Distributed outer approximation of the intersection of ellipsoids

Rodrigo Aldana-López, Eduardo Sebastián, Rosario Aragüés, Eduardo Montijano and Carlos Sagüés

Abstract— The outer Löwner-John method is widely used in sensor fusion applications to find the smallest ellipsoid that can approximate the intersection of a set of ellipsoids, described by positive definite covariance matrices modeling the quality of each sensor. We propose a distributed algorithm to solve this problem when these matrices are defined over the network's nodes. This is of particular significance as it is the first decentralized algorithm capable of computing the covariance intersection ellipsoid by combining information from the entire network using only local interactions. The solution is based on a reformulation of the centralized problem, leading to a local protocol based on exact dynamic consensus tools. After reaching consensus, the protocol converges to an outer Löwner-John ellipsoid in finite time, and to the global optimum asymptotically. Formal convergence analysis and numerical experiments are provided to validate the proposal's advantages.

I. INTRODUCTION

The Löwner-John (L-J) methods [15] are a series of ellipsoidal approximations for convex sets. Of particular interest are the convex sets generated by the intersection of n-dimensional ellipsoids, described by symmetric and positive semidefinite n-dimensional matrices. L-J methods have a significant presence in a wide variety of applications [18], such as robust control [14] or statistical analysis [20]. Specially important is in the field of data fusion and state estimation [17], where the ellipsoids represent the measurements or estimates' uncertainty. However, there are no distributed algorithms to compute L-J ellipsoids, despite their potential application in sensor networks where the measurements are scattered over a communication network [29].

In sensor fusion, both the inner and outer L-J ellipsoidal approximations are widely explored under the name of Covariance Intersection Method (CIM) [17]. The CIM is posed as a convex optimization program yielding the largest outer approximation of the intersection of two ellipsoids described by two covariance matrices. Since then, different works have applied variants of the CIM. For example, a sequential procedure of fusion of two ellipsoids [8] can be used to extend the CIM to N ellipsoids [24]. On the other hand, recent distributed Kalman filters [13], [16], [30]

propose CIM variants which directly consider N ellipsoids via approximations. More recently, the outer L-J method [29] has been explored for the fusion of the estimates and to certify the estimation with optimality guarantees. All these works are particular instances of L-J methods. However, rather than finding the global L-J ellipsoids in a distributed manner, they use the CIM to approximate the fusion of neighboring estimates.

The computation of L-J ellipsoids can be posed as an optimization program [15]. In this sense, distributed optimization methods [22] have been intensively researched in recent years, where a general assumption [4], [9], [21], [25] is that the global cost function is the sum of local objectives. In the case of L-J methods, the cost function is not separable. This separability issue extends to the constraints of the outer L-J methods, where a coupled equality constraint holds. To address coupled equality constraints in sensor fusion, the Inverse CIM [26] computes local bounds to ensure consistency, whereas other methods rely on fusion centers [6] to gather the distributively pre-processed covariance matrices. Instead, we propose a distributed projected gradient flow [12] protocol that converges to the coupled equality constraint in a prescribed time. In more general settings, the literature proposes distributed dual sub-gradient algorithms [5], [28], primal relaxations [27], the so-called "subgradient push" [23] or neural-network-based approaches [19]. In contrast, our method is suitable for non-separable optimization objectives.

Motivated by this discussion, we develop, for the first time, a distributed protocol that computes the outer approximation of the intersection of ellipsoids when only local information and distributed communications are available. The proposal is based on a continuous-time Exact Dynamic Consensus protocol (EDC), which extends previous protocols [1], [2], [11] to converge before a prescribed time. Consequently, we can certify the moment in which an outer L-J ellipsoid is already available. The quality of the outer L-J ellipsoid is further improved through a distributed optimization step based on the projected gradient flow [12], such that the global optimum is asymptotically found. We discuss how our proposal can be used for CIM in sensor fusion.

Notation: $\mathrm{tr}(\bullet)$, $\det(\bullet)$ denote the trace and determinant. $\mathbb{R}, \mathbb{R}_{\geq 0}, \mathbb{R}_{>0}$ denote the real, non-negative and positive reals respectively. $0 \preceq \mathbf{P}$ denotes when a matrix $\mathbf{P} \in \mathbb{R}^{n \times n}$ is positive semi-definite. $\mathbf{0}, \mathbf{I}$ denote the zero and identity matrices. For an undirected graph $\mathcal{G}$, denote with $\lambda_{\mathcal{G}}$ its standard algebraic connectivity (see [2]).

II. PROBLEM STATEMENT

Consider a set of N agents which communicate through a

All the authors are with the Instituto de Investigación en Ingeniería de Aragón, Universidad de Zaragoza, Spain (email:rodrigo.aldana.lopez@gmail.com, esebastian@unizar.es, raragues@unizar.es, emonti@unizar.es, csagues@unizar.es)

μ	$f(\mathbf{Q}(\mathbf{x}))$	$g_i(x_i, \mathbf{Q}(\mathbf{x}))$
0	$-\mathrm{tr}(\mathbf{Q}(\mathbf{x}))$	$-2x_i\mathrm{tr}(\mathbf{P}_i^{-1})$
1	$\log(\det(\mathbf{Q}(\mathbf{x})^{-1}))$	$-2x_i\mathrm{tr}(\mathbf{Q}(\mathbf{x})^{-1}\mathbf{P}_i^{-1})$
2	$\mathrm{tr}(\mathbf{Q}(\mathbf{x})^{-1})$	$-2x_i\mathrm{tr}(\mathbf{Q}(\mathbf{x})^{-1}\mathbf{P}_i^{-1}\mathbf{Q}(\mathbf{x})^{-1})$

TABLE I: Different ellipsoidal size measure functions $f(\bullet)$ with $\mathbf{Q}(\mathbf{x}) = (1/\mathsf{N})\sum_{i=1}^{\mathsf{N}} x_i^2 \mathbf{P}_i^{-1}$ as well as the components of the gradient $\nabla f(\mathbf{Q}(\mathbf{x})) = [g_1(x_1, \mathbf{Q}(\mathbf{x})), \dots, g_{\mathsf{N}}(x_{\mathsf{N}}, \mathbf{Q}(\mathbf{x}))]^\top /\mathsf{N}$. We label each choice by μ for its reference in Theorem 1.

communication network modeled by an undirected connected graph $\mathcal{G} = (\mathcal{I}, \mathcal{F})$, where $\mathcal{I}$ and $\mathcal{F}$ are the sets of nodes and edges respectively. We denote by $\mathcal{N}_i \subseteq \mathcal{I}$ the index set of neighbors for agent $i \in \mathcal{I}$. Each agent i is described by a $n-$dimensional ellipsoidal set $\mathcal{E}(\mathbf{P}_i) = \{\mathbf{y} \in \mathbb{R}^n : \mathbf{y}^\top \mathbf{P}_i^{-1}\mathbf{y} \le 1\}$, given $\mathbf{0} \prec \mathbf{P}_i \in \mathbb{R}^{n \times n}$.

The goal for the agents is to cooperate to find an ellipsoidal set $\mathcal{E}(\mathbf{P})$ covering $\check{\mathcal{E}} := \bigcap_{i=1}^{\mathsf{N}} \mathcal{E}(\mathbf{P}_i)$, characterized by some $\mathbf{P} \succ 0$. A wide family of ellipsoidal sets covering $\check{\mathcal{E}}$ is the one parameterized by $\mathbf{P}(\boldsymbol{\lambda})^{-1} := \sum_{i=1}^{\mathsf{N}} \lambda_i \mathbf{P}_i^{-1}$ with $\sum_{i=1}^{\mathsf{N}} \lambda_i = 1$ and $\boldsymbol{\lambda} = [\lambda_1, \dots, \lambda_{\mathsf{N}}]^\top \in \mathbb{R}_{\ge 0}^{\mathsf{N}}$. It can be verified that $\mathcal{E}(\mathbf{P}(\boldsymbol{\lambda})) \supset \check{\mathcal{E}}$ [15]. Therefore, designing $\mathbf{P}(\boldsymbol{\lambda})^{-1}$ as a convex combination of $\{\mathbf{P}_i^{-1}\}_{i=1}^{\mathsf{N}}$ defines a family of L-J outer ellipsoids for the intersection of $\{\mathcal{E}(\mathbf{P}_i)\}_{i=1}^{\mathsf{N}}$. The weights $\boldsymbol{\lambda}$ can then be optimized by solving the following optimization program:

$$\min_{\boldsymbol{\lambda} \in \mathbb{R}_{\ge 0}^{\mathsf{N}}} f(\mathbf{P}(\boldsymbol{\lambda})^{-1}), \quad \text{such that} \quad \sum_{i=1}^{\mathsf{N}} \lambda_i = 1, \qquad (1)$$

where $f(\bullet)$ is a function that measures the size of $\mathcal{E}(\mathbf{P}(\boldsymbol{\lambda}))$. For instance, $f(\bullet) = \log(\det(\bullet))$ can be used to minimize the volume of $\mathcal{E}(\mathbf{P}(\boldsymbol{\lambda}))$. Other popular choices of $f(\bullet)$ are in Table I. The purpose of this work is to design a distributed protocol that finds the optimum of (1). Due to the numerical difficulties found when dealing with equality constraints in practice, we recast (1) into

$$\min_{\mathbf{x} \in \mathcal{C}} f(\mathbf{Q}(\mathbf{x})), \quad \mathbf{Q}(\mathbf{x}) := 1/\mathsf{N}\sum_{i=1}^{\mathsf{N}} x_i^2 \mathbf{P}_i^{-1}, \qquad (2)$$

where instead of an equality constraint, the solution is restricted to a wider feasible manifold $\mathcal{C} = \{\mathbf{x} \in \mathbb{R}^{\mathsf{N}} : 1-\varepsilon \le s(\mathbf{x}) \le 1\}$ of $\mathbf{x} = [x_1, \dots, x_{\mathsf{N}}]^\top$, $s(\mathbf{x}) := \|\mathbf{x}\|^2/\mathsf{N}$ and $\varepsilon \in (0, 1)$. By using the change of coordinates $\lambda_i = x_i^2/\mathsf{N}$, we ensure $\lambda_i \in \mathbb{R}_{\ge 0}$ and that the unique minimizer of (1) follows from the minimizer of (2) when $\varepsilon = 0$. For any other $\varepsilon > 0$, we use (2) to approximate solutions of (1) with arbitrary accuracy dictated by the size of ε. In the following section, we provide a distributed algorithm to solve (2).

III. DISTRIBUTED OUTER ELLIPSE COMPUTATION

To solve problem (2) in a distributed fashion, we propose a novel distributed protocol based on Projected Gradient Flow (PGF) [12] methods. The idea is to drive the trajectories of $\mathbf{x}(t)$ towards the feasible manifold where the equality constraint in (1) holds. Once there, the trajectories of $\mathbf{x}(t)$

flow towards the optimum while fulfilling the equality constraint. To do so, we set a suitable virtual control action $\dot{x}_i(t) = u_i(t)$ using only local information.

To design a PGF-based protocol, we first need agreement on some global quantities across the nodes of the network. Therefore, the first stage of our algorithm computes local estimates $\{\hat{s}_i(t), \hat{\mathbf{Q}}_i(t)\}_{i=1}^{\mathsf{N}}$ for $s(\mathbf{x}(t)), \mathbf{Q}(\mathbf{x}(t))$ using the following EDC protocols:

$$\begin{aligned}
\hat{s}_i(t) &= x_i(t)^2 - v_i(t) \\
\dot{v}_i(t) &= \kappa_s \sum_{j \in \mathcal{N}_i} \phi\left(\hat{s}_j(t) - \hat{s}_i(t); \zeta_s; q\right) \\
\hat{\mathbf{Q}}_i(t) &= x_i(t)^2 \mathbf{P}_i^{-1} - \mathbf{V}_i(t) \\
\dot{\mathbf{V}}_i(t) &= \kappa_{\mathbf{Q}} \sum_{j \in \mathcal{N}_i} \phi\left(\hat{\mathbf{Q}}_j(t) - \hat{\mathbf{Q}}_i(t); \zeta_{\mathbf{Q}}; q\right)
\end{aligned} \qquad (3)$$

with auxiliary variables $\mathbf{V}_i(t), v_i(t)$ initialized as $\mathbf{V}_i(0) = \mathbf{0}, v_i(0) = 0$. Moreover, $\kappa_{\mathbf{Q}}, \kappa_s, \zeta_{\mathbf{Q}}, \zeta_s > 0, q \in (0, 1)$ are design parameters. We use $\phi(\bullet; \zeta; q) = (|\bullet|^{1-q} + |\bullet|^{1+q} + \zeta)\mathrm{sign}(\bullet)$ for a scalar parameter $\bullet$, and component-wise for $\bullet$ of any other dimension. As will be proven in Section IV, after a transient of prescribed duration T_c, these estimates will comply $\hat{\mathbf{Q}}_i(t) \equiv \mathbf{Q}(\mathbf{x}(t)), \hat{s}_i(t) \equiv s(\mathbf{x}(t)), \forall t \ge T_c$ for suitable $\kappa_{\mathbf{Q}}, \kappa_s$. This is possible since $\zeta_{\mathbf{Q}}, \zeta_s$ introduce a discontinuous sliding mode term in ϕ allowing (3) to achieve EDC even with time-varying consensus inputs. Then, all agents estimate its local component of the gradient $g_i(x_i(t), \mathbf{Q}(\mathbf{x}(t))) \equiv g_i(x_i(t), \hat{\mathbf{Q}}_i(t)), \forall t \ge T_c$, as in Table I.

During the first consensus stage defined in (3), we set a control action $u_i(t) = 0, \forall t \in [0, T_c]$. The second stage of our algorithm consists of taking the arbitrary initial conditions $x_i(0) = x_i(T_c)$ and update $x_i(t)$ towards $\mathcal{C}$. Then, nodes do PGF to find the global optimum of (2). This is achieved by a discontinuous controller. For all $t \ge T_c$:

$$u_i(t) = \begin{cases} \kappa_{\mathcal{C}} x_i(t)\mathrm{sign}\left(\left(1-\frac{\varepsilon}{2}\right) - \hat{s}_i(t)\right), & \hat{s}_i(t) \notin [1 - \varepsilon, 1] \\ -\kappa_{\mathcal{C}} g_i(x_i(t), \hat{\mathbf{Q}}_i(t)), & \hat{s}_i(t) \in [1 - \varepsilon, 1] \end{cases} \qquad (4)$$

with design parameter $\kappa_{\mathcal{C}} > 0$, which leads to:

$$\dot{\mathbf{x}}(t) = \begin{cases} \kappa_{\mathcal{C}} \mathbf{x}(t)\mathrm{sign}\left(\left(1 - \frac{\varepsilon}{2}\right) - s(\mathbf{x}(t))\right), & \mathbf{x}(t) \notin \mathcal{C} \\ -\kappa_{\mathcal{C}} \mathsf{N}\nabla f(\mathbf{Q}(\mathbf{x}(t))), & \mathbf{x}(t) \in \mathcal{C} \end{cases} \qquad (5)$$

under the synchronization conditions $\hat{\mathbf{Q}}_i(t) \equiv \mathbf{Q}(\mathbf{x}(t))$, $\hat{s}_i(t) \equiv s(\mathbf{x}(t))$. Using the idea of PGF, when $\mathbf{x}(t) \notin \mathcal{C}$, $\mathbf{x}(t)$ flows towards $\mathcal{C}$, in order to fulfill the equality constraint of the outer L-J method in (1). On the other hand, for $\mathbf{x}(t) \in \mathcal{C}$, $\mathbf{x}(t)$ flows in the direction opposite to the gradient, towards the optimum of the outer L-J method in (1). As we discuss in Section IV, this results in $\mathbf{x}(t)$ flowing in the direction of the projected gradient of $f(\mathbf{Q}(\mathbf{x}(t)))$ with respect to $\mathcal{C}$ in any case, allowing to maintain feasible trajectories and converge to the optimum. In the following, we state our main result as well as a practical assumption under which it holds.

Assumption 1. *Let $0 < \underline{b} < \overline{b}, \overline{b} > 1$ and $\underline{\sigma}, \overline{\sigma} > 0$. Then, $x_i(0) \in [\underline{b}, \overline{b}]$ and $\underline{\sigma}\mathbf{I} \preceq \mathbf{P}_i^{-1} \preceq \overline{\sigma}\mathbf{I}, \forall i \in \mathcal{I}$, and with p the maximum scalar component among all $\{\mathbf{P}_i\}_{i=1}^{\mathsf{N}}$.*

Theorem 1. *Let $\mathcal{G}$ be a connected undirected graph with N nodes, ℓ edges, algebraic connectivity $\lambda_{\mathcal{G}}$, and consider*

protocols (3). *Let* $\kappa_{\mathcal{C}} > 0, \varepsilon \in (0,1)$, *and Assumption 1 hold. Let* $f(\bullet)$ *in Table I be labeled by* $\mu \in \{0,1,2\}$ *and*

$$h(\mathsf{N}) = \kappa_{\mathcal{C}} \max\{\sqrt{\mathsf{N}\bar{b}}, 2\bar{b}\bar{\sigma}\mathsf{N}^{\mu+1}(\underline{\sigma}\min\{\underline{b}^2, 1-\varepsilon\})^{-\mu}\}.$$

Let $\dot{x}_i(t) = u_i(t)$ *with* $u_i(t) = 0, \forall t \in [0, T_c]$ *and* $u_i(t)$ *defined as in* (4) *for* $t \geq T_c$. *For any* $\kappa_{\mathcal{C}} > 0$, *if*

$$\kappa_s, \kappa_{\mathbf{Q}} > \frac{\ell\pi}{q\lambda_{\mathcal{G}}T_c}, \ \zeta_s > \frac{4\bar{b}h(\mathsf{N})}{\kappa_s\sqrt{\lambda_{\mathcal{G}}}}, \ \zeta_{\mathbf{Q}} > \frac{4p\bar{b}h(\mathsf{N})}{\kappa_{\mathbf{Q}}\sqrt{\lambda_{\mathcal{G}}}}, \quad (6)$$

then, there exists $T_\varepsilon > 0$ *such that* $\mathbf{x}(t) \in \mathcal{C}, \forall t \geq T_c + T_\varepsilon$. *In addition,* $\mathcal{E}(\mathbf{Q}(\mathbf{x}(t))^{-1}) \supset \check{\mathcal{E}}, \forall t \geq T_c + T_\varepsilon$ *and* $f(\mathbf{Q}(\mathbf{x}(t)))$ *converges asymptotically towards the optimum of* (2).

Remark 1. *After consensus has been reached for* $t \geq T_c$, *each agent can check the condition* $\hat{s}_i(t) = s(\mathbf{x}(t)) \in [1 - \varepsilon, 1]$ *to verify if* $\mathbf{x}(t) \in \mathcal{C}$ *and compute* T_ε. *Hence, the proposed algorithm obtains an outer L-J ellipsoid from* $t = T_c + T_\varepsilon$, *since* $\mathbf{x}(t) \in \mathcal{C} \ \forall t \geq T_c + T_\varepsilon$. *Thus, the ellipsoid* $\mathcal{E}(\hat{\mathbf{Q}}_i(t)^{-1}) = \mathcal{E}(\mathbf{Q}(\mathbf{x}(t))^{-1})$ *is already valid to use. Note that the greater* t, *the tighter* $\mathcal{E}(\mathbf{Q}(\mathbf{x}(t))^{-1})$, *so the user can design the algorithm depending on the computing resources available and the desired accuracy for the L-J ellipsoid.*

Remark 2. *The powers* $1 - q, 1 + q$ *with* $q \in (0,1)$ *in* ϕ *are known to induce a fixed settling time bound, regardless of the initial conditions. This allows all agents to make sure consensus has been reached before* $t = T_c$ *without having to individually check this condition, avoiding additional synchronization issues. The function* ϕ *along with* q *and* κ_s, ζ_s *(similarly* $\kappa_{\mathbf{Q}}, \zeta_{\mathbf{Q}}$*) were introduced for static consensus with disturbances in [2].*

A. Application to sensor fusion

In this section, we describe how our proposal is applied to sensor fusion, and how it relates to other methods in this context. A standard sensor fusion problem consists of estimating a quantity of interest $\mathbf{p} \in \mathbb{R}^n$ where each agent has access to an estimate $\hat{\mathbf{p}}_i \in \mathbb{R}^n$ of $\mathbf{p}$ under Gaussian uncertainty with covariance $\mathbf{P}_i$. For example, such estimates can be obtained at discrete-time instants, by sampling from a noisy sensor and estimating $\mathbf{p}$ with a Kalman filter. In this case, estimates $\{\hat{\mathbf{p}}_i\}_{i=1}^{\mathsf{N}}$ are correlated such that a sub-optimal information fusion strategy might be adopted. A popular choice is the CIM [24], yielding an optimization program within the class of problems described by (1). Our method is used to compute the global CIM estimate across the network by the sensor fusion rule

$$\hat{\mathbf{p}} = \mathbf{P}(\boldsymbol{\lambda}) \sum_{i=1}^{\mathsf{N}} \lambda_i \mathbf{P}_i^{-1} \hat{\mathbf{p}}_i = \mathbf{P}(\boldsymbol{\lambda})/\mathsf{N} \sum_{i=1}^{\mathsf{N}} (\mathsf{N}\lambda_i) \mathbf{P}_i^{-1} \hat{\mathbf{p}}_i \ (7)$$

Note that $\mathsf{N}\lambda_i$ and $\mathbf{P}(\boldsymbol{\lambda})$ can be estimated at agent i using $x_i(t)^2$ and $\hat{\mathbf{Q}}_i(t)^{-1}$ respectively for $t \geq T_\varepsilon$. Hence, the average in (7) for $\{\mathsf{N}\lambda_i\mathbf{P}_i^{-1}\hat{\mathbf{p}}_i\}_{i=1}^{\mathsf{N}}$, equivalently $\{x_i^2\mathbf{P}_i^{-1}\hat{\mathbf{p}}_i\}_{i=1}^{\mathsf{N}}$ can be computed by average consensus techniques [1], [2].

The sensor fusion rule in (7) has shown to be effective in many applications as described in [17], [24], where a sequential solution method is used in [8] to reduce the computational burden of using all sensors at once at the expense of reduced estimation quality. However, these solutions mostly rely on a centralized node which gathers the information across the network and computes (7). Our proposal is used to compute (7) as well, inheriting the same accuracy as [8], [17], [24] for this task. Nonetheless, we are able to obtain $\hat{\mathbf{p}}, \mathbf{P}(\boldsymbol{\lambda})$ in a distributed fashion. Other distributed approaches such as [13], [16], [29], [30] use local versions of the CIM rule in (7) where only neighboring sensors are used at each node. In contrast, we are able to include the information of all sensors across the network, which always increases the accuracy when compared to the case in which a subset of the sensors are used [17].

IV. CONVERGENCE ANALYSIS

In this section, we provide a proof for Theorem 1. Before that, we prove some auxiliary results. Systems (3) and $\dot{x}_i(t) = u_i(t)$ are discontinuous due to the use of the sign$(\bullet)$ function and the discontinuous nature of (5) at the boundary of $\mathcal{C}$. Hence, these systems are better understood in the sense of Filippov [10], devised to study discontinuous dynamics.

We start by analyzing the consensus protocols in (3). Note that all these protocols have the structure:

$$\dot{\upsilon}_i(t) = \kappa_z \sum_{j \in \mathcal{N}_i} \phi\left(\hat{z}_j(t) - \hat{z}_i(t); \zeta, q\right), \hat{z}_i(t) = z_i(t) - \upsilon_i(t) \ (8)$$

for suitable input $z_i(t)$, either $x_i(t)^2$ or the elements of $x_i(t)^2 \mathbf{P}_i^{-1}$. Now, we show convergence of (8) before $t = T_c$.

Proposition 1. *[2, Special case of Theorem 6] Let* $\mathcal{G}$ *be a connected undirected graph with* N *nodes,* ℓ *edges, algebraic connectivity* $\lambda_{\mathcal{G}}$, *and consider*

$$\dot{e}_i(t) = d_i(t) - \kappa_z \sum_{j \in \mathcal{N}_i} \phi(e_j(t) - e_i(t); \zeta, q) \quad (9)$$

where $|d_i(t)| \leq L', \forall t \geq 0$. *Moreover, given* $T_c > 0$ *let* $\kappa_z \geq \ell\pi/(q\lambda_{\mathcal{G}}T_c), \zeta \geq L'/(\kappa_z\sqrt{\lambda_{\mathcal{G}}})$. *Then,* $e_i(t) \equiv (1/\mathsf{N}) \sum_{i=1}^{\mathsf{N}} e_i(0), \forall t \geq T_c$.

Lemma 1. *Let* $\mathcal{G}$ *be a connected undirected graph with* N *nodes,* ℓ *edges, algebraic connectivity* $\lambda_{\mathcal{G}}$ *and* $\sum_{i=1}^{\mathsf{N}} \upsilon_i(0) = 0$. *Moreover, let* $\bar{z}(t) = (1/\mathsf{N}) \sum_{i=1}^{\mathsf{N}} z_i(t)$ *and* $|\dot{z}_i(t)| \leq L, \forall t \geq 0$. *Then,* (8) *satisfies that* $\hat{z}_i(t) \equiv \bar{z}(t), \forall t \geq T_c$ *and* $\forall i$, *provided that* $\kappa_z \geq \ell\pi/(q\lambda_{\mathcal{G}}T_c), \zeta \geq 2L/(\kappa_z\sqrt{\lambda_{\mathcal{G}}})$.

Proof. Let $e_i(t) = \upsilon_i(t) - (z_i(t) - \bar{z}(t))$ such that: $\dot{e}_i(t) = (\dot{\bar{z}}(t) - \dot{z}_i(t)) - \kappa_z \sum_{j \in \mathcal{N}_i} \phi(e_j(t) - e_i(t); \zeta, q)$ equivalent to (9) with $d_i(t) = \dot{\bar{z}}(t) - \dot{z}_i(t)$. By assumption, $|d_i(t)| \leq 2L$. Thus, Proposition 1 is used with $L' = 2L$ to conclude that $e_i(t) \equiv (1/\mathsf{N}) \sum_{i=1}^{\mathsf{N}} e_i(0) = 0, \forall t \geq T_c$. Hence, $\upsilon_i(t) = z_i(t) - \bar{z}(t)$ and $\hat{z}_i(t) = \bar{z}(t)$ for $t \geq T_c$. $\square$

Now, we study the ideal PGF trajectories for $\mathbf{x}(t)$, dictated by (5). The following result shows that $\mathbf{x}(t)$ converge to the feasible region $\mathcal{C}$, a requirement for a solution of (2).

Lemma 2. *Let* (5) *and* $\underline{b} > 0$. *Then,* $\forall \mathbf{x}(T_0) \notin \mathcal{C}, x_i(T_0) \geq \underline{b}$ *there exist* $T_\varepsilon > 0$ *such that* $\mathbf{x}(t) \in \mathcal{C}, \forall t \geq T_0 + T_\varepsilon$.

Proof. We split the proof in two cases: (a) $(1 - \varepsilon/2) - s(\mathbf{x}(T_0)) > 0$: in this case, note that $\dot{x}_i(t) = \kappa_{\mathcal{C}} x_i(t)$ for $t \in [T_0, T_0 + T')$ with $T' = \inf\{t > T_0 :$

$\mathbf{x}(t) \in \mathcal{C}\}$. Thus, $x_i(t)$ is increasing in such interval and $\|\mathbf{x}(t)\| \geq \sqrt{\mathsf{N} \min_{i \in \mathcal{I}} x_i(t)^2} \geq \sqrt{\mathsf{N}}\underline{b}, \quad \forall t \in [T_0, T_0 + T')$. Now, consider a Lyapunov function candidate $V_1(\mathbf{x}(t)) = (1 - \varepsilon) - s(\mathbf{x}(t))$, whose time derivative is $\dot{V}_1(\mathbf{x}(t)) = -(2/\mathsf{N})\mathbf{x}(t)^\top \dot{\mathbf{x}}(t) = -(2/\mathsf{N})\kappa_\mathcal{C}\|\mathbf{x}(t)\|^2 \leq -2\kappa_\mathcal{C}\underline{b}^2$. Thus, integrating from T_0 to t, $V_1(\mathbf{x}(t)) \leq V_1(\mathbf{x}(T_0)) - 2\kappa_\mathcal{C}\underline{b}^2(t - T_0)$ which satisfies $V_1(\mathbf{x}(T_0 + T_\varepsilon)) = 0$ with $T' \leq T_\varepsilon := V_1(\mathbf{x}(T_0))/(2\kappa_\mathcal{C}\underline{b}^2)$. Hence, $s(\mathbf{x}(T_0 + T_\varepsilon)) = 1 - \varepsilon$ and thus $\mathbf{x}(T_0 + T_\varepsilon) \in \mathcal{C}$. (b) $(1 - \varepsilon/2) - s(\mathbf{x}(T_0)) < 0$: note that $\dot{x}_i(t) = -\kappa_\mathcal{C}x_i(t)$ implies $x_i(t)$ is decreasing for $t \in [T_0, T_0 + T_\varepsilon)$ and thus, $\|\mathbf{x}(t)\| = \sqrt{\mathsf{N}s(\mathbf{x}(t))} \geq \sqrt{\mathsf{N}}$ since $s(\mathbf{x}(t)) \geq 1$. The rest of the argument for this case follows as before using $V_2(\mathbf{x}) = s(\mathbf{x}) - 1$. Thus, $\mathbf{x}(t)$ reaches $\mathcal{C}$ in finite time. In addition, $\mathbf{x}(t)$ remain inside $\mathcal{C}$ for $t \geq T_0 + T_\varepsilon$ since $V_1(\mathbf{x}(t))$ (resp. $V_2(\mathbf{x}(t))$) cannot increase at the inner (resp. outer) boundary of $\mathcal{C}$. $\qquad\square$

Now, we recall the definition of the projected gradient with respect to a manifold and then apply the definition to the problem of interest in (2).

Definition 1. *[12] Let $F : \mathbb{R}^n \to \mathbb{R}$ be differentiable and $\mathcal{C} = \{\mathbf{x} \in \mathbb{R}^n : \gamma_i(\mathbf{x}) \leq 0, i \in \{1, \ldots, m\}\}$ for m differentiable constraint functions $\gamma_i : \mathbb{R}^n \to \mathbb{R}$. Then, the projected gradient of $F(\mathbf{x})$ with respect to $\mathcal{C}$ is given by:*

$$\text{proj}_\mathcal{C}(\mathbf{x}, -\nabla F(\mathbf{x})) = \underset{\mathbf{w} \in \mathbb{R}^n}{\arg\min} \|\mathbf{w} - (-\nabla F(\mathbf{x}))\|^2 \tag{10}$$
$$\text{such that} \quad \mathbf{w}^\top \nabla \gamma_i(\mathbf{x}) \leq 0, \forall i \in \mathcal{J}(\mathbf{x})$$

where $\mathcal{J}(\mathbf{x}) = \{i \in \{1, \ldots, m\} : \gamma_i(\mathbf{x}) = 0\}$.

Lemma 3. *Let $\mathcal{C}$ with $\gamma_1(\mathbf{x}) = \|\mathbf{x}\|^2/\mathsf{N} - 1$, $\gamma_2(\mathbf{x}) = (1 - \varepsilon) - \|\mathbf{x}\|^2/\mathsf{N}$ in Definition 1 with $m = 2$. Then, for any differentiable $F(\bullet)$ and any $\mathbf{x} \in \mathcal{C}$,*

$$\text{proj}_\mathcal{C}(\mathbf{x}, -\nabla F(\mathbf{x})) =$$
$$\begin{cases} -\nabla F(\mathbf{x}) + \left(\frac{\mathbf{x}^\top \nabla F(\mathbf{x})}{\mathsf{N}^2}\right)\mathbf{x} & \gamma_1(\mathbf{x}) = 0, \mathbf{x}^\top \nabla F(\mathbf{x}) \leq 0 \\ -\nabla F(\mathbf{x}) - \left(\frac{\mathbf{x}^\top \nabla F(\mathbf{x})}{(\mathsf{N}(1-\varepsilon))^2}\right)\mathbf{x} & \gamma_2(\mathbf{x}) = 0, \mathbf{x}^\top \nabla F(\mathbf{x}) \geq 0 \\ -\nabla F(\mathbf{x}) & \text{otherwise} \end{cases} \tag{11}$$

Proof. First, $\nabla\gamma_1(\mathbf{x}) = 2\mathbf{x}/\mathsf{N}, \nabla\gamma_2(\mathbf{x}) = -2\mathbf{x}/\mathsf{N}$. Consider the following cases: (a) $\gamma_1(\mathbf{x}) < 0, \gamma_2(\mathbf{x}) < 0$: since (10) is unconstrained, $\mathbf{w} = -\nabla F(\mathbf{x})$. (b) for $\gamma_1(\mathbf{x}) = 0, \mathbf{x}^\top \nabla F(\mathbf{x}) \geq 0$ or $\gamma_2(\mathbf{x}) = 0, \mathbf{x}^\top \nabla F(\mathbf{x}) \leq 0$: $\mathbf{w} = -\nabla F(\mathbf{x})$ satisfies the constraint $\mathbf{w}^\top \nabla\gamma_1(\mathbf{x}) \leq 0$ and $\mathbf{w}^\top \nabla\gamma_2(\mathbf{x}) \leq 0$ in (10) respectively. (c) $\gamma_1(\mathbf{x}) = 0, \mathbf{x}^\top \nabla F(\mathbf{x}) \leq 0$: let $h(\mathbf{w}) = \|\mathbf{w} + \nabla F(\mathbf{x})\|^2$ and $\mathbf{w}^* = -\nabla F(\mathbf{x}) + \left(\frac{\mathbf{x}^\top \nabla F(\mathbf{x})}{\mathsf{N}^2}\right)\mathbf{x}$. Note that $(\mathbf{w}^*)^\top \nabla\gamma_1(\mathbf{x}) = 0$ making $\mathbf{w}^*$ is feasible for (10) and $\nabla_\mathbf{w}h(\mathbf{w}) = 2(\nabla F(\mathbf{x}) + \mathbf{w}) = \lambda\mathbf{x}/\mathsf{N} = (\lambda/2)\nabla_\mathbf{w}(\mathbf{w}^\top \nabla\gamma_1(\mathbf{x}))$ is satisfied with $\mathbf{w} = \mathbf{w}^*$ and Lagrange multiplier $\lambda = 2\mathbf{x}^\top \nabla F(\mathbf{x})/\mathsf{N}$ where $\nabla_\mathbf{w}$ denotes gradient with respect to $\mathbf{w}$. Hence, $\mathbf{w} = \mathbf{w}^*$ satisfies the optimality conditions for (10) and is the unique minimizer [3, Proposition 3.1.1]. (d) $\gamma_2(\mathbf{x}) = 0, \mathbf{x}^\top \nabla F(\mathbf{x}) \geq 0$: the same reasoning of (c) applies. $\qquad\square$

In the following, we show that the ideal system (5) flows in the direction of the previously obtained projected gradient.

Lemma 4. *Let* (5) *with $\mathbf{x}(T_0) \in \mathcal{C}$. Then, under Definition 1, $\dot{\mathbf{x}}(t) = \alpha(\mathbf{x}(t)) \text{proj}_\mathcal{C}(\mathbf{x}(t), -\nabla f(\mathbf{Q}(\mathbf{x}(t))))$ for some positive scalar function $\alpha : \mathbb{R}^n \to \mathbb{R}_{>0}$.*

Proof. In this proof, we omit time dependence for conciseness. We compare the right hand side of (5) with (11) in Lemma 3 under $F(\mathbf{x}) = f(\mathbf{Q}(\mathbf{x}))$: (a) $\gamma_1(\mathbf{x}) < 0, \gamma_2(\mathbf{x}) < 0$: $\dot{\mathbf{x}} = -\alpha(\mathbf{x})\nabla F(\mathbf{x}) = \alpha(\mathbf{x}) \text{proj}_\mathcal{C}(\mathbf{x}, -\nabla F(\mathbf{x}))$ due to (11) with $\alpha(\mathbf{x}) = \kappa_\mathcal{C}\mathsf{N}$. (b) $\gamma_1(\mathbf{x}) = 0$: we invoke the Filippov interpretation of solutions to write $\dot{\mathbf{x}}$. Call $\mathbf{w}_1 = \kappa_\mathcal{C}\mathbf{x}\text{sign}((1 - \frac{\varepsilon}{2}) - s(\mathbf{x}))$ and $\mathbf{w}_2 = -\kappa_\mathcal{C}\mathsf{N}\nabla f(\mathbf{Q}(\mathbf{x}))$. Filippov solutions in this manifold have $\dot{\mathbf{x}}$ in the convex hull of $\{\mathbf{w}_1, \mathbf{w}_2\}$. Concretely, if $\gamma_1(\mathbf{x}) = 0, \mathbf{x}^\top \nabla F(\mathbf{x}) \geq 0$, then $\mathbf{w}_1, \mathbf{w}_2$ point inside $\mathcal{C}$, and $s(\mathbf{x}(t)) = 0$ is satisfied for an isolated instant t, hence ignored by the trajectory [7]. In the case of $\gamma_1(\mathbf{x}) = 0, \mathbf{x}^\top \nabla F(\mathbf{x}) \leq 0$, $\mathbf{w}_1, \mathbf{w}_2$ point in different directions with respect to the inner boundary of $\mathcal{C}$, inducing sliding motion along $\gamma_1(\mathbf{x}) = 0$. Therefore, $\dot{\mathbf{x}}$ is the convex combination of $\mathbf{w}_1, \mathbf{w}_2$, lying in the tangent plane of $\gamma_1(\mathbf{x}) = 0$ [7]. Note that $\mathbf{w} = -\nabla F(\mathbf{x}) + \left(\frac{\mathbf{x}^\top \nabla F(\mathbf{x})}{\mathsf{N}^2}\right)\mathbf{x} = \left(\frac{\mathbf{x}^\top \nabla F(\mathbf{x})}{\kappa_\mathcal{C}\mathsf{N}^2\text{sign}((1-\frac{\varepsilon}{2})-s(\mathbf{x}))}\right)\mathbf{w}_1 + \left(\frac{1}{\kappa_\mathcal{C}\mathsf{N}}\right)\mathbf{w}_2$ with $\mathbf{w}^\top \mathbf{x} = 0$, lying in such tangent plane. Hence, $\dot{\mathbf{x}}$ is proportional to $\mathbf{w}$ and $\dot{\mathbf{x}} = \alpha(\mathbf{x})\mathbf{w} = \alpha(\mathbf{x}) \text{proj}_\mathcal{C}(\mathbf{x}, -\nabla F(\mathbf{x}))$ for some $\alpha(\mathbf{x}) > 0$. (c) $\gamma_2(\mathbf{x}) = 0$: the proof follows from (b). $\qquad\square$

Lemma 5. *Consider* (5) *under Assumption 1 and $f(\bullet)$ in Table I labeled by $\mu \in \{0, 1, 2\}$. For any $\mathbf{x}(T_0) \notin \mathcal{C}$ it follows that $\|\dot{\mathbf{x}}(t)\| \leq h(\mathsf{N}), \forall t \geq T_0$ with $h(\bullet)$ as in Theorem 1.*

First, we show that $|x_i(t)| \leq \bar{b}, \forall t \geq T_0$ by contradiction. Assume there exists $T_2 \geq T_0$ such that $|x_i(T_2)| > \bar{b}$. By continuity of the solution, there must have existed $T_1 = \sup\{t \in [T_0, T_2] : |x_i(t)| = \bar{b}\}$ such that $a_i(t) = |x_i(t)|$ is increasing for $t \in [T_1, T_2]$. However, in this interval, $\mathbf{x}(t) \notin \mathcal{C}$ since $s(\mathbf{x}(t)) = \|\mathbf{x}(t)\|^2/\mathsf{N} \geq \min_{i \in \mathcal{I}} |x_i(t)|^2/\mathsf{N} \geq \bar{b}^2 > 1$. Therefore, due to (5), $\dot{a}_i(t) = -\kappa_\mathcal{C}a_i(t)$ which means that $a_i(t)$ is decreasing in $t \in [T_1, T_2]$ leading to a contradiction.

Now, we show $\text{tr}(\mathbf{Q}(\mathbf{x}(t))^{-1}) \leq \mathsf{N}(\underline{\sigma}\min\{\underline{b}^2, 1 - \varepsilon\})^{-1}, \forall t \geq T_0$. From the proof of Lemma 2 it follows that $s(\mathbf{x}(t)) = \|x_i(t)\|^2/\mathsf{N} \geq \underline{b}^2, \forall t \in [T_0, T_0 + T_\varepsilon]$ and $s(\mathbf{x}(t)) \geq 1 - \varepsilon, \forall t \geq T_0 + T_\varepsilon$. Henceforth, $\mathbf{Q}(\mathbf{x}(t)) \succeq (1/\mathsf{N})\sum_{i=1}^{\mathsf{N}} x_i(t)^2\underline{\sigma}\mathbf{I} \succeq \underline{\sigma}\min\{\underline{b}^2, 1 - \varepsilon\}\mathbf{I}$ for $t \geq T_0$. Then, $\mathbf{Q}(\mathbf{x}(t))^{-1} \preceq (\underline{\sigma}\min\{\underline{b}^2, 1 - \varepsilon\})^{-1}\mathbf{I}$ from which $\text{tr}(\mathbf{Q}(\mathbf{x}(t))^{-1}) \leq \mathsf{N}(\underline{\sigma}\min\{\underline{b}^2, 1 - \varepsilon\})^{-1}$ is obtained. From the previous bounds, and the gradient in Table I:

$$\|\nabla f(\mathbf{Q}(\mathbf{x}(t)))\| \leq \max_{i \in \mathcal{I}} 2|x_i|\text{tr}(\mathbf{P}_i^{-1})\text{tr}(\mathbf{Q}(\mathbf{x}(t))^{-1})^\mu$$
$$\leq 2\bar{b}\bar{\sigma}\mathsf{N}^\mu(\underline{\sigma}\min\{\underline{b}^2, 1 - \varepsilon\})^{-\mu},$$

and

$$\|\dot{\mathbf{x}}(t)\| \leq \kappa_\mathcal{C}\max\{\|\mathbf{x}(t)\|, \mathsf{N}\|\nabla f(\mathbf{Q}(\mathbf{x}(t)))\|\}$$
$$\leq \kappa_\mathcal{C}\max\{\sqrt{\mathsf{N}}\bar{b}, 2\bar{b}\bar{\sigma}\mathsf{N}^{\mu+1}(\underline{\sigma}\min\{\underline{b}^2, 1 - \varepsilon\})^{-\mu}\} \quad \square$$

Recalling from Lemma 2 that (5) flows in the direction of the projected gradient, the following result characterizes convergence of such flow towards the optimum of (2).

Proposition 2. *Let* $F(\bullet), \gamma_i(\bullet), \mathcal{C}$ *defined as in Definition 1 and* $\alpha : \mathbb{R}^n \to \mathbb{R}_{>0}$ *be a positive scalar function such that* $\dot{\mathbf{x}}(t) = \alpha(\mathbf{x}(t)) \, \mathrm{proj}_{\mathcal{C}}(\mathbf{x}(t), -\nabla f(\mathbf{x}(t)))$ *has unique forward solution for* $t \geq T_0$ *and any* $\mathbf{x}(T_0) \in \mathcal{C}$. *Then,* $\mathbf{x}(t)$ *converges asymptotically towards* $\arg\min_{\mathbf{x} \in \mathcal{C}} F(\mathbf{x})$.

Proof. The result follows using the same reasoning in [12, Theorem 3], where a proof was provided for $\alpha(\mathbf{x}) = 1$. $\square$

Proof of Theorem 1: We start by analyzing (3) for $t \in [0, T_c]$, in which $x_i(t)^2 \mathbf{P}_i^{-1}$ and $x_i(t)^2$ remain constant because $u_i(t) = 0$. The result from Lemma 1 is valid obtaining $\hat{s}_i(t) \equiv s(\mathbf{x}(t)), \hat{\mathbf{Q}}_i(t) \equiv \mathbf{Q}(\mathbf{x}(t))$ at $t = T_c$. Assume that the condition $\hat{s}_i(t) \equiv s(\mathbf{x}(t))$ is maintained for some open interval $\mathcal{T} \subset [T_c, \infty)$. Then, $\mathbf{x}(t)$ is dictated by (5) for any $t \in \mathcal{T}$ and thus $z_i(t) = x_i(t)^2$ comply $|\dot{z}_i(t)| \leq 2|x_i(t)| \|\dot{\mathbf{x}}(t)\| \leq 2\bar{b}h(\mathsf{N})$ due to Lemma 5. Hence, the conditions of Lemma 1 are fulfilled with $L = 2\bar{b}h(\mathsf{N})$ due to the choice of κ_s in Theorem 1 for such interval $\mathcal{T}$.

Now, we show that $\hat{s}_i(t) \equiv s(\mathbf{x}(t))$ is maintained $\forall t \in [T_c, \infty)$. We verify this by contradiction. Assume that there exists $T_2 = \inf\{t \geq T_c : \|\dot{\mathbf{x}}(t)\| > 2\bar{b}h(\mathsf{N})\}$. Then, there must have existed some $T_1 = \sup\{t < T_2 : \hat{s}_i(t) \neq s(\mathbf{x}(t))$ or $\hat{\mathbf{Q}}_i(t) \neq \mathbf{Q}(\mathbf{x}(t))\}$ since the system could not have been the ideal one in (5). Hence, the system is not synchronized for some time $t \in [T_1, T_2]$ for which $\|\dot{\mathbf{x}}(t)\| \leq 2\bar{b}h(\mathsf{N})$, which is impossible since this means that the inputs for the consensus protocols have bounded derivative, and Lemma 1 ensure synchronization. Hence, $\hat{s}_i(t) \equiv s(\mathbf{x}(t)), \forall t \geq T_c$. The same reasoning applies to the synchronization condition $\hat{\mathbf{Q}}_i(t) \equiv \mathbf{Q}(\mathbf{x}(t))$.

Now, use Lemma 2 with $T_0 = T_c$ to obtain $\mathbf{x}(t) \in \mathcal{C}, \forall t \geq T_c + T_\varepsilon$. We can verify $\mathcal{E}(\mathbf{Q}(\mathbf{x}(t))^{-1}) \supset \check{\mathcal{E}}, \forall t \geq T_c + T_\varepsilon$ by taking any $\mathbf{y} \in \check{\mathcal{E}}$ which satisfy $\mathbf{y}^\top \mathbf{Q}(\mathbf{x}(t))\mathbf{y} = \sum_{i=1}^{\mathsf{N}}(x_i(t)^2/\mathsf{N})\mathbf{y}^\top \mathbf{P}_i^{-1}\mathbf{y} \leq \sum_{i=1}^{\mathsf{N}}(x_i(t)^2/\mathsf{N}) = s(\mathbf{x}(t)) \leq 1$. Hence, $\mathbf{y} \in \mathcal{E}(\mathbf{Q}(\mathbf{x}(t))^{-1})$. Use Lemma 4 with $T_0 = T_c + T_\varepsilon$ to write (5) as $\dot{\mathbf{x}}(t) = \alpha(\mathbf{x}(t)) \, \mathrm{proj}_{\mathcal{C}}(\mathbf{x}(t), -\nabla f(\mathbf{Q}(\mathbf{x}(t))))$, implying convergence towards the optimal weights due to Proposition 2. $\square$

V. NUMERICAL EXPERIMENTS

In this section, we simulate the proposal to illustrate its properties in the context of a sensor fusion problem. We assume that each agent reads a noisy measurement from a sensor with covariance matrix $\{\mathbf{P}_i\}_{i=1}^{\mathsf{N}}$. For the sake of generality, we set $\mathbf{P}_i = \mathbf{M}_i^\top \mathbf{M}_i$ where $\mathbf{M}_i$ was drawn with uniformly distributed components. The purpose of each agent i is to compute $\mathsf{N}\lambda_i, \mathbf{P}(\boldsymbol{\lambda})$ from $x_i(t)^2, \hat{\mathbf{Q}}_i(t)^{-1}$ so that they can be used in the sensor fusion rule in (7). We choose $f(\mathbf{Q}(\mathbf{x})) = \mathrm{tr}(\mathbf{Q}(\mathbf{x})^{-1})$ as a performance index.

All the differential equations, namely, (3) and $\dot{x}_i(t) = u_i(t)$ under (4), where simulated using the forward Euler method with time step $\Delta t = 10^{-4}$. In the sake of interpretability, we set $n = 2$ and a graph $\mathcal{G}$ of $\mathsf{N} = 6$ agents with edges $\mathcal{F} = \{(1,4), (1,5), (2,4), (2,6), (3,6), (5,6)\}$. In addition, we set $\varepsilon = 0.05$ as well as parameters $T_c = 1, q = 1/2, \underline{b} = 0.1, \bar{b} = 1.1, \kappa_{\mathcal{C}} = 0.1, \kappa_s = \kappa_{\mathbf{Q}} = 10, \zeta_{\mathbf{Q}} = \zeta_s = 1, \zeta_g = 4$. Figure 1 shows the local estimates
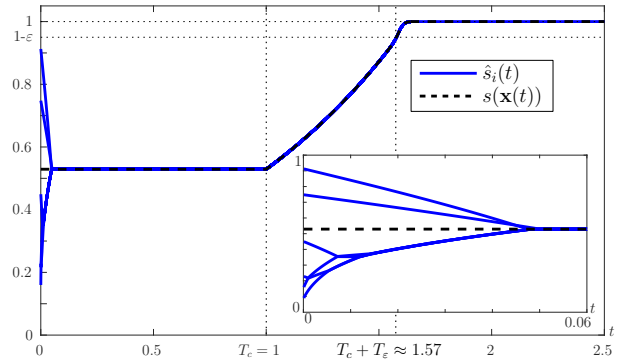

Fig. 1: Convergence of the estimates $\hat{s}_i(t)$ which reach consensus towards $s(\mathbf{x}(t))$ at $t \approx 0.05$, with zoom in $t \in [0, 0.06]$. Moreover, the evolution of the surface $s(\mathbf{x}(t))$ towards achieving $\mathbf{x}(t) \in \mathcal{C}$ is shown.
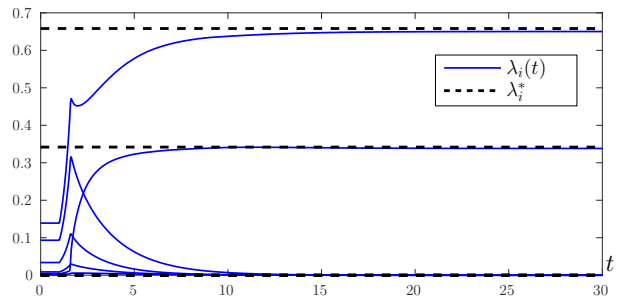

Fig. 2: Evolution of the trajectories $\lambda_i(t) = x_i(t)^2/\mathsf{N}$ which converge towards the global optimal values of (1) asymptotically, up to error less than $\varepsilon = 0.05$.

$\hat{s}_i(t)$ which reach consensus towards the global value of $s(\mathbf{x}(t))$ at $t \approx 0.05$, occurring before $t = T_c = 1$, which is a conservative prescribed convergence time bound. After $t = T_c$, $s(\mathbf{x}(t))$ increases until it reaches $s(\mathbf{x}(t)) = 1 - \varepsilon$, which means that $\mathbf{x}(t) \in \mathcal{C}$ from $t \geq T_c + T_\varepsilon$ with $T_\varepsilon \approx 0.57$. Figure 2 shows that, once $\mathbf{x}(t) \in \mathcal{C}$, then, the trajectories of $\lambda_i(t) = x_i(t)^2/\mathsf{N}$ converge towards the global minimizer of (2) which corresponds to $\lambda_1^* = \lambda_2^* = \lambda_3^* = \lambda_4^* = 0, \lambda_5^* = 0.6582, \lambda_6^* = 0.3418$ in this example, up to some error less than $\varepsilon = 0.05$. Finally, Figure 3 show the ellipses $\{\mathcal{E}(\mathbf{P}_i)\}_{i=1}^{\mathsf{N}}$ as well as $\mathcal{E}(\mathbf{Q}(\mathbf{x}(t))^{-1})$ with $t = T_c + T_\varepsilon$ and $t = 30$, which shows that $\mathcal{E}(\mathbf{Q}(\mathbf{x}(t))^{-1}) \supset \check{\mathcal{E}}$ in both cases, but the latter case leads to a tighter outer L-J ellipse.

Moreover, we evaluate the time it takes for the agents to reach consensus T_{cons} and T_ε by repeating the same experiment as before with a circular graph $\mathcal{G}$ of $\mathsf{N} = 10$ and 20 agents and with $n = 2$ and 4. We set $\kappa_s, \kappa_{\mathbf{Q}}, \zeta_s, \kappa_{\mathbf{Q}}$ the same in all cases, adjusted accordingly to account for $\ell, \lambda_{\mathcal{G}}$ in both networks as required in Theorem 1. The results are summarized in Table II where it is observed that T_{cons} increases with N as expected from [2]. On the other hand, T_{cons} is similar as before when n is increased. The reason is that (3) is executed in parallel for each component of $\hat{\mathbf{Q}}_i(t)$ such that there is no influence of n in the protocol convergence itself. Note that T_ε is similar in all cases since
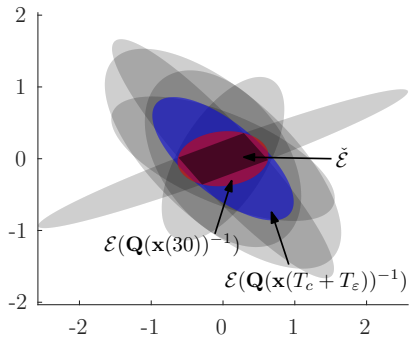
Fig. 3: Ellipses $\{\mathcal{E}(\mathbf{P}_i)\}_{i=1}^{\mathsf{N}}$ (shown in grey) as well as $\mathcal{E}(\mathbf{Q}(\mathbf{x}(t))^{-1})$ with $t = T_c + T_\varepsilon$ (blue) and $t = 30$ (red).

once consensus has been reached, the system is equivalent to (5) which does not depend on N, n for $\mathbf{x}(t) \notin \mathcal{C}$.

N	ℓ	$\lambda_{\mathcal{G}}$	n	T_{cons} [s]	T_ε [s]
10	9	0.3819	2	0.0371	0.169
			4	0.0368	0.172
20	19	0.0978	2	0.085	0.170
			4	0.088	0.181

TABLE II: Values for the consensus and feasible region reaching times $T_{\mathsf{cons}}, T_\varepsilon$, for a circular graph with N nodes.

VI. Conclusions

This work has developed a distributed method to compute a class of outer L-J ellipsoids. This is particularly useful for sensor fusion applications, as it allows the computation of the covariance intersection ellipsoid by combining information from the entire network using only local interactions. The algorithm reformulates the centralized problem and utilizes EDC tools to reach consensus and converge to an outer L-J ellipsoid in finite time, and to the global optimum asymptotically. The proposal's advantages are supported by formal convergence analysis and numerical experiments.

References

[1] Rodrigo Aldana-López, Rosario Aragüés, and Carlos Sagüés. ED-CHO: High order exact dynamic consensus. *Automatica*, 131:109750, 2021.

[2] Rodrigo Aldana-López, David Gómez-Gutiérrez, Michael Defoort, Juan Diego Sánchez-Torres, and Aldo Jonathan Muñoz-Vázquez. A class of robust consensus algorithms with predefined-time convergence under switching topologies. *Int. J. Robust Nonlinear Control*, 29(17):6179–6198, 2019.

[3] D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1999.

[4] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, 2011.

[5] Andrea Camisa, Ivano Notarnicola, and Giuseppe Notarstefano. Distributed stochastic dual subgradient for constraint-coupled optimization. *IEEE Control Syst. Lett.*, 6:644–649, 2022.

[6] Hao Chen, Jianan Wang, Chunyan Wang, Jiayuan Shan, and Ming Xin. Distributed diffusion unscented kalman filtering based on covariance intersection with intermittent measurements. *Automatica*, 132:109769, 2021.

[7] Jorge Cortes. Discontinuous dynamical systems. *IEEE Control Syst. Mag.*, 28(3):36–73, 2008.

[8] Zili Deng, Peng Zhang, Wenjuan Qi, Jinfang Liu, and Yuan Gao. Sequential covariance intersection fusion Kalman filter. *Inf. Sci.*, 189:293–309, 2012.

[9] Amir-Salar Esteki and Solmaz S Kia. Distributed unconstrained optimization with time-varying cost functions. *arXiv:2212.09472*, 2022.

[10] A. F. Filippov. *Differential equations with discontinuous righthand sides*, volume 18 of *Mathematics and its Applications (Soviet Series)*. Kluwer Academic Publishers Group, Dordrecht, 1988.

[11] Jemin George and Randy A. Freeman. Robust dynamic average consensus algorithms. *IEEE Trans. Autom. Control*, 64(11):4615–4622, 2019.

[12] Adrian Hauswirth, Saverio Bolognani, Gabriela Hug, and Florian Dörfler. Projected gradient descent on riemannian manifolds with applications to online power system optimization. In *Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 225–232, 2016.

[13] Xingkang He, Wenchao Xue, and Haitao Fang. Consistent distributed state estimation with global observability over sensor network. *Automatica*, 92:162–172, 2018.

[14] Didier Henrion and Jean-Bernard Lasserre. Inner approximations for polynomial matrix inequalities and robust stability regions. *IEEE Trans. Autom. Control*, 57(6):1456–1467, 2011.

[15] Didier Henrion, Sophie Tarbouriech, and Denis Arzelier. LMI approximations for the radius of the intersection of ellipsoids: Survey. *J. Optim. Theory Appl.*, 108(1):1–28, 2001.

[16] Jinwen Hu, Lihua Xie, and Cishen Zhang. Diffusion Kalman filtering based on covariance intersection. *IEEE Trans. Signal Process.*, 60(2):891–902, 2011.

[17] Simon Julier and Jeffrey K Uhlmann. General decentralized data fusion with covariance intersection. In *Handbook of multisensor data fusion*, pages 339–364. CRC Press, 2017.

[18] Jean B Lasserre. A generalization of Löwner-John's ellipsoid theorem. *Math. Program.*, 152:559–591, 2015.

[19] Xinyi Le, Sijie Chen, Zheng Yan, and Juntong Xi. A neurodynamic approach to distributed optimization with globally coupled constraints. *IEEE Transactions on Cybernetics*, 48(11):3149–3158, 2017.

[20] Haihao Lu, Robert M Freund, and Yurii Nesterov. Relatively smooth convex optimization by first-order methods, and applications. *SIAM J. Optim.*, 28(1):333–354, 2018.

[21] Chenxin Ma, Jakub Konečnỳ, Martin Jaggi, Virginia Smith, Michael I Jordan, Peter Richtárik, and Martin Takáč. Distributed optimization with arbitrary local solvers. *optimization Methods and Software*, 32(4):813–848, 2017.

[22] Angelia Nedić and Ji Liu. Distributed optimization for control. *Annu. Rev. Control Rob. Auton. Syst.*, 1:77–103, 2018.

[23] Angelia Nedić and Alex Olshevsky. Distributed optimization over time-varying directed graphs. *IEEE Trans. Autom. Control*, 60(3):601–615, 2014.

[24] W. Niehsen. Information fusion based on fast covariance intersection filtering. In *Proceedings of the Fifth International Conference on Information Fusion. FUSION 2002.*, volume 2, pages 901–904 vol.2, 2002.

[25] Boda Ning, Qing-Long Han, and Zongyu Zuo. Distributed optimization for multiagent systems: An edge-based fixed-time consensus approach. *IEEE Transactions on Cybernetics*, 49(1):122–132, 2017.

[26] Benjamin Noack, Joris Sijs, Marc Reinhardt, and Uwe D Hanebeck. Decentralized data fusion with inverse covariance intersection. *Automatica*, 79:35–41, 2017.

[27] Ivano Notarnicola and Giuseppe Notarstefano. Constraint-coupled distributed optimization: a relaxation and duality approach. *IEEE Transactions on Control of Network Systems*, 7(1):483–492, 2019.

[28] Licio Romao, Kostas Margellos, Giuseppe Notarstefano, and Antonis Papachristodoulou. Subgradient averaging for multi-agent optimisation with different constraint sets. *Automatica*, 131:109738, 2021.

[29] Eduardo Sebastián, Eduardo Montijano, and Carlos Sagüés. All-in-one: Certifiable optimal distributed kalman filter under unknown correlations. In *IEEE CDC*, pages 6578–6583, 2021.

[30] Shaocheng Wang and Wei Ren. On the convergence conditions of distributed dynamic state estimation using sensor networks: A unified framework. *IEEE Trans. Control Syst. Technol.*, 26(4):1300–1316, 2017.