Efficient near-ML Decoding via Statistical Pruning [1]

Radhika Gowaikar, Babak Hassibi
Dept. of Electrical Engineering,
California Institute of Technology, CA 91125
{gowaikar, hassibi}@caltech.edu

Abstract — **Maximum-likelihood (ML) decoding often reduces to finding the closest (skewed) lattice point in N-dimensions to a given point $x \in \mathcal{C}^N$. Sphere decoding is an algorithm that does this. We modify the sphere decoder to reduce the computational complexity of decoding while maintaining near-ML performance.**

I. System Model

We assume a discrete-time, block-fading, multiple-antenna channel model with N transmit and N receive antennas, where the channel is known to the receiver. If $\mathcal{S}$ is the signal space, the transmitted signal $\tilde{s} \in \mathcal{S}^{N\times 1}$ and the received signal $x \in \mathcal{C}^{N\times 1}$ are related by $x = \sigma_h H\tilde{s} + v$ where $H \in \mathcal{C}^{N\times N}$ is the known channel matrix and $v \in \mathcal{C}^{N\times 1}$ is the additive noise vector. Both are comprised of i.i.d. complex-Gaussian entries $\mathcal{CN}(0,1)$. σ_h determines the SNR. Under these conditions the ML criterion requires us to find $s \in \mathcal{S}^{N\times 1}$ that minimizes $\|x - Hs\|^2$.

II. Sphere Decoder

The sphere decoder finds lattice points in a hypersphere of radius r centered at x, i.e., it finds all $s \in \mathcal{S}^{N\times 1}$ that satisfy $r^2 \geq \|x - Hs\|^2$. For this, we decompose H as $H = QR$ where Q is unitary and R is upper triangular with positive diagonal. (Both are $N \times N$.) Then $\|x - Hs\|^2 = \|Q^*x - Rs\|^2$. Define $y' = Q^*x - Rs$ and $\lambda_i = |y'_{N-i+1}|^2$ for $i = 1, 2, \ldots, N$. We need to solve $\lambda_1 + \lambda_2 + \ldots + \lambda_N \leq r^2$. This is done by solving successively for $\lambda_1 \leq r^2; \lambda_1 + \lambda_2 \leq r^2; \ldots; \lambda_1 + \lambda_2 + \cdots + \lambda_N \leq r^2$. This can be done because the first condition gives an interval for s_N, whereas for any pre-determined $s_N, \ldots, s_{N-i+2}$, the i-th condition gives an interval for s_{N-i+1}. While the sphere decoder avoids exhaustive search it does incur very high computational complexities for very large N [1]. This happens because to have a high probability of finding at least one point in the hypersphere, r has to be proportional to N and a very large fraction of the points is retained in the early dimensions.

III. Statistical Pruning

To prune the search space right from the smaller dimensions, we modify the sphere decoder. We determine a schedule of radii $r_1 \leq r_2 \leq \ldots \leq r_N$ and solve for $\lambda_1 \leq r_1^2; \lambda_1 + \lambda_2 \leq r_2^2; \ldots; \lambda_1 + \lambda_2 + \cdots + \lambda_N \leq r_N^2$. Call this region $\mathcal{D}$. Since $\mathcal{D}$ is not hyperspherical, this is not exact ML decoding. If ϵ is the probability that the transmitted vector is not in $\mathcal{D}$, a loose upper bound on the probability of error is $P_e \leq P_e^{ML} + \epsilon$. The quantity ϵ can be determined exactly in terms of the r_is and so the r_is can be chosen to make ϵ as small as desired to ensure near-ML performance.

[1] This work was supported in part by the National Science Foundation under grant no. CCR-0133818, by the office of Naval Research under grant no. N00014-02-1-0578, and by Caltech's Lee Center for Advanced Networking.

IV. Results

The expected computational complexity C is given by $\sum_{k=1}^{N}$(expected # of points in $\mathcal{D}_k$) $\cdot$ (flops/point) where $\mathcal{D}_k$ is the restriction of $\mathcal{D}$ to the k-th dimension. The expected # of points in $\mathcal{D}_k$ is given by $\sum_{s^k \in \mathcal{S}^{k\times 1}} P(s^k \in \mathcal{D}_k)$ and flops/point $= 8k + 32$ for the k-th dimension. An upper bound on $P(s^k \in \mathcal{D}_k)$ leads to

$$C \leq \sum_{k=1}^{N}(8k+32)\sum_{l=0}^{2k}\binom{2k}{l}\Gamma\left(\frac{r_k^2}{1+\sigma_h^2 l}, k\right) \quad (1)$$

for QPSK constellations. Here $\Gamma(x,a) = \int_0^x \frac{e^{-t}}{\Gamma(a)} t^{a-1} dt$. An approximation to $P(s^k \in \mathcal{D}_k)$ leads to

$$C \approx \sum_{k=1}^{N}(8k+32)\sum_{s^k \in \mathcal{S}^{k\times 1}}\prod_{j=1}^{k}\min(1, \frac{X_k}{2(1+\sigma_h^2\|s^j - \tilde{s}^j\|^2})\quad (2)$$

where the X_is depend on $s^i - \tilde{s}^i$ and can be obtained recursively. This is computed efficiently with Monte Carlo simulations.

V. Simulations

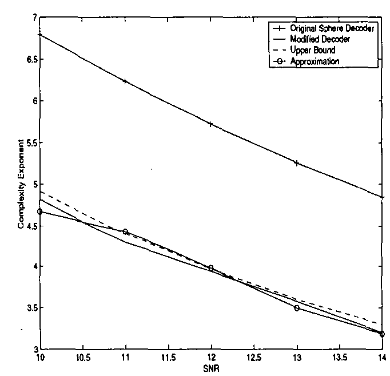

Figure 1: Complexity Exponent for N=50 with QPSK

Fig. (1) shows the complexity exponent ($\log C/\log N$) for $N = 50$ with the sphere decoder as well as the modified algorithm. The approximation and the upper bound are also plotted. A linear schedule of radii $r_i^2 = (\delta \log N + i)$ with δ chosen to make $\epsilon = 0.01$ was used. A computational savings of $50^2 = 2500$ is observed.

VI. Conclusions

The algorithm reduces decoding complexity by exploiting the statistics of the problem. Performance can be made arbitrarily close to ML by choice of ϵ.

References

[1] B. Hassibi and H. Vikalo, "The complexity of sphere decoding. pt. I Expected Complexity ," *Submitted to IEEE Trans. Sig. Proc.*; 2003

0-7803-7728-1/03/$17.00 ©2003 IEEE.