A Fundamental Need for Differentiated "Quality Of Service" Over Communication Links: An Information Theoretic Approach

Anant Sahai, Sanjoy Mitter *

Department of Electrical Engineering and Computer Science

Massachusetts Institute of Technology

Cambridge, Massachusetts 02139

{sahai,mitter}@lids.mit.edu

October 3, 2000

Abstract

We give a simple example of a vector estimation problem over a specific noisy communication channel with feedback. Assuming a layered architecture in which the reliable transmission of bits is separated from the encoding of the source into bits, we apply information theoretic bounds to show that it is impossible to estimate this source in a mean-squared sense if all the bits are treated alike by the reliable transmission layer. This is true no matter how large a finite end to end delay we permit or how we encode the source. However, if the reliable transmission layer "splits" the channel into two appropriate streams with differentiated "Quality of Service," it becomes possible to achieve finite end-to-end mean-squared error by sending appropriate bits over the two bit pipes. It is argued that this fundamental need for differentiated service is more general than this particular example.

1 Introduction

"A bit is a bit is a bit" seems to have become the motto of this digital age. Though we often take it for granted, the existence of such a common currency for information is far from obvious. While in practice it has been justified by the rapid advancement of digital electronics and the rise of a communications infrastructure to match, its meaningfulness rests upon the fundamental theorems of information theory.

Broadly speaking, there are two sides to information theory. The channel capacity theorems establish the possibility for reliable transmission of bits using only unreliable media. On the other side, the rate distortion theorems establish fundamental tradeoffs between fidelity to a source and the average length of the encoding in bits. These two sides of information theory are joined together by the information transmission theorems relating to source/channel separation.

*This research supported by U.S. Army Grant PAAL03-92-G-0115, Center for Intelligent Control Systems.

These information transmission theorems provide the philosophical justification for both a layered architecture and the practice of treating all bits alike when it comes to reliable transmission. However, in many real world applications, especially "multimedia" ones, it has been found that it is sometimes conceptually useful to completely break the abstraction layer and do joint source/channel coding. Another angle, explored in Tse's Thesis [12], is to consider sources which vary more slowly than the acceptable delay and to use an adaptive quantization approach which keeps end-to-end delay small while still giving good distortion-rate performance.

The flip idea of "loosely coupled joint source/channel coding" [4] suggests that the reliable transmission layer should treat bit streams differently based on their meanings. Communication links are then evaluated not just by their bit-rates, but by other "Quality of Service" parameters as well. Even across a single link, it has been found useful to provide differentiated service for bit streams being sent at the same time. [16, 15] In all these examples, this discrepancy between theory and practice is explained by an appeal to economics or the technical need for finite small end-to-end delay in applications. While generally information theory only applies in the limit of large delays, new results suggest that there is more to the need for QoS than just the issue of small delays.

We have recently found a new information transmission theorem for scalar unstable Markov processes in both the control [7] and estimation [9] contexts. This theorem shows that transmitting scalar unstable Markov processes needs more from a communication link than just a given bit-rate C. It requires that the α parameter (a close relative to the error exponent) at that bit rate also be high enough. Insofar as it is used alongside the rate to evaluate a channel, the parameter α in the α-any-time capacity thus has the appearance of a fundamental "Quality of Service" term.

In this paper, we justify the QoS interpretation by considering a unstable vector source and the problem of estimating it in a mean-squared sense across a specific binary erasure channel with feedback. After reviewing some definitions in section 2, we introduce the basic problem setup in section 3, and show that it is impossible to reliably transmit the source over the channel in a mean-squared sense if we insist on treating all bits alike in the reliable transmission layer. In section 4, we show that providing differentiated service (with different α parameters for different bit streams) can allow us to achieve finite end-to-end mean squared error. This example shows that we do not need to specify an end-to-end delay constraint or any other user-level QoS requirement *a priori.* The need for QoS requirements other than rate and hence differentiated service can emerge from the nature of the source and the distortion measure.

2 Review

2.1 Notions of Capacity

Definition 2.1 *The* Shannon classical capacity C *of a channel is the least upper bound of the rates at which the channel can be used to transmit data with an arbitrarily small probability of error. [3]*

$$C = \sup\{R|\forall\epsilon > 0 \exists N, \mathcal{E}_N, \mathcal{D}_N P_{error}(\mathcal{E}_N, \mathcal{D}_N) < \epsilon, Rate(\mathcal{E}_N, \mathcal{D}_N) = R\}$$

The encoder/decoder pair $\mathcal{E}_N, \mathcal{D}_N$ *has end-to-end delay less than or equal to* N.

The above is the operational meaning of Shannon capacity and should be distinguished from the many ways of calculating what it is for various channels. [14] The implicit meaning of "reliable transmission" in Shannon capacity is the ability to get a small probability of error on each of the bits being sent. Recently, we proposed a stronger sense of reliable transmission and its induced notion of capacity for a channel: [9]

Definition 2.2 *The* α-any-time capacity $C_{at}(\alpha)$ *of a channel is the least upper bound of the rates at which the channel can be used to transmit data with an arbitrarily small probability of error that decays with delay at least exponentially at a rate* α.

$$C_{at}(\alpha) = \sup\{R | \exists (K > 0, Rate(\mathcal{E}, \mathcal{D}^a) = R) \, \forall d > 0 \, P_{error}(\mathcal{E}, \mathcal{D}^a, Delay = d) \leq K 2^{-\alpha d}\}$$

This operational definition does not tell how to calculate α-any-time capacity. It is closely related to the traditional ideas of error exponents [3] but imposes a stronger restriction since it requires the encoder $\mathcal{E}$ to be the same for all end-to-end delays. This requirement arises because the bits to be transmitted are being generated in real-time as opposed to our transmitting a long message known in advance.

Viewing this from the decoder's perspective tells us why we are calling it any-time capacity. We can interrupt the decoding process for a given bit at "any time" and the resulting answer is increasingly meaningful the longer we wait. The α specifies the rate at which we want the answers to improve. The result is:

Lemma 2.1 *[8] Almost every bit is eventually decoded correctly. Let* $\tilde{S}_j^d = \hat{S}_j$ *when using delay* d. *For any* $\alpha > 0$, *if* $R < C_{at}(\alpha)$ *then* $\exists (\mathcal{E}, \mathcal{D}^a)$ *such that for all* $j > 0$ *the sequence* $[\tilde{S}_j^0, \tilde{S}_j^1, \ldots]$ *eventually converges to the correct value* S_j *and stays there with probability one.*

This property of getting every bit correct is analogous to the way in which TCP/IP provides reliable transmission over the Internet. [6]

2.2 Sources and Information Transmission

A Markov source $\{X_t\}$ with dynamics A and driven by noise $\{W_t\}$ is defined by the following:

$$X_{t+1} = AX_t + W_t \tag{1}$$

where A, X_t, and W_t are all of the appropriate dimensions for the recurrence to make sense. In the scalar case, they are all one dimensional.

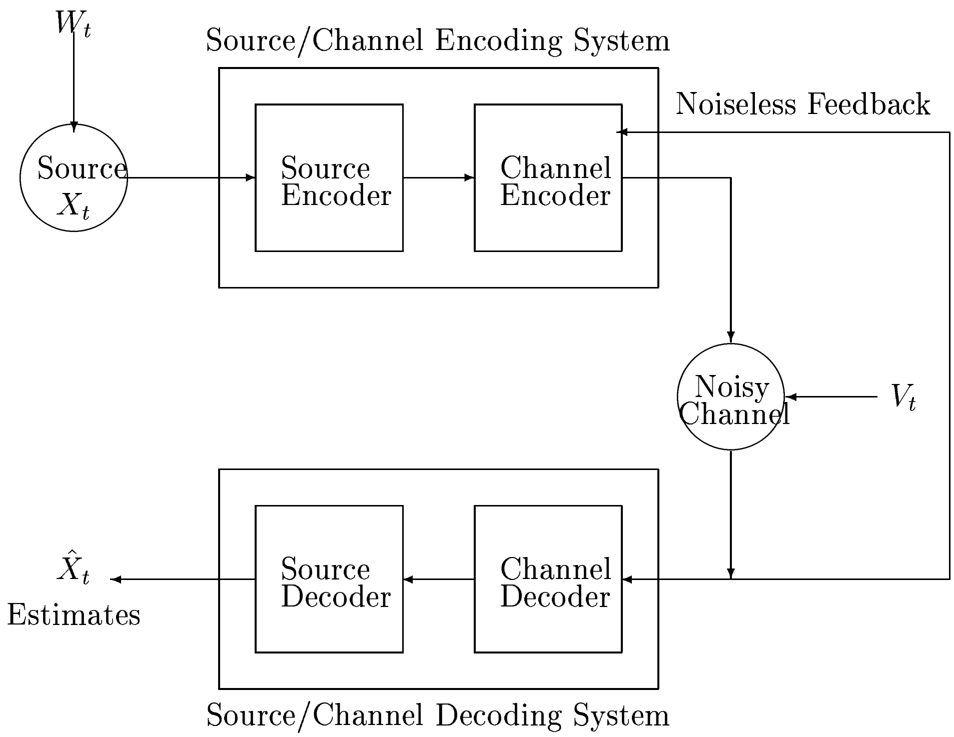
The natural problem of transmitting such a source across a noisy channel is the following: *Is it possible to design encoders and decoders within any specified end-to-end delay constraint so that the output of the decoder* $\{\hat{X}_t\}$ *achieves a desired mean-squared performance* $\sup_t E[\|\hat{X}_t - X_t\|^2] = K$*?* For the cases where A is a stable matrix (with all eigenvalues having magnitude less than 1), the problem has a well established complete solution [1] which can be naturally extended to a wide range of channels using the general source/channel separation results in [13].

In the case where the process is unstable (A having an eigenvalue with magnitude greater than 1), the first question is whether it is possible to get a finite K at all. After all, such unstable processes have asymptotically infinite variance. For the unstable scalar case with bounded noise, [9] provides the answer: (Proofs in [8])

Theorem 2.3 *If the source is scalar and driven by bounded noise, the source in (1) can be tracked with finite mean squared error across a noisy channel iff there is an* $\epsilon > 0$ *for which* $C_{at}(2\log_2 A + \epsilon) > \log_2 A$ *for the channel.*

3 A Simple Vector Problem

Consider a particular vector source and the problem of transmitting it with finite mean squared error across a binary erasure channel with noiseless feedback.



The specific vector source we consider has $X_t \in \Re^5$, a bound on the driving noise $\|W_t\|_\infty \leq \frac{\Omega}{2}$, and known initial condition $X_0 = 0$. It is defined by the matrix of dynamics

$$A = \begin{pmatrix} 1.178 & 0 & 0.04 & 0 & 0.04 \\ 1.08 & 1.058 & 0.36 & 0 & 0.36 \\ 0.36 & 0 & 1.178 & 0 & 0.12 \\ -0.12 & 0 & -0.04 & 1.058 & -0.04 \\ -0.12 & 0 & -0.04 & 0 & 1.018 \end{pmatrix} \quad (2)$$

In our simple example, we consider a binary erasure channel with erasure probability $e = 0.27$. The binary erasure channel models situations where errors can be reliably detected at the receiver. In the model, sometimes the single bit being sent does not make it through with probability e, but otherwise it makes it through correctly.

It can be shown that the Shannon classical capacity of this channel is $1 - e$ bits per channel use regardless of whether the encoder has feedback or not. [3] For the case where the encoder has access to instantaneous noiseless feedback, the anytime capacity can also be evaluated and is given by:

Theorem 3.1 *([7], Proof in [8]) For the binary erasure channel with noiseless feedback and probability of erasure e, let η range over $(0, \infty)$:*

$$C_{at}(\eta - \log_2(1 + (2^\eta - 1)e)) = 1 - \frac{1}{\eta}\log_2(1 + (2^\eta - 1)e)$$

3.1 Transform Coding

Looking at the source process in transformed coordinates is often of value in lossy coding. [5] In our case, it is illuminating to consider the transformed variable $\tilde{X}_t = TX_t$ where

$$T = \begin{pmatrix} 3 & 0 & 1 & 0 & 1 \\ 0 & 1 & -2 & 0 & 3 \\ 0 & 0 & 1 & 2 & 1 \\ 0 & 0 & 0 & 1 & -1 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

In that case, the transformed dynamics are given by a new $\tilde{A} = TAT^{-1}$ matrix:

$$\tilde{A} = \begin{pmatrix} 1.258 & 0 & 0 & 0 & 0 \\ 0 & 1.058 & 0 & 0 & 0 \\ 0 & 0 & 1.058 & 0 & 0 \\ 0 & 0 & 0 & 1.058 & 0 \\ 0 & 0 & 0 & 0 & 1.058 \end{pmatrix} \tag{3}$$

The initial condition is unchanged $\tilde{X}_0 = 0$ and the new driving noise $\tilde{W}_t$ has a new bound given by $\|\tilde{W}_t\|_\infty \leq 3\Omega$. It is also clear that having an encoder/decoder pair which achieves a finite mean squared error between the original $\{X_t\}$ and $\{\hat{X}_t\}$ processes is equivalent to having a pair which achieves a finite mean squared error between the transformed $\{\tilde{X}_t\}$ process and its corresponding transformed reconstruction.

Furthermore, being able to achieve a finite mean squared error with any finite end-to-end delay implies that we are able to achieve a finite mean squared error with no delay. This is because we could use an estimate for X_{t-d} to give an estimate for X_t by simply premultiplying it by A^d. After all,

$$\begin{aligned} E\left[\|X_t - A^d\hat{X}_{t-d}\|^2\right] &= E\left[\|(A^dX_{t-d} + \sum_{i=1}^{d} A^{i-1}W_{t-i}) - A^d\hat{X}_{t-d}\|^2\right] \\ &\leq E\left[\|A^d(X_{t-d} - \hat{X}_{t-d})\|^2\right] + E\left[\|\sum_{i=1}^{d} A^{i-1}W_{t-i}\|^2\right] + \\ & \quad 2E\left[\|A^d(X_{t-d} - \hat{X}_{t-d})\|\|(\sum_{i=1}^{d} A^{i-1}W_{t-i}\|\right] \end{aligned}$$

For any finite d, the three terms on the right hand side are all bounded for all t if we can achieve finite mean squared error for delay d. Hence the sum is as well.

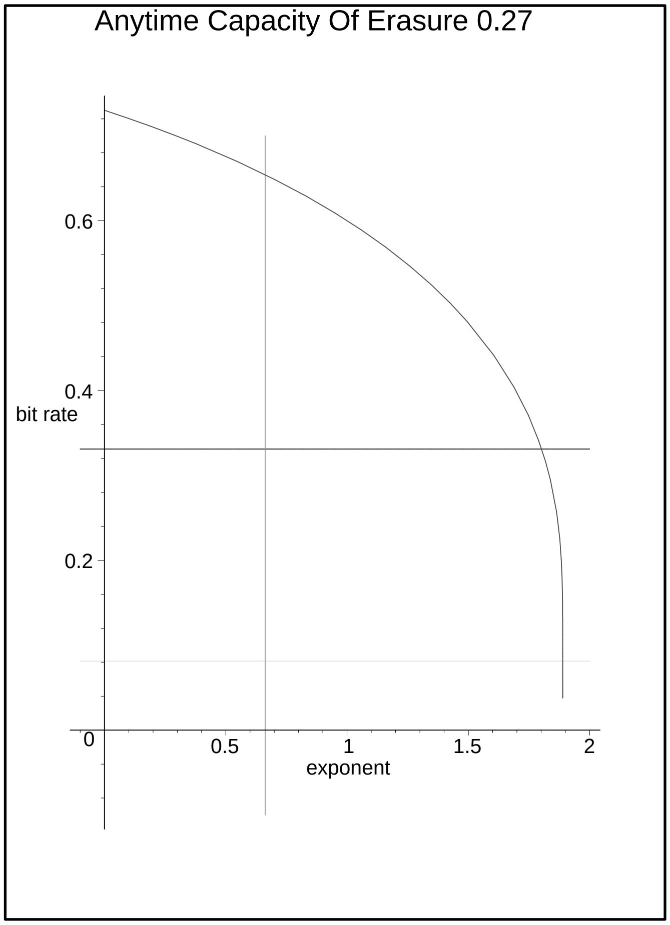
3.2 Fundamental Requirements

A necessary condition for achieving finite mean squared error for a vector valued process is getting a finite mean squared error for all the components. In the transformed domain, the dynamics of the components of our example are like those of five separate scalar unstable Markov sources. One of these is fast, and the four others are not as fast. Therefore, our Theorem 2.3 applies and we can easily conclude that we need the channel to satisfy both:

$$\begin{aligned} C_{\text{at}}(2\log_2(1.258)) &> \log_2(1.258) \\ C_{\text{at}}(2\log_2(1.058)) &> \log_2(1.058) \end{aligned}$$

Moreover, our work on sequential rate distortion [10, 11] tells us that the total rate must be larger than the the sum of the logs of the unstable eigenvalues, giving us the additional condition that:

$$C > \log_2(1.258) + 4\log_2(1.058)$$



It is easy to check that all these requirements are satisfied individually by the binary erasure channel with erasure probability $e = 0.27$. However as stated, these are only necessary conditions, not sufficient ones. In a sense, we need them all satisfied simultaneously for the different bitstreams in question.

3.3 Treating All Bits Alike

By using techniques from [8, 11], it is easy to construct recursive source codes for each of the five components of the transformed source. As long as we allow ourselves $R_1 > \log_2(1.258)$ bits per unit time for encoding the first component and $R_{2,3,4,5} > \log_2(1.1)$ bits per unit time time on the others, we can achieve finite mean squared error assuming no noise in transmitting these bits.

If we follow a strictly layered strategy and then require that all these bits be treated identically by the reliable transmission layer, we come up against a problem. For an erasure channel with $e = 0.27$, The maximum α for which the anytime capacity is larger than $(\log_2(1.258) + 4\log_2(1.058))$ is only around 0.646. This is enough for the four slow components which each require $\alpha > 2\log_2(1.058) \approx 0.163$. But it is less than the minimum α we require for the fast component: $\alpha > 2\log_2(1.258) \approx 0.662$. As long as we insist on treating all the bits alike in the reliable transmission layer, it is impossible to achieve a finite mean squared error on the first component of the transformed source. Because this is a consequence of our fundamental separation theorem for such processes, it is true regardless of how much end-to-end delay we are willing to tolerate or how we do the source encoding.

4 Differentiated Service

Consider an approach more in the spirit of "loosely coupled joint source/channel coding." [4] We get a higher priority for the bitstream representing the "faster" component requiring $\alpha > 2\log_2(1.258)$ by using the following strategy for reliable transmission:

- Store the incoming bits from the different streams into prioritized buffers — one buffer for each distinct priority level.
- At every opportunity for channel use, transmit the oldest bit from the highest priority buffer that is not empty.
- If the bit was received correctly, remove it from the appropriate buffer.

For our transformed source, we can use the scalar codes from [11, 8] to encode the first component at a rate $R_1 = \frac{1}{3} > \log_2(1.258)$ and assign this bitstream the higher priority. The first stream generates a bit every third time step. The other components are all encoded at a rate $R_{2,3,4,5} = \frac{1}{12} > \log_2(1.058)$ and gets the lower priority. These together also generates a bit every three steps (4 bits every 12 time steps). The total rate is therefore $R = \frac{2}{3} < 1 - e = 0.73$.

Since there is noiseless feedback and the incoming bitstreams are deterministic in their timing, the decoder can keep track of the encoder's buffer sizes and thus knows which incoming bit belongs to which stream. The source decoder takes all available bits and makes the best prediction of where it thinks the source process is.

4.1 The Source Codes

Our focus is not on optimizing performance and so we will just use a simple recursive source code [11, 8] which keeps the errors strictly bounded. An approach that asymptotically achieves optimal performance (rate-distortion bounds) can be found in [8]. Here, we simply apply the following theorem to each component of the transformed source.

Theorem 4.1 *[8] For all $\epsilon > 0$, every scalar discrete-time unstable linear Markov process with parameter $a > 1$ driven by bounded noise $-\frac{\Omega}{2} \leq W_t \leq \frac{\Omega}{2}$, can be tracked by an encoder F^R and decoder G^R with rate $R \leq \log_2 a + \epsilon$. Moreover, there exists a constant ν depending only on (a, Ω, R) such that $|X_t - \hat{X}_t| \leq \nu$ for all t.*

The source code works by encoding the predictive error signal $(X_{t+1} - a\hat{X}_t)$ and keeps it within a box of $[-\nu, \nu]$. Whenever the error signal threatens to leave that box, the encoder transmits a bit. Since each encoded bit cuts the uncertainty by a factor of 2, while the dynamics make it grow by a factor of a at each time, such a code can work for any $R > \log_2 a$. The closer R gets to that limit, the larger ν has to be. At the receiver's side, we use all the bits we have received to make our estimates. If we are missing some bits, we just predict by multiplying what we have by the appropriate power of that component's eigenvalue.

4.2 Analyzing The Channel Code

Notice that for any delay d time units, the channel decoder makes an error for bitstream i only if the encoder's buffer i contains more than dR_i bits still awaiting transmission. If there are less bits waiting, it means that the bit from d time units ago has already made it across. Thus, to analyze how the average probability of error varies with delay, we only need to study the steady-state distribution of the number of bits in each buffer.

4.2.1 The High Priority Stream

Since the highest priority stream preempts all lower priority streams, it effectively does not have to share the channel at all. So we can study its queue length using a simple Markov chain by blocking time into three time unit blocks. Then, the number of bits awaiting transmission at the end of a block is the Markov state with transition probabilities $p_{i,j}$ representing the probability that state i will go next to state j.

$$\begin{aligned}
p_{0,0} &= 3e^2(1-e) + 3e(1-e)^2 + (1-e)^3 \\
p_{i,i+1} &= e^3 \\
p_{i,i} &= 3e^2(1-e) \\
p_{i,i-1} &= \begin{cases} 3e(1-e)^2 + (1-e)^3 & \text{if } i = 1 \\ 3e(1-e)^2 & \text{if } i > 1 \end{cases} \\
p_{i,i-2} &= (1-e)^3
\end{aligned}$$

It is possible to calculate the steady state distribution π for this Markov chain. By some algebraic manipulation we can get the following recurrence relation:

$$(1-e)^3\pi_i = \begin{cases} (1-3e^2(1-e))\pi_{i-2} - 3e(1-e)^2\pi_{i-1} - e^3\pi_{i-3} & \text{if } i > 2 \\ (1-3e^2(1-e) - 3e(1-e)^2 - 3e^2(1-e))\pi_{i-2} - 3e(1-e)^2\pi_{i-1} & \text{if } i = 2 \end{cases}$$

It turns out that $\pi_i \propto \left(\frac{2e^3}{1+2e^3+(1-e)\sqrt{1+2e-3e^2}-3e^2}\right)^i$ as i gets large and thus:

$$\begin{aligned}
P_{\text{error}}(\text{Delay} = d) &\leq P(\text{Buffer State} > dR_1) \\
&\leq K\left(\frac{2e^3}{1+2e^3+(1-e)\sqrt{1+2e-3e^2}-3e^2}\right)^{\frac{d}{3}} \\
&= K2^{-\frac{1}{3}\log_2\left(\frac{1+2e^3+(1-e)\sqrt{1+2e-3e^2}-3e^2}{2e^3}\right)d}
\end{aligned}$$

which for $e = 0.27$ results in a fast enough $\alpha \approx 1.799 > 2\log_2(1.258) = 0.662$.

4.2.2 The Low Priority Streams

Notice that regardless of the realization of channel noise, the sum of the queue lengths for the two buffers is identical to the queue length for a hypothetical single stream at the combined rate. So it suffices to look at a single stream with rate $\frac{2}{3}$. Once again, we will group channel uses into threes so that the number of bits awaiting transmission at the end of a group is the Markov state for the system. This gives us:

$$\begin{aligned}
p_{0,0} &= 3e(1-e)^2 + (1-e)^3 \\
p_{i,i+2} &= e^3
\end{aligned}$$

$$
\begin{aligned}
p_{i,i+1} &= 3e^2(1-e) \\
p_{i,i} &= 3e(1-e)^2 \\
p_{i,i-1} &= (1-e)^3
\end{aligned}
$$

The steady state distribution π for the state in this case can be calculated just as before. By some algebraic manipulation we can get the following similar recurrence relation for it:

$$
(1-e)^3\pi_i = \begin{cases} (1-3e(1-e)^2)\pi_{i-1} - 3e^2(1-e)\pi_{i-2} - e^3\pi_{i-3} & \text{if } i > 2 \\ (1-3e(1-e)^2)\pi_{i-1} - 3e^2(1-e)\pi_{i-2} & \text{if } i = 2 \\ (1-3e(1-e)^2 - 3e^2(1-e))\pi_{i-1} & \text{if } i = 1 \end{cases}
$$

For large, i, we then have $\pi_i \propto \left(\frac{2e^2}{2e^2+\sqrt{4e-3e^2}-3e}\right)^i$. This rate of decay is clearly much slower than the one for the higher priority queue and thus, for large queue lengths, it dominates. Therefore we have for the bits in the lower priority queue:

$$
\begin{aligned}
P_{\text{error}}(\text{Delay} = d) &\leq P(\text{Combined Buffer State} > d(R_2 + R_3 + R_4 + R_5)) \\
&\leq K\left(\frac{2e^2}{2e^2 + \sqrt{4e - 3e^2} - 3e}\right)^{\frac{d}{3}} \\
&= K2^{-\frac{1}{3}\log_2\left(\frac{2e^2+\sqrt{4e-3e^2}-3e}{2e^2}\right)d}
\end{aligned}
$$

which for $e = 0.27$ results in an $\alpha \approx 0.285 > 2\log_2(1.058) \approx 0.163$ so our scheme is fast enough for all the slow components as well!

The direct part of Theorem 2.3 then shows that the cascaded source and channel codes achieve a finite mean squared error for all components of the transformed source. Thus, the system achieves a finite mean squared error on the original source as well.

5 Conclusion and Speculations

We have presented a simple example of a vector source and a specific channel for which it is impossible to achieve finite end-to-end mean-squared-error without using some form of differentiated service at the reliable transmission layer. Furthermore, in order to evaluate the bit-pipes provided by the reliable transmission layer, we have shown that the ideas of α-anytime-capacity are relevant and have used our previous separation results [9] to motivate a scheme which does achieve finite end-to-end performance.

It is hoped that this work goes some small distance towards meeting the challenge layed down by Ephremides and Hajek: [2]

> The interaction of source coding with network-induced delay cuts across the classical network layers and has to be better understood. The interplay between the distortion of the source output and the delay distortion introduced on the queue that this source output feeds into may hold the secret of a deeper connection between information theory.

This toy example with a single simple link opens the door towards a real Information Theory of "Quality of Service." Even though in the real world, few processes of interest actually have asymptotically infinite variance, we conjecture that these ideas will be a useful approximation whenever the upper limit of the tolerable end-to-end delay is within the range of time for which an unstable model is applicable. This may be a useful abstraction that underlies many "real-time" data sources. We suspect that this is true for not just control problems, but many others (including multimedia ones) as well.

References

[1] T. Berger, *Rate Distortion Theory.* Englewood Cliffs, NJ: Prentice-Hall, 1971.

[2] A. Ephremides and B. Hajek, "Information Theory and Communication Networks: An Unconsummated Union." IEEE Trans. on Information Theory, 44(6):2416–2434, 1998.

[3] R. Gallager, *Information Theory and Reliable Communication.* New York, NY: John Wiley and Sons, 1971.

[4] P. Haskell, D. G. Messerschmitt, and L. Yun, "Architectural Principles for Multimedia Networks," in *Wireless Communications: Signal Processing Perspectives*, H. V. Poor and G. W. Wornell, editors, Upper Saddle River, NJ: Prentice-Hall, 1998.

[5] A. K. Jain, *Fundamentals of Digital Image Processing*, Englewood Cliffs, NJ: Prentice-Hall, 1989.

[6] J. Postel, "Transmission Control Protocol", RFC 793, USC/Information Sciences Institute, September 1981 (http://www.ietf.org/rfc/rfc0793.txt)

[7] A. Sahai, "Evaluating Channels for Control: Capacity Reconsidered." Proc. 2000 American Control Conference, 2000.

[8] A. Sahai, "Any-time Information Theory." PhD Dissertation in progress.

[9] A. Sahai, ""Any-time" Capacity and A Separation Theorem For Tracking Unstable Processes," Proc. 2000 IEEE International Symposium on Information Theory, 2000.

[10] S. Tatikonda, A. Sahai, S. Mitter, "Control of LQG Systems Under Communication Constraints." Proc. 37th IEEE Conference on Decision and Control, 1998.

[11] S. Tatikonda, "Control Under Communication Constraints." PhD Dissertation, Massachusetts Institute of Technology, 2000.

[12] D. N. C. Tse, "Variable-rate Lossy Compression and its Effects on Communication Networks," PhD Dissertation, Massachusetts Institute of Technology, 1995.

[13] S. Vembu, S. Verdu, and Y. Steinberg, "The Source-Channel Separation Theorem Revisited." IEEE Trans. on Information Theory, 41(1):44–54, 1995.

[14] S. Verdu and T. S. Han, "A General Formula for Channel Capacity." IEEE Trans. on Information Theory, 40(4):1147–1157, 1994.

[15] X. Xiao and L. M. Ni, "Internet QoS: a big picture," IEEE Network, 13(2):8–18, 1999.

[16] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala, "RSVP: a new resource ReSerVation Protocol," IEEE Network, 7(5):8–18, 1993.