S. Oh, D.C. Park , R.J. Marks II and L.E. Atlas "Error detection and correction in multilevel algebraic optical processors", Optical Engineering, vol. 27, #4, pp.289-294 (1988).

Error detection and correction in multilevel algebraic optical processors

S. Oh
D. C. Park
Robert J. Marks II
L. E. Atlas
University of Washington
Interactive System Design Laboratory
Mail Stop FT-10
Seattle, Washington 98195

Abstract. The performance of inexact processors can be improved at the cost of throughput by using parallel redundant computations to correct errors at the processor's output. Error correcting codes applicable to binary data strings have previously been suggested for application to optical processors. We demonstrate that multilevel block codes can likewise be applied. Specific attention is given to error correction multilevel optical matrix-vector multipliers. The performance of the multilevel block code is compared to that of multilevel error correction codes formulated for VLSI processors and bar code readers. In residue-coded form (in which the matrix-vector multiplication is performed conventionally), the multilevel block code is shown overall to require fewer resolvable levels at the output.

Subject terms: optical signal processing; error detection and correction; multilevel algebraic optical processing; parallel redundant computation; vector-matrix multiplication.

Optical Engineering 27(4), 289-294 (April 1988).

CONTENTS

1. INTRODUCTION

The performance of an inaccurate algebraic optical processor can be improved at the cost of reduced throughput by parallel redundant computation. Performing a discrete algebraic operation on three different processors followed by a majority voter should, for example, increase the probability of the result being correct and, in a worst case scenario, indicate that the answer is inconclusive. A more sophisticated and computationally efficient approach is to use coded redundancy in such a way that processor errors can be detected and/or corrected at the output using simple decoding techniques.[1] Such error detection and correction coding techniques have been suggested for optical associative memories by Liebowitz and Casasent[2] and for composite matched filters by Marks and Atlas et al.[3–5] Each of these applications, however, makes use of those block codes applicable only to *binary* data strings. Optical processors, on the other hand, can operate at signal-to-noise ratios that allow multilevel output quantization levels.[6,7] Multilevel error *detection* has been suggested for optical systolic array processors by Caulfield and Putnam.[8]

In this paper, we show that conventional error *correction* block codes, when generalized to multilevels, can be straight-

Invited Paper OS-103 received Oct. 1, 1987; revised manuscript received Nov. 30, 1987; accepted for publication Dec. 8, 1987; received by Managing Editor Jan. 21, 1988.
© 1988 Society of Photo-Optical Instrumentation Engineers.

forwardly applied to increase the accuracy of multilevel optical matrix-vector multiplication processors. Additional redundant output values are computed to check the accuracy of the information elements.

Two generalizations of block codes to multilevel error correction are discussed in this paper. The first, which uses conventional integer addition and multiplication, is shown to require fewer output quantization levels for coded elements than are required by other error correction techniques.[9,10] The second code uses residue sums in the matrix coding and error decoding. The matrix multiplication, however, is still performed conventionally. Unlike the other coding methods considered, the number of required output quantization levels for residue coding is the same for both information and coded elements.

In all cases, matrix encoding is done off-line. A hybrid system of free-space and integrated optics can be used in the computation of the output syndromes. Alternatively, error decoding and correction can be performed totally by electronics.

2. A MULTILEVEL ERROR DECIMAL-BASED CORRECTING CODE

Our first coding technique is best illustrated by example, first as a code and then as applied to matrix-vector multiplication error correction.

Consider four integers g_3, g_5, g_6, and g_7. Similar to a Hamming block code, we form the sums

$$g_1 = g_3 + g_5 + g_7 ,$$
$$g_2 = g_3 + g_6 + g_7 , \qquad (1)$$
$$g_4 = g_5 + g_6 + g_7 .$$

Let $\mathbf{g}$ denote the 7-tuple vector of these numbers and let $\mathbf{d} = \mathbf{g} + \mathbf{n}$, where $\mathbf{n}$ is a vector of randomly selected integers. With reference to Eq. (1), we form the check sums

$$d'_1 = d_3 + d_5 + d_7 ,$$
$$d'_2 = d_3 + d_6 + d_7 , \qquad (2)$$
$$d'_4 = d_5 + d_6 + d_7 ,$$

and the syndromes

$$s_m = d'_m - d_m , \quad m = 1, 2, 4 . \qquad (3)$$

Three possible conclusions can be made from the syndromes: (1) If all of the syndromes are zero, we conclude that no error has been made. (2) If all nonzero syndromes are the same, we conclude that a single error has been made. We use the syndrome to locate and correct that error. (3) If the syndromes do not satisfy either of the above cases, two or more errors have been made. These errors cannot be corrected.

Example 1:

Consider the coded vector

$$\mathbf{g}^T = [\underline{7}\ \underline{8}\ 1\ \underline{9}\ 2\ 3\ 4] , \qquad (4)$$

where T denotes transposition. (The check-sum elements are underlined.) Assume that

$$\mathbf{d}^T = [7\ 8\ 1\ 9\ 4\ 3\ 4] ; \qquad (5)$$

then

$$\mathbf{s}^T = [s_1\ s_2\ s_4]$$
$$= -2 \cdot [1\ 0\ 1] . \qquad (6)$$

Thus, element $(1\ 0\ 1)_2 = 5$ in $\mathbf{d}^T$ (i.e., $d_5 = 4$) should be adjusted by −2. This is the correct result. If, on the other hand,

$$\mathbf{d}^T = [5\ 8\ 1\ 9\ 4\ 3\ 4] , \qquad (7)$$

then

$$\mathbf{s}^T = [-4\ 0\ 2] . \qquad (8)$$

Since the nonzero syndromes are different, we conclude that two or more mistakes were made. (There were two.)

In general, the minimum code length n, for m information elements in a single error correction code, is the smallest integer satisfying

$$2^m \leq \frac{2^n}{n+1} . \qquad (9)$$

For our example, n = 7 and m = 4.

This multilevel extension of a Hamming code can be straightforwardly applied to correcting errors in matrix-vector multiplication. Consider the matrix-vector multiplication

$$\mathbf{A}\mathbf{b} = \mathbf{c} , \qquad (10)$$

where, for an (n, m) = (7, 4) code, $\mathbf{A}$, $\mathbf{b}$, and $\mathbf{c}$ are $4 \times p$, $p \times 1$, and 4×1 matrices, respectively, and p is the dimension of the vector $\mathbf{b}$. We partition matrix $\mathbf{A}$ into rows

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_3^T \\ \mathbf{a}_5^T \\ \mathbf{a}_6^T \\ \mathbf{a}_7^T \end{bmatrix} . \qquad (11)$$

To allow for error correction, the matrix is augmented to a $7 \times p$ matrix $\mathbf{A}_+$. The n*th* row of $\mathbf{A}_+$ is

$$\{\mathbf{a}_n^T : 1 \leq n \leq 7\} . \qquad (12)$$

With reference to Eqs. (1), the new rows are computed via the Hamming recipe:

$$\mathbf{a}_1 = \mathbf{a}_3 + \mathbf{a}_5 + \mathbf{a}_7 ,$$
$$\mathbf{a}_2 = \mathbf{a}_3 + \mathbf{a}_6 + \mathbf{a}_7 , \qquad (13)$$
$$\mathbf{a}_4 = \mathbf{a}_5 + \mathbf{a}_6 + \mathbf{a}_7 .$$

The augmented matrix-vector multiplication is

$$\mathbf{A}_+\mathbf{b} = \mathbf{c}_+ . \qquad (14)$$

Owing to computational or other error, assume we receive

$$\mathbf{d}_{+} = \mathbf{c}_{+} + \mathbf{n}_{+} \ , \tag{15}$$

where the $\mathbf{n}_{+}$ is a vector of noise integers. Then $\mathbf{d}_{+}$ can be analyzed for errors using the multilevel Hamming coding procedure.

Example 2:

Assume that $\mathbf{b} = [0\ 1\ 1\ 2]^T$ and

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 2 & 0 & 2 & 0 \\ 0 & 2 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} . \tag{16}$$

It follows that

$$\mathbf{A}_{+} = \begin{bmatrix} 3 & 2 & 3 & 1 \\ 1 & 4 & 2 & 1 \\ 0 & 1 & 0 & 0 \\ 3 & 3 & 4 & 1 \\ 2 & 0 & 2 & 0 \\ 0 & 2 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} . \tag{17}$$

If this matrix were coded on an intensity transmittance, the resulting gray levels would not be exactly proportional to the elements of $\mathbf{A}_{+}$. We simulate this inexactness by adding zero mean Gaussian noise with a standard deviation of ⅛ to each element of $\mathbf{A}_{+}$. Motivated by the nonnegativity of intensity transmittances, negative elements are set to zero. One of our simulations yielded

$$\mathbf{A}_{+} + \text{noise} = \begin{bmatrix} 3.095 & 1.800 & 2.961 & 0.921 \\ 1.130 & 4.125 & 2.125 & 0.849 \\ 0.000 & 1.004 & 0.125 & 0.230 \\ 2.919 & 2.975 & 3.886 & 0.983 \\ 2.174 & 0.149 & 1.768 & 0.000 \\ 0.160 & 2.006 & 1.160 & 0.000 \\ 1.099 & 1.076 & 1.004 & 0.773 \end{bmatrix} . \tag{18}$$

The result of the matrix multiplication is

$$\mathbf{d} = \mathcal{R}\{(\mathbf{A}_{+} + \text{noise})\mathbf{b}\}$$

$$= [7\ 8\ 2\ 9\ 2\ 3\ 4]^T \ , \tag{19}$$

where the $\mathcal{R}$ operator rounds to the nearest integer. Using the decoding procedure, this vector corrects to $\mathbf{g}$ in Eq. (4). We have thus corrected an error due to processor inexactness.

3. OTHER MULTILEVEL ERROR CORRECTING CODES

For purposes of comparison to the multilevel Hamming code, we quickly review the multilevel error correction codes of Jou and Abraham[9] (JA) and Redinbo and Hemmann[10] (RH). Both are designed for single error correction in a multilevel matrix vector of length m. We augment the vector with two additional numbers corresponding to the inner product of the vectors,

$$\beta_1 = [1\ 1\ 1\ 1\ \ldots\ 1]^T \tag{20}$$

and

$$\beta_2 = [a_1\ a_2\ a_3\ \ldots\ a_m]^T \ . \tag{21}$$

JA uses $\alpha_k = 2^{k-1}$ and RH uses $\alpha_k = k$. If, for example,

$$\mathbf{f} = [1\ 2\ 3\ 4]^T \ , \tag{22}$$

then the corresponding codes are

$$\mathbf{h}_{JA} = [1\ 2\ 3\ 4 : 10\ 49]^T \tag{23}$$

and

$$\mathbf{h}_{RH} = [1\ 2\ 3\ 4 : 10\ 30]^T \ . \tag{24}$$

More generally,

$$\mathbf{h} = [h_1\ h_2\ \ldots\ h_m : h_{m+1}\ h_{m+2}]^T \ , \tag{25}$$

where

$$h_{m+1} = \sum_{k=1}^{m} h_k \tag{26}$$

and

$$h_{m+2} = \sum_{k=1}^{m} \alpha_k h_k \ . \tag{27}$$

Assume we receive

$$\mathbf{d} = [d_1\ d_2\ \ldots\ d_m : d_{m+1}\ d_{m+2}]^T \ , \tag{28}$$

which differs from $\mathbf{h}$ in one information element. We find the magnitude of the error from the syndrome

$$s_1 = d_{m+1} - \sum_{k=1}^{m} d_k \ . \tag{29}$$

The second syndrome points to the location of the error. Specifically,

$$s_2 = d_{m+2} - \sum_{k=1}^{m} \alpha_k d_k$$

$$= \sum_{k=1}^{m} \alpha_k (h_k - d_k) \ . \tag{30}$$

Since $h_k = d_k$ at all except the error location,

$$s_2 = \alpha_L s_1 \ , \tag{31}$$

where L is the error location.

Example 3:

For a JA code, suppose we receive

$$\mathbf{d}_{JA} = [1\ 2\ 1\ 4 : 10\ 49]^T \ . \tag{32}$$

The error is

$$s_1 = 10 - (1 + 2 + 1 + 4) = 2 \ . \tag{33}$$

The second syndrome is

$$s_2 = 49 - (1 + 2 \cdot 2 + 2^2 + 4 \cdot 2^3) = 8 \ . \tag{34}$$

Since $\alpha_k = 2^{k-1}$, we conclude that the L = third element should be adjusted by 2. Comparing Eq. (32) with the properly coded sequence in Eq. (23) confirms our result.

Codes of the form discussed in this section can be straightforwardly extended to correcting errors in matrix-vector multiplies. Matrix rows rather than vector elements are added.

Example 4:

The JA coding of matrix **A** in Eq. (16) results in

$$\mathbf{A}_{JA+} = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 2 & 0 & 2 & 0 \\ 0 & 2 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ \cdots & \cdots & \cdots & \cdots \\ 3 & 4 & 4 & 1 \\ 12 & 17 & 16 & 9 \end{bmatrix} \ . \tag{35}$$

Decoding of the vector $\mathbf{d}_+ = \mathbf{A}_{JA+}\mathbf{b}$ is performed as in Ex. 3.

4. CODE COMPARISON

A limited dynamic range and a finite signal-to-noise ratio will limit the number of resolvable levels (NRL) at the output of an optical processor. For the three codes considered thus far, the NRL for the coded portion of the matrix exceeds that of the information portion. The NRL of the output check-sum elements, as a consequence, is much higher than that for information elements.

Consider an (n,m) code. (For JA and RH, $n = m + 2$.) Suppose matrix **A** has p columns and r rows and that each element of **A** can take on only L_A values. If the input **b** can take on L_b resolvable values, then the total number of levels required for the elements of $\mathbf{c} = \mathbf{Ab}$ is

$$L = pL_AL_b \ . \tag{36}$$

The required NRL in the output check-sum elements, however, is substantially larger. For the multilevel Hamming (MH) check-sum elements, the required NRL is

$$L_{MH} \approx m\frac{L}{2} \ . \tag{37}$$

For the JA and RH codes, the required NRL are

$$L_{JA} = pL_AL_b \sum_{k=1}^{m} \alpha_k \approx 2^m L \tag{38}$$

and

$$L_{RH} \approx m^2 \frac{L}{2} \ . \tag{39}$$

Of the three NRL, L_{MH} is the smallest.

5. MATRIX CODING USING RESIDUE CHECK SUMS

In this section, we show that a slight modification of the MH code results in decoding that requires the same NRL for both information and check-sum elements. Thus, there are uniform output detection requirements. The technique makes use of residue coding and decoding.[1] The matrix-vector multiplication, however, is performed using conventional integer multiplication and addition. The technique is identical to the multilevel Hamming code but uses residue sums to code the **A** matrix. We illustrate in the following example.

Example 5:

Consider matrix **A** in Eq. (16) and assume that the maximum NRL is $L_A = 3$. Instead of forming new rows as integer sums, we add modulo 3. Equivalently, our coded modulo matrix is the element-by-element modulo base 3 of $\mathbf{A}_+$ in Eq. (17):

$$\mathbf{A}_\oplus = ((\mathbf{A}_+))_3 = \begin{bmatrix} 0 & 2 & 0 & 1 \\ 1 & 1 & 2 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 2 & 0 & 2 & 0 \\ 0 & 2 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} , \tag{40}$$

where $((g))_k$ denotes the residue of g base k. For an input of $\mathbf{b} = [0\ 1\ 1\ 2]^T$, the output using conventional multiplication is

$$\mathbf{c}_\oplus = \mathbf{A}_\oplus \mathbf{b} = [\underline{4}\ \underline{5}\ 1\ \underline{3}\ 2\ 3\ 4]^T \ . \tag{41}$$

Note that the check-sum elements are equal to their corresponding sums in modulo 3:

$$((4))_3 = ((1 + 2 + 4))_3 = 1 \ ,$$
$$((5))_3 = ((1 + 3 + 4))_3 = 2 \ , \tag{42}$$
$$((3))_3 = ((2 + 3 + 4))_3 = 0 \ .$$

Suppose we detected the vector

$$\mathbf{d}_\oplus = [\underline{4}\ \underline{5}\ 1\ \underline{3}\ 1\ 3\ 4]^T \ . \tag{43}$$

To detect negative errors, we compute the syndromes using the principal values (−1, 0, 1) rather than (0, 1, 2):

$$s_1 = (((1 + 1 + 4) - 4))_3 = -1 \ ,$$
$$s_2 = (((1 + 3 + 4) - 5))_3 = 0 \ , \tag{44}$$
$$s_3 = (((1 + 3 + 4) - 3))_3 = -1 \ ,$$

which means there is an error of −1 in the $(101)_2$ = fifth element. This is the correct result.

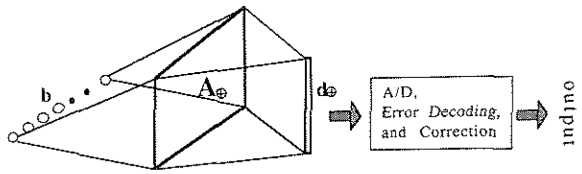


Fig. 1. An optical vector-matrix multiplier with electronic error decoding and correction.

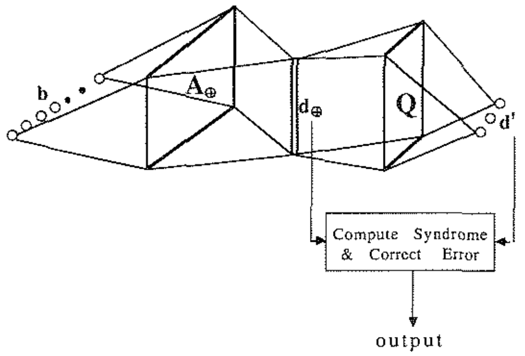


Fig. 2. A second architecture for correcting errors in an optical matrix-vector multiplier. The information portion of the processor output d is input into a second matrix-vector multiplier to compute d'.

As example 5 illustrates, use of such residue coding allows a uniform requirement on quantization levels within the coded matrix and at the processor output; i.e.,

$$L_{HR} = L \ , \tag{45}$$

where HR denotes the Hamming residue code.

The price that is paid using residue coding is a reduction of the dynamic range of the errors that can be corrected. For the MH, JA, and RH codes, a single error of any magnitude can be corrected. If L_A is odd, the HR code can correct errors only within a range of $(L_A - 1)/2$. If, however, errors are due to distributed inexactness as exemplified in Eqs. (17) and (18), we would expect the error magnitude to be small when only one output error occurs.

6. OPTICAL IMPLEMENTATIONS

Figure 1 shows a basic processor architecture for fault-tolerant computing using the multilevel Hamming residue code. The processor input corresponding to **b** is implemented by a linear array of p point source LEDs. The matrix multiplication is performed by the Stanford optical matrix multiplier.[11] (The astigmatic spreading and focusing optics have been deleted from the figure for clarity of presentation.)

The levels in the $\mathbf{A}_\oplus$ matrix are computed off-line. The matrix-vector multiply is performed in the conventional manner. Decoding is performed electronically.

Alternatively, as shown in Fig. 2, a second Stanford matrix-vector multiplier can be used to assist in computing the syndromes.[2] Each syndrome is a nonweighted conventional sum of chosen information elements in $\mathbf{d}_\oplus$. This can be performed with a **Q** matrix whose elements are either 1 or 0. For our (7,4) Hamming code example,

$$\mathbf{Q} = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} . \tag{46}$$

Then the syndrome vector is computed as

$$\mathbf{s} = ((\mathbf{d}' - [d_1 \ d_2 \ d_4]^T))L_A \ , \tag{47}$$

where $\mathbf{d}' = \mathbf{Q}\mathbf{d}_\oplus$. The detectors for the syndrome vector clearly require more resolvable levels than the detector array for **d**. The residue operations required at the decoding stage can also be performed optically.[12,13]

7. CONCLUSIONS

We have contrasted four multilevel error correction codes applicable to optical matrix-vector multipliers. The Jou and Abraham,[9] Redinbo and Hemmann,[10] and multilevel Hamming codes place severe performance constraints on the detectors for the output-weighted check sums. Use of residue-based matrix encoding, however, requires the same number of output resolvable levels for both information and check-sum elements. Matrix output vector multiplication is performed conventionally.

The multilevel Hamming codes require only the operations of addition and multiplication in the decoding process, while the other codes require a division operation as well.

Clearly, the Hamming code technique can be generalized to higher order Bose-Chaudhuri-Hocquenguem (BCH) and block codes.[1] Application to other operations, such as matrix-matrix multiplication, is also evident.

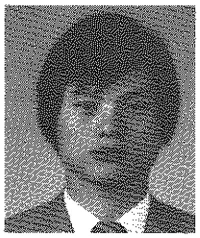
8. ACKNOWLEDGMENTS

This work was supported by the Strategic Defense Initiative Organization/Innovative Science and Technology program in Ultra High Speed Computing administered through the Office of Naval Research in conjunction with Texas Tech University. L. E. Atlas was supported by a National Science Foundation Presidential Young Investigator award.

9. REFERENCES

1. R. E. Blahut, *Theory and Practice of Error Control Codes*, Addison-Wesley, New York (1983).
2. S. Liebowitz and D. Casasent, "Error-correction coding in an associative processor," Appl. Opt. 26(6), 999–1006 (1987).
3. R. J. Marks II and L. E. Atlas, "Composite matched filtering with error correction," Opt. Lett. 12(2), 135–137 (1987).
4. R. J. Marks II, J. A. Ritcey, L. E. Atlas, and K. F. Cheung, "Composite matched filter output partitioning," Appl. Opt. 26(11), 2274–2278 (1987).
5. K. F. Cheung, L. E. Atlas, J. A. Ritcey, C. A. Green, and R. J. Marks II, "Conventional and composite matched filters with error correction: a comparison," Appl. Opt. 26(19), 4235–4239 (1987).
6. B. Braunecker, R. Hauck, and A. W. Lohmann, "Optical character recognition based on nonredundant correlation measurements," Appl. Opt. 18(16), 2746–2753 (1979).
7. A. Huang, Y. Tsunoda, J. W. Goodman, and S. Ishihara, "Optical communication using residue arithmetic," Appl. Opt. 18(2), 149–162 (1979).
8. H. Caulfield and R. S. Putnam, "Fault tolerance and self-healing in optical systolic array processors," Opt. Eng. 24(1), 65–67 (1985).
9. J-Y. Jou and J. A. Abraham, "Fault-tolerant matrix arithmetic and signal processing on highly concurrent computing structures," Proc. IEEE, 74(5), 732–741 (1986).
10. R. Redinbo and J. Hemmann, "Error-correcting coding for banking document bar codes," Electron. Lett. 22(5), 269–271 (1986).
11. J. W. Goodman, A. R. Dias, and L. M. Woody, "Fully parallel, high-speed incoherent optical method for performing discrete Fourier transforms," Opt. Lett. 2(1), 1–3 (1978).

12. C. D. Capps, R. A. Falk, and T. L. Houk, "Optical arithmetic-logic unit based on residue arithmetic and symbolic subtraction," Appl. Opt. 27 (1 May 1988).
13. R. A. Falk, C. D. Capps, and T. L. Houk, "Arithmetic-logic unit based on optical crossbar architecture," Appl. Opt. 27 (1 May 1988).



Seho Oh was born in Daejeon, Korea, in 1957. He received his BS degree in electronics engineering from Seoul National University and the MS degree in electrical engineering from Korea Advanced Institute of Science and Technology, Seoul. From 1981 through 1986, he was with Goldstar Central Research Laboratory in Seoul. Since September 1986, he has been a doctoral candidate in the Department of Electrical Engineering at the University of Washington. His present research interests are signal processing, optical neural nets, and pattern recognition.

Dong Chul Park was born in Buchon, Korea, in 1958. He received the BS degree in electronics engineering from Sogang University, Seoul, in 1980, and the MS degree in electrical engineering from the Korea Advanced Institute of Science and Technology, Seoul, in 1982. From 1982 to 1985 he was a research engineer in the Goldstar Central Research Laboratory, Seoul. Since September 1985 he has been working toward the Ph.D. in the Department of Electrical Engineering at the University of Washington. His research interests include fault tolerant computing, optical neural networks, signal processing, and pattern recognition.

R. Jackson Marks II received his Ph.D. degree in electrical engineering from Texas Tech University in 1977. He joined the Department of Electrical and Computer Engineering that same year and currently holds the rank of Professor. Dr. Marks is the cofounder and current president of the Puget Sound Optical Society of America. He is also chair of the IEEE Circuits and Systems Society's Technical Committee on Neural Systems and Applications. In 1984, he was awarded the IEEE Centennial Medal and Certificate. Dr. Marks has published more than 50 archival papers in the areas of neural networks, optical processing, pattern recognition, detection theory, and signal analysis.

Les Atlas is currently an assistant professor at the University of Washington. He received his Ph.D. in electrical engineering from Stanford University in 1983 and has been doing research in speech processing, artificial neural networks, and optics since. He is a recipient of a National Science Foundation Presidential Young Investigator Award.

S. Oh, D.C. Park, R.J. Marks II and L.E. Atlas "Error detection and correction in multilevel algebraic optical processors",
in SPIE Milestone Series: Selected Papers in Optical Computing edited by H. John Caulfield and G. Gheen,
SPIE vol.1142, pp.59-64, 1989 (The Society of Photo-Optical Instrumentation Engineers, Bellingham, WA),
reprinted from Optical Engineering, vol. 27, #4, pp.289-294 (1988).