

sion feedback equaliser, when the exponential weighting factor $\lambda = 0.99$ and the noise variance is 0.001, is shown in Fig. 2. The average number of iterations for no error reception was also determined for the linear and decision feedback equalisers with two different values of λ , and the results are presented in Table 1. The results in Fig. 2 and Table 1 are ensemble averages of 200 independent runs.

Table 1: Average number of iterations for no error reception

	Linear equaliser	DFE
$\lambda = 0.99$	73.25	20.23
$\lambda = 0.97$	71.63	20.07

Conclusions: A new decision feedback equaliser based on SPMLS has been proposed. The equaliser completely orthogonalises the two-channel input vector so that only scalar operations are required. Thus, the equaliser becomes simpler, modular and suitable for VLSI implementations.

© IEE 2001

Electronics Letters Online No: 20010602
DOI: 10.1049/el:20010602

26 April 2001

M.T. Özden and A.H. Kayran (Faculty of Electrical and Electronics Engineering, Istanbul Technical University, Maslak, 80626, Istanbul, Turkey)

References

- 1 LING, F., and PROAKIS, J.G.: 'Adaptive lattice decision-feedback equalizers - their performance and application to time-variant multipath channels', *IEEE Trans. Commun.*, 1985, COM-33, (4), pp. 348-356
- 2 ÖZDEN, M.T., KAYRAN, A.H., and PANAYIRCI, E.: 'Adaptive Volterra channel equalization with lattice orthogonalisation', *IEE Proc. Commun.*, 1998, 145, (2), pp. 109-115
- 3 LING, F., and PROAKIS, J.G.: 'A generalized multichannel least squares lattice algorithm based on sequential processing stages', *IEEE Trans.*, 1984, ASSP-32, (2), pp. 381-389

Path metric memory management for minimising interconnections in Viterbi decoders

S.-Y. Kim, H. Kim and I.-C. Park

To simplify the interconnection between processing elements and path metric memory banks in Viterbi decoders, a new path metric update scheme is proposed based on two techniques, named swapped state grouping and swapped computing. The proposed scheme leads to a simple interconnection consisting of 2×2 switches.

Introduction: Among many channel codes used to cope with error occurrence in channels, convolution codes that make code symbols serially with considering a fixed number of previous bits are often preferred in wireless digital communications for their superior coding gains. For decoding the convolution code, an efficient algorithm realising the maximum likelihood decoding was discovered and analysed by Viterbi [1]. To choose the globally maximum likelihood sequence, the Viterbi algorithm recursively computes the survivor path entering each state. The survivor path of a state is the sequence of symbols that is closest in distance to the received sequence of noisy symbols. The distance is called the path metric of that state. Each possible path of a state is determined by adding the branch metric entering that state to the path metric of the connecting survivor path, and then choosing the closest one as its survivor path.

Many Viterbi decoders have been proposed and implemented for various applications. To achieve high throughput, state parallel schemes that have as many add-compare-select (ACS) units as states have been proposed [2]. For moderate speed applications such as IS-95 CDMA mobile systems, sequential schemes, in

which one ACS unit processes all the states serially, have been used [3]. As mobile communications are changing to higher performance systems that extend the decode rates into the range of Mbit/s, it is necessary to design a fast Viterbi decoder using low power and area. One of the promising schemes to achieve this goal is to partition states into a number of groups and to make all the states in a group share one ACS unit [4, 5]. Although this scheme is effective in achieving high performance and low hardware cost, it suffers from complicated interconnection between ACS units and path metric memory banks. In this Letter, a new path metric update scheme is proposed to simplify the interconnection.

Path metric computation: Let us assume that the number of ACS units and the number memory banks storing the path metrics of states are both 2^n . Since two ACS units can share input path metrics, the lowest number of banks to make 2^n ACS units process simultaneously is 2^n if each memory bank provides only a read port. The problem to be solved is to partition 2^n states ($v > n$) into 2^n groups such that the path metrics of all the states in a group are stored in a memory bank, and to find a systematic path memory update scheme that leads to a simple interconnection between ACS units and memory banks.

If the least significant bit of the state corresponds to the oldest bit, the major path metric computation is given as follows:

$$pm_i = \min(pm_{2i} + bm_{2i,i}, pm_{2i+1} + bm_{2i+1,i})$$

$$pm_{i+2^{v-1}} = \min(pm_{2i} + bm_{2i,i+2^{v-1}}, pm_{2i+1} + bm_{2i+1,i+2^{v-1}})$$

where pm_i and $bm_{i,j}$ denote the path metric of state i and the branch metric from state i to state j , respectively. The path metrics of two adjacent states, S_{2i} and S_{2i+1} , are involved in the new path metric calculations of only two states, S_i and $S_{i+2^{v-1}}$. Therefore the path metrics of S_{2i} and S_{2i+1} have to be stored in different banks in order to read them simultaneously, and the new path metrics for S_i and $S_{i+2^{v-1}}$ that are computed in parallel using two ACS units should be stored in different banks. These constraints must be considered in state grouping. The following grouping called swapped state grouping satisfies the above constraints.

Given 2^n banks, a v -bit state denoted by $s_{v-1}s_{v-2}\dots s_1s_0$ is assigned into a bank $(s_{v-1} \oplus s_{v-2})s_{v-3}\dots s_1s_0$ where \oplus represents XOR operation.

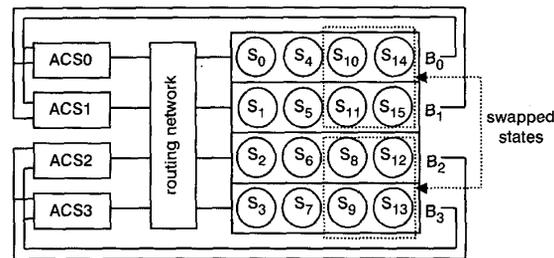


Fig. 1 Swapped state grouping for 16 state Viterbi decoder

Fig. 1 shows an example that consists of 16 states ranged from S_0 to S_{15} , four memory banks denoted from B_0 to B_3 , and four ACS units. According to the proposed state grouping, the connections from the banks to the ACS units are fixed, and the high half states are swapped to store four output path metrics calculated from four ACS units into different banks. In addition, we have to determine what states are computed in an ACS unit. The state assignment has a great influence on the routing network. The new path metrics of S_0 and S_8 are calculated from S_0 and S_1 path metrics, and the new path metrics of S_2 and S_{10} are computed from S_4 and S_5 path metrics. If ACS0 and ACS1 are assigned for (S_0, S_2) and (S_8, S_{10}) , respectively, ACS0 and ACS1 must have connections to both B_0 and B_2 . To simplify these connections, ACS0 and ACS1 can be assigned to compute (S_0, S_{10}) and (S_8, S_2) , respectively. In this case, it is sufficient to connect ACS0 to B_1 and ACS1 to B_2 . Since the high numbered state is computed in the low numbered ACS unit if $s_n = 1$ in the encoding of input states, this technique is called swapped computing. By applying the state grouping and the swapped computing, the output path metric

computed in ACS0 is stored in either B_0 or B_1 . Similarly, the other ACS units have connections to only two banks as summarised in Table 1. To control the connections, there are two 2×2 switches, each of which connects two ACS units and two memory banks as shown in Fig. 2a. All the switches are simultaneously controlled by s_{v-1} of the input state encoding. If $s_{v-1} = 1$, all the switches represent crossover connections, otherwise straight connections. Compared with the previous area-efficient architecture depicted in Fig. 2b, the proposed scheme leads to a very simple interconnection.

Table 1: State assignment and memory bank connection

ACS unit	(state, bank)			
ACS0	(S_0, B_0)	(S_{10}, B_0)	(S_5, B_1)	(S_{15}, B_1)
ACS1	(S_8, B_2)	(S_2, B_2)	(S_{13}, B_3)	(S_7, B_3)
ACS2	(S_1, B_1)	(S_{11}, B_1)	(S_4, B_0)	(S_{14}, B_0)
ACS3	(S_9, B_3)	(S_3, B_3)	(S_{12}, B_2)	(S_6, B_2)

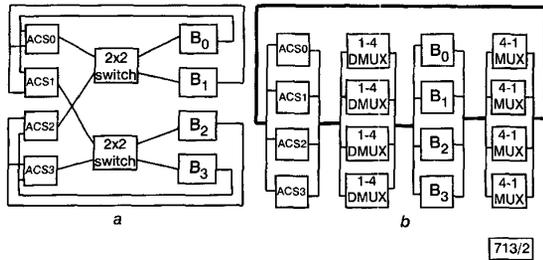


Fig. 2 Routing networks

a Proposed scheme
b Interconnection of [5]

Although the 16-state decoder is explained as an example, the above properties hold for general cases in which the numbers of states and banks are both powers of two.

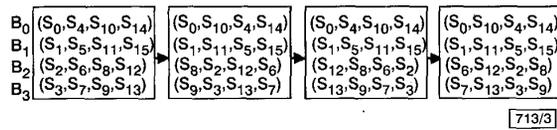


Fig. 3 In-place updating of proposed state grouping

One of the most important techniques in path metric management is the in-place updating that removes temporary buffers by storing output values into the position at which input values are read [3]. This technique can be applied without any conflict to the proposed scheme. Fig. 3 shows the change of path metric memory contents when the proposed state grouping is used along with the in-place updating.

Table 2: Gate count comparison of three Viterbi decoders

Block	State parallel	[5]	Proposed
ACS	33,280	2,940	2,940
Network	-	5,764	192
Trace-back	537	98	98
Address generator	223	564	672
etc.	15,960	500	198
Total	50,000	9,366	4,100

Design example: To validate the proposed path metric update scheme, we applied it to the design of a 256-state, 1/3-rate Viterbi decoder for CDMA2000. The decoder is composed of 16 ACS units, eight switches and 16 one-port memory banks. Due to the 16 ACS units, the decoder can process 16 states at a time, and thus 16 iterations are required to compute all the 256 states. Therefore the ACS units are processed with a clock that is 16 times faster than the data-sampling rate. To handle the 16-bit

decision vector generated from the ACS units, the survivor memory to be used for traceback operation is partitioned into 16 banks.

In Table 2, the decoder is compared to two other implementations in terms of gate counts. All the implementations are synthesised based on a 0.25 μm CMOS library. As can be seen in Table 2, the proposed architecture is the most area-efficient, especially the interconnection area is significantly reduced compared with the previous area-efficient architecture [5] that has the same performance as the proposed one. Although the state-parallel architecture can give the highest performance, its hardware complexity is more than 10 times greater than the proposed one.

Conclusions: A new path memory update scheme for Viterbi decoders has been proposed to minimise the interconnection overhead between ACS units and memory banks. To simplify the interconnection, states whose path metrics are read and written simultaneously are assigned into different banks, and path metric calculations are swapped. The proposed techniques lead to a simple interconnection that can be constructed by using only 2×2 switches, while the previous schemes require fully connected routing networks.

Acknowledgment: This work was supported in part by the Korea Science and Engineering Foundation through the MICROS Center at KAIST, Korea.

© IEE 2001
Electronics Letters Online No: 20010617
DOI: 10.1049/el:20010617

5 February 2001

S.-Y. Kim, H. Kim and I.-C. Park (Department of Electrical Engineering and Computer Science, Korea Advanced Institute of Science and Technology, 373-1, Kusung-dong, Yuseong-gu, Taejeon, 305-701, Korea)

References

- VITERBI, A.J.: 'Error bounds for convolutional codes and an asymptotically optimum decoding algorithm', *IEEE Trans. Inf. Theory*, 1967, **IT-13**, (4), pp. 263-269
- GULAK, P.G., and SHWEDYK, E.: 'VLSI structures for Viterbi receivers: Part I - general theory and applications', *IEEE J. Sel. Areas Commun.*, 1986, **4**, (1), pp. 142-154
- BIVER, M., KAESLIN, H., and TOMMASINI, C.: 'In-place updating of path metrics in Viterbi decoders', *IEEE J. Solid-State Circuits*, 1989, **24**, (8), pp. 1158-1159
- SHUNG, C.B., LIN, HORNG-DAR, CYPHER, R., SIEGEL, P.H., and THAPAR, H.K.: 'Area-efficient architectures for the Viterbi algorithm - Part I: theory', *IEEE Trans. Commun.*, 1993, **41**, (4), pp. 636-644
- SHIEH, MING-DER, SHEU, MING-HWA, WU, CHIEN-MING, and JU, WANN-SHYANG: 'Efficient management of in-place path metric update and its implementation for Viterbi decoders'. *IEEE Int. Symp. Circuits and Systems*, 1998, Vol. 4, pp. 449-452

Simple approach to nonlinear prediction

L. Vergara and P. Bernabeu

A simple procedure for improving linear prediction is presented. A zero memory nonlinear step at the output of the linear predictor may be adequate to obtain significant improvements. Two techniques are proposed for designing the nonlinearity, the object of each being to achieve the conditional mean of the nonlinear prediction given the linear one. The technique is applied to improve the signal-to-noise ratio in the automatic detection of fire by infrared signal processing.

Introduction: Prediction is a key area of signal processing and time series analysis. In statistical context, the minimum mean-squared error prediction is the conditional mean of the random variable corresponding to the predicted sample given the past samples from which prediction is to be made. Assuming Gaussianity, the conditional mean is a linear function of the samples, and we have several different standard methods for computing the predictor