Patents

R.J. Marks II
(1989-1990)

Robert J. Marks II

September 15, 1989

There exist a number of signal processing algorithms wherein a window, \( \varphi(k) \), shifts across a signal to give an alternate representation of the signal. Included are weighted running averages, spectrograms and zamograms \([1, 2, 3]\). Conventionally, weighted running averages are computed using the equivalent of an finite impulse response (FIR) filter the taps of which correspond to the window samples. Digitally computed spectrograms are traditionally computed by weighting the signal samples in a interval by the window weights followed by a fast Fourier transform (FFT). Digital zamograms also require the use of FFT’s for each point in time in which a spectral line is computed \([3]\).

For windows and that are uniform (i.e. rectangular or boxcar windows), the value of a signal representation generated from a sliding window can be obtained by adding to the current representation new data introduced by the shift and deleting data no longer included in the window. With non-rectangular windows, however, shifting alters the weights of all data and the procedure is no longer applicable. An approach with similar computational advantages occurs when the window is of the form \( \varphi(k) = e^{\pm s} \). Then, since \( \varphi(k \pm 1) = e^{\pm s} e^{\pm s} \), shifting from \( k \) to \( k \pm 1 \) is equivalent to multiplying each data point by \( e^{\pm s} \). Unfortunately, there are no useful windows that are exponential except the degenerate case of the rectangular window. There are, however, a number of commonly used windows that are superpositions of weighted exponentials. We refer to a weighted sum of exponentials as a Szasz series \([4, 5]\). Trigonometric polynomials are special cases. The Szasz components of the signal representation can be individually computed using
the exponential updating approach and the components superimposed to obtain the desired processing output. The generic procedure for the updating using Szasz windows, illustrated in Fig. 1 is:

1. In each Szasz component, subtract the terms that were in the previous window but not that in the current window. Likewise, add the newly introduced terms.

2. Multiply each of the elements common to both windows by the Szasz increment to effect the shift.

3. Add all of the Szasz components to obtain the desired outputs.

Two Szasz components may be complex conjugates of each other. In such cases, it is many times computationally convenient to combine the two components into a single composite component as shown in Fig. 2. Similarly, only the real portion of the output of a Szasz component may be required in certain cases.

In the next section, the Szasz series is reviewed. Application of the Szasz series to weighted running averages, spectrograms and zamograms are then presented.

1 Szasz Series Windows

A linear exponent Szasz series can be written as

\[ \varphi(k) = \sum_q \alpha_q e^{s_q k} \]  

(1)

where the \( \{\alpha_q\} \)'s and the \( \{s_q\} \)'s are possibly complex. We will assume that there are \( Q \) terms in the sum. In certain cases, we require the kernel to be even. We then use the alternate form

\[ \varphi_e(k) = \varphi(|k|) \]  

(2)

Some popularly used windows and their Szasz series representations are in Tables 1 through 4. In each case, the Szasz series is an even trigonometric polynomial so that \( \varphi_e(k) = \varphi(k) \). Each window is assumed to be zero for \( |k| > L \). Other windows that are not exactly equal to a Szasz series can always be approximated to an arbitrary accuracy by a Szasz series.
Table 4: Blackman: \( \varphi(k) = 0.42 + 0.5 \cos\left(\frac{\pi k}{L}\right) + 0.08 \cos\left(\frac{2\pi k}{L}\right), Q = 5. \)

Figure 3: An FIR implementation of the weighted running average filter. The gate denotes a unit delay.

2 Weighted Running Averages

The weighted running average, \( z(n) \), of a signal, \( v(n) \), is

\[
z(n) = \sum_{k=-L}^{L} \varphi(k)v(n-k)
\]

As is shown in Fig. 3, this process can be straightforwardly implemented on an FIR filter with \( 2L + 1 \) taps.

If the Szasz series in Eq. 1 is used, we can write Eq. 3 as

\[
z(n) = \sum_{q} z_q(n)
\]

where

\[
z_q(n) = \alpha_q \sum_{k=-L}^{L} e^{\alpha_k} v(n-k)
\]
Figure 4: An IIR implementation of the weighted running average filter.
Figure 6: When two Szasz components are related by a complex conjugate, then the two components (shown here at the top) can be replaced by a single one (shown at the bottom).
3.2 Spectrogram computation using Szasz series components

If the window in Eq. 12 is expressed in terms of the Szasz series in Eq. 1, then the spectrogram in Eq. 12 can be written as

\[ S(n, p) = \sum_q S_q(n, p) \quad (13) \]

where

\[ S_q(n, p) = \alpha_q \sum_{k=-L}^{L} e^{s_q k} v(n - k) e^{-j2\pi pk/M} \quad (14) \]

The \( q \)th Szasz component update is calculated as follows.

\[
S_q(n + 1, p) = \alpha_q \sum_{k=-L}^{L} e^{s_q k} v(n + 1 - k) e^{-j2\pi pk/M} \\
= \alpha_q \sum_{k=-L-1}^{L-1} e^{s_q (k+1)} v(n - \hat{k}) e^{-j2\pi p(k+1)/M} \\
= e^{s_q e^{-j2\pi p/M}} \sum_{k=-L-1}^{L-1} e^{s_q k} v(n - \hat{k}) e^{-j2\pi pk/M} \\
= e^{s_q e^{-j2\pi p/M}} S_q(n, p) + \alpha_q e^{-Ls_q e^{j2\pi pL/M}} v(n + L + 1) \\
- \alpha_q e^{(L+1)s_q e^{-j2\pi p(L+1)/M}} v(n - L) \quad (15) 
\]

We are again following the procedure outlined in Fig. 1. The new data is \( \alpha_q e^{-Ls_q e^{j2\pi pL/M}} v(n + L + 1) \), the old data is \( \alpha_q e^{(L+1)s_q e^{-j2\pi p(L+1)/M}} v(n - L) \) and the Szasz factor is \( e^{s_q e^{-j2\pi p/M}} \). Implementation of the specific iteration in Fig. 12 iteration is shown in Fig. 9. Since multiplication of the inputs by the arrays \( e^{j2\pi pL/M} \) and \( e^{-j2\pi p(L+1)/M} \) is common to each of the \( Q \) Szasz components, the alternate implementation shown in Fig. 10 is possible.
Figure 9: Computation of the spectrogram when the window is represented as a Qth order Szasz series. The thick lines correspond to signal flow directions of vectors parameterized by the frequency variable, $p$. The thin lines correspond to (possibly complex) scalars.
Figure 10: A second technique for computation of the spectrogram when the window is represented as a $Q$th order Szasz series.
Figure 11: When a Szasz component of a spectrogram is complex, its real and imaginary components can be realized as shown here. The real and imaginary components of the spectrogram are obtained by summing the real and imaginary components of the Szasz components.
Figure 12: The two Szasz components of a spectrogram indexed by \(q\) and \(\hat{q}\) shown on the left can be obtained by simple augmentation of the output of the \(q\)th Szasz component as shown on the right. Transposition replaces \(p\) by \(-p\) in the array \(S_q(n,p)\).

This relationship, as illustrated in Fig. 12, can be used to obtain the sum of two Szasz components, indexed by \(q\) and \(\hat{q}\), by a simple augmentation of the output of the Szasz component with index \(q\). The equivalent operation using the real and imaginary outputs of the Szasz component in Fig. 11 is shown in Fig. 13.

3.2.3 Example: Hanning and Hamming windowed spectrograms

In Fig. 14 we illustrate application of the Szasz series computation of a spectrogram for the \(Q = 3\) case when \(a_1\) is real, \(s_1 = 0\), \(a_2 = a_3\) and \(s_2 = s_3 = j\pi/L\). The Hanning (Table 2) and Hamming (Table 3) windows are special cases.

4 Zamograms

The zamogram is a display of high resolution time-frequency displays with good resolution in both domains. In the discrete domain, the zamogram of
Figure 13: The real and imaginary components of the $q$th component of a Szasz component can be straightforwardly augmented to give the sum of the real and imaginary parts of two Szasz components.
Figure 14: Generation of a spectrogram using Szasz components. Hanning & Hamming windowed spectrograms can both be thusly implemented.
The proof of these equations is straightforward. Let \( \lambda_n^- \) be the set on points in \( \Lambda_n \) but not in \( \Lambda_{n+1} \). Then
\[
\lambda_n^- = \{(m, k) | m = \frac{|k|}{2} + n + 1; |k| \leq 2L\} \tag{33}
\]
Similarly, let \( \lambda_n^+ \) denote the set of points in \( \Lambda_{n+1} \) that are not in \( \Lambda_n \). Thus
\[
\lambda_n^+ = \{(m, k) | m = -\frac{|k|}{2} + n; |k| \leq 2L\} \tag{34}
\]
Clearly, then
\[
C(n+1;p) = \sum_{(m,k) \in \lambda_n^-} \varphi(k) x(m + \frac{k}{2}) x(m - \frac{k}{2}) e^{-j2\pi pk/M} \tag{35}
\]
or, equivalently,
\[
C(n+1;p) = C(n;p) + B^+_n(m) - B^-_n(m) \tag{36}
\]
where
\[
B^+_n(p) = \sum_{(m,k) \in \lambda_n^+} \varphi(k) x(m + \frac{k}{2}) x(m - \frac{k}{2}) e^{-j2\pi pk/M} \tag{37}
\]
Equivalently, we can write
\[
B^+_n(p) = 2\Re x^*(n+1) \beta^+(n, p) \tag{38}
\]
and
\[
B^-_n(p) = 2\Re x^*(n) \beta^-(n, p) \tag{39}
\]
Substituting this and Equation(38) into Equation(36) establishes Equation(30) and the proof is complete.

4.1.1 Using Fast Fourier Transforms

We will now present two techniques to evaluate the iterations in Eq. 30.

A signal flow graph at time \( n \) is shown in Fig. 15 for direct evaluation of Equation(30). The sample signals are introduced into a shift register as
shown on the left. The shift register is tapped and each of the samples is multiplied by stored weights, \( \{ \varphi(k) \} \), as shown. The two vectors of the windowed samples are fed into two pipelined FFT processors. Transposition of the output of the lower FFT is required because there is a \( e^{j2\pi pk/M} \) term in Equation (32) rather than the \( e^{-j2\pi pk/M} \) used in Equation (31). The transposition replaces \( k \) with \( -k \) to take care of this. The delays in Fig. 15 are required to synchronize the samples \( x(n) \) and \( x(n+1) \) with the computational delays required in the processing to that point (e.g. by the FFT). These two samples are weighted by either \( \pm 2 \) after which they multiply every element of the output of the FFT processors. The real part of the resulting two vectors are summed. The sum is added to the current zogram register, and a new spectral line of the zogram emerges in vector form from the processor. The parameter \( \Delta \) is the total number of clock cycles required from input to output.

### 4.1.2 Using a Szasz Window

A second implementation is possible when the zogram’s kernel is expressed as the Szasz series in Eq. 1. The iteration in Equation (30) can be written as

\[
C(n + 1; p) = C(n; p) + 2\Re \{ x^*(n + 1) \sum_{q} b^+(n, p) \} \\
- x^*(n) \sum_{q} b^-(n, p) 
\]

where the Szasz components, \( b^\pm_q(n, p) \), can be updated as

\[
b^+_q(n, p) = e^{-i\frac{2\pi q}{M}} b^+_q(n - 1, p) \\
- \alpha_q x(n + 1) + \alpha_q e^{-2L(s_q - i\frac{2\pi q}{M})} x(n + 2L + 1) 
\]

and

\[
b^-_q(n, p) = e^{i\frac{2\pi q}{M}} b^-_q(n - 1, p) \\
+ \alpha_q x(n - 1) - \alpha_q e^{2L(s_q - i\frac{2\pi q}{M})} x(n - 2L - 1) 
\]

A proof will be presented after some discussion.
Figure 15: Iterative updating of a zamogram using FFT's.
Figure 16: Iterative updating of a zamogram using Szasz components $b_q^+$. 
A signal flow diagram for the recursion in Eq. 40 is shown in Fig. 16. Unlike the FFT implementation, we here need to tap the shift register at only five points \(x(n-2L-1), x(n-1), x(n), x(n+1)\) and \(x(n+2L+1)\).

We can express the complex \(b^\pm_q(n,p)\)'s in terms of their real and imaginary components as

\[
b^\pm_q(n,p) = b^r_q(n,p) + jb^i_q(n,p)
\]

Similarly, let

\[
x(n) = x^r(n) + jx^i(n)
\]

A corresponding implementation equivalent to that in Fig. 16 is shown in Fig. 17 using real arithmetic.

Note that both Eqs. 41 and 42 are iterations of Szasz components as illustrated in Fig. 1. The Szasz factors are \(\exp \pm (s_q - \frac{j2\pi p}{M})\). For Eq. 41, the new data is \(\alpha_q \exp[-2L(s_q - \frac{j2\pi p}{M})]x(n+2L+1)\) and the old data is \(\alpha_q x(n+1)\).

In Eq. 42, the old data is \(\alpha_q \exp[2L(s_q - \frac{j2\pi p}{M})]x(n-2L-1)\) and the new data is \(\alpha_q x(n-1)\). Implementation of the updates of the \(b^\pm_q\)'s in Eqs. 41 and 42 are illustrated in Fig. 18.

**Proof:** To show Eqs. 40, 41 and 42, we substitute Equation (1) into Eq. 37:

\[
B^\pm_q(p) = \sum_{(m,k)\in A^\pm_q} \sum_q \alpha_q e^{s_qk} x(m + \frac{k}{2}) x(m - \frac{k}{2}) e^{-j2\pi pk/M}
\]

Using the definition in Eq. 33, we find that

\[
B^+_q(m) = |x(n+1)|^2 \varphi(0) + 2Re x^*(n+1) \sum_q b^+_q(n,p)
\]

where

\[
b^+_q(n,p) = \alpha_q \sum_{k=1}^{2L} e^{-s_qk} x(n+k+1) e^{-j2\pi mk/M}
\]

The recursive form in Equation (41) can easily be established from Equation (47).

Similarly,

\[
B^-_q(p) = |x(n)|^2 \varphi(0) + 2Re x^*(n) \sum_q b^-_q(n,p)
\]

where

\[
b^-_q(n,p) = \alpha_q \sum_{k=1}^{2L} e^{-s_qk} x(n-k) e^{j2\pi mp/M}
\]

The recursion in Equation (42) follows and the proof is complete.
Figure 17: Iterative updating of a zamogram using Szasz components and real arithmetic.
Figure 18: Iterative updating of the Szasz components for the zamogram.
Realizing the real & imaginary parts of a Szasz component of a zamogram: Assume that the signal, \( x(n) \), is real. From Eq. 41, the real and imaginary components of \( b_q^+(n, p) \) follow as

\[
b_q^+(n, p) = \Re[e^{-(s_q - \frac{i2\pi}{M})}]b_q^+(n - 1, p) - \Im[e^{-(s_q - \frac{i2\pi}{M})}]b_q^+(n - 1, p) \\
- \Re[\alpha_q]x(n + 1) + \Re[\alpha_q e^{2L(s_q - \frac{i2\pi}{M})}]x(n + 2L + 1) \tag{50}
\]

and

\[
b_q^i(n, p) = \Im[e^{-(s_q - \frac{i2\pi}{M})}]b_q^+(n - 1, p) + \Re[e^{-(s_q - \frac{i2\pi}{M})}]b_q^+(n - 1, p) \\
- \Re[\alpha_q]x(n + 1) + \Re[\alpha_q e^{2L(s_q - \frac{i2\pi}{M})}]x(n + 2L + 1) \tag{51}
\]

The computational algorithm shown at the top of Fig. 19 implements these equations.

Similarly, from Eq. 42, the real and imaginary components of \( b_q^+(n, p) \) are

\[
b_q^-(n, p) = \Re[e^{-(s_q - \frac{i2\pi}{M})}]b_q^-(n - 1, p) - \Im[e^{-(s_q - \frac{i2\pi}{M})}]b_q^-(n - 1, p) \\
+ \Re[\alpha_q]x(n - 1) - \Re[\alpha_q e^{2L(s_q - \frac{i2\pi}{M})}]x(n - 2L - 1) \tag{52}
\]

and

\[
b_q^i(n, p) = \Im[e^{-(s_q - \frac{i2\pi}{M})}]b_q^-(n - 1, p) + \Re[e^{-(s_q - \frac{i2\pi}{M})}]b_q^-(n - 1, p) \\
+ \Re[\alpha_q]x(n - 1) - \Re[\alpha_q e^{2L(s_q - \frac{i2\pi}{M})}]x(n - 2L - 1) \tag{53}
\]

These two equations are implemented at the bottom of Fig. 19.

If \( x(n) \) is real and \( \phi(k) \) is real and even, then an inspection of Eq. 24 reveals that \( C(n, p) \) is also real. In this case, Eq. 40 can be written as

\[
C(n + 1; p) = C(n; p) + [x^2(n + 1) - x^2(n)]\phi(0) \\
+ 2x(n + 1) \sum_q b_q^+(n, p) \\
- 2x(n) \sum_q b_q^-(n, p) \tag{54}
\]

With reference to Fig. 19, the \( 2b_q^\pm(n, p) \) terms can be generated as shown in Fig. 20.
Figure 19: Evaluating the real and imaginary parts of $b_q^+(n, p)$ (top) and $b_q^-(n, p)$ (bottom).
Figure 20: When $\varphi(k)$ and $x(n)$ are real, only $b^+_q(n, p)$ contributes to $C(n, p)$. These real components can be generated as shown here.

30

Robert J. Marks II
Combining conjugately related Szasz components: If two Szasz components with indices $q$ and $\hat{q}$ are related by a complex conjugate as

$$b_q^\pm(n, p) = [b_q^\pm(n, p)]^*$$

then, for $\varphi(k)$ and $x(n)$ real, the contribution of the conjugate pair to $C(n, p)$ is simply $2b_q^\pm(n, p)$. The implementation follows directly from Fig. 19 and is shown in Fig. 21.

Example- Zamograms with Hanning & Hamming windows: To illustrate computation of zamograms using a Szasz series window, consider again the $Q = 3$ case where $\alpha_1$ is real and $s_1 = 0$. Let $\alpha_2 = \alpha_3$ and $s_2 = s_3^* = j\pi/L$. The Hanning (Table 2) and Hamming (Table 3) windows are special cases.

Implementation of our running example is shown in Figs. 22, 23 and 24. Figure 22 shows generation of $b_1^\pm(n - 1, p)$ on top and, for the conjugate terms, $2b_1^\pm(n - 1, p)$ on the bottom. The generation of $b_1^\pm(n - 1, p)$ and $2b_2^\pm(n - 1, p)$ is similarly shown in Fig. 23. The terms are gathered as shown in Fig. 24 to produce the zamogram, $C(n, p)$.

Note that in Figs. 22 and 23, the multiplication of $x(n + 2L + 1)$ and $x(n - 2L - 1)$, respectively, by the sinusoidal arrays is common to both the $q = 1$ and $q = 2$ stages. As in Fig. 10, the commonality allows a single sinusoidal array multiplication. Such modification of Fig. 22 is shown in Fig. 25. A similar modification is readily applicable to Fig. 23.

5 Notes

Some final remarks follow.

1. The Szasz series window is also potentially applicable to certain other generalized time-frequency representations (GTFR's) [6]. Kernels with Hourglass and diamond shapes [3] in the $(m, k)$ plane can be evaluated by Szasz series windows when, within the shape, the window is $\varphi(k)$. The zamogram has a cone-shaped kernel [3] in the $(m, k)$ plane.

2. In many spectrograms and GTFR's, output spectral lines are not computed at every signal sample point. The Szasz series
Figure 21: If two Szasz components with indices \( q \) and \( \hat{q} \) are related by a complex conjugate and \( \varphi(k) \) and \( x(n) \) are real, then the contributions of both terms to \( C(n,p) \) are simply \( 2b_{q}^{r}(n,p) \). As shown here, they can be generated as shown here by simply multiplying the outputs in the previous figure by 2.
Figure 22: Generation of the $b^{+\tau}_t(n-1,p)$'s for Hanning and Hamming windows.
Figure 23: Generation of the $b_2^{-r}(n-1,p)$'s for Hanning and Hamming windows.
Figure 24: Generation of the zamogram using the inputs generated in the previous two figures.
Figure 25: A modification wherein the sinusoidal array common to both components is computed but once.
window approach can be adapted to such cases in one of two ways. First, and most obvious, the iteration can proceed at each point with outputs generated periodically. Secondly, the iteration can be modified to the longer period. For example, in the weighted running average example, if there is to be an output at every other input sample point, then, at each iteration, two new samples would be introduced (instead of one) and two old samples would be deleted (instead of one). Each Szasz factor would be squared.

3. For the spectrogram (and the spectrogram component of the zamogram), computation of the output spectral line can be viewed as a number of multiplexed IIR filters parameterized by $p$. The only time one filter "talks" with another is in the operation of transposition.

4. There exist a number of modifications to the implementation of the Szasz signal processing algorithms that correspond directly to the commutative, distributive and associative laws applied to multiplication and addition. Performing a single sinusoidal array operation in Fig. 25 (compare with Fig. 22) is an example of a variation due to the distributive law.

References


July 26, 1989

Mr. Peter Odabashian
Director, External Affairs
Washington Technology Center
University of Washington
376 Loew Hall, M.S. FH-10
Seattle, WA  98195

Re: Title: OPTICAL NEURAL NET MEMORY
U.S. Patent No. 4,849,940
Issued: July 18, 1989
Patentee: R.J. Marks, II et al.
Your Reference: WTC #87-6
Our Reference: WTCC-1-3835

Dear Peter:

We are pleased to inform you that the subject patent issued on July 18, 1989. Typically, the official Letters Patent comes to us from the United States Patent and Trademark Office several weeks after the stated issue date. We will correspond with you at that time.

While it is not mandatory to use the patent number in marketing embodiments of the invention, the failure to do so may result in an inability to collect damages in the event the patent is infringed. The statute (Title 35, United States Code, Section 287) provides as follows:

Patentees, and persons making or selling any patented article for or under them, may give notice to the public that the same is patented, either by fixing thereon the word "patent" or the abbreviation "pat.", together with the number of the patent, or when, from the character of the article, this cannot be done, by fixing to it or to the package, wherein one or more of them is contained, a label containing a like notice. In the event of such failure so to mark, no damages shall be recovered by the patentee in any action for infringement, except on proof that the infringer was notified of the infringement and continued to infringe thereafter, in which event damages may be recovered only for infringement occurring after such notice.
If you are in doubt as to the proper use of the patent number on a product or in connection with a process, please let us know.

Very truly yours,

CHRISTENSEN, O'CONNOR, JOHNSON & KINDNESS

By
Michael G. Toner

MGT/jlm/lkb
Even before he entered high school, Gordon Gould knew he wanted to be an inventor. His heroes were Marconi, Bell, and Edison. He knew, too, that to invent anything truly significant he'd have to understand the physics of things, how things worked deep down in the invisible quanta. In high school, college, and graduate school he gathered the tools. He wanted to be ready when the light bulb flickered. On November 9, 1957, a Saturday night just given to Sunday, Gould was unable to sleep. He was 37 years old and a graduate student at Columbia University. The idea came to him, he remembers, about one o'clock. No mere Soft White, this bulb. For the rest of the night and the rest of the weekend, without sleep, Gould wrote down descriptions of his idea, sketched its components, projected its future uses.

On Wednesday morning he hustled two blocks to the neighborhood candy store and had the proprietor, a notary, witness and date his notebook. The pages described a way of amplifying light and of using the resulting beam to cut and heat substances and measure distance. "That notebook is absolutely incredible," says Peter Franken, a professor of physics and optical sciences at the University of Arizona, in Tucson. "It's as if God came down and whispered in Gordon's ear and said, 'Listen, buddy, this is what you're going to do.'"

Gould dubbed the process light amplification by stimulated emission of radiation, or laser, and he knew—he knew, no question—that this was the invention he'd been preparing himself for all along. The invention of a lifetime. It was indeed, in a way Gould did not anticipate. For it took nearly half a lifetime—the next 30 years—to win the patents for his ideas. At times the government's resistance to Gould's claims was so stubborn, its behavior so unusual, that he and his allies began to fear a concerted government-industry effort to keep Gould from ever getting a patent.

Gould's vindication came only last year, when he won the last of a series of victories that left him in control of patent rights to perhaps 90% of the lasers used and sold in the United States, lasers that weld auto parts, destroy skin cancers, aim weapons, and register prices at the checkout counter. Gould's patents directly affect some half-billion dollars in annual sales of lasers; ironically, had they been granted 30 years ago these patents would have expired while the industry was still tiny, and would have captured only a fraction of their current revenue. The company formed to license the Gould patents, Patlex Corp., now sits atop a rapidly growing mountain of cash.
and last summer it hired Frank Borman, moon pilot and former chief executive of Eastern Air Lines Inc., to be its new boss.

For Gould especially, victory is very, very sweet. Every other day a Federal Express truck arrives at his home in Virginia bearing license contracts to sign. Every quarter a check comes. A grin breaks across Gould's face, a Cheshire cat's grin flecked with canary feathers, as he matter-of-factly estimates that total royalties will be $46 million. "That's my share of it."

But Gould is 68 years old. He and his partners, men who gambled their futures to back him, spent more than $6 million fighting both the United States Patent and Trademark Office and the laser industry. The story is not one of courage and perseverance only on Gould's part. Gary Erbbaum liquidated his company and bet the proceeds on Gould. Richard Samuel, a patent attorney, gave up his law partnership to become Gould's master strategist. Gould fought history—and won.

GORDON GOULD, FOR NOW, LIVES IN A small, gray ranch house situated by a creek in Virginia's Northern Neck, two and a half hours from Washington, D.C. The place is modest because that's the way Gould likes to live, not because he can't afford better. He's already a millionaire. At the rear of the house is a huge all-weather porch, and Gould is sitting there in the smoke of an endless chain of cigarettes.

He is a lean, angular man, with heavy-framed glasses and a scalp that has yielded some to the advance of time. There is a war-torn aspect to the room symbolic of the battles so recently won. Smoke. Ragged butts jamming two ashtrays. Gamma, a German shepherd with one blood-fused eye and severe hip dysplasia, moves sideways across the room, a dog in serious misalignment. Gould lives with his longtime companion, Marilyn Appel. Of dragonish temperament, she is tough, energetic, and blunt, a screener of calls, guardian of the gate. Now and then she charges onto the porch, lights a cigarette, catapults herself into the conversation. Gould sits at rest, a portrait of physical entropy.

What kept him going all these years was sheer, blissful ignorance. "What you have to realize," he says, "is that at no point did I expect it was going to take more than a couple of years to resolve whatever problem existed at a given moment." Only once, he says, did he fear he would never get a patent.

"What'd you say?" Appel asks, squinting through wayward smoke. "That was the only moment? Or the first moment?"

"Well, OK. It was the first moment."

Gould was born on July 17, 1920, in New York City. He was the kid who fixed clocks for neighbors. At Union College, in Schenectady, N.Y., he studied physics and fell in love with light. He went to Yale in 1941 to begin work toward his doctorate, but war forced him to quit. Over the next two years, he worked on the Manhattan Project, the ultimate in applied physics. In 1945, indulging his girlfriend, he began attending meetings of a Marxist study group in Greenwich Village. The government yanked his security clearance. He took a job at a company that made specialized mirrors and spent the rest of his time trying to develop inventions.

In 1951 Gould resumed his doctoral work at Columbia. He taught part-time at City College of New York until 1954—Senator Joseph R. McCarthy's heyday—when he was called before a special panel of the state board of higher education commissioned to root communies from the halls of academe. Gould spent a day under interrogation but refused to testify against colleagues and friends. He was fired. His faculty adviser at Columbia, incensed by this treatment, got Gould a research assistantship at the university's radiation lab.

Meanwhile, a Columbia physicist, Charles H. Townes, had devised a method of amplifying microwave energy, an advance he dubbed the maser. To do the same with light required a radically different approach, and it was this process Gould conceived that night in 1957. "I almost immediately saw the tremendous potential of this device," Gould says. "It would do for light what the vacuum tube and later the transistor did for radio frequency electronics."

He envisioned lasers used to heat, weld, and cut; to machine parts; to measure distance; even to produce the heat necessary for nuclear fusion, technology only today being seriously investigated.

It was then that Gould made the mistake of a lifetime—a mistake that in the grandest of paradoxes promises to make him an extremely wealthy man.

IN COURTSROOMS AROUND THE COUNTRY, there are mounds of Gould paper. In the National Archives, a full cart of boxes accounts for a single lawsuit. At one point the patent office set aside a separate room for the Gould patents and took reservations from companies wanting a look.

The patent office lives paper, breathes paper—most of it precise, legal, notarized, certified, a massive white drift of painfully accurate prose. Even with the help of a patent lawyer, few applications succeed on the first round. Two out of three, however, eventually will become patents. In fiscal 1988 this rite of passage—the pendancy period—was 19.9 months. To understand why Gould needed 30 years, it's necessary first to know the ritual.
For the inventor, this bureaucracy becomes distilled in a single individual: the patent examiner, the high priest of invention. There are 1,400 examiners, each possessing a startling degree of control over the fate of an idea. On the average, an examiner will spend 17 hours on each case. How does 17 hours become 19.9 months? The initial processing takes a month. An examiner won't get the case for another two to three months. The inventor has three months to respond to each formal action the examiner takes. In a typical case there are two such actions. Throw in another three months for printing and publishing the final patent, and you've spent more than a year.

A challenge to a patent dramatically extends this pendency period. When two applications conflict, the patent office can begin what is called an interference proceeding to determine who was the first inventor. These proceedings last decades. In another type of proceeding, called reexamination, the challenger can trigger rejection of the patent by producing new evidence of prior art—new evidence that the invention was "not novel or was obvious."

Gould took the proper first step and consulted a patent attorney for advice on how to proceed. "I was so ignorant of the whole patent procedure that I came away from that meeting with the wrong impression, which was that I had to build a model in order to get a patent," Gould recalls. This is where he made his big mistake: patent law requires no such thing. Gould needed only to present enough detail to allow someone skilled in the art to build the device.

Gould was so excited about his laser ideas that he left Columbia without finishing his thesis. He joined Technical Research Group Inc. (TRG), a small scientific company on Long Island, hoping to develop laser applications. In 1959 he won TRG a $1-million contract for laser research, and filed for his patents. But he had lost precious time. Charles Townes and Arthur Schawlow, a Bell Labs physicist, had applied the previous July to patent the optical maser.

Soon after receiving notification that the contract had been awarded, Gould also learned that he had not disclosed enough detail to enable anyone to figure out whether they were dealing with his invention. What Gould needed were big extrapolations, perhaps best known for guaranteeing the college educations of an entire sixth-grade class in Harlem. "My own associates thought I was nuts."

Gould signed with REFA C, expecting that the company would help him win his patents. But REFA C contended it had agreed only to license Gould's inventions. What Gould needed were big legal guns and the big bucks to pay them.

Gould turned 55 that year. He still did not possess a single significant U.S. laser patent.

Gordon Gould, referred to the firm by REFA C, quietly explained to Samuel that he had invented the laser. Gould presented his application, all 113 pages and 19 drawings. He presented other official papers, including a document dated only six weeks earlier. This was especially striking. The ritual of the patent process demands that an inventor keep the chain of action and response going. Once this chain is broken, the patent office considers the application abandoned. "With Gould," Samuel says, "the more I delved, the more I believed he was right."

Samuel decided Gould's claims indeed had merit, and the firm agreed to pursue them for up to

In March 1969 Schawlow and Townes received the optical maser patent. For the next 17 years this and Townes's earlier patent would be considered the laser patents. Two months later Theodore Maiman, a scientist at Hughes Research Laboratories, built the first working laser.

Gould's application met its first serious resistance in the early 1960s, when it became mired in the first of five interference proceedings. Although Gould won some claims, he also lost important ground. The patent office ruled that he

TO: WTC/UW Principal Investigators  
FROM: Edwin B. Stear 
Executive Director  
SUBJECT: Technology Disclosures  

This memorandum, along with the enclosed materials, is intended to provide specific guidance on the handling of technology disclosures through the WTC, as well as clarify The Washington Technology Center's Patent and Copyright Policy in general.  

As you know, President Gerberding in October 1985 signed Administrative Order No. 17 which exempted the WTC from UW patent and copyright policies and delegated authority to the WTC to have and administer its own Patent and Copyright Policy subject to certain conditions (see the enclosed copy). Subsequently, the WTC Board of Directors approved a WTC Patent and Copyright Policy. Although a copy of this policy was distributed to you some months ago, it is included here to provide a self-contained information packet.  

To provide further background, I am enclosing copies of the WTC Principles governing patent and copyright policies and procedures, and the Agreement between The Washington Technology Center and the Washington Research Foundation (WRF).  

Finally, in accordance with the documents identified above, the enclosed technology disclosure policy is provided for your information and use in disclosing inventions related to WTC research projects. As noted in the instructions, the disclosure will generally be forwarded to the Washington Research Foundation (or other agent), at the discretion of the WTC, for evaluation of patents and commercial potential.  

Please feel free to contact me if you have any questions concerning these policies or procedures.  

EBS/bf  
Enclosures
INVENTION DISCLOSURE

Washington Technology Center

Instructions

This Invention Disclosure Form is used to report inventions and to record the circumstances under which the invention was made. The Disclosure is a legally important document; care should be taken in its preparation since it provides both the basis for determining patentability and the data for drafting a patent application.

New and potentially useful technology developed by WTC employees with WTC and/or industry grant and contract support should be reported promptly consistent with the Center's Patent and Invention Policy.

The following instructions apply to the correspondingly numbered sections of the form.

1. Use a brief title, sufficiently descriptive to aid in identifying the invention.

2. Provide a brief description, pointing out novel features of the invention. Attach additional material which covers the following points:
   a. General purpose
   b. Technical description with references to drawings, schematics, sketches, flow diagrams, etc., as appropriate
   c. Advantages and improvements over existing methods, devices or materials, and features believed to be new
   d. Possible variations and modifications
   e. State-of-the-art prior to invention, and similar or related patents (if known)

3. List all sources of support for the research which led to the conception or actual reduction to practice of the invention. Include WTC personnel, funds or materials as well as those of University or outside agencies, organizations and companies.

4. The invention history is legally important in determining the priority of invention and/or legal "bars" to patenting. The United States Patent law allows submission of a patent application up to one year after an enabling disclosure of the technology. Most foreign countries require a patent application prior to any enabling disclosure (an oral presentation or publication such as an article, abstract or theses, or other communication which would allow a knowledgeable person to duplicate the work).
5. List all reports, abstracts, papers, theses or patent applications which have been or are planned to be submitted by the inventor(s) describing the invention. Give dates of submission and actual or anticipated publication dates. Attach documents, if available. These documents may be used in part to respond to Section 2.

6. List any other known references, patents, patent applications or other publications pertinent to this invention. Attach copies, if available. These documents may also be used in part to respond to Section 2.

7. Describe and date any sale or public use of the invention in the United States. Specify if the use was operational, or for testing purposes, and if there was any effort or intent to maintain invention secrecy after operational use began.

8. List all co-inventors (any individuals who conceived an essential feature of the invention, either independently or jointly with others, during the evolution of the invention). In the event a patent application is filed, inventorship will be verified by the patent attorney.

9. Arrange for two technically qualified witnesses to read and sign this document verifying that they have understood the invention that is disclosed.

Submit the completed Disclosure together with the Transmittal form to Dr. Edwin B. Stear, Executive Director, Washington Technology Center, University of Washington, Mail Stop FH-10, Seattle, Washington 98195. Generally it will then be forwarded to the Washington Research Foundation (or another agent) for evaluation of patentability and commercial potential.

For further information, contact The Washington Technology Center, (206) 545-1920.
WASHINGTON TECHNOLOGY CENTER
INVENTION DISCLOSURE

This invention disclosure is an important legal document and should be completed carefully. Please refer to the attached instructions.

1. Title of Invention

2. Brief Description

3. Funding Source(s)

4. Invention History

<table>
<thead>
<tr>
<th>Date</th>
<th>Location and Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A. Initial Idea

B. First description of complete invention, oral or written

C. Invention development records, notes, drawings (evidence of diligence)

D. First successful demonstration, if any (first actual reduction to practice)

E. First publication with full description of invention (may bar patent)

F. First verbal description to others

5. List all reports, abstracts, papers, theses or patent applications related to the inventions which have been published or are planned to be submitted by the Inventor(s). Include copies if available.
6. List any other references, patents, patent applications or other publications which may be pertinent to the invention. Include copies if available.

7. Describe and date any sale or public use of the invention in the United States.

8. Inventor or Co-inventors

<table>
<thead>
<tr>
<th>Signature</th>
<th>Date</th>
<th>Signature</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name (Print)</td>
<td>Title</td>
<td>Name (Print)</td>
<td>Title</td>
</tr>
<tr>
<td>Address</td>
<td></td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>Telephone</td>
<td></td>
<td>Telephone</td>
<td></td>
</tr>
<tr>
<td>Signature</td>
<td>Date</td>
<td>Signature</td>
<td>Date</td>
</tr>
<tr>
<td>Name (Print)</td>
<td>Title</td>
<td>Name (Print)</td>
<td>Title</td>
</tr>
<tr>
<td>Address</td>
<td></td>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>Telephone</td>
<td></td>
<td>Telephone</td>
<td></td>
</tr>
</tbody>
</table>

9. Invention disclosed to and understood by (two witnesses required):

<table>
<thead>
<tr>
<th>Signature</th>
<th>Date</th>
<th>Signature</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name (Print)</td>
<td></td>
<td>Name (Print)</td>
<td></td>
</tr>
</tbody>
</table>

Submit completed Disclosure to the Washington Technology Center, University of Washington, 376 Loew Hall, M/S FE-10, Seattle, WA 98195.
THE WASHINGTON TECHNOLOGY CENTER

Form to Transmit Invention Disclosure
(For WTC Internal Use Only)

Instructions

Complete this form and the attached Invention Disclosure form and forward to The Washington Technology Center via WTC Program Director, Department Chairperson, and Dean of School/College for approval. If more than one Department is involved, obtain signatures from all Chairpersons and Deans (or their designate).

To: Washington Technology Center Date: __________________
Loew Hall 376, FH-10

From:

Inventor Name  Title  Department  Mail Stop

Inventor Name  Title  Department  Mail Stop

Inventor Name  Title  Department  Mail Stop

Inventor Name  Title  Department  Mail Stop

Re: Invention entitled:  ______________________________________

Verifyed and Approved:  Concurrence:
WTC Program Director  Dean of the School/College
Date:  __________________

Concurrence:  Accepted:
Department Chairperson  Edwin B. Stear, Executive Dir.
WASHINGTON TECHNOLOGY CENTER
Date:  __________________
ADMINISTRATIVE ORDER NO. 17

Effective October 18, 1985

SUBJECT: Exemption of the Washington Technology Center from the University of Washington Patent and Copyright Policies and delegation of authority to the WTC to have and administer its own Patent and Copyright Policy subject to certain conditions.


A. The Washington State Legislature, in Chapter 72, Section 11, Laws of the 1983 1st Extraordinary Session, with the concurrence of the Governor, has established The Washington Technology Center (WTC) at the University of Washington (UW) to be administered by the Board of Regents of the UW. Accordingly, unless otherwise specified, the WTC is subject to UW policies. However, the WTC Board of Directors and the UW Administration, acting under delegated authority from the UW Board of Regents, have agreed that in light of the purposes, goals, objectives and intended nature of the WTC, it should not be fully subject to UW Patent and Copyright Policies but should adopt its own Patent and Copyright Policy.

B. The WTC is exempted from UW Patent and Copyright Policies subject to certain conditions as follows:

1. the WTC may identify itself as the owner of inventions, patents and copyrights derived from WTC projects;

2. those inventions, patents and copyrights will be administered under a WTC Patent and Copyright Policy approved by the WTC Board and the UW Administration; and

3. the WTC will enter into a Technology Administration Agreement (TAA) with the Washington Research Foundation (WRF) that is identical in all substantive respects with the TAA between UW and WRF attached hereto as Exhibit A.

This Administrative Order No. 17 is pursuant to the authority cited above.

William P. Gerberding
President
PRINCIPLES GOVERNING
PATENT AND COPYRIGHT POLICIES AND PROCEDURES

1. The Washington Technology Center, hereafter referred to as the WTC, shall own all patents and copyrights arising from WTC sponsored research and technology development programs and projects.

2. The WTC shall negotiate all patent and copyright agreements and licensing arrangements so as to maximize technology transfer for the benefit of the economic development of the State of Washington.

3. Negotiations of patent and copyright agreements and subsequent licensing arrangements shall be the responsibility of the duly appointed individual in charge of the WTC Office at the appropriate participating university in accordance with WTC policies and procedures.

4. The WTC shall develop a Patent and Copyright Policy which will form the basis for negotiation of specific agreements on patents, copyrights, licensing, and distribution of royalty income with each of the participating universities.

5. The WTC shall negotiate up-front patent and copyright agreements, including licensing provisions, with all participating industrial sponsors of WTC programs and projects.

6. The WTC shall negotiate individual up-front patent and copyright agreements with all Industrial Fellows and their employers.

7. All individuals participating in WTC programs and/or projects shall sign an agreement requiring them to be bound by the WTC's Patent and Copyright Policy.

8. When investigators from more than one university work on a WTC project, there shall be a specific up-front agreement among all parties covering patent and copyright issues, including negotiation of agreements with industrial supporters of the project, negotiation of licenses for any intellectual property developed, distribution of royalty income, and ownership of any patents or copyrights in the event the WTC is terminated or ceases to operate for any reason.

WTC
10/22/85
THE WASHINGTON TECHNOLOGY CENTER

Patent and Copyright Policy

1. One of the primary missions of The Washington Technology Center (hereinafter referred to as WTC) is to develop new commercializable technology through joint industry-university research and technology development programs. Patents and copyrights are important in this process to:

(a) protect the economic interests of the WTC and the inventors.
(b) protect the economic interests of the industrial participants and the licensees.
(c) provide a firm legal basis for transferring the technology.

It is recognized that the value of the technology may diminish rapidly with time. Therefore, it will often be necessary to transfer technology immediately after disclosure and prior to application for or issuance of patents and copyrights.

Further, it is recognized that it will also be necessary to transfer technology without applying for patents or copyrights in those cases where the technology is not patentable or copyrightable, or where the value of the particular patent or copyright does not justify the expense.

The purpose of this document is to set forth the specific policies adopted by the WTC to assure that these requirements and goals are met.

2. As a condition of participation in WTC research projects, all personnel participating in WTC projects agree to assign their title and rights to all inventions and copyrightable material arising in connection with such research projects to the WTC, to an agent designated by the WTC, or to a sponsor, if required under agreements governing sponsored research. Such personnel shall execute documents of assignment and do everything reasonably required to assist the assignee(s) in obtaining, protecting, and maintaining patents, copyrights or other proprietary rights.

The WTC has no vested interest in inventions or copyrightable material conceived and developed by participants entirely on their own time and without the use of WTC facilities. However, in order to clarify the inventor's or creator's
title to such inventions and/or copyrightable material and to insure compliance with the requirements of any sponsors, all inventions and/or copyrightable material generated during participation in WTC programs and projects shall be reported to the WTC for determination of the degree of WTC interest.

If the WTC, in consultation with the appropriate participating universities, determines that it has no interest in an invention or copyrightable material or decides to forego the patenting, copyrighting, or other commercialization of an invention or copyrightable material, it shall waive its rights to the invention or copyrightable material in writing. Upon receipt of such a waiver, and assuming that no additional WTC or University resources will be invested, the inventor(s) or creator(s) may file a patent or copyright application and/or grant a license of his/her own.

3. WTC research funded wholly or in part by an outside sponsor is subject to this policy as modified by the provisions of negotiated agreement(s) covering such work. It is the general policy of the WTC to negotiate all such agreements, including any special provisions relating to the intellectual property, prior to initiation of the research effort being sponsored. Participants in such sponsored research are bound by the provisions of these agreements.

4. In general, title to any inventions and/or copyrightable material conceived and first reduced to practice in the course of research carried out in the WTC with the support of Federal agencies, industry, or other sponsors shall vest in the WTC. In rare cases, an industrial sponsor may possess a dominant patent or copyright position in a certain technology area so that any patent or copyright the WTC might seek would be of little value. For this or other such reasons, an exception to this WTC title policy may be approved when to do so would honor the general principles of this policy, protect the equities involved, and satisfy the requirements of the parties. In all cases, the granting of such exceptions must be explicitly covered in the agreements referred to above in Paragraph 3.

5. Interaction between the WTC and industry can take any one or more of the following forms: grants, contracts, consortial arrangements, equipment gifts, and appointment of industrial fellows. Industrial firms sponsoring WTC research programs through any one or more of these forms may be assured of at least a non-exclusive license to inventions and copyrights conceived and developed with their support. If necessary for the effective development and marketing of a WTC invention or copyright, an exclusive license may be granted for a limited period of time if the sponsor agrees to finance the cost of the WTC's patent or copyright application and observes due
diligence in bringing the technology involved into public use. In such cases, the patent or copyright costs may be treated as an offset against royalties payable when the invention or copyright is marketed.

Where the sponsor uses the invention or copyright entirely within its own operations, the license may be royalty-free. Where the sponsor, or a third party licensee, manufactures and sells products, services, or processes based on the invention or copyright, reasonable royalty payments to the WTC or its assignee are normally required.

In all cases involving industrial sponsorship of WTC research programs, the specific licensing rights of the sponsor(s) to any patentable and/or copyrightable technology generated in the research programs shall be explicitly covered in the up-front agreements referred to above in Paragraph 3.

6. Although the WTC reserves the right to patent and/or copyright intellectual property itself, it has designated the Washington Research Foundation as its primary patenting, copyrighting, and licensing agent. However, another comparable, mutually-acceptable patenting, copyrighting, and licensing agent can be used if so desired by an individual participating university.

7. Both the inventors and/or creators and the WTC are entitled to a share of royalty income from licensed patents and/or copyrights; the WTC on the basis of salary and/or facilities support for the inventor and/or creator and the cost of patent, copyright, and licensing administration; and the inventor and/or creator on the basis of the creative activity, documenting the invention or copyright, and assisting as necessary with commercialization. To recognize creativity and to encourage prompt disclosure of potential patents and copyrights, the WTC allocates the greater share of net early royalty income to the inventor or creator. The remainder is dedicated to further research by allocating shares to the WTC and to the home colleges/departments of the inventors and/or creators as appropriate. Unless amended in an agreement with a participating university, the specific allocation shall be as follows.

After deducting 15% for administrative services, net royalty income received from WTC inventions and/or copyrights handled by an outside agency is distributed as follows:
In the event that an invention and/or copyright is administered directly by the WTC, the direct costs of obtaining and maintaining the patent(s) and/or copyright(s) must be recovered in addition to the 15% service fee before distribution of royalty income begins under the above formula.

The royalty derived WTC Research Fund shall be used to promote additional research in areas identified for emphasis by the WTC.

When a proposed WTC program or project involves more than one university, it is the general policy of the WTC to negotiate an up-front agreement with the participating universities covering patent and copyright issues. Including negotiation of agreements with industrial supporters of the project, negotiation of licenses for any intellectual property developed and distribution of royalty income and ownership of any patents and copyrights in the event the WTC is terminated or ceases to operate for any reason.

8. As a public institution, the WTC should undertake sponsored research under conditions which permit timely publication of the research results. However, the WTC reserves the right to defer publication for a reasonable period of time during which the WTC and any sponsor(s) review the feasibility and desirability of patent and/or copyright protection of any intellectual property described in the proposed publication. Likewise, through consultation with appropriate university officials, graduate student theses or dissertations containing invention details may be withheld from the Library shelves for a limited period while this evaluation process is conducted.

Some research agreements may involve WTC access to a sponsor's proprietary data. In all such cases, a clause defining the conditions under which such data will be identified, accepted, used, and controlled shall be included in the up-front agreement referred to in Paragraph 3, or in an amendment thereto. (Where the work is related to a thesis, students must be able to participate in such research in a meaningful way without access to such proprietary data).
When publication of research results based on use of such proprietary data is contemplated, the WTC will agree to provide the sponsor with advance copy of any proposed publication prior to submission for publication to allow the sponsor an opportunity to identify any inadvertent disclosure of its proprietary data.

9. Consultation with commercial enterprises by WTC technical experts can be of significant benefit to the WTC, the employee, the commercial entity and the general public. However, such involvements include the potential for conflicts of interest, for the inhibition of the free exchange of information, and for interference with the experts' allegiance to the WTC and to their university if they also have university affiliations. In order to minimize the potential for such conflicts and as a condition for continued involvement in WTC research projects, all proposed consulting arrangements by WTC staff must be approved by the Executive Director of the WTC, in addition to approval by the appropriate authorities in their respective universities.

Invention clauses in any such consulting agreements must be consistent with the policy of the WTC, with WTC commitments under sponsored research agreements, and, where the consultant is employed by a university, with the policies of that university. Questions concerning potential conflicts should be referred to the Executive Director or Associate Director of the WTC through appropriate university authorities.

10. In the event that the WTC is terminated or ceases to operate for whatever reason, its ownership of inventions, patents and copyrights, whether administered directly by itself or assigned to WRF or another agent, shall revert to the university at which the research leading to the invention, patent or copyright was carried out in accordance with specific agreements when more than one university is involved.

11. The Technology Transfer Committee of the WTC's Board of Directors is responsible for oversight of the WTC Patent and Copyright Policy.

WTC
10/23/85
AGREEMENT

AGREEMENT made as of November 12, 1985 between the Washington Research Foundation (the "Foundation") and the Washington Technology Center (the "Center").

RECITALS

The Foundation has been formed to stimulate productive commercial applications of inventions and other technology discovered and developed at the Center as well as other research institutions in the State of Washington. The Center and the Foundation wish to provide for the disclosure to the Foundation of certain technology (the "Technology"), which may presently or hereafter be owned by the Center, for the purpose of development and management of such Technology by the Foundation, including licensing and marketing of such Technology, the pursuit of patent applications, and the development of commercial applications for such Technology.

AGREEMENTS

1. Submission and Evaluation of Technology. The Center may from time to time deliver to the Foundation, at the Center's sole discretion, disclosures of Technology (each such disclosure referred to herein as a "Technology Project"), and the Foundation agrees to evaluate each Technology Project expeditiously. If in the Foundation's judgment the Technology has significant commercial potential, the Foundation will use its best efforts to introduce the Technology Project into commercial use and to secure royalties or other compensation therefrom as it deems appropriate. If the Foundation decides not to pursue the development of the Technology Project, it will so inform the Center in writing no later than ninety (90) days after initial receipt by the Foundation of the Center's disclosure of the Technology Project and, with such notice, shall return to the Center all materials embodying, reflecting or describing the Technology Project. If the Foundation accepts the Technology Project for commercialization, the Foundation will promptly notify the Center of such acceptance in writing. Upon such notification, the Center will assign to the Foundation all rights of the Center in such Technology Project and will execute such instruments as may be necessary to secure the ownership, right, title and interest in the Foundation of such Technology Project, subject to the provisions of this Agreement. The Foundation will thereafter, with due diligence, undertake the commercialization of the Technology Project.
2. Confidentiality. All disclosures made by the Center to the Foundation with respect to Technology shall be treated by the Foundation as confidential in their entirety. It is understood by the Foundation that all disclosures under this Agreement with respect to Technology are made for the exclusive and limited purpose of providing the Foundation with information necessary for it to assess the development potential of the Technology to which such disclosures relate. Until the Foundation has decided to pursue development of a given Technology and until the Center and the Foundation have entered into the agreements contemplated by this Agreement with respect to the assignment of ownership rights in such Technology to the Foundation, the Foundation may not under any circumstances communicate such Technology or such disclosures to any other persons except as may be necessary on a strict need-to-know basis in order to accomplish the evaluations contemplated by this Agreement, nor may the Foundation put such Technology or disclosures to any use other than as provided in this Agreement. Such limited communication is to be restricted to the maximum extent practicable and shall in all cases be restricted to persons who are subject to this Agreement or who enter into equivalent agreements to preserve the secrecy of all such disclosures and Technology. Any agreement entered into between the Center and the Foundation with respect to the conveyance of ownership rights in Technology shall contain provisions adequate to protect the continuing interest of the Center in such Technology in light of any residual or reversionary interest which the Center may retain in such Technology under such conveyance. The provisions of this paragraph and the obligations imposed hereby shall survive the termination of this Agreement for any reason whatsoever.

3. Costs and Expenses. The Foundation will pay all costs and expenses of the evaluation, patenting, licensing or other administration of transfer of each Technology Project but shall be reimbursed therefor out of royalty income from the Technology Project received by the Foundation as set forth in Section 4.

4. Royalties.

4.1 Distribution. The Foundation shall pay to the Center 62.5% of all royalty income from any Technology Project, after reimbursement of all Directly Allocable Costs (as defined in Paragraph 6 hereof). Because of the interest of the Center and the Foundation in the successful development of the Foundation during its formative years, the parties agree that full distribution to the Center of the above-stated share of
royalties with respect to each Technology project shall commence with the 1986 calendar year and shall be payable from January 1, 1986, unless an earlier date for such full distribution of royalties is mutually agreed upon. Until such date as such full distribution becomes payable, the parties agree that 20% of gross royalty income received by the Foundation with respect to each Technology Project shall be paid to the Center.

4.2 Royalty Payments and Accounts. Payments to the Center shall be made annually on a calendar year basis no later than January 31 for the immediately preceding calendar year. Such payment will be accompanied by a full accounting of the previous year's transactions. The Foundation shall keep accounts and records in sufficient detail to enable the royalties to be determined. Upon reasonable notice to the Foundation, such records shall be made available for inspection by an authorized representative of the Center at reasonable times and places to the extent reasonably necessary (i) to verify the accuracy of the annual reports and royalties paid and (ii) to perform at the Center's expense an audit thereof if requested by the Center. If any audit conducted in accordance with the preceding sentence shall have disclosed an underpayment of 10% or more from what had been represented by the Foundation to the Center, the Foundation will pay the entire cost of such audit and will promptly pay to the Center as royalties an amount equal to the difference between the amount which it paid to the Center and the amount the audit discloses it should have paid to the Center.

5. Review of Foundation Financial Circumstances. A thorough review of the financial circumstances of the Foundation will be made by representatives of the Center and of the Foundation not less often than annually. Such review may also be made at any time upon the request of the Center with reasonable notice to the Foundation. On any such occasion, the Foundation will make available to the Center any financial records the Center may request.

6. Directly Allocable Costs. The term "Directly Allocable Costs" shall mean the Foundation's out-of-pocket expenses and similar costs related to a Technology Project whenever incurred during the term of this Agreement, including without limitation the costs of obtaining patents, consulting fees paid to third parties in respect to the Technology Project, travel expenses and telephone and reproduction costs, but excluding the costs of evaluating the Technology Project pursuant to paragraph 1 hereof. It does not include any portion of general salaries, rent and overhead of the Foundation.
7. Dissolution of Foundation.

In the event the Foundation ceases to operate or takes legal steps to dissolve, the Foundation will accomplish the following prior to dissolution:

7.1 Pay to the Center all cumulative royalty income due to the Center.

7.2 Reassign to the Center all rights, title and interest in all Technology and Technology Project previously assigned to the Foundation and assign to the Center all right, title and interest in any improvements and developments derived from such Technology and Technology Project. Such reassignment to the Center shall also involve a reassignment of any and all license, royalty or other agreements related to any Technology Project.

8. Termination.

8.1 In the event that the Foundation fails in its obligations hereunder either with respect to the payment of royalties or with respect to the prompt and vigorous development of any Technology or Technology Projects assigned to it by the Center as contemplated by this Agreement, the Center may at its option and upon thirty days written notice to the Foundation, terminate this agreement either with respect to the specific Technology Project as to which such failure of payment or development has occurred, or with respect to this Agreement as a whole. Upon such termination, any and all license agreements relating to any Technology Project shall not terminate but the Center shall automatically be substituted for the Foundation as a party to such agreements and all rights and obligations of the Foundation shall thereupon automatically be assigned to and become vested in the Center, provided, however that the Foundation shall continue to receive continuing payments in the same amount as it would have retained pursuant to Paragraph 4 of this Agreement after payment to the Center thereunder. All license, royalty and other agreements with respect to any Technology Project shall expressly identify that such agreement is subject to the terms and conditions of this Agreement and may be assignable to the Center pursuant to the terms of this Agreement.

8.2 Either the Foundation or the Center may terminate this Agreement at any time upon thirty days written notice, but in no event prior to December 31, 1986, with respect to any future assignments of Technology Projects by the Center to the
Foundation. In such event, all rights and obligations hereunder with respect to Technology or Technology Projects earlier assigned to the Foundation shall, subject to Sections 8.1 and 8.3 hereof, continue in full force and effect according to their terms and shall not be affected by a termination under this Section 8.2.

8.3 This Agreement may be terminated at any time by mutual agreement.


9.1 This Agreement constitutes the entire agreement between the parties with respect to the subject matter hereof, and supersedes any prior agreements, understandings, promises and representations made by either party to the other concerning the subject matter hereof and the terms applicable hereto. This Agreement may not be amended or modified except by an instrument in writing signed by duly authorized officers or representatives of both parties hereto.

9.2 If any provision of this Agreement is, becomes or is deemed invalid, illegal or unenforceable in any jurisdiction, such provision shall be deemed amended to conform to applicable laws so as to be valid and enforceable or, if it cannot be so amended without materially altering the intention of the parties, it shall be stricken and the remainder of this Agreement shall remain in full force and effect.

9.3 This Agreement shall be governed by and construed in accordance with the laws of the State of Washington.

9.4 No waiver of any right under this Agreement shall be deemed effective unless contained in a writing signed by the party charged with such waiver, and no waiver of any right arising from any breach or failure to perform shall be deemed to be a waiver of any future such right or of any other right rising under this Agreement.
9.5 All notices, reports and other communications required under this Agreement shall be in writing and shall be deemed given when delivered in person or five days after mailing by prepaid first-class mail, addressed as follows:

Center: Executive Director
The Washington Technology Center
376 Loew Hall FH-10
University of Washington
Seattle, WA 98195

Foundation: President
Washington Research Foundation
1107 N.E. 45 TH Street
Suite 322
Seattle, WA 98105

or to such other address as either party may specify by notice to the other.

9.6 Neither this Agreement nor any right or obligation arising hereunder may be assigned by either party in whole or in part, without the prior written consent of the other party, which consent may be withheld in the absolute discretion of the other party. This Agreement shall be binding upon any assignor and, subject to the restrictions on assignment herein set forth, inure to the benefit of the successors and assigns of each of the parties hereto.

IN WITNESS WHEREOF, the parties have executed this Agreement on the date first set forth above.

THE WASHINGTON TECHNOLOGY CENTER

By:    
DR. EDWIN B. STEAR

TITLE: EXECUTIVE DIRECTOR
DATED: NOVEMBER 12, 1985

WASHINGTON RESEARCH FOUNDATION

By:    
DR. PATRICK Y. TAM

TITLE: PRESIDENT
DATED: NOVEMBER 12, 1985
December 30, 1987

Mr. Robert J. Marks, II  
Dept. of Electrical Engineering, FT-10  
Electrical Engineering Building  
University of Washington  
Seattle, WA 98195

Re: U.S. Patent Application  
Serial No: 131,012  
OPTICAL NEURAL NET MEMORY - Marks et al.  
WTC 87-6  
Our Reference: WTCC-1-3835

Dear Bob:

The following five documents are included in our files relating to the above-referenced patent application.

1. Optical Processor Architectures for a Class of Continuous Level Neural Nets
2. An Introduction to Neural Networks for Solving Combinatorial Search Problems
3. Alternating Projection Neural Networks
4. Content Addressable Memories: A Relationship Between Hopfield's Neural Net and an Iterative Matched Filter
5. An All Optical Iterative Neural Net Recall Memory

Copies of the first one or two pages of each document are enclosed for your reference.

With respect to each of these documents, please let us know if either of the following conditions applies:

1. The document was published or otherwise made available to the public in printed form more than one year prior to the application filing date, i.e., before December 10, 1986; or
2. The document was published or made available to the public in printed form prior to the application filing date (December 10, 1987), and the document includes subject matter that is pertinent to the invention and that was contributed by a non-inventor, i.e., by Judson McDonnell, J.A. Ritecy or Qwan F. Cheung.
If the first condition applies to any document, then the document is "prior art" for patent examination purposes, and must be cited to the United States Patent and Trademark Office. If the second condition applies, then a more detailed analysis will be required to determine the status of the document.

Yours very truly

CHRISTENSEN, O'CONNOR, JOHNSON & KINDNESS

By

Michael G. Toner

MGT/mrw
Enclosure

cc: Mr. Peter Odabashian
ISDL REPORT

ALTERNATING PROJECTION NEURAL NETWORKS


submitted to

IEEE Trans on CAS

Report 11587

Interactive System Design Lab
Mail Stop FT-10
University of Washington
Seattle, Washington 98195

11-5-87
AN INTRODUCTION TO NEURAL NETWORKS FOR SOLVING COMBINATORIAL SEARCH PROBLEMS

AN INTRODUCTION TO NEURAL NETWORKS FOR SOLVING COMBINATORIAL SEARCH PROBLEMS

University of Washington
Seattle, WA 98195

INTRODUCTION

Birds have a mass density greater than that of air, which motivated early 20th century inventors to construct flying machines and, ultimately, to invent the airplane. Recent advances in artificial neural networks (ANN) research is similarly motivated by the similarity between the brains of birds and mammals, and the fact that nature's neural networks work quite well.

An ANN can be loosely defined as a large, interconnected array of simple processors. The processors, or neurons, can be homogeneous or heterogeneous, and can be partitioned into layers. The neural interconnections can be wired, or otherwise improve some performance. Many special designs and architectures for ANNs have been proposed.

ANN's have intrigued researchers from diverse disciplines, containing classic papers on neural networks.

7-7-87

Interactive System Design Lab
(Mail Stop FT-10)
University of Washington
Seattle, Washington 98195
Optical Processor Architectures for a Class of Continuous Level Neural Nets

Robert J. Marks II, Les E. Atlas and Kwan F. Cheung
Interactive Systems Design Laboratory
FT-10 University of Washington, Seattle, Washington 98195

ABSTRACT

Optical processing architectures are presented for a recently proposed class of continuous level neural networks. Both the feed forward and feedback paths are optical, i.e. no electronics or phase conjugators are used.

INTRODUCTION

Optical neural network architectures have been proposed by a number of researchers [1-5]. Neural net architectures are highly redundant in a distributed manner. As a result, they are resilient to computational inexactitude.

Based on the continuous level neural network (CLNN) model in Ref. [6], we present similar architectures wherein no electronics or phase conjugation is required in the forward or feedback paths. After a review of the basic CLNN model, these architectures are discussed in detail. Potential implementation problems and their solutions are also explored.

A MEMORY EXTRAPOLATION NET

Consider a set of N continuous level linearly independent vectors of length \( L > N \): \( \{ \mathbf{z}_n \} \), \( 0 \leq n \leq N \). We form the library matrix

\[
\mathbf{F} = [\mathbf{z}_1 | \mathbf{z}_2 | \ldots | \mathbf{z}_N ]
\]

and the interconnect matrix

\[
\mathbf{T} = \mathbf{F} (\mathbf{F}^\top \mathbf{F})^{-1} \mathbf{F}^\top
\] (1)
AN ALL OPTICAL ITERATIVE
NEURAL NET RECALL MEMORY

Robert J. Marks II
Interactive Systems Design Lab
Department of Electrical Engineering
University of Washington
Seattle, WA 98195

11-3-86
CONTENT ADDRESSABLE MEMORIES:
A RELATIONSHIP BETWEEN HOPFIELD'S NEURAL NET
AND AN ITERATIVE MATCHED FILTER *

Robert J. Marks II
Les E. Atlas
Interactive Systems Design Lab
Department of Electrical Engineering
University of Washington
Seattle, Washington 98195

ABSTRACT
Hopfield's neural net content addressable memory (CAM) is shown to be algorithmically equivalent to an iterative matched filter (IMF) CAM. The IMF CAM can be implemented with fewer operations per iteration. Hopfield's CAM, however, can operate asynchronously and is highly fault tolerant. The algorithms are described in a signal space setting where, for orthogonal library elements, each iteration corresponds to two successive projections -- one onto the subspace spanned by the library elements and the other onto a vertex of a hypercube.
CONTENT ADDRESSABLE MEMORIES: A RELATIONSHIP BETWEEN HOPFIELD’S NEURAL NET AND AN ITERATIVE MATCHED FILTER
Dear Micheal:

I write in response to your letter of Dec. 30, 1987 concerning the OPTICAL NEURAL NET MEMORY PATENT. All except paper #5 was made available to the public prior to Dec. 12, 1987.

1. The paper:

   R.J. Marks II, L.E. Atlas and K.F. Cheung "Optical processor architectures for a class of continuous level networks"

was submitted for publication to Optics Letters in 1987. The paper discusses the first design of the processor described in the subject patent. Cheung's contribution was a comparative literature search to assure that our effort did not overlap published reports of other neural network implementations. He contributed neither to the algorithm development nor to the processor architecture.

2. The paper:


was also submitted for publication in 1987. It is a tutorial of other works and deals neither with the algorithm nor the processor of the subject patent.

3. The paper:


was submitted for publication on November 8, 1987. It discusses in detail the algorithm implemented by the subject processor but does not address implementation. Ritcey's contribution was analysis of the algorithm convergence properties.

4. The paper:

   R.J. Marks II and L.E. Atlas "Content addressable memories: a relationship between Hopfield's neural net and an iterative matched filter"
was submitted for publication in prior to December 1986. The paper, however, is a tutorial introduction to neural networks previously proposed by others and does not impact on our Application.

5. The paper:

"An All Optical Iterative Neural Net Recall Memory"

was submitted to the Boeing High Technology Center as an internal document in November 1986. To my knowledge, no copies were made available to the public. The paper served as the first draft for paper #1 above.

I hope this is the information you need.

Best personal regards,

Robert J. Marks II
Professor

cc: Peter Odabashian
    Les Atlas
    Seho Oh
Attached is a copy of the patent application for the optical APNN. According to the WTC, we should keep the fact that there is a patent quiet. We can, however, talk about the technology in papers and at meetings.

A period of about a year and a half is the time typically taken to process the application.

cc. (memo only) Peter Odabastian, WTC
December 10, 1987

VIA FAR WEST TAXI

Robert J. Marks, II
Dept. of Electrical Engineering, FT-10
Electrical Engineering Building
University of Washington
Seattle, WA 98195

Re: U.S. Patent Application
For: OPTICAL NEURAL NET MEMORY
Our Reference: WTCC-1-3835

Dear Bob:

Enclosed please find a final draft of the above-referenced patent application, together with an attached three page document entitled Combined Declaration and Power of Attorney in Patent Application. Also enclosed is an Assignment of the invention to the Washington Technology Center.

Please arrange for final review of the application by yourself, Mr. Atlas and Mr. Oh. If the application is satisfactory, each of you should sign and date the Combined Declaration in the spaces provided on page 3 of that document. The Combined Declaration should at all times remain attached to the patent application. On the same day that each inventor signs the Combined Declaration, each inventor must also execute the Assignment before a notary public.

Once these steps have been completed, please arrange to have the patent application, attached Combined Declaration and Assignment returned to our office for filing later today in the United States Patent and Trademark Office. In order to accomplish filing today, we should receive the above-listed documents from you no later than 4 p.m.

In a copy of this letter sent to Peter Odabashian, we have enclosed a further document entitled Verified Statement Claiming Small Entity Status - Nonprofit Organization. This document should be executed by an authorized representative of the Washington Technology Center, and then returned to us no later than 4 p.m. for filing with the application.

Yours very truly,

CHRISTENSEN, O’CONNOR, JOHNSON & KINDNESS

By
Michael G. Toner

MGT/mrw
Enclosure
cc: Peter Odabashian
December 8, 1987

VIA FAR WEST TAXI

Robert J. Marks, II
Dept. of Electrical Engineering, FT-10
Electrical Engineering Building
University of Washington
Seattle, WA 98195

Re: U.S. Patent Application
For: OPTICAL NEURAL NET MEMORY
Our Reference: WTCC-1-3835

Dear Bob:

Enclosed please find a draft of the above-referenced patent application. During your review of the application, please keep in mind that the application must provide enough information to enable a person of ordinary skill in the art to make and use the invention, and must disclose the best mode known to the inventors at this time for carrying out the invention.

When you have completed your review, please call with your comments. If you will be sending us a marked-up copy of the enclosed draft, we will arrange for a delivery service if you wish. Once we have received and incorporated your corrections, we will then place the application in final form for the signatures of all inventors, and then transmit the application to the United States Patent and Trademark Office. The application must be transmitted no later than Friday, December 11.

Yours very truly,

CHRISTENSEN, O'CONNOR,
JOHNSON & KINDNESS

By
Michael G. Toner

MGT/mrw
Enclosure

cc: Peter Odabashian, w/ encl.
To: Peter A. Odabashian  
WTC, mail stop FH-10  
From: Robert J. Marks II  
Subject: APNN Patent

I talked with Mike Toner on the phone about some further developments on the patent. We decided that the best procedure is to write this memo with a copy to Mike.

The new issues are due, in part, to my colleague Les Atlas and student Seho Oh. Both were supported to some extent by the Boeing money. I understand from Mike that, although the Patent Office does not partition contributions in per cent, such can be done by us and kept on file at the Patent Office. I suggest the following partition:

Les. E. Atlas................ 15%  
Seho Oh...................... 10%  
Robert J. Marks II......... 75%  

(Oh is not a US citizens if that matters.) I have not spoken to these men, but have little doubt that they will agree.

cc: Mr. Mike Toner  
2700 Westin Bldg.  
2001 6th Ave.  
Seattle, WA 98195
ADDENDA TO OPTICAL IMPLEMENTATIONS OF
ALTERNATING PROJECTION NEURAL NETWORKS

Reference: The APNN paper refers to:
ISDL report 11587 (submitted to IEEE Trans. CAS)

1. Hidden Layers
   The number of input-output relationships that can be stored in an APNN is equal
to the number of clamped input neurons. The number of input neurons (and thus the
capacity of the APNN) can be increased artificially by establishing a hidden layer of
neurons. (See section 6 on p.17 of the APNN paper.) The states of the hidden layer
neurons can be nearly any nonlinear combination of the states imposed on the hidden
layers. The nonlinearity from one hidden neuron to the next, however, must be different.
In the optical APNN, this is done by using arbitrary nonlinear electronics prior to the
input source array to generate the states of these hidden neurons which, in turn, are used
to intensity modulate the input light source corresponding to that hidden neuron. In
contrast to the Hopfield model, we are, in essence, placing nonlinearities prior to the
input rather than in the feedback path.

2. Binary Outputs
   If there is a single output neuron in an APNN and the neural state is known to be
either 1 or -1, then the sign of the output state is the correct result after one iteration. As
is outlined in the APNN paper (Case 1 on p.12 and remark f on pp. 22-23), by
superposition, this result can be extended to an arbitrary number of output neurons as
long as each output neuron was trained on only plus and minus ones. The implication for
the optical APNN architecture is that no feedback is required. Furthermore, the problem
with absorptive losses is no longer an issue in this case.

3. Learning
   Learning addresses the matter in which the interconnect matrix transmittance is
updated when new library vectors are to be stored in the neural network. The Gram­
Schmidt learning procedure (Section 5 on p.16 of the APNN paper and remark c on p.22)
can be directly applied to the optical APNN architectures by making the following
changes:

(a) The entire transmittance matrix must be available (i.e. $T$ instead of just $T_Q$).

(b) The source array must be extended to include those neurons whose state, in playback,
is determined by the fibers, i.e. the output neurons. Similarly, the output detector array
must be extended to include sensing those states normally associated with the floating
(input) and hidden layer neurons.

The entire input array is excited corresponding to the new library vector. The
error vector, $e$, is read by the output array. The neural interconnects are updated in
accordance to the equation on p.17 of the APNN paper. This can be done with
conventional electronics.
November 10, 1987

TO: James B. Wilson, Senior Assistant Attorney General
University of Washington, AG-50

FROM: Lynn M. Fleming, Director of Administration

Subject: Appointment of Special Assistant Attorney General for Patent Counsel

This is to request the appointment of Mr. Mike Toner of Christensen, O'Connor, Johnson, and Kindness, 2701 Westin Building, 2001 Sixth Avenue, Seattle, WA 98121 as special assistant attorney general to assist The Washington Technology Center (WTC) at the University of Washington in the application for a patent covering an invention entitled An Optical Continuous Level Neural Network by R.J. Marks, II (WTC #87-6.) The inventor is a member of the University's Department of Electrical Engineering and developed the invention through a WTC research project supported by a WTC contract. Consistent with the UW/WTC Memorandum of Agreement and the WTC Patent and Copyright Policy, the inventor is in the process of assigning his rights to this invention to The Washington Technology Center, retaining certain rights to royalties as provided by the policy. Patent coverage is deemed to be essential for the effective commercialization of this invention.

Mr. Toner will be the responsible attorney for filing and prosecution of the patent application with reimbursement of actual hourly services at the rate of $145.00 per hour. Periodic billings will be based on the hourly rate multiplied by the number of hours expended on the case plus other actual out-of-pocket expenses. The total estimated cost for this patent application is $8,000 which will be charged to the Center's Technology Transfer budget account. The appointment should be for a period of four years.

It is in the best interest of the inventors, the Center, and the state to secure the patent rights to this property as soon as possible. Accordingly, it is requested that this request be processed expeditiously, thereby authorizing the services required for early filing of the patent application.

Thank you for your assistance.

cc. R.J. Marks, II

Logo: "The Raven" ... a Northwest Coast Indian design symbolizing the raven as a bringer of knowledge. A computer chip and DNA chain are held in the raven's beak. Artist: Bill Holm.
ADMINISTRATIVE ORDER NO. 17

Effective October 18, 1985

SUBJECT: Exemption of the Washington Technology Center from the University of Washington Patent and Copyright Policies and delegation of authority to the WTC to have and administer its own Patent and Copyright Policy subject to certain conditions.


A. The Washington State Legislature, in Chapter 72, Section 11, Laws of the 1983 1st Extraordinary Session, with the concurrence of the Governor, has established The Washington Technology Center (WTC) at the University of Washington (UW) to be administered by the Board of Regents of the UW. Accordingly, unless otherwise specified, the WTC is subject to UW policies. However, the WTC Board of Directors and the UW Administration, acting under delegated authority from the UW Board of Regents, have agreed that in light of the purposes, goals, objectives and intended nature of the WTC, it should not be fully subject to UW Patent and Copyright Policies but should adopt its own Patent and Copyright Policy.

B. The WTC is exempted from UW Patent and Copyright Policies subject to certain conditions as follows:

1. the WTC may identify itself as the owner of inventions, patents and copyrights derived from WTC projects;

2. those inventions, patents and copyrights will be administered under a WTC Patent and Copyright Policy approved by the WTC Board and the UW Administration; and

3. the WTC will enter into a Technology Administration Agreement (TAA) with the Washington Research Foundation (WRF) that is identical in all substantive respects with the TAA between UW and WRF attached hereto as Exhibit A.

This Administrative Order No. 17 is pursuant to the authority cited above.

William P. Gerberding
President
1. The Washington Technology Center, hereafter referred to as the WTC, shall own all patents and copyrights arising from WTC sponsored research and technology development programs and projects.

2. The WTC shall negotiate all patent and copyright agreements and licensing arrangements so as to maximize technology transfer for the benefit of the economic development of the State of Washington.

3. Negotiations of patent and copyright agreements and subsequent licensing arrangements shall be the responsibility of the duly appointed individual in charge of the WTC Office at the appropriate participating university in accordance with WTC policies and procedures.

4. The WTC shall develop a Patent and Copyright Policy which will form the basis for negotiation of specific agreements on patents, copyrights, licensing, and distribution of royalty income with each of the participating universities.

5. The WTC shall negotiate up-front patent and copyright agreements, including licensing provisions, with all participating industrial sponsors of WTC programs and projects.

6. The WTC shall negotiate individual up-front patent and copyright agreements with all Industrial Fellows and their employers.

7. All individuals participating in WTC programs and/or projects shall sign an agreement requiring them to be bound by the WTC's Patent and Copyright Policy.

8. When investigators from more than one university work on a WTC project, there shall be a specific up-front agreement among all parties covering patent and copyright issues, including negotiation of agreements with industrial supporters of the project, negotiation of licenses for any intellectual property developed, distribution of royalty income, and ownership of any patents or copyrights in the event the WTC is terminated or ceases to operate for any reason.

WTC
10/22/85
THE WASHINGTON TECHNOLOGY CENTER

Patent and Copyright Policy

1. One of the primary missions of The Washington Technology Center (hereinafter referred to as WTC) is to develop new commercializable technology through joint industry-university research and technology development programs. Patents and copyrights are important in this process to:

(a) protect the economic interests of the WTC and the inventors.

(b) protect the economic interests of the industrial participants and the licensees.

(c) provide a firm legal basis for transferring the technology.

It is recognized that the value of the technology may diminish rapidly with time. Therefore, it will often be necessary to transfer technology immediately after disclosure and prior to application for or issuance of patents and copyrights.

Further, it is recognized that it will also be necessary to transfer technology without applying for patents or copyrights in those cases where the technology is not patentable or copyrightable, or where the value of the particular patent or copyright does not justify the expense.

The purpose of this document is to set forth the specific policies adopted by the WTC to assure that these requirements and goals are met.

2. As a condition of participation in WTC research projects, all personnel participating in WTC projects agree to assign their title and rights to all inventions and copyrightable material arising in connection with such research projects to the WTC, to an agent designated by the WTC, or to a sponsor, if required under agreements governing sponsored research. Such personnel shall execute documents of assignment and do everything reasonably required to assist the assignee(s) in obtaining, protecting, and maintaining patents, copyrights or other proprietary rights.

The WTC has no vested interest in inventions or copyrightable material conceived and developed by participants entirely on their own time and without the use of WTC facilities. However, in order to clarify the inventor's or creator's
title to such inventions and/or copyrightable material and to
insure compliance with the requirements of any sponsors, all
inventions and/or copyrightable material generated during
participation in WTC programs and projects shall be reported
to the WTC for determination of the degree of WTC interest.

If the WTC, in consultation with the appropriate
participating universities, determines that it has no
interest in an invention or copyrightable material or decides
to forego the patenting, copyrighting, or other
commercialization of an invention or copyrightable material,
its shall waive its rights to the invention or copyrightable
material in writing. Upon receipt of such a waiver, and
assuming that no additional WTC or University resources will
be invested, the inventor(s) or creator(s) may file a patent
or copyright application and/or grant a license of his/her
own.

3. WTC research funded wholly or in part by an outside
sponsor is subject to this policy as modified by the
provisions of negotiated agreement(s) covering such work. It
is the general policy of the WTC to negotiate all such
agreements, including any special provisions relating to the
intellectual property, prior to initiation of the research
effort being sponsored. Participants in such sponsored
research are bound by the provisions of these agreements.

4. In general, title to any inventions and/or copyrightable
material conceived and first reduced to practice in the
course of research carried-out in the WTC with the support of
Federal agencies, industry, or other sponsors shall vest in
the WTC. In rare cases, an industrial sponsor may possess a
dominant patent or copyright position in a certain technology
area so that any patent or copyright the WTC might seek would
be of little value. For this or other such reasons, an
exception to this WTC title policy may be approved when to do
so would honor the general principles of this policy, protect
the equities involved, and satisfy the requirements of the
parties. In all cases, the granting of such exceptions must
be explicitly covered in the agreements referred to above in
Paragraph 3.

5. Interaction between the WTC and industry can take any one
or more of the following forms: grants, contracts, consortial
arrangements, equipment gifts, and appointment of industrial
fellows. Industrial firms sponsoring WTC research programs
through any one or more of these forms may be assured of at
least a non-exclusive license to inventions and copyrights
conceived and developed with their support. If necessary for
the effective development and marketing of a WTC invention or
copyright, an exclusive license may be granted for a limited
period of time if the sponsor agrees to finance the cost of
the WTC's patent or copyright application and observes due
diligence in bringing the technology involved into public use. In such cases, the patent or copyright costs may be treated as an offset against royalties payable when the invention or copyright is marketed.

Where the sponsor uses the invention or copyright entirely within its own operations, the license may be royalty-free. Where the sponsor, or a third party licensee, manufactures and sells products, services, or processes based on the invention or copyright, reasonable royalty payments to the WTC or its assignee are normally required.

In all cases involving industrial sponsorship of WTC research programs, the specific licensing rights of the sponsor(s) to any patentable and/or copyrightable technology generated in the research programs shall be explicitly covered in the up-front agreements referred to above in Paragraph 3.

6. Although the WTC reserves the right to patent and/or copyright intellectual property itself, it has designated the Washington Research Foundation as its primary patenting, copyrighting, and licensing agent. However, another comparable, mutually-acceptable patenting, copyrighting and licensing agent can be used if so desired by an individual participating university.

7. Both the inventors and/or creators and the WTC are entitled to a share of royalty income from licensed patents and/or copyrights; the WTC on the basis of salary and/or facilities support for the inventor and/or creator and the cost of patent, copyright, and licensing administration; and the inventor and/or creator on the basis of the creative activity, documenting the invention or copyright, and assisting as necessary with commercialization. To recognize creativity and to encourage prompt disclosure of potential patents and copyrights, the WTC allocates the greater share of net early royalty income to the inventor or creator. The remainder is dedicated to further research by allocating shares to the WTC and to the home colleges/departments of the inventors and/or creators as appropriate. Unless amended in an agreement with a participating university, the specific allocation shall be as follows.

After deducting 15% for administrative services, net royalty income received from WTC inventions and/or copyrights handled by an outside agency is distributed as follows:
Patent and Copyright Policy

<table>
<thead>
<tr>
<th>Cumulative Net Income</th>
<th>Inventor/Creator</th>
<th>Inventor's University Dept./College</th>
<th>WTC Research Fund</th>
</tr>
</thead>
<tbody>
<tr>
<td>First $10,000</td>
<td>100%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>$10,000-$40,000</td>
<td>50%</td>
<td>25%</td>
<td>25%</td>
</tr>
<tr>
<td>Above $40,000</td>
<td>30%</td>
<td>20%</td>
<td>50%</td>
</tr>
</tbody>
</table>

In the event that an invention and/or copyright is administered directly by the WTC, the direct costs of obtaining and maintaining the patent(s) and/or copyright(s) must be recovered in addition to the 15% service fee before distribution of royalty income begins under the above formula.

The royalty derived WTC Research Fund shall be used to promote additional research in areas identified for emphasis by the WTC.

When a proposed WTC program or project involves more than one university, it is the general policy of the WTC to negotiate an up-front agreement with the participating universities covering patent and copyright issues. Including negotiation of agreements with industrial supporters of the project, negotiation of licenses for any intellectual property developed and distribution of royalty income and ownership of any patents and copyrights in the event the WTC is terminated or ceases to operate for any reason.

8. As a public institution, the WTC should undertake sponsored research under conditions which permit timely publication of the research results. However, the WTC reserves the right to defer publication for a reasonable period of time during which the WTC and any sponsor(s) review the feasibility and desirability of patent and/or copyright protection of any intellectual property described in the proposed publication. Likewise, through consultation with appropriate university officials, graduate student theses or dissertations containing invention details may be withheld from the Library shelves for a limited period while this evaluation process is conducted.

Some research agreements may involve WTC access to a sponsor's proprietary data. In all such cases, a clause defining the conditions under which such data will be identified, accepted, used, and controlled shall be included in the up-front agreement referred to in Paragraph 3. or in an amendment thereto. (Where the work is related to a thesis, students must be able to participate in such research in a meaningful way without access to such proprietary data).
When publication of research results based on use of such proprietary data is contemplated, the WTC will agree to provide the sponsor with advance copy of any proposed publication prior to submission for publication to allow the sponsor an opportunity to identify any inadvertent disclosure of its proprietary data.

9. Consultation with commercial enterprises by WTC technical experts can be of significant benefit to the WTC, the employee, the commercial entity and the general public. However, such involvements include the potential for conflicts of interest, for the inhibition of the free exchange of information, and for interference with the experts' allegiance to the WTC and to their university if they also have university affiliations. In order to minimize the potential for such conflicts and as a condition for continued involvement in WTC research projects, all proposed consulting arrangements by WTC staff must be approved by the Executive Director of the WTC, in addition to approval by the appropriate authorities in their respective universities.

Invention clauses in any such consulting agreements must be consistent with the policy of the WTC, with WTC commitments under sponsored research agreements, and, where the consultant is employed by a university, with the policies of that university. Questions concerning potential conflicts should be referred to the Executive Director or Associate Director of the WTC through appropriate university authorities.

10. In the event that the WTC is terminated or ceases to operate for whatever reason, its ownership of inventions, patents and copyrights, whether administered directly by itself or assigned to WRF or another agent, shall revert to the university at which the research leading to the invention, patent or copyright was carried out in accordance with specific agreements when more than one university is involved.

11. The Technology Transfer Committee of the WTC's Board of Directors is responsible for oversight of the WTC Patent and Copyright Policy.

WTC
10/23/85
AGREEMENT

AGREEMENT made as of November 12, 1985 between the Washington Research Foundation (the "Foundation") and the Washington Technology Center (the "Center").

RECITALS

The Foundation has been formed to stimulate productive commercial applications of inventions and other technology discovered and developed at the Center as well as other research institutions in the State of Washington. The Center and the Foundation wish to provide for the disclosure to the Foundation of certain technology (the "Technology"), which may presently or hereafter be owned by the Center, for the purpose of development and management of such Technology by the Foundation, including licensing and marketing of such Technology, the pursuit of patent applications, and the development of commercial applications for such Technology.

AGREEMENTS

1. Submission and Evaluation of Technology. The Center may from time to time deliver to the Foundation, at the Center's sole discretion, disclosures of Technology (each such disclosure referred to herein as a "Technology Project"), and the Foundation agrees to evaluate each Technology Project expeditiously. If in the Foundation's judgment the Technology has significant commercial potential, the Foundation will use its best efforts to introduce the Technology Project into commercial use and to secure royalties or other compensation therefrom as it deems appropriate. If the Foundation decides not to pursue the development of the Technology Project, it will so inform the Center in writing no later than ninety (90) days after initial receipt by the Foundation of the Center's disclosure of the Technology Project and, with such notice, shall return to the Center all materials embodying, reflecting or describing the Technology Project. If the Foundation accepts the Technology Project for commercialization, the Foundation will promptly notify the Center of such acceptance in writing. Upon such notification, the Center will assign to the Foundation all rights of the Center in such Technology Project and will execute such instruments as may be necessary to secure the ownership, right, title and interest in the Foundation of such Technology Project, subject to the provisions of this Agreement. The Foundation will thereafter, with due diligence, undertake the commercialization of the Technology Project.
2. Confidentiality. All disclosures made by the Center to the Foundation with respect to Technology shall be treated by the Foundation as confidential in their entirety. It is understood by the Foundation that all disclosures under this Agreement with respect to Technology are made for the exclusive and limited purpose of providing the Foundation with information necessary for it to assess the development potential of the Technology to which such disclosures relate. Until the Foundation has decided to pursue development of a given Technology and until the Center and the Foundation have entered into the agreements contemplated by this Agreement with respect to the assignment of ownership rights in such Technology to the Foundation, the Foundation may not under any circumstances communicate such Technology or such disclosures to any other persons except as may be necessary on a strict need-to-know basis in order to accomplish the evaluations contemplated by this Agreement, nor may the Foundation put such Technology or disclosures to any use other than as provided in this Agreement. Such limited communication is to be restricted to the maximum extent practicable and shall in all cases be restricted to persons who are subject to this Agreement or who enter into equivalent agreements to preserve the secrecy of all such disclosures and Technology. Any agreement entered into between the Center and the Foundation with respect to the conveyance of ownership rights in Technology shall contain provisions adequate to protect the continuing interest of the Center in such Technology in light of any residual or reversionary interest which the Center may retain in such Technology under such conveyance. The provisions of this paragraph and the obligations imposed hereby shall survive the termination of this Agreement for any reason whatsoever.

3. Costs and Expenses. The Foundation will pay all costs and expenses of the evaluation, patenting, licensing or other administration of transfer of each Technology Project but shall be reimbursed therefor out of royalty income from the Technology Project received by the Foundation as set forth in Section 4.

4. Royalties.

4.1 Distribution. The Foundation shall pay to the Center 62.5% of all royalty income from any Technology Project, after reimbursement of all Directly Allocable Costs (as defined in Paragraph 6 hereof). Because of the interest of the Center and the Foundation in the successful development of the Foundation during its formative years, the parties agree that full distribution to the Center of the above-stated share of
royalties with respect to each Technology project shall commence with the 1986 calendar year and shall be payable from January 1, 1986, unless an earlier date for such full distribution of royalties is mutually agreed upon. Until such date as such full distribution becomes payable, the parties agree that 20% of gross royalty income received by the Foundation with respect to each Technology Project shall be paid to the Center.

4.2 Royalty Payments and Accounts. Payments to the Center shall be made annually on a calendar year basis no later than January 31 for the immediately preceding calendar year. Such payment will be accompanied by a full accounting of the previous year's transactions. The Foundation shall keep accounts and records in sufficient detail to enable the royalties to be determined. Upom reasonable notice to the Foundation, such records shall be made available for inspection by an authorized representative of the Center at reasonable times and places to the extent reasonably necessary (i) to verify the accuracy of the annual reports and royalties paid and (ii) to perform at the Center's expense an audit thereof if requested by the Center. If any audit conducted in accordance with the preceding sentence shall have disclosed an underpayment of 10% or more from what had been represented by the Foundation to the Center, the Foundation will pay for the entire cost of such audit and will promptly pay to the Center as royalties an amount equal to the difference between the amount which it paid to the Center and the amount the audit discloses it should have paid to the Center.

5. Review of Foundation Financial Circumstances. A thorough review of the financial circumstances of the Foundation will be made by representatives of the Center and of the Foundation not less often than annually. Such review may also be made at any time upon the request of the Center with reasonable notice to the Foundation. On any such occasion, the Foundation will make available to the Center any financial records the Center may request.

6. Directly Allocable Costs. The term "Directly Allocable Costs" shall mean the Foundation's out-of-pocket expenses and similar costs related to a Technology Project whenever incurred during the term of this Agreement, including without limitation the costs of obtaining patents, consulting fees paid to third parties in respect to the Technology Project, travel expenses and telephone and reproduction costs, but excluding the costs of evaluating the Technology Project pursuant to paragraph 1 hereof. It does not include any portion of general salaries, rent and overhead of the Foundation.
7. Dissolution of Foundation.

In the event the Foundation ceases to operate or takes legal steps to dissolve, the Foundation will accomplish the following prior to dissolution:

7.1 Pay to the Center all cumulative royalty income due to the Center.

7.2 Reassign to the Center all rights, title and interest in all Technology and Technology Project previously assigned to the Foundation and assign to the Center all rights, title and interest in any improvements and developments derived from such Technology and Technology Project. Such reassignment to the Center shall also involve a reassignment of any and all license, royalty or other agreements related to any Technology Project.

8. Termination.

8.1 In the event that the Foundation fails in its obligations hereunder either with respect to the payment of royalties or with respect to the prompt and vigorous development of any Technology or Technology Projects assigned to it by the Center as contemplated by this Agreement, the Center may at its option and upon thirty days written notice to the Foundation, terminate this agreement either with respect to the specific Technology Project as to which such failure of payment or development has occurred, or with respect to this Agreement as a whole. Upon such termination, any and all license agreements relating to any Technology Project shall not terminate but the Center shall automatically be substituted for the Foundation as a party to such agreements and all rights and obligations of the Foundation shall thereupon automatically be assigned to and become vested in the Center, provided, however that the Foundation shall continue to receive continuing payments in the same amount as it would have retained pursuant to Paragraph 4 of this Agreement after payment to the Center thereunder. All license, royalty and other agreements with respect to any Technology Project shall expressly identify that such agreement is subject to the terms and conditions of this Agreement and may be assignable to the Center pursuant to the terms of this Agreement.

8.2 Either the Foundation or the Center may terminate this Agreement at any time upon thirty days written notice, but in no event prior to December 31, 1986, with respect to any future assignments of Technology Projects by the Center to the
Foundation. In such event, all rights and obligations hereunder with respect to Technology or Technology Projects earlier assigned to the Foundation shall, subject to Sections 8.1 and 8.3 hereof, continue in full force and effect according to their terms and shall not be affected by a termination under this Section 8.2.

8.3 This Agreement may be terminated at any time by mutual agreement.


9.1 This Agreement constitutes the entire agreement between the parties with respect to the subject matter hereof, and supersedes any prior agreements, understandings, promises and representations made by either party to the other concerning the subject matter hereof and the terms applicable hereto. This Agreement may not be amended or modified except by an instrument in writing signed by duly authorized officers or representatives of both parties hereto.

9.2 If any provision of this Agreement is, becomes or is deemed invalid, illegal or unenforceable in any jurisdiction, such provision shall be deemed amended to conform to applicable laws so as to be valid and enforceable or, if it cannot be so amended without materially altering the intention of the parties, it shall be stricken and the remainder of this Agreement shall remain in full force and effect.

9.3 This Agreement shall be governed by and construed in accordance with the laws of the State of Washington.

9.4 No waiver of any right under this Agreement shall be deemed effective unless contained in a writing signed by the party charged with such waiver, and no waiver of any right arising from any breach or failure to perform shall be deemed to be a waiver of any future such right or of any other right rising under this Agreement.
9.5 All notices, reports and other communications required under this Agreement shall be in writing and shall be deemed given when delivered in person or five days after mailing by prepaid first-class mail, addressed as follows:

Center: Executive Director
The Washington Technology Center
376 Loew Hall PH-10
University of Washington
Seattle, WA 98195

Foundation: President
Washington Research Foundation
1107 N.E. 45 TH Street
Suite 322
Seattle, WA 98105

or to such other address as either party may specify by notice to the other.

9.6 Neither this Agreement nor any right or obligation arising hereunder may be assigned by either party in whole or in part, without the prior written consent of the other party, which consent may be withheld in the absolute discretion of the other party. This Agreement shall be binding upon any assignor and, subject to the restrictions on assignment herein set forth, inure to the benefit of the successors and assigns of each of the parties hereto.

IN WITNESS WHEREOF, the parties have executed this Agreement on the date first set forth above.

THE WASHINGTON TECHNOLOGY CENTER

By: [Signature]
DR. EDWIN B. STEAR

TITLE: EXECUTIVE DIRECTOR
DATED: NOVEMBER 12, 1985

WASHINGTON RESEARCH FOUNDATION

By: [Signature]
DR. PATRICK Y. TAM

TITLE: PRESIDENT
DATED: NOVEMBER 12, 1985
5. List all reports, abstracts, papers, theses or patent applications which have been or are planned to be submitted by the inventor(s) describing the invention. Give dates of submission and actual or anticipated publication dates. Attach documents, if available. These documents may be used in part to respond to Section 2.

6. List any other known references, patents, patent applications or other publications pertinent to this invention. Attach copies, if available. These documents may also be used in part to respond to Section 2.

7. Describe and date any sale or public use of the invention in the United States. Specify if the use was operational, or for testing purposes, and if there was any effort or intent to maintain invention secrecy after operational use began.

8. List all co-inventors (any individuals who conceived an essential feature of the invention, either independently or jointly with others, during the evolution of the invention). In the event a patent application is filed, inventorship will be verified by the patent attorney.

9. Arrange for two technically qualified witnesses to read and sign this document verifying that they have understood the invention that is disclosed.

Submit the completed Disclosure together with the Transmittal form to Dr. Edwin B. Stear, Executive Director, Washington Technology Center, University of Washington, Mail Stop FH-10, Seattle, Washington 98195. Generally it will then be forwarded to the Washington Research Foundation (or another agent) for evaluation of patentability and commercial potential.

For further information, contact The Washington Technology Center, (206) 545-1920.
This invention Disclosure is an important legal document and should be completed carefully. Please refer to the attached instructions.

1. Title of Invention
   **An Optical Continuous Level Neural Network**

2. Brief Description
   A library of continuous level object vectors is stored in two dimensions on an optical transmittance. When a portion of a library vector is input into the optical processor by an array of point light sources, the remainder of the vector is iteratively recovered at lightspeed.

3. Funding Source(s)
   Boeing High Technology Center

4. Invention History

<table>
<thead>
<tr>
<th></th>
<th>Date</th>
<th>Location and Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Initial Idea</td>
<td>Oct '86</td>
</tr>
<tr>
<td>B</td>
<td>First description of complete invention, oral or written</td>
<td>Nov '86</td>
</tr>
<tr>
<td>C</td>
<td>Invention development records, notes, drawings (evidence of diligence)</td>
<td>Feb '87</td>
</tr>
<tr>
<td>D</td>
<td>First successful demonstration, if any (first actual reduction to practice)</td>
<td>none</td>
</tr>
<tr>
<td>E</td>
<td>First publication with full description of invention (may be barred patent)</td>
<td>12/12/86</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2/10/87</td>
</tr>
<tr>
<td>F</td>
<td>First verbal description to others</td>
<td>10/86</td>
</tr>
</tbody>
</table>

5. List all reports, abstracts, papers, theses or patent applications related to the inventions which have been published or are planned to be submitted by the Inventor(s). Include copies if available.

*See attached reference list
6. List any other references, patents, patent applications or other publications which may be pertinent to the invention. Include copies if available.

See attached reference list

7. Describe and date any sale or public use of the invention in the United States.

none

8. Inventor or Co-inventors

Robert J. Marks II, Assoc. Prof

<table>
<thead>
<tr>
<th>Signature</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4/27/87</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Signature</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name (Print)</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Robert J. Marks II</td>
<td>Assoc. Prof</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name (Print)</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>UWEE Dept, FT-10, 98195</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Telephone</th>
</tr>
</thead>
<tbody>
<tr>
<td>(206) 543-6990</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Telephone</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

9. Invention disclosed to and understood by (two witnesses required):

<table>
<thead>
<tr>
<th>Signature</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Signature</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name (Print)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

Submit completed Disclosure to the Washington Technology Center, University of Washington, 376 Loew Hall, M/S FE-10, Seattle, WA 98195.

Date Received:

Washington Technology Center
THE WASHINGTON TECHNOLOGY CENTER

Form to Transmit Invention Disclosure
(For WTC Internal Use Only)

Instructions

Complete this form and the attached Invention Disclosure form and forward to The Washington Technology Center via WTC Program Director, Department Chairperson, and Dean of School/College for approval. If more than one Department is involved, obtain signatures from all Chairpersons and Deans (or their designate).

To: Washington Technology Center
Date:____________________
Loew Hall 376, FH-10

From:

Inventor Name    Title    Department    Mail Stop

Inventor Name    Title    Department    Mail Stop

Inventor Name    Title    Department    Mail Stop

Inventor Name    Title    Department    Mail Stop

Re: Invention entitled:________________________________________

Verified and Approved:_______________________________________

WTC Program Director
Date:____________________

Concurrence:

Dean of the School/College
Date:____________________

Accepted:

Department Chairperson
Date:____________________

Edwin B. Stear, Executive Dir.
WASHINGTON TECHNOLOGY CENTER
Date:____________________
REFERENCES (and further comments)

1. R.J. Marks II "A Class of Continuous Level Associative Memory Neural Nets" to appear in the 15 May issue of Applied Optics. (this paper contains the description of the algorithm performed by the processor).

2. R.J. Marks II "An All Optical Iterative Neural Net Recall Memory" (this paper, sent to the BHTC, was an internal document. It first presents optical feedback in an optical neural net architecture. Others have used (slow) feedback electronics).

3. R.J. Marks II "A class of continuous level neural nets and their optical implementation" (these are copies of the slides used at the BHTC seminar on 12-12-86).

4. R.J. Marks II "Optical architectures for a continuous level neural net" (to date, this has been an internal report but will soon be submitted for publication to Applied Optics. The use of optical switches is first suggested here).

5. R.J. Marks II "A continuous level neural net and its optical implementation" (copies of the slides used at a 2-10-87 U.W. seminar. Optical switches were included in the proposed architecture).


Other literature:


8. "Optics and Neural Nets" Computer Design, March '87. (a similar but more recent paper).

9. Psaltis and Farhat, Optics Letters 10, Feb. '85. (the first journal paper on optical neural nets. As with other designs, slow electronics is used in the feedback path).

A Class of Continuous Level Associative Memory Neural Nets

Robert J. Marks II
Interactive Systems Design Lab
University of Washington, FT-10
Seattle, WA 98195

to appear in Applied Optics
ABSTRACT

A neural net capable of restoring continuous level library vectors from memory is considered. As with Hopfield's neural net content addressable memory, the vectors in the memory library are used to program the neural interconnects. Given a portion of one of the library vectors, the net extrapolates the remainder. Necessary and sufficient conditions for convergence are stated. Effects of processor inexactitude and net faults are discussed. A more efficient computational technique for performing the memory extrapolation (at the cost of fault tolerance), is derived. The special case of table-look-up memories is addressed specifically.
INTRODUCTION

Hopfield's neural net content addressable memory (CAM) [1] has stirred great interest in the signal processing community. The net has been implemented both optically [2-5] and electronically [6]. For optical implementation, intensive neural interconnects are possible since light paths can cross without interference. Planar VLSI implementations, on the other hand, are restricted to nearest neighbor interconnects. The interconnects in Hopfield's CAM are programmed by a set of binary library vectors. Given a noisy subset of one of the library vectors, the neural net ideally converges to the library vector closest to the initialization. The net can operate asynchronously or synchronously. It is also tolerant of both lumped and distributed faults [3,6]. Thus, analog optical processor inexactitude is of less significance than usual.

The neural net introduced in this paper allows for library vectors with continuous elements. The interconnects are determined analogous to Hopfield's recipe. The net can also operate asynchronously and is fault tolerant. It differs from Hopfield's in that the initially known neural states are imposed on the net each iteration. That is, the known states act as the net stimulus and the remaining nodes catalog the response. A human memory analogy is our ability to recall a well known painting by continuously viewing only a portion of it.

After a brief introduction to the mathematics of the neural net, we specifically define the extrapolation neural net. Borrowing from some recent results in iterative signal recovery and synthesis [7-11], important insights into the net's performance are generated. These include sufficient conditions for convergence to the proper library vector and
effects of known state perturbations. A short section on fault tolerance contains empirical evidence that the net still works "well" for both quantized and deleted interconnects. A table look-up net is one where the same P nodes are always used as the net stimulus. Neural net architectures for these specific memory extrapolation problems are presented. Some final remarks tying the net's operation to some other well known iterative algorithms are made in the conclusions.

PRELIMINARIES

Consider a neural net of L nodes. The transmission from the kth to the ith node is t_{ik}. We will assume a symmetric net (t_{ki} = t_{ik}) and will allow for autointerconnects (t_{kk} \neq 0). The state, s_k, of the kth node, will be assumed to be a function of the sum of its inputs. For synchronous operation (i.e., all delays between node pairs are identical), we have at time M

\[ \hat{s}_M = T \hat{s}_M \]  \hspace{1cm} (1)

where \( \hat{s}_M \) is a vector of the L neural states at time M, \( \hat{i}_M \) is the vector of the L input sums at time M, and T is the matrix of \( t_{ik} \)'s. Let N denote the node operator that determines the next set of states from the input sum:

\[ \hat{s}_{M+1} = N \hat{i}_M \]  \hspace{1cm} (2)

Since the state of the kth node depends only on its input sum, N must be a pointwise operator. That is, the kth element of \( \hat{s}_{M+1} \) depends only on the kth element of \( \hat{i}_M \).

Substituting (1) into (2) gives the state iteration equation:

\[ \hat{s}_{M+1} = N T \hat{s}_M \]  \hspace{1cm} (3)

We illustrate with two short examples, saving our memory extrapolation net for a more detailed treatment.
Solving Simultaneous Equations

Consider the L linear equations

\[ \hat{g} = K \hat{f} \]

Given \( \hat{g} \) and \( K \), we wish to find \( \hat{f} \). Design a neural net with

\[ I = I - K \]

and let the neural operator be defined for an arbitrary vector \( \hat{i} \), by (see Fig. 1a)

\[ N \hat{i} = \hat{i} + \hat{g} \]

Thus, the \( k \)th node adds \( g_k \) to the sum of the node's inputs. Then with initialization \( \hat{s}_0 = \hat{g} \), (3) can be inductively shown to be equivalent to

\[ \hat{s}_M = \sum_{m=0}^{M} I^m \hat{g} \]

If \( ||I|| < 1 \), we can use a generalized geometric series and write:

\[ \hat{s}_\infty = (I - I)^{-1} \hat{g} = \hat{f} \]

The net thus ideally converges to our desired result. [12]

Hopfield's Neural Net

Let \( \{ b_n \mid 1 \leq n \leq N \} \) denote \( N \) library vectors each with only \( \pm 1 \) elements.

Define the library matrix

\[ B = [ b_1 \quad b_2 \quad \ldots \quad b_N ] \]

From this, we form the interconnect matrix

\[ I = B B^T - N I \]

where the superscript \( T \) denotes transformation. (Note that \( t_{kk} = 0 \)). Let the node operator be (see Fig. 1b):

\[ N = \text{sgn} \]

where \( \text{sgn} \) performs a signum operation on each vector element. The resulting neural net is Hopfield's CAM. For an initialization, \( \hat{g} \), and \( N \ll L \), the
net's state many times will converge to the library vector closest to \( \hat{g} \) in the Hamming sense.

A MEMORY EXTRAPOLATION NET

Consider a set \( F \) of \( N \) continuous level linearly independent vectors of length \( 2 \geq N \):

\[
F = \{ \tilde{f}_n \mid 1 \leq n \leq N \}
\]

and the corresponding library matrix:

\[
F = [ \tilde{f}_1 : \tilde{f}_2 : \ldots : \tilde{f}_N ]
\]

We form a neural net with interconnects\(^*\)[5]

\[
T = F (F^T F)^{-1} F^T
\]

(4)

Given a portion of one of the library vectors, a memory extrapolator, using the library, will reconstruct the remainder of that vector. For our net, we will divide the nodes into two sets: one in which states are known and the remainder, in which the states are unknown. This node partition may change from application to application. That is, any node may be used to stimulate or to respond. Without loss of generality, assume that states 1 through \( P \) (corresponding to the first \( P \) elements in some given \( \tilde{f} \in F \) are known for a given application. Define the node operator by

\[
\begin{bmatrix}
1_1 & 1_2 & \cdots & 1_P & 1_{P+1} & \cdots & 1_L \\
\end{bmatrix}^T
\]

\[
= [ \delta_1 \delta_2 \cdots \delta_P : \delta_{P+1} \cdots \delta_L ]^T
\]

(5)

where \( \delta_k \) is the \( k \)th element of \( \delta \) (Fig. 1c). That is, for \( 1 \leq k \leq P \), the node state is kept at \( \delta_k \). Otherwise, the node state is the input sum. The \( P \)

\* If \( F \) is not full rank, then we use

\[
T = F^* (F^* F^*)^{-1} F^* T
\]

where \( F^* \) is a full rank matrix obtained from discarding appropriate redundant columns from \( F \).
known states thus act as the input or stimulus to the net and the remaining steady state node states are the response.

In summary, the algorithm is this:

1. Initialize with all unknown states set to zero. The known states are equated to the known portion of the library vector.
2. Multiply the state vector by I in (4).
3. Replace states 1 through P with their known values.
4. Go to step 2 and repeat.

In many cases of interest, we claim that this iterative procedure will converge to the desired library vector. The uniqueness of convergence to the proper library element is addressed in the next section.

PERFORMANCE ANALYSIS

In this section, we derive important convergence properties of the memory extrapolation net and analyze the effects of input uncertainty on the net's performance. Some empirical results on the net's fault tolerance are also discussed.

Insight into the net's performance is gained by viewing the corresponding iterative algorithm in an L dimensional Hilbert space, H. Consider first, the N dimensional subspace**, T, spanned by the N library vectors (i.e., T is the closure of F). The matrix I in (4) (orthogonally) projects any vector onto that subspace [130]. That is, for any $\tilde{h} \in H$,$$
\inf_{\tilde{f} \in T} \| \tilde{h} - \tilde{f} \| = \| \tilde{h} - \Pi \tilde{h} \|$$

---

*If convergence is unique, any initialization will converge to the correct result.

**Also called a closed linear manifold.
where \( \| u \| ^2 = ^u T a \). Specifically, note that \( T^2 = T \), \( T F = F \) and that, for any element \( v \) orthogonal to \( T \), \( T v = 0 \) where \( 0 \) is the zero vector.

To similarly analyze the \( N \) operator in (5), we adopt the vector partitioning notation

\[
\begin{bmatrix}
v^p \\
\vdash v_q
\end{bmatrix}
\]

where \( v^p \) is a \( P \) and \( v_q \) is a \( Q \times P \) dimensional vector. Then, for example, the zero vector can be written as \( \vdash 0 = [ \vdash v^p : \vdash v_q ]^T \) and (5) becomes

\[
N \vdash v = [ \vdash v^p : \vdash v_q ]^T
\]

Note that the operator

\[
\leq \vdash v = [ \vdash v^p : \vdash v_q ]^T
\]

(orthogonally) projects \( \vdash v \) onto the \( Q \) dimensional subspace, \( S \), spanned by the unit vectors

\[
\vdash v_q = [ \vdash v^p : \vdash \delta_k ]^T, \quad 1 \leq k \leq Q
\]

where the vector \( \vdash \delta_k \) is 1 in its \( k \)th position and is otherwise zero. Thus, our operator

\[
N \vdash v = [ \vdash v^p : \vdash v_q ]^T + S \vdash v
\]

projects \( \vdash v \) onto the linear variety, \( N \), which is the translation of \( S \) by the vector \([ \vdash v^p : \vdash v_q ]^T \).

**Algorithm Convergence**

As illustrated in Fig. 2, by alternately projecting between the subspace \( T \) and linear variety \( N \), one expects convergence to a common point to both [6]. Of principal concern is whether our net's iteration:

\[
\vdash v_{n+1} = N T \vdash v_n
\]

will converge to \( \vdash \delta \in F \). A sufficient condition for unique convergence is that

\[
P \geq N
\]
and the matrix

\[ F_p = [\hat{\tau}_1 p \; \hat{\tau}_2 p \; \ldots \; \hat{\tau}_m p] \]  

is full rank.

Proof: A fundamental contribution of Youla and Webb [9] states that alternating projections between two (or more) convex sets* converge to a point common to both (all) sets. Since both \( N \) (a linear variety) and \( T \) (a subspace) are convex, the theorem is applicable here. Furthermore, since both of these sets are linear varieties, convergence is strong [9]. That is, there exists a vector \( \hat{h} \) in both sets (i.e., \( \hat{h} \in T \) and \( \hat{h} \in N \)) such that

\[ \lim_{n \to \infty} \| S_m - \hat{h} \| = 0 \]

Clearly, we would like to have \( \hat{h} = \hat{a} \). We can be assured of this if \( T \) and \( N \) intersect only at a single point. Let's explore this notion. If \( \hat{h} \in T \), then there exists an \( N \) dimensional vector \( \hat{a} \), such that

\[ \hat{h} = F \hat{a} \]

Similarly, if \( \hat{h} \in N \), then \( \hat{h}_p = \hat{D}_p \). Any \( \hat{h} \) common to both sets must then satisfy

\[ F_p \hat{a} = \hat{D}_p \]  

(10)

If \( P < N \), there are a continuum of solutions. If \( P \geq N \), there is at least one solution. If \( D = \hat{f}_m \) the solution is:

\[ \hat{a} = \hat{g}_m \]

A sufficient condition for this to be the unique solution is that \( F_p \) be full rank.

* A set \( C \) is convex if \( a + \lambda (1-a) b \in C \) for all \( a, b \in C \) and \( 0 \leq \lambda \leq 1 \).
A more general approach to the question of the degree of subspace intersection, in which our theorem is subsumed, is given by Youla [7-8].

Relaxation Parameters

The speed of convergence of the net iteration can be painfully slow. (Consider, for example, when the angle between $T$ and $N$ in Fig. 1 is very small.) One technique to offset this slow convergence is use of relaxation parameters [9,14-15]. Specifically, we select two constants, $\lambda_T$ and $\lambda_N$, both of which lie on the interval [0, 2] and redefine the interconnect and node operators by

$$I_r = (1 - \lambda_T) I + \lambda_T I$$

and

$$N_r = (1 - \lambda_N) + \lambda_N N$$

The autointerconnects are now

$$(t_r)_{kk} = \lambda_T (t_{kk} + 1) - 1$$

and the remaining interconnects become

$$(t_r)_{jk} = \lambda_T t_{jk} ; k \neq j$$

Effects of Input Node Operator Error

Consider the perturbed node operator $N_\delta$ defined by

$$N_\delta \hat{h} = [\hat{\Delta}_p + \hat{\Delta}_q : h_q]^T$$

where $\hat{\Delta}_p$ is a $P$ dimensional error vector corresponding to faulty library information or processor inexactitude. Define $\hat{\Delta} = [\hat{\Delta}_p : \hat{\Delta}_q]^T$. If $\hat{\Delta} \in T$, then a perturbed fixed point is clearly at $\hat{\delta} + \hat{\Delta}$. Otherwise, we ask whether the linear variety $N_\delta$ intersects $T$. If it does, then convergence will be to a common point in each set. If not, we can appeal to a result.
of Goldburg and Marks [10] who proved that iteration between two non-intersecting finite dimensional convex sets strongly converges to a cycle between two points in each set -- each, a closest point in its set to the other convex set. In either case, the fixed point of iteration is not affected by translation of the linear variety in a direction orthogonal to both sets.

Fault Tolerance

To obtain an empirical feel for the fault tolerance of the extrapolation net, we used $M=5$ orthogonal sampled sine wave vectors of length $L = 40$. Each vector had norm $\| \hat{\mathbf{m}} \| = \sqrt{20}$. In all cases, we deleted half of a library vector's elements. With only single precision computing error, the mean square error

$$e_n = \| s_n - \hat{\mathbf{m}} \|^2$$

reduced in 10 iterations from $e_0 = 10.5$ to $e_{10} = 0.3$. Quantizing each element of the $T$ matrix to

seven quantization levels yielded surprisingly similar results. Doubling the quantization interval resulted in divergence.

A number of simulations were performed wherein a percentage of the elements in $I$ were randomly set to zero. Convergence was strongly dependent upon the chosen library vector. Under the scenario above, for example, for 10% of $I$ set to zero, $e_{10}$ typically varied from 0.4 to 0.7. For 20%, $0.7 < e_{10} < 2.8$. A more exhaustive analysis of the fault tolerance is in order.

Tradeoff of Fault Tolerance with Operations per Iteration

The extrapolation net requires $L^2$ multiplications per iteration. Note, however, that
\[ R = F^T F \]
is a non-negative definite (correlation) matrix, and thus its inverse can be written as:

\[ R^{-1} = D^T \Lambda D \]

where the diagonal matrix \( \Lambda \) contains the eigenvalues of \( R^{-1} \) and \( D \) is the corresponding matrix of eigenfunctions. Therefore, (4) can be written as

\[ T = \Phi \Phi^T \]

where

\[ \Phi = F D^T \sqrt{\Lambda} \quad (11) \]
is an \( L \times N \) matrix. As was done by Marks and Atlas [16], one iteration can be performed by first, multiplying \( S \) by \( \Phi^T \) and second, multiplying this vector result by \( \Phi \). Each step costs \( NL \) multiplies and, if \( N \ll L \), a significant number of multiplies per iteration is saved using this outer product technique at, of course, the loss of fault tolerance and the neural net structure.

**TABLE LOOK-UP**

An assumption thus far is that any set of \( P \) known values in a vector \( \Phi \) can be used to drive the remaining \( Q \) nodes. Due to this generality, every node must be connected to every other node. If, on the other hand, the same \( P \) nodes are always used as inputs, then the number of interconnects can be reduced. Indeed, the states of the \( P \) input nodes are not determined by their inputs. Thus, the interconnects to these nodes can be discarded. As we shall see, such table look-up nets can be reconfigured to \( Q \ll L \) nodes. As with the extrapolation net, the number of operations per iteration can be reduced at the cost of fault tolerance.
A Table Look-Up Net

Again, without loss of generality, assume that the first P elements of \( \hat{\delta} \) are our input. Since the first P elements of \( \hat{\delta}_M \) and \( \hat{\delta} \) are the same, (1) can be written as:

\[
\hat{\delta}_M = \begin{bmatrix}
\hat{\delta}_{M,1} \\
\hat{\delta}_{M,2}
\end{bmatrix} = \begin{bmatrix}
I_3 & I_4
\end{bmatrix}
\begin{bmatrix}
\hat{\delta}_1 \\
\hat{\delta}_2
\end{bmatrix}
\]

where we have partitioned the I matrix. For the node operator in (5), we need not be concerned with \( \hat{\delta}_{M,1} \) since the nodes will transform it to \( \hat{\delta}_2 \). Thus, the \( I_1 \) and \( I_2 \) partitions have no contribution to the final result.

Such "don't care" portions in extrapolation matrices have been noted elsewhere [17]. Setting \( \hat{\delta}_{M+1,1} = \hat{\delta}_{M,1} \), the informational part of (12) is

\[
\hat{\delta}_{M+1,1} = I_3 \hat{\delta}_1 + I_4 \hat{\delta}_2
\]

where

\[
\hat{\delta}_1 = I_3 \hat{\delta}_p
\]

can be computed from the library and the memory address, \( \hat{\delta}_p \). A net for this operation using \( Q \) nodes can be formed akin to that discussed in the Preliminaries section. Our interconnect matrix is \( I_4 \).

> and the node operator is defined by

\[
\hat{N} \hat{f} = \hat{f} + \hat{g}
\]

If the sufficient criteria in (8) and (9) are applicable, then \( \hat{\delta}_{M+1} = \hat{\delta}_q \) with \( Q^2 \) multiplications per iteration. The node used in this net is that in Fig. 1a.

Outer Product Equivalent

The matrix in (11) can be partitioned as:
where \( \mathbf{q}_p \) contains the first \( p \) rows of \( \mathbf{q} \) and \( \mathbf{q}_q \) the remaining \( q \). Then

\[
\mathbf{T}_4 = \mathbf{q}_q \mathbf{q}_q^T
\]

and (13) can be written

\[
\tilde{\mathbf{s}}_{M+1,2q} = \mathbf{g} + \mathbf{q}_q \mathbf{q}_q^T \tilde{\mathbf{s}}_{M,2q}
\]

Performing the iteration in this non-net format requires \( 2NQ \) multiplications per iteration.

**FINAL REMARKS**

1. A summary of the operations per iteration for each of the four extrapolation techniques are in Table 1.

2. The analysis of the extrapolation net drew strongly from results previously derived for signal synthesis and recovery purposes [7-12,14-15]. In these cases, the equivalent of a library set was chosen either due to a design or constraint motivation rather than for memory purposes. The celebrated Papoulis-Gerchberg algorithm [7-8,12,14,17-20] (in discrete form), for example, used a similar \( N \) as ours, but chose as a "library" those vectors whose DFTs were identically zeros in specified bins. The extrapolation net performs this algorithm when the library vectors are the corresponding complimentary rows of the DFT matrix. The continuous form of the Papoulis-Gerchberg algorithm has been performed optically [12,21,23].

3. We have applied the powerful results of convex set projection in our analysis. Any net with a correspondingly convex \( N \) can be similarly analyzed. Also, two or more convex operations can be combined at a node. If, for example, we knew that the library vector's elements were between minus and plus one, then the output nodes
could perform an additional convex operation which for \( P + 1 \) is defined by

\[
s_k = \begin{cases} 
1 & \text{if } i_k > 1 \\
i_k & \text{if } |i_k| \leq 1 \\
-1 & \text{if } i_k < -1 
\end{cases}
\]

For \( 1 \leq k \leq P \), \( N \) is as before. One can view this as a projection onto a (convex) hypercube centered at the origin.

4. One advantage of the Hopfield CAM net is that a finite number of iterations can result in the exact correct answer, whereas the extrapolation net generally only gets iteratively closer and closer. A step towards a multilevel net, however, can be obtained from the extrapolation net by requiring each library vector to contain only integers. In lieu of (7), we perform the iteration

\[
\hat{s}_{M+1} = I N I \hat{s}_M
\]

where the vector operator \( I \) rounds each vector element to the nearest integer. Geometrically, \( I \) projects onto the nearest vector with all integer components. Although (14) generally converges in a finite number of iterations and gets us "close" to the desired library element, convergence can be to an element not contained in our library. Consider, for example, Fig. 3 where, as in Fig. 2, the subspaces \( T \) and \( N \) are shown. The lattice of dots denote vectors with integer components. Beginning with the \( \hat{s}_0 \) in the lower right corner, in accordance to (14), we project onto \( N \) and then onto \( T \) and finally onto the nearest lattice point. Continuing, we eventually converge to \( \hat{s}_\infty \) shown as the vertex of the steady state (\( \hat{s}_\infty, b, c \)) triangle in
Fig. 3. Although the process has converged in a finite number of iterations, the result is not our desired $\tilde{f}$. Note similar steady state triangles (e.g., $t$ in Fig. 3) exist closer to $\tilde{f}$.
ACKNOWLEDGEMENTS
The author gratefully acknowledges the support of this work by the SDIO/IST'S Ultra High Speed Computing Program administered through the U.S. Office of Naval Research in conjunction with the Optical Systems Lab at Texas Tech University and, in part, by the Boeing High Technology Center. Significant contributions to the clarification of result interpretation were made by Dziem Nguyen and Fred Holt at the Boeing High Technology Center. Also appreciated are the stimulating discussions with the author's ISDL colleagues: Les Atlas, Kwan Cheung and Jim Ritcey.
References


References


Figure and Table Captions

Figure 1: The three types of nodes used in this paper. The input into the $k^{th}$ node, $i_k$, is the sum of the contributions of all $L$ nodes through transmittances $t_{ik}$. (a) A node useful for linear equation solution and table look-up nets. (b) The node used in Hopfield CAM nets. (c) Nodes useful for our extrapolation net.

Figure 2: Illustration of the iterative convergence to the library vector. Beginning with $S_0 = [d_p; 0_q]$, we alternately orthogonally project between $T$ and $N$ as shown with the dashed lines. Note that $S_0$ is orthogonal to the subspace.

Figure 3: When rounding the states to the nearest integer, the iteration converges in a finite number of steps -- but not to the desired integer vector, $f$.

Table 1: Multiplies per iteration for four memories. For each, there are $N$ library vectors of length $L$. $P$ elements of one of these elements are used to regenerate the remaining $Q = L - P$. Each memory scheme executes the same restoration algorithm. Thus, in the absence of processor inexactitude, all perform identically.
October 13, 1986

TO: WTC/UW Principal Investigators
FROM: Edwin B. Stear
Executive Director

SUBJECT: Technology Disclosures

This memorandum, along with the enclosed materials, is intended to provide specific guidance on the handling of technology disclosures through the WTC, as well as clarify The Washington Technology Center's Patent and Copyright Policy in general.

As you know, President Gerberding in October 1985 signed Administrative Order No. 17 which exempted the WTC from UW patent and copyright policies and delegated authority to the WTC to have and administer its own Patent and Copyright Policy subject to certain conditions (see the enclosed copy). Subsequently, the WTC Board of Directors approved a WTC Patent and Copyright Policy. Although a copy of this policy was distributed to you some months ago, it is included here to provide a self-contained information packet.

To provide further background, I am enclosing copies of the WTC Principles governing patent and copyright policies and procedures, and the Agreement between The Washington Technology Center and the Washington Research Foundation (WRF).

Finally, in accordance with the documents identified above, the enclosed technology disclosure policy is provided for your information and use in disclosing inventions related to WTC research projects. As noted in the instructions, the disclosure will generally be forwarded to the Washington Research Foundation (or other agent), at the discretion of the WTC, for evaluation of patents and commercial potential.

Please feel free to contact me if you have any questions concerning these policies or procedures.

EBS/bf

Enclosures

cc: John Rusin
Janell Douglas
John Piety

Logo: "The Raven" ... a Northwest Coast Indian design symbolizing the raven as a bringer of knowledge.
A computer chip and DNA chain are held in the raven's beak. Artist: Bill Holm.
INVENTION DISCLOSURE

Washington Technology Center

Instructions

This Invention Disclosure Form is used to report inventions and to record the circumstances under which the invention was made. The Disclosure is a legally important document; care should be taken in its preparation since it provides both the basis for determining patentability and the data for drafting a patent application.

New and potentially useful technology developed by WTC employees with WTC and/or industry grant and contract support should be reported promptly consistent with the Center's Patent and Invention Policy.

The following instructions apply to the correspondingly numbered sections of the form.

1. Use a brief title, sufficiently descriptive to aid in identifying the invention.

2. Provide a brief description, pointing out novel features of the invention. Attach additional material which covers the following points:
   a. General purpose
   b. Technical description with references to drawings, schematics, sketches, flow diagrams, etc., as appropriate
   c. Advantages and improvements over existing methods, devices or materials, and features believed to be new
   d. Possible variations and modifications
   e. State-of-the-art prior to invention, and similar or related patents (if known)

3. List all sources of support for the research which led to the conception or actual reduction to practice of the invention. Include WTC personnel, funds or materials as well as those of University or outside agencies, organizations and companies.

4. The invention history is legally important in determining the priority of invention and/or legal "bars" to patenting. The United States Patent law allows submission of a patent application up to one year after an enabling disclosure of the technology. Most foreign countries require a patent application prior to any enabling disclosure (an oral presentation or publication such as an article, abstract or theses, or other communication which would allow a knowledgeable person to duplicate the work).
As you can see by the attached memo, I have sent our work to the right channels. If you have an additional paper, please send it to Fleming with a cover memo and a courtesy copy to Graham, Dziem and me.

Thanks!
ELECTRICAL ENGINEERING, FT-10  

DATE: February 26, 1987 

TO: Lynn Fleming  

FROM: Robert J. Marks II 

SUBJECT: Publications corresponding to "Analysis of Neural Nets" sponsored by the Boeing HTC. 

Enclosed are four papers generated totally or in part under the support of the subject grant:


2. Report on "One Step Convergence of Hopfield's Neural Net CAM", by Lawrence Wong, is a senior project final report prepared in the fall of 1986. 


4. "Optical Architectures for a Continuous Level Neural Net", contains material which is to be given at the IEEE Conference on Neural Information Processing Systems-Natural and Synthetic, this year at Boulder, Colorado. 

Please contact me at 543-6990 if you have any questions. 

cc: R. Graham  
    D. Nguyen
OPTICAL ARCHITECTURES FOR A CONTINUOUS LEVEL NEURAL NET

Robert J. Marks II
ISDL
2/20/87
INTRODUCTION

We propose optical processing architectures for implementing a recently proposed class of continuous level neural net (CLNN) associative memories [1]. As with other optical neural net architectures, the processors perform iteratively. They have the advantage, however, of requiring no electronic or phase conjugating optics in the feedback path. Thus, the neural net's stable states are iteratively generated at light speed. Furthermore, the processor components are all commercially available off-the-shelf items.

PRELIMINARIES

For purposes of continuity and establishing notation, we briefly review the CLNN. A more complete discussion can be found in Ref. [1].

In a system of L neurons, we store a total of N continuous level vectors \( [f_n] \subseteq \{n \leq N \} \). Define the library matrix

\[
E = [f_1; f_2; \ldots; f_N]
\]

and the interconnect matrix

\[
I = E (E^T E)^{-1} E^T
\]

Thus, \( t_{i,j} \) is the interconnect value between the \( i^{th} \) and the \( j^{th} \) neuron.

Assume that \( P \) neural states are known for some library vector \( f \).

In general, any \( P \) of the neural states can be known. For notational
convenience and without loss of generality, assume that the first P states of f are known. Accordingly, we adopt the following partitioning notation:

\[ f = [f_P; f_O]^T \]

where \( f_P \) is the vector of the first P elements of f and \( f_O \) is the remaining \( Q = L - P \). The operation performed at the neural nodes can now be expressed as:

\[ \eta_i = \eta [i_P; i_O]^T = [f_P; i_O]^T \]

In synchronous form, the neural net iteratively performs the operation:

\[ s_{M+1} = \eta I s_M \quad (1) \]

where \( s_M \) is the L vector of neural states at time M. Thus, if a state is known, the corresponding node clamps to the known value. The remaining nodes, responding to this stimuli, have floating states that are equal to the sum of their inputs. Convergence to \( f \) is guaranteed if \( P \leq N \) and the first P rows of \( E \) form a matrix of full rank. This is true independent of the choice of the initial state vector, \( s_0 \).

Two seeming disadvantages of the CLNN with respect to Hopfield's are:
are:

(1) the relative inexactness of analog processor results and
(2) the generally infinite number of required iterations for convergence.

With regard to optical implementation, the responses to these objections, respectively, are

(1) As a function of the accuracy and dynamic range of the input and processing, the input library vectors can be restricted to a given number of discrete levels. Then, corresponding to some performance level, the processor output can be quantized accordingly.

(2) When iterations are being performed at light speed, the significance the convergence rate is reduced substantially.

A TABLE LOOK-UP NET

A table look-up net is one in which the same P nodes are always used as the net's stimulus and the remaining Q nodes iteratively converge to the desired response. Note that the iteration in (1) can be partitioned as:

\[
\begin{bmatrix}
  f_P \\
  \vdots \\
  s_{Q,M+1}
\end{bmatrix}
= 
\begin{bmatrix}
  I_P \\
  \vdots \\
  I_Q
\end{bmatrix}
\begin{bmatrix}
  f_P \\
  \vdots \\
  s_{Q,M}
\end{bmatrix}
\]

where \( I_P \) denotes the first P rows of I and \( I_Q \) is the remaining Q. Since the first P neural states are always clamped to the known values, there is no need to know \( I_P \). Indeed, an equivalent expression is:
A basic methodology for optical implementation of this iteration is illustrated in Fig. 1. The known portion of the library vector, \( f_p \), is input into the processor by an intensity modulated point source array (e.g. LED's). Multiplication by the \( I_o \) matrix is performed by a standard vector-matrix multiplication architecture [2].

(The astigmatic optics are not shown). The vector output, \( s_{o,m+1} \), is input into a the fiber bundle shown on the right. The bundle is then fed back into the input on the left hand side. This provides the \( s_{o,n} \) portion of the input vector required in (2). We are thus performing the iteration required by the table look-up net at light speed. Feedback could also be provided by mirrors.

The astute reader will have already noted three major problems with this processor:

(1) There is no provision to detect the output.

(2) There is no provision for compensating for absorbtive and other losses in the feedback loop.

(3) The \( I_o \) matrix and the input generally contain both positive and negative numbers. Incoherent optics can only add and multiply positive numbers.

Each of these problems has a straightforward solution:
(1) The output can be detected by placing a highly transmitting pellicle in the feedback path and using appropriate focusing optics. This clearly increases absorptive losses and contributes further to problem number two:

(2) If the matrix transmittance can be amplified, then we can compensate for absorptive loss. One can easily show that if \( N \ll L \), then \( t_{ij} \ll 1 \). In such scenarios, we can then "amplify" the matrix transmittance significantly and still not exceed the maximum passive transmittance value of unity.

(3) The problem of performing bipolar operations with incoherent optics has a number of solutions. One straightforward technique is to rewrite each matrix and vector as the sum of a positive and negative matrix or vector:

\[
f_p = f_p^+ + f_p^-
\]

\[
s_{a,m} = s_{a,m}^+ + s_{a,m}^-
\]

\[
I_a = I_a^+ + I_a^-
\]

The matrix \( I_a^+ \), for example, is formed by setting all of the negative elements in \( I_a \) to zero. Then (2) can be written as:

\[
s_{a,m+1}^+ = \begin{bmatrix} f_p^+ \\ \vdots \\ f_p^+ \end{bmatrix} + \begin{bmatrix} f_p^- \\ \vdots \\ f_p^- \end{bmatrix}
\]

and
The corresponding optical implementation, although somewhat more involved, requires only positive multiplications and additions and is a straightforward generalization of the architecture in Fig. 1. The positive and negative components are added electronically at the output.

\[ s^{-a,m+1} = I^+ \begin{bmatrix} f^+ \\ \cdots \\ f^+ \end{bmatrix} + I^- \begin{bmatrix} f^- \\ \cdots \\ f^- \end{bmatrix} \]

AN OPTICAL IMPLEMENTATION OF THE CLNN

We now address optical implementation of the CLNN under the condition that any P neurons can act as the net stimulus. An architecture similar to that for the table look-up net is shown in Fig. 2 for \( L = 4 \) neurons. In the figure, the middle two neural states are known and are input into the net structure by the middle two sources in the point source array. The bottom four by four transmittance represents the I matrix. Multiplication is performed as before and the output is fed into the fibers on the right. The fiber bundle is positioned so that its other end provides the input to the net in the upper left hand corner. Since all of the input from the middle two neurons should come from only the corresponding sources, the light from the middle two fibers should not be reintroduced into the system. This can be done with either an electro-optic or optic-optic toggle switch that turns off the fibers corresponding to the locations of the neurons (sources) with known states. Such switches can operate in the gigahertz range with small attenuation.
As is shown in Fig. 2, the output of the switch is input into the net and multiplies the top four by four transmittance which also corresponds to the to the $I$ matrix. This top transmittance, however, is adjusted for feedback losses as previously discussed. The contribution from the known states (sources) and the unknown states (switch outputs) are thus multiplied by their respective $I$ matrices and the superposition of their contributions are collected by the fiber bundle on the right. The iteration therefore proceeds towards convergence at light speed. The processor can be straightforwardly augmented as before to allow for the required bipolar operations.

CONCLUSIONS

Using the continuous level neural net (CLNN) algorithm developed in [1], we have proposed two corresponding optical implementations that require no electronics or phase conjugation optics in the feedback path. After a more detailed feasibility study, we propose to prototype these architectures and investigate their ultimate performance.

REFERENCES

ISDL SEMINAR

"A Continuous Level Neural Net and its Optical Implementation"

Prof. Robert J. Marks II

Tues, Feb. 10, 2:30 in Rm 108 EEB, U of W.

For more information on the seminar series, please contact:

Prof. Robert Marks
Interactive System Design Laboratory
Dept. of Electrical Engineering, FT-10
University of Washington
Seattle, WA 98195
(206) 543-6990
A CONTINUOUS LEVEL NEURAL NET AND ITS OPTICAL IMPLEMENTATION
Contents:
1. Introduction to CAM's
2. A Homogeneous Neural Net
3. The CLNN
4. Optical Implementation
5. Conclusions
Associative Memory

4 Memory Objects:

1. Robot Hand

2. Walking Candles

3. Letters

4. Scope Trace
Properties of CAM:

1. Recall from partial memory:

   ![Character Image]

2. Works better with more information:

   ![Diagram]

3. Recognize Perturbed objects

   ![Graph]

4. Error Correction

   ![Image]
5. Fault Tolerance

6-7. Works better for small libraries
- Recognize uncorrelated objects better
A Homogeneous Neural Net

L neurons.

\( S_k = \text{state of } k^{th} \text{ neuron} \)

\( \vec{S} = L \text{ vector of neural states} \)

Interconnects: \( t_{ij} \)

\( i_k = \text{sum of inputs into } k^{th} \text{ neuron} \)

\[ i_k = \sum_{i=1}^{L} t_{ik} S_i \]

\( n_k \) = operator at \( k^{th} \) node

Iteration: \( S_{k+1} = n_k \frac{\dot{i}_k}{i_k} = n_k \sum_{i} t_{ik} S_i \)

Synchronous form:

\( \vec{S}_{k+1} = \mathcal{N} \bigwedge \vec{S}_k \)
Application to CAM's

Idea: Memories

Program memory into interconnects

Two Methods:
1. Hopfield
2. CLNN (Continuous Level Neural Net)
The CLNN

Library matrix:
\[ F = [\vec{f}_1; \vec{f}_2; \ldots; \vec{f}_N]^T \]

Interconnect Matrix:
\[ I = F (F^T F)^{-1} F^T \]

Q: Why?
A: \( I \) projects onto the column space of \( F \):

\[ \vec{f}_1 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \]
\[ \vec{f}_2 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \]
**Neural Operator for CLNN**

Let \( f \in \mathcal{F} \) library.

We know \( P \) of the elements of \( \mathcal{F} \) and wish to recall the remaining \( Q = L - P \). WLOG, let the first \( P \) state be known. Then, for any vector \( \mathbf{i} \):

\[
\mathbf{n} \cdot \mathbf{i} = \mathbf{n} \left[ \begin{array}{c} \mathbf{i}_P \\ \mathbf{i}_Q \end{array} \right] = \left[ \begin{array}{c} \mathbf{f}_P \\ \mathbf{f}_Q \end{array} \right]
\]

:: Two types of neural operators

1. **Clamped:**
   
   \( k^{th} \) neuron
   
   \( s_k = f_k \)
   
   state is known to be \( f_k \)

2. **Floating:**
   
   \( s_k = \mathbf{i}_k \)
   
   state is unknown.
   
   State = \( \mathbf{i}_k \) = sum of inputs
Synchronous Interpretation:

\[
\begin{bmatrix}
\vec{r}_P \\
\vdots \\
\vec{r}_Q
\end{bmatrix}
= \begin{bmatrix}
I \\
\vec{s}_M
\end{bmatrix}
\]

Q: Does \( \vec{s}_M \overset{M=M+1}{\rightarrow} \vec{f} \)?

A: Usually
Signal Space Interpretation of the Neural Operator, $R$

- Recall: \[ R \vec{i} = \begin{bmatrix} \vec{f}_P \\ \vec{i}_Q \end{bmatrix} \]

- Consider:
  \[ S \vec{i} = \begin{bmatrix} \vec{O}_P \\ \vec{i}_Q \end{bmatrix} \]

$S$ projects $\vec{i}$ onto a $Q$ dimensional subspace.

e.g. $\vec{i} = [i_1, i_2]^T$:

\[ x_2 \]

\[ S \vec{i} = [0, i_2]^T \]

$\vec{i}$

subspace

$x_1$
Note:
\[ n \vec{i} = \begin{bmatrix} \vec{f}_P \\ \vec{0}_Q \end{bmatrix} + S \vec{i} = \begin{bmatrix} \vec{f}_P \\ \vec{0}_Q \end{bmatrix} \]

Thus, \( n \) projects onto a linear variety:

\[
\begin{bmatrix} f_1 \\ i_2 \end{bmatrix} = n \vec{i} \]

The linear variety is the subspace translated by the (orthogonal) vector \( \begin{bmatrix} \vec{f}_P \\ \vec{0}_Q \end{bmatrix} \).
Our Story So Far:

- Library matrix: \( F = [ \hat{f}_1, \hat{f}_2, \ldots, \hat{f}_N ] \)
- Interconnect Matrix: \( T = F (F^T F)^{-1} F^T \) (projects onto the space spanned by the library).
- Let \( \vec{f} \in \text{library} \). The neural operator
  \[ n = n \hat{t} = [ \vec{f}_p, \vec{t}_q ]^T \]
  projects onto the linear variety.
- Our neural net performs the operation:
  \[ \vec{S}_{M+1} = n^T T \vec{S}_M \]
 \( \vec{f} \) is in both the subspace & the linear variety.
Q: When is \( \vec{f} \) the only point of intersection?
A: When \( P \geq N \) and the first \( P \) rows of \( \mathbf{F} \) form a full rank matrix.

Q: Is this nec. for \( \vec{s}_\infty = \vec{f} \)?
A: No. If the varieties intersect in more than one point, convergence is to that intersection point closest to the initialization.

![Diagram of a linear variety intersecting a plane]  

Problem: Convergence can be slow:

Solutions:
1. Use relaxation parameters.
2. Iterate at the speed of light.
OPTICAL IMPLEMENTATION

An optical matrix-vector multiplier:

\[ \vec{b} = A \vec{a} \]

LED array

matrix transmittance, $A$

(photo detector array)

(astigmatic optics not shown).
A Symptom-Diagnosis Neural Net
(i.e. table look up)

Same P nodes always provide the input. The remaining Q are the response.

Algorithm:

\[
\begin{bmatrix}
\vec{i}_p^+ \\
\vec{i}_q^+
\end{bmatrix} =
\begin{bmatrix}
T_p \\
T_q
\end{bmatrix}
\begin{bmatrix}
\vec{f}_p \\
\vec{f}_q
\end{bmatrix}
\]

There is no need to compute \( \vec{i}_p^+ \).
\[\therefore \text{ We don't need } T_p\]
An Optical Implementation:
• Problems:
  1. Detecting Output
  2. Absorptive Losses

• Solutions:
  1. Place pellicle in feedback path.

Q: What is \((t_{ij})_{\text{max}}\)?
A: For orthogonal bipolar library:

\[
(t_{ij})_{\text{max}} = \frac{N}{L}
\]

Here, we can tolerate an absorptive loss of \(N/L\) per iteration.
Alternate Form:

\[ \vec{i}_Q^+ = \mathbf{I}_Q \begin{bmatrix} \vec{f}_P \\ \vec{i}_Q \end{bmatrix} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{I}_4 \end{bmatrix} \begin{bmatrix} \vec{f}_P \\ \vec{i}_Q \end{bmatrix} = \mathbf{I}_3 \vec{f}_P + \mathbf{I}_4 \vec{i}_Q = \vec{g} + \mathbf{I}_4 \vec{i}_Q \]

\( \mathbf{I}_4 \) is \( Q \times Q \).

Use \( Q \) neurons with interconnect matrix \( \mathbf{I}_4 \). Neural operator:

\[ \mathbf{\eta} \vec{\lambda} = \vec{\lambda} + \vec{g} \]

Note that, in steady state, \( \vec{i}_Q^+ = \vec{f}_Q \).

Thus:

\[ \vec{f}_Q = \mathbf{I}_3 \vec{f}_P + \mathbf{I}_4 \vec{f}_Q \]

or

\[ \vec{f}_Q = [\mathbf{I} - \mathbf{I}_4]^{-1} \mathbf{I}_3 \vec{f}_P \]
An Optical CLNN

\[ \hat{S}_{M+1} = \mathcal{H} \mathcal{T} \hat{S}_M \]
Future Work:

Extension to convex sets:

Application to CLNN:

For response nodes, instead of $S_k = i_k$,

Projects onto box.

Limited dynamic range accelerates convergence!
EFFORTS TO DEVELOP ARTIFICIAL NEURAL NETWORKS MODELED ON THE BRAIN'S HIGHLY FAULT-TOLENT, MASSIVELY PARALLEL COMPUTING CAPABILITY ARE QUICKLY PICKING UP SPEED. BUT RESEARCHERS TRYING TO BUILD THESE NETWORKS WITH VERY LARGE-SCALE INTEGRATED CIRCUITS ARE RUNNING INTO A SPATE OF SIGNAL-DISTRIBUTION PROBLEMS. NOW A TEAM AT THE UNIVERSITY OF PENNSYLVANIA HAS TAKEN A BIG STEP FORWARD IN THIS WORK BY TURNING TO OPTOELECTRONICS INSTEAD OF VLSI TO BUILD WHAT CHIEF RESEARCHER NAHIL H. FARHAT CALLS THE FIRST PRACTICAL ARTIFICIAL NEURAL NET.

THE PENN RESEARCHERS WERE ABLE TO AVOID THE PROBLEMS THAT CROPPED UP IN AN ARTIFICIAL NEURAL NETWORK'S PRODIGIOUS INTERCONNECTIONS WHEN IMPLEMENTED IN SILICON. THEY ACHEIVED THIS BY TAKING ADVANTAGE OF A SIMPLE PRINCIPLE OF PHYSICS: LIGHT MULTIPLEXES AND INTEGRATES THROUGH LENSES WITHOUT CROSSTALK. THE TEAM'S WORK GOES BEYOND SUCH A NEURAL NETWORK MEMORY; BY TEAMING IT WITH A HIGH-RESOLUTION IMAGING RADAR--WHICH THEY DEVELOPED--THEY CAN PRODUCE IMAGES SHOWING DETAILS AS SMALL AS 50 CM ON FULL-SIZED AIRCRAFT--THE HIGHEST RESOLUTION REPORTED IN THE UNCLASSIFIED LITERATURE.

THE PENNSYLVANIA NEURAL-NET MEMORY IS AN OPTICAL CONTENT-ADDRESSABLE ASSOCIATIVE MEMORY (CAM), WHERE THE ELEMENTS ARE SEARCHED IN PARALLEL BY THEIR CONTENT RATHER THAN BY ADDRESS. THE RADAR AND CAM WORK WITH A LIBRARY OF AIRCRAFT CHARACTERIZERS AND NEED AS LITTLE AS 10% OF THE RADAR'S FULL DATA SET TO FIND THE CLOSEST MATCH TO A CHARACTERIZER AND THEREBY SUCCESSFULLY IDENTIFY A TARGET MODEL AIRCRAFT (FIG. 1).

BASED ON RECENT LABORATORY TESTS, THE PENN RESEARCHERS BELIEVE THEIR SYSTEM SHOULD BE ABLE TO IDENTIFY AN INCOMING TARGET AIRCRAFT AT A RANGE OF A FEW HUNDRED KILOMETERS. "ITS RANGE IS LIMITED ONLY BY TRANSmitter POWER AND THAT WILL BE EXTENDED CONSIDERABLY AS EQUIPMENT IS DEVELOPED," FARHAT SAYS. IN COMMERCIAL APPLICATIONS, IMAGING RADAR OPERATING IN THE S OR X BANDS WITH A 0.5-GHZ BANDWIDTH COULD PROVE USEFUL FOR A VARIETY OF NEAR-AIRPORT TASKS, SUCH AS TELLING A PILOT IF HIS LANDING GEAR HAS BEEN DEPLOYED.

THE SYSTEM WILL NOT BE LIMITED TO INTERROGATING LARGE OBJECTS, HOWEVER. WHEN UPGRADED TO OPERATE IN THE 60-TO 100-GHZ BANDWIDTH, IT WILL BE ABLE TO DISCERN MILLIMETER-SIZED DETAIL AT A RANGE OF SEVERAL METERS THROUGH MANY OPAQUE MATERIALS. SUCH A CAPABILITY MAKES THE NONDESTRUCTIVE EVALUATION OF MICROWAVE-PENETRABLE MATERIALS A NATURAL APPLICATION, FARHAT SAYS.

Over the past few years, theorists have taken giant strides in describing how a simple neural network might process information. But attempts to implement neural nets in VLSI circuitry have been mired in the maze of complex signal-distribution and interconnection problems among the many artificial neurons. For example, scientists at AT&T Bell Laboratories who are grappling with these problems in VLSI are making progress, according to a representative, but they have yet to engineer a solution they are willing to discuss publicly.

TWO DECADES OF RESEARCH

The need for a powerful parallel processor grew out of two decades of research at Penn into imaging radar. Farhat, zeroing in on his goal of near-visual-quality images, knew that real-time data generated for nearest-neighbor image searches would overwhelm all but the most powerful serial computers. A visit to the Jet Propulsion Laboratory, Pasadena, Calif., in 1980 introduced him to the neural-network concept, which resulted in lab versions of the CAM. The CAM can pare and interpret the flood of real-time data from the radar.

Dovetailing imaging radar and optical-memory technology and refining the CAM are the tasks at hand in Penn's lab. Although Farhat will not speculate on a commercialization time line, he is confident the CAM can be transferred from the lab to an optoelectronic circuit with present-day fabrication technologies. The CAM has uses in a wide spectrum of image-identification tasks, he says, especially those plagued by substantial amounts of missing or incorrect data.

Using off-the-shelf hardware such as light-emitting diodes, magneto-optic spatial light modulators, anamorphic lenses, photodiodes, and an electronic nonlinear feedback loop, Farhat's team of graduate students is probing the limits of the CAM's associative powers. In several tests, it has identified a scale-model aircraft from 10% of the full data set used to characterize the model aircraft with the high-resolution radar. Under real-world conditions, the memory will have to deal with spurious data from target vibration and wind buffeting, but Farhat expects its fault tolerance to be equal to the task.

The memory's phenomenal fault tolerance and robustness can be traced to a binary-coded memory.

1. RADAR IMAGE. The optical neural-net memory creates an identifiable space shuttle image from limited data.
matrix that attempts to mimic the synaptic connections in a simple biological neural network. The memory matrix is computed from the Hopfield algorithm, which is the basis for all neural-network-memory implementations. In its simplest form, the algorithm creates a two-dimensional memory matrix from a library of one-dimensional binary inputs known as characterizers. For practical image-identification problems, Farhat uses 2-d characterizers that expand to 4-d memory matrices.

Given a library of binary-coded characterizers each having n bits, the algorithm begins by taking the first characterizer and computing a simple numerical relationship between each bit and the remaining n-1 bits. This yields an expression of n² bits arranged in a matrix twice the dimensions of the input characterizer’s matrix. The same operation is performed for each characterizer, with each result summed in the memory matrix’s relevant position to create a decimal expression. Each characterizer, then, is equal to a stable energy state for the memory matrix. When excited by partial information, the matrix comes to rest at or near the closest stable energy state of the input.

The precise commercial architecture for an optical CAM is still to be determined but will probably include two major functional systems—one to create the memory matrix, the other for nearest-neighbor searches.

FIVE MAJOR SUBSYSTEMS

To create and store a memory matrix from a series of library characterizers consisting of a 32-by-32-bit array of data points, an optoelectronic processor would consist of five major subsystems (Fig. 2):

- A single-chip, 32-by-32-bit array of GaAs LEDs for the display of each characterizer pattern.
- A 32-by-32-bit array of molded plastic anamorphic lenslets to multiplex the displayed patterns.
- A 32-by-32-bit photodiode array to record the output of the memory mask and integrate the result.
- A digital memory device to drive the LED display with characterizers. When the programmable mask is implemented, another memory unit will be used to store the memory matrix.
- A photoreceptor, using a photodiode to capture the image (all but one lenslet is covered at a time). The multiplexed image interacts with the mask programmed to the receptor that represents the characterizer. The result recorded by the photodiode represents the first submatrix. The same procedure is followed for each lenslet. The results from each characterizer are summed in the matrix but finally clipped.

Nearest-neighbor searches require the addition of more electronic components. The two most notable are an array of masks, so the memory can be displayed in its entirety, and a nonlinear feedback loop to amplify multiplexed optical signals attenuated in the iterative cycles. Other circuits address the

2. BIG FIVE. The five major subsystems in the optical neural memory are 32-by-32-bit arrays of LEDs, lenslets, photodiodes, a digital memory device, and a medium upon which to record the memory matrix mask.
memory matrix and monitor the stability of the iterative result. During the search, the target's characterizer is multiplexed simultaneously by the entire lenslet array to interact with the complete memory matrix. Single photodiodes positioned behind the masks integrate the light, which, when clipped, yields a 32-by-32-bit iterative result.

Once the CAM stabilizes, it still has to interpret the result in terms of one of the library characterizers. There will seldom be an exact match. In a remarkably clever solution, Farhat has utilized the human observer's ability to recognize a less-than-perfect image. In addition to creating homo-associative memories—relationships of the information with itself—the CAM can form a hetero-associative memory. This means it can relate the same information to another image on the screen, such as an alphanumeric expression. In other words, the system output displayed on a cathode-ray tube is an imperfect four-character code for the object being identified—but something a technician can interpret nonetheless.

The CAM's ability to zero in on a target depends on the memory matrix's size and the number of characterizers that the matrix incorporates. For a 32-by-32-bit matrix, the CAM has a near-100% probability of stabilizing on a hit if the library consists of 30 or fewer characterizers, says Farhat. Though fewer than the hundreds needed for a library of characterizers to identify military and commercial aircraft, it is not a limitation. The CAM simply loads the first 30 into the memory matrix. If it doesn't succeed with that matrix, it loads additional sets.

As robust as the CAM is, to meet the needs of its intended applications in aircraft image identification, robotics, and a variety of other recognition tasks, it requires relatively precise images drawn by smart sensors that eliminate important information. By harnessing an innovative combination of frequency diversity, holography, and Fourier analysis, Farhat hurdled three persistent problems besetting imaging radar: enormous aperture size (and its corresponding high cost), noise, and image orientation.

The high resolution of Farhat's imaging radar is itself a breakthrough, achieved by a combination of frequency diversity, polarization, and multiple views of the target. His chief innovation is wavelength diversity—stepping across a range of frequencies and using different polarizations (Fig. 3). Fourier analysis turns that data into the mathematical equivalent of the impulse response of the target.

4. ROBUST. The CAM's ability to fill in missing information is illustrated by setting the last 12 bits of library word D to zero. After only four iterations, the CAM found the correct word in a library of four.
bility makes it possible to produce a near-visual-quality projection image from any Fourier slice of the object, even though the radar's looks are from a variety of slant angles. "If you've looked at a target from head on to broadside, which is you've probably characterized it totally because aircraft are symmetrical," says Farhat. "You might argue that you have to look at it from the rear, but in most cases you aren't interested on identifying something that left you. We store 90°-characterizers in memory—60° would be enough, but we're looking at 90°."

Visual representation of the projection image does not provide the best CAM characterizers. Range profile data can also yield sinograms—sinosoidal traces that produce a more distinctive signature. The research team currently constructs characterizers from sinograms; but options such as polarization are worthy of consideration and are being studied, says Farhat.

The imaging radar is closer to commercial implementation than the CAM. Existing radar technology operating between 300 and 500 MHz in 3-MHz steps could be adapted to achieve 30- to 50-cm resolution, says Farhat. "What would be required was intermediate, rapid frequency stepping to acquire the data. But using our laboratory equipment and the proper radio-frequency amplifiers, you could set up such a system." Several radar tracking stations would independently measure the target's frequency response from different directions. The data would be transferred to a central computer bank and corrected for phase differences to access a slice in the target's Fourier space.

About 200 frequency steps and 500 looks within a 90° azimuthal angle would generate a complete high-resolution image for a 50-m aircraft. The number of looks is determined by the target's size: the larger the target, the more looks are required. But this does not mandate a system of 500 discrete sensors distributed around the target. First, the aircraft's motion allows each sensor to address the target at more than one angle. Second, the CAM's sinogram-derived associative memory can be counted upon to fill large blocks of missing information (Fig. 9). Using actual radar-retrieved data of a model airplane, the CAM (simulated on a computer because the optical version has not attained a 32-by-32-bit array size) has identified the target with as few as 12 looks when 128 looks and 128 frequency steps were used to create the library characterizers.

Having coaxed the imaging-radar system through research to the verge of development, Farhat has two remaining goals: achieving millimeter-level image resolution in the laboratory and refining the optical CAM to process the radar data in real-time. A Department of Defense University Research Instrumentation Grant is funding a major upgrading of the radar laboratory to add equipment for millimeter-level image resolution, scheduled for completion this fall. It will facilitate frequency stepping as high as 60 GHz and enable the study of such real-world problems as target vibration.

Another advantage is economic: millimeter resolution will give researchers the ability to characterize full-sized aircraft from detailed models in the laboratory, says Farhat.

The CAM technology must be upgraded in two key areas to match with the radar for real-time operation. Using a simple 5-by-5-bit neural network, researchers have finished ironing out the generic wrinkles of optoelectronic CAMs. Next they will implement a 16-by-16-bit neural network; within a year, Farhat expects to be using 32-by-32-bit sinogram characterizers derived from target aircraft models available in the radar lab. "At that point, we will want to find out how well it recognizes the models on a statistical basis from any aspect angle," he says. Computer simulations indicate the 32-by-32-bit optical CAM might make do with 10% of the total characterizer data.

MOVING TO A MASK

In principle, moving from the present technique of storing interconnection data on transparent photographic film to a programmable mask should not pose serious difficulties, says Farhat. Litton Industries, Van Nuys, Calif., markets 48-by-48-bit magneto-optic spatial light modulators that can be used as the storage mechanism for 1-d neural networks; he says, and adapting the system to a 2-d neural network is a relatively simple matter of partitioning the 4-d memory matrix into 2-d components.

Over the long-term, the CAM could have an impact on such technologies as robotics, machine vision, artificial intelligence, and supercomputers. Recognition schemes could include ultrasound, colors, textures, infrared, and—perhaps the CAM's first commercial application—speech processing. Coupled with the imaging radar's smart sensing of primal images, the research will prove fruitful in gaining insights into imaging as a whole, including the eye-brain system, says Farhat.

TAKING LESSONS FROM MOTHER NATURE

For more than 20 years, Nabil H. Farhat has been extracting images through opaque media with constantly improving results. Beginning with microwave holography in 1964, he and his ever-changing team of University of Pennsylvania graduate students managed by 1969 to derive fuzzy holographic views of concealed objects such as a handgun in a suitcase. Though the images were impressive, his research convinced him that single-wavelength holography was stuck with the inherent limitations of speckle noise, range, and cost.

Turning to nature, Farhat reasoned that if bats and dolphins can resolve their environment with great precision using multifrequency clicks and chirps, then spectral diversity might also provide a key for high-resolution radar imaging.

From the project's holographic beginnings, Farhat was constantly on the lookout for a hybrid optical and electronic system for real-time data processing. While on a sabbatical trip to the California Institute of Technology in 1983, Farhat visited the Jet Propulsion Laboratory and became intrigued by the work in associative memory and neural networks that was being done by John Lamb and his colleagues.

"They handed me a paper on the Hopfield model [an algorithm developed by John J. Hopfield], and everything fit together," he recalls. "It was perfect, especially for an optical model."

A week later, he discovered that Caltech's Demetri Psaltis had similar interests. "We put our heads together and wrote a paper drawing the optical community's attention to how well neural networks dovetail with optics." In the human brain, individual neurons can become inoperative without damaging the neural network's integrity—a highly desirable trait for computer or imaging systems that must function for extended periods, as on future space missions that could last 50 to 100 years.

Together, imaging radar and associative memory have numerous applications from determining the condition of heat-resistant panels on the space shuttle to checking rush-hour traffic conditions around New York. Yes, Farhat says, a representative of New York's Port Authority has already contacted him.
Optics and neural nets: trying to model the human brain

Attempts to build computers that work on the same principles as the brain will require radical rethinking of what we consider to be computing. The communications power of optics may play a vital role in this endeavor.

Humans are not logical. This familiar Vulcan proverb illuminates one of the issues frustrating computer scientists and users. While computers outperform the human brain in solving certain classes of mathematical and logical problems, they appear woefully inadequate for other tasks that humans can do instantly, such as pattern recognition and association in real time, using incomplete or distorted input. Why is there such a difference in ability? Is it possible to construct machines that can compete with the human brain in solving the problems that seem to come to it most naturally?

It appears that the quest for ever-faster switching speeds in digital circuits will not provide the solution. Even supercomputers using silicon and gallium arsenide circuits with subnanosecond gate delays bog down at true real-time pattern-recognition tasks, while our brains can perform such problems instantly. And the response time of a neuron is in the millisecond range—not even close to the speed of silicon ICs. An avenue of research is emerging that seeks to understand not only the structure of the brain but also the differences in the class of problems that it's best designed to solve.

Professor Demetri Psaltis of the California Institute of Technology (Pasadena, CA) believes that today's computers lend themselves to solving problems that, by nature, are structured in such a way that they use algorithms having many short steps. Computers break down, however, when confronted with problems that are inherently random, such as pattern recognition. Structured problems, even those that lend themselves to parallelization, are deemed difficult in terms of the time-complexity they involve—or, in other words, the number of steps they take. Humans, however, don't recognize scenes by executing sequential steps, but rather by a process of global associations—processing all the received data at
Optical computers of many designs, sizes and functions are taking shape on blackboards and in laboratories around the world, as work goes on to improve the component used as input, output, scratchpad memory, interconnector and processor. This component is the spatial light modulator (SLM). A recent survey done by the Naval Research Laboratories listed 50 versions of the modulator with a wide array of properties. But what is an SLM, and why is it so critical?

An SLM is a transducer. It converts a two-dimensional pattern of light into a spatial pattern that can vary its brightness. Both continuous and binary outputs are available. There are many other forms the modulation can take. Many SLMs produce a spatial variation in polarization. Others produce patterns of relative phase. And still others produce coupled changes in two or more of these properties.

The SLMs can also do processing. The output spatial pattern doesn't need to be a faithful copy of the input pattern. Some of the input-output relationships include those shown below.

<table>
<thead>
<tr>
<th>Input Pattern</th>
<th>Output Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Incoherent, wide band</td>
<td>Coherent, narrow band</td>
</tr>
<tr>
<td>Continuous in intensity</td>
<td>Binary in intensity or other property</td>
</tr>
<tr>
<td>Well-defined intensity pattern</td>
<td>Reversed-intensity pattern</td>
</tr>
<tr>
<td>Continuous image</td>
<td>Edge-enhanced, dynamic-range-compressed image</td>
</tr>
</tbody>
</table>

With a steeply nonlinear I/O pattern, the SLM can perform logic operations (AND, OR, NAND, NOR) on input light patterns. Also, such SLMs can do thresholding, level restoring, clocking, and so forth. One particular form of I/O nonlinearity is called bistability. In bistable SLMs, picture elements or pixels, that are turned on may stay on until switched off. This provides for memory and for clocking.

SLMs can provide a type of scratchpad memory simply by time integration of multiple input patterns. In the same way, they can convert a serially scanned input pattern into a parallel read-out pattern. With appropriate input and output optics, we can cause each of N input points of light to be connected to each of N output points via an N x N SLM, as shown schematically below.

This simple arrangement is very powerful. By blocking N-1 of the N sources going to any output, we can affect any desired I/O interconnect pattern.

By regarding the inputs and outputs as vector components, we can view the SLM as a matrix. This yields a parallel matrix-vector multiplier. The matrix may represent something as simple as an algebraic problem or something more complex such as a neural network.

SLMs can also be used to compare in parallel many "symptoms" with many indicated "disease patterns" for optical expert systems. Using holograms to address the SLMs, we can access between $10^4$ and $10^6$ different patterns at a very high random-access rate. Unfortunately, current SLMs don't respond that fast. Using moderate values ($10^5$ patterns of $10^5$ pixels accessed in $10^3$), we arrive at a phenomenal pixel usage for a vast store of $10^{18}$ pixels, accessible in 100 s. This capability is well beyond the current capability of electronics, and well below the ultimate limits of optics.

These are only some of the many diverse forms and uses of SLMs. Many experts throughout the world agree that the SLM is the most critical and versatile of the required components of all optical computers, whether they're neural networks or numerical processors.
once. To expect a digital computer to be able to approximate this process in step-by-step algorithms is unrealistic.

The goal of computer scientists is to design a machine that can approach these random-type problems on the basis of global association and with incomplete data to reach valid conclusions. The most successful approach to date appears to be the neural-net model based on the interconnection pattern of neurons observed in the brain.

While there are many differing opinions on how to best implement the neural-net model, and while any such neural model would be a greatly simplified version of the brain's organization, all agree that neural nets represent a radical departure from any previous digital computer architecture. The hallmark of the neural net is massive parallelism and high interconnectivity between a large number of relatively simple processors. The information in a neural processor is stored in the interconnection pattern rather than at specific spatial locations uniquely defined by a memory address.

The need for interconnects between each of a large number of neurons makes implementation in silicon of any system approaching the brain's complexity look intimidating to many researchers who see the inherent high bandwidth, high parallelism and global communications properties of optics as a possible solution. In addition, research is showing that there are certain types of processing tasks, such as matrix operations, that are highly parallel and could lend themselves to solutions by optical processors even though they're not modeled after the brain. Some feel there may be a natural overlap between the needs of neural networks and the capabilities of optics.

Both neural nets and highly parallel numeric problems—if implemented optically—would represent analog processes and would be a radical departure from the von Neumann sequential architectures that have characterized digital computers for the last 30 years. It must be noted, however, that there's a good deal of research into bistable optical devices—including some using gallium arsenide—that could offer a quantum leap in switching speeds beyond that of current machines. But given the nature of time-complexity, such machines would excel at the tasks at which today's architectures already perform well. They wouldn't come considerably closer to the category of random problems at which neural nets are taking aim.

Indeed, resolving such problems requires a fresh look not only at the unique requirements of the machines, but also at the capabilities offered by neural-net and optical technology. Consider the optical technology most familiar to current computer users—the optical disk. Today's optical disks are used as if they're simply high-capacity magnetic media. Data is optically read bit-by-bit with a single laser. Light, however, has the ability to shine on an entire surface at once and has the potential to decode all the recorded bits in parallel at once. If there were some way to utilize that capability, optical storage would be faster, and more important, would represent an entirely different way of using data in a computer system.

The job of building a functional model of the human brain has only just begun. The work is in its most tentative and fundamental stages, and functioning intelligent systems won't be built on neural networks for decades, if then. Nonetheless, there's widespread recognition that it's possible to build hardware models of certain brain-like structures that act like analogous neural circuits observed by neurophysiologists. We know that a working model exists in nature—our own brains—and it's now possible to see the direction we must take to emulate that system.

In addition, work has emerged from academia in the form of startup companies. Such companies are looking for certain applications that would lend themselves to solutions on the basis of what has already been learned. "This is an important step," says Lauren Yasolino, president of Synaptics (San Jose, CA). "Applying the research will provide a feedback loop and aid development of the technology." Synaptics' vice-president of research, Federico Faggin, cautions that neural-net computers will probably not operate like human brains. "If we modeled airplanes after nature, they would have feathers," he says. "The goal is to understand and grasp some fundamental principles and translate those into silicon."

Since research in neural nets is at such a fundamental stage, no agreement has been reached on how to best implement an actual neural circuit. In fact, even an attempt to pin down a definition of neural network leads to lively debates among neurophysiologists, engineers and information scientists. Still, a general consensus is emerging that such circuits will behave like neurons in that their stored information will be distributed among the various nodes and their connections rather than being located at discrete memory addresses. The circuits exhibit high parallelism and interconnectivity and are basically analog in nature. Qualifying "basically analog" is important here because neurons in the brain exhibit analog gradients of stimulation in the area of the dendrites, but send information along...
their axons in pulses, which is what the term “firing rate” applies to when speaking of nerves. The fact that these pulses carry information is clear, but they vary in pulse width, amplitude and frequency, and the mechanism of information encoding is not yet understood.

Researchers’ attention has recently started shifting from these nerve pulses, or action potentials, to the synapses of nerves where the real processing occurs and the real information exists. Nerve pulses take place along axons, and many brain cells don’t even have axons. “Concentrating on action potentials is like walking into Electronics 101 and hearing the professor say ‘OK, the really important thing about electronics is touch-tone dialing,’” says Carver Mead, an electrical engineering professor at California Institute of Technology. Mead is also affiliated with Synaptics.

Given the high connectivity and distributed nature of information in neural nets, it’s clear that for a neural network to store and process useful information, the number of nodes has to reach a certain level of complexity. As John Neff, project manager for the Defense Advanced Research Projects Administration (Arlington, VA), says, “For these [neural nets] to be practical, they’re going to have to have a million or more nodes.” Such a system would be inherently robust because removing a given node wouldn’t destroy vital data, although at some point, the number of mistakes caused by disabling nodes would cease to be acceptable. Synaptics’ Faggin is among those who see a real possibility in implementing neural nets in silicon. “We can now really think of doing wafer-scale integration,” he says. “Flaws that inevitably occur on a whole wafer would only represent a statistical factor in the wafer’s quality, they wouldn’t render the whole wafer useless.”

Professor John Hopfield, professor of chemistry and biology at the California Institute of Technology has proposed a theoretical model that illustrates how a neural network might lend itself to problems that entail combinatorial complexity. Hopfield’s model involves a network of interconnected neurons that’s set up to reveal an optimization in terms of a global minimal energy state for the system. In other words, when the circuit is started, it will reach a stable state that represents the minimum sum of energy for the whole circuit. Certain nodes, how-
Neural net models for computations

Creating an artificial intelligence system that has the flexibility and the creativity of the human mind is one of the oldest goals of computer science. Ironically, it has remained the most elusive, despite the tremendous strides made in microelectronics and in general-purpose digital computing. The AI field has made substantial progress during recent years, performing difficult but well-defined tasks. These tasks include playing chess, diagnosing diseases and inferring biochemical structures of complex molecules.

Yet the staggering computational capabilities that can be achieved by the supercomputers and complex, powerful organizations of the AI systems of today seem far from accomplishing some elementary but poorly defined tasks, such as understanding and producing continuous speech, moving in a complex and dynamic three-dimensional space with only the aid of somewhat noisy two-dimensional detectors, and making inferences using common sense. Most human beings find these tasks easy to perform, whereas the more well-defined tasks that AI can solve require long and arduous training for humans.

This observation has led scientists in the neuroscience, psychology, mathematics, physics, computer science and electrical engineering fields to conclude that intelligent biological systems, such as the human brain, are organized along fundamentally different lines than most AI systems. In spite of the differences in their backgrounds and in the approaches that they follow, the researchers share a common goal of trying to gain a basic understanding of how intelligent biological systems solve incompletely defined problems, and applying these principles to the design and construction of AI systems so that they can solve those problems.

However, will be more "on" than others and from them, one can read the solution to the problem.

One of the characteristics of synaptic interfaces is that they include connections that stimulate a neighboring neuron as well as connections that inhibit stimulation of that neuron. Some even have inhibitory feedback loops onto themselves. It's clear that information is stored not only in the state of a node, or in the existence of a connection, but also in the strength of that connection.

One of Hopfield's examples involves the problem of a salesperson having to visit a given number of cities, visiting each city once, in such a sequence that the total distance traveled is minimized. Hopfield has arranged a set of connections with neurons in rows and columns. The rows, which are labeled alphabetically, will correspond to the cities on the tour. The columns correspond to the locations of the cities in the tour. Thus, if the node in row A, column S is most strongly energized when the system reaches stability, city A will be the fifth city in the sequence of the tour.

Setting up such a problem requires what Hopfield calls "a complex topology of syntax-rein-
Another critical feature of biological systems is the emphasis that is placed on self-organization and learning. Digital electronic systems rely on software for guiding the flow of control and data signals. This feature may prove to be crucial to useful applications of neural nets because the network can form a complete internal representation from a partial description of the problem. It may no longer be necessary to completely understand the problem in order to solve it. In addition, the same established learning principles may be applied successfully to all problems within a given class, changing the goal from understanding the differences between problems to discovering the similarities.

In traditional computing, solving a problem usually involves several distinct stages including defining the problem, selecting the methodology and algorithm, coding the algorithm, and performing the computation. Computation usually requires a way to represent the distances forcing connections. Each node must first be capable of directly stimulating every other node. Also, each node must also have connections to inhibit other nodes. In this example, if city A, position 5 node is going to be “on” at the end of the problem, all other nodes in row A and column 5 must be suppressed somehow, requiring another set of connections. Another aspect of the connectivity syntax requires a way to represent the distances between the cities. This is done by adding resistances into the connection pattern that correspond to these distances.

It's important to note that the circuit is operated in an analog range. At the beginning, all neurons are in a nonzero, low-energy state. For a small number of cities, the circuit rapidly computes the right answer. When the number of cities is increased, the processing time remains about the same, and the circuit settles on a set of the best answers. A 30-city tour, for example, requires 900 neurons and has $10^{30}$ possible tours. The neural circuit is able to find the $10^7$ best solutions in a few time constants or in about 1 μs. This represents a selection factor of $10^{23}$, according to Hopfield.

"Forcing connections." Each node must first be capable of directly stimulating every other node. Also, each node must also have connections to inhibit other nodes. In this example, if city A, position 5 node is going to be “on” at the end of the problem, all other nodes in row A and column 5 must be suppressed somehow, requiring another set of connections. Another aspect of the connectivity syntax requires a way to represent the distances between the cities. This is done by adding resistances into the connection pattern that correspond to these distances.
Pattern recognition by filtered Fourier transforms

It's no surprise that our computers don't understand us yet—we've been trying to communicate with them via an inferior medium language. Most of the input to a normal human brain is through the visual system. Far less data is received through the auditory channel. Since most biological systems are designed for utmost efficiency, expecting auditory-based linguistic information to be the most important component of human thought processes seems inconsistent with evolution.

The brain's ability to perform pattern-recognition tasks sets it apart from machines of von Neumann architecture. The filtered Fourier transform pattern-recognition technique is representative of early attempts to understand and simulate these abilities on von Neumann machines.

In 1966, Matthew Kabrisky of the Air Force Institute of Technology (W-Patterson AFB, OH) published the book, "A Proposed Model for Visual Information Processing in the Human Brain." In subsequent works, Kabrisky and several of his students investigated the possibility that two-dimensional filtered Fourier transforms are involved in the computational processes that occur in the human brain.

In 1967, Radoy, one of Kabrisky's students, demonstrated a pattern-recognition system that can recognize alphabetic characters. In essence, such a system overlays a pattern with a grid, extracts the brightness of the grid squares, enters those brightness values in a complex-pattern matrix and calculates a discrete two-dimensional Fourier transform, which is also a complex matrix of the same order as the pattern matrix. It then stores the transform matrix. Pattern recognition is performed by saving the transform matrices of various patterns and then comparing the transform of a new pattern with the transforms of the stored patterns. A new pattern is recognized as the stored pattern whose transform is most closely matched with the transform of the new pattern. This operation is done by calculating the Euclidean distance between the transforms.

Radoy found that ignoring the terms in the transform matrix that were associated with high-frequency components only minimally affected recognition of alphabetic characters. Using a technique known as low-pass spatial filtering, he reduced storage requirements of pattern transforms by a factor of 100 without seriously degrading the machine's ability to recognize patterns.

In 1969, Tallman, another of Kabrisky's students, experimented with hand-printed samples of all 26 alphabetic characters from 25 different people. By using the filtered Fourier transform technique, Tallman was able to achieve a 95 percent recognition rate for the set of 650 characters.

Kabrisky has pointed out that written characters, whether Arabic numerals or Chinese Kanji characters, evolved so that they are distinguishable by people. The filtered Fourier transform technique seems to identify the essence of a character—that which distinguishes it from other characters.

For an even better set of solutions, a technique known as annealing has been proposed. Annealing lets the system reach one stable state and then energizes it to find an even lower global energy level. This is similar to heating a crystalline structure to some temperature and then cooling to produce a more perfect crystalline pattern.

The fact that the neural net doesn't pick out one single best answer (often there are several) also fits the analog or "fuzzy" nature of deciding among conflicting solutions. Nevertheless, the network is able to consider the solutions simultaneously. By comparison, a typical microcomputer can find a comparably good solution in about 0.1 s, according to Hopfield. But the microcomputer has about 10^4 times as many devices as neural net.

This reveals several facts, the most significant being the role the connectivity pattern plays in representing the data, the problem and the program for the solution. The pattern of the interconnections programmed the system, and the use of the network to find a global minimum depends on the pattern of interconnection. In addition, the solution in this example is a static state, whereas the brain operates in real time with ever-changing input. Nevertheless, the neural net arrives at an acceptably correct solution. The pattern of the interconnections that are repeatedly stimulated are strengthened. As a result, the brain not only repre-
For example, note the T pattern in the figures below. Intrinsic brightness of the elements of this pattern is indicated by the size of the dark squares that make up the image. Negative values (those that appear in the inverse transforms) are shown by dashes of various lengths. Longer dashes indicate more negative values. If you take the Fourier transform of the pattern of the first image, and then take the Fourier transform of the transform, you will get the original pattern—that is, Fourier transforms are invertible. If you filter (eliminate high-frequency terms) the Fourier transform of the T before inverting it, you will produce the middle image: the 5x5 filtered inverse transform. It’s interesting that a pedestal forms at the base of the T in the filtered inverse transform and that serifs form at the ends of the horizontal bar. Compare this with the Triplex Roman T of the Hershey font set shown in the third figure.

Is it possible that the serifs and pedestal came into vogue in printer font sets because that’s the form of the most distinguishable T? Some think that it’s the most aesthetically pleasing form. Does the concept “aesthetically pleasing” derive from peculiarities of our internal image processors? Whether actual Fourier transform processes are occurring in the brain remains a matter of speculation. In any case, in terms of speed, the von Neumann architecture does not seem to be the appropriate architecture for simulating the brain’s pattern-recognition processes. Neural-net machines demonstrating self-organization of memory seem to be on the right track. These new machines will help to revolutionize the computer industry.

(continued on page 58)
As mentioned earlier, work on optical bistable devices is making progress. Hitachi (Tokyo, Japan) has developed an optical switch that can switch between two channels at 833 MHz, and spatial light modulators exist that operate in a nonlinear, or binary, mode. It is also true that optical bistable devices will probably find more immediate applications in real-world designs—especially in telecommunications—than some of the proposed optical systems discussed here. In computer systems, there's a branch of research that's looking into using optics to communicate among circuit boards as well as among VLSI components on the same board. One problem that's badly in need of a solution is clock skew between high-speed components. Optics are also being considered for this field. Still, the possibilities of using the high bandwidth and global communications abilities of light to aid in the neural-network class of future computers is also a lively area of research, although it's still very much confined to the university and the laboratory level. In addition, the use of light for all-optical processors is receiving serious attention.

The potential computational power of optical processors can be shown by the use of two-dimensional spatial light modulators (SLMs) in matrix processing. Matrix operations with light take advantage of the inherent global communications and the ability to easily integrate intensities to

Optics...
(continued from page 55)

Mead's model takes advantage of the fact that the cones in the eye are stimulated by light, and that they have outputs that feed back onto them and inhibit stimulus in somewhat the same manner as in the Hopfield model. The eye responds to the changes in light value rather than to absolute light intensity. The cones and Mead's CMOS photodetectors output a time derivative of intensity on a logarithmic scale rather than a linear one. Another layer, the amicrine layer, computes a spatial derivative of the time derivative provided by the sensors. The amicrine cells provide a passive resistive network that modifies the output of neighboring cells/detectors.

The silicon retina, like the natural one, relies on rates of change to detect moving objects, and it can do so in real time, unlike digital computers. The human eye is constantly undergoing minute motions to create the images we see. If that motion were to stop, the time-dependent rate-of-change computations would cease, and the image would fade. The silicon retina responds in the same way.

One means of using light to connect signals between switching elements in a multilayer hybrid optoelectric circuit is to use beams that may be redirected by changing the characteristics of the diffraction gratings, which are implemented as holograms.
A proposal for a digital all-optic computer shows beams from a source array directed at a bistable switching array whose output is directed to both memory and to an output array (a). A feedback loop with mirrors allows input to the bistable array to be altered. Bistable elements of the array (b) can be logically grouped to form functional processor components. Altering the configuration via the beam controller would alter the system architecture.

Another operation involves multiplying two vectors element-by-element to arrive at an inner product, which is the output scalar. Here the output light from each element of the first vector is sent only to the corresponding element of the second vector encoded on the SLM. The outputs of each element of the SLM are then summed by focusing them onto the single detector, which represents the resulting scalar value.

These two simple examples can be increased in complexity to produce vector-matrix and matrix-matrix multipliers and even higher order functions. Just as different types of optics are needed for different types of operations, different types of SLMs and detectors may be used for different purposes. For instance, a time-integrating detector array can be used to sum results. Such an array holds the input of one element and adds the weight of the next as indicated by the intensity of light. The result is the sum of successive elements. Another type of detector receives the sum of several elements simultaneously as a focused beam. Obviously, the kind of...
Matrix processors described here show the potential of the technology rather than describing practical systems. One challenge is how to design hardware that can handle the demands placed on it by the different computation tasks.

Also important is the fact that the examples were analog designs. The value of each element as vector or matrix was coded in the SLM as an analog intensity value via the transmittance function of one of the SLM's cells. Since the output result—the transmitted light—has an amplitude proportional to the product of the two numbers encoded as light, the accuracy of such values depends on the accuracy of the SLM. The SLM must have uniformly linear characteristics to accurately represent numerical values as gradations of light. It must be recognized from the start that this type of optical system doesn't lend itself to high numerical precision.

It's possible to construct SLMs with nonlinear characteristics that can represent digital numbers. In such a device, each cell would be a 1 or a 0, and multiple cells would be grouped to represent bytes or words. Encoding data as binary numbers, however, works against the efforts toward parallelism, forces the system to work with digital logic and introduces many of the repetitive operations that the parallel optical approach hopes to avoid.

Considering that there are devices such as SLMs that allow combinatorial operations using light, and that the analog nature of these devices fits well with the distributed and "fuzzy" character of neural processing, how might optics be used in the service of the neural model? According to Darpa's Neff, the communications abilities of optics could be used in hybrid optoelectronic systems that emphasize the need for reconfigurable connectivity between switching elements, such as between laser diodes and detectors. "As the emphasis on switching decreases, the emphasis on connectivity rises, and optics becomes more of an option," he notes.

One proposed hybrid scheme involves layers of hybrid optoelectronic chips containing laser diodes and detectors and a system of reconfigurable diffraction gratings that act as frequency-selectable filters to pass and/or direct the various beams containing data to appropriate places on different layers of circuit boards. In Neff's proposal, the diffraction gratings are holograms created by mixing waves of four different frequencies. Diffraction-grating writing beams would also be used to change the characteristics of the hologram to redirect the switching beams.

The next step, suggests Neff, might be an all-optical computer in which an optical source array (such as an SLM or a laser diode array) acts upon a processing array, which could also be a type of SLM or an array of optical bistable devices. In the latter case, the bistable devices would give it a more digital character, since they would act as logic gates. If bistable gates were used, they could be grouped to make up processing elements such as ALUs, shift registers, clock signals and so forth.

Even if such a machine were implemented in a bistable mode, there would still be great emphasis on the configuration of connectivity, since rearranging the connections would redefine not only the functional logic elements but also the entire architecture of the system. The critical element used to control the interconnection would be some kind of beam controller, such as a large diffraction grating, that could be programmed for the desired interconnects. This controller would also interact with the processing unit via a feedback loop to the input side of the array. Neff stresses that no one has built such a computer, but, he says, "It's technically believable to achieve such a system consisting of 1 million parallel channels."

If this scale of parallelism is possible in a bistable system, what about an analog machine with global connectivity? Numbers vary, but California Institute of Technology's Psaltis envisions arbitrarily connecting $10^4$ neurons, which would translate to $10^8$ connections in which each neuron could connect directly to every other neuron. Making that
connectivity programmable on such a scale requires techniques that aren't yet understood.

It's possible to build simple associative memories using SLM and detector arrays, but the high connectivity envisioned by researchers such as Neff and Psaltis requires some kind of medium with many more resolvable spots in a given area or volume than today's SLMs. Such resolvable spots would be used to refract and redirect individual beams of light to make or break connections between neurons. Two candidates that have been suggested are magneto-optic surfaces and photorefractive crystals.

Those searching for erasable optical media in optical disks are looking closely at magneto-optics, and when a solution is found, the very dense bit pattern on optical disks can be reconfigured. Such disks or surfaces implemented with the same technology used in magneto-optical disks could theoretically be used to specify the connections between several thousand lasers and detectors, according to Psaltis. Thus, the optical disk, which is currently used as if it were merely a denser form of magnetic media, might be used to its fuller potential in optical/neural systems.

But to truly achieve massive connectivity and dynamic reconfigurability, Psaltis suggests holography using photorefractive crystals. This is especially interesting because the global distributed manner in which the brain stores and processes information has often been compared to a hologram. One of the most striking characteristics of holograms is that the stored image can be reconstructed from only part of the hologram. This has led researchers to look into implementing associative memories using holograms—a field that looks promising. Further, recording connections as a hologram in a photorefractive crystal increases the number of possible connections by virtue of being in three dimensions, and makes the connections programmable.

Light in a photorefractive crystal generates free charges that are eventually trapped in a pattern similar to the intensity pattern of the incoming light. The spatially varying charge density that results creates internal fields that change the index of refraction within the crystal and produces in the hologram. When light shines into the crystal, it's refracted in directions determined by these varying refraction indices, giving the image of the hologram. In the case of our hypothetical computer, the image represents the pattern of interconnects. And that pattern—the hologram—can be modified by light from a feedback loop, giving the desired dynamic reconfigurability.

In the Hopfield model, the program as well as the information is stored in the communications network in a neural computer, and to be at all flexible, the system must be quickly reconfigurable. As a result, some form of intelligence is needed to process the information received and to determine how
to adapt its configuration pattern to the ongoing process of real-time computation. The system needs the ability to learn.

Some experiments have shown that holography can correlate matrices of output devices and detectors. One such experiment has demonstrated an associative memory that can pick out one of four recorded pictures of human faces, given only a partial picture as input. Fourier transforms of the images are stored in two holograms. Each image is recorded at different spatial frequencies so that they appear to be on separate planes. Partial-image data for the desired image is shined through a beam splitter into a loop formed with the holograms. The first hologram acts as a detector, and its output causes the holographic representation of only the selected image to begin to appear from the second hologram. This output is fed back into the system through a threshold device, an amplifying SLM. After a few iterations, the output is strong enough to pass through the beam splitter and be projected as the selected image.

An interesting extension of implementing connectivity via holograms has been proposed by Psaltis and a graduate student, Kelvin Wagner. The system, which is called a backward error propagation (BEP) learning network, would use holograms in photorefractive crystals to make connections between an input array and an output array. But processing in the output array—or in subsequent neural layers associated with it—would generate error signals based on a desired connection pattern.

The output array would also need the ability to send signals back to the photorefractive crystal and then to a system of polarizers and a phase conjugate mirror, which would adjust the phase of the error signal to alter the hologram in the direction of the image, producing the desired connectivity. Such an error signal could be continuous or pulsed, but would die away as the forward signal approached the desired connection pattern.

The models described here are representative of a wide range of research going on in both neural networks and in optical computing. None of them represents a practical working computer system and even those that have been implemented are experiments to prove principles and test hypotheses.

There is a realization that the need for a so-called "new paradigm" for neural-like computing systems carries with it the need for a new approach to the information science describing such machines. Neff, Psaltis, Hopfield, Mead and Faggin all caution that the idea of neural networks and learning systems doesn't imply a heterogeneous "mush" of infinitely replicated and interconnected neurons. Just as the brain is highly structured, these new systems will need a structure and hierarchy as well as an organizational basis to determine how they will learn, how they will preprocess and select input information, and how different parts of such intelligent systems will perform specific functions.

This is a science still in its most rudimentary stages. It will build in a kind of feedback loop as people try to solve relatively specialized problems using neural models and learn from their experiences. "The nervous system is based on a set of organizing principles different from any computational paradigm we know," says Mead. The process of understanding that paradigm, he argues, must start from the bottom up. Neural "primitives" are computationally powerful in their own right and include such things as exponential functions and integration with respect to time. "At the bottom level, the power of neural networks comes from the fact that they don't insist on taking a beautiful thing that creates an exponential and turning it into a 1 or a 0," Mead says. "They take what is there and use it."

Although building a working model of the brain is still a distant dream, using neural network models to perform certain special tasks is within reach. As applications are found, techniques will be developed and neurobiologists will take advantage of neural models just as computer scientists learn from neurobiology in an ongoing cooperative research effort. As for a working brain, it may be far off, but it's not impossible. As Lee Giles, program manager for the Air Force Office of Scientific Research at Bolling Air Force Base (Washington, DC) notes, "We have existing proof—us!"
Optical information processing systems can have high processing power because of the large degree of parallelism as well as the interconnection capability that is achievable. Typically, more than $10^5$ parallel processing channels are available in the optical system, and furthermore each of these channels can be optically interconnected (broadcasted) to $10^6$ other channels. The majority of optical processors are analog systems, designed to perform linear operations. The accuracy of an analog processor is limited by the linear dynamic range of the devices used (detectors, light modulators). In principle, the accuracy and the repertoire of achievable operations can be improved with systems that perform nonlinear operations on binary encoded data using bistable optical devices. Optical bistability is a subject that has received considerable attention recently as a means of achieving efficient high-speed logic, and it has been demonstrated with several nonlinear optical materials and devices. If we are to use such bistable devices to realize powerful, nonlinear optical computers, it is important to find algorithms that are well matched to the characteristics of the optical processor and utilize effectively its parallelism and interconnection capability. In this Letter we examine a method for synthesizing optical processing systems, based on optical associative memory and threshold logic, that appears to meet these requirements well.

Associative (or content-addressable) memories are of interest in computer science, and it is theorized that information is stored in the human brain in this manner. Holographic associative memories have been described by Gabor, who also commented on the similarity of the holographic memory to the way information may be stored in the human brain. More recently, Hopfield introduced an associative-memory model to describe the collective behavior of neural networks. Hopfield's model consists basically of an associative memory similar to the holographic, with the addition of threshold and feedback. The incorporation of nonlinear feedback enhances dramatically the error-correcting capability of the holographic memory.

Let $v_i^{(m)}$ be a binary word that is $N$ bits long. $M$ such words are stored in a matrix $T_{ij}$ according to

$$ T_{ij} = \begin{cases} \sum_{m} [2v_i^{(m)} - 1][2v_j^{(m)} - 1] & \text{if } i \neq j \\ 0 & \text{if } i = j \end{cases} \quad (1) $$

If $T_{ij}$ is multiplied by one of the stored binary vectors $v_i^{(m)}$, the product $\hat{v}_i^{(m)}$ is an estimate of the stored vector $[2v_i^{(m)} - 1]$

$$ \hat{v}_i^{(m)} = \frac{1}{2} \sum_{j=1}^{N} T_{ij} v_j^{(m)} = \frac{N}{2} (2^{N-1} + 1) $$

where the last term accounts for $T_{ij} = 0$ and $N_0$ is the number of 1's in $v_i^{(m)}$. We assume that for $m \neq m_0$ the binary words $v_i^{(m)}$ are statistically described in the following simple manner:

$$ P(v_i^{(m)}) = \frac{1}{2}, \quad P(v_i^{(m)}) = 0 $$

where $v_i^{(m)}$ are independent for all $i$ and $m$. Then $E(\hat{v}_i^{(m)}) = (N/2) [2v_i^{(m)} - 1]$ and $\text{var} \hat{v}_i^{(m)} = N(M - 1)/2$. We define the signal-to-noise ratio (SNR) of the estimate $\hat{v}_i^{(m)}$ as the ratio of the magnitude of the expected value of $\hat{v}_i^{(m)}$ to the standard deviation of the estimate:

$$ \text{SNR} = \frac{||E(\hat{v}_i^{(m)})||}{||\text{var} (\hat{v}_i^{(m)})||} = \frac{N}{N(M - 1)^{1/2}} \quad (4) $$

If $N$ is sufficiently larger than $M$, then with high probability $TH[\hat{v}_i^{(m)}] = v_i^{(m)}$, where $TH[\hat{v}_i^{(m)}] = 1$ if $\hat{v}_i^{(m)} > 0$ and zero otherwise. Thus the vector-matrix product in Eq. (2) combined with the thresholding operation results in a pseudoeigensystem in that the output vector equals the input. Now suppose that the full vector $v_i^{(m)}$ is in fact such a pseudoeigenvector of the system but that only $N_1$ of $N$ bits ($N_1 \leq N$) of $v_i^{(m)}$ are known. In this case we define an input vector consisting of the $N_1$ known bits, and the rest are set equal to zero. When this vector is multiplied by the matrix $T_{ij}$, an estimate of the complete vector $[2v_i^{(m)} - 1]$ is obtained. The SNR of the estimate is now $\text{SNR} = [N_1/2(M - 1)]^{1/2}$.
estimate \(|N_1/(2M - 1)^{1/2}\)| is sufficiently large, then the probability of \(N_2\) being bigger than \(N_1\) will be high; in this case the nonlinear, iterative procedure described \(\ast\) we will be likely to converge to the correct vector \(v_0\). Ideally, each of the \(M\) stored binary words is a pseudo-eigenvector of the nonlinear system. Notice that each pseudo-eigenstate is a stable state of the system, whereas any other input vector (state) will cause a change to occur in the next cycle. In general, the system converges to the stable state that is at the shortest Hamming distance away from the initial state.

This model has been studied computationally by Hopfield.\(^2\) In simulations, correct convergence was obtained reliably for \(M \leq 0.15N\) and \(N_1 = 0.75N\), taking \(N = 30\). At present there is no (adequate) theoretical prediction of the maximum number of words that can be stored or the maximum Hamming distance between the input vector and one of the stored words that is required for convergence. Several interesting properties were observed. The model does not require synchronism. Convergence can be obtained if the output vector is fed back to the input as a whole or, randomly, one element at a time. There is some evidence that asynchronous operation is actually preferable. The system is quite insensitive to imperfections such as nonuniformities, the exact form of the threshold operation, and errors in the \(T_{ij}\) matrix. Convergence to the correct vector was obtained even when the \(T_{ij}\) matrix was thresholded. Such properties are most desirable when an optical implementation is considered.

One possible optical implementation of the Hopfield model is through the arrangements shown in Fig. 1, in which the array of light-emitting diodes (LED's) represents \(N\) logic elements with binary states \(v_j = 0, 1, 1, \ldots N\) (LED on or off), which are to be interconnected in accordance with the model. This is achieved by the addition of nonlinear feedback (feedback, thresholding, and gain) to the well-known optical vector-matrix multiplier.\(^3\) Gain is included in the feedback loop to compensate for losses. Two possible feedback schemes are shown. One uses electronic wiring and the other is optical, with the thresholding (point nonlinearity) and the gain concentrated between the photodiode (PD) array and the LED array, which can be fabricated monolithically on GaAs. Furthermore, with the accelerating pace of research in thin-film nonlinear light amplifiers\(^4\) and optical bistable devices,\(^5\) it is possible to substitute a single distributed bistable light-amplifier device for the PD/LED arrays and the intervening thresholding and amplifying electronics.

Multiplication of the vector \(v_i\) by the \(T_{ij}\) matrix in these schemes is accomplished by horizontal imaging and vertical smearing of \(v_i\) using anamorphic optics (omitted from Fig. 1 for simplicity). A bipolar \(T_{ij}\) can be realized optoelectronically with incoherent light by assigning its negative and positive values to adjacent rows. Light passing through each row is focused onto adjacent pairs of photodiodes of the PD array that are electronically connected in opposition, as shown in Fig. 1. Here the positive and negative elements of each row of the \(T_{ij}\) matrix are separated into two subrows, one for positive values and one for negative. The light transmitted through the two subrows is integrated horizon-

tally with the aid of another set of anamorphic lenses (omitted from Figs. 1 and 2) and brought to focus on two adjacent photodiodes of the PD array connected in opposition. The output of the first diode-pair circuit will be proportional, \(v_j = \Sigma_i T_{ij} v_i\). This output is applied through an electronic thresholding circuit to the first element of the LED array, as shown in Fig. 1. Similar connections are made between other detector pairs of the photodector array and corresponding elements in the LED array. Thus each LED assesses the state of its input \(v_i = \Sigma_j T_{ij} v_j\) and fires accordingly to whether \(v_i\) exceeds the threshold or not.

We now consider the possibility of optically storing two-dimensional (2-D) functions (images). Let \(u^{(m)}(i, i')\) be the bipolar binary \((1, -1)\) images to be stored. If we directly extend the Hopfield model to two dimensions, then these images must be stored in a four-dimensional function in the following general form:

\[
T(i, i', j, j') = \sum_{m} v^{(m)}(i, i') u^{(m)}(j, j').
\]

In order to implement a 2-D Hopfield memory optically, we need to realize a 2-D, linear optical system whose spatial impulse response is the four-dimensional function defined in Eq. (5). Since we have only two spatial coordinates to work with in an optical system, it is difficult to implement such a system directly for the nonseparable, shift-variant kernel defined in Eq. (5). One possible solution is the use of wavelength multiplexing and/or time-domain processing to obtain additional independent variables. Another solution is based on holographic associative memories, as we have discussed earlier.\(^6\) Here we present an implementation based on spatial-frequency multiplexing.

The entire optical system, including nonlinear feedback, is shown in Fig. 3. The system accepts a 2-D
amplifier at light at plane sired output that is produced by the interconnection interferometrically detected by the nonlinear optical prescription given in Eq. (5). The modulation of the input and stored images. This is precisely the detude at weights being proportional to the inner product between gram, the transform of the mth image being centered transform between planes transparency, T_2(k, k'), consists of a 2-D array of Fourier-transform holograms in P_3 with a randomly chosen, uniform phase.

We have described several specific optical implementations of the Hopfield model; undoubtedly others are also possible. The most important feature of all such implementations is the robustness of a system that utilizes nonlinear feedback. The systems that we have described behave basically as associative memories (the whole is retrieved from a partial input), even with open-loop operation. However, the nonlinear feedback can correct errors of the open-loop system since it forces the state of the system to change continuously until a stable condition is reached. The nonlinearity plays a crucial role; if linear feedback were used, the system would either be unstable or converge to the eigenstate of the open-loop system with the highest eigenvalue, independently of the initial condition.

This error-correcting capability can provide the accuracy that is lacking from analog optical processors without, however, sacrificing the processing power that can be derived from the global processing capability of optics; the class of processors that we described are fully interconnected optical systems and hence utilize fully the parallelism and the interconnective capability of optics. In general, there is an excellent match between the global, linear operations and local, point nonlinearities that are required for the implementation of the Hopfield model, and the capabilities and limitations of optical techniques.

The authors thank John Hong and Yaser Abu-Mostafa for many helpful discussions on this subject. * On scholarly leave from the University of Pennsylvania, Philadelphia, Pennsylvania 19104.

References
Optical implementation of the Hopfield model

Nabil H. Farhat, Demetri Psaltis, Aluizio Prata, and Eung Paek

Optical implementation of content addressable associative memory based on the Hopfield model for neural networks and on the addition of nonlinear iterative feedback to a vector–matrix multiplier is described. Numerical and experimental results presented show that the approach is capable of introducing accuracy and robustness to optical processing while maintaining the traditional advantages of optics, namely, parallelism and massive interconnection capability. Moreover a potentially useful link between neural processing and optics that can be of interest in pattern recognition and machine vision is established.

I. Introduction

It is well known that neural networks in the eye–brain system process information in parallel with the aid of large numbers of simple interconnected processing elements, the neurons. It is also known that the system is very adept at recognition and recall from partial information and has remarkable error correction capabilities.

Recently Hopfield described a simple model for the operation of neural networks. The action of individual neurons is modeled as a thresholding operation and information is stored in the interconnections among the neurons. Computation is performed by setting the state (on or off) of some of the neurons according to an external stimulus and, with the interconnections set according to the recipe that Hopfield prescribed, the state of all neurons that are interconnected to those that are externally stimulated spontaneously converges to the stored pattern that is most similar to the external input. The basic operation performed is a nearest-neighbor search, a fundamental operation for pattern recognition, associative memory, and error correction. A remarkable property of the model is that powerful global computation is performed with very simple, identical logic elements (the neurons). The interconnections provide the computation power to these simple logic elements and also enhance dramatically the storage capacity; approximately N/4 lnN bits/neuron can be stored in a network in which each neuron is connected to N others. Another important feature is that synchronization among the parallel computing elements is not required, making concurrent, distributed processing feasible in a massively parallel structure. Finally, the model is insensitive to local imperfections such as variations in the threshold level of individual neurons or the weights of the interconnections.

Given these characteristics we were motivated to investigate the feasibility of implementing optical information processing and storage systems that are based on this and other similar models of associative memory. Optical techniques offer an effective means for the implementation of programmable global interconnections of very large numbers of identical parallel logic elements. In addition, emerging optical technologies such as 2-D spatial light modulators, optical bistability, and thin-film optical amplifiers appear to be very well suited for performing the thresholding operation that is necessary for the implementation of the model.

The principle of the Hopfield model and its implications in optical information processing have been discussed earlier. Here we review briefly the main features of the model, give as an example the results of a numerical simulation, describe schemes for its optical implementation, then present experimental results obtained with one of the schemes and discuss their implications as a content addressable associative memory (CAM).

II. Hopfield Model

Given a set of M bipolar, binary (1,−1) vectors $v_i^{(m)}$, $i = 1,2,3 \ldots N$, $m = 1,2,3 \ldots M$, these are stored in a synaptic matrix in accordance with the recipe

$$T_{ij} = \sum_{m=1}^{M} v_i^{(m)} v_j^{(m)}, \quad i,j = 1,2,3 \ldots N, \quad T_{ii} = 0, \quad (1)$$

$v_i^{(m)}$ are referred to as the nominal state vectors of the neurons.
memory. If the memory is addressed by multiplying the matrix \( T_{ij} \) with one of the state vectors, say \( \psi_{i}^{(m)} \), it yields the estimate

\[
\psi_{i}^{(m)} = \sum_{j} T_{ij} \psi_{j}^{(m)} = \sum_{j} \sum_{m} \psi_{j}^{(m)} v_{j}^{(m)} v_{j}^{(m)} = (N - 1) v_{i}^{(m)} + \sum_{m,m \neq \gamma} \alpha_{m,m} \psi_{i}^{(m)}
\]

where

\[
\alpha_{m,m} = \frac{N}{N} \sum_{j} \psi_{j}^{(m)} v_{j}^{(m)}
\]

\( \psi_{i}^{(m)} \) consists of the sum of two terms: the first is the input vector amplified by \((N - 1)\); the second is a linear combination of the remaining stored vectors and it represents an unwanted cross-talk term. The value of the coefficients \( \alpha_{m,m} \) is equal to \( \sqrt{N} - 1 \) on the average (the standard deviation of the sum of \( N - 1 \) random bits), and since \((M - 1)\) such coefficients are randomly added, the value of the second term will on the average be equal to \( \sqrt{(M - 1)(N - 1)} \) if \( N \) is sufficiently larger than \( M \), with high probability the elements of the vector \( \psi_{i}^{(m)} \) will be positive if the corresponding elements of \( \psi_{i}^{(m)} \) are equal to \( +1 \) and negative otherwise. Thresholding of \( \psi_{i}^{(m)} \) will therefore yield \( \psi_{i}^{(m)} \):

\[
\psi_{i}^{(m)} = \text{sgn}[\psi_{i}^{(m)}] = \begin{cases} +1 & \text{if } \psi_{i}^{(m)} > 0 \\ -1 & \text{otherwise.} \end{cases}
\]

When the memory is addressed with a binary valued vector that is not one of the stored words, the vector–matrix multiplication and thresholding operation yield an output binary valued vector which, in general, is an approximation of the stored word that is at the shortest Hamming distance from the input vector. If this output vector is fed back and used as the input to the memory, the new output is generally a more accurate version of the stored word and continued iteration converges to the correct vector.

The insertion and readout of memories described above are depicted schematically in Fig. 1. Note that in Fig. 1(b) the estimate \( \psi_{i}^{(m)} \) can be viewed as the weighted projection of \( T_{ij} \). Recognition of an input vector that corresponds to one of the state vectors of the memory or is close to it (in the Hamming sense) is manifested by a stable state of the system. In practice unipolar binary \((0,1)\) vectors or words \( b_{i}^{(m)} \) of bit length \( N \) may be of interest. The above equations are then applicable with \( 2b_{i}^{(m)} - 1 \) replacing \( v_{i}^{(m)} \) in Eq. (1) and \( b_{i}^{(m)} \) replacing \( v_{i}^{(m)} \) in Eq. (2). For such vectors the SNR of the estimate \( \psi_{i}^{(m)} \) can be shown to be lower by a factor of \( \sqrt{2} \).

An example of the \( T_{ij} \) matrix formed from four binary unipolar vectors, each being \( N = 20 \) bits long, is given in Fig. 2 along with the result of a numerical simulation of the process of initializing the memory matrix with a partial version of \( b_{i}^{(4)} \) in which the first eight digits of \( b_{i}^{(4)} \) are retained and the remainder set to zero. The Hamming distance between the initializing vector and \( b_{i}^{(4)} \) is 6 bits and it is 9 or more bits for the other three stored vectors. It is seen that the partial input is recognized as \( b_{i}^{(4)} \) in the third iteration and the output remains stable as \( b_{i}^{(4)} \) thereafter. This convergence to a stable state generally persists even when the \( T_{ij} \) matrix is binarized or clipped by replacing negative elements by minus ones and positive elements by plus ones evidencing the robustness of the CAM. A binary synaptic matrix has the practical advantage of being more readily implementable with fast programmable spatial light modulators (SLM) with storage capability such as the Litton Lightmod. 7 Such a binary matrix, implemented photographically, is utilized in the optical implementation described in Sec. III and evaluated in Sec. IV of this paper.

Several schemes for optical implementation of a CAM based on the Hopfield model have been described earlier. 5 In one of the implementations an array of light emitting diodes (LEDs) is used to represent the logic elements or neurons of the network. Their state (on or off) can represent unipolar binary vectors such as the state vectors \( b_{i}^{(m)} \) that are stored in the memory matrix \( T_{ij} \). Global interconnection of the elements is realized as shown in Fig. 3(a) through the addition of nonlinear feedback (thresholding, gain, and feedback) to a conventional optical vector–matrix multiplier 8 in which the array of LEDs represents the input vector and an array of photodiodes (PDs) is used to detect the output vector. The output is thresholded and fed back in parallel to drive the corresponding elements of the LED array. Multiplication of the input vector by the \( T_{ij} \) matrix is achieved by horizontal imaging and vertical smearing of the input vector that is displayed by the LEDs on the plane of the \( T_{ij} \) mask [by means of an anamorphic lens system omitted from Fig. 3(a) for simplicity]. A second anamorphic lens system (also not shown) is used to collect the light emerging from each row of the \( T_{ij} \) mask on individual photosites of the PD array. A bipolar \( T_{ij} \) matrix is realized in incoherent light by dividing each row of the \( T_{ij} \) matrix into two subrows, one for positive and one for negative values and bringing the light emerging from each subrow to focus on two adjacent photosites of the PD array. A bipolar \( T_{ij} \) matrix is realized in incoherent light by dividing each row of the \( T_{ij} \) matrix into two subrows, one for positive and one for negative values and bringing the light emerging from each subrow to focus on two adjacent photosites of the PD array. A bipolar \( T_{ij} \) matrix is realized in incoherent light by dividing each row of the \( T_{ij} \) matrix into two subrows, one for positive and one for negative values and bringing the light emerging from each subrow to focus on two adjacent photosites of the PD array. A bipolar \( T_{ij} \) matrix is realized in incoherent light by dividing each row of the \( T_{ij} \) matrix into two subrows, one for positive and one for negative values and bringing the light emerging from each subrow to focus on two adjacent photosites of the PD array.
Fig. 3. Concept for optical implementation of a content addressable memory based on the Hopfield model. (a) Matrix–vector multiplier incorporating nonlinear electronic feedback. (b) Scheme for realizing a binary bipolar memory mask transmittance in incoherent light.

monolithic structure that can also be made to contain all ICs for thresholding, amplification, and driving of LEDs. Optical feedback becomes even more attractive when we consider that arrays of nonlinear optical light amplifiers with internal feedback or optical bistability devices (OBDs) can be used to replace the PD/LED arrays. This can lead to simple compact CAM structures that may be interconnected to perform higher-order computations than the nearest-neighbor search performed by a single CAM.

We have assembled a simple optical system that is a variation of the scheme presented in Fig. 3(a) to simulate a network of \( N = 32 \) neurons. The system, details of which are given in Figs. 5–8, was constructed with an array of thirty-two LEDs and two multichannel silicon PD arrays, each consisting of thirty-two elements. Twice as many PD elements as LEDs are needed in order to implement a bipolar memory mask transmittance in incoherent light in accordance with the scheme of Fig. 3(b). A bipolar binary \( T_{ij} \) mask was prepared for \( M = 3 \) binary state vectors. The three vectors or words chosen, their Hamming distances from each other, and the resulting \( T_{ij} \) memory matrix are shown in Fig. 4. The mean Hamming distance between the three vectors is 16. A binary photographic transparency of \( 32 \times 64 \) square pixels was computer generated from the \( T_{ij} \) matrix by assigning the positive values in any given row of \( T_{ij} \) to transparent pixels in one subrow of the mask and the negative values to transparent pixels in the adjacent subrow. To insure that the image of the input LED array is uniformly smeared over the memory mask it was found convenient to split the mask in two halves, as shown in Fig. 5, and to use the resulting submasks in two identical optical arms as shown in Fig. 6. The size of the subrows of the memory submasks was made exactly equal to the element size of the PD arrays in the vertical direction which were placed in register.
of a memory submask was collected (spatially multichannel).

Fig. 4. Stored words, their Hamming distances, and their clipped $T_{ij}$ memory matrix.

Clipped memory matrix:

\[
\begin{array}{ccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
\end{array}
\]

against the masks. Light emerging from each subrow of a memory submask was collected (spatially integrated) by one of the vertically oriented elements of the multichannel PD array. In this fashion the anamorphic optics required in the output part of Fig. 3(a) are disposed of, resulting in a more simple and compact system. Pictorial views of the input LED array and the two submask/PD array assemblies are shown in Figs. 7(a) and (b), respectively. In Fig. 7(b) the left memory submask/PD array assembly is shown with the submask removed to reveal the silicon PD array situated behind it. All electronic circuits (amplifiers, thresholding comparators, LED drivers, etc.) in the thirty-two parallel feedback channels are contained in the electronic amplification and thresholding box shown in Fig. 6(a) and in the boxes on which the LED array and the two submask/PD array assemblies are mounted (see Fig. 7). A pictorial view of a composing and display box is shown in Fig. 8. This contains an arrangement of thirty-two switches and a thirty-two element LED display panel whose elements are connected in parallel to the input LED array. The function of this box is to compose and
display the binary input word or vector that appears on the input LED array of the system shown in Fig. 7(a).
Once an input vector is selected it appears displayed on the composing box and on the input LED box simultaneously. A single switch is then thrown to release the system into operation with the composed vector as the initializing vector. The final state of the system, the output, appears after a few iterations displayed on the input LED array and the display box simultaneously. The above procedure provides for convenient exercising of the system in order to study its response vs stimulus behavior. An input vector is composed and its Hamming distance from each of the nominal state vectors stored in the memory is noted. The vector is then used to initialize the CAM as described above and the output vector representing the final state of the CAM appearing, almost immediately, on the display box is noted. The response time of the electronic feedback channels as determined by the 3-dB roll-off of the amplifiers was ~60 msec. Speed of operation was not an issue in this study, and thus low response time was chosen to facilitate the experiment.

IV. Results

The results of exercising and evaluating the performance of the system we described in the preceding section are tabulated in Table I. The first run of initializing vectors used in exercising the system were error laden versions of the first word $b^{11}_1$. These were obtained from $b^{11}_1$ by successively altering (switching) the states of 1, 2, 3... up to $N$ of its digits starting from the $N$th digit. In doing so the Hamming distance between the initializing vector and $b^{11}_1$ is increased linearly in unit steps as shown in the first column of Table I whereas, on the average, the Hamming distance between all these initializing vectors and the other two state vectors remained approximately the same, about $N/2 = 16$. The final states of the memory, i.e., the steady-state vectors displayed at the output of the system (the composing and display box) when the memory is prompted by the initializing vectors, are listed in column 2 of Table I. When the Hamming distance of the initializing vector from $b^{11}_1$ is <11, the input is always recognized correctly as $b^{11}_1$. The CAM is able therefore to recognize the input vector as $b^{11}_1$ even when up to 11 of its digits (37.5%) are wrong. This performance is identical to the results obtained with a digital simulation shown in parenthesis in column 2 for comparison. When the Hamming distance is increased further to values lying between 12 and 22, the CAM is confused and identifies erroneously other state vectors, mostly $b^{11}_3$, as the input. In this range, the Hamming distance of the initializing vectors from any of the stored vectors is approximately equal making it more difficult for the CAM to decide. Note that the performance of
The complementary vectors that were not the nominal state vectors of the CAM and results of digital simulation in the range of Hamming distance are comparable except for the appearance of oscillations (designated by OSC) in the digital simulation when the outcome oscillated between several vectors that were not the nominal state vectors of the CAM. Beyond a Hamming distance of 22 both the optical system and the digital simulation identified the initializing vectors as the complement \( b^{(0)} \) of \( b^{(1)} \). This expected because it can be shown using Eq. (1) that the \( T_{ij} \) matrix formed from a set of vectors \( b^{(m)} \) is identical to that formed by the complementary set \( b^{(m)} \). The complementary vector can be viewed as a contrast reversed version of the original vector in which zeros and ones are interchanged. Recognition of a complementary state vector by the CAM is analogous to our recognizing a photographic image from the negative.

Similar results of initializing the CAM with error laden versions of \( b^{(2)} \) and \( b^{(3)} \) were also obtained. These are presented in columns 2 and 3 of Table I. Here again we see when the Hamming distance of the initializing vector from \( b^{(3)} \), for example, ranged between 1 and 14, the CAM recognized the input correctly as \( b^{(3)} \) as shown in column 3 of the table and as such it did slightly better than the results of digital simulation. Oscillatory behavior is also observed here in the digital simulation when the range of Hamming distance between the initializing vector from all stored vectors approached the mean Hamming distance between the stored vectors. Beyond this range the memory recognizes the input as the complementary of \( b^{(3)} \).

In studying the results presented in Table I several observations can be made: The optically implemented CAM is working as accurately as the digital simulations and perhaps better if we consider the absence of oscillations. These are believed to be suppressed in the system because of the nonsharp thresholding performed by the smoothly varying nonlinear transfer function of electronic circuits compared with the sharp thresholding in digital computations. The smooth nonlinear transfer function and the finite time constant of the optical system provide a relaxation mechanism that substitutes for the role of asynchronous switching required by the Hopfield model. Generally the system was able to conduct successful nearest-neighbor search when the inputs to the system are versions of the nominal state vectors containing up to ~30% error in their digits. It is worth noting that this performance is achieved in a system built from off-the-shelf electronic and optical components and with relatively little effort in optimizing and fine tuning the system for improved accuracy, thereby confirming the fact that accurate global computation can be performed with relatively inaccurate individual components.

V. Discussion

The number \( M \) of state vectors of length \( N \) that can be stored at any time in the interconnection matrix \( T_{ij} \) is limited to a fraction of \( N \). An estimate of \( M \approx 0.1N \) is indicated in simulations involving a hundred neurons or less\(^1\) and a theoretical estimate of \( M \approx N/4 \) \( \ln N \) has recently been obtained.\(^2\) It is worthwhile to consider the number of bits that can be stored per interconnection or per neuron. The number of pixels required to form the interconnection matrix is \( N^2 \). Since such a \( T_{ij} \) memory matrix can store up to \( M \approx N/4 \) \( \ln N \) \( \text{(N-tuples)} \), the number of bits stored is \( M/N = N^2/4 \ln N \). The number of bits stored per memory matrix element or interconnection is \( M/N^2 = (4 \ln N)^{-1} \), while the number of bits stored per neuron is \( M/N = M \).

The number of stored memories that can be searched for a given initializing input can be increased by using a dynamic memory mask that is rapidly addressed with different \( T_{ij} \) matrices each corresponding to different sets of \( M \) vectors. The advantage of programmable SLMs for realizing this goal are evident. For example, the Litton Lightmod (magnetooptic light modulator), which has nonvolatile storage capability and can provide high frame rates, could be used. A frame rate of 60 Hz is presently specified for commercially available units of 128 \( \times \) 128 pixels which are serially addressed.\(^7\) Units with 256 \( \times \) 256 pixels are also likely to be available in the near future with the same frame rate capability. Assuming a memory mask is realized with a Litton Lightmod of 256 \( \times \) 256 pixels we have \( N = 256, M \approx 0.1N \approx 26 \) and a total of \( 26 \times 60 = 1560 \) vectors can be searched or compared per second against an initializing input vector. Speeding up the frame rate of the Litton
Lightmod to increase memory throughput beyond the above value by implementing parallel addressing schemes is also possible. Calculations show that the maximum frame rate possible for the device operating in reflection mode with its drive lines heat sunk is 10 kHz. This means the memory throughput estimated above can be increased to search $2.6 \times 10^6$ vectors/sec, each being 256 bits long, or a total of $6.7 \times 10^6$ bits/sec. This is certainly a respectable figure, specially when we consider the error correcting capability and the associative addressing mode of the Hopfield model; i.e., useful computation is performed in addition to memory addressing.

The findings presented here show that the Hopfield model for neural networks and other similar models for content addressable and associative memories suit well the attributes of optics, namely, parallel processing and massive interconnection capabilities. These capabilities allow optical implementation of large neural networks based on the model. The availability of non-linear or bistable optical light amplifiers with internal feedback, optical bistability devices, and nonvolatile high speed spatial light modulators could greatly simplify the construction of optical CAMs and result in compact modules that can be readily interconnected to perform more general computation than nearest-neighbor search. Such systems can find use in future generation computers, artificial intelligence, and machine vision.

The work described in this paper was performed while one of the authors, N.F., was on scholarly leave at the California Institute of Technology. This author wishes to express his appreciation to CIT and the University of Pennsylvania for facilitating his sabbatical leave. The work was supported in part by the Army Research Office and in part by the Air Force Office of Scientific Research.

The subject matter of this paper is based on a paper presented at the OSA Annual Meeting, San Diego, Oct. 1984.

References

\[ s_k = i_k + g_k \]

\[ \text{sgn } i_k \]

\[ 1 \leq k \leq P \]

\[ P < k \leq L \]
<table>
<thead>
<tr>
<th>RECALL TECHNIQUE</th>
<th>MULTIPLIES/ITERATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extrapolation Net</td>
<td>$L^2$</td>
</tr>
<tr>
<td>...Outer Product Technique</td>
<td>$2NL$</td>
</tr>
<tr>
<td>Table Look-Up Net</td>
<td>$Q^2$</td>
</tr>
<tr>
<td>...Outer Product Technique</td>
<td>$2NQ$</td>
</tr>
</tbody>
</table>

Table 1.
November 5, 1986

Dr. Robert Graham
Boeing High Technology Center
Boeing Electronics Co.
P.O. Box 3707, MS 7J-05
Seattle, WA 98124-2207

Dear Rob,

Attached are two papers. The first is "A Continuous Level Associative Memory Neural Net" submitted to the Second Topical Meeting on Optical Computers at Lake Tahoe in March 1987. Although your support is greatfully acknowledged, the paper's contents are subsumed in an archival journal paper "A Continuous level Memory Extrapolation Neural O Net" submitted to Applied Optics in August of this year. This was prior to Boeing's support of our work.

The second paper, entitled "An All Optical Iterative Neural Net Recall Memory" outlines an architecture for a neural net based processor that operates at light speed. I wish to present this result at the Tahoe conference and submit the paper to an archival journal. This, of course, will be done in accordance to the "Analysis and Application of Neural Net" contract. I will assume, unless informed otherwise, that the time period for your review begins today.

Best regards,

Robert J. Marks II
Associate Professor

cc: J.A. Ritcey
    L.E. Atlas
    A. Somani
    E. Stear, Washington High Tech Center
AN ALL OPTICAL ITERATIVE NEURAL NET RECALL MEMORY

Robert J. Marks II
Interactive Systems Design Lab
Department of Electrical Engineering
University of Washington
Seattle, WA 98195
ABSTRACT

We propose an architecture for a continuous level discrete valued table look-up memory. Unlike other iterative memory recall optical processors, the processor performs at optical speeds, e.g. there are no electronics or slow optics (such as phase conjugators) in either the forward or feedback paths. Techniques to compensate for processor losses are presented.
INTRODUCTION

Hopfield's neural net content addressable memory\(^4\) has stirred a flurry of interest in the signal processing community. Optical implementations of such nets have been proposed and implemented\(^4-5\). Unlike planar VLSI, optical implementations are not restricted to nearest neighbor interconnects.

In a previous paper\(^4\) the author has described a class of neural net associative and table look-up memory algorithms based on convex set projection theory. This paper describes a processor for performing one of these algorithms. Unlike other iterative recall memories, the proposed processor operates at light speed.
PRELIMINARIES

In a previous paper, the author described a class of table look-up artificial neural networks for generating a vector in a specified library when given only a portion of that vector. We outline here one of these nets.

Let \( G = \{ \vec{f}_n \mid 1 \leq n \leq N \} \) denote a set of \( N \) real continuous level element library vectors of length \( L \). We form the library matrix

\[
E = [\vec{f}_1; \vec{f}_2; \ldots; \vec{f}_N]
\]

and the neural net interconnect matrix

\[
I = E (E^T E)^{-1} E^T
\]

Let \( \vec{f} \in G \). With knowledge of the first \( P \) elements of \( \vec{f} \), we wish to extrapolate the remainder. For a given \( \vec{f} \), define the vector operator

\[
\mathcal{N} \vec{a} = [\vec{f}_P; \vec{a}_Q]^T
\]

where \( \vec{f}_P \) denotes a vector containing the first \( P < L \) elements of \( \vec{f} \) and \( \vec{a}_Q \) contains the last \( Q = L - P \) elements of \( \vec{a} \). Then the iteration

\[
\vec{a}_{n+1} = \mathcal{N} I \vec{a}_n
\]

will, for any initialization, converge to \( \vec{f} \) if \( P > N \) and the first \( P \) rows of \( E \) form a matrix of full rank.
Some of the operations performed in (1) are not used since the \( \mathcal{N} \) operator replaces the first \( P \) vector elements with \( \hat{\mathbf{f}}_m \).

Thus, (1) can equivalently be written as

\[
\hat{\mathbf{f}}_{m+1} = \mathbf{I}_0 \left[ \hat{\mathbf{f}}_m \mid \hat{\mathbf{c}}_{m,0} \right]^T
\]

(2)

where \( \hat{\mathbf{c}}_{m,0} \) is the vector of the last \( Q \) elements of \( \hat{\mathbf{c}}_m \) and the matrix \( \mathbf{I}_0 \) consists of the last \( Q \) rows of \( \mathbf{I} \). The neuron operator, \( \mathcal{P} \), is not required in this equation since, for the last \( Q \) nodes, it is an identity operator.
The iterative neural net memory described by (2) can be straightforwardly implemented on an optical processor. While similar iterative memories have been implemented optically\(^{a-e}\), each requires either electronics or slow optics (e.g. phase conjugation mirrors) in the processor. As we will show, there are architectures for implementing (2) that are completely optical. This is because no nonlinear operations need to be performed in the processor's feedback path.

The basic processor architecture is shown in Fig. 1. The processor input corresponding to \( \tilde{f} \) is supplied by a linear array of \( P \) point source LED's. The feed-forward path consists of a standard optical matrix-vector multiplier\(^{a-e}\). (The astigmatic spreading and focusing optics have been deleted from the figure for presentation clarity.) The processor output, \( \tilde{x} \), is then fed back to the input through fibers. Once the input array is on, the iteration in (2) is thus performed at an optical speed.

The astute reader will immediately notice three fundamental problems with this processor: (1) there is no provision made for detecting the processor output (2) absorptive and other losses can significantly inhibit performance and (3) we require bipolar multiplication and addition operations rather than just the non-negative operations directly available in processors such as ours. We now address and offer solutions for each of these problems.
Although the feedback is linear, the processor is not. For any constant c, for example, $c \mathbb{N} \neq \mathbb{N} c$. The homogeneity property of linearity is thus violated. For linear processors, the mask and the input can be scaled independently. The corresponding multiplicative proportionality constant at the output is equivalently altered by either. For the processor in Fig. 1, on the other hand, we are allowed only one scaling parameter. If, for example, the mask transmittance is scaled so that no gain is required, the input LED irradiance must be similarly scaled.

We now address the problem of negative number operations in the processor. Methods of encoding both positive and negative (and even complex) number operations on incoherent algebraic processors have been proposed. Such an extension of our processor is shown in Fig 2. We decompose the $I_\alpha$ matrix as

$$I_\alpha = I_\alpha^+ + I_\alpha^-$$

where all of the elements of $I_\alpha^+$ are nonnegative and those in $I_\alpha^-$ are nonpositive. All negative elements in $I_\alpha$, for example, are set to zero to form $I_\alpha^+$. Similarly, we can write

$$\bar{f}_p = \bar{f}_p^+ + \bar{f}_p^-$$

and

$$\bar{g}_{a,m} = \bar{g}_{a,m}^+ + \bar{g}_{a,m}^-$$
FINAL REMARKS

We have proposed an architecture for an all optical table look-up processor based on a neural net model presented previously by the author. Once the optical input is made available, each iteration is performed at the time it takes light to circle the processor once.

Numerous variations on the processor are possible and are in need of further study. The optical couplers in Fig. 2, for example, can be avoided by placing fiber pairs together at the input to simulate a single point source. Also, the feedback can be performed with planar or concave mirrors rather than fibers.
fiber bundle

LED array

FIG 1
FIG. 2
A Class of Continuous Level Neural Nets and Their Optical Implementation

Robert J. Marks II
ISDL
University of Washington

Contents

1. POCS - What is it?
   - example CS's
   - projections
   - interactive POCS

2. Application to Neural Nets
   - a continuous level neural net based on POCS
   - relaxation for accelerated convergence
   - other convex constraints

3. Optical Implementation
   - fundamental architecture
   - alteration for absorbtive losses
   - alteration for bipolar operations
Q: What is POC S?
A: CS
In Hilbert space, $\mathcal{H}$, the set $\mathcal{C}$ is convex if, for $0 \leq \alpha \leq 1$,
$$\alpha \vec{x} + (1 - \alpha) \vec{y} \in \mathcal{C} \quad \forall \vec{x}, \vec{y} \in \mathcal{C}$$

i.e:

not convex:
Example Convex Sets in $\mathbb{R}^L$:

**Subspace**: Given $N$ vectors, 
\[ \{ \hat{f}_n \mid 1 \leq n \leq N \leq L \} \]
Then 
\[ C = \{ \hat{x} \mid \hat{x} = F^T \hat{\alpha} \} \]
where 
\[ F = [ \hat{f}_1 \mid \hat{f}_2 \mid \ldots \mid \hat{f}_N ] \]

**Ball**: 
\[ C = \{ \hat{x} \mid \| \hat{x} \| \leq R \} \]

**Box**: 
\[ C = \{ \hat{x} \mid | x_1 | \leq 1, 1 \leq l \leq L \} \]

**Linear Variety**: 
Given \( \{ f_1, f_2, \ldots, f_P \}, P < L \) 
\[ C = \{ \hat{x} \mid x_1 = f_1, 1 \leq l \leq P \} \]

**First Orthant**: 
\[ C = \{ \hat{x} \mid x_1 > 0, 1 \leq l \leq L \} \]

**Bandlimited vectors**: Given $P < L$ integers between $1 \leq l \leq L$:
\[ C = \{ \hat{x} \mid (D \hat{x})_l = 0 \mid l \in \text{Integers} \} \]
where 
\[ D = \text{DFT matrix} \]

etc.
POCS

\( \overset{\wedge}{x} \in C \) is the projection of \( \overset{\gamma}{y} \in \mathbb{R}^l \) onto \( C \) if

\[
\inf_{\overset{\wedge}{x} \in C} ||\overset{\wedge}{x} - \overset{\gamma}{y}|| = ||\overset{\wedge}{x} - \overset{\gamma}{y}||
\]

i.e., \( \overset{\wedge}{x} \) is the closest element in \( C \) to \( \overset{\gamma}{y} \):

Notation:

\( \overset{\wedge}{x} = P_C \overset{\gamma}{y} \)

If \( \overset{\gamma}{y} \in C \), then \( P_C \overset{\gamma}{y} = \overset{\gamma}{y} \)
Iterative POCS

★ Case 1: Intersecting CS's

\[ \text{Define } \bigcap_{1 \leq m \leq M} C_m = C \neq \emptyset \]

Then, if

\[ \tilde{z}_{N+1} = P_1 P_2 \cdots P_M \tilde{z}_N \]

we are assured that

\[ \lim_{N \to \infty} \tilde{z}_N \in C \]

Example:

Result may or may not be unique.

* Youla & Webb
Case 2: Two Nonintersecting CS's:

\[ \vec{S}_{M+1} = P_1 P_2 \vec{S}_M \]

Convergence is to:

\[ \inf \| \vec{x}_1 - \vec{x}_2 \| = \| \vec{S}_\infty - P_2 \vec{S}_\infty \| \]

That is, convergence is to the closest distance between \( C_1 \) and \( C_2 \):

Convergence may not be unique:

* Goldburg and Marks
Neural Nets: A neural net model

Let neurons, $t_{ij} = t_{ji}$

$\vec{S}_M$ = neural state at time $M$

Synchronous Operation: $\vec{S}_{M+1} = \mathcal{N} \cdot I \cdot \vec{S}_M$

$I$ = matrix of interconnects

$\mathcal{N}$ = pointwise vector operator

(performed at nodes)
A Continuous Level Neural Net

$N$ continuous level library vectors:
$$\mathcal{F} = \{ \vec{f}_n | 1 \leq n \leq N \}$$

Library matrix:
$$E = [ \vec{f}_1; \vec{f}_2; \ldots; \vec{f}_N ]$$

Projection interconnect matrix:
$$I = E (E^T E)^{-1} E^T$$

$I$ projects any $\vec{a} \in \mathbb{R}^l$ onto
$$S = [\mathcal{F}] = \text{subspace generated by } \mathcal{F}$$
The set
\[ \mathcal{V} = \{ \mathbf{x} \mid \mathbf{x} = [\mathbf{f}_P ; \mathbf{u}_Q]^T; \mathbf{u}_Q \in \mathbb{R}^{L-P} \} \]
is a linear variety.

\[ \mathbf{a} \] projects \( \mathbf{a} \in \mathbb{R}^L \) onto \( \mathcal{V} \)
Synchronous Net Operation

Let \( \vec{f} \in \mathcal{F} \), \( \vec{s}_0 = [\vec{f}_P : \vec{0}_Q]^T \)

\[
\begin{bmatrix}
\vec{u} \\
\vec{s}_N
\end{bmatrix}
= 
\begin{bmatrix}
I \\
\vec{s}_N
\end{bmatrix}
\begin{bmatrix}
\vec{f}_P \\
\vec{u}_Q
\end{bmatrix}
\]

\[\vec{s}_{N+1} = \begin{bmatrix}
\vec{f}_P \\
\vec{u}_Q
\end{bmatrix}\]

View in \( \mathbb{R}^L \):

[Diagram showing the relationship between \( \vec{s}_0 \) and \( \vec{s}_{N+1} \) through \( \vec{f}_P \) and \( \vec{u}_Q \).]
Clearly:
\[ \vec{f} \in \mathcal{V} \cap \mathcal{S} \]

Q: When is \( \vec{f} = \mathcal{V} \cap \mathcal{S} \)?
   (Then convergence is unique.)
A: Sufficient conditions:
   (1) \( P = \) number of known elements
       \( \geq N = \) number of library vectors.
   (2) \( \mathbb{F}_P = [ \vec{f}_{1P} ; \vec{f}_{2P} ; \ldots ; \vec{f}_{NP} ] \)
       is full rank.
Relaxation Parameters

Problem: Slow convergence:

A Solution: Relaxation Parameters

\[ I_r = (1 - \lambda_t) I + \lambda_t I \]
\[ N_r = (1 - \lambda_n) N - \lambda_n N \]

(neurons now have memory)

\[ 0 < \lambda_t, \lambda_n < 2 \]
Other Convex Constraints

e.g. Box \( B = \{ \hat{x} \mid \max |x_n| \leq 1 \} \)
\[ \forall \hat{f} \in \mathcal{F}, \ |f_n| \leq 1 \]

Revised operator

\[ \tilde{\mathcal{H}} \tilde{a} = [\tilde{\hat{f}}_p : \tilde{b}_Q]^T \]

where:

\[(b_Q)_n = \begin{cases} 
  a_n & ; |a_n| \leq 1 \\
  1 & ; a_n \geq 1 \\
  -1 & ; a_n \leq -1 
\end{cases} \]

View in \( \mathbb{R}^l \):

[Diagram of a mathematical illustration showing a box and various vectors and planes.]
Optical Implementation

- A table look up net:

\[
\begin{bmatrix}
\tilde{u}_p \\
\tilde{u}_q
\end{bmatrix} = \begin{bmatrix}
T_p \\
T_q
\end{bmatrix} \begin{bmatrix}
\tilde{s}_{N_p} \\
\tilde{s}_{N_q}
\end{bmatrix}
\]

\[\tilde{s}_{N+1} = \begin{bmatrix}
\tilde{f}_p \\
\tilde{f}_q
\end{bmatrix} \begin{bmatrix}
\tilde{s}_{N+1} \\
\tilde{N}_{N+1}
\end{bmatrix}
\]

If the same nodes are always used for the input (table look-up), an equivalent iteration is:

\[
\begin{bmatrix}
\tilde{s}_{N+1} \\
\tilde{s}_{N+1}
\end{bmatrix} = \begin{bmatrix}
T_p \\
T_q
\end{bmatrix} \begin{bmatrix}
\tilde{s}_{N+1} \\
\tilde{s}_{N+1}
\end{bmatrix}
\]

\[N = N + 1\]
Question #1:
(a) How do we detect the result?
(b) What about absorptive losses?

Answer #1:
Scale the mask

Question #2:
How do we handle bipolar operations?

Answer:
Separate $+$ and $-$ operations:

$$T_Q = T_Q^+ + T_Q^-$$
$$f_P = f_P^+ + f_P^-$$
$$\tilde{S}_{Q,M} = \tilde{S}_{Q,M}^+ + \tilde{S}_{Q,M}^-$$

Then:
$$\tilde{S}_{Q,M+1} = T_Q [\tilde{f}_P : \tilde{S}_{Q,M}]$$
becomes:
$$\tilde{S}_{Q,M+1}^\pm = T_Q^\pm [\tilde{f}_P^\pm : \tilde{S}_{Q,M}^\pm] + T_Q^- [\tilde{f}_P^+: \tilde{S}_{Q,M}^+]$$
Future Work:

1. Underdetermined continuous NN performance.

2. Effects of input noise and inaccurate processing on convergence.

3. Use of stochastic processing (BHTC).

4. (a.) Identification of optical architecture.
   
   (b.) Prototype.

5. Imposing other convex constraints.
made by bipolar integrated (somethings) CO. He's getting more information for us.

6800 CPU and is serial. The delta floating-point processor also uses a serial chip

P.S. Update: I just talked to a nuclear net graduate student that says the anza used an MC

Attachment

RM:ce

Professor
Robert MARVINI

Sincerely,

I'll try to find out more information.

DiGeFer with a grain of salt! However, 'Hech-Nielsen is a real snake oil salesman.'

Here is a copy of a paper outlining our complication (see the table on p. 41).

Dear HAI:

Bellevue, WA 98005
13219 Northrup Way, Suite 208
Philipp Technology Corporation

July 20, 1988

July 20, 1988

University of Washington

SEATTLE, WASHINGTON 98195
This Agreement shall remain in force and effect for one (1) year from the effective date hereof, except to the extent of the provisions hereof which are not in accordance with law.

This Agreement was entered into in accordance with Paragraph (2) above. The exchange of data shall be determined by the date attested below.

Date: 12/20/28

Title:

By: [Signature]

Les E. Alizes

[Company Name]

Washington, D.C.
Non-Disclosure Agreement

This Agreement shall remain in force and effect for one (1) year from the effective date hereof, except to the extent provided in paragraph (2) above. The Executive shall provide in triplicate (2) copies of this Agreement. This Agreement shall be governed by and construed in accordance with the laws of the State of Washington.

1. Any conceived of or claims arising out of or relating to this Agreement or the breach thereof, shall be settled by arbitration in accordance with the Commercial Arbitration Rules of the American Arbitration Association. The decision of the arbitrator in any such arbitration shall be final and binding on the parties. The costs of arbitration shall be shared equally by the parties.

2. The Executive shall remain in force and effect for one (1) year from the effective date hereof, except to the extent provided in paragraph (2) above. Each party agrees to keep confidential and not disclose to any third party any information received from the other party in connection with this Agreement. The information shall remain the property of the Executive. The Executive shall not use or disclose such information for any purpose unless expressly authorized in writing by the other party.

3. If any provision of this Agreement shall be held to be invalid, illegal or unenforceable, the validity, legality and enforceability of the remaining provisions shall not be affected or impaired thereby.

4. Confidential information shall mean: (a) information which the Executive has made available to the Executive in confidence and is marked as confidential or otherwise identified as confidential by the Executive; and (b) all information which the Executive has received from the Executive in connection with the Agreement.

5. This Agreement shall be governed by and construed in accordance with the laws of the State of Washington.

6. This Agreement shall remain in force and effect for one (1) year from the effective date hereof, except to the extent provided in paragraph (2) above. The Executive shall provide in triplicate (2) copies of this Agreement. This Agreement shall be governed by and construed in accordance with the laws of the State of Washington.

7. Any conceived of or claims arising out of or relating to this Agreement or the breach thereof, shall be settled by arbitration in accordance with the Commercial Arbitration Rules of the American Arbitration Association. The decision of the arbitrator in any such arbitration shall be final and binding on the parties. The costs of arbitration shall be shared equally by the parties.
Date 9-30-88

The Contracting Officer

By

Pacific Northwest Laboratories

Non-Disclosure Agreement

Pacific Northwest Laboratories
Battelle Memorial Institute

Sealed as follows:

A. All discussions of confidential information will be written and marked "Confidential" at the time such whispers.
AGREEMENT

We, the undersigned, Pieter J. van Heerden, Robert J. Marks II, and Seho Oh agree that the owner's rights and the financial benefits of the patent we will apply for "A Computer Chip Realizing Learning in a Digital Computer," will be distributed in the following manner: P. J. van Heerden 60%, R. J. Marks 20%, and Seho Oh 20%.

This means, of course, that the cost of obtaining the patent, specifically the cost of the patent lawyers, will be subtracted from the income from potential licensing rights and the sale of the patent.

Signed:

P. J. van Heerden
March 30, 1988

R. J. Marks II
April 4, 1988

Seho Oh
April 4, 1988
Introduction

The invention relates to a computer chip which is the 'brain' of a digital computer which can learn, that is improves and perfects its performance from previous experience. The computer may be designed to operate any kind of mechanical or electronic equipment normally operated by human beings. If we call the learning to improve the operation "Intelligence," then we may call the computer an intelligent machine. Intelligence is a quality observed in living beings, humans and animals. Since, in learning, the machine imitates the learning behavior of living beings, its operation is based on a theory of human and animal behavior. This is the theory of psychology of William MacDougall, given in his book "An Introduction to Social Psychology" Barnes and Noble, N.Y. 1960 (originally 1908). This book is out of print, but the present champion of this theory is Margaret Boden with her book "Purposive Explanation in Psychology" Harvard Un. Press 197....

The psychological theory is that man or animal does everything with a purpose. They have drives or instincts which want to be satisfied. The simplest example is hunger. Hunger drives the intelligent being to seek means to satisfying its hunger, by eating. How else would an animal know that, speaking physiologically, its body needs food to stay alive? Internal observations on the body, which could be for instance cells which measure the sugar content of the blood, are communicated, as a communication channel, to the brains. The measurement of a low-level of the sugar content of the blood results in a feeling of hunger.
MacDougall hypothesis is that man has, besides hunger, a whole spectrum of drives which want to be satisfied. These drives take care, not only of the bodily needs to the living being, but also of its social needs. Examples of these drives are: The curiosity propensity, which is the instinct to explore strange places and things; the self-assertive propensity, the instinct to domineer, to lead, to assert oneself over, or display oneself before one's fellows; the submissive propensity, the instinct to defer, to obey, to follow, to submit in the presence of others who display superior powers; the gregarious propensity, the instinct to remain in the company with fellows, and, if isolated, to seek that company; the anger propensity, which is the instinct to resent and forcibly break down any thwarting or resistance offered to the free exercise of any other propensity; the fear propensity, the instinct to flee for cover in response to violent impressions that inflict or threaten pain or injury; the constructive propensity, the instinct to construct shelters and implements; the acquisitive propensity, the instinct to acquire, possess, and defend whatever is found useful or otherwise attractive; the sex propensity, the instinct to court and mate; the parental or protective propensity, the instinct to feed, protect and shelter the young; the laughter propensity, the instinct to laugh at the defects and failures of our fellow creatures. In Boden are listed 18 propensities, which list is pretty complete, but obviously not claimed to be exhaustive or final. Many of these instincts are readily observed in animals, and, just like hunger, it is hard to imagine how the individual would survive without having these instincts. P. J. van Heerden has translated this psychological theory into a quantitative mathematical theory of intelligence. "The Foundation of Empirical Knowledge, with a Theory of Artificial Intelligence" Wistik, Wassenaar, Netherland 1968. The book is out of print, but available in many college libraries.

The brain is postulated to be a computer. Its only function is to process mathematically the input of information into an output. The input information consists of two kinds of information channels. The first kind is a spectrum of drives;
they are all of the same kind. They represent functions of time, $f_1(t)$, which represent some internal observation of an aspect of the state of the body, like hunger observes that the body needs food. In a primitive animal, the needs are few. In an animal of higher intelligence, these needs are differentiated and more refined. But their mathematical representation is always the same, $f_1(t)$, a drive which drives the individual to action, to satisfy that drive.

The second kind of input information channels, $f_2(t)$, is from the senses like eye and ear. It is obvious that without eyes and ears, the individual could never carry out intelligent actions. The senses therefore, in general, are observations on the state of the outside world which help the instincts $f_1(t)$, which are observations on the internal world of the individual, to satisfy their purposes. The output information $f_3(t)$ is of one kind. It gives commands, through the nerves, to the muscles of the body, of hand, foot and mouth. Speaking is also action, and it is hard to imagine intelligent human life without this means of communication with its fellow men. Of course, in our society, it is partly replaced by muscle action of the hand in writing.

Consequently, the mathematical theory translates the psychological theory of MacDougall into a universal quantitative theory of intelligence, and this theory is valid for all intelligence, whether in man, animal or machine, the brain is an organ, like the heart is an organ which pumps blood. The brain is a computer. It takes the input information, the drives and the senses, $f_1(t)$ and $f_2(t)$, and processes this information to arrive at an output, which is a command to the muscles to action. This command to the muscles has always only one purpose, which is to satisfy, silence, the drive which happens to be active. The sense information, $f_2(t)$, helps in performing intelligent action, that is action which leads to a better, easier, quicker satisfaction of the drives. In the case of a machine, the mathematical theory simply imitates the behavior of intelligence which we observe in man and animals. The output function $f_3(t)$ can be a typewriter, but in general any kind of machinery which one wants
the computer to drive. The input functions \( f_2(t) \), the senses, are any kind of information about the outside world one wants to make available to the computer. In the most advanced machine intelligence, this could be a television camera to see, and a microphone to hear. The functions \( f_1(t) \), the drives, are conceptually the most difficult part of the machine. The builder of the machine of course wants to put in functions \( f_1(t) \) which satisfy the builder's purpose, but makes the machine so that it works independently, without further instructions. The more refined the drives, the higher the intelligence the machine will be able to realize. Let us say that we consider the design of the functions \( f_1(t) \) as an open art, which requires psychological insight in what motivations lead to higher intelligence in humans. But, to construct machines, we should be able to give a simple function \( f_1(t) \) which accomplishes the basic goal of a primitive intelligence. This is the element of reward and punishment, without which no intelligent being could survive. And we know of no better way of demonstrating this than in an eating experience, of humans or animals. When we see an apple, and we know apples as delicious to be eaten, and as satisfying, stilling our hunger drive, we are inclined to take a bite. If we take a bite, then the first bite gives us the delightful taste of the apple juices flowing in our mouth. This experience encourages us to take a second bite. On the other hand, a bite of an apple which, by some treatment, is foul tasting, or is rotten, or has a worm in it, it will cause us to spit it out, throw the apple far from us, give us a warning to look carefully at an apple before we take a bite. This is the clearest, simplest example of reward or punishment in human situations which requires no further explanation. It is a rock-bottom experience. Yet, when we think about putting the principle in a machine, we are stopped in our tracks: how do you reward a machine, how do you punish a machine? Unless we can do this, we cannot build an intelligent machine. We want to propose a way in which this simple human, or animal sensation is realized in a machine. We think that in humans or animals, the reward causes a feeling of well being, which at all times encourages muscle actions, while punishment causes a feeling of diminished well
being, which therefore discourages muscle action. In all our actions, there is a desire to act before it reaches a level of actions. Humans and animals always have a certain caution, a feeling that action might be harmful, dangerous. Unless the individual has a high confidence level, partly caused by a clear signal from the eyes and ears that nothing is wrong, that the situation is clear cut, the individual will not act. An example is that at night, under circumstances of less visibility, we won't drive our car with as much confidence as in daylight. Another example is that one wants to ask a question at a public meeting, but hesitates to ask it because one thinks one has not properly understood the situation, and might make a fool of oneself. Confidence, desire for action, is partly caused by internal factors, for instance feeling robust and healthy, partly by a clear recognition, by the senses, of the situation one is in. We therefore want to propose a state of the body always present, of a bias level or rather the reverse, a "boldness level," that discourages action when it is low, and encourages action when it is high. It is like the grid voltage on a power tube which operates the motion of a power tool. At high voltage, the power tube works at full power, at low voltage, the current in the tube is cut off, the tube does not generate power.

This "boldness level," which like all our mechanical descriptions of humans, is felt as a sensation, is raised by a positive experience, that is felt as a reward; it is lowered by a bad experience, that is felt as a punishment. A bite of the apple, which tastes good, heightens the activity of the muscles in what we are doing, eating. A bad tasting bite lowers the activity of the eating muscles. So the taste cells in our mouth and nose, we hardly realize this in everyday life, form an essential mechanism given us by nature to raise our intelligence, because it increases our power of discrimination between good and bad. So, in general, such observation cells, which discriminate between good and bad action, are essential in an intelligent machine.

In teaching humans, we use reward and punishment by a show of either approval and affection, or disapproval. There is no
reason why in teaching intelligent machines we cannot use the same principle. A proper operation raises the boldness level; a bad operation, a barrier, a wrong executed motion, lowers the boldness level. But, also a simple push button, operated by the teacher, raises the boldness level. Another button lowers the boldness level.

So, in general, we must realize that in all human intelligent action, there are present multiple drives. While in eating the general drive is hunger, the very pleasure of eating something wholesome, appetite encourages the action of eating. In reaching out for instance to grasp an object, the brain cannot give precise instructions to the muscles, because the distance and the shape of the object is unknown. The eye monitors the movement of the arm to the object, and when the hand touches, the feeling of touch guides the muscles of the fingers to perform the proper grip action. This may signify that there is a certain innate pleasure of gripping, similar to appetite in eating, which has to be simulated in a machine to achieve proper learning. In the same way, in speech, hearing one's own speech, and the pleasure derived from it, may be an essential element in proper speech, and therefore has to be incorporated, simulated, in a machine.

In all human intelligent actions we have this principle that action of the muscles cannot be prescribed in detail at the beginning of the action, but must be specified in the course of the action by closer observations, and, as we have shown this may require the postulation of special appetite-like mechanisms. One may compare this with the order of an army general to move his army forward to engage and defeat the enemy. This order is intelligent action, motivated by his sense of patriotism, loyalty to his country, or personal ambition. However, he cannot prescribe to the individual platoon, or to the individual soldier how he has to move forward, since this depends on the exact terrain, and the conditions under which the soldier operates. There must therefore be a detailed motivation of the soldier, a desire for individual combat, a desire to avoid enemy fire, a willingness to obey orders, or, at the lowest level, to put his
foot forward without stumbling. But all this detailed action is integrated in the total intelligent action of the general, to defeat the enemy.

This discussion shows clearly that, before we can properly describe intelligent action in detail, and therefore incorporate its imitation in machines, a lot of thought and experimentation will still be necessary. However, this does not influence in the least the nature and purpose of our invention, which merely is aimed at finding the information which is appropriate to the situation at hand, no matter what the instincts, no matter what the information from the senses, no matter what the muscle action. We claim that intelligence is nothing but the modification of the behavior, from the automatic reactions of a newborn babe to its inborn instincts, by its life experiences that its instincts, like hunger, can be satisfied in different ways. Learning is always choosing the best experience in the past, and of the past, those experiences are stored in the permanent memory which have best satisfied the inborn instincts. The object of our invention therefore is merely a circuit which can carry out a search of the past experience, as fast, and therefore through as many parallel channels, as possible. Such a circuit, for information in the binary form, can be made on the surface of a silicon crystal, by the modern, conventional methods used for instance in the well-known manufacturing methods of a RAM device. The methods are micro photography of the circuit, or parts thereof on a photo-sensitive layer on the surface, etching, oxidation, vapor deposition or vapor reaction, etc. The idea is simply to process information in as many parallel channels as possible.

To sum it up, we believe that intelligent action in animals and man is not based on a mysterious, magic principle but on learning, that is rational information processing. It is only the enormous amount of information processing involved in the brain which is difficult to comprehend. Certainly the details of the kind of information, and information processing, and in particular, the information from drives and senses, will require a great deal of experimental and theoretical research. But it
seems clear that at the center of the operation of intelligence lies the fast search for accurate information of positive experiences of the past, and that the circuit of our invention carries that out in a machine. How far the operation of such a circuit alone will go in achieving intelligence in a machine, whether only of a simple nature or advanced, only future investigations can show.
The Mathematical Principle Involved

The mathematical principle involved in the theory was described in the book "The Foundation of Empirical Knowledge" mentioned before and realized in the optical machine described there and in two U.S. patents of P. J. van Heerden, #3,296,594 and #3,492,652. It is basically the principle that at every moment of the life of an intelligent individual the brain carries out a rapid search in its permanent, or temporary, storage for that positive life experience which most closely matches the present situation it finds itself in. If the information is given in the form of three functions \( f_1(t) \), \( f_2(t) \) and \( f_3(t) \), mentioned in the introductions, then this search produces automatically the function \( f_3(t) \), the muscle commands, for hand, foot or tongue, which the individual needs in the present situation. This muscle motion \( f_3(t) \), which, in a specific case for human beings, may be nothing more than speaking the right words, as learned from a previous experience, is now produced automatically. It is but the muscle motion \( f_3(t-k) \) which was successful in a previous satisfying experience \( k \) units of time ago. It is the claim of the theory that all muscle motions are learned by practice as successful in satisfying the drive function \( f_1(t) \).

The optical machine, with a hologram, described in the book mentioned and the patents, formed a fast and accurate way in which this rapid search for matching information could be carried out in a large memory. It was believed, at the time, that an equivalent search for information could never be carried out by a digital computer. At present, because of the great development of making complicated digital circuitry on the surface of a semiconductor, the digital computer has the capability of matching the performance of the hologram principle in the optical machine. Stating it mathematically, if the number of binary digits equivalent to the information storage in a memory of an intelligent individual is the number \( n \), then the number of elementary algebraic operations for search (like \( A+B=C \), when \( A, B \) and \( C \) are binary digits) in a life time is \( n^2 \). At present, the informa-
tion stored in the human brain in a lifetime is estimated of the order of $10^9$ binary bits (Th. K. Landauer, Cognitive Science 10, 477, 1986). This means that $10^9$ to $10^{10}$ elementary binary operations have to be carried out every second. With a clock time of a micro second ($10^{-6}$ sec), $10^6$ such operations can be carried out in one channel per second, and therefore a computer chip with $10^4$ parallel channels is necessary for carrying out the equivalent of the information processing that goes on, according to our theory, in the human brain.

The complexity of the circuit of our invention, described here, necessary to carry out this amount of information processing is estimated to be like that of a 400K RAM, which units at present are manufactured on a large scale. The circuit of our invention therefore can be manufactured in the conventional ways of making the complex circuitry required for modern computers. Improvements in technology, leading to more parallel channels, and a faster clock time, will improve machine operations.

The Circuit

The circuit has as an input one binary time series $f(t)$, which therefore is a series of standard pulses, which represent a "one," or the absence of a standard pulse, which represents a "zero." However, according to our mathematics, this procedure can also be reversed, in that a "zero" can be represented by a standard pulse, and a "one" by the absence of a standard pulse. The function $f(t)$ is periodically divided in the three functions $f_1(t)$, $f_2(t)$ and $f_3(t)$ mentioned before, so that for instance a fixed period of a sequence of pulses represents $f_1(t)$, a second sequence represents $f_2(t)$, and a third sequence represents $f_3(t)$, a fourth sequence represents again $f_1(t)$, a fifth sequence $f_2(t)$, a sixth sequence $f_3(t)$, a seventh sequence again $f_1(t)$, and so forth, so that $f(t)$ represents the full history of positive experiences of the machine. This history of experiences of the machine will be stored, temporarily or permanently, on a magnetic tape or disc or other electronic storage medium. As we will see,
memories of a different content will have to be present in the machine, and our invention does not cover the wiring of together of these memories. Our invention only covers the circuit on the chip, which is the same independent of the kind of information it processes. It is always involved in search for the best match with the content of the memory.

The complete function \( f(t) \) is fed into the circuit on the chip, and the operation of the circuit is to form the binary functions \((1+D_1) f(t), (1+D_2) f(t), (1+D_3) f(t), \) and so on, in general \((1+D_k) f(t)\). Here "\(+\)" for the binary function, means "plus modulo two": \( 1+1=0; \ 1+0=1; \ 0+1=1, \ 0+0=0, \) and \( D_\) the so called "Huffman operator," is defined as \( D^1 f(t) = f(t-1), \ D^k f(t) = f(t-k) \). Here the limit of time is chosen the clock time, the time in which a pulse, or "absence of a pulse," repeats itself. Therefore \((1+D^k) f(t)\) represents the new function \([f(t)+f(t-k)]\).

The operation of the circuit is now to select that function \((1+D^{k*}) f(t)\) which produces the highest percentage of zero's (over one's) in the most recent past interval, the length of which interval can be specified by appropriate circuitry and may be variable. This selection of the function \((1+D^{k*}) f(t)\) is realized in the circuit drawings by the method used for selecting the best player in tennis tournaments. One first matches each two players, and then matches the winners again two by two, and so on, until finally one winner emerges of the tournament. So, in the circuit of our invention, each pair of adjacent functions, \((1+D^k) f(t)\) and \((1+D^{k+1}) f(t)\) are compared on this excess of zero's content, by having a counter for each function counts the excess of zero's content, and then have the highest excess of zero's counter determine the switch setting. The "winner" is thus admitted to the "next round," and the winner of the "pairing of winners of the first round" is determined by the B- or comparator circuits. These comparators B again operate switches to admit the winners to the next round. Finally, one winner \((1+D^{k*}) f(t)\) emerges as the one who has produced the largest excess of zeros of all the circuits \((1+D^k) f(t)\) on the chip.
This function is added, in the binary way, to f(t), according to the formula \((1 + D^k) f(t) + f(t) = D^k f(t) = f(t - k)\). The part \(f_3(t - k)\) represents the "muscular" output of the machine. It operates the mechanical or electric apparatus one wants the intelligent computer to operate.

However, there is a mismatch of the operations of the computer circuit and the operation of intelligence in man and animals. This mismatch is the fact that a computer may have a clock time of one micro second, and processes information at a speed of \(10^6\) digits per second, while the human intelligence receives only a fraction of \(10^6\) digits per second. Let us say \(10^4\), \(10^3\), \(10^2\) or less digits. That would mean that the circuit would have to be idle the larger part of the time. Without further tricks, to match the inherent speed of the computer with the estimated capability of the human brain to process \(10^9\) to \(10^{10}\) digits per second, the very speed of the computer would be useless.

Therefore, some of these circuits present in an intelligent machine work not in real time, but from a memory of storage of past information, call it \(g(t)\), to differentiate it from the real lifetime experiences \(f(t)\), in such a way that it scans successive segments of \(g(t)\), given by a segment of pulses \(g(t-a)\) to \(g(t)\), \(g(t-2a)\) to \(g(t-a)\), \(g(t-3a)\) to \(g(t-2a)\), \(g(t-4a)\) to \(g(t-3a)\) and so on. However, the winner \(f(t) + D^k g(t)\) in each segment, has to be added to \(f(t)\) to give an output \(D^k g(t) = g(t - k)\). Therefore, at the point in the circuit \(T\) we have for every segment repeat the function \(f(t-a)\) to \(f(t)\), while in the shift register \(D\) should appear, successively, the segments of \(g(t)\), to wit \(g(t-a)\) to \(g(t)\), \(g(t-2a)\) to \(g(t-a)\), and so on. Clearly, we can only achieve a smooth operation if the number \(a\) (in units of time of one micro second) is equal to the number of processing units in the shift register, on a simple fraction there of \((1/2, 1/3, 1/4, \text{ etc.})\).
Discussion

We must realize that in achieving human intelligence - and even much more primitive intelligence in animals - in computers a great deal of experimenting and thought will be necessary. And the kind of experimenting involved will have to be both of an engineering nature, in the way the circuits execute their purposes, and the way they are connected in the general organization of the intelligent machine, and of a scientific nature, on what kind of drive - and sense - information is conducive to develop intelligent behavior.

For instance, one must realize that in intelligent action different kinds of intelligence are involved (as in our example of the general and the soldier), which therefore require different kinds of information storage, permanent or temporary, and circuits searching them. The claim is however that in this general organization of an intelligent machine the circuit of our invention plays the essential role. The action of the circuit is to search fast for that kind of information that is applicable to the present situation. That is all it does, and that is, as is claimed by our theory, the basic element in developing intelligence. No doubt, by this principle, the machine will learn, since it will recollect past positive experiences, and apply them to the present. Like in all scientific theories, future experiments with these circuits will show us the level of intelligence that can be reached.

Intelligence is learning by experience; that is learning by experience that kind of actions - muscular activity, including speech - which satisfies the drives Nature has endowed us with. These drives of course are "survival instincts." They are necessary for the individual to survive, and thrive, in its surrounding. And this surrounding can be its group, its tribe, its society. No doubt, life developed these drives in the millions of years of evolution of life, and its changing circumstances.
We think that all drives the baby has at birth are accompanied by an automatic response, in muscle actions. When a baby is hungry, it cries; when it is offered the mother's breast, it sucks. Then, in the course of time, it discovers, by experience, that there are other ways to satisfy its hunger.

While, in the life of the intelligent individual, the response to a drive has to be modified, to reach the age of a mature individual, and some drives may develop in adolescence, other drives present in the baby stage may not require modification. Let me give two examples. When an object comes close to the eyes, or touches the eyeball, we will automatically blink to protect the eyes. This is an automatic and intelligent response which does not need modification (except in exceptional cases, like a prize fighter who is taught not to blink when he sees a fist coming). When we touch a hot object with the fingers, so we burn ourselves, we will automatically pull back our hands. This is again an automatic response which does not need modification. P. J. van Heerden, in his book, has also pointed out that learning to see what it sees, a baby requires the curiosity drive from birth to direct the eyeballs, and focus the eye lenses, to an object appearing in its field of view. All these mechanisms may be imitated in machines, or mechanisms may be invented to serve the particular purpose.

It is clear that those inborn intelligent responses, and also the learned intelligent responses, like in speech, or moving the hands and fingers as we learn it in the crafts, professions and sports, a fast response is necessary. To scan the full stored intelligent memory in that short a time is physically impossible. A limited search, through a smaller memory, is necessary. This makes it clear that in intelligent machines, as in intelligent living beings, several kinds of memory storage are necessary, in which also different scanning times, and zero count integration times, are required. For instance, in speech, the amount of information flowing from our lips may be expressed in hundreds or tens, of binary digits per second. According to our theory, this is accomplished by a circuit with a fast counting integration
time scanning a small memory containing only words, and short sequences of words, of the language. But it is hard to imagine that this fast circuit also would contain the main purpose a person has with his conversation. Therefore, one must imagine that speech is guided by several circuits, or rather (since circuits only deal with intelligence in machines), one should say several memories. One memory controls the details of correct speech, from a limited content, and one which controls the overall purpose of the conversation one has. In the efforts to build intelligent machines, this must be taken into account.

To sum it up, our theory of intelligence is like a theory of human flight. The Wright brothers proved that flight is possible. The essential elements were: a wing to support the airplane, a motor-driven propeller to give it speed, and a steering mechanism to guide its motion. However, nobody could build a Boeing 747 in 1905. That would take humanity 70 years of learning. But, the same elements used by the Wright brothers are still the elements of flight: wing, motor and steering, except that now we use jet engines. In the same way, our theory of intelligence says that three elements are necessary to make intelligent machines: drives, senses and muscles, and that the essential operation is learning from past experience. In the digital computer, this learning is carried out in circuits of our design, and the availability of these circuits will be vital for developing intelligence in digital computers.
November 4, 1988

Dr. Pieter J. van Heerden
18217 - 145th Court, N.E.
Woodinville, WA 98072

Dear Pieter:

I hope the enclosed meets the needs of the patent attorney. If not, please let me know.

Best personal regards,

Robert J. Marks II
Professor

RJM:cc

Enclosure

cc: S. Oh
ELECTRONIC CIRCUITS
Digital and Analog

CHARLES A. HOLT
Virginia Polytechnic Institute and State University

JOHN WILEY & SONS, New York • Chichester • Brisbane • Toronto
that the $R$ input is inverted ahead of the $K$ terminal. This feature is useful in counting and sequence-generation applications. When the $JK$ inputs are connected together, the first stage is a D flip-flop, which is examined in the next section. All flip-flops are master-slave (MS) types with static operation, in contrast to the dynamic mode of multiphase configurations such as that of Fig. 8-7 of Sec. 8-3.

9-4. CMOS FLIP-FLOP CIRCUITS

D FLIP-FLOP

RS, JK, and T flip-flops have been examined. Another configuration of importance is the $D$ type, with $D$ representing delay. The output after a clock pulse equals the input before the pulse. In Fig. 9-15 are shown the symbol and characteristic table. Optional are the clear-preset inputs and the complement $Q$ of the output. A D flip-flop can be made from a $JK$ flip-flop by connecting the $J$ and $K$ inputs, with the connection serving as the data input. When periodic pulses are applied to the clock input, the output is that of the input delayed by one clock pulse.

![Figure 9-15 D flip-flop and characteristic table.](image)

A clocked $D$ latch differs from a D flip-flop in that the one-bit delay is eliminated. The network is designed so that when the clock pulse triggers the gate, the output is coupled directly to the input $D$, and $Q$ equals $D$. The output is then held, or latched, in this state until the next pulse triggers the gate. The clock simply acts as an enable input to the latch. It has important applications in registers, especially for temporary data storage.

The logic diagram of a CMOS clocked D latch is shown in Fig. 9-16. Transmission gate $TG_2$ turns off, and $TG_1$ then turns on, during a clock-pulse rise from low to high. The reason $TG_2$ turns off is to prevent the output at $Q_2$ from interacting with the data input. When $TG_1$ turns on, the input bit enters
the latch. This stored bit appears at the buffered output terminal \( Q \) with very little delay. The propagation delays of the inverters are small compared with the clock period.

When the clock pulse drops from high to low, \( TG_1 \) cuts off, \( TG_2 \) then turns on, and the bit remains stored until the next pulse appears. The reason for the inclusion of inverter 2 and \( TG_2 \) is to maintain the proper stored charge on the insulated gate terminals of inverters 1 and 4. If they were eliminated, any charge stored on these insulated gates would soon be lost by leakage. With \( TG_2 \) on, inverters 1 and 2 constitute a cross-coupled latch. Transmission gates are used instead of NOR gates to control the operation. The use of two inverters for the clock circuitry provides buffering to reduce the loading of the clock and to improve the pulse waveforms.

Integrated circuit CD4042A is classified as a CMOS quad clocked D latch. It consists of four separate D latches, each strobed by a common clock. The configuration is that of Fig. 9-16. A polarity circuit of two cascaded inverters can be used to program the pulse transition, either positive or negative, that switches the output. The gate propagation delay is typically 50 ns with a 10 V supply and a load capacitance of 15 pF, corresponding to a fan-out of three. In the low state the gate can sink about 2 mA while maintaining an output less than 0.5 V, and in the high state it can supply 2 mA with the voltage held above 9.5. A toggle frequency up to about 8 MHz is reasonable. Typical applications include buffer storage and use as a holding register in digital systems.

A D type master-slave (MS) flip-flop can be made simply by cascading two D latches of the form of Fig. 9-16, with the transmission gates clocked so that only one latch receives data at a time. By replacing inverters 1 and 2 of each latch with NOR gates, preset and clear controls can be added, which are often referred to as set-reset controls. Such a configuration is shown in Fig. 9-17, along
Figure 9-17 Logic diagram of D-type master-slave flip-flop.

Figure 9-18 D-type master-slave flip-flop circuitry.
with the truth table. When a clock pulse rises from low to high, which is a positive transition, the logic level present at the \( D \) input becomes the \( Q \) output.

Data enter the master on negative transitions and are transferred to the slave on positive transitions.

The first two rows of the truth table are those of a \( D \) flip-flop, with the symbols under the clock column indicating the level change at which the \( D \) input becomes the \( Q \) output. The bottom three rows simply represent the truth table for the case in which one or both of the preset-clear inputs is a logical 1. The states and clock transitions marked \( X \) have no effect on the output. These are referred to as don't-care conditions. When a logical 1 is present at a preset or clear input, the output is independent of the data input and the clock pulses.

### CMOS FLIP-FLOPS

Circuitry accomplishing the logic of Fig. 9-17 is shown in Fig. 9-18 with each gate identified. Both the numbers and the relative positions of the gates of Fig. 9-18 correspond with those of Fig. 9-17. Note the symbol used for the transmission gate. Because transmission through a gate is possible in both directions, the position of the gate terminal is centered. All transistors are enhancement-mode devices.

Integrated circuit CD4013A consists of dual \( D \) type flip-flops. Each of the two identical flip-flops has the circuitry of Fig. 9-18. Operation is static, rather than dynamic, with the state of the flip-flop retained indefinitely when the clock input is constant at either a high-level or low-level voltage. A toggle rate of about 8 MHz is typical with a 10 V supply, and the respective high-level and low-level output impedances are typically 400 and 200 ohms. The dc supply \( V_{DD} \) should be between 3 and 15 V. By connecting the \( Q \) output to the \( D \) input the flip-flop toggles at each clock pulse. Applications include shift registers, counters, and control circuits.

A \( D \) flip-flop can be converted to a \( JK \) configuration in a number of ways, one of which is shown in Fig. 9-19. In addition to the logic arrangement, the figure includes the characteristic table, assuming zero preset and clear inputs. Let us consider the first row of the table. With the present state of \( Q \) equal to 0 and the input at \( J \) a logical 1, the outputs of gates 1 and 2 are logical zeros regardless of \( K \), which is a don't-care state. Therefore, the output of NOR gate 3 is 1. As this is the \( D \) input, \( Q \) becomes 1 after the positive pulse transition. Verification of the other rows is left as an exercise. The process is simplified by recognizing that the output \( D \) of NOR gate 3 is given by

\[
D = KQ + J + Q
\]

with \( Q \) denoting the present state. Output \( D \) is the next state of \( Q \).

Figure 9-20 shows suitable circuitry that performs the logic of (9-1), along with the proper connections to the \( D \) flip-flop. All p-channel transistors have a
TO: Prof. Tom Seliga, Chair
   Department of Electrical Engineering

   Dr. Ray Bowen, Dean
   College of Engineering

   Prof. Ed Stear, Director
   Washington Technology Center

   Peter Odabashian, External Affairs Director
   Washington Technology Center

FROM: Robert J. Marks II

SUBJECT: Patent Disclosure

This attached disclosure requires approval from Profs. Seliga, Bowen and Stear. If approval is given, please forward this memo to the next person on the list. Otherwise, it should be returned to me.

The subject of this disclosure is a computational procedure to adapt training in an artificial neural network to nonstationary training data. The technology was developed by Prof. El-Sharkawi, Mr. D.C. Park and me. The work was performed under the sponsorship of Puget Sound Power and Light Company and was motivated by the need to adapt load forecasting to the changing load profiles. The adaptive technique, however, is potentially applicable to a large number of similar problems where the training data for the neural network comes from a slowly varying nonstationary process.

cc: Prof. M. El-Sharkawi
    D.C. Park
    M.L. Bruce, Puget Sound Power and Light Company
Additional fault tolerance may be inherent in the ANN algorithm.

Implementation and include:

- Other positive attributes of the VANN are those normally associated with digital
- In complexity,
- In software,
- In choice of algorithm,
- Repeatability
- Density factor
- Programmatically interconnected density capability (and thus an extremely high speed
- ARCHITECTURE, FULL TOLERANCE
- Modular structure
- Required electronics are currently available

Networks (VANN's) with the following characteristics:

- We propose development of an electronic architecture for volumetric artificial neural
- This white paper outlines a method for overcoming these problems. Specifically,
- The approach, the plan of approach to both serial digital and analog VANN VLSI
- Currently used ad hoc and learning algorithms (e.g., back propagation [5-8]),
- High speed VANN's are not available in two.
- The high-concurrent available in three dimensions is clearly
- The biologically inspired neural systems. However, there are numerous parallel and serial implementation severely degraded potential
- The serial electronics have been primarily marketed as simulation tools, ANN's,
- Have used high speed parallel analog electronics.
- Frequently used conventional high-speed serial computers designed on a
- Circuit implementation electronics, most efforts to date have either
- Artificial neural networks (ANN's) attempt to simulate the architecture and

Introduction

There are nine spaces plus six figures in this document
88/12/71
H. Phillips and R. J. Marks II
By
a White Paper
Algorithm programmability
Non volatility
Ease in establishment of architecture fault tolerance
No thermal drift in operating characteristics

The observations to this point strongly suggests a digital three dimensional ANN as the preferred architecture for adaptation and learning. The remainder of this white paper addresses more in detail how a VANN meets these objectives.

Volumetric Artificial Neural Network Description

Architecture

The VANN architecture is based conceptually on a cellular building block approach. The basic construction element is three-dimensional. Such a neural cell is most easily visualized as use a cube, but other arbitrary three-dimensional shapes (such as are found is crystal lattices) can also be used. A hexagonal cell, for example, is shown in Figure 1. Each cell contains a processing element such as a microcomputer and, in general, has the ability to simulate a number of neurons. A cell is directly connected electrically to each cell to which it is in physical contact. These connections carry information relating to the state of one or more neural cells, plus electrical power to permit the cells to function.

These cells may be stacked in volumetric fashion, e.g. the 8x5x4 cubic array as shown in Figure 2. Other arbitrary stackings may be obtained by simply ordering cubes differently. Nor is it necessary to have three stacking dimensions; an array could be laid out as a planar geometry, for example as simply 5x5x1, or as a linear array, for example 5x1x1. Neither do we require the same number of neurons in each layer. The resulting dimensions of the ANN is dictated only by the geometry of the basic construction element.

External Interface

Signals external to the array must be interfaced in such a manner as to permit large amounts of data throughput. The sides of the array and the open connections found on the sides may be so used. Both data input and output may be so facilitated. It is also possible to focus an image of data on one or more sides of the array by incorporating photodetectors and appropriate detection electronics into neurons on each such side. Alternatively, special cells may be affixed to each such side with photoreceptive properties, and little or no neural simulation ability.

Cell Connectivity

How high of a cell connectivity can be achieved? If every other layer in the cubic cellular structure was phased as illustrated in the top of Figure 3, then each cube makes physical contact with 12 adjacent cubes. Sides of 14 adjacent cubes can be made to have physical contact if adjacent rows in a layer are phased as is illustrated at the bottom of Figure 3. If similar phasing is applied to the hexagonal structure in Figure 1, then each unit will also make contact with 14 other units.
Operation

Operation Modes

The VANN will operate in three modes: programming, learning and operational:

(1) The type of ANN algorithm to be used is established in the programming mode.
   The operations here include establishment of the set of neurons to which a given
   neuron is (directly or indirectly) connected and the (sigmoidal) nonlinearity to be
   used by the neuron.

(2) In the learning mode, the interconnect weights among neurons are established
    using training data or, in certain applications such as combinatorial search
    problems [7-8], some training algorithm. When training data are used, some or
    all of the neurons are assigned certain states. The interconnect weights are then
    determined internal to the VANN by algorithms both known and yet to be
    discovered. In certain training algorithms, the initial interconnect weights are
    algorithmically specified by, say, a random number generator.

(3) In the operational mode, the neuron cubes perform three primary functions:
    a) computation of the neuron state which is a function of the neurons to
       which it is connected,
    b) conversion of the neuron’s state into an electrical signal,
    c) retransmission of neuron states from other adjacent neurons to yet other
       neurons in a message passing type of procedure.

Inter-Cell Communication

The interconnects from a neuron to the set of neurons with which it
communicates are stored within the neural cell with the corresponding cell addresses. In
the learning process, these values are established algorithmically (possibly iteratively) as
a function of the states desired in the operational mode. This is done internally to the
VANN, for example, by imposing desired states on a class of neural cells, letting the
ANN compute the states at some other group of cells, and computing the difference of
this value and the states desired. This error is then used to alter the interconnect weights
to reduce or compensate for this error.

A neuron’s state is typically computed as the (interconnect) weighted sum of
connected neural states nonlinearly altered using some memoryless nonlinearity such as a
sign function or a (biologically motivated) sigmoid. The conversion to an electrical
signal of the state possibly involves scaling of the state value and generation of a
destination address (each cell contains within it an address locater number which may be
used to designate its position within the cell array) if required. Retransmission of
adjacent state signals is done using a messenger function. They are employed to
distribute state signals from a first cell which generates the signal to another cell (or a
number of neurons) not adjacent to the first neuron.

The function of retransmission is employed to simulate the action of biological
neurons which have a high degree of connectivity to numerous other neurons, some at a
great distance from the source neuron. In any physical geometry of electronic neurons, this connectivity aspect represents a real problem. Allowing autoconnects, for example, in a 10x10x10 neuron array, it is possible to require up to one million interconnection paths in some algorithms. Wiring such a set of interconnections is clearly extremely difficult physically.

In the structure outlined here, all interconnects among non-adjacent neural cells are performed by having other neurons retransmit the sending state signal until the signal reaches its destination. Additionally, it is possible for a signal to be broadcast to a defined subset of all neurons, or even all neurons, via specially encoded messages. This is taken care of in the address portion of the signal.

Each cell must contain a communications handler whose purpose is to receive, redirect, and generate state signals. Each cell must also contain a computational element for computing state changes, and for applying weights to signals received from other neurons and also perhaps to weight its own outgoing signal. It must contain memory for program storage, which may be in the form of read-write, read-only, or read-mostly memory. It must contain read-write memory for storing parameters associated with changes in state and state weighting functions.

Neuron addresses may be either programmed permanently into each neuron prior to assembly of the array, or, preferably, would be self-programmed on power-up of the array. For example, a neural cell in the top left corner could through internal software ascertain it position simply via the fact that certain of its sides are not connected to other cells. It could then communicate to adjacent cells its position, allowing adjacent cells to determine their locations and hence addresses. The process can propagate automatically through the entire array until completed and all cells have assigned themselves addresses. The addresses would be stored in read-write memory or read-mostly memory in each neuron.

The flow of signals must be organized in such a fashion as to avoid collision of moving packets of information. For ANN algorithms that require each neuron to communicate with every other neuron, this can be achieved by alternating signal flow directions as is illustrated in Figure 4. At one instance, communication can be with neuron elements in a specified direction. In the next communication cycle, this direction would change. The technique can also be modified for the less severe case to algorithms where a neuron is only required to be connected to each neuron in an adjacent layer.

**Downloading and Uploading Features of the VANN**

Since cells imbedded deeply in the array are unreachable by direct electrical contact, the program may be ‘downloaded’ into each neuron via the retransmission process, or into just a subset of the array. A single neuron may be used as an entry node to facilitate the downloading. The programs may be loaded into the array via a conventional computer. Weights and communications paths may also be loaded into the array on a neuron by neuron basis if required by a similar process.

The ability to download neural information may be complemented by an ‘upload’ feature used to extract all neuron state and program information, especially information and programming of a variable nature. This is a critical feature for saving neural state information permanently onto hard media, such as a magnetic or optical disk. On power down of the network, all such information may be otherwise lost. Also, if a neural
network is to be replicated in mass production with specific programming, such uploads are crucial to extracting the information required for duplication. Only then can the extracted information be reprogrammed into one or more other similar neural networks which, for example, may utilize a higher speed operational mode dedicated architecture or be fabricated using analog VLSI. If this process were not performed, it would be necessary to teach each network individually, a process which can be tedious and impractical. The upload/download techniques are a form of cloning akin to software duplication of a conventional computer's programs and information.

Neuron per Cell Ratio

Since each neuron contains a digital computing element, it is possible and indeed, desireable, for each neuron to simulate a number of neurons at once. The 8x5x4 array shown may actually be made to simulate not 160 neurons but 640 neurons if each neuron cube simulates the action of four neurons. Communications among such 'internal' neurons may be facilitated with appropriate software. Communications among neurons would be quite similar except that additional burden would be placed on the inter-cell electrical connections.

Fault Tolerance

Another related issue is fault tolerance. If thousands of neurons are employed in a network, failures of neurons are inevitable. The software in each neuron must be designed to tolerate failures. For example, a communications failure of a single neuron may block transmission of messages among many other neurons. Considerable thought must be given to making communications automatically reroutable if such failures occur. It is possible to design a neuron algorithm such that an adjacent neuron could 'take over' the functioning of a bad cell or neuron.

Performance

The potential performance of a VANN is illustrated by the following analysis. We assume:

- A message handler can decode and route a byte or other parallel word of data and move it from one of the faces or edges of a neural cell to another face or edge to which it has physical contact at a constant rate, $V$ bytes/second. Alternately, at this same rate, the handler can intercept a word and queue it to a neuron inside a neural cell.

- The VANN has linear dimension of $N$ and thus is composed on the order of $N^3$ neural cells.

- A cell has $K$ connection faces to adjacent cells.

- Each data packet travels an average distance of $D$ cells from source to destination corresponding to $D$ intercell transfers.
CONFIDENTIAL

From these assumptions, it follows that:

- At any given moment there can be a maximum of $K N^3$ bytes of information pending within the VANN communication interfaces.
- At an intercell transfer rate of $V$, there exists a $V K N^3$ bytes/second maximum transfer limit, and a limit of

$$T = V K N^3 / (L D)$$

on the number of packets/second transmitted and delivered where $L$ is the communications packet length in bytes.

In order to better appreciate this analysis, let’s assume we require $L = 72$ bits/packet ($= 9$ bytes/packet using 8 bit bytes) parsed as follows:

- 24 bits of destination address or specific destination code.
- 16 bits of data (neural state)
- 24 bits of source address
- 8 bits of special handling code information (multiple destinations, etc.)

Let’s further assume that

- $N = 10$
- $V = 10^7$
- $K = 12$
- $L = 9$
- $D = N / 2$ (average)

Then the effective transfer rate in terms of messages transmitted and received is:

$$T = 2.22 \times 10^9$$

per second (maximum)

If we assume the reasonable inefficiency factor of 2 due to collisions, a realistic transfer rate would be

$$T \approx 10^9$$

messages/second delivered

Assume further that each cell contains 1,000 artificial neurons. Then there would be a total of $10^6$ neural simulations per second. This would only leave time for each neural simulation to be computed and retransmitted in only one microsecond. The neural computer imbedded in each cell would thus need to process $10^6$ neural simulations per second, requiring perhaps an optimized DSP chip for the task or even several DSP chips running in tandem.

The problem then becomes inverted relative to more traditional ANN hardware: the communications, using conventional CPU hardware, becomes faster than the ability to compute.

In reality, data transfers can be made at least twice as fast as our example (50 nsec/byte) using relatively slow low power CMOS logic. With ECL logic, transfers can easily be made in about 10 nsec. As we have indicated, however, the transfer rates seem not to be the relevant issue with VANN’s until processing speed can approach the sustainable transfer rates.
Packaging

Electronic coupling via mechanical joined electrical contacts is highly unreliable and thus not suitable for use in avionics. There are at least three potentially attractive alternatives:

- Highly reliable capacitive coupling can be achieved using an appropriate thin layer of dielectric for the cell walls.
- If the physical dimensions of the array are fixed, interconnects can simply be hard wired.
- Communication among neural cells can be done optically. (Note that, however, unless power can be provided internal to the construction element or through some other externally applied field, alternate interconnects would still be required to provide power.) As is shown if Figure 5, optical sources, such as LED’s, would be aligned to optical detectors at the construction element’s surface through a skin of optically transparent material. Inter-element communication could be established by any one of a number of commonly used modulation techniques.

Power Dissipation

It may be seen that as each neuron cube consumes power, the power is converted to heat which must dissipated in some manner. The geometry of the basic construction element can be modified to commit a large percentage of the volume to coolant flow. An example that can be used in lieu of the cube cell is shown in Figure 6. A single construction element is shown on top. A 2X2 array of these elements is shown on the bottom.

Final Remarks

The volumetric artificial neural network (VANN) is a neural network packaging with potentially high accurate performance capabilities using conventional electronics. We hope to propose a three year program wherein the VANN can be developed as a highly flexible and reliable computational tool for avionic and other applications. The milestones for this project are:

**Year 1**: Detailed performance of the VANN using state-of-the art electronics, including comparison with other more abstract connectionist architectures such as hypercubes and multicubes [9]. Initiate development of VANN software.

**Year 2**: Packaging study including materials, reliability analysis, cell coupling techniques and heat dissipation. Software finalization.

**Year 3**: Prototype the VANN.


FIGURE CAPTIONS:

FIGURE 1: Geometrical shapes such as the hexagonal one shown here can be used as a neural cell.

FIGURE 2: An 8x5x4 array of cubic neural cells. Possible geometries are dictated only by the shape of the neural cell.

FIGURE 3: (top) Phasing the layers of a cubic neural cell allows each cell to interact with the 12 other cells that it touches. (bottom) Additional phasing of adjacent rows allows a cell to directly connect to 14 other cells.

FIGURE 4: Illustration of cyclically changing signal flow directions. The technique is used to reduce collisions of traveling information packets. (All required direction flows for intense interconnection are not shown.) Alternately, the direction of flow in adjacent layers can be different at different points of time.

FIGURE 5: Illustration of the manner that adjacent cells can be optically coupled.

FIGURE 6: (left) An example of a construction element that allows ample coolant flow. (right) A 2x2 array of these elements.
Figure 2
Figure 5
Figure 6
Another mechanical method of interconnecting such arrays is to have each cube snap together with adjacent cubes, thus creating the need for external pressure. Cubes may be stacked together in a rectangular array of the basic connection element shown in Fig. 3. Other stacking configurations may be obtained by simply adding cubes in plan view. The stacking cubes may be stacked in volumetric fashion, e.g. the 8x8x8 cube array shown in Fig. 3.

The resulting connection element shown in Fig. 4 is where the connectivity of the basic connection element can be modulated to control its desired connectivity to the underlying circuitry when desired. As shown, the connectivity could easily be displayed in a number of other manners. The fact that the cubes are connected to form a higher connectivity element is used to overcome these problems.
Physically, some set of interconnections is clearly extremely difficult to reproduce. With such a set of interconnections, it is possible to produce up to one million AON (Addressable Neurons) and, by combining them into a computer-like structure, to solve a variety of problems. In a computer-like environment, some of the neurons are connected to a set of neurons which have high degrees of connectivity to numerous other neurons, some of the neurons are connected to a set of neurons which have low degrees of connectivity to numerous other neurons, and some of the neurons are connected to a set of neurons which have medium degrees of connectivity to numerous other neurons. This combination of high, low, and medium degrees of connectivity is employed to simulate the action of biological neurons.

The function of information transmission is employed to simulate the action of biological neurons.

A neuron in the ANN (Artificial Neural Network) is modeled as a single (interconnected) weighted sum of connected neurons. The input to a neuron is the weighted sum of the outputs of other neurons, with the weights being adjusted to optimize the performance of the ANN. The output of a neuron is the weighted sum of the inputs of other neurons, with the weights being adjusted to optimize the performance of the ANN.

The learning mode of the ANN is used to adjust the weights of the connections between neurons, with the goal of optimizing the performance of the ANN. The learning process involves adjusting the weights of the connections between neurons, with the goal of optimizing the performance of the ANN. The learning process is performed by an optimization algorithm, such as backpropagation, which adjusts the weights of the connections between neurons to minimize the error between the ANN's output and the desired output.

The operational mode of the ANN is used to process input data, with the goal of generating an appropriate output. The operational mode is performed by a set of neurons, with the goal of generating an appropriate output. The operational mode is performed by a set of neurons, with the goal of generating an appropriate output.

The ANNs will operate in these modes: programming, learning, and operational.

Also be simply connected together or addressed via any of a number of continuously available means, through the action of neurons embedded in each one.
Any one of a number of commonly used modulation techniques may be used to encode addresses of communication could be accomplished by frequently transmitted material. Under normal communication could be accomplished by one or more other addressed signals, thus providing for efficient, high-speed instruction. Interconnections would be established through optical means at the communication element's interface through a scheme that is similar to the use of optical fiber. In this manner, any number of other similarly addressed signals could be transmitted simultaneously. Simultaneous transmission through microphone-coupled transistors can be done in the event that the microphone is coupled to two or more other elements. The microphone may be simply mouted to a microphone, pedal, or other element, which may be connected to the microphone, pedal, or other element. The microphone may be connected to any other microphone, pedal, or other element, which may be connected to the microphone, pedal, or other element. Each module may also be used for communication, providing the same result as described in previous sections.
The amount of data throughput the size of the array and the open connections found on

Since each neuron contains a digital computing element, it is possible for each neuron to

The function of a neural network, which is possible to design a network architecture such that an adjacent neuron could, take over.

Another related issue is that the network is constructed using many other neurons, considering the interactions between neurons. The network is made of thousands of neurons and are employed in a

Convolutional computing operations and information.

The ability to downsample neural information may be complemented by an "isolate" process.

One primary characteristic of a neuron is its interpretability, in the sense that the

The flow of signals must be organized in such a fashion as to avoid collection of moving

**CONFIDENTIAL**
Further use of light or other radiative means to couple either into or out of such cells in the array, e.g., through an external computer or controller, either for storage, analysis, or other purposes.

9. An ability for each cell to perform calculations on data it receives from other cells and the communications means.

8. An ability for each cell to perform calculations on data it receives from other cells and the communications means.

7. An ability for cells to self-determine their locations within an array by an algorithm.

6. An ability of each cell to perform computations on data received from other cells and transmissions among cells without requiring the reconfiguration function among cells.

5. A communications interconnection among cells which permits global or large-subset transmission among cells.

4. Several electro-mechanical means for interconnecting cells by stacking, interlocking, one on another, transmitting, receiving, and communicating as a function of such communication.

3. An ability of each cell within the array to electrically or optically communicate one cell of similar type.

2. An ability to construct an arbitrary stacking of such cells into an array essentially and operational modes.

and operational modes.

characteristic of a neuron to varying degrees of modification in programming, training.

1. A design for a neural network comprising a plurality of interdimensional structures limited to the following:

The inventive aspects of the proposed neural network we believe include but are not

INVENTIVE ASPECTS

Confidential
11. An ability of functional cells to ignore malfunctioning cells via communications methods and algorithms governing the communications paths. A further ability of other cells to simulate the functions of malfunctioning cells if required.

12. An ability of a cell to stimulate more than one neuron via computational algorithms, and to communicate information from such simulations to other cells in the array via similar communications means.

Figure 1: A single neuron cube - edges and faces may be used for interconnects. Cooling channels are constructed for modular connection. Sprawling interconnects, shown here, are one of a number of available techniques for mechanical coupling.

Figure 2: Other geometrical shapes such as the hexagonal one shown here can be used as a neural element.

Figure 3: An 8x4 array of neural cubes. Possible geometries are dictated only by the shape of the neural unit.

Figure 4: Left) An example of a construction element that allows ample coolant flow. Right) A 2x2 array of these elements.

Figure 5: (top) Phasing the layers of a cubic neuron unit allows each neural unit to interact with the other units that it touches. (bottom) Additional phasing of adjacent rows allows a cube to directly connect to 14 other cubes.

Figure 6: Illustration of the manner that adjacent construction elements can be optically coupled.

Figure 7: Illustration of cyclically changing signal-flow directions. The technique is used to reduce collisions of traveling information packets. (All required direction flows for intense interconnection are not shown.) Alternately, the direction of flow in adjacent layers can be different at different points of time.
Figure 2

neuron unit
Figure 3
A maximum of 8 key words that describe the Project:

Artificial Neural Networks, Artificial Intelligence, VLSI, Concurrency Machinits

Far proposed and may eventually evolve into a staple for ANN implementation. This proposal, is in parallel with all of the ANN algorithms thus far proposed and may eventually evolve into a staple for ANN implementation. The proposed architecture is potentially compatible with all of the ANN algorithms. This proposal is the result of a highly flexible, architecturally fault tolerant electronic neural network. The result is a highly flexible, architecturally fault tolerant electronic neural network. We propose a massively parallel architecture for implementation of artificial neural network architectures. (Limit your abstract to 200 words, no classified or proprietary Informational)

Technical Abstract

Volume/Architectures

Proposal Title

Dr. Robert J. Marks II, Research Associate

Name and Title of Principal Investigator

Bellevue, WA 98003
1329 Northup Way, Suite #203
Multi-dimensional System Associates

Name and Address of Proposing Small Business Firm

Reference Department: AFOSR

Project Summary

Phase I–FY 1999
Small Business Innovation Research (SBIR) Program
U.S. Department of Defense

Appendix B

DOD No. 99-1
circuitry resulting in faster yet less accurate and less flexible processing ability.

The architecture we propose for the VANN can also clearly be used with and

* Additional fault tolerance may be inherent in the ANN algorithm.

No inherent third in operation characteristics
Non Volatility
Algorithmic Programmability

Detailed implementations and include

One possible attributes of the VANN are those normally associated with

- in connectivity
- in adaptability
- in choice of activation
- Hebbian
- Context Reuse
- Ambiguity
- Algorithmic Fault Tolerance
- Algorithmic Flexibility
- Regular Structure
- Required equations are currently available.

The following characteristics:

To overcome these limitations, we propose investigation and initial development

of an electronic architecture for volumetric artificial neural networks (VANN,s) with

obviously not available in two.

The high connectivity available in three dimensions is

Further more, the paper approach to both serial digital and analog ANN VLSI

have used high speed planar analog electronics.

Highly parallel structure or

connected on using conventional high speed serial computers designed on a

Introduction

C. IDENTIFICATION AND SIGNIFICANCE OF THE PROBLEM OR

Corporate Confidential Information

Use or disclosure of the proposed data on these specifically identified by an asterisk (*) are subject to the

Confidentiality Proprietary Information

}
developed to a more in-depth discussion of these proposed areas of research.

Following a more specific description of the VANNN, the remainder of this section is

- In connection with other current implantation technologies.
- Using currently available electronics.
- Overall performance metrics evaluation.
- Extensive interface design.
- Cell shapes and size effects on power dissipation.
- Performance advantages over present electronics.
- Comparison between alternative packaging schemes and conventional VANNN
- Performance impact of architectural packaging options such as
- Projectable and conformable capability.
- Inter-cell communication abilities and limitations.
- Projectable, learning and recall modularization.
- Optional (software capability) of the VANNN including

VANNN including the following:

The technical objective of Phase I will be to evaluate feasibility aspects of the

**D. PHASE I TECHNICAL OBJECTIVES**

VANN ARCHITECHTURE

VANN ARCHITECHTURE

- Proposal addresses more in detail how a VANNN meets these objectives.
- The remainder of this

The observation to the portion strongly suggested a different three dimensional ANN

Use of disclosure of the proposed data on these specific technical by an artisan(s) are subject to the

Confidential proprietary information
THE SHAPE OF THE NEURAL CELL

Figure 2: Ax 8x3x4 array of cubic neural cells. Possible connections are dictated only by cell-neural cell.

* Figure 1: Geometric shapes such as the hexagonal one shown here can be used as a

Confidential proprietary information

Restoration of the cover page of this proposal
Use or disclosure of the proposed data on these specifically identified by an asterisk (*) are subject to the

Translation of the cover page of this proposal
Use or disclosure of the proposed data on these specifically identified by an asterisk (*) are subject to the
The function of neurotransmission is employed to simulate the action of biological communication. For a neuron's state to be significantly changed, it is necessary to modify its weights (interconnected weights). When the input from another neuron exceeds a threshold, the neuron's state will change, and if the change is significant, it can influence the state of other neurons. This process is known as the ANNs (Artificial Neural Networks) as they attempt to achieve the same level of processing as biological neural networks. The ANNs are used to model and simulate cognitive processes such as learning and memory.

In the learning process, these values are established through a Hebbian-type learning rule, where the strength of the connections between neurons is modified based on the input and output patterns. The learning rate is adjusted to optimize the performance of the network.

The ANNs are used in various applications, including pattern recognition, classification, and prediction. They are particularly useful in problems where the input data is complex and non-linear, such as image and speech recognition.

The operation of the ANNs is determined by the learning and recall processes, where the weights of the connections between neurons are adjusted to minimize the error between the predicted and actual outputs. The learning process is an iterative process where the weights are updated based on the error and the input data.

The ANNs are trained using algorithms such as backpropagation, where the error is propagated backward through the network to adjust the weights. The recall process is used to reproduce the output for a given input, and it is essential for the performance of the ANNs.

The ANNs are versatile and can be used in various applications, including medical diagnosis, financial forecasting, and natural language processing. They are an essential tool in the field of artificial intelligence and have revolutionized the way we process and understand complex data.
The ability to download neural network may be completed by an "\( \text{VANN} \)".

Download and Uploading Features of the \( \text{VANN} \)

adjacent layer.

- Algorithms where a neuron is only required to be connected to each neuron in an
  - dirección would change. The \( \text{VANN} \) can also be modified for the less secure case to
  - direction chosen in a specific direction. In the next communication cycle, this
  - shows how directions as illustrated in Figure 4. All one-dimensional cells can be activated by the\n  - collection of non-adjacent neurons with every other neuron this can be achieved by choosing
  - the flow of signals must be organized in such a fashion as to avoid or minimize

- Read-modify-memorize in each neuron.
  - assigned memory addresses. The addresses would be stored in read-write memory or
  - can propagate emergently through the entire array and all cells have
  - addressing and excitation cells to determine their location and hence address. The process
  - connected to other cells. If a cell then becomes an "\( \text{VANN} \)" cell in the top or bottom instead of the
  - stored as a "\( \text{VANN} \)". Each cell must have some non-appearance of each neuron prior

- Neuron addresses may be either programmed permanently into each neuron prior

- Associated with changes in state and state weighting functions.

- This is taken care of in the address portion of the signal.

- In contrast to the more complex sensory processing

- Each cell must contain a communication header whose purpose is to receive.
Neuron Per Cell Ratio

Since each neuron contains a digital computer element, it is possible and indeed desirable, for each neuron to simulate a number of neurons at once. The 8x5x4 array shown may actually be made up of neurons but 640 neurons if each "neuron," neurons may be facilitated with appropriate software. Communications among neurons would be similar except that additional burden would be placed on the inter-cell electrical connections.

Packaging Impact on Performance

- There are at least four potentially attractive techniques to couple neural cells:
  - Direct electrical contact, although unreliable, is an obvious interconnect option.
  - Highly reliable capacitive coupling can be achieved using an appropriate thin layer of dielectric for the cell walls.
  - If the physical dimensions of the array are fixed, interconnects can simply be hard wired.
- Communication among neural cells can be done optically. (Note that, however, unless power can be provided externally to the construction element or through some other externally applied field, alternate interconnects would still be required to provide power.) As shown in Figure 5, optical sources, such as LED's, would be aligned to optical detectors at the construction element's surface through a skin of optically transparent material. Inter-element communication could be established by any one of a number of commonly used modulation techniques.

Power Dissipation

It may be seen that as each neuron cube consumes power, the power is converted to heat which must be dissipated in some manner. The geometry of the basic construction element can be modified to commensurate amount of this heat, in this manner. An example that can be used in lieu of the cube cell is shown in Figure 6. A single construction element is shown on top, A 2x2 array of these elements is shown on the bottom.
From these assumptions, it follows that:

- Destination corresponding to internal transfers.
- Each data packet travels an average distance of \( D \) cells from source to
  \( A \) cell has \( N \) connection faces to adjacent cells.
- The VANN has lower dimension of \( N \) and thus is composed on the order of \( N^2 \)
- A neural cell inside a neural cell.
- Alternately, at this same rate, the handler can inspect a word and glue it to a
  \( A \) edge to which it has physical connection at another face of a neural cell in another face of a
  \( A \) message handler can decode and route a byte or other packet of data.

We assume:

- Section a graph calculation showing the potential performance of the VANN is given.
- According to the overall potential performance of the VANN can be calculated. In this
- Once reliable operation is assumed and the architectural limitations of the VANN

**Overall Performance**

When each unit will also make contact with \( 4 \) other units.

Portions of Figure 3. If similar behavior is applied to the external structure in Figure 1
- When physical contact is effective in layers a layer of parallel links can be made in
  \( 4 \) adjacent planes. Sides of \( 4 \) adjacent planes can be made to
- Cellular structure was based on the hypothesis that every other layer in the cubic

**Cell Connectivity**

- Protocols for propelling, and little or no neural simulation ability.
- Such sites, alternatively, special cells may be added to each such site with
  incorporation of protocols and appropriate detection electronics into neurons on each
- It is also possible to focus an image of data on one or more sides of the array by
  located on the side of the array and the open connections
- Signals external to the array must be interfaced in such a manner as to permit

**External Interface**

- Take over the function of a bad cell or neuron.
- Occurs if possible to design a neuron algorithm such that an adjacent neuron could
  must be given to making communication functionally recognizable by such features
  2 makes transmission of messages among many other neurons. Considerable thought
  no longer to invoke relays. For example, a communication failure of a single neuron
- A network's failure of neurons are inevitable. The solution in each neuron must be
  Another related issue is fault tolerance. If thousands of neurons are employed in

Use or disclosure of the proposed data on this page specifically identified by an asterisk (*) are subject to the

Confidential Proprietary Information
Cell 1 directly connects to 14 other cells.

The 12 other cells that touch the bottom additional phasing of adjacent rows allows a

Figure 3: (Top) Phasing the layers of a cubic neural cell allows each cell to interact with

* redaction message: "Restriction on the cover page of this proposal. Use or disclosure of the proposed data on these specifically identified by an asterisk (*) are subject to the confidential proprietary information"
Figure 6: (Left) An example of a construction element that allows ample coolant flow.

(source) A 2x2 array of these elements.

Figure 5: Illustration of the manner that adjacent cells can be optically coupled.
Phase I: a detailed performance analysis of the VANN using static or-the-art electronics.

Through simulation, analysis, and first-order prototype, we hope to establish in

sustainable transistor limits.

seen not to be the relevant issue with VANN's until processing speed can approach the

can easily be made in about 100 nsec. As we have indicated, however, the transistor

use of MCs logic, and with ECL logic, transistors

In reality, data transfer can be made at least twice as fast as our example (50

In conclusion.

The communication's use of conditional CPs hardware, becomes faster than the

several DSP chips running in unison.

simultaneous per second, requiring perhaps an optimized DSP chip for the task or even

near-computer embedded in each cell would need to process 106 neural

near-computer to be connected and running, in only one microsecond. The

near-computer to be connected and running, in only one microsecond. This would only leave time for each

Assume further that each cell contains 1,000 artificial neurons. Then there would

Thus would be

If we assume the reasonable efficiency factor of 2 due to collisions, a realistic transfer

Then the effective transfer rate is in terms of messages transmitted and received is:

where: L = 72 x 10^9 per second (maximum)

Then let's further assume that

8 bits of special handling code information (multiple destinations, etc.)

24 bits of source address

16 bits of data (normal size)

24 bits of destination address or special destination code.

Thus a packet (of 9 bytes/packet using 8 bits bytes) packet as follows:

in order to better appreciate this analysis, let's assume we require L = 72

on the number of packets/second transmitted and delivered where L is the

\[(D/L)/N = K \Lambda = L\]

transfer limit, and a limit on

If an internal transfer rate of \(L\) bytes/second, maximum

any given moment that can be a maximum of \(N\) packets of information

restriction on the server side of the protocol

Use of disclosure of the proposal data is subject to the

Confidential Proprietary Information
References


C. Relationship With Future Research or R&D

The total cumulative funding of the above projects is well in excess of one half million.


Power Systems Stability and Security Assessments, Using Artificial Neural

Networks, Computer Science and Artificial Intelligence Laboratory, California Institute

of Technology, 1987-89.

Neural network computer architectures, The Washington Technology Center.

Optical Systems Lab at Texas Tech University (1987-88).

Increasing the accuracy of optical processors, SDI/IST through ONR and the

Air Force.

And with the other PIs listed, is currently involved in the following projects:

Industrial Ethernet

Analysis and application of neural nets: Proposing High Technology Center (1996-88),

with L.E. Adams.

The principle investigator has participated extensively in University level funded

research in artificial neural networks and related topics. He was one of two PIs on the

project, Technology, Inc.

As is evident from the biographical information in sections 1 and 2, the key

architect, L. E. Adams, and in parallel with this proposal, are pursuing a patent filing for the YANN

a fundamental description of the YANN Disclosure Document (1997 Dated from 8/12/88) (8)

The principle investigators have recently disclosed to the United States Patent Office

R. RELATED WORK

Use of disclosure of the proposed data on these selected topics is allowed, and no

Confidential Proprietary Information
H. POTENTIAL POST APPLICATIONS

I. Initiation of industrial drive in application of Phase III.

II. Supercapacitors.

III. Investigation of potential use of the VANN architecture in other aspects of

IV. Development of a more sophisticated VANN prototype.

V. Study of these packaging characteristics.

VI. Design with stress product development group of sealable ceramics.

VII. Conduct and test dispersion. The principle materials have recently initiated cell performance study including materials, reliability and stress cell.

VIII. Development of VANN software for the more commonly used ANN algorithms.

* Actual material

will freshly propel the VANN into the stage of a simple neural network computer.

* At the heart of computers, success in the current mass production research into ANNs's (artificial neural networks) has been a significant development.

* Spatial dimension, the application of VANN's and their effective implementation as pattern recognition engines.

* In areas such as voice, speech, signal processing, pattern recognition topics, there have been significant demonstrations of ANN applications in recent times.

* However, ANN’s are not new in operation at least as well as their biological counterparts. Although ANN’s are not new in operation at least as well as their biological counterparts, their obvious potential to

A fundamental reason for the enthusiasm for ANN's is their obvious potential to

[*] In these dimension, VANN's have been initiated in the last three years. VANN's have been initiated in the last three years. They are effective, comprehensive, established in many countries, including Germany and England, and others. VANN is in the United States, Federal, Government, through DOD, NSF, and in the last few years. The United States, Federal, Government, through DOD, NSF, and in the last few years...

* Especially in the field of electronic computer technology, has continued an

...
ARCHIVAL PUBLICATIONS

Robert J. Marks II
Principle Investigator

1. KEY PERSONAL

Disclosure on the cover page of this proposal. Use of disclosure of the proposed data on these pages is subject to the
confidentiality and proprietary information
Avoid disclosure of the proposed data on privacy specifically identified by any means. Applicable to the confidential proprietary information.

HARALD PHILLIPP, Research Associate

Learning in a Digital Computer, (pending)

Peter J. Van Heerden, Robert I. Marks II, and Seho Oh, „A Computer Chip Realizing
Washington Technology Center (pending)

R. I. Marks II, L.E. Alvis, and S. Oh, „An optical neural network„ assigned to the

PATENTS

(R) 1980; D. and H. I. Methods of implementing electrical neural networks
(Values of 1980; D. and H. I. Methods of implementing electrical neural networks

Chair of the session on artificial neural networks at the International Symposium on
Publicity committees, Washington D.C.

1989 International Joint Conference on Neural Networks Conference Planning and

Artificial Neural Networks: Fundamentals and Applications, organizer and chair, Northco,

Chair of Working Group on Perception at the Workshop on Optical Artificial

Artificial Neural Systems and Applications, Session Organizer and co-chair, 1987

SPECIAL SESSIONS AND WORKSHOPS

sup I, p. 4 (1988)

model classification problems, applications, and optical, Neural Networks, vol. 1

L.E. Alvis, R I. Marks II and L.W. Taylor, „Network Learning algorithms for multi-


A Potential benchmark for analyzing the performance of artificial neural networks.
Recent Publications of H. Phillip in the Areas of Electronics and Photonics:

- Volume Computing: Software and Hardware, with R.J. Marxs (in progress)
- Waveform Synthesis and Control
- High-Speed Digital Sampling, Timebase Circuitry
- Logarithmic Successive Approximation Conversion
- Compound Successive Approximation Conversion
- Monolithic Signal Acquisition Conversion
- Frequency Multiplication, Signal Conversion
- Optical Motion Sensor
- Optoelectronic Circuits
- Switching Amplifier Circuit
- Waveform Acquisition System
- Instrumentation Engineering
- Systems, Electro-Optics, and Switching Amplifier Technology

H. Phillip has 5 patents issued in the fields of image acquisition, switching, and numerous others.

Contribution: Photon Kinetics, Soliton, Beam AB (Sweden) and numerous others.

H. Phillip is currently Chairman of the Board of the Society of Photo-Optical Instrumentation Engineers.
2 SUN 386 Workstations
AT&T DSP21r Real-time DSP development system
Texas Instruments TMS320C31 Real-time DSP development system
Multi-Chip A/D and D/A converters
SN74100 Workstation
SUN 3/0 Workstation
Symboles 3660 Lisp Machine
DEC Micro-VAX I

programs included in the lab is the following computational equipment:

Additional simulation and co-simulation using simulation software available including ISDL Simulink and Signal Processing Blockset. The SUN workstations have been programmed to develop new neural network models. The SUN workstations, a SUN 4/30 workstation, and a SUN 4/110 system, were used for signal simulation and other applications.

Simulation studies of the VANN will be performed on a supercomputer. A supercomputer.

CA 93123,
The first two items are available from Xilinx Corporation, 2069 Hamilton Ave., San Jose.

Extended memory ($3300)
- XC-7288 Configuration PROM Programmer ($450)
- XC-DSII Programmable Gated Development System ($1950)

The system requires the purchase of the following software and hardware:

- Various CAD, compiler and assembler software packages are also available.
- Various electronic instrumentation (scopes, meters, etc.)
- CPE 7228 EPROM/Intel processor programmer
- Xilinx Simulation & hardware development systems
- Xilinx Floating Point/Intel processor development system

Corporation has the following equipment:

Multidimensional Systems Associates in affiliation with Phillip Technologies

1. FACILITIES/EQUIPMENT

Submission will be so notified immediately.

There is no prior current or pending support for a proposal similar to this one by

1. PRIOR, CURRENT OR PENDING SUPPORT

Confidential proprietary information

Use or disclosure of the proposed data on these specifically identified on an asterisk (*) are subject to the

Confidential proprietary information

22
neural networks to large artificial intelligence problems, neural architectures and training
strategies.

In 1985, Dr. Kaplan founded Intelligent Computer, Inc. in Seattle, Washington, and started working on neural network research. In 1986, he returned to the University of Washington to work on artificial intelligence and neural networks. His dissertation concerned application of neural networks to

Dmitry Kaplan received his PhD in Electrical Engineering in 1988 from the

University of Washington. His dissertation concerned application of neural networks in

the field of control.

---

I.E. E. A. A.上升的, 瑞典人, 今朝的, 今朝的.


---

K. CONSULTANTS

---

2 Next Workshops

Restriction on the cover page of this proposal do not disclose of the proposed data on this specifically identified by an asterisk (*). are subject to the

Confidential and proprietary information
<table>
<thead>
<tr>
<th>Description</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Direct &amp; Indirect Costs</td>
<td>$4,9937</td>
</tr>
<tr>
<td>1% of direct costs</td>
<td>$549</td>
</tr>
<tr>
<td>Washington State Business and Occupation Tax</td>
<td></td>
</tr>
<tr>
<td>35% of direct costs</td>
<td></td>
</tr>
<tr>
<td>Overhead</td>
<td></td>
</tr>
<tr>
<td>Indirect Costs</td>
<td>$3,6984</td>
</tr>
<tr>
<td>Total Direct Costs</td>
<td>$3,6984</td>
</tr>
<tr>
<td>The ASID @ The University of Washington</td>
<td></td>
</tr>
<tr>
<td>Subcontractors</td>
<td></td>
</tr>
<tr>
<td>Dr. David Kaplan</td>
<td></td>
</tr>
<tr>
<td>Dr. Leslie Alles</td>
<td></td>
</tr>
<tr>
<td>Consultants:</td>
<td></td>
</tr>
<tr>
<td>Other Direct Costs</td>
<td>$835</td>
</tr>
<tr>
<td>Equipment</td>
<td></td>
</tr>
<tr>
<td>Total Salaries</td>
<td>$1,068</td>
</tr>
<tr>
<td>1/4 person months</td>
<td>$387</td>
</tr>
<tr>
<td>1/4 person months</td>
<td>$333</td>
</tr>
<tr>
<td>1 person month</td>
<td>$609.4</td>
</tr>
<tr>
<td>1/2 person months</td>
<td>$304.7</td>
</tr>
<tr>
<td>1 person month</td>
<td>$609.4</td>
</tr>
</tbody>
</table>

**M. Cost Proposal**

Restriction on the cover page of this proposal:
Use or disclosure of the proposed data on items specifically identified by an asterisk (*) are subject to the
Confidential Property Information
Enclosures

cc: H. Phillip

Multidimensional Systems Associates

Sincerely,

Robert J. Marks II, President

Multidimensional Networks, Inc., New Architectures, and Models of Computation. Enclosed are five copies of a proposal volumetric architectures for artificial neural network

Address: Cameron Hernandez
Washington, D.C. 20332-5000

BOEING AFB
Bldg. 410, Room A 113
SBIR Program Manager
AP0SR/XOT

(303) 746-0566
Telephone: (206) 746-1652
Believe, Washington 98003
13229 Nordling Way Suite #203

Multidimensional Systems Associates
CONFIDENTIALITY AGREEMENT

[Redacted]

No reproduction allowed.

[Redacted]
CONFIDENTIALITY AGREEMENT

This Confidentiality Agreement (the "Agreement") is entered into as of [date], by and between [COMPANY NAME], a [STATE] corporation (the "Company"), and [RECIPIENT NAME], an individual (the "Recipient").

1. Definitions.
   - "Confidential Information" means any information disclosed by the Company to the Recipient, including but not limited to trade secrets, proprietary information, and technical data.
   - "Recipient" means the individual or entity receiving the Confidential Information.
   - "Confidentiality Period" means the period during which the Recipient is bound by the terms of this Agreement.

2. Confidentiality Obligations.
   - The Recipient shall maintain the Confidential Information in confidence and shall not disclose the Confidential Information to any third party without the prior written consent of the Company.
   - The Recipient shall use the Confidential Information solely for the purpose of fulfilling the obligations under this Agreement.
   - The Recipient shall return or destroy any copies or records of the Confidential Information upon request of the Company.
   - The Recipient shall not use the Confidential Information for any purpose other than the intended use.

   - The Recipient shall not disclose the Confidential Information to any third party without the prior written consent of the Company.
   - The Recipient shall not use the Confidential Information for any purpose other than the intended use.
   - The Recipient shall return or destroy any copies or records of the Confidential Information upon request of the Company.

4. Use of Confidential Information.
   - The Recipient shall not use the Confidential Information for any purpose other than the intended use.
   - The Recipient shall not disclose the Confidential Information to any third party without the prior written consent of the Company.
   - The Recipient shall return or destroy any copies or records of the Confidential Information upon request of the Company.

5. Return of Documents.
   - Prior to the expiration of the Confidentiality Period, the Recipient shall return all documents containing the Confidential Information to the Company.
   - The Recipient shall not retain any copies of the Confidential Information after the expiration of the Confidentiality Period.

   - The Company shall have the right to seek injunctive relief to enforce the terms of this Agreement.
   - The Company shall have the right to recover actual damages incurred as a result of a breach of this Agreement.

7. Termination.
   - This Agreement shall terminate upon the expiration of the Confidentiality Period.
   - This Agreement shall also terminate upon the occurrence of a material breach of the terms of this Agreement by the Recipient.

   - This Agreement shall be governed by the laws of the state of [STATE], without giving effect to principles of conflicts of laws.

9. Entire Agreement.
   - This Agreement constitutes the entire agreement between the parties and supersedes all prior negotiations, understandings, and agreements.

10. Counterparts.
    - This Agreement may be executed in multiple counterparts, each of which shall be deemed an original.

IN WITNESS WHEREOF, the Company and the Recipient have executed this Agreement as of the date first above written.

[COMPANY NAME]

By: ____________________________
    [Signature]

[RECIPIENT NAME]

By: ____________________________
    [Signature]
CONFIDENTIALITY AGREEMENT

PROGRAM/PROJECT: ______________________________________

Whereas, The Washington Technology Center (WTC) is the owner of certain proprietary, confidential information relating to the above technology (TECHNOLOGY); and

Whereas, __________________________________________ (COMPANY) wishes to receive the proprietary, confidential information to facilitate analysis and evaluation of the technology for commercial exploitation; and

Therefore, to assure WTC that all such proprietary information will be maintained by COMPANY under circumstances of strict confidentiality, COMPANY acknowledges and agrees as follows:

1. Proprietary information means any information relating directly or indirectly to the TECHNOLOGY not generally known to the public provided to COMPANY by WTC or its assignors/inventors. Proprietary information may be conveyed in written, graphic, aural or physical form and may include scientific knowledge, know-how, processes, inventions, techniques, formulae, products, business operations, customer requirements, data, plans or other records and information.

2. Proprietary information does not include information which COMPANY can demonstrate:
   (a) was in its knowledge or possession prior to disclosure by WTC or its assignors/inventors;
   (b) was public knowledge or has become public knowledge through no fault of COMPANY; or
   (c) was properly provided to COMPANY by an independent third party who has no obligation of secrecy to WTC or its assignors.

3. COMPANY agrees to maintain the disclosed proprietary information as confidential and agrees not to use this information for its own benefit or for the benefit of any other person or entity.

4. COMPANY may use the disclosed proprietary information only for the purposes of analyzing and evaluating the potential commercial uses of this information. The following restrictions apply:
   (a) COMPANY may duplicate or reproduce the disclosed proprietary information; if duplicated or reproduced in whole or in part, the disclosed information must carry a proprietary notice similar to that with which it was submitted to COMPANY.
   (b) COMPANY may not use, duplicate or disclose proprietary information for purposes of manufacture or procurement of the invention contained within the disclosed proprietary information.
   (c) COMPANY shall not use the disclosed proprietary information for research purposes nor to develop products or technologies for commercialization.

5. COMPANY agrees to protect WTC's proprietary information from further disclosure by taking equivalent precautions used to protect confidential information of COMPANY. In the event of unauthorized disclosure, COMPANY shall indemnify WTC for damages incurred as a result of the disclosure.

6. Upon completion of COMPANY's evaluation of proprietary information or at WTC's request, COMPANY will discontinue the use of and promptly return all proprietary information without retaining copies of that information and will promptly return samples or specimens embodying that information.

7. COMPANY agrees that violation of the Agreement will cause irreparable harm to WTC and that money damages will be inadequate to compensate WTC for its losses or damage. Therefore, COMPANY will stipulate to a motion for injunctive relief prohibiting violation or further violations of this Agreement should WTC desire such relief.

8. Any action arising out of this Agreement shall be decided in King County, Washington. This Agreement shall be construed under the laws of the State of Washington.

If COMPANY agrees to the foregoing, please indicate acceptance thereof by executing this Confidentiality Agreement.

Agreed to and Accepted this:

______________ day of ______________ _____________ , 19

Signature: ____________________________________________

Title: _____________________________________________

Company: __________________________________________
the sides may be so used. Both data input and output may be so facilitated. It is also possible to focus an image of data on one or more sides of the array by incorporating photodetectors and appropriate detection electronics into neurons on each such side. Alternatively, special cubes may be affixed to each such side with photoreceptive properties, and little or no neural simulation ability. Energy fields other than light may also be used, such as microwave, sound, radiation, etc.

INVENTIVE ASPECTS

The inventive aspects of the proposed neural network we believe include but are not limited to the following:

1. A design for a neural network comprising a plurality of three dimensional structures or cells, each such cell having an ability to electrically or optically interconnect on a plurality of sides or edges of each such cell and each having an ability to simulate the characteristics of a neuron to varying degrees of modification in programming, learning and operational modes.

2. An ability to construct an arbitrary stacking of such cells into an array essentially without restriction or limit except for a requirement of physical contact with adjacent cells of similar type.

3. An ability of each cell within the array to electrically or optically communicate one or more of programs, data, or commands, the cells in general having an ability to originate, retransmit, receive and reconfigure as a function of such communications.

4. Several electro-mechanical means for interconnecting cells by stacking, involving one or more of: compression mated contacts, plug-together mechanisms, adhesive mating methods, or magnetic attraction.

5. A communications interconnection among cells which permits global or large-subset transmissions among cells, without requiring the retransmission function among cells.

6. An ability of each cell to perform computations on data received from other cells within the array or external to the array. A further ability of each cell to originate communications to one or more other similar cells, the communicated data or programming being dependent on an algorithm and on the nature of communications from other cells prior to the communication.

7. An ability for cells to self-determine their locations within an array by an algorithm and the communications means.

8. An ability for such an array and its component cells to propagate programs and data from an external source, either to all cells in an array or to a subset thereof.

9. An ability for such an array and its component cells to have programs and data extracted from it via an external computer or controller, either for storage, analysis, or duplication purposes.

10. The use of specially designed or programmed interface cells on one or more faces of the array, engineered to permit communications to and/or from external sources. The further use of light or other radiative means to couple either into or out of such cells in
order to simplify the task of connection, and the use of radiatively active transducers such as phototransistors and light emitting diodes to facilitate such external interface coupling.

11. An ability of functional cells to ignore malfunctioning cells via communications methods and algorithms governing the communications paths. A further ability of other cells to simulate the functions of malfunctioning cells if required.

12. An ability of a cell to simulate more than one neuron via computational algorithms, and to communicate information from such simulations to other cells in the array via similar communications means.

**Figure 1:** A single neuron cube - edges and faces may be used for interconnects. Cooling channels are constructed for modular connection. Springy interconnects, shown here, are one of a number of available techniques for mechanical coupling.

**Figure 2:** Other geometrical shapes such as the hexagonal one shown here can be used as a neural element.

**Figure 3:** An 8x5x4 array of neural cubes. Possible geometries are dictated only by the shape of the neural unit.

**Figure 4:** (left) An example of a construction element that allows ample coolant flow. (right) A 2x2 array of these elements.

**Figure 5:** (top) Phasing the layers of a cubic neuron unit allows each neural unit to interact with the 12 other neural cubes that it touches. (bottom) Additional phasing of adjacent rows allows a cube to directly connect to 14 other cubes.

**Figure 6:** Illustration of the manner that adjacent construction elements can be optically coupled.

**Figure 7:** Illustration of cyclically changing signal flow directions. The technique is used to reduce collisions of traveling information packets. (All required direction flows for intense interconnection are not shown.) Alternately, the direction of flow in adjacent layers can be different at different points of time.
cooling channels

contains:
- microprocessor
- communications handler

spring contacts (all 12 edges)

Figure 1
neuron unit

figure 2
figure 3
figure 5
Figure 6

faces to be joined

source

detector
figure 7
September 28, 1988

Ron Melton
Batelle Northwest
FAX: 509-376-3876
VERI: 509-375-2580

Dear Ron:

Here is a draft of the nondisclosure agreement we have been using. See you Monday!

Best regards,

Bob Marks
Enclosure
cc: H. Philipp

RMCC

Robert J. Marks II

Best personal regards,

Robert J. Marks II

Dear Gene:

I hope some exciting things result from my notes, a contribution you made beyond the patent description was use of as a result, I'll keep you up to date on any developments.

Please let me know if I've missed anything. Thanks for spending the time with me. I hope some exciting things result from my notes, a contribution you made beyond the patent description was use of as a result, I'll keep you up to date on any developments.

Here's a copy of the nondisclosure agreement. I hope some good things happen.

Eugene V. Ochs

Monroe, WA 98272
7720 Woods Creek Road
Electronic Systems Inc.

Phone: (206) 542-0828
Seattle, Washington 98133
16515 Ashworth Ave. N.

September 29, 1988

Neural Processing, Optical Computers & Signal Analysis

ROBERT J. MARKS II
6. This agreement shall be governed by and construed in accordance with the laws of the State of Washington.

5. Promply after a receipt of a written request from PTG, and in the absence of such a request or implied, PTG, in its discretion, may disclose to any person, corporation, partnership, firm, or other legal entity, such portions of the Technology Disclosed heretofore or to be disclosed hereunder, as it may determine from time to time, in its discretion.

4. Confidential Information Regarding the Technology Disclosed heretofore or to be disclosed hereunder shall remain the property of PTG. No license under any patent, copyright, trademark, or trade secret is granted.

3. It is understood by the parties hereto that this obligation of confidentiality shall not apply to information which is publicly available or otherwise becomes generally available to the public.

2. Oceans Electronics and its representatives shall maintain the identity of confidential information which contains trade secrets.

1. All disclosures of confidential information will be in writing and marked "Confidential" at the time such writings are first furnished to the other party.

II. All disclosures of confidential information will be as follows:

1. Oceans Electronics and its representatives will be disclosed to Oceans Electronics, the recipient of such disclosures.

2. Oceans Electronics, the recipient of such disclosures, may disclose such disclosures only to its personnel who are designated as "Confidential".

3. Such personnel will be informed that such disclosures are "Confidential".

4. Confidentiality shall be maintained by all such personnel.

5. Any controversy or claim arising out of or relating to this agreement shall be submitted to the Commercial Arbitration Association, and judgment or decree upon any award of decision shall be final.

Commercial Arbitration Association, and judgment or decree upon any award of decision shall be final.

This agreement shall be governed by and construed in accordance with the laws of the State of Washington.
to allow for better pressurized mechanical coupling. The geometry can be used to fill out the geometry to a rectangular box. For example, Figure 1, dummy constructing electronic contacts can be made from SLR material at the expense of pressure plates from all sides. This can be accomplished with external pressurization plates through-out.

A stack of neurons with integrated interconnections must be somewhat made couple when the units are connected.Esch cascade, as shown, these cascades would be designed to automatically stack the layers. As shown, the number of neurons in each layer, the resulting dimensions of the ANN is dictated only by the number of neurons. Each layer must be modified to perform the required function. For example, as shown in Figure 2, where the output of neurons can be read out as a pattern. Other architectures may be obtained arrays as shown in Pattern 2. Other architectures may be obtained.

These cubes may be stacked in a volumetric fashion, e.g., the 3D cubic function.

overcoming these problems.

By a large number of neurons, this disclosure describes a method for complex electronic ANN's. It is the degree of interconnection required, that is the primary obstacle to realizing such electronic systems. As a result, a primary obstacle to realizing such electronic systems is the high speed required. In many photocolored neural networks that permit modular fabrication, multiple conservation of the ANN's effort to the construction of a building block approach. The basic construction element is based on a building block approach. The basic construction element is based on a building block approach.

In this disclosure, a three-dimensional ANN architecture is described. Dr. Robert J. Marks II

THEREE-DIMENSIONAL ARTIFICIAL NEURAL NETWORK ARRAY

PATENT DISCLOSURE
Another mechanical method of interconnecting such arrays is to have each cube snap together with adjacent cubes, obviating the need for external pressure plates. Cubes may also be simply cemented together or adhered via any of a number of commercially available means, or through the attraction of magnets imbedded in each cube.

The ANN will operate in three modes: programming, learning and operational:
(1) The type of ANN architecture to be used is established in the programming mode. The operations here include establishment of the set of neurons to which a given neuron is (directly or indirectly) connected and the (sigmoidal) nonlinearity to be used by the neuron.
(2) In the learning mode, the interconnect weights among neurons are established using training data or, in certain applications such as combinatorial search problems, some training algorithm. When training data are used, some or all of the neurons are assigned certain states. The interconnect weights are then determined internal to the ANN by algorithms both known and yet to be discovered. In certain training algorithms, the initial interconnect weights are algorithmically specified by, say, a random number generator.
(3) In the operational mode, the neuron cubes perform three primary functions: a) computation of the neuron state which is a function of the neurons to which it is connected, b) conversion of the neuron’s state into an electrical signal, c) retransmission of neuron states from other adjacent neurons to yet other neurons in a message passing type of procedure.

The interconnects from a neuron to the set of neurons with which it communicates are stored within the neuron cube with the corresponding cube addresses. In the learning process, these values are established algorithmically (possibly iteratively) as a function of the states desired in the operational mode. This is done internally to the ANN, for example, by imposing desired states on a class of neuron cubes, letting the ANN compute the states at some other group of neuron cubes, and computing the difference of this value and the states desired. This error is then used to alter the interconnect weights to reduce or compensate for this error.

A neuron state is typically computed as the (interconnect) weighted sum of connected neuron states nonlinearly altered using some memoryless nonlinearity such as a sign function or a (biologically motivated) sigmoid. The conversion to an electrical signal of the state possibly involves scaling of the state value and generation of a destination address (each neuron contains within it an address locator number which may be used to designate its position within the neuron array) if required. Retransmission of adjacent state signals is done using a messenger function. They are employed to distribute state signals from a first neuron which generates the signal to another neuron (or a plurality of neurons) not adjacent to the first neuron.

The function of retransmission is employed to simulate the action of biological neurons which have a high degree of connectivity to numerous other neurons, some at great distance from the source neuron.
In any physical geometry of electronic neurons, this connectivity aspect represents a real problem. Allowing autoconnects, for example, in a 10x10x10 neuron array, it is possible to require up to one million interconnection paths in some algorithms. Wiring such a set of interconnections is clearly extremely difficult physically.

In the structure outlined here, all interconnects among non-adjacent neurons are performed by having other neurons retransmit the sending state signal until the signal reaches its destination. Additionally, it is possible for a signal to be broadcast to a defined subset of all neurons, or even all neurons, via specially encoded messages. This is taken care of in the address portion of the signal. As a simple example, one neuron may transmit a signal to one full layer of the array with a single transmission properly encoded with address information. Or, it could address all elements of the array at once.

In cases where a neuron typically communicates with a very large number of other neurons, the interconnects may also provide for a global communications path. Such a path would consist of an electrical interconnection common to all neurons (or perhaps a large subset of all neurons), which would facilitate the transmission of a signal from any one neuron so connected to all other neurons on the common connection, simultaneously. The design would require fault tolerance to any failure of a neuron on the interconnect which might 'hog' or clamp the global interconnect, rendering it useless. Fortunately, as with biological neural networks, such fault tolerance is characteristic in many ANN algorithms.

Algorithms for inter-neuron communication need to be designed to facilitate such relayed state information. Alternatively, each neuron could also contain a separate communications processor, perhaps hard wired in silicon (i.e. not implemented in software) for higher speed. The microcomputer would then be free to compute its new state from its existing state and new transmissions received from other neurons.

Each neuron must thus contain a communications handler whose purpose is to receive, redirect, and generate state signals. Each neuron must also contain a computational element for computing state changes, and for applying weights to signals received from other neurons and also perhaps to weight its own outgoing signal. It must contain memory for program storage, which may be in the form of read-write, read-only, or read-mostly memory. It must contain read-write memory for storing parameters associated with changes in state and state weighting functions.

Neuron addresses may be either programmed permanently into each neuron prior to assembly of the array, or, preferably, would be self-programmed on power-up of the array. For example, a neuron cube in the top left corner could through internal software ascertain its position simply via the fact that certain of its sides are not connected to other cubes. It could then communicate to adjacent cubes its position, allowing adjacent neurons to determine their locations and hence addresses. The process can propagate automatically through the entire array until completed and all neurons have assigned
themselves addresses; the addresses would be stored in read-write memory or read-mostly memory in each neuron.

The interconnects may be simple mechanical contacts, perhaps spring loaded, which touch and make contact with adjacent neurons. If each neuron is a cube having 12 edges and 6 faces, then each neuron may communicate with up to 18 adjacent neurons. A neuron cube with corners modified and connectors placed on the corners may communicate with up to 26 adjacent neurons (Fig. 3). Power may be obtained from these connectors as well. External power applied to the sides of the array would flow through these interconnects.

One primary characteristic of a neuron is its reprogrammability, in the sense that the other neurons it communicates with may be reprogrammed to be more or less restrictive. A neuron may "grow" communications paths to other neurons during a learn cycle, or similarly destroy such paths. It may also modify state weights on its own. Also, it may be desirable to modify the actual structure of the microcomputer program, either on its own through a learning process or through external intervention. For example, during development of a neural network computer the cubes may require program modification. A human programmer may then create a new microcomputer program and load this program into the array. Since neurons imbedded deeply in the array are unreachable by direct electrical contact, the program may be 'downloaded' into each neuron via the retransmission process, or into just a subset of the array. A single neuron may be used as an entry node to facilitate the downloading. The programs may be loaded into the array via a conventional computer. Weights and communications paths may also be loaded into the array on a neuron by neuron basis if required by a similar process.

The ability to download neural information may be complemented by an 'upload' feature used to extract all neuron state and program information, especially information and programming of a variable nature. This is a critical feature for saving neural state information permanently onto hard media, such as a magnetic or optical disk. On power down of the network, all such information may be otherwise lost. Also, if a neural network is to be replicated in mass production with specific programming, such uploads are crucial to extracting the information required for duplication. Only then can the extracted information be reprogrammed into one or more other similar neural networks which, for example, may utilize a higher speed operational mode dedicated architecture. If this process cannot be performed, it may be required to unnecessarily teach each network individually, a process which can be tedious and impractical. The upload/download techniques are a form of cloning akin to software duplication of a conventional computer's programs and information.

Another related issue is fault tolerance. If thousands of neurons are employed in a network, failures of neurons are inevitable. The software in each neuron must be designed to tolerate failures. For example, a communications failure of a single neuron may block transmission of messages among many other neurons. Considerable thought must be given to making communications automatically
reroutable if such failures occur. It is possible to design a neuron algorithm such that an adjacent neuron could ‘take over’ the functioning of a bad neuron.

Since each neuron contains a digital computing element, it is possible for each neuron to simulate a number of neurons at once. The 5x5x8 array shown may actually be made to simulate not 200 neurons but 800 neurons if each neuron cube simulates the action of four neurons. Communications among such ‘internal’ neurons may be facilitated with appropriate software. Communications among neurons would be quite similar except that additional burden would be placed on the inter-cube electrical connections.

Signals external to the array must be interfaced in such a manner as to permit large amounts of data throughput. The sides of the array and the open connections found on the sides may be so used. Both data input and output may be so facilitated. It is also possible to focus an image of data on one or more sides of the array by incorporating photodetectors and appropriate detection electronics into neurons on each such side. Alternatively, special cubes may be affixed to each such side with photoreceptive properties, and little or no neural simulation ability. Energy fields other than light may also be used, such as microwave, sound, radiation, etc.

INVENTIVE ASPECTS

The inventive aspects of the proposed neural network we believe include but are not limited to the following:

1. A design for a neural network comprising a plurality of three dimensional structures or cells, each such cell having an ability to electrically interconnect on a plurality of sides or edges of each such cell and each having an ability to simulate the characteristics of a neuron to varying degrees of modification.

2. An ability to construct an arbitrary stacking of such cells into an array essentially without restriction or limit except for a requirement of physical contact with adjacent cells of similar type.

3. An ability of each cell within the array to electrically communicate one or more of programs, data, or commands, the cells in general having an ability to originate, retransmit, receive and reconfigure as a function of such communications.

4. Several electro-mechanical means for interconnecting cells by stacking, involving one or more of: compression mated contacts, plug-together mechanisms, adhesive mating methods, or magnetic attraction.

5. A communications interconnection among cells which permits global or large-subset transmissions among cells, without
November 24, 1987

Les E. Atlas  
Assistant Professor, Electrical Engineering  
401 Electrical Engineering Building, FT-10  
University of Washington  
Seattle, WA 98195

Dear Les:

Physio-Control Corporation is interested in supporting your work in artificial neural networks. We recognize that your research in this area is part of an ongoing project within the Washington Technology Center, and we plan to give an unrestricted gift of $10,000 to the Washington Technology Center at the University of Washington to supplement this research.

We approve of matching funds from the National Science Foundation Presidential Young Investigator Award and we approve of the publication of our participation in funding your work.

We are looking forward to many potential applications of artificial neural networks in solving challenging problems in science and industry.

Sincerely yours,

PHYSIO-CONTROL CORPORATION

Tom Lyster  
Senior Research Engineer  
TDL/msm

cc: Clif Alferness  
   John Adams
requiring the retransmission function among cells.

6. An ability of each cell to perform computations on data received from other cells within the array or external to the array. A further ability of each cell to originate communications to one or more other similar cells, the communicated data or programming being dependent on an algorithm and on the nature of communications from other cells prior to the communication.

7. An ability for cells to self-determine their locations within an array by an algorithm and the communications means.

8. An ability for such an array and its component cells to propagate programs and data from an external source, either to all cells in an array or to a subset thereof.

9. An ability for such an array and its component cells to have programs and data extracted from it via an external computer or controller, either for storage, analysis, or duplication purposes.

10. The use of specially designed or programmed interface cells on one or more faces of the array, engineered to permit communications to and/or from external sources. The further use of light or other radiative means to couple either into or out of such cells in order to simplify the task of connection, and the use of radiatively active transducers such as phototransistors and light emitting diodes to facilitate such external interface coupling.

11. An ability of functional cells to ignore malfunctioning cells via communications methods and algorithms governing the communications paths. A further ability of other cells to simulate the functions of malfunctioning cells if required.

12. An ability of a cell to simulate more than one neuron via computational algorithms, and to communicate information from such simulations to other cells in the array via similar communications means.

Disclosed by the undersigned this day, ______________, 1988.

Harold Phillipp
Dear Tom,

I would like to thank you and the Physio-Control Corporation for the $10,000 gift to help support my research at the Washington Technology Center. Our work in artificial neural networks will greatly benefit from the needed help and, I hope, future collaboration in problems of mutual interest. I am now in the state of research where the identification of important applications of this new neural network technology is crucial. The fast and accurate automatic identification of temporal patterns such as ECG signals is one of these applications. I look forward to speaking with you about this application in the future and would be willing, of course, to present more talks on artificial neural networks at your company.

I have put you on the mailing list for weekly seminars which our research group (the Interactive Systems Design Lab) hosts. These seminars are held at 3:30 PM Wednesdays when classes are in session. As you will see in the upcoming announcements, many of these seminars relate to artificial neural networks. Please let me know if anyone else at Physio-Control would like to be on this mailing list.

Sincerely,

Les Atlas
Les Atlas, Asst. Professor

Phone:(206)545-1315

cc: Prof. Ed Stear, Director, Washington Technology Center
    Prof. Robert Marks, Director, Interactive Systems Design Lab
Date: December 11, 1987

To: Prof. Stear, Washington Technology Center, FH-10

From: Prof. Atlas, Electrical Engineering, FT-10

Subject: Gift from Physio-Control

I have received a $10,000 check from Physio-Control for my unrestricted use in artificial neural networks research within the Washington Technology Center. This form of funding is very appropriate for our current research direction. While it would be hard for us to offer short-term deliverables to industrial sponsors, the potential for industrial support is very high. Many other companies have recently expressed an interest in gift support to maintain and enhance our research program in order to "keep a foot in the door" of artificial neural networks. I therefore intend to pursue (with the help of Bob Marks) putting together a Neural Network Research Consortium to formalize this gift program. If it is possible, I would like the gift account which is established by this check to be general enough to incorporate future gifts without new budget numbers.

cc: Prof. Robert Marks
    Prof. Robert Porter
Hi!

Regard,

I will still make a first pass at the patent, and then hopefully have Tom look at it and so forth.

I've also asked the mechanical design group (Stratos on Capitol Hill) to forward one of their recommend, an incredible CAD system, and some excellent design experience. They come highly recommended.

PN with Brochures to you, as I mentioned, they have a high level contact at Microsoft, access to individual (happily), you will need to modify it depending on who you are talking to (private formal). A disk with the intro is also on his way in Microsoft.

Attached is the nondisclosure agreement.

Bob,

Seattle, WA
Department of Electrical Engineering
University of Washington

Dr. Robert J. Marks II

September 22, 1988

FX: (206) 746-0566
(206) 746-1642
Bellevue, Washington 98005
13219 Northing Way, Suite 203

Philipp Technologies Corporation
1. This Agreement shall remain in force and effect for one (1) year from the effective date hereof described by the date attested hereto by the party last signing this Agreement.

2. The Agreement shall be governed by and construed in accordance with the laws of the State of Washington.

3. If a dispute arises as to the accuracy of any party's obligations under this Agreement, the dispute shall be submitted to arbitration in accordance with the rules of the American Arbitration Association and judgment of any award of decision shall be binding on the parties.

4. Confidential Information. No license under any patent or copyright included in this Agreement is granted unless specifically provided in this Agreement.

5. Right of Each Party to Use and Transfer the Confidential Information is grantable, and the right of each party to use and transfer the Confidential Information is expressly granted to each party.

6. Legal counsel in good faith, may prepare and file suit, or take any other action, that they deem necessary to enforce the terms of this Agreement.

7. Non-disclosure Agreement

Non-disclosure Agreement

I.$
November 4, 1988

Dr. Dmitry Kaplan
208 Mountain Park Boulevard
Apt. E302
Issaquah, WA 98027

Dear Dmitry:

Here’s the BAA from China Lake and the SDI effort. If you do call either Swenson (China Lake) or Bromley (SDI), and you actually talk to them, please mention that you’re talking about the project that I called them about so they know that they don’t have to call me back.

Let’s get some contracts, have fun and get rich!

Best personal regards

Robert J. Marks II
Professor & all round swell guy
Introduction

**OVER THE COURSE OF THE TWENTIETH CENTURY, A**

few ambitious initiatives have captured the imagination and intellect of the nation's leading scientists and engineers: the Manhattan project, Apollo moon missions, and, now, the Strategic Defense Initiative (SDI).

SDI's goal—to eliminate the nuclear threat—demands the best and brightest. Its enabling technologies, spanning advanced computing, materials, propulsion and energy sources, create exciting opportunities for researchers. These include:

- The opportunity to contribute to a critically important defense science initiative.
- The opportunity to work with leaders in academia, government, and industry on next-generation technologies.
- The opportunity to pursue promising innovations.

IST nurtures and supports programs related to the SDI mission from fundamental research into scientific feasibility of concept, to exploration of engineering feasibility, to demonstrating practicability.

By providing a responsive, flexible, and stable management structure, IST has fostered innovation. It provides the direction, coordination, and funding necessary to carry out a large-scale diversified research effort.

We welcome the interest and involvement of scientists and researchers. Specific program administration information, as well as a list of Science and Technology Agents, begins on p.4. The program summaries and case examples provide an in-depth look at the types of innovation sought by IST.
THE STRATEGIC DEFENSE INITIATIVE ORGANIZATION was created to explore the development of a defense system envisioned by President Reagan in his address to the nation on March 23, 1983.

To fulfill its mission, SDIO is organized into two primary areas, "technology" programs and "systems" programs. The Innovative Science and Technology Office is the technical directorate within SDIO tasked with seeking out innovative approaches to all aspects of ballistic missile defense. It funds research in these approaches and assures that the other technical directorates within SDIO are apprised of new results and breakthroughs from IST programs.

The IST office has several roles. First, it establishes a technology base for strategic defense via fundamental research conducted in universities, government and national laboratories, small businesses, and large industries. Second, it brings infant technologies to a stage where they can be validated. These technologies either transition into applications or go on the shelf for future exploitation. Third, the IST Office administers the SDIO Small Business Innovation Research Program.

THE CURRENT RESEARCH PROGRAM SUPPORTED BY IST focuses on six general areas:

- High speed computing
- Sensing, discrimination and signal processing
- Space power and power conditioning
- Directed and kinetic energy concepts
- Materials and structures
- Propulsion and propellants

Other areas may be added in the future. While the range of programs is broad, they all fulfill the IST criteria: each is directed toward revolutionary (not evolutionary) advances; each relates to some aspect of the strategic defense system—its architecture, weapon system or sensing components, or command and control; each blends the best thinking in academia, government and industry.

Vacuum outgassing experiments help scientists understand electricity behavior in the ionosphere.
Scientists at the Naval Research Laboratory are coordinating their studies of ultra-short wavelength lasers with research being conducted at the University of Rochester, University of Texas, Stanford, and Physics International Company—as well as x-ray and gamma-ray laser applications being demonstrated by innovative small businesses.

At the Jet Propulsion Laboratory (JPL), researchers have achieved major gains in computing speed using a "hypercube" network of multiple computers working simultaneously on different pieces of a problem. JPL researchers are also studying neural networks as a faster, fault-tolerant alternative to conventional numerical processing.

The Innovative Nuclear Space Power Institute (INSPI), a consortium of universities and small businesses, promotes new technologies to meet the needs of SDI space platforms for efficient, lightweight, high-power energy sources. INSPI is concentrating its efforts in two most promising areas: the gas core reactor and TRICE (thermonic reactor inductively coupled elements).

These examples of innovation and multidisciplinary teamwork are the norm, not the exception, at IST. They exemplify the importance IST places on its mission of disseminating technical knowledge.

Looking Ahead

IST's efforts aim at the development of an effective strategic defense system. As the architecture of the United States' SDI system emerges and moves toward deployment, IST's role will adjust accordingly. Some research programs will become more narrowly focused on key enabling technologies to support SDI system specifications.

IST's research findings will also extend beyond strategic defense. Advances in computing, sensing, and electronic materials will contribute to the nation's entire defense effort, as well as to civilian applications in industries like electronics, telecommunications, and automotive. Likewise, specialized lasers being developed under IST sponsorship will find medical diagnosis and treatment applications. The benefits of advanced energy and propulsion technologies will flow to NASA, commercial space ventures, and consumer products.

In a very direct way, IST's efforts are key to the nation's technology future—as a catalyst for scientific and technical achievements of the 21st century.
TECHNICAL MANAGEMENT OF THE WIDE DIVERSITY OF IST-sponsored research is conducted by the directorate's Science and Technology Agents (STAs). The STAs are affiliated with such defense research agencies as the Office of Naval Research, the Air Force Office of Scientific Research, and the Army Research Office. Administration, procurement, and reporting are generally carried out by the parent agencies of the cognizant STAs.

The STAs are the official representatives of SDIO and IST. Generally research will be funded by IST only with STA review and recommendation. Thus, proposals and inquiries should be sent to the appropriate STA, not to the IST directorate office. Each program description includes the name, address and phone number of the responsible STA. The information is also summarized in Appendix A of the brochure.

Researchers wishing to propose IST-sponsored projects should identify a program component. Then, they should directly contact the appropriate STA to initiate a dialogue regarding IST support. If, as a result of this dialogue, an STA determines that a proposed effort is of interest to SDIO, the researcher will be encouraged to follow up with additional documentation—for example, a white paper or a formal proposal.

When an investigator wishes to propose a program in an area for which no STA is identified, a brief two-page summary of the program should be sent to IST. The summary should stress the innovative nature of the proposed work, its relationship to perceived SDIO needs, and potential results. Appendix B of the brochure contains a sample format.

IST encourages early contact with STAs regarding novel and innovative concepts or approaches in any scientific or technology discipline applicable to the Strategic Defense Initiative. We are seeking revolutionary advances that can have high payoffs in enhancing strategic defense—and, we are seeking broad participation from the entire research community.

The following pages describe IST program components and the crucial technology challenges they address—challenges that point the way to tomorrow's technology frontiers.
Contents

HIGH SPEED COMPUTING
6 Optical Computing and Optical Signal Processing
7 Parallel Processing
7 Mathematical Methods and Algorithms
8 Self Adaptive Processing and Simulation

SENSING, DISCRIMINATION AND SIGNAL PROCESSING
9 Detectors for Sensing and Discrimination
10 Optical Sensors
10 Reliable Advanced Electronics
11 Integrated Detection Estimation and Communication Theory
12 Laser Satellite Networking
13 Boost Phase Detection
14 Terahertz Technology
15 Interactive Discrimination

SPACE POWER AND POWER CONDITIONING
16 Non-Nuclear Space Power and Power Conditioning
17 Advanced Pulse Power Physics
18 Nuclear Space Power
19 Advanced Electro-Chemical Prime Power

DIRECTED AND KINETIC ENERGY CONCEPTS
20 Electromagnetic Propagation and Directed Energy Concepts
21 Short-Wavelength Chemical Lasers
22 High-Power Microwave Sources
23 Advanced Beam Combining Concepts
23 Advanced Accelerators
24 Particle Beams
25 KE Interceptor Integration
26 Ultra Short Wavelength Lasers
26 Propagation Through Disturbed Environments
27 Mid-Atmospheric Effects

MATERIALS AND STRUCTURES
28 Advanced Composite Materials
29 Electronic and Optical Materials
30 Diamond Technology
30 Electronic-Materials Interfacing
31 Optical Glass and Macromolecular Materials
32 Space Structures and Dynamics
33 High Pressure Metastable Materials
33 Optical Sensor Survivability
34 Superconducting Materials
35 Interactive Space Technologies

PROPULSION AND PROPELLANTS
36 Electric Propulsion
37 Advanced Propellants
38 Low Emission Propellants
39 APPENDIX A
41 APPENDIX B
INNOVATIVE RESEARCH PROGRAM COMPONENTS

HIGH SPEED COMPUTING

Optical Computing and Optical Signal Processing

STA: Dr. William Miceli

Office of Naval Research
495 Summer Street
Boston, MA 02210-2109
(617) 451-3172

Objective:
Optical computing refers to the exploitation and application of suitable optical technology within a computational environment. This program is predicated upon the inherent parallelism of optical systems and addresses all computational aspects associated with the SDI, particularly sensor signal processing, target/decoy discrimination, and the data management functions associated with BM/C3. This research addresses all aspects of optics, opto-electronics, and acousto-optics applicable to analog signal processing, digital computing, and biologically inspired neuromorphic computing—commonly referred to as neural networks.

Program Description:
The program consists of research efforts in the following areas:

- Optical (Analog) Signal Processing
- Optical Digital Computing
- Optical Neural Networks

Emphasis is placed upon devising suitable processing architectures in each of these areas, as well as developing the technology base necessary to optically implement these architectures.

Opportunities:
The following topics have been identified as particular areas of interest:

- 2-dimensional arrays of bistable optical devices with acceptable power dissipation, packing densities, speed, signal/noise, fabrication costs, etc.
- Spatial light modulators with suitable data formatting, speed, modulation depth, power consumption, optical quality, etc.
- Dynamically reconfigurable optical interconnections between cascaded 2-dimensional arrays of optical devices. The need for real-time holographic elements, preferably with gain, is anticipated.
**Objective:**
The proposed SDI mission requires improvement in the performance of current sensor technology and signal processing. This research intends to provide extremely high throughput computing techniques for use in SDI signal processing.

**Program Description:**
The program's primary research centers on the investigation of algorithmically specialized systolic processors. Unlike past systolic work, study is being conducted on system level issues. Efforts to expand the systolic style of design into the middle ground between algorithmically specialized devices and general purpose parallel machines are being considered. One technique already in this middle ground is the programmable systolic array.

**Opportunities:**
Future studies will continue to exploit systolic processor technology for advancement in signal processing. Approaches which will be investigated will include SATCOM and control computations. Because of long SDI mission times and the likelihood for serious battle damage, fault tolerance techniques will be pursued. Approaches include reconfiguration in a wafer scale integration context and behavior-based fault detection at the system level.

---

**Mathematical Methods and Algorithms**

**STA:** Dr. Jagdish Chandra

**Army Research Office**
P.O. Box 12211
Research Triangle Park, NC
27709-2211
(919) 549-0641

**Objective:**
The command, control, and data manipulation phases of proposed SDI systems demand state-of-the-art parallel supercomputers, algorithmic processes and technologies. This research intends to explore applicability of related fields in large scale scientific computing.

**Program Description:**
The program concerns itself with studies in the following areas:
- Parallel methods in high-speed computing
- Systolic algorithms for signal processing
- High resolution imaging
- 3-dimensional robotic vision and shape recognition
- Segmentation and image detection in natural images.

**Opportunities:**
SDI requires the use of supercomputers and parallel computers in the design, testing, and implementation phases. Software for these systems needs extensive algorithmic development and should be able to run on a
variety of parallel architectures. Accomplishments in signal and image processing center on algorithmic improvements.

Special interest lies in the interaction between mathematical methods and algorithms, and systolic array architectures (in reference to large-scale parallel computation). Specific targets for the investigation of large-scale optimization problems with parallel computers will include:

- Understanding the problem
- Formulation of mathematical models appropriate to parallelism
- Development of appropriate computer languages, data structures and implementations for parallelism and adaptivity on various parallel architectures
- Development and implementation of systolic algorithms for linear algebra
- Signal processing
- High resolution imaging and switching applications
- Development and implementation of parallel algorithms for graphical display and animation of solutions.

| Self Adaptive Processing And Simulation | STA: Mr. Doyce Satterfield |

Objective:
This research intends to accomplish a two-fold task. It will devote part of its efforts to the study of innovative concepts for advancing processing technology for system self-modification while deployed in a dynamically evolving threat environment. In addition, research will be conducted to facilitate the simulation of the SDI battle management and C² network. The effort will demonstrate the advanced features of:

- Automated model generation
- Automated analysis of simulation results
- Goal-directed instrumentation
- Integration of adaptive hardware and software with the simulation(s).

Program Description:
The program focuses on the self-adaptive simulation research. Modeling of the entire BM/C² system will require the correct modeling of subsystems, the various threats, and the multidimensional environment. This effort will entail millions of lines of code and necessitate advancement of the state-of-the-art in computer-aided model generation, instrumentation of models, and simulation analysis.

Opportunities:
Future study will deal with the extensive SDI C² network with its many local nodes. Advancements in intelligent communicating agents that would reside at these local nodes are sought, since having the agents reside at the nodes will allow for adaptation to changing environments. Other required technologies and approaches include:

- Search, acquisition, and track networks using self-adaptive processing for in-circuit reconfigurability of logic and architecture
- Adaptive hardware and software for self-diagnosis, self-repair, and self-modification during operation.
A single construction element is shown in the lower left of the Figure 1. The construction element as shown in Figure 1 is similar to a single construction element of the array of construction elements shown in Figure 1. The construction element is shown in Figure 1 is similar to a single construction element of the array of construction elements shown in Figure 1. The construction element is shown in Figure 1 is similar to a single construction element of the array of construction elements shown in Figure 1.

Thus, these channels would be displaced in some manner through their channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1. Other additional channels may be displaced by simply adding another channel in Figure 1.
also be simply cemented together or adhered via any of a number of commercially available means, or through the attraction of magnets imbedded in each cube.

The ANN will operate in three modes: programming, learning and operational:

1. The type of ANN architecture to be used is established in the programming mode. The operations here include establishment of the set of neurons to which a given neuron is (directly or indirectly) connected and the (sigmoidal) nonlinearity to be used by the neuron.

2. In the learning mode, the interconnect weights among neurons are established using training data or, in certain applications such as combinatorial search problems, some training algorithm. When training data are used, some or all of the neurons are assigned certain states. The interconnect weights are then determined internal to the ANN by algorithms both known and yet to be discovered. In certain training algorithms, the initial interconnect weights are algorithmically specified by, say, a random number generator.

3. In the operational mode, the neuron cubes perform three primary functions: a) computation of the neuron state which is a function of the neurons to which it is connected, b) conversion of the neuron's state into an electrical signal, c) retransmission of neuron states from other adjacent neurons to yet other neurons in a message passing type of procedure.

The interconnects from a neuron to the set of neurons with which it communicates are stored within the neuron cube with the corresponding cube addresses. In the learning process, these values are established algorithmically (possibly iteratively) as a function of the states desired in the operational mode. This is done internally to the ANN, for example, by imposing desired states on a class of neuron cubes, letting the ANN compute the states at some other group of neuron cubes, and computing the difference of this value and the states desired. This error is then used to alter the interconnect weights to reduce or compensate for this error.

A neuron state is typically computed as the (interconnect) weighted sum of connected neuron states nonlinearly altered using some memoryless nonlinearity such as a sign function or a (biologically motivated) sigmoid. The conversion to an electrical signal of the state possibly involves scaling of the state value and generation of a destination address (each neuron contains within it an address locater number which may be used to designate its position within the neuron array) if required. Retransmission of adjacent state signals is done using a messenger function. They are employed to distribute state signals from a first neuron which generates the signal to another neuron (or a plurality of neurons) not adjacent to the first neuron.

The function of retransmission is employed to simulate the action of biological neurons which have a high degree of connectivity to numerous other neurons, some at great distance from the source neuron. In any physical geometry of electronic neurons, this connectivity aspect represents a real problem. Allowing autoconnects, for example, in a 10x10x10 neuron array, it is possible to require up to one million interconnection paths in some algorithms. Wiring such a set of interconnections is clearly extremely difficult physically.

In the structure outlined here, all interconnects among non-adjacent neurons are performed by having other neurons retransmit the sending state signal until the signal reaches its destination. Additionally, it is possible for a signal to be broadcast to a defined subset of all neurons, or even all neurons, via specially encoded messages. This is taken care of in the address portion of the signal. As a simple example, one neuron
may transmit a signal to one full layer of the array with a single transmission properly encoded with address information. Or, it could address all elements of the array at once.

In cases where a neuron typically communicates with a very large number of other neurons, the interconnects may also provide for a global communications path. Such a path would consist of an electrical interconnection common to all neurons (or perhaps a large subset of all neurons), which would facilitate the transmission of a signal from any one neuron so connected to all other neurons on the common connection, simultaneously. The design would require fault tolerance to any failure of a neuron on the interconnect which might 'hog' or clamp the global interconnect, rendering it useless. Such fault tolerance is characteristic with biological neural networks.

Algorithms for inter-neuron communication need to be designed to facilitate such relayed state information. Alternatively, each neuron could also contain a separate communications processor, perhaps hard wired in silicon (i.e. not implemented in software) for higher speed. The microcomputer would then be free to compute its new state from its existing state and new transmissions received from other neurons.

Each neuron must thus contain a communications handler whose purpose is to receive, redirect, and generate state signals. Each neuron must also contain a computational element for computing state changes, and for applying weights to signals received from other neurons and also perhaps to weight its own outgoing signal. It must contain memory for program storage, which may be in the form of read-write, read-only, or read-mostly memory. It must contain read-write memory for storing parameters associated with changes in state and state weighting functions.

Neuron addresses may be either programmed permanently into each neuron prior to assembly of the array, or, preferably, would be self-programmed on power-up of the array. For example, a neuron cube in the top left corner could through internal software ascertain it position simply via the fact that certain of its sides are not connected to other cubes. It could then communicate to adjacent cubes its position, allowing adjacent neurons to determine their locations and hence addresses. The process can propagate automatically through the entire array until completed and all neurons have assigned themselves addresses; the addresses would be stored in read-write memory or read-mostly memory in each neuron.

The interconnects may be simple mechanical contacts, perhaps spring loaded, which touch and make contact with adjacent neurons. If, for example, every other layer in the cube structure was phased as illustrated in the top of Figure 5, then each cube makes physical contact with 12 adjacent cubes. Sides of 14 adjacent cubes can be made to have physical contact if adjacent rows in a layer are phased as is illustrated at the bottom of Figure 5. If similar phasing is applied to the hexagonal structure in Figure 2, then each unit will also make contact with 14 other units.

Alternately, communication among construction elements can be done optically thereby eliminating the need for transmitting signals through mechanically coupled interconnects. (Note that, however, unless power can be provided internal to the construction element or through some other externally applied field, mechanical interconnects would still be required to provide power.) As is shown if Figure 6, optical sources, such as LED’s, would be aligned to optical detectors at the construction element’s surface through a skin of optically transparent material. Inter-element communication could be established by any one of a number of commonly used modulation techniques.
The flow of signals must be organized in such a fashion as to avoid collision of moving packets of information. For artificial neural network algorithms that require each neuron to communicate with every other neuron, this can be achieved by alternating signal flow directions as is illustrated in Figure 7. At one instance, communication can be with neuron elements in a specified direction. In the next communication cycle, this direction would change. The technique can also be modified for the less severe case to algorithms where a neuron is only required to be connected to each neuron in an adjacent layer.

One primary characteristic of a neuron is its reprogrammability, in the sense that the other neurons it communicates with may be reprogrammed to be more or less restrictive. A neuron may "grow" communications paths to other neurons during a learn cycle, or similarly destroy such paths. It may also modify state weights on its own. Also, it may be desirable to modify the actual structure of the microcomputer program, either on its own through a learning process or through external intervention. For example, during development of a neural network computer the cubes may require program modification. A human programmer may then create a new microcomputer program and load this program into the array. Since neurons imbedded deeply in the array are unreachable by direct electrical contact, the program may be 'downloaded' into each neuron via the retransmission process, or into just a subset of the array. A single neuron may be used as an entry node to facilitate the downloading. The programs may be loaded into the array via a conventional computer. Weights and communications paths may also be loaded into the array on a neuron by neuron basis if required by a similar process.

The ability to download neural information may be complemented by an 'upload' feature used to extract all neuron state and program information, especially information and programming of a variable nature. This is a critical feature for saving neural state information permanently onto hard media, such as a magnetic or optical disk. On power down of the network, all such information may be otherwise lost. Also, if a neural network is to be replicated in mass production with specific programming, such uploads are crucial to extracting the information required for duplication. Only then can the extracted information be reprogrammed into one or more other similar neural networks which, for example, may utilize a higher speed operational mode dedicated architecture. If this process cannot be performed, it may be required to unnecessarily teach each network individually, a process which can be tedious and impractical. The upload/download techniques are a form of cloning akin to software duplication of a conventional computer’s programs and information.

Another related issue is fault tolerance. If thousands of neurons are employed in a network, failures of neurons are inevitable. The software in each neuron must be designed to tolerate failures. For example, a communications failure of a single neuron may block transmission of messages among many other neurons. Considerable thought must be given to making communications automatically reroutable if such failures occur. It is possible to design a neuron algorithm such that an adjacent neuron could ‘take over’ the functioning of a bad neuron.

Since each neuron contains a digital computing element, it is possible for each neuron to simulate a number of neurons at once. The 8x5x4 array shown may actually be made to simulate not 160 neurons but 640 neurons if each neuron cube simulates the action of four neurons. Communications among such ‘internal’ neurons may be facilitated with appropriate software. Communications among neurons would be quite similar except that additional burden would be placed on the inter-cube electrical connections.

Signals external to the array must be interfaced in such a manner as to permit large amounts of data throughput. The sides of the array and the open connections found on