

Control with actuation anticipation

Ena Hariyoshi, Gireeja Ranade, and Anant Sahai

Abstract—Reachability traditionally thinks about the controllability Grammian of a system, i.e. the actuation vectors as they pass through the plant dynamics. This perspective depends on fully knowing the future plant dynamics even as we apply controls in the present, since the ability to plan is important for control.

This paper explores a toy model with uncertain random actuation vectors. Our ability to plan is modulated by how much we can anticipate about the plant’s future actuation. The results here philosophically build on the concept of “control capacity” introduced earlier and the model itself is inspired by earlier work on intermittent Kalman filtering and problems of networked control with packet drops.

A simple greedy strategy is optimal for our toy model and we can easily characterize the informational value of knowing future actuation vectors. Furthermore, the control capacity of the toy system can be stated in a way that is suggestive of a dimensional-sense of “signal-to-noise-ratio.”

I. INTRODUCTION

This paper studies a simple toy model with unpredictable actuation and follows up on our earlier paper at Allerton [1], where an even simpler form of this toy was introduced. Readers are directed to the introduction in our earlier paper to see a more extensive discussion of how the toy model explored here was inspired (see [2]) by the problems of dropped observation packets [3], [4] as well as dropped control packets [5], [6] in networked control systems. The discussion in [1] also connects this problem to earlier work on “control capacity” with multiplicative noise on the observation channel [7], [8] and actuation channel [9], [10], as well as work on the value of non-causal information such as [11], [12].

The problem introduced in [1] is an idealized version of trying to catch a stationary target in a high dimensional space, but where we are only allowed to move along randomly chosen actuation vectors that are revealed to us as we go. The state is taken to be our relative position to the target.

Explicitly, we considered a d -dimensional vector control system with random state $\mathbf{X}[n]$, such that at each time we are allowed to move along a single random actuation direction $\mathbf{B}[n]$:

$$\mathbf{X}[n+1] = \mathbf{X}[n] + \mathbf{B}[n]U[n]. \quad (1)$$

The actuation vector $\mathbf{B}[n]$ is assumed to be drawn uniformly from the d -dimensional unit hypersphere [13], and our control $U[n]$ is a scalar. In [1], the value of non-causal information was studied by giving us access to a constant

number k of future vectors $\mathbf{B}[n+1], \dots, \mathbf{B}[n+k-1]$ and seeing how this impacted the rate at which we could make the state $\mathbf{X}[n]$ approach zero in a mean-squared sense.

In this paper, we change two aspects of the toy model from [1]. First, we consider vector controls $\mathbf{U}[n]$ and consequently have the ability to move along many different randomly drawn actuation directions at once. Second, we allow both the number of actuation directions and the amount of lookahead to vary (possibly non-stationarily) with time. Explicitly, the new model for a system \mathcal{S} is:

$$\begin{aligned} \mathbf{X}[n+1] &= \mathbf{X}[n] - \mathbf{R}[n]\mathbf{U}[n] \\ \mathbf{Y}[n] &= (\mathbf{X}[n], \mathbf{T}[n]), \end{aligned} \quad (2)$$

with deterministic initial condition $\mathbf{x}[0]$.

Notice that the d -dimensional random state $\mathbf{X}[n] \in \mathbb{R}^d$ still evolves in basically the same manner. Now, instead of a single random $\mathbf{B}[n]$ we have an ordered set (a list) $\mathbf{R}[n]$ consisting of $r[n]$ different column vectors representing the actuation directions available to move along at time n . These vectors are chosen uniformly at random from the d -dimensional unit hypersphere. For simplicity, we slightly abuse notation and interpret this ordered set of columns as an $\mathbb{R}^{d \times r[n]}$ matrix so that the $r[n]$ dimensional control vector $\mathbf{U}[n]$ can represent the amount we move along each of those actuation directions using regular matrix multiplication.

At each time step n , we also learn $t[n]$ new actuation directions and this is represented by the list $\mathbf{T}[n]$. Each vector we learn comes with a timestamp of when we will be able to act on it: some might be available to act on at the current time n , but others might only be available at future times. Certainly, every vector in $\mathbf{R}[n]$ is learned by time n and so is in $\mathbf{T}[i]$ for $i \leq n$. Both the lengths $r[n]$ and $t[n]$ can vary but are subject to the constraint that $\forall n \sum_{i=0}^n r[i] \leq \sum_{i=0}^n t[i]$. There is assumed to be a terminal time N by which time everything learned has been used so $\sum_{i=0}^N r[i] = \sum_{i=0}^N t[i]$.

The control vector $\mathbf{U}[n]$ is allowed to causally depend on all of the information \mathbf{Y} we have up to and including time n , i.e. $\mathbf{U}[n] = f(\mathbf{Y}[0], \dots, \mathbf{Y}[n])$. This information represents all the states so far as well as all the actuation directions learned so far. The goal is to minimize the terminal cost, $\mathbb{E}[\|\mathbf{X}[N]\|^2]$.

For this problem, optimal results are obtained (by means of dynamic programming) for driving the second-moment to zero. These results are interpreted in a few different ways. First, we take an informational perspective and interpret the result as an asymptotic control capacity that bounds how fast we can dissipate the state per unit time. The resulting expression bears a striking resemblance to the communication capacity of an additive Gaussian noise channel. Second,

E. Hariyoshi ehariyoshi@berkeley.edu and A. Sahai sahai@eecs.berkeley.edu are with UC Berkeley EECS

G. Ranade giranade@microsoft.com is with Microsoft Research, Redmond, WA

we interpret this control capacity in the large-dimensional limit. Third, we pivot from a per unit time perspective to a per-actuation-vector perspective and show how the decay rate can be understood in terms of the “information context” in which each of the actuation vectors are used or learned, and we precisely define this notion.

II. PROBLEM SETUP AND NOTATION

In this paper we consider the basic system \mathcal{S} that was introduced in (2). Throughout this paper, we use upper-case letters to represent random variables, lower-case to represent deterministic quantities, boldface to represent vectors, and sans-serif (e.g. S) to represent lists/matrices. We use standard elementary linear-algebraic ideas like span and projections with natural notation, e.g. $\text{proj}_S \mathbf{x}$ means the projection of \mathbf{x} onto the subspace spanned by the columns of S .

As in [1] the underlying randomness here is in the choice of actuation vectors. Each vector in $\mathbf{T}[n]$ is assumed to have been chosen i.i.d. uniformly from the d -dimensional unit hypersphere. These may be generated by drawing i.i.d. components from a standard Normal and then normalizing the vector to be on the unit hypersphere. Alternatively, they can be generated by taking a standard basis unit vector and multiplying it by a random rotation matrix chosen uniformly from $\text{SO}(d)$.

One additional piece of basic notation is helpful. Since we assume that we can use each actuation direction exactly once, we let $S[n]$ be the set of $s[n]$ actuation vectors at time n that we know about but have not used up yet. By assumption, $R[n] \subseteq S[n]$. WLOG assume that the list $R[n]$ is a prefix of $S[n]$. In other words, the set of actuation vectors that we can move along at time n is a subset of the actuation vectors we know about at time n — in particular, the first few vectors.

$\mathbf{T}[n]$ populates $S[n]$ and $R[n]$ according to these equations:

$$\begin{aligned} S[0] &= \mathbf{T}[0] \\ S[n] &= (S[n-1] \setminus R[n-1]) \cup \mathbf{T}[n]. \end{aligned} \quad (3)$$

where the \cup means the concatenation of the ordered sets (lists) and the \setminus means dropping entries from the beginning of the list. The sizes of these lists viewed as sets also must satisfy $s[n] = t[n] + (s[n-1] - r[n-1])$ since the vectors involved will be distinct almost surely. Notice that each of the cardinalities $r[n], t[n], s[n]$ are all going to always be strictly less than d in this paper. (If any one were d or greater, then because a collection of independently randomly drawn vectors will be maximal rank almost surely, we will be able to catch the target exactly since the ambient dimension of the state-space is also d .) The Venn diagram in Figure 1 shows the relationship between these sets.

As a measure of performance, we are interested in the N -step expected decay of the system:

$$\frac{\mathbb{E}[\|\mathbf{X}[N]\|^2]}{\|\mathbf{x}[0]\|^2}. \quad (4)$$

We will find that this value is independent of the initial condition $\mathbf{x}[0]$. This is motivated by our desire to understand a notion of “control capacity” for vector actuation. Building

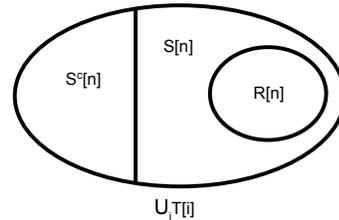


Fig. 1. The relationship between $S[n]$, $R[n]$ and $\mathbf{T}[n]$. $S^c[n] = \bigcup_{i=0}^{n-1} R[i]$ is the set of vectors that have been used so far. $\bigcup_{i=0}^n \mathbf{T}[i]$ represents the total vectors we have learned up to time n , regardless of whether we have acted on them or not yet.

on the definition in [9], we define the 2nd-moment control capacity for the system \mathcal{S} :

Definition 2.1: The 2nd-moment control capacity, $C_2(\mathcal{S})$ of the d -dimensional system \mathcal{S} as in (2) with initial state $\mathbf{x}[0]$ is defined as

$$\liminf_{N \rightarrow \infty} - \min_{U_0^{N-1}} \frac{d}{2N} \log_2 \mathbb{E} \left[\frac{\|\mathbf{X}[N]\|^2}{\|\mathbf{x}[0]\|^2} \right]$$

where the notation U_0^{N-1} is shorthand for the list U_0, U_1, \dots, U_{N-1} .

The units of control capacity here are bits since the logarithm is base 2. The scaling by d in front is because multiplying a generic d -dimensional random variable by 2 doesn’t increase the differential entropy by 1 bit but by d bits.

III. MAIN RESULT

Theorem 3.1: The minimum N -step expected decay for the system in (2) is

$$\frac{\mathbb{E}[\|\mathbf{X}[N]\|^2]}{\|\mathbf{x}[0]\|^2} = \prod_{n=0}^{N-1} \left(\frac{d - s[n]}{d - (s[n] - t[n])} \right). \quad (5)$$

Further, let $\mathbf{U}_{ls}[n]$ be the least square solution to $(S[n])\mathbf{U}[n] = \mathbf{X}[n]$, namely

$$\text{proj}_{S[n]} \mathbf{X}[n] = (S[n])\mathbf{U}_{ls}[n]. \quad (6)$$

Let $\mathbf{U}^*[n] \in \mathbb{R}^{r[n]}$ be given by those entries of $\mathbf{U}_{ls}[n] \in \mathbb{R}^{s[n]}$ which correspond to $R[n] \subseteq S[n]$. This $\mathbf{U}^*[n]$ is the optimal control achieving the performance given by (5).

The proof of this main theorem is given in Section V. However, to understand the statement of the theorem itself as regards the optimal choice of control, suppose that we have $\mathbf{X}[n] = [1 \ 2 \ 0]^T$, $S[n] = \{[0 \ 0 \ 1]^T, [0 \ 1 \ 1]^T\}$, and $R[n] = \{[0 \ 0 \ 1]^T\}$. Then, $\text{proj}_{S[n]} \mathbf{X}[n] = [0 \ 2 \ 0]^T$ and

$$\begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} -2 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ 0 \end{bmatrix}$$

Therefore, we have $\mathbf{U}^*[n] = [-2]$ since that is the entry that corresponds to the actuation vector that we actually get to act along in this period.

To further interpret the result of Theorem 3.1, we are interested in a case where $s[n] = k$, $t[n] = v$, and $r[n] = v$

are constant in steady state. This corresponds to $(k-v)$ -step lookahead, since $k-v$ vectors additional to those acted upon are known per timestep.

Specifically, we start off with $s[0] = k$, $t[0] = k$, and $r[0] = v$. Then we have $s[n] = k$, $t[n] = v$, and $r[n] = v$ for all $n \leq N'$, where $N' = N - \lceil \frac{k}{v} \rceil - 1$. For times in the endgame $n > N'$, we have $t[n] = 0$. $r[n] = v$ until the penultimate step $N-1$, which uses up the rest of $s[n]$ — this is v or less depending on whether it divides cleanly. It turns out that the specifics of the steps $n > N'$ do not matter much for analysis, since Theorem 3.1 shows that $t[n] = 0$ makes those factors in the product be 1 for the corresponding terms.

A direct application of Theorem 3.1 yields:

$$\frac{\mathbb{E}[\|\mathbf{X}[N]\|^2]}{\|\mathbf{x}[0]\|^2} = \frac{d-k}{d} \left(\frac{d-k}{d-(k-v)} \right)^{N'} \quad (7)$$

This in turn is easily converted into a control capacity.

Corollary 3.2: The 2nd-moment control capacity of this system, with $(k-v)$ vectors of anticipatory lookahead where we steadily get to use v vectors per time step, is

$$\frac{d}{2} \log_2 \left(1 + \frac{v}{d-k} \right). \quad (8)$$

Notice the striking similarity of this formula with the capacity of a continuous-time (complex) additive white Gaussian noise channel [14]:

$$C = W \log_2(1 + SNR) \quad (9)$$

The correspondence shows that the number of dimensions d is acting analogously to the bandwidth W . The number v of new dimensions you learn/act-upon per timestep, is acting like the “signal strength”, whereas the total number of other unknown dimensions $d-k$ play the role of the “noise”.

In Section VII, we will further interpret the main result on a vector by vector basis (paralleling the “waterfilling” interpretation of Gaussian capacity in [14]) while in Section VI we will see what happens in the limit of large dimensions d (paralleling the wideband limit of Gaussian channel capacity).

IV. EXPERIMENTAL RESULTS

To better understand how this toy system behaves and to see just how well our main theorem predicts its behavior, we ran a computer simulation of the system with steady lookahead.

We started each sample path with $\mathbf{x}[0]$ being a random unit vector drawn uniformly from the d -dimensional unit hypersphere, so the starting norm is always 1.

Figure 2 shows the squared norm of the vector over time, for 10000 sample paths, for $d = 10$ dimensions, $v = 2$ actuation directions acted upon and learned at each time, $k = 5$ total actuation directions known at each time (so 2 from the current time plus another 3 from the next two time steps), and $N' = 100$. The box-and-whiskers plot shows the median and quartiles every 10 timesteps. The red path corresponds to the empirical means, whereas the black line

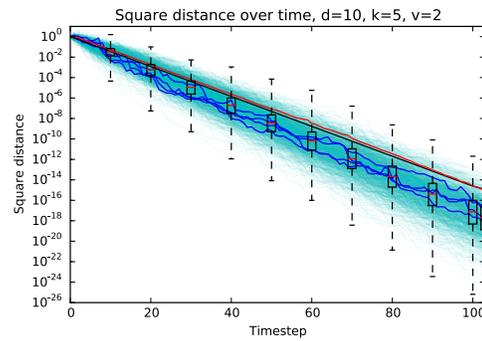


Fig. 2. 10000 simulated sample paths plotted together with some quartiles (with the medians marked red), the empirical means (red path) and predicted means (black). We have chosen four sample paths to highlight what typical paths look like.

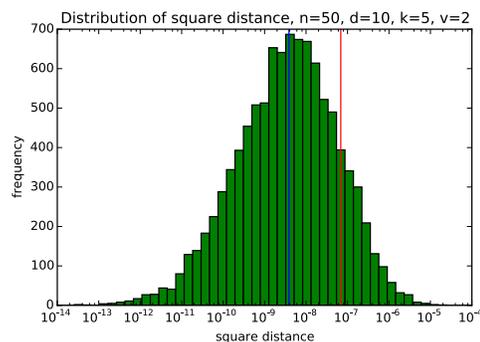


Fig. 3. An empirical histogram of what the squared distance is at time-step 50. The blue line marks the median, and the red line marks the mean.

corresponds to the prediction of the mean from (7). We can see that these two lines track each other closely.

Figure 3 shows the empirical distribution of the squared distance at a particular time step $n = 50$. The histogram looks Gaussian, and since the horizontal axis is on a log scale, we guess that the underlying distribution is approximately lognormal.

Figure 2 shows that our theorem predicts the value of the empirical mean very closely. To verify that indeed the theorem holds for various values of d, v , and k , we plot the predicted expected decay versus the actual decay in Figure 4. We use (7) for the prediction curve¹ corresponding to the $\frac{d-k}{d-(k-v)}$, the predicted steady-state decay rate. The red dots plot $\text{mean}(\|\mathbf{X}[N]\|^2)^{1/100}$ where the average is over 10000 samples, with $N = 100$.

Although the predictions for the mean of the squared distance are very good, looking at the sample paths in Figure 2 shows a discrepancy between the mean and the median. Not only is the median below the mean, the slopes of the lines are also different — the gap is growing. This is consistent with the conjecture that the distribution of

¹Because we are using a finite $N' = 100$ for this simulation, the actual prediction curve that we are plotting is $(\mathbb{E}[\|\mathbf{X}[N]\|^2])^{1/100} = \left(\frac{d-k}{d} \left(\frac{d-k}{d-(k-v)} \right)^{100} \right)^{1/100}$.

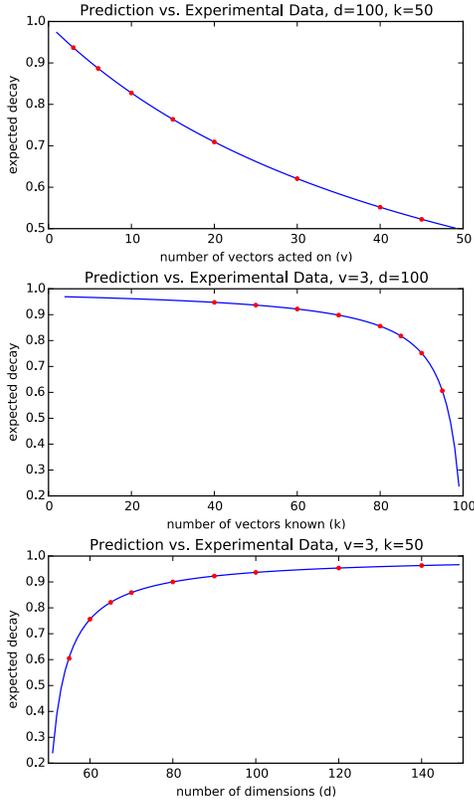


Fig. 4. This figure shows three slices through the formula predicting the long-term decay rate of the average squared distance. The top plot varies the number of actuation vectors that we learn/act-along at each time, but restricts us to always know 50 vectors. The second plot varies the amount of look-ahead but restricts us to acting along 3 of them at each time. The third plot varies the dimension of the system but keeps us acting along 3 vectors at a time while knowing 50 of them.

the squared distance is lognormal (a product of many i.i.d. random variables), and hence has heavier tails. The mean is dominated by rare events that are exponentially larger than the typical case. This is why [9] defines different senses of control capacity depending on the moment of stability we are interested in.

V. PROOF OF MAIN RESULT

The proof of the main theorem parallels the proof in [1]. We use dynamic programming to obtain the optimal policy, showing that it is greedy. The system objective is to minimize the squared distance of the system vector at the final step N . So the cost to minimize is the terminal cost:

$$J_N^*(\mathbf{X}[N]) = J_N(\mathbf{X}[N]) = \|\mathbf{X}[N]\|^2.$$

The optimal cost-to-go at time n in general is thus:

$$J_n^*(\mathbf{X}[n], \mathbf{S}[n]) = \min_{\mathbf{U}[n]} (J_n(\mathbf{X}[n], \mathbf{S}[n], \mathbf{U}[n])),$$

where

$$J_n(\mathbf{X}[n], \mathbf{S}[n], \mathbf{U}[n]) = \mathbb{E}_{\mathbf{T}[n+1]} [J_{n+1}^*(\mathbf{X}[n+1], \mathbf{S}[n+1])].$$

Definition 5.1: Let \mathbf{x} be a vector and \mathbf{L} a subspace. Define

$$\mathbf{x}^{\perp \mathbf{L}} = \mathbf{x} - \text{proj}_{\mathbf{L}} \mathbf{x}, \quad (10)$$

as the component of \mathbf{x} that is orthogonal to the subspace \mathbf{L} .

We recall two lemmas from [1], which are key elements in the proof. Lemma 5.1 restates a basic fact about standard linear least squares.

Lemma 5.1: Let $\mathbf{x} \in \mathbb{R}^d$ and $\mathbf{M} \in \mathbb{R}^{m \times d}$, then

$$\min_{\mathbf{U}} \|\mathbf{x} - \mathbf{M}\mathbf{U}\|^2 = \|\mathbf{x}^{\perp \text{span}(\mathbf{M})}\|^2. \quad (11)$$

Furthermore, the minimizing solution for \mathbf{U} is unique whenever \mathbf{M} has linearly independent column vectors.

Lemma 5.2 that captures a basic fact concerning the symmetry of projecting a vector onto a subspace chosen uniformly at random.

Lemma 5.2: Let \mathbf{L} be an ℓ -dimensional subspace of \mathbb{R}^d that is chosen uniformly at random over d -space, and let \mathbf{x} be any vector in d -space, then

$$\mathbb{E}_{\mathbf{L}} [\|\text{proj}_{\mathbf{L}} \mathbf{x}\|^2] = \frac{\ell}{d} \|\mathbf{x}\|^2. \quad (12)$$

Now consider N' , the earliest time at which all actuation vectors that will ever be used (until final horizon N), are known. At step N' , we are guaranteed that $\mathbf{S}[N']$ contains all the actuation vectors until the end. Hence there is no more randomness in the system following timestep N' and only a sequence of minimization remains.

Using Lemma 5.1, we can find the expression for the cost-to-go at timestep N' :

$$\begin{aligned} J_{N'}^*(\mathbf{X}[N'], \mathbf{S}[N']) &= \min_{\mathbf{U}[N'], \dots, \mathbf{U}[N-1]} \|\mathbf{X}[N]\|^2 \\ &= \|\mathbf{X}[N']^{\perp \text{span}(\mathbf{S}[N'])}\|^2. \end{aligned} \quad (13)$$

We combine this idea with the other basic lemmas to find the cost-to-go by induction:

Lemma 5.3: For $0 \leq n \leq N'$,

$$J_n^*(\mathbf{X}[n], \mathbf{S}[n]) = \prod_{m=n+1}^{N-1} \left(\frac{d-s[m]}{d-(s[m]-t[m])} \right) \|\mathbf{X}[n]^{\perp \text{span}(\mathbf{S}[n])}\|^2$$

Unfortunately, space prohibits a full proof. However, the argument is similar to that of [1] and the full proof will be included in a longer version.

The main result Theorem 3.1 follows from combining Lemma 5.3 and Lemma 5.2.

VI. LIMITING BEHAVIOR

Given the similarities between the formulae (8) and (9), it is interesting to explore the analogy between state dimension and bandwidth, as well as energy and actuation freedom. In information theory, the wideband limit of $W \rightarrow \infty$ is used to establish a famous connection between energy and information. In the limit of large bandwidth, communication capacity is energy limited and $k_{\text{Boltzmann}} T_{\text{temperature}} \ln 2$ units of energy is required to communicate a bit, a fact leveraged further as Landauer's Principle in physics [15]. So, what happens if we consider the control capacity as the dimension d of a system goes to infinity?

If the number v of actuation degrees of freedom per time is constant while the total number of actuation vectors known

at each time could increase affinely with d according to $k = k_0 + gd$ for some $0 \leq g < 1$, we can see that:

$$\begin{aligned} & \lim_{d \rightarrow \infty} \frac{d}{2} \log_2 \left(1 + \frac{v}{d-k} \right) \\ &= \frac{1}{2(1-g)} \log_2 \left(\lim_{d \rightarrow \infty} \left(1 + \frac{v}{d-k} \right)^{d-k} \right) \\ &= \frac{1}{2(1-g)} \log_2 (e^v) = v \frac{1}{(1-g)2 \ln 2} \end{aligned} \quad (14)$$

For the interesting special case of $g = 0$, it means that having a single generic degree of actuation freedom in a high dimensional system is worth $\frac{1}{2 \ln 2}$ bits.

VII. INFORMATION CONTEXT

The asymptotic analysis above suggests a change of perspective from time to the actuation vectors themselves.

Each term in the product within (5) is a ratio $\frac{d-s[n]}{d-(s[n]-t[n])}$. This ratio can be written as a telescoping product of $t[n]$ individual ratios, one for each actuation vector learned:

$$\frac{d-s[n]}{d-s[n]+1} \cdot \frac{d-s[n]+1}{d-s[n]+2} \cdots \frac{d-s[n]+t[n]-1}{d-s[n]+t[n]}$$

Each term in the telescoping product can be considered as arising from each individual actuation vector. Think of them being learned one at a time. The first one comes in with $s[n]-t[n]$ vectors known prior, the second one comes in with $s[n]-t[n]+1$ known prior, and the i -th one comes with $s[n]-t[n]+i-1$ known prior. This is what we want to call the learned information context for each actuation vector. It measures the informational context in which it was learned.

Definition 7.1: The vector $\mathbf{t}_{(n,i)}$ is defined to be the i -th vector learned at the beginning of time step n . In other words, $\mathbf{t}_{(n,i)}$ is the i -th column of $\mathbf{T}[n]$. The learned information context is defined to be $I(\mathbf{t}_{(n,i)}) = s[n] - t[n] + i - 1$.

With the learned information context, we can express

$$\begin{aligned} \prod_{n=0}^{N-1} \frac{d-s[n]}{d-(s[n]-t[n])} &= \prod_{n=0}^{N-1} \left(\prod_{i=1}^{t[n]} \frac{d-(s[n]-t[n]+i)}{d-(s[n]-t[n]+i-1)} \right) \\ &= \prod_{\mathbf{t}_{(n,i)}} \frac{d-(I(\mathbf{t}_{(n,i)})+1)}{d-I(\mathbf{t}_{(n,i)})} \end{aligned} \quad (15)$$

There is an analogous definition of information context for actuation vectors when they are acted along.

Definition 7.2: The vector $\mathbf{r}_{(n,i)}$ is defined to be the i -th vector used during time step n . In other words, $\mathbf{r}_{(n,i)}$ is the i -th column of $\mathbf{R}[n]$. The acted information context $I(\mathbf{r}_{(n,i)})$ for this vector is defined to be $I(\mathbf{r}_{(n,i)}) = s[n] - r[n] + i - 1$.

There is a key relationship between the learned information context and the acted information context. Either set can be used to compute the final optimal decay.

Theorem 7.1:

$$\prod_{\mathbf{t}_{(n,i)}} \frac{d-(I(\mathbf{t}_{(n,i)})+1)}{d-I(\mathbf{t}_{(n,i)})} = \prod_{\mathbf{r}_{(n,i)}} \frac{d-(I(\mathbf{r}_{(n,i)})+1)}{d-I(\mathbf{r}_{(n,i)})} \quad (16)$$

where all the $\mathbf{t}_{(n,i)}$, $\mathbf{r}_{(n,i)}$ are learned and acted upon within the epoch of N timesteps.

Although we omit the proof due to space, we can illustrate the intuition behind ‘‘information contexts’’ with an example. Consider an example where $N = 3$:

$$\begin{array}{lll} t[0] = 2 & r[0] = 1 & s[0] = 2 \\ t[1] = 3 & r[1] = 2 & s[1] = 4 \\ t[2] = 1 & r[2] = 3 & s[2] = 3 \end{array}$$

Here, a total of 6 actuation vectors are learned and acted along.

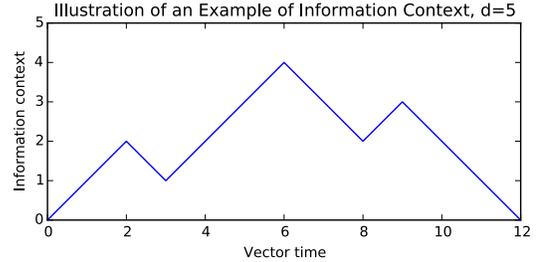


Fig. 5. Information contexts illustrated as vectors are first learned (rising segments) and then acted along (falling segments). The learned information contexts of vectors are on the rising segments and the acted information contexts on the falling segments.

To see the information contexts that each actuation vector experiences, consider Figure 5. The horizontal axis is a ‘‘vector virtual time’’, that ticks up when an actuation vector is learned or acted along. A learned vector represents an increase in the information context, and an acted-along vector represents a decrease. We can see that $s[0]$, $s[1]$, and $s[2]$ correspond to the peaks since some vectors are acted along at each time.

Because what goes up must come down, we can see from this graph that for any vector that is learned at a certain information context, there must be another vector that is acted along at the same information context.

At this point, the language of ‘‘types’’ from information theory and probability becomes useful. A type of a sequence is a count of how many times different items occur in that sequence [14]. Using that language, we see that $\{I(\mathbf{t}_{(n,i)})\}$ and $\{I(\mathbf{r}_{(n,i)})\}$ must have the same type when viewed as sequences of information contexts.

Consider the products in (16). On both sides, we could group terms by the actual value taken for the information context $I(\mathbf{r}_{(n,i)})$ (or equivalently $I(\mathbf{t}_{(n,i)})$). What values are possible? 0 is possible if a vector is learned or used with no other vector known. The highest possible information context is $c = d - 2$ since having $d - 1$ would mean that at some point, we knew a full d actuation vectors, which would let us set the state exactly to zero. Formally defining $n(c)$ to be

the number of times that $I(\mathbf{r}_{(n,i)}) = c$, we see:

$$\begin{aligned} \prod_{\mathbf{t}_{(n,i)}} \frac{d - (I(\mathbf{r}_{(n,i)}) + 1)}{d - I(\mathbf{r}_{(n,i)})} &= \prod_{c=0}^{d-2} \prod_{\mathbf{r}_{(n,i)}: I(\mathbf{r}_{(n,i)})=c} \frac{d - (I(\mathbf{r}_{(n,i)}) + 1)}{d - I(\mathbf{r}_{(n,i)})} \\ &= \prod_{c=0}^{d-2} \left(\frac{d - (c + 1)}{d - c} \right)^{n(c)} \\ &= \prod_{c=0}^{d-2} (d - c)^{n(c-1)} (d - c)^{-n(c)} \quad (17) \\ &= \prod_{c=0}^{d-2} (d - c)^{n(c-1) - n(c)}. \quad (18) \end{aligned}$$

Where (17) comes from realizing that the previous term in the product has the same form and so we can regroup terms if we add the definition $n(-1) = 0$. The above (18) suggests that what really matters is the “ignorance context” where we look at $d - c$. This captures how the ignorance of future actuation directions attenuate the effect of the control that we apply now.

VIII. CONCLUSIONS AND FUTURE WORK

In this paper, we introduced a very simple toy problem in which a target is pursued by moving along actuation directions provided to us uniformly at random. The idea of information context reveals that what matters for performance is how many future directions we already know while choosing how far to move along any particular direction. The more we know, the fewer dimensions we are ignorant of, and the faster we can approach the target.

All of the results in this paper were for the mean-squared-distance. However, both the simulation results and the nature of the proofs suggest a possible way forward that might give us all the moments of the distance. Of particular interest would be the “zeroth” (logarithmic) moment since that would give us the typical behavior. We can guess by looking at Figure 3 that the equivalent of a central limit theorem might result in the distribution of the distance to the target approaching a lognormal distribution. The typical behavior would then be governed by the location parameter of the lognormal.

IX. ACKNOWLEDGEMENTS

The authors would like to thank the NSF for CNS-0932410, CNS-1321155, and ECCS-1343398, and also for a CSOI fellowship for Ena Hariyoshi.

REFERENCES

- [1] E. Hariyoshi, G. Ranade, and A. Sahai, “Vector control systems with unpredictable actuation,” in *Communication, Control, and Computing, 53rd Annual Allerton Conference on*, 2015.
- [2] G. Ramnarayan, G. Ranade, and A. Sahai, “Side-information in control and estimation,” in *Information Theory (ISIT), IEEE International Symposium on.*, 2014.
- [3] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Sastry, “Kalman filtering with intermittent observations,” *Automatic Control, IEEE Transactions on.*, vol. 49, no. 9, pp. 1453–1464, 2004.

- [4] S. Park and A. Sahai, “Intermittent Kalman filtering: Eigenvalue cycles and nonuniform sampling,” in *American Control Conference (ACC)*, 2011, pp. 3692–3697.
- [5] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. S. Sastry, “Foundations of control and estimation over lossy networks,” *Proceedings of the IEEE*, vol. 95, no. 1, pp. 163–187, 2007.
- [6] N. Elia, “Remote stabilization over fading channels,” *Systems & Control Letters*, vol. 54, no. 3, pp. 237–249, 2005.
- [7] G. Ranade and A. Sahai, “Non-coherence in estimation and control,” in *Communication, Control, and Computing, 51st Annual Allerton Conference on*, 2013.
- [8] J. Ding, Y. Peres, and G. Ranade, “A tiger by the tail: When multiplicative noise stymies control,” in *2016 IEEE International Symposium on Information Theory (ISIT)*, July 2016, pp. 1994–1998.
- [9] G. Ranade and A. Sahai, “Control capacity,” in *International Symposium on Information Theory (ISIT)*, 2015.
- [10] —, “Control with side information,” in *Indian Control Conference (ICC)*. IEEE, 2016, pp. 405–411.
- [11] S. Flåm, “Nonanticipativity in stochastic programming,” *Journal of optimization theory and applications.*, vol. 46, no. 1, pp. 23–30, 1985.
- [12] N. C. Martins, M. A. Dahleh, and J. C. Doyle, “Fundamental limitations of disturbance attenuation in the presence of side information,” *IEEE Trans. Automatic Control*, vol. 52, no. 1, pp. 56–66, 2007.
- [13] P. Diaconis and M. Shahshahani, “The subgroup algorithm for generating uniform random variables,” *Probability in the Engineering and Informational Sciences*, vol. 1, no. 1, 1987.
- [14] T. Cover and J. Thomas, *Elements of information theory*. Wiley, 2006.
- [15] C. H. Bennett, “Notes on Landauer’s principle, reversible computation, and Maxwell’s Demon,” *Studies In History and Philosophy of Modern Physics*, vol. 34, no. 3, pp. 501–510, 2003.