Rapid Transitions with Robust Accelerated Delayed Self Reinforcement for Consensus-based Networks

Anuj Tiwari and Santosh Devasia

Abstract—Rapid transitions are important for quick response of consensus-based, multi-agent networks to external stimuli. While high-gain can increase response speed, potential instability tends to limit the maximum possible gain, and therefore, limits the maximum convergence rate to consensus during transitions. Since the update law for multi-agent networks with symmetric graphs can be considered as the gradient of its Laplacian-potential function, Nesterov-type accelerated gradient approaches from optimization theory, can further improve the convergence rate of such networks. An advantage of the accelerated-gradient approach is that it can be implemented using accelerated delayed-self-reinforcement (A-DSR), which does not require new information from the network nor modifications in the network connectivity. However, the accelerated-gradient approach is not directly applicable to non-symmetric graphs since the update law is not the gradient of the Laplacian-potential function. The main contributions of this work are to (i) extend the accelerated-gradient approach to general graph networks (whose Laplacians have real spectrum) using DSR, and (ii) develop analytical design criteria for a Robust A-DSR approach that maximizes both structural robustness and transition speed. Simulation results are presented to illustrate the performance improvement with the proposed Robust A-DSR of 40% in structural robustness and 50% in convergence rate to consensus, when compared to the case without the A-DSR. Moreover, experimental results are presented that show a similar 37% faster convergence with the Robust A-DSR when compared to the case without the A-DSR.

Index Terms—Consensus control, Multi agent systems, Decentralized control, Multirobot system, Network control.

I. INTRODUCTION

The performance of consensus-based, multi-agent networks, such as the response to external stimuli, depends on rapidly transitioning from one operating point (consensus value) to another, e.g., in flocking of natural systems, [1, 2], as well as engineered systems such as autonomous vehicles, swarms of robots, e.g., [3–5] and other networked systems such as aerospace control [6] microgrids [7, 8], flexible structures [9]. Rapid cohesive transitions, e.g., in the orientation of the agents from one consensus value to another, is seen in flocking behaviour during predator attacks and migration [10, 11]. Thus, there is interest to increase the convergence to consensus for such networked multi-agent systems.

There is a fundamental limit to the achievable rate of convergence using existing neighbor-based update laws for a given network, e.g., of the form

$$\dot{X}[k+1] = \dot{X}[k] + u[k] = \dot{X}[k] - \alpha L \hat{X}[k],$$  \hspace{1cm} (1)

where the current state is $\dot{X}(k)$, the updated state is $\dot{X}[k+1]$, $\alpha$ is the update gain, $L$ is the graph Laplacian, and $k$ represents the time instants $t_k = k\delta_t$. The convergence rate depends on the eigenvalues of the Perron matrix $P = 1 - \alpha L$, which in turn depends on the eigenvalues of the graph Laplacian $L$. For example, if the underlying graph is undirected and connected, it is well known that convergence to consensus can be achieved provided the update gain $\alpha$ is sufficiently small, e.g., [12]. The iteration gain can be selected to maximize the convergence rate, and typically, a larger gain $\alpha$ tends to increase the convergence rate. Nevertheless, for a given graph (i.e., a given graph Laplacian $L$), the range of the acceptable update gain $\alpha$ is limited, which in turn, limits the achievable rate of convergence [13]. Typically, the convergence rate tends to be slow if the number of agent inter-connections is small compared to the number of agents, e.g., [14]. Faster convergence can be achieved using randomized time-varying connections, as shown in, e.g., [14]. The update sequence of the agents can also be arranged to improve convergence, e.g., [15]. The problem is that the graph connectivity might be fixed and therefore the Laplacian $L$ cannot be varied over time. In such cases, with a fixed Laplacian $L$, the need to maintain stability limits the range of acceptable update gain $\alpha$, and therefore, limits the rate of convergence. This convergence-rate limitation motivates ongoing efforts to develop new approaches to improve the network performance, e.g., [16]. Furthermore, in addition to convergence-rate, an important consideration is robustness of the approach, e.g., as studied in [17, 18].

Since the neighbor-based update ($u$ in Eq. (1)) can be obtained from the gradient of the Laplacian potential $\Phi_{G} = X^T L \hat{X}$ for undirected graphs, i.e., $u = -\langle \alpha/2 \rangle \nabla \Phi_{G}$, Nesterov-type accelerated approaches, used to speed up gradient-based optimization [19–23], can be used to improve the convergence rate. Previous works have considered the use of some parts of the accelerated gradients (from optimization theory) for graph-based multi-agent networks. For example, the addition of a momentum term (of the form $\dot{X}[k] - \dot{X}[k-1]$), as in, e.g., [19] in the update law has been shown to improve the response speed under update-bandwidth limits [13, 24]. These works have also shown that the use of such reinforcement can lead to non-diffusive, wave-like response propagation seen in natural systems such as bird flocks [25]. Similarly, the addition of a Nesterov term...
without the momentum term, also referred to as an outdated feedback (of the form \( L(\hat{X}[k] - \hat{X}[k-1]) \), as in e.g., [21]), has been shown to result in faster convergence in [23], [27], and to enable a linear rate of convergence using a time-varying gain in [28]. Time-varying gains, however, require a global resetting of each agent’s gain at start of each transition, which might not be always feasible because the start of a transition might not be known to all agents. The combination of both, the momentum term and the outdated-feedback term, can further improve the convergence rate of cohesion-based networks when compared to the use of either term alone [29], [30]. Note that an advantage of such accelerated-gradient-based approach is that the update can be implemented by using an accelerated delayed-self-reinforcement (A-DSR), where each agent only uses current and past information from the network. This use of already existing information is advantageous since the convergence improvement is achieved without the need to change the network connectivity and without the need for additional information from the network. Nevertheless, the update law for more general graphs with non-symmetric Laplacian (e.g., directed graphs with real-valued spectrum) cannot be obtained from the gradient of the graph potential [31], [32]. Therefore, the Nesterov-based approach and its stability analysis (which relies on the Lyapunov function) cannot be directly applied for general directed graphs, which is addressed in the current work.

The main contribution of this article is to design a Nesterov-type accelerated update for more general graph networks (with Laplacians that have real-valued spectrum) using a local potential function for each agent. However, since the resulting update law does not necessarily reduce the overall Laplacian potential [32], the convergence studies from optimization methods cannot be used to establish stability [22], [23]. Moreover, while Lyapunov functions can be found to study stability for general directed graphs [32], the gradient of these Lyapunov function does not lead to the control update law, and hence accelerated methods cannot be directly applied using these Lyapunov functions. Prior methods that use either the momentum term alone or the outdated-feedback term alone also do not address the stability when both terms are used for general directed graphs. In this context, a contribution of this article is to develop stability conditions for the proposed generalized accelerated approach, with both the momentum and outdated-feedback terms. The current article expands on our prior work in [29], which used a fixed ratio between the momentum and outdated-feedback terms, by (i) proposing the general case with varying ratios between the momentum and Nesterov terms; (ii) developing a stability condition for the generalized approach, (iii) designing the A-DSR to achieve fast response while maximizing structural robustness, and (iv) presenting experimental results to comparatively evaluate the performance, with and without A-DSR.

The article begins by presenting the structurally-robust, convergence-rate improvement problem, along with the limits of standard consensus-based update in Section II. The proposed A-DSR based approach is introduced in Section III and the stability conditions of the A-DSR approach are developed in Section III-B, followed by the derivation of analytical Robust A-DSR approach for maximizing robustness in Section III-C. Section IV-A comparatively evaluates the performance with and without A-DSR through simulation, and Section IV-B presents experimental results. Lastly, conclusions from the article are reported in Section V.

II. PROBLEM FORMULATION

This section introduces graph based consensus dynamics used to model networked systems, and describes the convergence limits with structural robustness achievable due to stability bounds on the update gain in standard neighbor-based consensus dynamics. Finally, the problem statement of the article is stated.

A. Background: graph-based control

Let the multi-agent network be modeled using a graph representation, where the connectivity of the agents is represented by a directed graph (digraph) \( \mathcal{G} = (\mathcal{V}, \mathcal{E}) \), e.g., as defined in [12]. Here, the agents are represented by nodes \( \mathcal{V} = \{1, 2, \ldots, n+1\}, \) \( n > 1 \) and their connectivity by edges \( \mathcal{E} \subseteq \mathcal{V} \times \mathcal{V} \), where each agent \( j \) belonging to the set of neighbors \( N_j \subseteq \mathcal{V} \) of the agent \( i \) satisfies \( j \neq i \) and \((j, i) \in \mathcal{E})

The evolution of the multi-agent network is defined using the graph \( \mathcal{G} \), as in Eq. (1). The terms \( l_{i,j} \) of the \((n+1)\times(n+1)\) Laplacian \( L \) of the graph \( \mathcal{G} \) are real and given by

\[
l_{i,j} = \begin{cases} 
-\alpha_{i,j} < 0, & \text{if } j \in N_i \\
\sum_{m=1}^{n+1} a_{i,m}, & \text{if } j = i, \\
0, & \text{otherwise},
\end{cases}
\]

(2)

where the weight \( a_{i,j} \) is nonzero (and positive) if and only if \( j \) is in the set of neighbors \( N_i \subseteq \mathcal{V} \) of the agent \( i \), each row of the Laplacian \( L \) adds to zero, i.e., from Eq. (2), the \((n+1)\times1\) vector of ones \( \mathbf{1}_{n+1} = [1, \ldots, 1]^T \) is a right eigenvector of the Laplacian \( L \) with eigenvalue 0,

\[
L \mathbf{1}_{n+1} = 0 \mathbf{1}_{n+1}.
\]

(3)

1) Network dynamics: One of the agents is assumed to be a virtual source agent, which can be used to specify a desired consensus value \( X_s \). Without loss of generality, the state \( X_{n+1} \) of last \( n+1 \) node is assumed to be a virtual source agent \( X_s \). Moreover, each agent in the network should have access to the virtual source agent \( X_s \) through the network, as formalized below.

Assumption 1 (Connected graph): The digraph \( \mathcal{G} \) is assumed to have a directed path from the source node \( n+1 \) to any other node \( i \) in the graph, i.e., \( i \in \mathcal{V} \setminus \{n+1\} \).

Some properties of the graph \( \mathcal{G} \) without the source node \( n+1 \), i.e., \( \mathcal{G} \setminus s \), are listed below. In particular, consider the \( n \times n \) pinned Laplacian matrix \( K \) associated with \( \mathcal{G} \setminus s \) obtained by removing the row and column associated with the source node \( n+1 \) through the partitioning of the Laplacian \( L \), i.e.,

\[
L = \begin{bmatrix} K & -B \\
*_{1 \times n} & *_{1 \times 1}
\end{bmatrix}
\]

(4)
where $B$ is an $n \times 1$ matrix
\[
B = [a_1, a_2, \ldots, a_n]^T
\]
and non-zero values of $B_j$ implies that the agent $j$ is directly connected to the source $X_s$. The properties of the pinned Laplacian $K$ follow from Assumption 1, e.g., see [12].

1. The pinned Laplacian matrix $K$ is invertible, i.e.,
\[
\det\{I(K)\} \neq 0.
\]

2. The eigenvalues of the pinned Laplacian $K$ have strictly-positive, real parts.

3. The product of the inverse of the pinned Laplacian $K^{-1}$ with $B$ leads to a $n \times 1$ vector of ones, i.e.,
\[
K^{-1}B = I_n.
\]

The dynamics of the non-source agents $X$ represented by the remaining graph $G \setminus s$, can be given by
\[
X[k+1] = X[K] - \alpha KX[k] + \alpha BX_s[k]
= (I - \alpha K)X[k] + \alpha BX_s[k]
= PX[k] + \alpha BX_s[k],
\]
where $P$ is Perron matrix,
\[
P = I_{n \times n} - \alpha K,
\]
and $I_{n \times n}$ is the $n \times n$ identity matrix. A sufficiently-small selection of the update gain $\alpha$ will stabilize the dynamics in Eq. (8), i.e., all eigenvalues $\lambda_{P,m}$ of the Perron matrix will lie inside the unit circle. Note that if $\lambda_{K,m}$ is an eigenvalue of the pinned Laplacian $K$ with a corresponding eigenvector $V_{K,m}$, i.e.,
\[
KV_{K,m} = \lambda_{K,m}V_{K,m},
\]
then
\[
\lambda_{P,m} = 1 - \alpha \lambda_{K,m}
\]
is an eigenvalue of the Perron matrix $P$ for the same eigenvector $V_{K,m}$, since
\[
PV_{K,m} = [I_{n \times n} - \alpha K]V_{K,m} = (1 - \alpha \lambda_{K,m})V_{K,m}.
\]

When the pinned Laplacian $K$ has a real-valued spectrum, the stability of the network dynamics in Eq. (8) depends on the extremal eigenvalues.

**Assumption 2 (Real spectrum):** The pinned Laplacian $K$ is assumed to have real eigenvalues, ordered as
\[
0 < \lambda_{K,1} \leq \lambda_{K,2} \leq \ldots \leq \lambda_{K,n} = \overline{\lambda} = \sigma(K)
\]
where $\lambda > 0$ and $\overline{\lambda}$ are extremal eigenvalues, and $\sigma_K = \overline{\lambda}$ is the spectral radius of the pinned Laplacian $K$.

Under Assumption 2, from Eq. (11), for network stability the update gain $\alpha$ needs to satisfy
\[
-1 < 1 - \alpha \lambda_{K,m} < 1
\]
or $0 < \alpha < \frac{2}{\lambda_{K,m}}$ for all eigenvalues $\lambda_{K,m}$ resulting in the following network-stability condition on the update gain $\alpha$
\[
0 < \alpha < \frac{2}{\lambda_{K,m}} = \overline{\sigma} < \infty.
\]

2) **Stable consensus:** With a stabilizing update gain $\alpha$, the state $X$ of the network (of all non-source agents) converges to a fixed source value $X_s$, e.g., for a step change in the source value $X_s$ from $x_i$ to $x_f$, i.e., $X_s[k] = x_f, \forall k < 0$ (initial desired state) and $X_s[k] = x_f, \forall k \geq 0$. Since the eigenvalues $\lambda_{P,m}$ of the Perron matrix $P$ are inside the unit circle, the solution to Eq. (8) for the step input converges,
\[
X[k+1] - X[k] = P^{k+1}1_n(x_f - x_i) \to 0,
\]
as $k \to \infty$ because $P^{k+1} \to 0$. Therefore, defining
\[
x_\Delta = (x_f - x_i)
\]
and taking the limit as $k \to \infty$ in Eq. (8), and from invertibility of the pinned Laplacian $K$ from Eq. (6),
\[
X[k] \to K^{-1}Bx_f
\]
as $k \to \infty$. Then, from Eq. (7), the state $X[k]$ of the non-source agents reaches the desired state $1_n x_f$ as time step $k$ increases, i.e.,
\[
X[k] \to 1_n x_f
\]
as $k \to \infty$. Thus, the control law in Eq. (6) achieves consensus.

The rate of convergence to consensus depends on the spectral radius $\sigma(P)$ of the Perron matrix $P$ given by
\[
\sigma(P) = \max_m |\lambda_{P,m}| = \max_m |1 - \alpha \lambda_{K,m}|.
\]

Note that for any $\epsilon > 0$ there exists a nonsingular matrix $Q$ such that the modified vector norm $\|X\| = \|QX\|_\infty$ with the corresponding induced matrix norm $\|\cdot\|$ satisfies, see (Section 5.3.5),
\[
\|P\| \leq \sigma(P) + \epsilon.
\]

Hence, from Eq. (16),
\[
\|X[k+1] - X[k]\| \leq \|P\|^{k+1}\|1_n(x_f - x_i)\| \leq (\sigma(P) + \epsilon)^{k+1}\|1_n(x_f - x_i)\|.
\]

Since $\epsilon$ can be chosen to be arbitrarily small, minimizing the spectral radius $\sigma(P)$ of the Perron matrix $P$ results in faster convergence.

**B. Convergence with structural robustness**

The structural robustness of the network’s stability depends on the spectral radius $\sigma(P)$ of the Perron matrix $P$. For the network to be stable, the eigenvalues of the Perron matrix $P$ need to be inside the unit circle. Hence, the distance $d$ of its eigenvalues $\lambda_{P,m}$ from the unit circle is a measure of the network’s structural stability, i.e., robustness to perturbations, where
\[
d = 1 - \sigma(P).
\]

Minimizing the spectral radius $\sigma(P)$ results in increased structural robustness. Therefore, rapid structurally-robust convergence is achieved during transitions if the spectral radius $\sigma(P)$ is minimized.
Lemma 1: [Optimal no-DSR] The update gain \( \alpha^* \) that minimizes the spectral radius \( \sigma(P) \) is given by
\[
\alpha^* = \frac{2}{\overline{\lambda} + \underline{\lambda}},
\]
where \( \overline{\lambda} \) and \( \underline{\lambda} \) are extremal eigenvalues of the pinned Laplacian \( K \) as in Eq. (13). The minimal spectral radius of the corresponding optimal Perron matrix \( P^* \), with \( \alpha = \alpha^* \) in Eq. (9), is
\[
\sigma^* = \sigma(P^*) = \frac{\overline{\lambda} - \underline{\lambda}}{\overline{\lambda} + \underline{\lambda}}.
\]
Proof: Bounds on the Perron eigenvalues \( \lambda_{P,m} \) can be established by multiplying Eq. (11) with the positive update gain \( \alpha > 0 \), as in Eq. (15), resulting in
\[
\alpha \Delta \leq \alpha \lambda_{K,m} \leq \alpha \overline{\lambda}.
\]
Subtracting each term in the above equation from 1 results in the following range for the Perron eigenvalues \( \lambda_{P,m} \)
\[
\overline{\lambda}_P = 1 - \alpha \Delta \geq \lambda_{P,m} \geq 1 - \alpha \overline{\lambda} = \underline{\lambda}_P.
\]
Since the bounds \( \overline{\lambda}_P, \underline{\lambda}_P \) of the Perron eigenvalues are achieved, the singular value of the Perron matrix \( P \) is the maximal value of the bounds. The optimal update gain minimizes the maximal bound, i.e.,
\[
\alpha^* = \arg \min_{\alpha} \left( \max \{ |1 - \alpha \overline{\lambda}|, |1 - \alpha \underline{\lambda}| \} \right)
\]
because, from Eq. (27), if \( \lambda_{P,m} \) is positive, then its magnitude cannot exceed \( |\lambda_P| \), and if \( \lambda_{P,m} \) is negative, then its magnitude cannot exceed \( |\lambda_P| \). Note that when the two bounds \( \overline{\lambda}_P, \underline{\lambda}_P \) on the Perron eigenvalues in Eq. (27) are equal in magnitude with opposite signs, i.e.,
\[
\overline{\lambda}_P = [1 - \alpha \Delta] = \frac{\overline{\lambda} - \underline{\lambda}}{\overline{\lambda} + \underline{\lambda}},
\]
the update gain is optimal, i.e., \( \alpha = \alpha^* \), and from Eq. (27),
\[
|1 - \alpha^* \overline{\lambda}| = |1 - \alpha^* \underline{\lambda}| = \sigma(P^*).
\]
When the update gain \( \alpha \) deviates from the optimal value, i.e.,
\[
\alpha = \alpha^* + \hat{\alpha},
\]
the bounds on the Perron eigenvalues \( \lambda_{P,m} \) can be rewritten as
\[
\overline{\lambda}_P = [1 - (\alpha^* + \hat{\alpha}) \Delta] = -\hat{\alpha} \Delta + \frac{\overline{\lambda} - \underline{\lambda}}{\overline{\lambda} + \underline{\lambda}},
\]
which is positive and increases in magnitude above \( \sigma(P^*) \) if the deviation in update gain is negative, i.e., \( \hat{\alpha} < 0 \), and similarly
\[
\underline{\lambda}_P = [1 - (\alpha^* + \hat{\alpha}) \Delta] = -\hat{\alpha} \Delta - \frac{\overline{\lambda} - \underline{\lambda}}{\overline{\lambda} + \underline{\lambda}},
\]
which is negative and also increases in magnitude above \( \sigma(P^*) \) if the deviation in update gain is positive, i.e., \( \hat{\alpha} > 0 \). The minimal is achieved with no deviation, i.e., \( \hat{\alpha} = 0 \) and \( \alpha = \alpha^* \).

Remark 1: The update gain \( \alpha^* \) from Lemma 1 (for minimum spectral radius) satisfies the network-stability condition in Eq. (15), as for any network, with \( 0 < \Delta \leq \overline{\lambda} \),
\[
0 < \alpha^* = \frac{2}{\overline{\lambda} + \underline{\lambda}} < \frac{2}{\overline{\lambda}}.
\]

Remark 2: The spectral radius of the Perron matrix \( \sigma(P) \) can be made the ideal value of zero when all the eigenvalues \( \lambda_{K,m} \) of the pinned Laplacian \( K \) have the same value, i.e., \( \lambda_{K,m} = \lambda \), e.g., in platoon networks. In this case, with \( \overline{\lambda} = \underline{\lambda} \) in Eq. (25), the minimal spectral radius is \( \sigma^* = 0 \) resulting in maximally fast convergence.

C. The robust convergence optimization problem

The research problem addressed is to reduce the spectral radius of Perron matrix, i.e. to improve structural robustness and convergence rate, when each agent can modify its update law

1) using only existing information from the network neighbors, and
2) without changing the network structure (network connectivity \( K \)).

III. PROPOSED SOLUTION

This section introduces the proposed Accelerated Delayed Self Reinforcement (A-DSR) approach to achieve structurally-convergent and establishes stability conditions.

A. The A-DSR approach

1) Graph’s Laplacian potential: For undirected graphs, the control law \( u \) in Eq. (1) can be considered as a gradient-based search on the graph’s Laplacian potential \( \Phi_G \).
\[
u(\hat{X}) = -\frac{\alpha}{2} \nabla \Phi_G(\hat{X}),
\]
where \( \Phi_G(\hat{X}) = \frac{1}{2} \sum_{i,j=1}^n a_{i,j} (\hat{X}_j - \hat{X}_i)^2 = \hat{X}^T L \hat{X} \) results in
\[
u(\hat{X}) = -\frac{\alpha}{2} \nabla \Phi_G(\hat{X}) = -\alpha L \hat{X}.
\]
This results in the standard graph-based update law as in Eq. (1).
\[
\hat{X}[k + 1] = \hat{X}[k] - \frac{\alpha}{2} \nabla \Phi_G(\hat{X}[k]) = \hat{X}[k] - \alpha L \hat{X}[k].
\]
2) Nesterov’s accelerated-gradient-based update: In general, the convergence of the gradient-based approach as in Eq. (32) can be improved using accelerated methods. In particular, applying the Nesterov modification [19], [20] of the traditional gradient-based method to Eq. (32) results in

\[ u(\hat{X}[k]) = -\frac{\alpha}{2} \nabla \Phi_G \left\{ \hat{X}[k] + \beta \left( \hat{X}[k] - \hat{X}[k - 1] \right) \right\} \\
+ \beta \left( \hat{X}[k] - \hat{X}[k - 1] \right) \\
= -\alpha L \left\{ \hat{X}[k] + \beta \left( \hat{X}[k] - \hat{X}[k - 1] \right) \right\} \\
+ \beta \left( \hat{X}[k] - \hat{X}[k - 1] \right). \quad (36) \]

This accelerated-gradient-based input results in the modification of the system Eq. (1) to

\[
\hat{X}[k + 1] = \hat{X}[k] - \alpha L \left( \hat{X}[k] + \beta \left( \hat{X}[k] - \hat{X}[k - 1] \right) \right) \\
+ \beta \left( \hat{X}[k] - \hat{X}[k - 1] \right). \quad (37)
\]

Consequently, the dynamics of the non-source agents \(X\) represented by the remaining graph \(\mathcal{G}\)'s, i.e., Eq. (31), becomes

\[
X[k + 1] = X[k] - \alpha K \left\{ X[k] + \beta \left( X[k] - X[k - 1] \right) \right\} \\
+ \beta \left( X[k] - X[k - 1] \right) \\
+ \alpha B \{ X_s[k] + \beta \left( X_s[k] - X_s[k - 1] \right) \} \\
\]

The additional third term \(\beta \left( X[k] - X[k - 1] \right)\) on the right hand side of Eq. (38) is referred to as the momentum term (this term alone forms the Heavy ball method in [35]) and the similar terms inside the curly brackets of the second and fourth terms are referred to as the outdated-feedback addition. The above update is referred to as the Nesterov-update in the following.

3) Directed graphs: For general directed graphs, the potential function \(\Phi_G\) in Eq. (33) does not lead to the standard update equations (31), (32). Nevertheless, motivated by the gradient-based approach, for each non-source agent, \(1 \leq i \leq n\), a modified potential can be considered as

\[
\Phi_{G,i}(\hat{X}) = \sum_{j=1}^{n+1} a_{i,j} \left( \hat{X}_i - \hat{X}_j \right)^2. \quad (39)
\]

Here \(\Phi_{G,i}(\hat{X})\) is a localized version of the graph’s Laplacian potential [31], [32], whose gradient with respect to \(\hat{X}_i = X_i\)

\[
u_i(\hat{X}) = -\frac{\alpha}{2} \partial \Phi_{G,i}(\hat{X}) = -\alpha K_i X + \alpha B_i X_s \quad (40)
\]

with \(K_i\) as the \(i^{th}\) row of \(K\), \(B_i\) the \(i^{th}\) row of the source connectivity matrix \(B\), will lead to the standard update equations for each agent’s state \(X_i\) in the state vector \(X\) of non-source agents, as

\[
X_i[k + 1] = X_i[k] - \alpha K_i X[k] + \alpha B_i X_s[k]. \quad (41)
\]

The application of the accelerated-gradient approach (which does not necessarily decrease the graph potential \(\Phi_{G,i}(\hat{X}_i)\)) in Eq. (33), (19), (20) leads to the same Eq. (38).

4) A-DSR update: The Nesterov-update law in Eq. (38) uses the same gain \(\beta\) for the momentum and the outdated-feedback terms (Nesterov’s accelerated method in [36]). A generalization of this is to use different gains \(\beta_1, \beta_2\) for the outdated-feedback and momentum terms (respectively), as used before in optimization theory [37].

\[
X[k + 1] = X[k] - \alpha K \left\{ X[k] + \beta_1 \left( X[k] - X[k - 1] \right) \right\} \\
+ \beta_2 \left( X[k] - X[k - 1] \right) \\
+ \alpha B \{ X_s[k] + \beta_1 \left( X_s[k] - X_s[k - 1] \right) \} \\
\]

The above accelerated approach, is referred to as the accelerated delayed self reinforcement (A-DSR) in the following, since it does not require additional information from the network, or having to change the network connectivity. Rather, each agent uses delayed versions of known information to reinforce its own update. To illustrate, for each non-source agent \(i\), let \(x_i\) be the information obtained from the network, i.e.,

\[
x_i[k] = \alpha K_i X[k] \quad (43)
\]

where \(K_i\) is the \((i^{th})\) row of the pinned Laplacian \(K\). Then, the update of agent \(X_i\) is, from Eq. (42),

\[
X_i[k + 1] = X_i[k] - \alpha K_i \left\{ X_i[k] + \beta_1 \left( X_i[k] - X_i[k - 1] \right) \right\} \\
+ \beta_2 \left( X_i[k] - X_i[k - 1] \right) \\
+ \alpha B_i \{ X_s[k] + \beta_1 \left( X_s[k] - X_s[k - 1] \right) \} \\
= X_i[k] - \{ x_i[k] + \beta_1 (x_i[k] - x_i[k - 1]) \} \\
+ \beta_2 (X_i[k] - X_i[k - 1]) \\
+ \alpha B_i \{ X_s[k] + \beta_1 (X_s[k] - X_s[k - 1]) \} \quad (44)
\]
where $B_i$ is the $i^{th}$ row of the source connectivity matrix $B$. The delayed self-reinforcement (DSR) approach, however, requires each agent to store delayed versions $X_i[k-1]$ and $x_i[k-1]$ of its current state $X_i[k]$ and information $x_i[k]$ from the network, as illustrated in Fig. 1. □

Remark 3: The A-DSR method in Eq. (42) without the momentum term (i.e., $\beta_2 = 0$) is referred to as the outdated-feedback method, without the outdated-feedback term (i.e., $\beta_1 = 0$) is referred to as the momentum method, and with equal parameters (i.e., $\beta_1 = \beta_2 = \beta$) is referred to as the Nesterov-update method.

B. Stability of A-DSR

The stability conditions for the A-DSR approach are presented below.

1) Diagonalizing the pinned Laplacian: The network with A-DSR in Eq. (42) is decomposed into subsystems using an invertible transformation matrix $P_K$ as

$$X[k] = P_K X_j[k],$$

where the transformation matrix $P_K$ is selected to diagonalize the pinned Laplacian $K$ as

$$K_j = P_K^{-1} K P_K$$

where the diagonal terms of matrix $K_j$ are the eigenvalues $\lambda_{K,m}$ for $i = 1, 2, ..., n$ (which can be complex and with multiplicity greater than 1). Since input doesn’t affect stability, setting $X_i[k] = 0$, $\forall k$, and pre-multiplying the Eq. (42) with $P_k^{-1}$ results in

$$X_j[k + 1] - X_j[k] + \alpha K_j(X_j[k] + \beta_1 (X_j[k] - X_j[k-1])) - \beta_2 (X_j[k] - X_j[k-1]) = 0.$$  

(47)

The stability of network with A-DSR in Eq. (42) is equivalent to the stability in the transformed coordinate in Eq. (47).

2) Stability conditions:

Lemma 2: [A-DSR stability] The network with the generalized A-DSR in Eq. (42) is stable if and only if the A-DSR gains $\alpha, \beta_1$ and $\beta_2$ satisfy

$$0 < \alpha < \alpha \lambda_{K,m} (\beta_1 + \frac{1}{2}) - 1 < \beta_2 < (\alpha \beta_1 \lambda_{K,m} + 1)$$

for each eigenvalue $\lambda_{K,m}$ of the pinned Laplacian $K$.

**Proof** Taking the z-transform of Eq. (47) results in

$$(z^2 I - z [(1 + \beta_2) I - \alpha (1 + \beta_1) K_j] - (\alpha \beta_1 K_j - \beta_2 I) X_j(z) = 0.$$  

(49)

Thus, the network with A-DSR update in Eq. (42) is stable if and only if the roots of the following characteristic equation $D(z) = 0$ have magnitude less than one for each eigenvalue $\lambda_{K,m}$ of the pinned Laplacian $K$, where

$$D(z) = z^2 + z [\alpha (1 + \beta_1) \lambda_{K,m} - (1 + \beta_2)] + (\beta_2 - \alpha \beta_1 \lambda_{K,m}).$$  

(50)

The Jury test leads to the following three sufficient and necessary conditions for the roots of the characteristic equation to have magnitude less than one.

1) Nestorov-update method in Eq. (48) with $\beta_1 = \beta_2 = \beta$:

$$\frac{\alpha \lambda_{K,m}}{2} - 1 < \beta (1 - \alpha \lambda_{K,m}) < 1.$$  

(51)

2) Momentum method ($\beta_1 = 0$): $\frac{\alpha \lambda_{K,m}}{2} - 1 < \beta_2 < 1.$

(52)

3) Outdated-feedback method ($\beta_2 = 0$): $-1 < \alpha \lambda_{K,m} \beta_1 - 1 < \frac{\alpha \lambda_{K,m}}{2}.$

(53)

Corollary 1: The network update as in Eq. (42), for the following accelerated methods, is stable if and only if $\alpha > 0$, and the gains satisfy the following for each eigenvalue $\lambda_{K,m}$ of the pinned Laplacian $K$.

1) Nesterov-update method in Eq. (48) with $\beta_1 = \beta_2 = \beta$:

$$\frac{\alpha \lambda_{K,m}}{2} - 1 < \beta (1 - \alpha \lambda_{K,m}) < 1.$$  

(55)

2) Momentum method ($\beta_1 = 0$):

$$\frac{\alpha \lambda_{K,m}}{2} - 1 < \beta_2 < 1.$$  

(56)

3) Outdated-feedback method ($\beta_2 = 0$):

$$-1 < \alpha \lambda_{K,m} \beta_1 - 1 < \frac{\alpha \lambda_{K,m}}{2}.$$  

(57)

**Proof** For the Nesterov-update method ($\beta_1 = \beta_2 = \beta$), the stability condition in Eq. (48) becomes

$$\frac{\alpha \lambda_{K,m}}{2} - 1 < \beta (1 - \alpha \lambda_{K,m}) + 1,$$  

(58)

and subtracting $\alpha \beta \lambda_{K,m}$ from both sides results in Eq. (55).

For the momentum method, Eq. (48) becomes Eq. (56) with $\beta_1 = 0$. For the outdated-feedback method, with $\beta_2 = 0$, Eq. (48) becomes

$$\frac{\alpha \lambda_{K,m}}{2} - 1 < (\alpha \beta_1 \lambda_{K,m} + 1).$$  

(59)
The left inequality in Eq. (59) can be simplified to
\[ \alpha \lambda_{K,m} \beta_1 < 1 - \frac{\alpha \lambda_{K,m}}{2} \] (60)
and the right inequality becomes
\[ \alpha \lambda_{K,m} \beta_1 > -1, \] (61)
resulting in the stability condition in Eq. (57).

The application of Lemma 2 or Corollary 1 requires knowledge of all eigenvalues \( \lambda_{K,m} \) of the pinned Laplacian \( K \). The following corollary provides sufficient conditions for stability in terms of the range \( [\lambda, X] \) of the eigenvalues \( \lambda_{K,m} \) from Eq. (13).

**Corollary 2:** The network update as in Eq. (42), for the following accelerated methods, is stable if and only if \( \alpha > 0 \), and the gains satisfy the following, where
\[
\lambda_* = \{ \begin{array}{ll} \lambda & \text{if } \beta_1 \leq -\frac{1}{2} \\ \frac{\lambda}{\lambda} & \text{if } \beta_1 > -\frac{1}{2} \end{array} \\
\lambda^* = \{ \begin{array}{ll} \lambda & \text{if } \beta_1 \leq 0 \\ \lambda & \text{if } \beta_1 > 0 \end{array} 
\]

1) Generalized A-DSR method:
\[
\left[ \alpha \lambda_*(\beta_1 + \frac{1}{2}) - 1 \right] < \beta_2 < (\alpha \beta_1 \lambda^* + 1). \] (63)

2) Nesterov-update method (\( \beta_1 = \beta_2 = \beta \)):
\[
\left[ \alpha \lambda_*(\beta_1 + \frac{1}{2}) - 1 \right] < \beta < (\alpha \beta \lambda^* + 1). \] (64)

3) Momentum method (\( \beta_1 = 0 \)):
\[
\frac{\alpha \lambda_1}{2} - 1 < \beta_2 < 1. \] (65)

4) Outdated-feedback method (\( \beta_2 = 0 \)):
\[
\left[ \alpha \lambda_*(\beta_1 + \frac{1}{2}) - 1 \right] < 0 < (\alpha \beta_1 \lambda^* + 1). \] (66)

**Proof** This follows from Lemma 2 and the proof of Corollary 1 since
\[
\alpha \lambda_{K,m}(\beta_1 + \frac{1}{2}) \leq \alpha \lambda_* \beta_1, \quad \alpha \lambda^* \beta_1 \leq \alpha \lambda_{K,m} \beta_1
\]
for all eigenvalues \( \lambda_{K,m} \) of the pinned Laplacian \( K \). Therefore, the conditions in this corollary are more stringent that the conditions in Lemma 2 and Corollary 1.

**Remark 4 (Stability for complex spectrum):** Although the article focuses on graphs with real-valued spectrum, the diagonalization approach can be used to infer stability for the complex-valued case. Consider the generalized A-DSR in Eq. (42) for a given Laplacian \( K \) with a complex eigenvalue pair \( \lambda_{K,m} = a + jb \), and \( \lambda_{K,m} = a - jb \), such that \( a, b \) are real-valued scalars. The Jordan block associated with these eigenvalues can be written as
\[
z^2 I_{2x2} - \left( (1 + \beta_2)I_{2x2} \right) - \alpha(1 + \beta_1) \begin{pmatrix} a & b \\ -b & a \end{pmatrix} - \alpha \beta_1 \begin{pmatrix} a & b \\ -b & a \end{pmatrix} - \beta_2 I_{2x2} \] (67)

with a corresponding fourth-order characteristic equation. Conditions for stability can then be found using the standard Jury test.

**C. Robust A-DSR**

Convergence with structural robustness for A-DSR is presented below, which is similar to the structurally-robust convergence without A-DSR in Section II-B. Note that the characteristic Eq. (50) with A-DSR is equivalent to that of a standard second order system of the form,
\[
D(z) = z^2 + 2\zeta_{K,m} \omega_{K,m} z + \omega_{K,m}^2 = 0, \] (68)
where
\[
\omega_{K,m}^2 = (\beta_2 - \alpha \beta_1 \lambda_{K,m}), \quad \zeta_{K,m} = \alpha(1 + \beta_1 \lambda_{K,m} - (1 + \beta_2)) \] (69)
with two roots \( z_{K,m,i} \), \( i \in \{1, 2\} \) associated with each eigenvalue \( \lambda_{K,m} \) of the pinned Laplacian \( K \). As in the case without A-DSR, the goal is to select the roots \( z_{K,m,i} \) of the characteristic equation in Eq. (50) for A-DSR, associated with the extremal eigenvalues \( \lambda \) of the pinned Laplacian \( K \), to be equidistant from origin (for similar structural robustness)
\[
|z_{K,m,1}| = |z_{K,m,2}| = |z_{K,m,1}| = |z_{K,m,2}| = |z_{K,m}| \] (70)
and are farthest away from the unit circle (for fast convergence), i.e., by choosing the A-DSR parameters \( \alpha, \beta_1, \beta_2 \) to solve the following minimization problem
\[
\min_{\alpha, \beta_1, \beta_2} \left[ |z_{K,m,1}| = |z_{K,m,2}| \right]. \] (71)

Furthermore, the roots of Eq. (50) associated with the dominant eigenvalue \( \lambda \) of the pinned Laplacian are critically damped and positive, i.e.,
\[
\zeta_{\lambda} = -1, \quad z_{\lambda,1} = z_{\lambda,2} > 0, \] (72)
as in the case without A-DSR, which can help to reduce oscillations in the response.

**Lemma 3:** [Parameter selection for Robust A-DSR] With the A-DSR approach, given that the A-DSR gain \( \alpha \) is positive for stability, the roots \( z_{K,m,i}, \zeta_{K,m,i}, i \in \{1, 2\} \) of the characteristic equation in Eq. (50) associated with distinct extremal eigenvalues \( \lambda \neq \overline{\lambda} \) of the pinned Laplacian \( K \), are equidistant as in Eq. (70), maximally robust as in Eq. (71), and critically damped for the dominant eigenvalue \( \lambda \) as in Eq. (72), provided the A-DSR parameters \( \alpha, \beta_1, \beta_2 \) are chosen as
\[
\alpha = \alpha = \frac{4}{(\sqrt{\lambda} + \sqrt{\lambda})^2}, \quad \beta_1 = 0, \quad \beta_2 = \frac{(\sqrt{\lambda} - \sqrt{\lambda})^2}{(\sqrt{\lambda} + \sqrt{\lambda})^2}. \] (73)

**Proof** This is shown below in four steps.

**Step 1** is to show that the roots of Eq. (50) associated with the extremal eigenvalue \( \lambda \) of the pinned Laplacian cannot be overdamped. Note that if the damping ratio \( \zeta_{\lambda} \) of the roots
\(z_{\lambda,1}, z_{\lambda,2}\) in Eq. \((50)\) associated with the extremal eigenvalue \(\lambda\) is larger than one in magnitude, i.e., \(|\zeta_\lambda| > 1\), then the roots
\[
\begin{align*}
  z_{\lambda,1} &= -\left(\zeta_\lambda \omega_\lambda\right) + \omega_\lambda \sqrt{\frac{\lambda^2 - 1}{\lambda}} \\
  z_{\lambda,2} &= -\left(\zeta_\lambda \omega_\lambda\right) - \omega_\lambda \sqrt{\frac{\lambda^2 - 1}{\lambda}} ,
\end{align*}
\]  
(74)
are real and distinct and have different magnitudes \(|z_{\lambda,1}| \neq |z_{\lambda,2}|\), which cannot satisfy the lemma’s equidistant condition as in Eq. \((70)\). Therefore, the roots \(z_{\lambda,1}, z_{\lambda,2}\) of Eq. \((50)\) associated with the extremal eigenvalue \(\lambda\) of the pinned Laplacian cannot be overdamped, i.e.,
\[|\zeta_\lambda| \leq 1.\]  
(75)

**Step 2** is to show that the equidistant condition of the lemma, as in Eq. \((70)\), leads to a zero outdated-feedback gain, \(\beta_1 = 0\). Since the magnitude of the damping ratio is not more than one, \(|\zeta_\lambda| \leq 1\) from Eq. \((75)\), the term \(\zeta_\lambda^2 - 1\) becomes negative in Eq. \((74)\), and therefore its square root is complex and the magnitudes of the roots become
\[|z_{\lambda,1}| = |z_{\lambda,2}| = |\omega_\lambda| = \omega_\lambda = \sqrt{\beta_2 - \alpha \beta_1 \lambda}.\]  
(76)
Similarly, the magnitudes of the roots associated with the extremal value \(\lambda\) with damping ratio \(\zeta_\lambda = -1\) in Eq. \((72)\), are
\[|z_{\Delta,1}| = |z_{\Delta,2}| = |\omega_\Delta| = \omega_\Delta = \sqrt{\beta_2 - \alpha \beta_1 \Delta}.\]  
(77)
To satisfy the equidistant condition,
\[|z_\lambda| = \sqrt{\beta_2 - \alpha \beta_1 \lambda} = \sqrt{\beta_2 - \alpha \beta_1 \lambda} = |z_\lambda|,
\] and since \(\alpha > 0\) and \(\Delta \neq \lambda, \beta_1 = 0\). Thus, the magnitude of the roots (associated with the extremal eigenvalues) are
\[|z_\lambda| = |z_\Delta| = |\omega_\lambda| = |\omega_\Delta| = \sqrt{\beta_2 - \alpha \beta_1 \lambda}.\]  
(78)

**Step 3** is to show that the roots of Eq. \((50)\) associated with the extremal eigenvalue \(\lambda\) are critically damped. Using the damping ratio definition for the extremal modes, \(\zeta_\lambda\) and \(\zeta_\Delta\) in Eq. \((69)\), with \(\beta_1 = 0\) and \(\zeta_\Delta = -1\), and substituting for \(\omega_\lambda, \omega_\Delta\) from Eq. \((78)\), results in
\[-1 = \frac{\alpha \lambda - (1 + \beta_2)}{2 \sqrt{\beta_2}} \]  
(79)
\[
\zeta_\lambda = \frac{\alpha \lambda - (1 + \beta_2)}{2 \sqrt{\beta_2}}.
\]
Solving the two equations in Eq. \((79)\) for the magnitude \(\sqrt{\beta_2}\) of the extremal roots results in
\[\sqrt{\beta_2} = \frac{\alpha (\lambda - \Delta)}{2 (1 + \zeta_\lambda)},\]  
(80)
which is minimized over damping ratio \(|\zeta_\lambda| \leq 1\) by selecting
\[\zeta_\lambda = 1.\]  
(81)
Note that the magnitude of the roots (associated with the extremal eigenvalues) becomes, from Eqs. \((78)\), and \((80)\),
\[|z_\lambda| = |z_\Delta| = \sqrt{\beta_2} = \frac{\alpha (\lambda - \Delta)}{4}.\]  
(82)

**Step 4** is to find the optimal A-DSR gains \(\alpha\) and \(\beta_2\). Substituting \(\zeta_\lambda = 1\) from Eq. \((81)\) into Eq. \((79)\), results in
\[
\begin{align*}
\alpha \lambda &= (1 + \beta_2) + 2 \sqrt{\beta_2} \\
\alpha \Delta &= (1 + \beta_2) - 2 \sqrt{\beta_2}.
\end{align*}
\]  
(83)
Dividing the two equations to eliminate \(\alpha\) yields a quadratic equation for \(\sqrt{\beta_2}\), the magnitude of the roots,
\[\lambda (1 + \beta_2) - 2 \lambda \sqrt{\beta_2} = \lambda (1 + \beta_2) + 2 \Delta \sqrt{\beta_2}.\]  
(84)
or
\[\lambda - \lambda |\beta_2| - 2 (\lambda + \Delta) \sqrt{\beta_2} + (\lambda - \Delta) = 0,\]  
(85)
with solutions
\[\sqrt{\beta_2} = \frac{\lambda + \Delta \pm 2 \sqrt{\lambda \Delta}}{(\lambda - \Delta)}.\]  
(86)
Since \(\lambda > \Delta > 0\), the smaller root in Eq. \((86)\) is chosen for maximizing structural robustness, resulting in
\[\sqrt{\beta_2} = \frac{\lambda + \Delta}{(\lambda - \Delta)} = \frac{\sqrt{\lambda - \sqrt{\lambda}}}{\sqrt{\lambda + \sqrt{\lambda}}},\]  
(87)
and from Eq. \((82)\),
\[\alpha = \frac{4}{(\lambda - \Delta)} \sqrt{\beta_2} = \frac{4}{(\sqrt{\lambda + \sqrt{\lambda}})^2}.\]  
(88)

**Lemma 4:** [Stability of Robust A-DSR] With the optimal parameters \(\alpha, \beta_1\) and \(\beta_2\) in Eq. \((73)\) from Lemma 3, the A-DSR is stable.

**Proof** With the optimal parameters in Eq. \((73)\). The damping ratio \(\zeta_{K,m}\) of the roots \((z_{\lambda_{K,m},i}, i \in \{1, 2\})\) of Eq. \((55)\) associated with each eigenvalue \(\lambda_{K,m}\) of the pinned Laplacian \(K\) is given by
\[\zeta_{\lambda_{K,m}} = \frac{\alpha \lambda_{K,m} - 1 - \beta_2}{2 \sqrt{\beta_2}},\]  
(89)
which makes the damping ratio \(\zeta_{\lambda_{K,m}}\) linear in the eigenvalue \(\lambda_{K,m}\), and varying between \(\zeta_{\lambda_{K,m}} = -1\) to \(\zeta_{\lambda_{K,m}} = 1\). This implies that any eigenvalue between the extremal ones is underdamped, i.e.,
\[|\zeta_{\lambda_{K,m}}| < 1, \forall \lambda < \lambda_{K,m} < \lambda\]  
(90)
As a result, the magnitude roots of the characteristic polynomial for \(\lambda_{K,m}\) is
\[|z_{\lambda_{K,m}}| = |\bar{z}_{\lambda_{K,m}}| = \sqrt{\beta_2} = \frac{\sqrt{\lambda - \sqrt{\lambda}}}{\sqrt{\lambda + \sqrt{\lambda}} < 1, \forall \lambda > \Delta > 0,}\]  
(91)
which shows that the roots are strictly within the unit circle resulting in stability.

**Remark 5 (Balanced structural robustness):** All the roots of the characteristic Eqs. \((55)\), associated with the Robust A-DSR, have the same magnitude and lie on a circle centered at the origin. Therefore, the roots are equally structurally robust,
i.e., they are equidistant from the unit circle. Thus, the A-DSR with optimal parameters, as in Eq. (73) from Lemma 3, leads to balanced structural robustness.

IV. RESULTS AND DISCUSSION

This section comparatively evaluates the Optimal no-DSR and the Robust A-DSR approaches using simulation results for an example network’s structural robustness and convergence rate during transition. Additionally, the improvements in convergence rate with the Robust A-DSR are validated with an experimental system.

A. Simulation results

1) Example transition problem: The network considered here has four agents \((n = 4)\) represented by nodes \(X_i\), where \(1 \leq i \leq 4\), with node connectivity represented by the graph in Figure 2.

![Graph of example network with four agents](image)

Fig. 2. Graph of example network with four agents \((n = 4)\). Non-source agents are \(X_i, 1 \leq i \leq 4\), and the source agent is \(X_5\). The edge between agents \(X_4\) (the agent with source input) and \(X_3\) is undirected, the others are directed.

The virtual source agent \(X_s\) determines the desired consensus value for the network and is connected to the agent \(X_4\), i.e., the leader. The connecting edges are all directed in the non-source graph network, except for the undirected edge between the leader \(X_4\) and follower agent \(X_3\) which makes the graph Laplacian asymmetric. The system dynamics with no-DSR for the example network, is given by Eq. (5), with the pinned-Laplacian \(K\) and \(B\) given as

\[
K = \begin{bmatrix}
1 & -1 & 0 & 0 \\
0 & 1 & -1 & 0 \\
0 & 0 & 1 & -1 \\
0 & 0 & -1 & 2
\end{bmatrix}, \quad B = \begin{bmatrix}
0 \\
0 \\
0 \\
1
\end{bmatrix}.
\] (92)

As discussed in Section II-A1, the gradient of the asymmetric Laplacian potential \(\Phi_0(X) = X^T L X\) in Eq. (33) does not lead to standard neighbor-based update in Eq. (8), where \(L\) is the graph Laplacian (from Eq. (3)) and \(X\) is the state vector including source agent.

2) Optimal no-DSR for example network: The optimal update gain \(\alpha^*\) from Eq. (24), for minimum spectral radius \(\sigma(P) = \sigma(P^*)\), is determined using the extremal eigenvalues \(\bar{\lambda} = 2.618\) and \(\underline{\lambda} = 0.382\) of the graph Laplacian \(K\) in Eq. (92), using Eq. (24), as

\[
\alpha^* = \frac{2}{\bar{\lambda} + \underline{\lambda}} = \frac{2}{2.6180 + 0.3820} = 0.6667.
\] (93)

The measure of structural robustness \(d^*\) with Optimal no-DSR is, from Eq. (23),

\[
d^* = 1 - \sigma^* = 0.255,
\] (94)

with the optimal spectral radius \(\sigma^* = 0.745\), as illustrated in Figure 3.

To assess the transition response, a simulation was performed with the virtual agent’s state \(X_s\) changing from an initial value \(X_s[k] = x_i\) for all \(k < 0\) to a final value \(X_s[k] = x_f\) for all \(0 \leq k\). It was assumed that the non-source agents are initially at consensus, i.e., \(X[0] = x_1 I_n\). With the update gain from Eq. (93), the simulated response of the Optimal no-DSR method for a change in virtual agent state \(X_s\) from \(x_i = 0\) to \(x_f = 100\) is shown in Figure 4. The settling time \(T_s\) of the network’s response, defined as the time taken for all the agents’ states to achieve and remain within 95% of the desired change \(x_{\Delta} = x_f - x_i = 100\) in the consensus state was found to be 14 sampling time periods \((k = 14)\) from the simulated response.
3) A-DSR improves structural robustness: The A-DSR approach in Eq. (44) under Subsection III-A2 is used to improve the example network’s structural robustness. The spectral radius of the network is minimized over the range of A-DSR parameters $\alpha, \beta_1$ and $\beta_2$.

$$\sigma^* = \min_{\alpha, \beta_1, \beta_2} \left[ \max_i \left( \max_{1 \leq i \leq 2} |\lambda_{K,m,i}| \right) \right],$$  

(95)

where $\lambda_{K,m,i}$ with $i \in \{1, 2\}$ are the roots of the characteristic Eqs. (68) associated with eigenvalue $\lambda_{K,m}$ of the pinned Laplacian $K$, and the search space is constrained by the stability conditions in Eq. (65). The optimum parameters for minimum spectral radius, found through a numerical search, and the resulting performance are tabulated in Table I. With these optimal parameter selection, the corresponding roots of the characteristic polynomial with A-DSR, in Eq. (42), for each eigenvalue $\lambda_{K,m}$, are shown in Figure 5. The optimal spectral radius is given by $\sigma^* = 0.447$, which is a reduction of 40% when compared to the Optimal no-DSR case for this example network. For the same state transition from $x_i = 0$ to $x_f = 100$ in the consensus state, the corresponding 5% settling time is 7 sampling time periods ($k = 7$), which is a 50% improvement over the Optimal no-DSR case. Thus, the A-DSR approach improves both the structural robustness and the convergence rate when compared to the Optimal no-DSR case.

4) Robust A-DSR’s performance similar to A-DSR: Instead of a numerical search to optimize the parameters as in the A-DSR case, the Robust A-DSR, proposed in Subsection III-C, yields closed-form expressions for selection of its parameters as in Eq. (73). With the Robust A-DSR, the corresponding roots of the characteristic polynomials in Eq. (42), for each eigenvalue $\lambda_{K,m}$, are shown in Figure 5. Note that the roots corresponding to the extremal eigenvalues $\lambda, \overline{\lambda}$ are real valued and critically damped, as in Lemma 3. Furthermore, the other roots of characteristic equation, for intermediate eigenvalues $\lambda$ satisfying $\Delta < \lambda < \overline{\lambda}$, lie on a circle with radius equal to the magnitude of the critically damped extremal modes as shown in Figure 5 which follows from Lemma 2. Overall, the spectral radius $\sigma^*$ of the example network, with Robust A-DSR, is equal to the magnitude of the roots, i.e., $\sigma^* = \sqrt{\beta_2} = 0.447$.

The performance of the Robust A-DSR is similar to the optimized search-based A-DSR (see Table I). In particular, the spectral radius of $\sigma^* = 0.447$ with Robust A-DSR is smaller by 40% when compared to $\sigma(P^*) = 0.745$ with the Optimal no-DSR method (see Table I), thus improving the structural robustness. Additionally, the settling time $T_s$ with Robust A-DSR was found to be 7 sampling time periods from the simulation result (which corresponds to a 50% improvement in convergence rate) as shown in Figure 5.

Remark 6 (Momentum term $\beta_2$ and settling time $T_s$): For the Robust A-DSR approach, the settling time $T_s$ can be estimated analytically in terms of the momentum term $\beta_2$. Since all the roots of the characteristic equation in Eq. (71) have the same magnitude, the dynamics associated with the under-damped roots of the Robust A-DSR converge faster than critically-damped, real-valued roots $\sqrt{\beta_2}$. The corresponding real-valued continuous-time roots $s_{\text{cont}}$ are at $s_{\text{cont}} = (\ln(\sqrt{\beta_2}))/\delta_i$, which can be used to predict the 5% settling time $T_s$ as (in number of sampling time periods)

$$T_s \approx \frac{5}{|s_{\text{cont}}|\delta_i} \frac{5}{|\ln(\sqrt{\beta_2})|} = 6.2,$$

(96)

which matches the simulation-based value of 7 sampling time periods. Thus, a larger momentum term $\beta_2$ tends to results in faster settling.

In summary, the Robust A-DSR approach provides similar improvements as with the general A-DSR approach, in both the structural robustness and the convergence rate when compared to the Optimal no-DSR approach. The advantage of the Robust A-DSR approach is that it provides an analytical approach for selecting the control parameters instead of the numerical search with the general A-DSR.

5) Comparison of constrained accelerated approaches: Although constrained, the Robust A-DSR (with $\beta_1 = 0$) outperforms both the Nesterov-update method (with $\beta_1 = \beta_2 = \beta$) as well as the Outdated-feedback method (with $\beta_2 = 0$). Optimal parameters for the Nesterov-update as well as the Outdated-feedback methods were also found using the same optimization in Eq. (75) with the additional constraints $\beta_1 = \beta_2 = \beta$ for Nesterov-update method and $\beta_2 = 0$ for Outdated-feedback method. The search space for parameters were constrained as in Corollary 1. The optimal parameters of Nesterov-update and Outdated-feedback methods and the performance are provided in Table II. When compared to the Optimal no-DSR case, the Nesterov-update improves the spectral radius by 23.4% which is less than the improvement of 40% with the Robust A-DSR approach. The Outdated-feedback method also improves the spectral radius when compared to the no-DSR case, but the improvement (19.9%) is even smaller than the Nesterov-update case with 23.4%. Similarly, the settling time improvement of 50% with Robust
A-DSR when compared to Optimal no-DSR is larger than the improvement of 21.43% with the Nesterov-update and 42.9% improvement with the Outdated-feedback. Thus, while the Robust A-DSR is constrained, it still matches the performance of the general optimal A-DSR, and outperforms both the Nesterov-update method as well as the Outdated-feedback method.

Remark 7 (Outdated-feedback versus momentum): When simultaneously improving both the structural robustness and the convergence rate, of the two components of the A-DSR, the momentum component (associated with $\beta_2$) has more significant impact than the outdated-feedback component (associated with $\beta_1$).

| Method               | min of $\alpha$ | $\beta_1$ | $\beta_2$ | $\sigma$ | $T_s(k)$ |
|----------------------|-----------------|-----------|-----------|----------|----------|
| Robust A-DSR         | 0.80            | 0.20      | 0.4472    | 7        |
| A-DSR                | $T_s$           | 0.6303    | 0.2376    | 0.3868   | 0.6634   | 7        |
| Momentum             | $T_s$           | 0.7995    | 0.2006    | 0.4477   | 7        |
| Nesterov -update     | $T_s$           | 0.8388    | 0.2347    | 0.4845   | 6        |
| Nesterov -feedback   | $T_s$           | 0.5212    | 0.4684    | 0.4684   | 0.7599   | 11       |
| Outdated-feedback    | $T_s$           | 0.9638    | -0.1414   | 0.5973   | 8        |
| Optimal no-DSR       | 0.6667          | 0         | 0         | 0.745    | 14       |

7) Convergence improvement without structural robustness: The above results focused on increasing both the structural robustness and convergence rate. However, the parameters of the accelerated update methods can be chosen purely for optimizing the convergence rate (i.e. minimizing the settling time $T_s$). The resulting optimized parameters (found through a numerical search) and the performance are quantified in and Table I.

The accelerated methods achieve smaller settling time $T_s$ when the parameters are optimized for achieving a faster convergence rate. For instance, the settling time $T_s$ with A-DSR (search based) improves to 6 sampling time periods (see Table I), which is faster than Robust A-DSR and Nesterov-update each taking 7 sampling time periods, and an improvement of 57.1% over the Optimal no-DSR case. However, this improvement in settling time $T_s$ is accompanied by a decrease in structural robustness of the network. For example, with A-DSR parameters selected for fast convergence, the spectral radius $\sigma$ increased to $\sigma = 0.6237$ from $\sigma = \sigma^* = 0.4472$ for the case when the parameters were selected to maximize bot the structural robustness and convergence rate. Among the other accelerated approaches, the Momentum method also achieves the same settling time of 6 sampling time periods as the A-DSR case, indicating the importance the momentum term in improving the convergence rate of the given example network. A similar loss in structural robustness is seen with the Momentum and Outdated-feedback approaches when the parameters are optimized purely for faster convergence rate, as seen in Table I. The loss in structural robustness (for this example) is more with the Outdated-feedback than with the Momentum method.

The simulation results show that the network’s convergence-rate alone can be improved with the general A-DSR further than that achieved with Robust A-DSR. However, this increase in convergence-rate alone involves a loss in structural robustness. Moreover, the A-DSR parameters are found using a numerical search method.

In contrast, the parameters of the Robust A-DSR can be found analytically and achieves similar convergence rate as the A-DSR optimized for convergence-rate alone. Moreover, the performance improvement with the Robust A-DSR (as well as the A-DSR), in terms of both the structural robustness and the rapidity of transition, is better than the performance with the standard no-DSR consensus method.

B. Experimental results

A mobile-bot network is used for experimental evaluation of the proposed A-DSR approach.

1) System description: The experimental setup consists of four mobile-bot agents that move in a straight line. The network connectivity is the same as in the simulation example. The bots aim to maintain a spacing of $d_o$ between them, and the state $X_i$ of each bot $i$ is defined as the displacement from the initial equally-spaced configuration, as shown in Figure 6. The virtual source input $X_s$ determines the desired position of the network.

Fig. 6. Experimental test bed consisting of four mobile-bot agents moving in a straight line, with the same connectivity as in the example simulation network in Figure 2. Each agent $i$’s state is its displacement $X_i$ from its initial position.

2) Bot’s update computation: The desired displacement $X_i[k+1]$ at the next time step $k+1$ is computed using local relative-distance measurements available at time step $k$ by each bot $i$ using distance sensors (Ultrasonic HC-SR04 to the front, and Infrared GP2Y0A21YK at the back). These measurements of each bot $i$ include

$$X_{f,i}[k] = (X_{i+1}[k] - X_i[k]) + d_0,$$  

(97)

the relative displacement w.r.t. the front bot $i+1$ (which is $X_s$ for leader bot $i=4$), and

$$X_{b,i}[k] = d_0 - (X_{i-1}[k] - X_i[k]),$$  

(98)
the relative displacement w.r.t. the back bot $i$ where $2 < i - 1 < 4$, where $d_0$ is the desired offset distance between the bots in the experimental setup. These relative-distance measurements ($X_{f,i}[k], X_{b,i}[k]$) are used to determine the neighbor information needed to evaluate the update law, i.e., to obtain $K_iX[k]$, where $K_i$ is the $i^{th}$ row of the pinned-Laplacian in Eq. (92). For example,

$$K_iX[k] = K_{i,i+1}(X_i[k] - X_{i+1}[k]) + K_{i,i-1}(X_i[k] - X_{i-1}[k]) = K_{i,i+1}(d_0 - X_{f,i}[k]) + K_{i,i-1}(X_{b,i}[k] - d_0). \tag{99}$$

Thus, the relative-distance measurements ($X_{f,i}[k], X_{b,i}[k]$) at time step $k$ enable each bot $i$ to compute its update, i.e., to find the desired position $X_i[k+1]$ at the next time step according to Eq. (44), where parameters $\beta_1$ and $\beta_2$ are zero for the no-DSR case.

3) Bot’s feedback control: Each $i^{th}$ bot’s controller aims to match its state (displacement) $X_i$ to the desired state $X_i[k+1]$ by the next time step, i.e., when time $t = t_{k+1}$. This is accomplished using a velocity-feedback inner-loop and a position-feedback outer-loop, as shown in Figure 7 using measurements of the agent state $X_i(t)$ from magnetic encoders on each bot $i$.

![Fig. 7. Each $i^{th}$ bot’s control system includes: a) distance sensors to the front and back, c) micro-controller for on-board computation, and d) wheels with magnetic encoders on motors to estimate each bot’s displacement, $X_i(t)$. To ensure that the bot achieves $X_i[k+1]$, an inner-loop controller with gain $k_v$ to track desired velocity $V_i[k]$ in Eq. (100) and an outer-loop controller with gain $k_x$ for position error $(X_i(t)$ in Eq. (101) are implemented.](image)

In particular, the desired velocity for the time period $[t_k, t_{k+1})$ is computed as

$$V_i[k] = \frac{X_i[k+1] - X_i[k]}{\delta t}, \tag{100}$$

where $\delta t$ is the discrete time step (in seconds) for the update method. The desired velocity $V_i[k]$ is then tracked using an inner-loop controller with gain $k_v$, as shown in Figure 7. Additionally, an outer-loop feedback with gain $k_x$ is used to correct for position error $(X_i(t))$ at any time $t \in [t_k, t_{k+1})$, determined as

$$\dot{X}_i(t) = (X_i[k] + \Delta X_i(t)) - X_i(t), \tag{101}$$

where $\Delta X_i(t) = V_i[k](t - t_k) = V_i[k]\Delta t$, as shown in Figure [7].

The selection of parameters for the experiment were based on velocity limits of 20 cm/s for the bots. The initial position was $x_i = 0$, and the final position was $x_f = 100$ cm. Therefore the sampling-time period $\delta t$ was chosen as 4 s to ensure that the bots could meet the position transitions of 80 cm in one sampling-time period $\delta t$, seen in simulations in Figure 4 with the bot’s feedback gains $k_v = 5$ and $k_x = 1$.

4) Convergence rate improvement of the multi-agent network: The improvement in convergence rate of transition response in the example network with Robust A-DSR, over Optimal no-DSR, is evaluated through the experimental mobile-bot network.

A transition in desired position (defined using virtual source $X_d$ from $x_i = 0$ cm to $x_f = 100$ cm, similar to simulations), is implemented on the mobile-bot network. Each bot, initially in consensus with position zero, responds as the transition information propagates through the bot network (in Figure 6, which is measured using the magnetic encoders. This state transition is implemented using Optimal no-DSR and Robust A-DSR, with parameters given in Table I, and the observations of convergence rates from seven trials (with both the approaches) are tabulated in Table III. The position responses of the bots during the transition are plotted in Figure 8 for each of the seven trials with Optimal no-DSR (in light blue) and Robust A-DSR (in light red). The mean responses for both approaches, obtained from averaging over the seven trials, are also shown in Figure 8.

| Method             | Trial | $T_s(k)$ |
|--------------------|-------|----------|
| Robust A-DSR       | Trial1| 11       |
|                    | Trial2| 11       |
|                    | Trial3| 10       |
|                    | Trial4| 11       |
|                    | Trial5| 11       |
|                    | Trial6| 10       |
|                    | Trial7| 9        |
| Mean Response      |       | 10       |
| Optimal no-DSR     | Trial1| 17       |
|                    | Trial2| 16       |
|                    | Trial3| 16       |
|                    | Trial4| 15       |
|                    | Trial5| 15       |
|                    | Trial6| 18       |
|                    | Trial7| 15       |
| Mean Response      |       | 16       |

Robust A-DSR shows improvement in convergence rate of the bot network’s transition response, improving the settling time (within 5% of the final position) by 4 to 9 time periods (27% to 50%), when compared with Optimal no-DSR, similar to that observed in simulations. The mean response converges 6 time periods faster with Robust A-DSR (an improvement of 37.5%) when compared with Optimal no-DSR, see Table III.

Thus, the convergence rate improvements observed in simulations with Robust A-DSR, with analytically determined parameters, over Optimal no-DSR are verified with similar results from experimental studies of position transition in the mobile-bot network.
ing the structural robustness and convergence rate beyond the
ment (A-DSR) approach, based on local potential, for improv-
limits of standard consensus-based networks. Of the two ter-
with analytical expressions for its parameters, that closely
and robustness. A Robust A-DSR approach was developed,
alleviates the need for numerical search when selecting
parameters of the general A-DSR. Moreover, experimental
results verified the improved convergence rate with Robust
A-DSR over Optimal no-DSR.

The proposed Robust A-DSR approach, can be used to
accelerate convergence and improve performance of networks
with uncertainty, for instance, distributed sensing in presence
of communication delays, operation of multi-agent networks
with a human-in-the-loop where the human or network model
is uncertain, and transporting flexible structures with uncertain
stiffness values using mobile bots. Further work is needed
to explore the suitability of the Robust A-DSR for these
applications.

V. CONCLUSIONS

The article introduced an accelerated delayed self reinforce-
ment (A-DSR) approach, based on local potential, for improv-
ing the structural robustness and convergence rate beyond the
limits of standard consensus-based networks. Of the two terms
in the accelerated approach, it was shown that the moment-
tum term has substantially more impact when compared to
the outdated-feedback term for improving convergence rate
and robustness. A Robust A-DSR approach was developed,
with analytical expressions for its parameters, that closely
matches the performance of the general A-DSR approach,
which alleviates the need for numerical search when selecting
parameters of the general A-DSR. Moreover, experimental
results verified the improved convergence rate with Robust
A-DSR over Optimal no-DSR.

REFERENCES

[1] A Huth and C Wissel. The simulation of the movement of fish schools.
Journal of Theoretical Biology, 156(3):365–385, Jun 7 1992.
[2] Tamás Vicsek, András Czörk, Eshel Ben-Jacob, Iton Cohen, and Ofer
Shochet. Novel type of phase transition in a system of self-driven
particles. Phys. Rev. Lett., 75:1226–1229, Aug 1995.
[3] A. Jaddda, Jie Lin, and A. S. Morse. Coordination of groups of mobile
autonomous agents using nearest neighbor rules. IEEE Transactions
on Automatic Control, 48(6):988–1001, June 2003.
[4] Wei Ren and R. W. Beard. Consensus seeking in multiagent systems
under dynamically changing interaction topologies. IEEE Transactions
on Automatic Control, 50(5):655–661, May 2005.
[5] R. Olfati-Saber. Flocking for multi-agent dynamic systems: algorithms
and theory. IEEE Transactions on Automatic Control, 51(3):401–420,
March 2006.
[6] August Mark, Yunjun Xu, and Benjamin T. Dickinson. Consensus-
Based Decentralized Aerodynamic Moment Allocation Among Synthetic
Jets and Control Surfaces. IEEE Transactions on Control Systems
Technology, 27(6):2718–2726, NOV 2019.
[7] Michele Cucuzzella, Sebastian Trip, Claudio De Persis, Xiaodong
Cheng, Antonella Ferrara, and Arjan van der Schaft. A Robust
Consensus Algorithm for Current Sharing and Voltage Regulation in
DC Microgrids. IEEE Transactions on Control Systems Technology,
27(4):1583–1595, JUL 2019.
[8] Johannes Schiffer, Thomas Seel, Joerg Raisch, and Tevfik Sezi. Voltage
Stability and Reactive Power Sharing in Inverter-Based Microgrids With
Consensus-Based Distributed Voltage Control. IEEE Transactions
on Control Systems Technology, 24(1):96–109, JAN 2016.
[9] Naiming Qi, Qifan Yuan, Yanfang Liu, Mingying Hao, and Shilei Cao.
Consensus Vibration Control for Large Flexible Structures of Spacecraft
With Modified Positive Position Feedback Control. IEEE Transactions
on Control Systems Technology, 27(4):1712–1719, JUL 2019.
[10] C. C. Ioannou, V. Guttal, and I. D. Couzin. Predatory fish select
for coordinated collective motion in virtual prey. Science, 337(6099):1212–
1215, Sep 7 2012.
[11] Maté Nagy, Zsuza Akos, Dora Biro, and Tamás Vicsek. Hierarchical
group dynamics in pigeon flocks. Nature, 464(7290):890, 2010.
[12] R. Olfati-Saber, J.A. Fax, and R.M. Murray. Consensus and cooperation
in networked multi-agent systems. Proceedings of the IEEE, 95(1):215–
233, Jan 2007.
[13] S. Devasia. Faster Response in Bounded-Update-Rate, Discrete-time
Networks using Delayed Self-Reinforcement. International Journal
of Control, Accepted, 2019. DOI: 10.1080/00207179.2019.1644537.
[14] Ruggero Carli, Fabio Fiumi, Alberto Speranzon, and Sandro Zampieri.
Communication constraints in the average consensus problem. Automat-
ica, 44(3):671–684, Mar 2008.
[15] Maria Pia Fant, Agostino Marcello Mangini, Francesca Mazzia, and
Walter Ukovich. A new class of consensus protocols for agent networks
with discrete time dynamics. Automatica, 54:1–7, Apr 2015.
[16] Xiaoming Duan, Jianping He, Peng Cheng, and Jiming Chen. Exploiting
a Mobile Node for Fast Discrete Time Average Consensus. IEEE
Transactions on Control Systems Technology, 24(6):1993–2001, NOV
2016.
[17] Zhenhong Li and Zengtao Ding. Robust Cooperative Guidance Law
for Simultaneous Arrival. IEEE Transactions on Control Systems
Technology, 27(3):1366–1367, MAY 2019.
[18] Eduardo Montijano, Sonia Martinez, and Carlos Sagues. Distributed
Robust Consensus Using RANSAC and Dynamic Opinions. IEEE
Transactions on Control Systems Technology, 23(1):150–163, JAN 2015.
[19] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning Internal
Representations by Error Propagation, pp. 318-362, in D. E. Rumelhart
and J. L. McClelland (eds.) Parallel Distributed Processing, Vol. 1.
MIT Press, Cambridge, MA, 1986.
[20] Ning Qian. On the momentum term in gradient descent learning
algorithms. Neural Networks, 12(1):145 – 151, 1999.
[21] Y. E. Nesterov. A Method of Solving a Convex Programming Problem
with Convergence Rate of O(1/k^2). Soviet Mathematics Doklady,
27(3):372–376, 1983.
[22] D. Jakoveti, J. M. F. Xavier, and J. M. F. Moura. Convergence rates
of distributed nesterov-like gradient methods on random networks. IEEE
Transactions on Signal Processing, 62(4):868–882, Feb 2014.
[23] B. Van Scoy, R. A. Freeman, and K. M. Lynch. The fastest known
globally convergent first-order method for minimizing strongly convex
functions. IEEE Control Systems Letters, 2(1):49–54, Jan 2018.
[24] S. Devasia. Rapid Information Transfer in Swarms under Update-Rate-
Bounds using Delayed Self-Reinforcement. ASME Journal of Dynamic
Systems Measurement and Control, 141(8):081009-19, Aug Aug,
2019.
[25] A. Attanasi, A. Cavagna, L. Del Castello, I. Giardina, T.S. Grigera,
A. Jelic, S. Mellilo, L. Parisi, O. Pohl, E. Shen, and M. Viale.
Information transfer and behavioural inertia in starling flocks. Nature
Physics, 10(9):615–698, Sep 1 2014.
[26] Yongcan Cao and Wei Ren. Multi-Agent Consensus Using Both Current
and Outdated States with Fixed and Undirected Interaction. Journal of
Intelligent & Robotic Systems, 58(1):95–106, April 2010.
[27] Hossein Moradian and Solmaz Kia. Accelerated average consensus
algorithm using outdated feedback. In 2019 European Control Conference
ECC, June 25-28, Napoli, Italy, 2019.
[28] J. Bu, M. Fazel, and M. Mesbahi. Accelerated consensus with linear rate
of convergence. In 2018 Annual American Control Conference (ACC),
pages 4931–4936, June 2018.
[29] S. Devasia. Accelerated Consensus for Multi-Agent Networks through
Delayed Self Reinforcement. IEEE International Conference on Indus-
trial Cyber-Physical Systems, Taipei, Taiwan - May 6-9, 2019.
[30] S. Devasia. Cohesive Networks using Delayed Self-Reinforcement.
Automatica, 112:108699, 1–13, Feb 2020.
[31] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, Sep. 2004.

[32] Hui Zhang and Junmin Wang. Robust two-mode-dependent controller design for networked control systems with random delays modelled by Markov chains. *International Journal of Control*, 88(12):2499–2509, Dec 2 2015.

[33] J.M. Ortega. *Matrix Theory, A Second Course, The University Series in Mathematics, Classics in applied mathematics ; vol. 3*. Plenum Press, New York, 1987.

[34] H. Zhang, F. L. Lewis, and Z. Qu. Lyapunov, adaptive, and optimal design techniques for cooperative systems on directed communication graphs. *IEEE Transactions on Industrial Electronics*, 59(7):3026–3041, July 2012.

[35] Euhanna Ghadimi, Hamid Reza Feyzmahdavian, and Mikael Johansson. Global convergence of the heavy-ball method for convex optimization. In *2015 European Control Conference (ECC)*, pages 310–315. IEEE, 2015.

[36] Yuri Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer Publishing Company, Incorporated, 1 edition, 2014.

[37] Laurent Lessard, Benjamin Recht, and Andrew Packard. Analysis and design of optimization algorithms via integral quadratic constraints. *SIAM Journal on Optimization*, 26(1):57–95, 2016.