Technical Final Report
Construction of a Connectionist Network Supercomputer
University of California, Berkeley
ONR URI Grant No. N00014-92-J-1617

1 Abstract

This report provides an indepth examination of the accomplishments of the recently completed CNS project. Research contributions of this project span a wide range of fields: digital microsystems design, computer architecture, multiprocessor system design, neuromorphic analog processing, parallel programming languages, and neural network applications. Though numerous conference presentations and peer-reviewed journal articles, these advances have had broad impact in the university, industrial, and government communities. The research project also served to train many graduate students in a variety of disciplines in computer science, electrical engineering, and related scientific disciplines.

2 CNS: A Five Year Overview

Our primary motivation for the CNS project sprung from a simple question: given the neural network paradigms in common use in 1991, would neural networks with several million parameters be useful for engineering applications and neuroscience research?

Experimenting with networks of this size on real problems was not feasible with off-the-shelf computing hardware available at the start of the project. Neural network researchers, working by themselves with commercial hardware, could not experimentally attack the question. However, we believed that a team of computer system designers and neural network experts could answer this question, by:

- Defining a flexible, general-purpose computing architecture for neural computing. This architecture should included provisions for real-time input from real-world sensors.

- Designing a hardware implementation of this architecture that would support the simulation of neural systems with millions of weights, at a peak computation rate of 1 billion connection updates per second.

- Defining software tools and programming techniques that enable application-level programmers to realize the full computational power of this hardware.

- Providing a complete computing solution to the CNS applications community based on these new technologies, that functions as a reliable platform for research.

- Construct dozens of copies of this prototype hardware, for use by a community of neural network researchers, and guide this community through experiments with multi-million-parameter systems.

Judged by this list of goals, the CNS project was a success. During its five year lifespan, we:

- Created the CNS computing architecture, featuring a specialized microprocessor of our own design, T0, and specialized analog pre-processing chips for real-time auditory input.

- Designed and fabricated the T0 microprocessor, a 750,000 transistor CMOS chip that ran at 40 Mhz, and manufactured hundreds of copies of this fully-functional part.

- Constructed over 50 complete SPERT computing systems, and distributed these systems to active researchers in the neural-network community.

- Created a complete software development and operating system environment, to support neural-network application development on T0 and SPERT.

- Developed and distributed worldwide the object-oriented programming language Sather and pioneered several basic ideas in parallel software for irregular problems in the pSather version.

- Created complete neural-network applications that used SPERT systems. These applications used neural networks with millions of parameters to solve significant engineering problems.

- And finally, by combining multiple copies of the SPERT system, we produced a multiprocessing neural network acceleration system, TetraSPERT. TetraSPERT is capable of simulating a neural network with 5 million weights, while achieving a computational throughput of over a billion connection updates per second. We ported complete applications to TetraSPERT, closing the loop between supercomputer system design and neural network research.

The preceding description highlights the project engineering success of the CNS project: chip designs completed, numbers of machines fabricated, etc. However, as an academic research group, our primary mission is not project engineering. Exploring new approaches to problems, creating new engineering techniques, and formulating scientific knowledge are the true missions of an academic research project. The research contributions of the CNS project span several disciplines in computer science and electrical engineering, as detailed below.

2.1 Computer Architecture

At the beginning of the CNS project, the literature on digital systems for neural networks focused on embedding specific neural training and evaluation algorithms in architectures

optimized for maximum throughput. To begin our project, we studied the suitability of this specialized acceleration to an important application for the CNS research community – state-of-the-art speech recognition using neural networks.

We found that speech recognition applications spend a significant fraction of execution time performing algorithms unrelated to neural networks – tasks such as speech preprocessing and hidden Markov model state decoding. These tasks would not be able to use specialized neural network hardware – and thus the total speedup of the speech recognition applications would be limited, regardless of how fast the neural-network hardware computed.

This observation – which we dubbed "Amdahl's Law for Neurocomputing" – drove the CNS project in a new direction, towards hardware architectures that accelerated neural computations in a general-purpose framework. We found that a classical machine architecture for scientific computation – a vector unit combined with a fast scalar processor – could be reworked to serve the needs of the CNS project. Compared to other approaches to acceleration, a vector processor is an easy platform for both compilers and application programmers to target. Our reworking of the vector machine concept for the CNS project, embodied in our T0 microprocessor architecture, features these novel enhancements:

- In addition to the standard arithmetic ALU operations, our vector ALUs include special opcodes optimized for fast neural network acceleration, including instructions which support the evaluation of many weight multiplications in parallel.

- Instead of the large floating-point formats of classical vector machines, our vector registers and ALUs support a lean fixed-point data format, with sufficient accuracy and precision for both neural-network and general signal processing tasks.

- Special data movement and combination instructions between vector registers and the scalar unit allow many algorithms to sustain the maximum bandwidth throughout a task.

- Support for efficient vector load and store operations over a high-bandwidth interface to off-chip memory.

These architectural concepts have found academic and industrial acceptance far beyond the neural-network acceleration community. The multimedia extensions to the Intel Pentium architecture, MMX, are a clear application of the "Amdahl's Law for Neurocomputing" principle of general-purpose architecture for special-purpose acceleration, targeted to graphics and multimedia applications. In addition, the vector architecture of T0 has led to an exploration of vector architectures for mainstream computing. For example, the Intelligent DRAM (IRAM) project, led by Professor Dave Patterson of UC Berkeley, adopts a vector architecture for embedded DRAM computing that shares many features with T0.

2.2 Digital VLSI Design

The T0 microprocessor was a major VLSI design effort: a 750,000 transistor chip in a 1.2u CMOS process, that achieved fully functional first silicon. Research advances were made in circuit design and microarchitecture as part of this VLSI effort.

As part of the T0 project, we developed a novel technique for high-speed chip-to-chip communication. We designed a serial link interface that transfers bits between chips at a rate of 550 Mbps. This interface implements a low voltage signalling strategy, using an on-chip voltage reference. Two delay-locked loops are used to maintain data synchronization. We fabricated a test chip using this interface technology, and measurements on the chip confirmed reliable communications performance at full speed (275 MHz clock, two bits per cycle, to yield 550 Mbps).

2.3 Software Design

In addition to VLSI chip design and board-level hardware design, software design was a key part of the CNS project. A full operating system was designed for SPERT, and included multiprocessing features to support TetraSPERT. A custom neural-network training package, QuickNet, was the key middleware application of the CNS project. This training system ran on SPERT, TetraSPERT, and conventional workstations, and was the interface speech-recognition applications used to access CNS machines.

Another software package developed during the CNS project was the PHiPAC (Portable High Performance Ansi C) system. PHiPAC is a system that generates optimal matrix-matrix multiply code for a specific machine architecture. As neural-network software makes liberal use of matrix-matrix operations, PHiPAC can be used to optimize the network training and evaluation speed on a particular machine architecture. We have facilitated technology transfer by offering PHiPAC distribution via the Internet; the package has attracted a substantial user base in the numerical software community.

2.4 Neural Network Applications

Continuous speech recognition was the driving application of the CNS project. The speech recognition algorithms used in the CNS project are a blend of neural networks (multi-layer perceptrons (MLPs) trained with the backpropagation algorithm for phonetic classification) and traditional techniques (hidden Markov models (HMMs) for word recognition).

The theoretical underpinnings of this hybrid HMM-MLP architecture were well understood at the start of the CNS project, and demonstration systems were developed for speech recognition tasks of moderate size. During the course of the CNS project, hybrid HMM-MLP systems were developed for very large, state-of-the-art speech recognition tasks. These systems performed as well as traditional speech recognition algorithms on these benchmark tasks, while using significantly fewer parameters. Motivated by the advantages of smaller

parameter sets (faster recognition time, smaller RAM requirements), several commercial speech recognition vendors have adopted our HMM-MLP architecture for their product lines.

The hardware developments in the CNS project were central to the successful development of large-vocabulary hybrid HMM-MLP recognizers. A key bottleneck in developing these recognizers is the training of large (1-10 million parmeters) neural networks to evaluate candidate algorithmic ideas. During the later years of the CNS project, each speech recognition researcher had a dedicated SPERT system, using the T0 microprocessor, installed in his or her workstation. In addition to these dedicated systems, the TetraSPERT system, with 4 SPERT cards, was a shared computational resource for the group.

This hardware enabled the fast training of these multi-million parameter MLPs, dramatically increasing research productivity. Algorithmic advances by CNS researchers in speech recognition includes:

- New algorithms to lessen the effect of variable speaking rates on recognition accuracy.

- New algorithms to lessen the effect of foreign accents on recognition accuracy.

- Several generations of speech pre-processing systems that improve recognition accuracy in the presence of background noisy environments.

- Several generations of adaptive techniques that lesson the the effect of a microphone's frequency response on recognition accuracy.

- Advanced formulations of the statistical foundations of hybrid MLP-HMM systems, which better capture the underlying structure of the speech code.

- Advancements in the parallelization of hidden Markov model state decoding, to improve the recognition time of very-large-vocabulary recognition systems.

- New methods of optimally combining the results of several different speech recognition alogorithms.

- New ways of applying psycholinguistic knowledge to speech recognition.

In addition to speech recognition, the CNS project also supported application developers in vision and image compression. UCB-based researchers in these fields were trained to target their applications to SPERT, and benchmarks were performed that showed the advantages of vector microprocessing on low-level vision applications.

2.5 Neuromorphic Analog VLSI Design

At the beginning of the CNS project, basic circuit techniques for analog VLSI implementations of neuromorphic systems were well established. Several research laboratories had

developed prototype chips that showed the promise of biologically-inspired approaches to vision and auditory sensory processing. Vision chips that combined photoreceptors and spatiotemporal post-processing had progressed through several generations of designs. Many auditory chips were also developed, including several generations of cochlear designs, as well as post-cochlear processing for pitch, spectral shape, and spatial localization.

These prototype chips, however, were designed primarily to prove out circuit and architectural techniques. A main challenge of the CNS project was to leverage this knowledge base, and create sensory chips that were useful as input devices for the CNS computing system. We focused on auditory sensory chips.

2.5.1 Special Purpose A/D Converters for Audio

The traditional way to add real-world input to a digital system is to use a general-purpose analog-to-digital (A/D) converter. To incorporate analog processing in this model, we needed to develop a replacement for general-purpose A/D's. We call these replacement parts *special-purpose analog-to-digital converters for auditory processing.*

Analogous to general-purpose A/D converters, special-purpose A/Ds take as input an analog audio signal, and produce a digital output suitable for computer input. However, the digital output is not digitized shape of the analog waveform. Rather, the audio input is first processed by analog circuits that perform operations unique to the application. The final representation is then digitized, using a representation that codes the signal in an efficient way. In addition to a digital output, the converters also receive digital input, to customize the analog processing for a specific application.

For the CNS project, we designed and fabricated special-purpose converters for speech and audio processing. These converters feature neuromorphic algorithms cast in analog circuits, including a silicon cochlea circuit, temporal adaptation circuits, and temporal autocorrelation circuits. The final output is coded using the event-address protocol, a technique optimized for the efficient digital transmission of neuromorphic representations. The output bus of the converters is cascadable, supporting systems that include several converters, each computing a different representation on the audio input in parallel. The converters include non-volatile analog storage to hold control parameters; this storage is programmable under digital input.

2.5.2 Speech Recognition Experiments Using Silicon Auditory Models

We built a multi-representation speech processing system, using three copies of our special-purpose converter chip. We interfaced this system to a workstation, and designed a real-time software system for data display and capture. This system let us speak into a microphone, and see real-time output of three different neural representations on the workstation screen. The workstation could also generate analog audio signals: by using the workstation to play digitized speech recordings into the converter system, we could do large-scale automated

experiments using our analog hardware.

Using this system, we designed a speech recognition system for a 13 word, speaker-independent, telephone-quality recognition task. We used our analog hardware for speech pre-processing, and used the MLP-HMM speech recognition software described earlier in this report to complete the system.

This hybrid speech recognition system performs well on this 13-word task, yielding a 4 percent error rate. Speech recognition systems with performance in this range are usable in commercial applications, if good error recovery strategies are employed. However, it is dissapointing to note that the best non-neuromorphic software-based speech pre-processing do significantly better on this task, yielding a 2 percent error rate. In addition, the neuromorphic front end does not do well when processing input speech corrupted by background noise, yielding very high error rates.

We believe these difficulties belie the shortcomings of joining neuromorphic sensory representations with traditional "back-ends" for speech recognition. These back-ends have been optimized over several decades, by hundreds of researchers, to work well with traditional signal-processing front-ends. It is not surprising that such back-ends work relatively poorly if interfaced with a radically different front-end approach. To realize the promise of the neuromorphic approach in speech recognition, a reformulation of the complete recognition process may be in order.

2.5.3   Micropower Speech Recognition

The computing architectures used in analog neuromorphic auditory processing have extraordinary power efficiency. For example, a 51-channel silicon cochlea consumes a few microwatts of power, a specification that supports years of operation of a small wristwatch battery. Could complete speech recognizers could be built using this circuit technology, and achieve a micropower dissipation? If so, the micropower advantage would be a sufficient reason to develop speech recognition systems using neuromorphic front ends, even if the recognition performance is no better than current software techniques.

We explored the architecture of a micropower speech recognition system, using the hybrid speech recognizer described in the last section as a model. This system includes two major algorithms which would require analog micropower implementation: neural networks and hidden Markov model state decoding. Several researchers have focused on micropower neural network technology, with encouraging results. In the CNS project, we focused on implementing hidden Markov model state decoding using micropower analog circuits. This research, done in collaboration with Richard Lippmann at MIT Lincoln Laboratories, resulted in the design, fabrication of a functional prototype for Baum-Welch state decoding with microwatt power consumption.

2.6 Parallel Languages and Programming

The parallel programming aspect of the project also made a great deal of progress, but was not as closely integrated as we had originally planned. In the early years, we hoped that the CNS could turn into a general purpose machine and designed the parallel Sather language to be compatible with the CNS design. This combined effort had a positive effect on both sub-projects, but the integration was not carried through because it was decided to make CNS more specialized and to retain Sather as a general purpose multi-platform system.

The Sather project on Object-Oriented programming and its parallel version, pSather, have both been quite successful. Many of the innovations pushed in the Sather project are now main stream including automatic storage management, separation of inheritance and subtyping and robust abstraction libraries. The widely used Java language incorporates these and other Sather features and some of our students and post-docs are playing a central role in future Java development. The Sather project trained four doctoral students and six post-docs in addition to having many shorter term visitors and collaborators. This effort has led to basic advances in our understanding of storage management involving caches, active thread management and the mapping of large irregular tasks to parallel computers.

3 Senior Personnel

One of the key senior staff providing critical and essential research contributions was Post-Doctoral Researcher John Lazzaro. John was responsible for setting directions for the research and was solely responsible for the computational aspects of the research.

4 Students

Students supported by this grant who have graduated with MS degrees were: William Chang (December 1997), Todd Hodes (December 1997), Nathan McNamara (May 1995), and David Stoutamire (December 1997).

Students who have graduated with their Ph.D. degrees were: Krste Asanovic (December 1997), David Bailey (April 1997), Yochai Konig (May 1996), Srinivas Narayanan (May 1997), and Thorsten von Eicken (December 1993).

Students still working on their research supported by this grant were: Jeffrey Bilmes, Christoph Bregler, Timothy Callahan, John Hauser, Randy Huang, Stylianos Perissakis, Warner Warren, and Su-Lin Wu.

5 Publications

5.1 Journals

Lazzaro, J., Wawrzynek, J., Lippmann, R. P. (1997). A micropower analog circuit implementation of hidden Markov model state decoding. *IEEE Journal Solid State Circuits* **32**:8, 1200–1209.

Lazzaro, J. P., Wawrzynek, J. (1997). Speech recognition experiments with silicon auditory models. *Analog Integrated Circuits and Signal Processing,* **13**:1-2, 37-51.

Murer, S., Omohundro, S., Stoutamire, D., and Szyperski, C. (1996). Iteration abstraction in Sather. *Transactions on Programming Languages and Systems,* Vol. 18, No. 1, pp. 1-15.

Wawrzynek J., Asanović, K., Kingsbury, B., Beck, J., Johnson, D., and Morgan, N. (1996). SPERT-II: A Vector Microprocessor System. *IEEE Computer.* Vol. 29, No. 3, pp. 79–86.

Bourlard, H., Hermansky, H., and Morgan, N. (1996). "Towards Increasing Speech Recognition Error Rates," Speech Communications, May, pp. 205-231.

Morgan, N., and Bourlard, H. (1995). Continuous Speech Recognition: An Introduction to the Hybrid HMM/Connectionist Approach. *Signal Processing Magazine,* pp 25-42, May 1995.

Morgan, N., and Bourlard, H. (1995). Neural Networks for Statistical Recognition of Continuous Speech, *Proceedings of the IEEE,* pp 742-770, May 1995.

Stoutamire, D. and Kennel, M. (1995). Sather Revisited: A High-Performance Free Alternative to C++. *Computers in Physics,* Vol. 9, No. 5, pp. 519-524.

Lazzaro, J. P., Wawrzynek, J., and Kramer, A (1994). Systems technologies for silicon auditory models. *IEEE Micro,* 14:3. 7-15.

Asanović, K., Beck, J., Feldman, J., Morgan, N., and Wawrzynek, J. (1993). Designing a connectionist network supercomputer. *International Journal of Neural Systems,* Vol. 4, No. 4, pp. 317-326.

Omohundro, S. (1993). The Sather Programming Language. *Dr. Dobb's Journal,* Vol. 18, No. 11, pp. 42-48.

Wawrzynek, J., Asanović, K., and Morgan, N. (1993). The Design of a Neuro- Microprocessor. *IEEE Transactions on Neural Networks,* Vol. 4, No. 3, pp. 394-399.

Asanović, K., Morgan, N., and Wawrzynek, J. (1993). Using Simulations of Reduced Precision Arithmetic to Design a Neuro-Microprocessor. *Journal of VLSI signal processing,* Vol 6, pp. 33-44.

Lazzaro, J. P., Wawrzynek, J., Mahowald., M., Sivilotti, M., Gillespie, D. (1993). Silicon auditory processors as computer peripherals. *IEEE Journal of Neural Networks* 4:3 523–

528.

5.2 Conference Proceedings

Weissman, B., Gomes, B., Quittek, J. W., and Holtkamp, M. (in review). Efficient Fine-Grain Thread Migration with Active Threads, submitted to the 12th International Parallel Processing Symposium and 9th Symposium on Parallel and Distributed Processing (IPPS/SPDP 1998).

Faerber, P. and Asanovic, K. (in press). Parallel Neural Network Training on Multi-Spert. *IEEE 3rd International Conference on Algorithms And Architectures for Parallel Processing.*

Asanović, K. (1997). A Fast Kohonen Net Implementation for Spert-II. In Mira, J., Moreno-Díaz, R., and Cabestany, J. (eds). *Biological and Artificial Computation: From Neuroscience to Technology,* Lecture Notes in Computer Science, Springer, pp. 792–800.

Greenberg, S. and Kingsbury, B. E. D., (1997). "The modulation spectrogram: In pursuit of an invariant representation of speech," ICASSP-97, 1997 IEEE International Conference on Acoustics, Speech and Signal Processing.

Kingsbury, B. E. D, and Morgan, N. (1997). "Recognizing reverberant speech with RASTA-PLP," ICASSP-97, 1997 IEEE International Conference on Acoustics, Speech and Signal Processing.

Wu, S., Shire, M., Greenberg, S., and Morgan, N. (1997). Integrating Syllable Boundary Information Into Speech Recognition. *Intl. Conf. Acoustics, Speech and Signal Processing,* pp. 987-990, Munich.

Konig, Y., Bourlard, H., and Morgan, N. (1996). REMAP - Experiments with Speech Recognition. *ICASSP 1996,* pp.3350-3353.

Lazzaro, J. P., Wawrzynek J., and Lippmann, R. (1996). A micropower analog VLSI HMM state decoder for wordspotting. In Jordan, M., Mozer, M., and Petsche, T. (eds), *Advances in Neural Information Processing Systems 9.* Cambridge, MA: MIT Press.

Mirghafori, N., Fosler, E., and Morgan, N. (1996). Towards Robustness to Fast Speech in ASR. *ICASSP 1996,* pp.335-338.

K. Asanović, J. Beck, B. Irissou, B. Kingsbury and J.Wawrzynek (1996). "T0: A Single-Chip Vector Microprocessor with Reconfigurable Pipelines." Proceedings of the 22nd European Solid-State Circuits Conference, 1996, pp. 344–347.

Wawrzynek J., Asanović, K., Kingsbury, B., Beck, J., Johnson, D., and Morgan, N. (1996). SPERT-II: A Vector Microprocessor System and its Application to Large Problems in Back-propagation Training. *Proceedings Fifth International Conference on Microelectronics for Neural Networks and Fuzzy Systems,* IEEE Computer Society Press, pp. 227–231.

Asanović, K., Beck, J., Irissou, B., Kingsbury, B.E.D., Morgan, N. and Wawrzynek, J. (1995). The T0 Vector Microprocessor. *Proceedings of Hot Chips VII,* August 1995.

Bourlard, H., Konig, Y., and Morgan, N. (1995). REMAP: Recursive Estimation and Maximization of a Posteriori Probabilities in Connectionist Speech Recognition. *Proceedings of Eurospeech 1995*, pp 1663-1666, Madrid, Spain.

Fleiner, C., Feldman, J., and Stoutamire, D. (1995). Recent advances in parallel Sather. *Proceedings POOMA conference, December 1995.*

Konig, Y., Bourlard, H., and Morgan, N. (1995). REMAP: Recursive Estimation and Maximization of A Posteriori Probabilities - Application to Transition-Based Connectionist Speech Recognition. In Mozer, M., Touretsky, D., and Hasselmo, M. (eds), *Advances in Neural Information Processing Systems 8.* Cambridge, MA: MIT Press, pp.388-394.

Lazzaro, J. P., Wawrzynek J. (1995). Silicon models for auditory scene analysis. In Mozer, M., Touretsky, D., and Hasselmo, M. (eds), *Advances in Neural Information Processing Systems 8.* Cambridge, MA: MIT Press.

Lazzaro, J. P. and Wawrzynek, J. (1995). A multi-sender asynchronous extension to the address-event protocol. In Dally, W. J., Poulton, J. W., Ishii, A. T. (eds), *16th Conference on Advanced Research in VLSI,* pp. 158–169.

Mirghafori, N., Fosler, E., and Morgan, N. (1995). Fast Speakers in Large Vocabulary Continuous Speech Recognition: Analyses and Antidotes. *Proceedings of Eurospeech 1995*, pp 491-494, Madrid, Spain.

Wawrzynek J., Asanović, K., Kingsbury, B., Beck, J., Johnson, D., and Morgan, N. (1995). SPERT-II: A Vector Microprocessor System and its Application to Large Problems in Back-propagation Training. In Mozer, M., Touretsky, D., and Hasselmo, M. (eds), *Advances in Neural Information Processing Systems 8.* Cambridge, MA: MIT Press.

Asanović', K., Beck, J., Feldman, J., Morgan, N., and Wawrzynek, J. (1994). A Supercomputer for Neural Computation. *Proc. International Conference on Neural Networks,* Vol. I. pp. 5-9.

Mirghafori, N., Morgan, N., and Bourlard, H. (1994). Parallel Training of MLP Probability Estimators for Speech Recognition: A Gender-Based Approach. *IEEE Workshop on Neural Networks for Signal Processing,* Greece, pp.289-298.

Morgan, N., Tajchman, G., Mirghafori, N., Konig, Y., and Wooters, C., (1994). Scaling a Hybrid HMM/MLP System for Large Vocabulary CSR, it ARPA Spoken Language Technology Workshop, Morgan Kaufmann, pp 123-124.

Morgan, N. (1994). Using a Million Connections for Continuous Speech Recognition. *ICONIP'94-Seoul,* pp. 439-1444.

Morgan, N. (1994). Big Dumb Neural Nets: A Working Brute Force Approach to Speech Recognition. *Proceedings of the International Conference on Neural Networks,* Vol. VII,

pp. 4462-4465.

Asanović, K., Beck, J. Feldman, J. Morgan, N. and Wawrzynek, J. (1993). Development of a Connectionist Network Supercomputer. *Proceedings of the Third International Conference on Microelectronics for Neural Networks* pp. 253-262.

Feldman, J., Lim, C., and T. Rauber (1993). The Shared-Memory Language pSather on a Distributed Memory Multiprocessor. *Proceedings of the 1993 Workshop on Languages, Compilers, and Run-Time Environments for Distributed Memory Multiprocessors*, pp. 17-20, Boulder, CO, September 30 – October 2, 1992, ACM SIGPLAN Notices 28(1), January, 1993.

Lazzaro, J. P., Wawrzynek, J., Mahowald., M., Sivilotti, M., Gillespie, D. (1993). Silicon auditory processors as computer peripherals. In Hanson, S., Cowan, J., and Giles C., (eds), *Advances in Neural Information Processing Systems 5*. San Mateo, CA: Morgan Kaufmann Publishers, 820–827.

Lim, C., Feldman, J. and Murer, S. (1993). Unifying control- and data-parallelism in an object-oriented language. *Proceedings of the 1993 Joint Symposium on Parallel Processing*, pp. 261-268, Waseda University, Tokyo, May 17-19.

Asanović, K., Beck, J., Kingsbury, B.E.D, Kohn, P., Morgan, N., and Wawrzynek, J. (1992). SPERT: A Neuro-Microprocessor. *Proc. 3rd Int. Workshop on VLSI for Artifical Intelligence and Neural Networks.*

Asanović, K., Beck, J., Kingsbury, B.E.D, Kohn, P., Morgan, N., and Wawrzynek, J. (1992). SPERT: A VLIW/SIMD Microprocessor for Artificial Neural Network Computations. *Proceedings Application Specific Array Processors 1992*, Berkeley, CA, 1992.

Asanović, K., Beck, J., Kingsbury, B.E.D, Kohn, P., Morgan, N., and Wawrzynek, J. (1992). SPERT: A VLIW/SIMD Neuro-Microprocessor *Proc. IJCNN, Baltimore, USA, June 1992.*

Lazzaro, J. P. (1992). Low-power silicon spiking neurons and axons. *IEEE International Symposium on Circuits and Systems* San Diego, CA, pp. 2220-2224.

Asanović, K. and Morgan, N. (1991). Experimental Determination of Precision Requirements for Back-Propagation Training of Artificial Neural Networks. *Proceedings 2nd International Conference on Microelectronics for Neural Networks,* Munich, October 1991.

Lazzaro, J. P. (1991). Biologically-based auditory signal processing in analog VLSI. *IEEE Asilomar Conference on Signals, Systems, and Computers*, pp. 790-794.

5.3 Books and Book Chapters

Asanović, K., Beck, J., Johnson, D., Wawrzynek, J., Kingsbury, B.E.D., Morgan, N. (1997). Training Neural Networks with SPERT-II. In Sundararajan, N. and Saratchandran, P. (eds). *Parallel Architectures for Artificial Neural Networks — Paradigms and Implementations.*, IEEE Computer Scociety Press, pp. 609–641.

Lim, C. and Feldman, J. (1996). *Distributed Memory Implementation of a Shared-Address Parallel Object-Oriented Language,* Kluwer Academic Publishers, 1996, 27 pages.

J. Feldman (1996). Tutorial Manual for pSather. On-line.

Morgan, N. (1995). Programmable Neurocomputing Systems. In Arbib, M. (ed), *Neurocomputing Handbook,* MIT Press, pp. 764-768.

Asanović, K., Beck, J., Kingsbury, B.E.D, Kohn, P., Morgan, N., and Wawrzynek, J. (1994). SPERT: A Neuro-Microprocessor. In Delgado-Frias, José G., and Moore, William, R. (eds) *VLSI for Neural Networks and Artifical Intelligence,* Plenum Press, pp. 103–107.

Feldman, J. (1994). Universal High Performance Computing – We Have Just Begun. In Uzi Vishkin, (ed), *Developing a Computer Science Agenda for High-Performance Computing,* pp. 26-29, ACM Press.

Lazzaro, J. P., and Wawrzynek, J. (1994). Low-power silicon axons, neurons, and synapses. In Zaghloul, M. E., Meador, J. L., and Newcomb, R. W., (eds) *Silicon Implementations of Pulse Coded Neural Networks.* Norwell, MA: Kluwer Academic Publishers, pp. 153-164.

Szyperski, C., Omohundro, S. and Murer, S. (1993). Engineering a Programming Language: The Type and Class System of Sather. In Jurg Gutknecht, (ed), *Programming Languages and System Architectures,* pp. 208-227, Springer Verlag, Lecture Notes in Computer Science 782, November 1993. Available as technical report ICSI TR-93-064.

Schmidt, H. and Omohundro, S. (1993). CLOS, Eiffel, and Sather: A comparison. In Andreas Paepcke, (ed), *Object-Oriented Programming: The CLOS Perspective,* pp. 181-213, MIT Press Cambridge, Massachusetts, London, England, 1993. Available as technical report ICSI TR-91-047.

5.4 Technical Reports

Asanović, K, and Johnson, D. (1997). Torrent Architecture Manual, Revision 2.11. *Computer Science Division, University of California at Berkeley,* CSD-97-930.

Asanović, K, and Beck, J. (1997). T0 Engineering Data, Revision 0.14, Revision 2.11. *Computer Science Division, University of California at Berkeley,* CSD-97-931.

Philippsen, M. (1995). Imperative Concurrent Object-Oriented Languages: An Annotated Bibliography. *International Computer Science Institute Technical Report,* TR-95-049,

Philippsen, M. (1995). Enabling Compiler Transformations for pSather 1.1. *International Computer Science Institute Technical Report,* TR-95-040.

H. Haertig (1994). Near or far. *International Computer Science Institute Technical Report,* TR-94-04.

Omohundro, S. (1994). The Sather 1.0 Specification *International Computer Science Insti-*

tute Technical Report.

Philippsen, M. (1994). Sather 1.0 Tutorial. *International Computer Science Institute Technical Report,* TR-94-062.

Asanović, K, Beck, J. Callahan, T., Feldman, J. Irissou, B., Kingsbury, B.E.D, Kohn, P., Lazzaro, J., Morgan, N., Stoutamire, D., and John Wawrzynek (1993). CNS-1 Architecture Specification *International Computer Science Institute Technical Report,* TR-93-021.

Lim, C. (1993). A Parallel Object-Oriented System for Realizing Reusable and Efficient Data Abstractions.*International Computer Science Institute Technical Report,* TR-93-063.

Murer, S., Feldman, J., Lim, C. and Seidel, M. (1993). pSather: Layered Extensions to an Object-Oriented Language for Efficient Parallel Computation. *International Computer Science Institute Technical Report,* TR-93-028.

Murer, S., Omohundro, S., and Szyperski, C. (1993). Sather Iters: Object-Oriented Iteration Abstraction. *International Computer Science Institute Technical Report,* TR-93-045.

Asanović, K and N. Morgan (1991). Experimental Determination of Precision Requirements for Back-Propagation Training of Artificial Neural Networks. *International Computer Science Institute Technical Report,* TR-91-036.

Asanović, K, Beck, J., Kingsbury, B. E. D., Kohn, P., Morgan, N. and Wawrzynek, J. (1991). SPERT: A VLIW/SIMD Microprocessor for Artificial Neural Network Computations. *International Computer Science Institute Technical Report,* TR-91-072.

Kingsbury, B.E.D, Asanović, K., Wawrzynek, J, Irissou, B., and Morgan, N. (1991). Recent Work in VLSI Elements for Digital Implementations of Artificial Neural Networks. *International Computer Science Institute Technical Report,* TR-91-074.