2013 IEEE Conference on Computer Aided Control System Design (CACSD)
Part of 2013 IEEE Multi-Conference on Systems and Control
Hyderabad, India, August 28-30, 2013

Sequential Randomized Algorithms for Sampled Convex Optimization

Mohammadreza Chamanbaz, Fabrizio Dabbene, Roberto Tempo,
Venkatakrishnan Venkataramanan, Qing-Guo Wang

Abstract— **Motivated by the complexity of solving convex scenario problems in one-shot, two new algorithms for the sequential solution of sampled convex optimization problems are presented, for full constraint satisfaction and partial constraint satisfaction, respectively. A rigorous analysis of the theoretical properties of the algorithms is provided, and the related sample complexity is derived. Extensive numerical simulations for a non-trivial example testify the goodness of the proposed solution.**

I. Introduction

In recent years, research on randomized and probabilistic methods for control of uncertain systems has successfully evolved along various directions, see e.g. [17] for an overview of the state of the art on this topic. In particular, different approaches and techniques have been developed and tested in several applications, see e.g. [11]. For convex control design, two main classes of algorithms, sequential and non-sequential, have been proposed in the literature, and their theoretical properties have been rigorously studied.

Regarding non-sequential methods, the approach that has emerged is the so-called scenario approach, which has been introduced in [8], [9]. Taking random samples of the uncertainty $q \in \mathbb{Q}$, the main idea of this particular line of research is to reformulate a semi-infinite convex optimization problem as a sampled optimization problem subject to a finite number of constraints. Then, a key problem is to determine the sample complexity, i.e. the number of random constraints that should be generated, so that the so-called probability of violation is smaller than a given accuracy $\epsilon \in (0,1)$, and this event holds with a suitably large confidence $1-\delta \in (0,1)$. A very nice feature of the scenario approach is that the sample complexity is determined a priori, that is before the sampled optimization problem is solved, and it depends only on the number of design parameters, accuracy and confidence. On the other hand, if accuracy and confidence are very small, and the number of design parameters is large, then the sample complexity may be huge, and the sampled convex optimization problem cannot be easily solved in practice.

For this reason, a parallel line of research, mainly focused on deriving sequential methods, has been developed for various specific control problems, which include linear quadratic regulators, linear matrix inequalities and switched systems as particular cases of a general framework, based on various update rules and probabilistic oracles, presented in [17], [11]. The main idea of these sequential methods is to introduce the concept of validation samples. That is, at step k of the sequential algorithm, a "temporary solution" is constructed and, using a suitably generated validation sample set, it is verified whether or not the probability of violation corresponding to the temporary solution is smaller that a given accuracy ϵ, and this event holds with confidence $1-\delta$. To study the properties of these algorithms, the sample complexity of the validation set should be derived, but in this case, unlike the scenario approach, the sample complexity is a random variable which cannot be derived a priori. However, due to their sequential nature, these sequential algorithms might have wider applications than the scenario approach, in particular when fast computations are needed because of very stringent time requirements due to on-line implementations.

In this paper, we study two new sequential algorithms for full constraint satisfaction and partial constraint satisfaction, respectively, and we provide a rigorous analysis of their theoretical properties regarding the probability of violation. The sample complexity of both algorithms is derived and it enters directly into the validation step. The sample complexity increases very mildly with probabilistic accuracy, confidence and number of design parameters, and also on a termination parameter which is chosen by the user. In the worst case, an optimization problem having the same size of the scenario approach should be solved.

In the second part of the paper, using a non-trivial example regarding the control of a multivariable model for the lateral motion of an aircraft, we provide extensive numerical simulations which compare upfront the sample complexity of the scenario approach with the number of iterations required in the two sequential algorithms previously introduced. We remark again that the sample complexity of the scenario approach is computed a priori, while for sequential algorithms, the numerical results regarding the size of the validation sample set are random. For this reason, mean values, standard deviation and other related parameters are experimentally computed for both proposed algorithms by means of extensive Monte Carlo simulations.

M. Chamanbaz and V. Venkatakrishnan are with Data Storage Institute, Singapore (emails: {Mohammad_C, Venka_V}@dsi.a-star.edu.sg).

F. Dabbene and R. Tempo are with CNR-IEIIT Torino, Italy (emails: {roberto.tempo, fabrizio.dabbene}@polito.it).

M. Chamanbaz and Q-G. Wang are with Department of Electrical and Computer Engineering, National University of Singapore (emails: {Chamanbaz,Elewqg}@nus.edu.sg).

978-1-4799-1565-1/13/$31.00 ©2013 IEEE

II. Problem Formulation and Preliminaries

An uncertain convex problem has the form

$$\begin{aligned} &\underset{\theta\in\Theta}{\text{minimize}} \quad c^T\theta \\ &\text{subject to } f(\theta,q)\leq 0 \text{ for all } q\in\mathbb{Q} \end{aligned} \tag{1}$$

where $\theta\in\Theta\subset\mathbb{R}^{n_\theta}$ is the vector of optimization variables and $q\in\mathbb{Q}\subset\mathbb{R}^{\ell}$ denotes the vector of uncertain parameters bounded in the set $\mathbb{Q}$, $f(\theta,q):\Theta\times\mathbb{Q}\to\mathbb{R}$ is convex in θ for any fixed value of $q\in\mathbb{Q}$ and Θ is a convex and closed set. We note that most uncertain convex problems can be reformulated as (1). In particular, multiple scalar-valued constraints $f_i(\theta,q)\leq 0,\ i=1,\ldots,m$ can always be recast into the form (1) by defining $f(\theta,q)=\max\limits_{i=1,\ldots,m} f_i(\theta,q)$.

In this paper, we study a probabilistic framework in which the uncertainty vector q is assumed to be a random variable and the constraint in equation (1) is allowed to be violated for some $q\in\mathbb{Q}$, provided that the rate of violation is sufficiently small. This concept is formally expressed using the notion of "probability of violation".

Definition 1 (Probability of Violation): The probability of violation of θ for the function $f:\Theta\times\mathbb{Q}\to\mathbb{R}$ is defined as

$$V(\theta)\doteq\Pr\left\{q\in\mathbb{Q}\ :\ f(\theta,q)>0\right\}. \tag{2}$$

The exact computation of $V(\theta)$ is in general very difficult since it requires the computation of multiple integrals associated to the probability in (2). However, this probability can be estimated using randomization. To this end, assume a probability measure is given over the set $\mathbb{Q}$, we generate N independent identically distributed (i.i.d) samples within the set $\mathbb{Q}$

$$\mathbf{q}=\{q^{(1)},\ldots,q^{(N)}\}$$

based on the given density function. Next, a Monte-Carlo type approach is employed to obtain the so called "empirical violation" which is introduced in the following definition.

Definition 2 (Empirical Violation): For given $\theta\in\Theta$ the empirical violation of $f(\theta,q)$ with respect to the multisample $\mathbf{q}=\{q^{(1)},\ldots,q^{(N)}\}$ is defined as

$$\widehat{V}(\theta,\mathbf{q})\doteq\frac{1}{N}\sum_{i=1}^{N}\mathbb{I}_f(\theta,q^{(i)}) \tag{3}$$

where $\mathbb{I}_f(\theta,q^{(i)})$ is an indicator function defined as

$$\mathbb{I}_f(\theta,q^{(i)})\doteq\begin{cases}0 & \text{if } f(\theta,q)\leq 0\\ 1 & \text{otherwise}\end{cases}. \tag{4}$$

It is clear that, based on the definition of $\mathbb{I}_f(\theta,q^{(i)})$, the empirical violation is a random variable bounded in the closed interval $[0,1]$. Next, we need to obtain an explicit sample bound N for which it is guaranteed that the empirical violation (3) is within a desired pre-specified accuracy $\varepsilon\in(0,1)$ with high confidence $\delta\in(0,1)$. Such a characterization is provided by the additive Chernoff inequality, see e.g. [17] for more details.

One-sided Additive Chernoff Inequality: For fixed θ, ε and for any $f:\Theta\times\mathbb{Q}\to\mathbb{R}$, the one-sided additive Chernoff inequality is defined as

$$\Pr\left\{V(\theta)-\widehat{V}(\theta,\mathbf{q})\geq\varepsilon\right\}\leq 2e^{-2N\varepsilon^2}. \tag{5}$$

Bounding the right hand side of (5) with δ results in the additive Chernoff bound

$$N\geq\frac{1}{2\varepsilon^2}\ln\frac{1}{\delta}. \tag{6}$$

The Scenario Approach

In this subsection, we briefly recall the so-called scenario approach, also known as random convex programs, which was first introduced in [8], [9], see also [12], [7] for additional results regarding the so-called discarded constraint problems. In this approach, a set of independent identically distributed random samples of cardinality N is extracted from the uncertainty set and the following random convex program is formed

$$\begin{aligned} &\underset{\theta\in\Theta}{\text{minimize}} \quad c^T\theta \\ &\text{subject to } \ f(\theta,q^{(i)})\leq 0,\ i=1,\ldots,N. \end{aligned} \tag{7}$$

The function $f(\theta,q)$ is convex for fixed $q\in\mathbb{Q}$ and a further assumption is that the problem (7) attains a unique solution $\widehat{\theta}_N$. These assumptions are now formally stated.

Assumption 1 (Convexity): $\Theta\subset\mathbb{R}^{n_\theta}$ is a convex and closed set and $f(\theta,q)$ is convex in θ for any fixed value of $q\in\mathbb{Q}$.

Assumption 2 (Uniqueness): If the optimization problem (7) is feasible, it admits a unique solution.

We remark that the uniqueness assumption can be relaxed in most cases by introducing a tie-breaking rule (see Section 4.1 of [8]). The probabilistic property of the optimal solution obtained from (7) is stated in the next lemma which was first proven in [12] under strict feasibility assumption.

Lemma 1: Suppose that Assumptions 1 and 2 hold and let $\delta,\varepsilon\in(0,1)$ and N satisfy the following inequality

$$\sum_{i=0}^{n_\theta}\binom{N}{i}\varepsilon^i(1-\varepsilon)^{N-i}\leq\delta. \tag{8}$$

Then, either the optimization problem (7) is infeasible which means the original problem (1) is also infeasible or, if feasible, with probability no smaller than $1-\delta$, its optimal solution $\widehat{\theta}_N$ satisfies the inequality $V(\widehat{\theta}_N)\leq\varepsilon$.

Proof: See Theorem 3.3 in [7]. ■

There are a number of results in the literature for deriving the appropriate sample bound N which satisfies (8). The least conservative one, which is proved in [3], is stated here.

Lemma 2: Let Assumptions 1 and 2 hold. Then, for given $\varepsilon\in(0,1)$ and $\delta\in(0,1)$, Lemma 1 holds for

$$N\geq\inf_{a>1}\frac{1}{\varepsilon}\left(\frac{a}{a-1}\right)\left(\ln\frac{1}{\delta}+n_\theta\ln a\right). \tag{9}$$

Proof: See Theorem 4 in [3]. ■

We remark that any choice of $a > 1$ results in a sub-optimal sample bound N. For instance, if we choose $a = e$ where e is the Euler number, the bound (9) results in

$$N \geq \frac{1.58}{\varepsilon}\left(\ln\frac{1}{\delta} + n_\theta\right). \tag{10}$$

The sample bounds (9) and (10) can be very large even for problems with moderate number of decision variables. Therefore, the computational complexity of the formulated random convex program (7) might be beyond the capability of the available computational tools. Motivated by this limitation, in the next section we propose two sequential randomized algorithms which may have computational complexity smaller than the scenario approach.

III. The Sequential Randomized Algorithms

The main philosophy behind the proposed sequential randomized algorithms lies on the fact that it is easy from the computational point of view to evaluate a given "candidate solution" for a large number of random samples extracted from $\mathbb{Q}$, while it is clearly more expensive to solve the optimization problem (7) when the sample bound N is large. The sequential randomized algorithms, which are presented next, mitigate the potential conservativeness of the bounds (9) and (10) by generating a sequence of "design" sample sets $\{q_d^{(1)}, \ldots, q_d^{(N_k)}\}$ with increasing cardinality N_k which are used in (7) for solving the optimization problem. Parallel "validation" sample sets $\{q_v^{(1)}, \ldots, q_v^{(M_k)}\}$ of cardinality M_k are also generated by both algorithms in order to check if the given candidate solution, obtained from solving (7), satisfies the desired violation probability or not.

A. Full Constraint Satisfaction

The first algorithm is in line with the algorithm presented in [16] and [10] in the sense that it uses similar strategy to validate the candidate solution. A generalization of the algorithm/bound derived in [16] and [10] is presented in [1] using the so-called cardinality function. The sequential randomized is presented in Algorithm 1, and its theoretical properties are stated in the following theorem.

Theorem 1: Suppose that Assumptions 1 and 2 hold. If at iteration k the Algorithm 1 exits with a probabilistic solution $\widehat{\theta}_{N_k}$, then it holds that $V(\widehat{\theta}_{N_k}) \leq \varepsilon$ with probability no smaller than $1 - \delta$

$$\Pr\left\{V(\widehat{\theta}_{N_k}) \leq \varepsilon\right\} \geq 1 - \delta.$$

Proof: See Theorem 1 in [13]. ∎

Remark 1 (Optimal Value of α): The sample bound (12) is similar to the one derived in [11] Theorem 2 and originally proven in [15], with the exception that we are not using the Riemann Zeta function because the summation is a finite sum. The Riemann Zeta function does not converge for the case that α is smaller than one. However, in the presented bound (12) we can let α to be smaller than one which improves the sample complexity specially for the case that k_t is large. The optimal value of α which minimizes the sample bound (12) is computed by running simulations for different values of the termination parameter k_t. The almost optimal value of α minimizing (12) for a wide range of k_t is $\alpha = 0.1$. The bound (12) (for $\alpha = 0.1$) improves upon the bound derived in [11], by 5% to 15% depending on the termination parameter k_t. It also improves upon the bound in [16] which uses finite sum but in a less effective way.

Algorithm 1 The Sequential Randomized Algorithm: Full Constraint Satisfaction

1) Initialization
   Set the iteration counter to zero ($k = 0$). Choose the desired probabilistic levels ε, δ and the desired number of iterations $k_t > 1$.
2) Update
   Set $k = k + 1$ and $N_k = \left\lceil N\frac{k}{k_t}\right\rceil$ where N is chosen based on (10) and $\lceil x\rceil$ denotes the smallest integer greater than or equal to x.
3) Design
   - Draw N_k i.i.d samples $\mathbf{q}_d = \{q_d^{(1)} \ldots q_d^{(N_k)}\}$ from the uncertainty set $\mathbb{Q}$ based on the underlying distribution.
   - Solve the following random convex program
   $$\widehat{\theta}_{N_k} = \arg\operatorname*{minimize}_{\theta\in\Theta} \quad c^T\theta \tag{11}$$
   $$\text{subject to} \quad f(\theta, q_d^{(i)}) \leq 0, \quad i = 1, \ldots, N_k.$$
   - If the optimization problem (11) is not feasible, the original problem (1) is not feasible as well. Else, continue to the next step.
4) Validation
   - Draw
   $$M_k > \left\lceil \frac{\alpha\ln k + \ln\left(\mathcal{S}_{k_t}(\alpha)\right) + \ln\frac{1}{\delta}}{\ln\left(\frac{1}{1-\varepsilon}\right)} \right\rceil \tag{12}$$
   i.i.d samples $\mathbf{q}_v = \{q_v^{(1)} \ldots q_v^{(M_k)}\}$ from the uncertainty set $\mathbb{Q}$ based on the underlying distribution. The parameter $\mathcal{S}_{k_t}(\alpha)$ in (12) is a finite hyperharmonic series also known as p-series that is
   $$\mathcal{S}_{k_t}(\alpha) = \sum_{k=1}^{k_t}\frac{1}{k^\alpha}.$$
   - If
   $$\mathbb{I}_f(\widehat{\theta}_{N_k}, q_v^{(i)}) = 0 \text{ for } i = 1, \ldots, M_k$$
   then, $\widehat{\theta}_{N_k}$ is a probabilistic robust solution to (1) with confidence δ and accuracy ε and Exit. Else, goto step (2).

B. Partial Constraint Satisfaction

In the "validation" step of the Algorithm 1, *all* elements of the validation sample set $\mathbf{q}_v = \{q_v^{(1)} \ldots q_v^{(M_k)}\}$ are required to satisfy the constraint in (1). However, it is sometimes impossible finding a solution satisfying the constraint in (1)

for the entire set of uncertainty. In the next sequential randomized algorithm, we allow a limited number of validation samples to violate the constraint in (1).

Algorithm 2 THE SEQUENTIAL RANDOMIZED ALGORITHM: PARTIAL CONSTRAINT SATISFACTION

1) INITIALIZATION
Set the iteration counter to zero ($k = 0$). Choose the desired probabilistic levels ε, δ, the desired number of iterations $k_t > 1$ and define the following parameters:

$$\beta_v \doteq \max\left\{1, \beta_w \left(k_t \ln \frac{k_t}{\delta}\right)^{-1}\right\} \tag{13}$$

$$\beta_w \doteq \frac{1}{4\varepsilon} \ln \frac{1}{\delta}. \tag{14}$$

2) UPDATE
Set $k = k + 1$ and $\left\lceil N_k = N\frac{k}{k_t}\right\rceil$ where N is chosen based on (10).

3) DESIGN
- Draw N_k i.i.d samples $\mathbf{q}_d = \{q_d^{(1)} \dots q_d^{(N_k)}\}$ from the uncertainty set $\mathbb{Q}$ based on the underlying distribution.
- Solve the following random convex program:

$$\widehat{\theta}_{N_k} = \arg\underset{\theta\in\Theta}{\text{minimize}} \quad c^T\theta \tag{15}$$
$$\text{subject to} \quad f(\theta, q_d^{(i)}) \leq 0, \quad i = 1, \dots, N_k.$$

- If the optimization problem (15) is not feasible, the original problem (1) is not feasible as well. Else, continue to the next step.

4) VALIDATION
- Draw $M_k = \left\lceil 2k\beta_v \frac{1}{\varepsilon} \ln \frac{k_t}{\delta}\right\rceil$ i.i.d samples $\mathbf{q}_v = \{q_v^{(1)} \dots q_v^{(M_k)}\}$ from the uncertainty set $\mathbb{Q}$ based on the underlying distribution.
- If

$$\frac{1}{M_k}\sum_{i=1}^{M_k} \mathbb{I}_f(\widehat{\theta}_{N_k}, q_v^{(i)}) \leq \left(1 - (k\beta_v)^{-1/2}\right)\varepsilon \tag{16}$$

then, $\widehat{\theta}_{N_k}$ is a probabilistic robust solution to (1) with confidence δ and accuracy ε and Exit. Else, goto step (2).

We now state a theorem explaining the theoretical properties of the Algorithm 2.

Theorem 2: Suppose that Assumptions 1 and 2 hold. If at iteration k Algorithm 2 exits with a probabilistic solution $\widehat{\theta}_{N_k}$, then it holds that $V(\widehat{\theta}_{N_k}) \leq \varepsilon$ with probability no smaller than $1 - \delta$

$$\Pr\left\{V(\widehat{\theta}_{N_k}) \leq \varepsilon\right\} \geq 1 - \delta.$$

Proof: See Theorem 2 in [13]. ■

Remark 2 (Choice of N_{k_t}): The cardinality of the design sample set at the last iteration in both algorithms N_{k_t}, is chosen to be exactly equal to the bound (10). This choice implies if Algorithms 1 and 2 reach the last iteration, they use exactly the same sample bound as the one used by the scenario approach. Therefore, the complexity of the last iteration, if it is reached, is exactly equal to that of the scenario approach.

The Algorithm 2 is different from the algorithm presented in [2], which was derived for non-convex problems, in a number of aspects. That is, the cardinality of the sequence of sample sets used for design and validation increase linearly with iteration counter k, while they increase exponentially in [2]. Furthermore, the cardinality of the validation sample set at the last iteration M_{k_t} in [2] is chosen to be equal to the cardinality of the sample set used for design at the last iteration N_{k_t} while, in the presented algorithm M_{k_t} and hence β_w are chosen based on the additive Chernoff bound (6) which is less conservative.

C. Termination Parameter k_t

The termination parameter k_t defines the maximum number of iterations of the algorithm which can be chosen by the user. We note that the choice of k_t directly affects the cardinality of the sample sets used for design N_k and validation M_k at each iteration, although they converge to fixed values (independent of k_t) at the last iteration. For problems with large scenario bound (10), we would suggest to use larger k_t. The sequence of sample bounds N_k starts from a small number and does not increase dramatically with the iteration counter k. We also remark that the right hand side of the inequality (16) in the Algorithm 2 cannot be negative which in turn requires β_v to be greater than one. This condition is taken into account in defining β_v in (13). However, we can avoid generating $\beta_v < 1$ by an appropriate choice of k_t. In order to do so, we solve the inequality $\beta_v \geq 1$ for k_t as follows

$$\begin{aligned}\beta_v \doteq &\beta_w \left(k_t \ln \frac{k_t}{\delta}\right)^{-1} \geq 1\\ &k_t \ln \frac{k_t}{\delta} \leq \beta_w\\ &\frac{k_t}{\delta} \ln \frac{k_t}{\delta} \leq \frac{\beta_w}{\delta}.\end{aligned}$$

For implementation purposes, it is useful to use the function "LambertW" also known as "Omega function" or "product logarithm"[1]. Then, we solve the previous inequality for k_t

$$k_t \leq \frac{\beta_w}{\text{LambertW}\left(\frac{\beta_w}{\delta}\right)}.$$

By substituting β_w, the termination parameter k_t is obtained as

$$k_t \leq \left\lceil \frac{\frac{1}{4\varepsilon}\ln\left(\frac{1}{\delta}\right)}{\text{LambertW}\left(\frac{\frac{1}{4\varepsilon}\ln\left(\frac{1}{\delta}\right)}{\delta}\right)} \right\rceil. \tag{17}$$

[1]This function is the inverse function of $f(W) = We^W$. In other words, $W = \text{LambertW} f(W)$; see e.g. [14] for more details. The Matlab command is `W = lambertw(f(W))`.

L_p	L_β	L_r	g/V	Y_β	$N_{\dot\beta}$	N_p	N_β	N_r	L_{δ_a}	Y_{δ_r}	N_{δ_r}	N_{δ_a}
-2.93	-4.75	0.78	0.086	-0.11	0.1	-0.042	2.601	-0.29	-3.91	0.035	-2.5335	0.31

TABLE I

UNCERTAINTY VECTOR q AND ITS NOMINAL VALUE $\overline{q}$

ε	δ	k_t	*Design Samples*			*Validation Samples*			*Iteration Number*			*Computational Time (Sec)*		
			Mean	*Standard Deviation*	*Worst Case*	*Mean*	*Standard Deviation*	*Worst Case*	*Mean*	*Standard Deviation*	*Worst Case*	*Mean*	*Standard Deviation*	*Worst Case*
0.1	10^{-4}	10	211.8	53.4	291	109.9	0.31	110	5.1	1.3	7	10.9	11.2	58
0.05	10^{-5}	12	308.5	134.4	451	272.7	1.25	274	4.1	1.8	6	24.7	25.5	135.1
0.02	10^{-6}	20	1035.2	499.1	1827	832.5	2.5	836	8.5	3.7	15	65.1	56.1	239.1
0.01	10^{-8}	25	2149.8	1045.7	4030	2152.5	4.7	2160	9.6	4.7	18	144.6	124	643.1
0.005	10^{-9}	30	4689	434.9	5166	4813.6	2.2	4816	11.8	1.1	13	325	338.2	1774.7

TABLE II

SIMULATION RESULTS OBTAINED USING ALGORITHM 1

ε	δ	k_t	*Design Samples*			*Validation Samples*			*Iteration Number*			*Computational Time (Sec)*		
			Mean	*Standard Deviation*	*Worst Case*	*Mean*	*Standard Deviation*	*Worst Case*	*Mean*	*Standard Deviation*	*Worst Case*	*Mean*	*Standard Deviation*	*Worst Case*
0.1	10^{-4}	10	108.2	23	125	599	125.9	691	2.6	0.54	3	12.8	4.4	25.6
0.05	10^{-5}	12	266	53	301	1680	395.9	2240	3	0.7	4	31	12.5	58.7
0.02	10^{-6}	20	389.8	133.1	487	5380.2	1841.4	6725	3.2	1.1	4	115.7	51.7	301.1
0.01	10^{-8}	25	761.6	122.7	896	14715	2370.5	17312	3.4	0.54	4	324.1	127.6	557.7
0.005	10^{-9}	30	1431.2	466.4	2385	49737	16217	82894	3.6	1.17	6	854	355.5	2170.7

TABLE III

SIMULATION RESULTS OBTAINED USING ALGORITHM 2

IV. NUMERICAL SIMULATION

In this section, we employ the developed sequential randomized algorithms of Section III to solve a non-trivial control problem. The plant under consideration is a multivariable model for the lateral motion of an aircraft. The example is studied in [4] and [17], and it is adopted originally from [19]. The state space description of the model is given by

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & L_p & L_\beta & L_r \\ g/V & 0 & Y_\beta & -1 \\ N_{\dot\beta}(g/V) & N_p & N_\beta + N_{\dot\beta}Y_\beta & N_r - N_{\dot\beta} \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ 0 & L_{\delta_a} \\ Y_{\delta_r} & 0 \\ N_{\delta_r} + N_{\dot\beta}Y_{\delta_r} & N_{\delta_a} \end{bmatrix} u \tag{18}$$

where x_1 is the bank angle, x_2 its derivative, x_3 is the sideslip angle, x_4 the yaw rate, u_1 the rudder deflection and u_2 the aileron deflection. The objective is to design a state feedback controller of the form $u = Kx$ which, firstly, stabilizes the plant (18); secondly, enforces all the eigenvalues of the closed loop system to have real part smaller than $-\alpha\,(\alpha > 0)$ and, finally, we would like the trace of Lyapunov matrix, used for testing the stability of closed loop system, to be minimized. All parameters appearing in the model (18) are considered to be uncertain. Hence, representing the plant (18) as

$$\dot{x} = A(q)x + B(q)u$$

the uncertainty vector q is of dimension 13 and the uncertainty set $\mathbb{Q}$ is assumed to be a 13 dimensional hyperrectangle centered at the nominal value $\overline{q}$ with radius R i.e.

$$\mathbb{Q} = \{q_i \in \mathbb{R} :\, q_i \in [(1-R)\overline{q}, (1+R)\overline{q}],\, i = 1, \ldots, 13\}\,.$$

Using the approach of [6], we solve the following optimization problem

$$\begin{aligned} &\underset{P,Y}{\text{minimize}}\ \text{Tr}\ P \\ &\text{subject to}\ A(q)P + PA^T(q) + B(q)Y + Y^TB^T(q) \\ &\qquad + 2\alpha P \preceq 0 \end{aligned} \tag{19}$$

where $P \in \mathbb{R}^{4\times4}$ is a symmetric positive definite matrix and $Y \in \mathbb{R}^{2\times4}$ is an auxiliary variable. The feedback gain K can be constructed as $K = YP^{-1}$. We note that the constraint (19) is in the form of a linear matrix inequality (LMI) and by introducing the convex function $\lambda_{\max}(.)$, which represents

the maximum eigenvalue of the matrix, the problem can be recast in the form (1).

Sequential algorithms of Section III are implemented in Matlab using the the toolbox Randomized Algorithm Control Toolbox (RACT) [18]. In the simulation, we assumed that the relative uncertainty is 15% $(R = 0.15)$ and the probability density function of all uncertain parameters is uniform. The choice of uniform distribution is chosen due to its worst case nature [5]. The vector of nominal values $\overline{q}$ is given in Table I. The optimization problem (19) is solved for different values of ε and δ. Furthermore, we run the simulation 100 times for each pair of ε and δ. The mean, standard deviation and worst case values of the number of design samples, optimization samples, the iteration number in which the algorithm exits and the total computational time are tabulated in Tables II and III[2]. Table IV shows the scenario bound along with the computational time required for solving the random convex problem for the same probabilistic levels as Tables II and III. It can be seen that using the proposed algorithm, we can achieve the same probabilistic levels with much smaller number of design samples. The computational time of Tables II and III is much smaller than Table IV which further proves the effectiveness of the proposed sequential randomized algorithms. We note that the number of validation samples in Table III is bigger than the one in Table II which means Algorithm 2 needs more validation samples to verify the feasibility of the candidate solution. The increase in the number of validation samples in Algorithm 2 has two reasons: i) since we allow a number of validation samples to violate the constraint in (1), the algorithm requires more samples to be checked; and ii) the bound M_k in Algorithm 2 is purely based on Chernoff inequalities which is more conservative than log-over-log bounds.

ε	δ	*The Scenario Bound*	*Computational Time (Sec)*
0.1	1×10^{-4}	414	13
0.05	1×10^{-5}	901	24.6
0.02	1×10^{-6}	2434	108.4
0.01	1×10^{-8}	5596	479.3
0.005	1×10^{-9}	11920	2295

TABLE IV

THE SCENARIO BOUND AND THE REQUIRED COMPUTATIONAL TIME FOR THE SAME PROBABILISTIC LEVELS AS TABLES. II AND III

V. CONCLUSIONS

We proposed two sequential methods for solving in a computational efficient way the sampled convex optimization problems arising from the application of the scenario approach. The main philosophy behind the proposed sequential randomized algorithms stems from the consideration that it is easy, from a computational viewpoint, to validate a given "candidate solution" for a large number of random samples. The algorithms have been tested on a numerical example, and extensive numerical simulations show how the total computational effort is "diluted" by applying the proposed methodology.

ACKNOWLEDGEMENTS

The results presented in this paper were obtained when the first author was visiting CNR-IEIIT for a period of six months under DSI funding.

REFERENCES

[1] T. Alamo, A. Luque, D.R. Ramirez, and R. Tempo. Randomized control design through probabilistic validation. In *Proc. American Control Conference (ACC)*, pages 839–844, 2012.

[2] T. Alamo, R. Tempo, and E.F. Camacho. Randomized strategies for probabilistic solutions of uncertain feasibility and optimization problems. *IEEE Transactions on Automatic Control*, 54(11):2545 – 2559, 2009.

[3] T. Alamo, R. Tempo, and A. Luque. On the sample complexity of probabilistic analysis and design methods. In J. Willems, S. Hara, Y. Ohta, and H. Fujioka, editors, *Perspectives in Mathematical System Theory, Control, and Signal Processing*, volume 398 of *Lecture Notes in Control and Information Sciences*, pages 39–50. Springer Berlin / Heidelberg, 2010.

[4] B.D.O. Anderson and J.B. Moore. *Optimal Control: Linear Quadratic Methods*. Prentice Hall, Upper Saddle River, NJ, USA, 1990.

[5] E. Bai, R. Tempo, and M. Fu. Worst-case properties of the uniform distribution and randomized algorithms for robustness analysis. *Mathematics of Control, Signals, and Systems*, 11:183–196, 1998.

[6] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. SIAM, Philadelphia, 1994.

[7] G.C. Calafiore. Random convex programs. *SIAM Journal on Optimization*, 20:3427–3464, 2010.

[8] G.C. Calafiore and M.C. Campi. Uncertain convex programs: randomized solutions and confidence levels. *Mathematical Programming*, 102:25–46, 2004.

[9] G.C. Calafiore and M.C. Campi. The scenario approach to robust control design. *IEEE Transactions on Automatic Control*, 51:742–753, 2006.

[10] G.C. Calafiore and F. Dabbene. A probabilistic analytic center cutting plane method for feasibility of uncertain LMIs. *Automatica*, 43:2022–2033, 2007.

[11] G.C Calafiore, F. Dabbene, and R. Tempo. Research on probabilistic methods for control system design. *Automatica*, 47:1279–1293, 2011.

[12] M.C. Campi and S. Garatti. The exact feasibility of randomized solutions of uncertain convex programs. *SIAM J. on Optimization*, 19:1211–1230, 2008.

[13] M. Chamanbaz, Fabrizio Dabbene, Roberto Tempo, V. Venkataramanan, and Q-G Wang. Sequential randomized algorithms for convex optimization in the presence of uncertainty. *IEEE Transactions on Automatic Control*, 2013, submitted. See also `arXiv:1304.2222 [cs.SY]`.

[14] R. M. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey, and D. E. Knuth. On the LambertW function. *Advances in Computational Mathematics*, 5:329–359, 1996.

[15] F. Dabbene, P. S. Shcherbakov, and B. T. Polyak. A randomized cutting plane method with probabilistic geometric convergence. *SIAM Journal on Optimization*, 20:3185–3207, 2010.

[16] Y. Oishi. Polynomial-time algorithms for probabilistic solutions of parameter-dependent linear matrix inequalities. *Automatica*, 43:538–545, 2007.

[17] R. Tempo, G.C. Calafiore, and F. Dabbene. *Randomized Algorithms for Analysis and Control of Uncertain Systems: With Applications*. Springer, 2nd edition, 2013.

[18] A. Tremba, G.C. Calafiore, F. Dabbene, E. Gryazina, B. Polyak, P. Shcherbakov, and R. Tempo. RACT: randomized algorithms control toolbox for MATLAB. In *Proc. 17th World Congress of IFAC, Seoul*, pages 390–395, 2008.

[19] J. Tyler and F. Tuteur. The use of a quadratic performance index to design multivariable control systems. *IEEE Transactions on Automatic Control*, 11:84–92, 1966.

[2] All the simulations are carried on a work station with $2.83\,GHz$ Core 2 Quad CPU and $8\,GB$ RAM.