

Performance of a Neural Net Scheduler used in Packet Switching Interconnection Networks

M. Mittler and P. Tran-Gia

Institute of Computer Science, University of Würzburg, Am Hubland, W-8700 Würzburg, F. R. Germany
Tel.: +49-931-8885513, Fax: +49-931-8884601, e-mail: mittler@informatik.uni-wuerzburg.de

Abstract—In this paper we present an analysis of an interconnection network for packet switching controlled by a neural network. The interconnection network is a crossbar switch of size N which maintains N queues at each of the inputs. The neural net controller is of Hopfield type and the performance evaluation is done under realistic input traffic assumptions. In order to investigate the influence of the random characteristics of the packet traffic on the neural net performance, we model the offered packet streams by means of batch inputs, where the interarrival time and the packet length can be arbitrarily chosen. The incoming traffic indicating source-destination packet streams is symmetrically distributed over the crosspoints of the interconnection network. Simulation results for various system parameters are presented with respect to the performance measures such as switch throughput and transfer delay. The performance of the neural net as switch fabric controller is compared to conventional control schemes, where the issue of scheduling fairness is addressed.

I. NEURAL NET AS INTERCONNECTION NETWORK SCHEDULER

A. Introduction

The implementation of the future high-speed telecommunication networks requires powerful high-capacity switch fabrics or interconnection networks. Due to the recommendation of the *asynchronous transfer mode* (ATM) to be used in B-ISDN (broadband integrated services digital network), data or information will be partitioned into packets or *cells* (in ATM) and transmitted over the network. This leads to the development of a number of switch fabric structures which handle cells, slots or minipackets

as information units. The most common operation mode of appropriate switch fabrics is the synchronous mode, i.e. the time axis is slotted and the cells are transmitted from the input to the output lines within one time slot.

Important packet switching fabric types under discussion in modern cell-based switching systems are mainly crossbar and banyan switches. Both types require a switch controller to select the cells for transmission within the next time slot taking into account certain conditions and constraints (such as minimizing the delay or maximizing the throughput of the switch). The advantage of crossbar switches is that they are internally nonblocking. Blocking may only occur at the inputs or at the outputs. The disadvantage is a costly second-order increase of the number of crosspoints by increasing number of input and output lines.

In general the switch fabric task can be formulated as a scheduling problem. The aim is to maximize throughput while taking into account fairness criteria. Furthermore in switching systems operating in high-speed environments, the time interval left to run the scheduling task is very short. It must be done during one cell duration; for a 140 Mbps (megabit per second) network and a standardized cell size of 53 bytes, the cell duration is about 3 μ sec. In such a short interval to perform scheduling task, suboptimal solutions are already of great interest.

It is well-known that some classes of neural nets are suitable to be used in optimization and scheduling problems (cf.[1]). In [2] a Hopfield net is proposed as scheduler in interconnection networks. In order to estimate the throughput of the switch and the time to perform the scheduling task the neural states were simulated. Further, a VLSI implementation of the scheduler was described. The same sche-

duler was investigated in [3] and [4] using different extensions of the energy function presented in [2]. The throughput of the switch and the transfer delay were observed. Most of the papers did not investigate systematically the performance measures considering stochastic traffic. In this paper we devote attention on performance evaluation of the proposed neural net controller under stochastic packet load conditions and compare the performance to conventional control schemes.

B. Neural net structure and working mode

We consider a crossbar switch with N input and N output lines. The switch is assumed to queue the cells at the input, each input maintains one queue for each output (see Fig. 1). Furthermore the switch operates in the full-duplex mode, i.e. within one time slot only one cell can be transmitted on a particular link in each direction.

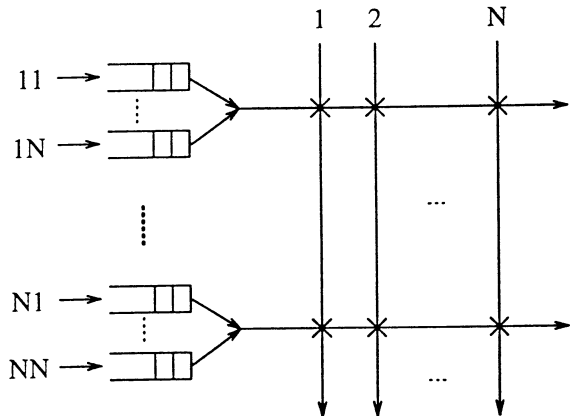


Figure 1: Crossbar interconnection network

A time slot is now observed. All pending transmission requests for this slot can simply be mapped onto a binary matrix $\mathbf{R} = (r_{ij})_{N \times N}$, which will be referred to as the request matrix, whereby $r_{ij} = 1$ indicates that there is at least one cell in queue j at input i waiting for transmission. During this slot, cells will be transmitted according to a schedule matrix \mathbf{S} , which has to be determined by the switch scheduler. As mentioned above, this has to be done within a slot duration, which is quite short in high-speed systems. To maximize the throughput of the switch within one

slot, the schedule matrix \mathbf{S} must be chosen in such a way that the overlap with the request matrix is maximized. Due to the full duplex transmission mode at most one entry $s_{ij} = 1$ is allowed within each row and each column.

For the switch scheduler, a hardware implementation of a Hopfield neural net is presented in [2]. This net maintains one neuron for each crosspoint of the crossbar switch and the neurons are only connected by rows and columns via inhibitory connections (see Fig. 2).

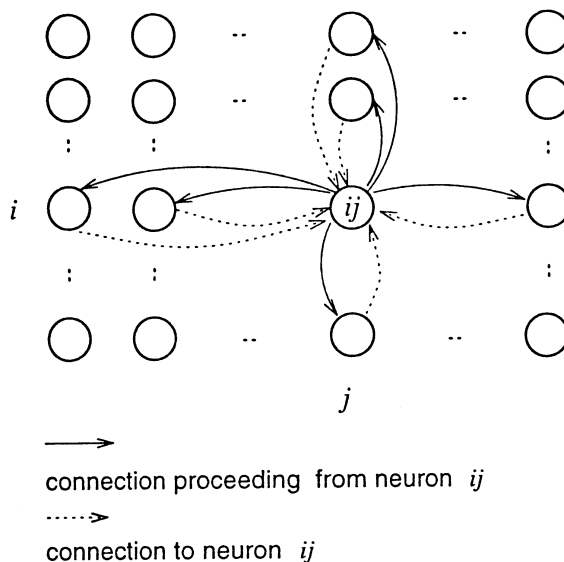


Figure 2: Neural net structure

The following energy function (without bias term) is assumed:

$$E = -\frac{A}{2} \sum_{i=1}^N \sum_{j=1}^N v_{ij} \sum_{\substack{k=1 \\ k \neq j}}^N v_{ik} - \frac{B}{2} \sum_{j=1}^N \sum_{i=1}^N v_{ij} \sum_{\substack{l=1 \\ l \neq i}}^N v_{lj}. \quad (1)$$

The first (second) term reaches its global minimum if at most one neuron is turned on within each row (column). The connection strength from one neuron to another within the same row (column) is denoted by A (B). As the authors pointed out, the results obtained for several 8×8 request matrices are optimal

for more than 98% of the cases. In all other cases the Hopfield net achieved a stable final state but the solutions correspond to non-optimal schedule matrices. The authors warranted the omission of a bias term in the energy function by the obtained results.

Note that the energy function also reaches the global minimum in the case of all neuron outputs being zero. Therefore in [3] the following bias term is added to the energy function given in eqn. (1):

$$+ \frac{C}{2} \left(\sum_{i=1}^N \sum_{j=1}^N v_{ij} - N \right)^2. \quad (2)$$

This additional term forces the schedule matrices to be permutation matrices, corresponding to final stable states with N neurons activated. Since the number of cells that can be transmitted without blocking may be lower than N , the schedule matrices have not always to be permutation matrices. A function to compute this number is presented in [4] but it does not work accurately. Moreover, the considered task is known as a matching problem appeared in graph theory and there are efficient methods to solve this class of problems.

The connection strengths are determined by partially differentiation of the energy function. Another method not taken into consideration in this paper is presented in [1].

Assuming the energy function to be the function given in eqn. (1) with a bias term added and applying $\frac{\partial^2 E}{\partial v_{ij} \partial v_{kl}}$ yields:

$$T_{ij,kl} = -A \delta_{ij} (1 - \delta_{jl}) - B \delta_{jl} (1 - \delta_{ij}). \quad (3)$$

The external bias is CN^2 . After the derivation of the connection strengths it remains to evaluate the parameters A , B and C . For reason of the symmetry of the connections we let $A = B$. Through several test runs we found that the Hopfield net performed well for $A = 100$ and an external bias of value 40. Nevertheless, it is imaginable that other parameter configurations might perform as well or even better. The evaluation of the parameters is right the crucial step in applying Hopfield nets to optimization problems. This step is a trial-and-error procedure and has to be done for each switch configuration.

As transfer function the following function was assumed:

$$v_{ij} = \frac{1}{2} \left[1 + \tanh(\beta u_{ij}) \right], \quad (4)$$

where u_{ij} is the input of neuron ij . Following [3] we set the gain factor β to 50 and initialize the starting neuron inputs with respect to the given request matrix:

$$u_{ij}^o = \begin{cases} -\beta^{-1} & \text{if } r_{ij} = 0 \\ \beta^{-1} & \text{if } r_{ij} = 1. \end{cases}$$

Simulating a Hopfield neural net with continuous neuron outputs requires to solve the following coupled set of ordinary differential equations:

$$\begin{aligned} \frac{du_{ij}}{dt} = & -A \sum_{\substack{c=1 \\ c \neq j}}^N V_{ic} - B \sum_{\substack{r=1 \\ r \neq i}}^N V_{rj} \\ & - \frac{u_{ij}}{\tau} + CN^2, \quad i, j \in \{1, \dots, N\}. \end{aligned} \quad (5)$$

The most commonly used methods for this purpose are explicit one-step methods. Since the eigenvalues of the weight matrices are often distinct (cf. [5]), the set of ordinary differential equations is characterized as *stiff*. In spite of the fact that explicit methods are not appropriate for integrating stiff differential equations, the results obtained can be used for practical purposes. Currently we are testing more efficient methods.

Since we update the neurons synchronously, the initial values u_{ij}^o are randomized to ensure the convergence to global minima in the following way:

$$\tilde{u}_{ij}^o := u_{ij}^o + \text{Uniform} \left[-0.1 \cdot \beta^{-1}; 0.1 \cdot \beta^{-1} \right],$$

where $\text{Uniform}[a; b]$ denotes the uniform distribution over the interval $[a; b]$.

II. NEURAL NET PERFORMANCE AND COMPARISON

In this section we first describe the model and subsequently present the results obtained using simulation.

A. Configuration and parameters

We take the simple example of a 8×8 crossbar interconnection network. Packets arrive according to

an interarrival time distribution function $A(t)$ with mean $\frac{1}{\lambda}$ and a distribution function of the packet length $X(k)$ with mean EX . The transmission time of a cell is equal to the length of one time slot. Thus within every slot at most N cells can be transmitted over an $N \times N$ crossbar interconnection network. Therefore the normalized utilization ρ of such a switch fabric is chosen as $\frac{\lambda \cdot EX}{N}$. The queueing strategy of the N queues at each input is assumed to be first-come, first-served.

As major performance measures of the interconnection network, we consider the following metrics:

- the *access delay* D , defined as the time passing from the packet arrival until the first appertaining cell will be transmitted and
- the *transmission time* T , the time interval from the arrival until the packet is completely transmitted.

The influence of the packet size distribution and the traffic intensity on the performance of the neural net scheduler, represented by these delays, is one object of the parameter study. The second object is a comparison of the performance of the neural net scheduler with a few conventional approaches, where the following methods are chosen:

- *round robin*: the scheduler picks up cells to be transmitted in a cyclic order
- *Hungarian algorithm*: the schedule matrix is calculated using the Hungarian algorithm which is well-known from graph theory [6].

The results shown in Fig. 3 - 5 below are obtained with 97.5% confidence intervals, which are too small to be drawn explicitly.

B. Results

First of all we devote attention to the fraction of optimal resulting schedule matrices obtained by the neural net scheduler under various switch utilizations, as shown in Table 1. The neural net controller yielded optimal throughput for more than 97% of the request matrices. The remaining schedule matrices are not optimal, although the stable states of the Hopfield net always represented global minima of the energy function. This fact is a consequence of

ρ	0.3	0.4	0.5	0.6	0.7	0.8	0.9
	1	0.998	0.995	0.988	0.978	0.976	0.969

Table 1: *Fraction of optimal schedule matrices*

the choice of the starting neuron activities. The network always approaches the equilibrium state being the local minimum closest to the starting point. This minimum does not always correspond to an optimal solution. For that reason, the mean access delay and mean transmission times of the neural net controller are usually longer than the corresponding times achieved by the matching algorithm which yields maximal throughput at each time slot (see Fig. 3).

In Fig. 3 the mean delay time and the mean transmission time are depicted as functions of the switch utilization. The packet size is hereby geometrically distributed. It is known from the *head of the line blocking* effect that the normalized throughput of a crossbar switch with only one queue per input has an upper bound of approximately 0.586 ([7]). The round robin strategy in this case is denoted by *round robin single request*. It should be mentioned that the computation using the Hungarian algorithm is most complex and time consuming among the three methods. We can observe in Fig. 3 that the neural net performance is practically comparable to the round-robin scheme.

To show the dependency of the neural net performance on the packet size and the interarrival distributions we take into account the following three parameter sets:

- (1) geometric interarrival time distribution and constant packet length,
- (2) the interarrival time and the packet length are geometrically distributed,
- (3) the interarrival time follows the hyperexponential distribution of order 2 with coefficient of variation $c = 5$ and the packet length is geometrically distributed.

Since the matching algorithm always yields the optimal schedule matrix we restrict our performance comparison in the following to the neural net scheduler and to the controller with round robin strategy.

In Fig. 4 the performance measures are depicted in dependency of the switch utilization. For the parameter sets (1) and (2) the mean access delay achieved by the round robin strategy is smaller. With higher variation of the arrival process the relation turns reversely. Further, the results of the sets (1) and (2) show that the mean transmission times are equal for both of the considered controllers. In the case of set (3) the neural net controller yields the best results. As illustrated in Fig. 5 these observations are confirmed for a fixed utilization and an increasing value of the mean packet length.

In the neural net scheduler solution discussed above we intentionally neglect the issue of scheduling fairness. The highest throughput does not always correspond to a fair schedule. While optimizing only the switch throughput, an individual cell could possibly be forced to wait a very long time until it will be transmitted. The resulting delay and transmission time of the affiliated packet can be very high. From this viewpoint the round robin schedule seems to offer an a priori fairness. It does not use the whole switch capacity within each time slot, but each (input, output)-pair is selected for transmission at least once every $2N$ slots.

III. CONCLUSION

In this paper we presented simulation results for a switch fabric in packet switching systems controlled by a neural network of Hopfield type. The performance of the neural net was compared with other conventional methods using simulation results. We focused on the switch throughput, the mean delay time and the mean transmission time of a packet while paying attention to the fairness of the scheduling algorithms. For the majority of request matrices under consideration, the neural net achieved stable states corresponding to optimal schedule matrices, i.e. the throughput was nearly maximized.

Acknowledgement The authors would like to thank H. Gold for the stimulating discussions.

REFERENCES

- [1] Tagliarini, R.G.; Christ, J.F.; Page, E.W.: *Optimization Using Neural Networks*, IEEE Transactions on Computers, vol. 40, no. 12, December 1991, pp. 533 - 541
- [2] Marrakchi, A.; Troudet, T.: *A Neural Net Arbitrator for Large Crossbar Packet-Switches*, IEEE Transactions on Circuits and Systems, vol. 36, no. 7, July 1989, pp. 1039 - 1041
- [3] Ali, M.M.; Nguyen, H.T.: *A Neural Network Implementation of an Input Access Scheme in a High-Speed Packet Switch*, in Proc. of the GLOBECOM Conference 1989, pp. 1192 - 1197
- [4] Ali, M.M.; Nguyen, H.T.: *A Neural Network Controller for a High-Speed Packet Switch*, in Proc. of the International Telecommunications Symposium 1990, pp. 493 - 497
- [5] Aiyer, S.V.B.; Niranjan, M.; Fallside, F.: *A theoretical Investigation into the Performance of the Hopfield Model*, IEEE Transactions on Neural Networks, vol. 1, no. 2, Juni 1990
- [6] Jungnickel, D.: *Graphen, Netzwerke und Algorithmen* (in German), Bibliographisches Institut, Mannheim, 1990
- [7] Karol, M.J.; Hluchij, M.G.; Morgan, S. P.: *Input Versus Output Queueing on a Space-Division Packet Switch*, IEEE Transactions on Communications, vol. Com-35, no. 12, December 1987, pp. 1347 - 1356
- [8] Ahmadi, H.; Denzel, W.E.: *A Survey of Modern High-Performance Switching Techniques*, IEEE Journal on Selected Areas in Communications, vol. 7, no. 7, September 1989
- [9] Roberts, J.W. (ed.): *EUR 14152 - COST 224 - Performance evaluation and design of multiservice networks*, Office for Official Publications of the European Communities, Luxembourg, 1992
- [10] Fuhrmann, S.W.: *Performance of a Packet Switch with Crossbar Architecture*, IBM Research Report, Zürich Research Laboratory 1989
- [11] Hopfield, J.J.: *Neurons with graded response have collective computational properties like those of two-state neurons*, in Proc. of the National Academy of Sciences USA 81, May 1984, pp. 3088 - 3092
- [12] Hopfield, J.J.; Tank, D.W.: *"Neural" Computations of Decisions in Optimization Problems*, Biological Cybernetics 52, 1985, pp. 141 - 152
- [13] Tank, D.T.; Hopfield, J.J.: *Simple "Neural" Optimization Networks: An A/D Converter, Signal Decision Circuit, and a Linear Programming Circuit*, IEEE Transactions on Circuits and Systems 33(5), May 1986, pp. 533 - 541

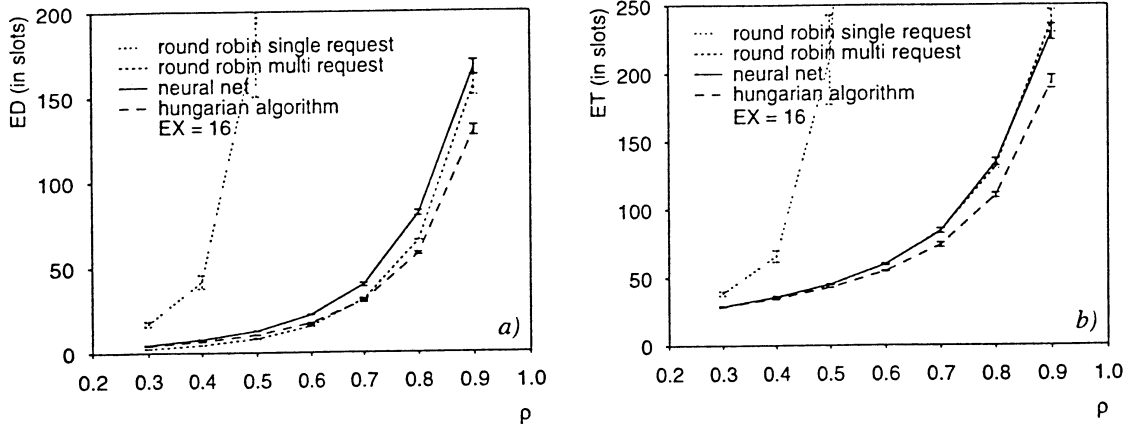


Figure 3: Performance comparison of various controllers

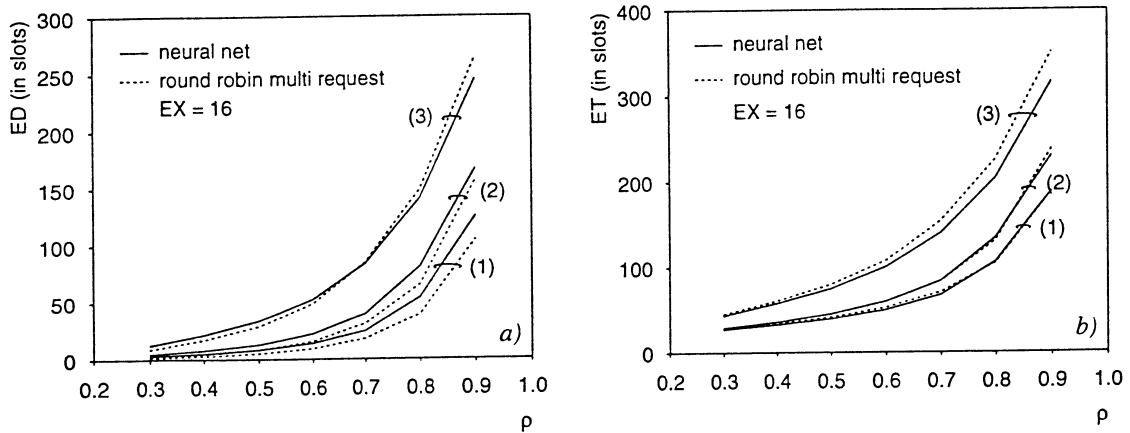


Figure 4: Influence of switch utilization on scheduler performance

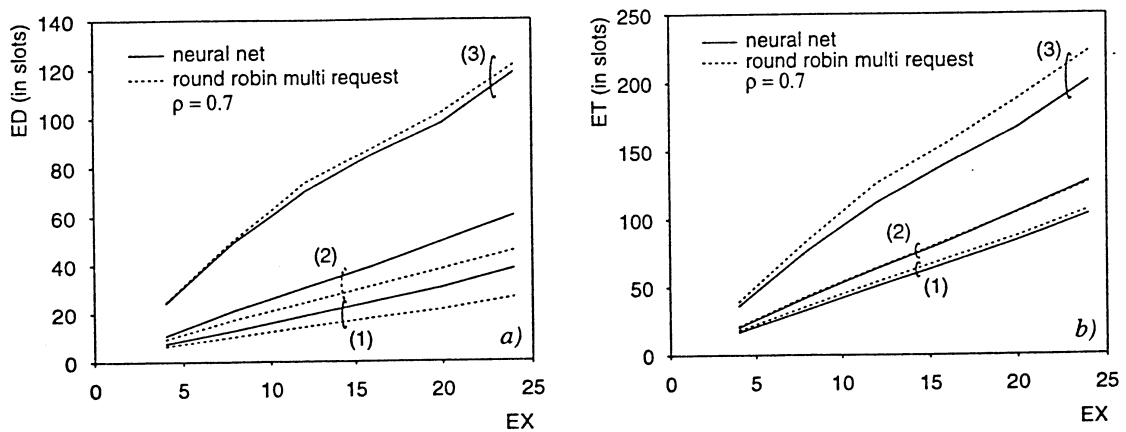


Figure 5: Influence of packet size on scheduler performance