Improved Methods for Search Radius Estimation in Sphere Detection Based MIMO Receivers

Patrick Marsch, Ernesto Zimmermann, Gerhard Fettweis
Vodafone Chair Mobile Communications Systems
Department of Electrical Engineering and Information Technology
D-01062 Dresden, Germany
Email: {marsch,zimmere,fettweis}@ifn.et.tu-dresden.de

***Abstract*— Recently, communication systems with multiple transmission and reception antennas (MIMO) have been introduced and proven to be suitable for achieving a high spectral efficiency. Assuming full channel knowledge at the receiver, so-called sphere detectors perform a tree search and generate multiple hypotheses about the transmitted symbol, from which soft output information can be derived for each bit. Most schemes constrain the search to a certain radius, estimated from the desired number of hypotheses and statistical channel properties. However, the actual number of found hypotheses in this case still deviates strongly from the desired quantity.**

In this paper, we introduce a new algorithm that determines a more precise search radius by considering the current channel realization and received signal. Simulation results show an improved performance, a strong reduction of the deviation in hypotheses quantity, and thus a low variance in complexity, which is very important for practical implementation.

I. Introduction

In modern mobile communications systems, it is essential to use bandwidth very efficiently, i.e. to transmit many data bits per channel access. Existing modulation schemes are limited due to power constraints and the achievable signal-to-noise ratio (SNR). An alternative are systems with multiple transmission and reception antennas (MIMO) that use spatial diversity for a parallel data transmission, achieving a higher spectral efficiency for the same SNR.

A. System Model

A linear channel with N_T transmission and N_R reception antennas can be defined by a complex matrix $\mathbf{C}_{N_R \times N_T}$. We assume the channel to be fast Raleigh fading and ergodic, such that consecutive channel accesses observe a quasi-static channel, but a high number of accesses reveals the statistical properties of the channel, i.e. it is passive, and all elements are independent, identically distributed (i.i.d) Gaussian random variables with mean zero and $E\{|C_{ij}|^2\} = 1$. For later detection algorithms, we also introduce a channel representation with real elements

$$\mathbf{H}_{M \times L} = \begin{bmatrix} \text{Re}\{\mathbf{C}\} & -\text{Im}\{\mathbf{C}\} \\ \text{Im}\{\mathbf{C}\} & \text{Re}\{\mathbf{C}\} \end{bmatrix}$$

with $M = 2N_R$ and $L = 2N_T$. A transmission is stated as

$$\mathbf{y} = \mathbf{Hs} + \mathbf{n} \tag{1}$$

where $\mathbf{s} \in S$ is one of Q^L possible discrete signal vectors transmitted by the N_T antennas during one channel access. In this paper, we use $Q = 4$ or $Q = 8$ for 16-QAM or 64-QAM modulation, respectively. Factor a is chosen so that $E\left\{s_l^2\right\} = 1$, and $\mathbf{n}$ represents additive white Gaussian noise (AWGN) at the receiver, defined as real i.i.d distributed Gaussian random variables with zero mean and $E\left\{n_i^2\right\} = \sigma^2 = \frac{N_0}{2}$. We can calculate the SNR, based on the code rate R, as

$$\text{SNR [dB]} = 10 \cdot \log_{10} \frac{E\{E_b\}}{E\{N_0\}} = 10 \cdot \log_{10} \frac{1}{\log_2 Q \cdot R \cdot N_0}$$

B. The Integer Least Squares Problem

A common detection problem is to determine the symbol that has been transmitted with the highest *a posteriori* probability, given the received signal. If all symbols are equiprobable, we can use Bayes' theorem to rewrite this problem as a search for the *maximum-likelihood* (ML) symbol, i.e.

$$\hat{\mathbf{s}}_{ML} = \arg\max_{\mathbf{s} \in S} p(\mathbf{s}|\mathbf{y}) \equiv \arg\max_{\mathbf{s} \in S} p(\mathbf{y}|\mathbf{s})$$

From our system model in section I-A, we can derive

$$p(\mathbf{y}|\mathbf{s}) = \frac{1}{(2\pi\sigma^2)^{\frac{M}{2}}} \exp\left(-\frac{\|\mathbf{y} - \mathbf{Hs}\|^2}{2\sigma^2}\right)$$

and conclude that finding the ML symbol is equivalent to solving the so-called *integer least-squares problem* [1]

$$\hat{\mathbf{s}}_{ML} = \arg\min_{\mathbf{s} \in S} \|\mathbf{y} - \mathbf{Hs}\|^2 \tag{2}$$

II. Review on Sphere Detection

A solution for equation (2) can be approximated by *linear equalization* [2]. However, the result is not optimal any more after discretization. So-called *sphere detectors* alleviate the problem by discretizing every real signal component (i.e. every signal *layer*), before proceeding with the next component.

A. QR-decomposition

Under the assumption $N_R \geq N_T$, we can transform the channel matrix by performing a so-called QR-decomposition

$$\mathbf{H} = \mathbf{Q} \begin{bmatrix} \mathbf{R} \\ \mathbf{O} \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{bmatrix} \begin{bmatrix} \mathbf{R} \\ \mathbf{O} \end{bmatrix}$$

where $\mathbf{Q}_{1,M\times L}$ and $\mathbf{Q}_{2,M\times(M-L)}$ are orthonormal matrices, $\mathbf{R}_{L\times L}$ is upper triangular, and $\mathbf{O}_{(M-L)\times L}$ is a zero matrix. We can then rewrite equation (2) to [3]

$$\begin{aligned}\hat{\mathbf{s}}_{ML} &= \arg\min_{\mathbf{s}\in S} \left\| \left[\begin{array}{c} \mathbf{Q}_1^T \\ \mathbf{Q}_2^T \end{array} \right] \mathbf{y} - \left[\begin{array}{c} \mathbf{R} \\ \mathbf{O} \end{array} \right] \mathbf{s} \right\|^2 \\ &= \arg\min_{\mathbf{s}\in S} \left\| \mathbf{Q}_1^T \mathbf{y} - \mathbf{R}\mathbf{s} \right\|^2 + \left\| \mathbf{Q}_2^T \mathbf{y} \right\|^2 \\ &\equiv \arg\min_{\mathbf{s}\in S} \left\| \mathbf{y}' - \mathbf{R}\mathbf{s} \right\|^2 \end{aligned} \tag{3}$$

where the term $\left\| \mathbf{Q}_2^T \mathbf{y} \right\|^2$ is omitted, as it has no impact on the *arg min* operation. The upper triangular form of matrix $\mathbf{R}$ now allows us to iteratively calculate estimates for the originally transmitted signals s_L, s_{L-1}, ..., s_1 as

$$\tilde{s}_l = \frac{y'_l - \sum_{m=l+1}^{L} r_{lm} \cdot \hat{s}_m}{r_{ll}} \tag{4}$$

and perform a discretization $\hat{s}_l = \lfloor \tilde{s}_l \rceil$, then used for calculating $\tilde{s}_{l-1}$. After processing all L layers we obtain a full signal vector $\hat{\mathbf{s}}$, referred to as a *candidate*. Alternative discrete signals can be chosen in each layer to create a *search tree* with multiple candidates, i.e. hypotheses on transmitted symbols. We denote the set of candidates as C.

B. Limiting Sphere Searches through Metrics

From equations (3) and (4), we can derive that [3]

$$\begin{aligned}\|\mathbf{y} - \mathbf{H}\hat{\mathbf{s}}\|^2 &= \left\| \mathbf{Q}_1^T \mathbf{y} - \mathbf{R}\hat{\mathbf{s}} \right\|^2 + \left\| \mathbf{Q}_2^T \mathbf{y} \right\|^2 \\ &= \sum_{l=1}^{L} \left(y'_l - \sum_{m=l}^{L} r_{ml} \hat{s}_m \right)^2 + K \\ &= \sum_{l=1}^{L} (\tilde{s}_l - \hat{s}_l)^2 \cdot r_{ll}^2 + K \end{aligned} \tag{5}$$

where $K = 0$ if the number of transmission and reception antennas is equal, which we will assume for simplification. Hence, the spacing $\Delta_l = |\tilde{s}_l - \hat{s}_l|$ between the calculated and discretized signal in every layer is related to the Euclidian distance of the received signal $\mathbf{y}$ to the projection $\mathbf{H}\hat{\mathbf{s}}$ of the obtained candidate. This can be used as a metric to evaluate partial or completed paths in the search tree and to limit the search to those candidates within a predefined Euclidian distance r, by assuring in each detection step that

$$M(\hat{\mathbf{s}})_l^L = \sum_{m=l}^{L} (\hat{s}_m - \tilde{s}_m)^2 \cdot r_{mm}^2 \leq \|\mathbf{y} - \mathbf{H}\hat{\mathbf{s}}\|^2 \leq r^2$$

C. Basic Sphere Search Algorithms

Most sphere detectors use either the *Fincke-Pohst* [4], [5] or *Schnorr-Euchner* [6] algorithm to perform a *depth-first* search and find all candidates within an estimated radius r, so that a desired number of candidates is found on average. Alternatively, the search radius can be continuously reduced to the metric of the N-th best candidate. We then obtain at least the N best candidates with a reduced effort, especially if Schnorr-Euchner is used, as it outputs candidates with low metrics first and thus quickly reduces the radius. Both methods require the estimation of an initial search radius, where fixed radius schemes are obviously more sensitive to a bad estimation than those using radius reduction.

Alternatively, a *List-Sequential Sphere Detector* (LISS) [7] performs a *breadth-first* search by sorting all tree nodes, and always processing that with the lowest metric first, regardless of its respective signal layer. Candidates are automatically found in the order of an ascending metric, so that the search can be stopped after a sufficient output of candidates, and does not require a search radius. However, the downside is a high effort for constantly ordering tree nodes.

For coded transmission, a detector has to provide *soft output*, i.e. *a posteriori* probabilities about each transmitted bit, which can be derived from the set of candidates as so-called *log-likelihood ratios* (LLR) [8].

III. Statistical Radius Estimation

In general, the problem of estimating a search radius is equivalent to predicting the number of candidates for a given radius. In this section, we will review several known approaches that are based on statistical properties of the *average* channel and transmitted symbol, and add some novel ideas in order to improve or simplify these approaches.

A. Analyzing the System Noise

A search radius for finding only the ML point can be derived from the noise term in equation (1). Here, $\mathbf{s}$ is the originally transmitted signal, and we can see that the distance $d^2 = \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 = \|\mathbf{n}\|^2$ is the sum of M squared Gaussian distributions, hence a chi-square random variable ξ with M degrees of freedom, i.e. $d^2 = \sigma^2 \xi$. Thus, [11] suggests to set

$$r^2 = k \cdot E\left\{d^2\right\} = k \cdot M\sigma^2$$

where $k \geq 1$ is chosen manually so that the search region contains $\mathbf{H}\mathbf{s}$ with high probability. In [1], a target probability is defined, e.g. $\rho = 0.99$, the pdf of ξ is integrated as

$$\int_0^{\Xi} \frac{\xi^{\frac{M}{2}-1}}{2^{\frac{M}{2}} \cdot \Gamma(\frac{M}{2})} \cdot e^{-\frac{\xi}{2}} d\xi = \rho$$

and solved for $r = \sigma \cdot \sqrt{\Xi}$. This approach can strongly reduce search complexity if the noise level is low compared to the signal amplitude after transmission. However, there remains a non negligible probability that no candidate is found at all.

B. Analyzing the Signal Lattice

If a higher number of candidates is desired, we have to analyze the geometry of the set of all possibly transmitted signals as they appear at the receiver, also referred to as the *signal lattice*. Some papers (e.g. [1], [12]) estimate the number of found candidates by dividing the volume of an

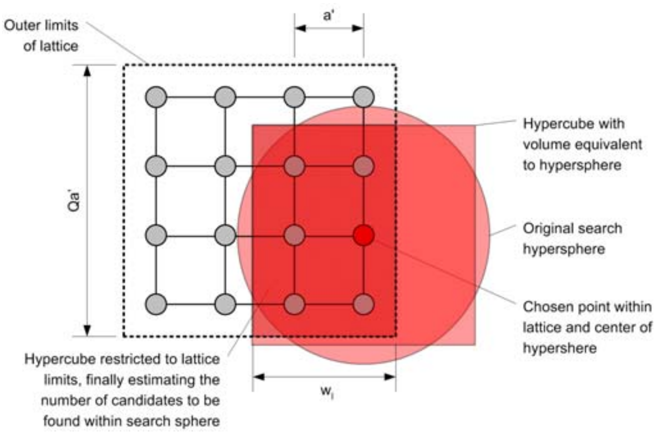

Fig. 1. Restricting a hypercube to the limits of the signal lattice.

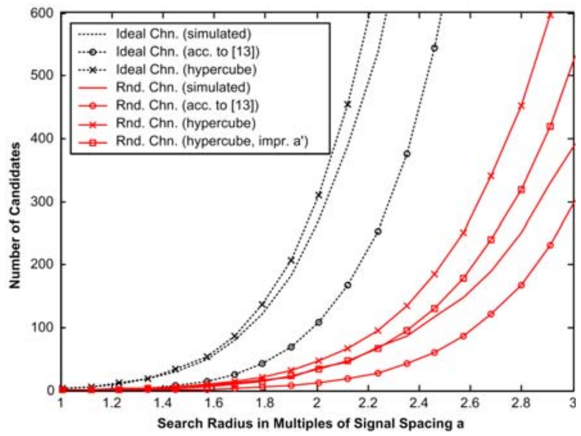

Fig. 2. Average number of candidates within search radius, simulated and predicted using statistical approaches.

L-dimensional search sphere of radius r around the received point through the volume of one candidate, i.e.

$$N(r) \approx \frac{V_L^\circ(r)}{V_c^\Box} = \frac{\pi^{L/2} \cdot r^L}{\Gamma(L/2+1) \cdot a^L \,|\det \mathbf{H}|} \tag{6}$$

where $V_c^\Box = a^L \cdot |\det \mathbf{H}|$ expresses the volume in which one candidate resides, if we assume the received signal lattice to have the same cubic shape as the transmitted lattice. A suitable search radius for finding N candidates could now be derived by solving (6) for r. However, this does not consider the finiteness of the signal lattice, e.g. the number of candidates around a point near the lattice border will be less. Hence, [13] suggests to divide the expected quantity by two for each layer in which the signal is on the edge of the lattice, i.e. where $s_l \in S_{edge} = \{-(Q-1)/2, (Q-1)/2\}$, and update (6) to

$$N(r) \approx \sum_{n=0}^{L} \frac{N_L^{[n]}}{Q^L} \cdot \frac{V_L^\circ(r)}{V_c^\Box} = \left(\frac{3}{4}\right)^L \cdot \frac{V_L^\circ(r)}{a^L \cdot |\det \mathbf{H}|}$$

where $N_L^{[n]}$ is the number of possible symbols that are positioned on an edge in n dimensions, calculated as

$$N_L^{[n]} = (Q-2)^{L-n} \cdot 2^n \begin{pmatrix} L \\ n \end{pmatrix}$$

C. Hypercube Restriction

A better estimation requires us to consider the lattice finiteness for all possibly transmitted symbols. A mathematically complete approach in [1] does this and finds a complex closed form for the number of nodes and candidates in the search tree. However, we suggest a strongly simplified approach that approximates hyperspheres by L-dimensional *hypercubes* with an equivalent volume and the side length

$$w = r \sqrt[L]{\frac{\pi^{L/2}}{\Gamma(L/2+1)}}$$

This hypercube can then be restricted to the lattice borders in each dimension. For 16-QAM modulation, the $Q = 4$ possible signals in every layer can be divided into those on the lattice edge ($s_l \in S_{ed}$) and those inside ($s_l \in S_{in}$), and the hypercube side length w_l in every dimension can be set to

$$w_l(w) = \begin{cases} w_{ed} = \min\left(\frac{7}{2}a', \frac{w}{2}\right) + \min\left(\frac{1}{2}a', \frac{w}{2}\right) & s_l \in S_{ed} \\ w_{in} = \min\left(\frac{5}{2}a', \frac{w}{2}\right) + \min\left(\frac{3}{2}a', \frac{w}{2}\right) & \text{otherwise} \end{cases}$$

as illustrated in figure 1. Here, $a' = a \cdot \sqrt[L]{|\det \mathbf{H}|}$ is the average signal amplitude after transmission. We define the limits of the lattice to lie $a'/2$ outside the outer symbols, so that $w_l(\infty) = 4a'$, corresponding to $Q = 4$ possible signals, and state

$$N(r) \approx \frac{\sum_{n=0}^{L} N_L^{[n]} \cdot (w_{ed})^n \cdot (w_{in})^{L-n}}{Q^L \cdot V_c^\Box} \tag{7}$$

D. Channel Preprocessing

Many sphere detectors use channel preprocessing to reduce the complexity of searches based on radius reduction. E.g. a *sorted QR-decomposition* (SQRD) [14] orders the matrix columns by their SNR, so that more reliable layers are detected first. It is also reasonable to consider noise and achieve a *minimum mean square error* (MMSE) by extending the channel matrix prior to decomposition [14], [1]. This reduces search complexity, but leads to slightly incorrect metrics and false ML candidates, as equation (5) is then not valid any more. An MMSE channel extension also facilitates radius estimation, but we have to consider that the detector now left-multiplies $\mathbf{y}$ with $\bar{\mathbf{Q}}_1^T$ and $\left|\det \bar{\mathbf{Q}}_1\right| \leq 1$. The amplitude a' is thus attenuated before detection, and should now be calculated as

$$a' = a \cdot \frac{|\det \mathbf{H}|^{\frac{1}{L}}}{\left(|\det \mathbf{H}|^{\frac{2}{L}} + \sigma^2\right)^{\frac{1}{2}}} \tag{8}$$

We generally use column ordering, either with MMSE channel extension (*MMSE-SQRD*) or without (*ZF-SQRD*).

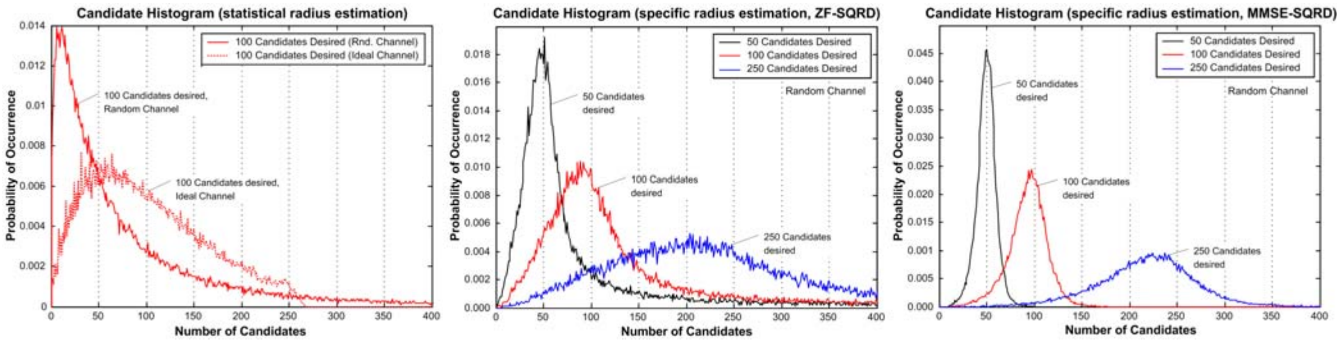

Fig. 3. Observed number of candidates in a sphere around the received signal.

E. Considering Lattice Distortion

We have so far assumed that the received lattice has a cubic shape. However, the channel will cause correlation between signal layers and thus distort the lattice, so that the volume of one candidate will be less than $V_c^{\square} = a^L \cdot |\det \mathbf{H}|$. In [13] this is compensated by a factor μ, which is chosen according to the lattice *acuteness*. As a more precise solution, we suggest to exploit the fact that the lattice is orthogonalized again during QR-decomposition, and estimate the candidate volume as

$$V_c^{\square} = \prod_{l=1}^{L} \bar{r}_{ll} \qquad \text{and use} \qquad a'' = a' \cdot \sqrt[L]{\prod_{l=1}^{L} \bar{r}_{ll}}$$

where $\bar{r}_{ll}$ is the average diagonal entry in $\mathbf{R}$ for many channel realizations. The values $\bar{r}_{ll}$ are difficult to state in a closed form, as they depend on the applied channel preprocessing and the average correlation between signal layers. Figure 2 shows the simulated and calculated number of candidates obtained by the different statistical schemes as a function of search radius. The hypercube restriction scheme using a'' predicts the number of candidates very well, especially for small radii. As a reference, the simulation and estimation of an ideal channel, i.e. where $\mathbf{H}$ is the identity matrix, is also shown.

IV. Specific Search Radius Estimation

We have so far observed an *average* channel, and the results were averaged out over all possibly transmitted symbols. However, the left histogram in figure 3 shows that even if a search radius is determined to precisely find 100 candidates on average, the actual quantity found for each channel realization and received signal varies strongly, so that the complexity and quality of output are difficult to predict. Thus, we will now introduce a novel scheme that determines a specific $N(r|\mathbf{H}, \mathbf{y})$ for a given channel and received signal. The algorithm in table I estimates the transmitted signal in each layer, based on the idea of sphere detection from section II-A, but without discretization, and the hypercube restriction technique from equation (7) is used to estimate the candidate quantity. Note that the algorithm can be used for any modulation or pre-processing scheme. However, it can not be inverted, i.e. in order to estimate a radius for a given number of candidates, a detector will have to make an initial guess, calculate the corresponding number of candidates and adjust the radius iteratively. Experiments have shown that 8 such iterations, corresponding to a choice of $2^8 = 256$ different radius values, enable a sufficiently precise radius estimation.

NumCandidates()

Input: Upper-triangular $\mathbf{R}_{L\times L}$, signal $\mathbf{y}'_{L\times 1}$, amplitude a, radius r
Output: Estimated number of candidates N

(1)	$w := r \cdot \sqrt[L]{\frac{\pi^{L/2}}{\Gamma(L/2+1)}}$	equivalent radius of a hypercube
(2)	$V := 1$	initially set volume to 1
(3)	$l := L$	start with upmost layer
(4)	while $(l \leq L)$ {	go through all layers
(5)	$d_{sum} := \sum_{m=l+1}^{L} r_{lm} \cdot \hat{s}_m$	interference of previous layers
(6)	$\tilde{s}_l := (y'_l - d_{sum}) / r_{ll}$	estimation of signal in this layer
(7)	$w_l := \min(Qa/2, \tilde{s}_l + w/2r_{ll})$ $- \max(-Qa/2, \tilde{s}_l - w/2r_{ll})$	width of hypercube in this layer
(8)	$V := V \cdot w_l$	update hypercube volume
(9)	}	
(10)	$N := V/a^L$	divide volume thr. cand. volume

TABLE I

Estimating the number of candidates within a radius r.

The two right diagrams in figure 3 show histograms of the number of candidates found with the specific estimation scheme. We can see that especially when MMSE channel extension is used, the variance of the number of candidates is strongly reduced compared to the statistical estimation scheme.

V. Simulation

We will compare the complexity and performance of different search schemes using either statistical or specific radius estimation, based on a 4x4 MIMO system, matrix ordering and MMSE channel extension (*MMSE-SQRD*).

A. Setup

All simulations are based on Turbo coded transmission, using a standard PCCC with $(7_R, 5)$ constituent convolutional codes. The block length is 8920 bits, and a code rate $R = 1/2$ is achieved by alternately puncturing parity bits at the output of the constituent encoders. At the receiver, one initial detection process is succeeded by 8 internal decoder iterations. All schemes except the LISS approach use the Schnorr-Euchner

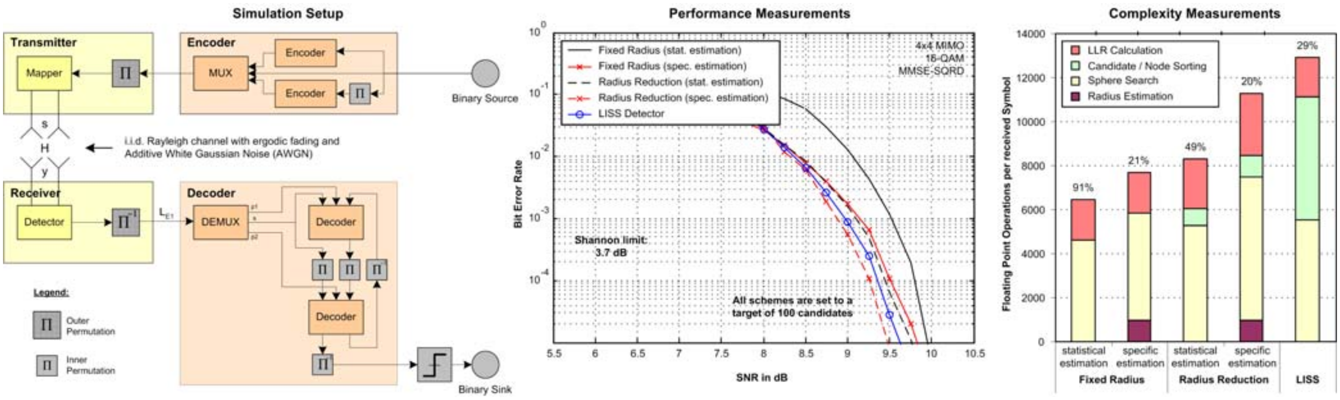

Fig. 4. Simulation setup, performance and complexity measurements, for 16-QAM modulation and MMSE-SQRD preprocessing.

algorithm. The fixed radius schemes are set to find exactly 100 candidates on average, those using radius reduction determine an initial radius containing 200 candidates on average, yielding about 125 or 150 found candidates on average, for the statistical or specific estimation scheme, respectively.

B. Performance

Figure 4 shows the setup and the bit error rate of the schemes as a function of SNR. Differences in performance are very small, and often within simulation inaccuracy. However, a search with a fixed radius using statistical radius estimation obviously performs worse than its counterpart using specific estimation, as the variance in the number of candidates strongly deteriorates the performance of the decoder. The difference between the two radius reduction schemes is smaller, as these are less sensitive to the choice of the initial search radius, as stated in section II-C. In general, the radius reduction schemes perform slightly better than those using a fixed radius, as the average number of candidates is higher. The LISS approach always finds exactly 100 candidates, which leads to a better performance than that of the two fixed radius approaches, as the soft output supplied to the decoder is of a constant quality.

C. Complexity

The main advantage of our new radius estimation scheme can be seen in the right part of figure 4, showing the average number of floating point operations per detected symbol. This quantity might not be representative for hardware implementation, but it still allows a valid comparison of all schemes, as the type of operations required is similar. Simulations have been performed at an SNR of 8dB, and the numbers above the bars show the standard deviation of complexity. For fixed radius searches, the average complexity is identical, except for the additional effort for the radius estimation algorithm itself. However, the deviation is reduced from 91% to 21%. The same effect is observable for the radius reduction schemes, though here the deviation is less for both the old and new scheme, and the new scheme appears more complex due to a higher average number of candidates. The LISS detector shows a high effort for node ordering and is thus the most complex.

VI. Conclusions

Our new specific radius estimation scheme strongly decreases the standard deviation of the number of found candidates and the corresponding search complexity, also leading to a better performance than approaches based on statistical radius estimation. Especially a fixed radius search using our new algorithm shows a predictable complexity and a good ratio of performance over complexity.

References

[1] B. Hassibi and H. Vikalo, *On the expected Complexity of Sphere Decoding*, Record of the Thirty-Fifth Asilomar Conference on Signals, Systems and Computers (2001), 1051–1055.

[2] J. G. Proakis, *Digital communications*, 4 ed., McGraw-Hill, 2000.

[3] M. O. Damen, H. El Gamal, and G. Caire, *On Maximum-Likelihood Detection and the Search for the closest Lattice Point*, IEEE Transactions on Information Theory **49** (2003), no. 10, 2389–2402.

[4] M. Pohst, *On the Computation of Lattice Vectors of minimal Length, successiv Minima and reduced Bases with Applications*, ACM SIGSAM Bulletin **15** (1981), 37–44.

[5] U. Fincke and M. Pohst, *Improved Methods for calculating Vectors of short Length in a Lattice, including a Complexity Analysis*, Math. of Comput. **44** (1985), 463–471.

[6] C. P. Schnorr and M. Euchner, *Lattice Basis Reduction: Improved practical Algorithms and solving Subset Sum Problems*, Math. Prog. **66** (1994), 181–191.

[7] S. Bäro, J. Hagenauer, and M. Witzke, *Iterative Detection of MIMO Transmission using a List-Sequential (LISS) Detector*, IEEE International Conference on Communications **4** (2003), 2653–2657.

[8] J. Hagenauer, E. Offer and L. Papke, *Iterative decoding of binary and block convolutional codes*, IEEE Transactions on Information Theory **42** (1996) 429–445.

[9] A. J. Viterbi, *Very Low Rate Convolutional Codes for Maximum Theoretical Performance of Spread-Spectrum Multiple-Access Channels*, IEEE JSAC **8** (1990), no. 4, 641–649.

[10] R. Kohno et al., *Combination of an Adaptive Array Antenna and a Canceller of Interference for Direct-Sequence Spread-Spectrum Multiple-Access System*, IEEE JSAC **8** (1990), no. 4, 675–682.

[11] B. M. Hochwald and S. ten Brink, *Achieving Near-Capacity on a Multiple-Antenna Channel*, IEEE Transactions on Communications **51** (2003), no. 3, 389–399.

[12] E. Agrell and T. Eriksson, *Closest Point Search in Lattices*, IEEE Transactions on Information Theory **48** (2002), no. 8, 2201–2214.

[13] J. Boutros, N. Gresset, L. Brunel, and M. Fossorier, *Soft-Input Soft-Output Lattice Sphere Decoder for Linear Channels*, IEEE Global Telecommunications Conference **22** (2003), no. 1, 1583–1587.

[14] D. Wübben, R. Böhnke, V. Kühn, and K.-D. Kammeyer, *MMSE Extension of V-BLAST based on Sorted QR Decomposition*, IEEE Semiannual Vehicular Technology Conference **1** (2003), 508–512.