Improved MIMO Detection based on Successive Tree Approximations

Jacob Goldberger
Engineering Faculty, Bar-Ilan University, Israel
Email: jacob.goldberger@biu.ac.il

***Abstract*—This paper proposes an efficient high-performance detection algorithm for MIMO communication systems that is based on a sequence of optimal tree approximations of the Gaussian density of the unconstrained linear system. The finite-set constraint is then applied to obtain a cycle-free discrete distribution that is suitable for message-passing algorithms. The proposed GTA-SIC algorithm is iterative and is based on first decoding the most reliable symbol, then canceling its contribution and applying the message-passing decoding to the smaller system. The computational complexity of the proposed GTA-SIC algorithm and the MMSE-SIC are comparable. The significantly improved MIMO decoding performance of the algorithm proposed here compared to lattice-reduction aided MMSE-SIC is demonstrated on several examples of large MIMO systems with high-order QAM constellations.**

I. Introduction

We consider a MIMO communication system with n transmit antennas and m receive antennas. The tap gain from transmit antenna j to receive antenna i is denoted by $\mathbf{H}_{ij}$. In each use of the MIMO channel a vector $x = (x_1, ..., x_n)^\top$ is independently selected from a finite set of complex numbers $\mathcal{A}$ according to the data to be transmitted, so that $x \in \mathcal{A}^n$. We further assume that in each use of the MIMO channel, x is uniformly sampled from $\mathcal{A}^n$. The received vector y is given by

$$y = \mathbf{H}x + \epsilon. \tag{1}$$

Here, noise is modeled by the random vector ϵ which is independent of x and whose components are assumed to be i.i.d. according to a complex Gaussian distribution with mean zero and with known variance σ^2. The $m \times n$ matrix $\mathbf{H}$ comprises i.i.d. elements drawn from a complex normal distribution of unit variance. The MIMO detection problem consists of finding the unknown transmitted vector x given $\mathbf{H}$ and y. The task, therefore, boils down to solving a linear system in which the unknowns are constrained to a discrete finite set. The maximum likelihood (ML) solution is

$$\hat{x} = \arg\min_{x \in \mathcal{A}^n} \|\mathbf{H}x - y\|^2. \tag{2}$$

However, going over all the $|\mathcal{A}|^n$ vectors is practically unfeasible when either n or $|\mathcal{A}|$ are large. In fact, the MIMO decoding problem is known to be NP hard [1].

A simple sub-optimal solution, known as the Zero-Forcing (ZF) algorithm, is based on a linear decision that ignores the finite-set constraint and then, neglecting the correlation between the symbols, finding the closest point in $\mathcal{A}$ for each symbol independently. This scheme performs poorly due to its inability to handle ill-conditioned realizations of matrix $\mathbf{H}$ (if $m < n$ then $\mathbf{H}^\top\mathbf{H}$ is even singular). Somewhat better performance can be obtained by using a minimum mean square error (MMSE) Bayesian estimation for the continuous linear system. A vast improvement over the linear approaches described above can be achieved by the MMSE with Successive Interference Cancelation (MMSE-SIC) algorithm which is based on sequential decoding with optimal ordering [2].

Many alternative methods have been proposed to approach the ML detection performance. The sphere decoding (SD) algorithm finds the exact ML solution by searching for the nearest lattice point [1], [3]. Although SD reduces computational complexity compared to the exhaustive search of the ML solution, sphere decoding is not feasible for high-order QAM constellations. While SD has been empirically found to be computationally very fast for small to moderate problem sizes, the sphere decoding complexity is prohibitive for large n, higher order QAM and/or low SNRs [4].

A preprocessing step based on lattice reduction (LR) has been proposed in order to enhance the performance of low-complexity suboptimal detectors and decrease time complexity of tree-search sphere decoding [5]. The performance gap of ML detection and LR based linear decoders increases greatly for a large number of antennas.

This study attempts to solve the MIMO decoding problem using the Belief Propagation (BP) paradigm. We provide an algorithm that is both efficient and achieves near-optimal results. A recent study proposed a modified version of the MMSE algorithm based on Gaussian tree approximations (GTA) [6]. In this study we take this approach one step further and develop an iterative version of GTA we call the GTA-SIC, which is based on successive interference cancelation. We derive the optimal ordering of symbol decoding and obtain an efficient version of the BP algorithm that is suitable for our problem. The GTA-SIC algorithm is both computationally efficient and achieves better results than state-of-the-art methods that are based on lattice reduction.

II. The Gaussian Tree Approximation Algorithm

In this section we briefly review the Gaussian tree approximation (GTA) method for MIMO decoding [6] and develop an explicit expression for the error covariance matrix of the GTA method.

Given the constrained linear system $y = \mathbf{H}x + \epsilon$, and a uniform prior distribution on x over a finite set of points $\mathcal{A}^n$, the probability function of the discrete random vector x given y is

$$p(x_1, .., x_n|y) \propto \exp\left(-\frac{1}{2\sigma^2}\|\mathbf{H}x - y\|^2\right), \qquad x \in \mathcal{A}^n. \quad (3)$$

It can be easily verified that (3) can be written as follows:

$$p(x|y) \propto \exp\left(-\frac{1}{2}(x-z)^\top \mathbf{C}^{-1}(x-z)\right), \qquad x \in \mathcal{A}^n \quad (4)$$

where $z = (\mathbf{H}^\top\mathbf{H})^{-1}\mathbf{H}^\top y$ is the least-squares estimator and $\mathbf{C} = \sigma^2(\mathbf{H}^\top\mathbf{H})^{-1}$ is its variance.

We next look for a good approximation of $p(x|y)$ (4) that enables a successful implementation of the Belief Propagation paradigm. Since the BP algorithm is optimal and efficient on connected cycle-free factor graphs, i.e., on trees, a reasonable approach is to find an optimal tree approximation of the exact distribution (3). This problem is, unfortunately, NP hard [6].

To overcome this obstacle, we search for a tree approximation of the distribution corresponding to the unconstrained version of the linear system (1):

$$f(x|y) \propto \exp\left(-\frac{1}{2}(x-z)^\top \mathbf{C}^{-1}(x-z)\right), \qquad x \in \mathcal{R}^n. \quad (5)$$

This distribution is Gaussian and therefore, as we show below, we can find a closed-form expression for the optimal Gaussian tree approximation. We can then apply the finite-set constraint and utilize the Gaussian tree distribution to form a discrete loop-free approximation of $p(x|y)$ (3) which can be efficiently globally maximized using the BP algorithm.

An n-node tree graph can be represented by the cycle-free parent relations $\{p(i)\}_{i=1}^n$ such that $p(i)$ is the parent node of i. Each node, excluding the tree root, has exactly one parent. To simplify notation we do not describe the root node separately. The parent of the root is implicitly assumed to be the empty set. A distribution $g(x_1, ..., x_n)$ is aligned with a tree $\{p(i)\}$ if it can be written as $g(x) = \prod_{i=1}^n g(x_i|x_{p(i)})$. The following theorem provides an explicit formula for the optimal Gaussian tree approximation of a given Gaussian distribution $f(x)$.

Theorem 1: [6] Let $f(x) = f(x_1, ..., x_n)$ be a multivariate Gaussian distribution. The Gaussian tree distribution $g(x) = \prod_{i=1}^n g(x_i|x_{p(i)})$ whose KL divergence $KL(f||g)$ is minimal is:

$$f_{gta}(x) = \prod_{i=1}^n f(x_i|x_{p(i)}), \qquad x \in R^n \quad (6)$$

and the tree structure of f_{gta} is the maximum spanning tree of the weighted n-node graph where the weight of the ij edge is the square of the correlation coefficient between x_i and x_j (based on $f(x)$). The Gaussian density $f_{gta}(x)$ with the optimal tree structure is called the Gaussian Tree Approximation (GTA) of $f(x)$.

A spanning tree of a connected graph is a subgraph that contains all the vertices and is a tree. Prim's algorithm [7] is an efficient and simple greedy approach to find the maximum spanning tree. It begins with some vertex v in the given graph, defining the initial set of vertices T. Then, in each iteration, we choose the edge with maximal weight out of all the edges (u, v), where u is outside of T and v is in T. Then vertex u is brought into T. This process is repeated until a spanning tree is formed. A simple implementation of Prim's algorithm based on searching an array of weights to find the maximum weight edge to add requires $O(n^2)$ operations. By applying the Prim algorithm we obtain the tree structure of the optimal Gaussian tree approximation.

Given the Gaussian tree approximation (6) of the Gaussian distribution $f(x|y)$ (5), the next step is applying the finite-set constraint to form a discrete loop-free approximation of $p(x|y)$ (3):

$$p_{gta}(x_1, ..., x_n|y) \propto \prod_{i=1}^n f(x_i|x_{p(i)}), \qquad x \in \mathcal{A}^n. \quad (7)$$

Using the BP algorithm we can efficiently obtain the exact marginal distributions of $p_{gta}(x|y)$:

$$p_{gta}(x_i = a|y), \qquad a \in \mathcal{A}, \quad i = 1, ..., n \quad (8)$$

that provide a soft decision result. Taking the most likely symbol, we obtain the GTA algorithm.

Theorem 2: Let $f(x) = f(x_1, ..., x_n)$ be a multivariate Gaussian distribution with mean z and covariance matrix $\mathbf{C}$. Let

$$g(x) = \prod_{i=1}^n f(x_i|x_{p(i)}), \qquad x \in R^n \quad (9)$$

be a Gaussian tree approximation of $f(x)$ defined by a spanning tree $\{p(i)\}$ (not necessarily the optimal tree). Denote the mean and variance of the Gaussian distribution $g(x)$ by z_g and $\mathbf{C}_g$ respectively. Then $z_g = z$ and

$$\mathbf{C}_g(i,j) = \mathbf{C}(i,i)\prod_{s=0}^{k-1}\frac{\mathbf{C}(v_s, v_{s+1})}{\mathbf{C}(v_s, v_s)} \quad (10)$$

such that $i = v_0, v_1, ..., v_{k-1}, v_k = j$ is the (unique) tree path from x_i to x_j.

Proof: Assume, without loss of generality, that the tree root is x_1. We prove that $f(x_i) = g(x_i)$ for every i by induction on the path length from the root x_1 to x_i. For the root node x_1, the definition of $g(x)$ implies that $g(x_1) = f(x_1)$. Let x_j be the parent of x_i in the tree representation of $g(x)$, i.e., $j = p(i)$. The induction assumption implies that $g(x_j) = f(x_j)$ and the definition of $g(x)$ implies that $g(x_i|x_j) = f(x_i|x_j)$. Therefore, $g(x_i, x_j) = f(x_j, x_j)$ which implies that $g(x_i) = f(x_i)$. To prove the covariance formula (10) for $\mathbf{C}_g(i,j)$ we use an induction on the length of the tree path from i to j. For path lengths that are either zero or one we showed above that $g(x_i, x_j) = f(x_i, x_j)$ and therefore $C_g(i,j) = C(i,j)$. The triplet of jointly Gaussian random variables x_i, x_{v_1}, x_j, based on the density $g(x)$, satisfies:

$$\text{cov}(x_i, x_j|x_{v_1}) = \mathbf{C}_g(i,j) - \frac{\mathbf{C}_g(i,v_1)\mathbf{C}_g(v_1,j)}{\mathbf{C}_g(v_1,v_1)} \quad (11)$$

Node v_1 is part of the tree path from i to j and therefore $\text{cov}(x_i, x_j|x_{v_1}) = 0$. Since the path from i to j is longer than the path from v_1 to j, we can apply the induction assumption on $\mathbf{C}_g(v_1, j)$ in (11) to obtain the covariance expression (10). □

Applying Theorem 2 to the optimal tree approximation f_{gta} defined in Theorem 1, we obtain that $f(x_i, x_j) = f_{gta}(x_i, x_j)$ for every x_i, x_j that are connected by an edge in the tree graph of f_{gta}. Hence, the ij entry of the covariance matrix of $f_{gta}(x)$ is equal to the ij entry of the covariance matrix of $f(x)$.

The MMSE is known to be a better MIMO decoding method than the ZF solution. An MMSE version of the GTA algorithm can be obtained by finding the optimal Gaussian tree approximation of:

$$f(x|y) \propto \exp\left(-\frac{1}{2}(x-E_{x|y})^\top V_{x|y}^{-1}(x-E_{x|y})\right) \qquad (12)$$

such that $E_{x|y} = (\mathbf{H}^\top\mathbf{H}+\frac{\sigma^2}{e}I)^{-1}\mathbf{H}^\top y$ and $V_{x|y} = \sigma^2(\mathbf{H}^\top\mathbf{H}+\frac{\sigma^2}{e}I)^{-1}$ where e is the mean symbol energy..

III. The Successive Interference Cancelation GTA

In this section we develop a successive interference cancelation (SIC) version of the GTA algorithm. We dub this algorithm the GTA-SIC. The main idea is similar to MMSE-SIC. The difference is that instead of applying the MMSE to obtain a hard decision of one symbol at a time, we apply the GTA algorithm. Applying a SIC procedure to the GTA is done in the following way. In each iteration we apply the GTA algorithm and obtain a hard decision of the single (most reliable) symbol. We cancel its contribution and obtain a system with a smaller number of unknown variables. This process is iterated until all the message symbols are decoded. The MMSE-SIC decoder is significantly better than the MMSE. The GTA algorithm can be viewed as an improved version of the MMSE based on Gauss-Markov density instead of the product of marginal Gaussian densities. In terms of performance, GTA outperforms MMSE. It is even better than MMSE-SIC [6]. In this study we show that a SIC version of the GTA algorithm provides a significant improvement over MMSE-SIC with only small increase in computational complexity.

A. Efficient message scheduling

Since at each iteration we are only decoding a single symbol, we show next that there is no need to apply all the message-passing steps of the BP algorithm used by the GTA decoder. An optimal BP schedule, when applied to a tree, requires passing a message once in each direction of each edge [8]. The BP messages are first sent from the leaf variables 'downward' to the root. Next, BP messages are sent 'upward' from the root back to the leaves. After the downward-upward message passing procedure is completed, we can compute the 'belief' at each node.

In our case of GTA combined with successive interference cancelation, in each round we are only interested in decoding a single node. Hence, the BP algorithm boils down to an instance of dynamic-programming. When applying BP to a tree graph we can consider each vertex as the tree root. Hence, without any loss of generality we can assume that the single node we want to decode is the root of the tree. Since we only want to compute the belief of the root, we can avoid the 'upward' part of the BP scheduling.

Assume we are given the following tree distribution rooted at x_1:

$$p_{gta}(x_1, ..., x_n|y) \propto \prod_{i=1}^{n} f(x_i|x_{p(i)}), \qquad x \in \mathcal{A}^n,$$

and we want to find the marginal distribution $p_{gta}(x_1|y)$. The 'downward' BP message from a variable x_i to its parent variable $x_{p(i)}$ is computed based on all the messages x_i received from its children

$$m_{i\to p(i)}(x_{p(i)}) = \sum_{x_i\in\mathcal{A}} f(x_i|x_{p(i)}) \prod_{j|p(j)=i} m_{j\to i}(x_i). \quad (13)$$

The 'belief' in the root node is:

$$\text{belief}(x_1) = f(x_1) \prod_{j|p(j)=1} m_{j\to 1}(x_1), \qquad x_1 \in \mathcal{A}. \quad (14)$$

To obtain the hard-decision decoding for the next round of the iterative procedure (see Fig. 1), we choose the symbol value whose posterior probability is maximal

$$\hat{x}_1 = \arg\max_{a\in\mathcal{A}} \text{belief}(a). \quad (15)$$

B. Optimal Ordering

Successive canceling algorithms can suffer from error-propagation. If a symbol is estimated incorrectly it can have an adverse effect on estimation of the remaining unknown symbols. To minimize the effects of error propagation, it is advantageous to perform interference canceling from the most reliable to the least reliable transmitted signal. This is the decoding order proposed by the V-BLAST method [2]. In [2] it is proved that this greedy local strategy is also globally optimal.

To perform optimal interference cancelation ordering in the MMSE-SIC algorithm, we consider the covariance matrix of the estimation error

$$\mathbf{C} = E((x-\hat{x})(x-\hat{x})^\top|y) = \sigma^2(\mathbf{H}^\top\mathbf{H} + \frac{\sigma^2}{e}I)^{-1}. \quad (16)$$

Matrix $\mathbf{C}$ is the variance of the linear estimation ignoring the finite-set constraint and $\hat{x} = (\mathbf{H}^\top\mathbf{H} + \frac{\sigma^2}{e}I)^{-1}\mathbf{H}^\top y$ is the MMSE unconstrained solution. The most reliable decoded symbol, is the x_i with the smallest error variance, i.e., x_i for which c_{ii} is the smallest. The MMSE-SIC is based on hard decoding of x_i and canceling its contribution from the linear system.

From Theorem 2 we obtain that the variance of the decoded symbol x_i is the same for the two decoding algorithms, MMSE and GTA (and is equal to the c_{ii} diagonal entry of the matrix $\sigma^2(\mathbf{H}^\top\mathbf{H}+\frac{\sigma^2}{e}I)^{-1}$. Hence the most reliable symbol for GTA-SIC is the one with the minimum error variance, i.e, the symbol x_i such that c_{ii} is minimal. The same argument which

Input: A constrained linear system: $\mathbf{H}x + \epsilon = y$ where $\mathbf{H}$ is an $m \times n$ matrix, a noise level σ^2 and a finite symbol set $\mathcal{A}$ whose mean symbol energy is denoted by e.

Algorithm:

For $k = n, n-1, ..., 1$

- **Compute:**

$$\begin{aligned} z &\leftarrow (\mathbf{H}^\top \mathbf{H} + \frac{\sigma^2}{e} I)^{-1} \mathbf{H}^\top y \\ \mathbf{C} &\leftarrow \sigma^2 (\mathbf{H}^\top \mathbf{H} + \frac{\sigma^2}{e} I)^{-1}. \end{aligned}$$

- **Optimal Ordering**: Assume that c_{kk} is the smallest diagonal element of $\mathbf{C}$ (otherwise relabel the variables).
- **Optimal Tree**: Compute the maximum spanning tree (rooted at node x_k) of the k-node graph where the weight of the i-j edge is the square of the correlation coefficient $c_{ij}^2/(c_{ii}c_{jj})$. Denote the parent of node i by $p(i)$.
- **Max Product**: Send messages from leaves to root. The message from i to its parent $j = p(i)$ is:

$$m_{i\to j}(x_j) = \max_{x_i \in \mathcal{A}} \left(-\frac{(x_i - z_i - \frac{c_{ij}}{c_{jj}}(x_j - z_j))^2}{c_{ii} - \frac{c_{ij}^2}{c_{jj}}} + \sum_{s|p(s)=i} m_{s\to i}(x_i) \right), \qquad x_i \in \mathcal{A}$$

- **Hard decision**:

$$\hat{x}_k = \arg\max_{a \in \mathcal{A}} \left(-\frac{(a - z_k)^2}{c_{kk}} + \sum_{s|p(s)=k} m_{s\to k}(a) \right).$$

- **Interference Cancelation**: Assume $\mathbf{H} = (h_1, ..., h_k)$.

$$\begin{aligned} y &\leftarrow y - h_k \hat{x}_k \\ \mathbf{H} &\leftarrow (h_1, ..., h_{k-1}) \end{aligned}$$

End

Fig. 1. The Gaussian Tree Approximation with Successive Interference Cancelation (GTA-SIC) Algorithm.

demonstrated that for V-BLAST the greedy local strategy is also the globally optimal, can be utilized to show that same is true for GTA-SIC optimal ordering. Therefore, the same ordering that is optimal for MMSE-SIC is also optimal for GTA-SIC.

C. Implementation and complexity

The belief-propagation algorithm has two variants, the sum-product and the max-product [8]. Above we described a sum-product version of GTA algorithm that computes soft marginal probabilities $p_{gta}(x_i|y)$. We did not observe any significant performance difference using either the sum-product or the max-product variants of GTA-SIC for decoding the MIMO systems that are presented in the experiment section. The max-product is more computationally efficient since the BP messages can be entirely computed in the log-domain whereas the sum-product is based on 'log-sum-exp' type operations. A detailed implementation of the max-product version of the proposed GTA-SIC algorithm is shown in Fig. 1.

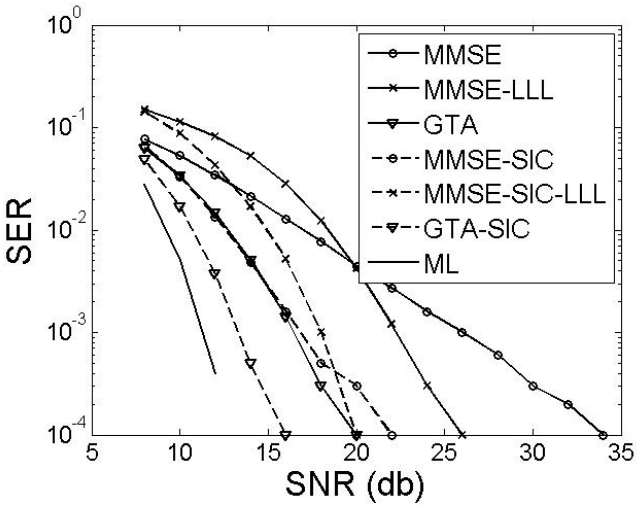

Fig. 2. Results for a 12×12, QPSK, MIMO system.

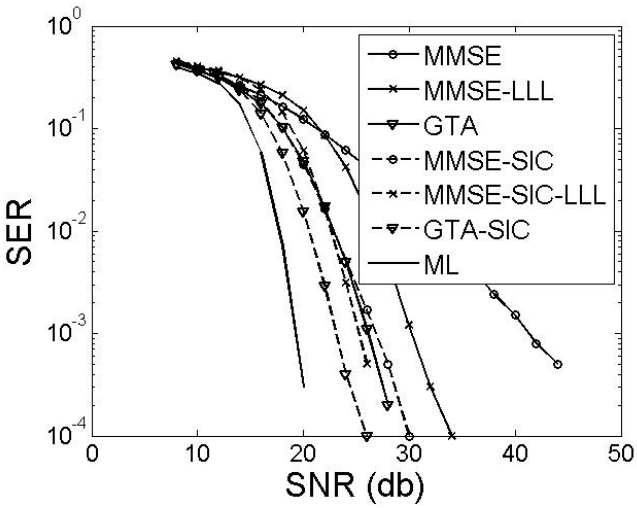

Fig. 3. Results for a 12×12, 16-QAM, MIMO system.

The computational complexities of both MMSE-SIC and GTA-SIC are dominated by the complexity of solving the underlying unconstrained least-squares problem, i.e., calculating the inverse of $(\mathbf{H}^\top \mathbf{H} + \frac{\sigma^2}{e} I)$ at each step of the algorithm. When $m = n$, (i.e. same numbers of transmit and received antenna) we need to evaluate the inverse of a series of matrices with dimensions $k \times k$, where $k = n, n-1, ..., 1$. The computational complexity of performing this series of operations is clearly of the fourth order, i.e., $O(n^4)$. In the GTA-SIC there are two additional operations that do not exist in MMSE-SIC,

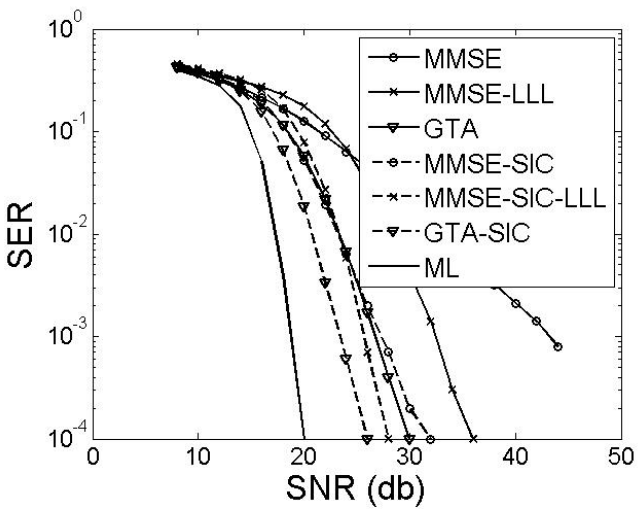

Fig. 4. Results for a 16×16, 16-QAM, MIMO system.

namely finding the optimal tree and applying the BP algorithm in each iteration. The computational complexity of applying Prim's algorithm to find n optimal trees is $O(n^3)$. It can be easily verified from Fig. 1 that the dynamic programming performed by the BP algorithm takes $k|\mathcal{A}^2|$ operations. When applying the BP algorithm on graphs of sizes $k = n, n–1, ..., 1$. the computational complexity of the BP step in the GTA-SIC algorithm is $O(n^2|\mathcal{A}^2|)$. Hence the complexity of the GTA-SIC algorithm is dominated by the factor $O(n^4)$ which is the same complexity as MMSE-SIC. By applying the square-root algorithm proposed by Hassibi [9] for efficient computation of error covariance matrices, the computational complexity can be reduced from $O(n^4)$ to $O(n^3)$.

IV. Experimental Results

In this section we provide simulation results for the proposed GTA-SIC detector over various MIMO systems. We assume a frame length of 100, i.e., channel matrix $\mathbf{H}$ is constant for 100 channel uses. The channel matrix comprised iid elements drawn from a zero-mean normal distribution of unit variance. We used 10,000 realizations of the channel matrix. This resulted in 10^6 vector messages. The performance of the proposed algorithm is shown as a function of the variance of the additive noise σ^2. The signal-to-noise ratio (SNR) is defined as $10\log_{10}(E_s/N_0)$ where $E_s/N_0 = \frac{ne}{\sigma^2}$ (n is the number of variables, $e = \frac{1}{|\mathcal{A}|}\sum_{a\in\mathcal{A}} a^2$ and σ^2 is the variance of the Gaussian additive noise).

Here we show the performance of the proposed GTA-SIC approach based on successive Gaussian tree approximations. The GTA-SIC is first compared to MMSE, GTA [6] and MMSE-LLL which is a MMSE combined with lattice reduction [5] based on the Lenstra-Lenstra-Lovasz (LLL) algorithm (with δ=3/4) [10] that is applied to the extended channel matrix [11]. Then, GTA-SIC is compared to MMSE-SIC and its lattice reduction variant (using optimal ordering) denoted by MMSE-SIC-LLL [11]. This method achieves the best results among LR based MIMO detection methods [5]. Finally, GTA-SIC is compared to maximum-likelihood (ML) detection. The ML score was implemented using the Schnorr-Euchner variant of sphere decoding (SD-SE) [12], [3].

Fig. 2 shows the symbol error rate (SER) versus SNR for a 12×12 QPSK MIMO system. Figures 3-4 show SER vs. SNR results for several other system sizes and constellations. First, we can revalidate the results reported in [6] that the GTA algorithm is much better than MMSE and is comparable and slightly better than MMSE-SIC. The results also show that the MMSE-LLL is better than MMSE in high SNR, however it is worse than MMSE-SIC. This already indicates the potential of the GTA-SIC algorithm. In all experiment results shown above the GTA-SIC performed significantly better than the MMSE-SIC. The GTA-SIC was also found to be better than the MMSE-SIC-LLL. Note that even in the case of high QAM constellation and a large number of antennas (Fig. 4), the improvement of GTA-SIC over MMSE-SIC and MMSE-SIC-LLL is significant.

To summarize, we showed that our approach is more suitable to MIMO detection than the state-of-the-art lattice reduction aided algorithms. We showed that using a comparable computational complexity we can achieve significantly improved MIMO decoding performance especially in large MIMO systems with high-order QAM constellations.

References

[1] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Mathematics of Computation*, vol. 44, no. 170, pp. 463–471, 1985.

[2] G. D. Golden, G. J. Foschini, R. A. Valenzuela, and P. W. Wolniansky, "Detection algorithm and initial laboratory results using V-BLAST space-time communication architecture," *Electron. Letters*, vol. 35, no. 1, pp. 14–16, 1999.

[3] C. Schnorr and M. Euchner, "Lattice basis reduction: Improved practical algorithms and solving subset sum problems," *Journal of Mathematical Programming*, vol. 66, no. 1-3, pp. 181–199, 1994.

[4] J. Jalden and B. Ottersten, "On the complexity of sphere decoding in digital communications," *IEEE Trans. Signal Processing*, vol. 53, no. 4, pp. 1474–1484, 2005.

[5] D. Wübben, D. Seethaler, J. Jalden, and G. Matz, "Lattice reduction: A survey with applications in wireless communications," *IEEE Signal Processing Magazine*, vol. 28, no. 3, pp. 70–91, 2011.

[6] J. Goldberger and A. Leshem, "MIMO detection for high-order QAM based on a Gaussian tree approximation," *IEEE Trans on Information Theory*, vol. 57, no. 8, pp. 4973–4982, 2011.

[7] R. C. Prim, "Shortest connection networks and some generalizations," *J. Bell System Tech.*, pp. 1389–1401, 1957.

[8] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[9] B. Hassibi, "An efficient square-root algorithm for BLAST," *Proc. IEEE Intl. Conf. Acoustic, Speech, Signal Processing*, pp. 5–9, 2000.

[10] A. K. Lenstra, H. W. Lenstra, and L. Lovasz, "Factoring polynomials with rational coefficients," *Math Ann.*, vol. 261, no. 3, pp. 515–534, 1982.

[11] D. Wübben, R. Böhnke, V. Kühn, and K. Kammeyer, "Near-maximum-likelihood detection of MIMO systems using MMSE-based lattice reduction," *IEEE Int. Conf. Communications*, 2004.

[12] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Trans. Inf. Theory*, vol. 48, no. 8, pp. 2201–2214, 2002.