Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 56-2 (2023) 5919–5924

A novel decentralized approach to large-scale multi-agent MILPs ⋆

Lucrezia Manieri Alessandro Falsone Maria Prandini

Politecnico di Milano, Milano, Italy (email: name.surname@polimi.it)

Abstract: We address the optimal operation of a large-scale multi-agent system where agents have to set their own continuous and/or discrete decision variables so as to jointly minimize the sum of local linear performance indices while satisfying local and global linear constraints. When the number of discrete decision variables is large, solving the resulting Mixed Integer Linear Program becomes computationally demanding, and often impossible in practice. Inspired by some recent methods in the literature, we propose a decentralized iterative scheme that recovers computational tractability by decomposing the dual of the MILP problem into lower-dimensional MILPs, one per agent, and obtains feasibility of the recovered primal solution by introducing a fictitious tightening of the global constraints. The tightening is updated in an adaptive fashion according to an heuristic strategy which allows it to both increase and decrease throughout the iterations, depending on the mismatch between the recovered mixed-integer primal solution and the solution to the relaxed linear problem associated with the current tightening. The procedure is shown to be effective and to outperform state-of-the-art alternative resolution schemes in a benchmark example on optimal charging of a fleet of electric vehicles.

Keywords: MILP; large-scale systems decentralized optimization; multi-agent systems

1. INTRODUCTION

In this paper we address the resolution of large-scale mixed-integer optimization programs where m agents cooperate to jointly minimize the sum of their local cost functions, while accounting for local constraints modeling their operational limitations, and global constraints modeling the usage of some shared resources, which couple their decisions. We consider a framework where the cost functions and the constraints are linear in the decision variables and, thus, the resulting *multi-agent constraint-coupled* mixed-integer linear program (MILP) takes the following form

$$\min_{x_1,\dots,x_m} \quad \sum_{i=1}^{m} c_i^\top x_i \tag{1a}$$

$$\text{subject to:} \quad \sum_{i=1}^{m} A_i x_i \le b \tag{1b}$$

$$x_i \in X_i, \quad i = 1,\dots,m, \tag{1c}$$

where $x_i \in \mathbb{R}^{n_i}$, represents the vector of decision variables of agent i and has $n_{c,i}$ continuous components and $n_{d,i}$ discrete ones, $i = 1,\dots,m$. Each decision x_i is associated with a cost $c_i^\top x_i$ and takes value in a non-empty local feasibility set $X_i = \{x_i \in \mathbb{R}^{n_{c,i}} \times \mathbb{Z}^{n_{d,i}} : D_i x_i \le d_i\}$, defined by a matrix D_i and a vector d_i of appropriate dimensions. The p coupling constraints (1b) are defined by the matrices $A_i \in \mathbb{R}^p \times \mathbb{R}^{n_i}$, $i = 1,\dots,m$, and the vector of resources $b \in \mathbb{R}^p$. All inequalities between vectors has to be intended as component-wise.

Due to their mixed-integer nature, MILPs have an intrinsic complexity that grows exponentially with the number of discrete decision variables. Finding an optimal solution is often computationally prohibitive and even retrieving a feasible solution can be challenging, especially when the size of the problem increases. The multi-agent structure of (1), however, can be exploited to decompose the problem and reduce the computational effort. Some works in the literature such as Vujanic et al. (2016), Falsone et al. (2018, 2019), and Camisa et al. (2021) took this perspective to develop heuristic procedures that are guaranteed to find feasible (but possibly sub-optimal) solutions.

Despite their simplicity, problems in the form of (1) can model the optimal operation of a wide range of systems, including energy systems (La Bella et al. (2021)), buildings (Ioli et al. (2015)), and plug-in electric vehicles (Vujanic et al. (2016)). They can be found in finite-horizon optimal control of aggregates of Mixed Logical Dynamical (MLD) systems (see Bemporad and Morari (1999)) that cooperate to jointly optimize a linear performance index while sharing some resources. When the finite-horizon control problem has to be solved repeatedly, within a model predictive control scheme, then, it is important to timely determine an applicable solution and optimality can be sacrificed for feasibility.

1.1 Background and related literature

The presence of the p coupling constraints hampers the decomposition of (1) into m separate MILPs. In order to recover decomposability, one can introduce a vector $\lambda \in \mathbb{R}^p$ of p non-negative Lagrange multipliers to dualize the constraints (1b) and formulate the dual problem

⋆ This paper is supported by PNRR-PE-AI FAIR project funded by the NextGeneration EU program.

10.1016/j.ifacol.2023.10.616

$$\max_{\lambda\geq 0} -\lambda^\top b + \sum_{i=1}^{m} \min_{x_i\in X_i} (c_i^\top + \lambda^\top A_i)x_i, \tag{2}$$

whose optimal solution $\lambda^\star$ can be used to construct a primal solution $x(\lambda^\star) = [x_1(\lambda^\star)^\top \cdots x_m(\lambda^\star)^\top]^\top$, where each $x_i(\lambda^\star)$ is computed solving the following minimization problem with $\lambda = \lambda^\star$

$$x_i(\lambda) \in \arg\min_{x_i\in X_i}(c_i^\top + \lambda^\top A_i)x_i \qquad i = 1,\ldots,m. \tag{3}$$

Note that despite (1) is non-convex, (2) is convex and can be solved to optimality through a decentralized version of the sub-gradient iterative algorithm (see Section 2.1 in Shor (1985)), where at each iteration the central unit updates the dual variable according to a gradient step

$$\lambda(k+1) = \left[\lambda(k) + \alpha(k)\Big(\sum_{i=1}^{m} A_i x_i(\lambda(k)) - b\Big)\right]_+ \tag{4}$$

whereas agents compute in parallel $x_i(\lambda(k))$, $i = 1,\ldots,m$, by solving the local MILPs in (3) with $\lambda = \lambda(k)$. The operator $[\,\cdot\,]_+$ in (4) denotes the projection of its argument onto the non-negative orthant, and is essential to account for the non-negativity constraints on λ.

If the step-size $\alpha(k)$ is chosen so that $\sum_{k=0}^{\infty}\alpha(k) = \infty$ and $\sum_{k=0}^{\infty}\alpha^2(k) = 0$, the sequence $\lambda(k)$ converges to the optimal solution of (2). However, the primal solution $x(\lambda^\star)$ recovered resorting to this dual decomposition approach may not satisfy (1b), because the violation of the coupling constraints is penalized but not prevented and the standard recovery procedures for the convex case (see Shor (1985)) do not apply in a mixed-integer setting (see, e.g., (Falsone et al., 2019, Appendix)).

In Vujanic et al. (2016), feasibility of the solution obtained via dual decomposition is enforced by introducing a fictitious tightening of the resource vector, as explained next.

Let $\rho \in \mathbb{R}^p$, be a tightening vector $\rho \geq 0$, and consider the following tightened problem

$$\min_{x_1,\ldots,x_m} \quad \sum_{i=1}^{m} c_i^\top x_i \tag{5a}$$

$$\text{subject to:} \quad \sum_{i=1}^{m} A_i x_i \leq b - \rho \tag{5b}$$

$$x_i \in X_i,\ i = 1,\ldots,m \tag{5c}$$

and its dual problem

$$\max_{\lambda\geq 0} -\lambda^\top (b-\rho) + \sum_{i=1}^{m} \min_{x_i\in X_i} \left(c_i^\top + \lambda^\top A_i\right) x_i, \tag{6}$$

obtained again by dualizing the coupling constraints. Denoting with $\lambda_\rho^\star$ the optimal solution of (6), Vujanic et al. (2016) explores the connection between the primal tentative solution $x(\lambda_\rho^\star)$, obtained via (3) with $\lambda = \lambda_\rho^\star$, and the solution x_ρ^{LP} of the following convexified problem

$$\min_{x_1,\ldots,x_m} \quad \sum_{i=1}^{m} c_i^\top x_i \tag{7a}$$

$$\text{subject to:} \quad \sum_{i=1}^{m} A_i x_i \leq b - \rho \tag{7b}$$

$$x_i \in \mathrm{conv}(X_i), \quad i = 1,\ldots,m, \tag{7c}$$

where each mixed-integer feasibility set X_i is replaced with its convex hull $\mathrm{conv}(X_i)$, $i = 1,\ldots,m$. Under the assumption that the sets X_i, $i = 1,\ldots,m$, are bounded we have that the dual problem of (7) coincides with (6). Building upon this observation, in Vujanic et al. (2016) it is shown that the solutions x_ρ^{LP} and $x(\lambda_\rho^\star)$ are closely related (cf. (Vujanic et al., 2016, Theorem 2.4)) and that there exists a (worst-case) value $\tilde{\rho}$ of ρ that can be computed a-priori and such that $x(\lambda_{\tilde{\rho}}^\star)$ satisfies the coupling constraint (1b), $\lambda_{\tilde{\rho}}^\star$ being the optimal solution of (6) with $\rho = \tilde{\rho}$. Moreover, Vujanic et al. (2016) shows that the performance degradation of such solution is limited and worsens as the infinity norm of the tightening vector increases (cf. Theorem 3.3 in Vujanic et al. (2016)).

Inspired by Vujanic et al. (2016), in Falsone et al. (2019) a decentralized iterative resolution scheme is proposed, where the tightening vector ρ is adaptively increased while computing the corresponding $\lambda_\rho^\star$. The scheme is extended to a distributed setting in Falsone et al. (2018). In both cases, the approach is provably less-conservative than the inspiring one in Vujanic et al. (2016) in that the tightening ρ converges to a value $\bar{\rho}$ for which $x(\lambda_{\bar{\rho}}^\star)$ is feasible and such that $\bar{\rho} \leq \tilde{\rho}$, meaning that also the performance bound is no-worse than that of Vujanic et al. (2016).

A different method to compute a feasible solution for (1) is proposed in the recent work by Camisa et al. (2021). It resorts to primal decomposition to recast (1) into a master-sub-problem architecture. In particular, the master problem handles the coupling by assigning a portion of the shared resources to each agent, whilst the m independent sub-problems retrieve the best solutions compatible with such resource allocation. To guarantee feasibility of the retrieved solution for both local and coupling constraints, the shared resource is fictitiously reduced by a quantity that is smaller or equal than the worst-case tightening in Vujanic et al. (2016). Although the method allows to effectively decompose (1), it requires to solve sub-problems involving $\mathrm{conv}(X_i)$, whose description in terms of linear inequalities is, in general, hard to obtain.

1.2 Contribution

In this work, we propose a decentralized iterative scheme to find a feasible solution to (1) that resorts to dual decomposition and resource tightening. Similarly to Vujanic et al. (2016), we employ a fictitious tightening and, similarly to Falsone et al. (2019), the tightening vector is updated in an iterative fashion, but, differently from Falsone et al. (2019), we allow ρ both to increase and to decrease. Finally, differently from Camisa et al. (2021), we do not require a description of $\mathrm{conv}(X_i)$. While we admittedly do not provide feasibility guarantees, the procedure is shown to outperform those in Vujanic et al. (2016); Falsone et al. (2019) on a benchmark example.

1.3 Paper organization

The paper is structured as follows. In Section 2 we introduce a new decentralized scheme to recover a feasible primal solution of (1) via dual decomposition and resource tightening. We provide both a high-level description of the procedure and a more detailed discussion on the algorithm to perform the involved steps. In Section 3 we assess the performance of the proposed algorithm on a benchmark electric vehicle charging problem. Finally, in Section 4 we conclude the paper and discuss possible future research directions.

2. PROPOSED PROCEDURE

We next recall some key elements of the work in Vujanic et al. (2016) to describe the intuition behind the decentralized resolution scheme proposed in this paper.

Fix $\rho \geq 0$ and recall that $\lambda_\rho^\star$ denotes the optimal solution of the dual problem (6).

Let $x(\lambda_\rho^\star) = [x_1(\lambda_\rho^\star)^\top \cdots x_m(\lambda_\rho^\star)^\top]^\top$ be a corresponding primal solution with $x_i(\lambda_\rho^\star), i = 1,\dots,m$, given by (3) with $\lambda = \lambda_\rho^\star$. Recall also that $x_\rho^{\mathrm{LP}} = [x_{1,\rho}^{\mathrm{LP}\top} \cdots x_{m,\rho}^{\mathrm{LP}\top}]^\top$ is the solution of the convexified tightened problem (7).

We can characterize the resource usage of $x(\lambda_\rho^\star)$ as follows

$$\begin{aligned}\sum_{i=1}^m A_i x_i(\lambda_\rho^\star) &= \sum_{i=1}^m A_i x_i(\lambda_\rho^\star) \pm \sum_{i=1}^m A_i x_{i,\rho}^{\mathrm{LP}} \\ &= \sum_{i=1}^m A_i x_{i,\rho}^{\mathrm{LP}} + \sum_{i=1}^m A_i\big(x_i(\lambda_\rho^\star) - x_{i,\rho}^{\mathrm{LP}}\big), \quad (8)\end{aligned}$$

where we added and subtracted the resource usage of x_ρ^{LP} and re-arranged the terms in the expression. By optimality of x_ρ^{LP} for (7), we know that x_ρ^{LP} satisfies (7b), i.e., $\sum_{i=1}^m A_i x_{i,\rho}^{\mathrm{LP}} \leq b - \rho$, which can be used to upper-bound the right hand side of (8) to obtain

$$\sum_{i=1}^m A_i x_i(\lambda_\rho^\star) \leq b - \rho + \sum_{i=1}^m A_i\big(x_i(\lambda_\rho^\star) - x_{i,\rho}^{\mathrm{LP}}\big). \quad (9)$$

We can then enforce satisfaction of (1b) by imposing that

$$b - \rho + \sum_{i=1}^m A_i\big(x_i(\lambda_\rho^\star) - x_{i,\rho}^{\mathrm{LP}}\big) \leq b, \quad (10)$$

ultimately deriving the following sufficient condition on ρ

$$\rho \geq \sum_{i=1}^m A_i\big(x_i(\lambda_\rho^\star) - x_{i,\rho}^{\mathrm{LP}}\big). \quad (11)$$

Computing a ρ satisfying (11) is, however, not trivial. Vujanic et al. (2016) exploits the intimate relationship between $x_i(\lambda_\rho^\star)$ and $x_{i,\rho}^{\mathrm{LP}}$ to show that only p component of the sum in (11) are non-zero. This property together with the fact that $x_i(\lambda_\rho^\star) \in X_i$ and $x_{i,\rho}^{\mathrm{LP}} \in \mathrm{conv}(X_i)$ and that the X_i's are bounded, allows to derive an upper bound for the right hand side of (11) which does not depend on ρ. Specifically, setting the s-th component of ρ equal to

$$[\tilde{\rho}]_s = p \max_{i\in\{1,\dots,m\}} \Big\{ \max_{x_i\in X_i} [A_i]_s\, x_i - \min_{x_i\in X_i} [A_i]_s\, x_i \Big\}, \quad (12)$$

renders (11) trivially satisfied with $\rho = \tilde{\rho}$, provided that $(x_{\tilde{\rho}}^{\mathrm{LP}}, \lambda_{\tilde{\rho}}^\star)$ exist and are unique, see (Vujanic et al., 2016, Theorem 3.1).

Starting from (11), Falsone et al. (2019) proposes to adapt the value of ρ in an iterative fashion, where the value of ρ for the next iteration $t+1$ is determined by the tentative primal solutions explored by the algorithm up to iteration t as

$$[\rho(t+1)]_s = p \max_{i\in\{1,\dots,m\}} \Big\{ \max_{\tau\leq t}[A_i]_s x_i(\tau) - \min_{\tau\leq t}[A_i]_s x_i(\tau) \Big\}, \quad (13)$$

for $s = 1,\dots,p$. In (13), $x_i(t)$ is computed from (3) using a sequence $\lambda(t)$ that is chasing the optimal solution of (6) while ρ is being updated. Since $x_i(t) \in X_i$ for all iterations t, then clearly $[\rho(t+1)]_s \leq [\tilde{\rho}]_s$ for all $s = 1,\dots,p$. Despite the approach by Falsone et al. (2019) is less conservative, it is clear from (13) that $[\rho(t)]_s$ may be oversized if, for example, $A_i x_i(t)$ varies considerably in early iterations and then settles.

In this work, we start from (11) to derive a less conservative iterative update law for the tightening vector ρ. Let $k \geq 0$ denote the iteration index, and let $\rho(k)$ be the value of tightening vector at k. Then, in order to find a value of $\rho \geq 0$ satisfying (11), we propose to compute $\rho(k+1)$ as follows

$$\rho(k+1) = \left[\sum_{i=1}^m A_i\big(x_i(\lambda_{\rho(k)}^\star) - x_{i,\rho(k)}^{\mathrm{LP}}\big)\right]_+ \quad (14)$$

where a projection operator is introduced to prevent components of ρ from becoming negative, thus avoiding a fictitious increase of the shared resource.

Note that $\rho(k)$, $k \geq 0$, is bounded because of the boundedness of the local constraint sets X_i, $i = 1,\dots,m$, and of their convex hulls.

Moreover, since (14) considers exactly the quantities $x_i(\lambda_\rho^\star)$ and $x_{i,\rho}^{\mathrm{LP}}$ appearing in (11) with $\rho = \rho(k)$ (as opposed to their conservative estimates in (12) and (13)), the components of the tightening vector are likely to assume smaller values, thus resulting in a less conservative tightening and feasible primal solutions with a better cost.

From (14) and the definition of $[\cdot]_+ = \max\{0,\cdot\}$ we have

$$\begin{aligned}\rho(k+1) &\geq \sum_{i=1}^m A_i\big(x_i(\lambda_{\rho(k)}^\star) - x_{i,\rho(k)}^{\mathrm{LP}}\big) \\ &\geq \rho(k) + \sum_{i=1}^m A_i x_i(\lambda_{\rho(k)}^\star) - b, \quad (15)\end{aligned}$$

where we used the fact that $-\sum_{i=1}^m A_i x_{i,\rho}^{\mathrm{LP}} \geq \rho(k) - b$ in the second inequality.

Let $v(h) = \sum_{i=1}^m A_i x_i(\lambda_{\rho(h)}^\star) - b$ denote the violation vector. Then, from (15) it follows that

$$\rho(k+1) \geq \rho(0) + \sum_{h=0}^k v(h),$$

and since $\rho(k)$ is upper bounded, then, each component of the violation vector must become negative throughout the iterations for the series to be upper bounded. When $p = 1$, this proves that a feasible solution will eventually be found. As for $p > 1$, proving feasibility requires all components to be jointly negative at the same iteration index k.

Since (3) with $\lambda = \lambda_{\rho(k)}^\star$ typically admits multiple mixed-integer solutions, $x_i(\lambda_{\rho(k)}^\star)$ is likely to oscillate across iterations and hence we do not expect $\rho(k)$ to converge, but rather to settle on a limit cycle. Unfortunately, this

Algorithm 1 Proposed decentralized resolution scheme

1: $\lambda(0) = 0$
2: $\rho(0) = 0$
3: $\tilde{J}^\star = \infty$
4: **for** $k = 0, 1, 2, \dots$ **compute**
5: $\quad (x^{\mathrm{LP}}_{\rho(k)}, \lambda^\star_{\rho(k)})$ primal-dual pair of solutions to (7)
6: $\quad$ **for** $i = 1, \dots, m$ **compute**
7: $\quad\quad x_i(k) \in \arg\min_{x_i \in X_i} (c_i^\top + {\lambda^\star_{\rho(k)}}^\top A_i)\, x_i$
8: $\quad$ **end for**
9: $\quad$ **if** $\sum_{i=1}^m A_i x_i(k) \leq b \wedge \sum_{i=1}^m c_i^\top x_i(k) \leq J^\star$
10: $\quad\quad \tilde{x}^\star \leftarrow [x_1(k)^\top \cdots x_m(k)^\top]^\top$
11: $\quad\quad \tilde{J}^\star \leftarrow \sum_{i=1}^m c_i^\top x_i(k)$
12: $\quad$ **end if**
13: $\quad \rho(k+1) = \left[\sum_{i=1}^m A_i \Big(x_i(k) - x^{\mathrm{LP}}_{i,\rho(k)} \Big) \right]_+$
14: **end for**

feature of the update rule (14), makes its analysis much more challenging, leaving it as a future research effort.

The overall proposed procedure is summarized in Algorithm 1 and described next.

At each iteration, problem (7) and its dual (6) with ρ equal to the current value of the tightening vector $\rho(k)$ are solved to optimality to obtain a primal-dual solution pair $(x^{\mathrm{LP}}_{\rho(k)}, \lambda^\star_{\rho(k)})$ (cf. Step 5). The optimal dual solution is then used to reconstruct a tentative primal solution $x(k)$ having components $x_i(k), i = 1, \dots, m$, computed setting $\lambda = \lambda^\star_{\rho(k)}$ in (3) (cf. Step 7). If the newly explored feasible primal solution $x(k)$ is better than any previous feasible solution found in terms of cost, the new solution is stored in $\tilde{x}^\star$ and its cost in $\tilde{J}^\star$ (cf. Steps 10 and 11). Finally, the tightening vector is updated according to (14) in Step 13.

As already mentioned in Section 1.1, the optimal dual solution $\lambda^\star_{\rho(k)}$ in Step 5 and the associated primal solution $x(\lambda^\star_{\rho(k)})$ can be computed via a decentralized sub-gradient method. Specifically, for $\rho = \rho(k)$ a solution $\lambda^\star_{\rho(k)}$ to (6) can be obtained as the limit of the following iterations

$$\lambda(\kappa) = \left[\lambda(\kappa) + \alpha(\kappa) \Big(\sum_{i=1}^m A_i x_i(\lambda(\kappa)) - b + \rho(k) \Big) \right]_+, \tag{16}$$

as stated in (Shor, 1985, Theorem 2.2). Moreover, it can be shown that since the dual variable $\lambda(\kappa)$ converges to $\lambda^\star_{\rho(k)}$ as $\kappa \to \infty$, then there exists a $\bar{\kappa}$ such that $x_i(\lambda(\kappa))$ is a solution of (3) with $\lambda = \lambda^\star_{\rho(k)}$ for all $\kappa \geq \bar{\kappa}$. To see this, consider the tentative solution $x_i(\kappa)$, $i = 1, \dots, m$ obtained setting $\lambda = \lambda(\kappa)$ in (3). By adding and subtracting ${\lambda^\star_{\rho(k)}}^\top A_i$ to the cost vector, we have that

$$x_i(\kappa) \in \arg\min_{x_i \in X_i} (c_i^\top + {\lambda^\star_{\rho(k)}}^\top A_i + (\lambda(\kappa) - \lambda^\star_{\rho(k)})^\top A_i) x_i. \tag{17}$$

Thus, $x_i(\kappa)$ can be seen as the solution obtained adding a perturbation $\delta(\kappa) = (\lambda(\kappa) - \lambda^\star_{\rho(k)})^\top A_i$ to the cost vector in (3) when $\lambda = \lambda^\star_{\rho(k)}$. Since $\lambda(\kappa) \to \lambda^\star_{\rho(k)}$, the norm of the perturbation $\delta(\kappa)$ tends to 0 as $k \to \infty$. Then there exists a $\bar{\kappa}$ such that $\|\delta(\kappa)\|$ is sufficiently small for all $\kappa \geq \bar{\kappa}$ and we can invoke the result of (Falsone et al., 2019, Lemma 1), which states that the solutions of (17) are also solutions of (3) with $\lambda = \lambda^\star_{\rho(k)}$. This shows how Step 7 can be performed in practice.

As for $x^{\mathrm{LP}}_{\rho(k)}$, the tentative primal solutions explored by the sub-gradient algorithm can be used also to estimate $x^{\mathrm{LP}}_{\rho(k)}$ without obtaining an explicit description of the sets $\mathrm{conv}(X_i)$, $i = 1, \dots, m$. Consider the sequence $\{\hat{x}(\kappa)\}_\kappa$ defined as

$$\hat{x}(h) = \frac{\sum_{\tau=1}^h \alpha(\kappa) x(\kappa)}{\sum_{\kappa=1}^h \alpha(\kappa)}, \tag{18}$$

where $x(\kappa) = [x_1(\kappa)^\top \cdots x_m(\kappa)^\top]^\top$, $\kappa = 1, 2, \dots$, with $x_i(\kappa)^\top$ satisfying (17), and $\alpha(\kappa)$ is the step-size in (16). Then, it can be shown that each accumulation point of the sequence $\hat{x}(h)$ is a solution to (7) (we refer the reader to (Shor, 1985, p. 117-118) for a proof). Thus, the solution $x^{\mathrm{LP}}_{\rho(k)}$ in Step 5 can be computed from $\hat{x}(h)$, letting $h \to \infty$.

Algorithm 2 translates the above discussion into pseudocode for the practical implementation of Steps 5-8 of Algorithm 1. In particular, Steps 3-7 of Algorithm 2 apply the sub-gradient method to (6) for the current value of the tightening vector $\rho(k)$, effectively implementing (16) and (3) with $\lambda = \lambda(\kappa)$. After the dual variable λ has achieved convergence for w consecutive iteration, the procedure keeps performing the primal and dual update (cf. Steps 4 and 7) and starts computing the sequence $\hat{x} = [\hat{x}_1^\top \cdots \hat{x}_m^\top]^\top$ with $\hat{x}_i$ defined in (18) (cf. Step 10). When also $\hat{x}$ has reached convergence for w consecutive iterations, the procedure selects the most recent value of $\hat{x}$ as the estimate of x^{LP}_ρ (cf. Step 14). All the $x_i(\kappa)$, $i = 1, \dots, m$, explored in Step 4 after $\lambda(\kappa)$ has converged satisfy (17). Since Algorithm 1 in Step 7 needs only a single $x_i(\lambda^\star_{\rho(k)})$ for all $i = 1, \dots, m$, we can select one among those satisfying (17) by means of any tie-break rule. In Algorithm 2, among those $x_i(\kappa)$ explored after convergence of λ, we select the one that minimizes the cost (cf. Steps 16 and 18), for all $i = 1, \dots, m$.

Both estimates $\hat{x}^{\mathrm{LP}}_\rho$, and $x(\lambda^\star_\rho)$ are returned in Step 19.

3. NUMERICAL EXAMPLE

In this section we assess the performance of the proposed decentralized resolution scheme on the Plug-in Electric Vehicles charging problem described in Vujanic et al. (2016). The problem consists in computing the optimal charging pattern of a fleet of m vehicles that must be charged overnight to reach a user-defined state of charge of the battery by the morning after. The schedule is formulated over a time horizon of 8 hours, divided into $N = 24$ time-slots.

We consider a Vehicle-to-Grid (V2G) setup, where each vehicle i, $i = 1, \dots, m$, can either draw or inject power from/into the grid at a constant rate P_i, in order to, respectively, charge or discharge its internal battery. We assume that the power exchanged between the fleet and the grid within each time-slot is limited by the network capacity, that does not allow all vehicles to charge or

Algorithm 2 Strategy to compute x_ρ^{LP} and $x(\lambda_\rho^\star)$

Input: ρ

$h = 1$
for $\kappa = 1, 2, \ldots$ **do**
  for $i = 1$ to m **do**
    $x_i(\kappa) \leftarrow \arg\min_{x_i \in X_i} (c_i^\top + \lambda(\kappa)^\top A_i) x_i$
  end for
  $\mu(\kappa) = \sum_{i=1}^m A_i x_i(\kappa) - b$
  $\lambda(\kappa+1) = [\lambda(\kappa) + \alpha(\kappa)(\mu(\kappa) + \rho)]_+$
  if λ at convergence for more than w iterations
    for $i = 1$ to m **do**
      $\hat{x}_i(h) = \sum_{\tau=\kappa-h}^{\kappa} \alpha(\tau)\, x_i(\tau) / \sum_{\tau=\kappa-h}^{\kappa} \alpha(\tau)$
    end for
    $\hat{x}(h) = [\hat{x}_1^\top(h) \cdots \hat{x}_m^\top(h)]^\top$
    if $\hat{x}$ at convergence for w iterations
      $\hat{x}_\rho^{\mathrm{LP}} = \hat{x}(\kappa)$
      for $i = 1, \ldots, m$ **select**
        $x_i(\lambda_\rho^\star) \in \arg\min_{\{x_i(\tau): \tau \geq \kappa - h\}} c_i^\top x_i(\tau)$
      end for
      $x(\lambda_\rho^\star) = [x_1(\lambda_\rho^\star)^\top \cdots x_m(\lambda_\rho^\star)^\top]^\top$
      return $\hat{x}_\rho^{\mathrm{LP}}$, $x(\lambda_\rho^\star)$
    end if
    $h \leftarrow h + 1$
  end if
  $\kappa \leftarrow \kappa + 1$
end for

discharge their batteries at the same time. The fleet has, thus, to be coordinated by an aggregator that must compute the optimal charging schedule of each vehicle taking into account local requirements and operational limitations (e.g. battery capacity, logical conditions, user-defined state of charge) as well as global network capacity constraints that create a coupling between the charging patterns.

We compare Algorithm 1 to the decentralized procedures proposed in Falsone et al. (2019) and Vujanic et al. (2016), measuring the level of conservativeness and the performance improvement based on, respectively, the infinity norm of the tightening and the cost attained by the computed primal solution. To make the assessment independent from the parameters of the problem, we consider 100 different fleets of vehicles extracted independently from the standard uniform distribution for a given m. We then show how the approaches scale with the number of vehicles on a given problem instance.

The decentralized algorithms are implemented in MATLAB R2020b and local MILPs are solved using CPLEX v12.10. Simulations are performed on a laptop equipped with an Intel Core i7-9750HF CPU @2.60GHz and 16GB of RAM.

Let ρ_A be the tightening vector associated to the solution computed by Algorithm 1, and denote with $\bar{\rho}$ and $\tilde{\rho}$ the convergence values of the tightening vectors computed by the procedure in Falsone et al. (2019) and Vujanic et al.

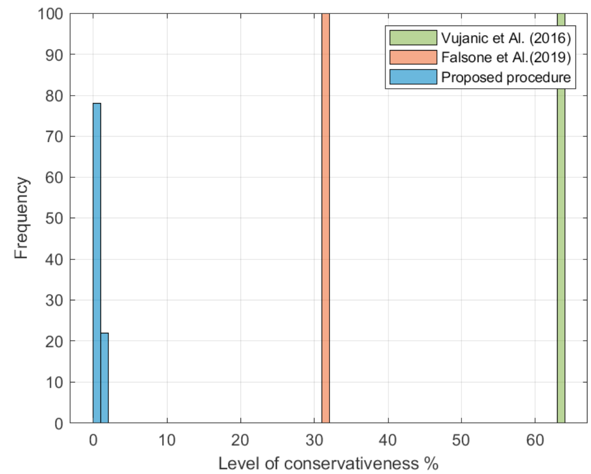

Fig. 1. Histogram of the level of conservativeness achieved by Algorithm 1(blue), and the procedures in Falsone et al. (2019) (orange) and Vujanic et al. (2016) over 100 runs with fixed number of vehicles $m = 250$.

(2016), respectively. Then, we can measure the level of conservativeness of each method through the indices

$$\Delta\rho_{A,\%} = \frac{\|\rho_A\|_\infty}{\|b\|_\infty} \cdot 100 \tag{19}$$

$$\Delta\rho_{F,\%} = \frac{\|\bar{\rho}\|_\infty}{\|b\|_\infty} \cdot 100 \tag{20}$$

$$\Delta\rho_{V,\%} = \frac{\|\tilde{\rho}\|_\infty}{\|b\|_\infty} \cdot 100, \tag{21}$$

where the operator $\|\cdot\|_\infty$ denotes the infinity norm of its argument and b is the right-hand side of the coupling constraints (1b). Figure 1 shows the distribution of $\Delta\rho_{A,\%}$ (blue), $\Delta\rho_{F,\%}$ (orange) and $\Delta\rho_{V,\%}$ (green) over the 100 fleets with different realizations of the vehicles parameters and $m = 250$ vehicles. The level of conservativeness achieved by Algorithm 1 is concentrated between 0% and 1.85% and is significantly smaller than the ones of the procedures in Falsone et al. (2019) and Vujanic et al. (2016), suggesting that Algorithm 1 is less conservative and, thus, more likely to compute solutions attaining smaller costs.

Let J_A, J_F and J_V the values of the objective function achieved applying, respectively, Algorithm 1, the resolution scheme in Falsone et al. (2019) and the one in Vujanic et al. (2016). Let also J_D be the lower-bound on the optimal cost $J^\star$ of (1) obtained solving the dual problem (2). Then, the performance of the three resolution schemes can be evaluated based on the sub-optimality level of the retrieved primal solutions, quantified by the following relative optimality gaps:

$$\Delta J_{V,\%} = \frac{J_V - J_D}{\|J_D\|} \cdot 100, \tag{22a}$$

$$\Delta J_{F,\%} = \frac{J_F - J_D}{\|J_D\|} \cdot 100, \tag{22b}$$

$$\Delta J_{A,\%} = \frac{J_A - J_D}{\|J_D\|} \cdot 100, \tag{22c}$$

where the dual optimal cost J_D is used in place of the optimal cost of the centralized problem $J^\star$, which was not

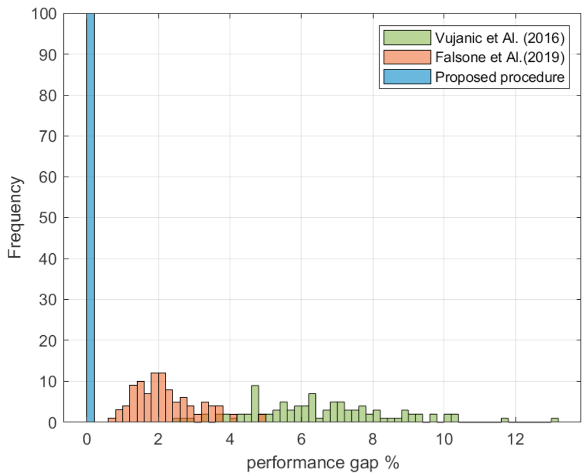

Fig. 2. Histograms of the relative performance gap attained by the primal solutions computed by Algorithm 1 and the procedure in Falsone et al. (2019) (orange) and Vujanic et al. (2016) (green).

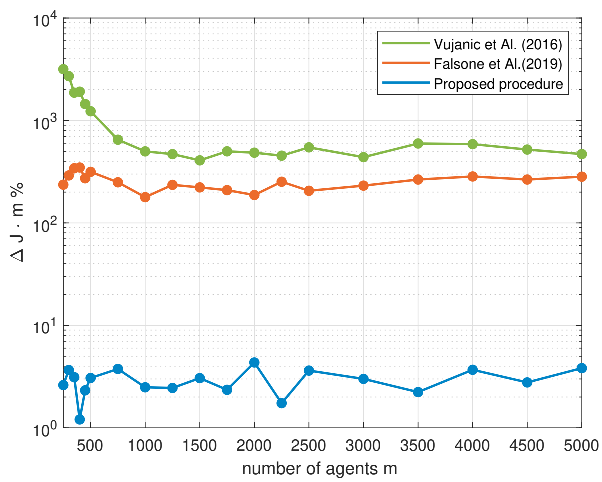

Fig. 3. Rescaled performance gap obtained by Algorithm 1 (blue) and the procedures in Falsone et al. (2019) (orange) and Vujanic et al. (2016) (green) for fleet of vehicles of increasing size.

possible to compute for a fleet of $m > 15$ vehicles given our computational resources.

Figure 2 shows the distribution of $\Delta J_{A,\%}$ (in blue), $\Delta J_{F,\%}$ (in orange) and $\Delta J_{V,\%}$ (in green) over 100 instances with a fixed number $m = 250$ of vehicles and different realizations of the parameters of the fleet. The plot shows a net improvement in the performance obtained by Algorithm 1, that exhibits significantly smaller gaps than its competitors over all runs. In particular, Algorithm 1 obtains optimality gaps concentrated around 0.016%, thus associated with near-optimal primal solutions, whereas the procedures in Falsone et al. (2019) and Vujanic et al. (2016) attain an average gap of 2.220% and 6.525%, respectively.

Figure 3 reports the results obtained for fleets of increasing dimensions, with a number of vehicles m varying from 250 to 5000. Since the values of $\Delta J_{A,\%}$, $\Delta J_{F,\%}$ and $\Delta J_{V,\%}$ tend to 0 as m tends to infinity (see the remark after (Vujanic et al., 2016, Theorem 3.3)), the values reported in the plot are rescaled by a factor m to have a more informative comparison. The graph shows that Algorithm 1 achieves better performance than its competitors irrespective of m.

4. CONCLUSIONS AND FUTURE WORKS

In this work, we proposed a decentralized resolution scheme to compute a feasible solution of large-scale constraint-coupled MILPs. The problem is decomposed in m smaller instances via dual decomposition to recover computational tractability. Feasibility of the obtained primal solution is enforced introducing a fictitious tightening of the shared resource. Extensive simulations on a benchmark electric vehicle charging problem show that the proposed approach outperforms existing state-of-the art schemes in that the obtained primal solutions attain a cost that is closer to the optimal cost of the centralized problem, irrespective of the problem parameters and the number of agents.

Although the adopted update rule on the tightening vector enforces feasibility of each coupling constraint separately, providing theoretical guarantees that the proposed algorithm provides a feasible solution is still an open problem and an interesting future research direction. Once feasibility of one of the explored solutions will be proven, performance guarantees can be derived following the derivations in Vujanic et al. (2016) based on the value of the tightening vector associated with the feasible solution.

REFERENCES

Bemporad, A. and Morari, M. (1999). Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3), 407–427.

Camisa, A., Notarnicola, I., and Notarstefano, G. (2021). Distributed primal decomposition for large-scale MILPs. *IEEE Transactions on Automatic Control*, 1–1.

Falsone, A., Margellos, K., and Prandini, M. (2018). A distributed iterative algorithm for multi-agent MILPs: finite-time feasibility and performance characterization. *IEEE Control Systems Letters*, 2(4), 563–568.

Falsone, A., Margellos, K., and Prandini, M. (2019). A decentralized approach to multi-agent MILPs: finite-time feasibility and performance guarantees. *Automatica*, 103, 141–150.

Ioli, D., Falsone, A., and Prandini, M. (2015). An iterative scheme to hierarchically structured optimal energy management of a microgrid. In *2015 54th IEEE Conference on Decision and Control (CDC)*, 5227–5232. IEEE.

La Bella, A., Falsone, A., Ioli, D., Prandini, M., and Scattolini, R. (2021). A mixed-integer distributed approach to prosumers aggregation for providing balancing services. *International Journal of Electrical Power & Energy Systems*, 133, 107228.

Shor, N.Z. (1985). *Minimization methods for non-differentiable functions*. Springer.

Vujanic, R., Mohajerin Esfahani, P., Goulart, P.J., Mariéthoz, S., and Morari, M. (2016). A decomposition method for large scale MILPs, with performance guarantees and a power system application. *Automatica*, 67, 144–156.