

Distributed Linear Quadratic Optimal Control: Compute Locally and Act Globally

Junjie Jiao^{ID}, Harry L. Trentelman^{ID}, *Fellow, IEEE*, and M. Kanat Camlibel^{ID}, *Member, IEEE*

Abstract—In this letter we consider the distributed linear quadratic (LQ) control problem for networks of agents with single integrator dynamics. We first establish a general formulation of the distributed LQ problem and show that the optimal control gain depends on global information on the network. Thus, the optimal protocol can only be computed in a centralized fashion. In order to overcome this drawback, we propose the design of protocols that are computed in a decentralized way. We will write the global cost functional as a sum of *local* cost functionals, each associated with one of the agents. In order to achieve “good” performance of the controlled network, each agent then computes its own local gain, using sampled information of its neighboring agents. This decentralized computation will only lead to suboptimal global network behavior. However, we will show that the resulting network will reach consensus.

Index Terms—Distributed control, linear quadratic optimal control, decentralized computation, consensus, multi-agent systems.

I. INTRODUCTION

THE DISTRIBUTED linear quadratic (LQ) optimal control problem is the problem of interconnecting a finite number of identical agents according to a given network graph to achieve consensus optimally. Each agent receives input only from its neighbors, in the form of a linear feedback of the relative states amplified by a certain constant gain. Such control law is called a *distributed diffusive control law*. The problem of minimizing a given quadratic cost functional over all distributed diffusive control laws that achieve consensus is then called the distributed LQ problem corresponding to this cost functional.

In the case that the agent dynamics is given by a general state space system, this optimal control problem is non-convex and difficult to solve, and it is unclear whether a solution exists in general, see [1]. In contrast, for the case of single integrator dynamics it is fairly easy to find an explicit expression for the optimal distributed diffusive control law,

Manuscript received February 28, 2019; revised May 13, 2019; accepted June 3, 2019. Date of publication June 11, 2019; date of current version June 21, 2019. Recommended by Senior Editor Z.-P. Jiang. (Corresponding author: Junjie Jiao.)

The authors are with the Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence, University of Groningen, 9742JD Groningen, The Netherlands (e-mail: j.jiao@rug.nl; h.l.trentelman@rug.nl; m.k.camlibel@rug.nl).

Digital Object Identifier 10.1109/LCSYS.2019.2922189

see, for example, [2]. Although a solution to the problem is available, it turns out however that *global information* on the network is needed to compute this optimal control law. More specifically, the optimal distributed diffusive control law can be computed only by a (virtual) supervisor that knows the network graph and the initial states of all the agents. Thus, although the resulting optimal control law *operates* in a distributed fashion, its actual computation can only be performed in a *centralized* way.

Formulating the distributed LQ problem as a problem of minimizing a *global* cost functional is therefore not practical. Indeed, the centralized computation requires that the local optimal gains needs to be re-designed by the supervisor in case that changes in the network occur. For example, by adding or removing agents from the network, its graph will change, and new initial states will occur while existing ones will disappear.

In this letter we will address this drawback and present a *decentralized* design method to compute a distributed controller: each agent will compute its own local control law. For this computation, the agent will not need knowledge of the network graph or the initial states of all other agents. This will then enable ‘plug-and-play’ operations on the network, since each agent will be able to automatically recompute its local gain whenever a new agent is added or removed.

In order to achieve this decentralized computation scheme we will write the original global cost functional as the sum of *local* LQ tracking cost functionals, each associated with one of the agents. The agents can not solve these optimal tracking problems explicitly because the reference signals depend on the future dynamics of the neighbours. However, using sampling, suboptimal local gains are obtained. This decentralized computation will not necessarily result in optimality of the global network behavior. We will however show that the resulting network will reach consensus.

The distributed LQ control problem has attracted much attention in the past, see, e.g., [2]–[5]. In [3], a suboptimal distributed controller for a global cost functional was developed to stabilize a network with general agent dynamics. A similar cost functional was also considered in [6] for designing distributed controllers with guaranteed performance. The distributed LQ control problem with general agent dynamics was also dealt with in [7] and [8] by adopting an inverse optimal control approach. In [9] a game theoretic approach was considered to obtain a suboptimal solution. Also, [1] considers a suboptimal version of this problem. In [10], a suboptimal

consensus controller design was developed by employing a hierarchical LQ control approach for an appropriately chosen global performance index, and a similar idea for constructing a particular cost functional was employed in [11] to design a reduced order distributed controller. In [12] a distributed optimal control method was adopted to decouple a class of linear multi-agent systems with state coupled nonlinear uncertainties.

The common feature of all work referred to above is that the computation of the control gains needs global information on the network. This disadvantage can be avoided by adopting adaptive control methods [13] or by using reinforcement learning [14], [15]. In [16] and [17], it was shown that diffusive couplings are necessary for minimization of cost functionals of a particular form, involving the weighted squared synchronization error.

Below we list the contributions of this letter.

- 1) We show that for agents with single integrator dynamics, in any distributed LQ cost functional, the state weighting matrix must be equal to a weighted square of the Laplacian of the network graph.
- 2) We give a solution to this general distributed LQ problem, and show that computation of the optimal protocol requires exact knowledge of the Laplacian as well as the initial state of the entire network.
- 3) We represent the global cost functional as a sum of local LQ tracking cost functionals, one for each agent. Using sampling, suboptimal local gains are obtained. Computation of these gains is completely decentralized.
- 4) We show that these gains lead to a protocol that achieves consensus of the network.

The outline of this letter is as follows. In Section II, we derive a general formulation of the distributed LQ problem. In Section III, we show that computation of the optimal control laws requires complete knowledge of the network graph and the initial state of the entire network. In Section IV, we propose a decentralized method to compute suboptimal (local) control laws. In order to do this, we need to apply ideas from linear quadratic tracking, and these are reviewed in Section V. Then, in Section VI, we compute these local control laws, and show that the network reaches consensus if all agents apply their own local gain. Finally, in Section VII, we will give some concluding remarks.

Notation: We denote by \mathbb{R} the field of real numbers. The space of n -dimensional real vectors is denoted by \mathbb{R}^n . The vector in \mathbb{R}^n with all components equal to 1 is denoted by $\mathbf{1}_N$. The identity matrix of dimension $n \times n$ is denoted by I_n . For a symmetric matrix P , we write $P > 0$ ($P \geq 0$) if P is positive (semi-)definite. We use $\text{diag}(a_1, a_2, \dots, a_n)$ to denote the $n \times n$ diagonal matrix with a_1, a_2, \dots, a_n on its diagonal. For a linear map $A : \mathcal{X} \rightarrow \mathcal{Y}$, the kernel and image of A are denoted by $\ker(A) := \{x \in \mathcal{X} | Ax = 0\}$ and $\text{im}(A) := \{Ax | x \in \mathcal{X}\}$, respectively.

In this letter, a graph is denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $\mathcal{V} = \{1, 2, \dots, N\}$ the node set and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ the edge set. For $i, j \in \mathcal{V}$, an edge from node i to j is represented by $(i, j) \in \mathcal{E}$. The neighboring set of node i is defined as $\mathcal{N}_i := \{j \in \mathcal{V} | (i, j) \in \mathcal{E}\}$. The adjacency matrix of \mathcal{G} is

equal to $A = [a_{ij}] \in \mathbb{R}^{N \times N}$, where $a_{ij} = 1$ if $(j, i) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. The degree matrix of \mathcal{G} is given by $D = \text{diag}(d_1, d_2, \dots, d_N)$ with $d_i = \sum_{j=1}^N a_{ij}$, and the Laplacian matrix is defined as $L := D - A$. A graph is called simple if \mathcal{E} only contains edges (i, j) with $i \neq j$, and it is called undirected if $(i, j) \in \mathcal{E}$ implies that $(j, i) \in \mathcal{E}$. Obviously, a graph is undirected if and only if L is symmetric. A simple undirected graph is called connected if for each pair of nodes i and j there exists a path from i to j . Throughout this letter, it will be a standing assumption that the network graph is a connected simple undirected graph.

II. THE GENERAL FORM OF A DISTRIBUTED LQ COST FUNCTIONAL

In this section we will show that in any distributed LQ cost functional, the state weighting matrix must be a weighted square of the Laplacian of the network graph. We will also give two important examples of distributed LQ cost functionals.

We consider a network of agents described by scalar single integrator dynamics

$$\dot{x}_i(t) = u_i(t), \quad x_i(0) = x_{i0}, \quad i = 1, 2, \dots, N, \quad (1)$$

with $x_{i0} \in \mathbb{R}$ the initial state of agent i . By collecting the states and inputs of the individual agents into the vectors $x = (x_1, x_2, \dots, x_N)^\top$ and $u = (u_1, u_2, \dots, u_N)^\top$, (1) can be written as

$$\dot{x}(t) = u(t), \quad x(0) = x_0. \quad (2)$$

A general class of LQ cost functionals are those of the form

$$J(x_0, u) = \int_0^\infty x^\top(t) Q x(t) + u^\top(t) R u(t) dt, \quad (3)$$

where $Q \in \mathbb{R}^{N \times N}$, $R \in \mathbb{R}^{N \times N}$ and $Q \geq 0$ and $R > 0$.

In the context of distributed LQ control we only allow distributed diffusive control laws that achieve consensus, i.e., the controlled trajectories converge to $\text{im}(\mathbf{1}_N)$, the span of the vector of ones. Thus the class of control laws over which we want to minimize (3) consists of those of the form $u = -gLx$, with $L \in \mathbb{R}^{N \times N}$ the Laplacian of the network graph and where $g > 0$, see, e.g., [18].

We will now show that for a cost functional (3) to make sense in this context, the weighting matrix Q must be of the form $Q = LWL$ for some positive semi-definite matrix W .

Lemma 1: $J(x_0, u) < \infty$ for all $x_0 \in \mathbb{R}^N$ and control laws of the form $u = -gLx$ with $g > 0$ only if there exists a positive semi-definite $W \in \mathbb{R}^{N \times N}$ such that $Q = LWL$.

Proof: Write $Q = C^\top C$ for some C . Now, let $\bar{x}(t)$ denote any nonzero state trajectory generated by the control law $u = -gLx$ with $g > 0$ and let $\bar{u}(t) = -gL\bar{x}(t)$. It is well known that this control law achieves consensus (see [18]) so we have $\bar{x}(t) \rightarrow c\mathbf{1}_N$ for some nonzero constant c . Now assume that the control law $u = -gLx$ gives finite cost, i.e., $J(x_0, \bar{u}) < \infty$. This implies $\int_0^\infty \bar{x}^\top(t) C^\top C \bar{x}(t) dt < \infty$ and hence $C\bar{x}(t) \rightarrow 0$. Thus we obtain $\mathbf{1}_N \in \ker(C)$, equivalently, $\ker(L) \subset \ker(C)$. We thus conclude that there exists a matrix V such that $C = VL$ so the state weighting matrix Q must be of the form $Q = LV^\top VL$ for some matrix V . This proves our claim. ■

We have thus shown that, for a general LQ cost functional to make sense in the context of distributed diffusive control for multi-agent systems, it must necessarily be of the form

$$J(u, x_0) = \int_0^\infty x^\top(t) LWLx(t) + u^\top(t) Ru(t) dt, \quad (4)$$

for some $W \geq 0$ and $R > 0$. The corresponding distributed LQ problem is to minimize, for the system (2) with initial state x_0 , the cost functional (4) over all control laws of the form $u = -gLx$ with $g > 0$.

As an illustration, we will now provide two important special cases of LQ cost functionals. The first one was studied before in [1] and [2]:

$$J(u, x_0) = \sum_{i=1}^N \int_0^\infty \sum_{j \in \mathcal{N}_i} q(x_i(t) - x_j(t))^2 + ru_i^2(t) dt, \quad (5)$$

where q and r are positive real numbers. Clearly, (5) is equal to $J(x_0, u) = \int_0^\infty x^\top(t) 2qLx(t) + ru^\top(t)u(t) dt$. Note that $2qL = L(2qL^\dagger)L$ with L^\dagger the Moore-Penrose inverse of L (which is indeed positive semi-definite). Thus this cost functional is of the form (4) with $W = 2qL^\dagger$ and $R = rI$.

As a second example, consider

$$J(x_0, u) = \sum_{i=1}^N \int_0^\infty q(x_i(t) - a_i(t))^2 + ru_i^2(t) dt, \quad (6)$$

with

$$a_i(t) := \frac{1}{d_i + 1} (x_i(t) + \sum_{j \in \mathcal{N}_i} x_j(t)). \quad (7)$$

Here, q and r are positive weights, d_i denotes the node degree of agent i and \mathcal{N}_i its set of neighbors. The idea of the cost functional (6) is to minimize the sum of the deviations between the state $x_i(t)$ and the average $a_i(t)$ of the states of its neighbors (including itself) and the control energy. In order to put this in the form (4), define

$$G := (D + I_N)^{-1} (A + I_N) \in \mathbb{R}^{N \times N}, \quad (8)$$

where $D \in \mathbb{R}^{N \times N}$ is the degree matrix and $A \in \mathbb{R}^{N \times N}$ the adjacency matrix. Then clearly $a(t) = Gx(t)$, where $x = (x_1, x_2, \dots, x_N)^\top$ and $a = (a_1, a_2, \dots, a_N)^\top$. It is then easily seen that

$$J(x_0, u) = \int_0^\infty qx^\top(t)(I_N - G)^\top(I_N - G)x(t) + ru^\top(t)u(t) dt.$$

Since $(I_N - G)^\top(I_N - G) = L(D + I_N)^{-2}L$, we conclude that (6) is a special case of (4) with $W = q(D + I_N)^{-2}$ and $R = rI_N$.

III. CENTRALIZED OPTIMAL GAIN

In this section we will briefly give a solution to the general distributed LQ problem with cost functional (4) as introduced in Section II, thus generalizing the result from [2] to general distributed LQ cost functionals. We will show that, indeed, computation of the optimal protocol requires global information on the network graph and the initial state of the entire network.

Consider the cost functional (4) together with the dynamics (2) with given initial state x_0 . Since the admissible control laws are given by $u = -gLx$, the associated state trajectory is

$x(t) = e^{-gLt}x_0$ and $u(t) = -gLx(t)$. Substituting this into the cost functional yields

$$J(g) := x_0^\top \left(\int_0^\infty e^{-gLt} (LWL + g^2LRL) e^{-gLt} dt \right) x_0 \quad (9)$$

Clearly, we need to minimize $J(g)$ over $g > 0$. Substituting $g = \tau$, we find

$$J(g) := x_0^\top \int_0^\infty e^{-\tau L} \left(\frac{1}{g} LWL + gLRL \right) e^{-\tau L} d\tau x_0.$$

Define $X_0 := \int_0^\infty e^{-\tau L} LWL e^{-\tau L} d\tau$ and $Y_0 := \int_0^\infty e^{-\tau L} LRL e^{-\tau L} d\tau$. It turns out that both integrals indeed exist, and can be computed as particular solutions of the Lyapunov equations

$$-LX - XL + LWL = 0, \quad (10a)$$

$$-LY - YL + LRL = 0. \quad (10b)$$

Indeed, although L is not Hurwitz, these equations do have positive semi-definite solutions X and Y and, in fact, X_0 is the unique positive semi-definite solution X to (10a) with the property that $\text{im}(\mathbf{1}_N) \subset \ker(X)$. Likewise Y_0 is the unique positive semi-definite solution Y of (10b) with the property that $\text{im}(\mathbf{1}_N) \subset \ker(Y)$ (see [19, Proposition 1]). It follows from (10b) that, in fact, $\ker(Y_0) = \text{im}(\mathbf{1}_N)$. Thus we see that $J(g) = \frac{1}{g} x_0^\top X_0 x_0 + g x_0^\top Y_0 x_0$.

In order to minimize $J(g)$ we distinguish three cases. (i) If $x_0 \in \ker(Y_0) = \text{im}(\mathbf{1}_N)$ then we must have $x_0 \in \ker(X_0)$ as well, so $J(g) = 0$ for all θ and every $g > 0$ is optimal. (ii) If $x_0^\top Y_0 x_0 > 0$ and $x_0^\top X_0 x_0 = 0$ then no optimal $g > 0$ exists. (iii) If $x_0^\top Y_0 x_0 > 0$ and $x_0^\top X_0 x_0 > 0$ then an optimal $g > 0$ exists and can be shown to be equal to $g^* = \left(\frac{x_0^\top X_0 x_0}{x_0^\top Y_0 x_0} \right)^{\frac{1}{2}}$. It is clear that the computation of the optimal gain g requires exact knowledge of the network graph in the form of the Laplacian L . Also, the optimal gain clearly depends on the global initial state of the network.

IV. TOWARDS DECENTRALIZED COMPUTATION

In this section we will propose a new approach to compute ‘good’ local gains that can be computed in a decentralized way. Instead of doing this for the general LQ cost functional (4), we will zoom in on the particular case given by (6)-(7).

In order to decentralize the computation, instead of minimizing the global cost functional (6) for the multi-agent system (2), we write it as a sum of *local* cost functionals, one for each agent in the network.

More specifically, the associated local cost functional for agent i is given by

$$J_i(u_i) = \int_0^\infty q(x_i(t) - a_i(t))^2 + ru_i^2(t) dt, \quad (11)$$

where $a_i(t)$ is defined in (7), for $i = 1, 2, \dots, N$. This local cost functional penalizes the squared difference between the state of the i th agent and the average of the states of its neighboring agents (including itself), and the local control energy. By minimizing (11), agent i would make the difference between its own state and the average of the states of its neighbors (including itself) small. Note, however, that it is impossible for agent i to minimize this local cost functional since the trajectory $a_i(t)$ for $t \in [0, \infty)$ associated with the

neighboring agents is *not known*, so also not available to the i th agent. Thus, because direct minimization of (11) is impossible, as an alternative we will replace each of these local optimal control problems by a *sequence of linear quadratic tracking problems* that do turn out to be tractable.

More specifically, we choose a sampling period $T > 0$, and introduce the following sampling procedure. For each non-negative integer k , at time $t = kT$ the i th agent receives the sampled state value $x_j(kT)$ of its neighboring agents and takes the average of these, which is given by

$$a_i(kT) = \frac{1}{d_i + 1} (x_i(kT) + \sum_{j \in \mathcal{N}_i} x_j(kT)). \quad (12)$$

Then, the i th agent minimizes the cost functional

$$J_{i,k}(u) = \int_0^\infty e^{-2\alpha t} (q(x_i(t) - a_i(kT))^2 + ru_i^2(t)) dt. \quad (13)$$

In fact, this is a discounted linear quadratic tracking problem with constant reference signal $a_i(kT)$ and discount factor $\alpha > 0$. By solving this linear quadratic tracking problem, agent i obtains an optimal control law over an infinite time interval. However, agent i applies this control law only on the time interval $[kT, (k+1)T)$.

Then, at time $t = (k+1)T$ the above procedure is repeated, i.e., agent i receives the updated average $a_i((k+1)T)$, and subsequently solves the discounted tracking problem with cost functional $J_{i,k+1}(u)$ which involves the constant updated reference signal $a_i((k+1)T)$. By performing this control design procedure sequentially at each sampling time kT , we then obtain a single control law for agent i over the entire interval $[0, \infty)$.

Based on this control design procedure for the individual agents, we will obtain a distributed control protocol for the entire multi-agent system, simply by letting all agents compute their own control law. In the sequel we will analyze this protocol and show that it achieves consensus for the network.

Definition 1: A distributed control protocol is said to achieve consensus for the network if $x_i(t) - x_j(t) \rightarrow 0$ as $t \rightarrow \infty$ for all initial states of agents i and j , for all $i, j = 1, 2, \dots, N$.

In order to obtain an explicit expression for the control protocol proposed above, we will study the linear quadratic tracking problem for a single linear system. This will be done in the next section.

V. THE DISCOUNTED LQ TRACKING PROBLEM

In this section, we will deal with the discounted linear quadratic tracking problem for a given linear system. The linear quadratic tracking problem has been studied before, see, e.g., [20]. Here, however, we will solve it by transforming it into a standard linear quadratic control problem.

Consider the continuous-time linear time-invariant system

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0, \quad (14)$$

with $A \in \mathbb{R}^{n \times n}$ and $B \in \mathbb{R}^{n \times m}$, and where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ denote the state and the input, respectively. We assume that the pair (A, B) is stabilizable. Given is also a constant reference

signal $r_{\text{ref}}(t) = r$ with $r \in \mathbb{R}^n$. Next, we introduce a discounted quadratic cost functional given by

$$J(u) = \int_0^\infty e^{-2\alpha t} [(x(t) - r)^\top Q(x(t) - r) + u^\top(t)Ru(t)] dt \quad (15)$$

where $Q \in \mathbb{R}^{n \times n}$, $R \in \mathbb{R}^{m \times m}$ and $Q > 0$ and $R > 0$ are given weight matrices and $\alpha > 0$ is a discount factor [20]. The linear quadratic tracking problem is to determine for every initial state x_0 a piecewise continuous input function $u(t)$ that minimizes the cost functional (15).

To solve this problem, we introduce the variables

$$z(t) = e^{-\alpha t}x(t), \quad z_r(t) = e^{-\alpha t}r, \quad v(t) = e^{-\alpha t}u(t), \quad (16)$$

and denote $\xi(t) = (z^\top(t), z_r^\top(t))^\top$. Then we obtain an auxiliary system in terms of ξ and v , given by

$$\dot{\xi}(t) = A_e \xi(t) + B_e v(t), \quad \xi_0 = (x_0^\top, r^\top)^\top,$$

where $\xi_0 \in \mathbb{R}^{2n}$ is the initial state and

$$A_e = \begin{pmatrix} A - \alpha I_n & 0 \\ 0 & -\alpha I_n \end{pmatrix}, \quad B_e = \begin{pmatrix} B \\ 0 \end{pmatrix}.$$

In terms of the new variables ξ and v , the cost functional (15) can be written as $J(v) = \int_0^\infty \xi^\top(t)Q_e\xi(t) + v^\top(t)Rv(t) dt$, where $Q_e = \begin{pmatrix} Q & -Q \\ -Q & Q \end{pmatrix} \in \mathbb{R}^{2n \times 2n}$. The problem is now to find, for every initial state ξ_0 , a piecewise continuous input function $v(t)$ that minimizes this cost functional. This is a so-called a *free endpoint* standard LQ control problem, see [21, p. 218]. Since the pair (A, B) is stabilizable, the pair (A_e, B_e) is also stabilizable and hence the input function $v(t)$ that minimizes the cost functional $J(v)$ is generated by the feedback law

$$v(t) = -R^{-1}B_e^\top P_e^- \xi(t), \quad (17)$$

where $P_e^- \in \mathbb{R}^{2n \times 2n}$ is the smallest positive semi-definite solution of the Riccati equation

$$A_e^\top P_e^- + P_e^- A_e - P_e^- B_e R^{-1} B_e^\top P_e^- + Q_e = 0. \quad (18)$$

Now, partition $P_e^- := \begin{pmatrix} P_1 & P_{12} \\ P_{12}^\top & P_2 \end{pmatrix}$, where all blocks have dimension $n \times n$. Recalling (16) and (17), we then immediately find an expression for the input function $u(t)$ that minimizes the cost functional (15) for the system (14) and reference signal $r_{\text{ref}}(t) = r$.

Theorem 1: The input function $u(t)$ that minimizes the cost functional (15) is generated by the control law

$$u(t) = K_1 x(t) + K_2 r, \quad (19)$$

where $K_1 = -R^{-1}B^\top P_1$ and $K_2 = -R^{-1}B^\top P_{12}$.

The proof follows immediately from the above considerations. See also [20].

Remark 1: Let $e(t) := x(t) - r$ denote the tracking error. Because $Q > 0$, the control law (19) only guarantees that $\bar{e}(t) := e^{-\alpha t}e(t)$ tends to zero as t goes to infinity. Thus, the feedback law that minimizes the LQ tracking cost functional (15) only guarantees the actual tracking error $e(t)$ to be

exponentially bounded with growth rate $\alpha > 0$. Note that $\alpha > 0$ can be taken arbitrarily small.

It will be shown however that, for the multi-agent system case, the control design method established in this section will, nevertheless, lead to a protocol that achieves consensus.

VI. CONSENSUS ANALYSIS

In this section, we will show that, by adopting the control design method for the multi-agent system (2) as proposed in Section IV, the resulting distributed control protocol achieves consensus for the entire network.

As already explained in Section IV, we choose a sampling period $T > 0$ and introduce a sampling procedure. For each nonnegative integer k , at time $t = kT$ the i th agent receives the sampled state value of its neighboring agents (including itself) and minimizes the cost functional (13), which is a discounted linear quadratic tracking problem with constant reference signal $r_{\text{ref}}(t) = a_i(kT)$ and discount factor $\alpha > 0$.

According to the theory on the discounted LQ tracking problem described in Section V, the local optimal control law for agent i at time $t = kT$ over the whole time horizon $[0, \infty)$ is therefore of the form

$$u_{i,k}(t) = g_{i,k}x_i(t) + g'_{i,k}a_i(kT), \quad (20)$$

in which the control gains $g_{i,k}$ and $g'_{i,k}$ can be computed explicitly by solving the Riccati equation (18) associated with the LQ tracking problem for agent i .

Lemma 2: Consider, at time $t = kT$, the i th agent of the multi-agent system (1) with associated local cost functional (13). Denote

$$\bar{A} = \begin{pmatrix} -\alpha & 0 \\ 0 & -\alpha \end{pmatrix}, \quad \bar{B} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \bar{Q} = \begin{pmatrix} q & -q \\ -q & q \end{pmatrix}.$$

Let $\bar{P} := \begin{pmatrix} p_1 & p_{12} \\ p_{12} & p_2 \end{pmatrix}$ be the smallest positive semi-definite solution of the Riccati equation

$$\bar{A}^\top \bar{P} + \bar{P} \bar{A} - r^{-1} \bar{P} \bar{B} \bar{B}^\top \bar{P} + \bar{Q} = 0. \quad (21)$$

Then the local control law (20) with $g_{i,k} := -r^{-1}p_1$ and $g'_{i,k} := -r^{-1}p_{12}$ minimizes the cost (13) for agent i .

Proof: This follows immediately from Theorem 1. \blacksquare

Next, agent i applies the control law (20) only on the time interval $[kT, (k+1)T)$. Then, at time $t = (k+1)T$ the above procedure is repeated.

Since, for all $i = 1, 2, \dots, N$ and $k = 0, 1, \dots$, the matrices \bar{A} , \bar{B} and \bar{Q} are independent of i and k , the same holds for the gains $g_{i,k}$ and $g'_{i,k}$. In the sequel, we will therefore drop the subscripts in the control gains $g_{i,k}$ and $g'_{i,k}$ and denote them by g and g' , respectively. Moreover, using (21), we compute $g = r^{-1}(\alpha - \sqrt{\alpha^2 + rq})$ and $g' = -g$.

By performing this procedure sequentially at each sampling time kT , we then obtain a single control law for agent i over the entire interval $[0, \infty)$ as

$$u_{i,k}(t) = gx_i(t) - ga_i(kT), \quad t \in [kT, (k+1)T), \quad (22)$$

where $g = r^{-1}(\alpha - \sqrt{\alpha^2 + rq}) < 0$.

Recall that $a(t) = Gx(t)$, with G given by (8), and that $a(kT) = Gx(kT)$. Therefore, the local control laws for the individual agents lead to a distributed control protocol

$$u_k(t) = gx(t) - gGx(kT), \quad t \in [kT, (k+1)T). \quad (23)$$

Now, by applying the protocol (23) to the multi-agent system (1), we find that the controlled network is represented by

$$\dot{x}(t) = gx(t) - gGx(kT), \quad t \in [kT, (k+1)T). \quad (24)$$

In the remainder of this section, we will analyze this representation, and show that consensus is achieved, i.e., for each initial state $x(0) = x_0$ we have $x_i(t) - x_j(t) \rightarrow 0$ as t tends to infinity.

In order to do this, note that the solution of (24) with initial state $x(0) = x_0$ is given by

$$x(t) = e^{g(t-kT)}x(kT) - \int_{kT}^t e^{g(t-\tau)}gGx(kT) d\tau, \quad (25)$$

for $t \in [kT, (k+1)T)$, $k = 0, 1, 2, \dots$. Obviously, for each initial state x_0 , the corresponding solution $x(t)$ is continuous. From (25) we see that the sequence of network states $x(kT)$ evaluated at the discrete time instances kT , $k = 0, 1, \dots$ satisfies the difference equation

$$\begin{aligned} x((k+1)T) &= \Gamma x(kT), \\ \Gamma &= e^{gT}I_N - (e^{gT} - 1)G \in \mathbb{R}^{N \times N}. \end{aligned} \quad (26)$$

Clearly, the network reaches consensus if and only if for each x_0 , $x_i(kT) - x_j(kT) \rightarrow 0$ as t tends to infinity.

We proceed with analyzing the eigenvalues of G .

Lemma 3: The matrix G has an eigenvalue 1 with algebraic multiplicity equal to one and associated eigenvector $\mathbf{1}_N$. The remaining eigenvalues of G are all real and have absolute value strictly less than 1.

Proof: Since $L = D - A$, we have $G = I_N - (D + I_N)^{-1}L$. Hence we have $\tilde{D}^{\frac{1}{2}}G\tilde{D}^{-\frac{1}{2}} = I_N - \tilde{D}^{-\frac{1}{2}}L\tilde{D}^{-\frac{1}{2}}$ where $\tilde{D} = D + I_N$. Note that the right hand side is symmetric and hence has only real eigenvalues. Thus, by matrix similarity, G also has only real eigenvalues.

Next, we show that G has a simple eigenvalue 1 with associated eigenvector $\mathbf{1}_N$. First note that

$$G\mathbf{1}_N = (I_N - (D + I_N)^{-1}L)\mathbf{1}_N = \mathbf{1}_N. \quad (27)$$

Hence, indeed, 1 is an eigenvalue of G with eigenvector $\mathbf{1}_N$. Since G is similar to a symmetric matrix, it is diagonalizable, so the algebraic multiplicity of its eigenvalue 1 must be equal to its geometric multiplicity. Suppose now that 1 is not a simple eigenvalue. Then there must exist a second eigenvector, say v , which is linearly independent of $\mathbf{1}_N$. This implies $Gv = v$. Then $Lv = 0$, so v must be a multiple of $\mathbf{1}_N$. This is a contradiction. We conclude that the eigenvalue 1 is indeed simple.

Finally, it follows from Gershgorin's Theorem [22] that every eigenvalue λ of G satisfies $-1 < \lambda \leq 1$. \blacksquare

Before we give the main result of this letter, we first review the following proposition.

Proposition 1: Consider the discrete-time system

$$x(k+1) = Ax(k), \quad x(0) = x(0), \quad y(k) = Cx(k)$$

with $A \in \mathbb{R}^{n \times n}$ and $C \in \mathbb{R}^{p \times n}$, where $x(k) \in \mathbb{R}^n$ is the state, x_0 is the initial state and $y(k) \in \mathbb{R}^p$ is the output. Then,

$y(k) \rightarrow 0$ as $k \rightarrow \infty$ for all initial states x_0 if and only if $X_+(A) \subset \ker(C)$. Here, $X_+(A)$ is the unstable subspace, i.e., the sum of the generalized eigenspaces of A associated with its eigenvalues in $\{\lambda \in \mathbb{C} \mid |\lambda| \geq 1\}$.

Proof: A proof can be given by generalizing the results [21, p. 99] to the discrete time case. ■

We are now ready to present the main result of this letter.

Theorem 2: Consider the multi-agent system (1). Let $T > 0$ be a sampling period, $\alpha > 0$ a discount factor, and let $q, r > 0$ be given weights. Let \bar{P} be the smallest positive semi-definite solution of the Riccati equation (21) and partition $\bar{P} := \begin{pmatrix} p_1 & p_{12} \\ p_{12} & p_2 \end{pmatrix}$. Then the distributed control protocol (23) with $g = -r^{-1}p_1$ and $g' = -r^{-1}p_{12}$ achieves consensus for the controlled network (24).

Proof: The network reaches consensus if and only if $Lx(kT) \rightarrow 0$ as $k \rightarrow \infty$. Since $\ker(L) = \text{im}(\mathbf{1}_N)$, it then follows from Proposition 1 that consensus is achieved if and only if $X_+(\Gamma) \subset \ker(L)$, equivalently, the sum of the generalized eigenspaces of Γ corresponding to the eigenvalues λ with $|\lambda| \geq 1$ is equal to $\text{im}(\mathbf{1}_N)$.

Indeed, we will show that all eigenvalues λ of Γ are real and satisfy $-1 < \lambda \leq 1$, and $\lambda = 1$ is a simple eigenvalue with associated eigenvector $\mathbf{1}_N$.

Recall that $\Gamma = e^{gT}I_N - (e^{gT} - 1)G$. Hence, μ is an eigenvalue of Γ if and only if $\mu = e^{gT} - \lambda(e^{gT} - 1)$ where λ is an eigenvalue of G . It was shown in Lemma 3 that all eigenvalues λ of G are real and satisfy $-1 < \lambda \leq 1$ and, moreover, $\lambda = 1$ is a simple eigenvalue. Using the fact that $g < 0$ we thus obtain that the eigenvalues μ of Γ satisfy $-1 < \mu \leq 1$ and $\mu = 1$ is a simple eigenvalue of Γ .

Finally, we will show $\mu = 1$ has eigenvector $\mathbf{1}_N$. Indeed, this follows from $\Gamma \mathbf{1}_N = (e^{gT}I_N - (e^{gT} - 1)G)\mathbf{1}_N = \mathbf{1}_N$. This completes the proof. ■

Remark 2: By analyzing the eigenvalues μ of Γ satisfying $-1 < \mu < 1$, it can be seen that, for given α , the convergence rate of the difference equation (26) increases with increasing sampling period T . The total time it takes to reach a disagreement smaller than a given tolerance is then the product of the number of iterations in (26) and this sampling period. It might therefore be more advantageous to use a smaller sampling period with a larger number of required iterations, but yet leading to a smaller total time. In other words, the choice of sampling period is a trade-off between the total time required to obtain an acceptable disagreement, and the number of iterations in (26).

For a simulation example, we refer to [23].

VII. CONCLUSION

We have studied the distributed linear quadratic control problem for a network of agents with single integrator dynamics. We have shown that the computation of control gains that minimize global cost functionals need global information, in particular the initial states of all agents and the Laplacian matrix. We have also shown that this drawback can be overcome by transforming the global cost functional into discounted local cost functionals and assigning each of these to an associated agent. In such a way, each agent computes its

own control gain, using sampled information of its neighboring agents. Finally, we have shown that the resulting control protocol achieves consensus for the network.

REFERENCES

- [1] J. Jiao, H. L. Trentelman, and M. K. Camlibel. (2018). *A Suboptimality Approach to Distributed Linear Quadratic Optimal Control*. [Online]. Available: <https://arxiv.org/abs/1803.02682>
- [2] Y. Cao and W. Ren, "Optimal linear-consensus algorithms: An LQR perspective," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 3, pp. 819–830, Jun. 2010.
- [3] F. Borrelli and T. Keviczky, "Distributed LQR design for identical dynamically decoupled systems," *IEEE Trans. Autom. Control*, vol. 53, no. 8, pp. 1901–1912, Sep. 2008.
- [4] A. Mosebach and J. Lunze, "Optimal synchronization of circulant networked multi-agent systems," in *Proc. Eur. Control Conf. (ECC)*, Jul. 2013, pp. 3815–3820.
- [5] A. Mosebach and J. Lunze, "Synchronization of autonomous agents by an optimal networked controller," in *Proc. Eur. Control Conf. (ECC)*, Jun. 2014, pp. 208–213.
- [6] P. Deshpande, P. P. Menon, C. Edwards, and I. Postlethwaite, "A distributed control law with guaranteed LQR cost for identical dynamically coupled linear systems," in *Proc. Amer. Control Conf. (ACC)*, Jun. 2011, pp. 5342–5347.
- [7] K. H. Movric and F. L. Lewis, "Cooperative optimal control for multi-agent systems on directed graph topologies," *IEEE Trans. Autom. Control*, vol. 59, no. 3, pp. 769–774, Mar. 2014.
- [8] H. Zhang, T. Feng, G.-H. Yang, and H. Liang, "Distributed cooperative optimal control for multiagent systems on directed graphs: An inverse optimal approach," *IEEE Trans. Cybern.*, vol. 45, no. 7, pp. 1315–1326, Jul. 2015.
- [9] E. Semsar-Kazerooni and K. Khorasani, "Multi-agent team cooperation: A game theory approach," *Automatica*, vol. 45, no. 10, pp. 2205–2213, 2009.
- [10] D. H. Nguyen, "A sub-optimal consensus design for multi-agent systems based on hierarchical LQR," *Automatica*, vol. 55, pp. 88–94, May 2015.
- [11] D. H. Nguyen, "Reduced-order distributed consensus controller design via edge dynamics," *IEEE Trans. Autom. Control*, vol. 62, no. 1, pp. 475–480, Jan. 2017.
- [12] V. Rezaei and M. Stefanovic, "Distributed decoupling of linear multiagent systems with mixed matched and unmatched state-coupled nonlinear uncertainties," in *Proc. Amer. Control Conf. (ACC)*, May 2017, pp. 2693–2698.
- [13] Z. Li and Z. Ding, "Fully distributed adaptive consensus control of multi-agent systems with LQR performance index," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2015, pp. 386–391.
- [14] K. G. Vamvoudakis, F. L. Lewis, and G. R. Hudas, "Multi-agent differential graphical games: Online adaptive learning solution for synchronization with optimality," *Automatica*, vol. 48, no. 8, pp. 1598–1611, 2012.
- [15] H. Modares, S. P. Nageshroo, G. A. D. Lopes, R. Babuška, and F. L. Lewis, "Optimal model-free output synchronization of heterogeneous systems using off-policy reinforcement learning," *Automatica*, vol. 71, pp. 334–341, Sep. 2016.
- [16] J. M. Montenbruck, G. S. Schmidt, G. S. Seyboth, and F. Allgöwer, "On the necessity of diffusive couplings in linear synchronization problems with quadratic cost," *IEEE Trans. Autom. Control*, vol. 60, no. 11, pp. 3029–3034, Nov. 2015.
- [17] H. J. van Waarde, M. K. Camlibel, and H. L. Trentelman, "Comments on 'On the necessity of diffusive couplings in linear synchronization problems with quadratic cost,'" *IEEE Trans. Autom. Control*, vol. 62, no. 6, pp. 3099–3101, Jun. 2017.
- [18] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.
- [19] H.-J. Jongsma, P. Mlinarić, S. Grundel, P. Benner, and H. L. Trentelman, "Model reduction of linear multi-agent systems by clustering with \mathcal{H}_2 and \mathcal{H}_∞ error bounds," *Math. Control Signals Syst.*, vol. 30, p. 6, Mar. 2018.
- [20] H. Modares and F. L. Lewis, "Linear quadratic tracking control of partially-unknown continuous-time systems using reinforcement learning," *IEEE Trans. Autom. Control*, vol. 59, no. 11, pp. 3051–3056, Nov. 2014.
- [21] H. L. Trentelman, A. A. Stoorvogel, and M. Hautus, *Control Theory for Linear Systems*. London, U.K.: Springer-Verlag, 2001.
- [22] R. Horn and C. Johnson, *Matrix Analysis*. New York, NY, USA: Cambridge Univ. Press, 1990.
- [23] J. Jiao, H. L. Trentelman, and M. K. Camlibel. (2019). *Distributed Linear Quadratic Optimal Control: Compute Locally and Act Globally*. [Online]. Available: <https://arxiv.org/abs/1902.11244>