AI and Neural Networks (Assignment –II)

Submitted in partial fulfilment of the requirements for the degree of

Master of Technology in Information Technology

by

Vijayananda D Mohire

(Enrolment No.921DMTE0113)

Information Technology Department
Karnataka State Open University
Manasagangotri, Mysore – 570006
Karnataka, India

(2010)

AI and Neural Networks

CERTIFICATE

This is to certify that the Assignment-II entitled AI and Neural Networks, subject code: MT22 submitted by Vijayananda D Mohire having Roll Number 921DMTE0113 for the partial fulfilment of the requirements of Master of Technology in Information Technology degree of Karnataka State Open University, Mysore, embodies the bonafide work done by him under my supervision.

Place: _______________ Signature of the Internal Supervisor

Name

Date: _______________ Designation

For Evaluation

Question Number	Maximum Marks	Marks awarded	Comments, if any
1	5		
2	5		
TOTAL	10		

Evaluator's Name and Signature Date

Preface

This document has been prepared specially for the assignments of M.Tech – IT II Semester. This is mainly intended for evaluation of assignment of the academic M.Tech – IT, II semester. I have made a sincere attempt to gather and study the best answers to the assignment questions and have attempted the responses to the questions. I am confident that the evaluator's will find this submission informative and evaluate based on the provide content.

For clarity and ease of use there is a Table of contents and Evaluators section to make easier navigation and recording of the marks. Evaluator's are welcome to provide the necessary comments against each response; suitable space has been provided at the end of each response.

I am grateful to the Infysys academy, Koramangala, Bangalore in making this a big success. Many thanks for the timely help and attention in making this possible within specified timeframe. Special thanks to Mr. Vivek and Mr. Prakash for their timely help and guidance.

Candidate's Name and Signature Date

Table of Contents

Table of Figures

AI AND NEURAL NETWORKS
RESPONSE TO ASSIGNMENT - II

Question 1 Explain intelligent control and scheduling.

Answer 1

Intelligent control is a class of control techniques that use various AI computing approaches like neural networks, Bayesian probability, fuzzy logic, machine learning, evolutionary computation and genetic algorithms.

Intelligent control can be divided into the following major sub-domains:

- Neural network control
- Bayesian control
- Fuzzy (logic) control
- Neuro-fuzzy control
- Expert Systems
- Genetic control
- Intelligent agents (Cognitive/Conscious control)

New control techniques are created continuously as new models of intelligent behavior are created and computational methods developed to support them.

Neural network controllers

Neural networks have been used to solve problems in almost all spheres of science and technology. Neural network control basically involves two steps:

- System identification
- Control

It has been shown that a feed forward network with nonlinear, continuous and differentiable activation functions have universal approximation capability. Recurrent networks have also been used for system identification. Given, a set of input-output data pairs, system identification aims to form a mapping among these data pairs. Such a network is supposed to capture the dynamics of a system.

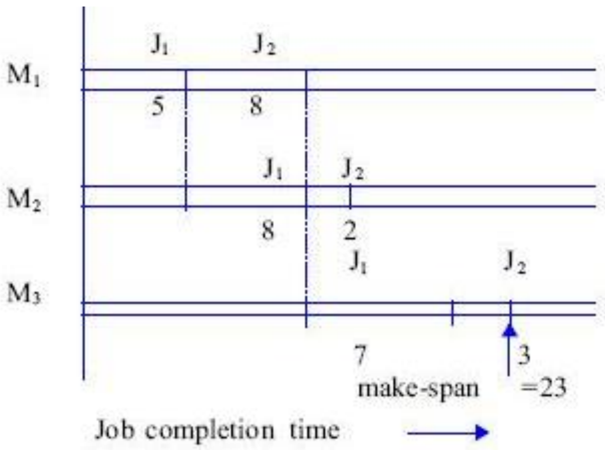
Bayesian controllers

Bayesian probability has produced a number of algorithms that are in common use in many advanced control systems, serving as state space estimators of some variables that are used in the controller.

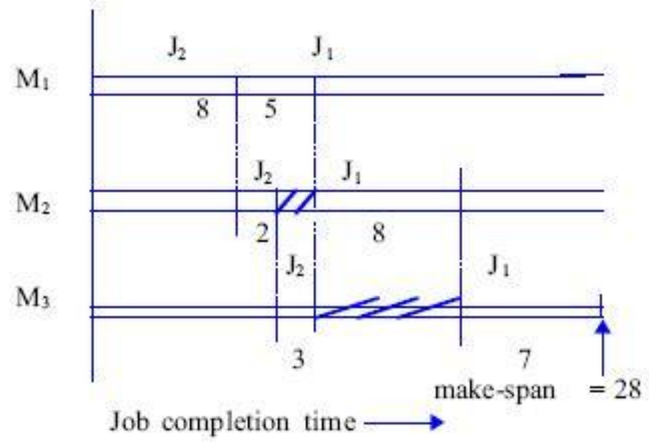
The Kalman filter and the Particle filter are two examples of popular Bayesian control components. The Bayesian approach to controller design requires often an important effort in deriving the so-called system model and measurement model, which are the mathematical relationships linking the state variables to the sensor measurements available in the controlled system. In this respect, it is very closely linked to the system-theoretic approach to control design.

Intelligent Control in process control: In process control, the controller is designed from the known models of the process and the required control objective. When the dynamics of the plant is not completely known, the existing techniques for controller design no longer remain valid. Rule-based control is appropriate in such situations. In a rule-based control system, the controller is realized by a set of production rules intuitively set by an expert control engineer. The antecedent (premise) part of the rules in a rule-based system is searched against the dynamic response of the plant parameters. The rule whose antecedent part matches with the plant response is selected and fired. When more than one rule is fireable, the controller resolves the conflict by a set of strategies. On the other hand, there exist situations when the antecedent part of no rules exactly matches with the plant responses. Such situations are handled with fuzzy logic, which is capable of matching the antecedent parts of rules partially/ approximately with the dynamic plant responses. Fuzzy control has been successfully used in many industrial plants. One typical application is the power control in a nuclear reactor. Besides design of the controller, the other issue in process control is to design a plant (process) estimator, which attempts to follow the response of the actual plant, when both the plant and the estimator are jointly excited by a common input signal. The fuzzy and artificial neural network-based learning techniques have recently been identified as new tools for plant estimation.

Intelligent Scheduling: In a scheduling problem, one has to plan the time schedule of a set of events to improve the time efficiency of the solution. For instance in a class-routine scheduling problem, the teachers are allocated to different classrooms at different time slots, and we want most classrooms to be occupied most of the time. In a flow shop scheduling problem, a set of jobs J1 and J2 (say) are to be allocated to a set of machines M1, M2 and M3. (say). We assume that each job requires some operations to be done on all these machines in a fixed order say, M1, M2 and M3. Now, what should be the schedule of the jobs (J1-J2) or (J2 -J1), so that the completion time of both the jobs, called the make-span, is minimized? Let the processing time of jobs J1 and J2 on machines M1, M2 and M3 be (5, 8, 7) and (8, 2, 3) respectively. The Gantt charts in fig. (a) and (b) describe the make-spans for the schedule of jobs J1 - J2 and J2 - J1 respectively. It is clear from these figures that J1-J2 schedule requires less make-span and is thus preferred.



(a) The J_1 - J_2 schedule.



(b): The J$_2$ - J$_1$ schedule where the hatched lines indicate waiting time of the machines.

Figure 1 The Gantt charts for the flow shop scheduling problem with 2 jobs and 3 machines.

Flow shop scheduling problems are a NP complete problem and determination of optimal scheduling (for minimizing the make-span) thus requires an exponential order of time with respect to both machine-size and job-size. Finding a sub-optimal solution is thus preferred for such scheduling problems. Recently, artificial neural nets and genetic algorithms have been employed to solve this problem. The heuristic search has also been used for handling this problem.

Evaluator's Comments if any:

Question 2 Write a short note on Synaptic Dynamics Models?

Answer 2

Synaptic Dynamics is attributed to learning in a biological neural network. The Synaptic weights are adjusted to learn the pattern information in the input samples. Typically, learning is a slow process, and the samples containing a pattern may have to be presented to the network several times before the pattern information is captured by the weights of the network. A large number of samples are normally needed for the network to learn the pattern implicit in the samples. Pattern information is distributed across all the weights, and it is difficult to relate the weights directly to the training samples. The only way to demonstrate the evidence of learning pattern information is that, given another sample from the same pattern source, the network would classify the new sample into the pattern class of the earlier trained samples. Another interesting feature of learning is that the pattern information is slowly acquired by the network from the training samples, and the training samples themselves are never stored in the network. That is why we say that we learn from examples and not store the examples themselves.

The adjustment of the synaptic weights is represented by a set of learning equations, which describe the synaptic dynamics of the network. The learning equation describing a synaptic dynamics models given as an expression for the first derivative of the synaptic weight w_{ij} connecting the units j to the unit i. The set of equations for all the weights in the network determine the trajectory of the weight states in the weight space from a given initial weight state.

Learning laws refer to the specific manners in which the learning equations are implemented. Depending on the synaptic dynamics model and the manner of implementation, several learning laws have been proposed in the literature. Following are some of the requirements of the learning laws for effective implementation.

a) The learning law should lead to convergence of weights
b) The learning or training time for capturing the pattern information from samples should be as small as possible.
c) An on-line learning is preferable to an off-line learning. That is the weights should be adjusted on presentation of each sample containing

the pattern information.

d) Learning should use only the local information as far as possible. That is, the change in the weight on a connecting link between two links should depend on the states of these two units only. In such a case, it is possible to implement the learning law in parallel for all the weights, thus speeding up the learning process.

e) Learning should be able to capture complex non linear mapping between input-output pattern pairs, as well as between adjacent patterns in a temporal sequence of patterns.

f) Learning should be able to capture as many patterns as possible into the network. That is, the pattern information storage capacity should be as large as possible for a given network.

Evaluator's Comments if any: