal equations:

$$D^T D\mathbf{u} = D^T\mathbf{b}, \quad \forall \mathbf{u} \in \mathbf{u}_S.$$

Moreover, $\mathbf{u}_S$ is not empty, with at least one solution $\hat{\mathbf{u}} = D^+\mathbf{b}$, where $D^+$ is the pseudo inverse of $D$ that can be obtained using the singular value decomposition of $D$. If $D^T D$ is a positive definite matrix, then $D^+ = (D^T D)^{-1}D^T$ and $\hat{\mathbf{u}}$ is the unique solution to the least squares problem. We denote by $S$ the minimum value of the function $f(\mathbf{u})$:

$$S = f(\hat{\mathbf{u}}) = ||(DD^+ - I)\mathbf{b}||^2. \tag{19}$$

## IV. Network with Adversary in Discrete Time

Suppose that there is an adversary that can add noise onto the weights of the links. The adversary's objective is to drive the system away from consensus. Considering only one stage optimization ($N = 1$), the state equation would become

$$\begin{aligned}\mathbf{x}(1) &= W(\mathbf{u},\mathbf{v})\mathbf{x}(0)\\ &= (I_n - Q\text{diag}(\mathbf{u}+\mathbf{v})Q^T)\mathbf{x}(0),\end{aligned} \tag{20}$$

where $W(\mathbf{u},\mathbf{v})$ is the weight matrix that depends on the control $\mathbf{u} \in U_1 = \mathbb{R}^m$ and the noise of the adversary $\mathbf{v} \in U_2 = \{y; y \in \mathbb{R}^m, ||y|| \leq C\}$, where $C$ is a given positive constant and can be seen as the power constraint of the adversary (the larger $C$ the more powerful is the adversary). The cost function is

$$\begin{aligned}J(\mathbf{u},\mathbf{v}) &= \mathbf{x}(1)^T\mathbf{x}(1)\\ &= ||(I_n - Q\text{diag}(\mathbf{u}+\mathbf{v})Q^T)\mathbf{x}(0)||^2\\ &= ||D(\mathbf{u}+\mathbf{v}) - \mathbf{b}||^2,\end{aligned} \tag{21}$$

where $D$ and $\mathbf{b}$ are given by (18). The adversary ($\mathbf{v}$) is the maximizer of $J(\mathbf{u},\mathbf{v})$ while the network designer ($\mathbf{u}$) is the minimizer in this zero-sum two-person game.

**Definition 1.** *A pair $(\mathbf{u}^* \in U_1, \mathbf{v}^* \in U_2)$ is a saddle point in pure strategies of $J(\mathbf{u},\mathbf{v})$ if the following holds:*

$$J(\mathbf{u}^*,\mathbf{v}) \leq J(\mathbf{u}^*,\mathbf{v}^*) \leq J(\mathbf{u},\mathbf{v}^*), \text{ for all } (\mathbf{u} \in U_1, \mathbf{v} \in U_2).$$

The *lower value* $\underline{V}$ and the *upper value* $\overline{V}$ of the game are defined by

$$\underline{V} = \sup_{\mathbf{v}\in U_2}\inf_{\mathbf{u}\in U_1} J(\mathbf{u},\mathbf{v}) \text{ , and } \overline{V} = \inf_{\mathbf{u}\in U_1}\sup_{\mathbf{v}\in U_2} J(\mathbf{u},\mathbf{v}).$$

Since the strategy spaces are decoupled, $\underline{V} \leq \overline{V}$. If furthermore $\underline{V} = \overline{V}$, then the common value is called the *value* of the game. Existence of a saddle point guarantees existence of the value [21]. As $J$ is a quadratic function of $\mathbf{u}$, and $J(\mathbf{u},\mathbf{v}) \geq 0$ for all $(\mathbf{u} \in U_1, \mathbf{v} \in U_2)$, then for any given $\mathbf{v} \in U_2$, $J$ attains a minimum on $U_1$ [22]. Moreover, since $U_2$ is compact, and $J$ is a continuous function on its domain of definition, for any given $\mathbf{u} \in U_1$, $J$ attains a maximum on $U_2$ by the Weierstrass Theorem. Therefore, we can replace $\inf_{\mathbf{u}\in U_1}$ by $\min_{\mathbf{u}\in U_1}$ and $\sup_{\mathbf{v}\in U_2}$ by $\max_{\mathbf{v}\in U_2}$ in the definitions of the upper and lower values. In the sequel, we will show that actually the game does not have a value, and hence does not have a saddle point (in pure strategies). It however has a saddle point in mixed strategies (shortly to be defined).

### A. The max-min solution

In the max-min solution, the network designer has access to the strategy played by the adversary.

$$\underset{\mathbf{u}}{\text{argmin}}\ J(\mathbf{u},\mathbf{v}) = \underset{\mathbf{u}}{\text{argmin}}\ ||D(\mathbf{u}+\mathbf{v}) - \mathbf{b}||^2 = D^+\mathbf{b} - \mathbf{v}.$$

Then we have,

$$\max_{\mathbf{v}}\min_{\mathbf{u}} J(\mathbf{u},\mathbf{v}) = \max_{\mathbf{v}} J(D^+\mathbf{b} - \mathbf{v},\mathbf{v}) = \max_{\mathbf{v}} S = S,$$

where $S$ is the value of the one player optimization problem, given by (19) and is independent of $\mathbf{v}$.

### B. The min-max solution

In the min-max solution, the adversary has access to the strategy of the controller. Note that $J$ can be written as:

$$\begin{aligned}J(\mathbf{u},\mathbf{v}) &= ||D(\mathbf{u}+\mathbf{v}) - \mathbf{b}||^2\\ &= \mathbf{b}^T\mathbf{b} + \mathbf{u}^T D^T D\mathbf{u} - 2\mathbf{b}^T D\mathbf{u}\\ &\quad + \mathbf{v}^T D^T D\mathbf{v} + 2\mathbf{v}^T\left(D^T D\mathbf{u} - D^T\mathbf{b}\right).\end{aligned}$$

Consider the following strategy $\mathbf{v}_1$ by the adversary:

$$\begin{cases}\mathbf{v}_1 \in \mathcal{R}(D^T D) \cap U_2 & \text{if } D^T D\mathbf{u} - D^T\mathbf{b} = \mathbf{0}\\ \mathbf{v}_1 = C\frac{(D^T D\mathbf{u} - D^T\mathbf{b})}{||D^T D\mathbf{u} - D^T\mathbf{b}||} & \text{otherwise,}\end{cases} \tag{22}$$

where $\mathcal{R}(D^T D)$ is the range of the matrix $D^T D$. Therefore,

$$\begin{aligned}\min_{\mathbf{u}}\max_{\mathbf{v}} J(\mathbf{u},\mathbf{v}) &\geq \min_{\mathbf{u}} J(\mathbf{u},\mathbf{v}_1)\\ &= \min_{\mathbf{u}}(\underbrace{\mathbf{v}_1^T D^T D\mathbf{v}_1 + 2\mathbf{v}_1^T\left(D^T D\mathbf{u} - D^T\mathbf{b}\right)}_{>0}\\ &\qquad + \mathbf{b}^T\mathbf{b} + \mathbf{u}^T D^T D\mathbf{u} - 2\mathbf{b}^T D\mathbf{u})\\ &> \min_{\mathbf{u}}\left(\mathbf{b}^T\mathbf{b} + \mathbf{u}^T D^T D\mathbf{u} - 2\mathbf{b}^T D\mathbf{u}\right) = S.\end{aligned}$$

Hence, $\max_{\mathbf{v}}\min_{\mathbf{u}} J(\mathbf{u},\mathbf{v}) < \min_{\mathbf{u}}\max_{\mathbf{v}} J(\mathbf{u},\mathbf{v})$, which means that there is no saddle point in pure strategies.

### C. A Saddle-Point Equilibrium (SPE) in Mixed Strategies

Since an SPE does not exist in pure strategies, we allow players to randomize their actions through mixed strategies. A mixed strategy for the network designer is a probability distribution $\mu$ on $U_1$, and we denote the space of all such probability distributions by $M_1$. Similarly, a mixed strategy for the adversary is a probability distribution $\nu$ on $U_2$, and the space of all such probability distributions is denoted by $M_2$. The average cost corresponding to a pair $(\mu \in M_1, \nu \in M_2)$ is given by

$$\bar{J}(\mu,\nu) = \int_{U_1\times U_2} J(\mathbf{u},\mathbf{v})d\mu(\mathbf{u})d\nu(\mathbf{v}).$$

**Definition 2.** *A pair $(\mu^* \in M_1, \nu^* \in M_2)$ is a saddle point equilibrium in mixed strategies if the following holds:*

$$\bar{J}(\mu^*,\nu) \leq \bar{J}(\mu^*,\nu^*) \leq \bar{J}(\mu,\nu^*), \text{ for all } (\mu \in M_1, \nu \in M_2).$$

**Proposition 4.** *Consider the following strategies:*

$$\mu^*(\mathbf{u}) : \mathbf{u} = D^+\mathbf{b} \text{ with probability 1}, \tag{23}$$

*and*

$$\nu^*(\mathbf{v}) : \begin{cases} \mathbf{v} = C\mathbf{p} & \text{with probability 1/2} \\ \mathbf{v} = -C\mathbf{p} & \text{with probability 1/2}, \end{cases} \tag{24}$$

*where* $\mathbf{p}$ *is any unit eigenvector of the matrix* $D^TD$ *corresponding to the largest eigenvalue* $\lambda_{\max}(D^TD)$*. Then the pair* $(\mu^*, \nu^*)$ *is an SPE in mixed strategies.*

*Proof.* Let us recall the cost function:

$$\begin{aligned} J(\mathbf{u}, \mathbf{v}) &= \mathbf{b}^T\mathbf{b} + \mathbf{u}^TD^TD\mathbf{u} - 2\mathbf{b}^TD\mathbf{u} \\ &\quad + \mathbf{v}^TD^TD\mathbf{v} + 2\mathbf{v}^T\left(D^TD\mathbf{u} - D^T\mathbf{b}\right) \\ &= ||D\mathbf{u} - \mathbf{b}||^2 + \mathbf{v}^TD^TD\mathbf{v} + 2\mathbf{v}^T\left(D^TD\mathbf{u} - D^T\mathbf{b}\right). \end{aligned}$$

Then the average cost under the given pair of strategies is,

$$\begin{aligned} \bar{J}(\mu^*, \nu^*) &= ||DD^+\mathbf{b} - \mathbf{b}||^2 + (C\mathbf{p})^TD^TD(C\mathbf{p}) \times (1/2) \\ &\quad + (-C\mathbf{p})^TD^TD(-C\mathbf{p}) \times (1/2) \\ &= S + C^2\lambda_{\max}. \end{aligned} \tag{25}$$

But we have,

$$\begin{aligned} \bar{J}(\mu^*, \nu) &= ||DD^+\mathbf{b} - \mathbf{b}||^2 + \int_{U_2} \nu^TD^TD\nu \, \mathrm{d}\nu(\mathbf{v}) \\ &\leq S + \max_{\mathbf{v}, ||\mathbf{v}|| \leq C} \mathbf{v}^TD^TD\mathbf{v} \\ &= S + C^2\lambda_{\max} = \bar{J}(\mu^*, \nu^*), \end{aligned} \tag{26}$$

$$\begin{aligned} \bar{J}(\mu, \nu^*) &= C^2\lambda_{\max} + \int_{U_1} ||D\mu - \mathbf{b}||^2 \, \mathrm{d}\mu(\mathbf{u}) \\ &\geq C^2\lambda_{\max} + \min_{\mathbf{u}} ||D\mathbf{u} - \mathbf{b}||^2 \\ &= S + C^2\lambda_{\max} = \bar{J}(\mu^*, \nu^*). \end{aligned} \tag{27}$$

Since we have for any pair $(\mu \in M_1, \nu \in M_2)$,

$$\bar{J}(\mu^*, \nu) \leq \bar{J}(\mu^*, \nu^*) \leq \bar{J}(\mu, \nu^*),$$

then $(\mu^*, \nu^*)$ is a saddle point equilibrium. □

**Remark:** The saddle point is not unique, as any $(\mu, \nu)$ where $\mu$ is a point distribution in the set $\mathbf{u}_S$ of (17) (or any distribution on this set due to the interchangeability property of saddle points [21]), and $\nu$ as in (24) where $\mathbf{p}$ is any eigenvector corresponding to $\lambda_{\max}(D^TD)$ (or any distribution on these vectors) is also a saddle point. However, if $D$ is full column rank, and $\lambda_{\max}$ has geometric multiplicity of 1, then the saddle point is unique.

## V. Simulations

### A. Optimal control

We illustrate the results obtained on a numerical example. Given the sample network of Fig. 1 and the initial values, we are interested in selecting the controls on links, $\mathbf{u}_k = (u_{12}^{(k)}, u_{23}^{(k)}, u_{34}^{(k)})^T$, so that the system reaches consensus. We limit the number of stages to $N = 3$ because in that case the diameter is equal to 3 and an optimal control that drives the system to consensus exists.

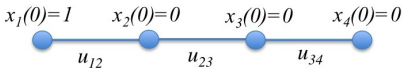


Fig. 1. Network with 4 communicating nodes. $x_i(0)$ is the initial value of node $i$, and $u_{ij}$ is the control value (or weight) of link $(ij)$.

| $k=0$ | | $k=1$ | |
|---|---|---|---|
| $\mathbf{x}(0)$ | $\mathbf{u}_0^*$ | $\mathbf{x}(1)$ | $\mathbf{u}_1^*$ |
| $\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$ | $\begin{pmatrix} 0.8665 \\ 0 \\ 0 \end{pmatrix}$ | $\begin{pmatrix} 0.1335 \\ 0.8665 \\ 0 \\ 0 \end{pmatrix}$ | $\begin{pmatrix} 0.2201 \\ 0.6051 \\ 0 \end{pmatrix}$ |
| $J_0 = \mathbf{x}(0)^T\mathbf{x}(0) = 1$ | | $J_1 = 0.7686$ | |
| $k=2$ | | $k=3$ | |
| $\mathbf{x}(2)$ | $\mathbf{u}_2^*$ | $\mathbf{x}(3)$ | |
| $\begin{pmatrix} 0.2949 \\ 0.1808 \\ 0.5243 \\ 0 \end{pmatrix}$ | $\begin{pmatrix} 0.3934 \\ 0.0708 \\ 0.4768 \end{pmatrix}$ | $\begin{pmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{pmatrix}$ | |
| $J_2 = 0.3945$ | | $J_3 = 0.25$ | |

TABLE I

Optimal control results for the network in Fig. 1.

Table I shows the optimal control $(\mathbf{u}_k^*, k = 0, 1, 2)$ for the given network. The control is obtained by the gradient descent iterative procedure of Proposition 3 where the initial starting point of the gradient was selected 0 on all links of the three stages. The results indicate that with only three iterations, the system reaches consensus. To compare with other weight selection algorithms, we apply the algorithm given in [10] obtained for a related semi-definite program (SDP). That algorithm finds a fixed set of weights for all iterations that guarantee fastest convergence independent of initial values (worst-case analysis). For the network example in Fig. 1, the SDP assigns a value 0.5 to all weights for all iterations, and the resulting state vector after three iterations is $\mathbf{x}_{SDP}(3) = (0.375, 0.375, 0.125, 0.125)^T$, which has a cost of $J_3 = 0.3125$ (thus higher cost than our time-varying weights) and needs an infinite number of iterations to converge. It is worth mentioning that the SDP weights are designed for worst-case node initial values, and thus have the advantage that they guarantee convergence starting from any initial values. However, the optimal control in this paper is designed for a given starting value, and thus if the initial node values change, the control values must be readjusted.

### B. Adversarial intervention

In this subsection, we study the effect of an adversary disrupting the communication on networks with connected random geometric graphs (RGGs) topology where $n$ nodes are thrown uniformly at random on a unit square, and any two nodes within a connectivity radius $r$ are connected by a link (the simulations are done with $r = \sqrt{0.6 \times \frac{\log(n)}{n}}$ given that the graph is connected). RGGs are generally used as models for wireless sensor networks, and disruption of communication can be accomplished by insertion of high intensity

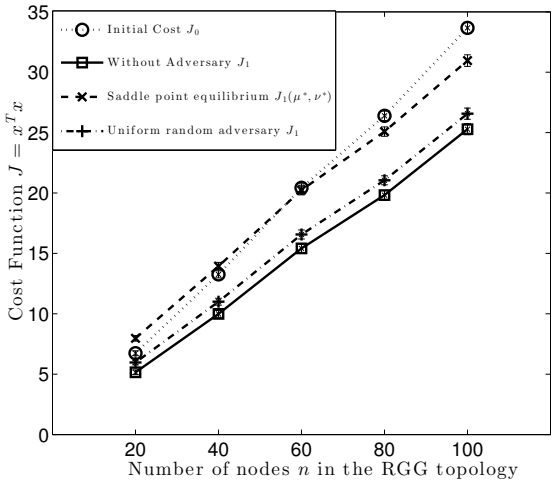


Fig. 2. The cost function due to different adversary settings: absence of adversary, uniform random adversary that adds a random noise to the control values, and saddle point adversary that randomizes its strategy in accordance with the saddle point equilibrium.

signals on communication links. The additive white noise can also be considered as an adversarial input in our settings. We compare the results on different RGGs with different sizes (number of nodes $n$) for $n \in \{20, 40, 60, 80, 100\}$. Fig. 2 depicts the different costs on the resulting network with and without the presence of the adversary, averaged over 150 independent runs. We consider only one-stage games where the initial cost function is given by $J_0 = \mathbf{x}(0)^T\mathbf{x}(0)$. For any node $i$, the initial node value $x_i(0)$ is selected at random uniformly within the interval $[0, 1]$. We assume that the adversary power constraint is $||\mathbf{v}|| \leq 1$ (i.e., $C = 1$). We see from Fig. 2 that the network without an adversary achieves the least cost $J_1$. An adversary selecting uniformly random strategy from the $n$-dimensional unit sphere does not substantially affect the cost; however, an adversary with the same power constraint playing the strategy of the saddle point equilibrium (equation (24)) achieves significantly higher cost than the uniform random adversary (even larger cost than $J_0$ for graphs of $n = 20$ and $n = 40$ nodes).

## VI. Conclusion

In this paper, we have studied a finite-horizon discrete-time optimal control for a network designer to achieve faster consensus given the network structure and the initial node values. The optimal control is obtained using gradient methods. Moreover, we have studied the saddle point equilibrium (SPE) of the consensus problem in the presence of an adversary, and found that an SPE does not exist in pure strategies. Nevertheless, an SPE exists in mixed strategies, where the adversary selects the noise using a randomized strategy, whereas the network designer's strategy is still pure.

For future work, distributed implementation of the optimal control would be an important direction because of the distributed nature of the average consensus protocols. So far, the adversary has access to initial values; it would be interesting to remove the dependence of the strategies on the initial values. Moreover, considering a broader class of adversaries (as malicious and misbehaving nodes, or adversaries that break links) is also one of our future interests.

## References


[1] J. Fax and R. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, 2004.

[2] G. Cybenko, "Dynamic load balancing for distributed memory multiprocessors," *Journal of Parallel and Distributed Computing*, vol. 7, no. 2, pp. 279–301, Oct. 1989.

[3] O. Vukovic and G. Dan, "Security of fully distributed power system state estimation: Detection and mitigation of data integrity attacks," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 7, pp. 1500–1508, July 2014.

[4] K. Avrachenkov, M. El Chamie, and G. Neglia, "A local average consensus algorithm for wireless sensor networks," in *Proceedings of the 2011 International Conference on Distributed Computing in Sensor Systems and Workshops (DCOSS)*, June 2011, pp. 1–6.

[5] W. Ren and R. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 655–661, May 2005.

[6] A. Nedic, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis, "On distributed averaging algorithms and quantization effects," *IEEE Trans. on Automatic Control*, vol. 54, no. 11, pp. 2506–2517, Nov 2009.

[7] M. El Chamie, J. Liu, and T. Başar, "Design and analysis of distributed averaging with quantized communication," in *Proceedings of the 53rd IEEE Conference on Decision and Control CDC 2014 (Los Angeles, California, Dec. 15-17)*, December 2014, p. 6.

[8] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, Sept 2004.

[9] A. Khanafer, B. Touri, and T. Başar, "Robust distributed averaging on networks with adversarial intervention," in *Proceedings of the 52nd IEEE Conference on Decision and Control CDC 2013 (Florence, Italy, Dec. 10-13)*, December 2013.

[10] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems and Control Letters*, vol. 53, pp. 65–78, 2004.

[11] M. El Chamie, G. Neglia, and K. Avrachenkov, "Distributed Weight Selection in Consensus Protocols by Schatten Norm Minimization," INRIA, INRIA Research Report RR-8078, Oct 2012, accepted to IEEE Transactions on Automatic Control as Technical Note. [Online]. Available: http://hal.inria.fr/hal-00738249

[12] C.-K. Ko and X. Gao, "On matrix factorization and finite-time average-consensus," in *Proceedings of IEEE CDC/CCC 2009*, Dec 2009.

[13] J. M. Hendrickx, R. M. Jungers, A. Olshevsky, and G. Vankeerberghen, "Graph diameter, eigenvalues, and minimum-time consensus," *Automatica*, vol. 50, no. 2, pp. 635 – 640, 2014.

[14] V. Schwarz and G. Matz, "Mean-square optimal weight design for average consensus," in *Proceedings of the 2012 IEEE SPAWC*, 2012.

[15] R. Olfati-Saber, J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, Jan 2007.

[16] D. Acemoglu, G. Como, F. Fagnani, and A. Ozdaglar, "Opinion fluctuations and persistent disagreement in social networks," in *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC 2011)*, Dec. 2011, pp. 2347 –2352.

[17] W. Ben-Ameur, P. Bianchi, and J. Jakubowicz, "Robust average consensus using total variation gossip algorithm," in *Proceedings of the 6th International Conference on Performance Evaluation Methodologies and Tools (VALUETOOLS 2012)*. IEEE, Nov. 2012.

[18] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," in *Proceedings of the 16th ACM Conference on Computer and Communications Security*, ser. CCS '09. New York, NY, USA: ACM, 2009, pp. 21–32.

[19] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal control. 3rd ed.* Hoboken, NJ: John Wiley Sons. , 2012.

[20] N. Shor, *Minimization Methods for Non-Differentiable Functions*, ser. Springer Ser. in Computational Mathematics. Springer, Berlin, 1985.

[21] T. Başar and G. J. Olsder, *Dynamic Noncooperative Game Theory*, ser. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, 1999.

[22] T. H. Hildebrandt, "Existence of a minimum of a quadratic function," *The American Mathematical Monthly*, vol. 15, no. 3, pp. 57–59, 1908.