Space-Time Processing for Wireless Communications

Babak Hassibi

Department of Electrical Engineering
California Institute of Technology

NSF/ONR Workshop on Ad-Hoc Networks
May 31-June 1, 2001

Cornell University, Ithaca, New York

Overview

Multiple-antennas systems have generated great interest for *high data rate* wireless communications, since they can

- significantly boost channel capacity
- lower the probability of error

of a wireless communications link. (Key: *spatial diversity*)

Applications abound and include:

- wireless LAN, fixed wireless access, mobile wireless, wireless Internet, etc.

What does all this mean for ad-hoc networks?

Why Multiple Antennas?

Spatial diversity is not a new thing (antenna arrays have been around for quite a while)

- In fact, the idea of diversity has also been around and has been exploited in a variety of forms
  - temporal diversity
  - frequency-domain diversity
  - multi-user diversity

  All these make communications more reliable, but they do not *directly* increase the channel capacity.

To increase capacity, it was traditionally believed that one has to either increase the transmit power or increase the bandwidth:

$$C = B \log(1 + \rho).$$

Neither of which is particularly exciting :(

But what about mutliple antennas? Well, pre-1995:

- fading is bad, scattering environment is bad
- line-of-sight is good
- beam-forming, angle-of-arrival estimation are the way to go
- capacity grows logarithmically in number of receive antennas

Things changed around 1995 (Foschini, Telatar). Now we know better:

- Fading is good! Rich-scattering environment is good!
- Capacity increases *linearly* in the minimum of the number of receive and transmit antennas.

This is now an exciting solution :)

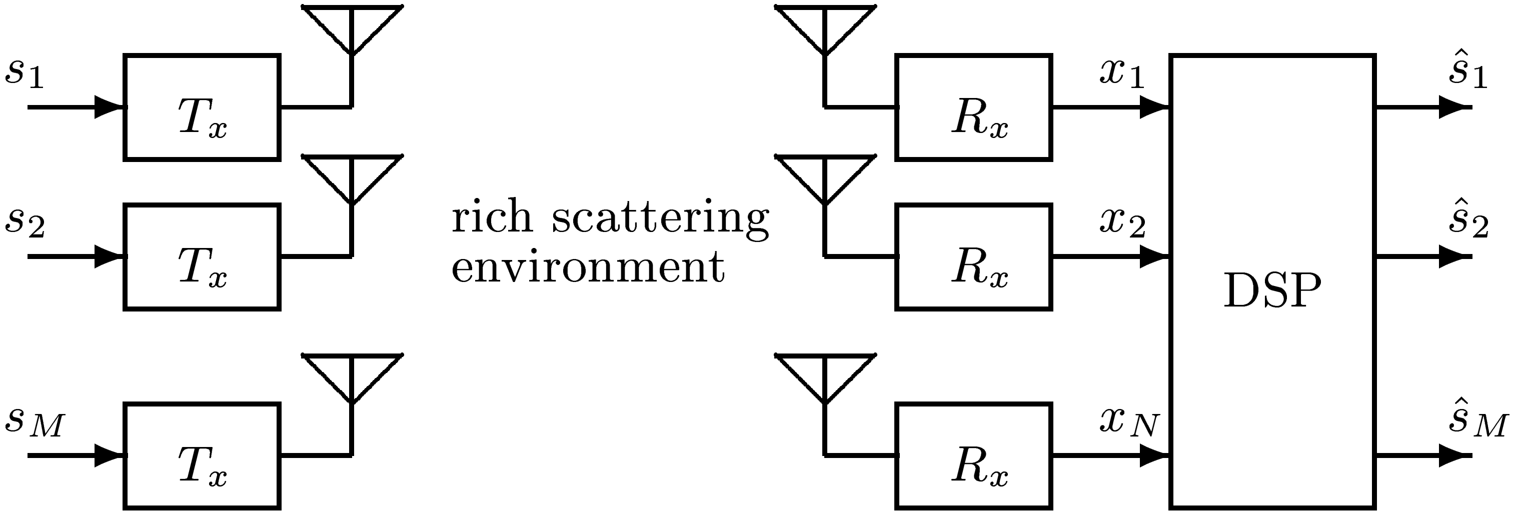
Research Challenges

- information theory
- coding theory (space-time codes)
- signal processing
- experimental — the propagation environment
  - Rayleigh vs. Rician fading, rich-scattering vs. line-of-sight
- RF circuits, antenna design
- system issues, network issues, multiple access, etc.

Will focus on the first three items.

(What does this all mean for ad-hoc networks?)

Information Theory



$$x = \sqrt{\frac{\rho}{M}} sH + v, \quad x = \left[\begin{array}{ccc} x_1 & \dots & x_N \end{array} \right], \; s = \left[\begin{array}{ccc} s_1 & \dots & s_M \end{array} \right]$$

If the channel H is known at the receiver, Wyner showed in the 70's that

$$C = E \log \det \left(I_N + \rho \frac{H^* H}{M} \right)$$

In 1995, Foschini and Telatar observed that, if H is rich-scattering,

$$C = \min(M, N) \log \rho + O(1).$$

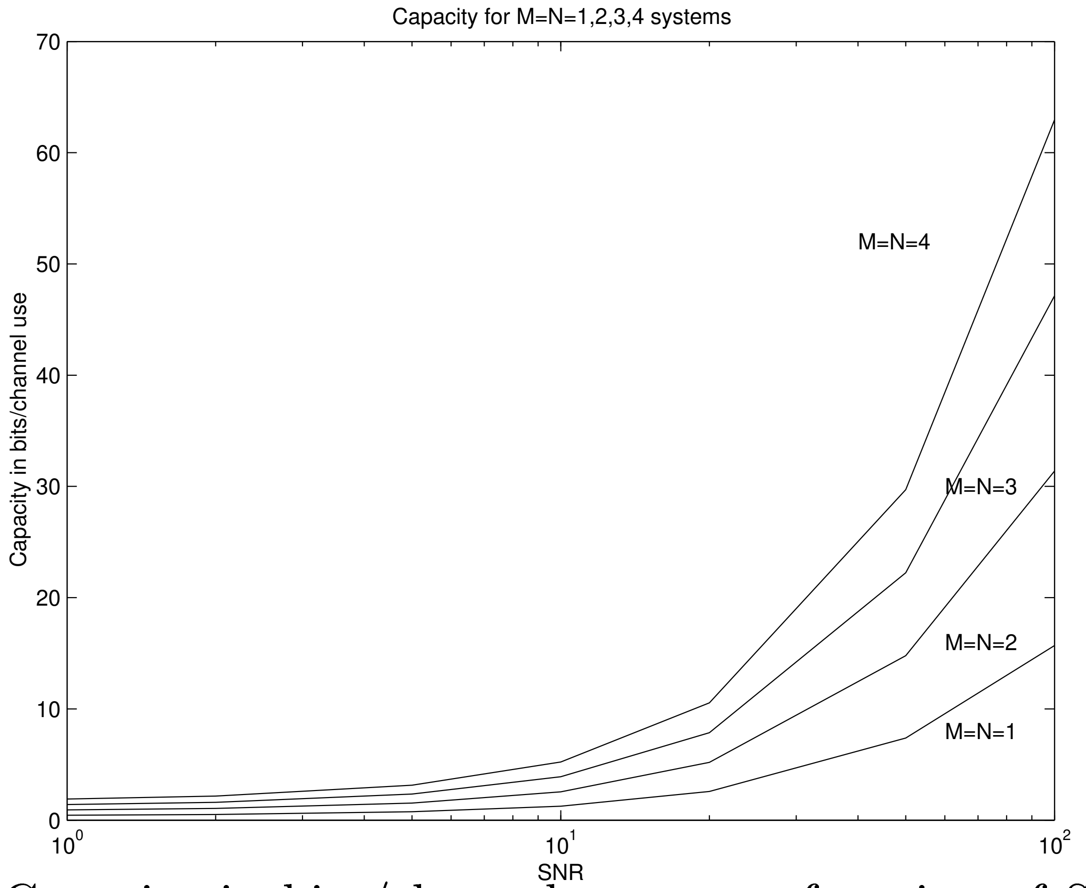


Figure 1: Capacity in bits/channel uses as a function of SNR for $M = N = 1, 2, 3, 4$ multi-antenna systems.

- Take $\rho = 20$db. Then the capacity of a single-antenna system is 5.9 bits/sec/Hz, whereas the capacity of a 10×10 multi-antenna system is 55.0 bits/sec/Hz.
  - this implies that for a 10×10 system it is theoretically possible to transmit at rate 1.65 Mbit/sec over a 30kHz bandwidth
  - to get the same rate with a single-antenna system requires a bandwidth of 280kHz, or an SNR of 170db!
- Of course, all this depends on the receiver knowing the channel. This is a reasonable assumption if M is not too large, or if the channel is not fading too rapidly.
  - so it makes sense for fixed-wireless access
  - but what about mobile wireless? (industry is beginning to consider 4×2, 4×4 and 8×4 systems)

The Block Fading Channel

A somewhat realistic model of a fading channel is the *block-fading* model: H is unknown to the receiver, but is fixed for a "coherence interval" of T channel uses, after which it changes to an independent value.

- computing the capacity for this channel is an open problem
- the high SNR capacity is (Zheng and Tse 2000, Hassibi and Marzetta 2001)

$$C = K\left(1 - \frac{K}{T}\right)\log\rho + O(1), \qquad K = \min(M, N, \frac{T}{2})$$

- *Autocapacity* (Hassibi, Marzetta and Hochwald 2000): If T and M are large enough ($T \geq 8$ and $M \geq 4$ appear to suffice) then reliable communication can be achieved by coding over a single coherence interval.

How to achieve this capacity?

- One method is to employ a training-based scheme:
  - use a portion of the coherence interval to send training symbols so that the receiver can learn the channel
  - use the rest of the coherence interval to transmit data, assuming that the receiver knows the channel
- Analysis of training-based schemes shows that (Hassibi and Hochwald 2000)
  - if optimized correctly, training-based schemes can achieve capacity at high SNR
  - training-based schemes are–by their very nature–highly suboptimal at low SNR

How to Convey Information?

In the block-fading model, it is useful to gather all the transmit and receive signals during one coherence interval into $T \times M$ and $T \times N$ matrices

$$S = \begin{bmatrix} s_{11} & \dots & s_{1M} \\ s_{21} & \dots & s_{2M} \\ \vdots & & \vdots \\ s_{T1} & \dots & s_{TM} \end{bmatrix} , \quad X = \begin{bmatrix} x_{11} & \dots & x_{1N} \\ x_{21} & \dots & x_{2N} \\ \vdots & & \vdots \\ x_{T1} & \dots & x_{TN} \end{bmatrix}$$

so that we may write

$$X = \sqrt{\frac{\rho}{M}} SH + V.$$

Thus, in multi-antenna systems, we transmit matrices and receive matrices.

But how can we convey information, given that H is unknown?

If we assume high SNR (or ignore the additive noise term V),

$$X \approx \sqrt{\frac{\rho}{M}} S H.$$

Key observation: H cannot alter the subspace spanned by the columns of S.

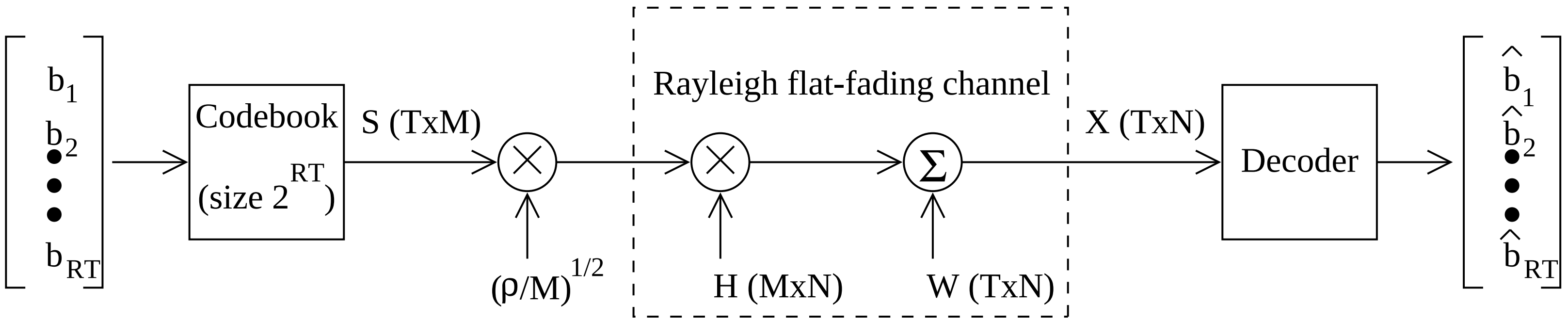
- Therefore what we can convey is this subspace information.
- The subspace information is best represented when the columns of S are orthonormal.
- Such transmission schemes are referred to as *unitary space-time modulation* (USTM).

Space-Time Codes

In multi-antenna systems the codes/constellations/signals transmitted and received are matrices.

- The extra spatial dimension adds a whole new twist to the problem of designing codes and constellations.

A *space-time code* (space-time constellation) is any set $\mathcal{S} = \{S_1, \ldots, S_L\}$ of $L = 2^{RT}$, $T \times M$ complex matrices.



Space-time codes fall under two general categories

- **Known channel codes** (coherent detection): here there is no restriction on the S_i
  - the maximum-likelihood decoder is given by

$$\arg \min_{i=1,\ldots,L} \|X - HS_i\|_F \, .$$

- **Unknown channel codes** (noncoherent detection): here the S_i have orthonormal columns
  - the maximum-likelihood decoder is given by

$$\arg \max_{i=1,\ldots,L} \|X^* S_i\|_F \, .$$

Linear Space-Time Codes

The most widely used class of known channel space-time codes are *linear*.

- The first such code was introduced by Alamouti in 1997 and is called an *orthogonal design*

$$S = \begin{bmatrix} s_1 & s_2 \\ -s_2^* & s_1^* \end{bmatrix}.$$

These have many desirable properties, but they cannot be easily generalized to more than two transmit antennas.

A general linear space-time code has the form

$$S = \sum_{q=1}^{Q} \left(s_q A_q + s_q^* B_q \right),$$

where the $\{A_q, B_q\}_{q=1}^{Q}$ are fixed $T \times M$ matrices, and the scalars s_q carry the information.

Design of Space-Time Codes

The quality of a space-time code is determined by the probability of mistaking one element of $\mathcal{S}$ by another. At high SNR, the probability of mistaking S_i with S_j is dominantly dependent on $|\det(S_i - S_j)|$.

- This leads us to the so-called *determinant criterion* (Fitz et al 1997, Tarokh et al 1998)

$$\max_{\mathcal{S}} \min_{i \neq j} |\det(S_i - S_j)| \, .$$

  - this criterion is very difficult to use for the design of high rate codes, especially when $M > 2$
  - it also has the unsatisfactory property of not depending on N

Any code for which $\det(S_i - S_j)$ is non-zero for all $i \neq j$ is called *fully-diverse.*

A more useful criterion is based on maximizing mutual information (Hassibi and Hochwald 2000). This leads to the following criterion

$$\max_{\{A_q, B_q\}} E \log\det\left(I + \frac{\rho}{M}\mathcal{H}\mathcal{H}^t\right), \tag{1}$$

where $\mathcal{H} = \mathcal{H}(\{A_q, B_q\}, H)$ is determined by

$$\operatorname{vec}\begin{bmatrix} \operatorname{Re}(SH) \\ \operatorname{Im}(SH) \end{bmatrix} = \mathcal{H}(\{A_q, B_q\}, H)\begin{bmatrix} \operatorname{Re}(s_1) \\ \operatorname{Im}(s_2) \\ \vdots \\ \operatorname{Re}(s_Q) \\ \operatorname{Im}(s_Q) \end{bmatrix}.$$

- The optimization problem (1) can be numerically solved for any rate R and for any number of transmit antennas M.
- The optimal codes are referred to as *linear dispersion* (LD) codes and they depend explicitly on N.

Here, for example, is an optimal three-antenna LD code (Hassibi and Hochwald 2000):

$$S = \begin{bmatrix} \alpha_1+\alpha_3+j[\frac{\beta_2+\beta_3}{\sqrt{2}}+\beta_4] & \frac{\alpha_2-\alpha_4}{\sqrt{2}}-j[\frac{\beta_1}{\sqrt{2}}+\frac{\beta_2-\beta_3}{2}] & 0 \\ \frac{-\alpha_2+\alpha_4}{\sqrt{2}}-j[\frac{\beta_1}{\sqrt{2}}+\frac{\beta_2-\beta_3}{2}] & \alpha_1-j\frac{\beta_2+\beta_3}{\sqrt{2}} & -\frac{\alpha_2+\alpha_4}{\sqrt{2}}+j[\frac{\beta_1}{\sqrt{2}}-\frac{\beta_2-\beta_3}{2}] \\ 0 & \frac{\alpha_2+\alpha_4}{\sqrt{2}}+j[\frac{\beta_1}{\sqrt{2}}-\frac{\beta_2-\beta_3}{2}] & \alpha_1-\alpha_3+j[\frac{\beta_2+\beta_3}{\sqrt{2}}-\beta_4] \\ \frac{\alpha_2-\alpha_4}{\sqrt{2}}+j[\frac{\beta_1}{\sqrt{2}}+\frac{\beta_2-\beta_3}{2}] & -\alpha_3+j\beta_4 & -\frac{\alpha_2+\alpha_4}{\sqrt{2}}+j[\frac{\beta_1}{\sqrt{2}}-\frac{\beta_2-\beta_3}{2}] \end{bmatrix}$$

where $s_q = \alpha_q + j\beta_q$, $q = 1, \ldots, 4$.

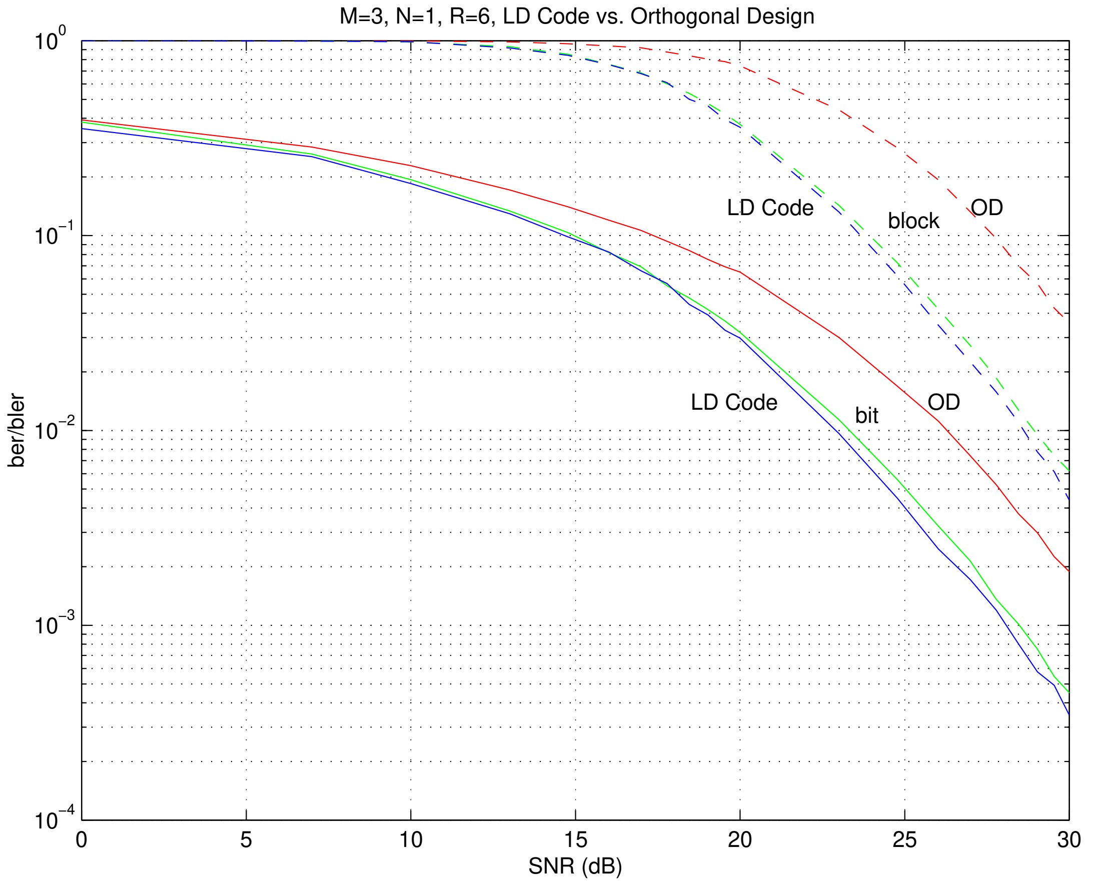


Figure 2: Block and bit error performance of OD versus LD code for $M = 3$ and $N = 1$ antennas at rate $R = 6$. The OD is ML decoded and the LD is decoded both with nulling/cancelling and with ML.

Unitary Space-Time Codes

Unknown channel space-time codes must be *unitary*.

- Tarokh etal (1999) derived such a code from Alamouti's scheme

$$S = \begin{bmatrix} x & y \\ -y^* & x^* \end{bmatrix}, \quad \text{where } |x|^2 + |y|^2 = 1.$$

Due to the unitarity constraint, designing unknown channel space-time codes is much more difficult/challenging.

- To break the logjam, we have recently considered the case where the space-time code forms a group under matrix multiplication.
- The resulting constellations can be thought of as multi-antenna analogs of PSK constellations.

Space-Time Codes from Groups

- Any fully-diverse code that forms a group must be *fixed-point-free*: all non-identity elements must have no eigenvalue at one. Indeed:

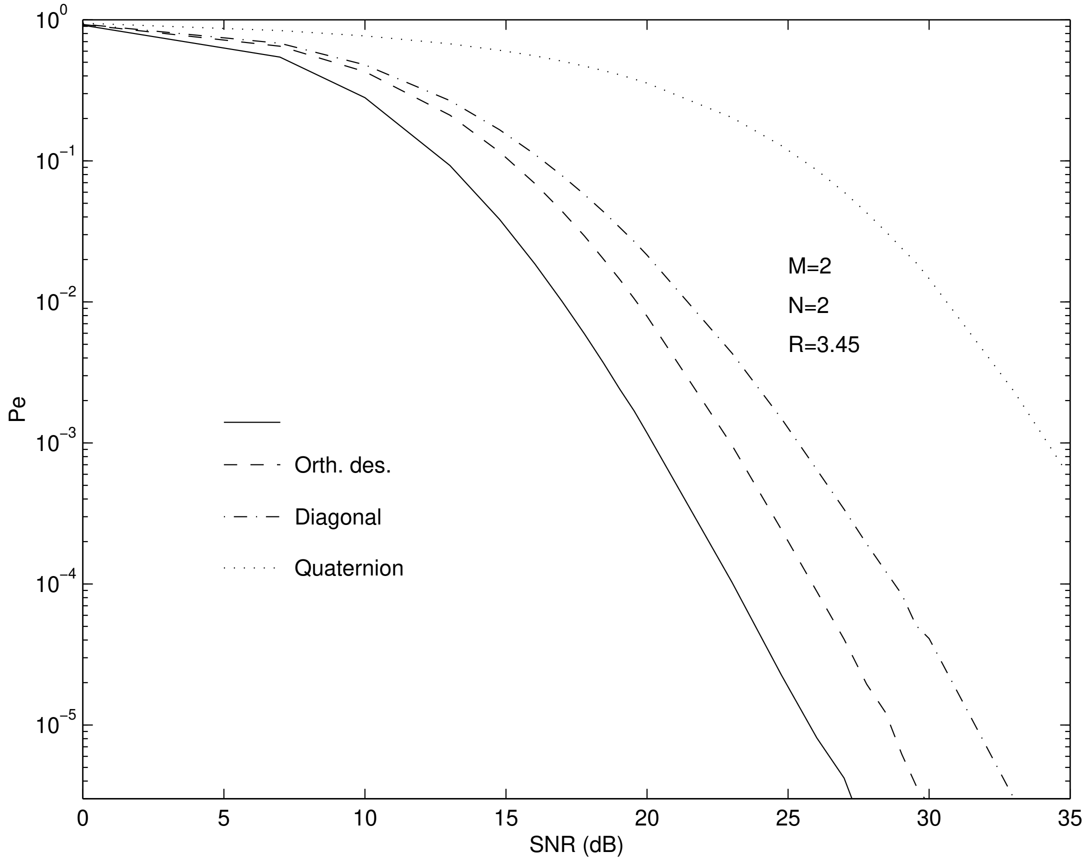
$$|\det(S_i - S_j)| = |\det(S_i)| \cdot \left|\det(I - S_i^{-1}S_j)\right| = |\det(I - S_k)|$$

- In Hassibi etal (2000) and Hassibi and Khorrami (2001), we have classified all finite and Lie fixed-point-free groups and have constructed the corresponding constellations.

- An optimal 120-element code, for example, is obtained from the 2×2 representation of the group $SL_2(\mathcal{F}_5)$ and is given by all possible products of μ and γ below where $\eta = e^{j2\pi/5}$

$$\mu = \frac{1}{\sqrt{5}} \begin{bmatrix} \eta^2 - \eta^3 & \eta - \eta^4 \\ \eta - \eta^4 & \eta^3 - \eta^2 \end{bmatrix} \quad , \quad \gamma = \frac{1}{\sqrt{5}} \begin{bmatrix} \eta - \eta^2 & \eta^2 - 1 \\ 1 - \eta^3 & \eta^4 - \eta^3 \end{bmatrix}.$$

$M = 2$, $N = 2$ **and** $R = 3.45$



Cayley Codes

- Fixed-point-free groups yield excellent constellations for low to moderate rates, but not for high rates.
- In Hassibi and Hochwald (2001) we have used the Cayley transform to construct high rate unitary space-time codes
  - the Cayley transform maps the nonlinear Stiefel manifold of unitary matrices to the linear space of skew-Hermitian matrices
  - the codes have the following form

    $$S = (I + jA)^{-1}(I - jA), \quad A = \sum_{q=1}^{Q} \alpha_q A_q$$

    where the $\{A_q\}$ are fixed $M \times M$ Hermitian matrices and the real scalars α_q carry the information.
  - code design is based on maximizing mutual information and ML decoding is reduced to an integer least-squares problem.

The Signal Processing Challenge

- Information theory suggests high data rates are possible in multi-antenna systems
- Space-time codes (in conjunction with error correcting codes) attempt to achieve these rates

Challenge: practical space-time transmission schemes must be simple yet effective: *all the processing done in real-time*

Size of the problem: We need to decode a set of $L = 2^{RT}$, $T \times M$ matrices. With $T = 8$ and $R = 16$, this is $L = 3.4 \times 10^{38}$ matrices!

Can this even be done?

Integer Least-Squares Problems

The problem of maximum likelihood decoding of linear space-time codes, as well as the class of Cayley unitary codes, reduces to an *integer least-squares problem*

$$\min_{s \in \mathcal{D} \subset \mathcal{Z}^m} \|x - As\|_2 ,$$

where $x \in \mathcal{R}^n$, $A \in \mathcal{R}^{n \times m}$ and $\mathcal{D}$ is a subset of the integer lattice $\mathcal{Z}^m$.

- For linear space-time codes A is simply the matrix $\mathcal{H}$ defined earlier. For a frequency-selective channel, A is block-Toeplitz.
- Interpretation: Given the "skewed" lattice As, find the "closest" lattice point to a Given n-dimensional vector x.

This problem is known to be NP-hard.

Some Heuristics

All practical systems employ approximations/heuristics:

- Invert and round to the closest integer (zero-forcing equalization):

$$\hat{s}_B = \left[H^{\dagger}x\right]_{\mathcal{Z}}.$$

  The above $\hat{s}_B$ is called the *Babai* estimate.

- Null and cancel (decision-feedback equalization):
  - only use the Babai estimate for one of the entries of s, say s_1
  - assume that s_1 is known and subtract out its effect to obtain a reduced integer least-squares problem with $m-1$ unknowns
  - solve similarly for s_2, etc.

- Nulling and cancelling with optimal ordering (BLAST):
  - perform nulling/cancelling from "strongest" to "weakest" signal

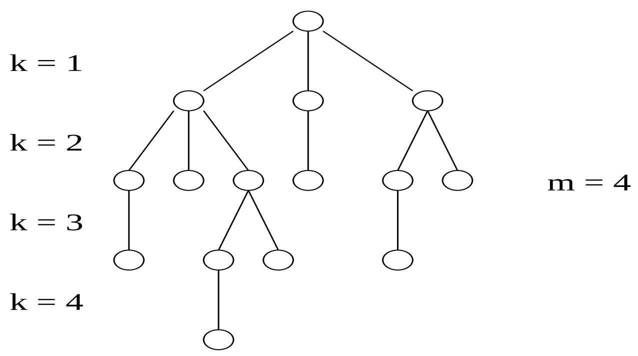
Exact Methods

- All the heuristic methods require $O(m^3)$ computations.
- There exist several exact methods for solving the integer least-squares problem that are a bit more sophisticated than doing a full search over the integer lattice.
- One is *sphere decoding* (Fincke and Post, 1985): search only over lattice points lying in a certain hypersphere centered around x.

Seems like a neat idea. But there are two important questions:

1. *How to choose r?* Clearly, if r is too big, we get too many points, but if r is too small, we get no points.
2. *How can we tell which lattice points are inside the sphere?*

- When $m = 1$ the answer to the second question is an interval.
- We can use this observation to go from dimension k to $k+1$: for every k-dimensional point in a sphere of radius r, the values of the $k+1$-th dimensional coordinate that lie in the higher dimensional sphere of radius r form an interval.
- Therefore the algorithm searches over all lattice points in spheres of radius r and dimensions $1, 2, \ldots m$.
- The algorithm constructs a tree, where the branches in the k-th level of the tree correspond to the lattice points inside the sphere of radius r and dimension k.



The Algorithm

Input: R, x, $\hat{s}$, r.

1. Set $k = m$, $r_m^{'2} = r^2 - \|x\|^2 + \|H\hat{s}\|^2$, $\hat{s}_{m|m+1} = \hat{s}_m$
2. (Bounds for s_k) Set $z = \frac{r_k^{'}}{r_{kk}}$, $UB(s_k) = \lfloor z + \hat{s}_{k|k+1} \rfloor$,
   $s_k = \lceil -z + \hat{s}_{k|k+1} \rceil - 1$
3. (Increase s_k) $s_k = s_k + 1$. If $s_k \leq UB(s_k)$ go to 5, else to 4.
4. (Increase k) $k = k + 1$ and go to 3.
5. (Decrease k) If $k = 1$ go to 6. Else $k = k - 1$,
   $\hat{s}_{k|k-1} = \hat{s}_k + \sum_{j=k+1}^{m} \frac{r_{kj}}{r_k k}(s_j - \hat{s}_j)$,
   $r_k^{'2} = r_{k+1}^{'2} - r_{k+1,k+1}^2 (s_{k+1} - \hat{s}_{k+1|k+2})$.
6. Solution found. Save s_k and go to 3.

A First Look at Complexity

Here is a very handwavy argument (that can be made rigorous):

- For an arbitrary point x, the expected number of lattice points inside a k-dimensional sphere of radius r is proportional to the volume
$$\frac{\pi^{k/2}}{\Gamma(k/2+1)} r^k.$$
Therefore the expected total number of points visited is
$$\sum_{k=1}^{m} \frac{\pi^{k/2}}{\Gamma(k/2+1)} r^k < \sum_{k=1}^{\frac{m}{2}} \frac{\pi^k}{\Gamma(k+1)} r^{2k} \approx e^{\pi r^2}, \quad \text{for large } m.$$
- To have a nonvanishing probability of finding a point in the m-dimensional sphere, its volume must be $\frac{\pi^{m/2}}{(m/2)!} r^m = O(1)$. But from Stirling's formula this implies that $r^2 = O(m)$ and that the complexity of the algorithm is exponential, $e^{O(m)}$.

Though not unexpected, this is a discouraging result.

- In our application, however, the vector x is not arbitrary, but is a lattice point perturbed by additive noise with known statistical properties:

$$x = As + v,$$

where the entries of v are independent $N(0, \sigma^2)$ random variables.

Under the above assumption, we have been able to *analytically* compute the expected complexity of sphere decoding (averaged over the noise and the lattice) (Hassibi and Vikalo 2001).

- When the entries of s are 2-PAM modulated (4-QAM, in the complex case), we have

$$C(m,\rho) = \sum_{k=1}^{m} (2k+17) \sum_{n=0}^{k} \left(\begin{array}{c} k \\ n \end{array} \right) \gamma \left(\frac{\alpha m}{1 + \frac{12\rho n}{m(L^2-12)}}, \frac{k}{2} \right),$$

where $\gamma(y,k) = \int_0^y \frac{\lambda^{k-1}}{\Gamma(k)} e^{-\lambda} d\lambda$ and α is such that $\gamma(m\alpha, m) = 1 - \epsilon$.

- When the entries of s are 4-PAM modulated (16-QAM, in the complex case), we have

$$C(m,\rho)=\sum_{k=1}^{m}(2k+17)\sum_{n}\frac{1}{2^k}\sum_{l=0}^{k}\left(\begin{array}{c}k\\ l\end{array}\right)g_n(k,l)\gamma\left(\frac{\alpha m}{1+\frac{12\rho n}{m(L^2-12)}},\frac{k}{2}\right)$$

where $g_n(k,l)$ is the coefficient of x^n in $\phi_0^l(x)\phi_1^{k-l}(x)$, where

$$\phi_0(x)=1+x+x^4+x^9 \quad \text{and} \quad \phi_1(x)=1+2x+x^4.$$

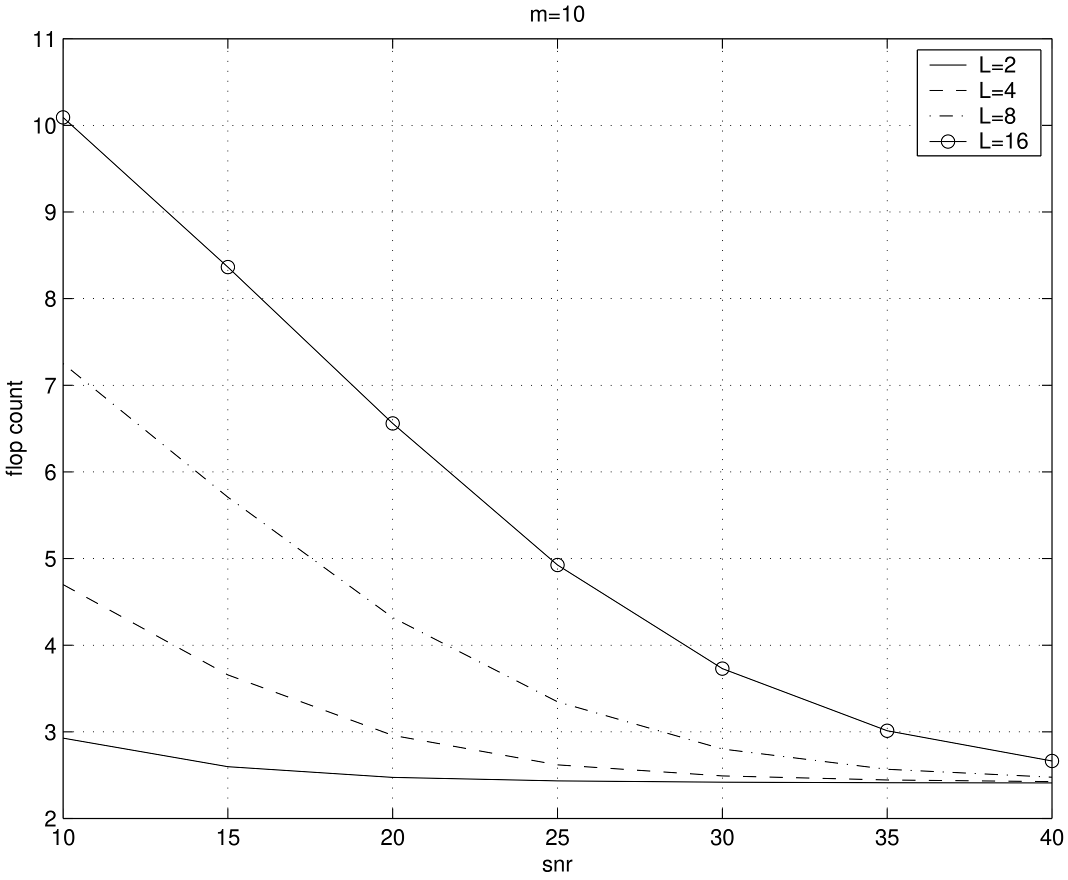
- Similar formulas can be developed for 8-PAM, 16-PAM, etc.

- It is often useful to look at

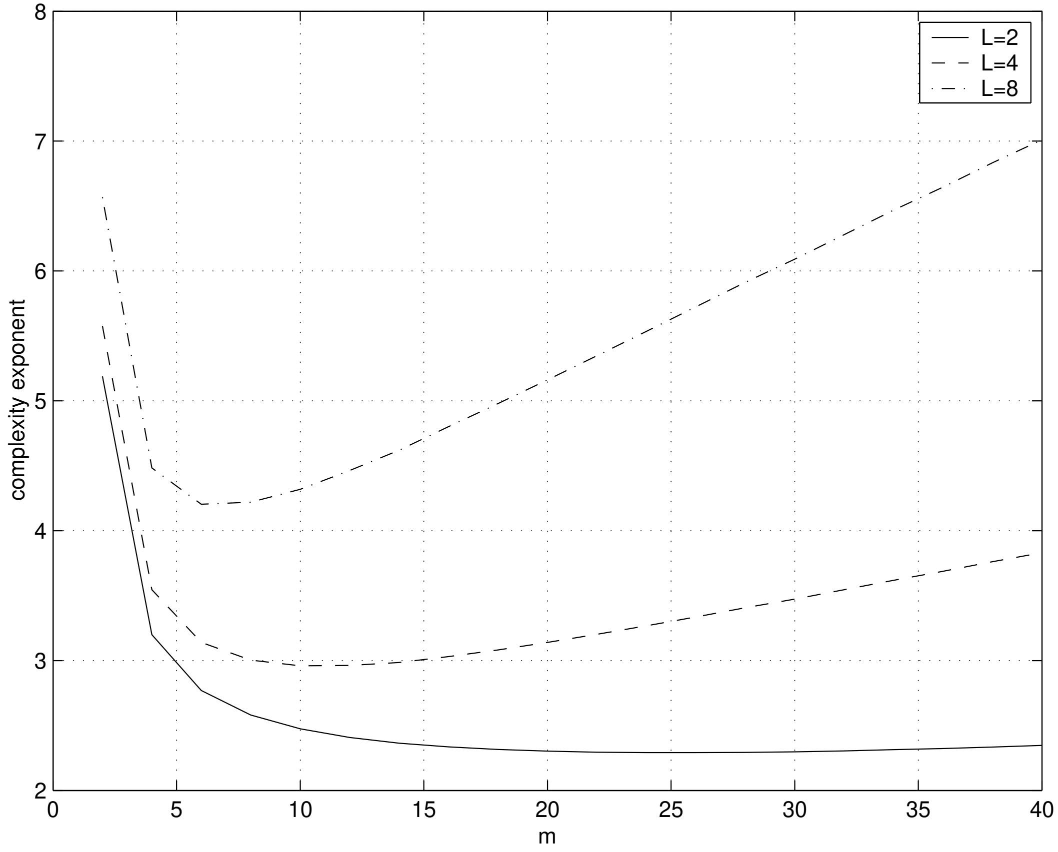
$$\frac{\log C(m,\sigma^2)}{\log M},$$

which approaches a constant if the expected complexity is polynomial, and grows like $\frac{M}{\log M}$ if it is exponential.

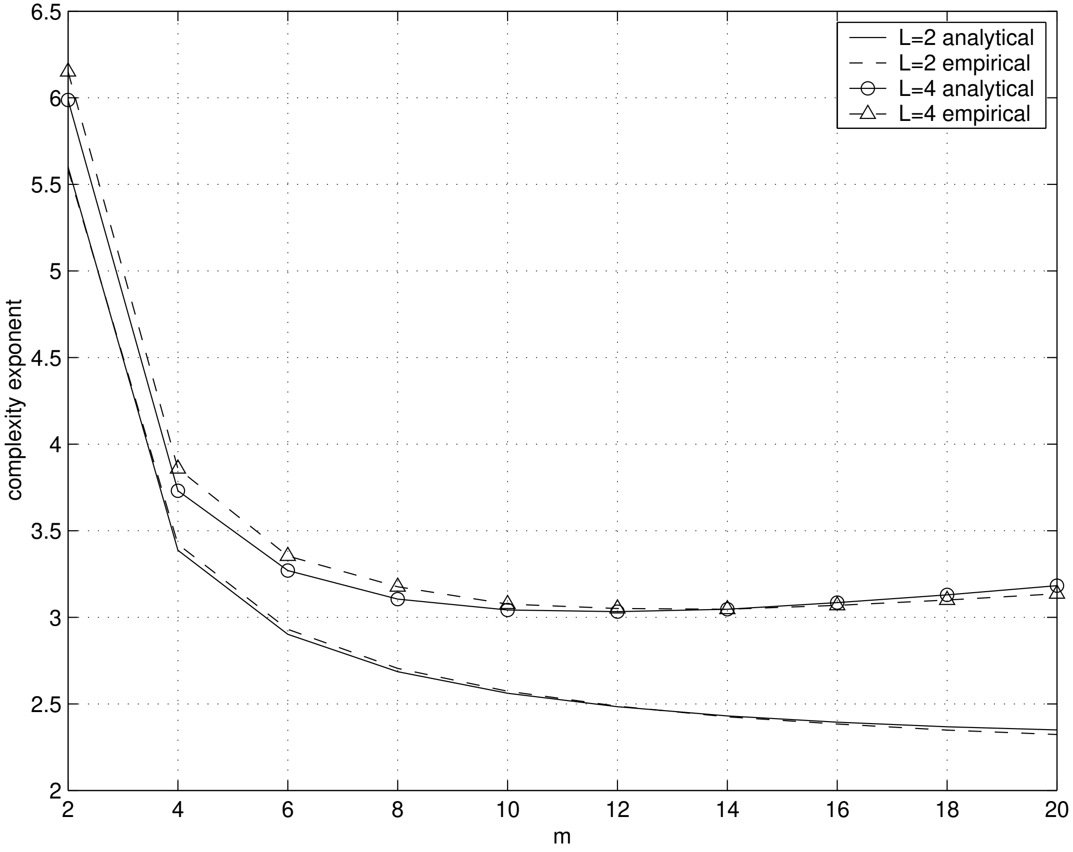
Complexity as a Function of SNR



Complexity as a Function of m



Empirical vs. Analytical Complexity



Computational Complexity and Shannon Capacity

Each entry of s carries $\log L$ bits of information so that the data transmission rate is

$$R = m \log L \quad \text{bits/channel use}$$

The simulations presented show that for a fixed m and L (and hence fixed transmission rate R) the computational complexity increases as we decrease the SNR.

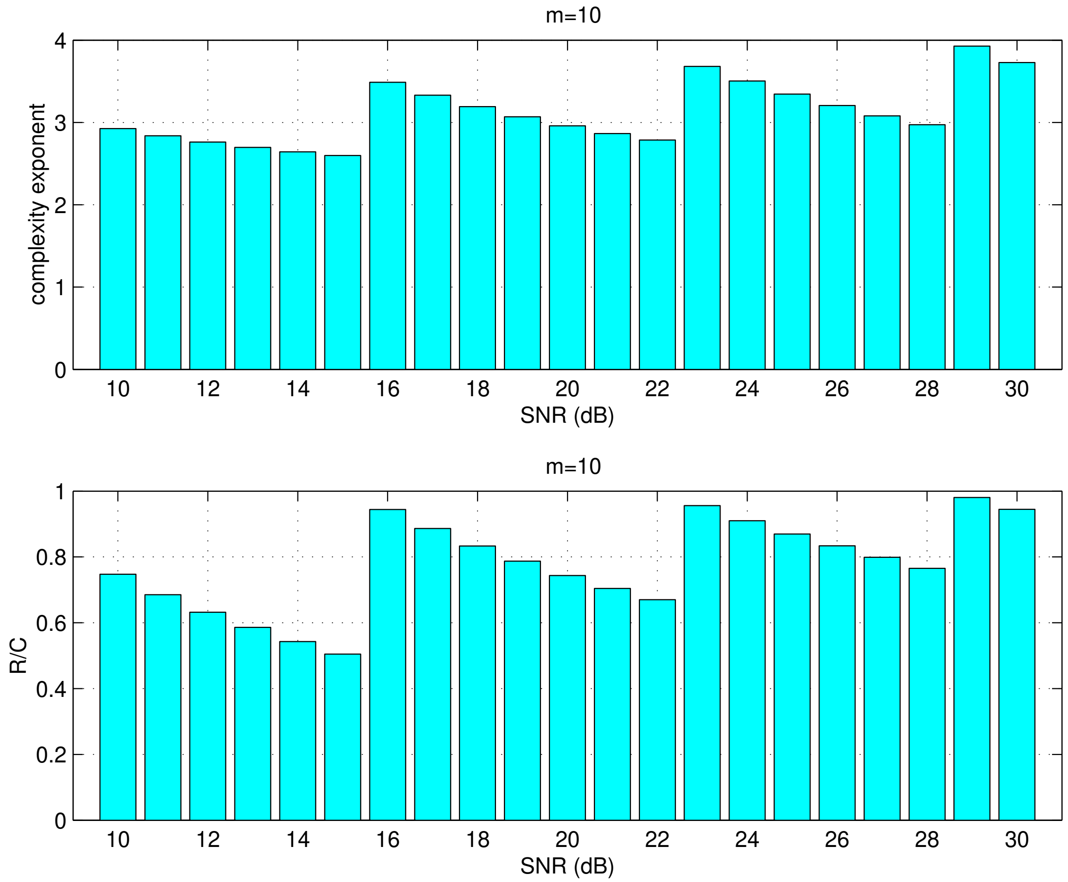
- In practice, however, we would never want the SNR to be such that the Shannon capacity

  $$C_{shannon}(m, \rho) = \frac{1}{2} E \log\det\left(I_m + \frac{\rho}{M} H H^T\right),$$

  is not able to support the rate R.

So what is the expected complexity like for $R < C_{shannon}(m, \rho)$?

Complexity vs. R/C



Speculation

Suppose we have a set $\mathcal{S} = \{S_1, \ldots, S_L\}$ of $L = 2^{RT}$ input signals of length T. Suppose further that we send these through a communications channel to obtain the output signal X and perform maximum-likelihood (ML) decoding at the output of the channel to recover the input signal

$$\hat{S} = \arg\max_{S \in \mathcal{S}} \log p(X|S).$$

Then information theory states that, as long as R is strictly less than the Shannon capacity C, then as $T \to \infty$ the probability of error in the ML decoder goes to zero.

- In practice ML decoding is rarely used since it requires an exponential search over the set $\mathcal{S}$.
- However, given that we are transmitting below capacity, is it possible that ML decoding can be done in polynomial-time?

Nulling/Cancelling vs. Polynomial-time ML

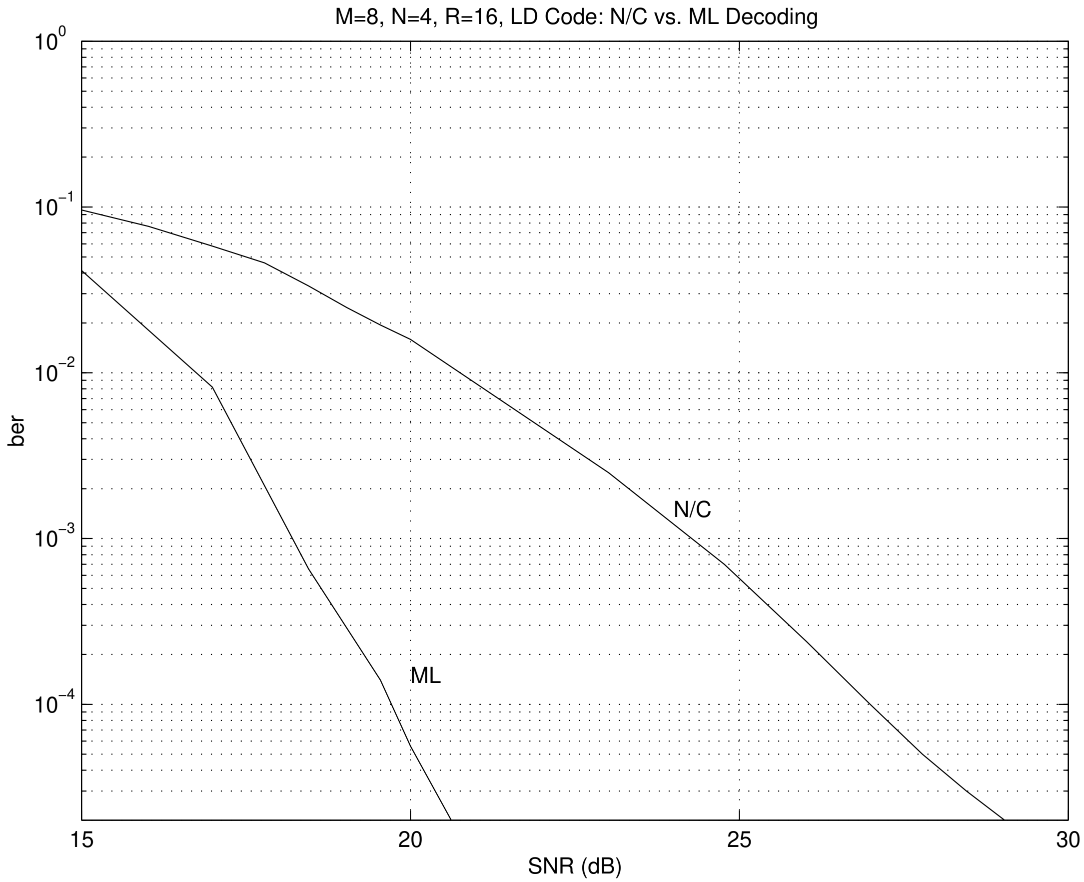


Figure 3: Bit error performance of an optimal linear space-time code for $T = 8$, $M = 8$ and $N = 4$, at rate $R = 16$. Note that $L = 2^{16\cdot 8} \approx 3.4\times 10^{38}$. The ML complexity was roughly twice that of nulling/cancelling.

Summary

Multiple-antenna systems promise very high data rates for wireless communications. To deliver on this promise, there are still many challenges and open problems.

- information theory
  - what are the fundamental limitations? how does fading affect things? training issues?...
- coding theory (space-time codes)
  - how to achieve capacity? known channel codes, unknown channel codes, groups, Cayley transforms...
- signal processing
  - how to do all the processing in real-time? sphere decoding, polynomial-time ML? equalization and frequency-selective channels...

What Does This all Mean for Ad-Hoc Networks?

- rich-scattering vs. line-of-sight
- interference and diversity vs. beam-forming
- link capacity vs. network capacity
- mobility vs. fixed
- how much feedback?
- training issues
- etc.