URAN VLSI chip을 이용한 숫자음 인식

김기철°, 한일송*, 이준희, 이황수, 이영란
한국과학기술원 정보 및 통신공학과,
*한국통신 연구센터

Spoken Digit Recognition Using URAN(Universally Reconstructable Artificial Neural-network)VLSI Chip

Ki-Chul Kim, Il-Song Han*, Jun-Hee Lee, Hwang-Soo Lee, and Yung N. Yi
Dep of Information and Communication Eng. KAIST
*Korea Telecom Research Center

Abstract

In this paper, we explore the possibility of URAN(Universally Reconstructable Artificial Neural-network) VLSI chip for speech recognition. URAN, a newly developed analog-digital hybrid neural chip, is discussed in respects to its input, output, and weight accuracy and their relations to its performance on speaker independent digit recognition. Multi-layer perceptron(MLP) nets including a large frame input layer are used to recognize a digit syllable at a forward retrieval. The simulation results using the full and limited floating precision computations for the input, output, and weight variables of the network give the comparable classification performance. An MLP with piecewise linear hidden and output units is also trained successfully using low accuracy computation.

1 Introduction

As the neural networks have the ability to learn from experience and of generalization to new data, they provide a promising approach to the real world problems such as speech recognition tasks that have variabilities between examples of the same class. Neural networks also offers potential advantage over existing approaches by providing highly parallel architectures. The large computational requirements of neural networks and their massively parallel architecture have led to a number of hardware implementations. Though hardware implementation of the neural networks has many advantages, it involves practical difficulties in speed, scale, and accuracy implementation. In addition, the entire system must be engineered, considering such issues as test and debug, memory bandwidth and latency, communication between processors, I/O, and software for real systems[1]. We are now developing a real time speech recognition system using neural network VLSI chips.

In this paper, URAN, the new analog-digital hybrid VLSI neural network, is described with digital interface to conventional computers. The implementation of the Back Propagation(BP) algorithm employing limited accuracy available to URAN chip is considered and applied to speaker independent digit recognition. Though various kinds of neural network models have been proposed and showed distinct performance in speech recognition areas, we examine simple MLPs to find their adaptability to reduced precision of the network parameters and the linear output function which are available to URAN.

The simulation results show that the performance of speaker independent digit recognition is not degraded for the limited floating point precision equivalent to 8 bit accuracy for input, output, and weight as in the previous works[3]. The MLP used in our simulations has a large number of input units to allow even the largest input pattern to be applied to input neurons at once as a whole. The temporal acoustic contexts are modeled inherently without time normalization by padding zeros to the input units without input data, which yields the investigation of the fundamental issues related to BP implementation on URAN chip.

In addition, we perform a simulation of on-chip learning by employing a piecewise linear function for hidden and output units instead of sigmoidal activation function. The BP simulations using linear hidden and output units gave similar performance to those using sigmoidal units, once training is completed with sigmoidal units. In addition, an MLP with piecewise linear hidden and output units is trained successfully and shows comparable performance to the ones trained with sigmoidal units.

Section 2 describes the URAN chip architecture. The critical issues related with BP implementations are presented in section 3 with the network architecture and learning algorithm. Section 4 includes the speaker independent digit recognition experiments and performance figures according to the input, output, and weight accuracy. Preliminary investigation of on-chip learning using a piecewise linear function is described in section 5.

2 URAN–Universally Reconstructable Artificial Neural-network

2.1 Chip Architecture

In general, most of digital, analog, or analog-digital mixed neuro-chips are constrained in accuracy, speed, size or flexibility. There has been made new advancement in those aspects with the suggested analog-digital hybrid neural network circuit. The accuracy is improved by the linear voltage-controlled MOSFET linear resistance circuit for the synapse emulation. The speed is increased by the digital neural state. The general flexibility is realized by the inherent electrical characteristic of each synapse cell and modular architecture of chip.

Chip features are summarized in Table 1. As in Table 1, the chip performs under the flexible control, that is, the various mode of synaptic connection per neuron, the extendible weight accuracy and the unlimited asynchronous/direct interchip expansion in size and speed. In fact, 16 fully connected module is selected from external and independently - either one by one selection or all at one selection is possible as shown in Fig. 1. Additionally, the speed or modularity of each module is improved by introducing the individual external weight input. The neural hardware of huge size and high speed is straightforwardly implementable with chips in the same way as the chip is with module.

Considering the operation of the circuit itself, all circuits over the chip except digital decoder unit are operated in analog. And as they are almost virtually static except switching transistor controlled by neural input, the computation speed is high and even can be improved substantially with the advance of memory production technology. As a basic cell, 9 transistors are used per cell including weight memory. The cell size including interconnection area for URAN-I is reduced to less than 40 μm in diameter.

With the linear voltage-controlled bipolar current source of each synapse cell, the synaptic function of multiplication is done with the switching transistor, i.e. half-in-analog and half-in-digital. The use of bipolar pulse improved stray effect from switching. Pulse of neural input for switching are not limited in style, time and numbers, that is, they are fully independent from each other.

The linearity is based on the compensated channel resistance of balanced configuration in the triode region, and proved to have more than 8 bit if necessary. The accuracy extendibility and flexible modularity are inherent in electrical wired-OR characteristics from each independent bipolar current source. NO clocking or any synchronous operation is needed in this case, while it is indispensable in most of conventional digital neural hardware of analog-digital neural chip. Any size of network can be integrated of implemented by merely placing the cell in 2 dimensional array without considering the timing requirement of digital or the load effect of analog. In the case of URAN-I, the delay of control signal is considered in designing the logic interface due to the long path on the chip. Off-chip digital interface is depicted in Fig. 2.

Speed	200×10^6 connections/s
Synaptic connections	135,424
Weight accuracy	8 bit
Organization (synapses/neuron)	$92 \times n$, n=1 to 16 (electrically programmable)
Function of interchip expansion	Fully asynchronous and direct electrical wired-OR at output
Supply voltage	3 V, -3 V
Chip size	$13 \times 13\ mm^2$
Technology	1.0μ digital CMOS
PGA package	257 pin

Table 1: URAN-I features

2.2 The Critical Issues of BP Implementations on URAN Chip

A fundamental limitation of the use of the BP in the training of an MLP is the high degree of required computational accuracy. Three critical issues must be addressed in the parallel implementation of BP on efficient hardware[2]. These are availability of weight values for back propagating the error, the scaling and precision of computations, and the efficient implementation of output transfer function. Among them weight accuracy and output function are considered. URAN can be used for retrieval if low weight accuracy and linear output function are allowed.

Three critical issues must be addressed in the parallel implementation of BP on efficient hardware. These are availability of weight values for back propagating the error, the scaling and precision of computations, and the efficient implementation of output transfer function[ ]. And 16 bit integer weights are known to be sufficient for BP training and much lower values adequate for use after training[ ]. We can use URAN for retrieval only if low weight accuracy and linear output function is allowed for retrieval. Because the input and output accuracy of URAN depend upon the clock frequency.

The weight accuracy of URAN chip is 8 bit. To perform BP simulations using the weight accuracy available to URAN chip, we first do retrieval tests using the reduced floating point weight values which are saved after normal training phase. We also evaluate an extremely efficient feature vector for input, peak-weighted binary spectrum, to exploit fully URAN chip architecture[4].

As the exact analysis of the non-linear output transfer function on the analog hardware is not possible, the output of the neuron chip is devised to provide linear function. We second perform retrieval tests with the linear output function for the synapse trained with non-linear output function.

Finally, we perform simulations of training weights and retrieving using the piecewise linear function for the learning equation derived from the sigmoid output function. It was impossible to training weights with pure linear functions. Though weight update precision should be considered also for the on-chip learning to be possible, elaborate output precision using the fast clock frequency and efficient weight update mechanism using the off-chip synapse memory could lead to realization of the on-chip learning with the piecewise linear output neuron as described in section 2.

3 Simulations Using Low Accuracy Computation

3.1 Speech Database and Preprocessing

The simulations were performed to recognize isolated 10 Korean digits. The database was recorded in a silent office room from 10 male and 10 female speakers. Each digit was pronounced 10 times by each speaker. There were thus 2000 tokens in total. Among them 5 repetitions of each digit from 5 male and 5 female speakers ($5\ repetitions \times 10\ digits \times 10\ speakers = 500\ tokens$) were used for training, and the other data set (500 tokens) from the training speakers was used for multi-speaker recognition test, and the remaining 10 speakers' data ($10\ repetitions \times 10\ digits \times 10\ speakers = 1000\ tokens$) were used for speaker independent test.

Each utterance was low-pass filtered up to 4.7 kHz, then digitized at a 10 kHz sampling rate with 12 bit quantization. Manually endpointed speech data were preemphasized with a transfer function $H(z) = 1 - 0.98z^{-1}$. After passing through a 20 ms Hamming window at a rate of 10 ms, 17-channel critical-band filter bank analysis was performed to get a 17 dimensional feature vector for each frame. The 17-channel critical-band filter-bank was simulated by 512 point FFT. The data presented to the networks were simply scaled so that the range of the largest coefficient was approximately from -1 to 1.

A binary feature vector which is consisted of only 17 bits for each frame was also used for input. The binary spectrum was obtained by thresholding the second derivative of the normalized LPC spectrum, that is, by clamping spectral peaks above threshold as 1's and the others as 0's. We have not performed word length normalization so that variable number of feature vectors are applied to the input layer as a whole.

3.2 MLP Architecture and Training Algorithm

All MLPs trained have 17 × 70 input neurons, a variable number of hidden neurons from 20 to 50 and 10 output neurons. The network architecture used is illustrated in Fig. 3.

In general, MLP consists of weighted-sum with sigmoidal output functions. The activation value of output neuron k is defined as follows.

$$O_k = \frac{1}{1+e^{-\sum_k w_{kj}O_j}} \qquad (1)$$

where O_j is the activation value of hidden neuron j, and w_{kj} is the weight from hidden neuron j to output neuron k. The activation value of hidden neuron is also defined by the same function using the activation value of input neuron and weights from input to hidden neurons.

During the training process of MLP, the errors defined in equation (2) to be propagated between the target and output values are calculated after every feedforward of input.

$$E = \frac{1}{2}\sum_{\text{all patterns}}\sum_k(Target_k - O_k)^2 \qquad (2)$$

Changing the weights by a small amount in the direction of steepest descent minimizes the error and adjusts the internal parameters so as to better model the target input/output pairs.

To accelerate the learning speed a momentum γ is used in the weight changes.

$$w_{kj}(t) = \eta\Delta w_{kj} + \gamma w_{kj}(t-1) \qquad (3)$$

where η is the learning rate, and t is the learning step. The learning rate η and the momentum γ are set to 0.1 and 0.9, respectively.

The initial strength of the weights are distributed uniformly and randomly from -0.5 to 0.5. The learning procedure was repeated until the total-sum-of-squared-error of the network reaches at 0.01. The recognition phase employs a winner-take-all rule which allows the network to keep the most highly activated neuron in output layer.

3.3 Speaker Independent Digit Recognition Results

First, we performed simulations of BP training and testing with reduced accuracy of 4, 2, 1 decimal places for weight, input, and output using sigmoidal output function. Each MLP has 30 hidden units and is the learning step was constrained to 100. The speaker independent digit recognition results for all of the reduced floating point accuracy were around 97 %, while the result of binary feature vector was 88.4 %.

Second, the above simulations were repeated in the recognition phase using the linear output function as in equation (4).

$$y = a(\textstyle\sum_k w_{kj}O_j) + b, \quad -A \le \sum_k w_{kj}O_j \le A \qquad (4)$$

The results were not sensitive to a and b if the domain of the linear function broadly covers the sum-of-weighted-input distributions. The simulation results are listed in Table 2 and Table 3, in which a = 0.1, b = 0.5, and A = 5.

Each MLP output has 2 decimal places. The MLP using 1 decimal place input has weight accuracy of 1 decimal place, the MLP using binary input has weight accuracy of 2 decimal places. MLP using reduced floating point input is trained until 200 learning steps, and MLP using binary input until 500 steps.

hidden	floating point input		binary input
units #	2 decimal	1 decimal	1 bit
20	99.2 %	98.6 %	96.6 %
30	98.8 %	98.8 %	96.4 %
40	98.8 %	98.8 %	96.0 %
50	98.8 %	98.8 %	95.4 %

Table 2: Multi-speaker digit recognition results

hidden	floating point input		binary input
units #	2 decimal	1 decimal	1 bit
20	96.4 %	96.1 %	89.2 %
30	97.0 %	97.1 %	89.3 %
40	96.0 %	96.3 %	89.7 %
50	96.2 %	96.5 %	87.8 %

Table 3: Speaker independent digit recognition results

We can conclude that the BP retrieval is not affected from the accuracy of weight, input, and output, when training is completed. In addition, training was also possible with 2 or 1 decimal places for input and output, which are equivalent to 8 bit or 4 bit precisions.

Finally, we performed training using linear output functions. As the training using the simple linear functions such as equation (4) was not accomplished, we used piecewise linear functions combined with 3 linear functions. By minimizing the mean squared error(MSE) between the sigmoid and piecewise linear functions, a piecewise linear function is found as in equation (5). The MSE becomes zero when we calculate it up to 3 decimal places.

The recognition results, of which MLPs are trained and tested using the piecewise linear function with 2 decimal places of output resolution, are listed in Table 4. All MLPs have 30 hidden neurons, and learning step was the same as the previous conditions. The influence of the output precision was negligible for all cases as shown in the table. The performance of MLP using binary input was increased to 92.5 % with more slow slope function, that is, the performance was not sensitive to the slope of the piecewise linear function.

$$\begin{cases} y = 0.0, & x < -7.6 \\ y = 0.0087x + 0.066, & -7.6 \le x < -2.2 \\ y = 0.206x + 0.5, & -2.2 \le x < 2.2 \\ y = 0.0087x + 0.931, & 2.2 \le x < 7.6 \\ y = 1.0, & x > 7.6 \end{cases} \qquad (5)$$

output	floating point input		binary input
precision	2 decimal	1 decimal	1 bit
floating	96.6 %	96.3 %	90.1 %
2 decimal	96.6 %	96.1 %	90.0 %

Table 4: Recognition results of MLPs using piecewise linear units

4 Conclusions

In this paper, we described simulation results of the BP algorithms adapted for URAN chip, and showed that the reduced weight accuracy using the linear output function is enough to obtain high performance in speaker independent digit recognition experiments, once training was completed. We also illustrates that piecewise linear functions is useful for training when learning equations derived from the sigmoidal units are employed, using URAN chip and the simulated BP algorithms.

References

[1] IN. Morgan, "Making Useful Neurocomputers," in *Proc. of MICRONEURO'93*, Apr. 1993, pp. 297-305.

[2] J.L. Holt and J.N. Hwang, "Finite Precision Error Analysis of Neural Network Electronic Hardware Implementations," in *Proc. of IJCNN'91*, Vol. I, 1991, pp. 519-525.

[3] J.L. Holt and T.E. Baker, "Back Propagation Simulations Using Limited Precision Calculations," in *Proc. of IJCNN'91*, Vol. II, 1991, pp. 121-126.

[4] J.J. Choi, Seho Oh, and R.J. Marks II, "Training Layered Perceptrons Using Low Accuracy Computation," in *Proc. of IJCNN'91*, Vol. I, 1991, pp. 551-559.

[5] R.M. Debenham and S.C.J.Garth, "Investigations into the Effect of Numerical Resolution on the Performance of Back Propagation," in *Neural Networks from Models to Applications*, L. Personnaz and G. Dreyfys, Eds. Paris: I.D.S.E.T., 1989, pp. 752-755.

[6] Il-Song Han and Ki-Hwan Ahn, "Neural Network VLSI Chip Implementation of Analog-Digital Mixed Operation for more than 100,000 Connections," in *Proc. of MICRONEURO'93*, Apr. 1993, pp. 159-162.

[7] H. McCarter, "Back Propagation Implementation on the Adaptive Solutions CNAPS Neurocomputer Chip," in *Advances in Neural Information Processing Systems 3*, R.P. Lippman, J.E. Moody, and D.S. Touretzky, Eds. Dan Mateo, CA: Morgan Kaufman, 1991, pp. 1028-1030.

[8] J.L. Holt and T.E. Baker, "Back Propagation Simulations Using Limited Precision Calculations," in *Proc. of IJCNN'91*, Vol. II, 1991, pp. 121-126.

[9] K.C. Kim and J.W. Cho, "Robust Speech Recognition Using Frequency Weighted All-Pole Model Spectrum," *Computer Processing of Chinese & Oriental Languages*, Vol. 5, No. 3 & 4, Nov. 1991, pp. 203-216.

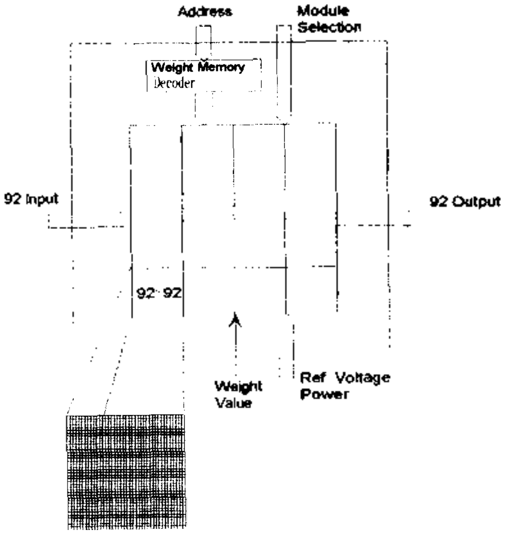

Fig 1 URAN-I block diagram

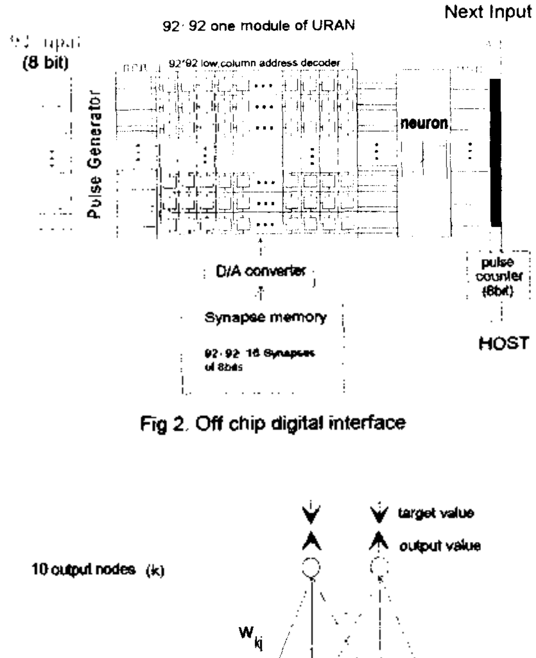

Fig 2. Off chip digital interface

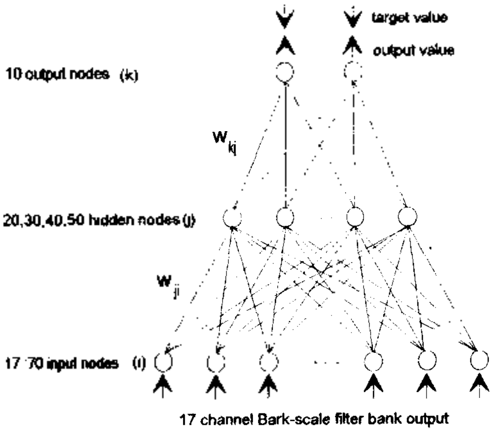

Fig 3. MLP architecture