UNIVERSITI SAINS MALAYSIA

Second Semester Examination
2012/2013 Academic Session

June 2013

EPE 401 – Artificial Intelligence In Manufacturing

[Kecerdikan Rekaan Dalam Pembuatan]

Duration : 2 hours
Masa : 2 jam

Please check that this paper contains **SEVEN (7)** printed pages and **FOUR (4)** questions before you begin the examination.

*[Sila pastikan bahawa kertas soalan ini mengandungi **TUJUH (7)** mukasurat bercetak dan **EMPAT (4)** soalan sebelum anda memulakan peperiksaan.]*

INSTRUCTIONS : Answer **ALL** questions. You may answer all questions in **English** OR **Bahasa Malaysia** OR a combination of both.

[ARAHAN : *Jawab **SEMUA** soalan. Calon boleh menjawab semua soalan dalam **Bahasa Malaysia** ATAU **Bahasa Inggeris** ATAU kombinasi kedua-duanya.]*

Answer to each question must begin from a new page.

[Jawapan untuk setiap soalan mestilah dimulakan pada mukasurat yang baru.]

In the event of any discrepancies, the English version shall be used.

[Sekiranya terdapat sebarang percanggahan pada soalan peperiksaan, versi Bahasa Inggeris hendaklah diguna pakai.]

Formulas given for feedforward multilayer perceptron neural networks

Formula yang diberikan bagi rangkaian perceptron masuk depan berbilang neural

Activation functions

$$X = \sum_{i=1}^{n} x_i w_i - \emptyset$$

$$Y^{step} = \begin{cases} 1, if\ X \geq 0 \\ 0, if X < 0 \end{cases}$$

$$Y^{sign} = \begin{cases} +1, if\ X \geq 0 \\ -1, if X < 0 \end{cases}$$

$$Y^{sigmoid} = \frac{1}{1 + e^{-X}}$$

$$Y^{linear} = X$$

Error back propagation functions

$$e(p) = Y_d(p) - Y(p)$$
$$w_i(p+1) = w_i(p) + \alpha \times x_i(p) \times e(p)$$
$$\Delta w_{jk}(p) = \alpha \times \gamma_j(p) \times \delta_k(p)$$

Error gradient for neurons in the output layer

$$\delta_k(p) = y_{k(p)} \times [1 - y_k(p)] \times e_k(p)$$
$$e_k(p) = y_{d,k}(p) - y_k(p)$$
$$\Delta w_{jk}(p) = \alpha \times \gamma_j(p) \times \delta_k(p)$$
$$w_{jk}(p+1) = w_{jk}(p) + \Delta w_{jk}(p)$$

Error gradient for neurons in the hidden layer

$$\delta_j(p) = y_{j(p)} \times [1 - y_j(p)] \times \sum_{k=1}^{l} \delta_k(p) \times w_{jk}(p)$$

$$\Delta w_{ij}(p) = \alpha \times x_i(p) \times \delta_j(p)$$

$$w_{ij}(p+1) = w_{ij}(p) + \Delta w_{ij}(p)$$

Euclidean distance

The Euclidean distance between a pair of n-by-1 vectors X and W_j is defined by

$$d = \|X - W_j\| = \left[\sum_{i=1}^{n} (x_i - w_{ij})^2\right]^{1/2}$$

Where x_i and w_{ij} are the *i*th elements of the vectors X and W_j, respectively.

Q1. [a] A purchasing department manager intends to build a rule-based expert system to classify the company global suppliers based on product pricing, product quality and delivery capability. The decision chart is provided in the Figure Q1[a]. Construct the rule-based expert system for this application. Each rule carries a maximum of two antecedents.

Seorang pengurus jabatan pembelian bercadang untuk membina sebuah sistem pakar berasaskan peraturan untuk mengklasifikasikan pembekal global syarikat berdasarkan harga produk, kualiti produk dan keupayaan penghantaran. Carta keputusan disediakan dalam Rajah S1[a]. Bina sebuah sistem pakar untuk aplikasi tersebut. Setiap peraturan masing-masing membawa maksimum dua bahagian penentu.

(70 marks/*markah*)

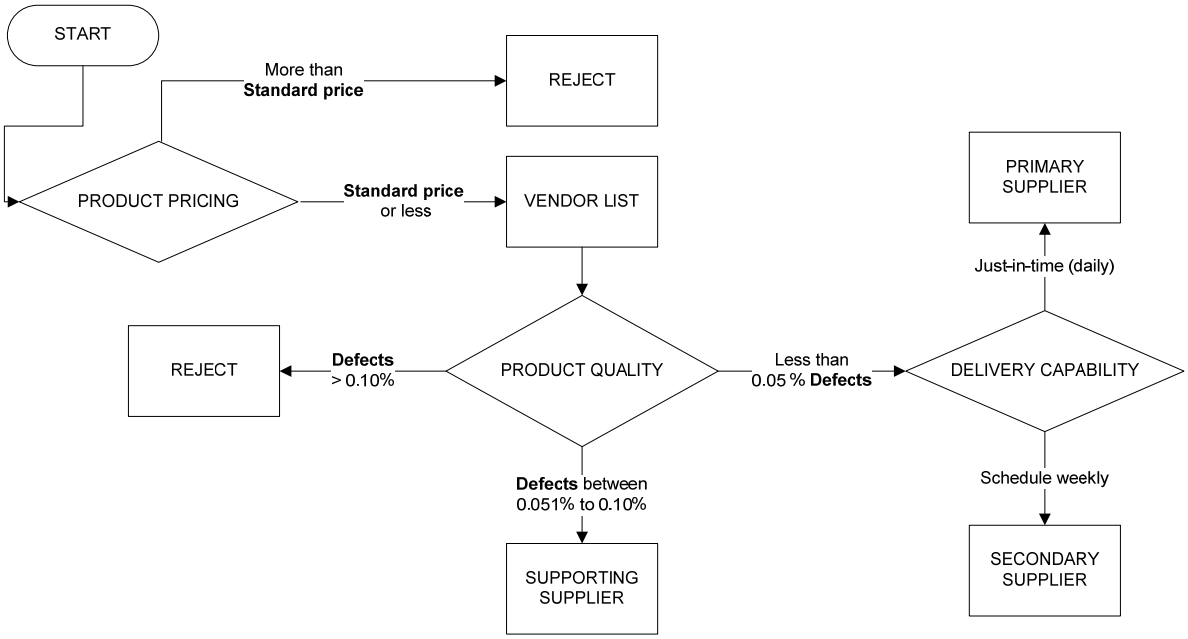

Figure Q1[a]
Rajah S1[a]

[b] Based on the expert system constructed in Q1[a], illustrate the forward chaining and backward chaining.

Berdasarkan sistem pakar yang dibina dalam S1[a], ilustrasi perantaian ke depan dan perantaian ke belakang.

(30 marks/*markah*)

Q2. As shown in Figure Q2[a], a robot contains four sensors, S1, S2, S3 and S4 around the body circumference. Each sensor reads the distance to the nearest object in its direction. The range of value for the sensors is 0 to 10. The higher value indicates that an object is close. The movement of the robot is controlled by speed controller consists of two wheels, Left wheel (LW) and Right wheel (RW). Both wheels are moved together. The speed of each wheel ranges from -10 to 10. Positive value of the speed means moving forward while negative value of the speed means moving backward. A third wheel (CW) is installed to control the direction of the robot. CW receives value between -90 to 90, indicating the degree of turning. A negative value of CW will turn the robot to the left while a positive value will turn the robot to the right. The robot needs to move away from any obstacle as far as possible.

Seperti yang ditunjukkan dalam Rajah S2[a], sebuah robot mempunyai empat sensor, S1, S2, S3 dan S4 pada sekitar lilitan badan. Setiap sensor membaca jarak ke objek yang terdekat dalam arahnya. Julat nilai untuk sensor adalah dari 0 hingga 10. Nilai tinggi menunjukkan bahawa suatu objek adalah dekat. Pergerakan robot dikawal oleh pengawal kelajuan yang terdiri daripada dua roda, roda kiri (LW) dan roda kanan (RW). Kedua-dua roda bergerak bersama-sama. Kelajuan setiap roda adalah antara -10 hingga 10. Nilai positif kelajuan bermakna bergerak ke hadapan manakala nilai negatif kelajuan bermakna bergerak mundur. Sebuah roda ketiga (CW) dipasang untuk mengawal arah robot. CW menerima nilai antara -90 hingga 90, menunjukkan tahap membelok. Nilai negatif pada CW akan mengarah robot ke kiri manakala nilai positif akan mengarah robot ke kanan. Robot ini perlu untuk bergerak dari sebarang halangan sejauh mungkin.

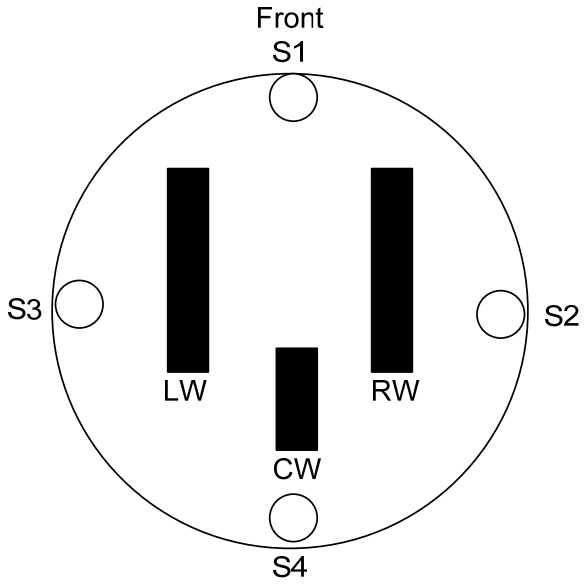

Figure Q2[a]
Rajah S2[a]

[a] Present the fuzzy sets in diagrams for sensors S1, S2, S3, S4, speed controller for LW and RW and direction controller for CW. Each sensor or controller is recommended to have two fuzzy sets.

Bentangkan set-set 'fuzzy' dalam rajah untuk sensor S1, S2, S3, S4, pengawal kelajuan bagi LW dan RW dan pengawal arah bagi CW. Setiap sensor atau pengawal dicadangkan mempunyai dua set fuzzy.

(40 marks/*markah***)**

[b] Develop required fuzzy rules where fuzzy sets from sensors S1, S2, S3 and S4 are the antecedents and fuzzy sets from speed controller (LW and RW) and direction controller (CW) are the output actions.

Bangunkan peraturan-peraturan 'fuzzy' yang diperlukan di mana set-set fuzzy dari sensor S1, S2, S3 dan S4 adalah bahagian penentu dan set-set fuzzy dari pengawal kelajuan (LW and RW) dan pengawal arah (CW) adalah tindakan-tindakan keluaran.

(30 marks/*markah***)**

[c] If in one case the sensor output (S1, S2, S3, S4) = (6, 0, 6, 2), calculate the value for speed controller and direction controller from your fuzzy logic system developed in Q2[a] and [b] based on Mamdani-style inference.

Jika dalam satu kes keluaran bagi sensor (S1, S2, S3, S4) = (6, 0, 6, 2), kirakan nilai bagi pengawal kelajuaan dan pengawal arah dari sistem logik 'fuzzy' yang dibangunkan dalam S2[a] dan [b] berdasarkan inferens gaya-Mamdani.

(30 marks/*markah***)**

Q3. Artificial neural networks are to be constructed to identify the class of bomber airplane based on speed and fuel tank capacity. Four data samples listed in Table Q3 are used to provide the training data to the artificial neural networks. Questions Q3[a] and Q3[b] will be based on this case.

Rangkaian-rangkaian saraf tiruan perlu dibina untuk mengenalpasti kelas kapal terbang 'bomber' berdasarkan kelajuan dan kapasiti tangki bahan api. Empat sampel data yang disenaraikan dalam Jadual S3 digunakan untuk menyediakan latihan kepada rangkaian-rangkaian saraf tiruan. Soalan S3[a] dan S3[b] akan berdasarkan kes ini.

Table Q3
Jadual S3

Speed	Tank capacity	Class
0.1	2.5	Bomber XS23
0.2	1.4	Bomber X-T
0.5	2.1	Bomber XS23
0.4	1.2	Bomber X-T

[a] Design a single perceptron to identify class of bomber airplane based on speed and fuel tank capacity. Show ONE (1) cycle of training iterations including all data samples.

Reka bentuk satu 'perceptron' tunggal untuk mengenalpasti kelas kapal terbang 'bomber' berdasarkan kelajuan dan kapasiti tangki bahan api. Tunjukkan SATU (1) lelaran pengiraan termasuk kesemua sampel data.

(50 marks/*markah***)**

[b] Design a "Winner-takes-all" neural network for the same task. Show ONE (1) cycle of training iteration by including all the data samples.

Reka bentuk satu rangkaian saraf tiruan "Winner-takes-all" untuk tugasan yang sama. Tunjukkan SATU (1) lelaran pengiraan termasuk kesemua sampel data.

(50 marks/*markah***)**

Q4. A genetic algorithm program is used to solve capacity allocation problem in a factory. Four products, X1, X2, X3 and X4 with different quantities are to be produced weekly. Each product has a standard process flow. They are required to go through four machines: machine A, machine B, machine C and machine D in series. The cycle times are different for individual products. Each machine has a maximum capacity of 10,000 time unit per week and production cannot exceed this limit. The information is shown in Table Q4. The genetic algorithm program has to maximize the total profit of the factory by deciding the quantity to produce for each product per week. Questions Q4[a] to Q4[c] have to be answered based on the above capacity allocation case.

Table Q4
Jadual S4

Product	Profit per unit (RM)	Cycle time per unit (in time unit)			
		Machine A	Machine B	Machine C	Machine D
X1	15	2	7	4	3
X2	9	3	2	4	1
X3	18	4	5	6	8
X4	7	5	5	7	0

Satu program algoritma genetik digunakan untuk menyelesaikan masalah peruntukan kapasiti di kilang. Empat produk, X1, X2, X3 dan X4, dengan kuantiti yang berbeza akan dihasilkan setiap minggu. Setiap produk mempunyai aliran proses standard. Produk-produk tersebut dikehendaki melalui empat mesin dalam siri: mesin A, mesin B, mesin C dan mesin D. Masa kitaran adalah berbeza bagi produk individu. Setiap mesin mempunyai kapasiti maksimum 10,000 unit masa seminggu dan pengeluaran tidak boleh melebihi had ini. Maklumat berkenaan ditunjukkan dalam Jadual S4. Program algoritma genetik ini perlu memaksimumkan jumlah keuntungan kilang dengan memutuskan kuantiti untuk setiap produk yang dihasilkan seminggu. Soalan-soalan Q4[a] hingga Q4[b] perlu dijawab berdasarkan kes peruntukan kapasiti di atas.

[a] Formulate the objective function.

Formulasikan fungsi objektif.

(20 marks/*markah***)**

[b] Design a suitable solution representation for the genetic algorithm program, with an example and calculation.

Reka bentuk suatu perwakilan penyelesaian yang sesuai untuk program algoritma genetik, dengan satu contoh dan pengiraan.

(30 marks/*markah***)**

[c] Illustrate with suitable examples:

(i) Roulette wheel selection mechanism.
(ii) One-point crossover.
(iii) Two-point crossover.

Ilustrasi dengan contoh-contoh yang sesuai:
(i) Mekanisma pemilihan roda roulette.
(ii) Operasi bertukar ganti satu titik.
(iii) Operasi bertukar ganti dua titik.

(50 marks/*markah***)**

-oooOOooo-