detected the modem needs to recover true synchronisation. It may be possible to do this by initiating a fast retrain of the modem. In half-duplex applications (such as facsimile), it will probably be quicker for the synchronisation method to recover synchronisation itself. To do this the receiver compensation should be turned off. This is necessary because this compensation may mask the effect of the transmitter inversions. The inversions will (in general) become clearer without the receiver compensation. Similar techniques to the detection of synchronisation loss can now be used to find the inversion position. In simple cases this will be the end of the recovery process and the receiver can start to pass valid data to the user again.

However, it may be the case that a modem is capable of using several symbol rates and in this case it is desirable to synchronise to a common boundary between the blocks of each symbol rate. In the cases where two modem pairs are used either side of a digital demod/remod link this is essential for changing data rate. This means that some method of finding 'superframe' synchronisation must be devised. Superframe synchronisation may also be necessary where the shortest available synchronisation frame is much greater than the preferred inversion frequency. In this case detection of loss of synchronisation and the recovery from this loss will be quicker if a superframe is used and inversions are kept more frequent.

Recovering superframe synchronisation: To enable superframe synchronisation a unique pattern of inversions must be used across the superframe. The simplest method of doing this is to keep the same interval between inversions but occasionally miss an inversion out. The number of inversions that are missed out must be kept low to enable the loss of synchronisation detection algorithm to function properly. The more inversions that are missed out the easier it is to spot the superframe position. It has been found that retaining 75% of the inversions is sufficient to enable fast loss of synchronisation detection. The pattern of these inversions will determine how quickly and reliably superframe synchronisation can be found.

One approach would be to have a pattern of inversions in which every binary b-tuple was unique, e.g. every 6-tuple in the pattern 0 1 1 1 1 1 0 1 0 1 1 1 0 1 1 is unique. By minimising b (as it is here for the weight of the pattern) it could be possible to find the superframe position as quickly as possible. This method is not very reliable because in normal operating conditions, the noise on the channel is capable of masking the presence or absence of inversions. This means an averaging process is needed which slows down the recovery of superframe synchronisation.

Another approach is to design the pattern so that its cyclic shifts are as far apart as possible. For short patterns this is the most effective method of choosing the pattern, e.g. all the cyclic shifts of the pattern 0 1 1 1 0 1 1 1 1 1 1 1 1 0 1 0 are at least a Hamming distance of 6 from each other. The reliability of finding superframe synchronisation is increased with the minimum distance between cyclic shifts of the pattern. It is possible to monitor enough inversion positions to cover the length of the pattern. A decoding algorithm may then be employed on the inversions (along with the channel noise on them) to find the closest codeword in the codebook of cyclic shifts of the pattern. The resulting cyclic shift will give the superframe position. The reliability of this decoding obviously increases with the distance between the cyclic shifts of the pattern. For m zeros in a pattern, the largest possible minimum distance between cyclic shifts is $2(m - 1)$

The pattern above has the maximum distance possible between its cyclic shifts. The truncated pattern of any length also has the maximum possible distance between its cyclic shifts. This pattern is proposed as the inversion pattern in the forthcoming V.fast recommendation.

Conclusion: A method of maintaining synchronisation in a modern modem by inverting the redundant bit from the trellis code has been presented. This is a reliable method that does not degrade performance when the modem is working properly. The possibility of maintaining superframe synchronisation has also been discussed. It has been argued that to maintain the speed of detection of loss of synchronisation, inversions need to happen quite often. This leads to a high weight pattern of inversions. It has also been argued that this pattern should be as distinct as possible from all of its cyclic shifts to enable reliable superframe recovery. The superframe position is found by effectively decoding a weighty cyclic code. The choice of pattern is a design problem for this cyclic code. A codeword must be found of length n with weight $>w$ that will form a codebook from all its cyclic shifts with the largest possible Hamming distance. The largest Hamming distance such a codebook can have is $2(n - w - 1)$.

Acknowledgments: This work has been driven by the need to find an acceptable method of synchronisation to include in the V.fast recommendation. As such, the conclusions reached in this letter are the result of several useful discussions with participating companies. The author would particularly like to thank S. Olafsson, V. Krishnan, D. Forney and K. Jones.

18 August 1993
Electronics Letters Online No: 19931197

R. G. C. Williams (*BT Laboratories, Martlesham Heath, Ipswich, IP5 7RE, United Kingdom*)

References

1 LANG, G.R., and LONGSTAFF, F.M.: 'A Leech lattice modem', *IEEE Sel. Areas Commun.*, 1989, **SAC-7**, pp. 968–973
2 WEI, L.-F.: 'Trellis-cpded modulation with multidimensional constellations', *IEEE Trans.*, 1987, **IT-33**, pp. 483–501
3 FORNEY JR., G.D., GALLAGER, R.G., and QUERSHI, S.U.: 'Efficient modulation for band-limited channels', *IEEE Sel. Areas Commun.*, 1984, **SAC-2**, pp. 632–647
4 WILLIAMS, R.G.C., DELIOT, E., and TAIT, D.J.: 'Practical constellation shaping'. Int. Symp. on Comm. Theory and App., 1993, (Amleside)
5 WILLIAMS, R.G.C.: 'Impulse noise protection with 4-D TCM', *Electron. Lett.*, 1993, **29**, pp. 121–122

Wavelet transform in scattering data interpolation

M.-H. Yaou and W.-T. Chang

Indexing terms: Interpolation, Wavelet transforms

A fast algorithm for scattering data interpolation is presented. Based on the multiresolution wavelet transform, a preconditioning scheme is proposed to expedite the slow computation speed in the interpolation problem. By applying the wavelet transform before and after any conventional iterative solving method, fast data interpolation can be easily achieved.

Introduction: Scattering data interpolation is used to recover a full signal representation when only partial information of the signal is available. This problem plays an important role in many early vision processes such as surface from contours, structure from motion, stereopsis etc. [1]. This is an ill-posed inverse problem and is often described as a regularisation problem. In general, a variation of the functional method which involves a second order maximum smoothness requirement [2, 3] is applied for the regularisation. Various discretisation methods [2, 3] can be used to discretise the problem into an objective function of discrete nodal variables such that an approximated solution can be solved numerically. This discrete formulation then leads to the minimisation of a quadratic energy function

$$\frac{1}{2}v^T Av - v^T b + c \qquad (1)$$

where v is a column vector containing the nodal variables to be solved and A is a real symmetric matrix called the stiffness matrix. b and c are the associated column vector and constant. According to the Euler-Lagrange formula, the optimisation of this quadratic function results in a linear equation system $Av - b = 0$. For an $N \times N$ interpolation problem, the size of matrix A will be $N^2 \times N^2$. The resultant equation system is thus usually large and sparse. To solve this problem, iterative methods are usually adopted. How-

ever, owing to the sparse stiffness matrix of the equation system, the convergence rate of the iterative computation is very low. To speed up the convergence, acceleration methods such as the multigrid method and hierarchical basis functions method [3, 4] have been proposed. Basically, those methods use the multi-resolution concept in improving the convergence rate. In this Letter, a more efficient multi-resolution approach based on the wavelet transform is introduced.

Preconditioning by wavelet transform: Basically, in the discrete formulation of the interpolation problem, the signal to be interpolated is directly expressed as the linear combination of the associated nodal basis functions. That is, the interpolated signal $F(x)$ is expressed as

$$F(x) = \sum_{0}^{N-1} v[n]\phi(x-n) \tag{2}$$

where the variables in sequence $v[n]$ are the nodal variables associated with the nodal basis functions $\phi(x-n)_{(n=0,\ldots,N-1)}$. This expression in eqn. 2 is of the same form as the multi-resolution wavelet representation [5]. In the language of wavelet transform, the function $\phi(x)$ is called the scaling function and the sequence $v[n]$ is called the discrete approximation of the continuous signal $F(x)$. With such an expression, the wavelet transform [5] can be readily applied to transform eqn. 2 as

$$F(x) = \sum_{k=1}^{J}\sum_{n=0}^{\frac{N-1}{2^k}} w_k[n]2^{\frac{-k}{2}}\psi(2^{-k}x-n) + \sum_{n=0}^{\frac{N-1}{2^J}} v_J[n]2^{\frac{-J}{2}}\phi(2^{-J}x-n)_{(J=1,2,3,\ldots)}$$

$$v_0[n] = v[n] \tag{3}$$

where the function $\psi(x)$ is called the wavelet and integer J stands for the resolution level. In this multi-resolution representation of signal $F(x)$, the original nodal basis $\phi(x-n)$ is transferred to the wider bases $\{\phi(2^{-J}x-n), \psi(2^{-k}x-n)_{(k=1,\ldots,J)}\}$. The nodal variables in $v_0[n]$ are transformed to the variables in $\{v_J[n], w_k[n]_{(k=1,\ldots,J)}\}$. This transform of nodal variables is called the discrete wavelet transform. The discrete wavelet transform can be easily implemented by a QMF structure [5] as, for $m = 0, 1, 2, \ldots$,

$$v_{m+1}[n] = \sum_k h_0[k-2n]v_m[k]$$

$$w_{m+1}[n] = \sum_k h_1[k-2n]v_m[k] \tag{4}$$

$$v_m[n] = \sum_k v_{m+1}[k]g_0[n-2k] + \sum_k w_{m+1}[k]g_1[n-2k] \tag{5}$$

The analysis filters $h_0[n]$, $h_1[n]$ and the synthesis filters $g_0[n]$, $g_1[n]$ are directly determined by the functions $\phi(x)$ and $\psi(x)$. In matrix form, we can write $\boldsymbol{v} = \boldsymbol{R}\tilde{\boldsymbol{v}}$. The vector $\tilde{\boldsymbol{v}}$ contains the transferred nodal variables $\{v_J[n], w_k[n]_{(k=1,\ldots,J)}\}$. The matrix $\boldsymbol{R}$ is the QMF matrix [6] which describes the synthesis filtering in the QMF structure. With this transform, the quadratic energy function in eqn. 1 can be rewritten as $1/2(\boldsymbol{R}\tilde{\boldsymbol{v}})^T\boldsymbol{A}(\boldsymbol{R}\tilde{\boldsymbol{v}}) - (\boldsymbol{R}\tilde{\boldsymbol{v}})^T\boldsymbol{b} + c$; this new expression results in a new equation system $\tilde{\boldsymbol{A}}\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{b}}$ where $\tilde{\boldsymbol{A}} = \boldsymbol{R}^T\boldsymbol{A}\boldsymbol{R}$ and $\tilde{\boldsymbol{b}} = \boldsymbol{R}^T\boldsymbol{b}$. In this new equation system, the number of variables is the same as that of the untransformed equation system. However, the convergence condition of the new stiffness matrix $\tilde{\boldsymbol{A}}$ is much better than that of the original matrix $\boldsymbol{A}$. The reason for this improvement is the transfer of the nodal basis. The basis transfer replaces the original basis functions by the new basis functions with larger region of support. This strengthens the connection among the interpolation nodes and expedites the convergence rate in the iterative computation [6]. Hence, wavelet transformation provides an effective preconditioning of the interpolation problem.

Fast interpolation algorithm: With the wavelet transform as a preconditioner to the interpolation problem, a fast interpolation algorithm can then be designed. For a given interpolation problem, we first derive the equation system $\boldsymbol{A}\boldsymbol{v} = \boldsymbol{b}$. We then choose a suitable resolution level J and derive the QMF matrix R to transfer the equation system into $\tilde{\boldsymbol{A}}\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{b}}$. The transferred equation system can be easily solved by the general iterative solving method. By transferring the solved $\tilde{\boldsymbol{v}}$ back to $\boldsymbol{v}$ by $\boldsymbol{v} = \boldsymbol{R}\tilde{\boldsymbol{v}}$, the solution can then be obtained. In this algorithm, the transfer of nodal variables is only applied before and after the iterative computation. The iterative solving method is independent of the algorithm. Any iterative solving method can be applied. The various choices of the scaling function $\phi(x)$ and wavelet $\psi(x)$ also provide much flexibility in the design of QMF matrix $\boldsymbol{R}$.

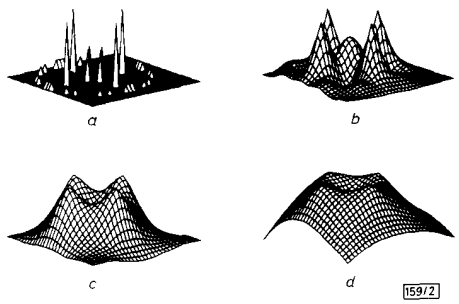

Fig. 2 *Interpolation of 32 × 32 surface*

a Constraints of interpolated surface
b Result after 50 iterations using resolution-level $J = 0$
c Result after 50 iterations using resolution level $J = 1$
d Result after 50 iterations using resolution level $J = 2$

Experiment: An experiment of the proposed algorithm is made for a 32 × 32 interpolation problem illustrated in Fig. 2*a*. In this problem, the constraints density is only 3.52% (i.e. there are 36 constrained nodes among 1024 unknown nodes). The goal is to find the smoothest surface that satisfies the constraints in those constrained nodes. This results a 1024 × 1024 linear equation system.

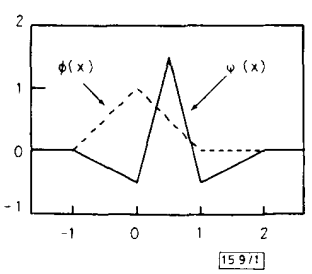

Fig. 1 *Wavelet $\psi(x)$, scaling function $\phi(x)$, and impulse responses of associated QMF filters*

$g_0(z) = 1/4(z^{-1} + 2 + z)$
$g_1(z) = 1/8(-z^{-2} - 2z^{-1} + 6 - 2z - z^2)$
$h_0(z) = 1/8(-z^{-2} + 2z^{-1} + 6 + 2z - z^2)$
$h_1(z) = 1/4(z^{-1} - 2 + z)$

The functions $\phi(x)$, $\psi(x)$ and the associated QMF filters used in this experiment are shown in Fig. 1. The Gauss-Seidel method is applied for the iterative computation. The convergence curves of the objective energy function with the proposed algorithm using different resolution levels J are plotted in Fig. 3. The top-most curve ($J = 0$) denotes the convergence status using direct iterative computation without preconditioning. It can be found that the convergence rate is significantly improved when the preconditioning is applied ($J = 1, 2$). The corresponding interpolated surfaces with different resolution levels after 50 iterations are shown in Fig. 2*b* – *d*. It can be seen that the performance is improved when the resolution level J is increased.

Conclusion: A fast algorithm has been derived for scattering data interpolation. Experimental results show that this algorithm yields significant improvement in computation speed. The proposed algorithm is easy to code and has high flexibility in implementation. The general concept of the algorithm also makes it applicable to various regularisation problems.

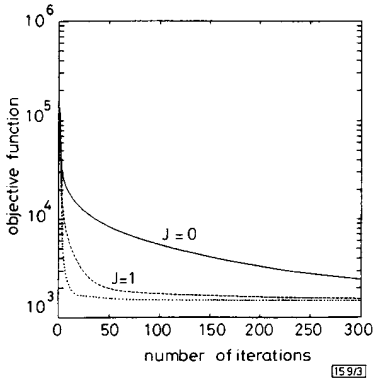

Fig. 3 *Convergence curves of energy function using different resolution levels*

16 August 1993
Electronics Letters Online No: 19931223

M.-H. Yaou and W.-T. Chang (*Institute of Communication Engineering, National Chiao-Tung University, Hsinchu, Taiwan 30039, Republic of China*)

W.-T. Chang is the corresponding author

References

1 MARR, D.: 'Vision: A computational investigation into the human representation and processing of visual information' (Freeman, San Francisco, CA, 1982)
2 GRIMSON, W.E.L.: 'An implementation of a computational theory of visual surface interpolation', *Comput. Vision Graphics Image Process.*, 1983, **22**, pp. 39–69
3 TERZOPOULOS, D.: 'Imsge analysis using multigrid relaxation methods', *IEEE trans.*, 1986, **PAMI-8**, (2), pp. 129–139
4 SZELISKI, R.: 'Fast surface interpolation using hierarchical basis functions', *IEEE Trans.*, 1990, **PAMI-12**, (6), pp. 513–528
5 MALLAT, S.G.: 'A theory for multiresolution signal decomposition: The wavelet representation', *IEEE Trans.*, 1989, **PAMI-11**, (7), pp. 647–693
6 YAOU, M.H., and CHANG, W.T.: 'Fast surface interpolation using multi-resolution wavelet transform'. Tech. Report NCTUCM-9301, National Chiao-Tung University, Taiwan, 1993

Bandwidth of crossbars for general reference model

H.-K. Chang and S.-M. Yuan

Indexing term: Multiprocessor interconnection networks

The bandwidth of crossbar multiprocessor systems is analysed for the general memory reference model. Previous solutions are restricted to several specified models: uniform memory reference, favorite memory reference and hot-spot; the presented analysis includes these as special cases.

Introduction: In a tightly coupled multiprocessor system, processors are connected via an interconnection network (IN) to memory modules so that the memory modules are shared by all processors. The readers are referred to [5] for a survey of INs.

Fig. 1 shows an $M \times N$ crossbar connecting M processors and N memory modules. A crossbar provides the capacity for all memory modules to be accessed simultaneously provided the requested memory modules are distinct. A memory conflict occurs when two or more processors attempt to access the same memory module. The bandwidth, which is defined as the expected number of requests accepted per unit time [5], is an important metric with which to estimate the performance of an IN.

Analyses of the bandwidth of crossbars for three specified

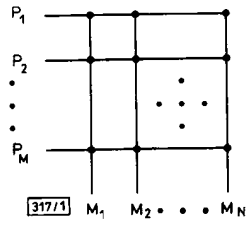

Fig. 1 $M \times N$ *crossbar*

reference models have appeared in the literature. They are the uniform memory reference [6], favourite memory reference [3] and hot-spot [1,2,8]. In [4], a survey was reported. In this Letter we analyse the bandwidth of crossbars for the general reference model.

Assumptions and notation: The analysis of this Letter is based on the following assumptions:

(i) The crossbar operates in a synchronous mode, i.e. a process can generate a request, if any, at the beginning of a memory cycle.

(ii) Requests from different processors are mutually independent.

(iii) When two or more requests are intended for the same memory modules, only one of the requests is accepted, and the others are rejected.

(iv) The requests which are rejected are discarded, i.e. the requests generated at successive cycles are independent.

Note that this Letter does not make any assumption of the memory reference model. Let P_i and M_j denote processor i and memory module j, respectively, $1 \le i \le M$, $1 \le j \le N$. The memory reference model is defined by a matrix $Q = \{q_{ij}\}_{M \times N}$, where q_{ij} is the probability that a request from P_i is intended for M_j and

$$\sum_{j=1}^{N} q_{ij} = 1$$

The probability that P_i makes a request at the beginning of each memory cycle is r_i, $0 \le r_i \le 1$. Thus the probability that P_i makes a request to M_j at the beginning of any memory cycle is $r_i q_{ij}$.

Bandwidth analysis: For $1 \le j \le N$, let X_j be a binary random variable such that $X_j = 1$ denotes the event that M_j receives one or more requests from the processors; then $X_j = 0$ denotes the event that no request from any processor is sent to M_j.

The expected value of X_j, $E\{X_j\}$, is

$$\begin{aligned} E\{X_j\} &= \Pr\{X_j = 1\} \\ &= 1 - \Pr\{X_j = 0\} \\ &= 1 - \prod_{i=1}^{M} (1 - r_i q_{ij}) \end{aligned} \tag{1}$$

The bandwidth of the crossbar is the expected value of $X_1 + \ldots + X_N$. According to assumption (ii), $X_1, \ldots, X_N$ are independent random variables. Thus, the bandwidth of the crossbar, BW, is

$$\begin{aligned} BW &= E\left\{\sum_{j=1}^{N} X_j\right\} \\ &= \sum_{j=1}^{N} E\{X_j\} \\ &= \sum_{J=1}^{N} \left(1 - \prod_{i=1}^{M} (1 - r_i q_{ij})\right) \end{aligned} \tag{2}$$

Comparisons: Eqn. 2 is compared with previous works including: the uniform reference model [6], favourite memory [3], and hot memory [2,8].