The Golden Ratio in Communication

...

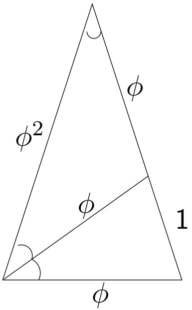
Blackwell's Trapdoor Channel

Task Assignment

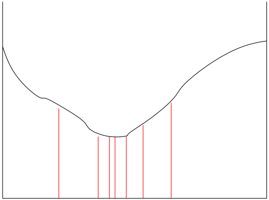
Paul Cuff
(Cover, Permuter, Weissman, Van Roy)

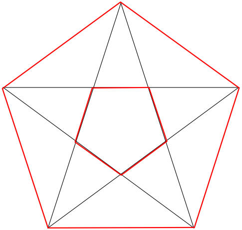
Stanford University

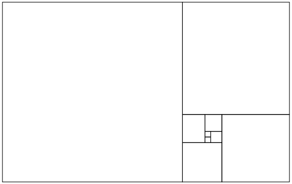
November 24, 2008



Golden Ratio

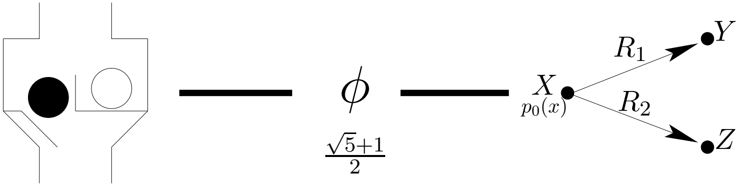






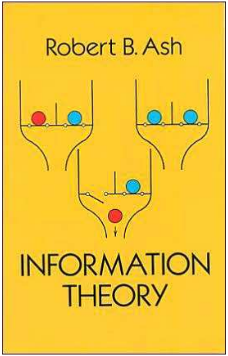
1,1,2,3,5,8,13,...

Outline - Markov Chain



The Trapdoor Channel

Introduced by David Blackwell in 1961.
[Ash 65], [Ahlswede & Kaspi 87], [Ahlswede 98], [Kobayashi 02 & 03].



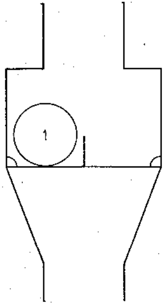
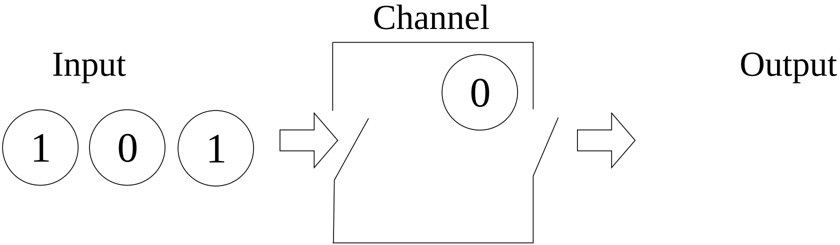


Fig. 7.1 A simple two-state channel.

A "simple two-state channel." - Blackwell

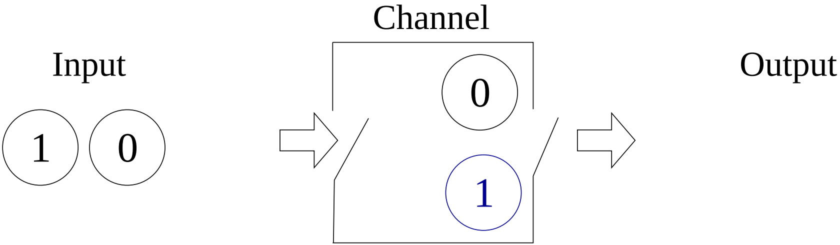
The Trapdoor Channel



$$s_t = s_{t-1} + x_t - y_t$$

$s_0 = 0$

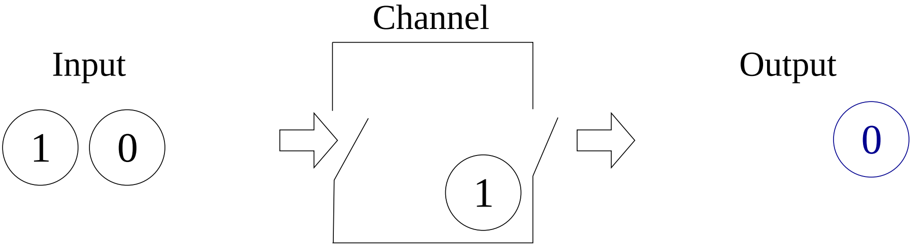
The Trapdoor Channel



$$s_t = s_{t-1} + x_t - y_t$$

$s_0 = 0$
$x_1 = 1,$

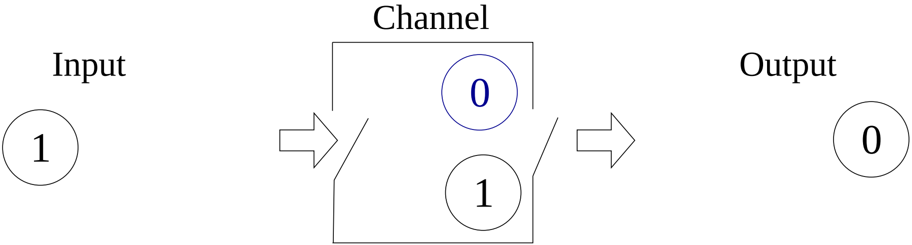
The Trapdoor Channel



$$s_t = s_{t-1} + x_t - y_t$$

$s_0 = 0$
$x_1 = 1,\ s_1 = 1,\ y_1 = 0,$

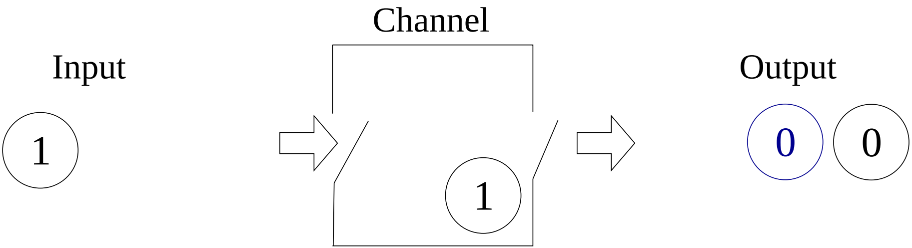
The Trapdoor Channel



$$s_t = s_{t-1} + x_t - y_t$$

$s_0 = 0$
$x_1 = 1,\ s_1 = 1,\ y_1 = 0,$
$x_2 = 0,$

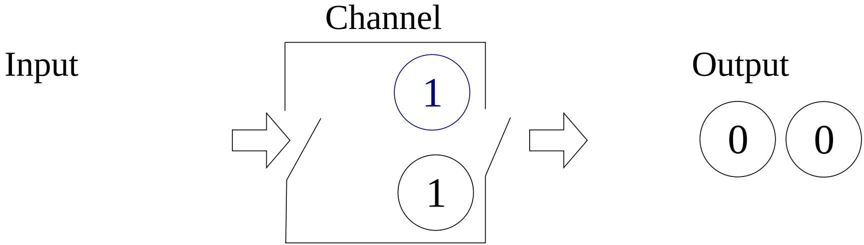
The Trapdoor Channel



$$s_t = s_{t-1} + x_t - y_t$$

$s_0 = 0$
$x_1 = 1,\ s_1 = 1,\ y_1 = 0,$
$x_2 = 0,\ s_2 = 1,\ y_2 = 0,$

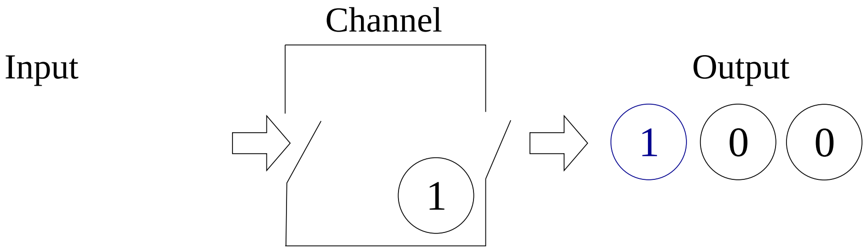
The Trapdoor Channel



$$s_t = s_{t-1} + x_t - y_t$$

$s_0 = 0$
$x_1 = 1,\ s_1 = 1,\ y_1 = 0,$
$x_2 = 0,\ s_2 = 1,\ y_2 = 0,$
$x_3 = 1,$

The Trapdoor Channel



$$s_t = s_{t-1} + x_t - y_t$$

$s_0 = 0$
$x_1 = 1,\ s_1 = 1,\ y_1 = 0,$
$x_2 = 0,\ s_2 = 1,\ y_2 = 0,$
$x_3 = 1,\ s_3 = 1,\ y_3 = 1.$

Communication without Feedback

- Repeat each bit three time: $R = 1/3$ bit.

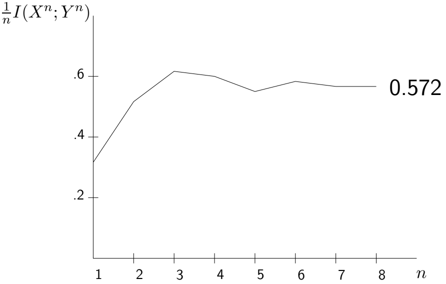
Communication without Feedback

- Repeat each bit three time: $R = 1/3$ bit.
- Repeat each bit twice: $R = 1/2$ bit. [Ahlswede & Kaspi 87]

Communication without Feedback

- Repeat each bit three time: $R = 1/3$ bit.
- Repeat each bit twice: $R = 1/2$ bit. [Ahlswede & Kaspi 87]
- $C \approx 0.572$ bits per channel use. [Kobayashi & Morita 03]

$$C = \lim_{n \to \infty} \max_{p(x_1,\ldots,x_n)} \frac{1}{n} I(X^n; Y^n).$$



Communication Setting (with feedback)

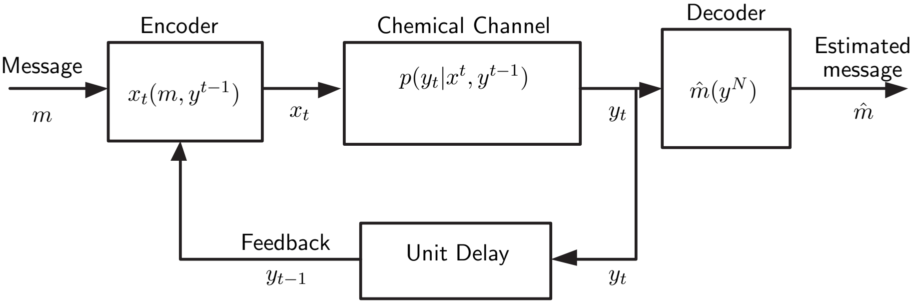


Figure: Communication with feedback

Feedback Capacity of Unifilar, Strongly Connected, FSC

- Capacity of the Trapdoor Channel

$$C = \lim_{n\to\infty} \frac{1}{n} \max_{p(x_1,\ldots,x_n)} I(X^n; Y^n)$$

- Feedback capacity of the Trapdoor Channel

$$C_{FB} = \lim_{n\to\infty} \frac{1}{n} \max_{\{p(x_i|x^{i-1},y^{i-1})\}_{i=1}^n} I(X^n \to Y^n)$$

[Permuter, Weissman & Goldmith ISIT06]

Directed Information

Mutual Information

$$I(X^n;Y^n) \;=\; \sum_{i=1}^{n} I(X^{n};Y_i|Y^{i-1})$$

Directed Information was defined by Massey in 1990.

$$I(X^n \to Y^n) \;\triangleq\; \sum_{i=1}^{n} I(X^{i};Y_i|Y^{i-1})$$

Directed Information

Mutual Information

$$I(X^n; Y^n) = \sum_{i=1}^{n} I(X^{n}; Y_i | Y^{i-1})$$

Directed Information was defined by Massey in 1990.

$$I(X^n \rightarrow Y^n) \triangleq \sum_{i=1}^{n} I(X^{i}; Y_i | Y^{i-1})$$

Intuition [Massey 05]:

$$I(X^n; Y^n) = I(X^n \rightarrow Y^n) + I(Y^{n-1} \rightarrow X^n)$$

Feedback Capacity of Unifilar, Strongly Connected, FSC

$$
\begin{aligned}
C_{FB} &= \lim_{N\to\infty} \frac{1}{N} \max_{\{p(x_t|x^{t-1},y^{t-1})\}_{t=1}^N} I(X^N \to Y^N) \\
&= \lim_{N\to\infty} \frac{1}{N} \max_{\{p(x_t|x^{t-1},y^{t-1})\}_{t=1}^N} \sum_{t=1}^N I(X^t; Y_t|Y^{t-1}) \\
&= \lim_{N\to\infty} \frac{1}{N} \max_{\{p(x_t|s_{t-1},y^{t-1})\}_{t=1}^N} \sum_{t=1}^N I(X_t, S_{t-1}; Y_t|Y^{t-1}) \\
&= \sup_{\{p(x_t|s_{t-1},y^{t-1})\}_{t\geq 1}} \liminf_{N\to\infty} \frac{1}{N} \sum_{t=1}^N I(X_t, S_{t-1}; Y_t|Y^{t-1})
\end{aligned}
$$

Dynamic Programming (infinite horizon, average reward)

Variable Assignments

State: $\beta_t = p(s_t|y^t)$

Action: $u_t = p(x_t|s_{t-1})$

Disturbance: $w_t = y_{t-1}$

Dynamic Programming Requirements

State evolution:

$$\beta_t = F(\beta_{t-1}, u_t, w_t)$$

Reward function per unit time:

$$g(\beta_{t-1}, u_t) = I(X_t, S_{t-1}; Y_t|\beta_{t-1})$$

Similar dynamic programming approaches:

- Yang, Kavčić & Tatikonda - 2005
- Chen & Berger - 2005
- Mitter & Tatikonda - 2006

Dynamic Programming (infinite horizon, average reward)

Dynamic Programming Operator T

The dynamic programming operator T is given by

$$T \circ J(\beta) = \sup_{u \in \mathcal{U}} \left(g(\beta, u) + \int P_w(dw|\beta, u) J(F(\beta, u, w)) \right).$$

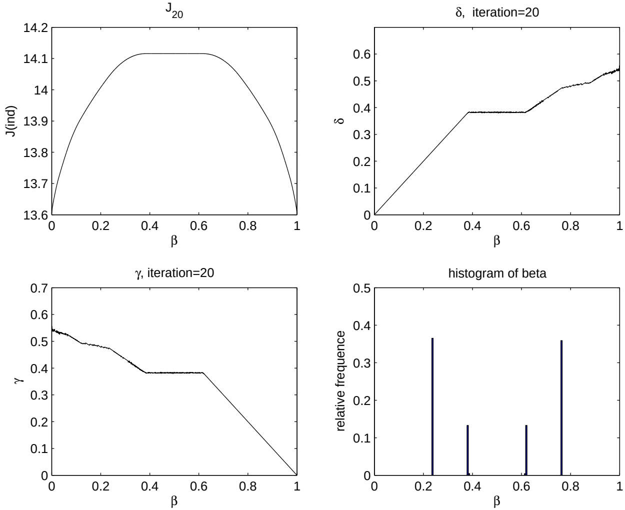
Bellman Equation

If there exist a function $J(\beta)$ and constant ρ that satisfy

$$J(\beta) = T \circ J(\beta) - \rho$$

then ρ is the optimal infinite horizon average reward.

Trapdoor Channel—20th Value iteration



Calculation and Coincidence

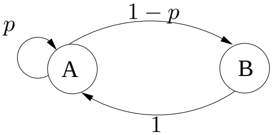
Trapdoor channel: $C_{FB} \approx 0.694$ bits

Calculation and Coincidence

Trapdoor channel: $C_{FB} \approx 0.694$ bits

Homework Question

Entropy rate. Find the maximum entropy rate of the following two-state Markov chain:

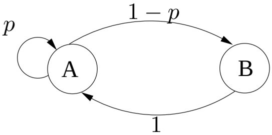


Calculation and Coincidence

Trapdoor channel: $C_{FB} \approx 0.694$ bits

Homework Question

Entropy rate. Find the maximum entropy rate of the following two-state Markov chain:



Solution: (Golden Ratio: $\phi = \frac{\sqrt{5}+1}{2}$)

$p^\star = \phi - 1 = \frac{1}{\phi}$

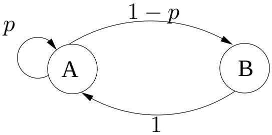
$H(\mathcal{X}) = \log \phi = 0.6942...$ bits

Calculation and Coincidence

Trapdoor channel: $C_{FB} \approx 0.694$ bits

Homework Question

Entropy rate. Find the maximum entropy rate of the following two-state Markov chain:



Solution: (Golden Ratio: $\phi = \frac{\sqrt{5}+1}{2}$)

$p^{\star} = \phi - 1 = \frac{1}{\phi}$

$H(\mathcal{X}) = \log \phi = 0.6942...$ bits

Equivalent Channel Formulation

Rename channel input: $\tilde{x}_t = x_t \oplus s_{t-1}$.
Rename channel output: $\tilde{y}_t = y_t \oplus y_{t-1}$.

Equivalent Channel Formulation

Rename channel input: $\tilde{x}_t = x_t \oplus s_{t-1}$.
Rename channel output: $\tilde{y}_t = y_t \oplus y_{t-1}$.

Case 1: $\tilde{x}_t = 0$

$$\tilde{x}_t = 0 \quad \Rightarrow \quad \tilde{y}_t = \tilde{x}_{t-1}.$$

Equivalent Channel Formulation

Rename channel input: $\tilde{x}_t = x_t \oplus s_{t-1}$.
Rename channel output: $\tilde{y}_t = y_t \oplus y_{t-1}$.

Case 1: $\tilde{x}_t = 0$

$$\tilde{x}_t = 0 \quad \Rightarrow \quad \tilde{y}_t = \tilde{x}_{t-1}.$$

Proof: $x_t = s_{t-1} = y_t = s_t$

Equivalent Channel Formulation

Rename channel input: $\tilde{x}_t = x_t \oplus s_{t-1}$.
Rename channel output: $\tilde{y}_t = y_t \oplus y_{t-1}$.

Case 1: $\tilde{x}_t = 0$

$$\tilde{x}_t = 0 \quad \Rightarrow \quad \tilde{y}_t = \tilde{x}_{t-1}.$$

Proof: $x_t = s_{t-1} = y_t = s_t$

$$x_{t-1} \oplus s_{t-2} \quad = \quad y_{t-1} \oplus s_{t-1}$$

Equivalent Channel Formulation

Rename channel input: $\tilde{x}_t = x_t \oplus s_{t-1}$.
Rename channel output: $\tilde{y}_t = y_t \oplus y_{t-1}$.

Case 1: $\tilde{x}_t = 0$

$\tilde{x}_t = 0 \quad \Rightarrow \quad \tilde{y}_t = \tilde{x}_{t-1}.$

Proof: $x_t = s_{t-1} = y_t = s_t$

$$\begin{aligned} x_{t-1} \oplus s_{t-2} &= y_{t-1} \oplus s_{t-1} \\ x_{t-1} \oplus s_{t-2} &= y_{t-1} \oplus y_t \end{aligned}$$

Equivalent Channel Formulation

Rename channel input: $\tilde{x}_t = x_t \oplus s_{t-1}$.
Rename channel output: $\tilde{y}_t = y_t \oplus y_{t-1}$.

Case 1: $\tilde{x}_t = 0$

$$\tilde{x}_t = 0 \quad \Rightarrow \quad \tilde{y}_t = \tilde{x}_{t-1}.$$

Proof: $x_t = s_{t-1} = y_t = s_t$

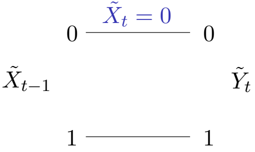
$$\begin{aligned} x_{t-1} \oplus s_{t-2} &= y_{t-1} \oplus s_{t-1} \\ x_{t-1} \oplus s_{t-2} &= y_{t-1} \oplus y_t \end{aligned}$$

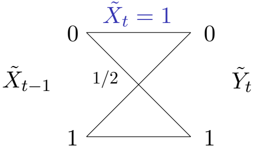
Case 2: $\tilde{x}_t = 1$

$\tilde{x}_t = 1 \quad \Rightarrow \quad \tilde{y}_t \sim$ Bern(1/2), independent of the past.

Equivalent Channel

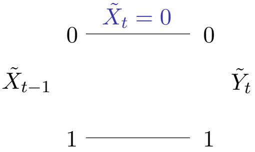
- Two states specify whether the channel is all noise or noise-free.



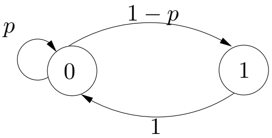


Equivalent Channel

- Two states specify whether the channel is all noise or noise-free.



- Use Markov input



Decoding Example

- Transmitter promises to never use $\tilde{x} = 1$ twice in a row.
- The number of such sequences of length n is the Fibonacci sequence.

Decoding rules

$$\tilde{x}_t = 0 \quad \Rightarrow \quad \tilde{x}_{t-1} = \tilde{y}_t$$
$$\tilde{x}_t = 1 \quad \Rightarrow \quad \tilde{x}_{t-1} = 0$$

$\tilde{x}^n$: 0
$\tilde{y}^n$: 1 1 1 1 0 0 1

Decoding Example

- Transmitter promises to never use $\tilde{x} = 1$ twice in a row.
- The number of such sequences of length n is the Fibonacci sequence.

Decoding rules

$$\tilde{x}_t = 0 \quad \Rightarrow \quad \tilde{x}_{t-1} = \tilde{y}_t$$

$$\tilde{x}_t = 1 \quad \Rightarrow \quad \tilde{x}_{t-1} = 0$$

$\tilde{x}^n$: 0 0

$\tilde{y}^n$: 1 1 1 1 0 0 1

Decoding Example

- Transmitter promises to never use $\tilde{x} = 1$ twice in a row.
- The number of such sequences of length n is the Fibonacci sequence.

Decoding rules

$$\tilde{x}_t = 0 \quad \Rightarrow \quad \tilde{x}_{t-1} = \tilde{y}_t$$

$$\tilde{x}_t = 1 \quad \Rightarrow \quad \tilde{x}_{t-1} = 0$$

$\tilde{x}^n$: 1 0 0
$\tilde{y}^n$: 1 1 1 1 0 0 1

Decoding Example

- Transmitter promises to never use $\tilde{x} = 1$ twice in a row.
- The number of such sequences of length n is the Fibonacci sequence.

Decoding rules

$$\tilde{x}_t = 0 \quad \Rightarrow \quad \tilde{x}_{t-1} = \tilde{y}_t$$

$$\tilde{x}_t = 1 \quad \Rightarrow \quad \tilde{x}_{t-1} = 0$$

$\tilde{x}^n$: 0 1 0 0

$\tilde{y}^n$: 1 1 1 1 0 0 1

Decoding Example

- Transmitter promises to never use $\tilde{x} = 1$ twice in a row.
- The number of such sequences of length n is the Fibonacci sequence.

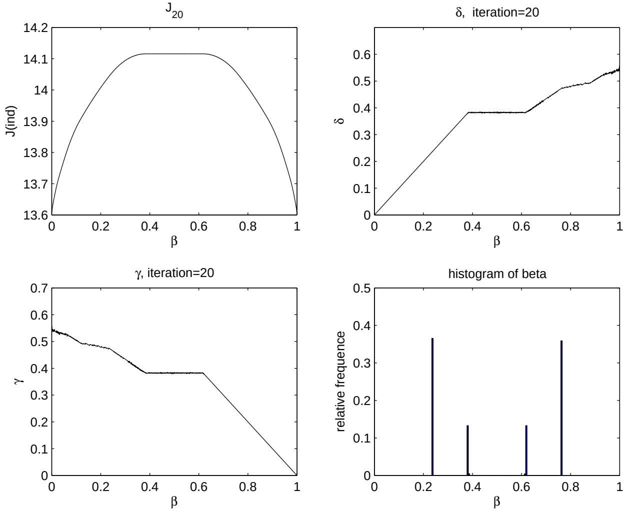
Decoding rules

$$\tilde{x}_t = 0 \quad \Rightarrow \quad \tilde{x}_{t-1} = \tilde{y}_t$$

$$\tilde{x}_t = 1 \quad \Rightarrow \quad \tilde{x}_{t-1} = 0$$

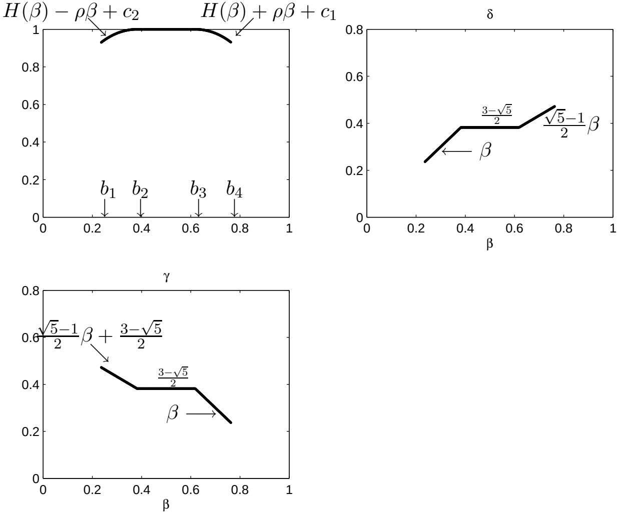
$\tilde{x}^n$: 1 0 1 0 0
$\tilde{y}^n$: 1 1 1 1 0 0 1

Dynamic Programming 20th Value iteration



Bellman Equation: $J(\beta) = T \circ J(\beta) - \rho$.

Conjectured Solution to the Bellman Equation



Bellman Equation: $J(\beta) = T \circ J(\beta) - \rho$.

Feedback Capacity and Zero-Error Capacity

Bellman Equation is satisfied.

Trapdoor channel feedback capacity:

$$
\begin{aligned}
C_{FB} &= \log \phi = 0.6942... \text{ bits.} \\
C &\approx .572 \text{ bits.}
\end{aligned}
$$

Feedback Capacity and Zero-Error Capacity

Bellman Equation is satisfied.

Trapdoor channel feedback capacity:

$$\begin{aligned} C_{FB} &= \log\phi = 0.6942... \text{ bits.} \\ C &\approx .572 \text{ bits.} \end{aligned}$$

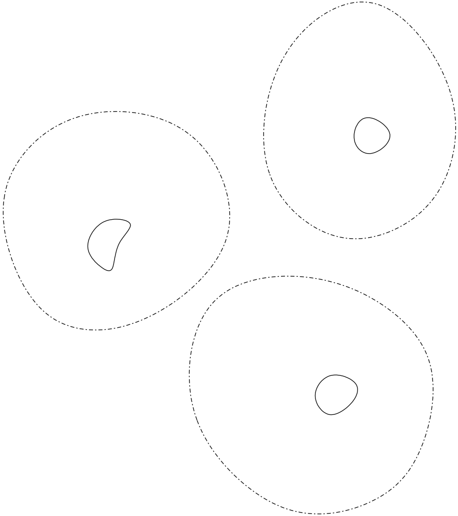
Trapdoor channel zero-error capacity:

$$\begin{aligned} C_{FB} &= \log\phi. \\ C &= .5 \text{ bits.} \end{aligned}$$

[Ahlswede & Kaspi 87]

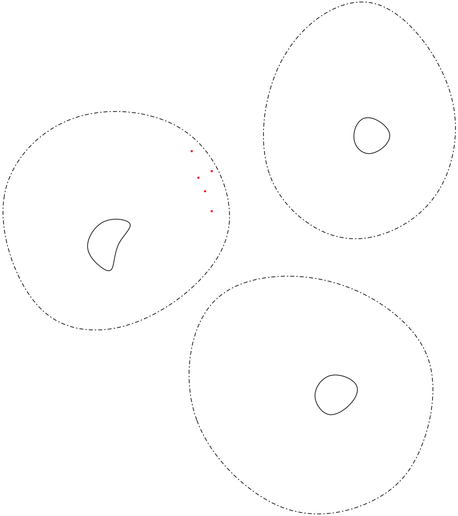
Biological Cells

Biochemical interpretation of the trapdoor channel [Berger 71]



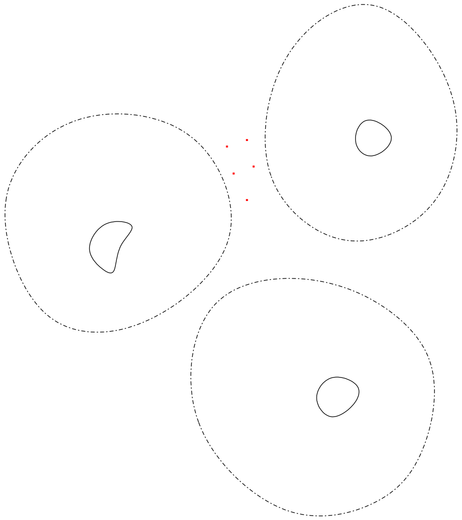
Biological Cells

Biochemical interpretation of the trapdoor channel [Berger 71]



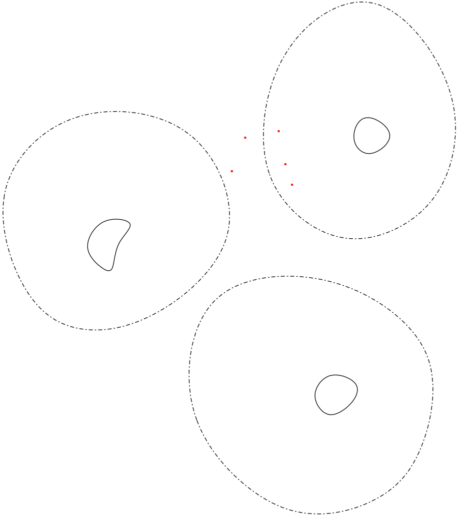
Biological Cells

Biochemical interpretation of the trapdoor channel [Berger 71]

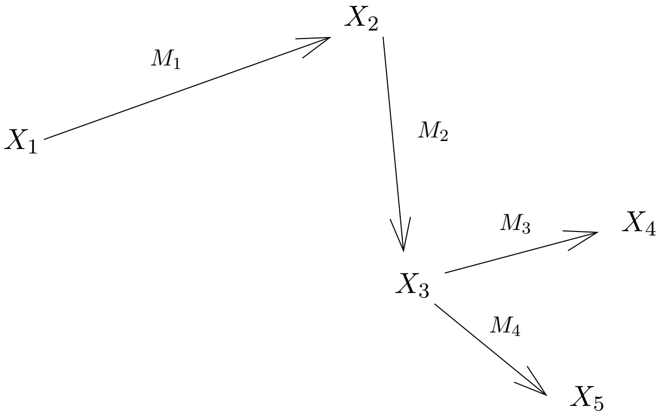


Biological Cells

Biochemical interpretation of the trapdoor channel [Berger 71]



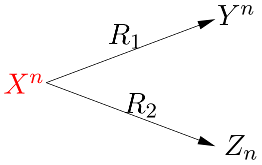
Coordinated Action in a Network



Task Assignment

Computation Tasks: $T = \{0, 1, 2\}$
Processors: X, Y, and Z

$X^n \sim iid, \;\; Unif(\{0, 1, 2\})$

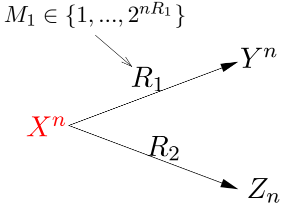


What are the required rates (R_1, R_2)?

Task Assignment

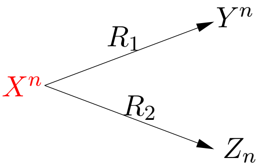
Computation Tasks: $T = \{0, 1, 2\}$
Processors: X, Y, and Z

$X^n \sim iid, \;\; Unif(\{0, 1, 2\})$



What are the required rates (R_1, R_2)?

Task Assignment: Achievable Rates

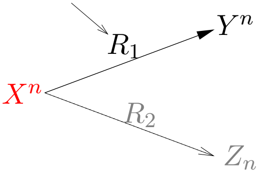


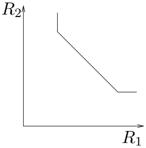
Idea 1—Describe X^n:

$$R_1 = R_2 = H(X) = \log(3).$$

A protocol tells Y and Z how to orient around X.

Task Assignment: Achievable Rates



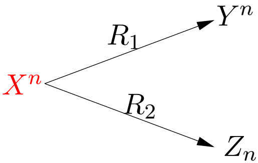


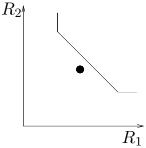
Idea 2—Minimize R_1:

$$
\begin{aligned}
R_1 &\geq \min_{p(y|x):X\neq Y} I(X;Y) \\
&= H(X) - \max_{p(y|x):X\neq Y} H(Y|X) \\
&= \log 3 - \log 2.
\end{aligned}
$$

Describe Z^n at full rate: $R_2 = \log 3$.

Task Assignment: Achievable Rates





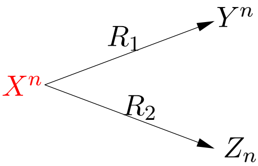
Idea 3—Restrict the support of Y and Z:

$$\begin{aligned} Y &\in \{0,1\} \\ Z &\in \{0,2\}. \end{aligned}$$

By default $Y = 1$ and $Z = 2$.
The encoder tells them when to move out of the way.

$$\begin{aligned} R_1 &\geq H(Y) = H(1/3) = \log 3 - 2/3 \text{ bits}, \\ R_2 &\geq H(Z) = H(1/3) = \log 3 - 2/3 \text{ bits}. \end{aligned}$$

Task Assignment: Achievable Rates



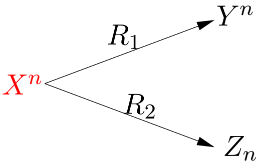
Achievable rates for general joint distribution $p(x, y, z)$*:*

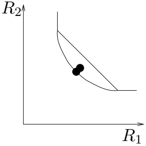
$$
\begin{aligned}
R_1 &\geq I(X; Y, U), \\
R_2 &\geq I(X; Z, U), \\
R_1 + R_2 &\geq I(X; Y, U) + I(X; Z, U) + I(Y; Z|X, U).
\end{aligned}
$$

for some U.

[Cover, Permuter 07], [Zhang, Berger 95]

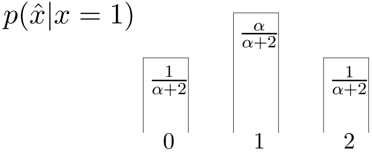
Task Assignment: Achievable Rates



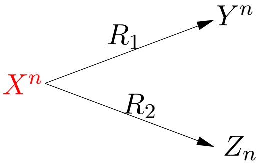


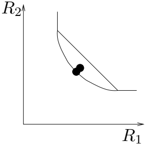
Idea 4—Restrict the support of Y and Z around $\hat{X}$:

First send $\hat{X}$ to both Y and Z, where $\hat{X} = X$ with probability $\frac{\alpha}{\alpha+2}$.



Task Assignment: Achievable Rates





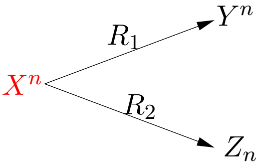
Idea 4—Restrict the support of Y and Z around $\hat{X}$:

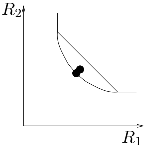
$$\begin{aligned} Y &\in \{\hat{X}, \hat{X}+1\} \\ Z &\in \{\hat{X}, \hat{X}+2\}. \end{aligned}$$

The encoder tells them when to move out of the way (onto $\hat{X}$).

$$\begin{aligned} R_1 &\geq I(X;\hat{X}) + H(Y|\hat{X}), \\ R_2 &\geq I(X;\hat{X}) + H(Z|\hat{X}). \end{aligned}$$

Task Assignment: Achievable Rates





Idea 4—Restrict the support of Y and Z around $\hat{X}$:

Optimize α (compression rate of $\hat{X}$):

$$\alpha^* \quad = \quad \phi = \frac{\sqrt{5}+1}{2}.$$

Resulting rates:

$$R_1 = R_2 = \log 3 - \log \phi.$$

Summary

1. Blackwell's Trapdoor Channel
   - $C_{FB} = \log \phi$.
   - Transform to equivalent channel
   - Simple zero-error communication scheme
   - Guess and check solution to Bellman equation
2. Three-node task assignment
   - Achievable symmetric rate: $H(X) - \log \phi$
   - Two phases of communication