Abstract— The complex sphere decoding algorithm has optimal bit error ratio performance for uncoded multiple-input multiple-output (MIMO) systems. The computational and hardware complexities of this algorithm increase significantly for detection of 64-QAM modulated signal streams. In this paper two modifications to the original sphere decoding algorithm are proposed that reduce both computational complexity and required hardware resources compared to the original sphere decoding algorithm. The improvements are achieved using a new definition for sphere radius and changing the symbol search strategy. Simulation results show the new algorithms have a small bit error ratio (BER) degradation compare to the original sphere decoding algorithm.

I. INTRODUCTION

Multiple-input multiple-output (MIMO) communication use multipath propagations and space-time coding to increase data rate and improve overall bit error ratio (BER) in the communication link. Recent IEEE standards for WLANs (IEEE 802.11n) and WMANs (IEEE 802.16) are based on the combination of MIMO with orthogonal frequency division multiplexing (OFDM) modulation. In mobile networks, MIMO is included in the new generation of UMTS standard and is also being considered for the fourth-generation mobile access standards [1], [2]. MIMO receivers defined by IEEE 802.11n and IEEE 802.16 should support a number of modulation schemes including 64-QAM modulation and achieve required throughput levels.

In a MIMO communication system comprising \( N_t \) transmitter and \( N_r \) receiver antennas, each of the \( N_t \) receivers will receive components from each of the \( N_r \) transmitters, including both line-of-sight and reflected components. The role of the MIMO detector is to use the received signals at each of the \( N_r \) receiver antennas and to recover the transmitted symbols. There is a considerable increase in complexity of signal processing tasks related to detection in MIMO receivers. The Maximum-likelihood (ML) detector is the optimum detection algorithm in MIMO communications systems. The major issue with ML detectors is the high level of computational complexity of these detectors. Lattice-based decoding algorithms, such as the sphere decoding algorithm [3], have lower complexity and are able to achieve close to optimal ML performance. The basic principle of sphere decoding is to reduce the number of symbols needed to be considered for finding the ML solution. In order to be able to achieve the throughput levels envisioned in the next generation wireless standards, the complexity of sphere decoding algorithm need to be further reduced specifically for higher constellation modulation schemes. Numerous algorithm modifications have been explored in the past to further reduce the complexity of sphere decoder by applying techniques such as reordering the computation sequence, performing fixed number of operations, modified norm definition and early termination of search [6], [7], [9].

In this paper, we will look at the challenges related to implementation of depth-first sphere decoding (DFSD) algorithm for detection of 64-QAM modulated transmitted symbols. We propose improvements to the original DFSD algorithm that increase the overall data throughput of SD algorithm and reduce the memory requirements in hardware implementation of detector.

The rest of the paper is organized as follows. After a brief review of MIMO detection and sphere decoding algorithm, in section III a number of performance enhancements to the original algorithm will be introduced. In section IV, the implementation of detector.

II. MIMO CHANNEL MODEL

A. MIMO System Model

In a flat-fading narrow band MIMO system with \( N_t \) transmitter and \( N_r \) receiver antennas (\( N_t \geq N_r \)), the channel can be modeled as \( N_r \times N_t \) matrix \( H = [h_{ij}] \), the signal at the transmitter and receiver are complex vectors \( s \ (N_t \times 1) \) and \( y \ (N_r \times 1) \). The relation between the transmitter and receiver in a MIMO channel can be modeled as:

\[
y = Hs + n
\]

(1)

The \( N_r \times 1 \) vector \( n \) represents the thermal noise as independent identically distributed additive complex gaussian noise at the receiver with zero mean and variance \( \sigma^2 \). We assume that in this model the channel matrix has been identified and known to the receiver. The transmit vector \( s \) corresponds to a binary vector, containing \( N_t \) bits, where \( b \) is the number of bits per symbol in the complex constellation \( \Lambda = \{e^{j\pi b/2 b+1} \} \). The Maximum Likelihood (ML) detector solves (1) by finding the transmitted symbol vector \( s \) that minimizes \( J(s) = \|y - Hs\|^2 \) as described in (2)
Applying QR decomposition on channel matrix $\mathbf{H}$ [5], we can further simplify (1) and (2) to obtain (3):

$$s_{at} = \arg \min_{s_{at}} \| \mathbf{y} - \mathbf{H} s \|$$

Expanding vector norm in (3) yields

$$s_{at} = \arg \min_{s_{at}} \sum_j | y_j - \sum_i R_{ji} s_i |$$

starting from $i=N_t$, (6) can be solved recursively as follows:

$$\| T_i(\mathbf{P}) \|^2 = \| T_{i+1}(\mathbf{P}_{i+1}) \|^2 + \| e_i(\mathbf{P}) \|^2$$

and

$$T_{N_t+1}(\mathbf{P}_{N_t+1}) = 0$$

where

$$T_i(\mathbf{P}) > T_{i+1}(\mathbf{P}_{i+1})$$

and

$$e_i(\mathbf{P}) = y_q - \sum_{j=i}^{N_t} R_{ji} s_j$$

In (7)-(9), $\mathbf{P} = \begin{bmatrix} s_0, s_1, \ldots, s_{N_t} \end{bmatrix}$ is commonly known as partial symbol vector. Using the above simplification, the original optimization problem (2) can be recursively solved by applying an iterative tree search methodology [6], [7].

The cost function of the resulting optimization problem is $T_i(\mathbf{P})$, which is known as “Partial Euclidean Distance (PED)” [6].

### B. Depth-First Sphere Decoding Algorithm

The Sphere Decoding (SD) algorithm can achieve close to optimal ML BER performance with reduced computational complexity [8]. In this detector the search is limited only to those vector symbols $\mathbf{s}$ for which $\mathbf{H} \mathbf{s}$ lies inside a hyperspace with radius $r$ around the received point $\mathbf{y}$. We can express this condition by the following inequality:

$$d(\mathbf{s}) = \| \mathbf{y} - \mathbf{H} \mathbf{s} \| < r$$

All possible combinations of transmitted symbols in a MIMO channel can be mapped on to a tree graph. The root of this tree is on level $N_t+1$ and each node at level $i$ corresponds to the transmitted symbols from $N_i$ to $i$-th antenna. The leaves on the first level show all the possible combinations of transmitted symbols. Depth-first search starts at the root of the tree ($i = N_t+1$) and progresses by expanding each node of the search tree into its children nodes and going deeper and deeper until it reaches the leaves at the first level of the tree, or it can not progress any further due to a failed test. In this case the search backtracks, returning to the most recent node it had not finished exploring. In a depth-first sphere detector, the search tree is traversed depth-first such that at each node after calculating the partial metric $T_i$ the sphere radius (SR) test is performed. The result of this test determines the direction of tree traverse. It is possible to improve the throughput of the sphere detector by shrinking the sphere radius dynamically.

The basic idea is that once a valid solution has been identified, the value of the sphere radius is updated to the value of the PED associated with this solution and the tree search continues with the updated radius value [5]. Following Schnorr-Euchner (SE) ordering rule can further optimize the tree search [4]. Based on SE ordering, at each level of the tree the priority is given to the nodes with the smallest PEDs.

The depth-first sphere decoding algorithm with dynamic radius reduction is outlined in the following steps.

1. Read channel parameters $\mathbf{y}_0$, $\mathbf{R}_0$, initial sphere radius $r_0$
2. Set $i=N_t$, $r=r_0$, $T_i(\mathbf{P}_{i=0})=0$, $P_{i=0}=[]$, $l_i=0$ $k=1,2,...,4$
3. Generate $\mathbf{P}(c_{i+1})=[c_{i+1}, \mathbf{P}_{i+1}]$, $\text{iteration}=$iteration+1
4. Calculate $T_i(\mathbf{P}(c_{i}))$ for all children nodes using (7)
5. If there is at least one $m : T_i(\mathbf{P}(c_{i}))<r$ (SR test)
   - $l_i =$length $\{ T_i(\mathbf{P}(c_{i})) : T_i(\mathbf{P}(c_{i}))<r \}$
   - $T_{i+1}(\mathbf{P}_{i+1})=$sort $\{ ( T_i(\mathbf{P}(c_{i})) : T_i(\mathbf{P}(c_{i}))<r_i ) \} \text{ } t=1,\ldots,1$
   - Save $\mathbf{P}_{i}(c_{i})$, $T_i(\mathbf{P}_{i}(c_{i}))$
   - Go to step 6
   - Else
     - Go to step 7
6. If $i = 1$
   - Update SR: $r=$min($T_{1,li})=T_{1,li}$
   - Go to step 7
   - Else
     - Set $T_{min}=T_{1,li}$, $P_{min}=P_{1,li}$
     - $l_i =$length $\{ T_i(\mathbf{P}(c_{i})) : T_i(\mathbf{P}(c_{i}))<r_i \}$
     - $T_{i+1}=T_{min}$, $P_{i+1}=P_{min}$
     - Go to step 3
9. Search result: $T_i(\mathbf{P})$

Throughput of depth-first sphere detector is a function of number of iterations through the algorithm and depends on the initial choice of the sphere radius $r_0$ and channel noise level and parameters. The number of iterations is incremented at step 3. It should be noted that if the initial sphere radius $r_0$ is too small the sphere constraint may not include the ML solution.
III. PERFORMANCE IMPROVEMENTS

A. Sphere Decoding for 64-QAM Modulation

The application of depth-first sphere decoding algorithm for detection of 16-QAM modulated signals has been previously reported in literature. Applying sphere decoding for detection of 64-QAM modulated signal streams requires overcoming a number of challenges. In this section we address two problems related to hardware implementation of the algorithm. In 64-QAM modulation the number of required memory cells for saving the intermediate results increases significantly compared to 16-QAM modulation. As an example, the memory space required to save the temporary search results in a 16-QAM implementation considering a 16-bit fixed point data representation is equal to 1536 bits. Under the same implementation constraints, the memory required for the 64-QAM implementation is 7680 bits which shows a 500% increase in the number of used memory elements. The second issue related to the implementation of sphere decoding algorithm for 64-QAM modulation is throughput. In depth-first sphere decoding, the throughput of the algorithm is related to the level of signal to noise ratio (SNR). The number of iterations through algorithm (or number of visited nodes in tree search graph) is a direct measure of the overall throughput of the algorithm. In the SD algorithm outlined in section II-B, a single iteration of the algorithm is considered to be going through steps 3 to 7. In a typical hardware implementation of sphere decoding algorithm this may take a number of clock cycles depending on the efficiency of implementation. In 64-QAM modulation the number of visited nodes increases significantly at lower SNR values; this is mainly due to the number of nodes that are saved in buffers at the early iterations of the algorithm when sphere radius is too big. The algorithm revisits all these nodes in the following iterations which amount to significant increase in the number of iterations.

B. Modified Norm Definition

In [6] a new norm definition for PED calculation is proposed which can reduce the algorithm complexity and improve the throughput. The proposed norm definition can also be applied to 64-QAM modulation. The \( \ell^p \)-norm definition in (7) can be approximated by \( \ell^\infty \)-norm definition (11).

\[
\begin{align*}
\|e_i(P)\| = \max(|\text{Re}(e_i(P))|, |\text{Im}(e_i(P))|) \\
\|\epsilon_i(P)\| = \max(\|\text{Real}(\epsilon_i(P))\|, |\text{Im}(\epsilon_i(P))|)
\end{align*}
\]

Simulation results show that using \( \ell^\infty \)-norm definition can decrease the number of iterations comparing to \( \ell^p \)-norm definition and the BER performance penalty is within acceptable range. The sphere decoding with \( \ell^\infty \)-norm results in sub-optimal detection. As a result of this observation we use \( \ell^\infty \)-norm for the modified versions of the algorithm.

C. Depth-First-2-Best (DF2B) Sphere Decoding

As mentioned before, one of the major challenges with 64-QAM sphere decoding, is the number of memory cells required to save the intermediate information. To address this issue we propose a combination of depth-first and best-first search strategy. In this approach at level 4 of the search tree all the children nodes that pass the sphere radius test are saved but in level 3 and 2 only the two children of each parent nodes with smallest associated PEDs are saved in the memory (Fig. 1). To apply the above modification, step 5 of the original sphere decoding algorithm needs to be changed such that only the first 2 elements of the sorted PEDs are saved and the rest are ignored. Simulations results show that the BER performance is close to original SD algorithm with \( \ell^\infty \)-norm definition but number of iterations has been significantly reduced at lower SNR values (< 25 dB). In this algorithm the required number of memory cells for storing intermediate results is reduced to 2720 bits which is a 65% reduction in memory usage comparing to the original implementation.

D. Level-Based Sphere Radius (LBSR)

In order to further reduce the number of visited nodes and iterations through the algorithm we need to apply a more effective tree pruning strategy to reduce the number of saved nodes at early stages of the search. In a tree graph eliminating a parent node removes all the children nodes connecting to it. A smarter method of finding parents nodes that their children will eventually fail the test helps in decreasing the number of visited node or iterations of the algorithm. In forward pruning strategy, the tree expansions that seem to fail are detected and pruned earlier. The real task in applying a forward pruning strategy is to identify which parent nodes are worth considering more closely by expanding to their children nodes and which can be pruned off with minimal risk of pruning the ML solution. Two main approaches can be considered for implementation a more effective pruning strategy in sphere decoding algorithm; faster shrinkage of the sphere radius or a new definition for sphere radius. In this paper we follow the later and propose a new definition of sphere radius which is a function of the tree level. The original SD algorithm with
dynamic radius update, updates the radius at level one of search tree; such that:

\[ r = \min(T(P)) \]  \hspace{1cm} (12)

This updated radius will be used to prune branches of tree at every level and will remain unchanged until algorithm reaches level 1 of the tree and finds a new result with smaller value. For further simplification, \(T_i(c_{\text{sd}})\) and \(P_i(c_{\text{sd}})\) are defined as:

\[ P_i(c_{\text{sd}}) = [c_{\text{sd}}, s_{\text{sd}}, \ldots, s_{\text{sd}}] \]  \hspace{1cm} (13)

\[ T(c_{\text{sd}}) = T_i(P_i(c_{\text{sd}})) \]  \hspace{1cm} (14)

In sphere decoding, the sphere radius test requires the following condition to be satisfied at every level:

\[ T(c_{\text{sd}}) < r \]  \hspace{1cm} (15)

In our proposed algorithm, at each level of the tree, PEDs are compared to the associated sphere radius for that level. The main idea is that if the PED of a parent node is growing faster than the previously detected smallest path, there is a higher risk for the children of that parent node to fail later in the search. As an example for a 4x4 system we have:

\[ P_1 = [c_{\text{sd}}, s_{\text{sd}}, s_{\text{sd}}, s_{\text{sd}}] \]  \hspace{1cm} (17)

\[ P_2 = [s_{\text{sd}}, s_{\text{sd}}, s_{\text{sd}}] \]  \hspace{1cm} (18)

\[ P_3 = [s_{\text{sd}}, s_{\text{sd}}] \]  \hspace{1cm} (19)

Due to definition of PED we have the following relationship for the sphere radius.

\[ r_1 \geq r_2 \geq r_3 \geq r_4 \]  \hspace{1cm} (20)

The effect of defining a sphere radius which is a function of the tree level is that at early stages of tree search \((l=4, 3)\), the PEDs are compared to smaller number than original definition of sphere radius and there is a higher possibility of pruning branches at earlier stages. Fig. 2 shows the effect of pruning of parent nodes at level 4 and level 3 of the search tree. In this case applying the new definition of sphere radius at level 4 eliminates close to 95% of nodes comparing to 37% elimination rate achieved by original sphere radius definition. At level 3 we observe a 17% improvement in the number of eliminated parent nodes. Simulation results show that the level based sphere radius (LBSR) is more effective in reducing the number of iterations when SNR is smaller than 25 dB. At higher SNR values the number of iterations merges to the minimum number of possible iterations which is four. Hence we only apply the new sphere radius definition for the smaller SNR values. In practice it would be easier to use the number of iterations as measure of applying the new definition rather than SNR which is not always an easy to measure or accessible parameter. In this case step 6 of SD algorithm changes to:

6. If \(i = 1\)

Update SR: if iteration \(< 5\)

\[ r = \min(T_{i=1}(P_{i=1})) \]  \hspace{1cm} (k=1,2,3,4)

Else

\[ r = T_{i=1}(P_{i=1}) \]

Save temporary search result: \(T_{i=1}, P_{i=1}\)

Go to step 7

Else ...

Fig. 2. Node elimination in level-based sphere radius

IV. SIMULATION RESULTS

Simulation results for bit error ratio (BER) and complexity as a function of signal to noise ratio (SNR) for the four algorithms discussed in section III has been presented for a 4x4 64-QAM MIMO system in Fig. 3 and Fig. 4. In Fig. 3, the complexity has been defined based on the number of iterations of the SD algorithm. In a typical hardware implementation for SD algorithm, the average decoder throughput is inversely proportional to the number of iterations. As the simulation results show, the original sphere decoding algorithm with \(l^\infty\)-norm and \(l^1\)-norm (SD-I2 and SD-linfl) has a poor performance at lower SNR values (SNR<25). In DF2B algorithm, number of iterations significantly decreases at low SNR values due to less number of saved nodes at earlier stage of search. At higher SNR the numbers of iterations for all the four cases converge to four which is the minimum number of iterations in a depth-first SD algorithm. Fig. 4 shows the BER performance of DF2B sphere decoding with original sphere radius and also level-based sphere radius (LBSR) compared to the performance of original SD algorithm with \(l^1\)-norm and \(l^\infty\)-norm. The original SD algorithm with \(l^1\)-norm definition (SD-I2) is ML type detector. The \(l^\infty\)-norm results in slightly higher BER at higher SNR compared to ML type detector. The BER performance of DF2B algorithms are close to \(l^\infty\)-norm BER performance. The performance degradation of DF2B with level based SR compared to the ML (SD-I2) detector at BER\(=10^{-3}\) is 1.3 dB while compared to SD-linfl, the degradation is 0.45 dB.
require modifications to the channel preprocessing as in FSD algorithm.

### Table I

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Original SD (l^2-norm)</th>
<th>Original SD (l^\infty-norm)</th>
<th>DF2B SD (l^2-norm)</th>
<th>DF2B SD (l^\infty-norm)</th>
<th>FSD-64 (1,1,1,64) (l^\infty-norm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of iterations (dB)</td>
<td>150</td>
<td>106</td>
<td>36</td>
<td>15</td>
<td>256</td>
</tr>
<tr>
<td>BER (dB)</td>
<td>1.2x10^{-3}</td>
<td>3.1x10^{-3}</td>
<td>3.6x10^{-3}</td>
<td>5.4x10^{-3}</td>
<td>1.2x10^{-3}</td>
</tr>
<tr>
<td>Number of memory cells (bits)</td>
<td>7680</td>
<td>7680</td>
<td>2720</td>
<td>2720</td>
<td>2640</td>
</tr>
</tbody>
</table>

### Acknowledgment

The authors would like to thank Telus Communications, Altera Corporation, Ontario Centres of Excellence (OCE), and Natural Sciences and Engineering Research Council (NSERC) for their financial support.

### References