Systems & Control Letters 94 (2016) 63–69

Contents lists available at ScienceDirect

Systems & Control Letters

journal homepage: www.elsevier.com/locate/sysconle

On the computational complexity and generalization properties of multi-stage and stage-wise coupled scenario programs

Nikolaos Kariotoglou [a,*], Kostas Margellos [b], John Lygeros [a]

[a] Automatic Control Laboratory, Department of Information Technology and Electrical Engineering, ETH Zurich, Switzerland
[b] Control Group, Department of Engineering Science, University of Oxford, United Kingdom

ARTICLE INFO

Article history:
Received 10 October 2015
Received in revised form
5 April 2016
Accepted 11 May 2016

Keywords:
Scenario approach
Randomized optimization
Uncertain systems
Approximate dynamic programming

ABSTRACT

We discuss the computational complexity and feasibility properties of scenario sampling techniques for uncertain optimization programs. We propose an alternative way of dealing with a special class of stage-wise coupled programs and compare it with existing methods in the literature in terms of feasibility and computational complexity. We identify trade-offs between different methods depending on the problem structure and the desired probability of constraint satisfaction. To illustrate our results, an example from the area of approximate dynamic programming is considered.

1. Introduction

One way of dealing with data uncertainty in robust optimization is to allow the optimal decision to violate problem constraints on a set of pre-specified measure. The authors in [1,2] provide explicit solutions to such problems under assumptions on the probability distribution of the uncertainty. To avoid such assumptions one can make use of uncertainty samples and construct decisions that only satisfy system constraints on the sampled uncertainty instances. The *scenario approach* [3,4] can be used to provide feasibility generalization statements for convex optimization problems, i.e., how likely it is for a sample-based decision to satisfy the problem constraints for a new realization of the uncertainty that was not included in the samples. Beyond feasibility guarantees, [5–7] provide bounds on the amount of constraint violation and are concerned with probabilistic performance issues.

We focus on scenario based convex optimization problems using the scenario approach [3,4,8,9]. We illustrate that the same guarantees on the feasibility of a scenario based solution may be obtained by formulating alternative scenario programs, each with a potentially different number of decision variables and constraints and hence different computational complexity. We argue that in the case of multi-stage scenario programs, stage-wise coupled via the constraint functions, it is often challenging to decide which algorithm to use and illustrate how different choices give rise to a significant trade-off in the total computation time. Motivated by such cases, we provide a framework to compare approaches in terms of computational complexity, while sharing the same joint constraint feasibility properties. In this context, our contributions are: (1) We show how the scenario approach paradigm can be deployed in stage-wise coupled programs and analyze the feasibility properties of the associated solutions (Sections 3.3 and 3.4). (2) We illustrate how the violation and confidence parameters can be treated as additional degrees of freedom and be selected by means of a convex program in view of reducing the computational complexity (Section 3.5). This is fundamentally different to the existing literature where violation and confidence levels are typically considered as fixed parameters when computing sample size bounds. (3) We compare alternatives with respect to computational complexity and identify underlying trade-offs (Section 4). (4) We demonstrate the results on an approximate dynamic programming (ADP) algorithm developed for reachability problems (Section 5). Applications are not limited to this algorithm since most ADP approaches based on [10] result in a sequence of coupled scenario programs.

Section 2 presents the general problem under consideration and Section 3 the different scenario based alternatives and their properties. In Section 4 we discuss the trade-off between feasibility and computational complexity of each alternative while Section 5 illustrates our results with a numerical example in ADP.

* Corresponding author.
E-mail addresses: nkarioto@ethz.ch (N. Kariotoglou), kostas.margellos@eng.ox.ac.uk (K. Margellos), lygeros@ethz.ch (J. Lygeros).

http://dx.doi.org/10.1016/j.sysconle.2016.05.009
0167-6911/

Let $\mathbb{R}$, $\mathbb{N}$, $\mathbb{N}_+$ denote the real, natural and positive natural numbers. Uncertainty samples are extracted from a possibly unknown set Δ according to a possibly unknown probability measure $\mathbb{P}$. $\mathbb{P}^S$ with $S \in \mathbb{N}_+$ denotes the corresponding product measure. We use i.i.d for identically and independently distributed. Operator $|\cdot|$ denotes the cardinality of its argument, $\dim(A)$ the dimension of a linear space A, and $x \models y$ that x satisfies statement y.

2. Programs with multiple robust constraints

Consider a compact convex set $\mathcal{X} \subseteq \mathbb{R}^d$, a possibly unbounded uncertainty set $\Delta \subseteq \mathbb{R}^w$, a convex cost function $f : \mathcal{X} \to \mathbb{R}$ and a set of $M \in \mathbb{N}_+$ convex constraint functions $g_i : \mathcal{X} \times \Delta \to \mathbb{R}$, $i = 1, \ldots, M$. We deal with robust convex optimization problems (RCP) of the form:

$$\text{RCP} : \begin{cases} \min\limits_{x \in \mathcal{X}} & f(x) \\ \text{s.t} & g_i(x, \delta) \leq 0, \quad \forall \delta \in \Delta, \ \forall i \in \{1, \ldots, M\}. \end{cases} \tag{1}$$

The set Δ may be infinite and possibly unbounded, rendering (1) a convex, semi-infinite optimization program. A common approach to approximate the solution is to impose the constraints on a finite number of uncertainty instances. Consider $S \in \mathbb{N}_+$ i.i.d samples $\{\delta^j\}_{j=1}^S$ extracted from Δ according to some underlying probability distribution, and a collection $\{\Delta_i\}_{i=1}^M$ of M subsets of $\{\delta^j\}_{j=1}^S$ such that for each $\delta \in \{\delta^j\}_{j=1}^S$ there exists i so that $\delta \in \Delta_i$, i.e., the sets may be overlapping but each δ belongs in at least one of them. The interpretation is that for each $i = 1, \ldots, M$, the corresponding constraint $g_i(x, \delta)$ should be satisfied for all $\delta \in \Delta_i$, but not necessarily for all $\delta \in \Delta$. Problem (1) is then approximated by a scenario convex optimization program (SCP) of the form:

$$\text{SCP}[\Delta_1, \ldots, \Delta_M] : \begin{cases} \min\limits_{x \in \mathcal{X}} & f(x) \\ \text{s.t} & g_i(x, \delta) \leq 0, \quad \forall \delta \in \Delta_i, \\ & \forall i \in \{1, \ldots, M\} \end{cases} \tag{2}$$

and can be solved to optimality by various solvers. We impose the following assumption on $\text{SCP}[\Delta_1, \ldots, \Delta_M]$:

Assumption 1. For any set $\{\delta^j\}_{j=1}^S$ and collection of subsets $\{\Delta_i\}_{i=1}^M$ with $S, M \in \mathbb{N}_+$, $\text{SCP}[\Delta_1, \ldots, \Delta_M]$ is feasible, its feasibility region has a non-empty interior and its minimizer $x^*[\Delta_1, \ldots, \Delta_M] : \Delta^S \to \mathcal{X}$ is unique.

We refer to [3,11] for details on how Assumption 1 can be relaxed. Measurability of $x^*[\Delta_1, \ldots, \Delta_M]$ is assumed as needed [7,12]. Same as the standard literature on the scenario approach [3,4] we focus on the feasibility properties of x^* as a function of the algorithm used to construct it; performance issues are discussed in [6,7].

3. Feasibility of scenario convex programs

We introduce four different approaches to formulate the SCP: the standard scenario approach, the multi-stage scenario approach, the stage-wise coupled scenario approach using the same samples at every step and the stage-wise coupled scenario approach using different samples at every step. Due to differences in the generation of samples, each approach provides different design choices. In particular, whenever the constraints in the SCP are sampled separately, additional degrees of freedom are introduced, allowing to choose different feasibility properties for each constraint. We compare all approaches on the same metric of jointly satisfying all of the constraints in (2), and exploit their structure to reduce computational complexity.

3.1. The standard scenario approach

Let $\bar{\Delta} = \{\delta^j\}_{j=1}^S$ and assume that $\Delta_1 = \cdots = \Delta_M = \bar{\Delta}$, i.e., enforce each constraint on all elements in $\bar{\Delta}$. Let d be the dimension of the decision space $\mathcal{X}$ and denote by $\text{SCP}[\bar{\Delta}]$, $x^*[\bar{\Delta}]$ the resulting instance of $\text{SCP}[\Delta_1, \ldots, \Delta_M]$ and its minimizer, respectively. According to [4, Theorem 2.4], one can choose violation and confidence levels $\varepsilon, \beta \in (0, 1)$, sample

$$S \geq S(\varepsilon, \beta, d) \tag{3}$$

with $S(\varepsilon, \beta, d) := \min \left\{ N \in \mathbb{N} \middle| \sum_{i=0}^{d-1} \binom{N}{i} \varepsilon^i (1-\varepsilon)^{N-i} \leq \beta \right\}$ points from the constraint set of (1) according to $\mathbb{P}$ and formulate $\text{SCP}[\bar{\Delta}]$ where $\Delta_1 = \cdots = \Delta_M = \bar{\Delta}$ are constructed using the extracted samples. Under Assumption 1, the minimizer of the resulting problem, $x^*[\bar{\Delta}]$, satisfies

$$\text{CCP}_\varepsilon : \mathbb{P}[\exists i \in \{1, \ldots, M\}, \ g_i(x^*[\bar{\Delta}], \delta) > 0] \leq \varepsilon \tag{4}$$

with confidence (measured with respect to $\mathbb{P}^S$) at least $1 - \beta$. The final statement can be compactly written as $\mathbb{P}^S[x^*[\bar{\Delta}] \models \text{CCP}_\varepsilon] \geq 1 - \beta$. The computational complexity associated with constructing $x^*[\bar{\Delta}]$, along with its feasibility properties depend on the choice of ε, β and the number of decision variables d that implicitly affect the number of constraints (inspect (3)). Note that the result remains unaffected if d in (3) is replaced by any upper bound on the number of the so-called support constraints (see [3] for a precise definition) other than the dimension of the decision space. Refinements along this direction are discussed in [8,13,14] where the authors present a tighter bound, defined as the constraint support rank.

3.2. The multi-stage scenario approach

We impose additional structure on the RCP by assuming that for any $\delta \in \Delta$ and each $i = 1, \ldots, M$, the constraint function $g_i(\cdot, \delta)$ does not necessarily depend on all decision variables. The set-up is then similar to the structure considered in [8], where the authors studied optimization programs with multiple chance constraints. For each $i = 1, \ldots, M$, let $\mathcal{X}_i \subseteq \mathcal{X}$ denote the domain of each $g_i(\cdot, \delta)$ and $d_i = \dim(\mathcal{X}_i)$, where $\dim(\mathcal{X}_i)$ denotes the dimension of the smallest subspace of $\mathbb{R}^d$ containing $\mathcal{X}_i$. We further assume that $d_i < d$ for at least one $i = 1, \ldots, M$ to exclude the case where all constraint functions depend on all decision variables; if this is not the case the subsequent analysis reduces to the standard scenario approach of Section 3.1. It was shown in [8, Theorem 4.1] that one can choose different violation and confidence levels $\varepsilon_i, \beta_i \in (0, 1)$ for each $i = 1, \ldots, M$, extract

$$S_i \geq \sum_{i=1}^{M} S(\varepsilon_i, \beta_i, d_i) \tag{5}$$

with

$$S(\varepsilon_i, \beta_i, d_i) := \min \left\{ N \in \mathbb{N} \middle| \sum_{j=0}^{d_i - 1} \binom{N}{j} \varepsilon_i^j (1-\varepsilon_i)^{N-j} \leq \beta_i \right\}$$

samples i.i.d from Δ according to a probability measure $\mathbb{P}$, construct $\{\Delta_i\}_{i=1}^M$ as in Section 2 with $|\Delta_i| = S_i$ and formulate $\text{SCP}[\Delta_1, \ldots, \Delta_M]$. Under Assumption 1, it holds that for each $i = 1, \ldots, M$, the minimizer $x^*[\Delta_1, \ldots, \Delta_M]$ of $\text{SCP}[\Delta_1, \ldots, \Delta_M]$ satisfies the chance constraint,

$$\text{CCP}_{\varepsilon_i} : \mathbb{P}[g_i\left(x^*[\Delta_1, \ldots, \Delta_M], \delta\right) > 0] \leq \varepsilon_i, \tag{6}$$

with confidence (measured with respect to $\mathbb{P}^{S_i}$) at least $1 - \beta_i$. As with the standard scenario approach in Section 3.1, each d_i can be replaced by a tighter upper bound on the support

constraints of g_i. The condition in (6) establishes the feasibility properties of $x^*[\Delta_1, \ldots, \Delta_M]$ for each separate constraint. In the following proposition, we provide guarantees on the probability that $x^*[\Delta_1, \ldots, \Delta_M]$ satisfies all constraints simultaneously, i.e., CCP_ε in (4).

Proposition 1. *Fix $\varepsilon, \beta \in (0, 1)$ and select $\varepsilon_i, \beta_i \in (0, 1)$, for $i = 1, \ldots, M$, such that $\sum_{i=1}^M \varepsilon_i = \varepsilon$ and $\sum_{i=1}^M \beta_i = \beta$. Under the set-up of [8, Theorem 4.1] and Assumption 1 we have $\mathbb{P}^S[x^*[\Delta_1, \ldots, \Delta_M] \models \text{CCP}_\varepsilon] \geq 1 - \beta$, with CCP_ε from (4).*

Proof. The proof of Proposition 1 is an application of the Boole–Bonferroni inequalities [15]. By [8, Theorem 4.1] we have that $\mathbb{P}^S[x^*[\Delta_1, \ldots, \Delta_M] \models \text{CCP}_{\varepsilon_i}] \geq 1 - \beta_i$, for all $i = 1, \ldots, M$. By the subadditivity of $\mathbb{P}^S$ we get $\mathbb{P}^S[x^*[\Delta_1, \ldots, \Delta_M] \models \text{CCP}_{\varepsilon_i}, \text{for all } i = 1, \ldots, M] \geq 1 - \sum_{i=1}^M \beta_i = 1 - \beta$. To complete the proof it suffices to show that $x^*[\Delta_1, \ldots, \Delta_M] \models \text{CCP}_{\varepsilon_i}$ for all $i = 1, \ldots, M$, implies that $x^*[\Delta_1, \ldots, \Delta_M] \models \text{CCP}_\varepsilon$, where CCP_ε is given in (4). By the subadditivity of $\mathbb{P}$, and since the statement $x^*[\Delta_1, \ldots, \Delta_M] \models \text{CCP}_{\varepsilon_i}$ is equivalent to the statement $\mathbb{P}[g_i(x^*[\Delta_1, \ldots, \Delta_M], \delta) > 0] \leq \varepsilon_i$, we have that $\mathbb{P}[\exists i \in \{1, \ldots, M\} \text{such that } g_i(x^*[\Delta_1, \ldots, \Delta_M], \delta) > 0] \leq \sum_{i=1}^M \varepsilon_i = \varepsilon$. Since $g(x, \delta) := \max_{i=1,\ldots,M} g_i(x, \delta)$, the last statement implies that $\mathbb{P}[g(x^*[\Delta_1, \ldots, \Delta_M], \delta) > 0] \leq \varepsilon$, and hence $x^*[\Delta_1, \ldots, \Delta_M] \models \text{CCP}_\varepsilon$. □

Note that the proof of Proposition 1 is based on measure subadditivity, hence the result is conservative. The complexity associated with obtaining $x^*[\Delta_1, \ldots, \Delta_M]$ with the properties of Proposition 1, depends on $\{d_i\}_{i=1}^M$ and the choices for $\{\varepsilon_i\}_{i=1}^M$, $\{\beta_i\}_{i=1}^M$. We elaborate on these choices in Section 3.5.

3.3. Stage-wise coupled scenario approach without re-sampling

We now consider RCP problems with constraint functions that allow tackling $\text{SCP}[\Delta_1, \ldots, \Delta_M]$ in a sequential manner. We assume that for $i = 1, \ldots, M-1$ the constraint functions $g_i(\cdot, \cdot, \cdot) : \mathcal{X}_i \times \mathcal{X}_{i+1} \times \Delta \to \mathbb{R}$ are pairwise coupled and convex with respect to their first argument but may have an arbitrary, possibly non-convex, dependency with respect to the other arguments. For $i = M$, $g_M(\cdot, \cdot) : \mathcal{X}_M \times \Delta \to \mathbb{R}$ is not coupled and is assumed to be convex on $\mathcal{X}_M$. The pairwise coupling structure can be relaxed to any form of stage-wise coupling as long as the constraint function at every stage is convex with respect to the decision variables. Consequently, we have by construction of RCP that $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_M$, which is a special case of the structure assumed in Section 3.2. Let $x = (x_1, \ldots, x_M)$ where $x_i \in \mathcal{X}_i$, for each $i = 1, \ldots, M$. We further assume that the objective function is separable, i.e., $f(x) = \sum_{i=1}^M f_i(x_i)$. Since we are only concerned with the feasibility properties of the associated scenario program, the assumption on the objective function is not restrictive; any separable function can be employed at the expense of a suboptimal solution. A similar approach of selecting arbitrary cost functions when only concerned with feasibility properties has been employed in [12]. As will be demonstrated in Section 5, ADP algorithms based on [10] exhibit this structure and cover a wide range of control problems that can be solved via dynamic programming.

The separable structure assumed, motivates the decomposition of $\text{SCP}[\Delta_1, \ldots, \Delta_M]$ into a sequence of coupled scenario programs. For each $i = 1, \ldots, M-1$ we define a parametric scenario program:

$$\text{SCP}_i[x_{i+1}, \Delta_i] : \begin{cases} \min\limits_{x_i \in \mathcal{X}_i} & f_i(x_i) \\ \text{s.t} & g_i(x_i, x_{i+1}, \delta) \leq 0, \quad \forall \delta \in \Delta_i \end{cases} \tag{7}$$

and $\text{SCP}_M[\Delta_M]$ analogously, with $g_M(x_M, \delta) \leq 0$ for all $\delta \in \Delta_M$, replacing the corresponding constraint in (7). We assume that all stage problems in (7) satisfy Assumption 1 for any fixed $x_{i+1} \in \mathcal{X}_{i+1}$; weaker assumptions are discussed in [16, Section IV]. Note that the indexing chosen to formulate $\text{SCP}_i[x_{i+1}, \Delta_i]$ is arbitrary; one can define an identical sequence of problems starting with a problem depending only on $x_1 \in \mathcal{X}_1$, letting SCP_i depend on x_{i-1} and Δ_i. In this way the structure resembles that of stochastic model predictive control (SMPC) problems where the system dynamics couple the constraints of stage i with the solution (system state and control input) of stage $i-1$ [13]. However, since each stage is handled separately, the approach is myopic and may cause suboptimal performance by not exploiting the coupling when minimizing the total cost. It does, however, achieve significantly reduced computational times (see Sections 4 and 5) and may be preferred in case of computational time limitations or of problems with inaccurate prediction models. The situation is different in ADP algorithms based on [10] which exhibit a structure that naturally fits the framework of (7) without sacrificing optimality.

Consider now the sequence of M pairwise coupled programs $\text{SCP}_i[x_{i+1}, \Delta_i]$ and let all sets Δ_i be identical, i.e., $\Delta_1 = \cdots = \Delta_M = \bar{\Delta}$. Such optimization problems were referred to as cascading programs in [16], where the authors study the feasibility properties of a solution generated by sequentially solving a pair of coupled problems using the same set of uncertain scenarios. Let $d_i = \dim(\mathcal{X}_i)$ be the dimension of the smallest subspace of $\mathbb{R}^d$ containing $\mathcal{X}_i$ and $\bar{d} = \sum_{i=1}^M d_i$. As a direct consequence of [16, Theorem 5], we infer that one can select overall violation and confidence levels $\varepsilon, \beta \in (0, 1)$ and a sample number $S \geq S(\varepsilon, \beta, \bar{d})$, where $S(\varepsilon, \beta, \bar{d})$ is given by (3) and construct $x^* := (x_1^*, \ldots, x_M^*)$, where each $x_i^*[\bar{\Delta}]$ is computed from (7) with $\Delta_i = \bar{\Delta}$ for $i = 1, \ldots, M$. We then have that $\mathbb{P}^S[x^*[\bar{\Delta}] \models \text{CCP}_\varepsilon] \geq 1 - \beta$. Moreover, if $x^* := (x_1^*, \ldots, x_M^*)$ is constructed in this way, for any fixed $x_{i+1} \in \mathcal{X}_{i+1}$, with probability at least $1 - \beta_i$, $x_i^*[x_{i+1}, \bar{\Delta}]$ satisfies $\mathbb{P}[g_i(x_i^*[x_{i+1}, \bar{\Delta}], x_{i+1}, \delta) > 0] \leq \varepsilon_i$, for any $\varepsilon_i, \beta_i \in (0, 1)$ satisfying the equation $S(\varepsilon_i, \beta_i, d_i) \leq S$. Note that, in general, no ordering between $S(\beta, \varepsilon, \bar{d})$, $S(\varepsilon_i, \beta_i, d_i)$ can be made. Moreover, the values of ε_i and β_i are not set a-priori and are not design choices; they are implicitly determined by the number of decision variables d_i of each subproblem and the number of samples S. Consequently, the computational complexity of the stage-wise coupled scenario approach without re-sampling only depends on $\bar{d}$ and the choice of ε, β. As in Sections 3.1 and 3.2, each d_i can be replaced by a tighter upper bound on the support constraints of g_i.

3.4. Stage-wise coupled scenario approach with re-sampling

Consider the structure of Section 3.3 and note that for a fixed $x_{i+1} \in \mathcal{X}_{i+1}$, $\text{SCP}_i[x_{i+1}, \Delta_i]$ is in the form of $\text{SCP}[\bar{\Delta}]$ considered in Section 3.1. Fix ε_i and β_i and let the number of samples S_i, $i = 1, \ldots, M$ be chosen according to (5). For any $x_{i+1} \in \mathcal{X}_{i+1}$, $\Delta_i \in \Delta^{S_i}$, let $x_i^*[x_{i+1}, \Delta_i] : \mathcal{X}_{i+1} \times \Delta^{S_i} \to \mathcal{X}_i$ be the minimizer of $\text{SCP}_i[x_{i+1}, \Delta_i]$. The result in Section 3.1 implies that for all $i = 1, \ldots, M-1$, $x_i^*[x_{i+1}, \Delta_i]$ satisfies the chance constraint $\text{CCP}_{\varepsilon_i}[x_{i+1}] : \mathbb{P}[g_i(x_i^*[x_{i+1}, \Delta_i], x_{i+1}, \delta) > 0] \leq \varepsilon_i$, with probability at least $1-\beta_i$, while for $i = M$, $x_M^*[\Delta_M]$ satisfies $\text{CCP}_{\varepsilon_M}$ with probability at least $1 - \beta_M$.

Using the parameterized problems in (7) we can construct a decision vector $x^* := (x_1^*, \ldots, x_M^*)$, where for each $i = 1, \ldots, M-1$ the optimizer $x_i^*[x_{i+1}, \Delta_i]$ can be written as $x_i^*[\Delta_i, \ldots, \Delta_M] : \Delta^{S_i} \times \cdots \times \Delta^{S_M} \to \mathcal{X}_i$, satisfying

$$\mathbb{P}^{S_i}\left[x_i^*[\Delta_i, \ldots, \Delta_M] \models \text{CCP}_{\varepsilon_i}\left[x_{i+1}^*[\Delta_{i+1}, \ldots, \Delta_M]\right]\right] \geq 1 - \beta_i, \tag{8}$$

and for $i = M$, $\mathbb{P}^{S_M}[x^*_M \models \text{CCP}_{\varepsilon_M}] \geq 1 - \beta_M$. Note that due to the stage-wise coupled process, x^*_i depends implicitly on all sets $\Delta_i, \ldots, \Delta_M$.

Proposition 2. *Fix $\varepsilon, \beta \in (0, 1)$ and select $\varepsilon_i, \beta_i \in (0, 1)$, for $i = 1, \ldots, M$, such that $\sum_{i=1}^M \varepsilon_i = \varepsilon$ and $\sum_{i=1}^M \beta_i = \beta$. Construct $x^* := (x^*_1, \ldots, x^*_M)$, where each $x^*_i[\Delta_i, \ldots, \Delta_M]$ is computed from (7), satisfying (8). Let $S = \sum_{i=1}^M S_i$ with $\{S_i\}_{i=1}^M$ chosen according to (5). We then have that $\mathbb{P}^S[x^*[\Delta_1, \ldots, \Delta_M] \models CCP_\varepsilon] \geq 1 - \beta$.*

Proof. Let $\bar{S}_i = \sum_{k=i}^M S_k$, $\bar{\varepsilon}_i = \sum_{k=i}^M \varepsilon_k$, $\bar{\beta}_i = \sum_{k=i}^M \beta_k$, $\bar{\Delta}_i = (\Delta_i, \ldots, \Delta_M)$ and $\bar{x}^*_i = (x^*_i, \ldots, x^*_M)$. The statement $\mathbb{P}^S[x^*[\Delta_1, \ldots, \Delta_M] \models \text{CCP}_\varepsilon] \geq 1 - \beta$, is then equivalent to

$$\mathbb{P}^{\bar{S}_i}\Big[\mathbb{P}\big[g_M(x^*_M[\bar{\Delta}_M], \delta) > 0 \quad \text{or} \quad \exists k \in \mathbb{N}_+, i \leq k < M : g_k(x^*_k[\bar{\Delta}_k], x^*_{k+1}[\bar{\Delta}_{k+1}], \delta) > 0\big] \leq \bar{\varepsilon}_i\Big] \geq 1 - \bar{\beta}_i \tag{9}$$

for $i = 1$. We will show that (9) holds for any $i = 1, \ldots, M$ using induction. For $i = M$, (9) holds since $\bar{\Delta}_M = \Delta_M$ and $\text{SCP}_M[\Delta_M]$ is in the form considered by [4, Theorem 2.4]. Assume now that (9) holds for some index i where $1 < i < M$. The statement in (8) is then equivalent to

$$\mathbb{P}^{S_{i-1}}\Big[\mathbb{P}\big[g_{i-1}(x^*_{i-1}[\bar{\Delta}_{i-1}], x^*_i[\bar{\Delta}_i], \delta) > 0\big] \leq \varepsilon_{i-1}\Big] \geq 1 - \beta_{i-1}. \tag{10}$$

Since samples are extracted independently, and (9), (10), hold for any uncertainty realization not in $\bar{\Delta}_i$ and $\bar{\Delta}_{i-1}$, respectively, we can replace $\mathbb{P}^{\bar{S}_i}$ and $\mathbb{P}^{S_{i-1}}$ with $\mathbb{P}^{\bar{S}_{i-1}}$. By the subadditivity of $\mathbb{P}^{\bar{S}_{i-1}}$ and $\mathbb{P}$, we can show analogously to the proof of Proposition 1 that (9) holds with $i - 1$ in place of i which is equivalent to $\mathbb{P}^{\bar{S}_{i-1}}[\bar{x}^*_{i-1} \models \text{CCP}_{\bar{\varepsilon}_{i-1}}] \geq 1 - \bar{\beta}_{i-1}$ and concludes the inductive claim. □

The computational complexity associated with obtaining $x^*[\Delta_1, \ldots, \Delta_M]$ with the feasibility properties of Proposition 2 depends on $\{d_i\}_{i=1}^M$ and the choices for $\{\varepsilon_i\}_{i=1}^M$, $\{\beta_i\}_{i=1}^M$. Again, the result remains the same if the value of each d_i is replaced by a tighter upper bound than the dimension of the relevant decision space. Notice that unlike the stage-wise coupled scenario approach without re-sampling, $\{\varepsilon_i\}_{i=1}^M$ and $\{\beta_i\}_{i=1}^M$ are again design parameters for $i = 1, \ldots, M$ and can be chosen to reduce the computational complexity of the algorithm used to solve the corresponding optimization problems (see Section 3.5).

3.5. Complexity optimization

The complexity of solving optimization problems depends on the number of decision variables and constraints. For Sections 3.2 and 3.4, the overall violation and confidence are chosen a priori but the stage-wise levels $\{\varepsilon_i\}_{i=1}^M$ and $\{\beta_i\}_{i=1}^M$ are typically not fixed by problem data and are design choices that affect the computational complexity due to the polynomial dependence of solvers on the total number of constraints and decision variables. Since the values of $\{d_i\}_{i=1}^M$ are fixed by problem data and generating samples from Δ can be hard, we minimize the total number of samples (and hence constraints) as an approximation to minimizing the total complexity. Throughout this section we replace the implicit sample size bound used in (5) to upper bound the required sample size by the explicit bound $\frac{e}{e-1}\frac{1}{\varepsilon_i}(d_i - 1 + \ln(\frac{1}{\beta_i}))$ due to [17]. For simplicity, we treat the right-hand-side as an integer. Using this explicit bound is more conservative than employing the implicit one and performing numerical inversion to compute the required number of samples [11], as demonstrated in Section 5. It allows however to select violation and confidence levels by means of a standard form convex optimization program (Proposition 4). Note that the resulting values are not guaranteed to remain optimal if the implicit bound is considered instead.

Proposition 3. *Consider the setup of Sections 3.2 and 3.4 where for each $i = 1, \ldots, M$ the values of $d_i = \dim(\mathcal{X}_i)$ and $d = \dim(\mathcal{X})$ are fixed by the problem data. Fix $\varepsilon, \beta \in (0, 1)$. The problem of selecting $\{\varepsilon_i, \beta_i \in (0, 1)\}_{i=1}^M$ with $\sum_{i=1}^M \varepsilon_i \leq \varepsilon$, $\sum_{i=1}^M \beta_i \leq \beta$ that minimize the sample number $\sum_{i=1}^M \frac{e}{e-1}\frac{1}{\varepsilon_i}(d_i - 1 + \ln(\frac{1}{\beta_i}))$, is a convex optimization program of the form:*

$$\min_{\{\varepsilon_i, \beta_i\}_{i=1}^M} \quad \sum_{i=1}^M \frac{e}{e-1}\frac{1}{\varepsilon_i}\left(d_i - 1 + \ln\left(\frac{1}{\beta_i}\right)\right)$$
$$\text{subject to :} \quad \left.\begin{array}{l} \sum_{i=1}^M \varepsilon_i \leq \varepsilon, \\ \sum_{i=1}^M \beta_i \leq \beta, \\ \varepsilon_i, \beta_i > 0. \end{array}\right\} \forall i \in \{1, \ldots, M\}. \tag{11}$$

Proof. The function $\frac{e}{e-1}\frac{1}{\varepsilon_i}(d_i - 1 + \ln(\frac{1}{\beta_i}))$ is convex with respect to ε_i, β_i since the Hessian matrix is positive definite for any $\varepsilon_i, \beta_i \in (0, 1)$. Hence, the cost function is the sum of convex functions. □

Note that in place of d_i in (11) one could employ the support rank [8]. In that case, the sample size computed via Proposition 3 will not be higher than the one computed in [8]. The objective function of problem (11) is not in a standard form compatible with all commercially available optimization software. As a result, one needs to implement a first or second order method to solve (11) (e.g., [18]) taking advantage of the fact that both the gradient and Hessian matrix of the objective function are bounded with respect to ε_i, β_i in $[\mu, 1)$ for any $\mu > 0$. Alternatively, a nonlinear optimization solver should be employed. It is, however, more likely for such solvers to exhibit numerical issues, in certain cases even failing to identify the global optimum. We thus relax (11) to a problem that is in standard SDP form, allowing the use of efficient solvers for semi-definite programming. To achieve this we fix the confidence levels β_i, e.g., $\beta_i = \beta/M$, for all $i = 1, \ldots, M$. The performance deterioration incurred by fixing β_i, $i = 1, \ldots, M$, is negligible, since they are typically set to values very close to zero and appear inside the logarithm in (11), hence their difference from the optimal values returned by the optimization program in (11) will have a minor effect on the total number of samples that need to be extracted.

Proposition 4. *Choose $\beta \in (0, 1)$ and fix the stage-wise confidence levels $\{\beta_i \in (0, 1)\}_{i=1}^M$ such that $\sum_{i=1}^M \beta_i = \beta$. Fix $\varepsilon \in (0, 1)$. For $c_i = \frac{e}{e-1}(d_i - 1 + \ln(\frac{1}{\beta_i}))$, $i = 1, \ldots, M$, the following SDP is equivalent to (11)*

$$\min_{\{t_i, \varepsilon_i\}_{i=1}^M} \quad \sum_{i=1}^M t_i$$
$$\text{subject to:} \quad \left.\begin{array}{l} \begin{bmatrix} t_i & \sqrt{c_i} \\ \sqrt{c_i} & \varepsilon_i \end{bmatrix} \succcurlyeq 0, \\ \sum_{i=1}^M \varepsilon_i \leq \varepsilon, \ \varepsilon_i > 0. \end{array}\right\} \quad \forall i \in \{1, \ldots, M\}. \tag{12}$$

Proof. The objective function in (11) can be written as $\sum_{i=1}^M c_i/\varepsilon_i$. Writing the problem in standard epigraph form and using Schur's complement we end up with the constraints in (12). □

4. Discussion and trade-offs

4.1. Structure and feasibility properties

In contrast to the standard scenario approach of Section 3.1, the multi-stage variant of Section 3.2 assumes that the domain

Table 1
Complexity characteristics of the methods presented in Sections 3.1–3.4.

	Section 3.1	Section 3.2	Section 3.3	Section 3.4
Number of problems	1	1	M	M
Samples per problem	$S \sim (3)$	$S_i \sim (5)$	$S \sim (3)$, $d = \sum_{i=1}^{M} d_i$	$S_i \sim (5)$
Total number of samples	S	$\sum_{i=1}^{M} S_i$	S	$\sum_{i=1}^{M} S_i$
Decision variables per problem	$d = \dim(\mathcal{X})$	$d = \dim(\mathcal{X})$	$d_i = \dim(\mathcal{X}_i)$	$d_i = \dim(\mathcal{X}_i)$
Constraints per problem	MS	$\sum_{i=1}^{M} S_i$	S	S_i
Total complexity (SOCP)	$\mathcal{O}((d+MS)^3)$	$\mathcal{O}\big((d+\sum_{i=1}^{M} S_i)^3\big)$	$\sum_{i=1}^{M} \mathcal{O}\big((d_i+S)^3\big)$	$\sum_{i=1}^{M} \mathcal{O}\big((d_i+S_i)^3\big)$

of each constraint function in RCP is restricted to a subset of $\mathcal{X}$. By investigating each constraint separately, the latter method provides guarantees on the probability that $x^*[\Delta_1, \ldots, \Delta_M]$ satisfies every individual constraint, something that cannot be achieved with the standard scenario approach. In the stage-wise coupled methodologies of Sections 3.3 and 3.4 we further restrict the structure of RCP by requiring the constraint functions to be pairwise coupled. In this way we relax the assumption regarding the convexity, requiring $g_i(x_i, x_{i+1}, \delta)$ to be convex only with respect to x_i.

All results in Section 3 lead to feasibility statements in the form of $\mathbb{P}^S[x^*[\Delta_1, \ldots, \Delta_M] \models \text{CCP}_\varepsilon] \geq 1-\beta$. Each method however requires a different number of samples to construct a solution and in turn the space on which the confidence related to the probability of constraint satisfaction is measured differs. In the standard scenario approach the total number of samples S is fixed by the value of d and the choice of violation and confidence levels ε, β (inspect (3)). Assuming the same choice of ε and β, the total number of samples in the multi-stage scenario approach $\sum_{i=1}^{M} S_i$ can be greater or less than S depending on the values of $\{d_i\}_{i=1}^{M}$ (inspect (5) and the first two columns in Table 1). In general, if each d_i is significantly smaller than d, the total number of samples is smaller in the multi-stage scenario approach. The situation is analogous between the stage-wise coupled scenario approach without and with re-sampling, where the total number of samples will be generally higher in the latter depending on the values of $\{d_i\}_{i=1}^{M}$ and the choices of $\{\varepsilon_i\}_{i=1}^{M}$, $\{\beta_i\}_{i=1}^{M}$ (see the last two columns in Table 1). Note that for the multi-stage scenario approach and the stage-wise coupled scenario approach with re-sampling we can use the methods of Section 3.5 to optimize over $\{\varepsilon_i\}_{i=1}^{M}$ and $\{\beta_i\}_{i=1}^{M}$ but there is no guarantee that this will lead to a smaller number of total samples since $\{d_i\}_{i=1}^{M}$ is fixed by problem data.

4.2. Complexity

We compare the computational complexity of different methods when a primal–dual interior point algorithm is used to solve second-order cone programs (SOCPs). Methods designed to address such problems require solving linear systems of size $(n+m)^3$ that are known to be of $\mathcal{O}\left((n+m)^3\right)$ complexity [19], where n, m denote the number of decision variables and constraints. The complexity of each method is reported in Table 1. Both the standard and multi-stage scenario approach of Sections 3.1 and 3.2 require solving a single problem with the same number of decision variables but a potentially different number of constraints. The number of decision variables d is given by problem data, while the number of constraints depends on d, $\{d_i\}_{i=1}^{M}$ and the chosen ε, β and $\{\varepsilon_i\}_{i=1}^{M}$, $\{\beta_i\}_{i=1}^{M}$. In the standard scenario approach, we use the same samples S (see (3)) for each constraint function leading to a total of MS constraints. In the multi-stage scenario approach, we use different samples S_i (see (5)) for each constraint function leading to $\sum_{i=1}^{M} S_i$ constraints, a number that can be minimized over $\{\varepsilon_i\}_{i=1}^{M}$, $\{\beta_i\}_{i=1}^{M}$ using the methods of Section 3.5. Depending on the ratio between the minimum value of $\sum_{i=1}^{M} S_i$ and MS, either of the two methods might be preferable. Note that, in contrast to other studies, we measure the computational efficiency of each method based on the complexity of the underlying algorithm, and not on the relevant sample size.

The computational complexity of the methodologies in Sections 3.3 and 3.4 depends on the number of decision variables and constraints per subproblem. Each subproblem involves a single constraint function and hence the number of samples coincides with the number of constraints. In the stage-wise coupled scenario approach without re-sampling, we use the same number of samples S in every subproblem which depends on $\bar{d} = \sum_{i=1}^{M} d_i$ and the choice of ε, β (see Section 3.3). If $\bar{d} = d$ (for example in some ADP problems, see Section 5), the number of samples coincides with that of the standard scenario approach. In general however, it might be that $\bar{d} > d$ (as is the case, for example, in some SMPC problems). In the stage-wise coupled scenario approach with re-sampling, the number of samples S_i in each subproblem coincides with the number of samples used in the multi-stage scenario approach and depends on d_i and the choice of ε_i, β_i (see Proposition 2). As in the multi-stage scenario approach, $\sum_{i=1}^{M} S_i$ can be minimized over $\{\varepsilon_i\}_{i=1}^{M}$, $\{\beta_i\}_{i=1}^{M}$ using the methods in Section 3.5. Whenever applicable, the stage-wise coupled methods can provide significant computational advantages, as illustrated in the next section.

5. Example: approximate dynamic programming

Dynamic programming (DP) recursions are generally intractable even for systems of moderate dimensions. An established methodology for approximate dynamic programming (ADP) is the linear programming approach [10] which projects the value function on the span of a selected set of functions. The authors in [20,21] developed an algorithm along the lines of Section 3.4 to approximate the value function of stochastic reachability DP recursions. We use this algorithm here and compare with the alternative formulations from Sections 3.1–3.3. We consider a simplified planar unicycle model with additive noise

$$\begin{bmatrix}\delta_1(i+1)\\ \delta_2(i+1)\end{bmatrix} = \begin{bmatrix}\delta_4(i)\cos(\delta_3(i)) + \delta_1(i)\\ \delta_4(i)\sin(\delta_3(i)) + \delta_2(i)\end{bmatrix} + w_i \tag{13}$$

where δ_1, δ_2 denote linear position, δ_3 yaw angle and δ_4 linear velocity. We use δ_3 and δ_4 as control inputs to the system and δ_1 and δ_2 as states. The noise $w_i \in \mathbb{R}^2$ is assumed to be i.i.d according to a multivariate normal distribution $\mathcal{N}(0, \Sigma)$ with diagonal covariance matrix. The combined state-action space is denoted by $\Delta = \Delta_x \times \Delta_u = \mathbb{R}^2 \times ([-0.5, 0.5] \times [-2\pi, 2\pi])$ where for $\delta = (\delta_1, \delta_2, \delta_3, \delta_4) = (\delta_x, \delta_u) \in \Delta$, δ_x corresponds to spatial coordinates while δ_u to control inputs. The symbol δ is used for the state-input variables to match the notation of Section 3. For a target set $T = [0.8, 1]^2$, an avoid set $A = [-0.45, 0.25] \times$

Table 2
Results of ADP for reach-avoid using the methods presented in Sections 3.1–3.4.

	Step	ε	$\hat{\varepsilon}$	$1-\beta$	d	Constraints	Constraints (implicit)	Optimal value	Solver (s)	Sampling (s)
Section 3.1	$i=3$	–	0.017	–	–	–	–	0.903e+03	–	–
	$i=2$	–	0.032	–	–	–	–	0.396e+03	–	–
	$i=1$	–	0.049	–	–	–	–	0.121e+03	–	–
	Overall	**0.1**	0.093	**0.97**	450	21 528	14 718	1.42e+03	73	0.509
Section 3.2	$i=3$	**0.023**	0.021	**0.99**	100	7 126	5 419	1.433e+03	–	0.546
	$i=2$	**0.035**	0.026	**0.99**	250	11 463	8 229	2.422e+03	–	0.797
	$i=1$	**0.042**	0.030	**0.99**	350	13 319	9 397	2.936e+03	–	0.879
	Overall	**0.1**	0.076	**0.97**	450	31 995	23 045	6.791e+03	105	2.22
Section 3.3	$i=3$	–	0.009	–	100	7 159	4 906	0.202e+03	1.15	–
	$i=2$	–	0.036	–	150	7 159	4 906	0.599e+03	2.05	–
	$i=1$	–	0.049	–	200	7 159	4 906	1.225e+03	6.91	–
	Overall	**0.1**	0.069	**0.97**	–	–	–	2.026e+03	10.11	0.4812
Section 3.4	$i=3$	**0.0276**	0.019	**0.99**	100	5 938	4 515	4.32e+03	1.12	0.399
	$i=2$	**0.0336**	0.033	**0.99**	150	7 232	5 351	2.83e+03	4.066	0.473
	$i=1$	**0.0387**	0.036	**0.99**	200	8 323	6 050	2.23e+03	7.57	0.538
	Overall	**0.1**	0.08	**0.97**	–	–	–	9.38e+03	12.756	1.41

$[-0.2, 0.15]$ and a collection of time indexed safe sets defined as $\{S_i\}_{i=1}^3 = \{[-1, 1]^2, [-0.3, 1]^2, [0.4, 1]^2\}$, the three step reach-avoid problem is to maximize the probability that (13) reaches T while staying in the corresponding safe region $S_i \setminus A$ for $i = 1, 2, 3$ (see Fig. 1). It can be solved for all states via the DP recursion [22]:

$$V_i^*(\delta_x) = \sup_{\delta_u \in \Delta_u} \left\{ \underbrace{\mathbb{1}_T(\delta_x) + \mathbb{1}_{(S_i \setminus A) \setminus T}(\delta_x) \int_{\Delta_x} V_{i+1}^*(y) Q(dy|\delta)}_{h(\delta_x, \delta_u)} \right\}$$

initialized by $V_4^*(\delta_x) = \mathbb{1}_T(\delta_x)$ where V_i^* denotes the value function at stage i, Q denotes the transition kernel of the process in (13) and $\mathbb{1}_T$, $\mathbb{1}_{(S_i \setminus A) \setminus T}$ denote the indicator functions of the sets T and $(S_i \setminus A) \setminus T$ respectively. We follow the ADP method for reach-avoid problems from [20,21] to approximate the recursion. We express each V_i^* as the solution to an infinite dimensional linear program:

$$\begin{aligned} V_i^* \in \arg \quad & \inf_{V(\cdot) \in \mathcal{F}} \quad \int_{\Delta_x} V(\delta_x) \nu(\mathrm{d}\delta_x) \\ & \text{subject to} \quad V(\delta_x) \geq h(\delta_x, \delta_u), \ \forall \delta \in \Delta \end{aligned} \tag{14}$$

where ν is a (positive) measure supported on Δ_x and $\mathcal{F}$ denotes the space of Borel-measurable functions in which V_i^* resides [20]. Problems in the form of (14) are generally intractable and it is common in the literature to restrict the decision space to a finite dimensional subspace of $\mathcal{F}$ to approximate V_i^*. We restrict the decision space to Gaussian radial basis functions (RBFs) with fixed centers and variances and use their span to approximate each V_i^*. Let $\{d_i\}_{i=1}^3 = \{200, 150, 100\}$ denote the cardinality of each basis set over the horizon and $x = \{x_i\}_{i=1}^3$ with $x_i \in \mathbb{R}^{d_i}$ vectors corresponding to the weights of each RBF in the set. The reduction in the number of basis elements over time is motivated by the shrinking of the safe sets S_i (Fig. 1). We denote by $L_i : \mathbb{R}^{d_i} \times \mathbb{R}^{d_{i+1}} \times \Delta \to \mathbb{R}$ the functions (linear in the first and second arguments) that for $i = 1, 2$ and each $\delta \in \Delta$ return the difference between the approximate value function at time i and the one-step-ahead reward at time $i + 1$ (see (14)). Each L_i implicitly depends on the safe, avoid and target sets at time i and the weights x_i, x_{i+1} completely determine its value over Δ. For $i = 3$, the function is defined as $L_3 : \mathbb{R}^{d_3} \times \Delta \to \mathbb{R}$ since the reach-avoid value function at $i = 4$ is known. The approximate reach-avoid value functions can be then computed via a sequence of programs:

$$\begin{aligned} \min_{x_i \in \mathbb{R}^{d_i}} \quad & x_i^\top I_i \\ \text{subject to:} \quad & L_i(x_i, x_{i+1}, \delta) \geq 0, \quad \forall \delta \in \Delta \end{aligned} \tag{15}$$

where I_i denotes the element-wise ν-weighted integral over Δ_x of each RBF in the basis set. The problems in (15) can be combined:

$$\begin{aligned} \min_{x \in \mathbb{R}^{d_1+d_2+d_3}} \quad & \sum_{i=1}^3 x_i^\top I_i \\ \text{subject to:} \quad & L_i(x_i, x_{i+1}, \delta) \geq 0, \quad \forall \delta \in \Delta, \ i = 1, 2 \\ & L_3(x_3, \delta) \geq 0, \quad \forall \delta \in \Delta. \end{aligned} \tag{16}$$

For the implementation, we used d_i to bound the number of support constraints since one can show that the support rank of each constraint is equal to the number of decision variables using the linear independence between basis functions [8, Example 3.5(b)]. Using the optimal weights, we construct the approximate value function of the stochastic reach-avoid problem for each $i = 1, 2, 3$.

We solved the problem with all methods and Table 2 compares the theoretical guarantees (ε) with the empirical ones ($\hat{\varepsilon}$) and the associated complexities ("Sampling" and "Solver"). The empirical violation values were calculated by uniformly sampling 1000 realizations from Δ, other than those used in the optimization process, and computing the ratio of violations to 1000. The bold parameters can be chosen by the user and are not fixed by problem data; for the multi-stage scenario approach and the stage-wise coupled scenario approach with re-sampling, the violation levels ε_i at each stage are chosen using (12). The associated confidence levels $1 - \beta_i$, $i = 1, 2, 3$ were fixed to 0.99 to achieve an overall confidence $1 - \beta$ of at least 0.97. The basis centers and variances were sampled uniformly from each safe set and $(0, 0.01]$ respectively. Computations were carried out on a 1.73 GHz CPU with 16 GB of memory, using the Gurobi software. Fig. 1 shows level sets of the approximation at $i = 1$ restricted on $[-1, 1]^2$, constructed using the stage-wise coupled scenario approach with re-sampling. The values of the approximation go above 1 since the algorithm in [20] provides an upper bound to the reach-avoid probability.

Table 2 suggests that for ADP it is favorable to solve problems in a stage-wise coupled manner since the same overall violation levels are respected while computation times are smaller. The number of constraints is calculated as the number of constraints for a given δ multiplied by the number of scenarios/realizations of δ. Consider for example the table entries corresponding to Section 3.1. Since problem (16) has three stages and one scalar-valued constraint per stage, we have three constraints for a given δ. According to ε, β and d reported in the corresponding entries of Table 2, 7176 samples were generated. The total number of constraints is then $3 \times 7176 = 21\,528$. The structure exhibited by the problem in the standard scenario approach, where we have the same samples for all stages, appears to be exploited by the

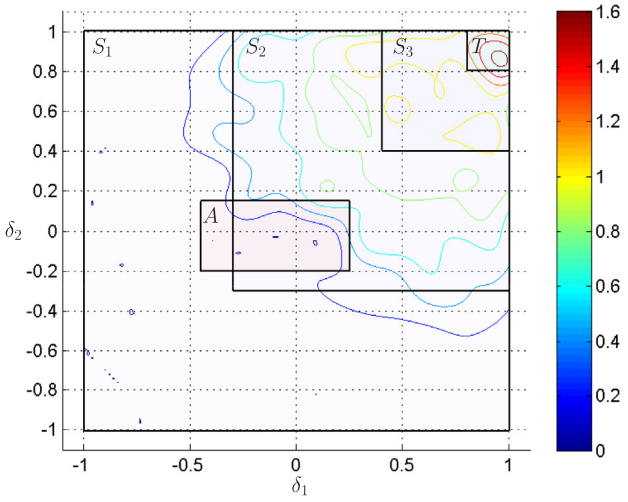

Fig. 1. Level sets of the approximate value function at $i = 1$ restricted on $[-1, 1]^2$, constructed using the method in Section 3.4.

solver when comparing to the multi-stage scenario approach (both methods solve the problem in one-shot) where different sample numbers are used for each constraint by employing the different support rank of the constraint of each stage. Sampling each constraint separately also consumes more time as observed in the multi-stage and stage-wise coupled approach with re-sampling. Differences in the reported sampling times are a consequence of the hit and run algorithm used to generate them [23]. We have also reported the sample numbers computed by inverting the implicit bound (see (3), (5)) to highlight the conservatism introduced by using the explicit bound in Section 3.5. Note that the same feasibility guarantees will hold if one uses the sample number obtained by inverting the implicit bound using the optimizers computed with the methods of Section 3.5. However, the optimal violation and confidence levels that minimize complexity may be different if the implicit bound is used in the corresponding complexity optimization problems.

From the optimal values reported by the solver, we notice that the stage-wise coupled scenario approach without re-sampling is achieving the best complexity/optimal value trade-off since it requires smaller problem instances than the standard and multi-stage variants while exploring more of the uncertainty set at each horizon step than the stage-wise coupled scenario approach with re-sampling. In terms of performance alone, the standard scenario approach achieves the best value since it optimizes over all weights in one-shot, as opposed to the stage-wise coupled methods, and the samples used for every step are taken from the union of all safe sets, as opposed to the multi-stage and stage-wise coupled approach with re-sampling where we only use samples from the safe set of the corresponding time-steps. All numbers have been averaged over 10 runs of each method.

6. Conclusion

We introduced an alternative way of dealing with a sequence of scenario programs with coupled constraints and compared its feasibility properties with the literature. Moreover, we showed how confidence and violation levels can be treated as optimization assets and selected by means of convex optimization problems to reduce computation time. We verified with a numerical example that the stage-wise coupled structure, often encountered in sequential decision making, can lead to computational savings.

Current work involves probabilistic performance considerations, extending the results of [5–7] to the class of scenario programs considered in this paper. Another promising research direction is developing a constructive algorithm to decompose the constraints of a robust convex problem into a sequence of stage-wise coupled constraints enabling one to take advantage of the proposed methodologies. In terms of applications, we expect similar computational advantages in stochastic model predictive control problems and more generally in control problems for multi-agent systems where the decisions of one agent depend on the decisions of another; in such cases using different samples between agents can have a significant impact on the required communication bandwidth.

Acknowledgment

The work of N. Kariotoglou was supported by the Swiss National Science Foundation under grant number 200021_137876.

References

[1] A. Ben-Tal, L. El Ghaoui, A. Nemirovski, Robust Optimization, Princeton University Press, 2009.
[2] D. Bertsimas, M. Sim, Tractable approximations to robust conic optimization problems, Math. Program. 107 (1–2) (2006) 5–36.
[3] G.C. Calafiore, M.C. Campi, The scenario approach to robust control design, IEEE Trans. Automat. Control 51 (5) (2006) 742–753.
[4] M.C. Campi, S. Garatti, The exact feasibility of randomized solutions of uncertain convex programs, SIAM J. Optim. 19 (3) (2008) 1211–1230.
[5] A. Lecchini, J. Lygeros, J. Maciejowski, Stochastic optimization on continuous domains with finite-time guarantees by Markov chain Monte Carlo methods, IEEE Trans. Automat. Control 55 (12) (2010) 2858–2863.
[6] T. Kanamori, A. Takeda, Worst-case violation of sampled convex programs for optimization with uncertainty, J. Optim. Theory Appl. 152 (1) (2012) 171–197.
[7] P. Mohajerin, T. Sutter, J. Lygeros, Performance bounds for the scenario approach and an extension to a class of non-convex programs, IEEE Trans. Automat. Control 60 (1) (2015) 46–58.
[8] G. Schildbach, L. Fagiano, M. Morari, Randomized solutions to convex programs with multiple chance constraints, SIAM J. Optim. 23 (4) (2013) 2479–2501.
[9] G.C. Calafiore, D. Lyons, Random convex programs for distributed multi-agent consensus, in: European Control Conference, IEEE, 2013, pp. 250–255.
[10] D.P. de Farias, B. Van Roy, The linear programming approach to approximate dynamic programming, Oper. Res. 51 (6) (2003) 850–865.
[11] G.C. Calafiore, Random convex programs, SIAM J. Optim. 20 (6) (2010) 3427–3464.
[12] S. Grammatico, X. Zhang, K. Margellos, P. Goulart, J. Lygeros, A scenario approach for non-convex control design, IEEE Trans. Automat. Control 61 (2) (2016) 334–345.
[13] G. Schildbach, L. Fagiano, C. Frei, M. Morari, The scenario approach for stochastic model predictive control with bounds on closed-loop constraint violations, Automatica 50 (12) (2014) 3009–3018.
[14] X. Zhang, S. Grammatico, G. Schildbach, P. Goulart, J. Lygeros, On the sample size of random convex programs with structured dependence on the uncertainty, Automatica 60 (2015) 182–188.
[15] A. Prékopa, Stochastic Programming, Springer, 1995.
[16] K. Margellos, M. Prandini, J. Lygeros, On the connection between compression learning and scenario based single-stage and cascading optimization problems, IEEE Trans. Automat. Control 60 (10) (2015) 2716–2721.
[17] T. Alamo, R. Tempo, A. Luque, On the sample complexity of probabilistic analysis and design methods, in: Persp. in Math. System Theory, Control, and Signal Processing, Springer, 2010, pp. 39–50.
[18] Y. Nesterov, Universal gradient methods for convex optimization problems, Math. Program. (2013) 1–24.
[19] A. Domahidi, E. Chu, S. Boyd, Ecos: An socp solver for embedded systems, in: European Control Conference, IEEE, 2013, pp. 3071–3076.
[20] N. Kariotoglou, T. Summers, M. Kamgarpour, J. Lygeros, A numerical approach to stochastic reach-avoid problems for mdps, Tech. Report. URL http://arxiv.org/abs/1411.5925.
[21] N. Kariotoglou, S. Summers, T. Summers, M. Kamgarpour, J. Lygeros, Approximate dynamic programming for stochastic reachability, in: European Control Conference, IEEE, 2013, pp. 584–589.
[22] S. Summers, J. Lygeros, Verification of discrete time stochastic hybrid systems: A stochastic reach-avoid decision problem, Automatica 46 (12) (2010) 1951–1961.
[23] D.P. Kroese, T. Taimre, Z.I. Botev, Handbook of Monte Carlo Methods, Vol. 706, John Wiley & Sons, 2011.