Multi-antenna Communications:

Information Theory and Algorithms

Babak Hassibi

Department of Electrical Engineering
California Institute of Technology

Information Theory Workshop
February 25 - March 1

Mathematical Sciences Research Institute, Berkeley, CA

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.

We shall focus on *some* of the more interesting mathematics in this area.

Multiple Antennas: A Brief History

Spatial diversity is not a new thing (antenna arrays have been around at least since the 1960's). It was believed that:

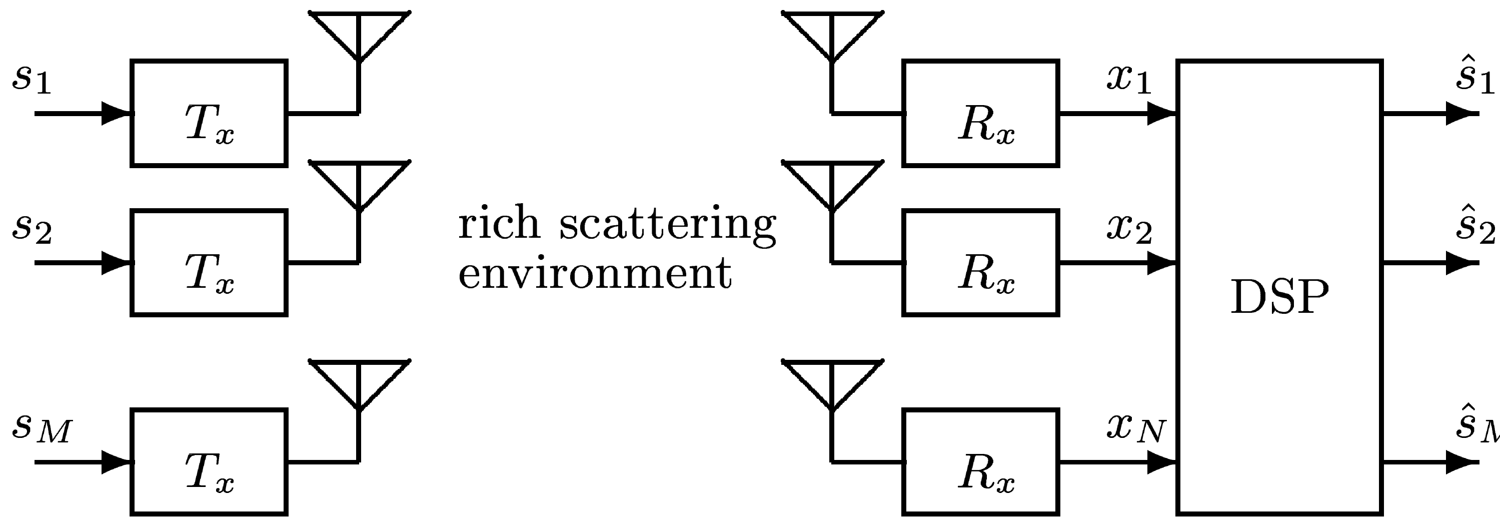
- fading is bad, scattering environment is bad
- line-of-sight is good
- beam-forming, angle-of-arrival estimation are the way to go
- capacity grows logarithmically in number of receive antennas

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 what has generated excitement!

The Model



$$x = \sqrt{\frac{\rho}{M}} sH + v,$$

where

$$x = \begin{bmatrix} x_1 & \dots & x_N \end{bmatrix}, \ s = \begin{bmatrix} s_1 & \dots & s_M \end{bmatrix}$$

$v \in \mathcal{C}^{1 \times N}$ has iid $\mathcal{CN}(0, 1)$ entries, and $H \in \mathcal{C}^{M \times N}$.

Capacity Results

- **Transmitter and receiver know H:**
  - capacity achieved by water-filling
  - $H = U\Sigma V^*$: the transmitter implements U and the receiver implements V^*, so the channel is diagonalized
- **Receiver knows H:**
  - In the 1970's Blankenburg and Wyner showed that

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

  - Rediscovered in 1995 by Foschini and Telatar, who further observed that, if H is rich-scattering,

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

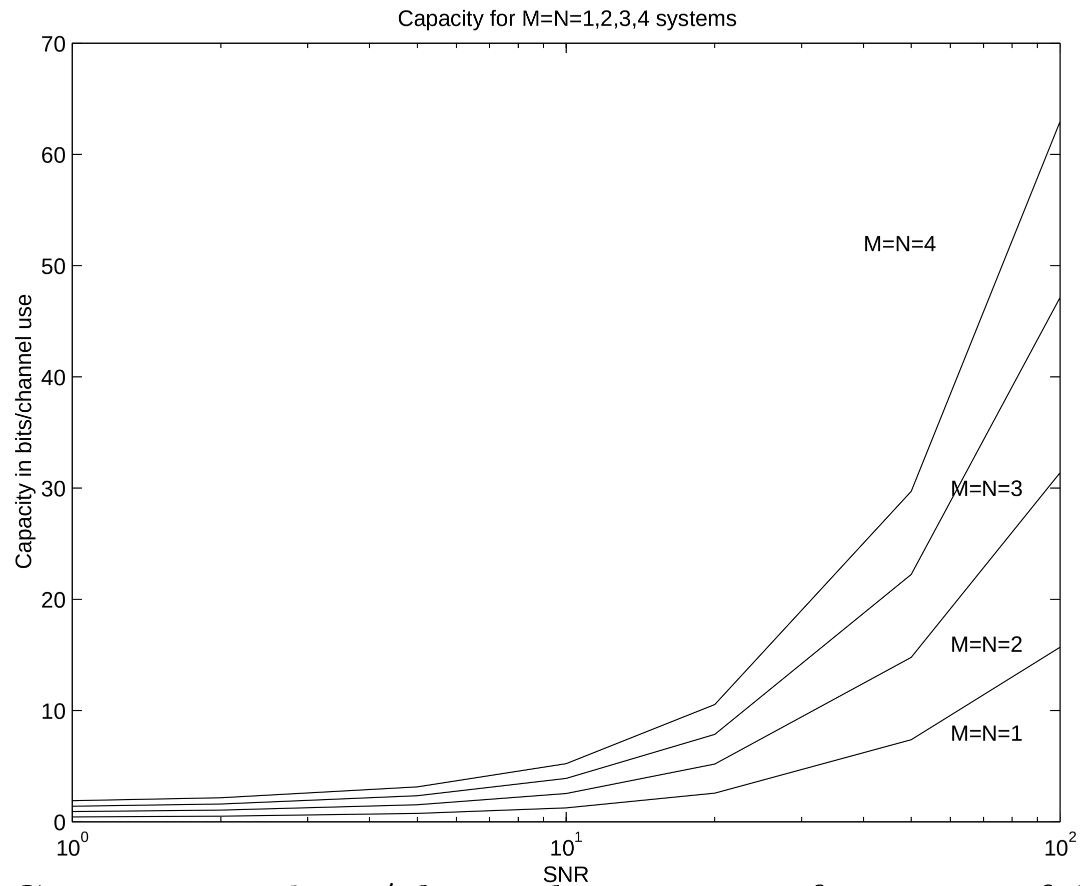


Figure 1: Capacity in bits/channel uses as a function of SNR for $M = N = 1, 2, 3, 4$ multi-antenna systems, assuming that the entries of H are iid $\mathcal{CN}(0, 1)$.

The above results depend on the the receiver (and transmitter) 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, where the channel is fast fading?

This requires us to say something about the temporal behaviour of H.

- The channel now is a random process, and not just a random matrix
- The problem is much more challenging

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
- structure of capacity-achieving distribution known
- 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* (Marzetta, Hochwald and Hassibi 2001): for large enough T and M 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),

$$X \approx \sqrt{\frac{\rho}{M}} SH.$$

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).

Structure of Capacity-Achieving Distribution

$$S = \Phi D, \qquad \text{(Marzetta and Hochwald 1999)}$$

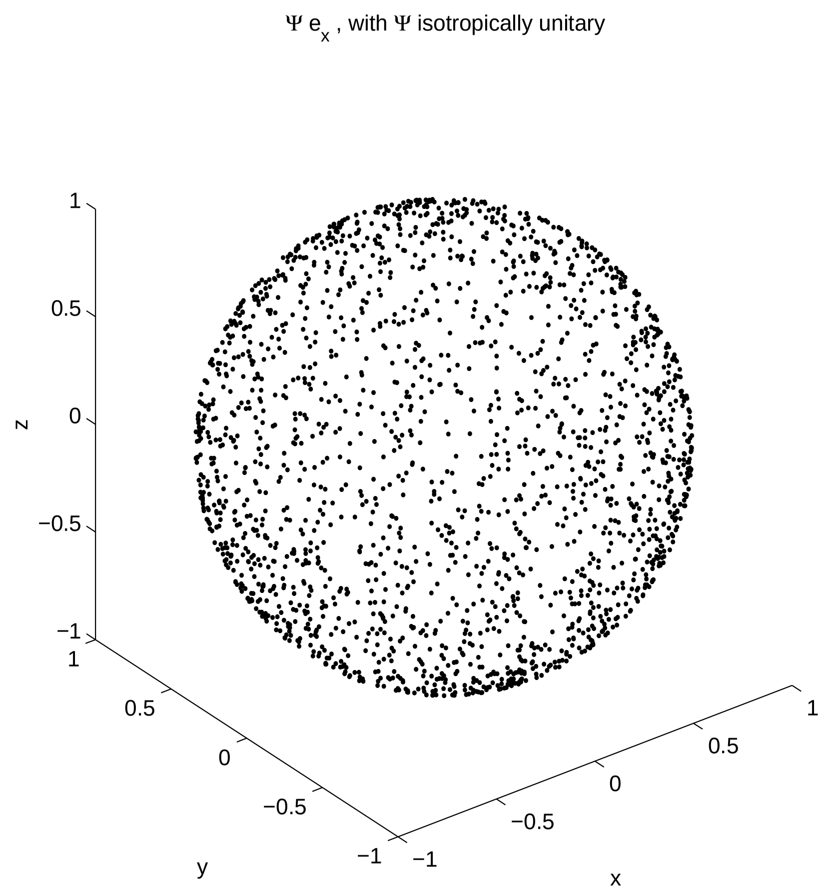
- $\Phi \in \mathcal{C}^{T \times M}$ is an *isotropically-random (i.r.) unitary* matrix
- $D \geq 0$ is an independent diagonal matrix with $E\text{tr}DD^* = TM$

An i.r. unitary matrix Ψ is one whose probability density function is invariant under pre- or post-multiplication by any fixed unitary matrix:

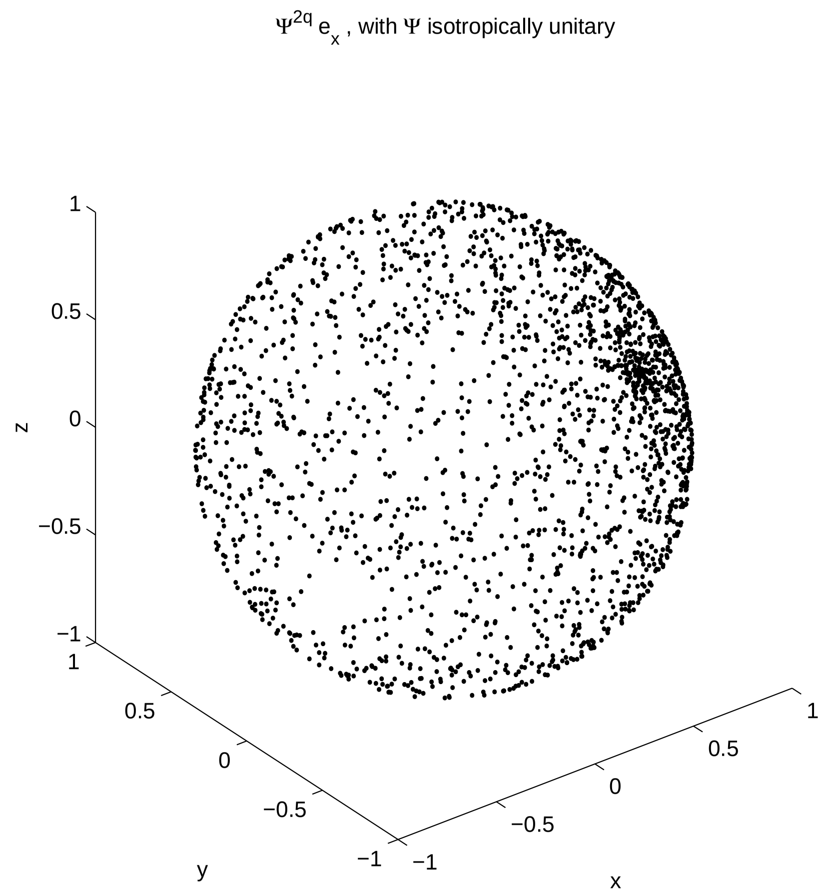
$$p(\Psi) = p(\Theta\Psi) = p(\Psi\Theta), \qquad \forall\Theta \text{ s.t. } \Theta\Theta^* = \Theta^*\Theta = I$$

- Also known as the *Haar* measure: the uniform measure on $U(n)$
- Is key to computing capacity, cut-off rates, error exponents, etc.
- One (of many) interesting facts (Wright and Diaconis 1998, Marzetta, Hassibi and Hochwald 2000): Ψ^ℓ is **not** i.r. for $n \geq 2$

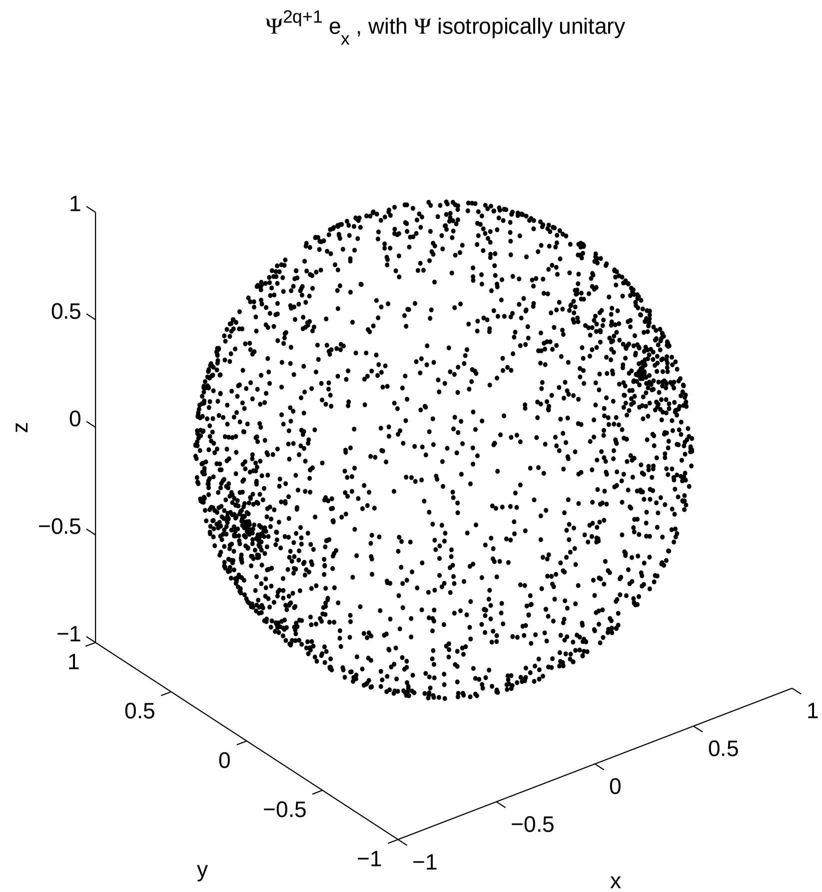
Ψe_x with Ψ Isotropically Unitary



$\Psi^2 e_x$ with Ψ Isotropically Unitary



$\Psi^3 e_x$ with Ψ Isotropically Unitary

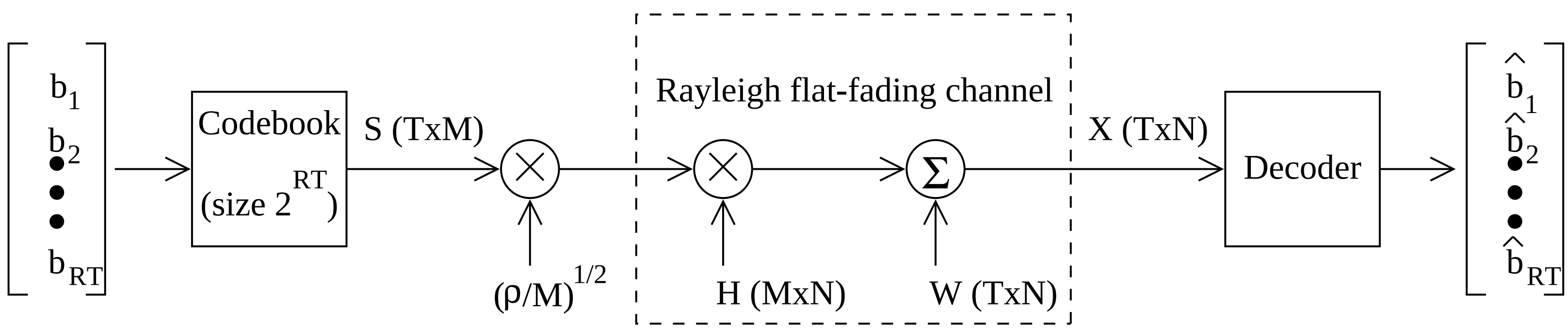


Space-Time Codes

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:

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

  Called an *orthogonal design*, it has many desirable properties (full-diversity, full-rate, decoupled ML decoding, etc.). But not clear how to generalize 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

One measure of 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)

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

  - 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*.

- Codes designed solely based on the determinant criterion tend to suffer from severe rate losses.
- To alleviate this, a design based on maximizing mutual information has been suggested (Hassibi and Hochwald, 2000)
  - called *linear dispersion* (LD) codes, they can be numerically found from the solution of a nonlinear optimization problem
  - they depend explicitly on N
  - they take very little rate hits
  - often exhibit surprising structure, which we do not understand
  - are nonunique and parametrized by a $2Q \times 2Q$ orthogonal matrix
- In general, there is a trade-off between rate and diversity
  - a possibility is to choose the $2Q \times 2Q$ orthogonal matrix to maximize the diversity of codes that achieve a certain rate

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

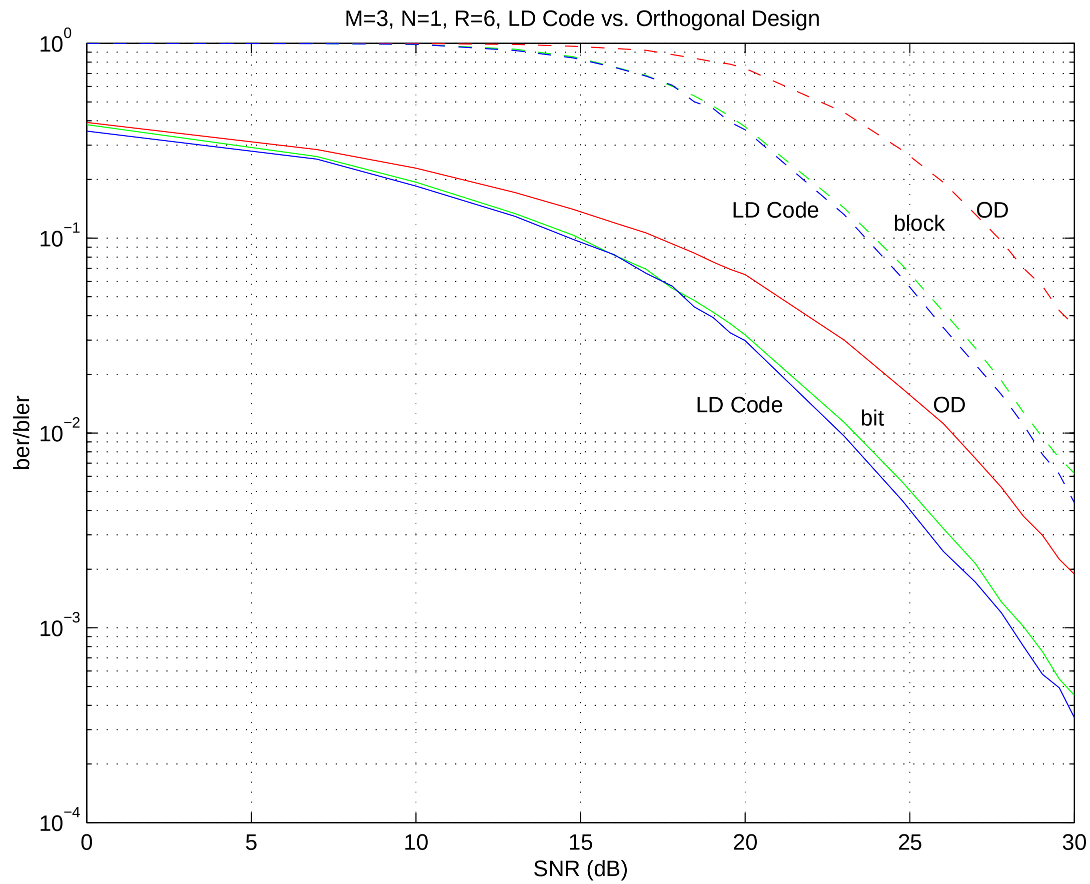


Figure 2: Block and bit error performance of OD versus LD code for $M = 3$ and $N = 1$ antennas at rate $R = 6$. The OD is ML decoded and the LD is decoded both with nulling/cancelling and with ML.

Unitary Space-Time Codes

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

- Tarokh and Jafarkhani (1999) derived such a code from Alamouti:

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

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.
- One motivation is that the resulting constellations can be thought of as multi-antenna analogs of PSK constellations.

Space-Time Codes from Groups

Any finite group can be represented as a set of unitary matrices. So which groups to choose...?

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

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

So we must look for fpf groups. *But what property should an abstract group $\mathcal{G}$ have such that, when represented as unitary matrices, all non-identity matrices have no eigenvalue at one?*

- Zassenhaus studied such groups in 1936 and gave an *almost* complete characterization

All Odd-Order Fixed-Point-Free Groups

Building upon Zassenhaus' work, here is the characterization of all odd-order fpf groups (Shokrollahi, Hassibi, Hochwald and Sweldens 2001)

Theorem 1 *A finite group G of odd order, L, is fixed point free if and only if it is isomorphic to the group*

$$G_{m,r} = \langle \sigma, \tau \mid \sigma^m = 1, \tau^M = \sigma^t, \tau\sigma\tau^{-1} = \sigma^r \rangle,$$

for some integers m and r such that:

(i) $L = mM$.

(ii) *M is the smallest integer such that* $r^M \equiv 1 \mathrm{mod}(m)$.

(iii) $\gcd(M, t) = 1$, *where* $t = \frac{m}{\gcd(r-1,m)}$.

(iv) *All prime divisors of M divide* $\gcd(r-1, m)$.

The Group Representation

The representation of the group takes the form:

$$\mathcal{V} = \left\{ \Delta(\sigma)^{\ell} \Delta(\tau)^{k} \,|0 \le \ell \le m-1, 0 \le k \le M-1 \right\},$$

where

$$\Delta(\sigma) = \begin{pmatrix} \eta & 0 & \cdots & 0 \\ 0 & \eta^{r} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \eta^{r^{M-1}} \end{pmatrix}, \quad \Delta(\tau) = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ \eta^{t} & 0 & 0 & \cdots & 0 \end{pmatrix}$$

and $\eta = e^{j2\pi/m}$.

Even-Order Fixed-Point-Free Groups

- Classification of even-order fixed-point-free groups is more involved
- In addition to $G_{m,r}$, there are five other group types
- One interesting even-order fixed-point-free group is $SL_2(\mathcal{F}_5)$. This group has 120 elements and can be expressed as

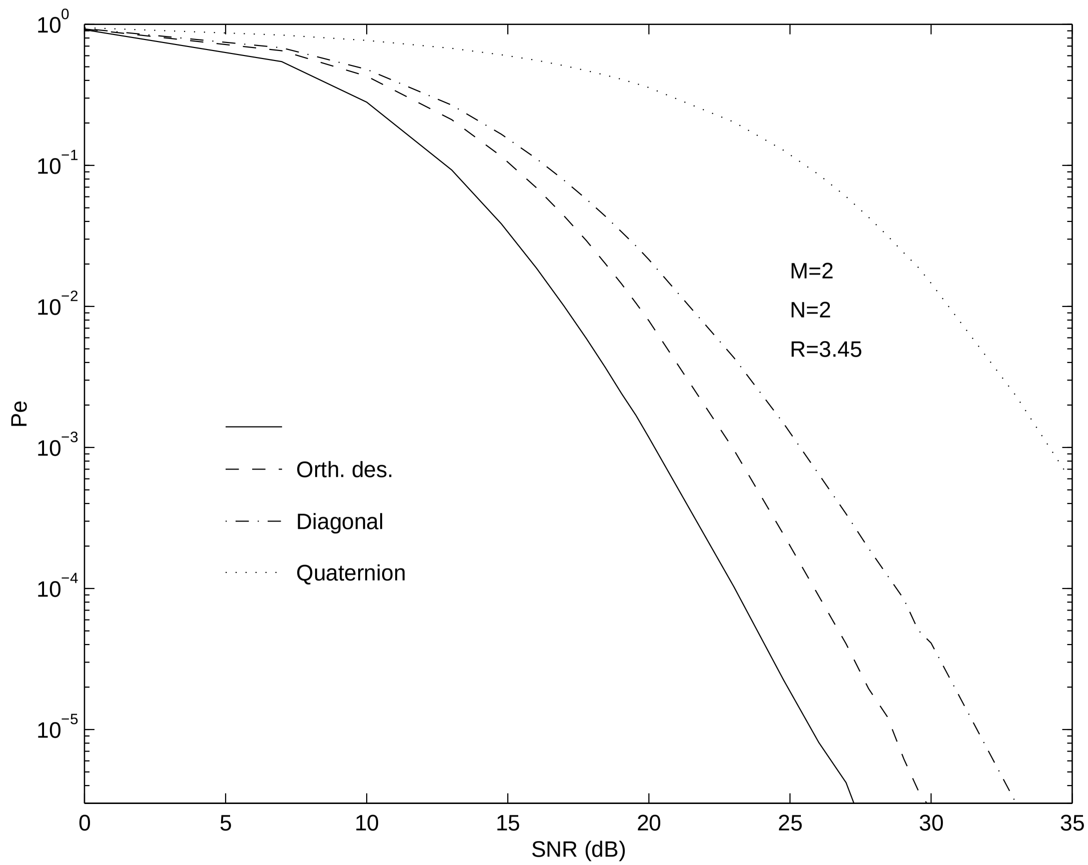
$$SL_2(\mathcal{F}_5) = \langle \mu, \gamma \mid \mu^2 = \gamma^3 = (\mu\gamma)^5, \mu^4 = 1 \rangle.$$

The representation of its generators is given by

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

where $\eta = e^{2\pi i/5}$.

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



Remarks

- We have thus classified all *finite* fpf groups
  - there are some star performers among these ($SL_2(\mathcal{F}_5)$)
  - they are generally few and far between
  - the best constellations are not obtained for very high rates or for a large number of antennas
- This brings up the question of whether there are any *infinite* fpf groups? It turns out that there are...

Phase modulation: $U(1)$, the group of unit-modulus complex scalars:

$$e^{j\omega}, \quad \omega \in [0, 2\pi[$$

Alamouti's scheme: $SU(2)$, unit-determinant 2×2 unitary matrices:

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

Other Infinite Fixed-Point-Free Groups?

But are there other infinite fpf groups?

- we will focus on Lie groups, which is the most interesting case (the above two examples are Lie groups)
- with a Lie group the problem of constellation design becomes one of sampling the group's underlying manifold (the unit-circle, in the first case, the 3-dimensional sphere, in the second)
- in fact, our star performer, $SL_2(\mathcal{F}_5)$ is an orthogonal design with an optimal sampling of 120 points on the 3-dimensional sphere

All FPF Lie Groups - Hassibi and Khorrami 2001

Lemma 1 *A Lie group has a representation as finite unitary matrices iff it is either $U(1)$, a compact semi-simple Lie group, or the direct sum of $U(1)$ and a compact semi-simple Lie group.*

Lemma 2 *If all non-identity elements in any unitary representation of a compact semi-simple Lie group have no more than k eigenvalues at unity, then the rank of the group is no more than $k - 1$.*

Therefore for fpf Lie groups we need only consider rank one groups. But there is only one such group: $SU(2)$.

Theorem 2 *The only fpf Lie groups are $U(1)$ and $SU(2)$. Their only irreducible fpf representations are 1- and 2-dimensional, respectively.*

The Symplectic Group $Sp(2)$

- The next best thing is to have no more than one eigenvalue at unity.
- This requires a rank of no more than two.
- There are three such semi-simple Lie groups: $SU(3)$, $Sp(2)$ and G_2.

$Sp(2)$ is the group of 4×4 unitary matrices Φ such that $\Phi^t J \Phi = J$, with $J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}$. Their representation is given by

$$\Phi = \begin{bmatrix} U\Sigma_x V & U\Sigma_y \bar{V} \\ -\bar{U}\Sigma_y V & \bar{U}\Sigma_x \bar{V} \end{bmatrix},$$

where U, V are unitary and $\Sigma_x, \Sigma_y \geq 0$ are diagonal with $\Sigma_x^2 + \Sigma_y^2 = I$.

- The above structure can be easily used to construct fully-diverse constellations (Jing and Hassibi 2002).

Cayley Codes

- Is there any other method (other than groups) to design unitary space-time codes?
- 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 i.r.u. matrix is transformed to a Cauchy random matrix
  - 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 Algorithmic 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?

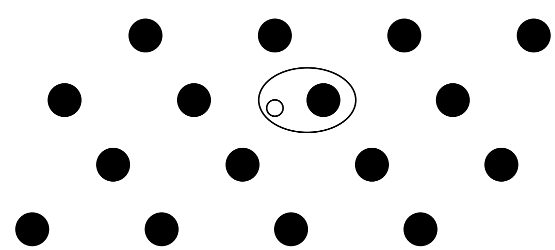
Integer Least-Squares Problems

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*

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

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

- Interpretation: Given the "skewed" lattice Hs, find the "closest" lattice point to a Given n-dimensional vector x.



This problem is well 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

How Good are the Heuristics?

- All the heuristic methods require $O(m^3)$ computations

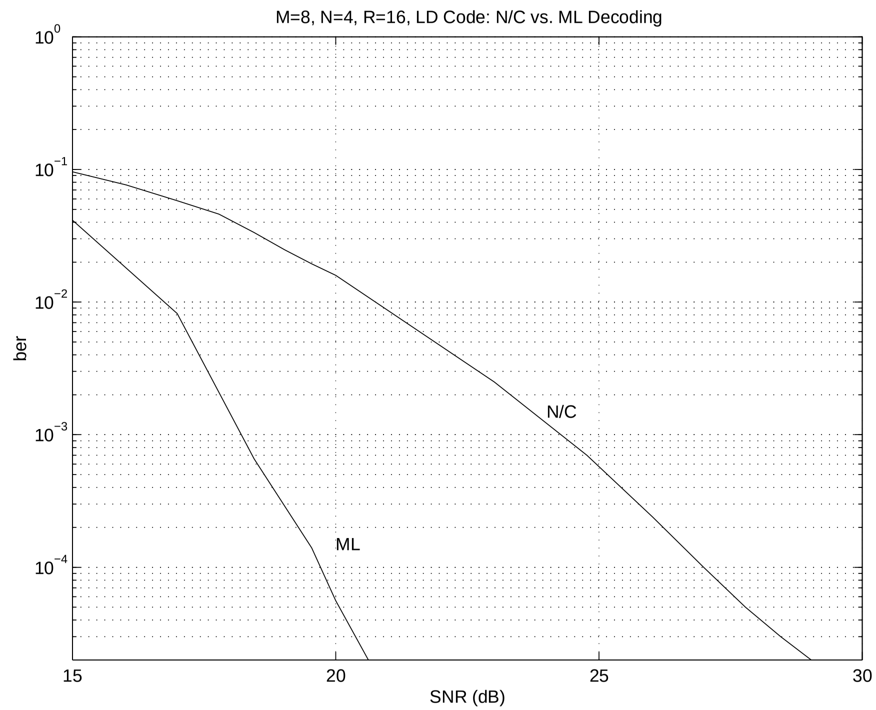
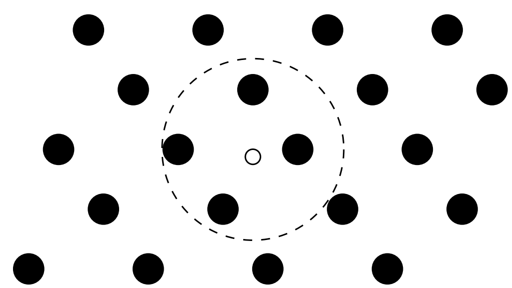


Figure 3: Bit error performance of a rate 16 linear space-time code, corresponding to $m = 64$. ML vs. nulling/cancelling with optimal ordering. (No. of lattice points= $2^{128} \approx 3.4 \times 10^{38}$).

Exact Methods

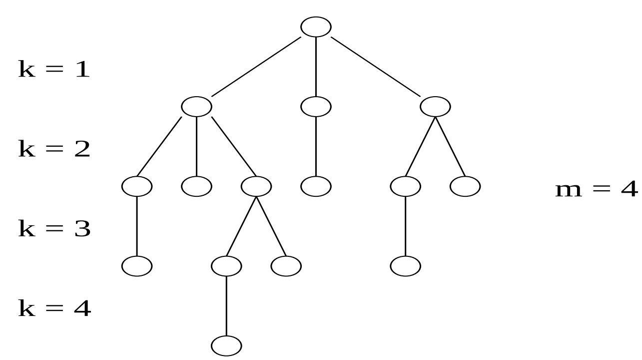
- There exist several exact methods that are a bit more sophisticated than doing a full search over the integer lattice
- One is *sphere decoding* (Fincke and Post, 1985): search only over lattice points lying in a certain hypersphere centered around x.



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 complexity of the algorithm depends on the *size* of the tree.

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, $e^{O(m)}$.

A Random Model

- Though not unexpected, this is a discouraging result.

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

$$x = Hs + v,$$

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

- A first by-product is that one should determine the radius r based on the noise, *not* on the lattice
- Clearly when $\sigma^2 = 0$, the exact solution can be found in $O(m^3)$
- When $\sigma^2 \to \infty$, the expected complexity is exponential

But what happens at intermediate noise levels?

The expected complexity is given by

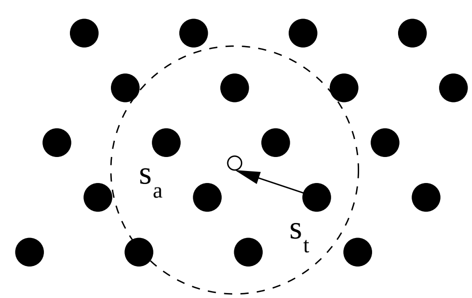
$$\sum_{k=1}^{m} (\text{expected \# of points in } k\text{-sphere of radius } r) \cdot (\text{flops/point})$$

$$= \sum_{k=1}^{m} E_p(k, r^2 = \alpha m \sigma^2) \cdot (2k + 17)$$

- How to compute $E_p(k, r^2)$?

Suppose that the lattice point s_t was transmitted and that the vector $x = Hs_t + v$ was observed. The probability that an arbitrary lattice point s_a lies in a hypersphere of radius r around x can be computed to be

$$\gamma\left(\frac{r^2}{\sigma^2 + \|s_a - s_t\|^2}, \frac{k}{2}\right) = \int_0^{\frac{r^2}{\sigma^2 + \|s_a - s_t\|^2}} \frac{\lambda^{k/2-1}}{\Gamma(k/2)} e^{-\lambda} d\lambda.$$



The probability just computed depends only on $\|s_a - s_t\|^2 = \|s\|^2$, i.e., on the squared norm of an arbitrary lattice point. Thus,

$$E_p(k, r^2) = \sum_{n=0}^{\infty} \gamma\left(\frac{r^2}{\sigma^2 + n}, \frac{k}{2}\right) \cdot (\#\text{ of lattice points with } \|s\|^2 = n).$$

Since

$$\|s\|^2 = s_1^2 + \ldots + s_k^2,$$

we basically, need to figure out how many ways a non-negative integer n can be represented as the sum of k squared integers.

This is denoted by $r_k(n)$ and is related to the classic Waring problem (1770). Euler introduced the following (now called Jacobi) theta function

$$\theta(x) = \sum_{m=-\infty}^{\infty} x^{m^2} = 1 + 2\sum_{m=1}^{\infty} x^{m^2},$$

and noted that

$$\theta^k(x) = 1 + \sum_{n=1}^{\infty} r_k(n) x^n.$$

Representing Integers as Sum of Squares

- Using the relationship between theta functions and elliptic functions Jacobi showed

$$r_2(n) = 4\left(d_1(n) - d_3(n)\right),$$

where $d_1(n)$ and $d_3(n)$ are the number of divisors of n congruent to 1 and 3 mod 4, respectively. Jacobi also obtained a similar formula for $r_4(n)$.

- Similar methods have been used to compute $r_k(n)$ for $k = 6, 8, 10, 12$. Ramanujan (and Hardy and Littlewood) computed explicit formulas for even $k \leq 24$. But that's about as far as it goes.
- A plethora of asymptotic results (in k and n) are available.
- In anycase, for any given k and n the value of $r_k(n)$ can be numerically computed using Euler's trick. It is also a built-in function in *Mathematica*, `SumOfSquaresR[k,n]`.

The Expected Complexity Over the Full Lattice

As a function of m, the lattice dimension, and σ^2, the noise variance, the expected complexity of sphere decoding therefore becomes (Hassib and Vikalo 2001):

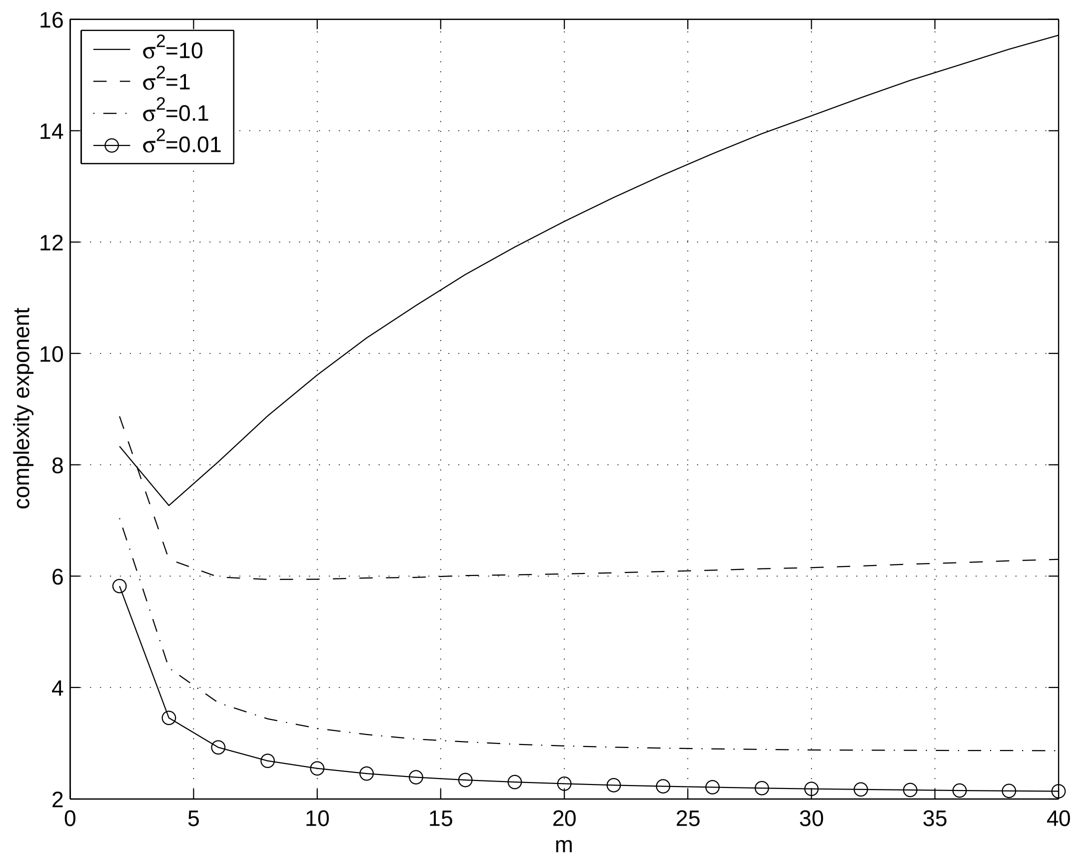
$$C(m, \sigma^2) = \sum_{k=1}^{m} (2k + 17) \sum_{n=0}^{\infty} r_k(n) \gamma \left(\frac{m\sigma^2}{\sigma^2 + n}, \frac{k}{2} \right).$$

It is often useful to look at the *complexity exponent*:

$$C_e = \frac{\log C(m, \sigma^2)}{\log m}.$$

- When $C = O(m^\alpha)$, then $C_e \approx \alpha$.
- When $C = O(\beta^m)$, then $C_e \approx \frac{m}{\log m}$.

Expected Complexity as a Function of m



Expected Complexity for Finite Constellations

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

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

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

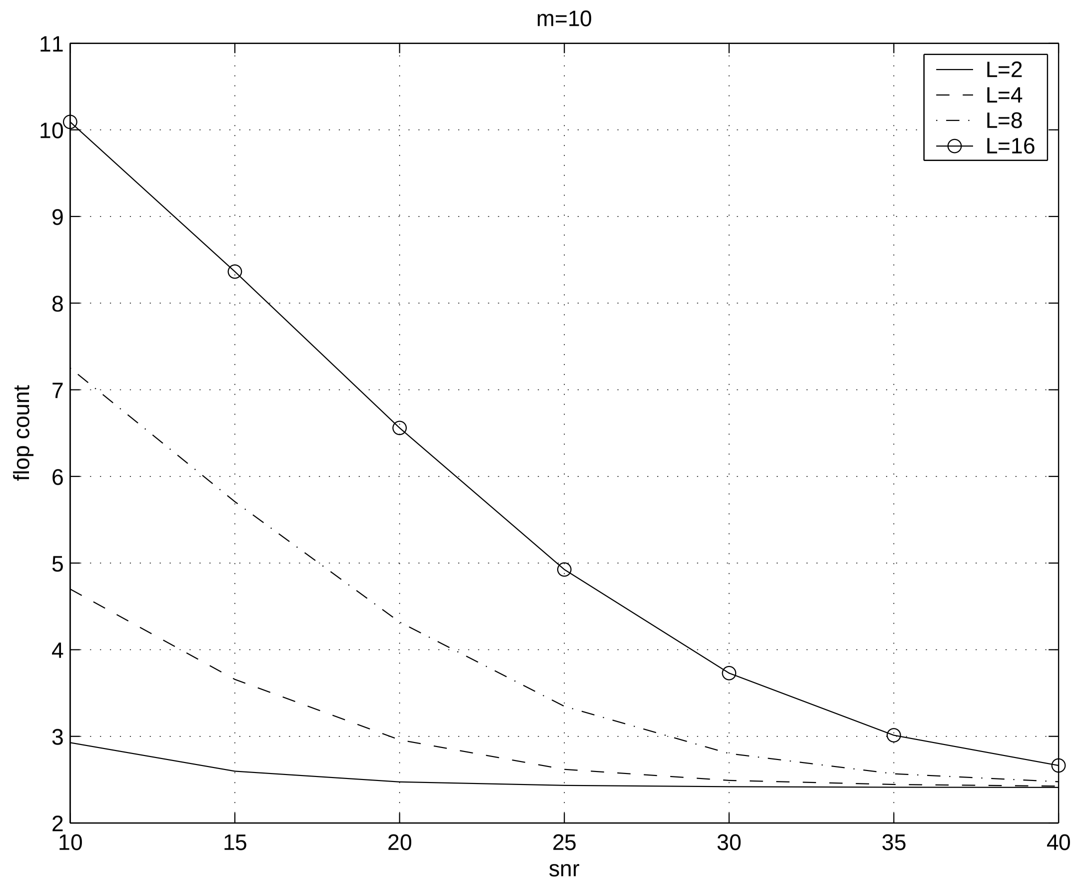
$$C(m,\rho) = \sum_{k=1}^{m} (2k+17) \sum_{n} \frac{1}{2^k} \sum_{l=0}^{k} \begin{pmatrix} k \\ l \end{pmatrix} g_n(k,l) \gamma \left(\frac{\alpha m}{1 + \frac{12\rho n}{m(L^2-12)}}, \frac{k}{2} \right)$$

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

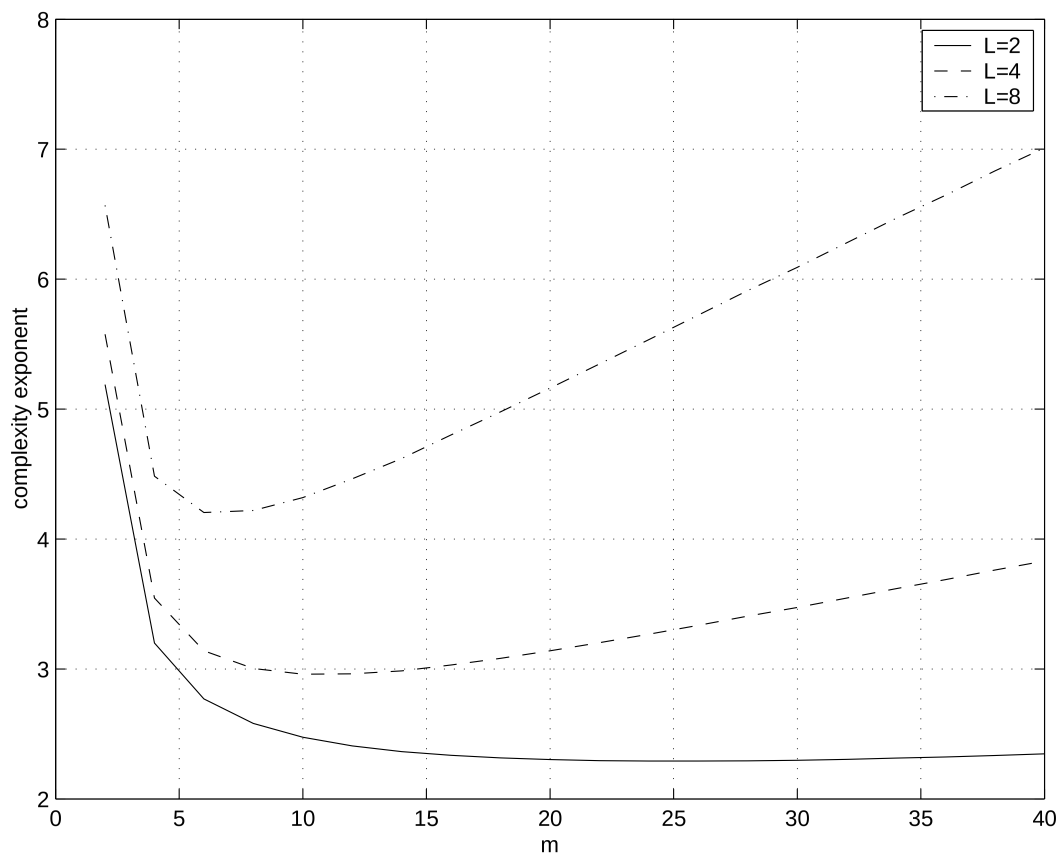
$$\phi_0(x) = 1 + x + x^4 + x^9 \quad \text{and} \quad \phi_1(x) = 1 + 2x + x^4.$$

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

Complexity as a Function of SNR



Complexity as a Function of m



Computational Complexity and Shannon Capacity

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

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

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.

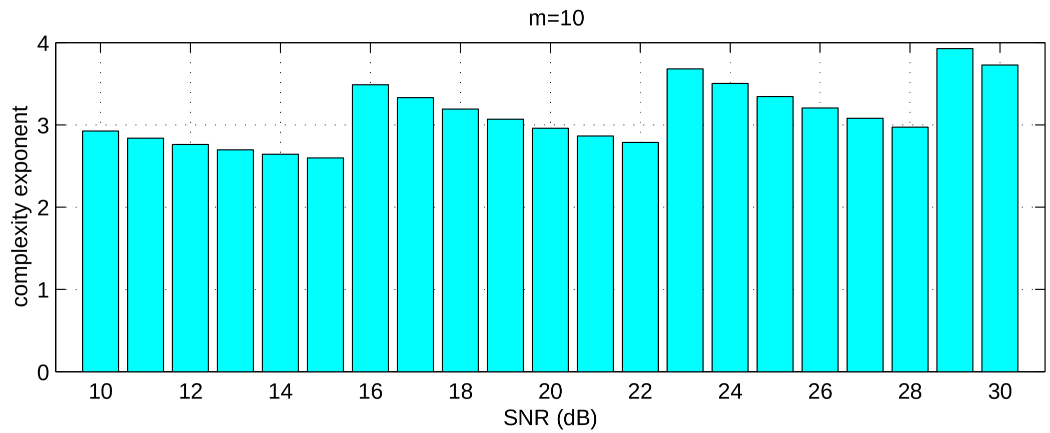
- In practice, however, we would never want the SNR to be such that the Shannon capacity

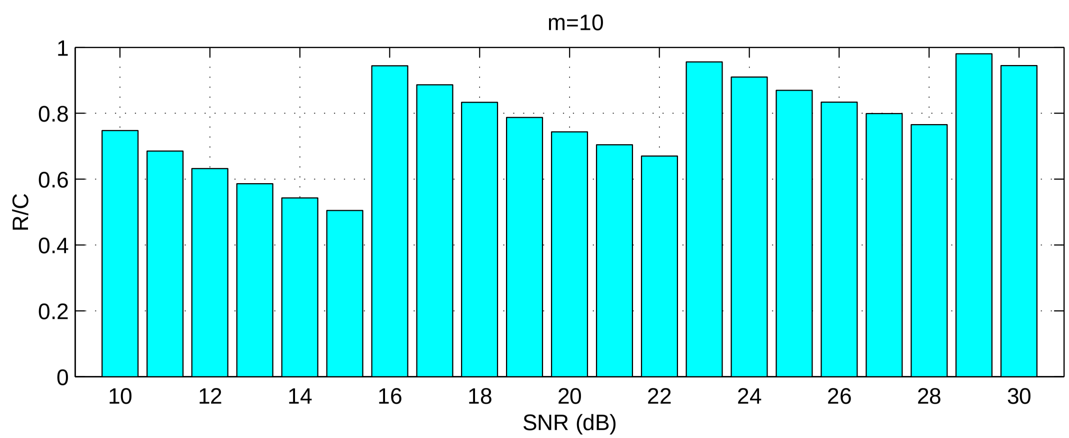
  $$C_{shannon}(m, \rho) = \frac{1}{2} E \log \det \left(I_m + \frac{\rho}{M} H H^T \right),$$

  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





Nulling/Cancelling vs. Polynomial-time ML

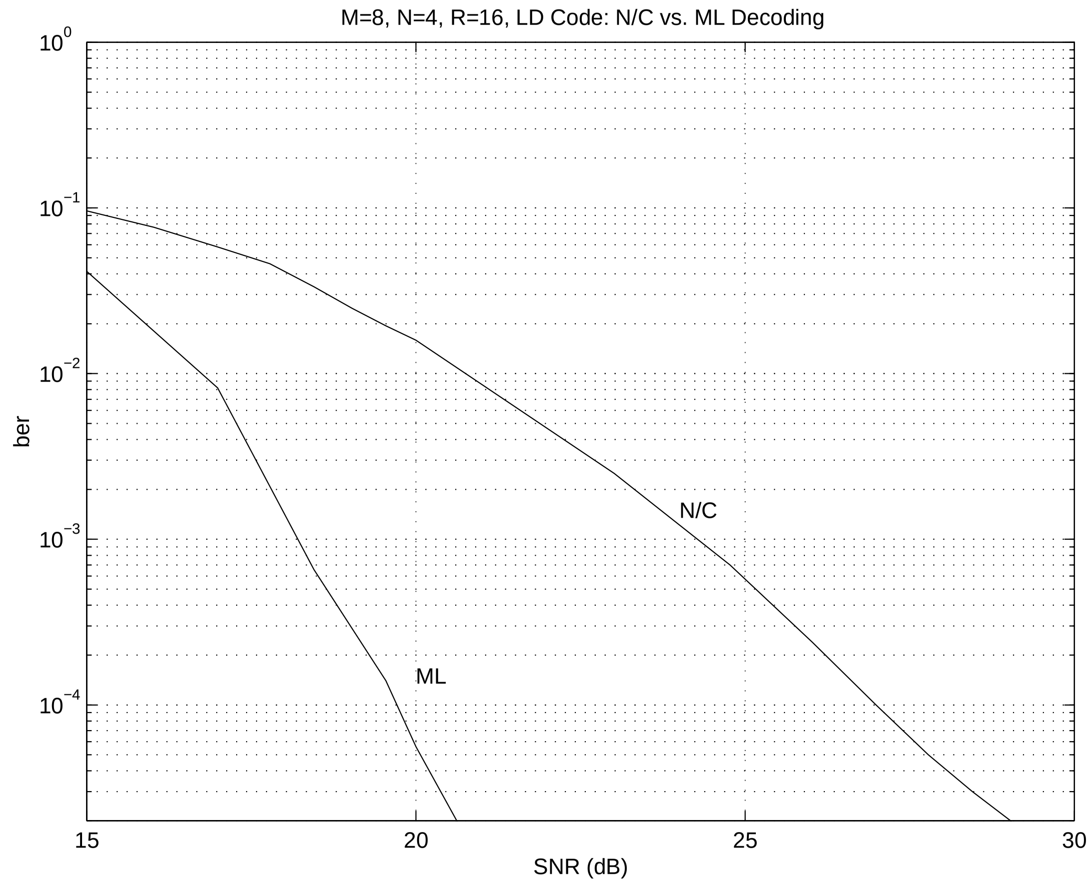


Figure 4: Bit error performance of an optimal linear space-time code for $T = 8$, $M = 8$ and $N = 4$, at rate $R = 16$. Note that $L = 2^{16\cdot 8} \approx 3.4\times 10^{38}$. The ML complexity was roughly twice that of nulling/cancelling.

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? random matrices...
- coding theory (space-time codes)
  - how to achieve capacity? known channel codes, unknown channel codes, group representations, Cayley transforms...
- algorithms
  - how to do all the processing in real-time? sphere decoding, polynomial-time ML? average vs. worst-case compelxity, equalization and frequency-selective channels...