Technical Notes and Correspondence

Remote Control Over Noisy Communication Channels: A First-Order Example

Sridevi V. Sarma and Munther A. Dahleh

Abstract—**In this note, we consider a set up in which the plant and controller are local to each other, but are together driven by a remote reference signal that is transmitted through a noisy discrete channel. Our goal is to design codeword lengths of block source and channel encoders, and a controller to meet a model matching performance objective. Such design problems are difficult in general, as there is a strong interplay between control objectives and communication constraints, which forces the synthesis of controllers and encoder-decoder pairs to be done simultaneously. Current approaches typically fix one, while the other is designed to meet some objective. We first construct a model matching performance metric that captures the tradeoffs between coding the reference command to achieve more accuracy at the remote site and designing a controller to meet performance. We then simultaneously synthesize the controller and encoder codeword lengths that meet the specified objective for a first-order plant and model case. Finally, we illustrate performance sensitivity to the poles of the plant and model, and to the channel noise.**

Index Terms—**Channel decoder, channel encoder, communication channel, remote control, source encoder.**

I. Introduction

Due to the information-rich world we live in today, control in distributed, asynchronous, networked environments is in demand. Communication links that have rate limitations, delays, and that are noisy, are now connected to control systems, and the interactions between the two cannot be ignored.

In this note, we consider the simple network, shown in Fig. 1, in which a plant and controller are both remote and separated from the reference command by a discrete communication channel. There is much work that focuses on the reconstruction of the reference command at the remote site (see [11] and the references therein). However, here we are interested in *driving* a remote system with the reconstructed command. Teleoperation systems, which often involve hazardous and unstructured tasks, can be addressed under this framework. Examples of such tasks include nuclear reactors, space applications, medical operations, and deep-sea and deep-space explorations [8]. Previous work involving communication constraints in teleoperation systems, mainly consider noiseless channels that simply add delays. When the delays are assumed to be constant, then the local and remote operators compute wave-variable transformations [9] and/or are delay compensators [1], which transform the channel into a passive connection, thereby ensuring stability under any delay. Smith predictors [5] at the local end are also applied to achieve performance. When the delays are assumed to be time-varying but bounded from above, then buffering techniques [6] can be applied. Finally, if the channel also erases some signals (e.g., packet losses), then one can develop a model of the channel using second-order statistics [2], or build an observer at the remote operator to reconstruct the data stream at the channel output [7] to maintain stability and performance.

Our approach views the local operator as a channel encoder, and the remote operator as a channel decoder (as defined in information theory [4]) to minimize errors between the actual and reconstructed command. We then parameterize all stabilizing controllers as a function of encoder and decoder parameters to ultimately design coding schemes and controllers to meet a specific performance objective. Therefore, our work combines information theory and robust control tools.

The performance metric that we construct illustrates the tradeoffs between sending the remote control system an accurate reference command, and designing a controller such that the remote system matches a given ideal transfer function. The longer the code lengths representing the input signal before it enters the channel, the more accurate the signal is that drives the remote control system. However, delays in receiving commands at the remote site negatively affect performance. We then simultaneously synthesize the controller and encoder lengths that meet specified model matching objectives, and the plant and ideal model are both first-order single-input–single-output (SISO) systems. In general, synthesis of each cannot be done separately due to the tight interplay between the communication link and control system. Finally, we illustrate performance sensitivity to the poles of the plant and model, and to the channel noise.

II. Problem Formulation

A. Setup

In this note, we ignore the feedback loop and only consider how the channel impacts performance in a feedforward setting. This makes sense when one is navigating "slowly." That is, a command is sent to a remote system when it is at rest and then it "moves" accordingly. Feedback as to where the system has moved to is given, but at a slower time-scale. Real-time feedback and navigation will be addressed in future work.

We consider the set up shown in Fig. 2, where

- $r \in \mathcal{C}_r$, where $\mathcal{C}_r$ is a given set of reference signals in l_p with finite covering $\{\mathcal{C}_{r,1}, \mathcal{C}_{r,2}, \ldots, \mathcal{C}_{r,L}\}$, i.e., $\cup_{i=1}^{L} \mathcal{C}_{r,i} = \mathcal{C}_r$, and a unique representative signal $r_j^* \in \mathcal{C}_{r,j}$ for each j [see Fig. 3(a)];
- $R : \mathcal{C}_r \to I_r$ maps a signal in $\mathcal{C}_{r,j}$ to the index $j \in I_r = \{1, 2, \ldots, L\}$ which represents the signal r_j^*;
- $SE : \{1, 2, \ldots, L\} \to \{0, 1\}^{\log_2(M)}$ is a source encoder that compresses information about the input signals $(M \leq L)$, i.e., SE defines a partition on the signal set $\{\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_M\}$ such that $\cup_{i=1}^{M} \mathcal{B}_i = \mathcal{C}_r$ and $\mathcal{B}_i \cap \mathcal{B}_j = \phi$ for all $i \neq j$, $i, j \in \{1, 2, \ldots, M\}$ [see Fig. 3(b)];
- $CE : \{0, 1\}^{\log_2(M)} \to \{0, 1, \ldots, k\}^N$ is a channel encoder that adds redundancy by mapping $\log_2(M)$ bits to N bits where $N \geq \log_2(M)$;
- C: is a discrete memoryless channel, that transmits one bit per time step, with input domain $X^N \in \{0, 1, \ldots, k\}^N$, range $W^N \in \{0, 1, \ldots, j\}^N$, and corresponding conditional probability distribution $P_N(\underline{w}|\underline{x})$;
- CD: is a channel decoder that maps $W^N \to \{0, 1\}^{\log_2(M)}$ to minimize the probability of decoding error, $P(\theta \neq \hat{\theta})$;

Manuscript received January 31, 2006; revised June 29, 2006. Recommended by Associate Editor L. Xie. This work was supported by the Air Force Office of Scientific Research under Grant 6892167.

The authors are with the Department of Electrical Engineering, the Massachusetts Institute of technology, Cambridge, MA 02139-4307 USA (e-mail: sree@mit.edu; dahleh@mit.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAC.2006.886539

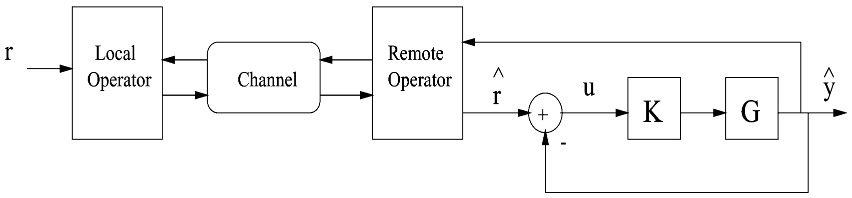

Fig. 1. Simple feed-forward network.

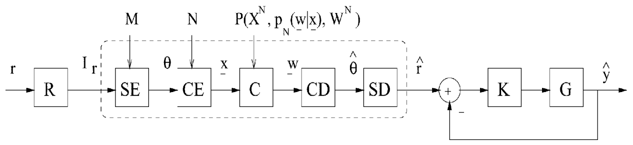

Fig. 2. Problem setup.

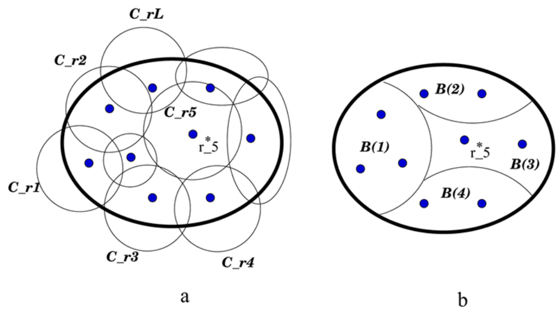

Fig. 3. Signal set, covering and representative signals, and source encoder partition.

- $SD : \{0,1\}^{\log_2(M)} \rightarrow \mathcal{C}_r$ first maps the estimate $\hat{\theta}$ to one of the M partitions $\mathcal{B}_i$ and then selects one of the reference signal centroids $r_j^* \in \mathcal{B}_i$ to send to the remote system[1];
- K is a causal, discrete-time, SISO controller;
- G is an unstable causal, discrete-time, SISO plant.

Note that G and C, are fixed, while SE, CE, and K are left for design. Both decoders are functions of the encoding schemes and are fixed once SE and CE are determined.

Before constructing a performance metric, we define two parameters that depend on the source encoding scheme and the set $\mathcal{C}_r$ which provide bounds on the error $\|r - \hat{r}\|_p$

1) $\beta_{\max}(\mathcal{C}_r) = \max_{i\in\{1,2,\ldots,M\}}, r_j \in \mathcal{C}_r \|r_j - r_i^*\|_p$;
2) $\beta_{\min}(\mathcal{C}_r, SE) = \max_{i\in\{1,2,\ldots,M\}} \max_{r_k\in\mathcal{B}_i} \|r_k - r_i^*\|_p$.

If the codeword (θ) for the index that represents the signal being sent is correctly decoded, i.e., $\theta = \hat{\theta}$, then the worst error between r and $\hat{r}$ is $\beta_{\min}(\mathcal{C}_r)$. On the other hand, if $\theta \neq \hat{\theta}$, then the worst error between r and $\hat{r}$ is $\beta_{\max}(\mathcal{C}_r)$.

To gain some insight into $\beta_{\min}$, we consider the following two extreme cases. If the channel is ideal with no noise, then source coding is not necessary, which is equivalent to $M = L$ and $SE = I$. In this case, $\beta_{\min} = 0$. If, on the other hand, the source encoder compresses all L signals into one "ball" or cover, *i.e.,* $M = 1$, then $\beta_{\min} = \beta_{\max}$. In general, $\beta_{\min}(\mathcal{C}_r, SE)$ is a function that monotonically decreases as M increases, and its shape depends entirely on the source encoder compression algorithm and the set of reference signals. Going forward, we suppress $\beta_{\min}$'s dependence on $\mathcal{C}_r$, and SE and $\beta_{\max}$'s dependence on $\mathcal{C}_r$, for an easier read.

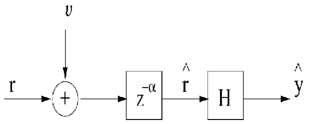

Fig. 4. Simpler setup.

B. A Simpler Setup

In this section, we present a simpler representation of the detailed set up described previously. It turns out, as we will see in Section II-C, that performance with respect to the channel in our original set up (Fig. 2) is equivalent to performance in the following simpler set up shown in Fig. 4.

In Fig. 4, we see that the reconstructed command, $\hat{r}$, is just a noisy delayed version of r. However, the noise and the delay are *not* independent of each other. The noise depends on the channel, the source resolution M, and the channel encoding length N, and the delay depends on M and N. Note also that $H = (I + GK)^{-1}GK$.

In this simple setup, we characterize the noise as follows. For a given channel, $\|\nu\|_p \leq \overline{\nu}$, where $\overline{\nu}$ is a random variable such that

$$\overline{\nu} = \begin{cases} \beta_{\max} & \text{Prob}(\theta \neq \hat{\theta}) \\ \beta_{\min} & \text{Prob}(\theta = \hat{\theta}). \end{cases}$$

$\text{Prob}(\theta \neq \hat{\theta})$ depends on N, M, and the channel as will be described in Section II-E. To obtain the above model of the noise's upper bound, we consider two scenarios. If the codeword for the index that represents the signal being sent is correctly decoded, i.e., $\theta = \hat{\theta}$, then the worst error between r and $\hat{r}$ is $\beta_{\min}$. On the other hand, if $\theta \neq \hat{\theta}$, then the worst error between r and $\hat{r}$ is $\beta_{\max}$.

The delay $\alpha = \log_2(M) + N$ is due to the delay in transmitting a total of $\log_2(M) + N$ bits through the channel, which has a rate of transmitting one bit per time step. We proceed with constructing our performance objective using this simpler representation.

C. Model Matching Performance Metric

In classical synthesis problems, we may be interested in designing K such that H is "close" to some given ideal model transfer function T. That is, we solve the following problem:

$$\begin{aligned} \min_{K} \quad & \|H - T\|_{p-ind} \\ \text{s.t.} \quad & H \text{ is stable.} \end{aligned}$$

Here, we consider the following modified problem that takes into account the communication link in our setup:

$$\begin{aligned} \min_{K} \quad & \max_{r\in\mathcal{C}_r} E\{d(\hat{y}, y_{\text{ideal}})\}(^*) \\ \text{s.t.} \quad & H \text{ is stable} \end{aligned}$$

where $d(\hat{y}, y_{\text{ideal}}) = \|H\hat{r} - Tr\|_p$. If we let $H_\alpha = z^{-\alpha}H$ (a delayed version of H), we then get that $d(\hat{y}, y_{\text{ideal}}) \triangleq \|H\hat{r} - Tr\|_p = \|H_\alpha(r + \nu) - Tr\|_p = \|(H_\alpha - T)r + H_\alpha \nu\|_p$, where ν is the noise signal. We

[1] We assume that the source decoder carries in its memory a bank of L reference signals in $\mathcal{C}_r$, and that it activates one of them when it receives $\hat{\theta}$.

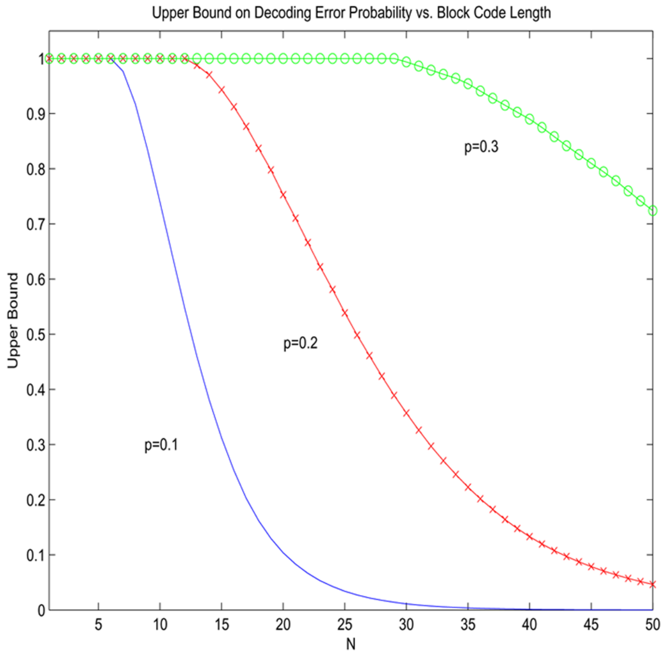

Fig. 5. Upper bound on probability of decoding error: BSC.

now take the expectation of the distance function d with respect to the noise upper bound $\overline{\nu}$, and get

$$\begin{aligned}E_{\overline{\nu}}(d) &= P(\theta = \hat{\theta})\|(H_\alpha - T)r + H_\alpha \nu_{\overline{\nu}=\beta_{\min}}\| \\ &\quad + P(\theta \neq \hat{\theta})\|(H_\alpha - T)r + H_\alpha \nu_{\overline{\nu}=\beta_{\max}}\| \\ &\leq P(\theta = \hat{\theta})\{\|(H_\alpha - T)r\| + \|H_\alpha\|\beta_{\min}\} \\ &\quad + P(\theta \neq \hat{\theta})\{\|(H_\alpha - T)r\| + \|H_\alpha\|\beta_{\max}\} \\ &\leq \|H_\alpha\|\{\beta_{\min} + (\beta_{\max} - \beta_{\min})P(\theta \neq \hat{\theta})\} \\ &\quad + \|(H_\alpha - T)\|\overline{r} \qquad (1)\end{aligned}$$

where $\overline{r} = \max_{i=1,\ldots,L} \|r_i^*\|_p$.

From [4, p. 135], we recall that an upper bound on the average probability of decoding error when one of M messages enter the channel encoder whose output is N bits and mapping arbitrary, $P(\theta \neq \hat{\theta})$, given any discrete memoryless channel is

$$\begin{aligned}P(\theta \neq \hat{\theta}) &\leq (M-1)^\rho \left\{ \frac{1}{k} \sum_{l=1}^{j} \left[\sum_{m=1}^{k} p(l|m)^{1/1+\rho} \right]^{1+\rho} \right\}^N \\ &\triangleq \overline{\omega}\end{aligned}$$

where $0 \leq \rho \leq 1$. We exhaustively search for ρ that minimizes this upper bound for our example described later. It is important to note that the upper bound decays quickly as N increases[2] when the channel has less noise. For example, we compute the bound for the binary symmetric channel (BSC) [4] in Fig. 5 for different noise levels, where p is the probability of an error.

Finally, we plug in the upper bound, $\overline{\omega}$, into inequality (1) and get

$$E_{\overline{\nu}}(d) \leq \|(H_\alpha - T)\| \cdot \overline{r} + \|H_\alpha\|\eta$$

where $\eta \triangleq \{\beta_{\min} + (\beta_{\max} - \beta_{\min})\overline{\omega}\}$.

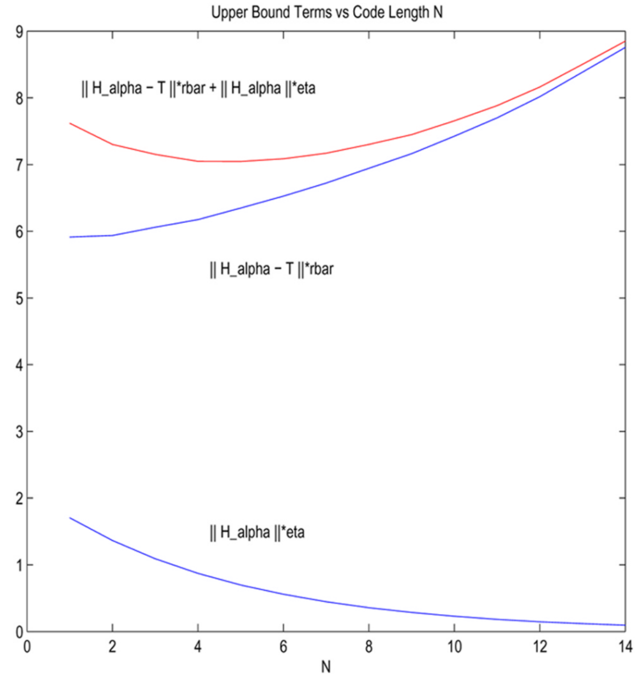

Fig. 6. Tradeoff of each cost component as N increases.

D. *Tradeoffs Between Communication and Control Objectives*

We now make some high-level observations on components of the upper-bound of $E_{\overline{\nu}}(d)$ computed previously as the code lengths vary

- $\|(H_\alpha - T)\| \cdot \overline{r}$: increases as a function of $N + \log_2(M)$;
- $\|H_\alpha\|\eta$: generally increases if $\log_2(M)$ increases and decreases if N increases.

Overall, if M is fixed ($\beta_{\min}$ is fixed) and N increases, the estimate of the reference signal improves, but the delay of the control system receiving $\hat{r}$ increases. Roughly speaking, $\|(H_\alpha - T)\| \cdot \overline{r}$ increases while $\|H_\alpha\|\eta$ decreases. See Fig. 6 for an example. On the other hand, if N is fixed and M increases, then the source encoder more accurately represents the input signals (less compression), but the probability of decoding error increases as there are more possible messages that can be sent through the channel. In addition, delay of the control system receiving $\hat{r}$ increases with $\log_2(M)$. Mathematically speaking, $\beta_{\min}$ decreases and hence η increases resulting in $\|H_\alpha\|\eta$ increasing, while $\|(H_\alpha - T)\| \cdot \overline{r}$ increases but at a slower logarithmic rate.

These observations indicate that there may exist a pair of code lengths M and N that will minimize the upper bound on the performance cost $E_{\overline{\nu}}(d)$. We set out to quantify these tradeoffs and optimal code lengths.

E. *Problem Statement*

In this section, we state questions that we are interested in answering for the aforementioned setup. We assume that the reference signals in $\mathcal{C}_r$ lie in l_2, and the output signals lie in l_∞. Thus, the induced norm between the input $r \in \mathcal{C}_r$, and output $\hat{y}$, is upper-bounded by the $\mathcal{H}_2$ norm of the network.

We observe that

$$\begin{aligned}E_{\overline{\nu}}(d) &\leq \|(H_\alpha - T)\|_{\mathcal{H}_2} \cdot \overline{r} + \|H_\alpha\|_{\mathcal{H}_2} \cdot \eta \\ &\leq \sqrt{2}\sqrt{\|(H_\alpha - T)\|_{\mathcal{H}_2}^2 \cdot \overline{r}^2 + \|H_\alpha\|_{\mathcal{H}_2}^2 \cdot \eta^2} \\ &= \sqrt{2}\, \| \left[(H_\alpha - T)\overline{r} \quad H_\alpha \eta \right] \|_{\mathcal{H}_2}.\end{aligned}$$

To get the second inequality shown previously, we let $|g_1| \triangleq \|(H_\alpha - T)\|_{\mathcal{H}_2} \cdot \overline{r}$, $|g_2| \triangleq \|H_\alpha\|_{\mathcal{H}_2} \cdot \eta$, and then use the fact that $|g_1| + |g_2| \leq \sqrt{2}\sqrt{|g_1|^2 + |g_2|^2}$.

[2] We note that $\overline{w}$ decreases as N increases only if the channel encoder rate, $(\log_2(M)/N)$, is less than the Shannon Capacity of the Channel, C [4]. The channel encoder rate is defined as the number of input symbols entering CE divided by the number of output symbols leaving CE

Now, instead of solving $(^{*})$, which, in general, is not easily computable for broad classes of encoders and channels, we seek to minimize the previous upper bound by solving the following problem:

$$\begin{aligned} \min_{K(M,N)} \quad & \sqrt{2}\,\|\,[\,(H_\alpha - T)\overline{r} \quad H_\alpha \eta\,]\,\|_{\mathcal{H}_2} (^{**}) \\ \text{s.t.} \quad & H \text{ is stable.} \end{aligned}$$

Note that if the channel is ideal ($\overline{w} = 0$), then no coding is necessary, which makes $\beta_{\min} = 0$, and therefore $\eta = 0$. The above cost function then reduces to the traditional model matching cost function.

1) Questions of Interest: Given a causal, unstable, DT, SISO plant, G, a causal, stable, DT, SISO ideal model, T, a discrete memoryless channel, C, and a decreasing function $\beta_{\min}(M)$.

1) Solve $(^{**})$ to synthesize a SISO LTI controller, K^o, as a function of (M, N).
2) Plug $K^o(M, N)$ back into the performance metric and find the code lengths, M and N, that minimize the cost function.
3) Describe the sensitivity of the optimal cost to the poles of the plant and ideal model, and to the channel noise.

III. First-Order Example

In this section, we consider the special case where

- $G(z) = (z/z - a)|a| > 1$;
- $T(z) = (z/z - \lambda)|\lambda| < 1$;
- C is a binary symmetric channel (BSC), shown in Fig. 10, with bit-flip probability p;
- $\beta_{\min}(M) = \begin{cases} \beta_{\max} = 1 & M = 1 \\ (1/\log_2(M)) & M \geq 2. \end{cases}$

A. Synthesis of Controller

To synthesize the controller as a function of the code lengths, we first parameterize the set of all stabilizing controllers of the remote system $H = (I + GK)^{-1}GK$ [3]. To do so, we first construct one observer-based controller by finding scalars f and l such that $a + af$ and $a + l$ are both stable (have magnitude inside the unit disk). We choose $f = -1$ and $l = -a$. Then, using the method and notation described in [3], we get the following coprime factorization over all stable proper rational functions of the plant G:

$$F(z) = 1 \quad J(z) = \frac{z - a}{z} \quad \tilde{Y}(z) = 1 \quad \tilde{X}(z) = \frac{a}{z}$$
where
$$G(z) = \frac{F(z)}{J(z)} \;\&\; F(z)\tilde{X}(z) + J(z)\tilde{Y}(z) = 1.$$

Then, all stabilizing controllers are of the form $((\tilde{X}(z) + J(z)Q(z))/(\tilde{Y}(z) - F(z)Q(z)))$, for $Q(z)$ being any proper rational stable function. This gives us the following closed-loop transfer function:

$$H(z) = F(z)\tilde{X}(z) + F(z)J(z)Q(z) = P(z) - U(z)Q(z)$$

where $P(z) \triangleq F(z)\tilde{X}(z)$, $U(z) \triangleq -F(z)J(z)$. The optimization problem $(^{**})$, using the parameterization of all stabilizing controllers, is then equivalent to solving

$$\begin{aligned} \min_{Q(M,N)} \quad & \sqrt{2}\|\overline{G} - S\|_{\mathcal{H}_2} \\ \text{s.t.} \quad & \{S \in (\overline{u}Q)\overline{V}| \\ & \qquad Q \text{ is a stable proper rational function}\} \\ \text{where} \quad \overline{G} = \; & [\,\overline{r}(z^{-\alpha}P - T) \quad \eta z^{-\alpha}P\,] \\ \overline{u} = \; & z^{-\alpha}U \quad \overline{V} = [\,\overline{r} \quad \eta\,]. \end{aligned}$$

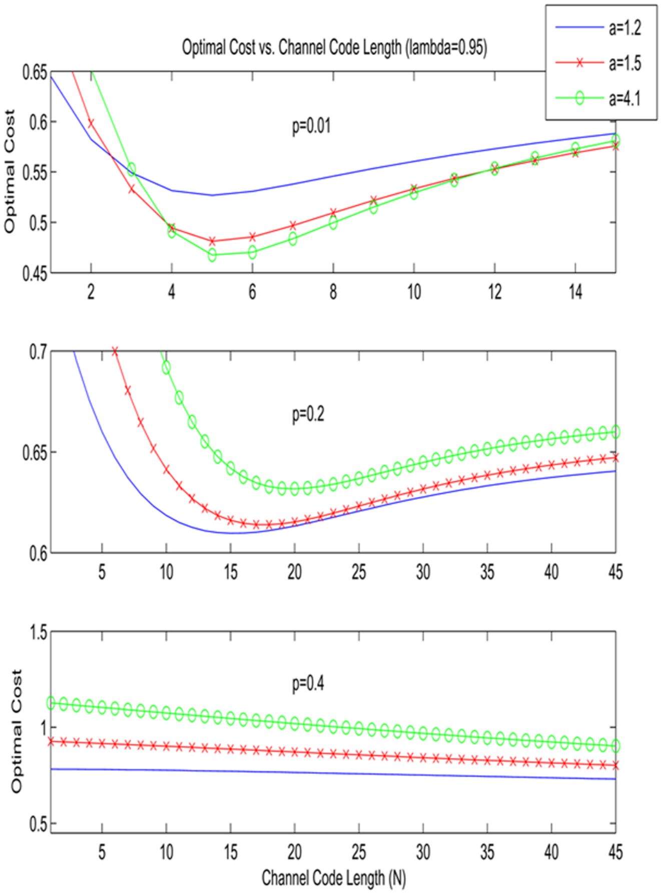

Fig. 7. Optimal cost versus channel code length (N): $(\lambda = 0.95)$.

Note that we suppress the z-dependence on z-transforms for a more compact notation (eg. $U = U(z)$).

Before solving for the optimal S^o, we recall that any stable proper rational function can be written as the product of an all-pass filter and a minimum-phase filter (see [3] for details). We can then factor $\overline{u}$ as follows:

$$\overline{u} = \overline{u}_{ap}\overline{u}_{mp} = \left\{ \frac{a - z}{(a - z^{-1})(z^{\alpha+1})} \right\} \{a - z^{-1}\}.$$

Finally, we define $S^o = \overline{u}Q^o\overline{V}$, and as shown in [3]

$$Q^o = \frac{(\overline{u}_{mp})^{-1}}{\overline{V}\overline{V}^*} \left[\overline{u}_{ap}^{-1}\overline{G}\right]_{\mathcal{H}_2} \overline{V}^*.$$

Recall that $[f]_{\mathcal{H}_2}$ is the projection of a function f onto the $\mathcal{H}_2$ subspace, and g^* denotes the complex-conjugate transpose of a complex-valued vector function g.

Note that $\overline{u}$ and $\overline{V}$ are functions of M and N, therefore, Q^o is also a function of M and N. Finally, the optimal controller, $K^o = (Q^o/1 - GQ^o)$, is a function of M and N. For our first-order example

$$H^o(M, N) = \frac{C1z}{az - 1} + \frac{C2z}{z - \lambda}$$

where $C_1 = (a - 1/a)\,[(a\gamma/k(a\lambda - 1)) + 1]$, $C_2 = (\gamma(\lambda - a)/k(a\lambda - 1))$, $k = \overline{r}^2 + \eta^2$, $\gamma = \lambda^\alpha \left[((a^2 - 1)\overline{r}^2/\lambda - a) + a\overline{r}^2\right]$.

B. Synthesis of Code Lengths

Now, that we have the optimal closed-loop transfer function as a function of code lengths, we look for the optimal (M^o, N^o) pair that

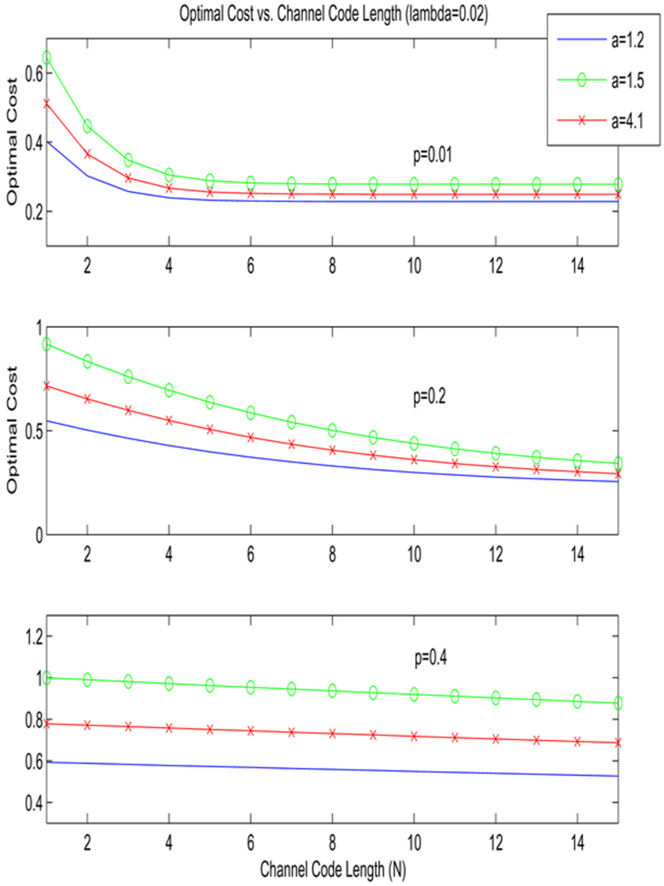

Fig. 8. Optimal cost versus channel code length (N): $(\lambda = 0.02)$.

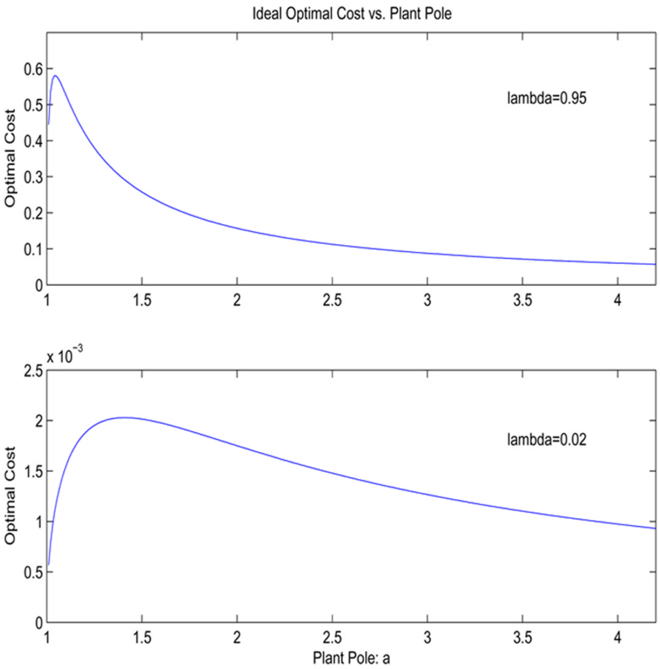

Fig. 9. Ideal optimal cost versus plant pole.

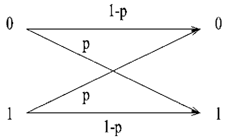

Fig. 10. Binary symmetric channel.

minimizes the cost function $\sqrt{2}\|\overline{G} - S\|_{\mathcal{H}_2}$, and the corresponding optimal closed-loop transfer function H^o. We set $a = 1.2$, $\lambda = 0.95$, $p = 0.01$, and $\overline{r} = 0.2$ and find that the minimum cost is 0.6194 and occurs when $M^o = 32$, and $N^o = 13$. The corresponding optimal control system is

$$H^o = \frac{0.0687z}{1.2z - 1} + \frac{0.1693z}{z - 0.95}.$$

C. *Performance Sensitivity*

In this section, we investigate the sensitivity of the optimal cost to the poles of the plant and model, and to the channel noise.

Fig. 7 illustrates how the channel code length, N, impacts the optimal cost for different unstable plant poles, when $\lambda = 0.95$ and the bit-flip probabilities of the BSC are 0.01, 0.2, and 0.4. Fig. 8 illustrates the optimal cost for different levels of channel noise when $\lambda = 0.12$. For these experiments, $\overline{r} = 0.2$, and $M = 1$, while N and a varied.

From Figs. 7 and 8, we make the following observations.

- *Sensitivity to plant pole*: The optimal code length increases as the magnitude of the unstable pole increases.
- *Sensitivity to channel noise*: As the channel noise increases, more coding is necessary to reach a minimum cost. However, for very noisy channels, the optimal code length is too long to be of use when implemented as the delay is too large. In such situations, recall that our upper bound on the cost, which was obtained using the upper bound on the probability of decoding error described in Section II-C, is not useful.
- *Sensitivity to ideal model pole*: The closer the model pole is to the unit disk, the more coding improves performance. That is, we see the tradeoffs between sending the remote system an accurate reference command and meeting performance.

D. *Ideal Solution*

In this section, we look at performance sensitivity of the ideal model matching problem (no channel or coding) by setting $\eta = \alpha = 0$. Fig. 9 illustrates how the optimal cost behaves as the plant pole becomes more unstable (as a ranges from 1.01 to 4.1) for different ideal model poles ($\lambda = 0.95$ and 0.02). As shown in Fig. 9, we see that when the ideal model pole is close to the unit disk, the ideal optimal cost $(\|H - T\|_{\mathcal{H}_2})$ is lowest when $a = 4.1$. This is consistent with what we see in Fig. 7 for very low channel noise $(p = 0.01)$, which shows that the optimal cost for $a = 4.1$ is lowest. However, when the ideal model pole is close to the origin, the ideal optimal cost is lowest when $a = 1.01$, which is also consistent with what we see in Fig. 8 for very low channel noise.

IV. Future Work

Future work entails making the set up more "real-time" and appropriate for navigation of the remote system. In this note, only one reference signal is sent through the link, and the system waits α time steps before it receives any signal. A more realistic set up includes switching the reference signal over time (as a function of real-time feedback signal), and allowing for $\hat{r}$ to reach the system without much delay, and to have it improve (be closer to r) over time as the input gets further coded. The latter scheme entails implementing a dynamic encoding scheme, much like that introduced in [10].

References

[1] R. J. Anderson and M. W. Spong, "Bilateral control of teleoperators with time delay," *IEEE Trans. Autom. Control*, vol. 34, no. 5, pp. 494–501, May 1989.

[2] J. C. Bolot, "End-to-end packet delay and loss behavior in the Internet," in *SIGCOMM*, Sep. 1993, pp. 289–298.

[3] Doyle, John, Francis, Bruce, Tannenbaum, and Allen*, Feedback Control Theory*. New York: MacMillan, c1992.

[4] Gallager and Robert*, Information Theory and Reliable Communication*. New York: Wiley, c1968.

[5] Ganjefar *et al.*, "Teleoperation systems design using augmented wave-variables and smith predictor method for reducing time-delay effects," in *Proc. IEEE Int. Symp. Intelligent Control*, Oct. 2002, pp. 333–338.

[6] R. Luck and A. Ray, "Experimental verification of a delay compensation algorithm for integrated communication and control systems," *Int. J. Control*, vol. 59, no. 6, pp. 1357–1372, 1994.

[7] R. Luck, A. Ray, and Y. Halevi, "Observability under recurrent loss of data," *AIAA J. Guid., Control, Dyna.*, vol. 15, pp. 284–287, 1992.

[8] C. Kitts, "Development and teleoperation of robotic vehicles," in *AIAA Unmanned Unlimited Systems, Technologies Operations Conf.*, San Diego, CA, 2003, AIAA 2003-6584.

[9] G. Niemeyer, "Using wave variables in time delayed force reflecting teleoperation," Ph.D. dissertation, MIT, Cambridge, MA, 1996.

[10] Sarma, Sridevi, Dahleh, A. Munther, Salapaka, and Srinivasa, "On time-varying bit-allocation maintaining stability: A convex parameterization," in *Proc. 43rd IEEE Conf. Decision and Control*, Dec. 2004, pp. 1430–1435.

[11] P. G. Voulgaris, C. Hadjicostis, and R. Touri, "A perfect reconstruction paradigm for digital communications," in *Proc. 42nd IEEE Conf. Decision and Control*, Dec. 2003, pp. 4196–4201.

A Remark on an Example By Teel–Hespanha With Applications to Cascaded Systems

Alessandro Astolfi

***Abstract*—The properties of a system, proposed by Teel and Hespanha, which is globally exponentially stable but with state that can be driven to infinity by an arbitrarily small exponentially decaying disturbance, are discussed in detail. These are used to propose a family of systems with a similar property and to argue that unstable behavior may be nongeneric and not detected by means of simulations. Finally, sufficient conditions for the existence of unbounded trajectories in cascaded systems are given.**

***Index Terms*—Cascaded systems, nonlinear systems, stability.**

I. Introduction

In [5], which builds upon the results in [4], the system (system (13) of [5])

$$\begin{aligned}\dot{x}_1 &= g(x_1x_2)x_1\\ \dot{x}_2 &= -2x_2+z\\ \dot{z} &= -z \end{aligned}\tag{1}$$

where $g(\cdot)$ satisfies Assumption 1 below, has been considered.

Assumption 1: [5, Ass. 1] The function $g(s)$ is such that

i) it is continuous;

ii) $|g(s)|\leq 1$ for all $s\in\mathbb{R}$;

iii) $g(s)=-1$ for all $s\in(-\infty,1/2]\cup[3/2,\infty)$;

iv) $g(1)=1$.

The authors have shown that if the initial condition $(x_1(0),x_2(0),z(0))$ is such that $x_1(0)\neq 0$, $x_2(0)=1/x_1(0)$ and $z(0)=x_2(0)$ then the resulting trajectory is such that $x_1(t)=e^tx_1(0)$, i.e., it is exponentially diverging. This property, together with the fact that the (x_1,x_2)-subsystem with $z=0$ is globally exponentially stable, is exploited to conclude that globally exponentially stable systems may be *destabilized* by arbitrarily small and exponentially decaying disturbances. This may not be a surprising fact, but the illuminating example in [5] shows that it may arise in very simple systems.

However, the aforementioned class of systems reserves further surprises. To begin with, one may be tempted to see the phenomenon highlighted in [5] by means of simulations. This yields the first surprise.

We run Matlab simulations using the ode45 command, setting the initial condition to (1, 1, 1) and selecting the function[1]

$$g(s)=\begin{cases}-1 & \text{if } s\leq\frac{1}{2}\\ 4s-3 & \text{if } \frac{1}{2}\leq s\leq 1\\ -4s+5 & \text{if } 1\leq s\leq\frac{3}{2}\\ -1 & \text{if } s\geq\frac{3}{2}\end{cases}$$

which is such that Assumption 1 holds. This yields the results in Fig. 1, displaying the time histories of the state $x_1(t)$ for different values of the "Relative Tolerance" Matlab internal variable (the "Absolute Tolerance" has been set to 1/100 the "Relative Tolerance"), together with the function e^t. Surprisingly, the state x_1 does not behave as theoretically forecast, but *follows* the signal e^t for an initial period of time (which depends upon the simulation parameters), and then converges exponentially to zero. At this point, after carefully rechecking all the calculations in [5], and after convincing ourselves that they are correct (as one would expect), we are left wondering why the forecast instability is not exposed by the simulation.

We devote the rest of this note to address this question and some related issues. In particular, we show that system (1) can be simply modified to display a family of unbounded trajectories, which are also *captured* by simulations, and we provide some sufficient conditions for the existence of unbounded trajectories in a class of cascaded systems. Note that the problem of existence of unbounded trajectories has also been studied, for general nonlinear systems, in [2].

II. A Class of Systems

Consider a system described by equations of the form

$$\begin{aligned}\dot{x}_1 &= g(x_1x_2)x_1\\ \dot{x}_2 &= -2x_2+z\\ \dot{z} &= -\lambda z \end{aligned}\tag{2}$$

where $g(\cdot)$ satisfies Assumption 1, and the following assumption.

Assumption 2: The function $g(s)$ is such that

i) $g(s)$ is C^1 for all $s\in(1/2,1)\cup(1,3/2)$;

ii) $g(s)=1\Rightarrow s=1$.

For system (2), the properties outlined in the following statement hold.

Proposition 1: Consider system (2) and assume the function $g(\cdot)$ satisfies Assumptions 1 and 2.

i) If $\lambda>1$, then all trajectories are bounded and converge exponentially to zero.

Manuscript received October 27, 2004; revised June 1, 2005, January 31, 2006, and July 26, 2006. Recommended by Associate Editor M.-Q. Xiao.

The author is with Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, U.K., and also with the Dipartimento di Informatica, Sistemi e Produzione, University of Rome "Tor Vergata," Rome 00133, Italy (e-mail: a.astolfi@imperial.ac.uk).

Digital Object Identifier 10.1109/TAC.2006.889870

[1] Similar conclusions can be drawn with other selections.