The Systolic Bidirectional Algorithm for Decoding Trellis Codes

Vojin Šenk[1]
ram_senk@uns.ns.ac.yu

Predrag Radivojac[2]
predrag@ist.temple.edu

Igor Radujkov[1]
radujkov@uns.ns.ac.yu

Abstract — **The systolic bidirectional algorithm (SBA) is a new parallel procedure for decoding trellis codes. Simulation results show a substantial reduction of decoding complexity when this algorithm is compared to stack algorithm (SA) and bidirectional stack algorithm (BSA).**

I. DESCRIPTION OF THE NEW ALGORITHM

The SBA is based on the same notation used for the BSA, [1], [2]. As SBA uses $2(L+\mathcal{M}-\mathcal{T})$ processing units, or less, half of them are intended for forward search while the others, using the reverse code, perform backward search. Processing units are arranged in two arrays so that the processing unit at depth l deals only with paths of depths $l-1$, l, and $l+1$ (except for the processors at the ends of each array). Each processing unit contains a stack of size M_l (l being the depth of the unit, $l = 1, 2, \ldots, L+\mathcal{M}-\mathcal{T}$). The steps of the algorithm are:

1. Put the root node into the first stack on the F side, and the (unique) terminal node into the first stack on the opposite side, associating them the zero metric.

   The steps 2-5 are performed simultaneously in all processing units from both directions. We describe the behavior of the unit at depth l.
2. Choose the node with the best metric and eliminate it from the stack. Link it via a tunnel (if a tunnel is possible, i.e. if the states match) to each of existing paths in the stack at depth $L-l+\mathcal{M}-\mathcal{T}$. If a tunnel is M branches long, then the best path from the stack can be linked to all the paths from the stack at depth $L-l$. The total length of the paths obtained in this way is $l+\mathcal{T}+(L-l+\mathcal{M}-\mathcal{T}) = L+\mathcal{M}$ branches. Store the best combined path into the tentative decision register. If there is already a path in the register, keep the better. Prune the paths remaining in all stacks according to any of discarding criteria established. If all the stacks are emptied in this way, output the tentative decision as the decoder's final decision and terminate the algorithm.
3. Calculate the metrics of all the successors of the processed path, and eliminate all of them that do not conform to the discarding criteria established.
4. Pass remaining successors to the processing unit at depth $l+1$. Receive all the successors from depth $l-1$ and sort them according to their metrics.
5. Return to step 2.

After each new tentative decision is formed, several discarding criteria for the paths stored in all processing units can be established. The first one is based on the nonselection principle [3], while the second one is based on the path distance. Denoting by $d^{\mathrm{inv}}(L-l+\mathcal{M}-\mathcal{T})$ the minimum path metric from any stack from the reverse direction at depths from $L-l+\mathcal{M}-\mathcal{T}$ to $L+\mathcal{M}-\mathcal{T}$, a path of length l and distance d can be discarded whenever $d + d^{\mathrm{inv}}(L-l+\mathcal{M}-\mathcal{T}) \geq d_{\mathrm{TD}}$, where d_{TD} is the total accumulated distance of the tentative decision. Moreover, a path may be discarded whenever it is ranked below the M_l-th place in its current stack, or when the metric difference between the best path ever from a stack and the path in question is greater than T_l.

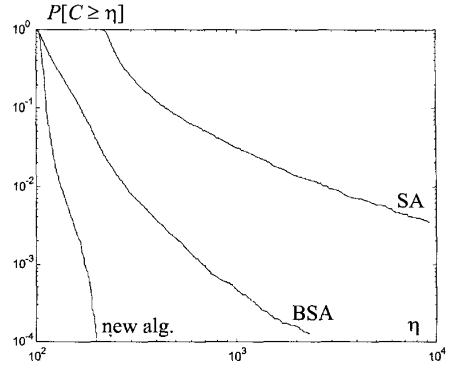

Fig. 1: Computational distribution for stack algorithm (SA or ZJ), bidirectional stack algorithm (BSA) with $\mathcal{T} = 20$ and the new algorithm at the moment of reaching the final decision

II. SIMULATION RESULTS

We have first constructed bidirectional optimum distance profile codes and then conducted a simulation, assuming Binary Symmetric Channel, $\mathcal{M} = 32$ bidirectional code with $K = 1$, $N = 2$. Generator polynomials in their usual octal form are $g_1 = 45107770451$ and $g_2 = 60235556203$. The size of the information block is $L = 180$. The crossover probability for BSC is set to $p = 0.045$, or $E_b/N_0 = 4.6$dB, meaning that the code rate equals the cutoff rate of the channel.

Figure 1 shows the computational distributions obtained for the SA, BSA, and the new algorithm (the latter two used $\mathcal{T} = 20$). In all cases we allowed unlimited stack size, i.e. we took $M_l \to \infty$, as well as $T_l \to \infty$, $\forall l$. C represents the number of parallel steps of the algorithm. In each step every processing unit performs two metric computations.

Analyzing the results obtained, we observe that the new algorithm is not only dramatically faster than the SA and BSA by Kallel and Li [2], but also than the tunneled version introduced in [1].

REFERENCES

[1] V. Šenk, P. Radivojac, "The Bidirectional Stack Algorithm," *Proceedings of IEEE International Symposium on Information Theory*, ISIT'97, p. 500, Ulm, Germany, July 1997.

[2] S. Kallel, K. Li, "Bidirectional Sequential Decoding," *IEEE Transactions on Information Theory*, vol. IT-43, No. 4, pp. 1319-1326, July 1997.

[3] J. L. Massey, "Error Bounds for Tree Codes, Trellis Codes and Convolutional Codes with Encoding and Decoding Procedures," CISM Courses and Lectures No. 216, *Coding and Complexity*, 1975.

---

[1] University of Novi Sad, Trg Dositeja Obradovića 6, 21000 Novi Sad, Serbia and Montenegro

[2] Temple University, Center for Information Science and Technology, PA 19122 Philadelphia, USA