4-26-2012

Application of Statistical Learning Control to the Design of a Fixed-Order Controller for a Flexible Beam

Chaouki T. Abdallah

M. Ariola

V. Koltchinskii

Application of Statistical Learning Control to the Design of a Fixed-Order Controller for a Flexible Beam

M. Ariola, V. Koltchinskii, C.T. Abdallah

Abstract

This paper shows how probabilistic methods and statistical learning theory can provide approximate solutions to "difficult" control problems. The paper also introduces bootstrap learning methods to drastically reduce the bound on the number of samples required to achieve a performance level. These results are then applied to obtain more efficient algorithms which probabilistically guarantee stability and robustness levels when designing controllers for uncertain systems. The paper includes examples of the applications of these methods.

Keywords

Statistical Learning, Radamacher bootstrap, Robust Control, Sample Complexity, $\mathcal{NP}$-hard problems, Decidability theory.

M. Ariola is with the Dipartimento di Informatica e Sistemistica, Università degli Studi di Napoli Federico II, Napoli, Italy. E-mail:ariola@unina.it. His research is partially supported by the MURST.

V. Koltchinskii is with the Department of Mathematics and Statistics, University of New Mexico, Albuquerque, NM 87131, USA. E-mail: vlad@math.unm.edu. His research is partially supported by NSA Grant MDA904-99-1-0031.

Corresponding Author: C.T. Abdallah is with the Electrical & Computer Engineering Department, University of Tennessee at Knoxville, 401 Ferris Hall Knoxville, TN 37996-2100, Usa. Phone: (865)974-0927, Fax: (865)974-5483 (Fax), E-mail: abdallah@mail.ee.utk.edu.

I. Introduction

Our paper is concerned with the problem of approximatley solving difficult control problems. Such problems are too costly to be solved exactly as discussed for example in [?]. Many authors have recently advanced the notion of probabilistic methods in control analysis and design for such problems. These methods build on the standard Monte Carlo approach with ideas advanced during the 1960s and 1970s [1] on the theory of empirical processes and statistical learning. In control theory, some of the original (Monte Carlo) ideas have already been used by Lee and Poolla [?], Ray and Stengel [2], Tempo et al. [3], [4], [5], Barmish et al. [6], [7], [8], [?], Chen and Zhou [9], [10], [11] and by Khargonakar and Tikku [12], to solve *robust analysis* problems while Vidyasagar used learning theory to solve *robust design* problems [13], [14].

Unfortunately, and as acknowledged by the various authors, probabilistic methods, while more efficient than gridding techniques (which suffer from the curse of dimensionality), still require a large number of samples in order to guarantee accurate designs. As an example, Vidyasagar in [14] calculates that more than 2 million samples are needed in order to probabilistically guarantee a certain performance level in a robust control design problem. On the other hand, it was conjectured and verified experimentally that much smaller bounds on the number of samples may be sufficient (tens of thousands instead of millions) to guarantee a certain level of performance [14]. In fact, Vidyasagar in [14] uses 200 samples instead of the millions implied by his bounds, while acknowledging that the theoretical guarantees of accuracy and confidence no longer hold.

This paper recovers the theoretical guarantees by invoking different versions of *bootstrap sequential learning* algorithms. For these algorithms, the necessary number of samples (known as the sample complexity of learning) is a random variable whose value is not known in advance and is to be determined in the process of learning. This value is bounded below by the sample size at which the algorithm starts to work, and bounded above by conservative upper bounds of the sample complexity, which are of the same order as the bounds well known in statistical learning theory, used, for instance, by Vidyasagar [13].

The remaining of this paper is divided as follows: section II contains a discussion of generic robust control problems, their difficulty, and their computational complexity. Section III presents an overview of statistical learning methods and section IV contains the bootstrap learning method and its applications to control problems. Section V contains numerical examples illustrating our approach and contrasting it with earlier results, while section VI contains conclusions and an outline for future research.

II. Robust Control, Decision Theory, and Computational Complexity

In studying control problems we are led to the conclusion that some robust control problems are actually undecidable. For example, the simultaneous stabilization problem of more than two plants was shown by Blondel [15] to be rationally undecidable using a general model of computing. More examples of such problems may be found in [16].

Most of the control problems we study in this paper are decidable and may be converted to a decision problem relating to the satisfiability of quantified multivariate polynomial inequalities (MPIs) which are then reduced using Tarski's quantifier elimination (QE) theory [17]. These problems include the fixed-structure control design problem for linear and nonlinear systems which remains one of the most practical and difficult problems [?], [?]. In fact, one can argue that most practical control designs involve fixed-structure (and fixed-order) controllers such as PID, or Lead-Lag compensators (see page 113 of [?], and page 3 of [?]). While this makes the control design problem theoretically intractable, it actually reduces some undecidable problems to decidable ones, and fits nicely within the randomized algorithms framework. As an example, the following problems are all decidable using Tarski's decision theory: robust stabilization problems [18], dead-beat control of discrete-time systems [19], Lyapunov stability of polynomial systems [20], and others [21]. The general control problem for an uncertain single input single output (SISO), linear time invariant (LTI) system stated as a decision problem is as follows,

Problem 1: Given a real rational function $G(s,X)$*, where* $X = [x_1 \quad x_2 \quad \cdots \quad x_k]$ *is a* k*-dimensional real vector, does there exist an* l*-dimensional real vector* $Y = [y_1 \quad y_2 \quad \cdots \quad y_l]$*,* $y_i)$ $[\underline{y_i} \leq y_i \leq \overline{y_i}]$*,* $1 \leq i \leq l$*, in the real rational* $C(s,Y)$ *such that for all* (x_i) $[\underline{x_i} \leq x_i \leq \overline{x_i}]$*,* $1 \leq i \leq k$*, the closed-loop system*

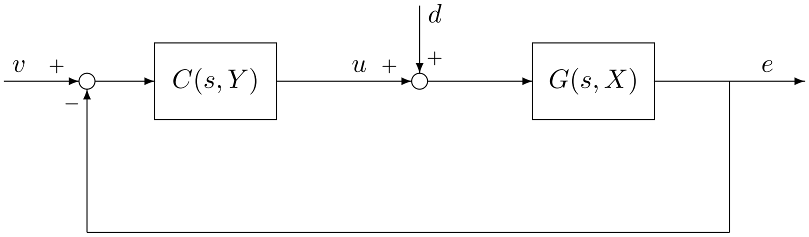

Fig. 1. Feedback Structure for Problem 1

$T(s, X, Y)$ satisfies some performance objectives placed on a scalar performance index $\Psi(X, Y)$?

This is the performance verification problem [**?**], and it includes the guaranteed-cost design problem. Note that if either X or Y are known, then the problem simplifies to a robust analysis problem. Typical examples are the linear quadratic regulator (LQR), and specialized guaranteed-cost problems. In the case where the problem and the performance objective are convex, Linear Matrix Inequalities (LMIs) may be used and the decision control problem is easy. The general decision control problem is however very hard because it leads to a nonlinear, partial differential Hamilton-Jacobi-Bellman (HJB) equation (for non-quadratic performance objectives) which in general is difficult to solve. Researchers in Control Theory have used QE in solving Problem 1 since the 1970's, but the tedious operations made the technique very limited [18]. Later, Collins [22] introduced a theoretically more efficient QE algorithm that uses a cylindrical algebraic decomposition (CAD) approach. However, this algorithm was not capable of effectively handling nontrivial problems. Then, Collins and Hong [23], and Hong [24], [25] introduced a significantly more efficient partial CAD QE algorithm, implemented in the software package QEPCAD. Recently, people have used the QEPCAD software to solve academic, but nontrivial problems [26], [21], [27], [20], [19].

One important concept in this theory is that of a *polynomial-time algorithm*. In practice, such an algorithm can be feasibly implemented on a real computer. This is in contrast to an *exponential-time algorithm*, which is only feasible if the problem being solved is extremely small. Unfortunately, it turns out that QE is at best exponential! [28]

The complexity class $\mathcal{P}$ consists of all decision problems that can be decided in polynomial-time, using a Turing machine model of computation. The simplicity of the Turing machine model appears to make it of little practical value; however, the Church-Turing Thesis holds that the class of problems solvable on a Turing machine in polynomial time is robust across all other reasonable models of computation (including the computers we use).

The complexity class $\mathcal{NP}$ consists of all decision problems that can be decided algorithmically in *nondeterministic* polynomial-time. An algorithm is nondeterministic if it is able to choose or guess a sequence of choices that will lead to a solution, without having to systematically explore all possibilities. This model of computation is not realizable, but it is of theoretical importance since it is strongly believed that $\mathcal{P} \neq \mathcal{NP}$. In other words, these two complexity classes form an important boundary between the tractable (or easy) and intractable (or difficult) problems. A problem is said to be $\mathcal{NP}$-hard if it is as hard as any problem in $\mathcal{NP}$. Thus, if $\mathcal{P} \neq \mathcal{NP}$, the $\mathcal{NP}$-hard problems can only admit deterministic solutions that take an unreasonable (i.e. exponential) amount of time, and they require (unattainable) nondeterminism in order to achieve reasonable (i.e. polynomial) running times.

The central idea used to demonstrate $\mathcal{NP}$-hardness evolves around the $\mathcal{NP}$-complete problems. A problem is said to be $\mathcal{NP}$-complete if every decision problem in $\mathcal{NP}$ is polynomial-time reducible to it. This means that the $\mathcal{NP}$-complete problems are as hard as any decision problem in $\mathcal{NP}$. Given two decision problems P_1 and P_2, P_1 is said to be polynomial-time reducible to P_2 (written as $\mathrm{P}_1 \leq_p \mathrm{P}_2$), if there exists a polynomial time algorithm R which transforms every input x for P_1 into an equivalent input $R(x)$ for P_2. By equivalent we mean that the answer produced by P_2 on input $R(x)$ is always the same as the answer P_1 produces on input x. Thus, any algorithm which solves P_2 in polynomial time can be used to solve P_1 on input x in polynomial time by simply computing $R(x)$, and then running

P_2. In order to show that a particular (control) decision problem P_2 is $\mathcal{NP}$-complete, one starts with a problem P_1 in $\mathcal{NP}$-complete, and attempts to show that $\mathrm{P}_1 \leq_p \mathrm{P}_2$. This shows that P_2 is $\mathcal{NP}$-hard. To complete the proof that P_2 is $\mathcal{NP}$-complete, it must be demonstrated that a candidate solution can be verified in polynomial time. In control theory, researchers have followed this "reduction" method to study the computational difficulty of some decidable problems and many decidable control problems have been shown to be $\mathcal{NP}$-complete (or $\mathcal{NP}$-hard) [29], [30], [31], [32], [33]. A recent overview of the computational complexity of many control problems may be found in [29].

The problem of simultaneous stabilization of N given linear systems with a LTI dynamic compensator is as previously mentioned rationally undecidable for $N > 2$ [15]. However, restricting the stabilizing compensator to be static (or dynamic but of a given order) makes the problem decidable (although inefficiently) using the Tarski approach as discussed before. So the question becomes: how do we deal with decidable but inefficient control problems? And moreover, can we deal with undecidable control problems? We actually have two possibilities in attempting to answer both questions:

1. Limit the class of systems (such as to linear, minimum-phase, passive systems, etc.). This is typically the approach taken by control designers.
2. Soften the goal for the class of systems we are interested in. This is a more recent idea in control pioneered in [9], [2], [34], [13], [35]. An example of goal softening is the randomized algorithms approach discussed next.

A re-formulation of LTI control problems may then be as follows [36],

Problem 2: Given a closed-loop system $T(s, X, Y)$ with a performance measure $\Psi(X, Y)$, where X, Y are random real-valued vectors, find a vector Y_0, if one exists, of controller parameters which has a high probability of minimizing the expected value with respect to X of an appropriate function $f(X, Y)$ of $\Psi(X, Y)$.

The related decision problem is to ascertain the existence of a vector Y_0 such that a certain level γ is achieved by $\mathbb{E}f(X, Y)$. Note that our problem has been changed from a deterministic decision problem to a probabilistic optimization problem. Also note that the randomness of X and Y is used to open the door for Monte-Carlo and statistical learning methods. Finally, we have converted a worst-case scenario (guaranteed-cost) into an average-case problem.

In the context of stabilization, let $\Psi(X, Y) = 0$ if $T(s, X, Y)$ is stable and $\Psi(X, Y) = 1$ otherwise. By minimizing $\mathbb{E}f(X, Y)$ we are actually maximizing the volume (or number in case of finite number of plants) which may be stabilized with $C(s, Y_0)$. In fact, let

$$f_Y(X) = f(X, Y) = \begin{cases} 1 & \Psi(X, Y) = 1 \\ 0 & \Psi(X, Y) = 0 \end{cases}$$

and $\mathcal{F} = \{f_Y(\cdot) : Y \in \mathcal{Y}\}$. The purpose of control is to choose Y_0, and thus the corresponding controller $C(s, Y_0)$ to stabilize the maximum number of plants. Note that if the structure and the order of $C(s, Y)$ are fixed, then the problem reduces to finding the set of parameters Y. This objective may be achieved by minimizing the expected value $\mathbb{E}[f_Y(X)]$. An interpretation of the minimization of the expectation $\mathbb{E}[f_Y(X)]$ is that we can then ascertain with confidence $1 - \mathbb{E}[f_{Y_0}(X)]$ that the controller $C(s, Y_0)$ stabilizes a random plant $G(s, X)$.

One limitation of this approach is that in practice, we do not have the necessary information to calculate $\mathbb{E}[f_Y]$ since all we have are sample plants and compensators. Moreover, how do we minimize $\mathbb{E}[f_Y]$ when all we have are the values of f at sample points? In [36], the empirical mean of $f_Y(X)$ is used instead of $\mathbb{E}[f_Y]$ for a given $Y \in \mathcal{Y}$,

$$\frac{1}{n} \sum_{j=1}^{n} f_Y(X_j), \tag{1}$$

which then leaves us with two questions:

1. Will $\frac{1}{n} \sum_{j=1}^{n} f_Y(X_j)$ be a good approximation of $\mathbb{E}[f_Y]$ *uniformly* in Y as n increases?

2. Will the minimum of $\frac{1}{n}\sum_{j=1}^{n} f_Y(X_j)$, obtained empirically as

$$\min_{1\le i\le m}\left[\frac{1}{n}\sum_{j=1}^{n} f_{Y_i}(X_j)\right]$$

be close to the actual minimum of $\frac{1}{n}\sum_{j=1}^{n} f_Y(X_j)$ as m increases?

It turns out that the first question has been studied thoroughly in the theory of empirical process and statistical learning theory. Minimization of a function defined by equation (1) in particular is a case of empirical risk minimization as discussed in the next section. Note that there are actually two separate questions to answer: a question of empirical averaging, and a question of empirical minimization. The empirical average question depends on the number n of plants, while the minimization question depends on both the number of plants n and the number of controllers m. Our main results in this paper offer a significant reduction in n but not in m. Our future papers will address the minimization problem and how to reduce m further. We will next review relevant results from Statistical Learning Theory and randomized algorithms.

III. Statistical Learning Theory

The basic notions of Probability Theory used in the paper can be found in any textbook on Advanced Probability, see, for instance, [**?**]. More special results on empirical processes and statistical learning theory can be found in [37], [**?**], [38], [39], [13]. We present now an overview of standard learning theory concepts and results obtained in [36] along with their application to control problems.

Let $(S,\mathcal{A})$ be a measurable space and let $\{X_n\}_{n\ge1}$ be a sequence of independent identically distributed (i.i.d) observations in this space with common distribution P. We assume that this sequence is defined on a probability space $(\Omega,\Sigma,\mathbb{P})$. Denote by $\mathcal{P}(S) := \mathcal{P}(S,\mathcal{A})$ the set of all probability measures on $(S,\mathcal{A})$. Suppose $\mathcal{P}\subset\mathcal{P}(S)$ is a class of probability distributions such that $P\in\mathcal{P}$. In particular, if one has no prior knowledge about P, then $\mathcal{P}=\mathcal{P}(S)$. In this case, we are in the setting of *distribution free learning.* One of the central problems of statistical learning theory is *the risk minimization problem.* It is crucial in all cases of learning (standard concept or function learning, regression problems, pattern recognition, etc.). It also plays an important role in randomized (Monte Carlo) algorithms for robust control problems, as has been shown by Vidyasagar [14] and as we will see in this paper. Given a class $\mathcal{F}$ of $\mathcal{A}$-measurable functions f from S into $[0,1]$ (e.g., decision rules in a pattern recognition problem or performance indices in control problems), the risk functional is defined as

$$R_P(f) := P(f) := \int_S f dP := \mathbb{E}f(X),\ f\in\mathcal{F}.$$

The goal is to find a function f_P that minimizes R_P on $\mathcal{F}$. Typically, the distribution P is unknown (or, as it occurs in many control problems, the integral of f with respect to P is too hard to compute) and the solution of the risk minimization problem is to be based on a sample $(X_1,\dots,X_n)$ of independent observations from P. In this case, the goal of statistical learning is more modest: given $\varepsilon>0,\delta\in(0,1)$, find an estimate $\hat{f}_n\in\mathcal{F}$ of f_P, based on the data $(X_1,\dots,X_n)$, such that

$$\sup_{P\in\mathcal{P}}\mathbb{P}\{R_P(\hat{f}_n)\ge\inf_{f\in\mathcal{F}}R_P(f)+\varepsilon\}\le\delta. \tag{2}$$

In other words, one can write that with probability $1-\delta$, $R_P(\hat{f}_n)$ is within ε of $\inf_{f\in\mathcal{F}}R_P(f)=R^*$. Denote by $\tilde{N}^L_{\mathcal{F},\mathcal{P}}(\varepsilon;\delta)$ the minimal number $n\ge1$ such that for some estimate $\hat{f}_n$ the bound (2) holds, and let $\tilde{N}^U_{\mathcal{F},\mathcal{P}}(\varepsilon;\delta)$ be the minimal number $N\ge1$ such that for some sequence of estimates $\{\hat{f}_n\}$ and for all $n\ge N$ the bound (2) holds. Let us call the quantity $\tilde{N}^L_{\mathcal{F},\mathcal{P}}(\varepsilon;\delta)$ *the lower sample complexity* and the quantity $\tilde{N}^U_{\mathcal{F},\mathcal{P}}(\varepsilon;\delta)$ *the upper sample complexity* of learning. These quantities show how much data we need in order to guarantee certain accuracy ε of learning with certain confidence level $1-\delta$. Clearly, $\tilde{N}^L_{\mathcal{F},\mathcal{P}}(\varepsilon;\delta)\le\tilde{N}^U_{\mathcal{F},\mathcal{P}}(\varepsilon;\delta)$, and it is easy to show that the inequality can be strict. The upper sample complexity is used rather frequently in statistical learning theory and is usually referred to simply as the sample complexity. But in this paper we will deal more with the lower sample complexity.

A method of *empirical risk minimization* is widely used in learning theory. Namely, the unknown distribution P is replaced by *the empirical measure* P_n, defined as

$$P_n(A) := \frac{1}{n}\sum_{k=1}^{n} I_A(X_k),\ A \in \mathcal{A}$$

where $I_A(x) = 1$ for $x \in A$ and $I_A(x) = 0$ for $x \notin A$. The risk functional R_P is replaced by the empirical risk R_{P_n}, defined by

$$R_{P_n}(f) := P_n(f) := \int_S f dP_n := \frac{1}{n}\sum_{k=1}^{n} f(X_k),\ f \in \mathcal{F}.$$

The problem is now to minimize the empirical risk R_{P_n} on $\mathcal{F}$, and we let $f_{P_n} \in \mathcal{F}$ be a function that minimizes R_{P_n} on $\mathcal{F}$.

In what follows, f_{P_n} is used as our learning algorithm, i.e. $\hat{f}_n := f_{P_n}$. Determining the sample complexity of the empirical risk minimization method is definitely one of the central and most challenging problems of statistical learning theory (see, e.g., [37], or Vidyasagar [36] for the relevant discussion in the context of robust control problems). A reasonable upper bound for the sample complexity can be obtained by finding the minimal value of n for which the expected value $\mathbb{E}f(X)$ is approximated uniformly over the class $\mathcal{F}$ by the empirical means with given accuracy ε and confidence level $1-\delta$. More precisely, denote

$$N(\varepsilon,\delta) := N^L_{\mathcal{F},\mathcal{P}}(\varepsilon,\delta) := \min\Big\{n \geq 1 : \sup_{P\in\mathcal{P}} \mathbb{P}\big\{\|P_n - P\|_{\mathcal{F}} \geq \varepsilon\big\} \leq \delta\Big\},$$

where $\|\cdot\|_{\mathcal{F}}$ is the sup-norm in the space $\ell^\infty(\mathcal{F})$ of all uniformly bounded functions on $\mathcal{F}$. Let us call the quantity $N(\varepsilon;\delta)$ *the (lower) sample complexity of empirical approximation* on the class $\mathcal{F}$. Then, clearly, $N^L_{\mathcal{F},\mathcal{P}}(\varepsilon/2;\delta) \geq \tilde{N}^L_{\mathcal{F},\mathcal{P}}(\varepsilon;\delta)$. To see this, it is enough to consider the following,

$$\begin{aligned} 0 &\leq R_P(f_{P_n}) - \inf_{f\in\mathcal{F}} R_P(f) \\ &\leq P(f_{P_n}) - P_n(f_{P_n}) + \inf_{f\in\mathcal{F}} P_n(f) - \inf_{f\in\mathcal{F}} P(f) \\ &\leq 2\|P_n - P\|_{\mathcal{F}}. \end{aligned} \tag{3}$$

Unfortunately, the quantity $N^L_{\mathcal{F},\mathcal{P}}(\varepsilon,\delta)$ is itself unknown for most of the nontrivial examples of function classes, and only rather conservative upper bounds for this quantity are available. These bounds are expressed in terms of various entropy characteristics and combinatorial quantities, such as VC-dimensions, which themselves are not always known precisely and are replaced by their upper bounds [36]. Going back to our control motivation, we note that our problem involves also the finding of the minimum of a certain performance objective or more precisely, finding the controller parameters which correspond to such minimum. This is the second separate question mentioned at the end of Section II and refers to the optimization part of the problem which we approach in the same manner as Vidyasagar. In [36], Vidyasagar introduced the following types of minima, in order to use statistical learning theory to design fixed-order robust controllers, which minimize the performance index in Problem 2.

Definition 1: Let $R : \mathcal{Y} \longrightarrow \mathbb{R}$ and $\varepsilon > 0$ be given. A number $R_0 \in \mathbb{R}$ is said to be an *approximate near minimum of R to accuracy ε* if

$$\left| R_0 - \inf_{Y\in\mathcal{Y}} R(Y) \right| \leq \varepsilon$$

■

Definition 2: Suppose $R : \mathcal{Y} \longrightarrow \mathbb{R}$, Q is a given probability measure on $\mathcal{Y}$, and $\alpha > 0$ be given. A number $R_0 \in \mathbb{R}$ is a *probable near minimum of R to level α* if there exists a measurable set $\mathcal{S} \subseteq \mathcal{Y}$ with $Q(\mathcal{S}) \leq \alpha$ such that

$$\inf_{Y\in\mathcal{Y}} R(Y) \leq R_0 \leq \inf_{Y\in\mathcal{Y}\setminus\mathcal{S}} R(Y).$$

where $\mathcal{Y} \setminus \mathcal{S}$ is the complement of the set $\mathcal{S}$ in $\mathcal{Y}$. ■

Definition 3: Suppose $R : \mathcal{Y} \longrightarrow \mathbb{R}$, Q is a given probability measure on $\mathcal{Y}$, and $\alpha > 0$, $\varepsilon > 0$ be given. A number $R_0 \in \mathbb{R}$ is a *probably approximate near minimum of R to accuracy ε and level α* if there exists a measurable set $\mathcal{S} \subseteq \mathcal{Y}$ with $Q(\mathcal{S}) \leq \alpha$ such that

$$\inf_{Y \in \mathcal{Y}} R(Y) - \varepsilon \leq R_0 \leq \inf_{Y \in \mathcal{Y} \setminus \mathcal{S}} R(Y) + \varepsilon.$$

■

Note in particular that the last 2 types of minima while useful in practice may not give an accurate picture of the performance of the closed-loop control system. Finally, let us define a version of probably approximate near minima in the case of a stochastic process R (say, $R := R_{P_n}$, see the definition above) as follows.

Definition 4: Suppose that $R : \mathcal{Y} \rightarrow \mathbb{R}$ is a stochastic process, that Q is a given probability measure on $\mathcal{Y}$, and that $\alpha \in (0,1)$, $\delta \in (0,1)$ and $\varepsilon > 0$ are given. A number R_0 is a *probably approximate near minimum of R with confidence* $1 - \delta$*, level α and accuracy ε*, if

$$\mathbb{P}\left\{ \inf_{Y \in \mathcal{Y}} R(Y) - \varepsilon \leq R_0 \leq \inf_{Y \in \mathcal{Y} \setminus \mathcal{S}} R(Y) + \varepsilon \right\} \geq 1 - \delta$$

with some measurable set $\mathcal{S} \subseteq \mathcal{Y}$ such that $Q(\mathcal{S}) \leq \alpha$. ■

An interpretation of definitions 2, 3, 4 is that we are not searching for the minimum over all of the set $\mathcal{Y}$ but only over its subset $\mathcal{Y} \setminus \mathcal{S}$, where $\mathcal{S}$ has a small measure (at most α). Unless the actual infimum R^* is attained in the exceptional set $\mathcal{S}$, R_0 is within ε from the actual infimum with confidence $1 - \delta$. It is exactly this goal softening that gets around the computational difficulty of these problems [35]. Although using Monte Carlo type minimization, it is unlikely to obtain a better estimate of R^* than R_0 (since the chances of getting into the set $\mathcal{S}$ are small), nothing can be said in practice about the size of the difference $R_0 - R^*$. Vidyasagar in [36] then proposes an "efficient" algorithm as follows.

Algorithm 1: Given:

- Sets $\mathcal{X}$ and $\mathcal{Y}$,
- Probability measures P on $\mathcal{X}$ and Q on $\mathcal{Y}$,
- A measurable function $f : \mathcal{X} \times \mathcal{Y} \longrightarrow [0,1]$, and
- An accuracy parameter $\varepsilon \in (0,1)$, a level parameter $\alpha \in (0,1)$, and a confidence parameter $\delta \in (0,1)$.

Let $R_P(\cdot) = \mathbb{E}_P[f(X, \cdot)]$ and denote $\mathcal{F} := \{f_Y : Y \in \mathcal{Y}\}$,

$$\begin{aligned} R_{P_n}(\cdot) &= \frac{1}{n} \sum_{j=1}^{n} f(X_j, \cdot) \\ q(n, \varepsilon, \mathcal{F}) &= \mathbb{P}\{\sup_{Y \in \mathcal{Y}} |R_{P_n}(Y) - R_P(Y)| > \varepsilon\}. \end{aligned}$$

Then, choose n and m such that

$$\begin{aligned} m &\geq \frac{\log(2/\delta)}{\log[1/(1-\alpha)]} \\ q(n, \varepsilon, \mathcal{F}) &\leq \delta/2 \end{aligned} \qquad (4)$$

and generate i.i.d. samples $Y_1, Y_2, \cdots, Y_m \in \mathcal{Y}$ from the distribution Q and $X_1, X_2, \cdots, X_n \in \mathcal{X}$ from the distribution P. Then let,

$$R_0 = \min_{1 \leq i \leq m} R_{P_n}(Y_i)$$

Then with confidence at least $1 - \delta$, R_0 is a probably approximate near minimum of $R_P(Y)$ to level α and accuracy ε. ◇

To guarantee the existence of n such that $q(n, \varepsilon, \mathcal{F}) \leq \delta/2$, in Algorithm 1 one can assume that $\mathcal{F}$ is a Glivenko-Cantelli class for P (see [?], [38] for the definition). The UCEM property considered in [13] means that for all $\varepsilon > 0$ $q(n, \varepsilon, \mathcal{F}) \to 0$ as $n \to \infty$; it is equivalent to the Glivenko-Cantelli property of the class $\mathcal{F}$. Note that in Algorithm 1, the bound on the quantity $q(n, \varepsilon, \mathcal{F})$ is no longer dependent on m.

Sufficient conditions for satisfying Glivenko-Cantelli (UCEM) property, which are convenient for the purposes of control theory, can be formulated in terms of the finiteness of VC-dimensions or P-dimensions of the class $\mathcal{F}$, [13], [36].

Definition 5: Let $\mathcal{C}$ be a family of subsets of $\mathcal{X}$. A finite set $F = \{x_1, \cdots x_n\} \subset \mathcal{X}$ is shattered by $\mathcal{C}$, if for every subset B of the 2^n subsets of F, there exists a set $A \in \mathcal{C}$ such that $F \cap A = B$. The Vapnik-Chervonenkis dimension of $\mathcal{C}$ denoted *VC-dim*$(\mathcal{C})$ is the largest integer n such that there exists a set F of cardinality n shattered by $\mathcal{C}$. ■

Given a class $\mathcal{F}$ of functions mapping $\mathcal{X}$ into $\{0, 1\}$, one can consider the class of sets $\mathcal{C} := \{\{x : f(x) = 1\} : f \in \mathcal{F}\}$ and define the VC-dimension of $\mathcal{F}$ as *VC-dim*$(\mathcal{C})$. It will be also denoted *VC-dim*$(\mathcal{F})$. The role of P-dimension (see e.g. [13]) is similar in the case of more general classes of functions. In particular, one can consider the class $\mathcal{F}_{k,l,r,t}$ arising from our MPIs and defined as follows. Given polynomials $p_1(X, Y), \ldots, p_t(X, Y)$ on $\mathbb{R}^k \times \mathbb{R}^l$ of degree $\leq r$ (with respect to Y), consider all the Boolean formulae obtained from expressions "$p_j(X, Y) > 0$", $j = 1, \ldots, t$ using the standard logical operations $\vee, \wedge, \neg$. Let $\Phi_{k,l,r,t}$ be the set of all such formulae. Each formula $\phi \in \Phi_{k,l,r,t}$ defines the function $f := f_\phi$ that takes value 1 if the formula is true and value 0 otherwise. We set $\mathcal{F}_{k,l,r,t} := \{f_\phi : \phi \in \Phi_{k,l,r,t}\}$. This class can be used to describe the control decidability questions.

We then have the following theorems that go back to the original work of Vapnik and Chervonenkis [39], [?] and that were used in [36].

Theorem 1: Let $\mathcal{F}$ be a family of measurable functions from $\mathcal{X}$ into $\{0, 1\}$ and suppose that VC-dim$(\mathcal{F}) \leq d < \infty$. Then, $\mathcal{F}$ has the UCEM property and moreover,

$$q(n, \varepsilon, \mathcal{F}) \leq 4 \left(\frac{2en}{d}\right)^d \exp(-n\varepsilon^2/8); \quad \forall n, \varepsilon \tag{5}$$

■

This then leads to the following bound on the sample complexity of empirical approximation on the class $\mathcal{F}$.

Theorem 2: Let $\mathcal{F}$ be a family of measurable functions from $\mathcal{X}$ into $\{0, 1\}$ and suppose that VC-dim$(\mathcal{F}) \leq d < \infty$. Let P be an arbitrary probability measure on $\mathcal{X}$, and let $\varepsilon, \delta \in (0, 1)$ be arbitrary constants. Then, $q(n, \varepsilon, \mathcal{F}) \leq \delta$ if

$$n \geq \max\left\{\frac{16}{\varepsilon^2}\log\frac{4}{\delta}, \frac{32d}{\varepsilon^2}\log\frac{32e}{\varepsilon^2}\right\}$$

■

The next theorem gives an upper bound for the VC-dimension of the class $\mathcal{F}_{k,l,r,t}$ and is due essentially to Karpinski and Macintyre [?]. We cite it from [13].

Theorem 3: The following upper bound holds:

$$VC - dim(\mathcal{F}_{k,l,r,t}) \leq 2l\log(4ert).$$

■

IV. Sequential Learning Algorithms

In this section, we present sequential algorithms for a general problem of empirical risk minimization. They are designed to overcome some of the difficulties encountered with the standard learning methods of Section III. This approach does not depend on the explicit calculation of the VC-dimension, although its finiteness remains critical to the termination of the design algorithm, in the distribution-free learning case. The sequential algorithms chosen are based on *Rademacher bootstrap* although other bootstrap techniques, developed in statistics (for instance, standard Efron bootstrap or various versions of weighted bootstrap), can also be adopted for our purposes. An important feature of our approach is the randomness of the sample size for which a given accuracy of learning is achieved with a guaranteed probability. Thus, the sample complexity of our method of learning is rather a random variable. Its value is not known in advance and is to be determined in the process of learning. The lower bound for this random variable is the value of the sample size which the sequential learning algorithm starts working with. The upper bounds for the random sample complexity are of the same order of magnitude as the standard conservative upper bounds for the sample complexity of empirical risk minimization algorithms. Thus, *in the worst case,* the sequential method of learning would take as much time (up to a numerical constant) as the standard methods do.

We start with some basic definitions. The proofs of all statements of this section can be found in [**?**].

Definition 6: Let $\{\Sigma_n\}_{n\geq 1}$ be a filtration of σ-algebras (i.e. for all $n \geq 1$ $\Sigma_n \subset \Sigma_{n+1}$) such that $\Sigma_n \subset \Sigma, n \geq 1$ and X_n is Σ_n-measurable. Less formally, Σ_n consists of the events that occur by time n (in particular, the value of random variable X_n is known by time n). A random variable τ, taking positive integer values, will be called *a stopping time* if and only if (iff), for all $n \geq 1$, we have $\{\tau = n\} \in \Sigma_n$. In other words, the decision whether $\tau \leq n$, or not, depends only on the information available by time n. ■

Given $\varepsilon > 0$ and $\delta \in (0, 1)$, let $\bar{n}(\varepsilon, \delta)$ denote the initial sample size of our learning algorithms. We assume that $\bar{n}$ is a non-increasing function in both ε and δ. Denote by $\mathcal{T}(\varepsilon, \delta) := \mathcal{T}_{\mathcal{F},\mathcal{P}}(\varepsilon, \delta)$ the set of all stopping times τ such that $\tau \geq \bar{n}(\varepsilon; \delta)$ and

$$\sup_{P\in\mathcal{P}} \mathbb{P}\big\{\|P_\tau - P\|_{\mathcal{F}} \geq \varepsilon\big\} \leq \delta.$$

If now $\tau \in \mathcal{T}(\varepsilon, \delta)$ and $\hat{f} := f_{P_\tau}$ is a function that minimizes the empirical risk based on the sample $(X_1, \ldots, X_\tau)$ then a bound similar to (4) immediately implies that

$$\sup_{P\in\mathcal{P}} \mathbb{P}\Big\{R_P(f_{P_\tau}) \geq \inf_{f\in\mathcal{F}} R_P(f) + 2\varepsilon\Big\} \leq \delta.$$

The questions, though, are how to construct a stopping time from the set $\mathcal{T}(\varepsilon, \delta)$, based only on the available data (without using the knowledge of P) and which of the stopping times from this set is best used in the learning algorithms. The following definition will be useful in this connection.

Definition 7: A parametric family of stopping times $\{\nu(\varepsilon, \delta) : \varepsilon > 0, \delta \in (0, 1)\}$ is called *strongly (statistically) efficient* for the class $\mathcal{F}$ with respect to $\mathcal{P}$ iff there exist constants $K_1 \geq 1, K_2 \geq 1$ and $K_3 \geq 1$ such that for all $\varepsilon > 0$ and $\delta \in (0, 1)$

$$\nu(\varepsilon, \delta) \in \mathcal{T}(K_1\varepsilon, \delta)$$

and for all $\tau \in \mathcal{T}(\varepsilon, \delta)$

$$\sup_{P\in\mathcal{P}} \mathbb{P}\big\{\nu(K_2\varepsilon, \delta) > \tau\big\} \leq K_3\delta.$$

■

Thus, using strongly efficient stopping time $\nu(\varepsilon; \delta)$ allows one to solve the problem of empirical approximation with confidence $1 - \delta$ and accuracy $K_1\varepsilon$. With probability at least $1 - K_3\delta$, the time required by this algorithm is less than the time needed for *any* sequential algorithm of empirical approximation with accuracy ε/K_2 and confidence $1 - \delta$.

Definition 8: We call a family of stopping times $\{\nu(\varepsilon, \delta) : \varepsilon > 0, \delta \in (0, 1)\}$ *weakly (statistically) efficient* for the class $\mathcal{F}$ with respect to $\mathcal{P}$ iff there exist constants $K_1 \geq 1, K_2 \geq 1$ and $K_3 \geq 1$ such that for all $\varepsilon > 0$ and $\delta \in (0, 1)$

$$\nu(\varepsilon, \delta) \in \mathcal{T}(K_1\varepsilon, \delta)$$

and

$$\sup_{P\in\mathcal{P}} \mathbb{P}\big\{\nu(K_2\varepsilon, \delta) > N(\varepsilon; \delta)\}\big\} \leq K_3\delta.$$

■

Using weakly efficient stopping time $\nu(\varepsilon;\delta)$ also allows one to solve the problem of empirical approximation with accuracy $K_1\varepsilon$ and confidence $1-\delta$. With probability at least $1-K_3\delta$, the time required by this algorithm, is less than the sample complexity of empirical approximation with accuracy ε/K_2 and confidence $1-\delta$.

Note that, under the assumption $N(\varepsilon;\delta) \geq \bar{n}(\varepsilon;\delta)$, we have $N(\varepsilon,\delta) \in \mathcal{T}(\varepsilon,\delta)$. Hence, any strongly efficient family of stopping times is also weakly efficient. The converse to this statement is not true. We show below how to construct efficient stopping times for empirical risk minimization problems. The construction is based on a version of bootstrap. Let $\{r_n\}_{n\geq 1}$ be *a Rademacher sequence* (i.e. a sequence of i.i.d. random variables taking values $+1$ and -1 with probability $1/2$ each). We assume, in addition, that this sequence is independent of the observations $\{X_n\}_{n\geq 1}$. Suppose that (with $\lfloor\cdot\rfloor$ denoting the floor of the argument)

$$\bar{n}(\varepsilon,\delta) \geq \left\lfloor \frac{4}{\varepsilon^2}\log(\frac{2}{\delta(1-e^{-\varepsilon^2/4})}) \right\rfloor + 1.$$

Let

$$\nu(\varepsilon,\delta) := \nu_{\mathcal{F}}(\varepsilon,\delta) := \min\{n \geq \bar{n}(\varepsilon,\delta) : \|n^{-1}\sum_{j=1}^{n} r_j\delta_{X_j}\|_{\mathcal{F}} \leq \varepsilon\}.$$

where $\delta_x(f) := f(x)$. Note that for all $\varepsilon > 0$ and for all $\delta \in (0,1)$, $\nu(\varepsilon,\delta)$, is a stopping time and it can be computed by Monte Carlo simulation of the sequence $\{r_j\}_{j\geq 1}$. The finiteness with probability 1 of the stopping time $\nu(\varepsilon;\delta)$ (and other stopping times, defined below) can be shown to follow from the Glivenko-Cantelli property for the class $\mathcal{F}$.

Theorem 4: $\{\nu_{\mathcal{F}}(\varepsilon,\delta) : \varepsilon > 0, \delta \in (0,1)\}$ is a strongly efficient family of stopping times for any class $\mathcal{F}$ of measurable functions from S into $[0,1]$ with respect to the set $\mathcal{P}(S)$ of all probability distributions. ■

The initial time of the previous algorithm could be too large if ε is very small. Here we construct another version of sequential risk minimization algorithm with smaller initial time. Define

$$\nu(\varepsilon,\delta) := \nu_{\mathcal{F}}(\varepsilon,\delta) := \min\{n : \|n^{-1}\sum_{j=1}^{n} r_j\delta_{X_j}\|_{\mathcal{F}} \leq \varepsilon,\ n := n_k := 2^k\bar{n}(\varepsilon,\delta),\ k = 0,1,\dots\}.$$

Theorem 5: Suppose that

$$\bar{n}(\varepsilon,\delta) \geq \left\lfloor \frac{4}{\varepsilon^2}\log(\frac{4}{\delta}) \right\rfloor + 1.$$

Then, for all $\varepsilon > 0, \delta \in (0,1)$,

1. $\nu(\varepsilon;\delta) \in \mathcal{T}(K_1\varepsilon;\delta)$ with $K_1 = 5$.
2. Moreover, suppose that

$$N(\varepsilon,\delta) \geq \bar{n}(\varepsilon,\delta) \geq \left\lfloor \frac{4}{\varepsilon^2}\log(\frac{4}{\delta}) \right\rfloor + 1.$$

Then $\{\nu_{\mathcal{F}}(\varepsilon,\delta) : \varepsilon > 0, \delta \in (0,1/2)\}$ is a weakly efficient family of stopping times for any class $\mathcal{F}$ of measurable functions from S into $[0,1]$ with respect to the set $\mathcal{P}(S)$ of all probability distributions on S.

■

The next proposition shows that if the family of stopping times defined above starts too late (namely, after the time $N(\varepsilon;\delta)$), then the stopping time is close to the initial time with high probability.

Proposition 1: Suppose that

$$\bar{n}(\varepsilon,\delta) \geq \left\lfloor \frac{4}{\varepsilon^2}\log(\frac{4}{\delta}) \right\rfloor + 1$$

and

$$12/\varepsilon \leq N(\varepsilon,\delta) \leq \bar{n}(\varepsilon,\delta).$$

Then, there exist constants $K_1 \geq 1, K_2 \geq 1$ such that

$$\sup_{P \in \mathcal{P}(S)} \mathbb{P}\{\nu_{\mathcal{F}}(K_1\varepsilon;\delta) > K_2 \bar{n}(\varepsilon;\delta)\} \leq \delta. \tag{6}$$

■

Based on the randomized algorithm introduced in section III, and on the sequential learning algorithm of this section, a probably approximate near minimum of f with confidence $1-\delta$, level α and accuracy ε, can be found with the following algorithm.

Algorithm 2 (Search for the Statistical Optimal Controller) Given:

- Sets $\mathcal{X}$ and $\mathcal{Y}$,
- Probability measures P on $\mathcal{X}$ and Q on $\mathcal{Y}$,
- A measurable function $f : \mathcal{X} \times \mathcal{Y} \longrightarrow [0,1]$, and
- An accuracy parameter $\varepsilon \in (0,1)$, a level parameter $\alpha \in (0,1)$, and a confidence parameter $\delta \in (0,1)$.

Let $R_P(\cdot) = \mathbb{E}_P[f(X,\cdot)]$ and

$$R_{P_n}(\cdot) = \frac{1}{n}\sum_{j=1}^{n} f(X_j,\cdot)$$

Then,

1. Choose m independent controllers with parameters having distribution Q where
$$m \geq \frac{\log(2/\delta)}{\log[1/(1-\alpha)]} \tag{7}$$
2. Choose n independent plants with parameters having distribution P, where
$$n = \left\lfloor \frac{4K_1^2}{\varepsilon^2}\log\left(\frac{8}{\delta}\right)\right\rfloor + 1 \tag{8}$$
with $K_1 = 5$
3. Evaluate the stopping variable
$$\gamma = \max_{1\leq i\leq m}\left|\frac{1}{n}\sum_{j=1}^{n} r_j f(X_j,Y_i)\right|$$
where r_j are *Rademacher* random variables, i.e. independent identically distributed random variables (also independent of the plant sample) taking values $+1$ and -1 with probability $1/2$ each.
If $\gamma > \frac{\varepsilon}{K_1}$, add n more independent plants with parameters having distribution P to the plant samples, set $n := 2n$ and repeat step 3
4. Choose the controller which minimizes the cost function R_{P_n}. Then with confidence at least $1-\delta$, this controller minimizes R_P to a level α and accuracy ε.

◇

Note that Algorithm 2 corresponds to Theorem 5 and other variations on this algorithm are possible.

V. Applications To Control Design

Example 1: In this example we consider a modified version of the control problem originally presented by Doyle et al. in [40] (Section 12.4).

Let us consider the feedback system in Figure 2. The plant $G(s)$ is a simplified model of a flexible beam. The input is the voltage to a power amplifier while the output is the tip deflection of the beam. The transfer function $G(s)$ is given by

$$G(s) = \frac{-6.4750s^2 + 4.0302s + 175.7700}{s(5s^3 + 3.5682s^2 + 139.5021s + 0.0929)}. \tag{9}$$

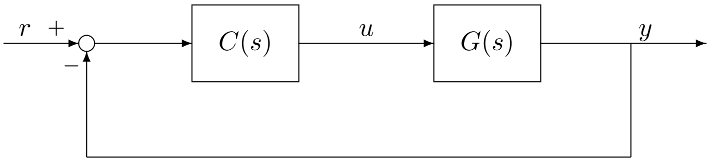

Fig. 2. The closed-loop system for Example 1

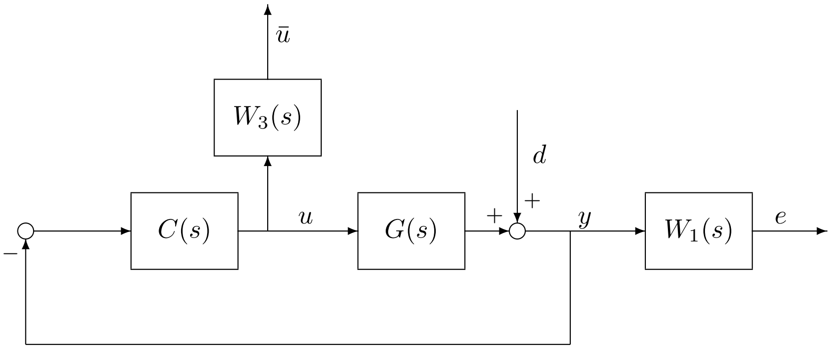

Fig. 3. The closed-loop system with the weighting functions

In [40] the author designed a controller $C(s)$ considering some time domain specifications and an amplitude constraint on the plant input. In order to use the $\mathcal{H}_\infty$ approach, some appropriate weighting functions $W_1(s)$ and $W_3(s)$ were chosen and the problem was reformulated as the minimization of the mixed-sensitity norm (see Figure 3)

$$\left\| \begin{pmatrix} W_1 S \\ W_3 C S \end{pmatrix} \right\|_\infty \tag{10}$$

where S is the sensitivity function $S = (1+GC)^{-1}$. After an iterative procedure, the weighting functions W_1 and W_3 turned out to be

$$W_1(s) = 0.8 \frac{s^2 + 1.2s + 1}{(s+0.001)(s+1.2)(0.001s+1)}$$

$$W_3(s) = \frac{s+\frac{1}{3}}{s+\frac{100}{3}}$$

The suboptimal controller found in [40] achieves an $\mathcal{H}_\infty$ norm of the mixed sensitivity (10) of about 0.938 and has order eight, the same order of the augmented plant (plant plus weighting function).

The same example was reconsidered in [12], where the authors tried to design a fixed-structure, third order controller with a randomized algorithm. After fixing some ranges for the controller zeros, poles and gain, $10,000$ controllers were randomly generated and the controller minimizing the norm (10) was chosen. This controller resulted to achieve an $\mathcal{H}_\infty$ norm of the mixed sensitivity (10) of about 1.02, not far from the optimal one.

Hereafter what we will do is solve a *robust* control problem using our randomized algorithm and then we will interpret the result using classical analysis tools.

First of all let us reconsider the beam transfer function (9) where we introduce uncertainties on the gain, on the damping factor and on the natural frequency of the two complex poles, which model the first flexible mode

$$\hat{G}(s) = K \frac{-6.4750s^2 + 4.0302s + 175.7700}{5s(s+6.66\cdot 10^{-4})(s^2+2\zeta\omega_n s+\omega_n^2)} \tag{11}$$

where the original plant $G(s)$ is recovered letting $K = 1$, $\zeta = 0.0675$, $\omega_n = 5.28$.

Rewriting (11) as

$$\hat{G}(s) = K\frac{-6.4750s^2 + 4.0302s + 175.7700}{5s(s + 6.66 \cdot 10^{-4})(s^2 + As + B)} \tag{12}$$

we consider the following intervals for the coefficients K, A and B

$$K \in (0.5, 1.5), \quad A \in (0, 1.4259), \quad B \in (19.53, 36.27). \tag{13}$$

which result in the following ranges for the original parameters ζ and ω_n

$$\zeta \in (0, 0.1184), \quad \omega_n \in (4.4176, 6.0201).$$

What we intend to find is a third order controller which minimizes the mixed sensitivity norm (10) over all the uncertain plants (11), guaranteeing at the same time some nominal performance

$$\min \left\| \begin{pmatrix} W_1(1 + \hat{G}C)^{-1} \\ W_3C(1 + \hat{G}C)^{-1} \end{pmatrix} \right\|_\infty \text{ subject to } \quad \left\| \begin{pmatrix} W_1(1 + GC)^{-1} \\ W_3C(1 + GC)^{-1} \end{pmatrix} \right\|_\infty < 2 \tag{14}$$

There is no analitycal way of solving this problem. A possible *classical* way to approach the problem could be to use the μ-synthesis to design a controller and then reduce its order hoping that its optimal properties would be retained. Hereafter we will try a randomized approach basing on our algorithm. As in [12], we look for a stable, minimum-phase, strictly proper, third order controller. Its coefficients are randomly generated in the following way.

Algorithm 3 (Random Generation of the Controllers [12]) Step 1. Randomly decide if the zeros will be real or complex conjugate.

Step 2. Randomly decide if there will be a pair of complex conjugate poles.

Step 3. Generate the real zeros and poles by sampling uniformly over `(-realbound,0)`.

Step 4. Generate the real parts of the complex zeros and poles by sampling uniformly over `(-realbound,0)` and their imaginary part by sampling uniformly over `(-imagbound,imagbound)`.

Step 5. Generate the controller gain by sampling uniformly over (0,gainbound).

◇

Next, we reformulate our original problem. We assume that the plant uncertain parameters have uniform distribution in the intervals (13). We denote by $X \in \mathcal{X} \subset \mathbb{R}^3$ the vector of the random coefficients of the plant and by $Y \in \mathcal{Y} \subset \mathbb{R}^6$ the the vector of the random coefficients of the controller. Let us introduce a cost function

$$\Psi(Y) = \max\{\psi_1(Y), \psi_2(Y)\} \tag{15}$$

where

$$\psi_1(Y) = \begin{cases} 0 \text{ if } \left\| \begin{pmatrix} W_1(1 + G(s)C(s,Y))^{-1} \\ W_3C(1 + G(s)C(s,Y))^{-1} \end{pmatrix} \right\|_\infty < 2 \\ 1 \text{ otherwise} \end{cases} \tag{16}$$

and

$$\psi_2(Y) = E\left(\zeta(X, Y)\right)$$

with

$$\zeta(X,Y) = \begin{cases} 1 \text{ if } (1 + \hat{G}(s,X)C(s,Y))^{-1} \text{ is unstable} \\ \left\| \begin{pmatrix} W_1(1 + \hat{G}(s,X)C(s,Y))^{-1} \\ W_3C(1 + \hat{G}(s,X)C(s,Y))^{-1} \end{pmatrix} \right\|_\infty \text{ otherwise} \end{cases}$$

Now we are ready to apply the randomized algorithm 2 to find a probably approximate near minimum of $\Psi(Y)$ with confidence $1 - \delta$, level α and accuracy ε [36]. In our simulation, we chose $\varepsilon = 0.1$, $\delta = 0.05$

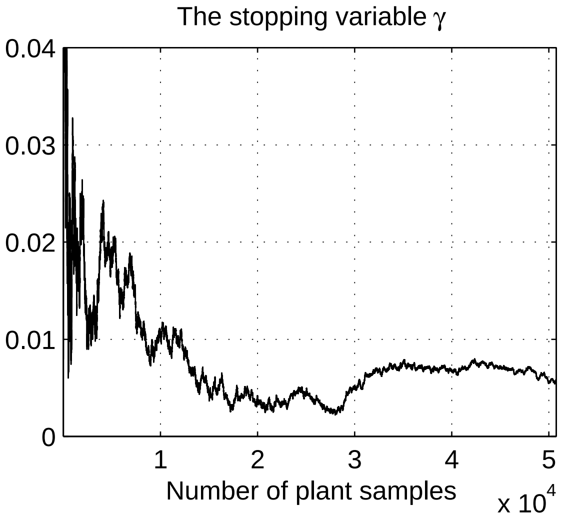

Fig. 4. The stopping variable γ for the design of the statistical controller of Example 1

and $\alpha = 0.001$. Therefore n evaluated to $3,688$ and m initially evaluated to $50,753$. For the generation of the controller we let `gainbound=realbound=imagbound=2`. Out of the n controllers, only $\hat{n} = 15$ gave $\psi_1 = 0$; the others were discarded. The stopping variable γ that we obtained for the first m plant samples is shown in Figure 4. Since the *final* value of γ is less than $\varepsilon/5 = 0.02$, one iteration was sufficient. The *best* controller is

$$C(s) = \frac{0.5781s^2 + 0.6602s + 0.1198}{0.2012s^3 + 0.9687s^2 + 11.7181s + 1}. \tag{17}$$

and the corresponding value of the cost function (15) is $\Psi = 0.600$.

Now let us go back to our original intent. We wanted to design a *good* third order controller attempting to minimize the norm in (14). How *good* our controller (17) is? To answer this question, in the following we will analyse the *nominal performance*, *robust stability* and *robust performance* of our closed-loop system. The tools we will use have become classical control tools; for a detailed discussion see e.g. [41] and the references therein.

To analyze the nominal performance, it sufficient to evaluate the mixed sensitivity norm (10). This norm evaluates to about 1.08, which is not far from the values gained by the $\mathcal{H}_\infty$ controller proposed in [40] and by the random controller proposed in [12].

The analysis of both robust stability and robust performance can be carried out using the μ-analysis. To this aim we need to write our uncertain system (11) using LFTs. First of all let us rewrite (12) as

$$\hat{G}(s) = K\frac{as^2 + bs + c}{5s(s+d)(s^2 + As + B)}$$

where

$$a = -6.4750, \quad b = 4.0302, \quad c = 175.77, \quad d = 6.66 \cdot 10^{-4}.$$

Let the uncertain parameters K, A and B have the expressions

$$K = \bar{K}(1 + \alpha_K\delta_K), \quad A = \bar{A}(1 + \alpha_A\delta_A), \quad B = \bar{B}(1 + \alpha_B\delta_B).$$

with

$$\begin{aligned} \bar{K} &= 1 & \alpha_K &= 0.5 \\ \bar{A} &= 0.7130 & \alpha_A &= 1 \\ \bar{B} &= 27.90 & \alpha_B &= 0.3. \end{aligned}$$

In this way we have normalized the uncertainties

$$-1 < \delta_K, \delta_A, \delta_B < 1$$

and the nominal plant (9) is recovered simply letting

$$\delta_K = 0, \quad \delta_A = 0, \quad \delta_B = 0.$$

Now, letting $\bar{G} = -a\bar{K}/5$, we can write our system isolating the uncertain terms as

$$\begin{aligned} \dot{x} &= \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -\bar{B}d \\ 0 & 1 & 0 & -\bar{B} - \bar{A}d \\ 0 & 0 & 1 & -\bar{A} - d \end{pmatrix} x + \begin{pmatrix} 0 & 0 & c & \bar{G}c \\ 0 & d & b & \bar{G}b \\ d & 1 & 1 & \bar{G} \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} u_A \\ u_B \\ u_K \\ u \end{pmatrix} \\ \begin{pmatrix} y_A \\ y_B \\ y_K \\ y \end{pmatrix} &= \begin{pmatrix} 0 & 0 & 0 & -\alpha_A \bar{A} \\ 0 & 0 & 0 & -\alpha_B \bar{B} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} x + \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \bar{G}\alpha_G \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} u_A \\ u_B \\ u_K \\ u \end{pmatrix} \end{aligned} \tag{18}$$

The input u_A, u_B and u_K and the outputs y_A, y_B and y_K are fictitious inputs and outputs used to model the uncertainties

$$\begin{pmatrix} u_A \\ u_B \\ u_K \end{pmatrix} = \begin{pmatrix} \delta_A & 0 & 0 \\ 0 & \delta_B & 0 \\ 0 & 0 & \delta_K \end{pmatrix} \begin{pmatrix} y_A \\ y_B \\ y_K \end{pmatrix} \qquad -1 < \delta_A, \delta_B, \delta_K < 1$$

Let now $u = -C(s)y$ in (18), with $C(s)$ given by (17), and denote by $G_c(s)$ the resulting closed-loop system

$$\begin{pmatrix} y_A \\ y_B \\ y_K \end{pmatrix} = G_c(s) \begin{pmatrix} u_A \\ u_B \\ u_K \end{pmatrix}$$

Consider the block structure

$$\mathbf{\Delta} := \left\{ \begin{pmatrix} \delta_1 & 0 & 0 \\ 0 & \delta_2 & 0 \\ 0 & 0 & \delta_3 \end{pmatrix}, \delta_i \in \mathbb{R} \right\}$$

We evaluated by means of the *μ-Analysis and Synthesis Toolbox* [42] the real structured singular value $\mu_{\mathbf{\Delta}}(G_c(s))$. Its plot is shown in Figure 5. Since its peak value is approximately equal to 0.97, our system (18) is guaranteed to be stable for all $\Delta \in \mathbf{\Delta}$ with $\|\Delta\|_2 < 1/0.97 \approx 1.03$. Anyway this result may be conservative, since the lower and upper μ bounds are significantly different[1]. For the sake of comparison, we evaluated $\mu_{\mathbf{\Delta}}(G_c(s))$ for the controller proposed in [40], which however had *not* been designed to be robust. The peak value of $\mu_{\mathbf{\Delta}}(G_c(s))$ is approximately equal to 1.04.

For the evaluation of the closed-loop performance let us refer to Figure 6 (see also Figure 3). We denote by $T(G_P, \Delta)$ the transfer matrix between d and $\begin{pmatrix} e & \bar{u} \end{pmatrix}^T$ (it is indeed the transfer matrix whose norm we minimized in (14)). Define an augmented block structure

$$\mathbf{\Delta}_P := \left\{ \begin{pmatrix} \Delta & 0 \\ 0 & \Delta_r \end{pmatrix} : \Delta \in \mathbf{\Delta}, \ \Delta_r \in \mathbb{C}^{1\times 2} \right\}$$

We evaluated the structured singular value $\mu_{\mathbf{\Delta}_P}(G_P(s))$. Its plot is shown in Figure 7. Since its peak value is approximately equal to 1.80, $\|T(G_P, \Delta)\|_\infty \leq 1.80$ for all $\Delta \in \mathbf{\Delta}$ with $\|\Delta\|_2 < 1/1.80 \approx 0.556$. We evaluated $\mu_{\mathbf{\Delta}_P}(G_c(s))$ also for the controller proposed in [40]. The peak value of $\mu_{\mathbf{\Delta}_P}(G_c(s))$ in this case is approximately equal to 1.60.

[1] It is well known that in the case of only real perturbations, the evaluation of the function μ is problematic, since it is not necessarely a continuous function [43]. The μ lower bound in these cases may not converge or may be significantely lower than μ itself [42]. In [42] it is shown how it is possible to overcome these problems by introducing fictitious small complex perturbations. Anyway this is beyond the scope of our analysis.

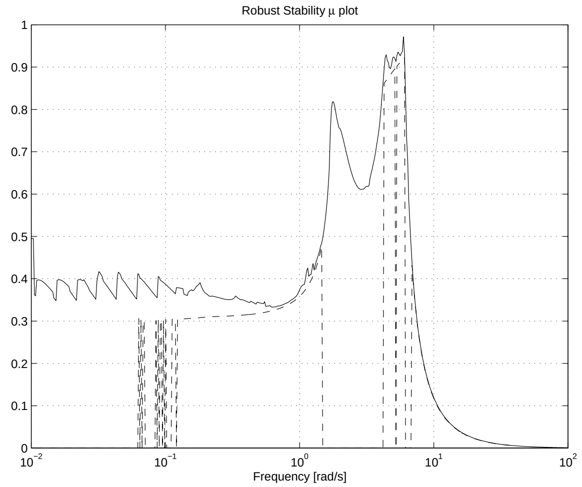

Fig. 5. The lower and upper bounds for $\mu_{\boldsymbol{\Delta}}(G_c(s))$

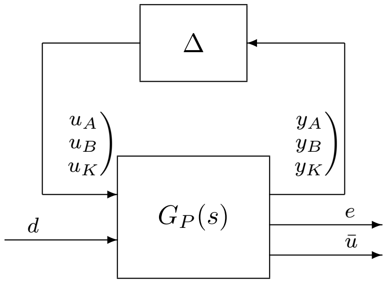

Fig. 6. The closed-loop for the evaluation of robust perfomance

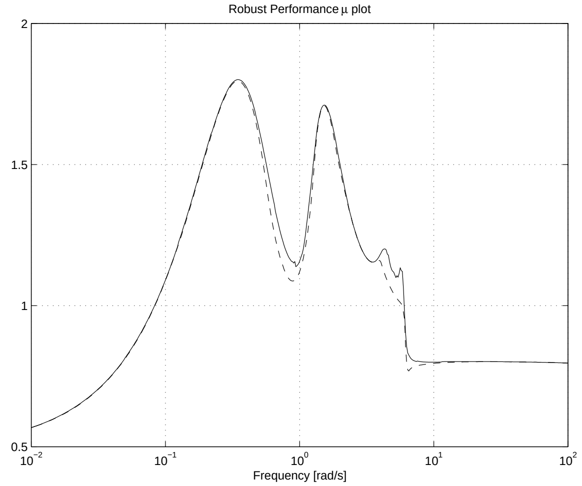

Fig. 7. The lower and upper bounds for $\mu_{\boldsymbol{\Delta_P}}(G(s))$

Remark 1: The values we chose for `gainbound`, `realbound` and `imagbound` in Algorithm 3 are arbitrary. Therefore we can repeat our design procedure choosing larger ranges for the controller parameters to see if we obtain a better controller, i.e. a controller which gives a smaller value for the cost function $\Psi(Y)$ (15). We tried `gainbound=realbound=imagbound=4` and making use of Algorithm 2 we obtained the controller

$$C(s) = \frac{0.4608s^2 + 0.7064s + 0.0803}{0.0848s^3 + 0.5769s^2 + 1.6465s + 1},$$

which gives a smaller value than before of $\Psi = 0.576$ for the cost function (15). With this controller we performed the same analysis as before. We obtained 1.01 for the mixed-sensitivity norm (10), a peak value of 0.96 for $\mu_{\mathbf{\Delta}}(G_c(s))$ and a peak value of 1.69 for $\mu_{\mathbf{\Delta}_P}(G_P(s))$, which improve the results of the controller (17).

△

This example shows how a combination of statistical methods and classical theoretical results can be a good way of approaching *difficult* control design problems.

VI. Conclusions

In this paper we have shown how drastic reduction in the number of samples needed to obtain performance guarantees in robust control synthesis problems are obtained. This reduction is achieved by introducing sequential bootstrapping algorithms and exploiting the fact that the sample complexity is itself a random variable. This has allowed us to present Algorithm 2 as an efficient design methodology for fixed-order robust control design problems [**?**]. Recall for example that the Static Output Feedback (SOF) was shown in [44] to be NP-hard when the gains of the feedback matrix were bounded, but that Algorithm 3, is well suited to address exactly the SOF problem under those conditions.

It should be noted that the methodology presented in this paper can be used in many other application areas: one only needs to have an efficient analysis tool in order to convert it to an efficient design methodology. This is due to the fact that the design problem is converted to a sequence of analysis or verification problems after sampling more plants and controllers than the minimum number required by Algorithm 2. It should also be noted that the computational complexity or the undecidability of the problems studied are not eliminated but only avoided by relaxing the design requirements from absolute (hard) to probabilistic (soft) ones.

The randomized algorithms approach may be applied to design fixed-structure controllers for nonlinear systems, and to delay-differential systems. Our future research is concentrating at the theoretical level in obtaining better optimization algorithms and at the application level in designing software modules for linear and nonlinear control design.

Acknowledgements: C.T. Abdallah acknowledges fruitful discussions with P. Dorato, G.L. Heileman, and P. Goldberg.

References

[1] V. Vapnik, *Estimation of Dependences Based on Empirical Data*, Springer-Verlag, New York, NY, 1982.

[2] L.R. Ray and R.F. Stengel, “A monte carlo approach to the analysis of control system robustness,” *Automatica*, vol. 29, pp. 229–236, 1993.

[3] E.W. Bai, R. Tempo, and M. Fu, “Worst case properties of the uniform distribution and randomized algorithms for robustness analysis,” in *Proc. IEEE American Control Conf.*, Albuquerque, NM, 1997, pp. 861–865.

[4] R. Tempo, E.W. Bai, and F. Dabbene, “Probabilistic robustness analysis: Explicit bounds for the minimum number of samples,” *Systems and Control Letters*, vol. 30, pp. 237–242, 1997.

[5] R. Tempo and F. Dabbene, “Probabilistic robustness analysis and design of uncertain systems,” Preprint, 1998.

[6] B.R. Barmish and C.M. Lagoa, “The uniform distribution: A rigorous justification for its use in robustness analysis,” in *Proc. IEEE Conf. on Dec. and Control*, Kobe, Japan, 1993, pp. 3418–3423.

[7] B.R. Barmish and C.M. Lagoa, “The uniform distribution: A rigorous justification for its use in robustness analysis,” *Mathematics of Control, Signals, and Systems*, vol. 10, pp. 203–222, 1997.

[8] B.R. Barmish, C.M. Lagoa, and R. Tempo, “Radially truncated uniform distributions for probabilistic robustness of control systems,” in *Proc. IEEE American Control Conf.*, Albuquerque, NM, 1997, pp. 853–857.

[9] X. Chen and K. Zhou, “On the probabilistic characterization of model uncertainty and robustness,” in *Proc. IEEE Conf. on Dec. and Control*, San Diego, CA, 1997, pp. 3816–3821.

[10] X. Chen and K. Zhou, “A probabilistic approach for robust control,” in *Proc. IEEE Conf. on Dec. and Control*, San Diego, CA, 1997, pp. 4894–4895.

[11] X. Chen and K. Zhou, "Constrained optimal synthesis and robustness analysis by randomized algorithms," in *Proc. IEEE American Control Conf.*, Philadelphia, PA, 1998, pp. 1429–1433.

[12] P.P. Khargonekar and A. Tikku, "Randomized algorithms for robust control analysis and synthesis have polynomial complexity," in *Proc. IEEE Conf. on Dec. and Control*, Kobe, Japan, 1993, pp. 3470–3475.

[13] M. Vidyasagar, *A Theory of Learning and Generalization with Applications to Neural Networks and Control Systems*, Springer-Verlag, Berlin, 1996.

[14] M. Vidyasagar, "Statistical learning theory and randomized algorithms for control," *IEEE Control Systems Magazine*, vol. 18, no. 6, pp. 69–85, 1998.

[15] V. Blondel, *Simultaneous Stabilization of Linear Systems*, Springer-Verlag, London, 1st edition, 1994.

[16] V.D. Blondel, E.D. Sontag, M. Vidyasagar, and J.C. Willems, Eds., *Simultaneous stabilization of linear systems and interpolation with rational functions*, Springer Verlag, London, 1999.

[17] A. Tarski, *A Decision Method for Elementary Algebra and Geometry*, Univ. of California Press, Berkeley, 1951, 2nd Ed.

[18] B.D.O. Anderson, N.K. Bose, and E.I. Jury, "Output feedback stabilization and related problems – solution via decision methods," *IEEE Trans. Automat. Control*, vol. 20, pp. 53–65, 1975.

[19] D. Neši'c and I.M.Y. Mareels, "Dead beat controllability of polynomial systems: Symbolic computation approaches," *IEEE Trans. Automat. Control*, vol. 43, pp. 162–175, 1998.

[20] M. Jirstrand, *Algebraic methods for modeling and design in control*, Ph.D. thesis, Linköping University - Sweden, 1996, Thesis Nr. 540.

[21] H. Anai, "On solving semidefinite programming by quantifier elimination," in *Proc. IEEE American Control Conf.*, Philadelphia, PA, 1998, pp. 2814–2818.

[22] G. E. Collins, "Quantifier elimination in the elementary theory of real closed fields by cylindrical algebraic decomposition," in *Lecture Notes in Computer Science*, Berlin, 1975, pp. 134–183, Springer-Verlag, Vol. 33.

[23] G. E. Collins and H. Hong, "Partial cylindrical algebraic decomposition for quantifier elimination," *J. Symb. Comp.*, vol. 12, no. 3, pp. 299–328, 1991.

[24] H. Hong, *Improvements in CAD-based Quantifier Elimination*, Ph.D. thesis, The Ohio State University, 1990.

[25] H. Hong, "Simple solution formula construction in cylindrical algebraic decomposition based quantifier elimination," in *ISSAC'92, International Symposium on Symbolic and Algebraic Computation, July 27-29, Berkeley, California*, P. S. Wang, Ed., New York, 1992, pp. 177–188, ACM Press.

[26] C.T. Abdallah, P. Dorato, W. Yang, R. Liska, and S. Steinberg, "Application of quantifier elimination theory to control system design," in *Proceedings of the 4th IEEE Mediterranean Symposium on Control & Automation*, Chania, Crete, Greece, 1996, pp. 41–44.

[27] P. Dorato, W. Yang, and C.T. Abdallah, "Robust Multi-Objective Feedback Design by Quantifier Elimination," *J. Symbolic Computation*, vol. 24, pp. 153–159, 1997.

[28] S. Basu, R. Pollack, and M.-F. Roy, ""On the combinatorial and algebraic complexity of quantifier elimination"," in *Proc. 35th Symposium on Foundations of Computer Science*, Santa Fe, NM, November 22–24 1994, pp. 632–641.

[29] V. Blondel and J. Tsitsiklis, "A survey of computational complexity results in systems and control," Preprint available at http://www.ulg.ac.be/mathsys/blondel/publications.html, 1998.

[30] G.E. Coxson, *Computational Complexity of Robust Stability and Regularity in Families of Linear Systems*, Ph.D. thesis, The University of Wisconsin–Madison, 1993.

[31] A. Nemirovskii, "Several NP-hard problems arising in robust stability analysis," *Mathematics of Control, Signals, and Systems*, vol. 6, pp. 99–105, 1993.

[32] S. Poljak and J. Rohn, "Checking robust nonsingularity is NP-hard," *Mathematics of Control, Signals, and Systems*, vol. 6, pp. 1–9, 1993.

[33] O. Toker and H. Özbay, ""On the NP-hardness of solving bilinear matrix inequalities and simultaneous stabilization with static output feedback"," *Proc. IEEE American Control Conf.*, pp. 2056–2064, June 1995.

[34] R.F. Stengel and L.R. Ray, "Stochastic robustness of linear time-invariant control systems," *IEEE Trans. Automat. Control*, vol. 36, pp. 82–87, 1991.

[35] X. Zhu, Y. Huang, and J. Doyle, "Soft vs hard bounds in probabilistic robustness analysis," in *Proc. IEEE Conf. on Dec. and Control*, Kobe, Japan, 1993, pp. 3412–3417.

[36] M. Vidyasagar, "Statistical learning theory and its applications to randomized algorithms for robust controller synthesis," in *European Control Conference (ECC97)*, Brussels, Belgium, 1997, vol. Plenary Lectures and Mini-Courses, pp. 162–190.

[37] L. Devroy, L. Györfi, and G. Lugosi, *A probabilistic theory of pattern recognition*, Springer-Verlag, Berlin, 1996.

[38] A. van der Vaart and J. Wellner, *Weak convergence and empirical processes*, Springer-Verlag, Berlin, 1996.

[39] V. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, New York, NY, 1998.

[40] J. C. Doyle, B. A. Francis, and A. R. Tannenbaum, *Feedback Control Theory*, Macmillan Publishing Company, New York, 1992.

[41] K. Zhou with J. C. Doyle, *Essentials of Robust Control*, Prentice Hall, Upper Saddle River, NJ, 1998.

[42] G. J. Balas, J. C. Doyle, K. Glover, A. Packard, and R. Smith, *μ-Analysis and Synthesis Toolbox*, The MathWorks, Inc., Natick, MA, 1998, User's Guide *Version 3*.

[43] B. Barmish, P. Khargonekar, Z. Shi, and R. Tempo, "Robustness Margin Need Not Be a Continuous Function of the Problem Data," *Systems and Control Letters*, vol. 15, pp. 91–98, 1989.

[44] V. Blondel and J.N. Tsitsiklis, "NP-hardness of some linear control design problems," *The European Journal of Control*, vol. 1, no. 1, pp. To appear, 1995.