ALEKSANDER SKRAASTAD & FREDRIK B. TØRNVALL

EXERCISE 1

TDT4136 - Intoduction to AI

August 2014

NTNU

Norges teknisk-naturvitenskapelige universitet

Table of contents

Chapter 1

Theoretical Questions

1.1 Question 1

Turing test is to check if a machine is able to exhibit intelligent behavior equivalent to, or indistinguishable from a human. The way it was shown in class is that we have a test person on one side of a wall and a machine or another person on the other side. The goal of the test is for the machine to convince the test person that he/she is talking to another human.

1.2 Question 2

- Rational behavior: doing the right thing.
- Thinking rationally: always yields correct conclusion when given correct premises.

The relationship between thinking rationally and acting rationally is that by coming to the right conclusion one will do the right thing in a given situation. Rational thinking is not an absolute condition for acting rationally, because rational actions do not necessary involve rational thinking. An example of this could be reflexes such as blinking.

1.3 Question 3

Tarskis theory of reference is an explanation of how to relate objects in the real world to objects in a logic statement.

1.4 Question 4

Rationality is defined in the dictionary as; the quality or state of being reasonable, based on facts or reason. A rational decision is one that is not just reasoned, but also optimal for achieving a goal or solving a problem.

Rationality as described in the book depends on four things:

- The performance measure that defines the criterion of success
- The agent´s prior knowledge of the environment
- The actions that the agent can performance
- The agent´s percept sequence to date

This leads to a definition of a rational agent: For each possible percept sequence, a rational agent should select an action that is expected to maximize its performance measure, given the evidence provided by the percept sequence and whatever built-in knowledge the agent has.

1.5 Question 5

1.5.1 Part A

The robot has no possible way of detecting the above threat, thus the robot is still rational in its own world, according to the definition of a rational agent.

1.5.2 Part B

There exists a sequence of actions that would give a better performance measure than the one described in this scenario. Obviously, the performance measure is crossing the road unscathed. We also assume that this is a simple reflex agent, as it would not make sense to incorporate previous traffic percepts that could trigger the robot to cross the road based on times of no traffic, without checking if there is a car coming. Either the robot has not used the look-left-look-right action to update its perception of the environment, or it has so, but proceeded to cross the road anyway, instead of waiting, thus failing to maximize the performance measure. Due to this, we state that the robot is not rational.

1.6 Question 6

1.6.1 Part A

The agent has no knowledge of the state of the square it is not in, therefor it can not act rationally. If it moves the agent is penalized. A simple reflex agent will therefore move between square A and B indefinitely and rank up minus points or not clean the floor at all.

1.6.2 Part B

The environment described on page 38 states that the cleaned squares stay cleaned thus the agent move from the starting square to the next one e.g A to B and stay in square until the 'life-time' has passed. This is rational behavior.

1.6.3 Part C

In this case the environment is perfectly observable thus the agent can act rationally. The reason for this is that the agent now knows when a square is dirty and only moves when it has to. This will minimize the number of penalizing points.

```
function vacuum-cleaner(percept) returns an action
    persistent: rules = (
        Dirty -> Suck
        Clean -> Move
    )

    state <- interpret-input(percept)
    rule <- rule-match(state, rules)
    action <- rule.action

    return action

function rule-match(state, rules) returns rule
    if Dirty not in state return No-op
    else
```

```
return rules[state.position]
```

The environment is perfectly observable. The agent will move to the square that needs cleaning. If all squares is clean the agent will not move.

1.7 Question 7

- Partially observable: the agent can only observe the state of the square it is in.
- Single agent: only one agent in the environment.
- Deterministic: the environment given in fig. 2.3 can be categorized as deterministic because the actions of the agent is determined by the current state and the action executed by the agent.
- Episodic: each action the agent performs is based on a single perception. The agent does not think ahead.
- Static: the square is dirty or it is clean. The environment is not dynamic. Only the agent will alter the environment with its actions.
- Discrete: the agent is either in square A or B.
- Known: the outcome for an action is knowns. Move to square and clean.

1.8 Question 8

1.8.1 Part A

A simple reflex agent is generally much smaller, and requires less processing power to operate. This makes it possibly quite efficient and fast. A drawback to the reflex agent is that is has no prior knowledge of percepts. This requires the environment to be fully observable to guarantee a best possible action. If there is unobservability in the environment, then prior knowledge of percepts would be needed to give the agent the possibility to reason better about what action to perform.

1.8.2 Part B

A model-based reflex agent has the advantage of keeping track of previous states. This allows the agent to use this information to determine the best possible action. Nonetheless, in a partially observable environment, the agent cannot guarantee an exact representation of the environment. It can only make a best possible guess, based on its model and previous states of the environment.

1.8.3 Part C

A goal-based agent implements the notion of .. yes, goals. This allows us to make the agent try to choose the best possible actions that would lead to reaching a goal. This allows for much flexibility, as we can define a goal, and the agent will use a model similar to the model-based reflex agents to choose which actions will lead to reaching one or more goals. The obvious advantage here is flexibility. The flexibility does however lead to drawbacks like processing time, efficiency and added complexity. Additionaly, goal based agents only really have a notion of "Goal reached" or "Goal not reached".

1.8.4 Part D

A utility-based agent expands the added flexibility of a goal-based agent. It is an internalization of performance measures, leading us to let the agent performance measure itself. There may be several ways to rome, but which road is the shortest, fastest, most safe etc. Goal-based agents therefore allows us to have a much more self-sustained agent, able to analyze its own performance and choose the best possible course of action. This again leads to even more complexity and computation time. And in the end, one cannot guarantee that agents like this will perform perfectly. It still only maximizes the expected utility based on the information it already has when uncertainty is involved.