A scenario-based approach to multi-agent optimization with distributed information

Alessandro Falsone *  Kostas Margellos **  Maria Prandini *  Simone Garatti *

* Dipartimento di Elettronica Informazione e Bioingegneria, Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133 Milano, Italy (e-mail: \{alessandro.falsone,maria.prandini,simone.garatti\}@polimi.it).
** Department of Engineering Science, University of Oxford, Parks Road, OX1 3PJ, Oxford, UK (e-mail: kostas.margellos@eng.ox.ac.uk)

Abstract: In this paper, we consider optimization problems involving multiple agents. Each agent introduces its own constraints on the optimization vector, and the constraints of all agents depend on a common source of uncertainty. We suppose that uncertainty is known locally to each agent through a private set of data (multi-agent scenarios), and that each agent enforces its scenario-based constraints to the solution of the multi-agent optimization problem. Our goal is to assess the feasibility properties of the corresponding multi-agent scenario solution. In particular, we are able to provide a priori certificates that the solution is feasible for a new occurrence of the global uncertainty with a probability that depends on the size of the datasets and the desired confidence level. The recently introduced wait-and-judge approach to scenario optimization and the notion of support rank are used for this purpose. Notably, decision-coupled and constraint-coupled uncertain optimization programs for multi-agent systems fit our framework and, hence, any distributed optimization scheme to solve the associated multi-agent scenario problem can be accompanied with our a priori probabilistic feasibility certificates.

Keywords: Uncertain systems, multi-agent systems, data-driven optimization, distributed algorithms.

1. INTRODUCTION

We consider cooperative optimization in multi-agent systems where the goal is to minimize some global cost function subject to local constraints. Prominent examples of systems involving multiple entities interacting with each other can be found in various application domains, such as power systems, Bolognani et al. (2015); Zhang and Giannakis (2016), wireless networks, Mateos and Giannakis (2012); Baingana et al. (2014), and robotics, Martínez et al. (2007). Most of the literature addressing cooperative optimization in multi-agent systems focuses on the design of algorithms that are compatible with the networked structure of the system, distribute the computations among agents, and preserve privacy of local information. Typically, they refer to a deterministic nominal setting and neglect the uncertainty affecting the system. However, this may result in an infeasible design when uncertainty takes a value different from the nominal one, which hampers the actual implementation of the computed optimal solution.

In this paper, we instead focus on multi-agent optimization problems affected by uncertainty, which is only known through data. More specifically, we consider \( m \) agents that communicate to cooperatively solve the following optimization problem:

\[
P_S: \min_{x \in X} f(x)
\]

subject to \( x \in \bigcap_{\delta \in \Delta} \bigcap_{i=1}^{m} X_i(\delta), \)

where \( x \in \mathbb{R}^n \) represents a vector of \( n \) decision variables that is constrained to take values in a convex set \( X \subseteq \mathbb{R}^n \), and \( \delta \)

* Research was partially supported by EPSRC UK, grant EP/P03277X/1.
be evaluated for the sets of scenarios $S_i$, $i = 1, \ldots, m$, thus obtaining the following multi-agent scenario program
\[ P_N : \min_{x \in X} f(x) \]
subject to $x \in \bigcap_{i=1}^{m} \bigcap_{\delta \in S_i} X_i(\delta)$,
where $N$ denotes the total number of independent scenarios, i.e., $N = \bar{N}$ if the scenarios are common, and $N = \sum_{i=1}^{m} N_i$ if the scenarios are private.

$P_N$ is a data-driven approximation of $P_S$, which can be solved by the agents, each contributing with its own piece of information. The main objective of this paper is that of assessing the robustness level of the solution to program $P_N$ with respect to the original problem $P_S$. This amounts to evaluating the probabilistic feasibility level of the solution to program $P_N$ for the constraint $x \in \bigcap_{i=1}^{m} X_i(\delta)$ with $\delta$ taking values in $\Delta$ according to $P$. Since the incurred probabilistic feasibility level will depend on the extracted multi-sample of scenarios $S = \bigcup_{i=1}^{m} S_i$, evaluations that hold with a certain confidence, measured according to the product probability $P^N$ on the multi-scenario space $\Delta^N$ will be given.

If the scenarios are common across agents – case (a) – then, standard results of the scenario approach (Calafiore and Campi (2006); Campi and Garatti (2008); Campi et al. (2009); Campi and Garatti (2011); Garatti and Campi (2013); Margellos et al. (2015)) can be applied to provide the sought priori probabilistic certificates on the feasibility of the solution. However, when scenarios constitute private information of each agent – case (b) – then, standard scenario theory does not apply anymore and has to be extended. Here, we study the general problem leveraging the recent results of Campi et al. (2015, 2018). We also provide a tighter result for the particular case where the agents impose their constraints on separate decision variables by exploiting the concept of support rank as in Schildbach et al. (2013). This is the main contribution of our paper.

In the final part of the paper, we also show that our framework accommodates two problem classes, namely, decision-coupled and constraint-coupled optimization programs, extensively treated in the literature on distributed optimization. Since the multi-agent problem $P_N$ can be treated as a deterministic program once the scenarios have been observed, any distributed algorithm that provides an optimal solution to $P_N$ without sharing private information can be accompanied with our a priori probabilistic certificates. The introduced multi-agent scenario approach is thus applicable to a large class of distributed algorithms, those that are adopted for decision-coupled programs (see e.g. Nedic and Ozdaglar (2009); Nedic et al. (2010); Margellos et al. (2018)) and for constraint-coupled programs (see e.g. Zhu and Martinez (2012); Chang et al. (2014); Boyd et al. (2010); Notarnicola and Notarstefano (2017); Falsone et al. (2017)). This is a further contribution of our work.

It is worth mentioning that distributed techniques taking into account uncertainty have recently appeared in Tçöting and Sayed (2014); Carlone et al. (2014); Chamanbaz et al. (2017); Lee and Nedic (2013, 2016); Margellos et al. (2018); Sayin et al. (2017). However, the techniques proposed in the literature are tailored to the considered algorithm and not of general applicability as the multi-agent scenario approach presented in this paper.\(^1\)

\(^1\) Specifically, the approaches in Tçöting and Sayed (2014); Lee and Nedic (2016) require some regularity conditions on the agents’ cost function; Sayin et al. (2017) and Lee and Nedic (2013) require to extract an infinite number of scenarios; the randomized algorithm of Carlone et al. (2014) requires to

\[ P_N : \min_{x \in X} f(x) \]
subject to $x \in \bigcap_{i=1}^{m} \bigcap_{\delta \in S_i} X_i(\delta)$,
where we changed the subscript from $N$ to emphasize the fact that there are $\bar{N}$ common scenarios. Let us denote by $x^*_N$ a solution of $P_N$ (which is well-defined based on Assumption 1), possibly adopting a convex tie-break rule to get a unique minimizer.

The problem we address here is the evaluation of the robustness level of $x^*_N$. In the present context, the theory of the scenario approach developed in Calafiore and Campi (2006); Campi and Garatti (2008) provides a full-fledged characterization, showing that $x^*_N$ is feasible for $P_S$ up to an explicitly quantified probabilistic level $\varepsilon$. To illustrate the result we need first to introduce the notion of support set of Campi et al. (2015).\(^2\) That is, for a given optimization program, a support set is a minimal cardinality subset of constraints that alone suffices to retrieve the solution to the original program where all constraints are in exchange constraints over a time-invariant communication network, whereas Chamanbaz et al. (2017) allows for time-varying communications but is confined to linear programs.

\(^2\) The support set was called compression scheme in Margellos et al. (2015) and in typical cases (referred to as non-degenerate) coincides with the set of support constraints (see Campi and Garatti (2008), Definition 2).

\[ P_N : \min_{x \in X} f(x) \]
subject to $x \in \bigcap_{i=1}^{m} \bigcap_{\delta \in S_i} X_i(\delta)$,
in a sense, the constraints that are not in the support set are inessential since removing all of them leaves the solution unchanged. It is well known that for convex optimization programs the cardinality of the support set is always no bigger than the number $n$ of decision variables, see (Calafiore and Campi, 2006, Theorem 3). In some cases\footnote{E.g., because of the structure of constraints or the presence of some regularization term, as e.g. in Campi and Caré (2013).} the maximal support set cardinality can be strictly smaller than $n$ and improved bounds can be obtained, see e.g. Schioldbach et al. (2013).

Referring back to $P_N$, which is convex by Assumption 1, we denote by $d \in \mathbb{N}_+$ any available upper-bound to the cardinality of the support set of $P_N$. The following theorem is a direct consequence of the results of Calafiore and Campi (2006).

**Theorem 1.** Fix $\beta \in (0, 1)$ and let
\[
\bar{\varepsilon} = 1 - \sqrt{n^{-d} \frac{\beta}{\binom{N}{d}}}.
\]
We then have that
\[
P^N \left\{ S \in \Delta^N : \mathbb{P} \left\{ \delta \in \Delta : x^*_N \notin \bigcap_{i=1}^m X_i(\delta) \right\} \leq \bar{\varepsilon} \right\} \geq 1 - \beta.
\]

Theorem 1 says that with confidence no smaller than $1 - \beta$, $x^*_N$ is feasible for $P_S$ except for a portion of uncertainty instances that has probability $\bar{\varepsilon}$ at most. Though $\bar{\varepsilon}$ depends on $N$, $\beta$ and $d$, this dependency is suppressed throughout to avoid notational cluttering.

**Remark 1.** (improved bound). Following Campi and Garatti (2008), an improved result could be given by replacing $\bar{\varepsilon}$ in (4) with the solution of the equation $\sum_{k=1}^{\frac{N}{d}-1} \binom{\frac{N}{d}}{k} \bar{\varepsilon}^k (1 - \bar{\varepsilon})^{N-k} = \beta$. For simplicity, we use (4) which gives an explicit – although conservative – expression for $\bar{\varepsilon}$.

If $N$ is too small, it may be that $\bar{\varepsilon}$ is larger than 1 and the theorem is not of practical interest. In this case, one may want to fix $\bar{\varepsilon}, \beta \in (0, 1)$ and use Theorem 1 the other way around to determine how many scenarios are needed for $S$ to hold. This amounts to solving (4) with respect to $N$. See (Calafiore and Campi, 2006, Theorem 1).

### 2.2 Private scenarios

Suppose now that scenarios are private resources collected independently by the agents. This means that, for $i = 1, \ldots, m$, agent $i$ is supplied with its own set $S_i \subseteq \Delta$ of $N_i \in \mathbb{N}_+$ independent scenarios extracted according to $\mathbb{P}$ and that scenarios belonging to different sets $S_i$ are also independent.

The resulting multi-agent scenario problem is given by the optimization program $P_N$ in (2) where the total number of independent scenarios is $N = \sum_{i=1}^m N_i$.

As in Section 2.1, we want to show that the minimizer $x^*_N$ of $P_N$ (again, well-defined thanks to Assumption 1) is feasible for $P_S$ in a probabilistic sense. This means that we have to assess the probability with which $x^*_N$ satisfies the “global” constraint $\bigcap_{i=1}^m X_i(\delta)$, where $\delta$ is the same for all the $X_i(\delta)$, $i = 1, \ldots, m$. On the other hand, in the computation of $x^*_N$ through $P_N$, $X_i(\delta)$, $i = 1, \ldots, m$ is evaluated for the scenarios in $S_i$, which are different from the scenarios for which the other $X_j(\delta), j \neq i$, are evaluated. This poses a major challenge in this private scenarios set-up.

Similar argument was also used in Kariotoglou et al. (2016).
where the last step follows from (7). This concludes the proof.

In words, Proposition 1 says that with confidence no smaller than $1 - \beta$, $x_N^*$ is feasible for $P_\delta$ except for a portion of uncertainty instances that has probability $\bar{\varepsilon}$ at most. Proposition 1 has an issue though, because $\bar{\varepsilon}$ is very conservative when the number of agents is large and this limits the applicability of the results. To see this, we perform a comparison between $\bar{\varepsilon}$ and $\varepsilon$, the bound to the probability of violation we have when the scenarios are common across the agents. Suppose that $N_i = N$ and $\beta_i = \beta/m$, for all $i = 1, \ldots, m$. Using (4) and (6), we then have that $\bar{\varepsilon} = m\varepsilon_i \approx m^{2/3}$. This simple calculation shows that $\bar{\varepsilon}$ approximately grows linearly with the number of agents $m$, a fact that is also apparent from a numerical simulation that will be presented next (see Fig. 2).

**An a priori bound using a posteriori results.**

The conservatism of Proposition 1 is due to the fact that it considers a fictitious situation where $d_{i,N} = d$ for all $i = 1, \ldots, m$, while the fact $\sum_{i=1}^{m} d_{i,N} \leq d$ reveals us that when $d_{i,N} = d$ for some $i$, then $d_{j,N} = 0$ for all other $j \neq i$. In this subsection, we want to exploit the inequality $\sum_{i=1}^{m} d_{i,N} \leq d$ to reduce the conservatism of Proposition 1 and, to this purpose, we use the results of Campi et al. (2015, 2018) that are based on a wait-and-judge, a posteriori, perspective. Following Theorem 1 and Remark 4 in Campi et al. (2018), for each $i = 1, \ldots, m$, fix $\beta_i \in (0, 1)$ and consider function $\varepsilon_i(\cdot)$ defined as follows:

$$\varepsilon_i(k) = 1 - \frac{\beta_i}{(d+1)(N_i^m)}^{k}, \text{ for all } k = 0, 1, \ldots, d. \quad (10)$$

Besides $k$, $\varepsilon_i(\cdot)$ depends on $N_i$, $\beta_i$ and $d$ as well, but this dependency is not explicitly indicated to ease the notation. By focusing on a given agent $i$, $i = 1, \ldots, m$, an application of Theorem 1 of Campi et al. (2018) conditional to the scenarios of all other agents $S \setminus S_i$ yields

$$\mathbb{P}^N\left\{ S \in \Delta^N : \exists \delta \in \Delta : \exists x_N^* \notin X_i(\delta) \right\} \leq \varepsilon_i(d_{i,N})$$

Integrating (11) with respect to the probability of realizing the scenarios $S \setminus S_i$, we then have that

$$\sum_{i=1}^{m} \varepsilon_i(d_{i,N}) \geq 1 - \beta_i. \quad (11)$$

Differently from (5) and (8), the assessment of the violation probability level in (13) is a-posteriori because $\varepsilon_i(d_{i,N})$ is a function of the seen scenarios. An a-priori assessment can be easily derived by simply computing a worst-case value for $\sum_{i=1}^{m} \varepsilon_i(d_{i,N})$ over the possible values of $d_{i,N}$, $i = 1, \ldots, m$, that satisfies $\sum_{i=1}^{m} d_{i,N} \leq d$. This amounts to solving

$$\varepsilon = \max_{\{d_i \in \mathbb{N}_i\}^m} \sum_{i=1}^{m} \varepsilon_i(d_i), \text{ subject to } \sum_{i=1}^{m} d_i \leq d \quad (14)$$

which is an integer maximization program that can be solved via numerical solver. Notice that $\{d_i^{\text{opt}}\}^m_{i=1}$ in (14) are integer optimization variables, which should not be confused with $\{d_i\}^m_{i=1}$. In conclusion, the following theorem holds true.

**Theorem 2.** Fix $\beta \in (0, 1)$ and choose $\beta_i, i = 1, \ldots, m$, such that $\sum_{i=1}^{m} \beta_i = \beta$. Set $\varepsilon$ according to (14). We then have that

$$\mathbb{P}^N\left\{ S \in \Delta^N : \exists \delta \in \Delta : \exists x_N^* \notin \bigcap_{i=1}^{m} X_i(\delta) \right\} \leq \varepsilon$$

Integrating (11) with respect to the probability of realizing the scenarios $S \setminus S_i$, we then have that

$$\sum_{i=1}^{m} \varepsilon_i(d_{i,N}) \geq 1 - \beta_i. \quad (11)$$

Enforcing the condition $\sum_{i=1}^{m} d_{i,N} \leq d$ when determining $\varepsilon$ in (14), provide a tighter estimate for the violation probability in Theorem 2 with respect to that in Proposition 1. This is also shown pictorially in Fig. 2, where we plot $\bar{\varepsilon}$ in Theorem 1, $\tilde{\varepsilon}$ in Proposition 1, and $\varepsilon$ in Theorem 2 as functions of the number of agents $m$. The conser-

---

5 The $\approx$ in the last step is because in (6) we have $\beta_i = \beta/m$ in place of $\beta$ in (4); yet, this dependence on $m$ via $\beta_i$ has a negligible effect.
m of agents, when $\beta = 10^{-5}$, $N_i = \bar{N} = 4500$, $\beta_i = \beta/m$, $i = 1, \ldots, m$, and $d = 50$. As it appears, $\tilde{\epsilon}$ grows as $m \cdot \bar{\epsilon}$, while $\epsilon$ is only moderately increasing with $m$.

When the number of agents is very large and/or there are few scenarios available, $\epsilon$ may still exceed 1, making the result of Theorem 2 trivial. Similarly to the discussion at the end of Section 2.1, note that Theorem 2 can be reversed to compute the number of scenarios $N_i$ that need to be extracted by agent $i$, $i = 1, \ldots, m$, for given values of $\epsilon, \beta \in (0, 1)$. This can be achieved by numerically seeking for values of $N_i, i = 1, \ldots, m$, that lead to a solution of (14) that attains the desired $\epsilon$.

Private scenarios with local decision vectors. We consider the case where the decision vector $x$ can be partitioned into $m$ parts, each one associated to an agent and each agent imposes constraints only on its own set of decision variables. More precisely, we have $x = [x_1^T \ldots x_m^T]^T$ where $x_i \in \mathbb{R}^{n_i}$ is associated with agent $i$, $i = 1, \ldots, m$, and the constraint set of agent $i$ takes the form

$$X_i(\delta) = \mathbb{R}^{n_1} \times \cdots \times X_i(\delta) \times \cdots \times \mathbb{R}^{n_m},$$

where $\delta \in \Delta, i = 1, \ldots, m$.

The structure of the problem is such that

$$\mathbb{P}^N \left\{ S \in \Delta^N : \mathbb{P} \left\{ \delta \in \Delta : x_N^i \notin X_i(\delta) \right\} \leq \epsilon_i \right\}$$

is identical to problem $P_N$ if we set $X = \mathbb{R}^n$, $f(\cdot) = \sum_{i=1}^m f_i(\cdot)$. For each $i = 1, \ldots, m$, $f_i(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}$ is the cost function of agent $i$, whereas, for any $\delta \in \Delta, X_i(\delta) \subseteq \mathbb{R}^n$ represents all constraints to the decision vector imposed by agent $i$. Algorithms like the one in Margellos et al. (2018) allow to compute a solution according to a distributed scheme where local information (set of scenarios $S_i$, cost function $f_i$, constraint $X_i$) is not disclosed to the other agents. The optimal solution returned by the chosen distributed algorithm can be accompanied by the probabilistic feasibility certificate of Theorem 2.

**3. APPLICATION TO DISTRIBUTED OPTIMIZATION**

In order to deal with the multi-agent nature of the problem, to avoid the presence of a central regulatory authority and to accommodate the need of not disclosing possibly private information of agents, distributed optimization methods could be adopted to solve the multi-agent scenario program $P_N$.

Let $S_i$ denote the $i$-th scenario available. Each $S_i$ is a subset of $\mathbb{R}^n$ and represents, in turn, an upper bound on the cardinality of the support set in Section 2.1 for the decision set $X_i(\delta) = \mathbb{R}^{n_1} \times \cdots \times X_i(\delta) \times \cdots \times \mathbb{R}^{n_m}$. This can be achieved to a solution of (14) that attains the desired $\epsilon$. 

**Decision-coupled optimization program**

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m f_i(x)$$

subject to $x \in \bigcap_{i=1}^m X_i(\delta), i = 1, \ldots, m$

is identical to problem $P_N$ if we set $X = \mathbb{R}^n$ and $f(\cdot) = \sum_{i=1}^m f_i(\cdot)$. For each $i = 1, \ldots, m$, $f_i(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}$ is the cost function of agent $i$, whereas, for any $\delta \in \Delta, X_i(\delta) \subseteq \mathbb{R}^n$ represents all constraints to the decision vector imposed by agent $i$. Algorithms like the one in Margellos et al. (2018) allow to compute a solution according to a distributed scheme where local information (set of scenarios $S_i$, cost function $f_i$, constraint $X_i$) is not disclosed to the other agents. The optimal solution returned by the chosen distributed algorithm can be accompanied by the probabilistic feasibility certificate of Theorem 2.

**Constraint-coupled optimization program**

$$\{x_i \in \mathbb{R}^{n_i} \}_{i=1}^m \sum_{i=1}^m g_i(x_i) \leq 0,$$

subject to $x_i \in \bigcap_{\delta \in S_i} X_i(\delta), i = 1, \ldots, m$

is identical to $P_N$ if we set $x = [x_1^T \ldots x_m^T]^T$, $X = \{x \in \mathbb{R}^n : \sum_{i=1}^m g_i(x_i) \leq 0\}, f(\cdot) = \sum_{i=1}^m f_i(\cdot)$, and $X_i(\delta) = \mathbb{R}^{n_1} \times \cdots \times X_i(\delta) \times \cdots \times \mathbb{R}^{n_m}, i = 1, \ldots, m$. This is identical to $P_N$ if we set $x = [x_1^T \ldots x_m^T]^T$, $X = \{x \in \mathbb{R}^n : \sum_{i=1}^m g_i(x_i) \leq 0\}, f(\cdot) = \sum_{i=1}^m f_i(\cdot)$, and $X_i(\delta) = \mathbb{R}^{n_1} \times \cdots \times X_i(\delta) \times \cdots \times \mathbb{R}^{n_m}, i = 1, \ldots, m$. In this case, each agent $i$, $i = 1, \ldots, m$, has a local decision vector $x_i \in \mathbb{R}^{n_i}$, its local cost function $f_i(x_i) : \mathbb{R}^{n_i} \rightarrow \mathbb{R}$, and its local constraint set $X_i(\delta) \subseteq \mathbb{R}^{n_i}$. Function $g_i(x_i) : \mathbb{R}^{n_i} \rightarrow \mathbb{R}$ quantifies the amount of $p$ resources that is required by agent $i$. 

![Fig. 3. Dashed green line: $\tilde{\epsilon}$ in Theorem 1; blue solid line: $\epsilon$ in Theorem 2; dotted magenta line: $\epsilon$ in Theorem 3.](image-url)
i to implement its decision $x_i$. The coupling among the agents’ decision is due to the constraint $\sum_{i=1}^{m} g_i(x_i) \leq 0$.

The algorithm based on proximal minimization and dual decomposition in Falsone et al. (2017) can be used to compute an optimal solution to the above constraint-coupled program according to a distributed scheme where local information (set of scenarios $S_i$, functions $f_i$ and $g_i$ and constraint $X_i$) is not disclosed to the other agents. It should be noted that in the case of the constraint-coupled problem, the probabilistic feasibility certificate derived in Theorem 3 can be used in place of the general one in Theorem 2.

4. CONCLUDING REMARKS

We extended the scenario approach to deal with multi-agent optimization problems affected by uncertainty. Specifically, we showed how to extend the probabilistic feasibility guarantee of the classical scenario theory to the case where scenarios are a private local information of each agent. Since our probabilistic feasibility guarantees are independent of the algorithm adopted to solve the multi-agent scenario problem, then, they apply also to the case of distributed optimization schemes. This allows to extend distributed solutions originally developed for deterministic set-ups to the uncertain case, accompanying them with an a priori probabilistic certificate of feasibility.

Current work concentrates towards applying the developed theoretical framework to energy management problems in building networks Belluschi et al. (2020), as well as to non-cooperative multi-agent programs Deori et al. (2018). From a theoretical point of view, we aim at improving the bounds using the recent a posteriori developments of the scenario theory in Campi and Garatti (2018), and at investigating non-convex variants of the multi-agent settings under study.

REFERENCES


