A distributed dual proximal minimization algorithm for constraint-coupled optimization problems

Alessandro Falsone *Member, IEEE,* and Maria Prandini *Fellow, IEEE*

***Abstract*—We address constraint-coupled optimization for a system composed of multiple cooperative agents communicating over a time-varying network. We propose a distributed proximal minimization algorithm that is guaranteed to converge to an optimal solution of the optimization problem, under suitable convexity and connectivity assumptions. The performance of the introduced algorithm is shown on a numerical example of a charging scheduling problem for a fleet of plug-in electric vehicles.**

***Index Terms*—Optimization algorithms, distributed control, agents-based systems**

I. Introduction

In this letter, we consider a system composed of multiple agents that have computational capabilities and can communicate with neighboring agents to solve a (convex) optimization problem involving the entire system. Each agent has its own decision variables, cost function and constraint set, and the goal is to minimize the sum of the agents' cost functions subject to a coupling constraint involving all decision variables (*Constraint-Coupled Problem* – CCP). The presence of the coupling element makes the problem solution challenging, especially in a distributed framework where no central authority is available to coordinate the agents and each agent has no knowledge of the local information of the other agents.

Even though CCPs arise naturally in practical applications, typically due to the fact that agents share resources with a finite capacity, most of the literature on distributed optimization focuses on optimization problems where the agents are coupled because they have to agree on common decision variables (*Decision-Coupled Problems* – DCPs).

The earliest distributed solutions to DCPs are algorithms based on a combination of standard (sub)gradient methods for the optimization of the agents local cost functions and consensus schemes to drive the agents towards a common optimal decision, see, e.g., [1], [2]. Subsequent works propose solutions resting on consensus-based primal-dual approaches, see, e.g., [3]. Finally, the work in [4] deals with DCPs combining a consensus scheme with the proximal minimization algorithm instead of the (sub)gradient method for the optimization of the agents local cost functions.

Note that, in principle, the distributed methods developed for DCPs could be applied to CCPs by defining as common decision variables the collection of the decision variables of all the agents. However, this would require each agent to store and update the tentative solutions of all other agents and have access to the whole coupling constraint rather than its portion only, thus increasing communication and computational burden and ultimately hampering the applicability of such strategies.

Approaches to directly tackle CCPs leverage Lagrangian duality to deal with the coupling constraint. Works based on primal-dual algorithms (e.g., [5]) seek to find the saddle point of the Lagrangian, but require an additional procedure to recover the optimal solution of the CCP. Methods based on dual decomposition (e.g., [6]) solve the dual of the CCP via a distributed subgradient method, and still require a recovery procedure. Finally, the approach in [7] is also based on a distributed subgradient scheme but adopts successive duality steps to obtain the optimal solution of the CCP without calling for any recovery procedure.

In this letter, we introduce a proximal-based algorithm for CCPs and analyze its convergence properties. The proposed approach is the counterpart of [4] for DCPs, exactly as [6] and [5] are the counterpart of [1] and [3], respectively. The dual of the addressed CCP has the structure of a DCP, but since it does not match the assumptions in [4], a nontrivial extension of the convergence analysis in [4] is required. Also, recovery of the primal optimal solution has to be worked out. Proximal-based algorithms are known to be more stable than their gradient counterpart ([8]), and indeed the proposed method presents a smoother behavior through iterates when compared to [6] and also [5], as shown in the numerical example section.

All the cited approaches, including the one proposed here, work under mild assumptions on the regularity of the cost function and on the communication network. However, they require a vanishing step-size for the gradient update or an increasing penalty for the proximal operator, which ultimately leads to a slow convergence rate. A first attempt to improve the convergence rate by using a fixed step-size is presented in [9] where a consensus scheme is used together with the Alternating Direction Method of Multipliers (ADMM, see [10]) in place of the (sub)gradient method, but the approach requires stricter assumptions on the communication network. Linear convergence rate for distributed algorithms with constant step-size solving DCPs has been recently achieved by the so-called gradient-tracking schemes (see, e.g., [11], [12]), where the gradient method is used together with a technique known as dynamic average consensus (firstly proposed in [13] and further elaborated in [14]) in place of the original consensus algorithm. In these methods, however, more restrictive assumptions on the agents' local cost functions are imposed. Similarly to DCPs, latest distributed algorithms for solving

The authors are with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Via Ponzio 34/5, 20133 Milano, Italy. Email: name.surname@polimi.it.

TABLE I
DISTRIBUTED ALGORITHMS FOR DCPS AND CCPS.

	Vanishing step-size		Constant step-size	
	DCP	CCP	DCP	CCP
Gradient	[1], [2]	[6], [7]	[11], [12]	
Proximal	[4]	this work		
Saddle point	[3]	[5]		[15]
ADMM	–	–	[9]	[16], [17]

CCPs are aiming at improving the convergence rate via a constant step-size. A primal-dual algorithm with constant step-size is proposed in [15], but requires smoothness of the local cost functions. An approach based on ADMM and consensus is derived in [16], and a strategy combining ADMM and dynamic average consensus with faster convergence is presented in [17]. Both approaches impose stricter assumptions on the communication network.

This short review of the literature is summarized in Table I, where contributions are classified in the (sub)gradient minimization, proximal minimization, saddle point seeking, and ADMM categories. Many variants of the mentioned algorithms have been proposed in the last decade but are here omitted in the interest of space.

This letter completes the picture of the methods in the vanishing step-size column, by introducing a proximal-based algorithm for CCPs. Our contribution then lays the foundations for future development of improved distributed proximal-based algorithms.

The rest of the paper is organized as follows. In Section II we present the set-up of constraint-coupled optimization problems. In Section III we derive the proposed distributed dual proximal algorithm and state its convergence properties. In Section IV we apply our algorithm to a realistic application related to the optimal charging schedule for a fleet of electric vehicles. In Section V we draw some conclusions. Proofs of the main results are reported in the Appendix.

II. CONSTRAINT-COUPLED OPTIMIZATION

We consider a multi-agent system composed of N agents that are willing to cooperate while addressing a decision making problem involving the whole system. Specifically, all agents shall set their local decision variables $x_i \in \mathbb{R}^{n_i}$, $i = 1, \dots, N$, so as to find an optimal solution to the following constrained optimization program

$$\begin{aligned} \min_{x_1,\dots,x_N} \quad & \sum_{i=1}^{N} f_i(x_i) \\ \text{subject to:} \quad & \sum_{i=1}^{N} A_i x_i = b \\ & x_i \in X_i \qquad i = 1, \dots, N, \end{aligned} \tag{$\mathcal{P}$}$$

where $A_i \in \mathbb{R}^{p \times n_i}$ and $b \in \mathbb{R}^p$ specify the coupling constraint, $f_i : \mathbb{R}^{n_i} \to \mathbb{R}$ and $X_i \subset \mathbb{R}^{n_i}$ are, respectively, the local cost function and constraint of agent i, $i = 1, \dots, N$, which satisfy the following assumption:

Assumption 1 (Convexity and compactness)**.** *For all $i = 1, \dots, N$, the function f_i is convex and the set X_i is convex and compact.* □

To deal with problems in the form of $\mathcal{P}$, a common practice is resorting to duality theory to handle the coupling constraint. Let $\boldsymbol{x} = [x_1^\top \ \cdots \ x_N^\top]^\top$, consider a vector $\lambda \in \mathbb{R}^p$ of Lagrange multipliers and let

$$\begin{aligned} L(\boldsymbol{x}, \lambda) &= \sum_{i=1}^{N} f_i(x_i) + \lambda^\top \left(\sum_{i=1}^{N} A_i x_i - b \right) \\ &= \sum_{i=1}^{N} f_i(x_i) + \lambda^\top (A_i x_i - \tfrac{b}{N}) \end{aligned} \tag{1}$$

be the Lagrangian function obtained by dualizing the coupling constraint $\sum_{i=1}^{N} A_i x_i = b$. The dual of $\mathcal{P}$ is then

$$\max_{\lambda \in \mathbb{R}^p} \min_{\boldsymbol{x} \in X} L(\boldsymbol{x}, \lambda) = \max_{\lambda \in \mathbb{R}^p} \ \sum_{i=1}^{N} \varphi_i(\lambda), \tag{$\mathcal{D}$}$$

where $X = X_1 \times \cdots \times X_N$, and the i-th contribution φ_i is defined as

$$\varphi_i(\lambda) = \min_{x_i \in X_i} f_i(x_i) + \lambda^\top (A_i x_i - \tfrac{b}{N}). \tag{2}$$

The next assumption ensures that $\mathcal{P}$ and $\mathcal{D}$ are well-posed.

Assumption 2 (Existence of optimal solutions)**.** *Problem $\mathcal{P}$ admits an optimal solution $\boldsymbol{x}^\star = [x_1^{\star\top} \ \cdots \ x_N^{\star\top}]^\top$ and problem $\mathcal{D}$ admits an optimal solution $\lambda^\star$.* □

Remark 1 (Inequality coupling constraint)**.** *Note that a linear coupling constraint of the form $\sum_{i=1}^{N} \alpha_i^\top x_i \leq \beta$, $\alpha_i \in \mathbb{R}^{n_i}$ and $\beta \in \mathbb{R}$, can easily be rewritten as an equality constraint by adding a slack variable s_i for each agent and imposing the equivalent constraint $\sum_{i=1}^{N} (\alpha_i^\top x_i + s_i) = \beta$, with $s_i \in [0, M_i]$ and $M_i \geq \max_{x_i \in X_i} (\frac{\beta}{N} - \alpha_i^\top x_i)$. This can be trivially generalized to the case of multiple linear inequality coupling constraints, thus showing that $\mathcal{P}$ includes also coupling constraints that are (linear) inequalities.* □

III. DISTRIBUTED DUAL PROXIMAL ALGORITHM

In this section we first introduce the distributed computational framework of interest, and then propose a novel proximal-based algorithm exploiting such a framework for solving $\mathcal{P}$. Finally, we analyze its convergence properties.

A. Distributed Computation Framework

In the considered distributed set-up, the cost function f_i, constraint set X_i, and contribution A_i to the coupling constraint have to be regarded as private information of agent i not to be disclosed to other agents, whereas the total amount b available to the overall multi-agent system is known to everybody.

To cooperatively solve $\mathcal{P}$, the agents must repeatedly exchange information through the communication network.

At each iteration k, communications among the agents are modeled as a directed graph $\mathcal{G}_k = (\mathcal{V}, \mathcal{E}_k)$, where the set $\mathcal{V} = \{1, \dots, N\}$ represents the agents, and the set $\mathcal{E}_k \subseteq \mathcal{V} \times \mathcal{V}$

collects the set of communication links that are active at k, with $(i,j) \in \mathcal{E}_k$ modeling the fact that agent i receives information from agent j during iteration k. We denote by $\mathcal{N}_{i,k} = \{j \in \mathcal{V} \mid (i,j) \in \mathcal{E}_k\}$ the set of *neighbors* of agent i at k. We assume that $(i,i) \in \mathcal{E}_k$ for all $i = 1, \dots, N$ and for all $k \geq 0$.

Let $\mathcal{E}_\infty = \{(i,j) : (i,j) \in \mathcal{E}_k \text{ for infinitely many } k\}$. To ensure that the agents communicate sufficiently often, we impose the following assumption.

Assumption 3 (Connectivity)**.** *The graph $(\mathcal{V}, \mathcal{E}_\infty)$ is strongly connected, i.e., for every pair of vertices in $\mathcal{V}$ there exists a directed path of arcs in $\mathcal{E}_\infty$ that connects them. Moreover, there exists a $T \geq 1$ such that for every $(i,j) \in \mathcal{E}_\infty$ agent i receives information from agent j at least once every consecutive T iterations.* □

B. Proposed Algorithm

We start by noticing that $\mathcal{D}$ is a DCP and thus it could be solved in a distributed way by applying the proximal Algorithm in [4]. Accordingly, agent i would run the following two steps at each iteration:

$$\ell_{i,k} = \sum_{j \in \mathcal{N}_{i,k}} w_{ij,k}\, \lambda_{j,k} \tag{3a}$$

$$\lambda_{i,k+1} = \operatorname*{argmax}_{\lambda_i \in \mathbb{R}^p} \varphi_i(\lambda_i) - \frac{1}{2c_k}\|\lambda_i - \ell_{i,k}\|^2. \tag{3b}$$

where c_k is a penalty coefficient and $w_{ij,k}$ is the weight associated to edge $(i,j) \in \mathcal{E}_k$ at iteration k and models how much agent i values the information received by agent j at k. For those $(i,j) \notin \mathcal{E}_k$, $w_{ij,k} = 0$, modeling the fact that agent i does not receive any information from agent j at k.

The penalty coefficient and the network weights should satisfy the following conditions to get convergence of (3) to an optimal solution $\lambda^\star$.

Assumption 4 (Penalty coefficient)**.** *Sequence $\{c_k\}_{k\geq 0}$ is positive, non-increasing, and satisfies $\sum_{k=0}^{\infty} c_k = +\infty$ and $\sum_{k=0}^{\infty} c_k^2 < +\infty$.*

Assumption 5 (Network weights)**.** *There exists $\eta > 0$ such that, for all $i, j = 1, \dots, N$ and $k \geq 0$, $w_{ij,k} \in [0, 1)$, $w_{ii,k} \geq \eta$, and $w_{ij,k} > 0$ implies $w_{ij,k} \geq \eta$. Furthermore, for all $k \geq 0$,*

- $\sum_{i=1}^{N} w_{ij,k} = 1$ *for all* $j = 1, \dots, N$,
- $\sum_{j=1}^{N} w_{ij,k} = 1$ *for all* $i = 1, \dots, N$,

and $w_{ij,k} > 0$ if and only if $(i,j) \in \mathcal{E}_k$. □

Assumptions 3–5 are common in the consensus-based distributed optimization literature, see, e.g., [1], [2]. For the choice of the network weights, see the discussion after Assumption 3 in [18].

In (3a), agent i constructs a weighted average $\ell_{i,k}$ of its own estimate $\lambda_{i,k}$ of the dual optimal solution and the estimates $\lambda_{j,k}$ of its neighboring agents, then, in (3b), it updates its local estimate by solving a local maximization problem where the cost function is given by its local contribution $\varphi_i(\cdot)$ to the dual function $\varphi(\cdot)$, plus a quadratic term that penalizes the distance of the new estimate from the average $\ell_{i,k}$ computed in (3a).

Algorithm 1 Distributed Dual Proximal Algorithm

Initialization

$x_{i,0} \in \mathbb{R}^{n_i}$, $\lambda_{i,0} \in \mathbb{R}^p$, $\hat{x}_{i,0} \in \mathbb{R}^{n_i}$, $k = 0$

For each iteration k do

$\ell_{i,k} = \sum_{j \in \mathcal{N}_{i,k}} w_{ij,k}\, \lambda_{j,k}$

$x_{i,k+1} \in \operatorname*{argmin}_{x_i \in X_i} f_i(x_i) + \ell_{i,k}^\top A_i x_i + \frac{c_k}{2}\|A_i x_i - \frac{b}{N}\|^2$

$\lambda_{i,k+1} = \ell_{i,k} + c_k(A_i x_{i,k+1} - \frac{b}{N})$

$\hat{x}_{i,k+1} = \hat{x}_{i,k} + \frac{c_k}{\sum_{s=0}^{k} c_s}(x_{i,k+1} - \hat{x}_{i,k})$

$k \leftarrow k + 1$

Note that the optimizer $\lambda_{i,k+1}$ in (3b) always exists and is unique due to the strict convexity of the quadratic penalty term.

Unfortunately, step (3b) involves maximizing a function $\varphi_i(\cdot)$ implicitly defined via a minimization (cf. (2)), which is not easy in general. We therefore shall first derive an equivalent formulation of (3) which is easier to implement.

Since the maximization in (3b) is unconstrained, for $\lambda_{i,k+1}$ to be the optimal solution it must hold that

$$0 \in \partial\varphi_i(\lambda_{i,k+1}) - \tfrac{1}{c_k}(\lambda_{i,k+1} - \ell_{i,k}), \tag{4}$$

where $\partial\varphi_i(\lambda_{i,k+1})$ is the set of all subgradients of $\varphi_i(\lambda)$ at $\lambda = \lambda_{i,k+1}$ (subdifferential). Moreover, as a consequence of the Danskin's theorem (see [19, Proposition B.25]), we have that the subdifferential can be characterized as

$$\partial\varphi_i(\lambda_{i,k+1}) = \Big\{ A_i x_{i,k+1} - \tfrac{b}{N} : x_{i,k+1} \in \operatorname*{argmin}_{x_i \in X_i} f_i(x_i) + \lambda_{i,k+1}^\top A_i x_i \Big\}. \tag{5}$$

Using (5) together with (4), we can conclude that there must exist some $x_{i,k+1}$ such that

$$x_{i,k+1} \in \operatorname*{argmin}_{x_i \in X_i} f_i(x_i) + \lambda_{i,k+1}^\top A_i x_i \tag{6a}$$

$$0 = A_i x_{i,k+1} - \tfrac{b}{N} - \tfrac{1}{c_k}(\lambda_{i,k+1} - \ell_{i,k}). \tag{6b}$$

Condition (6b) can be rewritten as

$$\lambda_{i,k+1} = \ell_{i,k} + c_k(A_i x_{i,k+1} - \tfrac{b}{N}) \tag{7}$$

and plugged into (6a) to obtain

$$x_{i,k+1} \in \operatorname*{argmin}_{x_i \in X_i} f_i(x_i) + \ell_{i,k}^\top A_i x_i + c_k(A_i x_{i,k+1} - \tfrac{b}{N})^\top A_i x_i$$

which is equivalent to

$$x_{i,k+1} \in \operatorname*{argmin}_{x_i \in X_i} f_i(x_i) + \ell_{i,k}^\top A_i x_i + \tfrac{c_k}{2}\|A_i x_i - \tfrac{b}{N}\|^2 \tag{8}$$

thanks to the result of [4, Lemma 4] interpreting $c_k(A_i x_{i,k+1} - \frac{b}{N})^\top A_i$ as the gradient of $\frac{c_k}{2}\|A_i x_i - \frac{b}{N}\|^2$ with respect to x_i evaluated at $x_{i,k+1}$. Note that computing $x_{i,k+1}$ using (8) does not require the knowledge of $\lambda_{i,k+1}$ as instead suggested by (6a) and computing $\lambda_{i,k+1}$ from (7) does not require to maximize $\varphi_i(\cdot)$ as instead suggested by (3b).

The resulting distributed strategy is summarized in Algorithm 1 from the perspective of agent i. Note that even though $x_{i,k+1}$ is typically not unique, $A_i x_{i,k+1}$ is unique as the

function to be minimized in (8) is strictly convex in $A_i x_i$ and this ensures that also $\lambda_{i,k+1}$ is unique. Algorithm 1 and the method in (3) thus generate the same $\{\lambda_{i,k}\}_{k\geq 0}$ sequences and are therefore equivalent.

In the next theorem, we state our first main result, which shows that Algorithm 1 converges to an optimal solution of $\mathcal{D}$. The result builds on the work in [4] for DCPs, which cannot be directly applied because it would require the common decision variable λ to belong to a compact set (see [4, Assumption 1]).

Theorem 1 (Dual optimality)**.** *Under Assumptions 1-5, the sequences $\{\lambda_{i,k}\}_{k\geq 0}$, $i = 1, \ldots, N$, generated by Algorithm 1, converge to an optimal solution $\lambda^\star$ of $\mathcal{D}$.* □

Unfortunately, as it is typically the case for duality-based strategies, convergence of the dual variables to an optimal solution of $\mathcal{D}$ does not necessarily mean that the primal tentative solutions $x_{i,k+1}$ computed in Step 5 converge to an optimal primal solution. However, in many applications, one is interested in obtaining an optimal solution $x^\star$ of $\mathcal{P}$ and not only its optimal cost. This is typically the case for finite-horizon optimal control of dynamical systems, where the optimal primal solution coincides with optimal strategy to control the system. To recover primal optimality each agent need to locally construct an auxiliary sequence (cf. Step 7) which appropriately averages the terms of its corresponding $\{x_{i,k}\}_{k\geq 0}$ sequence. The sequence generated by Step 7 can be equivalently obtained as

$$\hat{x}_{i,k+1} = \frac{\sum_{s=0}^{k} c_s x_{i,s+1}}{\sum_{s=0}^{k} c_s}. \tag{9}$$

If we define $\hat{\boldsymbol{x}}_k = [\hat{x}_{1,k}^\top \cdots \hat{x}_{N,k}^\top]^\top$, then we have the following result.

Theorem 2 (Primal optimality)**.** *Under Assumptions 1-5, all limit points of the sequence $\{\hat{\boldsymbol{x}}_k\}_{k\geq 0}$ are optimal solutions for $\mathcal{P}$.* □

The proofs of Theorems 1 and 2 are given in the Appendix.

IV. Numerical Example

The proposed algorithm is tested on a modified version of the plug-in electric vehicles optimal charging schedule problem described in [20]. We consider a fleet of N electric vehicles which need to find a minimum-cost overnight charging strategy while coping with local constraints (target state of charge and battery limits) and network constraints (maximum power that the grid can provide). For simplicity we consider the "only charging" case in which vehicles only draw energy from the network without injecting any, over each time slot of the charging time horizon. The problem can be written in the form of $\mathcal{P}$, using Remark 1. Due to space limitations, we refer the reader to [20], the only difference being that we allow each vehicle to also optimize the charging rate at any time slot instead of deciding only whether to charge or not the battery.

In our simulation we considered a fleet of $N = 50$ vehicles, each one with 24 decision variables representing the charging rate over each 20-minute time slot of the 8-hour charging horizon, and 24 additional slack variables to account for the 24 coupling constraints (see Remark 1). The communication network was taken to be fixed across iterations and was built according to the following randomized procedure. For each possible agent pair, the corresponding edge was included in the graph with probability 0.25, and we kept generating graphs until a strongly connected one was found. The corresponding weight matrix was also taken to be fixed across iterations and was constructed applying the procedure in [21] to the adjacency matrix of the randomly-generated graph, satisfying Assumption 5 with $\eta = 0.03$.

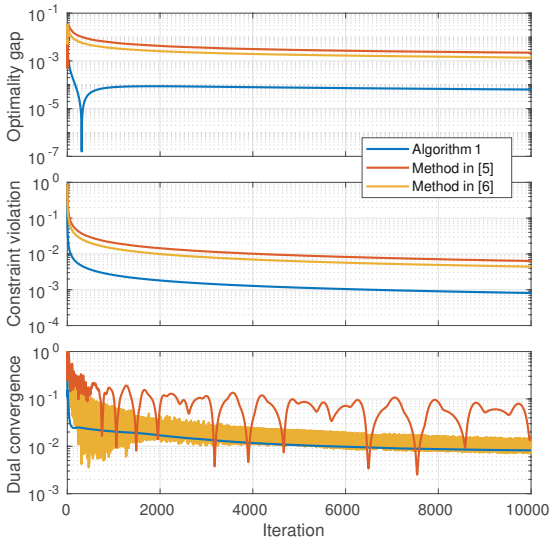

Fig. 1. Relative optimality gap (top plot), relative violation of the joint constraints (middle plot), and relative dual convergence (bottom plot), across iterations. Blue lines: proposed proximal method; red lines: primal-dual method in [5]; yellow lines: subgradient method in [6].

We run Algorithm 1 for 10^4 iterations, setting $c_k = 10^{-3}/(k+1)^{0.51}$. For comparison purposes we also run one algorithm for each category in the same column of Table I: the primal-dual algorithm in [5] and the distributed subgradient method in [6]. In Figure 1 we report, the behavior across iterations of the relative optimality gap between the value of the cost function achieved by the primal tentative solution $\hat{\boldsymbol{x}}_k$ and the optimal cost $f^\star$ computed by a centralized solver (top plot), the relative violation of the joint constraints (middle plot), and the relative dual convergence (bottom plot), measured as

$$\frac{|\sum_{i=1}^N f_i(\hat{x}_{i,k}) - f^\star|}{|f^\star|}, \quad \frac{\|\sum_{i=1}^N A_i \hat{x}_{i,k} - b\|}{\|b\|}, \quad \frac{\|\bar{\lambda}_k - \lambda^\star\|}{\|\lambda^\star\|},$$

with $\bar{\lambda}_k = \frac{1}{N}\sum_{i=1}^N \lambda_{i,k}$. Not surprisingly, the convergence rate of the three methods is similar, as they all make use of the same vanishing step-size sequence. However, the proposed method generates less oscillating iterates for the tentative dual variables (all $\lambda_{i,k}$'s behave similarly to $\bar{\lambda}_k$) compared to [5] and [6], which can make it easier for the agents to check the algorithm progress.

V. Conclusions

In this paper, we have extended the distributed proximal minimization-based approach presented in the literature for decision-coupled problems to constraint-coupled problems in

which the sum of local cost functions has to be minimized subject to both individual constraints and a linear coupling constraint. The proposed distributed algorithm guarantees convergence of the decision variables of each agent to an optimal solution, with a rate that is similar to its distributed subgradient counterpart.

REFERENCES

[1] A. Nedić and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 48–61, 2009.

[2] A. Nedić, A. Ozdaglar, and P. A. Parrilo, "Constrained consensus and optimization in multi-agent networks," *IEEE Transactions on Automatic Control*, vol. 55, no. 4, 2010.

[3] M. Zhu and S. Martínez, "On distributed convex optimization under inequality and equality constraints," *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 151–164, 2012.

[4] K. Margellos, A. Falsone, S. Garatti, and M. Prandini, "Distributed constrained optimization and consensus in uncertain networks via proximal minimization," *IEEE Transactions on Automatic Control*, vol. 63, no. 5, pp. 1372–1387, 2018.

[5] T.-H. Chang, A. Nedić, and A. Scaglione, "Distributed constrained optimization by consensus-based primal-dual perturbation method," *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1524–1538, 2014.

[6] A. Falsone, K. Margellos, S. Garatti, and M. Prandini, "Dual decomposition for multi-agent distributed optimization with coupling constraints," *Automatica*, vol. 84, pp. 149–158, 2017.

[7] I. Notarnicola and G. Notarstefano, "Constraint-coupled distributed optimization: a relaxation and duality approach," *IEEE Transactions on Control of Network Systems*, 2019.

[8] D. P. Bertsekas, "Incremental proximal methods for large scale convex optimization," *Mathematical Programming*, vol. 129, no. 2, pp. 163–195, 2011.

[9] J. F. Mota, J. M. Xavier, P. M. Aguiar, and M. Püschel, "D-ADMM: A communication-efficient distributed algorithm for separable optimization," *IEEE Transactions on Signal Processing*, vol. 61, no. 10, pp. 2718–2723, 2013.

[10] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and distributed computation: numerical methods*. Prentice hall Englewood Cliffs, NJ, 1989, vol. 23.

[11] A. Nedić, A. Olshevsky, and W. Shi, "Achieving geometric convergence for distributed optimization over time-varying graphs," *SIAM Journal on Optimization*, vol. 27, no. 4, pp. 2597–2633, 2017.

[12] G. Qu and N. Li, "Harnessing smoothness to accelerate distributed optimization," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 3, pp. 1245–1260, 2018.

[13] M. Zhu and S. Martínez, "Discrete-time dynamic average consensus," *Automatica*, vol. 46, no. 2, pp. 322–329, 2010.

[14] S. S. Kia, B. Van Scoy, J. Cortes, R. A. Freeman, K. M. Lynch, and S. Martinez, "Tutorial on dynamic average consensus: The problem, its applications, and the algorithms," *IEEE Control Systems Magazine*, vol. 39, no. 3, pp. 40–72, 2019.

[15] S. Liang, L. Y. Wang, and G. Yin, "Distributed smooth convex optimization with coupled constraints," *IEEE Transactions on Automatic Control*, 2019.

[16] T.-H. Chang, "A proximal dual consensus ADMM method for multi-agent constrained optimization," *IEEE Transactions on Signal Processing*, vol. 64, no. 14, pp. 3719–3734, 2016.

[17] A. Falsone, I. Notarnicola, G. Notarstefano, and M. Prandini, "Tracking-ADMM for distributed constraint-coupled optimization," *Automatica*, vol. 117, p. 108962, 2020.

[18] A. Falsone, K. Margellos, and M. Prandini, "A distributed iterative algorithm for multi-agent MILPs: finite-time feasibility and performance characterization," *IEEE Control Systems Letters*, vol. 2, no. 4, pp. 563–568, 2018.

[19] D. P. Bertsekas, *Nonlinear programming*. Athena Scientific, 1999.

[20] R. Vujanic, P. M. Esfahani, P. J. Goulart, S. Mariéthoz, and M. Morari, "A decomposition method for large scale MILPs, with performance guarantees and a power system application," *Automatica*, vol. 67, pp. 144–156, 2016.

[21] R. Sinkhorn and P. Knopp, "Concerning nonnegative matrices and doubly stochastic matrices," *Pacific Journal of Mathematics*, vol. 21, no. 2, pp. 343–348, 1967.

[22] B. T. Polyak, "Introduction to optimization. translations series in mathematics and engineering," *Optimization Software*, 1987.

APPENDIX

Proof of Theorem 1 (Dual Optimality)

We prove Theorem 1 by showing that [4, Thm. 1] holds for the method in (3) even if the λ domain in $\mathcal{D}$ is not compact. The statement of Theorem 1 then follows directly from the equivalence between (3) and Steps 4-6 in Algorithm 1.

We start the proof by checking that all the assumptions of [4] but the compactness in [4, Ass. 1] are satisfied.

To see that $\mathcal{D}$ fits the problem in [4, Eq. (2)] it is sufficient to see that it is equivalent to $\min_{\lambda\in\mathbb{R}^p} \sum_{i=1}^N -\varphi_i(\lambda)$ where all the local sets in [4, Eq. (2)] are given by $\mathbb{R}^p$. By definition of $\varphi_i(\lambda)$ in (2), $\varphi_i(\lambda)$ is the point-wise minimum of affine functions of λ and is therefore concave, meaning that $-\varphi_i(\lambda)$ is convex, which together with the convexity of $\mathbb{R}^p$, satisfies the convexity in [4, Ass. 1]. Also, setting the agents local constraint sets to $\mathbb{R}^p$ trivially satisfies [4, Ass. 3]. Assumptions 3, 5, and 4 are respectively equivalent to [4, Ass. 2, 5, and 4]. The only assumption that is not satisfied in our set-up is the compactness in [4, Ass. 1].

We now revisit the proofs of [4] and show how compactness can be removed in all steps where it is used. Symbols hereafter refer to quantities in [4].

In [4, Sec. V-A] the sequence $\bar{v}(k)$ is considered as a feasible counterpart for the average $v(k)$ of the tentative solutions $x_i(k)$, for each k. Indeed, even if $x_i(k) \in X_i$ for all k, $v(k)$ does not necessarily belong to $\bigcap_{i=1}^m X_i$, whereas $\bar{v}(k)$ does. Since in our set-up X_i are all equal to $\mathbb{R}^p$, then $v(k)$ readily belongs to $\bigcap_{i=1}^m X_i = \mathbb{R}^p$, meaning that we can safely replace $\bar{v}(k)$ with $v(k)$ and set $\mu = 1$ in all derivations without requiring compactness of all X_i's, effectively bypassing [4, Lemma 1].

The proof of [4, Lemma 3] uses compactness to bound the term $\|x_i(0)\|$ in [4, Eq. (63)] and the term $\|e_i(1)\|$ in [4, Eq. (66)]. Since $x_i(0)$ is finite for all $i = 1, \dots, m$, then $z_i(0)$ and $x_i(1)$ computed as in [4, Steps 7 and 8 of Algorithm 1] are also finite, which implies that also $e_i(1) = x_i(1) - z_i(0)$ is finite. The compactness requirement of [4, Lemma 3] can thus be relaxed replacing D in [4, Eq. (66)] with $\max\{\|x_i(0)\|, \|x_i(1)\|, \|z_i(0)\|\} < +\infty$.

The proof of [4, Lemma 5] leverages compactness under [4, Eq. (41)] to exploit Lipschitz continuity of the agents' local cost functions. Here we do not have compactness of the λ domain, but we can exploit compactness of the local constraint sets in our primal problem $\mathcal{P}$ to show Lipschitz continuity of each $\varphi_i(\cdot)$. To see this recall the definition of $\partial\varphi_i(\lambda)$ (cf. (5)) and note that, under Assumption 1, there exists a uniform upper bound L_i for $\|\partial\varphi_i(\lambda)\|$.

In the proof of [4, Prop. 2] compactness is required to use [4, Lemma 3] (which we have shown not to need compactness) and to bound the term $\|x_i(1) - x^\star\|$ in [4, Eq. (43)]. Similarly to our prior discussion, $x_i(1)$ is finite while $x^\star$ is finite due to our Assumption 2. Therefore [4, Prop. 2] also holds without compactness.

The proof of [4, Prop. 3] uses compactness to bound the term $\|x_j(s)\|$ in [4, Eq. (45)] for a *fixed* s. Since [4, Step 8 of Algorithm 1] is always well defined, for any fixed iteration s, $\|x_j(s)\|$ will be finite for all $j = 1, \dots, m$. We can thus take

the $\limsup_{k\to\infty}$ of the line above [4, Eq. (45)] and still get [4, Eq. (46)], thus relaxing the compactness assumption also for this result.

We now prove [4, Thm. 4], which requires compactness, in an alternative way. Consider [4, Eq. (37)]. First rearrange the terms so that $\sum_{i=1}^m \|x_i(k+1)-x^\star\|$ is the only one on the left hand side. Then notice that the terms $-\sum_{i=1}^m \|e_i(k+1)\|^2$ and $2c(k)\sum_{i=1}^m (f_i(x^\star)-f_i(\bar v(k+1)))$ on the right hand side are negative and can be discarded. We are left with

$$\underbrace{\sum_{i=1}^m \|x_i(k+1)-x^\star\|^2}_{u_{k+1}} \le \underbrace{\sum_{i=1}^m \|x_i(k)-x^\star\|^2}_{u_k} + \underbrace{2\bar L c(k)\sum_{i=1}^m \|x_i(k+1)-\bar v(k+1)\|}_{\beta_k}, \quad \text{(A.10)}$$

where $\sum_{k=0}^\infty \beta_k < +\infty$ due to [4, Eq. (47)]. We can then apply [22, Lemma 2 in Sec. 2.2.1] to (A.10) with u_k and β_k as defined in (A.10) and $\alpha_k = 0$, for all k, to get the result of [4, Thm. 4].

Finally, in the proof of [4, Thm. 1], the compactness assumption is used to bound the terms $\|x_i(1)-x^\star\|$, $i=1,\dots,m$, which are however finite owing to $x_i(1)$ and $x^\star$ being finite as per our discussion above, even without requiring compactness. This concludes the proof. □

Proof of Theorem 2 (Primal Optimality)

We start by showing that the limit points of the $\{\hat{\boldsymbol{x}}_k\}_{k\ge0}$ sequence are feasible for the coupling constraint. Consider the average violation of the coupling constraint of the tentative solution $\hat{\boldsymbol{x}}_{k+1}$

$$\begin{aligned}
&\frac{1}{N}\left(\sum_{i=1}^N A_i\hat x_{i,k+1} - b\right)\\
&\overset{\text{(a)}}{=} \frac{1}{N}\sum_{i=1}^N\left(A_i\frac{\sum_{s=0}^k c_s x_{i,s+1}}{\sum_{s=0}^k c_s} - \frac{b}{N}\right)\\
&\overset{\text{(b)}}{=} \frac{1}{N}\sum_{i=1}^N\left(\frac{\sum_{s=0}^k c_s (A_i x_{i,s+1} - \frac{b}{N})}{\sum_{s=0}^k c_s}\right)\\
&\overset{\text{(c)}}{=} \frac{1}{N}\sum_{i=1}^N\left(\frac{\sum_{s=0}^k \lambda_{i,s+1} - \ell_{i,s}}{\sum_{s=0}^k c_s}\right)\\
&\overset{\text{(d)}}{=} \frac{\sum_{s=0}^k\left(\frac{1}{N}\sum_{i=1}^N\lambda_{i,s+1} - \frac{1}{N}\sum_{i=1}^N\ell_{i,s}\right)}{\sum_{s=0}^k c_s}\\
&\overset{\text{(e)}}{=} \frac{\sum_{s=0}^k\left(\bar\lambda_{s+1}-\bar\lambda_s\right)}{\sum_{s=0}^k c_s}\\
&\overset{\text{(f)}}{=} \frac{\bar\lambda_{k+1}-\bar\lambda_0}{\sum_{s=0}^k c_s}
\end{aligned} \quad \text{(A.11)}$$

where in (a) we used the definition of $\hat x_{i,k+1}$ in (9), in (b) we brought A_i and $\frac{b}{N}$ inside the convex combination, in (c) we used Step 6 with s in place of k, in (d) we exchanged the two summations, in (e) the fact that, due to Assumption 5, $\frac{1}{N}\sum_{i=1}^N \ell_{i,s} = \frac{1}{N}\sum_{i=1}^N\lambda_{i,s} = \bar\lambda_s$, and in (f) the fact that the summation is telescopic. By taking the limit for $k\to\infty$ on both sides of (A.11), owing to the fact that $\lambda_{i,k}\to\lambda^\star$ for all $i=1,\dots,N$, and thus also $\bar\lambda_k\to\lambda^\star$, together with $\sum_{k=0}^\infty c_k = +\infty$ thanks to Assumption 4, we get

$$\lim_{k\to\infty}\frac{1}{N}\left(\sum_{i=1}^N A_i\hat x_{i,k+1} - b\right) = 0, \quad \text{(A.12)}$$

meaning that all limit points of $\{\hat{\boldsymbol{x}}_k\}_{k\ge0}$ are feasible for the coupling constraint.

Consider now (6a). By optimality of $x_{i,k+1}$ we have

$$\begin{aligned}
&f_i(x_{i,k+1}) + \lambda_{i,k+1}^\top(A_i x_{i,k+1} - b/N)\\
&\qquad\le f_i(x_i^\star) + \lambda_{i,k+1}^\top(A_i x_i^\star - b/N).
\end{aligned}$$

Adding $\lambda^{\star\top}(A_i x_{i,k+1} - b/N)$ and $\lambda^{\star\top}(A_i x_i^\star - b/N)$ on both sides and rearranging some terms we obtain

$$\begin{aligned}
&f_i(x_{i,k+1}) + \lambda^{\star\top}(A_i x_{i,k+1} - b/N)\\
&\quad\le f_i(x_i^\star) + \lambda^{\star\top}(A_i x_i^\star - b/N)\\
&\qquad + (\lambda^\star - \lambda_{i,k+1})^\top(A_i x_{i,k+1} - b/N)\\
&\qquad + (\lambda_{i,k+1} - \lambda^\star)^\top(A_i x_i^\star - b/N). \quad \text{(A.13)}
\end{aligned}$$

Setting $\boldsymbol{x}_k = [x_{1,k}^\top\ \cdots\ x_{N,k}^\top]^\top$, collecting $(\lambda_{i,k+1}-\lambda^\star)$ in the last two terms, summing (A.13) over $i=1,\dots,N$, and recalling (1), we have

$$\begin{aligned}
&L(\boldsymbol{x}_{k+1},\lambda^\star) \le L(\boldsymbol{x}^\star,\lambda^\star)\\
&\qquad + \underbrace{\sum_{i=1}^N(\lambda_{i,k+1}-\lambda^\star)^\top(A_i x_i^\star - A_i x_{i,k+1})}_{\gamma_{k+1}}, \quad \text{(A.14)}
\end{aligned}$$

where we know that $\lim_{k\to\infty}\gamma_k = 0$ thanks to the fact that $\lim_{k\to\infty}\lambda_{i,k} = \lambda^\star$, for all $i=1,\dots,N$, by Theorem 1 together with $x_{i,k}$ and $x_i^\star$ being bounded under Assumption 1.

By convexity of $L(\cdot,\lambda^\star)$ it holds that

$$\begin{aligned}
L(\hat{\boldsymbol{x}}_{k+1},\lambda^\star) &\le \frac{\sum_{s=0}^k c_s L(\boldsymbol{x}_{k+1},\lambda^\star)}{\sum_{s=0}^k c_s}\\
&\overset{\text{(a)}}{\le} \frac{\sum_{s=0}^k c_s L(\boldsymbol{x}^\star,\lambda^\star)}{\sum_{s=0}^k c_s} + \frac{\sum_{s=0}^k c_s\gamma_{s+1}}{\sum_{s=0}^k c_s}\\
&\overset{\text{(b)}}{=} L(\boldsymbol{x}^\star,\lambda^\star) + \frac{\sum_{s=0}^k c_s\gamma_{s+1}}{\sum_{s=0}^k c_s},
\end{aligned}$$

where in (a) we used (A.14) and in (b) the fact that $L(\boldsymbol{x}^\star,\lambda^\star)$ does not depend on s. Owing to the fact that $\lim_{k\to\infty}\gamma_k = \bar\gamma = 0$ together with $\sum_{k=0}^\infty c_k = +\infty$ by Assumption 4, we have that $\lim_{k\to\infty}\frac{\sum_{s=0}^k c_s\gamma_{s+1}}{\sum_{s=0}^k c_s} = \bar\gamma = 0$ and hence

$$\limsup_{k\to\infty} L(\hat{\boldsymbol{x}}_{k+1},\lambda^\star) \le L(\boldsymbol{x}^\star,\lambda^\star),$$

which, together with $L(\hat{\boldsymbol{x}}_{k+1},\lambda^\star) \ge L(\boldsymbol{x}^\star,\lambda^\star)$, implies $\lim_{k\to\infty}L(\hat{\boldsymbol{x}}_{k+1},\lambda^\star) = L(\boldsymbol{x}^\star,\lambda^\star)$. This shows that all limit points of $\{\hat{\boldsymbol{x}}_k\}_{k\ge0}$ achieve the optimal value of $\mathcal{P}$ and, since they are feasible for the coupling constraint (cf. (A.12)), we can finally conclude that they are optimal solutions of $\mathcal{P}$. □