Applications of Random-Pulse Machine Concept to Neural Network Design

Emil M. Petriu, *Senior Member, IEEE,* Kenzo Watanabe, *Fellow, IEEE,* and Tet H. Yeap

***Abstract*—Neural networks can reach their true potential only when they are implemented in hardware as massively parallel processors. This paper presents the random-pulse machine concept and shows how it can be used for the modular design of neural networks. Random-pulse machines deal with analog variables represented by the mean rate of random-pulse streams and use simple digital technology to perform arithmetic and logic operations. This concept presents a good tradeoff between the electronic circuit complexity and the computational accuracy. The resulting neural network architecture has a high packing density and is well suited for very large-scale integration (VLSI). Simulation results illustrate the performance of the basic elements of a random-pulse neuron.**

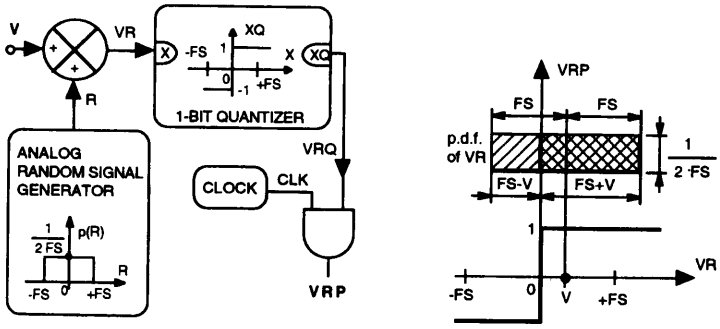

Fig. 1. Analog/random-pulse converter.

I. Introduction

LOOKING for a model to prove that algebraic operations with analog variables can be performed by logical gates, von Neuman advanced in 1956 the idea of representing analog variables by the mean rate of random-pulse streams [1]. Pursuing this idea, a number of similar stochastic data-processing concepts were reported in the 1960's: "noise computer" by Poppelbaum and Afuso [2], "random-pulse machine" by Ribeiro [3], and "stochastic computing" by Gaines [4].

Random-pulse machines deal with analog variables while using digital technology to perform arithmetic and logic operations on binary pulses which are the information carriers. As variables are represented by the statistical average of random pulse streams, the resulting data-processing system has a better tolerance to noise than the classical deterministic systems. The digital technology used to implement these systems offers a number of advantages over the analog technology: modular and flexible design, higher internal noise immunity, and simpler I/O interfaces.

On parallel tracks, in the late 1950's and 60's dither techniques were studied to reduce the effects of the quantization noise [5]–[9]. The effects of dither quantization have been further discussed during the 1970's and 80's from a signal-processing perspective [10]–[15]. More recently, dither quantization has also caught the attention of the instrumentation and measurement community [16]–[19].

Since first mentioned in 1987, pulse-stream VLSI neural networks are regularly reported in literature. However, like most new technologies, these networks are bottom-up implementations using ad hoc data-processing solutions. With few exceptions [22], these reports ignore earlier random-pulse technology developments [20], [21].

This paper presents a number of basic random-pulse data-processing techniques and shows how these can be used for a more structured design of neural network architectures.

II. Random-Pulse Data Representation

Random-pulse data appear as sequences of random binary pulses which carry analog information represented by the statistical mean value of the pulse sequence. Such a representation can be viewed as the probability modulation of a random-pulse carrier by a deterministic analog variable.

The "analog/random-pulse" converter shown in Fig. 1 illustrates the principle of this modulation. A deterministic analog input V, supposed to have a relatively low variation rate, is superimposed on an analog random signal R which is uniformly distributed between $+FS$ and $-FS$. The resulting analog random signal VR is uniformly distributed around a deterministic bias V as shown in the VR versus VRP quantization diagram. This signal is then 1-bit quantified to produce a random sequence of pulses VRQ which will have the binary value +1 if $VR \geq 0$ or -1 if $VR < 0$. These pulses are finally sampled by a clock signal CLK to produce the clocked random-pulse sequence VRP.

A probabilistic estimation of the deterministic component of the random-pulse sequence can be calculated from the VR

Manuscript received April 26, 1995; revised November 6, 1995. This work was supported by a grant from the Ministry of Education of Japan, and by grants from the Natural Sciences and Engineering Research Council of Canada and the Telecommunications Institute of Ontario.

E. M. Petriu and T. Yeap are with the Department of Electrical Engineering, University of Ottawa, Ontario, Canada K1N 6N5.

K. Watanabe is with Shizuoka University, Hamamatsu 432, Japan.

Publisher Item Identifier S 0018-9456(96)02488-6.

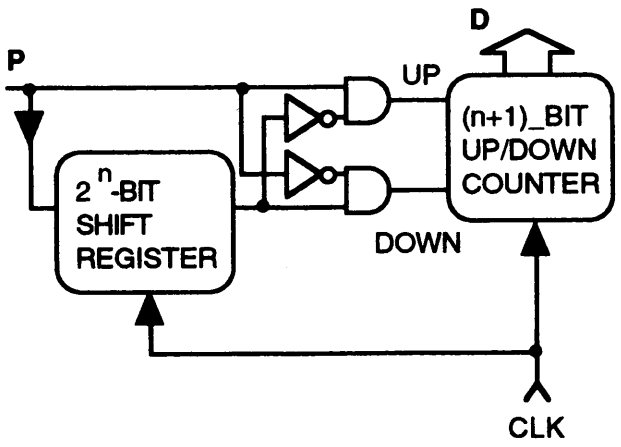

Fig. 2. Moving average random-pulse/digital converter.

versus VRP quantization diagram given in Fig. 1

$$\begin{aligned} E[VRP] &= (+1)\cdot p[VR \geq 0] + (-1)\cdot p[VR < 0] \\ &= p(VRP) - p(VRP') \\ &= \frac{(FS+V)}{(2\cdot FS)} - \frac{(FS-V)}{(2\cdot FS)} \\ &= \frac{V}{FS} \end{aligned} \tag{1}$$

which shows that the statistical mean value of the VRP sequence represents a measure of the deterministic analog input V.

From (1) the deterministic analog value V associated with the VRP sequence is

$$V = [p(VRP) - p(VRP')]\cdot FS \tag{2}$$

where the apostrophe sign ($'$) denotes a logical inversion of the respective binary signal.

A "digital/random-pulse" conversion [3], [4], can be obtained by comparing the digital input with a digital noise. Such a converter can be used as an input interface between a digital computer and a stochastic computer. It is also used to restore randomness after some algebraic operations, such as integration for instance, are performed on random-pulse data.

Different "random-pulse/digital" converters were developed to recover the deterministic component V by averaging a finite number of VRP pulses. The "moving-average" converter [23] shown in Fig. 2 gives a continuously updated digital average of the last $N = 2^n$ pulses of the random-pulse data stream P_i. It employs the stable averaging algorithm

$$\begin{aligned} D_i &= \frac{P_i + P_{i-1} + P_{i-2} + \cdots + P_{i-N} + 1}{N} \\ &= D_{i-1} + \frac{P_i - P_{i-N}}{N}. \end{aligned} \tag{3}$$

Pulses are digitally integrated by an $(n+1)$-bit up-down counter which is incremented if $P_i = +1$ and $P_{i-N} = -1$, and conversely decremented if $P_i = -1$ and $P_{i-N} = +1$, with nothing happening if P_i and P_{i-N} are equal. The N-bit shift register is used to store FIFO style all the pulses occurring in the previous N clock intervals. When more VRP samples are considered, the estimation accuracy is increased but the bandwidth of V is restricted [24].

The described analog/random-pulse and random-pulse/digital conversions are illustrated in Fig. 3. An analog signal x changing from $-0.3\cdot FS$ to $0.45\cdot FS$ is converted to a sequence of random-pulses X which is then reconverted as a moving average over $N = 16$ random-pulses to produce the analog estimation $xest$.

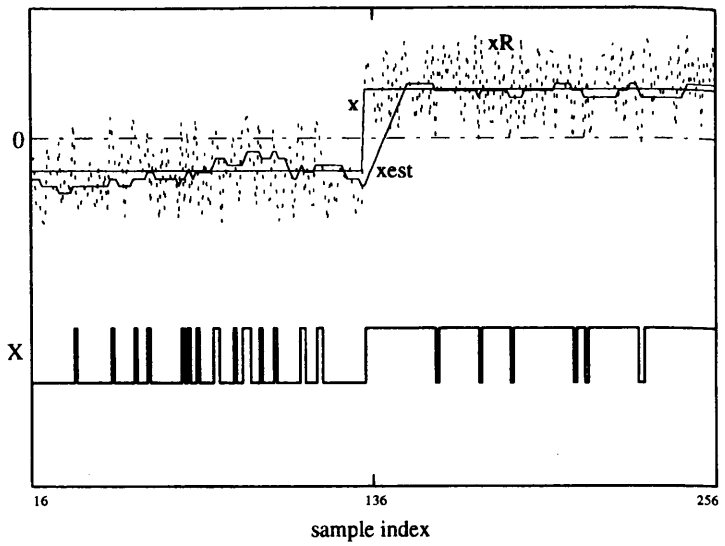

Fig. 3. Simulation results illustrating the analog/random-pulse and random-pulse/digital conversions.

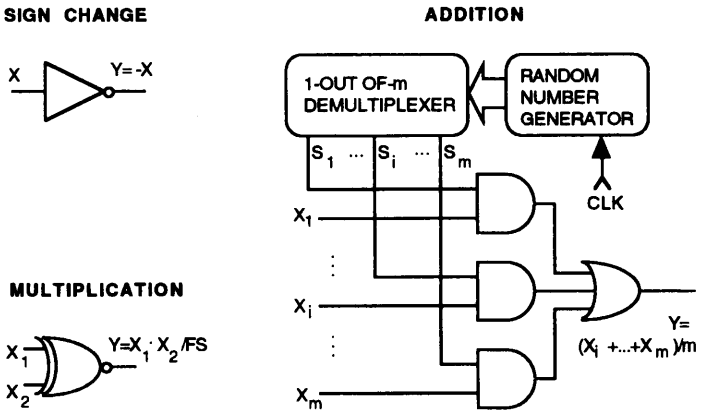

Fig. 4. Logic circuits for random-pulse arithmetic.

III. Random-Pulse Arithmetic

Simple logical operations with individual pulses allow carrying out arithmetic operations with the analog variables represented by their respective random-pulse sequences. Fig. 4 shows logic circuits for the random-pulse implementation of "sign change," "multiplication," and "addition."

The arithmetic "sign change" is carried out by an INVERTER circuit. If x is the random-pulse input, then the output random-pulse sequence is $y = x'$. The analog meaning of the output sequence y is

$$Y = -X. \tag{4}$$

The arithmetic "multiplication" is carried out by a COINCIDENCE circuit. If x_1 and x_2 are the two random-pulse inputs, their logic combination will produce an output random-pulse sequence $y = x_1\cdot x_2 + x_1'\cdot x_2'$. The analog meaning of the output sequence y is

$$Y = X_1\cdot\frac{X_2}{FS}. \tag{5}$$

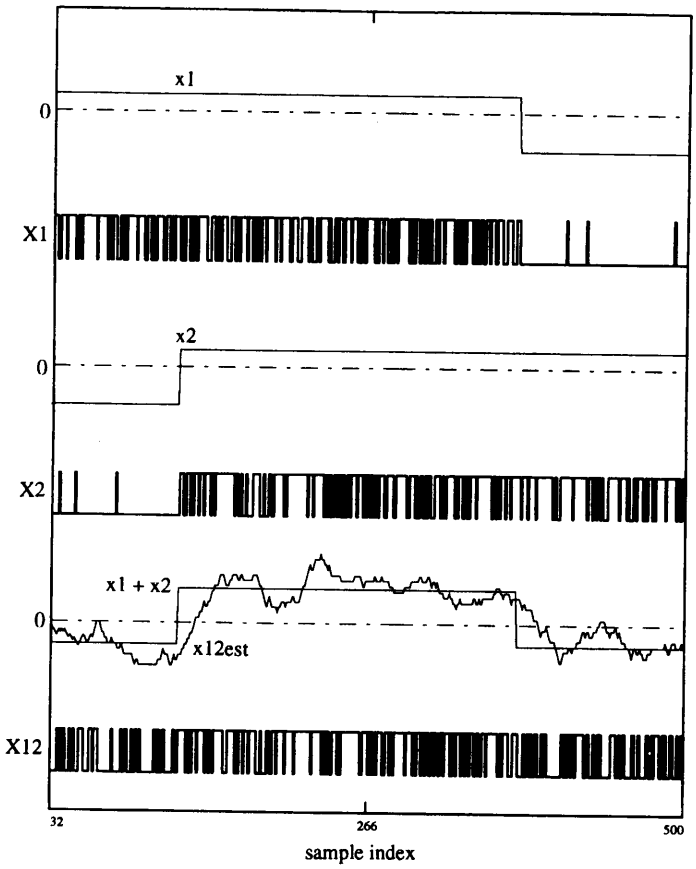

Fig. 5. Simulation results illustrating the random-pulse addition of two signals.

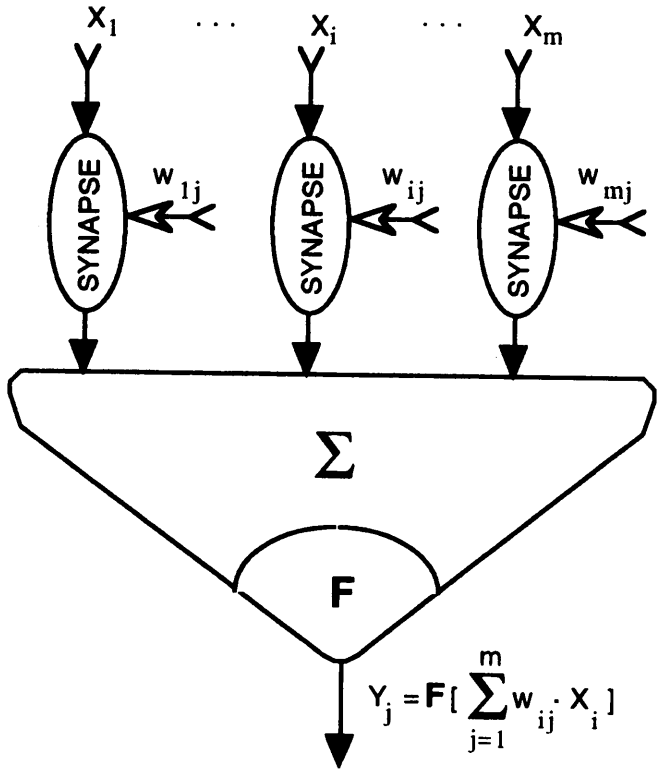

Fig. 6. Typical neuron structure.

The arithmetic "addition" of m random-pulse data $x_1, x_2, \cdots, x_m$ can be carried out as shown in Fig. 4 by a time multiplexing controlled by random signals S_i (for $i = 1, 2, \cdots, m$). This random scanning acts as a "stochastic isolator" [4] which removes unwanted correlations between sequences with similar patterns. The random scanning signals S_i are uniformly distributed having the same probability $p(S_i) = 1/m$. Because of this scanning the multiplexed samples are statistically independent. The analog meaning of the output sequence y is

$$Y = \frac{X_1 + \cdots + X_m}{m}. \tag{6}$$

The random-pulse addition is illustrated in Fig. 5. Two step signals, $x1$ which takes values from $0.2 \cdot FS$ to $-0.45 \cdot FS$, and $x2$ which takes values from $-0.45 \cdot FS$ to $0.2 \cdot FS$, have their random-pulse representations $X1$ and $X2$. The arithmetic addition of these two streams of random-pulses produces the random-pulse train $X12$ which is reconverted by a moving average over $N = 16$ running samples and a rescaling factor $m = 2$ to yield the analog estimation $x12est$.

IV. Random-Pulse Neural Network Architecture

A typical neuron consists of more synapses and a neuron-body, as shown in Fig. 6. Each synapse multiplies an incoming neural signal X_i, where $i = 1, 2, \cdots, m$, by a synaptic-stored variable weight value w_{ij}. The connection weights are adjusted during the learning phase. Connection weights which are positive-valued are "excitatory" connections, and those with negative values are "inhibitory" connections. The neuron-body integrates the signals from all the post-synaptic channels (also called dendrites). The result of this integration is then submitted to an "activation" function "F" to produce the neuron's output signal Y_j.

Random-pulse representation is used throughout the neural network, for synaptic weight storage as well as for arithmetic operations. In Fig. 7 is shown the implementation of a synapse around the 1-bit random-pulse multiplication module shown in Fig. 4. The synaptic weights are dynamically stored in a 2^n-bit shift register. Loading weight values from the DATIN input into each register selected by the synapse address SYNADD is done serially when a low logic signal is applied to the control input MODE.

The synaptic weight multiplication is illustrated in Fig. 8. An analog signal $x1 = -0.4 \cdot FS$ is converted to produce the random-pulse stream $X1$ to be multiplied by a synaptic weight $w1$ which changes from an excitatory value 0.75 to an inhibitory -0.5. The 16-bit random-pulse stream $W1$ representing this weight is then cyclically multiplied with $X1$ to produce the random-pulse stream $DT1$. The analog estimation $DT1est$ is the moving average over $N = 16$ samples of this dendrite random-pulse stream.

Fig. 9 shows the random-pulse implementation of the neuron body. The m-input addition module collects the post-synaptic data streams which are then integrated by a moving-average random-pulse/digital converter. The internal neuron body clock CLK* has a pulse rate at least m times higher than the general clock CLK. There are only five activation function "F" types which are usually used: linear, step, ramp, sigmoid, and Gaussian [25].

During the learning phase, some algorithms like back propagation require a smooth nonlinear activation function which

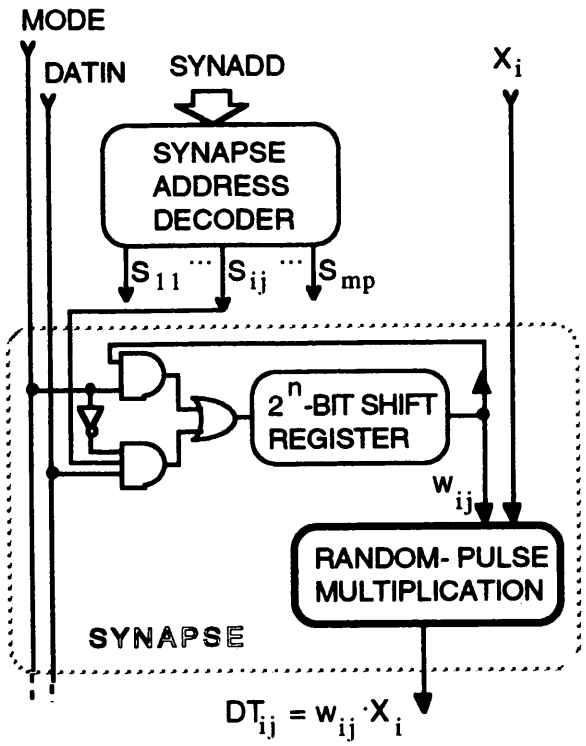

Fig. 7. Random-pulse implementation of a synapse.

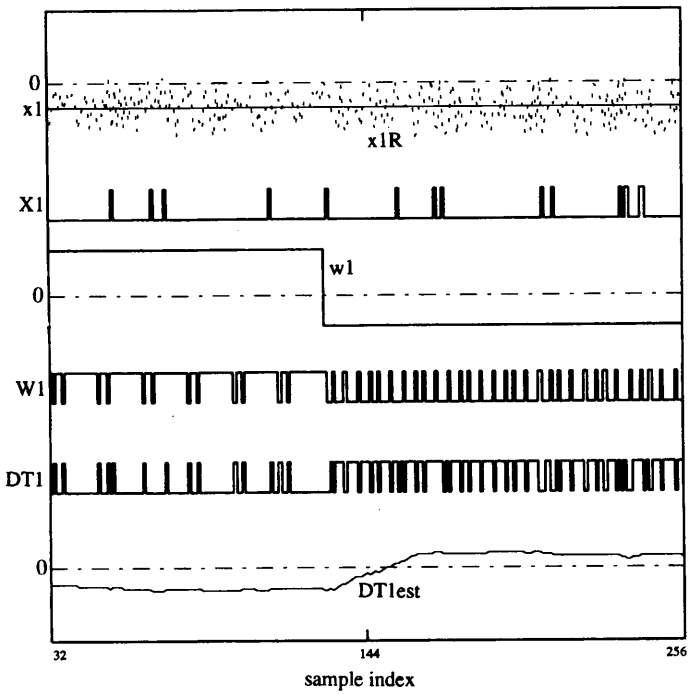

Fig. 8. Simulation results illustrating the random-pulse synaptic weight multiplication.

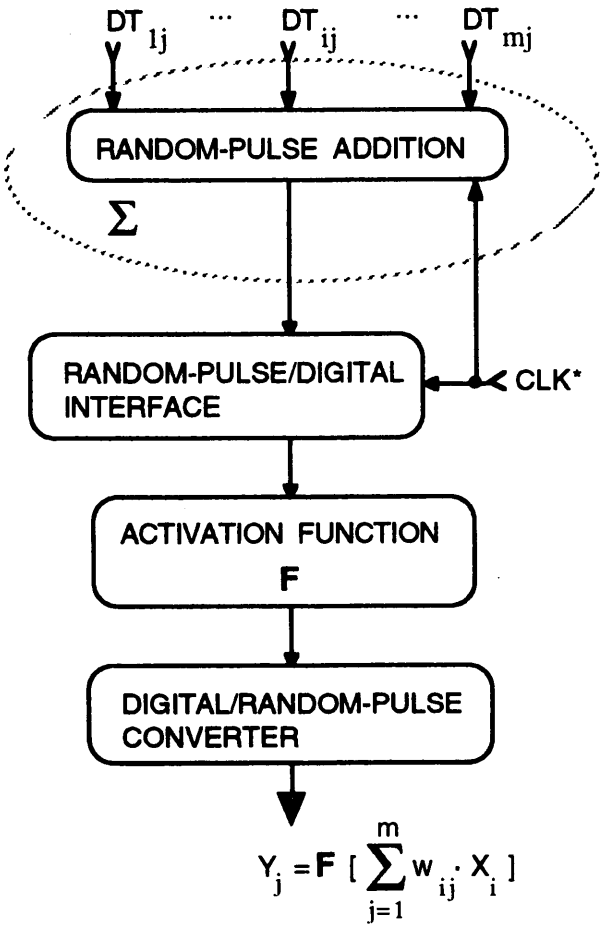

Fig. 9. Random-pulse implementation of the neuron body.

TABLE I
ACCURACY OF THE RANDOM-PULSE MULTIPLICATION (SIMULATION RESULTS)

Bit Stream Length	Mean Absolute Error	Absolute Error Standard Deviation	Mean Square Error	Square Error Standard Deviation
8	0.4266	0.2268	0.2329	0.2123
16	0.3559	0.1984	0.1656	0.1469
32	0.3341	0.1878	0.1466	0.1348
64	0.3078	0.1981	0.1336	0.1388

is differentiable everywhere [21]. In general, any desired activation function can be implemented as a look-up table. Since the neuron output will be used as a synaptic input to other neurons, a final digital/random-pulse converter stage is used to restore the randomness of the signal Y_j. An interesting implementation was recently reported in [22] for linear and sigmoid activation functions. The counter of the random-pulse/digital converter is preloaded at each conversion cycle with a given threshold value. At any time when the counter content reaches this threshold, an overflow into the most significant bit is generated and used further as the neuron's random-pulse output.

The neural network architecture described can be interfaced directly with the environment or other computers via either analog/random-pulse or digital/random-pulse input interfaces and random-pulse/digital output interfaces.

The proposed architecture has been simulated at the logic level. The purpose of the simulation is to determine the arithmetic accuracy that could be achieved by the proposed architecture and to find a suitable size of the shift registers which store connection weights. Two measures are used to evaluate the architecture performance: means and standard deviations of the absolute and square errors.

First, the accuracy on the multiplication of two analog values between -1 and 1 is measured. One hundred pairs of 16-bit random-pulse data are multiplied using the COINCIDENCE logic. The simulation results given in Table I show that both the mean square and absolute errors stabilize to approximately 0.3 and 0.1, respectively, for bit stream length greater than 16.

TABLE II
ACCURACY OF THE RANDOM-PULSE PRODUCT ADDITION (SIMULATION RESULTS)

Number of Products Per Sum	Mean Absolute Error	Absolute Error Standard Deviation	Mean Square Error	Square Error Standard Deviation
2	0.2244	0.1805	0.0414	0.0
4	0.3033	0.2180	0.0040	0.0
8	0.5154	0.3973	0.0387	0.0001

Next the accuracy of the addition of products is measured for 100 sums of 2, 4 and 8 products of 16-bit pseudo-random data representing analog values between -1 and 1. The resulting statistics for these sums of products are presented in Table II. It can be seen that when the number of products which are summed (Fig. 4) increases from 2–8, relatively small changes are observed in the mean square error.

V. CONCLUSIONS

The random-pulse machine concept has been used as a common analytical framework when dealing with quantization problems occurring in different areas such as instrumentation, signal processing, and control. This paper shows how this concept can be extended to the design of random-pulse neural networks. A neural network architecture based on 1-bit random-pulse data processing has been presented. The design represents a tradeoff between neuron circuit complexity and computational accuracy, aimed at obtaining a high packing density in an integrated circuit that is well suited for implementation using VLSI technology. The architecture is modular and has neurons constructed using simple digital circuits. Simulation results show that reasonably good computational accuracy can be achieved using relatively short, 16-bit long, random-pulse streams. More logic-level and electronic device-level simulations are needed before attempting the VLSI implementation of such a random-pulse neural network.

REFERENCES

[1] J. von Neuman, "Probabilistic logics and the synthesis of reliable organizms from unreliable components," in *Automata Studies,* C. E. Shannon, Ed. Princeton, NJ: Princeton University Press, 1956.

[2] W. J. Poppelbaum and C. Afuso, "Noise-computer," University of Illinois, Urbana, Dept. Comp. Sci., Quart. Tech. Progress Reports, Jan./Mar. 1965, Apr./June 1965, July/Sept. 1965, Jan./Mar. 1966.

[3] S. T. Ribeiro, "Random-pulse machines," *IEEE Trans. Electron. Comp.,* vol. EC-16, no. 3, pp. 261–276, June 1967.

[4] B. R. Gaines, "Stochastic computer thrives on noise," *Electronics,* pp. 72–79, July 1967.

[5] T. Ishikawa, "Linearization of contactor control systems by external dither signals," Tech. Rep. 2103-2, Electronics Laboratories, Stanford University, 1960.

[6] T. Sato, "Autocorrelation using random voltage method," *Automat. Contr.,* vol. 7, pp. 8–12, 1960.

[7] B. P. T. Veltman and H. Kwakernaak, "Theorie und technik der polaritatkorrelation fur die dynamische analyse niederfrequenter signale und systeme," *Regelungstechnik,* vol. 9, pp. 357–364, Sept. 1961.

[8] P. Jespers, P. T. Chu, and A. Fetweis, "A new method for computing correlation functions," in *Intl. Symp. Inf. Theory,* Brussels, Belgium, 1962.

[9] L. Schuchman, "Dither signals and their effect on quantization noise," *IEEE Trans. Commun. Technol.,* vol. CT-12, pp. 162–165, 1964.

[10] K.-Y. Chang and D. Moore, "Modified digital correlator and its estimation errors," *IEEE Trans. Inf. Theory,* vol. IT-16, no. 6, pp. 699–706, 1970.

[11] J. J. Hirsch, "Contributions a l'etude des conversions numerique-analogique et analogique-numerique au moyen d'une representation stochastique de l'information," These de Dr. Ing., Grenoble, France, 1970.

[12] F. Castanie, "Estimation de moments par quantification a reference stochastique," These Doctorat-es-Sciences, Toulouse, France, 1977.

[13] ——, "Signal processing by random reference quantizing," *Signal Processing,* vol. 1, no. 1, pp. 27–43, 1979.

[14] E. Pop and E. Petriu, "Influence of reference domain unstability upon the precision of random reference quantizer with uniformly distributed auxiliary source," *Signal Processing,* vol. 5, pp. 87–96, 1983.

[15] F. Castanie, "Linear mean transfer random quantization," *Signal Processing,* vol. 7, no. 2, pp. 99–117, 1984.

[16] M. F. Wagdy, "Effect of various dither forms on quantization errors of ideal A/D converters," *IEEE Trans. Instrum. Meas.,* vol. 38, no. 4, pp. 850–855, 1989.

[17] M. F. Wagdy and W. Ng, "Validity of uniform quantization error model for sinusoidal signals without and with dither," *IEEE Trans. Instrum. Meas.,* vol. 38, no. 3, pp. 718–722, 1989.

[18] M. F. Wagdy and M. Goff, "Linearizing average transfer characteristics of ideal ADC's via analog and digital dither," *IEEE Trans. Instrum. Meas.,* vol. 43, no. 2, pp. 146–150, 1994.

[19] P. Carbone and D. Petri, "Effect of additive dither on the resolution of ideal quantizers," *IEEE Trans. Instrum. Meas.,* vol. 43, no. 3, pp. 389–396, 1994.

[20] A. Hamilton, A. F. Murray, D. J. Baxter, S. Churcher, H. M. Reekie, and L. Tarasenko, "Integrated pulse stream neural networks: Results, issues, and pointers," *IEEE Trans. Neural Networks,* vol. 3, no. 3, pp. 385–393, May 1992.

[21] J. E. Tomberg and K. K. K. Kaski, "Pulse-density modulation technique in VLSI implementations of neural network algorithms," *IEEE J. Solid-State Circuits,* vol. 25, no. 5, pp. 1277–1286, 1990.

[22] M. van Daalen, T. Kosel, P. Jeavons, and J. Shawe-Taylor, "Emergent activation functions from a stochastic bit-stream neuron," *Electron Lett.,* vol. 30, no. 4, pp. 331–333, Feb. 1994.

[23] A. J. Miller, A. W. Brown, and P. Mars, "Moving-average output interface for digital stochastic computers," *Electron Lett.,* vol. 10, no. 20, pp. 419–420, Oct. 1974.

[24] A. J. Miller and P. Mars, "Optimal estimation of digital stochastic sequences," *Int. J. Syst. Sci.,* vol. 8, no. 6, pp. 683–696, 1977.

[25] P. K. Simpson, "Foundations of neural networks," in *Artificial Neural Networks* E. Sanchez-Sinencio and C. Lau, Eds. New York: IEEE Press, 1992, pp. 3–24.

Emil M. Petriu (M'86–SM'88) for a photograph and biography, see this issue, p. 412.

Kenzo Watanabe, (M'74–SM'86–F'93) received the B.E. and M.E. degrees in engineering from Shizuoka University, Hamamatsu, Japan, in 1962 and 1966, respectively, and the Dr.Eng. degree from Kyoto University in 1976.

He is a Professor at the Research Institute of Electronics, Shizuoka University. Except for a year as a Visiting Professor at UCLA, he has been on the faculty at Shizuoka University since 1962, serving progressively as Research Assistant, Associate Professor and Professor.

Dr. Watanabe is an AdCom member of the IEEE Instrumentation and Measurement Society, and is an Associate Editor of this TRANSACTIONS.

Tet H. Yeap received the B.A.Sc. degree in electrical engineering from the Queen's University, Kingston, Ontario, Canada, in 1982, and the M.A.Sc. and Ph.D. degrees in electrical engineering from the University of Toronto, in 1984 and 1991, respectively.

He is currently an assistant professor in the Department of Electrical Engineering, University of Ottawa, Ottawa, Ontario, Canada. His research interests include neural networks, VLSI architectures, optical comminucations, multi-media, dynamics and control and parallel computer architectures.