**University of Alberta**
**Dept. of Electrical and Computer Engineering**

**ECE449 Intelligent Systems Engineering**

# SAMPLE FINAL EXAMINATION

# WITH SOLUTIONS

This sample final exam shows the overall structure of the exam and *types* of questions that can be expected, not *specific questions* or *contents* that will be covered in the actual exam.

| Question(s) | Worth | Marks | Subject |
|---|---|---|---|
| I. (1-5) | 10 | | Various multiple choice |
| II. (6-8) | 10 | | Various short answer |
| III. (9-11) | 30 | | Neural Networks |
| IV. (12-14) | 30 | | Fuzzy Systems |
| V. (15-17) | 20 | | Evolutionary Computing |
| **Total** | **100** | | **-** |

**I. Various multiple choice** [10 marks total]

1. [2 marks] Which of the following statements about lateral inhibition is TRUE:

☐ After the iterative process is completed, multiple neurons remain active
☐ Outputs of each neuron are fed back to its neighbors' inputs through negative weights
☐ Outputs of each neuron are fed back to its input through a negative weight
☐ Outputs of each neuron are fed back to its neighbors' inputs through positive weights
☐ Outputs of all neurons are fed back to the inputs of neurons in the previous layer

2. [2 marks] Which of the following relationships CANNOT be classified as a fuzzy relation:

☐ Age – marital status
☐ Similarity of integers from the interval [1, 5]
☐ Canadian province – provincial sales tax
☐ Gender – weight
☐ Processor speed – processor cost

3. [2 marks] The purpose of the momentum term added to backpropagation learning algorithm is to

☐ Linearize the dependency of weight change with respect to error
☐ Eliminate local minima on error surface
☐ Increase the learning step unconditionally
☐ Increase the learning step when gradient changes direction
☐ Decrease the learning step when gradient changes direction

4. [2 marks] In Kohonen self-organizing maps, weights in the neigbourhood of the winning neuron are updated

☐ To avoid overtraining
☐ To make resulting clusters larger
☐ To preserve topology of the input space
☐ To filter out small disturbances
☐ To facilitate self-organization

5. [2 marks] Which of the following statements is NOT true about fuzzy sets

☐ They are based on concept of gradual membership
☐ They can express linguistic descriptions
☐ They provide generalization of conventional (crisp) set theory
☐ They are based on theory of probability
☐ They can be described by a membership function

**II. Various short answer** [10 marks total]

*1.* [2 marks] Briefly explain the concept of *neural plasticity*

Ability of some parts of the brain to developed through learning, especially in early stages of life, to adapt to the environment (new inputs); as opposed to the part of the neural structure of the brain present at birth.

*2.* [5 marks] Fill in the blanks in the following table with appropriate words from the following list {inputs; outputs; weights; totals, targets}:

| | | | |
|---|---|---|---|
| Percepron learning rule is based on comparing | outputs | and | targets |
| ADALINE learning rule is based on comparing | totals | and | targets |
| Backpropagation learning rule is based on comparing | outputs | and | targets |
| Grossberg instar learning is based on comparing | inputs | and | weights |
| Grossberg outstar learning is based on comparing | outputs | and | weights |

*3.* [3 marks] Name three selection procedures commonly used in genetic algorithms.

Fitness-proportional (roulette), ranked, tournament selection

**III. Fuzzy Systems** [30 marks total]

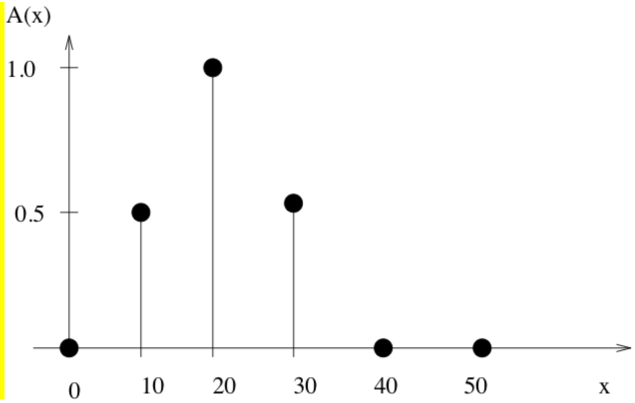
*4.* [6 marks] Suppose variable *x* is defined on a discrete universe of discourse

$$X = \{0, 10, 20, 30, 40, 50\}$$

and that *A* is a fuzzy set on this universe of discourse defined as

$$A = \{A(x)|x\} = \{0.0|0\ ,\ 0.5|10,\ 1.0|20,\ 0.5|30,\ 0.0|40,\ 0.0|50\}$$

a) [2 marks] Sketch the membership function of fuzzy set *A*




b) [4 marks] Determine the following properties of fuzzy set *A*:

Height: Hgt(*A*) = 1

Support: Supp(*A*) = {10, 20, 30} ([10, 30] is also acceptable}

Core: Core(*A*) = 20

Cardinality: $\text{Card}(A) = \sum_i A(x_i) = 0.5 + 1.0 + 0.5 = 2.0$

5. [6 marks total] Consider a fuzzy relation, R, defined in matrix form

$$R(x,y)=\begin{bmatrix} 1.0 & 0.4 & 0.8 & 0.3 & 0.0 \\ 0.5 & 1.0 & 0.6 & 0.7 & 1.0 \\ 0.9 & 1.0 & 0 & 0.6 & 0.8 \\ 1.0 & 0.5 & 0.2 & 0.0 & 0.9 \\ 0.3 & 0.5 & 0.3 & 0.1 & 1.0 \end{bmatrix}$$

a) [2 marks] Determine projections Proj$R_x$ and Proj$R_y$ of the relation $R$.

Proj$R_x(x)$ = $\sup_{x \in X} R(x,y)$ = $\begin{bmatrix} 1.0 \\ 1.0 \\ 1.0 \\ 1.0 \\ 1.0 \end{bmatrix}$

Proj$R_y(y)$ = $\sup_{y \in Y} R(x,y)$ = $\begin{bmatrix} 1.0 & 1.0 & 0.8 & 0.7 & 1.0 \end{bmatrix}$

b) [2 marks] Reconstruct the relation from these projections using min ***t***-norm.

$$R_{\text{rec}}(x,y)=\min_{x,y}\left(\text{Proj}_x(x),\text{Proj}_y(y)\right)=\begin{bmatrix} 1.0 & 1.0 & 0.8 & 0.7 & 1.0 \\ 1.0 & 1.0 & 0.8 & 0.7 & 1.0 \\ 1.0 & 1.0 & 0.8 & 0.7 & 1.0 \\ 1.0 & 1.0 & 0.8 & 0.7 & 1.0 \\ 1.0 & 1.0 & 0.8 & 0.7 & 1.0 \end{bmatrix}$$

c) [2 mark] Comment on quality of the reconstruction obtained in 10b).

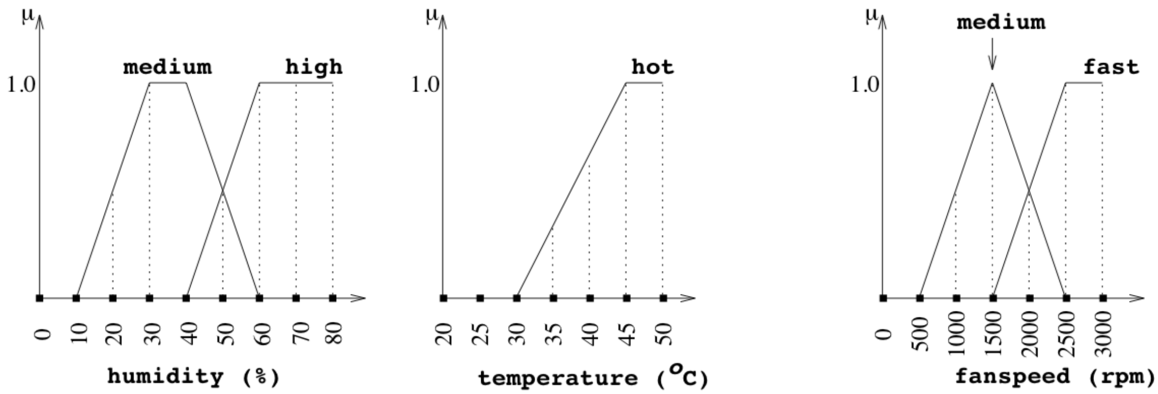
$$R_{\text{rec}}(x,y)\cap R(x,y)$$

The projection operations perform data compression; therefore some information is lost which cannot be recovered using reconstruction.

*6.* [18 marks total] For a fuzzy controller with the following algorithm

IF `humidity` is `high` AND `temperature` is `hot` THEN `fanspeed` is `fast` ELSE
IF `humidity` is `medium` AND `temperature` is `hot` THEN `fanspeed` is `medium`

and with fuzzy sets defined as follows:



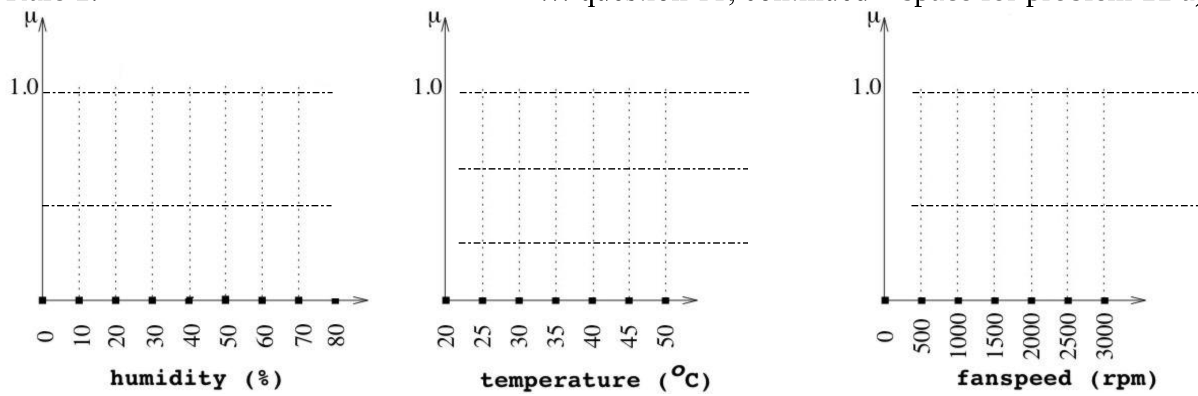

a) [8 marks] Using the axes provided on the following page, derive the output fuzzy set corresponding to crisp inputs of **humidity** = 50% and **temperature** = 40°C using graphical method. Use **min** (Mamdani) inference, and **min** ***t***-norm as the interpretation of the AND within the antecedents.

b) [4 marks] Calculate the crisp `fanspeed` corresponding to the fuzzy set found in a) using the SCOA method.

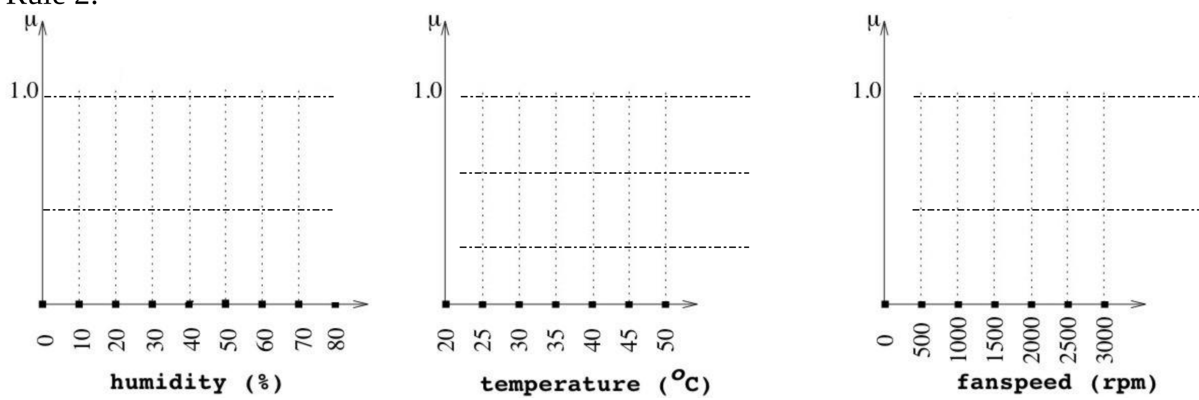
$$b = SCOA(B(y)) = \frac{\sum B(y) \cdot y}{\sum B(y)} = \frac{.5 \cdot 1000 + .5 \cdot 1500 + .5 \cdot 2000 + .5 \cdot 2500 + .5 \cdot 3000}{.5 + .5 + .5 + .5 + .5} =$$

= 2000 rpm

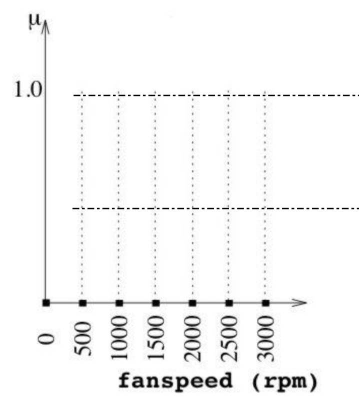
Rule 1:




Rule 2:




Output fuzzy set:

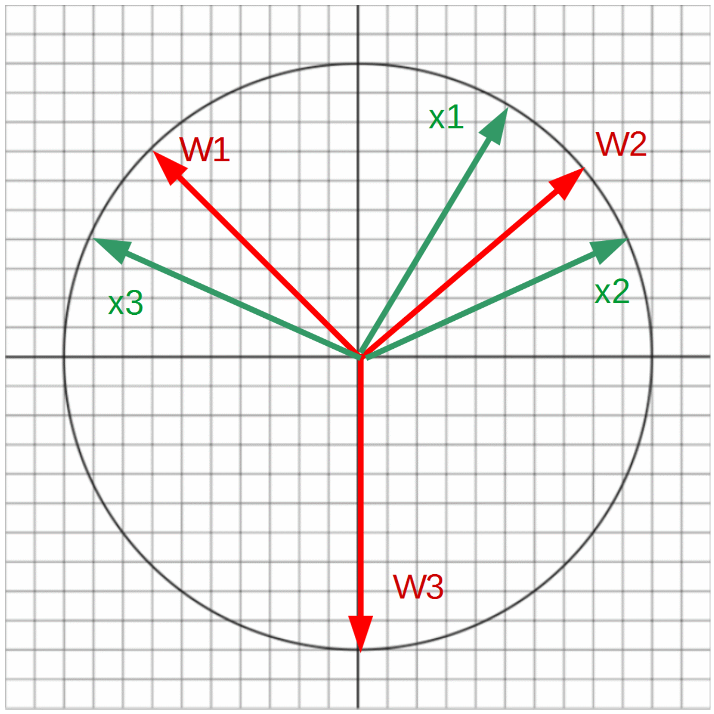



c) [6 marks] Comment on the expected performance of this controller [Hint: Use visual inspection to examine the static properties of the rule base.]

The fuzzy sets defined on input (humidity, temperature) and output (fan speed) do not cover their respective universes of discourse. Therefore, there are no rules for certain values of inputs (humidity < 10%, temperature < 30ºC) and the system would not provide any control signal for such values. At the same time, values of fan speed < 500 rpm are not covered, which means that fan speed cannot be set to a value within this range.

**IV. Neural Networks** [30 marks total]

*7.* [10 marks total] Consider a competitive network with two inputs and three neurons in the competitive layer. Plot of the input vectors and initial weights in a unit circle is provided below:




a) Determine the resulting weights (in terms of original weights and/or input vectors, i.e. not numerically) found after training with the competitive rule with learning rate $\eta = 1$, using the following sequence of training inputs: $\mathbf{x}(1)$, $\mathbf{x}(2)$, $\mathbf{x}(3)$ [3 marks]

| **Pattern presented** | **Winner ($\mathbf{w}_i$)** | **Winner updated to** |
|---|---|---|
| $\mathbf{x}(1)$: | $\mathbf{w}_2$ | $\mathbf{w}_2 = \mathbf{x}(1)$ |
| $\mathbf{x}(2)$: | $\mathbf{w}_2$ | $\mathbf{w}_2 = \mathbf{x}(2)$ |
| $\mathbf{x}(3)$: | $\mathbf{w}_1$ | $\mathbf{w}_1 = \mathbf{x}(3)$ |

| Summary of weights after training |
|---|
| $\mathbf{w}_1 = \mathbf{x}(3)$ |
| $\mathbf{w}_2 = \mathbf{x}(2)$ |
| $\mathbf{w}_3 = \mathbf{w}_3$ |

b) Analyze the resulting weights for case a) and elaborate on the final weight distribution with respect to the input vectors: which patterns ($\mathbf{x}(k)$) are represented by which weight vectors ($\mathbf{w}_i$)? [3 marks]

$\mathbf{w}_1$ - represents $\mathbf{x}(3)$

$\mathbf{w}_2$ - represents $\mathbf{x}(1)$ and $\mathbf{x}(2)$

$\mathbf{w}_3$ – does not represent any input training pattern (dead unit)

c) Consider that the sequence of training inputs was changed to $\mathbf{x}(3)$, $\mathbf{x}(1)$, $\mathbf{x}(2)$. Will the resulting set of weights be the same as in the case b) above? [2 marks]

| **Pattern presented** | **Winner ($\mathbf{w}_i$)** | **Winner updated to** |
|---|---|---|
| $\mathbf{x}(3)$: | $\mathbf{w}_1$ | $\mathbf{w}_1 = \mathbf{x}(3)$ |
| $\mathbf{x}(1)$: | $\mathbf{w}_2$ | $\mathbf{w}_2 = \mathbf{x}(1)$ |
| $\mathbf{x}(2)$: | $\mathbf{w}_2$ | $\mathbf{w}_2 = \mathbf{x}(2)$ |

| Summary of weights after training |
|---|
| $\mathbf{w}_1 = \mathbf{x}(3)$ |
| $\mathbf{w}_2 = \mathbf{x}(2)$ |
| $\mathbf{w}_3 = \mathbf{w}_3$ |

→ YES, weights are the same is in a)

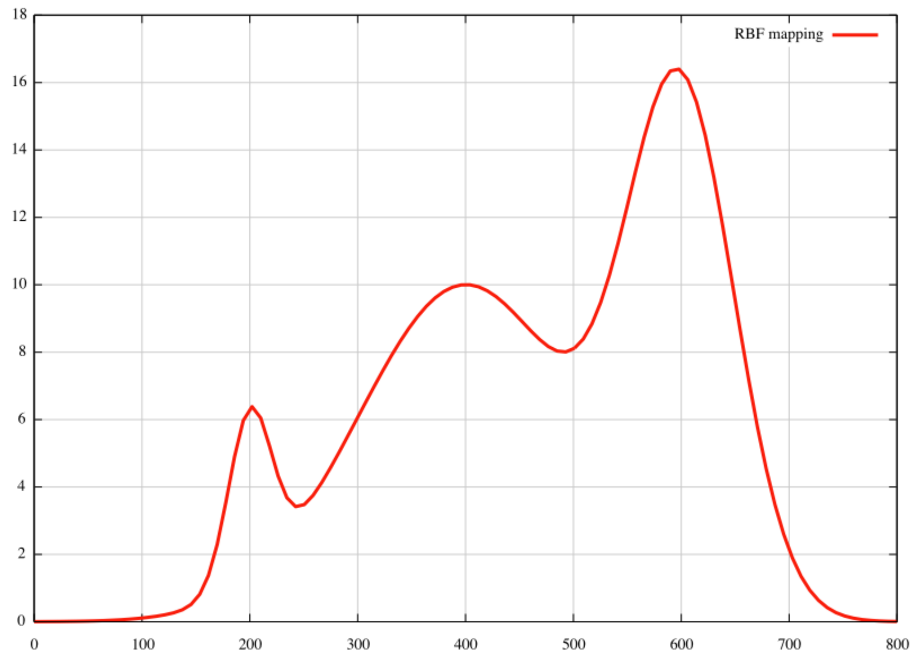
d) Consider that the sequence of training inputs was changed to $\mathbf{x}(3)$, $\mathbf{x}(2)$, $\mathbf{x}(1)$. Will the resulting set of weights be the same as in the case a) above? [2 marks]

| **Pattern presented** | **Winner ($\mathbf{w}_i$)** | **Winner updated to** |
|---|---|---|
| $\mathbf{x}(3)$: | $\mathbf{w}_1$ | $\mathbf{w}_1 = \mathbf{x}(3)$ |
| $\mathbf{x}(2)$: | $\mathbf{w}_2$ | $\mathbf{w}_2 = \mathbf{x}(2)$ |
| $\mathbf{x}(1)$: | $\mathbf{w}_2$ | $\mathbf{w}_2 = \mathbf{x}(1)$ |

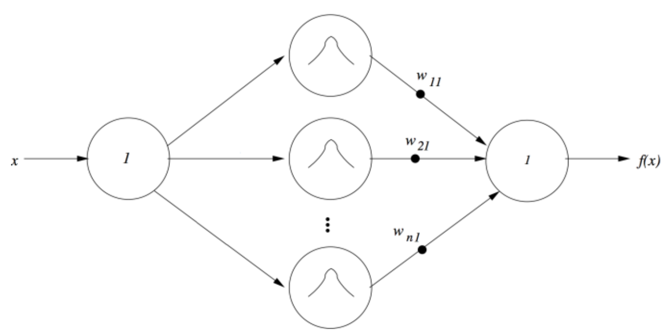
| Summary of weights after training |
|---|
| $\mathbf{w}_1 = \mathbf{x}(3)$ |
| $\mathbf{w}_2 = \mathbf{x}(1)$ |
| $\mathbf{w}_3 = \mathbf{w}_3$ |

→ NO, weights are not the same is in a)

*8.* Derive a Radial Basis Function (RBF) network that generates the following function:




The network will have the following architecture:




The activation function of the output neuron is a linear function with no bias:

$$f(x)=\sum_{i=1}^{n} w_{i1} o_i ,$$

where $n$ is the number of RBF nodes (determined by you), and $o_i$ denotes the output from the $i$-th neuron in the hidden layer. Assume all $n$ RBF nodes use Gaussian kernel function

$$o_i = e^{-\frac{\|x-m_i\|^2}{2s^2}}$$

(a) How many RBF nodes are best-suited to generate $f(x)$ ? Explain why. [2 marks]

There are three Gaussian components in $f(x)$, therefore three (3) nodes in the hidden layer are appropriate.

(b) Specify all parameters for the hidden layer nodes, i.e. centers of the basis functions $\mu_i$ and their standard deviations $\sigma_i$. The standard deviations can be expressed in terms of one-another. (e.g. $\sigma_i > \sigma_j > \sigma_k$). [4 marks]

Centers of the Gaussians can be read from the graph by identifying the peaks:

$\mu_1 = 200$, $\mu_2 = 400$, $\mu_3 = 600$

The standard deviations can be read by comparing widths of the Gaussians and ordered as follows:

$0 < \sigma_1 < \sigma_3 < \sigma_2$

(c) Derive approximate values of all weights ($w_{11}$, ..., $w_{n1}$). Note that although the weights are derived numerically in practice, graphical determination is possible here, based on the function plot. [4 marks]

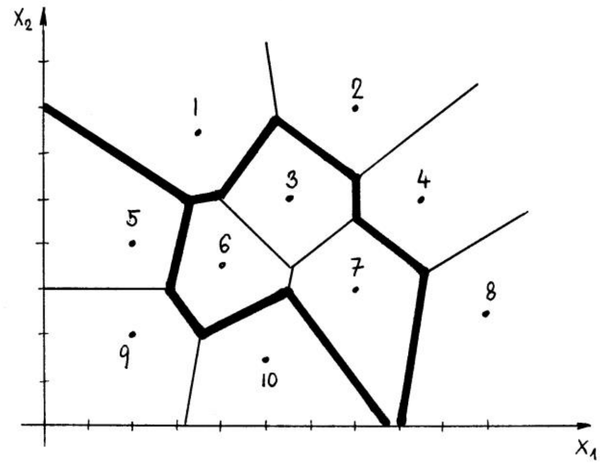
Approximate values of the weights can be read from the graph by identifying the heights of the Gaussians (they are only approximate due to potential interplay between the individual Gaussian components):

$w_1$ = 5-6

$w_2$ = 10

$w_3$ = 15-16

9. Consider a partition obtained by a competitive layer of a neural network, dividing the feature space into ten regions outlined in the figure below, and represented by ten prototype vectors labeled 1-10.




The ultimate goal of the neural network is to aggregate regions 1-10 to three larger areas indicated by the thick lines in the figure.

a) [6 marks] Sketch the structure of a neural network that could be used to implement both parts of this problem (i.e. the initial partition into 10 regions using competitive layer, and the final partition into 3 larger regions using vector quantization layer); label each layer to indicate what its purpose is.

2 input neurons (= dimension of the input space)

10 neurons in competitive layer (= number of small/simple regions)

3 neurons in LVQ layer (= number of larger regions)

b) [3 marks] Which learning rule can be used to implement the initial partition into ten convex regions (name the rule and provide its mathematical description).

Competitive learning.

$$Dw_j = \begin{cases} h\left(x - w_j\right) & \text{for} \quad j = k \\ 0 & \text{for} \quad j \neq k \end{cases}$$

**c)** [3 marks] Write down the weight matrix that can be used to aggregate the convex regions into the three larger regions shown in the figure.

$$\begin{array}{c|cccccccccc|}
\text{Region} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\
\text{Class 1} & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\
\text{Class 2} & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\
\text{Class 3} & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0
\end{array}$$

**V. Evolutionary Computing** [20 marks total]

*10.* [10 marks] Consider following population of $N$=6 individuals of a genetic algorithm

{[1, 1], [3, 2], [1, 6], [7, 4], [2, 3], [8, 6]}

Each individual represents a candidate solution in form [$x$, $y$] for a problem with integer parameters. The fitness function

$$f(x, y) = \max(|x-2|, |y-3|)$$

should be <u>minimized</u>.

a) [6 marks] Determine the intermediate population obtained using deterministic tournament method, assuming that the following groups of individuals have been formed for tournaments (the symbols represent individuals in the population listed above, starting with index "A" for the first individual on the left):

A-B, C-D, E-F, A-C, B-F, D-E

| Individual | x | y | f(x,y) |
|---|---|---|---|
| A | 1 | 1 | 2 |
| B | 3 | 2 | 1 |
| C | 1 | 6 | 3 |
| D | 7 | 4 | 5 |
| E | 2 | 3 | 0 |
| F | 8 | 6 | 6 |

| Tournament | | Winner | New [x,y] |
|---|---|---|---|
| A | B | B | [3,2] |
| C | D | C | [1,6] |
| E | F | E | [2,3] |
| A | C | A | [1,1] |
| B | F | B | [3,2] |
| D | E | E | [2,3] |

b) [4 marks] Comment on the composition of the intermediate population from the perspective of counts of individuals it contains and their fitness.

The individuals with high fitness (2 and 5) are represented by two copies each in the intermediate population, while the individuals with low fitness (4 and 6) are not present at all. This is a direct consequence of properly set up binary tournament process (each individual participates in exactly two tournaments).

*11.* [6 marks] Consider following two individuals of a genetic algorithm

[1.5, 0.1, 3.5, 2.2, 2.0, 5.5]
[1.5, 3.0, 4.2, 4.5, 3.5, 1.8]

Write down the resulting children obtained using

(a) Simple arithmetic recombination with parameters $k = 3$ and $\alpha = 0.6$

Child 1: [1.50, 0.10, 3.50, 3.12, 2.60, 4.02]

Child 2: [1.50, 3.00, 4.20, 3.58, 2.90, 3.28]

(b) Simple arithmetic recombination with parameters $k = 3$ and $\alpha = 0.6$

Child 1: [ see formula sheet ]

Child 2: [ see formula sheet ]

(c) Whole arithmetic recombination with parameters $k = 3$ and $\alpha = 0.6$

Child 1: [ see formula sheet ]

Child 2: [ see formula sheet ]

14. [4 marks] Consider the following schema: **`*1*110*`** Write down all individuals represented by this schema.

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 |

**ECE449 FINAL EXAM Fall 2013 – FORMULA SHEET**

<u>Learning rules:</u>

Perceptron: $Dw_i = hx_i(t-o)$

ADALINE: $Dw_i = hx_i(t-tot)$

Instar: $Dw_i = h(x_i - w_i)$

Outstar: $Dw_{ij} = h(t_j - w_{ij})$

Competitive: $k = \arg\min_i (\|x - w_i\|)\ ;\ Dw_{ik} = h(x_i - w_{ik})$

<u>Properties of fuzzy sets:</u>

Height: Hgt($A$) = $\sup_x A(x)$

Support: Supp($A$) = $\{x \in X \mid A(x) > 0\}$

Core: Core($A$) = $\{x \in X \mid A(x) = 1\}$

Cardinality: Card($A$) = $\Sigma_{x \in X} A(x)$

<u>Extension principle:</u>

$B(y) = \sup_{x=f^{-1}(y)} \min(A(x))$

<u>Fuzzy relations:</u>

Construction: $R(x,y) = A(x) \times B(y) = \min(A(x), B(y))$

Projections: $\mathrm{Proj}R_x(x) = \sup_{y \in Y} R(x,y)$, $\mathrm{Proj}R_y(y) = \sup_{x \in X} R(x,y)$

Cylindric extension: Cyl ($A$) ($x$,$y$) = A(x) for all $y \in Y$

<u>Implication operators:</u>

| Name | Implication operator $\Phi[A(x),B(y)]$ | ELSE connectives |
|---|---|---|
| Mamdani (min) | $\min[A(x),B(y)]$ | OR (**s**-norm) |
| Larsen (product) | $A(x) \cdot B(y)$ | OR (**s**-norm) |
| Zadeh (max-min) | $\max\{\min[A(x),B(y)],1-A(x)\}$ | AND (**t**-norm) |
| Lukasziewicz (arithmetic) | $\min[1-A(x)+B(y), 1]$ | AND (**t**-norm) |
| Boolean (Dienes-Rescher) | $\max[1-A(x), B(y)]$ | AND (**t**-norm) |

<u>Defuzzification operators:</u>

$$b = \mathrm{COA}(B(y)) = \frac{\int B(y)\,y\,dy}{\int B(y)\,dy}\ ; \quad b = SCOA(B(y)) = \frac{\sum B(y)\cdot y}{\sum B(y)}\ ;$$

$$b = MOM(B(y)) = \frac{\alpha + \beta}{2}$$