Current Topics in Information Theory at Bell Labs

Emre Telatar

Bell Labs, Lucent Technologies

May 18, 1998

Lucent Technologies
Bell Labs Innovations

When is Information Theory Useful?

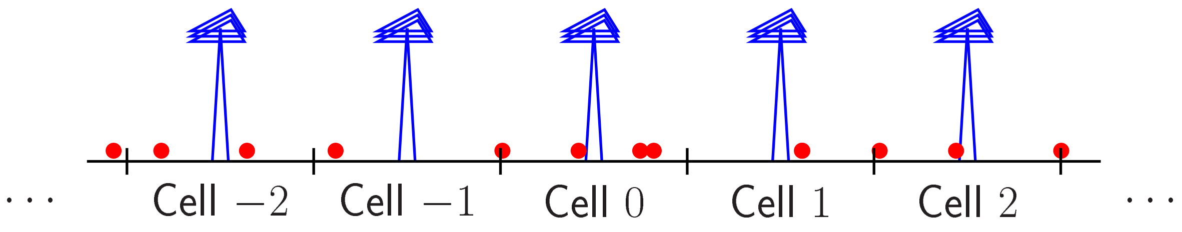
In communication systems, information theory is most useful when there is need to make efficient use of resources. It is not particularly useful if you have bandwidth and power to waste.

Fiber optics: Lots of bandwidth. Who cares if we are efficient. Bad news.

Wireless: Resources are inherently limited. Need to be smart to make good use of them. Good news.

Wireless Communication: Cellular Systems

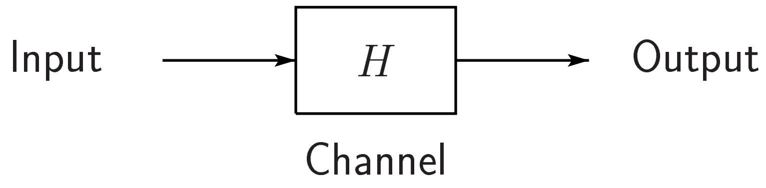
The most useful theoretical models are those abstracted of the details. Trade-offs can be understood clearly, and real insight gained.



Aaron Wyner's Model of Cellular Communication

Each mobile is received at its own cell, and also at the two neighboring cells at reduced strength.

Wireless Communication: Fading Channels

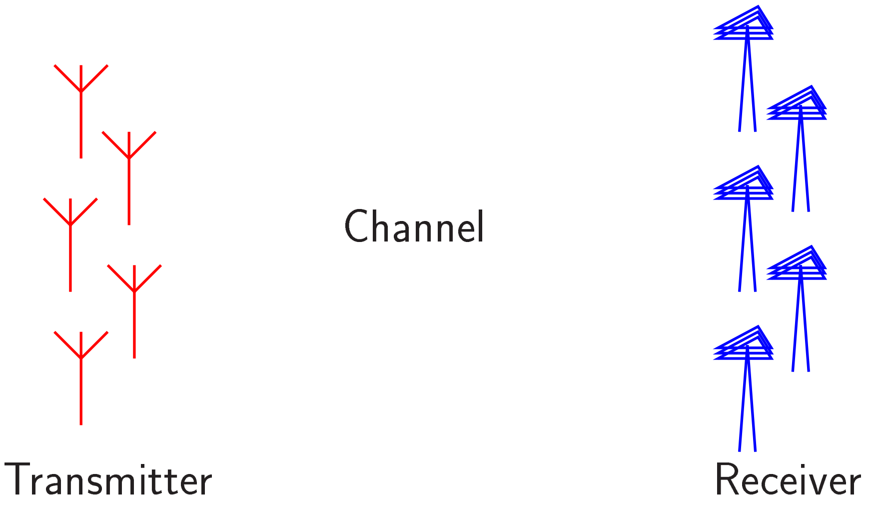


In wireless systems the channel H changes in time. To perform well, communication systems need to keep track of the changes in the channel.

This requires that the input signals be designed not only to combat noise but also to make estimation of the channel possible.

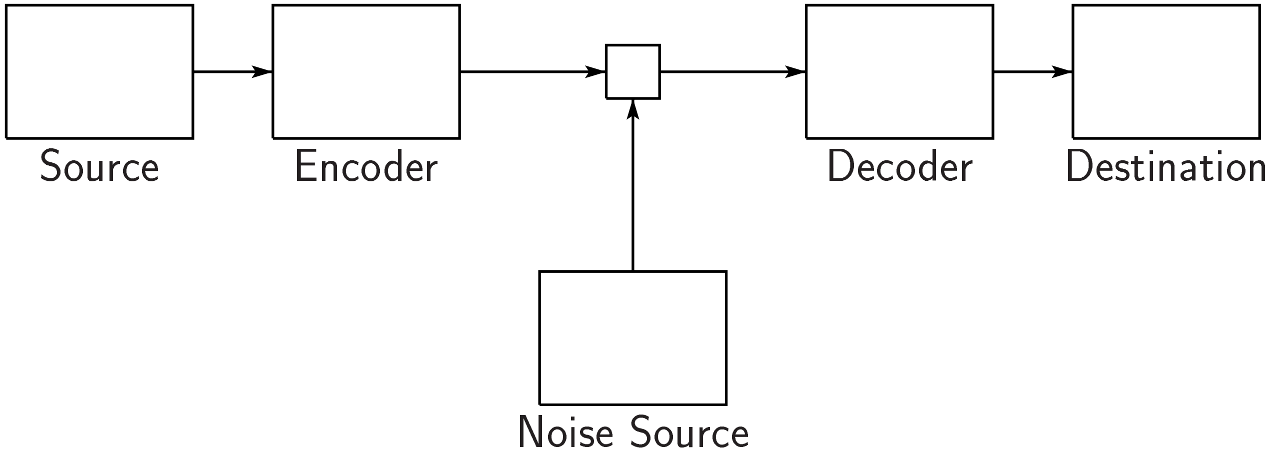
Information theory tells us which signaling schemes are not good and which may be good.

Wireless Communication: Multiple Antennas



Can one improve the wireless channel by the use of multiple antennas at the transmitter and receiver?

Turbo codes



The "coat of arms" of Information Theory

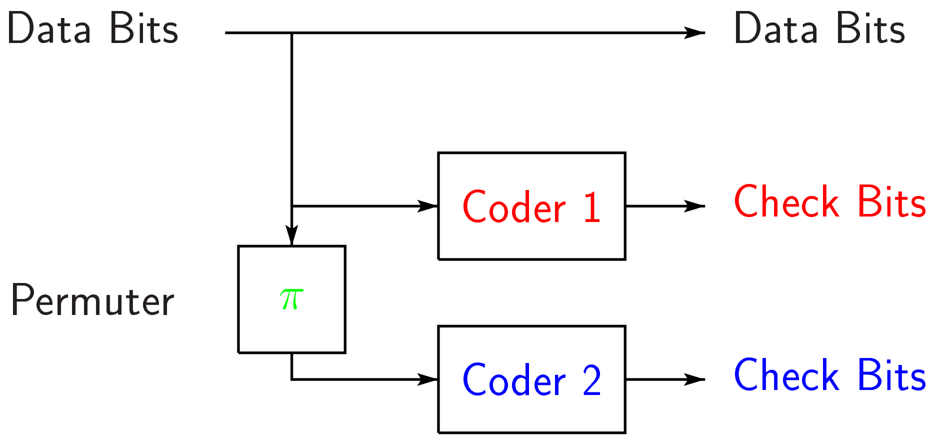
Shannon's promise: almost all codes are good.

Problem: optimal decoding of codes without structure is very hard; codes with structure do not seem to be very good.

Traditional coding theory: search for codes with structure so that we can decode them optimally — suboptimal codes with optimal decoders.

Turbo codes (continued)

Turbo codes: construct codes without structure. These are hard to decode optimally, so use suboptimal decoding.



Transmit data bits and the check bits over a noisy channel. Decode by iteration: obtain likelihoods of the data bits from the received data bits and the first set of check bits. Improve this using the second set of check bits. Repeat.

Why do Turbo Codes and Turbo Decoding Perform Well?

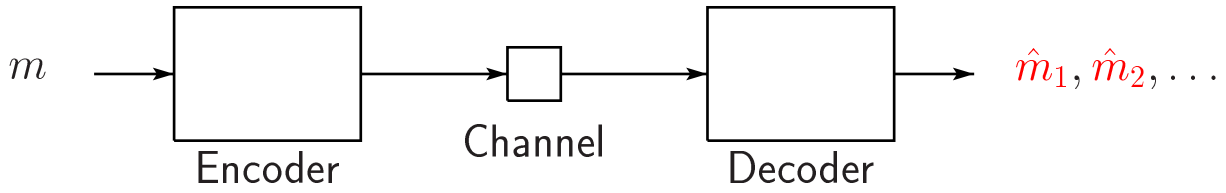
To understand why turbo codes are good, one needs to analyze the properties of the code, e.g., its distance profile.

To understand why this kind of iterative decoding works, one needs to use tools that are not conventional in communication theory.

Tom Richardson views the operation of the turbo decoding algorithm as the evolution of a dynamical system with a high dimensional state space. Using the methods of nonlinear dynamical systems analysis he can make statements about the convergence, stability and proximity to maximum-likelihood of a class of turbo-like decoding algorithms.

List Decoding

In traditional communication systems the receiver provides a single estimate of the message sent. In list decoding the receiver may provide multiple estimates (a list):



What is the trade-off between the list size and the probability of error?

Extreme point: insist that the correct message is always in the list. What is the highest communication rate possible while keeping the list size small on the average?

Zero-error vs. Regular Capacity

Adder channel with two inputs, X and Y, and one output $Z = X + Y$.

X-Code: $\{00, 10\}$, Y-Code: $\{00, 10, 11\}$. Good.

X-Code: $\{00, 10\}$, Y-Code, $\{00, 01, 11\}$. Bad.

Are there communication rates that are achievable with arbitrarily small probability of error, but not with zero error?

Universal Data Compression

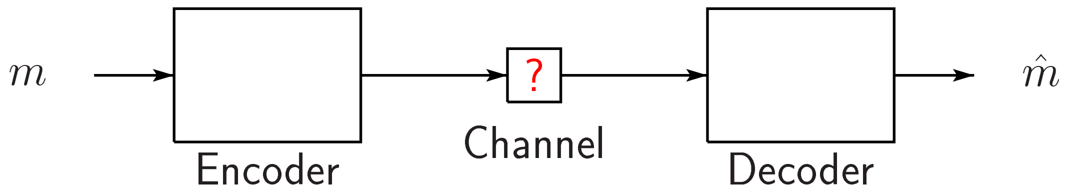
Modern data compression schemes operate by incrementally parsing the data stream:

0010110100110 1 ...

The length of the compressed string is intimately tied to the number of phrases in the parsing. Serap Savari has found simple ways to relate this number to the entropy of the source generating the string. One can then say how long a string needs to be so that it will compress well.

Communication over Unknown Channels

When can one design encoders and decoders such that they will perform well over every member of a given family of channels?



Recent results with Amos Lapidoth extend the known results from the 1950's to channels with memory.