Departamento de Informática
Universidad Carlos III de Madrid

Grado en Ingeniería en Informática

Artificial Intelligence
Partial exam

April 2015

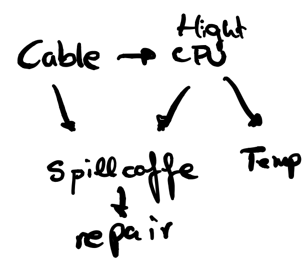
General indications

- Time assigned to the exam is **2 hours**
- You cannot leave the classroom during the exam, unless you have finished it
- Exams cannot be answered using a pencil

Exercise 1 (3p)

A coffee-serving robot is not working properly and has to be repaired. From the experience we know that a loose cable is a possible cause for loss of commands over the bus that controls the robot's arm. Also, it can induce a high CPU load of the robot's computer (because a certain task throws an exception and restarts constantly). In fact, either of these (loose of commands and high CPU load) could cause the arm to malfunction and spill coffee. Additionally, a high CPU load explains an increased temperature of the robot's computer.

1. Design a bayesian network to represent the variables and causal links defined in the text.
2. Define all necessary parameters to specify completely your network (invent the values).
3. Compute the probability of a loose cable after observing a spilled coffee.



Exercise 2 (4p)

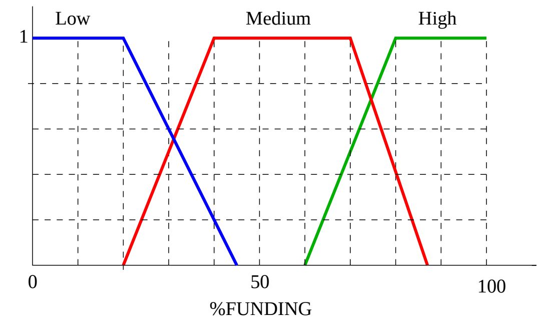
A recycling robot devotes two hours a day to collect the cans it finds in an environment. Periodically, it should take one of the following decisions: (1) search for a can; (2) wait for someone to bring it a can, or (3) go back to the base to recharge its battery. The best way to find cans is to search. Search provides more reward (more cans collected) than wait; and wait provides more reward than recharge. This is represented by action costs: the cost of searching is 5 while the cost of waiting is 10 and the cost of recharging is 20.

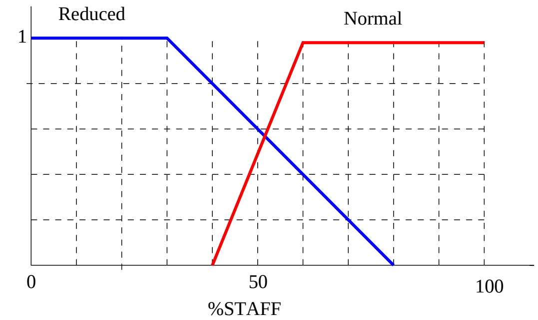
The decisions of the robot are only based on the energy level of the battery, that can be high, low or empty. Search runs down the battery, whereas waiting does not. When the energy level is high, after a period of searching, the energy level will be high with a probability of 0.6, otherwise the energy level will be low. When the energy is low, a period of searching leaves the battery low with probability 0.3 and exhausts the battery with the rest of probability. In the last case, a human should rescue the robot, and the only applicable action is *call human*. This action leads the robot to a state with the battery recharged but its cost is 100.

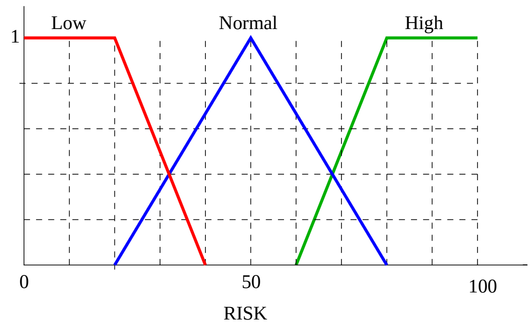
1. Define formally an MDP for the described problem.
2. Define the Bellmann equations for the states of your model.
3. Execute the first two iterations of value iteration.
4. Which is the policy after these two interations? Is that the optimal policy? Why?

Exercise 3 (3p)

The objective of a fuzzy rule-based system is to determine the risk of a software project. There are two inputs: percentage of project funding (in relation to other projects of the company) and the percentage of staff assigned to the project. Values for both inputs are described by fuzzy sets. The percentage of project funding can be low, medium or high. The precentage of staff can be reduced or normal. The degree of risk of the project, also defined by fuzzy sets, can be low, normal or high. The membership functions for the problem variables are:







The system includes the following rules:

R1 **IF** the project funding is high **OR** the staff is reduced **THEN** the risk is low

R2 **IF** the project funding is medium **AND** staff is normal **THEN** the risk is normal

R3 **IF** the project funding is low **THEN** the risk is high

The current project funding is 70 % and the assigned staff is 70 %. Execute the rule-based system (with Mamdani inference) to determine the risk of that project.

2.)

1. Estados { low L, high H, empty $\mathcal{E}$ }

Acciones { search S, wait W, recharge R, call human C }

Costes { $c(S) = 5$, $c(W) = 10$, $c(R) = 20$, $c(C) = 100$

Probabilidades :

$P_S(A/B)$

A \ B	H	L
H	0'6	0
L	0'4	0'3
$\mathcal{E}$	0	0'7

$P_W(X/X) = 1$

$X \in$ Estados

$P_R(H/X) = 1$

$P_C(H/\mathcal{E}) = 1$

2.

$$V_{t+1}(H) = \min_a [\,5 + 0'6 \cdot V_t(H) + 0'4\, V_t(L),\; 10 + 1\, V_t(H),\; 20 + 1\, V_t(H)\,]$$

$$V_{t+1}(L) = \min_a [\,5 + 0'3 \cdot V_t(L) + 0'7\, V_t(\mathcal{E});\; 10 + 1\, V_t(L),\; 20 + 1\, V_t(H)\,]$$

$$V_{t+1}(\mathcal{E}) = \min_a [\,10 + 1\, V_t(\mathcal{E}),\; 100 + 1\, V_t(H)\,]$$

3. $V_0(H)=0$ $V_0(L)=0$ $V_0(E)=0$

$V_1(H)=\min(5,10,20)=5$

$V_1(L)=\min(5,10,20)=5$

$V_1(E)=\min(10,100)=10$

$V_2(H)=\min(10,15,25)=10$

$V_2(L)=\min(13'5,15,25)=13'5$

$V_2(E)=\min(20,105)=20$

4. Politica

Para high es search

Para low es search

Para empty es wait

Para high y empty es la mejor opcion, ya que search cuando tiene toda la bateria proporciona muchas latas y no hay peligro de que se que tirado y para empty es mejor esperar ya que llamar es muy costoso y no recoge latas.

Sin embargo para low battery lo mejor no es buscar sino recargar, ya que hay muchas posibilidades de que no pueda continuar.

3. Project funding 70% PF
Assignedstaff 70% AS

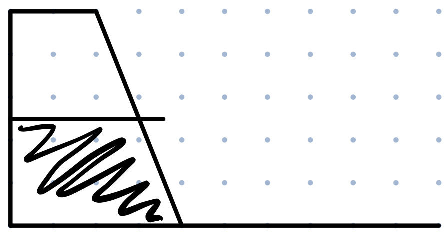
Borrosificar: Valor → Grado pertenencia

PF:
L = 0
M = 1
H = 0'5

AS:
R = 0'2
N = 1

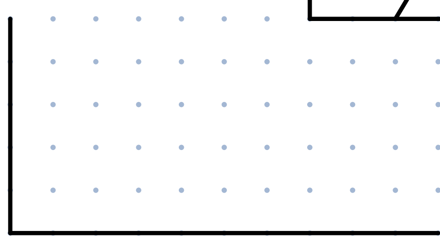
Analisis reglas:

R1: PF(H) or AS(R) ⇒ R(L)
max(0'5, 0'2) = 0'5

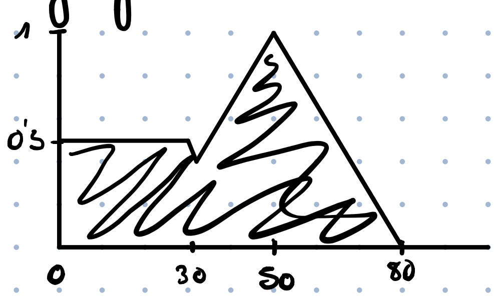


R2: PF(M) and AS(N) ⇒ R(N)
min(1, 1) = 1

R3: PF(L) ⇒ R(H)
$\overset{\shortparallel}{0}$



Agregación de consecuentes:



Desborrosificar: El min del maximo, que es 50.
Por centroide será aprox 45.