A near-ML Performance Adaptive Dijkstra Algorithm for Large Scale MIMO Detection

Une quasi ML performance par un algorithme adapté de Dijkstra pour une detection MIMO à grande échelle

Wassim Hellassa* & Karima Boukari

Faculty of Science of Engineering, Badji Mokhtar University , PO Box 12, Annaba, 23000, Algeria.

Article Info

Article history:

Received 10/12/2018
Revised 15/10/2019
Accepted 12/11/2019

Keywords

MIMO communications, Signal detection, tree search algorithm, optimisation.

Mots clés

Communications MIMO, détection de signal, algorithme de recherche d'arbre, optimisation.

ABSTRACT

Employing Maximum Likelihood (ML) algorithm for signal detection in large scale Multiple-Input- Multiple Output (MIMO) system with high modulation order is a computationally expensive approach. In this paper an adaptive search algorithm is proposed for ML detection based MIMO receiver that can be classified as a derivative of Dijkstra's Search (DS) algorithm based best first search algorithm; hence naming Adaptive Dijkstra's Search (ADS) algorithm. The proposed ADS exploits the resources available in the search procedure to reduce the required number of nodes to be visited in the tree. Results are obtained depending on a tunable parameter, which is defined to control the number of the best possible candidate nodes. Unlike the conventional DS, the ADS algorithm results in signal detection with low computation complexity and quasi-optimal performance for systems under low and medium SNR regimes. Simulation results demonstrate 25% computational complexity reduction, compared to the conventional DS. For Symbol Error Rate (SER) of 10^{-2}, such computation complexity reduction is also a trade-off with 2 dB SNR degradation, while attaining the same SER with conventional DS. The reduction of the computation complexity with the proposed ADS is non-linearly proportional to the dimension of MIMO combination as well as the modulation order.

RESUMÉ

Utiliser l'algorithme de vraisemblance maximale (ML) pour la détection de signal dans un système MIMO (Multiple-Input-Multiple Output) à grande échelle et avec un ordre de modulation élevé est une approche coûteuse en calcul. Dans cet article, un algorithme adaptatif de recherche est proposé pour la détection ML dans un récepteur MIMO, qui peut être considéré comme un dérivé de l'algorithme de recherche de Dijkstra (DS); d'où le nom de l'algorithme ADS (Adaptive Dijkstra's Search). L'ADS proposé exploite les ressources disponibles dans la procédure de recherche pour réduire le nombre de nœuds à visiter dans l'arbre de recherche. Les résultats sont obtenus en fonction d'un paramètre ajustable, défini pour contrôler le nombre des meilleurs nœuds candidats possibles. Contrairement à la DS conventionnelle, l'algorithme ADS permet une détection de signal avec une complexité de calcul faible et des performances quasi optimales pour les systèmes sous un SNR faible et moyen. Les résultats de la simulation démontrent une réduction de la complexité de 25% par rapport à la DS classique. Pour un taux d'erreur de symbole (SER) de 10^{-2}, une différence de 2 dB à la faveur l'algorithme ADS proposé par rapport à la DS conventionnelle. La réduction de la complexité de calcul avec l'ADS proposé est proportionnelle de manière non linéaire à la dimension de la combinaison MIMO ainsi qu'à l'ordre de modulation.

Corresponding Author
Wassim Hellassa
Faculty of Science of Engineering, Badji Mokhtar University, Po box 12, Annaba, 23000, Algeria.
Email: wassim.hellassa@gmail.com

1. INTRODUCTION

Multiple Input Multiple Output (MIMO) system is one of the major technologies adopted by current wireless communication standards such as Third Generation Partnership Project Long-Term Evolution (3GPP LTE), IEEE 802.11n, IEEE 802.16e as well as IEEE 802.11ac [1]. Employing the MIMO technology results in an increased data rate and reliability of the communication systems [2], without compromising the bandwidth or the signal power. However, it results in an additional signal processing and computations in the receiver. Hence, the application of MIMO technology in real time systems is expanded to be constrained with such additional computational complexity, while providing an intended quality of service (QoS) [3]. To attain a possible enhancement in data rate, the detection problem of spatial multiplexed MIMO signal has attracted many researchers' attentions in the recent years. The main aim of the reported research is to optimally recover the transmitted signals with a reasonable complexity for hardware implementation. In response to that, a wide range of detection techniques have been proposed in the existing literature[4-6]. By an exhaustive search over all the possible combination of the transmitted signals, Maximum likelihood (ML) detection scheme, can be argued to be the best detection technique which may not be feasible for higher dimension of MIMO system or high modulation order [7] [8]. To overcome the increasing computation complexity during detection, there are some techniques from the existing literature such as the linear detection algorithm family such as Zero-Forcing (ZF), Minimum Mean Squared Error (MMSE) [2] or the iterative detection using the Vertical Bell Laboratories Layered Space-Time (VBLAST) technique. Using these techniques decreases dramatically the detection complexity compared to the ML technique, but at the expense of a significant decrease in the detection performance. By representing the MIMO signal detection as a tree search [9], a wide range of optimal and suboptimal techniques have been proposed in the literature with different tradeoff results between performance and complexity. The tree representation reformulates the MIMO signal detection problem to find the shortest possible branch from the top to the bottom of the tree within optimal detection schema. Some algorithms initially based on finding the shortest path [10], are now adapted for the MIMO detection problem based on a tree search. These techniques can be classified into three different categories of tree search: depth first, breadth first and best first. In the MIMO signal detection, based on a tree search, all the proposed algorithms aimed at reducing the number of visited nodes and the total computation cost required for establishing the optimal solution. The Dijkstra's algorithm [10–12] reduces the number of visited nodes by maintaining a list of candidate nodes and equally search among them in the order of the best first. Despite the optimal performance obtained, the search involves to visiting unnecessary nodes with equal importance to each node. Hence, the number of visited nodes and the memory requirement remain high for practical implementation of such algorithm in the systems with high modulation order.

In this paper, a new MIMO signal detection algorithm has been proposed based on the best first category of a tree search. The proposed algorithm is adaptive; it defines some criteria to reduce the number of the combinations to be evaluated and visits only the most likely combination to result in an optimal solution and hence, a large amount of calculation can be avoided. The proposed algorithm is an adaptive version of the Dijkstra's algorithm with respect to transmit-receive antenna and modulation constellation dimension. The complexity reduction within the proposed algorithm is directly proportional to the modulation order with quasi-optimal performance in the low SNR regime. The proposed algorithm is hereafter named as Adaptive Dijkstra's algorithm (ADS). The rest of the paper is organized as follows. Section 2 presents the system model and conventional Djikstra's algorithm. The proposed algorithm is presented in section 3. Simulation results and discussions are presented in section 4. Finally, conclusions are given in section 5.

2. SYSTEM MODEL AND DIJKSTRA'S ALGORITHM

2.1. System model

We consider a spatial multiplexed MIMO system with Nt transmitter antennas and Nr receiver antennas with Rayleigh fading channel, the received signal vector can be expressed as:

$$y_c = H_c \times s_c + v_c \tag{1}$$

Where y_c is (Nr ×1)-dimensional vector and represents the received signal, s_c which is a (Nt×1)-dimensional vector represents the transmitted signal.Its elements are drawn from a set of complex elements such as M-QAM constellation where M is the modulation order, H_c is a (Nr × Nt) matrix representing the channel with independent and identically distributed (i.i.d.) Gaussian entries with zero mean and unitary variance and v_c is (Nr ×1)-dimensional vector of noise with an i.i.d complex entries with zero mean and variance σ_n^2. Figure 1 shows tree search for 2x 2 MIMO system with 16-QAM modulation m=2xNt and $\Omega = \{+3, +1, -1, -3\}$

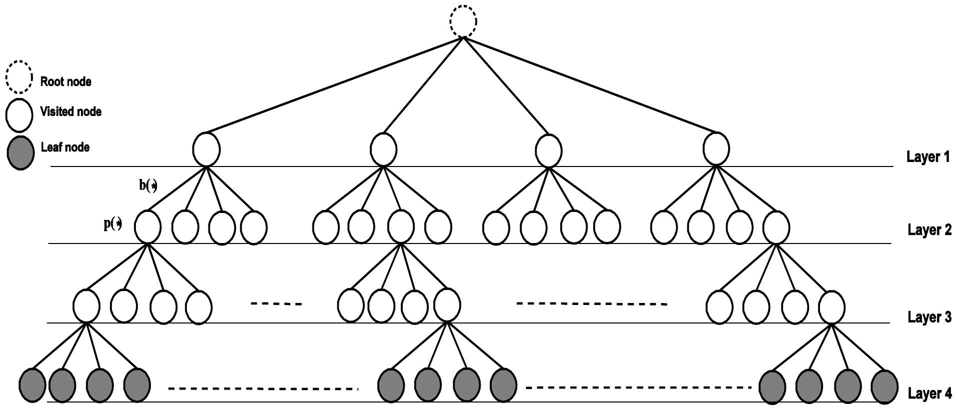


Figure 1. Tree search for 2x 2 MIMO system with 16-QAM modulation m=2xNt and $\Omega = \{+3, +1, -1, -3\}$

For simplicity, the number of transmit and receive antennas are assumed to be symmetrical, i.e. Nt=Nr, the channel state information is assumed to be known to the receiver.

The equivalent real presentation of the system (1) is defined by

$$\begin{bmatrix} \Re(y_c) \\ \Im(y_c) \end{bmatrix} = \begin{bmatrix} \Re(H_c) & -\Im(H_c) \\ \Im(H_c) & \Re(H_c) \end{bmatrix} \times \begin{bmatrix} \Re(s_c) \\ \Im(s_c) \end{bmatrix} + \begin{bmatrix} \Re(v_c) \\ \Im(v_c) \end{bmatrix} \tag{2}$$

Where $\Re(*)$ and $\Im(*)$ denote the real and the imaginary parts of its elements. The equivalent real representation of the system model in (2) is presented as follows:

$$y = H \times s + v \tag{3}$$

Where: $y \in R^n$, $s \in R^m$, $H \in R^{n \times m}$, $v \in R^n$ with $n = 2 \times N_r$, $m = 2 \times N_t$

2.2. MIMO Detection based on tree search

In the MIMO detection, the maximum likelihood ML detection achieves the optimal bit error rate performance by solving the minimization problem:

$$\hat{s} = \arg \min_{s \in \Omega^m} \| y - Hs \|^2 \tag{4}$$

Where: Ω is the constellation in the real valued system model, for example, in 16-QAM $\Omega = \{+3, +1, -1, -3\}$.

The ML detection makes an exhaustive search over all the candidates of "s". Hence, the complexity of the detection increases exponentially with the number of antennas and the modulation order M, making it impractical for real time implementations. By applying the QR decomposition on the channel matrix, (4) is reformulated by the equivalent expression (5).

$$\hat{s} = \arg \min_{s \in \Omega^m} \| y - Hs \|^2 \text{ with } P(s) = \| y' - Rs \|^2 \text{ and } y' = Q^H R \tag{5}$$

Where Q is a unitary matrix and R is an upper triangular matrix.

Because of the upper triangular matrix R, P(s) in (5) can be calculated in a recursive process as shown in (6). Thus, the MIMO detection is reformulated.

$$P(s_k^l) = P(s_{k-1}^l) + B(s_k^l) \text{ with } B(s_k^l) = (y'_k - \sum_{j=k}^{m} r_{k,j} s_j)^2 \tag{6}$$

Where: y'_k and s_j are the real elements of y and s respectively and $r_{k,j}$ is the (k, j)-entry of R.

Figure 2. shows an illustration of the Dijkstra's search algorithm (DS) for MIMO signal detection based on tree search. The number between brackets represents the path metric $P(s_k^l)$ of the node.

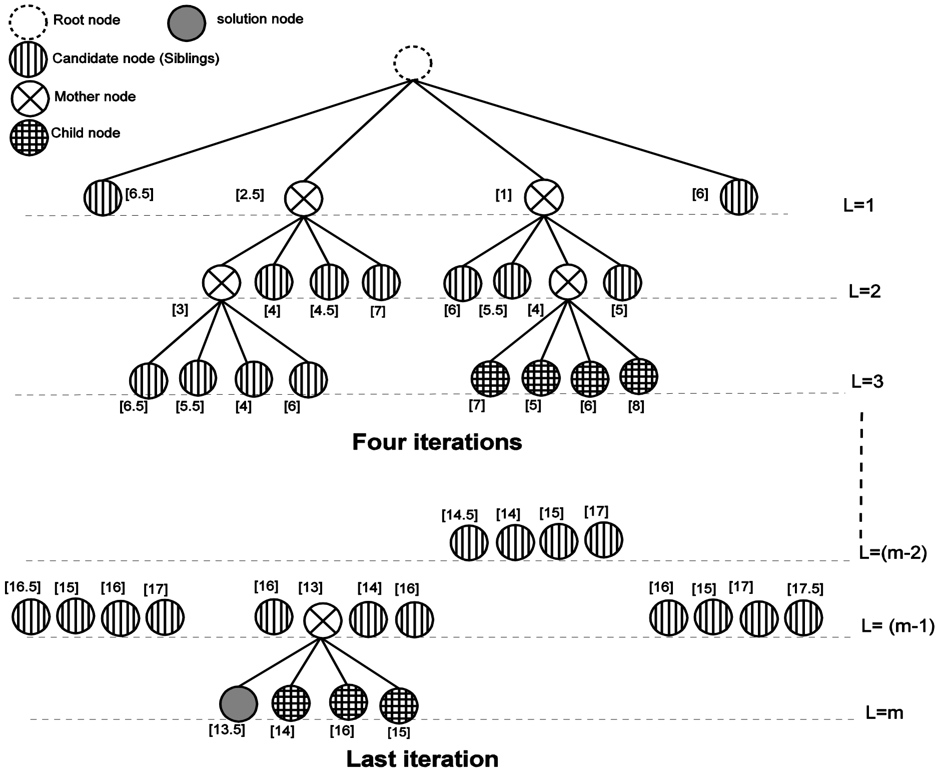


Figure 2. Illustration of the Dijkstra's search algorithm (DS) for MIMO signal detection based on tree search. The number between brackets represents the path metric $P(s_k^l)$ of the node

The minimization problem represented in Eq.(5) could be solved by a tree search with m layer(s) as depicted in figure 1. Each node in figure 1 represents a partial candidate symbol vector s_k^l which is weighted by two metrics; path metric $P(s_k^l)$ and branch metric $B(s_k^l)$. As illustrated in Figure 1, the tree search procedure starts from layer l = 0, denoted with a dotted circle, named as root node, which acts as reference to calculate branch metric at layer l = 1. Solid circled nodes and shaded-circle represent the visited nodes and the leaf nodes respectively, where an estimation of the path metric is required to make decision on a partial candidate vector or a candidate vector. In the tree search procedure, it is expected to visit the node and expand towards the child nodes if required. For conventional ML signal detection with such tree search algorithm, all the possible candidate nodes are required to be visited. Final decision on the detected symbol vector is the leaf node with the smallest path metric and is referred to herein by the ML solution. The objective of the optimization is to reduce the number of visited nodes and output the ML solution.

2.3. Dijkstra's tree search algorithm

Dijkstra's Search algorithm (DS) has been proposed initially for graph search [10]; then it has been applied on MIMO signal detection [13–15]. As shown in figure 2, the conventional Dijkstra's algorithm visits all the nodes, at first layer (l = 1), by estimating their respective path metrics; then it selects the node with the smallest path metric to be the mother node, while the rest of the nodes are the siblings. All the possible nodes expanded from the mother node at l = 2 are considered as child nodes and their respective path metrics are calculated. In the next iteration of the search procedure, all the candidate nodes, consisting of the siblings nodes and the new child nodes, are considered in the search of the new mother node. The new mother node is then expanded to its child nodes. This procedure continues until the selection of the mother node at last layer l = m which constitutes the detectable symbol of interest. The DS algorithm was also proposed for systems with memory constraint [13]. In these systems, the size of the candidate nodes list is explicitly assigned and it is equal to the maximum memory available to the system. The nodes with an index superior to the maximum list size are discarded.

Algorithm I: DS(L)

A0. (Initialization) Given an initial node list C that contains the root node only, and node list size constraint L.
A1. (Selecting the best node) Select the first node (the best node in this iteration) from C. If this node is in layer 1, stop the algorithm and output this node as the solution
A2. (Expanding the best node) Expand the best node by adding all its children nodes to C and removing itself from C.
A3. (Maintaining and sorting the node list) Order the nodes in C in ascending order of their path metric. Retain the first min(|C|, L) nodes and discard others. Go to A1

3. PROPOSED ADAPTIVE DIJKSTRA'S ALGORITHM (ADS)

The computational complexity of the conventional DS algorithm is exponentially proportional to the number of nodes required to visit in the detection process. In this paper, the authors propose a novel candidate search which requires a lower number of nodes to be visited; subsequently, expected to have reduced computational complexity. The proposed algorithm is based on an adaptation criterion to select the optimum nodes to be visited and is referred to herein as Adaptive Dijkstra’s Search algorithm (ADS).

For each iteration, the ADS performs three new steps which are:

3.1. Step 1 (Defining the number of child nodes)

The degree of closeness between the path metric of the mother node and the siblings is used as a selection criterion to define the number of child nodes to be visited and it is evaluated by exploiting two estimated parameters: γ and μ calculated by Eq.(7) and Eq.(8):

$$\gamma = \frac{\mathrm{P_i(3)\text{-}P_i(1)}}{2} \tag{7}$$

$$\mu = \mathrm{P_i(2)\text{-}P_i(1)} \tag{8}$$

In the case of $\mu \geq \gamma$, the mother node is too small compared to the second smallest node in the candidate list (first sibling). It means that the current mother node and its child nodes are the most promising nodes while the siblings are less likely to lead to the ML solution. From this point of view, all the child nodes have to be visited. Otherwise ($\mu < \gamma$), the mother node is comparable to the first sibling. In this case, each sibling is also expected to be viewed as a promising node to lead to the ML solution. In order to enhance the possibility for the siblings to be expanded, the ADS is designed to reduce the number of child nodes which correspond to the selected mother. The number of child nodes w to be visited is defined by Eq. (9):

$$w = \begin{cases} |\Omega| \, if \;\; \mu \geq \gamma \\ 0.25 \times c \times |\Omega| \, if \;\; \mu < \gamma \end{cases} \tag{9}$$

Where: $c = \{1, 2, 3\}$ is a tunable parameter used to predefine the number of child nodes to be visited in the ADS search procedure.

3.2. Step 2 (Ordering the child nodes)

If the number of child nodes required to be visited is less than $|\Omega|$, only the best child nodes will be concerned. A Symbol estimator is performed using the received signal vector, the channel state information and the just detected signal to obtain an estimated symbol $\hat{s}_k$ expressed in (10). The output of the estimator for the respective mother node leads towards sorting the child nodes in the order of the nearest first as presented in figure 3.

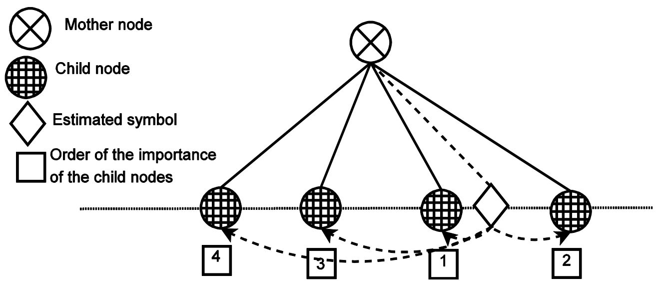


Figure 3. Ordering the child nodes based on the estimated symbol exp: 16-QAM

Figure 4 presents an illustration of the Adaptive Dijkstra's search algorithm (ADS) for MIMO signal detection based on tree search. The number between brackets represents the path metric $P(s_k^l)$ of the node.

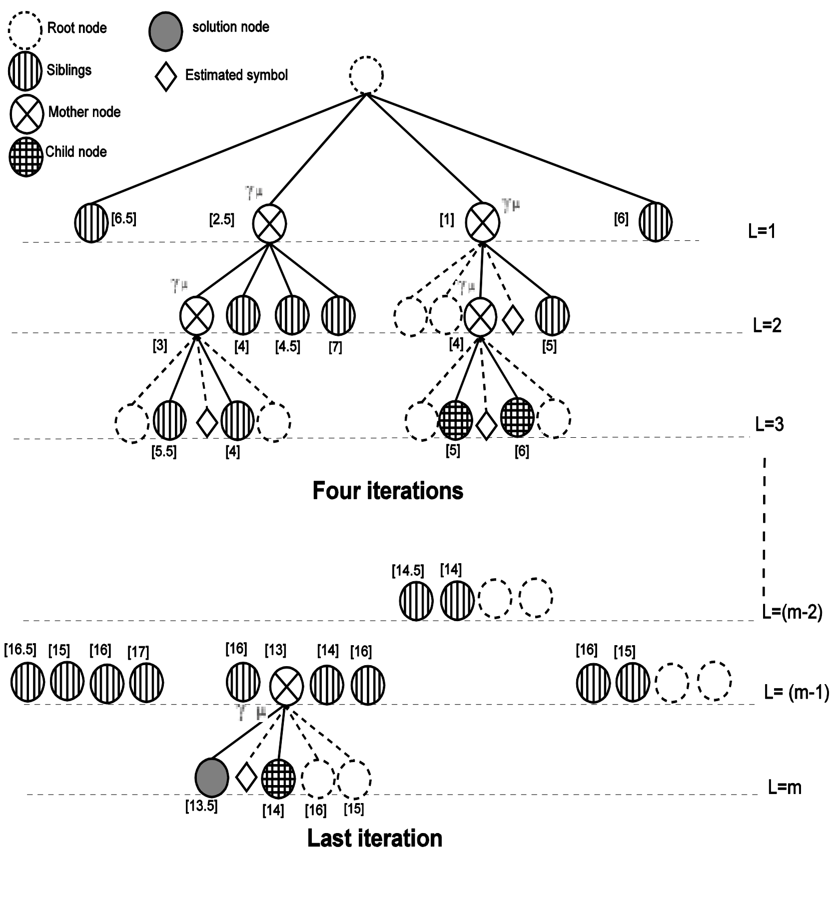


Figure 4. Illustration of the Adaptive Dijkstra's search algorithm (ADS) for MIMO signal detection based on tree search. The number between brackets represents the path metric $P(s_k^l)$ of the node

$$\hat{s}_k = \begin{cases} y_k / r_{k,k} \\ \left(y_k - \sum_{i=k+1}^{m} r_{k,i} s_i \right) \Big/ r_{k,k} \end{cases} \quad (10)$$

To avoid the additional computations of the sorting, a predefined look-up table (LUT), based on the output of the estim.

3.3. Step 3 (Visiting the child nodes)

From the ordered child nodes, the proposed ADS algorithm visits only the w best nodes. Figure 4 depicts the tree expansion of the ADS algorithm. In the first iteration, the number of visited child nodes is reduced to two, while it is maintained to four in the second iteration. the search continues by defining the number of child nodes in each iteration until reaching the last layer. The proposed ADS is based on an adaptive search on each expanded node. It is more suitable for transmitted signals with high modulation order and with a given SNR, which result in an increased number of expanding nodes. Unlike the conventional DS, which visits all the child nodes, the ADS visits adaptively a reduced number of child nodes. Subsequently, the ADS algorithm is expected to outperform the conventional DS algorithm in terms of computation complexity especially for systems with high modulation order. The reduction of computational complexity due to the elimination of one child node is equal to the additional computational requirement due to the mother node expansion. In the ADS algorithm, many more child nodes are more likely to be discarded as regulated by the proposed tunable parameter. Hence the reduction in computational complexity is expected to be significant compared to the conventional DS algorithm. These statements will be further demonstrated by simulation results, presented in the following sections.

Algorithm II :ADS(L)

A0–A1. Same as Algorithm I.

A2. (Defining the number of child nodes) calculate γ and μ by Eq.(7) and Eq.(8).
If $\mu \geq \gamma$ then all the child nodes have to be visited
Conversely, The number of child nodes w to be visited is defined by Eq. (9).

A3. (Ordering the child nodes) an estimated symbol $\hat{s}_k$ expressed in (10) leads towards sorting the child nodes in the order of the nearest first

A4. (Expanding the best node) Expand the best node by adding its w children nodes to C and removing itself from C.

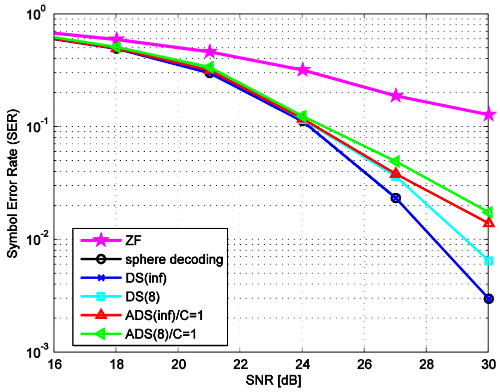
A5. (Maintaining and sorting the node list) Order the nodes in C in ascending order of their path metric. Retain the first min(|C|, L) nodes and discard others. Go to A1.

Table 1 presents parameters of simulated MIMO systems.

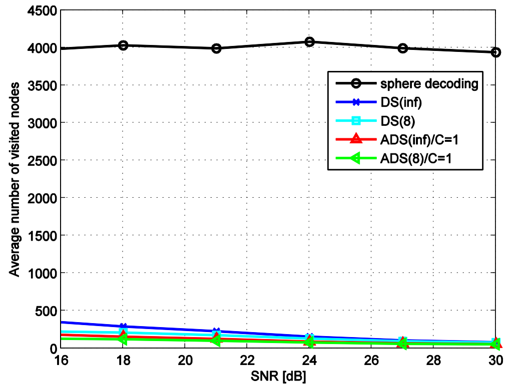
Table 1. Simulated MIMO systems

Parameters	Values
Number of transmit antennas	4
Number of receive antennas	4
Modulation order (M-QAM)	64
Tunable Parameter (c)	1 (25%), 2 (50%), 3 (75%)
Channel	Rayleigh

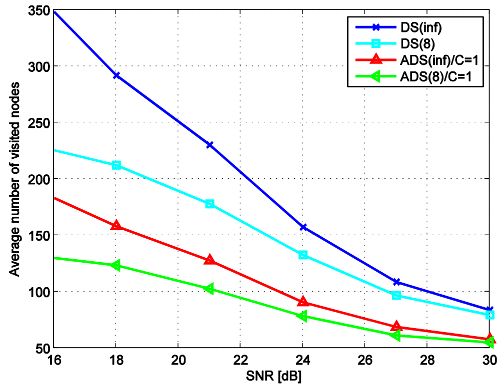
Figure 5 shows performance comparison of the ADS algorithm with c=1 in 4 x 4 MIMO system and 4-QAM modulation.



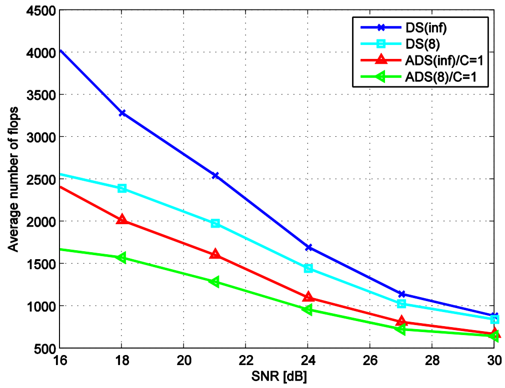
(a) SER performance



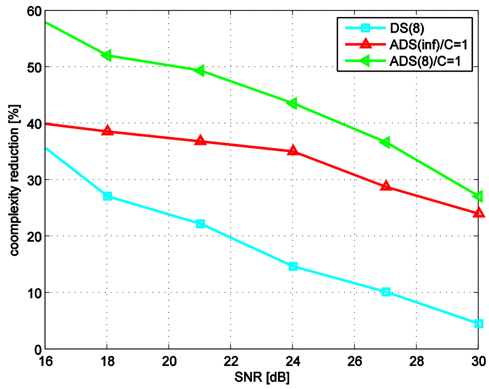
(b) Computation complexity
(number of visited nodes)



(c) Computation complexity
(number of visited nodes)



(d) Computation complexity
(number of flops)



(e) Reduction of computation complexity

Figure 5. Performance comparison of the ADS algorithm with c=1 in 4 x 4 MIMO system and 4-QAM modulation.

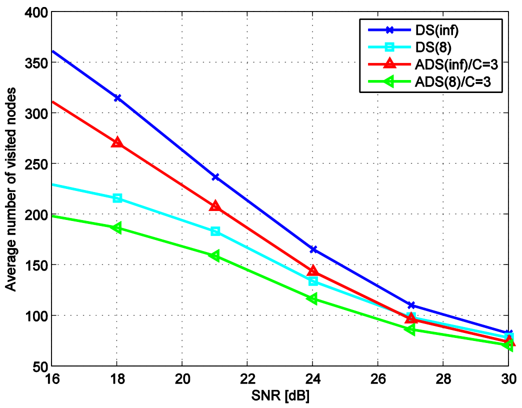
4. SIMULATION RESULTS AND DISCUSSIONS

4.1. Experiment design

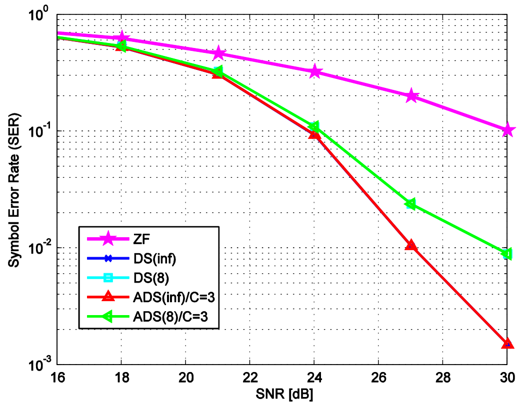
The aim of this section is to experimentally compare the efficiency of the proposed ADS algorithm over the conventional DS algorithm in MIMO signal detection, in terms of Symbol Error Rate (SER) performance (achieving the optimal solution for signal detection) and the associated computational complexity. MATLAB simulation environment is used to realize the simulations. Table I presents the different parameters settings for the experiments. The SER is employed to compare the performance of the systems with different signal detection schemes, while the average number of visited nodes and flops are calculated to evaluate the computational complexity. In this experiment, a 4x 4 MIMO system with64-QAM modulation order is considered. Four detection algorithms were tested on this system, ZF, SD, DS and ADS. As for the memory, two scenarios were considered for both DS and ADS algorithms (U = 1 and U= 8).

4.2. Results and discussion

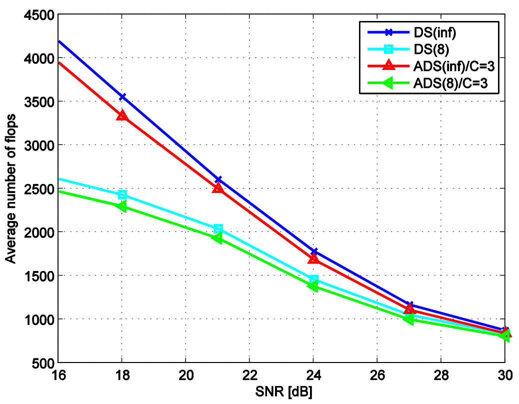
As shown in Figure 5, the SD algorithm was employed in order to verify that the conventional DS has an optimal error performance It is interesting to notice in Figure 5-a that for SNR <24 dB, the performance of the conventional DS and the proposed ADS with constrained (U= 8) and unconstrained memory (U= 1) is almost the same, with negligible differences. It is only when SNR>24 dB where performances of different algorithms start to differentiate but gradually and only slightly following this sequence from best to worst: DS (1), DS(8), ADS(1) and ADS(8). For SER= 10^{-2}, the performance of ADS(1) is degraded by about 3dB compared to the DS(1) while the performance of ADS(8) is degraded by about 2:5 dB compared to DS(8).



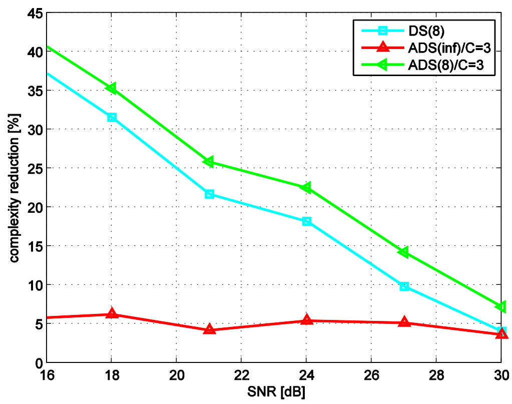
(a) SER performance



(b) Computation complexity
(number of visitednodes)



(c) Computation complexity
(number of flops)



(d) Reduction of computation complexity

Figure 6. Performance comparison of the ADS algorithm with c=1 in 4 x 4 MIMO system and 4-QAM modulation.

By referring to the computational complexity of the conventional DS(1), in Figure 5-e, the proposed ADS(1) reduces the complexity by 25% to 40% While the ADS(8) reduces the computational complexity by 20% compared to DS(8). The optimal curves of the SER and computational complexity are presented in contrast to the curves of the SER and number of flops of the conventional DS(1) respectively.

For low and medium SNR:

1) The noise affects the transmitted symbols, which appear in the tree as nodes equally probable to lead to the optimal solution and their path metrics are relatively close. Therefore, a large number of nodes have to be visited as presented in Figure 5c. Due to the equally searching of the DS(1) algorithm, all the child nodes are visited, even the weakest nodes (350 nodes are visited for SNR= 16dB).

2) On the contrary, the ADS(1) visits the strongest child nodes (25% in this experiment) and discards a large number of the weakest child nodes (75%). As a result, the total visited nodes is 180 for SNR= 16dB.

3) Given the large number of the visited nodes, the probability associated to each node to lead to the optimal solution has to be small, which results in a negligible SER degradation for discarding the weakest child nodes.

4) On the other hand, this pruning reduces the computational cost (about 40% compared to DS(1)) by conserving the calculations of the path metrics of the discarded child nodes.

For memory constraint, the ADS inherits also the behavior of the conventional DS. The ADS(8) has the same SER performance as DS(8) while reducing the computations by 21%.

For high SNR:

1) The degraded performance of the ADS(1) compared to the DS(1) is caused by discarding the child nodes. Knowing that the mother node in this region of SNR is the most likely to end to the ML solution, reducing the number of its child nodes enhances the probability to discard the optimal solution. As a result a high degradation of SER performance is observed where 75% of child nodes are discarded.

It is important to note that in cases when SNR is less than 24 dB, visiting only 25% of child nodes, ADS (1) with c= 1, seems to be sufficient to outperform the DS (1) with quasi-optimal performance as well as a significant reduction of computation complexity by up to 40%.

In Figure 6, the SER performance and the computation complexity are calculated for the ADS algorithm with c= 3 (75% of child nodes are visited).

For c= 3 , both the performances of ADS(1) and ADS(8) are the same as the performances of DS(1) and DS(8), respectively, across all SNR levels. The complexity reduction is about 5% for both ADS(1) and ADS(8) compared to DS(1) and DS(8), respectively. Visiting 75% of the best child nodes, conserves the optimality of the DS(1) algorithm and reduces its complexity by 5%.

CONCLUSION

Due to the growing interest in large-scale MIMO system for future generation of wireless communication system, reduction of the computational complexity within ML detection scheme has been addressed in this paper. A tunable adaptation parameter has been introduced, while attaining intended application specific quality of service (SER) as a tradeoff with computational complexity. More specifically, a novel tree search procedure has been proposed. The simulation results have been fairly consistent with the theoretical expectations to demonstrate the potential superiority of the proposed ADS over the conventional DS. For real world applications, the MIMO systems operate in low to medium SNR; where the proposed ADS proved to outperform the conventional DS. In this case, the results show that searching only 25% of the child nodes is sufficient to achieve an acceptable quasi-optimal solution, resulting in a computational complexity reduction of up to 40%. For high SNR regions, such reduction in computational complexity is attained at anominal cost of additional SNR. However, in the case of 75% of the tree search, the ADS outperforms the conventional DS across all SNR levels, still with some reduction in computational complexity. Adaptation of the dynamic modulation resolution as well as dynamic selectivity of active antennas within a large scale MIMO system to be potential focus to attain an intended QoS consistently with the proposed detection scheme.

REFERENCES

[1] A. Paulraj, D. Gore, R. Nabar, and H. Bolcskei, 2004. An overview of mimo communications - a key to gigabit wireless, *Proceedings of the IEEE,* vol. 92 (2), 198–218.

[2] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson., 2013.Scaling up MIMO: Opportunities and challenges with very large arrays, *IEEE Signal Processing Magazine*, Vol. 30 (1), 40–60.

[3] S. Yang and L. Hanzo, 2015. Fifty years of MIMO detection: The road to large-scale MIMOs, *IEEE Communications Surveys and Tutorials*, Vol. 17(4), 1941–1988.

[4] F. Jiang, C. Li and Z. Gong, 2018. Low Complexity and Fast Processing Algorithms for V2I Massive MIMO Uplink Detection, *IEEE Transactions on Vehicular Technology* .1–15.

[5] J. Zeng, J. Lin and Z. Wang, 2018. Low Complexity Message Passing Detection Algorithms for Large-Scale MIMO Systems, *IEEE Wireless Communications Letters*. 1–4.

[6] S. Wu, L. Kuang, Z. Ni, et al, 2014. Low-Complexity Iterative Detection for Large-Scale Multiuser MIMO-OFDM Systems Using Approximate Message Passing, *IEEE Journal of Selected Topics in Signal Processing*, vol. 8 (5), 902–915.

[7] X. Zhu and R. Murch, 2002. Performance analysis of maximum likelihood detection in a mimo antenna system, *Communications, IEEE Transactions on*, vol. 50 (2), 187–191.

[8] M.-O. Damen, H. El-Gamal, and G. Caire, 2003. On maximum likelihood detection and the search for the closest lattice point, *Information Theory, IEEE Transactions on*, vol. 49 (10), 2389–2402.

[9] E. Larsson, 2009. Mimo detection methods: How they work [lecture notes], *Signal Processing Magazine, IEEE*, vol. 26 (3), 91–95.

[10] E. W. Dijkstra, 1959. A note on two problems in connexion with graphs, *Numerische mathematik*, vol. 1 (1), 269–271.

[11] K. Su, 2005. Efficient maximum likelihood detection for communication over multiple input multiple output channels, Department of Engineering, University of Cambridge, United Kingdom, 44p.

[12] T. Fukatani, R. Matsumoto, and T. Uyematsu, 2004. Two methods for decreasing the computational complexity of the mimo ml decoder, *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 87 (10), 2571–2576.

[13] A. Okawado, R. Matsumoto, and T. Uyematsu, 2008. Near ml detection using dijkstra's algorithm with bounded list size over mimo channels in Information Theory. *ISIT 2008. IEEE International Symposium on*, 2022–2025.

[14] R. Chang and W.-H. Chung, 2012. Best-first tree search with probabilistic node ordering for mimo detection: Generalization and performance-complexity tradeoff, *Wireless Communications, IEEE Transactions on*, vol. 11 (2) , 780–789.

[15] T.-H. Kim and I.-C. Park, 2010. High-throughput and area-efficient mimo symbol detection based on modified dijkstra's search, *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 57 (7), 1756–1766.

[16] W. Hou, T. Fujino, and T. Kojima, 2012. Adaptive candidate selection scheme in qrm-mld algorithm for mimo detection, *in Wireless and Mobile Computing, Networking and Communications (WiMob), 2012 IEEE 8th International Conference on*, 502–506.