A LOW-COMPLEXITY SOFT-MIMO DETECTOR BASED ON THE FIXED-COMPLEXITY SPHERE DECODER

Luis G. Barbero, T. Ratnarajah and Colin Cowan

ECIT, Queen's University of Belfast
Queen's Island, Belfast, BT3 9DT, UK
e-mail: {l.barbero, t.ratnarajah, c.cowan}@ecit.qub.ac.uk

ABSTRACT

This paper presents a soft-output version of the fixed-complexity sphere decoder (FSD) previously proposed for uncoded multiple input-multiple output (MIMO) detection. Thus, the soft-FSD (SFSD) can be used in turbo-MIMO systems to exchange extrinsic soft-information with the outer decoder. For that purpose, the SFSD generates a list of candidates that approximates that of the list sphere decoder (LSD) while containing information about all the possible bit values, removing the need for clipping. In addition, it overcomes the two problems of the LSD: its variable complexity and the sequential nature of its tree search. Simulation results show that the SFSD can be used to approximate the performance of the LSD while having a considerably lower and fixed complexity, making the algorithm suitable for hardware implementation.

***Index Terms*—** MIMO, soft-detection, sphere decoder, turbo decoding.

1. INTRODUCTION

It has been recently shown that the capacity of multiple input-multiple output (MIMO) channels can be approached using a turbo-MIMO scheme that combines a spatially-multiplexed MIMO stage and an outer code with an interleaver operation in between [1]. At the receiver, a soft-MIMO detector is required to generate soft-information that can be used by the outer decoder. The list sphere decoder (LSD) is considered the most promising algorithm for soft-MIMO detection, reducing the prohibitive complexity of the maximum likelihood detector (MLD) for large number of antennas and/or constellation orders [1]. A large number of alternatives have been proposed to improve the performance and, in some cases, reduce the complexity of the LSD [2] -[6]. However, all those algorithms suffer from the same problems as the original LSD, namely, its variable complexity, depending on the noise level and the channel conditions, and the sequential nature of its tree search.

In this paper, we present a soft-output version of the fixed-complexity sphere decoder (FSD) previously proposed for uncoded MIMO detection [7]. The soft-FSD (SFSD) can be used to approximate the performance of the LSD while having a fixed complexity. This makes the algorithm especially suitable for a parallel and fully-pipelined hardware implementation, as previously shown for the FSD [8]. The SFSD applies the concepts of *bit-negating* [6] and *path-augmentation* [9] but in the context of the FSD. Previously proposed fixed-complexity soft-MIMO detectors with a similar level of performance are based on the M-algorithm [10],[11] which is shown here to have a higher complexity than the SFSD. Alternatives to reduce its complexity resort back to a variable complexity [10],[11].

This work was supported by the UK Engineering and Physical Sciences Research Council under grant number EP/C004132/1.

2. TURBO-MIMO SYSTEM MODEL

We consider the turbo-MIMO system presented in [1] for the transmission of frames of K_u bits. It consists of M transmit and N receive antennas, denoted as $M\times N$, where $N\geq M$. At the transmitter, the K_u information bits are encoded using an off-the-shelf convolutional or turbo code of rate R_c, where $K_\mathrm{u} = K_\mathrm{b}\cdot R_\mathrm{c}$. The K_b coded bits are then interleaved and the resulting bits, denoted as $\mathbf{b}$, are mapped to symbols taken independently from a a quadrature amplitude modulation (QAM) constellation $\mathcal{O}$ of P points, forming a sequence of $K_\mathrm{s} = K_\mathrm{b}/\log_2 P$ symbols. The sequence of symbols is then split into M substreams and blocks of $M\cdot K_\mathrm{ch}$ symbols are transmitted in each channel realization (i.e. block-Rayleigh fading channel). Therefore, a frame of K_b coded bits requires the transmission of $K_\mathrm{s}/(M\cdot K_\mathrm{ch})$ blocks of data. In this paper, $K_\mathrm{ch} > 1$ is assumed as opposed to [1], where $K_\mathrm{ch} = 1$. Assuming symbol-synchronous sampling at the receiver and ideal timing, the vector of received symbols $\mathbf{r}\in\mathbb{C}^{N\times 1}$ can be written as

$$\mathbf{r} = \mathbf{H}\mathbf{s} + \mathbf{v}\,,$$

where $\mathbf{s}\in\mathbb{C}^{M\times 1}$ denotes the vector of transmitted symbols with $\mathrm{E}[|s_i|^2] = 1/M$, for $1\leq i\leq M$, and $\mathbf{v}\in\mathbb{C}^{N\times 1}$ is the vector of independent complex Gaussian noise samples $v_i\sim\mathcal{CN}(0,\sigma^2)$, for $1\leq i\leq M$. The channel matrix $\mathbf{H}\in\mathbb{C}^{N\times M}$, assumed to be perfectly known at the receiver, has independent elements $h_{j,i}\sim\mathcal{CN}(0,1)$, for $1\leq j\leq N$ and $1\leq i\leq M$, representing the block-Rayleigh fading propagation environment. The set of all possible transmitted vectors form the M-dimensional complex transmit constellation $\mathcal{O}^M$.

At the receiver, the turbo-principle is applied, where extrinsic log-likelihood ratio (LLR) soft-information is exchanged iteratively between the soft-MIMO detector and the outer decoder with interleaving/deinterleaving operations in between until the desired performance is achieved [1]. Concentrating on the soft-MIMO detector, its task is to generate extrinsic LLR information about the interleaved bits $\mathbf{b}$, $L_\mathrm{E}(b_k|\mathbf{r})$, for $1\leq k\leq K_\mathrm{b}$, taking into account the channel observations $\mathbf{r}$ and the *a priori* LLR information, $L_\mathrm{A}(b_k)$, coming from the outer decoder. Given that the exact computation of $L_\mathrm{E}(b_k|\mathbf{r})$ has an exponential complexity $O(P^M)$ [1], a number of soft-MIMO detectors have been proposed to approximate $L_\mathrm{E}(b_k|\mathbf{r})$ with reduced complexity [1]-[6],[10],[11]. For the system under consideration, assuming that the bits b_k are statistically independent due to the interleaving operation and making use of the Max-log approximation $L_{\mathrm{E}}(b_k|\mathbf{r})$ can be approximated by [1]

$$L_{\mathrm{E}}(b_k|\mathbf{r}) \approx \frac{1}{2} \max_{\mathbf{b}\in\mathcal{L}\cap\mathbb{B}_{k,+1}} \left\{ \frac{-\|\mathbf{r}-\mathbf{Hs}\|^2}{\sigma^2/2} + \mathbf{b}_{[k]}^{\mathrm{T}}\mathbf{L}_{A[k]} \right\} - \frac{1}{2} \max_{\mathbf{b}\in\mathcal{L}\cap\mathbb{B}_{k,-1}} \left\{ \frac{-\|\mathbf{r}-\mathbf{Hs}\|^2}{\sigma^2/2} + \mathbf{b}_{[k]}^{\mathrm{T}}\mathbf{L}_{A[k]} \right\}, \quad (1)$$

for $1 \le k \le K_{\mathrm{b}}$, where, without loss of generality, $K_{\mathrm{b}} = M \cdot \log_2 P$ has been assumed to simplify the index notation. In (1), $\mathbf{b}_{[k]}$ denotes the subvector of $\mathbf{b}$ omitting b_k, $\mathbf{L}_{\mathrm{A}} = [L_{\mathrm{A}}(b_1), L_{\mathrm{A}}(b_2), \ldots, L_{\mathrm{A}}(b_{K_{\mathrm{b}}})]^{\mathrm{T}}$, $\mathbf{L}_{\mathrm{A}[k]}$ denotes the subvector of $\mathbf{L}_{\mathrm{A}}$ omitting $L_{\mathrm{A}}(b_k)$, $\mathbb{B}_{k,+1}$ and $\mathbb{B}_{k,-1}$ represent the sets of 2^{K_b-1} bit vectors $\mathbf{b}$ having $b_k = +1$ (logical '1') and $b_k = -1$ (logical '0'), respectively, $\mathcal{L}\cap\mathbb{B}_{k,+1}$ and $\mathcal{L}\cap\mathbb{B}_{k,-1}$ denote the subgroups of vectors of $\mathcal{L}$ that have $b_k = +1$ and $b_k = -1$, respectively, and $\mathbf{s} = \mathrm{map}(\mathbf{b})$[1], which represents the mapping of each group of $\log_2 P$ bits onto a QAM symbol. The subgroup of vectors or list of candidates $\mathcal{L} \subset \mathcal{O}^M$ depends on the specific soft-MIMO detector used and plays a key role in the performance and complexity of the turbo-MIMO receiver, given that $\|\mathbf{r}-\mathbf{Hs}\|^2$ needs to be computed for all $\mathbf{s} \in \mathcal{L}$.

3. SOFT-FSD

The SFSD presented in this paper is based on the FSD previously proposed for uncoded MIMO detection [7]. The FSD approximates the performance of the MLD combining a specific FSD channel matrix ordering with a search over a subset of the entire transmit constellation $\mathcal{S} \subset \mathcal{O}^M$. The process can be written as

$$\hat{\mathbf{s}}_{\mathrm{FSD}} = \arg\min_{\mathbf{s}\in\mathcal{S}} \|\mathbf{r}-\mathbf{Hs}\|^2 .$$

The FSD, analogously to the sphere decoder (SD), can be seen as a constrained tree search through a tree with M levels where P branches originate from each node [12]. The FSD performs a fixed two-stage constrained tree search, from $i = M$ to $i = 1$, depending on a parameter T. Initially, a full search is performed in the first T levels expanding all P branches per node. Secondly, a single search is performed in the remaining $M - T$ levels, expanding only one branch per node following the decision-feedback equalization (DFE) path. The M columns of $\mathbf{H}$ are ordered iteratively so that the signals with the *largest* and the *smallest* post-processing noise amplification are detected in the first T and the last $M - T$ levels, respectively [7]. In particular, for $N \ge M$, the FSD provides asymptotical maximum likelihood (ML) performance if $(N-M)(T+1)+(T+1)^2 > N$ and provides the same diversity as the MLD if $(N-M)(T+1)+(T+1)^2 = N$ [13].

It should be noted that the subset $\mathcal{S}$ could be directly used as the list of candidates in (1). However, the FSD focuses only on finding the best possible solution from a hard-output perspective. In the case of turbo-MIMO systems, the interest is not only in finding the ML solution, $\hat{\mathbf{s}}_{\mathrm{ML}}$, but also in obtaining a set of candidates around the ML solution with different bit values that can be used to calculate the extrinsic LLR information of $\mathbf{b}$. The SFSD approximates that set of candidates around the ML solution with fixed complexity, taking as a starting point the subset $\mathcal{S}$ of the FSD.

By its definition, $\mathcal{S}$ contains vectors where the bits corresponding to the signals detected in the first T levels take both possible values. However, that is not guaranteed for the bits corresponding to the signals detected in the last $M - T$ levels. For that reason, the SFSD creates a new subset $\tilde{\mathcal{S}} \subset \mathcal{O}^M$ to make sure that information about the two values of the bits corresponding to the signals detected in the last $M - T$ levels is present. The list of candidates provided by the SFSD is then generated as $\mathcal{L} = \mathcal{S} \cup \tilde{\mathcal{S}}$ where $\mathcal{S} \cap \tilde{\mathcal{S}} = \emptyset$ so that the transmitted vectors $\mathbf{s} \in \mathcal{S}$ are not reconsidered in $\tilde{\mathcal{S}}$. In addition, $\tilde{\mathcal{S}}$ is created so that it favors the transmitted vectors with the required bit values that are more likely to be closer to the FSD solution, in order to obtain an accurate approximation of $L_{\mathrm{E}}(b_k|\mathbf{r})$ in (1).

3.1. Soft-FSD Algorithm

The SFSD iteratively generates a subset $\tilde{\mathcal{S}}$ so that different levels of performance and complexity can be achieved at the receiver. As the number of iterations $1 \le N_{\mathrm{SFSD}} \le |\mathcal{S}|$ increase, the accuracy of (1) improves at the expense of a higher complexity. Let $\{\mathbf{s}^{(l)}\}$, for $1 \le l \le |\mathcal{S}|$, denote the elements of $\mathcal{S}$ such that $\|\mathbf{r}-\mathbf{Hs}^{(1)}\|^2 \le \|\mathbf{r}-\mathbf{Hs}^{(2)}\|^2 \le \ldots \le \|\mathbf{r}-\mathbf{Hs}^{(|\mathcal{S}|)}\|^2$ and $\mathbf{s}^{(1)} = \hat{\mathbf{s}}_{\mathrm{FSD}}$.

If only one iteration is performed, the SFSD selects the path in the tree that corresponds to $\mathbf{s}^{(1)} = [s_1^{(1)}, s_2^{(1)}, \ldots, s_M^{(1)}]^{\mathrm{T}}$, initializes $\tilde{\mathcal{S}} = \emptyset$ and performs the following steps for $i = M - T, \ldots, 1$:

1. Additional $\log_2 P$ branches are considered at level i, corresponding to the constellation points $\hat{s}_{i,p}^{(1)}$, for $1 \le p \le \log_2 P$, defined as

$$\hat{s}_{i,p}^{(1)} = \arg \min_{s_{i,p}^{(1)}\in\mathcal{O}_p} |s_{i,p}^{(1)} - s_i^{(1)}|^2 , \quad (2)$$

   where $\mathcal{O}_p \subset \mathcal{O}$ denotes the subset of the constellation points whose p-th bit is the negated of the p-th bit of $s_i^{(1)}$. This can be seen as a *local* bit-negating operation [6].
2. If $i > 1$, those additional branches are extended, following the DFE path, until the bottom of the tree is reached. The corresponding $\log_2 P$ transmitted vectors are then added to $\tilde{\mathcal{S}}$, $i \leftarrow i - 1$ and the algorithm continues from step 1. If $i = 1$, no path extension is required. The $\log_2 P$ transmitted vectors are directly added to $\tilde{\mathcal{S}}$ and the algorithm terminates with $\mathcal{L} = \mathcal{S} \cup \hat{\mathcal{S}}$.

Thus, after one iteration, $\tilde{\mathcal{S}}$ contains information about the bit values that are not represented in $\mathbf{s}^{(1)}$, although it does not necessarily contain the vectors $\mathbf{s} \in \mathcal{O}^M$ with bit values not represented in $\mathbf{s}^{(1)}$ *and* lowest Euclidean metric. In addition, the dependencies between the last $M - T$ levels are accounted for only in the DFE path extension of step 2 and not in the bit-negating operation of step 1, which is performed independently per level. However, the following justifies the use of $\tilde{\mathcal{S}}$ as a low-complexity alternative:

- By selecting $\mathbf{s}^{(1)}$ as the starting point, transmitted vectors that are relatively close to $\mathbf{s}^{(1)}$ are obtained in $\tilde{\mathcal{S}}$, where $\mathbf{s}^{(1)} = \hat{\mathbf{s}}_{\mathrm{ML}}$ with high probability [13].
- The columns of $\mathbf{H}$ selected in the last $M - T$ levels by the FSD ordering are more orthogonal between them compared to the columns obtained if no ordering or a vertical-Bell Labs layered space time (V-BLAST) ordering is applied [14]. This effectively reduces the dependencies between those $M - T$ levels.

Fig. 1 shows a tree diagram of the resulting list of candidates $\mathcal{L}$ if one iteration is performed in the SFSD in a 4×4 system with 4-QAM modulation, assuming Gray mapping, and $T = 1$ [7]. Al-

[1] For simplicity, $\mathbf{s}$ is used to denote both the transmitted vector and *a possible* transmitted vector

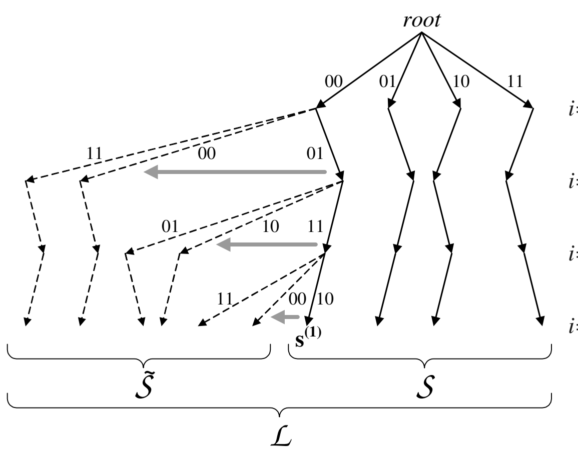

Fig. 1. List of candidates $\mathcal{L}$ generated by one iteration of the SFSD in a 4×4 system with 4-QAM modulation.

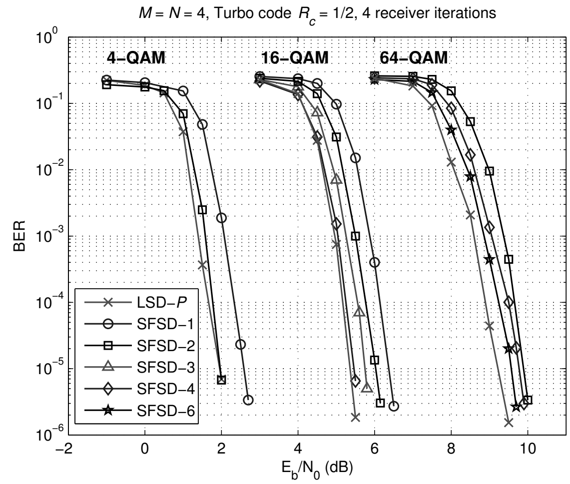

Fig. 2. BER performance of the SFSD and the LSD with a rate $R_\mathrm{c} = 1/2$ turbo code in a 4×4 system as a function of the SNR per bit.

though, in the 4-QAM case, the bit-negating operation corresponds to negating each bit independently, that is not always the case for higher modulations. However, that does not represent an increase in complexity, since the constellation points $\hat{s}_{i,p}^{(1)}$ in (2) can be obtained using a look-up table (LUT) of $P \times \log_2 P$ entries.

The performance of the SFSD can be improved by performing additional iterations according to N_SFSD, keeping $\tilde{\mathcal{S}}$ from the previous iteration and taking as a starting point $\mathbf{s}^{(2)}, \mathbf{s}^{(3)}, \ldots, \mathbf{s}^{(|\mathcal{S}|)}$ in each additional iteration, respectively. That performance improvement comes at the expense of only a linear complexity increase with N_SFSD. The total number of candidates in $\mathcal{L}$ is given then by $|\mathcal{L}| = P^T + N_\mathrm{SFSD} \cdot \log_2 P \cdot (M - T)$, where no clipping is required since all the bit values are represented in $\mathcal{L}$ [1]. The SFSD, like the FSD, has a fixed complexity independent of the noise level and the channel conditions. In addition, all the N_SFSD iterations can be performed in parallel, i.e. there are no dependencies between them. Thus, the SFSD is especially suited for a fully-pipelined real-time hardware implementation.

4. SIMULATION RESULTS

The bit error rate (BER) and complexity of the SFSD have been measured through Monte Carlo simulations, comparing them to those of the LSD, in a 4×4 system using 4-,16- and 64-QAM modulation. Frames of $K_\mathrm{b} = 9216$ bits with $K_\mathrm{ch} = 16$ have been transmitted using the rate $R_\mathrm{c} = 1/2$ parallel concatenated turbo code of memory 2 used in [1] and pseudo-random interleavers. The SFSD, denoted as SFSD-N_SFSD, takes as a starting point the corresponding FSD with $T = 1$ and performs a variable number of iterations N_SFSD depending on the modulation used ($N_\mathrm{SFSD} = \{1, 2\}, \{1, 2, 3, 4\}, \{2, 4, 6\}$ for 4-,16-,64-QAM, respectively). The LSD, denoted as LSD-P, generates a list of P candidates with the lowest Euclidean metric, setting its radius to that of the candidate with the largest Euclidean metric when the list is full. For simplicity, if the LSD obtains no information about one of the possible bit values, its extrinsic LLR value is clipped to ± 8 [1]. At the receiver, 4 iterations are considered, where one receiver iteration is defined as a complete cycle of extrinsic information exchange between the soft-MIMO detector and the turbo decoder with the turbo decoder performing 8 internal iterations. Finally, the soft-MIMO detectors are run only once at the beginning of the detection process. Although the performance of the detectors could be improved by incorporating *a priori* information and rerunning them in every iteration, that would cause an increase in complexity proportional to the number of iterations and it has not been considered in this paper.

Fig. 2 shows the BER performance of the SFSD and the LSD as a function of the signal to noise ratio (SNR) per bit, $E_b/N_0 = \log_2^{-1} P \,/\, \sigma^2$. It can be seen how the SFSD approximates the performance of the LSD independently of the modulation used, although more SFSD iterations are required for higher modulations. In particular, in the 16-QAM case, performing $N_\mathrm{SFSD} = 4$ iterations yields effectively the same performance as the LSD-16. On the other hand, if only $N_\mathrm{SFSD} = 1$ iteration is performed in the SFSD, the $|\mathcal{L}| = 28$ candidates obtained can only approximate the soft-information of the 16 candidates obtained by the LSD-16. Although $\mathcal{L}$ has the advantage of containing information about all the possible bit values, its elements are not guaranteed to have the lowest Euclidean distance per bit value, making it more difficult for the turbo-scheme to converge. In addition, although not included in this paper for brevity, further simulations show that the performance degradation decreases as the number of receiver iterations decreases.

The complexity of the SFSD and the LSD with 16-QAM modulation are shown in Fig. 3. In order to account for the computational complexity of the soft-detection algorithms, the curves include only real arithmetic operations (addition/subtraction and multiplication/division). For simplicity, all the operations have been considered to have the same effect on the final operations count. An optimized version of the LSD has been considered, using the direct Schnorr-Euchner (SE) enumeration proposed in [15]. In addition, a V-BLAST-zero forcing (ZF) channel matrix ordering has been considered to further reduce the complexity of the LSD [16]. The 90-percentile is plotted to indicate the number of operations required to generate the list of candidates in 90% of the cases, given the variable complexity of the LSD. For the number of SFSD iterations under study, the complexity of the SFSD is smaller than that of the LSD-16 with V-BLAST-ZF ordering, especially for the region of operation of turbo-MIMO systems, $E_b/N_0 < 15$ dB. Apart from the lower complexity, a very important advantage of the SFSD is

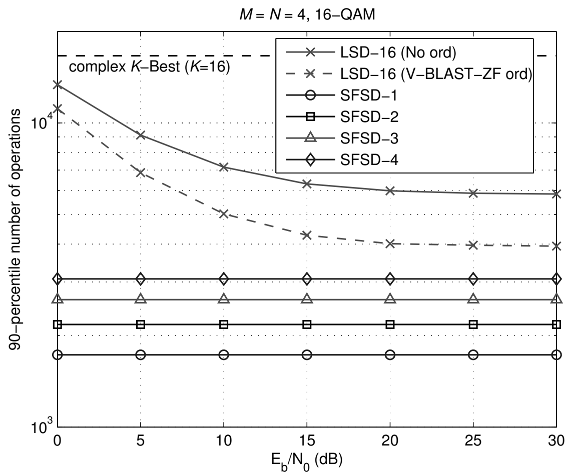

Fig. 3. Computational complexity of the SFSD, the SE-LSD and the K-Best lattice decoder in a 4×4 system with 16-QAM as a function of the SNR per bit.

its fixed complexity, that allows a fully-pipelined hardware implementation of the algorithm. The sequential nature of the LSD and its variable complexity can affect a mapping of the algorithm on a hardware platform. For comparison purposes, the complexity of the K-Best lattice decoder [11] with $K = 16$ is shown, given that it also has a fixed-complexity and yields a slightly better performance than the SFSD-2 if 4 iterations are run at the receiver (but worse than the SFSD-3). It can be seen how the complexity of the K-Best lattice decoder is considerably higher than that of the SFSD. Although different alternatives have been proposed to reduce it [11], they result in a variable complexity which affects its real-time implementation.

One important aspect of Fig. 3 is the fact that only arithmetic operations have been considered, thus *hiding* the additional logical operations required in the LSD due to the sequential tree search and the additional sorting operations required in both the LSD and the K-Best lattice decoder. On the other hand, in all cases $|\mathcal{L}| > P$, i.e. the SFSD generates more candidates than the LSD-P it is compared with. That has the side-effect of linearly increasing the number of additions required to evaluate $L_{\mathrm{E}}(b_k|\mathbf{r})$ in (1) compared to the LSD-P. However, that effect can be reduced by limiting the maximum number of candidates given by the SFSD without greatly affecting the performance. In this case sorting operations would be required to select the candidates with the lowest Euclidean metric and clipping values would need to be used if information about all bit values is not present.

5. CONCLUSION AND FUTURE WORK

In this paper, a soft-extension to a previously presented FSD has been proposed for iterative detection and decoding in turbo-MIMO systems. The SFSD uses the same channel matrix ordering and generates a list of candidates, relatively close to the hard-output FSD solution, that contains soft-information about all the possible bit values. It has been shown how the SFSD can approximate the performance of the LSD while having a lower and fixed complexity. In addition, the iterative nature of the SFSD allows for different levels of performance/complexity trade-off to be achieved. The SFSD, given its fixed complexity, is especially suited for a parallel and fully-pipelined real-time implementation.

In terms of current and future work, additional methods could be devised to improve the performance of the SFSD when a low number of iterations N_{SFSD} is performed, without resorting to a considerably higher nor variable complexity. In addition, a detailed comparison would need to be done between the SFSD, previously proposed alternatives to the LSD [4], [6] and soft-MIMO algorithms based on the sequential decoder [9].

6. REFERENCES

[1] B. M. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 389–399, Mar. 2003.

[2] J. Boutros, N. Gresset, L. Brunel, and M. Fossorier, "Soft-input soft-output lattice sphere decoder for linear channels," in *Proc. IEEE GLOBECOM '03*, San Francisco, CA, USA, Dec. 2003, vol. 3, pp. 1583–1587.

[3] H. Vikalo, B. Hassibi, and T. Kailath, "Iterative decoding for MIMO channels via modified sphere decoding," *IEEE Trans. Wireless Commun.*, vol. 3, no. 6, pp. 2299–2311, Nov. 2004.

[4] P. Marsch, E. Zimmermann, and G. Fettweis, "Smart Candidate Adding: A New Low-Complexity Approach Towards Near-Capacity MIMO Detection," in *Proc. 13th EUSIPCO '04*, Antalya, Turkey, Sep. 2005.

[5] M. S. Yee, "Max-log-MAP sphere decoder," in *Proc. IEEE ICASSP '05*, Philadelphia, PA, USA, Mar. 2005, vol. 3, pp. 1013–1016.

[6] R. Wang and G. B. Giannakis, "Approaching MIMO channel capacity with soft detection based on hard sphere decoding," *IEEE Trans. Commun.*, vol. 54, no. 4, pp. 587–590, Apr. 2006.

[7] L. G. Barbero and J. S. Thompson, "Performance analysis of a fixed-complexity sphere decoder in high-dimensional MIMO systems," in *Proc. IEEE ICASSP '06*, Toulouse, France, May 2006, vol. 4, pp. 557–560.

[8] L. G. Barbero and J. S. Thompson, "Rapid prototyping of a fixed-throughput sphere decoder for MIMO systems," in *Proc. IEEE ICC '06*, Istanbul, Turkey, June 2006, vol. 7, pp. 3082–3087.

[9] S. Bäro, J. Hagenauer, and M. Witzke, "Iterative detection of MIMO transmission using a list-sequential (LISS) detector," in *Proc. IEEE ICC '03*, Anchorage, AK, USA, May 2003, vol. 4, pp. 2653–2657.

[10] D. Le Ruyet, T. Bertozzi, and B. Özbek, "Breadth first algorithms for APP detectors over MIMO channels," in *Proc. IEEE ICC '04*, Paris, France, June 2004, vol. 2, pp. 926–930.

[11] Z. Guo and P. Nilsson, "Algorithm and implementation of the K-best sphere decoding for MIMO detection," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 3, pp. 491–503, Mar. 2006.

[12] A. D. Murugan, H. El Gamal, M. O. Damen, and G. Caire, "A unified framework for tree search decoding: Rediscovering the sequential decoder," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 933–953, Mar. 2006.

[13] J. Jaldén, L. G. Barbero, B. Ottersten, and J. S. Thompson, "Full diversity detection in MIMO systems with a fixed-complexity sphere decoder," in *Proc. IEEE ICASSP '07*, Honolulu, HI, USA, Apr. 2007, vol. 3, pp. 49–52.

[14] H. Zhang, H. Dai, Q. Zhou, and B. L. Hugues, "On the diversity order of spatial multiplexing systems with transmit antenna selection: A geometrical approach," *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5297–5311, Dec. 2006.

[15] A. Burg, M. Borgmann, M. Wenk, M. Zellweger, W. Fichtner, and H. Bölcskei, "VLSI implementation of MIMO detection using the sphere decoding algorithm," *IEEE J. Solid-State Circuits*, vol. 40, no. 7, pp. 1566–1577, July 2005.

[16] M. O. Damen, H. El Gamal, and G. Caire, "On maximum-likelihood detection and the search for the closest lattice point," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2389–2402, Oct. 2003.