SELECTIVE SPANNING WITH FAST ENUMERATION DETECTOR IMPLEMENTATION REACHING LTE REQUIREMENTS

Jarmo Niskanen, Janne Janhunen and Markku Juntti

*Uninord Ltd.
Oulu, Finland
jarmo.niskanen@uninord.fi

Centre for Wireless Communications
University of Oulu, Oulu, Finland
{janne.janhunen, markku.juntti}@ee.oulu.fi

ABSTRACT

We present a hardware implementations of the selective spanning with fast enumeration (SSFE) detection algorithm for a spatial multiplexing multiple-input multiple-output (MIMO) orthogonal frequency division multiplexing (OFDM) system. We compare the results to a corresponding implementation of well known K-best list sphere detection algorithm in a 3G long term evolution (LTE) system. We show that due to many favorable properties of the SSFE algorithm, the ASIC design achieves in 2×2 antenna system up to 210 Mbps decoding rate with 66k gate equivalents (GE) and in 4×4 antenna case up to 420 Mbps with 254 kGE.

1. INTRODUCTION

Peak data rate up-to 100 Mbps is required for the long term evolution (LTE) standard [1] and the LTE-A proposal goes beyond that, up-to a Gbps peak data rates. The multiple-input multiple-output (MIMO) antenna system combined with the orthogonal frequency division multiplexing (OFDM) technique has been proposed for many standards to increase capacity or diversity in the system. The multipath environment causes MIMO channel to be frequency-selective and OFDM can transform such a channel into a set of parallel frequency-flat MIMO channels, which decreases the receiver complexity. High data rate wireless communication needs power efficient solutions to process the increasing amounts of data with a limited hardware and low power consumption.

Linear minimum mean square error (LMMSE) and zero forcing (ZF) principles can be straightforwardly applied in MIMO detection. Unfortunately, the linear detectors can suffer a significant performance loss in fading channels, especially when there is a spatial correlation between antenna elements [2].

The maximum likelihood (ML) detector is optimal for finding the closest lattice point [3]. However, it is not often feasible for real implementations, because its computational complexity increases exponentially with the increasing number of transmit antennas. The sphere detector (SD) [4] calculates the ML solution with reduced complexity compared to full-complexity exhaustive search ML detectors [3]. The list sphere detector (LSD) [5] is a variant of the sphere detector that can be used to approximate the soft decision maximum *a posteriori* probability (MAP) detector. There are multiple variations of list sphere detectors such as increasing radius (IR) [6] and K-best [7]. In addition, there are detectors which have similarities to sphere detectors such as layered orthogonal lattice detector (LORD) [8] and selective spanning with fast enumeration (SSFE) [9]. In this paper, we consider K-best and SSFE detectors.

The remaining part of the paper is structured as follows: Section 2 presents the system model and briefly discusses the MIMO detection problem. Section 3 reviews the K-best and SSFE algorithms. Simulation model and results are presented in Section 4. Sections 5 discusses the implementation flow. Section 6 summarizes the results and compares the K-best and SSFE detector implementations presented in literature. Section 7 finally concludes the paper.

2. SYSTEM MODEL

We consider a MIMO–OFDM system with N transmit and M receive antennas, where $N \leq M$. Figure 1 illustrates the applied system model. Table 1 summarizes the 3G channel model parameters based on the International Telecommunication Union (ITU) specification. The model applies a layered space-time architecture with vertical encoding in 2×2 antenna system and horizontal encoding in 4×4 antenna system. The cyclic prefix of an OFDM symbol is assumed to be long enough to eliminate intersymbol interference, i.e., larger than $\frac{T_\mathrm{m}}{T_\mathrm{s}}$, where T_m is the maximum delay spread in channel and T_s denotes the symbol time. The maximum delay spread and OFDM symbol time are presented in Tables 1 and 2, respectively. The received signal with sth subcarrier can be presented as

$$\mathbf{y}_s = \mathbf{H}_s \mathbf{x}_s + \eta_s, \quad s = 1, 2 ..., S \tag{1}$$

where S is the number of subcarriers, $\mathbf{y}_s \in \mathbb{C}^M$, $\mathbf{x}_s \in \mathbb{C}^N$ denotes the transmitted symbol vector, $\mathscr{A} \in \mathbb{C}$ is the symbol alphabet and $\eta_s \in \mathbb{C}^M$ is an identically distributed complex Gaussian noise vector with variance σ^2. The symbol $\mathbf{H}_s \in \mathbb{C}^{M \times N}$ denotes the channel matrix. Bit-interleaved coded modulation (BICM) is applied. The entries of $\mathbf{x}_s$ are chosen independently of each other from a quadrature amplitude modulation (QAM) constellation.

Table 1: Channel model parameters

Number of paths	6
Path delays [ns]	[0 310 710 1090 1730 2510]
Path power [dB]	[0 -1 -9 -10 -15 -20]
BS antenna spacing	4 λ
MS antenna spacing	0.5 λ
BS avg angle of dept	50°
MS avg angle of arrival	67.5°
BS azimuth spread	5°
BS azimuth spread	35°

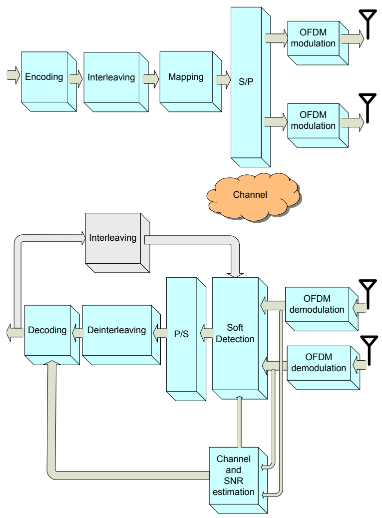

Figure 1: A MIMO–OFDM system model.

The ML detector minimizes the Euclidean distance between the received signal $\mathbf{y}$ and the lattice points $\mathbf{Hx}$ and selects the lattice point that minimizes the Euclidean distance to the received vector $\mathbf{y}$, i.e.,

$$\hat{\mathbf{x}} = \arg\min_{\widetilde{\mathbf{x}} \in \mathscr{A}^N} \| \mathbf{y} - \mathbf{H}\widetilde{\mathbf{x}} \|^2, \tag{2}$$

where $\| \cdot \|$ denotes the L_2 norm of a vector. The exhaustive search can be used to solve the ML detection problem. However, it becomes computationally infeasible as the set of lattice points increases. The sphere detection algorithm solves the ML approximation (2) by limiting the search to the lattice points that lie inside a M-dimensional hyper-sphere [3].

3. DETECTOR ALGORITHMS

The LSD algorithm approximates the MAP detection in channel coded systems with reduced computational complexity. Basically, the LSD algorithm traverses a tree, whose depth depends on the number of transmit antennas and the number of branches depends on the used constellation. The real signal model doubles the depth in the search tree compared to a complex signal model algorithm, but provides for instance a less complex distance calculation.

The computational complexity can be reduced by limiting the search inside a sphere with radius d using the sphere constraint $d^2 \geq \| \mathbf{y} - \mathbf{Hx} \|^2$. The channel matrix $\mathbf{H}$ can be QR decomposed (QRD) into two parts. If the number of transmit and receiver antennas are equal, the channel matrix can be presented as $\mathbf{H} = \mathbf{QR}$, where $\mathbf{Q}$ denotes a $N \times N$ orthogonal matrix and $\mathbf{R}$ is a $N \times N$ upper triangular matrix. After the QR decomposition, the equation can be rewritten as

$$\begin{aligned} & d^2 \geq \| \mathbf{y} - \mathbf{QRx} \|^2 \\ \Leftrightarrow\, & d^2 \geq \| \mathbf{Q}^{\mathrm{H}}\mathbf{y} - \mathbf{Rx} \|^2, \end{aligned} \tag{3}$$

where $\mathbf{Q}^{\mathrm{H}}$ denotes the Hermitian transpose of matrix $\mathbf{Q}$.

By denoting $\mathbf{Q}^{\mathrm{H}}\mathbf{y} = \mathbf{y}'$, we get

$$d^2 \geq \| \mathbf{y}' - \mathbf{Rx} \|^2 . \tag{4}$$

Let $\mathbf{x}_i^N = (x_i, x_{i+1}, \ldots, x_{N-1}, x_N)^{\mathrm{T}}$ denote the last $N-i+1$ components of the vector $\mathbf{x}$. The sphere search can be thought as a tree structure, where the root layer corresponds to $\mathbf{x}_N^N$. The last elements of the possible symbol vectors are calculated first, i.e., $x_N, x_{N-1}, \ldots, x_1$.

The partial Euclidean distance can be calculated as [10]

$$d(\mathbf{x}_i^N) = d(\mathbf{x}_{i+1}^N) + \left| y_i' - \sum_{j=i}^{N} r_{i,j} x_j \right|^2 , \tag{5}$$

where $i = N, N-1, \ldots, 1$ and $r_{i,j}$ is the i, jth term of the upper triangular matrix $\mathbf{R}$.

3.1 *K*-Best algorithm

The K-best LSD algorithm [11] is a breadth-first search algorithm based on the well known M-algorithm [12, 13]. The LSD algorithm proceeds a level by level repeating spanning-sorting-deleting process. The process will continue until the leaf nodes are reached. After the final level, the K best candidates are sorted and output as a final candidate list. The main complexity of the K-best LSD algorithm comes from the PED calculation and sorting the K best distances into the list.

Figure 2 presents the spanning-sorting-deleting processing in the tree search algorithm, where list size $K = 4$. The example illustrates a real-valued signal model with 2×2 antenna system and 16-QAM. The black arrows show the K best paths at each level and the grey arrows are the deleted paths, which did not succeed in the selection. Note that the impact of the node discarding becomes more significant when the number of transmit antennas increases, a high order modulation is used or the list size is small.

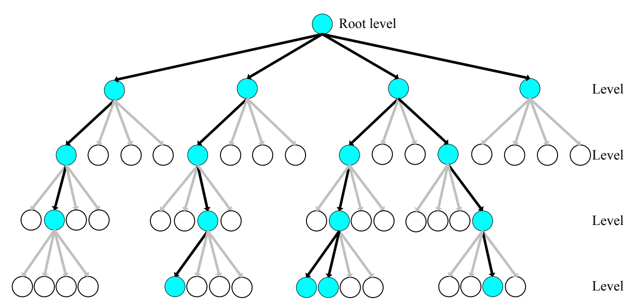

Figure 2: An example of K-best LSD tree search.

A large list size improves the decoding performance, but leads to an increasing computation burden and memory usage. Candidate list sorting is required, when the number of candidates exceeds the list size K. The candidate list updating requires a comparison between a new PED and the maximum PED in the list. If the new PED is smaller than the maximum PED in the list, the new PED is included in the list. Otherwise, the list stays untouched.

The complexity of the algorithm depends mostly on the number of transmit antennas, the list size and the modulation level. The algorithm maintains a list of the K best symbol candidates and the corresponding multidimensional constellation symbol identifiers. For example, in 64-QAM with a real-valued signal model, $\sqrt{64} = 8$ QAM symbols can be represented with $S_b = \log_2(8) = 3$ bits, 000 representing the first QAM symbol and 111 representing the last QAM symbol. By setting the sphere radius to infinity, $d = \infty$, a fixed number of nodes is processed in each step of the algorithm. The algorithm is serial between the PED calculation and sorting, which prevents writing a fully parallel code between the levels. High computing power is required to achieve real time requirements.

3.2 SSFE

Selective spanning with fast enumeration algorithm has many architecturally favorable features such as deterministic and regular dataflow [9]. The algorithm is characterized by a level update vector $\mathbf{m} = [m_1, ..., m_M]$ in complex-valued system and $\mathbf{m} = [m_1, ..., m_{2M}]$ in real-valued system. The level update vector defines the number of spans for each node on level i and also the length of the final candidate list. Hereafter we consider a real-valued system. Because there is no node deleting process in the algorithm, some extra computational complexity is created for the log-likelihood ratio (LLR) unit. For example in 16-QAM, 2×2 antenna system with real signal model, the vector $\mathbf{m} = [4,4,4,4]$ would lead to a full search and to the length of 256 candidates in the final list. The vector $\mathbf{m} = [1,2,2,4]$ or $\mathbf{m} = [1,2,2,3]$ would lead to a more realistic implementation of the algorithm, only 16 or 12 candidates in the final list.

A short Euclidean distance list keeps also the log-likelihood radio calculation unit simple. The spanned nodes are never deleted. Thus, "unnecessary" PED computing is not done like in the K-best algorithm. The total number of computed nodes in the search tree can be determined using vector $\mathbf{m}$ i.e. $\prod_{j=i}^{2M} m_j$.

The heart of the SSFE algorithm is a slicer unit. The slicer unit selects a set of closest constellation points $\mathbf{x}^i$ such that $\|d_i(\mathbf{x}^i)\|^2$ is minimized at each level. In Figure 3, the grey nodes present constellation points on the horizontal axis, whereas the white circle is the received symbol. If $\mathbf{m}$ vector requires for instance two constellation points to be sliced, the slice $\triangle_1$ is picked first and then the slice $\triangle_2$. Thus, the SSFE is a distributed and greedy algorithm. It is distributed because $\mathbf{m}$ defines locally the number of spanned nodes. This is different for instance to the conventional K-best algorithm, in which the spanning-sorting-deleting process is globally based on K. A K-best version with variable K for each level would resemble SSFE algorithm. However, in SSFE the symbol selection is based on the slicer operation which clearly differentiates these two algorithms.

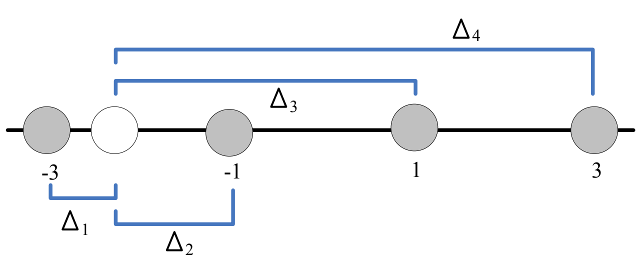

Figure 3: The principle of slicer operation in 16-QAM real system model.

Figure 4 presents a SSFE tree search with real-valued system model assuming two transmit antennas and 16-QAM. The vector $\mathbf{m} = [2,1,2,2]$, in which the first element (2) corresponds the number of slices on the level 1, the second element (1) corresponds slices on the level 2 and so forth. The final list size in this example is eight.

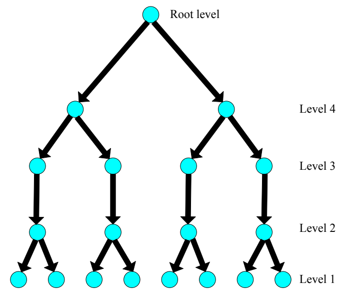

Figure 4: An example of SSFE tree search.

Instead of using a complex-valued system model [9], we prefer a real-valued one. It provides a simpler Euclidean distance calculation but also as a simpler slicer operation. The closest constellation point selection can be done on horizontal axis instead of selecting constellation point from the two dimensional grid.

4. SIMULATION

The parameters K and the vector $\mathbf{m}$ have a significant impact on the complexity of the K-best and SSFE algorithms. Floating-point simulations for K-best and SSFE algorithms have been carried out in a MATLAB environment. Simulation parameters are inspired by the 3G LTE specifications [14] and are summarized in Table 2.

In LTE, a radio frame period is 10 ms, which is divided to 1 ms subframes. The subframe is further divided into two slots both period of 0.5 ms. In case of normal cyclic prefix (CP) a single slot consists seven OFDM symbols, where the overall symbol time is the sum of useful symbol time and the length of CP. A resource block is defined in time-frequency domain. In time domain, the resource block lasts a slot period, which consists seven OFDM symbols with normal CP. In frequency domain, the resource block has 12 subcarriers. For LTE, the OFDM subcarrier spacing has been chosen to be $\triangle f = 15$ kHz. The LTE carrier can consist any number of resource blocks between 6 and 110, which roughly corresponds to a bandwidth from 1 MHz to 20 MHz. In simulations, a 5 MHz bandwidth is assumed, which corresponds to 512 (300 used) OFDM subcarriers.

The simulator takes into account the effect of log-likelihood ratio (LLR) clipping [15] with threshold $L_{\max} = 8$. The LSD output list is used to calculate the approximation of the probability LLR of each transmitted bit. By limiting the dynamic range of the LLR, the required LSD list size can be decreased and the computational complexity of the LSD decreases.

Figures 5 and 6 compare the LMMSE, MAP, K-best and SSFE detectors. We use a moderately correlating channel, which is based on the 3GPP vehicular A parameters specified by International Telecommunication Union (ITU). A linear detector does not perform well in the correlating channel. The MAP detector illustrates again the optimal receiver performance for the channel coded system. The K-best algorithm performs better over SSFE algorithm with a list size

Table 2: Simulation parameters

Number of subcarriers	512 (300 used)
Bandwidth	5 MHz
Carrier frequency	2.4 GHz
Cyclic prefix (CP) duration	4.69 μs
Symbol time T_s	66,7 μs
Encoding	VBLAST, HBLAST
Channel code	Turbo code
Code rate	1/2
Channel model	3GPP-VA ITU,
User velocity	120 km/h
Frames per SNR point	1500

$K = 8$, which can be considered as a feasible list size for an area and power efficient K-best implementation. However, in better channel the SSFE algorithm becomes an attractive alternative. For instance in 16-QAM, 4×4 antenna system, the K-best detector with $K = 8$ performs approximately 0.3 dB (10^{-2} FER) better than the SSFE with $\mathbf{m} = [11122223]$. The K-best computes 212 PEDs and sorts 152 times, whereas the SSFE computes 237 PEDs but replaces expensive sorting operations with 14 slicing operations.

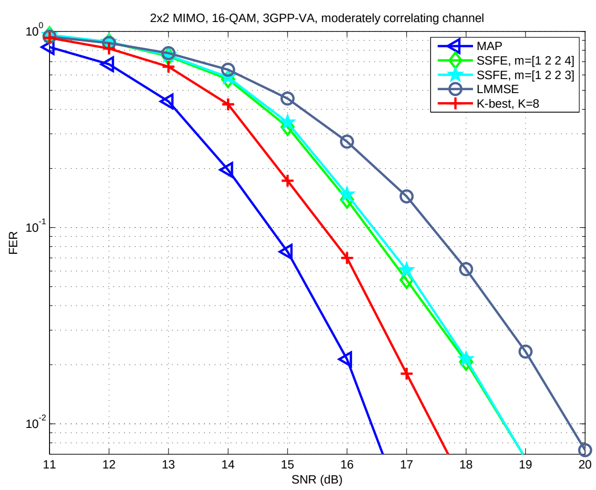

Figure 5: Frame error rate vs. SNR in a 2x2 system in a moderately correlating channel.

5. IMPLEMENTATION TOOL CHAIN

In the logic generation flow, we use a high-level Catapult C synthesis tool. A fixed-point ANSI C/C++ code is first simulated in software simulation and then we slightly modify the code to fit into the synthesis tool. From a high level synthesis we get a register transfer level (RTL) code. C/C++ coding is faster and less error sensitive compared to traditional RTL coding. The high level synthesis tool automates a large part of the interface and pipeline generation. The tool allows a designer to choose the best architecture for a given design specification making tradeoffs between performance, silicon area and power consumption. The same design can be easily tested for different pipeline and speed targets. After the high level synthesis is finished, we generate the logic synthesis with Synopsys Design Compiler using a 0.18 μm CMOS technology. The tool generates verilog netlist and a standard delay format (SDF) files. In the next two steps, we create a testbench and do the simulation with Modelsim tool. PrimePower gets input file from the simulation. Based on the input file, the tool provides an estimation of the power consumption.

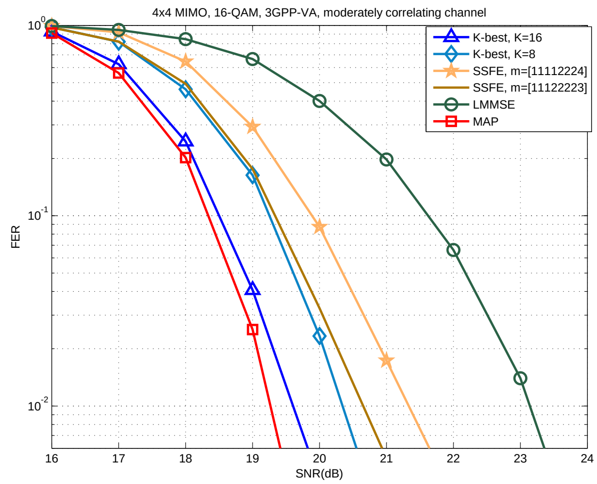

Figure 6: Frame error rate vs. SNR in a 4x4 system in a moderately correlating channel.

6. RESULTS AND COMPARISON

LTE standard set up the decoding rate target for both of our SSFE implementations. We utilize the $\mathbf{m} = [1224]$ for 2×2 and $\mathbf{m} = [11112224]$ for 4×4 antenna system. A new symbol data is taken every third clock cycle in both SSFE implementations. Due to efficient pipelining, low 35 MHz clock frequency enables the required decoding rate. The low clock frequency has significant influence on low power consumption.

We utilized a 16-bit fixed-point arithmetic. The word is divided in 5-bit integer part and 10-bit fraction. One bit is used for sign. The 16-bit fixed-point arithmetic has negligible frame error rate over double precision floating-point arithmetic. Since a half code rate is assumed, the maximum throughput for the actual data is half of the decoding rate. The goodput, which can be defined to be successfully received data, depends on the SNR level and is not considered in the results. We summarize the gate equivalent, power consumption and the decoding rate for implementations in Table 3.

We compare the Catapult C SSFE detector implementation to a hand coded SSFE detector implementation [16] and to a Catapult C implementation of the K-best detector. A fair comparison between designs is difficult. In addition to different design parameters, the used technologies may differ. Note that the power dissipation between SSFE implementations are not comparable due to different CMOS technologies. Scaling a CMOS technology from 180 nm to 65 nm can reduce the design power dissipation up to 75 percent.

In [16], a hand coded RTL of complex signal model SSFE is presented. The architecture supports 16-QAM and

64-QAM and is scalable from 2×2 to 8×8 antenna systems. The expensive multiplication operations are replaced with shift and add operations. The Euclidean norm (L2-norm) has been replaced with the Manhattan norm (L1-norm), which removes the square operation from the PED calculation, and thus, simplifies the detector. However, the L1-norm has not been used in our implementations due to significant performance loss in the coded channel.

The *K*-best algorithm gives a reliable data transmission throughput in correlating channel, but it is also found to be complex to implement. The *K*-best [17] and our SSFE implementation use the same tool flow and CMOS technology, and thus, they are somewhat comparable. The pipeline of the *K*-best implementations can receive new symbol data after every 8th clock cycle. The long pipeline needs a clock frequency of 150 MHz, which partly explains the high power consumption. The design supports QPSK, 16- and 64-QAM, which increases the number of gate equivalents. We summarize the implementation comparison in Table 3.

Table 3: Detector implementation comparison

Detector	MIMO	kGE	Power (mW)	Dec. rate (Mbps)
SSFE	2×2	66	23	210
SSFE	4×4	254	200	420
SSFE, [16]	2×2	45	9	200
SSFE, [16]	4×4	145	28	400
K-Best, K=8 [17]	2×2	110	120	140
K-Best, K=8 [17]	4×4	209	290	280

7. CONCLUSIONS

We implemented an SSFE detector for 16-QAM, 2×2 and 4×4 antenna systems using a high-level Catapult C synthesis tool. Our design target is set by the LTE requirements. The results show that the SSFE detector for 2×2 antenna system can be implemented with moderate silicon area and power consumption. In 4×4 case, we kept the same operating frequency but doubled the decoding rate due to doubled number of transmit antennas. Thus, the silicon area and power consumption are increased but up to 420 Mbps decoding rate is achieved.

REFERENCES

[1] 3rd Generation Partnership Project (3GPP); Technical Specification Group Radio Access Network, “Physical layer aspects for evolved UTRA (TR 25.814 version 1.5.0 (release 7)),” 3rd Generation Partnership Project (3GPP), Tech. Rep., 2006.

[2] H. Artés, D. Seethaler, and F. Hlawatsch, “Efficient detection algorithms for mimo channels: A geometrical approach to approximate ml detection,” *IEEE Transactions on Signal Processing*, vol. 51, no. 11, pp. 2808–2820, Nov. 2003.

[3] M. O. Damen, H. E. Gamal, and G. Caire, “On maximum–likelihood detection and the search for the closest lattice point,” *IEEE Transactions on Information Theory*, vol. 49, no. 10, pp. 2389–2402, Oct. 2003.

[4] U. Fincke and M. Pohst, “Improved methods for calculating vectors of short length in a lattice, including a complexity analysis,” *Math. Comput.*, vol. 44, no. 5, pp. 463–471, May 1985.

[5] B. Hochwald and S. ten Brink, “Achieving near-capacity on a multiple-antenna channel,” *IEEE Transactions on Communications*, vol. 51, no. 3, pp. 389–399, Mar. 2003.

[6] M. Myllylä, M. Juntti, and J. R. Cavallaro, “A list sphere detector based on Dijkstra’s algorithm for MIMO-OFDM system,” in *Proceedings of the IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications*, Athens, Greece, Sep. 3-7 2007, pp. 1–5.

[7] K. Wong, C. Tsui, R. K. Cheng, and W. Mow, “A VLSI architecture of a K-best lattice decoding algorithm for MIMO channels,” in *Proceedings of the IEEE International Symposium on Circuits and Systems*, vol. 3, Scottsdale, AZ, May 26-29 2002, pp. 273–276.

[8] M. Siti and M. Fitz, “A novel soft-output layered orthogonal lattice detector for multiple antenna communications,” in *Proceedings of the IEEE ICC 2006*, Jun.11–15 2006, pp. 1686–1691.

[9] M. Li, B. Bougart, E. Lopez, and A. Bourdoux, “Selective spanning with fast enumeration: A near maximum-likelihood mimo detector designed for parallel programmable baseband architectures,” in *Proceedings of the IEEE ICC 2008*, Beijing, China, May19-23 2008, pp. 737 – 741.

[10] A. Burg, M. Borgmann, M. Wenk, M. Zellweger, W. Fichtner, and H. Bölcskei, “VLSI implementation of MIMO detection using the sphere decoding algorithm,” *IEEE Transactions on Applied Superconductivity*, vol. 40, no. 7, pp. 1566–1577, Jul. 2005.

[11] M. Myllylä, P. Silvola, M. Juntti, and J. R. Cavallaro, “Comparison of two novel list sphere detector algorithms for MIMO-OFDM systems,” in *Proceedings of the IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications*, Helsinki, Finland, Sep. 11-14 2006, pp. 12–16.

[12] F. Jelinek and J. Anderson, “Instrumentable tree encoding of information sources,” *IEEE Transactions on Information Theory*, vol. 17, no. 1, pp. 118–119, Jan. 1971.

[13] J. Anderson and S. Mohan, “Sequential coding algorithms: A survey and cost analysis,” *IEEE Transactions on Communications*, vol. 32, no. 2, pp. 169–176, Feb. 1984.

[14] 3rd Generation Partnership Project (3GPP), “http://www.3gpp.org.”

[15] M. Myllylä, J. Antikainen, M. Juntti, and J. Cavallaro, “The effect of LLR clipping to the complexity of list sphere detector algorithms,” in *Proceedings of the Annual Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, USA, Nov. 4-7 2007, pp. 1559–1563.

[16] R. Fasthuber, D. Novo, P. Raghavan, L. V. D. Perre, and F. Catthoor, “Novel energy-efficient scalable soft-output ssfe mimo detector architectures,” in *International Symposium on Systems, Architectures, Modeling and Simulation*, Samos, Greece, Jul.20-23 2009, pp. 165–171.

[17] J. Ketonen, M. Juntti, and J. Cavallaro, “Performance–complexity comparison of receivers for a lte MIMO–OFDM system,” *IEEE Transactions on Signal Processing*, Oct. 2009.