Automatica 125 (2021) 109386

Contents lists available at ScienceDirect

Automatica

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

Statistical learning for analysis of networked control systems over unknown channels ☆

Konstantinos Gatsis [a,*], George J. Pappas [b]

[a] Department of Engineering Science, University of Oxford, Parks Road, Oxford, OX1 3PJ, UK
[b] Department of Electrical and Systems Engineering, University of Pennsylvania, 200 South 33rd Street, Philadelphia, PA 19104, USA

ARTICLE INFO

Article history:
Received 5 November 2019
Received in revised form 24 August 2020
Accepted 23 November 2020
Available online 18 December 2020

Keywords:
Learning algorithms
Statistical analysis
Networked control systems
Communication channels
Stability analysis

ABSTRACT

Recent control trends are increasingly relying on communication networks and wireless channels to close the loop for Internet-of-Things applications. Traditionally these approaches are model-based, i.e., assuming a network or channel model they are focused on stability analysis and appropriate controller designs. However the availability of such wireless channel modeling is fundamentally challenging in practice as channels are typically unknown a priori and only available through data samples. In this work we aim to develop algorithms that rely on channel sample data to determine the mean square stability and performance of networked control tasks. In this regard our work is the first to characterize the amount of channel modeling that is required to answer such a question. Specifically we examine how many channel data samples are required in order to answer with high confidence whether a given networked control system is stable or not. This analysis is based on the notion of sample complexity from the learning literature and is facilitated by concentration inequalities. Moreover we establish a direct relation between the sample complexity and the networked system stability margin, i.e., the underlying packet success rate of the channel and the spectral radius of the dynamics of the control system. This illustrates that it becomes impractical to verify stability under a large range of plant and channel configurations. We validate our theoretical results in numerical simulations.
© 2020 Elsevier Ltd. All rights reserved.

1. Introduction

Wireless communication is increasingly used in autonomous applications to connect devices in industrial control environments, teams of robotic vehicles, and the Internet-of-Things. To guarantee safety and control performance it is customary to include a model of the wireless channel, for example an i.i.d. or Markov link quality, alongside the model of the physical system to be controlled. In such model-based approaches one can characterize, for example, that it is impossible to estimate or stabilize an unstable plant if its growth rate is larger than the rate at which the link drops packets (Hespanha, Naghshtabrizi, & Xu, 2007; Schenato, Sinopoli, Franceschetti, Poolla, & Sastry, 2007), or below a certain channel capacity (Sahai & Mitter, 2006; Tatikonda & Mitter, 2004). Models also facilitate the allocation of communication resources to optimize control performance in, e.g., power allocation and scheduling over fading channels (Gatsis, Pajic, Ribeiro, & Pappas, 2015; Gatsis, Ribeiro, & Pappas, 2014; Quevedo, Ahlén, Leong, & Dey, 2012), or in event-triggered control (Araújo, Mazo, Anta, Tabuada, & Johansson, 2014; Mamduhi, Tolić, Molin, & Hirche, 2014; Mazo & Tabuada, 2011).

Even though physical system models are often available from first principles, wireless communication takes place over a channel subject to unpredictable variations and interferences. For example, this is the case when a given mechanical system with sensors and actuators is deployed in a new location in a factory. Channel conditions follow unknown distributions which are more often observable via a finite amount of collected channel sample measurements (Halperin, Hu, Sheth, & Wetherall, 2010; Rappaport, MacCartney, Samimi, & Sun, 2015). The purpose of this work is the analysis of networked control systems when only channel sample data are available instead of channel models. We use the data to learn whether a given networked control system is stable, and we also characterize how the learning procedure depends on the amount of channel samples and the control system parameters. This is in contrast to the extensive literature on model-based approaches mentioned above.

☆ This research is partially supported by National Science Foundation, USA CPS-1837253, by the Intel Science and Technology Center for Wireless Autonomous Systems (ISTC-WAS), USA and by ARL DCIST CRA W911NF-17-2-0181. The material in this paper was partially presented at the 57th IEEE Conference on Decision and Control, December 17–19, 2018, Miami Beach, Florida, USA. This paper was recommended for publication in revised form by Associate Editor Luca Schenato under the direction of Editor Christos G. Cassandras.
* Corresponding author.
E-mail addresses: konstantinos.gatsis@eng.ox.ac.uk (K. Gatsis), pappasg@seas.upenn.edu (G.J. Pappas).

https://doi.org/10.1016/j.automatica.2020.109386
0005-1098/© 2020 Elsevier Ltd. All rights reserved.

Learning methods have been most commonly used to control an unknown dynamical system from data within the reinforcement learning and approximate dynamic programming literature (Bertsekas, 2012). One specific approach within this framework is based on learning the system dynamics model first (Kumar & Varaiya, 2015). This is used for example in analyzing the sample complexity of the quadratic control of unknown linear systems (Abbasi-Yadkori & Szepesvári, 2011; Dean, Mania, Matni, Recht, & Tu, 2019). In contrast, our work is focused on collecting channel data and learning unknown channel models instead of system dynamics. In the context of networked control systems, recent works from the last two years are proposing data-based approaches including deep learning for allocating resources (Demirel, Ramaswamy, Quevedo, & Karl, 2018; Eisen, Gatsis, Pappas, & Ribeiro, 2019; Leong, Ramaswamy, Quevedo, Karl, & Shi, 2020; Wu, Ren, Jia, Johansson, & Shi, 2019), for controller design (Baumann, Zhu, Martiu, & Trimpe, 2018; Schuurmans, Sopasakis, & Patrinos, 2019; Yoo & Johansson, 2019), or to identify from data when given system models are inaccurate (Baumann, Solowjow, Johansson, & Trimpe, 2019; Solowjow, Baumann, Garcke, & Trimpe, 2018). To the best of our knowledge, our work is the first to examine the sample complexity of networked control systems analysis and the amount of channel data needed when channel model is not available while system dynamics are known.

Specifically, we consider the stability of a linear dynamical system over a Bernoulli packet-dropping channel with an unknown success rate (Section 2). Using channel sample data, i.e., a number of packet successes and failures, we develop an algorithm to learn whether the networked control system is stable or not (Section 3). To do this, we utilize confidence intervals obtained by concentration inequalities such as Hoeffding's inequality. As our algorithm depends on random channel samples there is always a probability of error, i.e., the algorithm returns that the system is stable while the true system is not. Our algorithm is guaranteed by design to have low probability of such errors *irrespective of the number of data points* provided. On the other hand, we characterize the statistical correctness of the algorithm (Theorem 1), i.e., the probability of correctly learning the stability of the networked control system. We further analyze the amount of channel sample data needed to correctly learn the system stability with a desired confidence level (Corollary 2).

Our most significant finding is that the sample complexity adversely depends on the system stability margin, i.e., the underlying packet success rate of the channel and the spectral radius of the dynamics of the control system. In other words, a significantly larger number of samples are needed to learn whether the system is stable if the networked control system over the channel is closer to instability. This means that it becomes impractical to verify stability under a large range of plant and channel configurations, but the derived sample complexity can be useful as follows: It describes the amount of channel samples required if we are willing to verify stability with high confidence up to a certain system stability margin. This paper extends preliminary results reported in Gatsis and Pappas (2018). Specifically we include a tighter sample complexity analysis that can be performed for very reliable or very unreliable links (Section 3.2). This is important because it shows that it is possible in such cases to rely on fewer channel samples. Apart from stability analysis, we also analyze from data whether the quadratic cost of the system is below some test value (Section 4). We validate our theoretical analysis and provide comparison with practical methods of constructing confidence intervals in numerical simulations (Section 5).

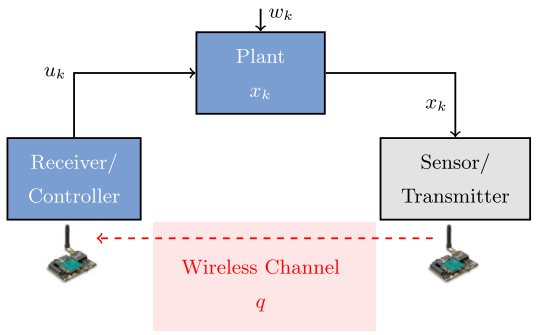


Fig. 1. Wireless Control System. A sensor measures the state of a plant perturbed by a random disturbance. The sensor transmits the measured information over a packet-dropping wireless channel to a receiver/controller providing control inputs.

2. Model-based networked control

We consider the evolution of a dynamical system over a packet dropping channel. This is a standard model for remote estimation or control over a network or a wireless channel, for example when a sensor measures the state of the plant and transmits it to a receiver — see Fig. 1 and Hespanha et al. (2007), Schenato et al. (2007) for related examples. Our goal is to analyze the stability properties and the control performance of the system, hence we assume the dynamics for the system are fixed, for example a controller has been already designed. We assume that the evolution of the system depends on whether a transmission occurs at time k or not, indicated with variables $\gamma_k \in \{0, 1\}$. In general we may consider dynamics described by a switched linear time invariant model of the form

$$x_{k+1} = \begin{cases} A_o\, x_k + w_k, & \text{if } \gamma_k = 0 \\ A_c\, x_k + w_k, & \text{if } \gamma_k = 1 \end{cases}. \tag{1}$$

Here $x_k \in \mathbb{R}^n$ denotes the state of the overall control system at each time k. At a successful transmission the system dynamics are captured by the matrix $A_c \in \mathbb{R}^{n\times n}$, where 'c' stands for closed-loop, and otherwise when the transmission fails the dynamics are in open loop described by $A_o \in \mathbb{R}^{n\times n}$, where 'o' stands for open-loop. We assume that A_c is asymptotically stable, implying that if system successfully transmits at each time step the state evolution $x_k, k \geq 0$ is stable. The open loop matrix A_o may be unstable. The additive terms $w_k, k \geq 0$ model an independent identically distributed (i.i.d.) noise process across time according to some known probability distribution with mean zero and positive definite covariance W.

In this paper we make the assumption that γ_k are independent Bernoulli random variables with a constant success probability q, and they are also independent from the system state x_k and noise w_k. This i.i.d. assumption is crucial for our results as we discuss in Section 6. We are also interested in the performance of the dynamical system. We will employ the usual quadratic system state cost at each time step k as $x_k^T\, Q\, x_k$ where Q is a positive definite matrix. Given the model of the transmission success, we can describe the effect on the performance of the control system with the average quadratic cost

$$J(q) = \limsup_{N\to\infty} \frac{1}{N} \sum_{k=0}^{N-1} \mathbb{E}[x_k^T\, Q\, x_k]. \tag{2}$$

The expectation at the right hand side accounts for the randomness introduced by the system disturbance and the channel. We choose to denote this cost as a function of the success rate of the channel q.

When the channel success rate q is known, we may analyze fundamentally whether the system is stable over this channel as well as characterize its performance. However in practice the channel success rate q is *unknown*. Instead we may have access to *channel sample data*. The problem we would like to answer is twofold:

- Data-driven Stability Analysis: how to check whether the system is stable using the channel sample data?
- Sample Complexity: what is the confidence of the method and how does it scale with the amount of channel samples and the control system and channel parameters?

2.1. Special case of system dynamics

To simplify the exposition of the main results we will examine these questions for the following special case of the system dynamics, which are further motivated by a canonical example in Remark 1.

$$x_{k+1} = \begin{cases} Ax_k + w_k, & \text{if } \gamma_k = 0 \\ w_k, & \text{if } \gamma_k = 1 \end{cases}. \tag{3}$$

At a successful transmission the system dynamics are reset to zero, and otherwise when the transmission fails the dynamics are in open loop described by $A \in \mathbb{R}^{n\times n}$. The open loop matrix A may be unstable, i.e., the eigenvalue with the largest modulus may be larger than unity, $\rho(A) = \max_{i=1,\dots,n} |\lambda_i(A)| > 1$. We have then the following result.

Proposition 1. *Consider the switched linear system (3) over an i.i.d. Bernoulli binary channel with a known success probability $q \in [0, 1]$. Then:*

(1) The system is mean square stable, i.e., $\sup_k \mathbb{E}x_k x_k^T < \infty$ if and only if

$$q > 1 - \frac{1}{\rho(A)^2}. \tag{4}$$

(2) Moreover, if condition (4) holds, the quadratic cost (2) is a strictly decreasing function of the success rate q given by

$$J(q) = Tr(PW), \tag{5}$$

where P is the unique positive definite solution of the (Lyapunov) matrix equation

$$P = Q + (1-q)A^T PA. \tag{6}$$

Proof. The stability condition (4) as well as the expression for the control cost in (5)–(6) follows from the random jump linear system theory (Costa, Fragoso, & Marques, 2006). The fact that the control cost (2) is a decreasing function can be shown as follows: It is straightforward to show that the solution to the Lyapunov equation (6) can be written as

$$P = \sum_{i=0}^{\infty} (1-q)^i (A^i)^T Q A^i, \tag{7}$$

where the sum converges due to (4). Plugging in this expression in (5) yields the expression for the control cost as

$$J(q) = \sum_{i=0}^{\infty} (1-q)^i \operatorname{Tr}((A^i)^T Q A^i W). \tag{8}$$

Taking the derivative with respect to q verifies that the function is non-increasing because all terms $\operatorname{Tr}((A^i)^T Q A^i W)$ are non-negative and is strictly decreasing because $\operatorname{Tr}(QW) = \operatorname{Tr}(W^{1/2} Q W^{1/2}) > 0$ as both Q, W are positive definite matrices. ■

The above result characterizes the absolute minimum channel success rate required for stability as a function of the eigenvalues of the dynamics A and gives an expression for the control performance as a function of the system and channel parameters. In the following section we introduce our approach for analyzing the stability of (3) when only channel samples are available instead of the channel model. We return to consider the general dynamics (1) in Section 3.3, where the same approach follows, only subject to the fact that the stability condition is more complex than (4).

Remark 1. The stability condition (4) of Proposition 1 is motivated from a more general and canonical networked control architecture. In particular suppose we are interested in controlling a linear plant in Fig. 1 with dynamics

$$x_{k+1} = Ax_k + Bu_k + w_k, \tag{9}$$

and we employ a controller of the form $u_k = K\hat{x}_k$ where K is a standard LQR controller gain and $\hat{x}_k$ is an estimator built at the controller side based on information sent from the sensor at the other side. Suppose the sensor transmits state measurements x_k over the unreliable link and the estimator evolves as

$$\hat{x}_k = \begin{cases} A\hat{x}_{k-1}, & \text{if } \gamma_k = 0 \\ x_k, & \text{if } \gamma_k = 1 \end{cases}. \tag{10}$$

Then for Bernoulli packet dropping channels it is known that condition (4) is again necessary and sufficient for stability of the system (Hespanha et al., 2007). This canonical example motivates the study of the simplified dynamics in (3), which are similar to the dynamics of the state estimation in this example.

3. Sample-based stability analysis of networked control

Suppose that instead of knowing the packet success rate q of the channel, we have available N samples denoted by $\gamma_k, k = 0, \dots, N-1$ drawn independently from the Bernoulli distribution with success q. In practice this data is easy to obtain, it suffices to send N packets and record whether they are received or not. Given this data the most natural approximation of the true success probability q is the sample average

$$\hat{q}_N = \frac{1}{N} \sum_{k=0}^{N-1} \gamma_k. \tag{11}$$

Indeed this approximation is in some sense optimal.[1] In the case of unlimited data samples, the sample average converges almost surely to the true underlying packet success rate by the Strong Law of Large Numbers (Durrett, 2010, Ch.2).

But in practice only finite amount of data will be available and this motivates us to investigate a finite sample analysis. Our approach is based on confidence intervals constructed by the channel sample data using concentration inequalities. In particular in this section we employ Hoeffding's inequality and in the following sections we expand on other constructions. Given a desired high confidence level $1-\delta$ where δ is a small positive value, for example of the order of 10^{-3}, and after collecting N samples, we may derive an interval where the true underlying channel success rate lies, as follows:

[1] It is easy to show that the sample average is the maximum likelihood estimate of the underlying channel success rate q. The likelihood of some success rate $q \in [0, 1]$ give the data $\gamma_k, k = 0, \dots, N-1$ is given by the probability that the data would be obtained under this success rate q. That is given by $\mathbb{P}(\gamma_k, k = 0, \dots, N-1 \,|\, q) = q^{\sum_k \gamma_k} (1-q)^{(N-\sum_k \gamma_k)}$. Maximizing the likelihood (or its logarithm) readily gives the sample mean as the most likely packet success.

Algorithm 1 Stability analysis using channel samples

Input: Dynamics A, Confidence level δ, Number of samples N, Channel samples $\gamma_0, \dots, \gamma_{N-1} \in \{0,1\}^N$

Compute the sample average

$$\hat{q}_N = \frac{1}{N}\sum_{k=0}^{N-1} \gamma_k \tag{13}$$

Compute the high confidence lower and upper bounds

$$q_{\min} = \max\{0, \hat{q}_N - \sqrt{\frac{\log(1/\delta)}{2N}}\} \tag{14}$$

$$q_{\max} = \min\{1, \hat{q}_N + \sqrt{\frac{\log(1/\delta)}{2N}}\} \tag{15}$$

if $1 - \frac{1}{\rho(A)^2} < q_{\min}$ **then**
  return 'Stable'
else
  if $1 - \frac{1}{\rho(A)^2} > q_{\max}$ **then**
    return 'Unstable'
  else
    return 'Undetermined'
  end if
end if

Lemma 1. *Consider a sequence $\{\gamma_k, k = 0, \dots, N-1\}$ of i.i.d. random variables taking values in $\{0,1\}$ with mean q. Let $\hat{q}_N = \frac{1}{N}\sum_{k=0}^{N-1}\gamma_k$ be the sample average. Then for any $\delta \in (0,1)$ it holds that*

$$\mathbb{P}\left(q \geq \hat{q}_N - \sqrt{\frac{\log(1/\delta)}{2N}}\right) \geq 1 - \delta, \tag{12}$$

where the probability is with respect to the random sequence $\{\gamma_k, k = 0, \dots, N-1\}$.

The result follows directly from Hoeffding's inequality (Boucheron, Lugosi, & Bousquet, 2003, Th. 1) and the fact that variables γ_k are bounded by 0 and 1. In this lemma the quantity $\hat{q}_N - \sqrt{\frac{\log(1/\delta)}{2N}}$ is a sample-based high-confidence lower bound on the true packet success rate of the channel. An upper bound can also symmetrically be defined. The result essentially states that there is a low probability that the sample average deviates much from the true packet success rate, and it reveals a closed form dependency between the number of samples N, the deviation amount, and the confidence level. For example, doubling the amount of collected channel samples from N to $2N$ only slightly tightens the lower bound (by a factor of $1/\sqrt{2}$).

3.1. Stability analysis using channel samples

Given some channel data, we would like to verify whether the system is stable, that is, whether the inequality (4) holds. We can construct an interval where the channel success rate lies with a desired high confidence using Lemma 1 and it suffices to check whether stability holds for the lower end of this interval. A symmetric argument can verify instability of the system. This data-based procedure is summarized in Algorithm 1.

First we note that it is possible that the algorithm returns a wrong answer, i.e., return 'Stable' in cases where the system is unstable and vice versa. Intuitively this can occur if the sample mean of the data is sufficiently different from the true mean. However *by design* we can guarantee that wrong answers are very unlikely (happen with probability less than δ) and also *irrespective of the number of data points N* provided. On the other hand the algorithm may not be able to correctly determine stability or instability for small amount of data, for example, return 'Undetermined' in cases where the system is actually stable. The remedy is to draw more samples and check stability again.

Our main result is to analyze the average performance of Algorithm 1 using a finite number of samples. By average performance, we mean how often would the algorithm return the correct or the wrong answer if we were to run it multiple times over independent data samples. Let A_N and W_N denote the events that the algorithm provides a correct answer and wrong answer respectively. That is, if the system is stable, i.e., condition $q > 1 - \frac{1}{\rho(A)^2}$ holds, then define the event

$$A_N = \left\{\{\gamma_0, \dots, \gamma_{N-1}\} \in \{0,1\}^N : \text{ Alg. 1 returns 'stable'}\right\} \tag{16}$$

Similarly define the event

$$W_N = \left\{\{\gamma_0, \dots, \gamma_{N-1}\} \in \{0,1\}^N : \text{Alg. 1 returns 'unstable'}\right\} \tag{17}$$

The complement of A_N and W_N is the event that the algorithm returns the answer 'Undetermined'. Alternatively if the system is strictly unstable, i.e., condition $q < 1 - \frac{1}{\rho(A)^2}$ holds, the events can be similarly defined. Due to symmetry, we analyze these two cases together. We can state then the following main result.

Theorem 1 (*Sample-based Stability Analysis*)**.** *Consider the switched linear system (3) over an i.i.d. Bernoulli binary channel with an unknown success probability $q \in [0,1]$ and assume $q \neq 1 - \frac{1}{\rho(A)^2}$. Consider the stability analysis procedure developed in Algorithm 1 using N i.i.d. channel samples drawn with success rate q. Let A_N denote the event that the algorithm provides the correct answer as defined in (16), and let W_N denote the event that the algorithm provides the wrong answer as defined in (17). Then for any $N \geq 1$ it holds that*[2]

$$\mathbb{P}(A_N) \geq 1 - \exp\left\{-2N\left[\left|q - 1 + \frac{1}{\rho(A)^2}\right| - \sqrt{\frac{\log(1/\delta)}{2N}}\right]_+^2\right\} \tag{18}$$

and

$$\mathbb{P}(W_N) \leq \delta \tag{19}$$

where the probability is with respect to the random channel samples.

Proof. Suppose the system is stable, i.e., according to Proposition 1 the packet success probability satisfies

$$q > 1 - \frac{1}{\rho(A)^2}. \tag{20}$$

The event A_N that Algorithm 1 returns the correct result is defined by (16), and corresponds to the case

$$A_N = \{\{\gamma_0, \dots, \gamma_{N-1}\} \in \{0,1\}^N \text{ s.t.} \tag{21}$$

$$\hat{q}_N - \sqrt{\frac{\log(1/\delta)}{2N}} > 1 - \frac{1}{\rho(A)^2}\}. \tag{22}$$

As a result we have that

$$\mathbb{P}(A_N) = 1 - \mathbb{P}\left[\hat{q}_N - \sqrt{\frac{\log(1/\delta)}{2N}} \leq 1 - \frac{1}{\rho(A)^2}\right]. \tag{23}$$

Adding and subtracting q at the right hand side we have that

$$\mathbb{P}(A_N) = 1 - \mathbb{P}\left[\hat{q}_N \leq q - \left(q - 1 + \frac{1}{\rho(A)^2} - \sqrt{\frac{\log(1/\delta)}{2N}}\right)\right] \tag{24}$$

[2] Here $[\ ]_+$ denotes the projection to the positives.

The term in the parenthesis can be in general both negative or positive, hence we consider two cases.

Case I: $q-1+\frac{1}{\rho(A)^2}-\sqrt{\frac{\log(1/\delta)}{2N}}>0$. This is the case where the term in the parenthesis in (24) is positive and we can directly apply Hoeffding's inequality to get the desired bound (18).

Case II: $q-1+\frac{1}{\rho(A)^2}-\sqrt{\frac{\log(1/\delta)}{2N}}\leq 0$. In this case the bound in (18) becomes

$$\mathbb{P}(A_N)\geq 1-\exp\{-2N\,0\}=0, \tag{25}$$

which trivially holds.

Using similar arguments we can show that the algorithm provides the wrong answer with probability bounded by

$$\mathbb{P}(W_N)\leq \exp\left\{-2N\left[\left|q-1+\frac{1}{\rho(A)^2}\right|+\sqrt{\frac{\log(1/\delta)}{2N}}\right]^2\right\} \tag{26}$$

In particular, since the term in the absolute value is non-negative, the bound (19) can be obtained by bounding this absolute value below by zero.

A symmetric argument verifies the bounds (18) and (19) when the system is strictly unstable, i.e., when the packet success probability satisfies $q<1-\frac{1}{\rho(A)^2}$. ■

Some remarks are in order. First, the probability that the algorithm returns the correct answer grows to one as the number of samples grows to infinity. This is expected from the Law of Large Numbers. But for finite number of samples, there will be a probability the algorithm may not be confident enough to correctly identify stability. As already mentioned, by design the algorithm is very unlikely to provide a wrong answer as verified by (19). The above result is important as it allows us to characterize the finite sample complexity of the algorithm, i.e., answer how many channel sample data we need in order to correctly determine stability with a desired high confidence given the system parameters.

Corollary 2 (*Sample Complexity*). *Consider the switched linear system (3) over an i.i.d. Bernoulli binary channel with an unknown success probability* $q\in[0,1]$ *and assume* $q\neq 1-\frac{1}{\rho(A)^2}$. *Consider the stability analysis procedure developed in Algorithm 1 using N i.i.d. channel samples drawn with success rate q and some parameter* $\delta\in(0,1)$. *If the number of samples is*

$$N\geq \frac{2\log(1/\delta)}{(q-1+\frac{1}{\rho(A)^2})^2}, \tag{27}$$

then the procedure correctly determines the stability or instability of the system with probability $(1-\delta)$.

Proof. Let the number of samples satisfy (27). Then it holds that

$$\left|q-1+\frac{1}{\rho(A)^2}\right|\geq\sqrt{\frac{2\log(1/\delta)}{N}}=2\sqrt{\frac{\log(1/\delta)}{2N}}, \tag{28}$$

which is equivalent to

$$\left[\left|q-1+\frac{1}{\rho(A)^2}\right|-\sqrt{\frac{\log(1/\delta)}{2N}}\right]_+\geq\sqrt{\frac{\log(1/\delta)}{2N}}, \tag{29}$$

or

$$-2N\left[\left|q-1+\frac{1}{\rho(A)^2}\right|-\sqrt{\frac{\log(1/\delta)}{2N}}\right]_+^2\leq-\log(1/\delta). \tag{30}$$

Hence by Theorem 1, Algorithm 1 returns the correct answer with probability at least $1-\delta$. ■

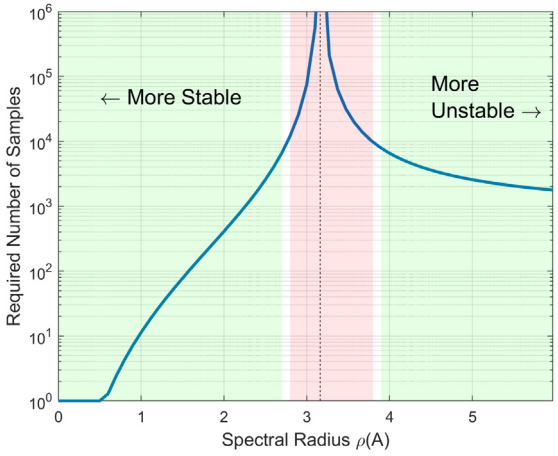


Fig. 2. Sample complexity according to Corollary 2. For a channel with fixed success rate $q=0.9$ and a desired confidence $1-\delta=0.99$ we plot the theoretically required number of channel samples N as a function of the system spectral radius. A small number of samples are needed when the networked system becomes more stable or more unstable, but it grows unbounded when the stability radius reaches the critical point of instability, i.e., when $q=1-\frac{1}{\rho(A)^2}$, denoted by the dotted line.

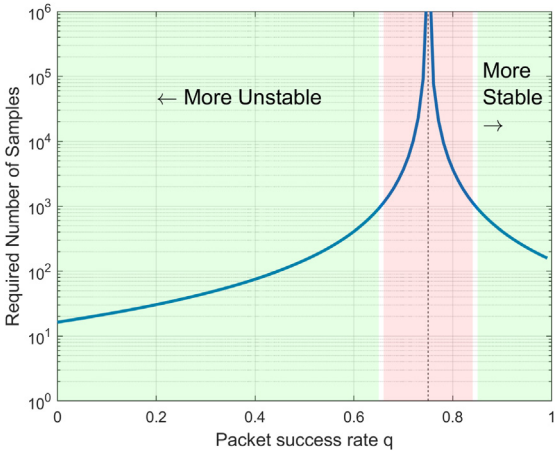


Fig. 3. Sample complexity according to Corollary 2. For a system with fixed spectral radius $\rho(A)=2$ and a desired confidence $1-\delta=0.99$ we plot the theoretically required number of channel samples N as a function of the packet success rate q of the channel. The required number of samples grows unbounded as the packet success reaches the critical point of instability, i.e., when $q=1-\frac{1}{\rho(A)^2}$, denoted by the dotted line.

The number of samples required depends on the true channel success rate q, which is unknown, but provides intuition. We observe from this result that the sample complexity scales well with the desired confidence level. An order of magnitude improvement in confidence can be guaranteed with just doubling the amount of data samples. On the other hand, the sample complexity does not scale well with the system stability margin. If the stability margin $|q-1+\frac{1}{\rho(A)^2}|$ is reduced by a factor of β, then β^2 more channel samples will be required.

A plot for this sample relation (27) is given in Fig. 2 as a function of the system spectral radius $\rho(A)$ and for a fixed channel success rate q. The required number of samples grows unbounded as the spectral radius reaches the critical point of instability, i.e., when $q=1-\frac{1}{\rho(A)^2}$. Otherwise, it becomes easier to verify stability with fewer samples if the system dynamics are very fast or very slow. Similarly in Fig. 3 we plot the theoretically required number of samples (27) for a fixed system $\rho(A)$ as a

function of the underlying packet success rate q of the channel, and we observe similar behavior. These observations mean that it becomes impractical to verify stability under all plant and channel configurations, but the above sample complexity can be useful. It describes the amount of channel samples required if we are willing to verify stability with high confidence up to a certain system stability margin.

Remark 2. The bound in (27) can be thought as asymptotically tight. For very large values N, by the Central Limit Theorem, the variables $q_{\min}$ and $q_{\max}$ are both distributed as normal distributions centered at the true mean q with a variance $\sigma^2 = q(1-q)\sqrt{N}$. The stability test that we conduct checks whether $\hat{q}_{\max} < 1 - 1/\rho(A)^2$ or $\hat{q}_{\min} > 1 - 1/\rho(A)^2$. Hence it gives the correct answer with probability $\mathbb{P}(A_N) \approx 1 - \mathbb{P}(\mathcal{N}_{0,\sigma} \geq |q - 1 + \frac{1}{\rho(A)^2}|) \approx 1 - \exp(-|q - 1 + \frac{1}{\rho(A)^2}|^2/(q(1-q)\sqrt{N})^2)$. From this we conclude that the probability of correct answer increases to unity exponentially with a rate $|q - 1 + \frac{1}{\rho(A)^2}|^2/N$.

3.2. Sharper sample complexity results for low-variance channels

The previous analysis uses Hoeffding's inequality to obtain a sample complexity result for any potential channel success rate $q \in [0, 1]$ we may encounter. In this section we will show that using sharper concentration inequality bounds, we can obtain more advantageous results in the case of very reliable or very unreliable links ($q \approx 0$ or $q \approx 1$). In particular, in these cases we will show that fewer samples are needed to obtain high performance in the data-driven stability problem. This may be practically relevant in cases where one a priori knows that we are dealing with very reliable link.

Specifically we can use Bernstein's inequality (Boucheron et al., 2003), which involves the variance of the random samples (equal to $q(1-q)$ for our Bernoulli random success samples). To achieve the more advantageous results we propose to employ estimation intervals for the true packet success rate q that *decay faster* than before. Specifically consider Algorithm 1 and change to

$$q_{\min} = \hat{q}_N - \frac{\log(1/\delta)}{N}, \tag{31}$$

$$q_{\max} = \hat{q}_N + \frac{\log(1/\delta)}{N}, \tag{32}$$

which decay as $1/N$ versus the $1/\sqrt{N}$ that we considered before. Then we can establish the following result.

Theorem 3 (*Sample Complexity for Low-Variance Channels*)**.** *Consider the switched linear system (3) over an i.i.d. Bernoulli binary channel with an unknown success probability $q \in [0, 1]$ and assume $q \neq 1 - \frac{1}{\rho(A)^2}$. Consider the stability analysis procedure developed in Algorithm 1 with the substitution (31) and (32) using N i.i.d. channel samples drawn with success rate q and some parameter $\delta \in (0, 1)$. If there exists a constant $c > 0$ such that*

$$q(1-q) \leq c \left| q - 1 + \frac{1}{\rho(A)^2} \right|, \tag{33}$$

then the procedure correctly determines the stability or instability of the system with probability $(1-\delta)$ if the number of samples is larger than

$$N \geq c' \frac{\log(1/\delta)}{\left| q - 1 + \frac{1}{\rho(A)^2} \right|}, \tag{34}$$

for some constant $c' = 2(4 + 3c)/3 > 0$.

Proof. Using the same arguments as in the proof of Theorem 1, the probability that this new algorithm returns a correct answer equals

$$\mathbb{P}(A_N) = 1 - \mathbb{P}\left[\hat{q}_N \leq q - \left[\left| q - 1 + \frac{1}{\rho(A)^2} \right| - \frac{\log(1/\delta)}{N} \right]_+ \right]. \tag{35}$$

This can be bounded using Bernstein's inequality (Boucheron et al., 2003, p. 199), which involves the variance of the random samples (equal to $q(1-q)$ for our Bernoulli random success samples) and the fact that they are bounded (by 1 in our case), to establish that

$$\mathbb{P}(\hat{q}_N \leq q - \varepsilon) \leq \exp\left(-\frac{N\varepsilon^2/2}{q(1-q) + \varepsilon/3} \right). \tag{36}$$

Applying this in (35), we have that the probability of correct answer is larger than $1 - \delta$ if

$$\frac{N\left(\left| q - 1 + \frac{1}{\rho(A)^2} \right| - \frac{\log(1/\delta)}{N} \right)^2/2}{q(1-q) + \left(\left| q - 1 + \frac{1}{\rho(A)^2} \right| - \frac{\log(1/\delta)}{N} \right)/3} \geq \log(1/\delta). \tag{37}$$

This is equivalent to a second degree polynomial with respect to N being larger than zero, and after a root analysis we can find that (37) is satisfied if

$$N \geq \frac{2\log(1/\delta)\left(4\left| q - 1 + \frac{1}{\rho(A)^2} \right| + 3q(1-q) \right)}{3\left(q - 1 + \frac{1}{\rho(A)^2} \right)^2}. \tag{38}$$

Using condition (33) in this inequality leads to (34). ■

Condition (33) essentially means the channel success rate is as close to 0 or 1 as to the critical rate for system stability. From (34) we see again that for channel rates close to the critical ones for stability we need a lot of samples. However comparing this with Corollary 2 we observe an *order of magnitude improvement* in this scaling law. In other words, we need an order of magnitude fewer samples to verify very reliable channels.

In the opposite regime where $q(1-q)$ is much larger than $\left| q - 1 + \frac{1}{\rho(A)^2} \right|$, then the sample complexity of the method in this section has the same rate as the method based on Hoeffidng's inequality. In other words in that regime we do not gain anything from reducing the confidence intervals faster.

As a side remark, by reducing the width of the confidence intervals as we do in this section, we are increasing the probability of wrong answer as defined in the previous section (event W_N). Technically, if we follow the steps in the proof of Theorem 1 using the confidence intervals in this section, then the probability of event W_N is not guaranteed to be less than δ for all $N \geq 1$. This is a shortcoming of the approach in this section and may not be desirable for safety reasons.

3.3. Analysis for general linear systems

Let us now revisit the general system dynamics (1) and examine how the approach generalizes. Under an i.i.d. channel γ_k with success rate q it is well known that the system (1) is mean square stable, i.e., $\limsup_{k\to\infty} \mathbb{E} x_k x_k^T < \infty$, if and only if the following condition holds

$$\rho\Big(q\, A_c \otimes A_c + (1-q) A_o \otimes A_o \Big) < 1, \tag{39}$$

where $\rho(.)$ denotes the spectral radius and $\otimes$ the Kronecker product. The result follows from the random jump linear system theory (Costa et al., 2006), and Proposition 1 is a special case. The method presented in Algorithm 1 can be extended to this more general case. For a fixed number of samples we can construct

an interval where the true packet success rate q lies with a desired high confidence level. This can be constructed in the same way as in the previous sections. Then we can check whether the above stability condition (39) holds for all values q in the interval. If this holds, we can declare stability. Symmetrically, if for all values of the packet success q within the high-confidence interval the above condition (39) does not hold, then we can declare instability. As before, as the number of samples increases the probability that this procedure provides the correct answer increases to one. The caveat is that the above stability condition (39) is in general non-convex in the variable $q \in [0, 1]$, it may be a union of disjoint intervals, so checking whether it holds for all values q in an interval may be computationally demanding. Only in special cases the procedure may be simplified, such as the case $A_c = 0$ we have examined before.

3.4. Practical algorithms for confidence intervals

Methods for constructing confidence intervals have some inherent tradeoffs. We would like to have intervals that are shorter (less conservative) and in our paper this would make checking stability easier, i.e., getting the correct answer more often and with fewer samples. But if the intervals are too short, there is a higher probability that they do not contain the true value and can result in wrong answers, especially for small sample sizes. In the previous sections we devised confidence intervals from concentration inequalities with the purpose of characterizing theoretically the sample complexity of the problem. However these general concentration inequalities are distribution free and hence can be conservative.

In this section we discuss more practical approaches. We can exploit the known distribution of the channel sample data, in particular the fact that γ_k are i.i.d. Bernoulli random variables. We see then that $\sum_{k=0}^{N-1} \gamma_k$ has a binomial distribution with parameters N and q. Hence we can construct confidence intervals estimating the binomial proportion q (Agresti & Coull, 1998). We detail a few methods next.

Exact method. An exact method, also known as Clopper–Pearson, aims to invert the binomial distribution in order to obtain a confidence interval. This is for example the method used in Matlab functions "binofit", "paramci" to find binomial confidence intervals. Given a sequence of packet successes or failures, we can provide confidence intervals about the packet success rate as follows:

$$q_{\min} = \min \left\{ \hat{q} \in [0, 1] : F(\sum_{k=0}^{N-1} \gamma_k; N, \hat{q}) \leq 1 - \delta \right\}, \tag{40}$$

$$q_{\max} = \max \left\{ \hat{q} \in [0, 1] : F(\sum_{k=0}^{N-1} \gamma_k; N, \hat{q}) \geq \delta \right\}, \tag{41}$$

where $F(r; N, \hat{q})$ is the Binomial distribution with parameters $N, \hat{q}$ at the points $r = 0, 1, \ldots, N$. We note that this approach has again the guarantee that the probability of getting a wrong answer is by design less than δ for any sample size N as in (19). A drawback of this method is that it is still practically conservative, in the sense that it gives wide confidence intervals as described at the beginning of this section — see also Agresti and Coull (1998).

Approximate method. A less conservative and popular method for finding binomial confidence intervals is using the normal approximation, also called the Wald interval. This method exploits the fact that by the Central Limit Theorem $\frac{\sum_{k=0}^{N-1}(\gamma_k - q)}{\sqrt{Nq(1-q)}}$ behaves like a standard normal distribution in the limit. Hence this method proposes intervals

$$q_{\min} = \hat{q}_N - z(\delta)\sqrt{\frac{\hat{q}_N(1-\hat{q}_N)}{N}}, \tag{42}$$

$$q_{\max} = \hat{q}_N + z(\delta)\sqrt{\frac{\hat{q}_N(1-\hat{q}_N)}{N}}, \tag{43}$$

where $z(\delta)$ is the $1-\delta$ quantile of the standard normal distribution. This method is approximate and does not have the guarantee that the probability of a wrong answer is by design less than δ for any sample size N as in (19). However practically it works well for sufficiently large number of samples and for probabilities q away from the extreme 0 and 1 — see also Agresti and Coull (1998).

We can substitute the confidence intervals produced by any of these practical methods in the stability analysis algorithm of Section 3.1. In Section 5 we compare these methods numerically — see specifically Figs. 6 and 7.

4. Control performance analysis using channel samples

Beyond verifying stability, given a finite number of channel samples we may be interested in verifying with accuracy whether the control cost of the system is below some required bound $J(q) \leq J_{\text{req}}$, supposing the system is stable. With the collected samples we can construct a high-confidence lower bound on the channel success q. Since the function $J(q)$ is non-increasing according to Proposition 1, we can check whether the control system performance under this high confidence lower bound on q is lower than the desired cost value J_{req}. This is shown in Algorithm 2.

Theorem 4. *Consider the switched linear system (3) over an i.i.d. Bernoulli binary channel with an unknown success probability $q \in [0, 1]$ and assume $q > 1 - \frac{1}{\rho(A)^2}$. Let q^* be the optimal solution to the problem*

$$\begin{aligned} &\textit{minimize} && \tilde{q} && \text{(49a)}\\ &\textit{subject to} && Tr(PW) \leq J_{req}, && \text{(49b)}\\ & && P = Q + (1-\tilde{q})A^T PA, && \text{(49c)}\\ & && P \succeq 0,\ 0 \leq \tilde{q} \leq 1, && \text{(49d)} \end{aligned}$$

and assume $q \neq q^*$. Consider the control cost analysis procedure developed in Algorithm 2 using N i.i.d. channel samples drawn with success rate q and some parameter $\delta \in (0, 1)$. Then:

(1) for any number of samples N the procedure returns a wrong answer with probability less than δ,

(2) if the number of samples is

$$N \geq \frac{2\log(1/\delta)}{(q-q^*)^2}, \tag{50}$$

then the procedure returns a correct answer with probability at least $(1-\delta)$.

The proof is included next and is primarily based on similar arguments as in the previous section.

Proof. We consider the case where the cost of the system is indeed lower than the desired value, i.e., $J(q) \leq J_{\text{req}}$ – a symmetric argument verifies the case when the cost is larger that the desired value. Note by Proposition 1 and the expression for the control cost $J(q)$ given in (2), we have that the value q is feasible for problem (49), and since q^* is the optimal solution for problem (49), this is implies that $q \geq q^*$, and since we have assumed that $q \neq q^*$, we have that $q > q^*$.

Proof of (2): First we show that the algorithm returns the correct answer with high probability. Following the same arguments as in the proof of Theorem 1 and Corollary 2 and since $q > q^*$, if the number of samples satisfies (50), then the event $q^* < q_{\min}$ occurs with high probability (at least $1-\delta$). Then since q^* is the

Algorithm 2 Performance verification

Input: Dynamics A, Noise covariance W, Desired cost level J_{req}, Confidence level δ, Number of samples N, Channel samples $\gamma_0, \dots, \gamma_{N-1} \in \{0,1\}^N$

Compute the sample average

$$\hat{q}_N = \frac{1}{N}\sum_{k=0}^{N-1} \gamma_k \tag{44}$$

Compute the high confidence lower and upper bounds

$$q_{\min} = \max\{0, \hat{q}_N - \sqrt{\frac{\log(1/\delta)}{2N}}\} \tag{45}$$

$$q_{\max} = \min\{1, \hat{q}_N + \sqrt{\frac{\log(1/\delta)}{2N}}\} \tag{46}$$

if there exists a matrix P such that

$$\text{Tr}(PW) \le J_{\text{req}}, \tag{47a}$$

$$P = Q + (1-q_{\min})A^T PA, \tag{47b}$$

$$P \succeq 0. \tag{47c}$$

then

  return 'Cost is less than J_{req}'

else

  if there exists a matrix P such that

$$\text{Tr}(PW) \ge J_{\text{req}}, \tag{48a}$$

$$P = Q + (1-q_{\max})A^T PA, \tag{48b}$$

$$P \succeq 0. \tag{48c}$$

  then

    return 'Cost is larger than J_{req}'

  else

    return 'Undetermined'

  end if

end if

optimal solution to problem (49), this also implies that the data-dependent random variable $q_{\min}$ is feasible for problem (49) with high probability. Hence there exists a matrix $P_{\min} \succeq 0$ such that $P_{\min} = Q + (1-\tilde{q})A^T P_{\min} A$ and $\text{Tr}(P_{\min} W) \le J_{\text{req}}$. This matrix makes also problem (47) feasible, and this proves that the algorithm returns the correct answer with high probability (at least $1-\delta$).

Proof of (1): Second we will show that the algorithm returns the wrong answer with low probability. For that we make use of the following fact:

Fact: If problem (48) is feasible, then $q_{\max} \le q^*$.

We have also argued that $q^* < q$. So if problem (48) is feasible, then we combine the last two statements to conclude that $q_{\max} \le q$. As a result the event that (48) is feasible implies the event that $q_{\max} \le q$. Hence we can bound the probability that the algorithm returns the wrong answer as

$$\mathbb{P}((48) \text{ feasible}) \le \mathbb{P}(q_{\max} \le q) \le \delta \tag{51}$$

where the latter follows by Hoeffding's inequality (Lemma 1).

To complete this proof we need to prove the above Fact. If problem (48) is feasible, then the cost of the system on a channel with rate $q_{\max}$ is larger than J_{req}, i.e., $J(q_{\max}) \ge J_{\text{req}}$. On the other hand by design of problem (49) we have that $q^* = \text{argmin}\{\tilde{q} : J(\tilde{q}) \le J_{\text{req}}\}$. Combining the two statements we conclude that $J(q_{\max}) \ge J_{\text{req}} \ge J(q^*)$. Finally we have shown in Proposition 1 that the function $J(q)$ is strictly decreasing. Hence $q_{\max} \le q^*$ and this verifies the above fact. ■

The problem (49) essentially searches for the minimum channel quality value that makes the control cost smaller than the desired bound, i.e., $q^* = \text{argmin}\{q \in [0,1] : J(q) \le J_{\text{req}}\}$. We note that the optimization problem (49) is not jointly convex in the variables q, P but can be solved with bisection. In contrast both problems (47) and (48) can be solved as convex optimization problems.

Interestingly the proposed cost performance analysis has qualitatively the same form of sample complexity as the stability analysis according to Corollary 2, even though these are two different questions. More specifically we see that what matters is the difference between the channel conditions q and the solution to the problem (49). Again we see that properties of the networked system over the channel adversely affects the required amount of samples.

We note here that unlike the stability verification question of the previous section, here even if the system under consideration is asymptotically stable ($\rho(A) < 1$), testing whether its performance satisfies a desired value may still require a very large number of samples.

5. Numerical simulations

We consider a system of the form (3) with spectral radius $\rho(A) = 2$ that evolves over a Bernoulli channel with success rate $q = 0.9$. For this value the system is stable because (4) holds. For 1000 trials we draw $N = 2000$ i.i.d. channel samples according to the success rate q. For each trial we run the stability test described in Algorithm 1.

In Fig. 4, for different trials we plot the value of the high-confidence lower bound $q_{\min}$ on the true packet success rate q computed by Algorithm 1 as the number of samples N grow. These lower bounds converge to the true packet success, also plotted in the figure. We also plot the minimum packet success rate required for stability which is $1 - 1/\rho(A)^2$ as described in Proposition 1. Algorithm 1 checks stability by checking whether the lower bounds exceed the minimum packet success rate. As the number of channel samples grows, on average more of the lower bounds exceed the threshold and the algorithm correctly verifies the stability of the system. In Fig. 4 this takes about 300 channel data samples.

We record the responses of the algorithm as 'Unstable', 'Stable', 'Undetermined'. Across all trials we average how many times the algorithm returns the correct answer 'Stable'. This is an empirical evaluation of the correctness of the algorithm, similar to the theoretical bound described by Theorem 1 (cf. (18)). In Fig. 5 we plot both the empirical average correctness of the algorithm as well as its theoretical bound as a function of the number of channel samples drawn. First, we observe that the theoretical bound indeed is a lower bound on the average correctness of the algorithm. We note also that the bound is not tight. That means that fewer channel samples are actually required to learn whether the system is stable or not than what is predicted by our theoretical bound. The reason is that Hoeffding's inequality is not tight. Empirically however the rate at which the algorithm correctly learns the system stability as the number of samples grows seems to match the rate at which the theoretical bound grows. Hence numerically it seems that our bound captures the complexity with respect to the number of samples. Note here that the spikes appearing in the figure are not noise due to the random samples, they appear because the number of packet successes is a discrete variable, not continuous.

Then in Fig. 6 we compare different algorithms for producing high confidence intervals. In particular we compare Algorithm

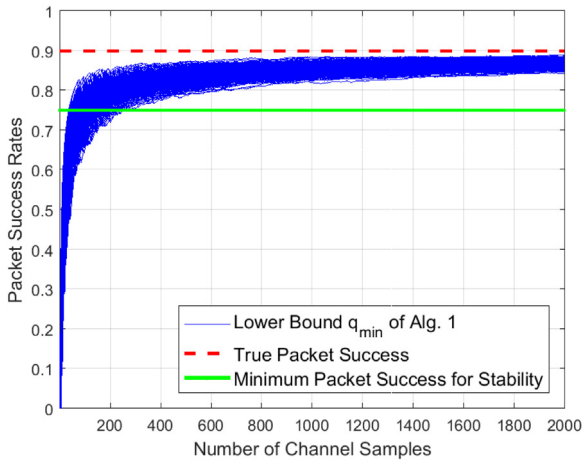


Fig. 4. We consider a system and channel that lead to stability. For different trials we plot the value of the high-confidence lower bound on the true packet success rate q computed by Algorithm 1 as the number of samples N grow. These lower bounds grow on average above the minimum packet success rate required for stability as described in Proposition 1, also plotted. As the number of channel samples grows, on average the algorithm correctly verifies the stability of the system.

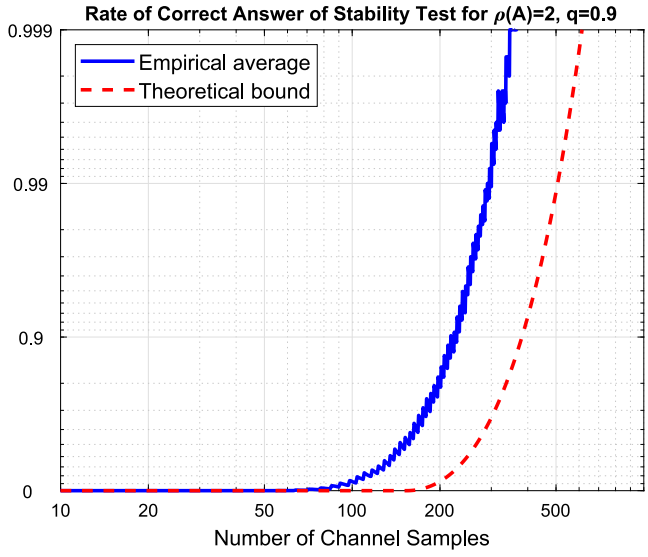


Fig. 5. We consider a system and channel that lead to stability. As the number of channel samples grows, the probability that Algorithm 1 correctly verifies the stability of the system grows. The theoretical bound by Theorem 1 is below the empirical bound obtained by simulation.

1 which is based on Hoeffding's inequality with the algorithms described in Section 3.4 derived based on the binomial distribution. In particular we consider the algorithm based on exact confidence intervals and the algorithm based on normal approximation. We perform again an empirical evaluation of the rate of correctness of the algorithm. For multiple trials we draw N i.i.d. channel samples according to the success rate q and we record the responses of the algorithms as 'Unstable', 'Stable', 'Undetermined'. Across all trials we average how many times each algorithm returns the correct answer 'Stable'. We observe that the

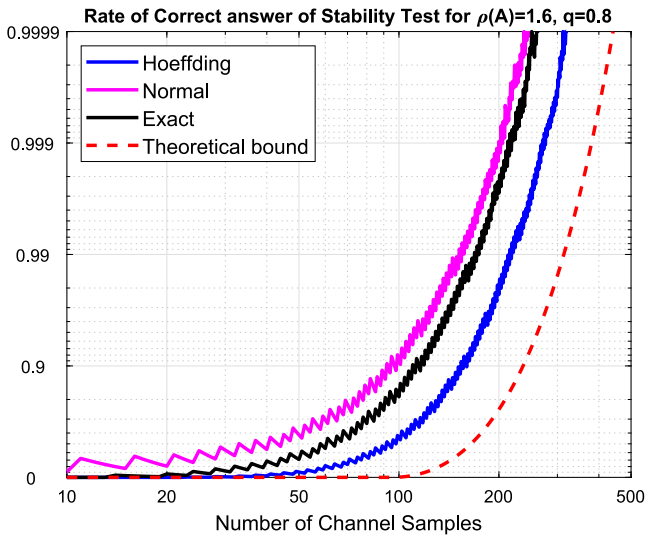


Fig. 6. Comparison of stability verification algorithms using different confidence interval methods. We compare methods based on Hoeffdings inequality, the exact method, and the normal approximation method (Section 3.4). We empirically plot how often the stability verification algorithms return the correct answer across a large number of iterations and number of channel samples. The normal approximation method gives the least conservative results and is more sample efficient.

algorithm based on Hoeffding's inequality is the most conservative one. The algorithm based on exact confidence intervals has better performance, i.e., requires fewer samples on average, while the algorithm based on normal approximation provides the best performance empirically. On the other hand we observe that the average correctness of all algorithms seems to increase at similar rates with the number of samples.

In Fig. 7 we empirically plot how often the stability verification algorithms return the *wrong answer* , i.e., return that the system is 'Unstable' while the system is stable, across a large number of iterations and number of channel samples. The normal approximation method often gives the wrong answer especially for few number of samples. On the other hand, both the method based on Hoeffding's inequality and the exact method give the wrong answers with probability less than the desired confidence ($\delta = 10^{-3}$ here). We note that for this figure we chose a system that has very small stability margin over the channel (specifically $1 - 1/\rho(A)^2 \approx 0.49$ is close to $q = 0.5$ here). This shows that even in such extreme scenarios the method has guaranteed low probability of error (except for the normal approximation method for small number of samples).

6. Discussion

The assumption that the collected channel samples are i.i.d. following a Bernoulli distribution is crucial in the above results. In practice, only the channel sample data is available and no a priori knowledge about their distribution class, e.g., whether they are i.i.d., as in this paper, or whether they are correlated or even non-stationary. This is an important practical concern. For example, if we utilize the stability test developed in Algorithm 1 under i.i.d. channel assumptions on data that is not i.i.d. the probability of error in the procedure does not necessarily obey the bounds given in Theorem 1. Ideally, we would like a more robust sample-based stability analysis. This is the topic of future work.

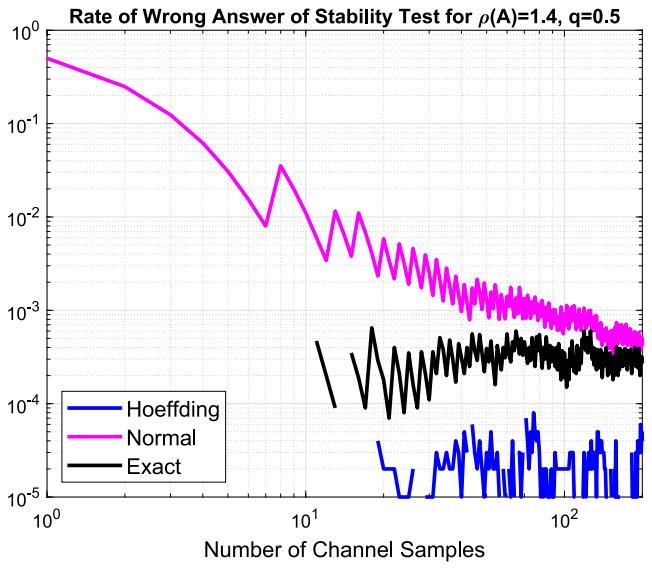


Fig. 7. Comparison of stability verification algorithms using different confidence interval methods. We compare methods based on Hoeffding's inequality, the exact method, and the normal approximation method (Section 3.4). We empirically plot how often the stability verification algorithms return the wrong answer across a large number of iterations and number of channel samples. The normal approximation method often gives the wrong answer especially for few number of samples.

7. Conclusions

In this paper we consider networked control systems over unknown channels. We utilize a learning procedure based on channel sample measurements to analyze whether the system is stable or not over a given channel, and we also analyze its control performance in terms of a quadratic control cost. As the stability analysis procedure is based on random samples there is a probability of error and we characterize this probability as a function of the true channel model, the system dynamics, and the number of collected channel samples. In particular we illustrate both theoretically and in simulations that the number of channel samples required adversely depends on the networked system stability margin, measured by the difference between the system spectral radius and the unknown channel packet success rate.

Future work involves the analysis of more complex channel models and the corresponding sample complexity, other modes of stability such as higher order moment, as well as controller synthesis for networked control systems over unknown channels using channel samples collected online while the system is controlled.

References

Abbasi-Yadkori, Y., & Szepesvári, C. (2011). Regret bounds for the adaptive control of linear quadratic systems. In *Proceedings of the 24th annual conference on learning theory* (pp. 1–26).

Agresti, A., & Coull, B. A. (1998). Approximate is better than "exact" for interval estimation of binomial proportions. *The American Statistician*, *52*(2), 119–126.

Araújo, J., Mazo, M., Anta, A., Tabuada, P., & Johansson, K. H. (2014). System architectures, protocols and algorithms for aperiodic wireless control systems. *IEEE Transactions on Industrial Informatics*, *10*(1), 175–184.

Baumann, D., Solowjow, F., Johansson, K. H., & Trimpe, S. (2019). Event-triggered pulse control with model learning (if necessary). In *2019 American control conference* (pp. 792–797). IEEE.

Baumann, D., Zhu, J.-J., Martiu, G., & Trimpe, S. (2018). Deep reinforcement learning for event-triggered control. in *2018 IEEE 57rd annual conference on decision and control*.

Bertsekas, D. P. (2012). *Dynamic programming and optimal control (vol. 2)* (4th ed.). MA: Athena scientific Belmont.

Boucheron, S., Lugosi, G., & Bousquet, O. (2003). Concentration inequalities. In *Summer school on machine learning* (pp. 208–240). Springer.

Costa, O. L. V., Fragoso, M. D., & Marques, R. P. (2006). *Discrete-time Markov jump linear systems*. Springer Science & Business Media.

Dean, S., Mania, H., Matni, N., Recht, B., & Tu, S. (2019). On the sample complexity of the linear quadratic regulator. *Foundations of Computational Mathematics,*, 1–47.

Demirel, B., Ramaswamy, A., Quevedo, D. E., & Karl, H. (2018). DeepCAS: A deep reinforcement learning algorithm for control-aware scheduling. in *2018 IEEE 57rd annual conference on decision and control*.

Durrett, R. (2010). *Probability: theory and examples*. Cambridge University Press.

Eisen, M., Gatsis, K., Pappas, G. J., & Ribeiro, A. (2019). Learning in wireless control systems over non-stationary channels. *IEEE Transactions on Signal Processing*, *67*(5), 1123–1137.

Gatsis, K., Pajic, M., Ribeiro, A., & Pappas, G. J. (2015). Opportunistic control over shared wireless channels. *IEEE Transactions on Automatic Control*, *60*(12), 3140–3155.

Gatsis, K., & Pappas, G. J. (2018). Sample complexity of networked control systems over unknown channels. in *57th IEEE conference on decision and control*.

Gatsis, K., Ribeiro, A., & Pappas, G. J. (2014). Optimal power management in wireless control systems. *IEEE Transactions on Automatic Control*, *59*(6), 1495–1510.

Halperin, D., Hu, W., Sheth, A., & Wetherall, D. (2010). Predictable 802.11 packet delivery from wireless channel measurements. *40*, In *ACM SIGCOMM Computer Communication Review* (4), (pp. 159–170). ACM.

Hespanha, J. P., Naghshtabrizi, P., & Xu, Y. (2007). A survey of recent results in networked control systems. *Proceedings of the IEEE*, *95*(1), 138–162.

Kumar, P. R., & Varaiya, P. (2015). *Stochastic systems: Estimation, identification, and adaptive control (vol. 75)*. SIAM.

Leong, A. S., Ramaswamy, A., Quevedo, D. E., Karl, H., & Shi, L. (2020). Deep reinforcement learning for wireless sensor scheduling in cyber–physical systems. *Automatica*, *113*, Article 108759.

Mamduhi, M. H., Tolić, D., Molin, A., & Hirche, S. (2014). Event-triggered scheduling for stochastic multi-loop networked control systems with packet dropouts. In *2014 IEEE 53rd annual conference on decision and control* (pp. 2776–2782). IEEE.

Mazo, M., & Tabuada, P. (2011). Decentralized event-triggered control over wireless sensor/actuator networks. *IEEE Transactions on Automatic Control*, *56*(10), 2456–2461.

Quevedo, D. E., Ahlén, A., Leong, A. S., & Dey, S. (2012). On Kalman filtering over fading wireless channels with controlled transmission powers. *Automatica*, *48*(7), 1306–1316.

Rappaport, T. S., MacCartney, G. R., Samimi, M. K., & Sun, S. (2015). Wideband millimeter-wave propagation measurements and channel models for future wireless communication system design. *IEEE Transactions on Communications*, *63*(9), 3029–3056.

Sahai, A., & Mitter, S. (2006). The necessity and sufficiency of anytime capacity for stabilization of a linear system over a noisy communication link&# 8212; part I: Scalar systems. *IEEE Transactions on Information Theory*, *52*(8), 3369–3395.

Schenato, L., Sinopoli, B., Franceschetti, M., Poolla, K., & Sastry, S. (2007). Foundations of control and estimation over lossy networks. *Proceedings of the IEEE*, *95*(1), 163–187.

Schuurmans, M., Sopasakis, P., & Patrinos, P. (2019). Safe learning-based control of stochastic jump linear systems: a distributionally robust approach. In *2019 IEEE 58th conference on decision and control* (pp. 6498–6503). IEEE.

Solowjow, F., Baumann, D., Garcke, J., & Trimpe, S. (2018). Event-triggered learning for resource-efficient networked control. In *2018 Annual American control conference* (pp. 6506–6512). IEEE.

Tatikonda, S., & Mitter, S. (2004). Control under communication constraints. *IEEE Transactions on Automatic Control*, *49*(7), 1056–1068.

Wu, S., Ren, X., Jia, Q.-S., Johansson, K. H., & Shi, L. (2019). Learning optimal scheduling policy for remote state estimation under uncertain channel condition. *IEEE Transactions on Control of Network Systems*.

Yoo, J., & Johansson, K. H. (2019). Event-triggered model predictive control with a statistical learning. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*.



Konstantinos Gatsis received the Ph.D. degree in electrical and systems engineering from the University of Pennsylvania, Philadelphia, PA, USA, in 2016. Currently, he is a Departmental Lecturer in the Department of Engineering Science, University of Oxford, UK. His research interests include control and optimization applied to cyber–physical systems. Dr. Gatsis received the Joseph, D16, and Rosaline Wolf Award for Best Doctoral Dissertation from the Department of Electrical and Systems Engineering at the University of Pennsylvania, the 2014, O. Hugo Schuck Best Paper Award, the Student Best Paper Award at the 2013 American Control Conference, and was a Best Paper Award Finalist at the 2014 ACM/IEEE International Conference on Cyber–Physical Systems.



George J. Pappas received the Ph.D. degree in electrical engineering and computer sciences from the University of California, Berkeley, CA, USA, in 1998. He is currently the Joseph Moore Professor and Chair of the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA, USA. He also holds a secondary appointment with the Department of Computer and Information Sciences and the Department of Mechanical Engineering and Applied Mechanics. He is a Member of the GRASP Lab and the PRECISE Center. He had previously served as the Deputy Dean for Research with the School of Engineering and Applied Science. His research interests include control theory and, in particular, hybrid systems, embedded systems, cyber–physical systems, and hierarchical and distributed control systems, with applications to unmanned aerial vehicles, distributed robotics, green buildings, and biomolecular networks. Dr. Pappas has received various awards, such as the Antonio Ruberti Young Researcher Prize, the George S. Axelby Award, the Hugo Schuck Best Paper Award, the George H. Heilmeier Award, the National Science Foundation PECASE award, and numerous best student papers awards at ACC, CDC, and ICCPS.