On Further Reduction of Complexity in Tree Pruning based Sphere Search

Byonghyo Shim, *Member, IEEE,* and Insung Kang, *Member, IEEE,*

***Abstract*—In this letter, we propose an extension of the probabilistic tree pruning sphere decoding (PTP-SD) algorithm that provides further improvement of the computational complexity with minimal extra cost and negligible performance penalty. In contrast to the PTP-SD that considers the tightening of necessary conditions in the sphere search using per-layer radius adjustment, the proposed method focuses on the sphere radius control strategy when a candidate lattice point is found. For this purpose, the dynamic radius update strategy depending on the lattice point found as well as the lattice independent radius selection scheme are jointly exploited. As a result, while maintaining the effectiveness of the PTP-SD, further reduction of the computational complexity, in particular for high SNR regime, can be achieved. From simulations in multiple-input and multiple-output (MIMO) channels, it is shown that the proposed method provides a considerable improvement in complexity with near-ML performance.**

***Index Terms*—Sphere decoding, multiple input multiple output, maximum likelihood decoding, sphere radius, probabilistic tree pruning.**

I. Introduction

Maximum likelihood (ML) detection of the sequence of finite alphabet symbols requires a search for the entire block of symbols. Although the ML solution is optimal for achieving the minimum probability of error, for a general matrix where no exploitable structure is available, this problem was shown to be NP-hard [1], [2]. Ever since the rediscovery of Fincke and Pohst's work [3], [4], an efficient algorithm so called *Sphere Decoding* (SD) has received much attention [1], [5]. Even though the SD algorithm offers computational efficiency in many communication scenarios [2], [6]–[8], in particular for multiple-input multiple-output (MIMO) antenna systems, still a considerable amount of computations is required, which limits the application in the real-time system [6], [15]. Since the heart of the SD algorithm lies on the choice of sphere radius within which the search space is limited, several approaches have been proposed to provide the efficient control of sphere radius including *increase radius search* (IRS) [6], *improved increasing radius search* (IIRS) [11], and *increasing radii algorithm* (IRA) [10]. It has been shown that these algorithms achieve an improvement in complexity over the SD algorithm at the expense of negligible performance loss.

This work was supported by a research grant from Qualcomm Inc. and KOSEF R01-2008-000-20292-0. This paper was presented in part at the ICASSP, Taipei, Taiwan, April 2009.

B. Shim is with School of Information and Communication, Korea University, Seoul, Korea (email: bshim@korea.ac.kr).

I. Kang is with Qualcomm Inc., CA 92121 USA (email: insungk@qualcomm.com).

In [12], we have proposed an algorithm relaxing the strict ML search to gain the benefit in computational complexity referred to as *probabilistic tree pruning SD* (PTP-SD). We have shown that an addition of the probabilistic noise constraint into the path metric, generated by the probabilistic model of the unvisited nodes, expedites the tree pruning. Since the sphere constraint is loose for most layers of the search tree, the addition of the estimated noise contribution tightens the necessary condition, and therefore facilitates the pruning of unlikely segments of the tree. Due to the elimination of subtrees without visiting, we could achieve considerable computational savings at the expense of negligible performance degradation.

In this letter, we propose an extension of the PTP-SD that provides further improvement of the computational complexity with minimal extra cost. While our previous approach tries to fortify the structural weakness of the sphere search by tightening the sphere radius per layer, proposed method focuses on the sphere radius control strategy when a candidate lattice point is found. In this respect, we can view the PTP-SD as *intra-search radius control* and the proposed method as *inter-search radius control* (ISRC). Notice that although the SD dynamically updates the radius whenever a new candidate is found and hence shrinks the volume of hypersphere, it does not guarantee the fast ML search. One example might be the case where lattice points are densely spaced in their cost function (e.g., low SNR scenario). Even when lattice points are spaced apart, if the sphere radius of the initial lattice point found is far larger than that of the ML point, the number of lattice points examined in SD would be substantial. Hence, a mechanism providing an aggressive radius control is crucial for achieving search space reduction.

In fact, two key requirements are considered in the design of the ISRC method: 1) ML lattice point should be included in the hypersphere with high probability for minimizing the performance loss and 2) two extreme situations (too large or too small radius) should be prevented. Note that too small radius is as detrimental as too large radius since no lattice point exists inside the sphere resulting in the search failure. In order to satisfy these requirements, we employ a hybrid between the dynamic radius update that naturally depends on the lattice point found and the lattice independent radius selection scheme relying only on the noise statistics. As a result, considerable number of lattice points is excluded in the search while performance close to the ML is achieved. From the simulation over MIMO channels, we demonstrate the near-ML performance and the additional considerable complexity savings of the proposed algorithm over the PTP-SD.

The organization of this letter is as follows. In section II, we briefly review the SD and the PTP-SD. In section III, we present the proposed ISRC algorithms. The simulation results are provided in section IV and we conclude in section V.

II. Probabilistic Tree Pruning

A. SD Algorithm

Consider the ML detection of a real-valued linear system described by

$$\mathbf{r} = \mathbf{H}\mathbf{s}_t + \mathbf{v} \tag{1}$$

where $\mathbf{s}_t$ is the $m \times 1$ transmitted symbol vector whose components are elements of a finite set $\mathcal{F}$, $\mathbf{r}$ and $\mathbf{v}$ are the $n \times 1$ received signal vector and the i.i.d. Gaussian noise vector, respectively, and $\mathbf{H}$ is the $n \times m$ channel matrix ($n \geq m$). Under the assumption that $\mathbf{H}$ is given, the ML solution becomes

$$\mathbf{s}_{ML} = \arg\min_{\mathbf{s}\in\mathcal{F}^m} ||\,\mathbf{r} - \mathbf{H}\mathbf{s}\,||^2. \tag{2}$$

Instead of searching all lattice points $\mathbf{H}\mathbf{s}$, the SD algorithm tests the lattice points inside the hypersphere with radius $\sqrt{r_0}$. That is, lattice points satisfying $||\,\mathbf{r} - \mathbf{H}\mathbf{s}\,||^2 < r_0$ is only investigated. In order to make the search systematic, actual search is performed in the QR-transformed domain given by $J(\mathbf{s}) = ||\,\mathbf{y} - \mathbf{R}\mathbf{s}\,||^2 \leq d_0$ where $\mathbf{H} = [\mathbf{Q} \quad \mathbf{U}][\mathbf{R}^T \quad \mathbf{0}^T]^T$, $\mathbf{y} = \mathbf{Q}^T\mathbf{r}$, and $d_0 = r_0 - ||\mathbf{U}^T\mathbf{r}||^2$. Denoting a branch metric at layer $m-k+1$ as $B_k(s_k^m) = (\,y_k - \sum_{l=k}^m r_{k,l}s_l\,)^2$, we further have

$$J(\mathbf{s}) = \sum_{k=1}^{m} B_k(s_k^m) \leq \quad d_0. \tag{3}$$

The SD algorithm is well explained by the tree search algorithm where the candidate of the first layer (bottom row in the matrix structure) is computed initially by the comparison $B_m(s_m) \leq d_0$. Once the candidate s_m satisfying this condition is found, s_{m-1} satisfying $B_{m-1}(s_{m-1}^m) + B_m(s_m) \leq d_0$ is examined in the next layer. By repeating this step and updating the radius whenever a new lattice point $\mathbf{R}\mathbf{s}$ is found, the SD algorithm outputs the ML point $\mathbf{s}_{ML}$ for which the cost function $J(\mathbf{s})$ is minimized.

B. Probabilistic Tree Pruning (PTP)-SD

Although the SD algorithm should test the condition described in (3), due to the causality of the search, the actual condition to be checked at layer $m-k+1$ becomes

$$P_k^m(s_k^m) = B_k(s_k^m) + \cdots + B_m(s_m^m) \leq d_0 \tag{4}$$

where $P_k^m(s_k^m)$ is the path metric obtained by the accumulation of branch metrics from layers 1 to $m-k+1$. The key idea behind the probabilistic tree pruning is to use (3) instead of (4) throughout all layers in the search. Since the branch metrics $B_1, \cdots, B_{k-1}$ are unavailable at layer $m-k+1$, assuming perfect decoding, they are modeled as Gaussian noise

$$B_l(s_l^m) = (\,y_l - \sum_{j=l}^{m} r_{l,j}\,s_j\,)^2 = v_l^2 \;\; \text{for} \;\; \mathrm{l} = 1, \cdots, \mathrm{k}-1 \tag{5}$$

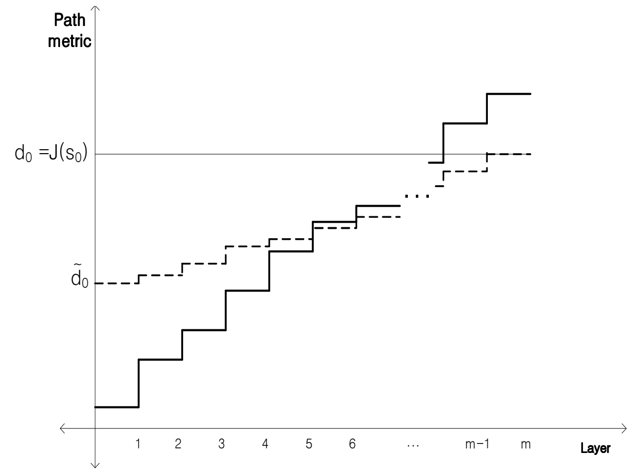


Fig. 1. Illustration of the PTP-SD principle (solid and dotted lines represent the path metric and modified sphere condition, respectively).

where v_l is the l-th component of the Gaussian noise vector $\mathbf{v}$. From (3) and (5), the new necessary condition becomes

$$\sum_{l=1}^{m} B_l(s_l^m) = P_k^m(s_k^m) + \sum_{l=1}^{k-1} v_l^2 \leq d_0. \tag{6}$$

Since $v_1, \cdots, v_{k-1}$ are values from i.i.d. Gaussian distribution, $\sum_{l=1}^{k-1} v_l^2$ becomes the χ^2-random variable with $k-1$ degrees of freedom (DOF). Denoting $\Phi_{k-1} = \sum_{l=1}^{k-1} v_l^2$, (6) becomes $P_k^m(s_k^m) + \Phi_{k-1} \leq d_0$. If the probability of this event is too small and thus less than a pre-specified threshold, we regard this event as an unlikely one and prune the subtree starting from the node. This condition can be summarized as $F_\Phi(d_0 - P_k^m(s_k^m)\,;\,k-1) = P_r(\Phi_{k-1} \leq d_0 - P_k^m(s_k^m)) < P_\epsilon$ where P_ϵ is the pre-specified pruning probability and $F_\Phi(\psi; k)$ is the cumulative distribution function (CDF) of χ^2-random variable with DOF k. By taking inverse, we get the pruning condition $d_0 - P_k^m(s_k^m) \; < \; \beta_{k-1}$ as well as the tightened necessary (survival) condition

$$P_k^m(\,s_k^m\,) \leq d_0 - \beta_{k-1} \equiv \tilde{d}_0. \tag{7}$$

where $\beta_{k-1} \; = \; F_\Phi^{-1}(P_\epsilon\,;\,k-1)$. The interpretation of (7) is that if the path metric in layer $m-k+1$ is larger than $d_0 - \beta_{k-1}$, the rest of search is unlikely to satisfy the sphere condition even for the best scenario (the remaining nodes are detected perfectly and their contributions are noises only). Hence, as depicted in Fig. 1, whenever a path s_k^m violates the modified sphere condition in (7), we stop the search and prune the path from the subtree. In [12], we reported a significant reduction of complexity of the PTP-SD over the SD in low and mid SNR regimes (roughly defined as the SNR regions such that $P_e(SNR) > 10^{-3}$ where $P_e(SNR)$ is the symbol error rate). However, the benefit of the PTP-SD vanishes as the SNR increases so that the complexity of the PTP-SD converges asymptotically to the SD complexity in the high SNR regime.

C. High SNR Scenario

As the SNR increases, the signal power is getting larger so that the distance between the cost function $\Delta(\mathbf{s}_i, \mathbf{s}_j) =$

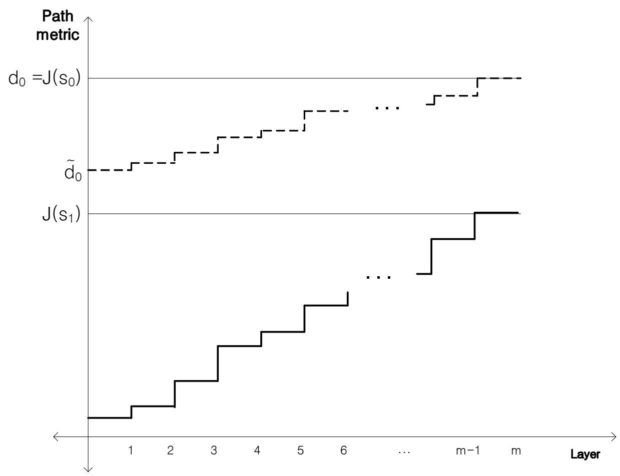


Fig. 2. Illustration of PTP-SD in high SNR scenario.

$|\, J(\mathbf{s}_i) - J(\mathbf{s}_j)\,|$ for two distinct symbol vectors $\mathbf{s}_i$ and $\mathbf{s}_j$ also becomes larger. For a candidate lattice point $\mathbf{Rs}_0$ found by the sphere search, the distance to the closest lattice point $\mathbf{Rs}_1$ with cost function smaller than $J(\mathbf{s}_0)$ can be expressed as

$$\Delta_{\min}(\mathbf{s}_0) = \min_{\mathbf{s}_1,\, J(\mathbf{s}_1) < J(\mathbf{s}_0)} \Delta(\mathbf{s}_0, \mathbf{s}_1). \tag{8}$$

Following lemma explains the shortcoming of the PTP-SD in high SNR scenario.

Lemma 1: If $\Delta_{\min}(\mathbf{s}_0) > \beta_{\max} = \max_i \beta_i$, then the pruning operation of the PTP-SD becomes invalid for every $\mathbf{s} \in B_{\mathbf{Rs}}(\mathbf{y}, J(\mathbf{s}_0))$ where $B_{\mathbf{Rs}}(\mathbf{y}, J(\mathbf{s}_0)) = \{\mathbf{s} \,|\, J(\mathbf{s}) \leq J(\mathbf{s}_0)\}$.

Proof: As discussed in II-B, the pruning operation of the PTP-SD occurs if $P_k^m(s_k^m) > J(\mathbf{s}_0) - \beta_k$. Rearranging this, we have,

$$\beta_k > J(\mathbf{s}_0) - P_k^m(s_k^m). \tag{9}$$

However, by hypothesis, we have $J(\mathbf{s}_0) - J(\mathbf{s}) \geq \Delta_{\min}(\mathbf{s}_0) > \max_i \beta_i \geq \beta_k$ for any $\mathbf{s}$ satisfying $J(\mathbf{s}) < J(\mathbf{s}_0)$. Noting further that $J(\mathbf{s}) = P_1^m(s_1^m) \geq P_k^m(s_k^m)$, we have

$$J(\mathbf{s}_0) - P_k^m(s_k^m) > J(\mathbf{s}_0) - J(\mathbf{s}) > \beta_k \tag{10}$$

for any $\mathbf{s} \in B_{\mathbf{Rs}}(\mathbf{y}, J(\mathbf{s}_0))$ and layer k, which contradicts the pruning condition in (9). ■

As illustrated in Fig. 2, if the cost function difference $J(\mathbf{s}_0) - J(\mathbf{s}_1)$ is larger than $\beta_{\max}$, $\mathbf{s}_1$ cannot be pruned by the PTP-SD operation. Therefore, in high SNR regime where the hypothesis of the lemma is satisfied, the pruning of the PTP-SD will be effective only for the lattice points in $B^c_{\mathbf{Rs}}(\mathbf{y}, d_0)$, the complement of $B_{\mathbf{Rs}}(\mathbf{y}, d_0)$. However, since these lattice points are rarely visited or they are pruned in the early layers of the search, the complexity of the PTP-SD becomes close to that of the SD.

III. Inter Search Radius Control

A. Existing Radius Control Stragies

In selecting the initial sphere radius, two well-known strategies exist:

1) Dynamic radius update using an arbitrarily large initial radius
2) Noise statistics based radius selection

First method simply uses an arbitrarily large initial radius ($d_0 = \infty$) to prevent the search failure. Typically, Schnorr-Euchner (SE) enumeration [2] is being used along with this where the candidates for each layer are sorted based on their branch metric values. Since the search starts from the candidate minimizing the branch metric, SE enumeration finds the right path earlier than the scheme exploiting the lexicographic order (Pohst enumeration) [1], [12]. The initial lattice point $\mathbf{Rs}_b$ found by the SE enumeration and corresponding cost $J(\mathbf{s}_b) = ||\,\mathbf{y} - \mathbf{Rs}_b\,||^2$ is called *Babai* point and *Babai* distance, respectively [13]. The drawback of the SE enumeration is that the Babai point might be a loose initial point. That is, when $J(\mathbf{s}_b) \gg J(\mathbf{s}_{ML})$, a lot of lattice points might be located between two and thus the number of lattice points visited is considerable.

The complementing strategy of the dynamic radius control is a lattice independent radius selection scheme proposed by Hassibi and Vikalo [6] referred to as *increase radius search* (IRS). Their assumption is that when the detection is done perfectly, branch metrics would contain the noise contribution only. Although it is an ideal scenario but provides a clue to choose the initial radius square d_0. With this assumption, $||\,\mathbf{y} - \mathbf{Rs}\,||^2 = \sum_{i=1}^n v_i^2$ becomes χ^2-random variable with n DOF. Thus, by denoting $\Phi_n = \sum_{i=1}^n v_i^2$ and setting a threshold probability P_{th} (say $P_{th} = 0.01$), a condition for the initial radius is obtained as $F_\Phi(d_0\,;\,n) = 1 - P_{th}$. Taking the inverse of χ^2-CDF, we directly have $d_0 = F_\Phi^{-1}(1 - P_{th}\,;\,n)$. Due to the fact that the radius is chosen by the noise statistics only, this approach has an advantage of skipping lots of unnecessary lattice points in the initial search. However, if the lattice points are densely packed, this method might not be effective (e.g., low SNR scenario). In addition, when the initial sphere radius square is chosen to be smaller than the ML distance ($d_0 < J(\mathbf{s}_{ML})$), sphere search fails. In this case, d_0 is re-computed with smaller P_{th} and the search should be restarted so that an additional loop is needed for the implementation.

B. Inter-Search Radius Control (ISRC)

We can marry the dynamic radius update and the noise statistics based radius selection in a way to take advantages of both (radius tightening by the dynamic radius update and probabilistic pruning by the IRS). In this subsection, two methods employing this principle referred to as ISRC-I and ISRC-II are presented.

1) ISRC-I: The key feature of this method is to speed-up the search by choosing a smaller sphere radius than the cost function of the lattice point found. In a normal SD operation, the detection error occurs when the last candidate $\mathbf{s}_f$ which always corresponds to $\mathbf{s}_{ML}$ is not equal to the transmitted symbol vector $\mathbf{s}_t$

$$\begin{aligned} P_{err}(\mathrm{ML}) &= P(\mathbf{s}_{ML} \neq \mathbf{s}_t) \\ &= P(J(\mathbf{s}_{ML}) < J(\mathbf{s}_t)) \\ &= P(J(\mathbf{s}_{ML}) < ||\mathbf{v}||^2). \end{aligned} \tag{11}$$

However, if an aggressive radius control is introduced, the search might be finished without reaching the ML point. In this case, the detection error probability is

$$\begin{aligned} P_{err}(\text{Near-ML}) &= P_{err}(\mathbf{s}_f = \mathbf{s}_{ML}) + P_{err}(\mathbf{s}_f \neq \mathbf{s}_{ML}) \\ &= P(\mathbf{s}_f \neq \mathbf{s}_t,\, \mathbf{s}_f = \mathbf{s}_{ML}) + P(\mathbf{s}_f \neq \mathbf{s}_t,\, \mathbf{s}_f \neq \mathbf{s}_{ML}). \quad (12) \end{aligned}$$

The first term in the right of (12) equals $P_{err}(\text{ML})$. The second term causing an additional increase in the error probability can be further expressed as

$$\begin{aligned} P(\mathbf{s}_f \neq \mathbf{s}_t,\, \mathbf{s}_f \neq \mathbf{s}_{ML}) &= P(J(\mathbf{s}_{ML}) < J(\mathbf{s}_f) < \|\mathbf{v}\|^2) \\ &+ P(J(\mathbf{s}_{ML}) \leq \|\mathbf{v}\|^2 < J(\mathbf{s}_f)). \quad (13) \end{aligned}$$

Since $J(\mathbf{s}_{ML})$ and $\|\mathbf{v}\|^2$ are equal or very close for the mid and high SNR regimes, the second term in the right-side of (13) becomes a dominating factor and thus

$$\begin{aligned} &P_{err}(\text{Near-ML}) - P_{err}(\text{ML}) \\ &\quad \sim P(J(\mathbf{s}_{ML}) \leq \|\mathbf{v}\|^2 < J(\mathbf{s}_f)). \quad (14) \end{aligned}$$

In the sphere search, the event $J(\mathbf{s}_{ML}) \leq \|\mathbf{v}\|^2 < J(\mathbf{s}_f)$ occurs when the sphere radius square d_0 is set aggressively to $d_0 < J(\mathbf{s}_{ML}) \leq \|\mathbf{v}\|^2 < J(\mathbf{s}_f)$. Since our goal is to design the sphere radius reducing the complexity while maintaining the performance close to ML detection (say $P_{err}(\text{Near-ML}) - P_{err}(\text{ML})$ is within P_δ), we should have

$$P(d_0 < J(\mathbf{s}_{ML}) \leq \|\mathbf{v}\|^2 < J(\mathbf{s}_f)) \leq P_\delta. \quad (15)$$

Again it is highly likely that $J(\mathbf{s}_{ML}) = \|\mathbf{v}\|^2$ for the mid and high SNR regimes, so we approximately have

$$F_\Phi(J(\mathbf{s}_f)\,;\, n) - F_\Phi(d_0\,;\, n) \leq P_\delta \quad (16)$$

and directly we have $F_\Phi^{-1}(F_\Phi(J(\mathbf{s}_f)\,;\, n) - P_\delta\,;\, n) \leq d_0$. Hence, a natural choice of sphere radius when a lattice point $\mathbf{s}$ is found might be

$$d_0 = F_\Phi^{-1}(F_\Phi(J(\mathbf{s})\,;\, n) - P_\delta\,;\, n). \quad (17)$$

Employing (17), further shrinking of the search space can be achieved. In fact, when the lattice points are packed locally in their cost function, even with small P_δ, this method might provide a beneficial effect on complexity.

2) ISRC-II: In this strategy, we set pre-defined threshold probabilities $P_{th} = \{0, 0.05, \cdots\}$ and compute the corresponding radii

$$d_{th}^{\{i\}} = F^{-1}(1 - P_{th}^{\{i\}}),\ i = 1, 2, \cdots. \quad (18)$$

Since P_{th} is an increasing sequence, d_{th} becomes a decreasing one. Note that the first threshold probability $P_{th}^{\{1\}}$ should be zero to ensure the infinite radius ($d_{th}^{\{1\}} = F^{-1}(1 - P_{th}^{\{1\}}\,;\, n) = \infty$).

Both the pre-computed d_{th} as well as the dynamically obtained radii $J(\mathbf{s})$ are being exploited in the sphere search. Specifically, we set $d_0^{\{1\}} = d_{th}^{\{1\}} = \infty$ as an initial sphere radius square. In doing so, the search failure can be prevented and the Babai point $\mathbf{s_b}$ always becomes the first candidate [12]. Once the Babai point is found, instead of choosing $d_0^{\{2\}} = J(\mathbf{s}_b)$, updated sphere radius square is chosen as a minimum between $J(\mathbf{s}_b)$ and $d_{th}^{\{2\}}$, i.e., $d_0^{\{2\}} = \min\{J(\mathbf{s}_b),\ d_{th}^{\{2\}}\}$. In general, for the $(k-1)$-th candidate lattice point $\mathbf{s}_{k-1}$, the updated sphere radius square becomes $d_0^{\{k\}} = \min\{J(\mathbf{s}_{k-1}),\ d_{th}^{\{k\}}\}$. This search and update operation is repeated till the search fails and the last lattice point found becomes the output of the algorithm. In extreme cases, the sphere radii selected might be either the cost functions of lattice points except the first one $\{d_{th}^{\{1\}} = \infty,\ J(\mathbf{s}_b)\ ,\ J(\mathbf{s}_b)\ \cdots\}$ or the pre-specified radii only $\{d_{th}^{\{1\}}, d_{th}^{\{2\}}, \cdots d_{th}^{\{n\}}\}$. Since the output in the former case is the ML solution while that of the latter case might be not, it might be judicious to set a small value for the difference between adjacent radii $|\, d_{th}^{\{i\}} - d_{th}^{\{i+1\}}\,|$ after a few steps.

3) Choice of P_δ and P_{th}: In order to achieve the near ML performance and complexity savings, P_δ of *ISRC-I* needs to be chosen deliberately. Generally, too large P_δ results in the performance loss and too small P_δ will not be helpful for reducing the complexity. From (17), it is clear that $F_\Phi(J(\mathbf{s})\,;\, n) - P_\delta > 0$ and thus

$$P_\delta = \epsilon\, F_\Phi(J(\mathbf{s})\,;\, n) \quad (19)$$

where $0 < \epsilon < 1$. As a rule of thumb, relatively large ϵ is preferred for a few initial candidates (say $\epsilon = 0.5$) and small ϵ would be a reasonable choice for the rest of candidates. P_{th} of *ISRC-II* needs to be chosen in a similar principle. By choosing a radius difference $|\, d_{th}^{\{i\}} - d_{th}^{\{i+1\}}\,|$ relatively large for initial i, we can exclude bunch of lattice points and hence expedite the search speed. Whereas, by assigning a small value after a few steps, we can minimize the performance loss (e.g., $|\, d_{th}^{\{i\}} - d_{th}^{\{i+1\}}\,| \leq 0.01$).

In summary, proposed ISRC strategies have the following features:

1) By the addition of the probabilistic radius control on top of the dynamic adjustment, tight sphere radius minimizing performance loss can be obtained: while the probabilistic control is added on top of the dynamic control in ISRC-I, statistically computed (d_{th}) and dynamically adjusted radius square ($J(\mathbf{s})$) are competing in ISRC-II method.
2) ISRC strategy and intra search radius control (PTP-SD) can be effectively combined. The former strategy is especially useful when the lattice points are spaced apart (e.g., high SNR) and the latter one is more effective for the densely packed lattice structure (e.g., low SNR).
3) The extra complexity of the intra search radius control is at most one subtraction per layer [12] and that for the ISRC is one compare operation when a candidate lattice point is found. Since the online computation of χ^2-CDF and inverse CDF is a bit cumbersome, lookup table might be an option for computing d_0 in ISRC-I. Whereas, an offline tabulation is enough for ISRC-II.

The combination of the ISRC and the PTP-SD algorithms is summarized in Table I. Notice that the PTP-SD and the ISRC are implemented only by modifying the step 5 and 6 of the SD algorithm, respectively.

TABLE I
ISRC ALGORITHM

Input:	$\mathbf{y}',\ \mathbf{R},\ d_0=\infty,\ \beta_0,\cdots,\beta_{m-1},\ P_\delta$ (**ISRC-I**), $\{d_{th}^{\{i\}}\,;\, i=1,2,\cdots\}$ (**ISRC-II**)
Output:	$\hat{\mathbf{s}}$
Variable:	k denotes the $(m-k+1)$-th layer being examined
	v denotes the count for the candidate lattice point found
	i_k denotes the lattice point index sorted by the SE enumeration in the $(m-k+1)$-th layer
Step **1** :	Set $k=m$, $\tilde{d}_0(m)=d_0-\beta_{m-1}$, $P_{m+1}^m=0$, $v=1$.
Step **2** :	Compute $s_{k,\max}$ and $s_{k,\min}$ satisfying (4) with d_0 replaced by $\tilde{d}_0(k)$.
	$N_s=s_{k,\max}-s_{k,\min}+1$.
	Compute the branch metrics $B_k(s_k^m)=\left(y_k-\sum_{j=k}^m r_{k,j}s_j\right)^2$ for every $s_k\in[s_{k,\min},s_{k,\max}]$.
	Obtain the sorted s_{k,i_k} by the SE enumeration.
	Set $i_k=0$.
Step **3** :	$i_k=i_k+1$.
	If $i_k>N_s$, go to step 4.
	Else go to step 5.
Step **4** :	$k=k+1$.
	If $k=m+1$, output the latest s and terminate.
	Else go to step 3.
Step **5** :	Update the path metric $P_k^m(s_k^m)=P_{k+1}^m(s_{k+1}^m)+B_k(s_k^m)$ (**PTP-SD**).
	If $k=1$, go to step 6.
	Else $k=k-1$, $\tilde{d}_0(k)=d_0-\beta_{k-1}$ go to step 2.
Step **6** :	If $P_1<d_0$, save s, $v=v+1$, and update
	(if **ISRC-I**) $d_0=F_\Phi^{-1}(F_\Phi(P_1^m\,;\,n)-P_\delta\,;\,n)$
	(if **ISRC-II**) $d_0=\min\{P_1^m,\ d_{th}^{\{v\}}\}$
	Go to step 4.

IV. SIMULATIONS

A. Simulation Setup

In this section, we observe the performance and complexity of the proposed method (ISRC+PTP-SD) over the SD, PTP-SD, and linear MMSE estimator. In a simulation of the SD algorithm, we test both IRS and $d_0=\infty$ based strategies described in Section III-A. The simulation setup is based on the 16-QAM transmission over the MIMO channel with Rayleigh fading ($h_{ij}\sim CN(0,1)$) [16].

The SD, PTP-SD, and proposed algorithm in our study employ the depth-first search and the SE enumeration so that the radius is updated whenever a new lattice point $\mathbf{Rs}$ having smaller cost function than the sphere radius is found. The threshold probability of the IRS is set to $P_{th}=\{0.1,0.01,0.001,\cdots\}$ and that of the ISRC-II is follows $P_{th}=\{0,0.05,0.09,0.12,\cdots\}$. Note, although the role of P_{th} is similar, the usage is distinct since P_{th} of the proposed method increases to reduce the volume of the hypersphere while that of the IRS decreases to enlarge the volume when the search fails. As a measure for the performance and complexity, we employ the symbol error rate (SER) and the average number of nodes visited. For each SNR point, we run at least $20,000$ channel realizations.

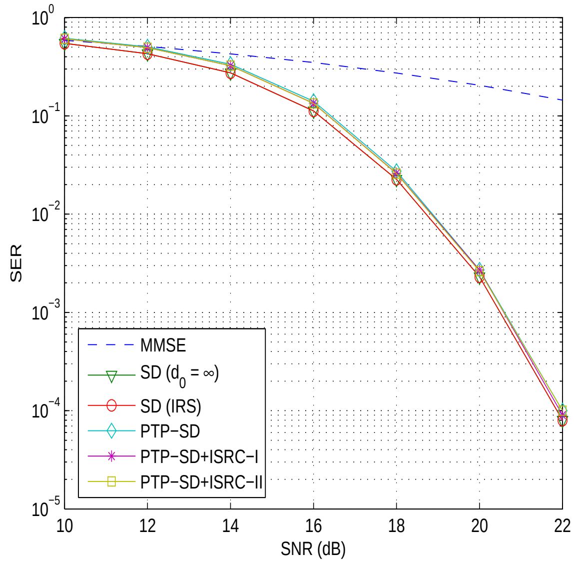
B. Simulation Results

Fig. 3(a) provides the performance results for 8×8 MIMO systems with 16-QAM modulation ($m=8$ and $|\mathcal{F}|=16$) where the worst case complexity is $|\mathcal{F}|^m=16^8\sim 4.3\times 10^9$. While the performance difference is negligible among the reference SD, PTP-SD, and proposed methods over the entire simulation range, we observe clear distinction in complexity as shown in Fig. 3(b). Although the PTP-SD achieves considerable complexity reduction in the low SNR regime as mentioned in Section II-C, its benefit disappears as the SNR increases. In contrast, the IRS-based SD shows relatively large reduction in complexity in high SNR regime so that the IRS-SD outperforms the PTP-SD in the right side of the crossover point at 14 dB. Since the proposed method (ISRC+PTP-SD) adopts the advantage of two schemes, it is no surprise that the complexity of the proposed method is better than both. In fact, the complexity reduction in the low SNR regime is mostly coming from the PTP and that in high SNR regime comes mainly from the ISRC.

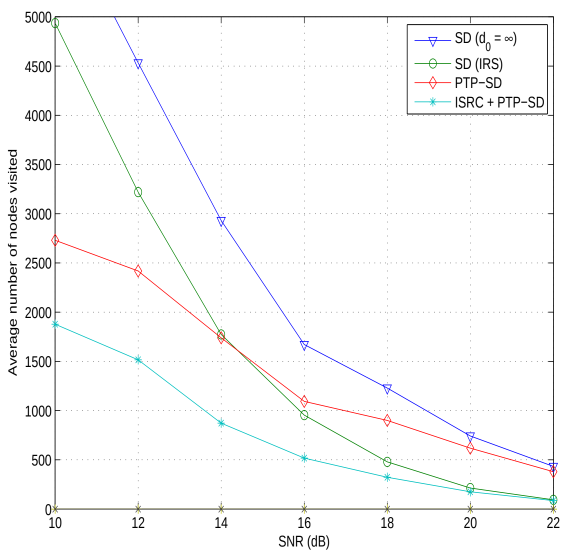
Since the computation of the sphere radius of ISRC-I requires χ^2-CDF and inverse CDF functions $d_0=F_\Phi^{-1}((1-\epsilon)F_\Phi(J(\mathbf{s})\,;\,n)\,;\,n)$, it is of interest to investigate the sensitivity when the CDF and the inverse CDF functions are quantized into finite levels. Towards this end, we check the performance of the ISRC-I with N-level quantized lookup tables. Specifically, for each $J(\mathbf{s})$ found from the sphere search, we first obtain the nearest quantized value $\tilde{J}(\mathbf{s})$ and the corresponding CDF value $F_\Phi(\tilde{J}(\mathbf{s}))$. Then, the inverse CDF value of the nearest quantized value of $(1-\epsilon)F_\Phi(\tilde{J}(\mathbf{s}))$ is used as an updated sphere radius square. As shown in Fig. 4, we observe that the performance of quantized ISRC-I is almost same as that of (infinite-level) ISRC-I expect for the $N=8$ case. This allows for an employment of ISRC-I scheme with a small lookup table without affecting performance.

V. CONCLUSIONS

The main intention of this letter is to provide a radius control strategy of the SD algorithm for providing further reduction in complexity with minimal performance loss. Contrary to



(a) SER



(b) Complexity

Fig. 3. Performance and complexity of SD algorithms for 8×8 MIMO system with 16-QAM modulation.

the PTP-SD that provides per-layer radius control by the probabilistic noise constraint, proposed ISRC methods focus on the radius control when the candidate lattice point is found. For this purpose, hybrid of the dynamic radius update and the noise statistics based radius control is jointly considered. Since the additional implementation overhead is tiny (one compare operation when the candidate lattice point is found), proposed methods find wide applications including list sphere decoding (LSD) [18], sphere encoding for multi-user MIMO system [17], or CDMA multi-user detection (MUD) with minimal code modifications.

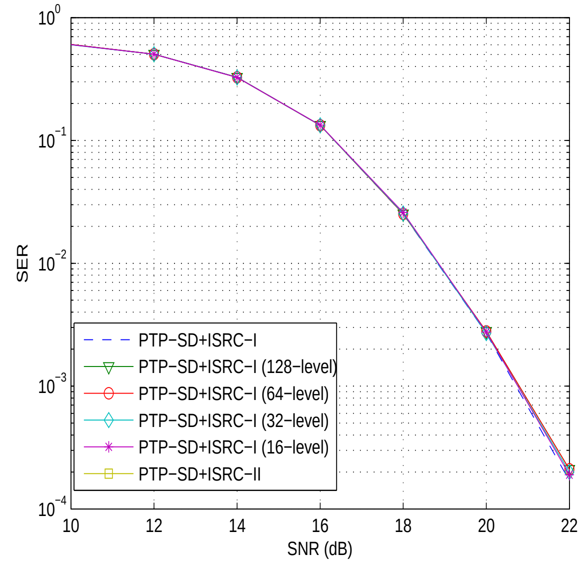


Fig. 4. Performance of ISRC-I with N-level quantization.

References

[1] E. Agrell, T. Eriksson, A. Vardy, and K. Zegar, "Closet point search in lattices," *IEEE Trans. Inf. Theory*, vol. 48, pp. 2201-2214, Aug. 2002.

[2] M. O. Damen, H. E. Gamel, and G. Caire, "On maximum-likelihood detection and the search for the closest lattice point," *IEEE Trans. Inf. Theory*, vol. 49, pp. 2389-2402, Oct. 2003.

[3] 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, pp. 463-471, 1985.

[4] M. Pohst, "On the computation of lattice vectors of minimal length, successive minima and reduced basis with applications," *ACM SIGSAM*, vol. 15, pp. 37-44, 1981.

[5] E. Viterbo and E. Giglieri, "A universal lattice decoder," in *GRESTSI 14-eme Colloque*, Juan-les-Pins, France, Sept. 1993.

[6] B. Hassibi and H. Vikalo, "On the sphere-decoding algorithm I. Expected complexity," *IEEE Trans. Signal Proc.*, vol. 53, pp. 2806-2818, Aug. 2005.

[7] D. Gesbert, M. Shafi, D. Shiu, P. Smith, and A. Naguib, "From theory to practice: an overview of MIMO space-time coded wireless systems," *IEEE J. Select. Areas Commun.*, vol. 21, pp. 281-302, April 2003.

[8] A. D. Murugan, H. E. Gamal, M. O. Damen, and G. Caire, "A unified framework for tree search decoding: rediscovering the sequential decoder," *IEEE Trans. Inf. Theory*, vol. 52, pp. 933-953, March 2006.

[9] C. P. Schnorr and M. Euchner, "Lattice basis reduction: Improved practical algorithms and solving subset sum problems," *Math. Programming*, vol. 66, pp. 181-191, 1994.

[10] R. Gowaikar and B. Hassibi, "Statistical pruning for near-maximum likelihood decoding," *IEEE Trans. Signal Proc.*, vol. 55, pp. 2661-2675, June 2007.

[11] W. Zhao and G. B. Giannakis, "Sphere decoding algorithms with improved radius search," *IEEE Trans. Commun.*, vol. 53, pp. 1104-1109, July 2005.

[12] B. Shim and I. Kang, "Sphere decoding with a probabilistic tree pruning," *IEEE Trans. Signal Proc.*, vol. 56, pp. 4867-4878, Oct. 2008.

[13] L. Babai, *On Lovasz' lattice reduction and the nearest lattice point problem*, Combinatorica, vol. 6, pp. 1-13, 1986.

[14] S. M. Ross, *Probability Models*, 2nd ed. Academic Press, 2000.

[15] J. Jalden and B. Ottersten, "On the complexity of sphere decoding in digital communication," *IEEE Trans. Signal Proc.*, vol. 53, pp. 1474-1484, April 2005.

[16] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge Univ. Press, 2005.

[17] B. Hochwald and C. B. Peel, and A. L. Swindlehurst, "A vector-perturbation technique for near capacity multiantenna multiuser communication - part II: perturbation," *IEEE Trans. Commun.*, vol. 53, pp. 537-544, March 2005.

[18] B. Hochwald and S. T. Brink, “Achieving near-capacity on a multiple-antenna channel,” *IEEE Trans. Commun.*, vol. 51, pp. 389-399, March 2003.