A Neural-Network Packet Switch Controller: Scalability, Performance, and Network Optimization

Keith J. Symington, *Student Member, IEEE*, Andrew J. Waddie, Mohammad R. Taghizadeh, and John F. Snowdon

***Abstract*—This paper examines a novel combination of architecture and algorithm for a packet switch controller that incorporates an experimentally implemented optically interconnected neural network. The network performs scheduling decisions based on incoming packet requests and priorities. We show how and why, by means of simulation, the move from a continuous to a discrete algorithm has improved both network performance and scalability. The system's limitations are examined and conclusions drawn as to its maximum scalability and throughput based on today's technologies.**

***Index Terms*—Neural networks, optical diffraction, optimization methods, packet switching.**

I. Introduction

THE IMPLEMENTATION of neural-network hardware in a practical and scaleable manner remains an obstacle to its deployment. Algorithmically, a neural network may prove to be infinitely scaleable, but if component interconnection requirements and signal to noise ratios are incorporated into the design not only can the hardware quickly become unfeasible to build, but unforeseen results may become apparent in the network itself. This paper examines the mapping of a neural-network algorithm onto optoelectronic hardware and describes how algorithmic adaptation has resulted not only in hardware minimization but also significantly improved network characteristics.

The neural network referred to here is designed to solve the assignment problem. In this case the problem is to maximize the use of available resources thus optimizing bandwidth utilization in a packet switch. It is interesting in two respects. First, it is unusual to use a neural network as packet switch controller. Such tasks are generally tackled by custom built digital hardware. Second, the network contains *optoelectronic components* which are used to provide the high degree of interconnectivity required [1], [2]. This approach makes the neural network highly scaleable.

It is important to note that applying this neural-network solution to the scheduling problem [3]–[5] can (in terms of mean packet delay) outperform digital schedulers [6] such as iSLIP [7] at higher load levels. Although our current hardware is not built for the high speed of an individual switch cycle, this is purely due to our system being designed as a proof of principle.

Manuscript received June 11, 2001; revised March 21, 2002. This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) under Grant GR/M22956, "Parallel optoelectronic controller for telecommunications switching networks."

The authors are with the Department of Physics at Heriot-Watt University, Edinburgh EH14 4AS, U.K. (e-mail: kjsymington@iee.org).

Digital Object Identifier 10.1109/TNN.2002.806646

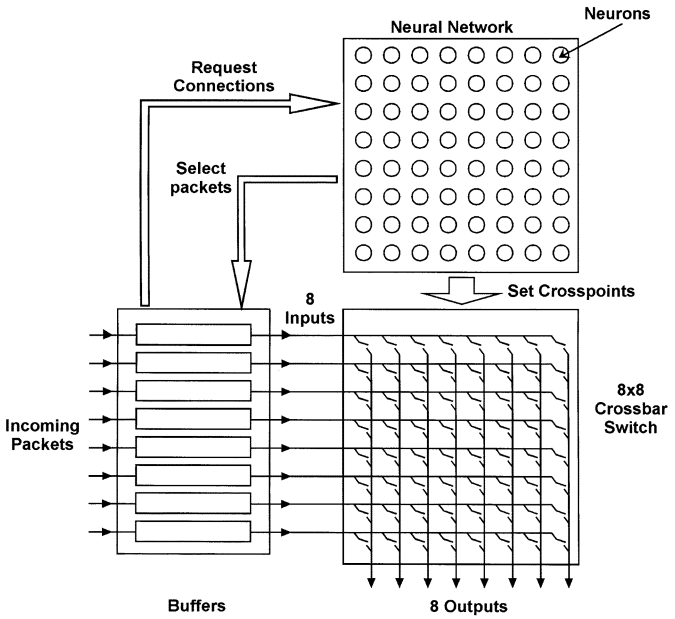

Fig. 1. Schematic of experimental neural-network crossbar switch controller. Based on the connections requested by incoming packets, the neural network chooses an optimal solution, sets the appropriate crosspoint switches, and then selects the chosen packets for transmission.

Indeed, the simplicity of the system is such that extremely high speed versions of the hardware are easily conceivable. This fact, combined with the encouraging simulation results presented herein on scalability, suggest that this system will provide an excellent packet switch scheduler.

All simulations are based on experimental component parameters to assess the scale to which neural-network hardware could be constructed given today's optoelectronic component tolerances.

II. Switching Problem

Packet switches are very common in computer networks and telecommunications systems such as asynchronous transfer mode (ATM) networks. With the exponential increase in traffic from sources such as the Internet, efficient packet switching is becoming a very important issue.

The crossbar switch is a common type of packet switch and can be seen in Fig. 1. In this switch every incoming packet is buffered and its requested output line examined. To rout the packet successfully it must be transmitted through the crossbar switch to its destination by closing the correct crosspoint. This operation is mutually exclusive in that any input or output lines

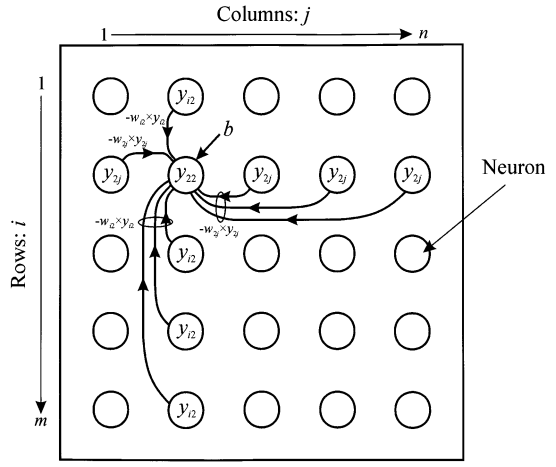

Fig. 2. Neural-network interconnection pattern mapped to an $N = 5$ element crossbar switch. Neuron y_{22} receives inhibitory input from all other neurons in the same row and column. Note that $m = n = N$.

that are in use cannot be reused, so two simultaneous requests for the same output line will result in one packet being blocked regardless of the routing algorithm used.

The neural-network architecture discussed here [6], [8]–[11] creates a near optimal solution by examining the requested output lines of all buffered packets and selecting a set of packets to be transmitted such that mean packet delay is minimized. The neurons are arranged in a two-dimensional array where each neuron corresponds directly to a crosspoint switch. If a crosspoint connection is requested by an incoming packet, the appropriate neuron is allowed to evolve. This neuron will attempt to inhibit all other active neurons in the same row and column, as shown in Fig. 2, thereby activating its associated crosspoint switch.

We can, therefore, define the network's updating rule

$$x_{ij}(t) = i_{ij}\left(x_{ij}(t-1) + \lambda_{ij} \cdot \left(-A\sum_{k\neq i}^{m} w_{kj}y_{kj} - B\sum_{k\neq j}^{n} w_{ik}y_{ik} + b\right)\right) \quad (1)$$

where x_{ij} is the summation of all inputs to the neuron referenced by ij. Inhibitory inputs from other neurons are represented by y_{ij} and multiplied by a fixed synaptic weight w_{ij}. A, B and b are two neuron optimization parameters and a bias, respectively. Evolution of the network is controlled by both λ_{ij}, which is a time constant for the neuron, and i_{ij}, which determines whether a connection has been requested and its priority. Alteration of the latter parameter i_{ij} allows adaptation of the neural network to cope with changing traffic patterns, thereby implementing prioritization. The neuron's final output y_{ij} is then determined using a monotonic activation function $y_{ij} = f(x_{ij})$ whose nature, be it continuous or discrete, will be discussed in this paper.

This neural network is derived from the Hopfield and Tank model [12] but is in fact a winner take all (WTA) network. Given that it solves a generalization of the assignment problem, algorithms such as the traveling salesman problem (TSP), processor load balancing, etc., can also be readily mapped. Indeed, the network has already been adjusted and tested on a banyan switch configuration with a high degree of success [6].

III. System Hardware

This project has seen the construction of two demonstrators. The first generation demonstrator was designed as a proof of principle to show that a neural network could be successfully constructed using optoelectronic components. The second generation demonstrator added programmability, and thereby flexibility, enabling network adaptation to differing problems. This created a generalized optoelectronic neural network capable of solving the assignment problem whose potential scalability and performance are simulated here.

All neural networks consist of a set of neurons, interconnected in an application specific manner, which perform some sort of transfer function on the summation of a set of input values each multiplied by an appropriate weight. Our hardware divides this functionality into two domains: one optical and one electronic (Fig. 3).

Interconnection and fixed weighting is performed optically using a diffractive optic element (DOE) [13]–[16] while a digital signal processor (DSP) calculates the transfer function electronically.

A. Optical System

In the optical domain, each neuron has control of one optical output channel in the form of a vertical cavity surface emitting laser (VCSEL) and one optical input channel in the form of a detector. Both output and input elements are arranged in a two dimensional array with each element corresponding to a crosspoint switch. Therefore, the optical interconnection scheme must allow interaction with other neurons in the same row and column. Fig. 4 shows how an output VCSEL signal is split by the DOE and inhibits all neurons in the same row and column by applying a signal to their detector inputs.

Note that there is no signal present on the central detector as this would result in self-inhibition and prevent the neural network from evolving. Therefore, the DOE stores the weight matrix for the neural network. Adapting the DOE can change both network interconnection and weighting, thus allowing different algorithms to be tackled. The interconnection pattern formed by the DOE is shift invariant. That is, no matter where the optical beam is incident, an identically proportioned pattern will be created relative to the position of this beam. Thus, the neurons must have an identical interconnection pattern, thus requiring adaptation of any neural-algorithm to the hardware. Such an approach separates this system from optical matrix vector solutions [17] by allowing two-dimensional arrays of optical emitters and detectors, thereby reducing hardware size and improving optical spatial bandwidth. An electrical implementation would require a separate wiring network for each neural output leading to an quadratic increase in routing complexity as network size increases.

This optical system makes use of inherent optical properties to aid convergence and perform input summation. Increasing the amount of light incident on a detector gives a near linear

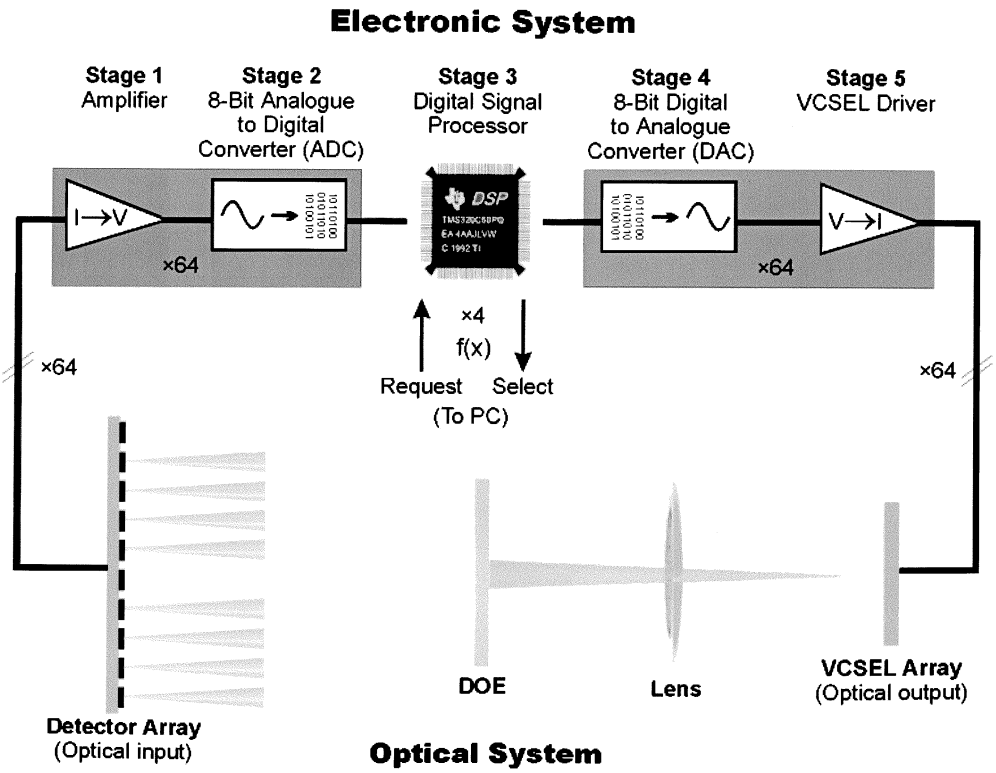

Fig. 3. Outline of the system currently under construction showing both optical and electronic pathways. This version is capable of routing eight input lines to eight output lines ($N = 8$, 64 channels in total).

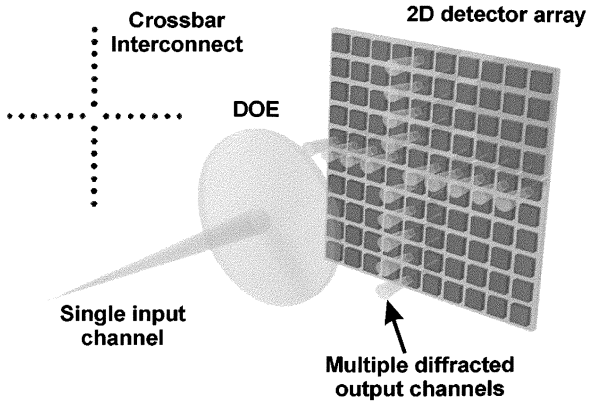

Fig. 4. Interconnect pattern formed by a single input channel incident on the DOE. All detectors and thereby associated neurons in the same row and column receive an inhibitory signal from incident light: thus the missing zeroth order—self inhibition would prevent network evolution.

increase in the detector's output signal. This results in analog summation of all beams that are incident on a single detector. Quantization error during conversion into the electronic domain leads to noise but neural networks tend to be tolerant to noise in many applications—indeed this system requires noise to aid convergence.

B. Electronic System

In the second-generation system, the electronics were modularized into five stages (Fig. 3). Stage 1, the optical input, is a detection system which converts a current generated by light incident on a photodiode into a voltage of magnitude specified by the preset amplification level of a transimpedance amplifier. The second stage is an analog-to-digital converter (ADC) which converts the voltage received from the first stage into digital information (normally 8 bits) and multiplexes 64 analog channels through eight octal ADC chips. The third stage consists of a Texas Instruments DSP which takes the digital information from the second stage and performs a transfer function based on previous and current neuron inputs. There are four DSPs in this system each handling 16 neurons (or channels) with each DSP under the control of a master controller (personal computer). The fourth stage consists of eight octal digital-to-analog converters (DACs) that are fed the new activation levels from the third stage and convert this information into appropriate voltages. The fifth and final stage turns a voltage into a drive current for the VCSELs, thus returning the signal into the optical domain. A single chip solution has been fabricated for stage five.

The use of off-the-shelf digital signal processors in what appears to be an analog loop has a number of advantages. First, interfacing of DSPs with input buffers and packet switch control logic is relatively easy. Second, analog component tolerances biased network convergence in the first generation demonstrator preventing certain neurons from winning against others. This equates to a "favoritism" of certain neurons, and thereby paths, which is unacceptable in an actual packet switch scheduler. Finally, the ability to alter neuron characteristics allows network functionality to be adjusted and new applications tackled with minimal reconstruction of hardware.

IV. Simulation and Results

Even though our first generation demonstrator had proven the viability of this system, detailed simulation was considered

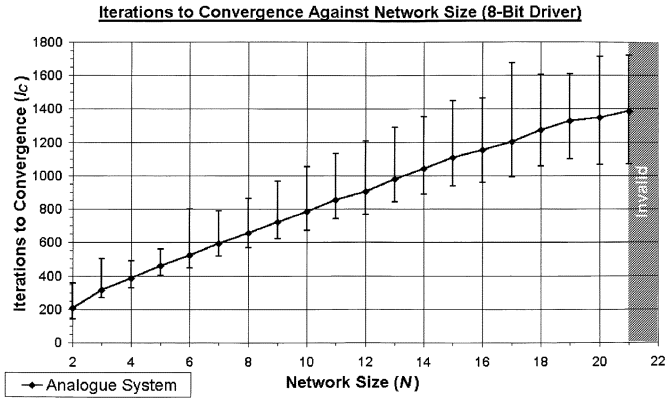

Fig. 5. Scalability of an 8-bit (analog like) driver which has a linear increase in the average number of iterations required as N increases. Average convergence times are indicated by the curve with error bars showing minimum and maximum outliers.

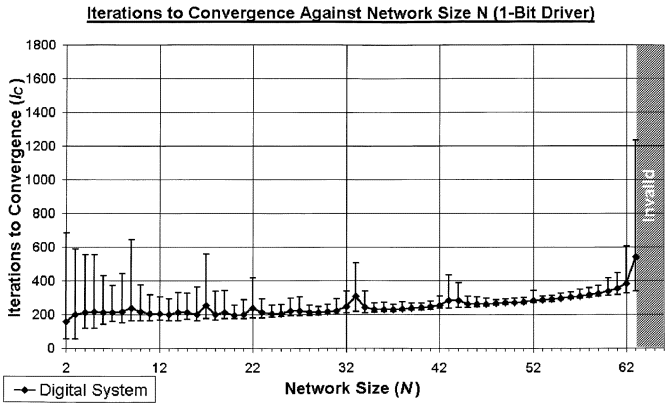

Fig. 6. Examination of the scalability of a digitally driven VCSEL system shows that convergence time is not greatly affected by network size. Average convergence times are indicated by the curve with error bars showing minimum and maximum outliers.

necessary to establish scalability limitations, potential hardware problems or possible optimizations in the design of the second generation. Scalability is quantified in terms of switch size, N. If a crossbar switch is of size $N = 8$ then it has eight input channels, eight output channels and $N^2 = 64$ neurons. Performance is quantified in terms of iterations to convergence I_C. Iterations to convergence are the number of complete system iterations required for the neural network to converge to a steady state. If the slowest component in the system performs its required operation at 10 MHz, be it VCSEL, detector, ADC or neural electronics, then 10×10^6 iterations can be performed per second, since the system is pipelined. Of these iterations, perhaps 400 may be required for the neural network to converge.

The simulation included component characteristics for conversion between analog (optical) and digital (electronic) domains. All conversion accuracies are expressed as bit depths so, for example, a value of two indicates $2^2 = 4$ discrete levels or eight gives $2^8 = 256$ levels. Q_{DAC} is the bit depth of digital-to-analog conversion and Q_{ADC} that of analog-to-digital conversion. Q_{ADC} was determined to be the largest source of random noise due to a quantization error of $\pm 1/2$ least significant bit (LSB) which is larger than normal background noise. This has been experimentally measured as true so long as $Q_{\mathrm{ADC}} < 16$.

A. Variation of Network Response With Q_{DAC}

During the initial design phase, a set of predicted component values were specified to give good performance without compromising functionality. An 8-bit DAC was selected as this gave reasonable, but not excessive, resolution. As expected, the average number of iterations required for convergence I_C scaled in a linear manner with respect to network size N as in (2)

$$I_C = 62N + 146. \tag{2}$$

This is graphed in Fig. 5 as a line intersecting the average convergence times for various network sizes.

Given current optical system component values, the upper size limit proved to be a crossbar switch of $N = 21$ (441 neurons). These simulations were done under full network load conditions with each point on the graph the average of 1000 network runs. Degradation thereafter resulted in the neural network producing invalid switch configurations when optical component signal to noise ratios were insufficient for the state of a single neural input to be reliably determined. Under such circumstances, the network still converges, however its output is not usable due to frequent errors.

Characterization of neural network response at various optical output bit depths was carried out in an attempt to find a relation between maximum scalability, convergence time and offered load. It was found that reduction of the bit depth to $Q_{\mathrm{DAC}} = 1$ gave an unexpected and significant result: the neural network converged faster and became more scalable. Such a bit depth can be considered the same as applying a hard threshold transfer function in the neuron [18] and will be referred to as driving the optical output digitally. Simulation has shown that digital driving of the neural network in this manner will allow the construction of networks of up to $N = 63$ (3969) neurons using optical components with the same characteristics as before. This is graphed in Fig. 6 and shows that the average number of iterations required, I_C, increases marginally with network size.

However, at $N = 63$ the number of iterations increases dramatically, after which point the network begins to fail.

The reason the digital system converges faster is that instead of gradually turning on, as the analog system does, the neurons switch from nothing to fully on in one step thus adding impetus to convergence [19]. Digital driving of the optical system also removes the need for both DACs and analog VCSEL drivers. It appears that a reduction in hardware confers increased scalability and a rapid decision.

Simulation of both analog and digitally driven networks was also performed under varying network load conditions. This was started at zero (0% or no connection requests) and increased until a load of one was generated (100% or a connection request for every crosspoint on the crossbar switch). Under full load conditions the number of neurons on should be N, however suboptimal solutions do arise with $N-1$ neurons on. Fig. 7 details the results for the analog network and Fig. 8 those for the digital.

In both graphs, the left axis indicates the number of iterations to convergence I_C and the right axis the average number

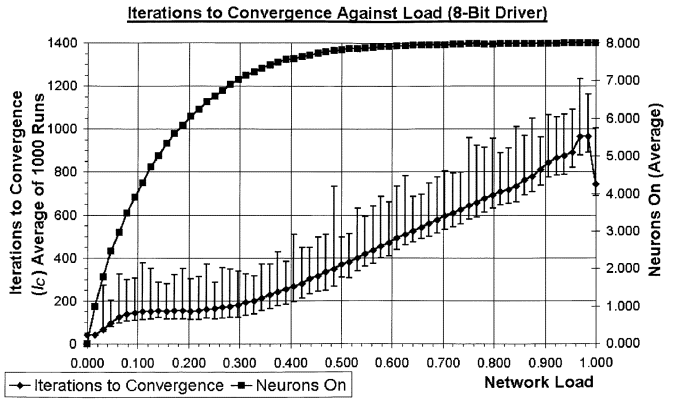

Fig. 7. Performance of an 8-bit analog-like driver under varying load conditions. Averages indicated by the curve with error bars showing minimum and maximum outliers.

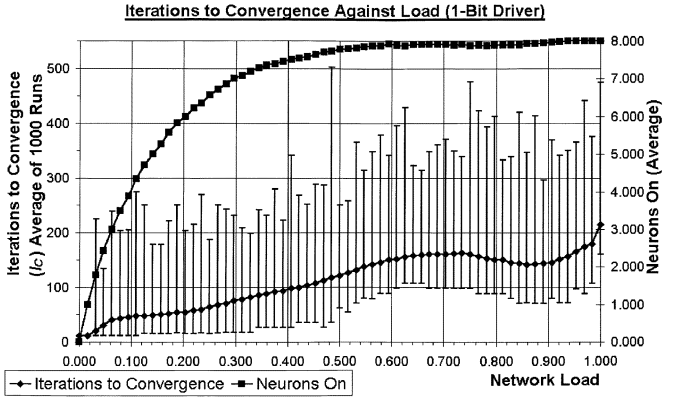

Fig. 8. Performance of a digitally driven network under varying load conditions. Averages indicated by the curve with error bars showing minimum and maximum outliers.

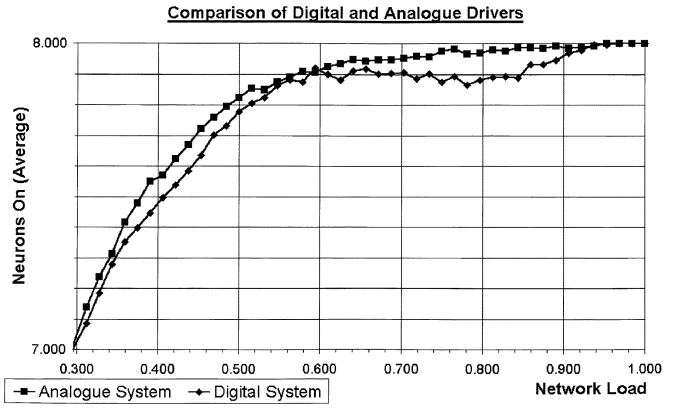

Fig. 9. Comparison of quality of solution between digital and analog systems. Averages indicated by the curve.

of neurons in an "on" state all averaged over 1000 test runs. The digital network copes particularly well as load increases with the number of iterations I_C required increasing slowly. It outperforms the analog network in all ways as far as convergence is concerned. However, speed of convergence and scalability are traded off against quality of solution. Expanding the second y-axis in Figs. 7 and 8 to give Fig. 9 shows that, on average, suboptimal solutions were generated by the digital system a little more often than by the analog one. Under high load conditions, the analog system gives a solution that is below optimal (one less active neuron/crosspoint) about 3% of the time whereas the digital system about 8%–10% of the time. This is because driving digitally prompts a faster descent through the solution space by leaving parts of it unexplored.

B. Variation of Network Response With Q_{ADC}

The amount of light incident at a neuron's optical input stage is converted into a voltage and then to a digital value with a bit depth of Q_{ADC} An attempt was made to define the number of bits required to successfully operate the system. This was done by varying Q_{ADC} along with offered load and examining the number of iterations required for convergence I_C.

When using a lower resolution ADC (in this case $Q_{\mathrm{ADC}} = 6$) is was possible to produce valid results a little beyond what was believed to be its resolution limit. This was discovered to be dependent on initial system request values. If a system starts off unbalanced then there is a distinct solution or descent gradient such that noise is no longer the major contributing factor driving convergence. Such systems are capable of converging with ADCs that have less resolution than theoretically required. The hypothesis is that any noise present can push them in one direction only. Values of $Q_{\mathrm{ADC}} < 6$ with $N = 8$ did not produce valid results as the resolution was no longer high enough to discern whether a neuron's optical output was on or not.

Also observed was that when using an accurate, higher resolution ADC (in this case $Q_{\mathrm{ADC}} = 16$) only a few neurons were active but the network did not produce a solution near optimal, i.e., it was not converging correctly. This is because if the system starts off balanced (full matrix) then noise is a very important factor in stimulating convergence. As bit resolution increases, LSB noise decreases and so the system fails to converge within the maximum number of iterations (in this case 5000)—indeed a complete lack of noise prevents the system from working at all. This problem is solved by adding noise which resulted in the system beginning to converge again.

Therefore, Q_{ADC} must be of sufficient resolution to clearly determine the state of a single neuron's optical output. However, excessive resolution will suppress noise and slow convergence critically. Results indicate that for an $N = 8$ network, $Q_{\mathrm{ADC}} = 8$ is a good compromise.

C. Validity and Stability of Design

Simulation was undertaken to prove that the distribution of solutions was of a normal nature given that the addition of noise was linear. This would indicate the presence of an attractor to which the network converged and therefore underline the relevance of an average convergence time. Without such an attractor, the neural network could be considered as unstable [20]. Each point on this graph (Fig. 10) represents the number of occurrences of iterations to convergence I_C, and, therefore, one complete simulation, within a certain window.

The normal curve can be seen to be positively skewed around an average number of iterations (bold line). Varying of network load consistently gave a normal distribution.

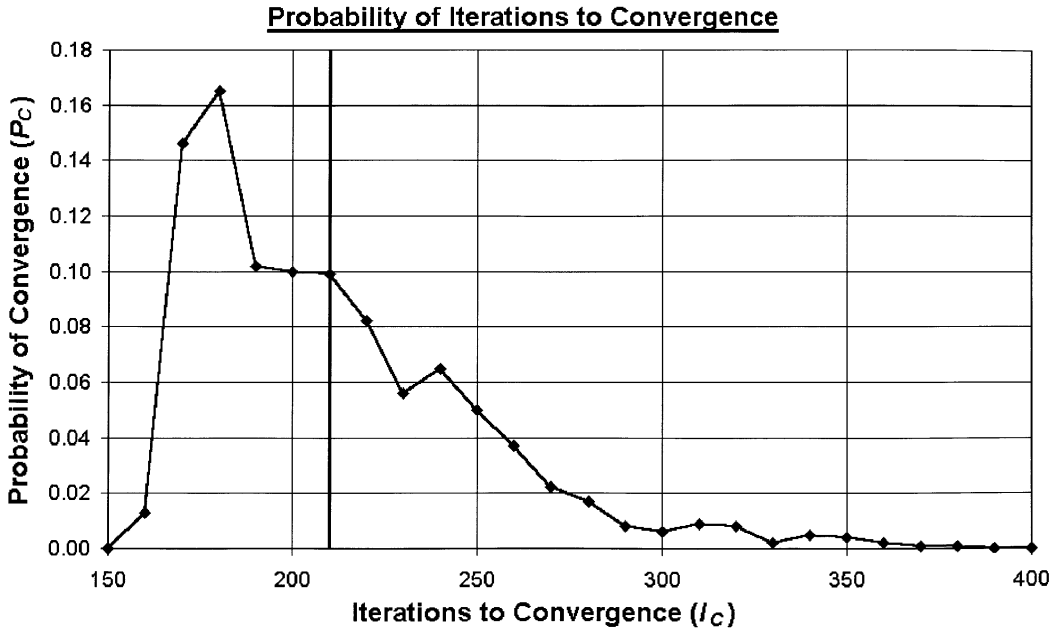

Fig. 10. Positively skewed normal curve indicates the presence of an attractor. Multiple network iterations have an average convergence time as indicated by the bold line.

D. Hardware

Although our results in this paper have been rigorously extrapolated from experimental data, it should be noted that there are restrictions on *current* scalability due to features of commercially available components. For example, VCSELs are only readily available at $N = 8$ at present (although $N = 16$ and $N = 32$ exist in the laboratory frame) due to fabrication difficulties and power dissipation problems. Similarly, DOE technology probably limits us to $N = 30$ due to noise considerations—again primarily due to fabrication difficulties. Photodetector arrays are relatively simple to acquire and highly scalable. There are obviously limits to the number of channels the electronics can handle, but this type of DSP solution was chosen for flexibility. A dedicated application specific integrated circuit (ASIC) or field programmable gate array (FPGA) solution would be necessary for a real application and is not envisaged as a bottleneck at present. It is presumed that the latest generation of fabrication techniques, when applied to these optoelectronic components, will lead to significant relaxation of the current system bounds.

V. Conclusion

These results show how nonintuitive algorithmic adaptation can bring hardware minimization, improved performance, and better scalability to an otherwise cumbersome system. The reason that digital thresholding converges faster is because instead of a neuron gradually being switched on, as the analog system does, the digital system switches from nothing to fully on in one step. This adds impetus to convergence but unfortunately also a hastiness that can lead to suboptimal results more often than in the analog case. Nevertheless, the possible hardware savings accrued by bypassing DACs and analog VCSEL drivers for a more rapid decision lead, in themselves, to performance enhancement.

Driving digitally, an $N \approx 60$ network would produce a solution every 400 iterations under maximum load. On the other hand, an 8-bit analog like system could be scaled up to $N = 21$ but would have a decision time of $\sim$1800 cycles at this level. The advantages of a digitally driven system are clear, provided that a suboptimal solution of $N-1$ neurons on can be tolerated less than 10% of the time under higher load conditions. If we were to implement such a system with a 1 GHz cycle time at $N = 8$, a fully loaded neural network (with convergence in a generous 400 iterations) could produce 2 500 000 million solutions per second. This compares favorably with current packet switch schedulers [6]. The impressive thing about such a system is that although hardware cost increases, decision time remains virtually constant as switch size N increases. On the other hand, existing algorithms such as iSLIP require an increasing number of iterations as modeled by $\log_2 N$.

Given the ability to reconfigure the interconnect, this neural network is capable of solving many variations on the assignment problem. What has been demonstrated here is how careful problem mapping to regular weight matrices and optimization can lead to viable and scalable hardware implementations.

Acknowledgment

The authors would like to thank S. Eitel and K.-H. Gulden of Avalon Photonics for supply of the VCSEL arrays used in these demonstrators.

References

[1] R. P. Webb and A. W. O'Neill, "Optoelectronic neural networks," *Brit. Telecom Technol. J.*, vol. 10, pp. 144–154, July 1992.
[2] R. P. Webb, "Optoelectronic implementation of neural networks," *Int. J. Neural Syst.*, vol. 4, pp. 435–444, Dec. 1993.
[3] A. Marrakchi and T. Troudet, "A neural net arbitrator for large crossbar packet switches," *IEEE Trans. Circuits Syst.*, vol. 36, pp. 1039–1041, July 1989.

[4] P. S. Rodríguez-Hernández, F. J. González-Castaño, J. M. Pousada-Carballo, and U. M. García-Palomares, "Stochastic neural scheduler for real-time input-buffered packet switching," *Electron. Lett.*, vol. 35, no. 16, pp. 1313–1314, Aug. 1999.
[5] J. M. Pousada-Carballo, F. J. González-Castaño, P. S. Rodríguez-Hernández, and U. M. García-Palomares, "High performance real-time neural scheduler for ATM switches," *IEEE Commun. Lett.*, vol. 4, pp. 372–374, Nov. 2000.
[6] R. P. Webb, A. J. Waddie, K. J. Symington, M. R. Taghizadeh, and J. F. Snowdon, "An optoelectronic neural network scheduler for packet switches," *Appl. Opt.*, vol. 39, no. 5, pp. 788–795, Feb. 2000.
[7] P. Gupta and N. McKeown, "Designing and implementing a fast crossbar scheduler," *IEEE Micro*, vol. 19, no. 1, pp. 20–28, Jan./Feb. 1999.
[8] K. J. Symington, J. F. Snowdon, A. J. Waddie, T. Yasue, and M. R. Taghizadeh, "Optoelectronic neural networks," in *Proc. Conf. Postgraduate Res. Electron., Photon. Related Fields*. Nottingham, U.K.: Inst. Elect. Eng., Apr. 2000, pp. 182–187.
[9] K. J. Symington, A. J. Waddie, T. Yasue, M. R. Taghizadeh, and J. F. Snowdon, "High performance optoelectronic neural network scheduler," in *Proc. Opt. Comput. 2001*. Lake Tahoe, NV: Opt. Soc. Amer., Jan. 2001, pp. 12–14.
[10] P. W. Protzel, D. L. Palumbo, and M. K. Arras, "Performance and fault-tolerance of neural networks for optimization," *IEEE Trans. Neural Networks*, vol. 4, pp. 600–614, July 1993.
[11] J. Ghosh, A. Hukkoo, and A. Varma, "Neural networks for fast arbitration and switching noise reduction in large crossbars," *IEEE Trans. Circuits Syst.*, vol. 38, pp. 895–904, Aug. 1991.
[12] J. J. Hopfield and D. W. Tank, "'Neural' computation of decisions in optimization problems," *Biol. Cybern.*, vol. 52, pp. 141–152, 1985.
[13] P. Blair, "Diffractive optical elements, design and fabrication issues," Ph.D. dissertation, Dept. Physics, Heriot-Watt Univ., Edinburgh, U.K., 1995.
[14] A. Vasara, M. R. Taghizadeh, J. Turunen, J. Westerholm, E. Noponen, H. Ichikawa, J. M. Miller, T. Jaakkola, and S. Kuisma, "Binary surface-relief gratings for array illumination in digital optics," *Appl. Opt.*, vol. 31, pp. 3320–3336, 1992.
[15] K. Ballüder and M. R. Taghizadeh, "Optimized phase quantization for diffractive elements using a bias phase," *Opt. Lett.*, vol. 24, no. 23, pp. 1576–1578, 1999.
[16] J. W. Goodman, *Introduction to Fourier Optics*, 2nd ed. New York: McGraw-Hill, 1996.
[17] D. Casasent, "Optical processing in neural networks," *IEEE Expert*, vol. 7, pp. 55–61, Oct. 1992.
[18] J. J. Hopfield, "Neural networks as physical systems with emergent collective computational abilities," *Proc. Nat. Acad. Sci.*, vol. 79, pp. 2554–2558, 1982.
[19] S. V. B. Aiyer, M. Niranjan, and F. Fallside, "A theoretical investigation into the performance of the Hopfield model," *IEEE Trans. Neural Networks*, vol. 1, pp. 204–215, June 1990.
[20] S. Haykin, *Neural Networks—A Comprehensive Foundation*. New York: Macmillan, 1994.

Keith J. Symington (S'00) received the Ph.D. degree in optically interconnected computing systems in 2002.

He is currently a Research Associate with the School of Engineering and Physical Sciences at Heriot-Watt University, Edinburgh. U.K. His current research is in the optical interconnection of processor to memory systems and the addition of optical interconnects to FPGAs.

Dr. Symington is a Member of the Institution of Electronic Engineers (IEE).

Andrew J. Waddie was born in Edinburgh, U.K. He received the B.Sc. degree in natural philosophy and physics from the University of Edinburgh in 1990, the M.Sc. degree in computing and computer modelling of optoelectronic materials and devices from the University of Newcastle-upon-Tyne, U.K., in 1991, and his Ph.D. degree in optoelectronic neural networks from Heriot-Watt University, Edinburgh, in 1996.

Since 1996, he has been employed as a Research Associate in the Diffractive Optics group of the Department of Physics (now the School of Engineering and Physical Sciences), Heriot-Watt University. His research interests include diffractive optical element design and fabrication, optoelectronic systems, and optical beam shaping.

Dr. Waddie is a Member of the Optical Society of America, the Society of Photonic and Instrumentation Engineers, and the Institute of Physics.

Mohmammad R. Taghizadeh is a Reader in Physics at the School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh, U.K. He is the head of the Diffractive and Micro-Optics Group, which he established in 1980s. He has more than 220 publications in the areas of diffractive optics, optical interconnects, and optical computing in various refereed journals and/or in the international conference proceedings.

Dr. Taghizadeh is a Member of the Optical Society of America and the SPIE.

John F. Snowdon is a Lecturer in Physics at Heriot-Watt University, with 15 years of experience in optoelectronic systems. He runs the Optically Interconnected Computing (OIC) Group and has been a Principal Investigator in EPSRC, EU, and industrially funded projects. The expertise of this group is in using optoelectronic techniques to defeat bottlenecks in computational and communication systems. The OIC group has a strong experimental program, which has been responsible for the design of a series of optoelectronic demonstrator systems. The group has an acknowledged leading international role with invited and contributed papers at all major conferences in the field. He has published more than 50 papers in these areas.