

Consensus with Robustness to Outliers via Distributed Optimization

Jixin Li

Ehsan Elhamifar

I-Jeng Wang

René Vidal

Abstract—Over the past few years, a number of distributed algorithms have been developed for integrating the measurements acquired by a wireless sensor network. Among them, average consensus algorithms have drawn significant attention due to a number of practical advantages, such as robustness to noise in the measurements, robustness to changes in the network topology and guaranteed convergence to the centralized solution. However, one of the main drawbacks of existing consensus algorithms is their inability to handle outliers in the measurements. This is because they are based on minimizing a Euclidean (L_2) loss function, which is known to be sensitive to outliers. In this paper, we propose a distributed optimization framework that can handle outliers in the measurements. The proposed framework generalizes consensus algorithms to robust loss functions that are strictly convex or convex, such as the Huber loss or the L_1 -loss. This generalization is achieved by posing the robust consensus problem as a constrained optimization problem, which is solved using distributed versions of classical primal-dual and augmented Lagrangian optimization methods. The resulting algorithms include the classical average consensus as a particular case. Synthetic experiments evaluate our robust consensus framework for several robust cost functions and show their advantages over the classical average consensus algorithm.

I. INTRODUCTION

We have reached a technology inflection point where the availability of miniature, network-enabled sensors provides a tremendous opportunity for using large-scale wireless sensor networks to monitor an environment and carry out automatic analysis. Such networks can include a variety of sensors (heat, magnetic field, cameras) and have found widespread applications in several domains, including environmental monitoring, health monitoring, multi-robot navigation, etc.

These applications have motivated the development of a number of distributed algorithms for integrating the acquired information across the network [1], [2]. Among them, *average consensus algorithms* [3] have gained significant attention [4]. In the classical *average consensus* algorithm, each node measures a scalar quantity, say temperature, and the average temperature over the entire network is obtained by iteratively updating the temperature reading at each node with the average temperature of its neighbors. Under mild network connectivity requirements, this iterative procedure is provably convergent to the global average. Moreover, this procedure can be easily extended to multivariate data $\mathbf{u}_i \in \mathbb{R}^D$ by applying the scalar algorithm to each coordinate of \mathbf{u}_i .

This work was supported by the grant NSF CNS 0834470.

J. Li, E. Elhamifar and R. Vidal are with the Center for Imaging Science, Johns Hopkins University, Baltimore MD 21218, USA, {jixin,ehsan,rvidal}@cis.jhu.edu.

I.-J. Wang is with the Johns Hopkins University Applied Physics Laboratory, Laurel, MD 20723, USA I-Jeng.Wang@jhuapl.edu

A number of works have extended the applicability of consensus algorithms to more general cost functions. [5] proposes distributed consensus algorithms to solve a least-squares estimation problem. [6] extends consensus algorithms to computing the harmonic mean, the geometric mean, the minimum and the maximum under the assumptions that the graph is balanced, the gradient of the cost function is distributed and the entries of the gradient are of the same sign. [7] extends consensus to Riemannian manifolds by using a chord distance to compute a projected arithmetic mean. [8] extends consensus to the space of rigid-body motions by using the geodesic distance to compute a Karcher mean. [9] uses an approach similar to ours, but for a completely different application area (classification and clustering).

Significant research has also been undertaken to enhance the robustness of the standard consensus algorithms to measurement noise [10], [11], [5], [12]. However, most of the existing results on the robustness of consensus to errors in the measurements assume independent random noise with stationary statistical properties and do not address the issue of outliers that results from unmodeled perturbations.

Among the few existing works addressing the issue of outliers are [13] and [14], which use a primal-dual formulation to extend consensus to strictly convex functions and to a class of maximum likelihood problems, respectively. However, the derivation in [13] is not correct, as we will point out in Section III. The work of [15] also addresses robust loss functions, but it is not based on consensus and requires the nodes to form a communication cycle.

The augmented Lagrangian method presented in this paper also represents an alternative to the recent results in [16] and [17] for distributed convex optimization based on subgradient methods. Our approach leads to a more straightforward convergence analysis with no increase in communications and no projection and averaging involved in the iterations. The dual decomposition method used in [18] is applicable to more generalized functions, but it is not completely decentralized and requires redundant variables when applied to the consensus problem that we address here.

Paper contributions. We propose a distributed optimization framework for dealing with outliers in consensus algorithms. More specifically, we pose the robust consensus problem as a constrained optimization problem in which we minimize a strictly convex or a convex loss function. In the case of a strictly convex loss, the Lagrangian function is naturally distributed, hence the robust consensus problem can be solved using a distributed primal-dual approach. We show that this approach reduces to a local optimization at each node plus

a consensus-like update of the Lagrange multipliers. In the case of a convex loss, we use an augmented Lagrangian approach that adds the square of the constraints to the loss function in order to make the problem strictly convex. However, the resulting augmented Lagrangian function is not naturally distributed. By introducing auxiliary variables and using a modified augmented Lagrangian approach, we still find a distributed solution, which involves solving a modified local optimization problem with an additional quadratic form plus the same consensus-like update of the Lagrange multipliers. We also show how to reduce the computational complexity of our approach by simply solving a local optimization problem plus a consensus-like update in an auxiliary state.

Paper outline. The paper is organized as follows. In Section II we briefly review the classical average consensus algorithm together with some basic results from duality theory. In Section III, we propose a generalized distributed consensus algorithm for strictly convex and convex loss functions. In Section IV we apply the proposed framework to the robust consensus problem using several robust loss functions, such as the Huber loss and the L_1 -loss.

II. PRELIMINARIES

In this section, we review the classical average consensus algorithm and the basics of duality theory, which will be used for deriving the results to be presented in Section III.

A. Classical Average Consensus

In the basic setup of average consensus, the sensor network is represented using a directed graph $G = (V, E)$, where the vertices $i \in V = \{1, \dots, n\}$ represent the nodes of the network, and the edges $(i, j) \in E \subseteq V \times V$ represent the communication links such that if $(i, j) \in E$ then there is a link from node j to i . The graph is assumed to be symmetric, i.e., if an edge (i, j) belongs to E , then (j, i) also belongs to E . The set of neighbors of node i is denoted as $\mathcal{N}_i = \{j \in V \mid (i, j) \in E\}$. The number of neighbors or degree of node i is denoted as $d_i = |\mathcal{N}_i|$ and the maximum degree of the graph G is denoted as $\Delta_G = \max_i \{|\mathcal{N}_i|\}$.

Each node measures a scalar quantity $u_i \in \mathbb{R}$, $i \in V$ and the goal is to compute the average of these measurements, $\bar{u} = \frac{1}{n} \sum_{i=1}^n u_i$, in a distributed fashion. Classical average consensus does so by iterating the difference equation

$$z_i(t+1) = z_i(t) + \epsilon \sum_{j \in \mathcal{N}_i} (z_j(t) - z_i(t)), \quad z_i(0) = u_i, \quad i \in V, \quad (1)$$

where $z_i(t)$ is the state of node i at iteration t and $\epsilon \leq \frac{1}{\Delta_G}$ is the step-size. One can verify that the mean of the states is preserved at each iteration, i.e.,

$$\bar{u} = \frac{1}{n} \sum_{i=1}^n z_i(t) = \frac{1}{n} \sum_{i=1}^n z_i(t+1). \quad (2)$$

One can also verify that (1) is in fact a gradient descent algorithm for minimizing the function

$$\varphi(z_1, z_2, \dots, z_n) = \sum_{i=1}^n \sum_{j \in \mathcal{N}_i} (z_i - z_j)^2. \quad (3)$$

The minimum of (3) is achieved when the nodes reach a *consensus* at \bar{u} , i.e., when $z_i = \bar{u}$, $i \in V$. It can be shown that $\lim_{t \rightarrow \infty} z_i(t) = \bar{u}$, $i \in V$, when the graph G is connected (see e.g., [19]). The rate of convergence is given by the second smallest eigenvalue of the graph's Laplacian matrix.¹

B. Review of Basic Duality Theory

In this section, we review some basic results from duality theory. We refer the readers to [20] and [21] for details.

Consider the (primal) optimization problem

$$\begin{aligned} \min_z \quad & f(z) \\ \text{s.t.} \quad & g_i(z) = 0, \quad i \in \{1, \dots, l\}, \\ & z \in S, \end{aligned} \quad (4)$$

where $f : \mathbb{R}^D \rightarrow \mathbb{R}$ is the objective function, $g_i : \mathbb{R}^D \rightarrow \mathbb{R}$ are l equality constraints, and $S \subseteq \mathbb{R}^D$ is a nonempty convex set. The Lagrange function for this problem is defined as

$$\mathcal{L}(z, \lambda) = f(z) + \sum_{i=1}^l \lambda_i g_i(z) = f(z) + \lambda^\top g(z), \quad (5)$$

where $\lambda = (\lambda_1, \dots, \lambda_l)^\top$ is the vector of Lagrange multipliers and $g(z) = (g_1(z), \dots, g_l(z))^\top$ is the vector of constraints. The Lagrange dual function is then defined as

$$q(\lambda) = \inf_{z \in S} \mathcal{L}(z, \lambda), \quad (6)$$

where the infimum is taken over all $z \in S \subseteq \mathbb{R}^D$.

The dual function $q(\lambda)$ provides a lower bound on the optimal value of the primal problem $f(z^*)$, i.e., $q(\lambda) \leq f(z^*) \leq f(z)$. This result, known as *weak duality*, allows us to find the tightest lower bound by solving the *dual problem*

$$\max_{\lambda} \quad q(\lambda). \quad (7)$$

This problem is always convex, regardless of whether the primal problem is convex or not. Moreover, when $f(z)$ is strictly convex and under certain conditions on g , the dual function $q(\lambda)$ is differentiable and the gradient is given by

$$\frac{\partial q(\lambda)}{\partial \lambda_i} = g_i(z^*(\lambda)). \quad (8)$$

When the primal problem is convex and some constraint qualification such as the Slater's condition holds, the optimal values of the primal and dual problems coincide, i.e., $q(\lambda^*) = f(z^*)$. This result is known as *strong duality*, allows us to first minimize the Lagrangian function over z and then maximize the dual function over the Lagrange multipliers. In some cases, this may be computationally simpler, because the maximization of the dual function is an unconstrained optimization problem. In particular, the iterative procedure

$$z(t+1) = \arg \min_{z \in S} \{f(z) + \lambda(t)^\top g(z)\}, \quad \lambda(0) = 0, \quad (9)$$

$$\lambda(t+1) = \lambda(t) + \epsilon g(z(t+1))$$

can be shown to converge to the optimal solution (z^*, λ^*) for a suitable choice of the step-size $\epsilon > 0$.

¹Recall that the smallest eigenvalue is always zero and the second smallest eigenvalue is strictly positive if the graph is connected.

III. GENERALIZED CONSENSUS FOR ROBUST CONVEX COST FUNCTIONS

Given n measurements $\{u_i \in \mathbb{R}^D\}_{i=1}^n$, the goal of average consensus is to minimize the cost function

$$f(z) = \sum_{i=1}^n \|z - u_i\|_2^2 \quad (10)$$

in a distributed fashion. To that end, the average consensus algorithm described in Section II-A uses multiple copies of the variable z , $\{z_i\}_{i=1}^n$, to define the cost function in (3). This cost function has several minima, namely $z_i = \alpha$, $i \in V$, for any choice of α . Nonetheless, the algorithm in (1) converges to only one such minima, $\alpha = \bar{u}$, because it is initialized at $z_i = u_i$, $i \in V$, and the average of the states $\{z_i\}$ is preserved at each iteration, as per (2).

In order to make the average consensus algorithm robust to outliers, we could try to minimize a robust version of (3),

$$\varphi(z_1, z_2, \dots, z_n) = \sum_{i=1}^n \sum_{j \in \mathcal{N}_i} \rho(z_i - z_j). \quad (11)$$

Here $\rho: \mathbb{R}^D \rightarrow \mathbb{R}$ is a penalty function on the error vector $e = (e_1, \dots, e_D)^\top$ of the form $\rho(e) = \sum_{k=1}^D \phi(e_k)$, where $\phi: \mathbb{R} \rightarrow \mathbb{R}$ is a robust loss function, such as the ‘‘fair loss’’ [13]

$$\phi(e_k) = \sigma^2 \left(\frac{|e_k|}{\sigma} - \log \left(1 + \frac{|e_k|}{\sigma} \right) \right), \quad (12)$$

which is strictly convex, or the L_p loss

$$\phi(e_k) = |e_k|^p, \quad (13)$$

which is strictly convex for $p > 1$ and convex for $p = 1$. Another important robust cost function is the Huber loss [22]

$$\phi(e_k) = \begin{cases} e_k^2 & \text{if } |e_k| \leq \delta \\ 2\delta|e_k| - \delta^2 & \text{if } |e_k| > \delta \end{cases}, \quad (14)$$

which is a combination of the L_1 and L_2 losses. $\delta > 0$ is a parameter that represents the threshold of outliers and determines the trade-off between robustness and efficiency. The Huber loss is also convex, but not strictly convex.

As before, the cost function in (11) has several minima, namely $z_i = z_j$ for all $i, j \in V$. However, in this case it is not clear that the minimization of (11) using gradient descent (or a similar method) will converge to the minimizer of

$$f(z) = \sum_{i=1}^n \rho(z - u_i), \quad (15)$$

as in the case of the L_2 loss.

In this section, we propose an algorithm analogous to that in (1) for computing the minimizer of (15) in a distributed fashion. We pose this problem as a constrained optimization problem, whose solution can be found using a distributed primal-dual approach. We first derive the solution for strictly convex cost functions and show that the classical average consensus algorithm becomes a particular case of our formulation. We then extend the algorithm to convex functions

by using an augmented Lagrangian approach. Although our derivation is based on general functions, the result can be directly applied to robust cost functions as particular cases.

A. Robust Consensus as a Constrained Optimization Problem

Let $f_i: \mathbb{R}^D \rightarrow \mathbb{R}$ be an arbitrary function defined at each node $i \in V$. For example, we can choose $f_i(z) = \rho(z - u_i)$ for the robust average consensus problem. Suppose that we want to estimate a common quantity $z \in \mathbb{R}^D$ by solving the following optimization problem

$$\min_z \left(f(z) = \sum_{i=1}^n f_i(z) \right). \quad (16)$$

As suggested in [13], under the assumption that the graph G is strongly connected, the above optimization problem is equivalent to the following constrained optimization problem

$$\min_{z_1, z_2, \dots, z_n} \sum_{i=1}^n f_i(z_i) \quad \text{s.t.} \quad z_j = z_i \quad \forall i \in V, j \in \mathcal{N}_i. \quad (17)$$

The variables $\{z_i \in \mathbb{R}^D\}_{i=1}^n$ can be interpreted as the states of each one of the nodes in the sensor network and the local constraints $z_j = z_i$, $i \in V, j \in \mathcal{N}_i$, ensure that the optimal solution of (17) coincides with that of (16), i.e.,

$$z_1^* = z_2^* = \dots = z_n^* = z^*. \quad (18)$$

The reformulation of the optimization problem in (16) as the constrained optimization problem in (17) allows us to derive distributed algorithms for solving (16) in the case of both strictly convex and convex functions, as we show next.

B. Consensus for Strictly Convex Cost Functions

Assume first that the functions $\{f_i\}_{i=1}^n$ are strictly convex. As noted in [21], in this case the Lagrangian function is separable and can be minimized using a primal-dual approach. More specifically, let $\lambda_{i,j} \in \mathbb{R}^D$ be the Lagrange multiplier associated with the constraint $z_i - z_j = 0$, for all $i \in V, j \in \mathcal{N}_i$. Therefore, the Lagrangian function for the problem in (17) is given by

$$\mathcal{L}(\{z_i\}, \{\lambda_{i,j}\}) = \sum_{i=1}^n f_i(z_i) + \sum_{i=1}^n \sum_{j \in \mathcal{N}_i} \lambda_{i,j}^\top (z_i - z_j) \quad (19)$$

$$= \sum_{i=1}^n [f_i(z_i) + \sum_{j \in \mathcal{N}_i} (\lambda_{i,j} - \lambda_{j,i})^\top z_i], \quad (20)$$

which can be decomposed as the sum of n sub-Lagrangians

$$\mathcal{L}_i(z_i, \{\lambda_{i,j}\}) = f_i(z_i) + \sum_{j \in \mathcal{N}_i} (\lambda_{i,j} - \lambda_{j,i})^\top z_i, \quad (21)$$

each one depending on only the state at node $i \in V$, z_i , and the Lagrange multipliers at edges from and to node $i \in V$.

According to the duality results from Section II-B, the problem in (17) is equivalent to:

$$\min_{\{z_i\}} \max_{\{\lambda_{i,j}\}} \sum_{i=1}^n \mathcal{L}_i = \max_{\{\lambda_{i,j}\}} \min_{\{z_i\}} \sum_{i=1}^n \mathcal{L}_i = \max_{\{\lambda_{i,j}\}} \sum_{i=1}^n \min_{z_i} \mathcal{L}_i. \quad (22)$$

It follows from the right hand side of (22) that, when $\{\lambda_{i,j}\}$ are fixed, we can minimize \mathcal{L} over $\{z_i\}$ in a distributed fashion by minimizing separately each \mathcal{L}_i over z_i , i.e.,

$$\min_{z_i} [f_i(z_i) + \sum_{j \in \mathcal{N}_i} (\lambda_{i,j} - \lambda_{j,i})^\top z_i]. \quad (23)$$

Moreover, since the functions f_i are strictly convex and the constraints on the states are linear and finitely many, the dual function $q(\{\lambda_{i,j}\})$ is differentiable and the gradient is given by (see page 669 in [21] for further details)

$$\frac{\partial q}{\partial \lambda_{i,j}} = z_i^*(\{\lambda_{i,j}\}) - z_j^*(\{\lambda_{i,j}\}), \quad (24)$$

where $z_i^*(\{\lambda_{i,j}\})$ is the optimal solution to (23). As a consequence, we can maximize q over $\{\lambda_{i,j}\}$ in a distributed fashion by using the iterative algorithm in (9). Therefore, we have shown the following result.

Proposition 1: If the functions $\{f_i\}_{i=1}^n$ are strictly convex, the generalized consensus problem in (17) can be solved in a distributed fashion by alternating the following two updates for all $i \in V$, $j \in \mathcal{N}_i$:

$$z_i(t+1) = \arg \min_{z_i} [f_i(z_i) + \sum_{j \in \mathcal{N}_i} (\lambda_{i,j}(t) - \lambda_{j,i}(t))^\top z_i] \quad (25)$$

$$\lambda_{i,j}(t+1) = \lambda_{i,j}(t) + \epsilon(z_i(t+1) - z_j(t+1)), \quad (26)$$

where $\epsilon > 0$ is a learning rate and $\lambda_{i,j}(0) = 0$.

The above generalized consensus algorithm is very simple. It only involves local optimization over the state of each node, z_i , and a simple consensus-like update on the Lagrange multipliers, $\lambda_{i,j}$. However, its ability to handle an arbitrary strictly convex cost function comes at the price of increasing the communication requirements. Specifically, each node $i \in V$ needs to know not only the neighboring states $\{z_j\}_{j \in \mathcal{N}_i}$ (as in classical average consensus), but also the Lagrange multipliers of outgoing and incoming neighbors $\{\lambda_{i,j}\}_{j \in \mathcal{N}_i}$ and $\{\lambda_{j,i}\}_{j \in \mathcal{N}_i}$, respectively.

Fortunately, we can exploit the structure of the update equations to reduce the communication requirements. Specifically, notice that if we define a new variable at each node as $\beta_i = \sum_{j \in \mathcal{N}_i} (\lambda_{i,j} - \lambda_{j,i})$, we obtain a new, simpler algorithm.

Proposition 2: If the functions $\{f_i\}_{i=1}^n$ are strictly convex, the generalized consensus problem in (17) can be solved in a distributed fashion by alternating the following two updates for all $i \in V$:

$$z_i(t+1) = \arg \min_{z_i} [f_i(z_i) + \beta_i(t)^\top z_i] \quad (27)$$

$$\beta_i(t+1) = \beta_i(t) + 2\epsilon \sum_{j \in \mathcal{N}_i} (z_i(t+1) - z_j(t+1)), \quad (28)$$

where $\epsilon > 0$ is a learning rate and $\beta_i(0) = 0$.

This new algorithm has the same communication requirement as the classical average consensus. The only additional cost is that each node needs to store and update one additional variable β_i , $i \in V$, which can be seen as an auxiliary state. Notice that the auxiliary states satisfy the constraint

$\sum_{i=1}^n \beta_i = 0$. Notice also that this constraint is automatically enforced, because each β_i is initialized as $\beta_i(0) = 0$ and the update in (28) is such that $\sum_{i=1}^n \beta_i(t+1) = \sum_{i=1}^n \beta_i(t) = 0$.

Remark 1 (Differences with [13]): The consensus algorithm proposed in [13] also tries to address the optimization problem in (17) under the assumption that each loss function f_i is strictly convex. However, their derivation decomposes the Lagrangian in (19) as the sum of the n sub-Lagrangians

$$f_i(z_i) + \sum_{j \in \mathcal{N}_i} \lambda_{i,j}^\top (z_i - z_j), \quad \forall i \in V \quad (29)$$

Notice that these sub-Lagrangians depend not only on the state z_i , but also on the states z_j , $j \in \mathcal{N}_i$. As a consequence, the duality theorem in (22) is not directly applicable.

Remark 2 (Average consensus as a particular case):

As stated at the beginning of this section, the classical average consensus can be formulated as

$$\min_{z_1, z_2, \dots, z_n} \sum_{i=1}^n \|z_i - u_i\|_2^2 \quad \text{s.t.} \quad z_i = z_j, \forall i \in V, j \in \mathcal{N}_i, \quad (30)$$

where $\{u_i\}_{i=1}^n$ are the measurements at the n nodes. If we apply our algorithm in (27)–(28) to this problem by substituting $f_i(z) = \|z_i - u_i\|_2^2$ in (27), we obtain the following update for the state at each node

$$z_i(t+1) = u_i - \beta_i(t)/2. \quad (31)$$

Substituting this expression for $\beta_i(t)$ into (28) yields the classical average consensus algorithm in (1).

C. Consensus for Convex Cost Functions

The algorithm described in the previous subsection is an extremely simple extension of classical consensus to strictly convex loss functions. However, many robust loss functions are not strictly convex, e.g., the L_1 loss $\rho(e) = \|e\|_1$.

In this subsection, we show how to extend consensus to convex functions. In principle, one may think that the approach for strictly convex functions applies directly to the case of convex functions. However, as we alluded to earlier, the assumption of strict convexity is needed in order to ensure the differentiability of the dual cost function. Therefore, for functions that are not strictly convex, the update of the Lagrange multipliers in (26) may not be applicable.

In order to address this issue, we could use the Augmented Lagrangian approach [23]. In this approach, a quadratic term is added to the cost function in order to make it strictly convex. The quadratic term is chosen as the sum of the squares of the constraints, which gives the following augmented Lagrangian

$$\sum_{i=1}^n f_i(z_i) + \sum_{i=1}^n \sum_{j \in \mathcal{N}_i} \lambda_{i,j}^\top (z_i - z_j) + \frac{c}{2} \sum_{i=1}^n \sum_{j \in \mathcal{N}_i} \|z_i - z_j\|^2, \quad (32)$$

where $c > 0$ is a parameter. Since the additional term is zero for any feasible solution, the optimal solution of the original problem does not change. However, although

the resulting augmented Lagrangian resolves the problem of differentiability of the dual function, it is not a separable function. That is, we cannot directly write it as the sum of n functions $\{\mathcal{L}_i\}_{i=1}^n$, where each \mathcal{L}_i depends only on z_i .

To address this problem, we use a modified augmented Lagrangian approach.² The main idea is to introduce a set of auxiliary variables into the optimization problem in (17) and build a separable augmented Lagrangian function using more constraints. This leads to the iterative optimization algorithm given in the following proposition.

Proposition 3: If the functions $\{f_i\}_{i=1}^n$ are convex, but not necessarily strictly convex, the generalized consensus problem in (17) can be solved in a distributed fashion by alternating the following two updates for all $i \in V, j \in \mathcal{N}_i$:

$$z_i(t+1) = \arg \min_{z_i} \{f_i(z_i) + \sum_{j \in \mathcal{N}_i} (\lambda_{i,j}(t) - \lambda_{j,i}(t))^\top z_i + \sum_{j \in \mathcal{N}_i} c(t) [z_i - \frac{1}{2}(z_i(t) + z_j(t))]^2\} \quad (33)$$

$$\lambda_{i,j}(t+1) = \lambda_{i,j}(t) + \frac{c(t)}{2} (z_i(t+1) - z_j(t+1)), \quad (34)$$

where $\{c(t) > 0\}$ is nondecreasing and $\lambda_{i,j}(0) = 0$.

Proof: If we introduce a set of auxiliary variables $\{w_{ij}\}$ and two new sets of constraints $\{z_i = w_{ij}\}$ and $\{z_j = w_{ij}\}$, the optimization problem in (17) transforms to:

$$\begin{aligned} \min_{\{z_i\}, \{w_{ij}\}} \quad & \sum_{i=1}^n f_i(z_i) \\ \text{s.t.} \quad & z_i = w_{ij}, z_j = w_{ij} \quad \forall i \in V, j \in \mathcal{N}_i. \end{aligned} \quad (35)$$

Since the above set of constraints implies that $z_i - z_j = 0$, $\forall i \in V, j \in \mathcal{N}_i$, the optimal solution for $\{z_i\}$ in this problem is identical to that of (17).

In order to solve this new optimization problem, let $\{\gamma_{ij}^1\}$ and $\{\gamma_{ij}^2\}$ be the Lagrange multipliers for the constraints $\{z_i = w_{ij}\}$ and $\{z_j = w_{ij}\}$, respectively. The modified augmented Lagrangian can be defined as

$$\begin{aligned} \mathcal{L}_c(\{z_i\}, \{w_{ij}\}, \{\gamma_{ij}^1\}, \{\gamma_{ij}^2\}) &= \sum_{i=1}^n f_i(z_i) \\ &+ \sum_{i=1}^n \sum_{j \in \mathcal{N}_i} [\gamma_{ij}^1{}^\top (z_i - w_{ij}) + \gamma_{ij}^2{}^\top (z_j - w_{ij})] \\ &+ \frac{c}{2} \sum_{i=1}^n \sum_{j \in \mathcal{N}_i} [(z_i - w_{ij})^2 + (z_j - w_{ij})^2]. \end{aligned} \quad (36)$$

Applying the method of augmented Lagrange multipliers to the above augmented Lagrangian leads to the following alternate minimizations with respect to $\{z_i\}$ and $\{w_{ij}\}$:

$$\begin{aligned} \{z_i(t+1)\} &= \arg \min_{\{z_i\}} \mathcal{L}_{c(t)}(\{z_i\}, \{w_{ij}(t)\}, \{\gamma_{ij}^1(t)\}, \{\gamma_{ij}^2(t)\}) \\ \{w_{ij}(t+1)\} &= \arg \min_{\{w_{ij}\}} \mathcal{L}_{c(t)}(\{z_i(t+1)\}, \{w_{ij}\}, \{\gamma_{ij}^1(t)\}, \{\gamma_{ij}^2(t)\}), \end{aligned} \quad (37)$$

²A similar approach is used in pages 243-253 of [21] for solving a related problem.

and the following updates for γ_{ij}^1 and γ_{ij}^2 :

$$\gamma_{ij}^1(t+1) = \gamma_{ij}^1(t) + c(t)(z_i(t+1) - w_{ij}(t+1)) \quad (38)$$

$$\gamma_{ij}^2(t+1) = \gamma_{ij}^2(t) + c(t)(z_j(t+1) - w_{ij}(t+1)), \quad (39)$$

for all $i \in V, j \in \mathcal{N}_i$, where $\{c(t) > 0\}$ is nondecreasing.

In order to solve the minimization problem in (37) with respect to z_i in a distributed fashion, we can use the same techniques as in (20) to make the augmented Lagrangian separable. The resulting update for $z_i, \forall i \in V$, is given by:

$$z_i(t+1) = \arg \min_{z_i} \{f_i(z_i) + \sum_{j \in \mathcal{N}_i} (\gamma_{ij}^1(t) + \gamma_{ji}^2(t))^\top z_i + \frac{c(t)}{2} \sum_{j \in \mathcal{N}_i} [(z_i - w_{ij}(t))^2 + (z_i - w_{ji}(t))^2]\}. \quad (40)$$

As per w_{ij} , a closed form update can be obtained by taking the partial derivative of the augmented Lagrangian and setting it to zero, which gives

$$w_{ij}(t+1) = \frac{1}{2} [z_i(t+1) + z_j(t+1) + \frac{\gamma_{ij}^1(t) + \gamma_{ij}^2(t)}{c(t)}]. \quad (41)$$

The rest of the proof is simply to combine the two sets of constraints and to effectively eliminate the update for the auxiliary variable w_{ij} . Adding (38) and (39) gives rise to

$$\begin{aligned} \gamma_{ij}^1(t+1) + \gamma_{ij}^2(t+1) &= \gamma_{ij}^1(t) + \gamma_{ij}^2(t) \\ &+ c(t)[z_i(t+1) + z_i(t+1) - 2w_{ij}(t+1)], \end{aligned} \quad (42)$$

from which we obtain update for w_{ij} as

$$\begin{aligned} w_{ij}(t+1) &= \frac{1}{2} [z_i(t+1) + z_j(t+1) \\ &+ \frac{\gamma_{ij}^1(t) + \gamma_{ij}^2(t)}{c(t)} - \frac{\gamma_{ij}^1(t+1) + \gamma_{ij}^2(t+1)}{c(t)}]. \end{aligned} \quad (43)$$

Comparing (41) and (43) gives the following important relation, which reduces the sets of Lagrangian multipliers to only one (denoted as $\{\lambda_{i,j}\}$):

$$\gamma_{ij}^1(t+1) = -\gamma_{ij}^2(t+1) \doteq \lambda_{i,j}(t+1). \quad (44)$$

Then the update equation for the multipliers can be computed by subtracting (39) from (38) and then changing the multipliers according to the above equation to obtain

$$\lambda_{i,j}(t+1) = \lambda_{i,j}(t) + \frac{c(t)}{2} (z_i(t+1) - z_j(t+1)). \quad (45)$$

Finally, since the equation for w_{ij} at iteration t reduces to

$$w_{ij}(t) = \frac{1}{2} [z_i(t) + z_j(t)], \quad (46)$$

the update for z_i in (40) reduces to

$$\begin{aligned} z_i(t+1) &= \arg \min_{z_i} \{f_i(z_i) + \sum_{j \in \mathcal{N}_i} (\lambda_{i,j}(t) - \lambda_{j,i}(t))^\top z_i \\ &+ \sum_{j \in \mathcal{N}_i} c(t) [z_i - \frac{1}{2}(z_i(t) + z_j(t))]^2\}. \end{aligned} \quad (47)$$

At last, we point out that the convergence of our iterative algorithm can be proved as a special case of that of Augmented Lagrangian Methods in page 245 of [21]. \blacksquare

Remark 3: As in Proposition 2, we can eliminate the Lagrange multipliers and replace them by an auxiliary state $\beta_i(t) = \sum_{j \in \mathcal{N}_i} (\lambda_{i,j}(t) - \lambda_{j,i}(t))$ to obtain the updates

$$z_i(t+1) = \arg \min_{z_i} \{f_i(z_i) + \beta_i(t)^\top z_i + \sum_{j \in \mathcal{N}_i} c(t) [z_i - \frac{1}{2}(z_i(t) + z_j(t))]^2\} \quad (48)$$

$$\beta_i(t+1) = \beta_i(t) + c(t) \sum_{j \in \mathcal{N}_i} (z_i(t+1) - z_j(t+1)), \quad (49)$$

where $\beta_i(0) = 0$, for all $i \in V$.

IV. EXPERIMENTS

In this section, we evaluate the performance of the proposed robust consensus algorithms on synthetic data. We consider a network with $n = 100$ nodes with the communication graph being a connected 6-regular lattice, *i.e.*, each node is connected to 6 nearest neighbors in the network. The nodes' measurements, $\{u_i\}$, are generated from the zero-mean Gaussian distribution with unit standard deviation. For $P\%$ of the nodes, where $P \in [0, 30]$, we corrupt the measurements by large outliers drawn from a uniform distribution on the interval $[30, 100]$.

We solve the optimization problem in (16) using the proposed distributed optimization algorithms with five loss functions: the fair loss in (12), the L_p loss (13) with $p = 2, 1.5$, and 1, and the Huber loss in (14). We emphasize that the fair, L_2 and $L_{1.5}$ losses are strictly convex, while the L_1 and Huber losses are convex, but not strictly convex. A step size of $\epsilon = 0.056$ is used for the L_2 and $L_{1.5}$ losses. For the fair loss, a parameter $\sigma = 3.5$ and a step size $\epsilon = 0.00056$ are used. For the other cases, we use a constant step size $c = 0.056$. The threshold for Huber loss is set to $\delta = 5$.

Figures 1a-1e plot the states $z_i(t)$ of all the nodes for each one of the five loss functions, respectively, when $P = 15\%$. Notice that all five algorithms converge to a consensus configuration, which coincides with the centralized solution for the corresponding loss function shown by the dash line. However, the consensus configuration does not coincide with the mean of the outlier-free data, which is zero in this case. This issue is further illustrated in Figure 1f, which shows the centralized solutions of the five loss functions as a function of the percentage of outliers. As expected, the classical average consensus algorithm, which uses the L_2 loss, is very sensitive to outliers. The performance of the $L_{1.5}$ loss is better than that of the L_2 loss, but the fair and Huber losses are even more robust. Overall, the most robust loss is the L_1 loss.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we considered the problem of distributed optimization in sensor networks with robustness to outliers in the measurements. To address this problem, we proposed a generalized consensus algorithm in which each node minimizes a local robust cost function, while neighboring nodes perform a consensus like-update on Lagrange multipliers defined for each edge of the network. We showed that

the proposed distributed algorithm converges to the optimal solution of the equivalent centralized optimization problem. We also studied the performance of the iterative approach for convex and strictly convex cost functions through simulation experiments. Future work includes extending the results to the class of quasi-convex functions as well as deriving more efficient updates on the dual variables which reduce the amount of Lagrange multipliers used at each node. Extensions to asymmetrically directed graph or time-varying graph topologies are also of interests to study.

REFERENCES

- [1] R. Olfati-Saber, J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.
- [2] W. Ren, R. Beard, and E. Atkins, "Information consensus in multivehicle cooperative control," *IEEE Control Systems Magazine*, pp. 71–82, April 2007.
- [3] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 3, pp. 1520–1533, 2004.
- [4] F. Bullo, J. Cortés, and S. Martínez, *Distributed Control of Robotic Networks*, Applied Mathematics Series. Princeton University Press, 2009, Electronically available at <http://coordinationbook.info>.
- [5] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Symposium on Information Processing of Sensor Networks (IPSN)*, 2005, pp. 63–70.
- [6] J. Cortés, "Distributed algorithms for reaching consensus on general functions," *Automatica*, vol. 44, no. 3, pp. 726–737, 2008.
- [7] A. Sarlette and R. Sepulchre, "Consensus optimization on manifolds," *SIAM J. Control and Optimization*, vol. 48, no. 1, pp. 56–76, 2009.
- [8] R. Tron, R. Vidal, and A. Terzis, "Distributed pose averaging in camera networks via consensus on SE(3)," in *International Conference on Distributed Smart Cameras*, 2008.
- [9] A. Forero, A. Cano, and G. Giannakis, "Consensus-based distributed linear support vector machines," in *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, New York, USA, 2010, pp. 35–46.
- [10] L. Xiao, S. Boyd, and S. Kim, "Distributed average consensus with least-mean-square deviation," *Journal of Parallel and Distributed Computing*, vol. 67, pp. 33–46, 2007.
- [11] M. Huang and J. Manson, "Stochastic approximation for consensus seeking: mean square and almost sure convergence," in *Proceedings of the 46th IEEE Conference on Decision and Control*, New Orleans, LA, USA, Dec 2007, pp. 306–311.
- [12] S. Stanković, M. Stanković, and D. Stipanović, "Decentralized parameter estimation by consensus based stochastic approximation," in *Proceedings of the 46th IEEE Conference on Decision and Control*, New Orleans, LA, USA, Dec 2007, pp. 1535–1540.
- [13] M. Rabbat, R. Nowak, and J. Bucklew, "Generalized consensus computation in networked systems with erasure links," in *Workshop on Signal Processing Advances in Wireless Communications*, June 2005, pp. 1088–1092.
- [14] I. Schizas, A. Ribeiro, and G. Giannakis, "Consensus in ad hoc WSNs with noisy links - Part I: Distributed estimation of deterministic signals," *IEEE Transactions on Signal Processing*, vol. 56, pp. 350–364, 2008.
- [15] M. Rabbat and R. Nowak, "Distributed optimization in sensor networks," in *International symposium on information processing in sensor networks*, New York, NY, USA, 2004, pp. 20–27, ACM.
- [16] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 48–61, 2009.
- [17] M. Zhu and S. Martínez, "On distributed optimization under inequality and equality constraints via penalty primal-dual methods," in *Proceedings of the 2010 American Control Conference*, Baltimore, USA, 2010, pp. 4863–4868.
- [18] S. Samar, S. Boyd, and D. Gorinevsky, "Distributed estimation via dual decomposition," in *Proceedings of the 2007 European Control Conference*, Kos, Greece, 2007, pp. 1511–1516.

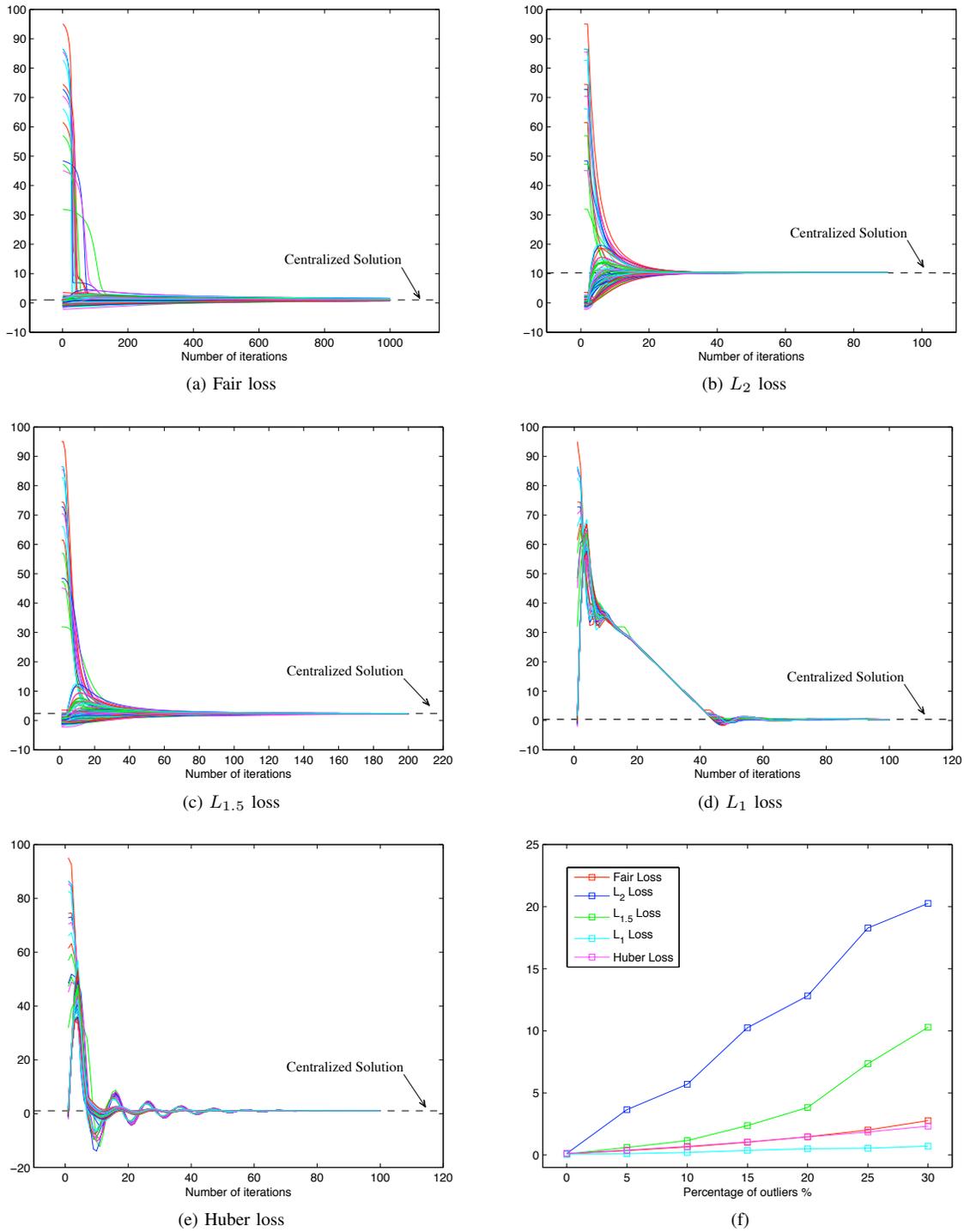


Fig. 1: Evaluation of the proposed robust consensus algorithms on synthetic data contaminated with outliers. Figures (a)-(e) show the states of $n = 100$ nodes as a function of the number of iterations for the fair, L_2 , $L_{1.5}$, L_1 , and Huber loss functions, respectively, when the percentage of outliers is $P = 15\%$. All the states converge to a consensus configuration given by the corresponding centralized algorithm. Figure (f) plots the centralized solution to which the algorithms converge as a function of the percentage of outliers for the different loss functions.

[19] A. Olshevsky and J. Tsitsiklis, "Convergence speed in distributed consensus and averaging," *SIAM Journal of Control and Optimization*, vol. 48, no. 1, pp. 33–55, 2007.

[20] R. Rockafellar, *Convex analysis*, Princeton, N.J., Princeton University Press, 1997, C1970.

[21] D. Bertsekas and J. Tsitsiklis, *Parallel and Distributed Computations*, Prentice-Hall, 1989.

[22] P. Huber, *Robust Statistics*, John Wiley & Sons, New York, 1981.

[23] M. Hestenes, "Multiplier and gradient methods," *Journal of Optimization Theory and Applications*, vol. 4, no. 5, pp. 303–320, 1969.