

Combining competitive scheme with slack neurons to solve real-time job scheduling problem

Ruey-Maw Chen ^a, Shih-Tang Lo ^b, Yueh-Min Huang ^{b,*}

^a Department of Computer Science and Information Engineering, National Chin-yi Institute of Technology, Taichung 411, Taiwan, ROC

^b Department of Engineering Science, National Cheng-Kung University, Tainan 701, Taiwan, ROC

Abstract

Generally, how to satisfy the deadline constraint is the major issue in solving real-time scheduling. Recently, neural network using competitive learning rule provides a highly effective method and deriving a sound solution for scheduling problem with less network complexity. However, due to the availability of resources, the machines may not reach full utilization. To facilitate the problem the extra neuron is introduced to the competitive neural network (CHNN). This study tries to impose slack neuron on CHNN with respect to process time and deadline constraints. Simulation results reveal that the competitive neural network imposed on the proposed energy function with slack neurons integrated ensures an appropriate approach of solving this class of scheduling problems of single or multiple identical machines.

© 2006 Elsevier Ltd. All rights reserved.

Keywords: Scheduling; Slack neuron; Competitive learning; Hopfield neural network

1. Introduction

Neural networks have been widely used in a large area of applications like image processing, learning processes, identification and control, etc. But, there is a lack for their use for approximate solving real-time scheduling problems. Most problems are confirmed to be NP complete or combinatorial problems, especially for large-scale scheduling problem. The traveling salesman problem (TSP) is a typical NP-complete problem which seeks a tour that has a minimum cost obtaining the optimal solution is quite time consuming.

In general, job scheduling problems are seen as involving allocations of resources (like machines or processors) to execute a set of jobs satisfying a given type of constraints and optimizing a given criterion. Jobs are assigned timing constraints like ready time and deadline, and they need a maximum number of time units of processing time

(Cardeira & Mammeri, 1996). Many different schemes have been developed for solving the scheduling problem. Liu and Layland (1973) was the pioneering paper giving an insight to real-time scheduling algorithms for mono-job or scheduling of independent and periodic tasks. Linear programming is widely approached to minimize cost function from the specific scheduling problem. Willems and Rooda (1994) translated the job-shop scheduling problem onto a linear programming format, and then mapped it into an appropriate neural network structure to obtain a solution. Furthermore, Foo and Takefuji (1998) adopted integer linear programming neural networks to solve the scheduling problem by minimizing the total starting times of all jobs by a precedence constraint. Zhang, Yan, and Chang (1991) developed a neural network algorithm derived from linear programming, in which preemptive jobs are scheduled according to their priorities and deadline. Silva, Cardeira, and Mammeri (1997) explored the multi-process or real-time scheduling with a Hopfield-type neural network. Above investigations concentrating on the preemptive jobs executed on multiple machines with job

* Corresponding author.

E-mail address: huang@mail.ncku.edu.tw (Y.-M. Huang).

Nomenclature

N	total number of jobs/processes to be scheduled	$H(G_{ijk})$	unit step function. Defined to check if the timing constraint satisfied. A non-zero value indicates the assigned schedule violating the timing constraint. On the other hand, a zero value is yield as meet the timing requirement
M	total number of machines/processors to be operated	W_{xyzijk}	synaptic weight between neuron (x,y,z) and neuron (i,j,k)
T	deadline of the jobs	θ_{ijk}	input bias from outside of neuron (i,j,k)
i, j, k	denotes the “job”, “machine”, and “time” variables, respectively	Net_{ijk}	net value of neuron (i,j,k) , a neuron (i,j,k) receives a community of neuron with interconnection strength W_{xyzijk} and an input bias, θ_{ijk} , from outside
x, y, z	denotes the “job”, “machine”, and “time” variables, respectively	$\delta(a,b)$	Kronecker delta function. The value is 1 if a equals b . Otherwise, the value is zero
V_{ijk}, V_{xyz}	represents the binary states of neurons (i,j,k) and (x,y,z) on Hopfield neural network	E	energy function
P_i	denotes the total execution time required by process i	C_1, C_2, C_3, C_4, C_5	weighting factors of energy terms
d_i	deadline of the process i		
G_{ijk}	defined to examine that whether the time of process x finished in processor i later than the time limit		

transfer by a neural network. Moreover, Hanada and Ohnishi (1993) presented a parallel algorithm based on a neural network for task scheduling problems by permitting task transfer among machines. A classical local search heuristic algorithm was embedded into the TSP optimization neural network by Park et al. (1994). Most investigations have constructed the energy functions for scheduling problems in terms of timing constraint, preemption, and migration features associated with the process. Meanwhile, the neural networks were applied to solve scheduling problems extensively.

This work aims to find a feasible solution to generic scheduling problem. Most scheduling problem are concentrated on minimizing the maximum complete time (*make-span*), or minimizing the tardiness. Such problem to obtaining the optimal solution is quite time consuming. An advantage of real-time task scheduling is owing to its ability to meet task timing constraints rather than optimize a given target. Examples of real-time scheduling include nuclear power plant control system, traffic control systems, flight mission control system and embedded tactical systems for military applications. In these applications, failure to meet timing constraints of system might not only lead to system degradation, but even it may lead to a hazardous situation. To solve generic scheduling problems containing timing constraints similar to the above examples is the major consideration in this study. This work investigates a job scheduling problem involving preemptive multitasking with processing time and deadline constraints on the condition of no job migration allowed. A modified neural network with slack neuron is constructed to solve the scheduling problems.

Hopfield and Tank (1985) started the applications in using the neural network to solve optimization problems. In the Hopfield neural networks, the state input information from a community of neurons is received to decide

neuron output state information. Each neuron exchanges information with other neurons in the network. These neurons apply this information to cooperatively move the network to achieve convergence. The energy function used in the Hopfield neural network is an appropriate Lyapunov function. Many researchers have recently applied this method to various applications. Dixon, Cole, and Bellgard (1995) applied the Hopfield neural network with mean field annealing to solve the shortest path problem in a communication network. In our previous work also, we solved a multi-constraint schedule problem for a multiprocessor or system by the Hopfield neural network (Huang & Chen, 1999).

A competitive Hopfield neural network (CHNN) applies a competitive learning mechanism to update the neuron states in the Hopfield neural network. A competitive learning rule cannot only reduce the time consumed in obtaining coefficients but also obtain an effective and sound solution. CHNN has been applied in various fields, mostly on image processing such as image clustering processes and specific image segmentation. Chung, Tsai, Chen, and Sun (1994) presented a competitive Hopfield neural network for polygonal approximation. Uchiyama and Arbib (1994) used competitive learning in color image segmentation application. The winner-take-all rule employed by the competitive learning mechanism ensures that only one job is executed on a dedicated machine at a certain time, enforcing the *1-out-of-N* constraint to be held. The maximum output value neuron of the set of neurons is activated. The monotonic of the maximum neuron follows the fact that it is equivalent to a McCulloch and Pitts neuron with a dynamic threshold (Lee, Funabiki, & Takefuji, 1992). A series of studies have been done to fully utilized processors scheduling problem (Chen & Huang, 1998; Huang & Chen, 1999). Hopfield neural network scheme and mean field annealing technique are utilized to obtain an adequate schedule. The

Download English Version:

<https://daneshyari.com/en/article/388591>

Download Persian Version:

<https://daneshyari.com/article/388591>

[Daneshyari.com](https://daneshyari.com)