Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 53-2 (2020) 2684–2689

A distributed optimization algorithm for Nash bargaining in multi-agent systems ⋆

Andrea Camisa * Philipp N. Köhler ** Matthias A. Müller *** Giuseppe Notarstefano * Frank Allgöwer **

* Department of Electrical, Electronic and Information Engineering, Alma Mater Studiorum Università di Bologna, Bologna, Italy
** Institute for Systems Theory and Automatic Control, University of Stuttgart, Stuttgart, Germany.
*** Institute of Automatic Control, Leibniz University Hannover, Hannover, Germany.

Abstract: In this paper, we consider a multi-objective optimization problem over networks in which agents aim to maximize their own objective function, while satisfying both local and coupling constraints. This set up includes, e.g., the computation of optimal steady states in multi-agent control systems. Since *fairness* is a key feature required for the solution, we resort to Cooperative Game Theory and search for the Nash bargaining solution among all the efficient (or Pareto optimal) points of a bargaining game. We propose a negotiation mechanism among the agents to compute such a solution in a distributed way. The problem is reformulated as the maximization of a properly weighted sum of the objective functions. The proposed algorithm is then a two step procedure in which local estimates of the Nash bargaining weights are updated online and existing distributed optimization algorithms are applied. The proposed method is formally analyzed for a particular case, while numerical simulations are provided to corroborate the theoretical findings and to demonstrate its efficacy.

Keywords: Cooperative game theory; Distributed optimization; Multi-Objective optimization; Nash bargaining; Distributed model predictive control; Network games.

1. INTRODUCTION

Distributed control of large-scale systems has become a ubiquitous requirement in many trending applications such as smart electric grids, self-organized factories or autonomous vehicles. The control goal in many of these applications is not mere setpoint stabilization, but rather optimal operation of the system with respect to certain real, economic cost criteria such as profit or operating costs. In cooperative multi-agent applications, the subsystems can be interested in determining the best steady-state operating point. However, the overall performance criterion is usually simply taken as the unweighted sum of the local performance criteria, which ignores the multi-objective nature of the original problem and can lead to very poor performance for some systems, while other systems are treated preferentially. Thus, such a solution will in general not be "fair". The computation of fair optimal steady states plays an important role in optimization-based distributed control techniques such as distributed economic model predictive control (MPC), see Christofides et al. (2013), Maestre and Negenborn (2013), Müller and Allgöwer (2017) for an overview.

The contributions of this paper are as follows. Starting from the main motivation, namely distributed computation of fair optimal steady-states for multi-agent control systems, we propose a distributed algorithm to solve general convex constraint-coupled multi-objective optimization problems. Among all the efficient (or Pareto optimal) solutions, the algorithm is designed to yield a fair solution. The proposed approach can be subsequently employed in existing distributed economic MPC schemes, such as Köhler et al. (2018), to obtain fair economically optimal closed-loop system operation. The notion of fairness is well defined in the context of Cooperative Game Theory (Nash, 1950), and in particular, in this work we regard the Nash bargaining solution as the fair solution. The setup under consideration consists of a group of agents with local objectives, depending on local optimization variables, that are intertwined by means of convex coupling constraints. The problem of finding a Pareto optimal solution of the multi-objective problem in a distributed way amounts to solving a constraint-coupled optimization problem, which has been investigated in the literature, see, e.g., Bürger et al. (2014), Notarstefano et al. (2019), Falsone et al. (2017), Notarnicola and Notarstefano (2019). In order to obtain the Nash bargaining solution, the approach proposed in this paper consists of the repeated solution of constraint-coupled problems over the network, while in

⋆ This result is part of a project that has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 638992 - OPT4SMART). P. Köhler, M. Müller, and F. Allgöwer thank the German Research Foundation (DFG) for support of this work within grant AL 316/11-2 and within the German Excellence Strategy under grant EXC-2075.

10.1016/j.ifacol.2020.12.402

an iterative fashion adapting the weighting of the local objective functions in such a way that asymptotically the Nash Bargaining solution is obtained. To the best of our knowledge, no such cooperative distributed algorithm has been proposed in the literature yet, whereas there exist numerous distributed algorithms for the computation of Nash equilibria in *non*-cooperative network games, see, e.g., Grammatico et al. (2016), Salehisadaghiani and Pavel (2016), Liang et al. (2017). Since Pareto optimality is in general not reached by Nash equilibria, cooperation among the agents is incentivized. Moreover, due to the presence of coupling constraints among the agents, cooperation is in fact necessary to obtain a feasible solution.

The paper is organized as follows. In Section 2 we introduce the main motivation together with the general problem setup. In Section 3, we introduce the needed game theoretical tools. In Section 4 we describe our proposed distributed algorithm and provide some initial analysis. In Section 5, we demonstrate its efficacy with numerical simulations. Due to space reasons, the theoretical proofs are omitted and will be provided in a forthcoming document.

2. MOTIVATING APPLICATION AND PROBLEM STATEMENT

In this section, we first present our overarching motivation, which is finding fair (economically) optimal steady states for a network of constraint-coupled linear systems. We then generalize this setup to arrive at a more general optimization problem, which will be studied throughout this work.

2.1 Multi-objective Optimal Steady States

Consider $N \in \mathbb{N}$ linear time-invariant systems of the form

$$z_i(t+1) = A_i z_i(t) + B_i u_i(t),$$

where, for all $i \in \{1, \dots, N\}$, $z_i(t) \in \mathbb{R}^{p_i}$ is the system's state at time t, $u_i(t) \in \mathbb{R}^{q_i}$ is the input to the system at time t, and the matrices A_i, B_i are of appropriate dimensions. Assume each system i is subject to local constraints on the state and input of the form $z_i(t) \in Z_i$, $u_i(t) \in U_i$, for all $t \geq 0$, where $Z_i \subseteq \mathbb{R}^{p_i}$ and $U_i \subseteq \mathbb{R}^{q_i}$. A vector $(z_i^{\mathrm{e}}, u_i^{\mathrm{e}})$ is a feasible steady state for system i if it satisfies

$$z_i^{\mathrm{e}} = A_i z_i^{\mathrm{e}} + B_i u_i^{\mathrm{e}} \quad \text{and} \quad z_i^{\mathrm{e}} \in Z_i, \ u_i^{\mathrm{e}} \in U_i.$$

In multi-agent scenarios, a common requirement is that the steady states must further satisfy coupling constraints, which can be written in the form $\sum_{i=1}^N G_i(z_i^{\mathrm{e}}, u_i^{\mathrm{e}}) \leq b$, where $b \in \mathbb{R}^m$ and each $G_i : \mathbb{R}^{p_i+q_i} \to \mathbb{R}^m$ (assumed to be convex) is used to express the i-th contribution to the coupling constraints; inequality signs are meant component-wise for vectors throughout the paper. Such coupling constraints appear in various applications as, e.g., when multiple systems share a common resource, or when mobile robots need to stay close to maintain connectivity.

Typically, there exist multiple feasible steady states satisfying the coupling constraints. Therefore, we suppose that each agent i is equipped with a utility function $\ell_i : \mathbb{R}^{p_i+q_i} \to \mathbb{R}$ that expresses the agent's individual preference. To determine an optimal overall steady state configuration the following *multi-objective* optimization problem needs to be solved

$$\begin{aligned}
\max_{\substack{z_1,\dots,z_N \\ u_1,\dots,u_N}} \quad & \big[\ell_1(z_1,u_1), \ \dots, \ \ell_N(z_N,u_N)\big] \\
\text{subj. to} \quad & z_i \in Z_i, \ u_i \in U_i, \qquad i \in \{1,\dots,N\}, \\
& z_i = A_i z_i + B_i u_i, \qquad i \in \{1,\dots,N\}, \\
& \sum_{i=1}^N G_i(z_i,u_i) \leq b,
\end{aligned} \tag{1}$$

where the vector-valued utility function means that the objective is to maximize all the components *simultaneously* and not, for instance, in lexicographic order.

2.2 General Setup

In the remainder of the paper, we focus on a more general problem formulation, which contains our motivating example in Problem (1) as a special case. To this end, consider a network of N agents that aim to cooperatively solve the multi-objective optimization problem

$$\begin{aligned}
\max_{x_1,\dots,x_N} \quad & \big[f_1(x_1), \ \dots, \ f_N(x_N)\big] \\
\text{subj. to} \quad & h_i(x_i) \leq 0, \qquad i \in \{1,\dots,N\}, \\
& \sum_{i=1}^N g_i(x_i) \leq 0,
\end{aligned} \tag{2}$$

where, for all $i \in \{1, \dots, N\}$, $x_i \in \mathbb{R}^{n_i}$ is agent i's component of the optimization variable, the function $h_i : \mathbb{R}^{n_i} \to \mathbb{R}^{r_i}$ expresses local constraints for x_i, and $f_i : \mathbb{R}^{n_i} \to \mathbb{R}$ is the i-th utility function. Moreover, the variables are intertwined by means of $m \in \mathbb{N}$ coupling constraints $\sum_{i=1}^N g_i(x_i) \leq 0$, where each $g_i : \mathbb{R}^{n_i} \to \mathbb{R}^m$ is used to model the contribution of x_i to the coupling constraints. We use the symbols $0, \mathbf{1}$ to denote the vectors of zeroes and ones, respectively. Note that Problem (1) fits in the general formulation (2) by defining x_i as the stack of z_i, u_i and by appropriately defining f_i, h_i and g_i. Note also that separable costs are not restrictive, since one can make copies of common optimization variables (if present) and add consistency-preserving coupling constraints.

We assume that each agent i knows only its local constraint function h_i, its local utility function f_i, and its own contribution g_i to the coupling. We make the following standing assumption.

Assumption 2.1. The feasible set of Problem (2) is non-empty and compact. Moreover, for all $i \in \{1, \dots, N\}$, *(i)* the function f_i is concave and continuously differentiable, *(ii)* the functions h_i and g_i are convex and continuously differentiable. □

In distributed optimization (with standard scalar cost function), agents are interested in computing an optimal solution to the overall problem. However, since we deal with multi-objective optimization, additional elaboration on the concept of desired optimality is in order.

2.3 Pareto Optimality

For a multi-objective optimization problem, it may be possible to compute a solution that is optimal for *all* objective function components, however this is an uncommon situation that is seldom found. A far more interesting and realistic case occurs when a high degree of optimality for one component corresponds to a low degree of optimality for the others. In these cases, one is typically interested in computing a so-called *Pareto optimal* solution, the formal definition of which is as follows.

Definition 2.2. A feasible vector $(x_1^\star, \dots, x_N^\star)$ for Problem (2) is Pareto optimal if there does not exist another feasible vector $(x_1', \dots, x_N')$ such that $f_i(x_i') \geq f_i(x_i^\star)$ for all $i \in \{1, \dots, N\}$ and $f_j(x_j') > f_j(x_j^\star)$ for at least one $j \in \{1, \dots, N\}$. □

Figuratively, a Pareto optimal solution "saturates" optimality in the sense that an improvement over one component of the objective function can only be obtained by worsening (at least) one other component. In order to practically compute Pareto optimal solutions of Problem (2), several techniques exist in the literature, see, e.g., Miettinen (2012). Here we employ the so-called *weighting method*, which is amenable for distributed computation. The original multi-objective problem (2) is transformed into a (standard) optimization problem with scalar objective function, i.e.,

$$
\begin{aligned}
\max_{x_1,\dots,x_N} \quad & \sum_{i=1}^{N} w_i f_i(x_i) \\
\text{subj. to} \quad & h_i(x_i) \leq 0, \qquad i \in \{1, \dots, N\}, \\
& \sum_{i=1}^{N} g_i(x_i) \leq 0,
\end{aligned}
\tag{3}
$$

where each $w_i \geq 0$ is a weight assigned to the i-th component of the overall objective function. Under the convexity Assumption 2.1, it can be shown that any Pareto optimal solution of Problem (2) corresponds to a set of weights for Problem (3), as detailed in the next lemma.

Lemma 2.3. (Miettinen (2012), Thm 3.1.4). Let Assumption 2.1 hold and let $(x_1^\star, \dots, x_N^\star)$ denote a Pareto optimal solution of Problem (2). Then, there exist $w_i \geq 0$, for $i \in \{1, \dots, N\}$, such that $(x_1^\star, \dots, x_N^\star)$ is an optimal solution of Problem (3). □

For preassigned, fixed weights w_i, Problem (3) may be solved cooperatively in a distributed way using an existing algorithm such as, e.g., Bürger et al. (2014), Falsone et al. (2017), Notarnicola and Notarstefano (2019). Clearly, depending on the Pareto optimal solution, each component of the objective function is weighted differently from the others. A natural question arising at this point is whether it is possible to find a *fair* Pareto optimal solution and, consequently, how to choose the weights w_i accordingly. Hence, the goal of this work is to develop a distributed algorithm that provides a fair set of the weights in Problem (3) online, where the notion of fairness is detailed in the following.

3. COOPERATIVE GAME THEORY

In this section we provide some background on cooperative game theory. We recall the definition of *bargaining problem* and we formulate a game associated to the multi-objective problem (2). Then, we recall the definition of Nash bargaining solution along with its main properties.

3.1 Bargaining Problems

Bargaining problems were first introduced by Nash (1950). A bargaining problem is a game in which several players (or agents) aim to share a surplus that they are able to generate through cooperation. The goal of the game is that agents agree on how the surplus should be split. Formally, an N-player bargaining game is defined by a tuple $(\mathcal{Y}, d)$, where $\mathcal{Y} \subseteq \mathbb{R}^N$ is the *feasible set* and $d \in \mathbb{R}^N$ is the *disagreement point* (Thomson, 1994). The feasible set represents all the possible outcomes of the game, while the disagreement point is a "fallback solution", which is selected if negotiation breaks down. The solution of the bargaining game is a vector $y^\star \in \mathcal{Y}$ on which agents unanimously agree (it can also be $y^\star = d$). Clearly, each agent i is interested in making its $y_i^\star$ as large as possible. In order to apply cooperative game theory results, the following assumption is made.

Assumption 3.1. We assume that *(i)* $d \in \mathcal{Y}$, *(ii)* $\mathcal{Y}$ is d-comprehensive: if $y \in \mathcal{Y}$ and $y \geq y' \geq d$, then $y' \in \mathcal{Y}$, and *(iii)* there exists $y \in \mathcal{Y}$ such that $y > d$. □

Let us briefly comment on this assumption. Item *(i)* requires the disagreement to be a possible outcome of the game, *(ii)* can be interpreted as the freedom of each agent to decrease its utility, while *(iii)* means that all the agents can benefit from cooperation.

Let us now connect the formalism of cooperative game theory with the definitions of the previous section by posing a bargaining problem associated to Problem (2). We define the feasible set $\mathcal{Y}$ of the game as the image of the feasible set of Problem (2) through the vector function $F(x) = [f_1(x_1), \dots, f_N(x_N)]$, i.e.,

$$\mathcal{Y} \triangleq F(\mathcal{X}) = \{F(x) \mid x \in \mathcal{X}\},$$

where $\mathcal{X}$ is the feasible set of Problem (2), i.e.,

$$\mathcal{X} \triangleq \Big\{(x_1, \dots, x_N) \mid h_i(x_i) \leq 0 \; \forall\, i \text{ and } \sum_{i=1}^{N} g_i(x_i) \leq 0\Big\}.$$

Therefore, $\mathcal{Y}$ represents the set of all the possible *utility combinations* that correspond to (at least) one feasible solution of problem (2). In such a way, all the feasible vectors of Problem (2) are represented by a point in $\mathcal{Y}$, which in turn represents how much each agent can benefit from cooperation.

Let us briefly comment on the definition of the disagreement point d. We suppose that the agents are equipped with a feasible vector $x^{\mathrm{d}} = (x_1^{\mathrm{d}}, \dots, x_N^{\mathrm{d}}) \in \mathcal{X}$ such that each agent i only knows x_i^{d} and $d_i = f_i(x_i^{\mathrm{d}})$ for all $i \in \{1, \dots, N\}$. In principle, we assume that the agents already know their x_i^{d}, because it represents the starting point of the game. For instance, it could be provided to the agents by a centralized authority regulating the game, or it could be computed by the agents as the Nash equilibrium of the underlying strategic game. In general, the way of how x^{d} should be computed is application specific.

3.2 Nash Bargaining Solution

There exist several notions of solutions to bargaining problems, see, e.g., Thomson (1994). Here, we focus on the *Nash bargaining solution*, the definition and properties of which are now recalled.

The Nash bargaining solution $(y_1^\star, \ldots, y_N^\star)$ is the unique vector maximizing $\prod_{i=1}^N (y_i - d_i)$ while satisfying $(y_1, \ldots, y_N) \in \mathcal{Y}$ and $y \geq d$. The significance of the Nash bargaining solution lies in the fact that it enjoys desirable properties, which are referred to as *axioms* (see also Thomson (1994)). Among them, we mention that *(i)* it satisfies Pareto optimality, *(ii)* all agents gain from cooperation, i.e., $y_i^\star > d_i$ for all $i \in \{1, \ldots, N\}$, and *(iii)* the solution is scale invariant, in the sense that any affine transformation applied to the set $\mathcal{Y}$ and to the disagreement point d results in the same transformation applied to the Nash bargaining solution $(y_1^\star, \ldots, y_N^\star)$. Note that the latter property is of particular significance in our setup. Indeed, this means that the Nash bargaining solution associated to Problem (2) takes into account the scale used by each agent to represent its utility.

We now comment on the computational aspect. To compute the nash bargaining solution, let us consider the problem

$$\begin{aligned} \max_{x_1,\ldots,x_N} \ & \prod_{i=1}^N (f_i(x_i) - d_i) \\ \text{subj. to } & h_i(x_i) \leq 0, \qquad i \in \{1, \ldots, N\}, \\ & f_i(x_i) \geq d_i, \qquad i \in \{1, \ldots, N\}, \\ & \sum_{i=1}^N g_i(x_i) \leq 0. \end{aligned} \tag{4}$$

Then, if $(x_1^\star, \ldots, x_N^\star)$ denotes an optimal solution of Problem (4), it follows that

$$y_i^\star = f_i(x_i^\star), \qquad \text{for all } i \in \{1, \ldots, N\}.$$

In order to compute the Nash bargaining solution in a distributed way, it would be desirable to use distributed algorithms for constraint-coupled problems (see Notarstefano et al. (2019) for an overview). All of these algorithms require the overall objective function to be expressed as the sum of local functions. In order to bring Problem (4) into such a structure, one could consider the logarithm of the objective function in (4). However, such an approach would require log-concavity of the local objective functions, which is a strong assumption. In the following, we propose a distributed algorithm that is shown to work without the log-concavity assumption.

4. DISTRIBUTED ALGORITHM

In this section, we propose a distributed algorithm to obtain the Nash bargaining solution of Problem (2), which is the optimal solution of Problem (4). First, we reformulate the problem to make it amenable for distributed optimization. Second, we formalize our Match Nash Weights distributed algorithm, which is then analyzed for a particular case.

4.1 Problem Reformulation

As mentioned above, a distributed solution of the Nash bargaining problem (4) is not directly possible using existing distributed optimization algorithms due to the objective being a product of local contributions. On the other hand, Problem (3) is amenable to state-of-the art distributed optimization algorithms. Therefore, in view of Pareto optimality of the Nash bargaining solution and of Lemma 2.3, we now elaborate on the computation of the individual objectives' weighting factors w_i corresponding to the Nash bargaining solution. In the following, we consider a modified version of Problem (3),

$$\begin{aligned} \max_{x_1,\ldots,x_N} \ & \sum_{i=1}^N w_i f_i(x_i) \\ \text{subj. to } & h_i(x_i) \leq 0, \qquad i \in \{1, \ldots, N\}, \\ & f_i(x_i) \geq d_i, \qquad i \in \{1, \ldots, N\}, \\ & \sum_{i=1}^N g_i(x_i) \leq 0, \end{aligned} \tag{5}$$

where we also added the constraint $f_i(x_i) \geq d_i$, which is present in Problem (4) but not in (3). Note that this constraint is convex and involves only local information for each agent i, therefore state-of-art distributed algorithms can be applied to Problem (5). We also make the following assumption on the problem, which is standard in the literature and allows for the application of duality.

Assumption 4.1. (Slater). There exist $\bar{x}_1, \ldots, \bar{x}_N$ such that $h_i(\bar{x}_i) < 0$ and $f_i(\bar{x}_i) > d_i$ for all i and $\sum_{i=1}^N g_i(\bar{x}_i) < 0$.□

As proposed in Waslander et al. (2004), let us compute the weights corresponding to the Nash bargaining solution by comparing the optimality conditions of Problems (5) and (4). The next theorem formalizes this fact.

Theorem 4.2. Let Assumptions 2.1, 3.1 and 4.1 hold. Moreover, for $\alpha > 0$, let the weights of Problem (5) be

$$w_i^\star = \frac{\alpha}{y_i^\star - d_i}, \qquad i \in \{1, \ldots, N\}. \tag{6}$$

Then, any optimal solution $(x_1^\star, \ldots, x_N^\star)$ of Problem (5) corresponds to the Nash Bargaining Solution of the bargaining problem, i.e., $f_i(x_i^\star) = y_i^\star$ for all i. □

Some comments are in order. First, there is a degree of freedom that is given by the choice of $\alpha > 0$. Indeed, the weights corresponding to the Nash bargaining solution are defined up to a constant, therefore they essentially identify a "direction" of optimization. Second, note that the weights (6) explicitly depend on the optimal solution. This fact directly suggests that we can design a distributed algorithm, proposed in the next section, in which the weights are iteratively adapted.

4.2 Algorithm Description

We can now state the proposed distributed algorithm, where $\lambda \in (0, 1)$ and $M > 0$ are two given design parameters.

Distributed Algorithm Match Nash Weights

Initialization: positive weights $w_i^0 > 0$ for all i

Perform distributed optimization and compute $(x_1^t, \ldots, x_N^t)$, an optimal solution of (5) with $w_i = w_i^t$

Each agent i **computes** $\widetilde{w}_i^t$ locally as

$$\widetilde{w}_i^t = \begin{cases} \dfrac{1}{f_i(x_i^t) - d_i} & \text{if } f_i(x_i^t) > d_i \\ M & \text{if } f_i(x_i^t) = d_i \end{cases} \tag{7}$$

Compute $\|w^t\|^2$ and $\|\widetilde{w}^t\|^2$ via average consensus

Each agent i **updates** w_i^{t+1} as

$$w_i^{t+1} = \lambda \frac{w_i^t}{\|w^t\|} + (1 - \lambda) \frac{\widetilde{w}_i^t}{\|\widetilde{w}^t\|} \tag{8}$$

Let us comment on the algorithm evolution. Initially, agents have an estimate w^0 of the optimal weights. With this estimate, they compute an optimal solution $(x_1^t, \ldots, x_N^t)$ of Problem (5) that corresponds to a possible outcome of the bargaining game via distributed optimization. Then, they use the iteration (7) to compute a new set of weights $\widetilde{w}^t$. In order to improve the convergence properties of the algorithm, they further perform a combination step (8), in which the updated set of weights is computed as a convex combination of w^t and $\widetilde{w}^t$. Intuitively, the algorithm is always changing the "direction" of optimization in Problem (5) by adapting w_i until the weights of the Nash bargaining solution are reached.

We do not provide a full analysis of Match Nash Weights, however in the next subsection we give a theoretical result for a special case. In Section 5, we demonstrate the efficacy of the proposed algorithm through numerical simulations.

4.3 Algorithm Analysis

In this section we present a convergence analysis of our distributed algorithm under simplifying assumptions that allow us to explicitly express the evolution of the algorithm in order to avoid technicalities. Although this example may seem oversimplified, its sole purpose is to highlight the main features of the analysis of the algorithm in full generality, which is subject of ongoing work.

Let us consider the following multi-objective optimization problem, which is a special case of Problem (2),

$$\begin{aligned} \max_{x_1,\ldots,x_N} \quad & [\sqrt{x_1}, \ \ldots, \ \sqrt{x_N}] \\ \text{subj. to} \quad & \sum_{i=1}^{N} x_i \leq N \\ & x_i \in [0, N], \qquad \text{for } i \in \{1, 2\}. \end{aligned} \tag{9}$$

Let us formulate the bargaining problem corresponding to Problem (9). The feasible set of the problem is $\mathcal{X} = \{(x_1, \ldots, x_N) \in [0, N]^N : \sum_{i=1}^N x_i \leq N\}$ and the vector-valued objective function is $F(x) = [\sqrt{x_1}, \ \ldots, \ \sqrt{x_N}]$. It follows from these definitions that the feasible set of the bargaining problem, i.e. $\mathcal{Y} = F(\mathcal{X})$, admits the expression

$$\mathcal{Y} = \left\{ (y_1, \ldots, y_N) \;\middle|\; y \geq 0, \ \sum_{i=1}^{N} y_i^2 \leq N \right\}. \tag{10}$$

The set $\mathcal{Y}$ is an N-sphere intersected with the non-negative orthant. This is a "normalized" bargaining problem, in the sense that it satisfies the symmetry axiom and the simplex $\sum_{i=1}^N y_i \leq N$ is a supporting hyperplane of the set $\mathcal{Y}$. For such a particular case, the Nash bargaining solution is already known to be equal to $y^\star = \mathbf{1}$ (Thomson, 1994).

We will now analyze the algorithm applied to this setup, assuming the disagreement point is the origin, i.e., $d = 0$. First, note that Assumptions 2.1, 3.1 and 4.1 are satisfied, except for differentiability of the objective functions at $x_i = 0$, which however is only required to apply Theorem 4.2. By using (10) and by applying the change of variables $y_i = \sqrt{x_i}$ for all i, it follows that the network-wide optimization problem with weighted sum of the objectives amounts to

$$\begin{aligned} \max_{y_1,\ldots,y_N} \quad & \sum_{i=1}^{N} w_i y_i \\ \text{subj. to} \quad & y \geq 0 \\ & \sum_{i=1}^{N} y_i^2 \leq N. \end{aligned} \tag{11}$$

By applying Theorem 4.2 to Problem (11) the weights corresponding to the Nash bargaining solution are $w^\star = \alpha \mathbf{1}$, for $\alpha > 0$. At any iteration t, if $(y_1^t, \ldots, y_N^t)$ denotes the optimal solution of Problem (11) for $w = w^t$, the computation of $\widetilde{w}_i^t$ reduces to

$$\widetilde{w}_i^t = \begin{cases} 1/y_i^t & \text{if } y_i^t > 0, \\ M & \text{if } y_i^t = 0. \end{cases}$$

Next we provide the convergence result for Match Nash Weights applied to Problem (9) in the case of two agents. To remove the extra degree of freedom $\alpha > 0$, the convergence result is stated in terms of the normalized version of w^t.

Theorem 4.3. Assume $N = 2$ and let $0 < \lambda < 1$. Consider the sequence $\{w^t\}_{t \geq 0}$ generated by Match Nash Weights applied to Problem (2), initialized with $w^0 > 0$. Then, it holds

$$\lim_{t \to \infty} u^t = u^\star \triangleq \frac{1}{\sqrt{2}} \mathbf{1}.$$

Proof. For space reasons, we only provide a sketch of the proof. It is based on deriving an explicit expression of u^t of the form

$$u_1^{t+1} = \frac{\lambda u_1^t + (1-\lambda) u_2^t}{\sqrt{\lambda^2 + (1-\lambda)^2 + 4\lambda(1-\lambda) u_1^t u_2^t}},$$

and a symmetric expression for u_2^t. Then, the main idea is to prove that, for $\lambda \in (0, 1)$, the angle between the vectors u^t and $u^\star$ decreases strictly at each iteration. This can be accomplished by defining a sequence V^t of the form

$$V^t \triangleq \frac{\mathbf{1}^\top u^t}{\|\mathbf{1}\|} = \frac{u_1^t + u_2^t}{\sqrt{2}}, \qquad \text{for all } t \geq 0,$$

which is equal to the cosine of the angle between u^t and $u^\star$ and thus satisfies $-1 \leq V^t \leq 1$. By using geometric observations, it is possible to show that V^t converges to 1 and, as a consequence, the vector u^t converges to $u^\star$.

5. NUMERICAL EXAMPLES

In this section we demonstrate the efficacy and the properties of Match Nash Weights in numerical simulations. In particular, we first provide simulation results of the setup (9) in order to illustrate the convergence analysis of the preceding section. Subsequently, we loop back to our very initial motivation and have our algorithm solve the problem of determining a fair economic optimal steady-state configuration for a set of constraint-coupled linear systems. This second, more elaborate example highlights that Match Nash Weights is indeed capable of finding the Nash bargaining solution in a distributed way for setups much more complex than the one exemplarily analyzed in the previous section.

5.1 Two-agent example

We consider the setup (9) with disagreement point $d = 0$. The initial condition of each agent $w_i^0 > 0$ is randomly chosen and the algorithm is run for different values of λ. A detailed analysis of the impact of the parameter λ is subject of ongoing work.

Figure 1 reports the cost error with respect to the optimal cost of Problem (4) Note that, for $\lambda = 0$, the algorithm does not converge. For $\lambda \in \{0.2, 0.9\}$, the algorithm converges, although with different rates, while for $\lambda = 0.5$ convergence occurs at the first iteration. Indeed, in this example, for $\lambda = 0.5$ any choice of $u^0 > 0$ immediately yields $u^1 = u^\star$.

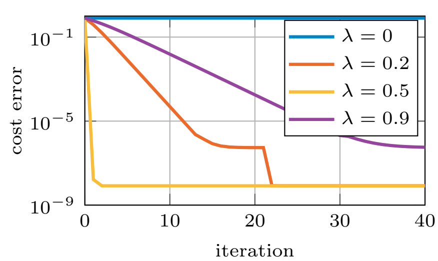

Fig. 1. Cost error for the simulation with $N = 2$ agents.

5.2 N-agent example

We now focus on an example scenario of our initial motivation, namely distributed computation of fair optimal steady states. The setup detailed next is a readapted version of Köhler et al. (2018, Section 5.1).

Let us consider $N = 50$ two-dimensional discrete-time systems of the form $x_i(k+1) = A_i x_i(k) + B_i u_i(k)$, with $x_i(k) \in \mathbb{R}^2$ and $u_i(k) \in \mathbb{R}$ with the system matrices $A_i = \left[\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right]$ and $B_i = [0, 1]^\top$. Local state and input constraints are $-1 \leq x_i, u_i \leq 1$ for all i. The objective function of each agent is $\ell_i(x_i, u_i) = -\eta_i\big((a_i - x_{i,1})^2 + (b_i - x_{i,2})^2 + (d_i - u_i)^2\big)$, with a_i, b_i, $d_i \in [0, 0.1]$ and $\eta_i \in [0, 1]$ randomly chosen. The coupling constraint is given by the resource constraint $\sum_{i=1}^N x_{i,1} \leq -0.5$. For each agent, the equilibrium $(x_i^{\mathrm{d}}, u_i^{\mathrm{d}})$ corresponding to the disagreement point is given by $x_i^{\mathrm{d}} = [-1, 0]^\top$ and $u_i^{\mathrm{d}} = 0$. First, we solved Problem (4) to optimality by using a centralized solver. Then, we executed our algorithm by initializing each agent with a random weight $w_i^0 > 0$ and by using different values of λ.

In Figure 2 we report the cost error and the error of x^t with respect the optimal cost and the optimal solution of Problem (4) (respectively). It is seen that the algorithm converges to the optimal weights and that the vector x^t approaches the optimal solution for all values of λ.

6. CONCLUSIONS

In this paper we presented a novel distributed optimization framework. Agents in a network aim to cooperatively solve a constraint-coupled multi-objective optimization problem, while ensuring fairness of the computed solution. Cooperative game theoretical tools have been used to characterize local objective weightings corresponding to the Nash bargaining solution. A distributed algorithm has been proposed to compute such weighting factors. Convergence of the algorithm in a particular case has been proven, while numerical simulations demonstrate the efficacy of the proposed approach. A detailed study of convergence conditions of the proposed algorithm for more general settings is the subject of ongoing work.

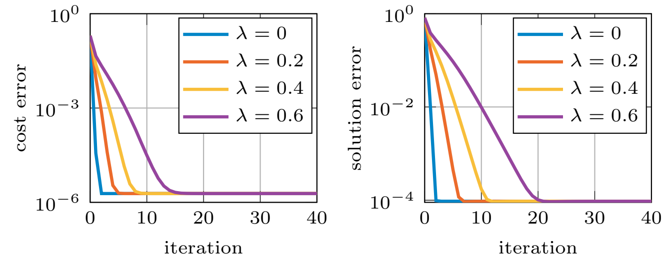

Fig. 2. Cost error (left) and solution error (right) for the simulation with $N = 50$ agents.

REFERENCES

Bürger, M., Notarstefano, G., and Allgöwer, F. (2014). A polyhedral approximation framework for convex and robust distributed optimization. *IEEE Transactions on Automatic Control*, 59(2), 384–395.

Christofides, P.D., Scattolini, R., de la Pena, D.M., and Liu, J. (2013). Distributed model predictive control: A tutorial review and future research directions. *Computers & Chemical Engineering*, 51, 21–41.

Falsone, A., Margellos, K., Garatti, S., and Prandini, M. (2017). Dual decomposition for multi-agent distributed optimization with coupling constraints. *Automatica*, 84, 149–158.

Grammatico, S., Parise, F., Colombino, M., and Lygeros, J. (2016). Decentralized convergence to nash equilibria in constrained deterministic mean field control. *IEEE Transactions on Automatic Control*, 61(11), 3315–3329.

Köhler, P.N., Müller, M.A., and Allgöwer, F. (2018). A distributed economic MPC framework for cooperative control under conflicting objectives. *Automatica*, 96, 368–379.

Liang, S., Yi, P., and Hong, Y. (2017). Distributed nash equilibrium seeking for aggregative games with coupled constraints. *Automatica*, 85, 179–185.

Maestre, J.M. and Negenborn, R.R. (2013). *Distributed model predictive control made easy*, volume 69.

Miettinen, K. (2012). *Nonlinear multiobjective optimization*, volume 12.

Müller, M.A. and Allgöwer, F. (2017). Economic and distributed model predictive control: Recent developments in optimization-based control. *SICE Journal of Control, Measurement, and System Integration*, 10(2), 39–52.

Nash, J.F. (1950). The bargaining problem. *Econometrica: Journal of the Econometric Society*, 155–162.

Notarnicola, I. and Notarstefano, G. (2019). Constraint-coupled distributed optimization: a relaxation and duality approach. *IEEE Transactions on Control of Network Systems*, (99), 1–10.

Notarstefano, G., Notarnicola, I., and Camisa, A. (2019). Distributed optimization for smart cyber-physical networks. *Foundations and Trends® in Systems and Control*, 7(3), 253–383.

Salehisadaghiani, F. and Pavel, L. (2016). Distributed nash equilibrium seeking: A gossip-based algorithm. *Automatica*, 72, 209–216.

Thomson, W. (1994). Cooperative models of bargaining. *Handbook of game theory with economic applications*, 2, 1237–1284.

Waslander, S.L., Inalhan, G., and Tomlin, C.J. (2004). Decentralized optimization via nash bargaining. In *Theory and algorithms for cooperative systems*, 565–585.