Departamento de Informática
Universidad Carlos III de Madrid

Grado en Ingeniería Informática

Artificial Intelligence

May 2014

General instructions

- Time assigned to the exam is **1.5 hours**
- Teachers will not answer any question about the exam
- You cannot leave the classroom during the exam, unless you have finished it
- Exams cannot be answered using a pencil

1 Questions on practice

1. Describe how to use the tool Genie to design a bayesian network.
2. Explain from your point of view how the game LuxDelux communicates with the bayesian networks designed by you (configuration, files, etc.)
3. In which moment/s of the game LuxDelux are your bayesian networks executed?
4. Describe your attack bayesian network. Then, explain: (a) what probability from this network is required by the game when your agent is playing; (b) how this probability determines the decision of executing an attack.
5. In your experiments against the minimal agent (IA2014_min), did your agent win in a percentage higher than 50%? What other experiments did you perform in your project?

2 Questions on theory

1. **(2 points)** In a production system, how do we name the set of instantiated rules that can be executed given a state of the working memory? What is the name of the step that computes this set in the execution cycle of production systems? What is the name of the step that decides which rule(s) to execute? Explain two/three alternatives for performing this last decision.
2. **(2 points)** What is the theoretical advantage of using heuristic search over uninformed search? What are the inputs to a heuristic search procedure in general? Describe an example.
3. **(2 points)** Explain what conditional probability stands for and one way we can use it when reasoning under uncertainty. Describe an example of its use.
4. **(2 points)** Explain modus ponens and how it is used in fuzzy logic.
5. **(2 points)** Present a potential architecture (as a figure) of a robotic application (main components and relations among them).

Solutions of Theory part.
May 2014

1. The answers are:

   (a) Conflict set

   (b) Matching

   (c) Conflict set resolution

   (d) Select first rule, select rule with more conditions, select rule according to priorities, use search to select a rule

2. Heuristic search can theoretically be more efficient than uniform search, since the use of a heuristic can guide the search technique towards the goal expanding a lesser number of nodes (less time to obtain solution). Note that it does not have to do with optimality (there are admissible uniform search techniques), nor with actions costs (again, there are uniform search techniques that deal with costs).

   The inputs are: a problem space (set of states, and set of actions), a problem (initial state and a description of the goal), and a heuristic function. In most cases, the problem space is programmed within the search procedure, so the inputs are reduced to just a problem and a heuristic function.

   An example could be a grid, where a robot is initially in a position with a key (initial state) and has to go to another position and get some tool from a closed chest, that can be opened with the key (goal description). A potential heuristic is Euclidean distance.

3. Conditional probability, whose notation is $P(A|B)$, defines the probability of some event A, given the knowledge on some other event B. It can be used to update the probability of A, once we know about the occurrence of B. Conditional probabilities are used in the Bayes Theorem, in the description of Bayesian Networks (when defining the conditional probability tables at nodes), when defining Markov processes as Bayesian Networks, or in Naïve Bayes for learning classifiers. An example of their use in Naïve Bayes consists on defining the class to which an example belongs as

$$c(X_i) = \arg\max_{C_k \in C} p(C_k) \prod_{j=1}^{a} p(A_j = V_{jv_j} \mid C_k)$$

4. Modus ponens is an inference rule in classical logic that allows us to derive the truth value of a formula given the truth value of another formula. It is widely used in logic, as well as in a generalized way in any programming language when using the IF-THEN structures. Formally it is defined as:

$$\frac{\begin{array}{l} p \\ p \rightarrow q \end{array}}{q}$$

   That is, if formula p is true and $p \rightarrow q$, then we can derive that formula q is true. Note that in production systems, $p \rightarrow q$ is usually written as `IF` p `THEN` q, where p and q can be arbitrary formulae.

   In the context of fuzzy logic, given that it handles fuzzy sets, the implementation of modus ponens is more complex. We could model it as (a formula x^f means that x is a fuzzy formula):

$$\frac{\begin{array}{l} p^f \\ s^f \rightarrow q^f \end{array}}{r^f}$$

   So, r^f is not exactly q^f, given that it is modified by the degree of matching between p^f and s^f. The usual procedure followed in fuzzy logic is the one defined by Mamdani fuzzy inference.

5. A possible architecture (as defined in the slides in robotics) is shown in Figure 1.

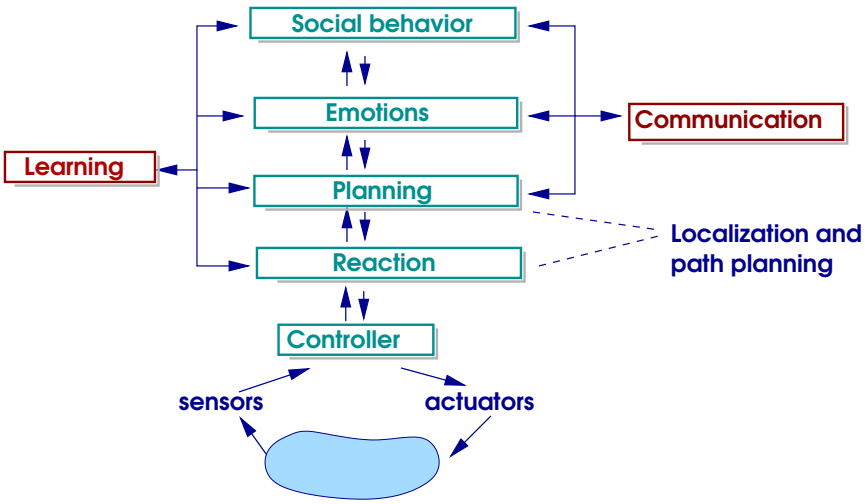

Figure 1: A robotic architecture.