K-best MIMO detection based on interleaving of distributed sorting

B. Kim and I.-C. Park

A simple yet efficient technique is proposed to improve the performance of distributed sorting based *K*-best detection. The proposed technique is to interleave partially sorted elements so as to reduce the distribution skew that is the major cause of performance degradation of distributed sorting. Experimental results show that the proposed interleaving almost fully recovers the degradation and makes the performance close to that of the original *K*-best detection.

Introduction: Multiple-input multiple-output (MIMO) systems are effective in achieving high spectral efficiency, but detecting signals becomes complex because of the interference among multiple data streams. The signal detection can be transformed into a tree search problem that can be efficiently solved by two tree search strategies: depth-first and breadth-first. The depth-first searching is applied to sphere decoding and the breadth-first searching to the *K*-best detection [1], which is attractive in VLSI implementation. The *K*-best detection algorithm delivers a deterministic throughput and provides a bit error rate (BER) performance close to the maximum likelihood performance. Delivering a fixed throughput is an important advantage over the sphere-decoding algorithm. A distinct feature of the *K*-best detection algorithm is the use of sorting that is serial in nature, but the sorting becomes the major bottleneck taking a long processing time. Some efforts have been made to reduce the sorting complexity [2, 3]. Among them, distributed sorting [2], which divides the elements to be sorted into a number of groups and then sorts each group independently, is effective in reducing the sorting complexity, but it degrades the performance so severely that applying it to *K*-best detection is often infeasible.

In this Letter, we introduce the concept of distribution skew, which is the main cause of performance degradation of distributed sorting. An interleaving technique is presented to reduce the distribution skew, resulting in a significant improvement of the BER.

Distribution skew in distributed sorting: In *K*-best detection, the child nodes of K nodes are exploited to select K child nodes to be searched next. For n^2-QAM, we calculate and sort the metric values of n^*K child nodes to select the best K child nodes for the next searching, if we employ real value decomposition [4]. The processing time is dominated by sorting, which is serial in nature. To reduce the processing time, K nodes are divided into a number of groups and each group is separately processed in the distributed sorting as shown in Fig. 1*a*. Let G be the number of groups, and K be the integer multiple of G, i.e. $K = m^*G$, where m is a positive integer. Each group has a metric computation unit (MCU) and a sorting unit (SU) to select m best child nodes. The distributed sorting reduces the sorting latency and the computational time of partial distance metrics by a factor of G.

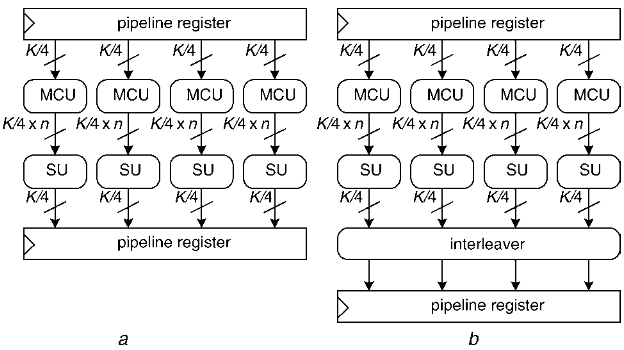

Fig. 1 *Pipeline stage of distributed sorting based K-best detection*

a Conventional structure
b Proposed structure with interleaving

As mentioned, distributed sorting has BER performance degraded, since the output nodes of the distributed sorting are not the globally best nodes. Let us consider a step of selecting K best child nodes, which is shown in Fig. 2, where $K = 8$, $n = 2$ and $G = 2$. A skewed distribution is assumed for case 1, meaning that good parent nodes associated with small distance metrics are clustered in a group. In this case, an erroneous child node of metric 6 survives instead of the node of metric 5. This erroneous behaviour would not occur if the distribution were rather even, as shown in case 2. Since a good parent node has a tendency to produce good child nodes, the distribution skew occurs frequently in conventional distributed sorting.

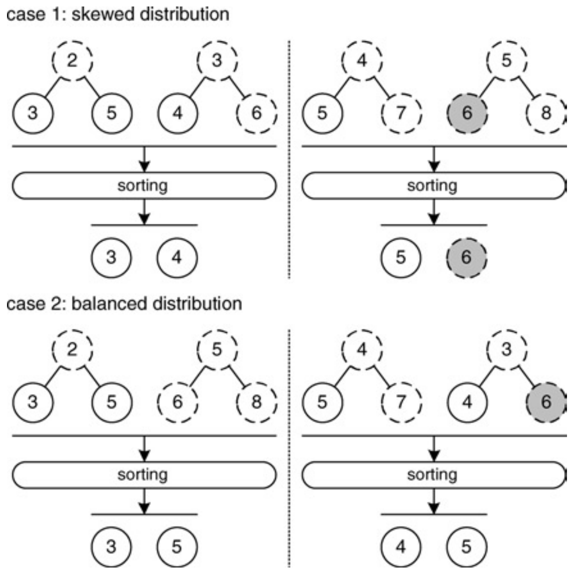

Fig. 2 *Motivation of proposed interleaving*

Distributed sorting with interleaving: Motivated by the above observation, we propose an interleaving scheme to effectively remove the skew and thus to make the pruning quality better. As shown in Fig. 1*b*, the outputs that are partially sorted in a group are interleaved across groups before they are fed into the next processing stage. In fact, there are a large number of interleaving patterns so that it is prohibitive to exhaustively investigate all the patterns. We propose a simple heuristic approach that can lead to an efficient interleaving pattern. The proposed approach is a systematic two-step procedure, as shown in Fig. 3, where $K = 16$ and $G = 4$. First, the node positions sorted in a group are rotated in the group. The rotation amount is different for each group, i.e. the wth group rotates left its nodes by w in Fig. 3. Then, in the second step, the nodes rotated in a group are shuffled across all the groups, i.e. the ith node in the wth rotated group is assigned to the wth node of the ith group. The two-step procedure is only for the sake of visualising how to construct the interleaving. Note that the interleaving is actually achieved only by changing wire interconnections in real implementations.

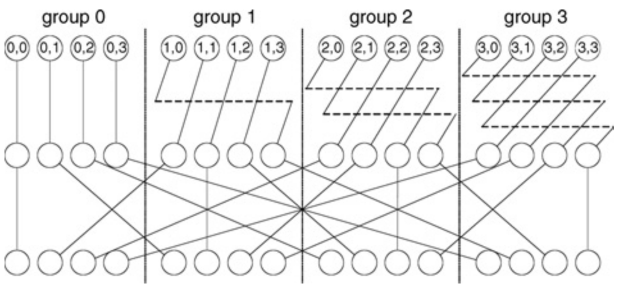

Fig. 3 *Proposed interleaving procedure*

To extend the proposed interleaving for general cases, let us first consider the common case that m is not less than G. In this case, the sorted nodes in the wth group are rotated by $w + ((G - (w^*m)\%G)\%G)$, and the ith node in the wth rotated group is assigned to the $((i + w^*m)/G)$th node in the $((i + w^*m)\%G)$th group, where % stands for the modulo operation and / represents the integer division. If m is less than G, the set of groups is divided into several partitions, each of which has at most G groups. The interleaving described above for the case $m > G$ is applied to each partition individually.

Experimental results: Fig. 4 shows the simulated BER performance for a 4 × 4 16-QAM system that conforms to the 802.11 WLAN standard.

The K-best algorithm is configured to process soft outputs and Viterbi decoding is performed to decode the 64-state convolution code specified in the standard. Parameter K is chosen to be 16 in order to achieve a BER performance close to that of the exhaustive search algorithm. For a BER of 10^{-4}, SNR degradations of about 1.1 and 2.6 dB are observed for two-way and four-way distributed cases, respectively. The degraded BER performance resulting from the conventional distributed sorting is almost fully recovered by applying the proposed interleaving. Specifically, the proposed interleaving shows about 2.5 dB SNR improvement for a BER of 10^{-4}, which is only 0.15 dB away from the original K-best detection algorithm in terms of SNR.

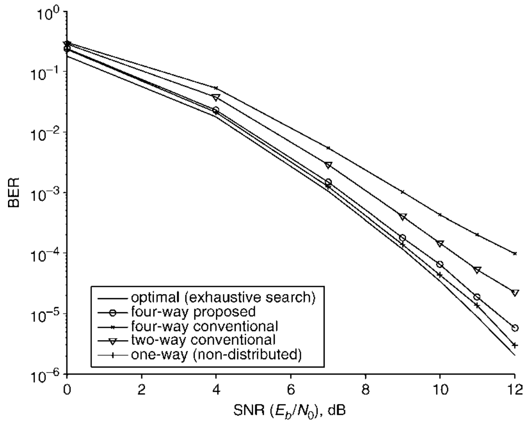

Fig. 4 *Comparison of BER performance*

The hardware overhead of the proposed interleaving is negligible, as the interleaving can be implemented only by changing the wire interconnections between the SUs and the next processing stage. A prototype detector based on the proposed interleaving is designed by employing four-way distributed sorting and by using the architecture proposed in [3] as the baseline architecture. Compared to the original K-best detection algorithm, the gate count is reduced by about 30% at the expense of slight BER degradation. In addition, the processing latency of the original K-best detection algorithm is much longer than that of the prototype detector, as the sorting time is proportional to the number of nodes to be sorted.

Conclusions: A simple interleaving technique has been proposed to reduce the distribution skew. The proposed interleaving is effective in recovering the BER degradation of the distributed sorting. Experimental results show that the proposed interleaving results in a significant performance improvement over the conventional distributed sorting, and achieves a performance similar to that of the original K-best detection.

Acknowledgment: This work was supported in part by the Institute of Information Technology Assessment through the ITRC.

21 July 2007
Electronics Letters online no: 20082129
doi: 10.1049/el:20082129

B. Kim and I.-C. Park (*Department of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST), 373-1, Guseong-dong, Yuseon-gu, Daejeon 301-701, Republic of Korea*)
E-mail: icpark@ee.kaist.ac.kr

References

1 Wong, K., Tsui, C., Cheng, R.S.K., and Mow, W.: 'A VLSI architecutre of a K-best lattice decoding algorithm for MIMO channels'. ISCAS, May 2002, Vol. 3, pp. 273–276
2 Chen, S., Zhang, T., and Goel, M.: 'Relaxed tree search MIMO signal detection algorithm design and VLSI implementation'. ISCAS, May 2006
3 Wenk, M., Zellweger, M., Burg, A., Felber, N., and Fichtner, W.: 'K-best MIMO detection VLSI architectures achieving up to 424 Mbps throughput'. ISCAS, May 2006
4 Damen, M.O., Chkeif, A., and Belfiore, J.-C.: 'Lattice code detector for space-time codes', *IEEE Commun. Lett.*, 2000, **4**, (5), pp. 161–163