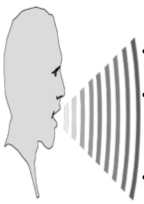
EEL5840: Elements of Machine Intelligence

MILECE

Announcements



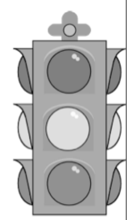
- Today's Handouts:
  - > Outline Class 3
- Web Site
  - > www.mil.ufl.edu/5840
  - > Software and Notes
  - > Programming Assignment Format
- Reading Assignment:
  - > Nilsson Chapter 2 & 3
  - > LISP Chapters 1-4
- Written Assignment
  - > Homework 1 Exercises 2.1-2.6 Due Tue. 9/1/09 in class

EEL5840: Elements of Machine Intelligence

MILECE

Today's Menu

- Stimulus-Response (*SR*) Agents
- Example of a Classical AI Production System
  - > Irrevocable Control Strategy
  - > Tentative Control Strategy



EEL5840: Elements of Machine Intelligence

MILECE

Stimulus-Response Agents

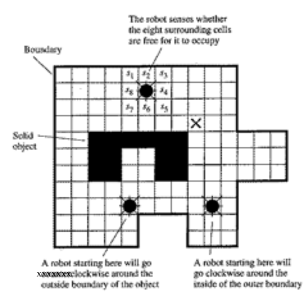
- S-R (stimulus-response) Agents—machines without an internal state representation that simply react to immediate stimuli in their world.
- At MIL and IMDL (EEL-5666) we have built a variety of simple robots that exhibit surprisingly interesting behavior.
- The first example in the text is shown in Figure 2.1 which is reproduced in the next slide.
- We want this robot to go to a cell adjacent to a wall or boundary then follow that boundary along its perimeter forever.
- The robot must sense whether certain tiles are free to occupy & perform certain primitive actions.

EEL5840: Elements of Machine Intelligence

MILECE

Sensory Inputs:
The binary vector
$s = \{s_1, s_2, \ldots, s_8\}$

Robot Actions:
1. North - up
2. East - right
3. South - down
4. West - left

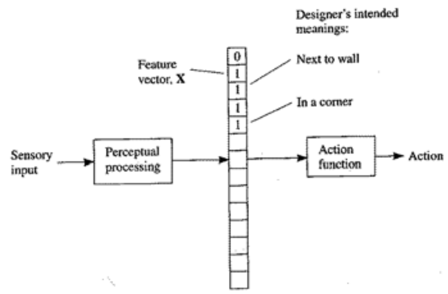


$x = \{0,0,0,0,0,0,1,0\}$

EEL5840: Elements of Machine Intelligence

MILECE

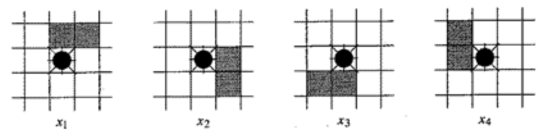
The designer's job is to specify a function of the sensory vector *s* that selects the appropriate action.



EEL5840: Elements of Machine Intelligence

MILECE

- The perceptual processing phase produces a feature vector, *x*, of features $\{x_1, x_2, \ldots, x_n\}$
- Deciding how to split the problem into its perception and action is "an art"



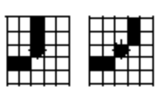
In each diagram, the indicated feature has value 1 if and only if at least one of the shaded cells is *not* free.

$x_1 = s_2 + s_3$ $x_2 = s_4 + s_5$ $x_3 = s_6 + s_7$ $x_4 = s_8 + s_1$

EEL5840: Elements of Machine Intelligence

MILECE

- The grid world has no tight spaces.
  The "no-tight-space" condition, intuitively, attempts to rule out configurations, such as the ones shown below, in which the robot would be confused about which boundary to follow:

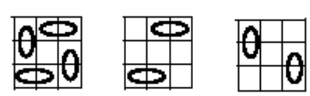


Since there are many such confusing configurations, the condition is difficult to state succinctly. We note that the robot perceives the world only through the values of the features $\{x_1, x_2, x_3, x_4\}$. Thus it suffices to define the no-tight-space condition in terms of those features alone.

EEL5840: Elements of Machine Intelligence

MILECE

Let us define the conditions diagrammatically as follows. We rule out any configurations in which each of the sets of cells marked by ellipses in the diagrams below have one or more cells occupied and in which the other cells are empty.



That is, we rule out any configurations for which the Boolean function $x_1x_2x_3x_4 + x_1x_3/x_2/x_4 + x_2x_4/x_1/x_3 = 1$.

EEL5840: Elements of Machine Intelligence

MILECE

Perception and Action

- Perception
  - > Convert *s* into *x*
  - > The possible number of inputs is 2^8 or 256
  - > Features were specified in a previous slide, e.g., x_1=1 iff s_2=1 or s_3=1 *{this is correct! Nilsson's Definition: If at least one of the shaded tiles is not empty}* $x_1=s_2+s_3$
  - > the processing involved in this example is very simple and is not characteristic of real robots which yield incomplete data and ambiguous readings.
- Action
  - > The specification of a function that selects the appropriate boundary-following action
  - > Default to a given direction if none of the inputs s_i=1, e.g., north. Analogous to "go straight unless {IR sensor value > threshold}"
  - > Boundary follow:
    - if x_1=1 and x_2=0 go right (east)
    - if x_2=1 and x_3=0 go down (south)
    - if x_3=1 and x_4=0 go left (west)
    - if x_4=1 and x_1=0 go up (north)

EEL5840: Elements of Machine Intelligence

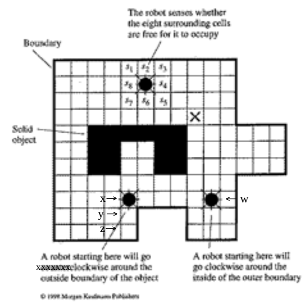
MILECE

Recall that $x_1=s_2+s_3$ $x_2=s_4+s_5$ $x_3=s_6+s_7$ $x_4=s_8+s_1$

if x_1=1 and x_2=0 go right; if x_2=1 and x_3=0 go down

if x_3=1 and x_4=0 go left; if x_4=1 and x_1=0 go up

Thus, at point x, S_x = {0,0,0,0,1,0,0,0} therefore x_1=0 x_2=1 x_3=0 x_4=0 the robot will go south to point y S_y = {0,0,0,1,1,0,0,0} and x_1=0 x_2=1 x_3=0 x_4=0 the robot will go south to point z S_z= {0,0,1,1,1,1,0,0} x_1=1 x_2=1 x_3=1 x_4=0 the robot will go west to the left of z S_{tlz}= {0,0,0,0,1,1,1,0} x_1=1 x_2=1 x_3=1 x_4=0 the robot goes west, i.e., clockwise around the boundary



At point w, S_w={0,0,0,0,0,0,0,0} therefore x_1=0 x_2=0 x_3=0 x_4=0 the robot will go north to the boundary where S_B={1,1,1,0,0,0,0,0} and x_1=1 x_2=0 x_3=0 x_4=1 the robot turns right and goes clockwise around the boundary

EEL5840: Elements of Machine Intelligence

MILECE

Perception and Action

- Boolean Algebra: I will assume you already know or can learn the principles of Boolean Algebra as per EEL-3701.
- Classes and Forms of Boolean Functions
  - > One form to write Boolean functions is as a conjunction of literals or a monomial, e.g., $\lambda_1\lambda_2...\lambda_n$, where λ_i are the literals. The conjunction itself is called a term.
  - > A clause is any expression of the form $\lambda_1+\lambda_2+...+\lambda_n$, where λ_i are the literals. This form is called a disjunction of literals.
  - > A term is the dual of a clause and vice-versa by DeMorgan's law.
  - > A function is said to be in *Disjunctive Normal Form* (DNF) if it can be written as a disjunction of terms.
  - > A function is said to be in *Conjunctive Normal Form* (CNF) if it can be written as a conjunction of clauses.

EEL5840: Elements of Machine Intelligence

MILECE

Perception and Action

Representing and Implementing Action Functions

In order to compute one of R possible actions we need an appropriate R-valued function of the feature vector.

- Production Systems
  - > An ordered set of rules (called productions) of the form $condition_i \rightarrow action_i$ or $c_i \rightarrow a_i$
  - > To process the rules, we start with the i=1 rule and look for the first rule whose *condition* part evaluates to 1 and then we select and execute the *action* part of that rule.
  - > $action_i$ can be either a primitive action, a call to another production system, or a set of actions to be executed simultaneously.
  - > The last rule in the set usually has 1 as its condition yielding a default action

EEL5840: Elements of Machine Intelligence

MILECE

Perception and Action

A possible production system representation for the boundary following routine is:

b-f: $x_4/x_1 \rightarrow$ north
$x_3/x_4 \rightarrow$ west
$x_2/x_3 \rightarrow$ south
$x_1/x_2 \rightarrow$ east
1 $\rightarrow$ north

A goal condition is specified by an action of *nil* on the first rule of the set and c_1 specifies the overall goal that we want the action program to achieve. For example:

corner-detect $\rightarrow$ *nil*
1 $\rightarrow$ *b-f* will get the robot to a corner

EEL5840: Elements of Machine Intelligence

MILECE

The Classical AI Paradigm

- In AI we use the following paradigm for problem-solving: (1) a global database (DATA), (2) production rules (OPERATIONS), and (3) a control strategy (CONTROL).
  The FIRST and most important step is to choose an APPROPRIATE LEVEL OF DESCRIPTION.
- Production Systems Definition
  - > global DB is a central, globally accessible data structure
    - – New knowledge can be added without reprogramming entire system
    - – Production systems are modular
  - > rules in the form of antecedent consequent
    - – all rules have access to the DB
    - – rules do not call one another
    - – all communication is through the DB
  - > control strategy chooses among the rules & ceases when the termination condition is satisfied

EEL5840: Elements of Machine Intelligence

MILECE

The Classical AI Paradigm

- A. Problem Representation
  - > To transform the problem into the three components of a production system, mainly, (1) problem states (Global DB), (2) the legal moves (rules) and goal and (3) selection of a suitable strategy
- B. General Programming Procedure
  1. DATA ← initial database
  2. until DATA satisfies the termination condition
  3. {select some rule R that applies
  4. DATA ← Apply R to DATA
  5. } end

EEL5840: Elements of Machine Intelligence

MILECE

The Classical AI Paradigm

- C. Control Strategy
  1. Irrevocable
  2. Tentative

- Another Production Systems Example
  1. The 8-puzzle

2 8 3		1 2 3
1 4 5	==>	8 _ 4
7 6 _		7 6 5

EEL5840: Elements of Machine Intelligence

MILECE

Production Systems Example

- The solution to an AI problem is to find a sequence of moves that transforms the initial state into a goal state. The set of all moves applied is called the state space of the problem
- We are often interested in solutions having minimum moves. Thus, a cost function is attached to each move so we can find solutions with minimal cost.
- The rules for the 8-puzzle can be summarized as follows: move the blank tile Right, Left, Up or Down

EEL5840: Elements of Machine Intelligence

MILECE

Production Systems Example

- Since the "selection of rules" is not specified in the algorithm, a production system is non-deterministic
- The control strategy selects the rule, keeps track of which rule was applied and modifies the DB
- Available information precludes selecting the best rule at each step. Thus, control strategies take the form of a search process; try the rules until a sequence is found that satisfies the goal condition
- Efficient control strategies require knowledge about the problem domain so that the rule selected is as optimum as possible.

EEL5840: Elements of Machine Intelligence

MILECE

Production Systems Example

- An Irrevocable Control Strategy
  - > Suppose we use a gradient search method and irrevocably select the rule that produces a DB with the largest increase in the value of a function
- Let f (DB)=$-\sum$(number of tiles out of place)
- f (initial DB) = -4 and f (goal DB) = 0
- If none of the applicable rules permits an increase in the value of the function (i.e., in case of ties), we select a rule based on an arbitrary scheme, say, {L,U,R,D}. If there are no applicable rules, the process halts.

EEL5840: Elements of Machine Intelligence

MILECE

Irrevocable Control Strategy

```
 (-4)              (-3)           (-3)
 2 8 3             2 8 3          2 _ 3
 1 4 _     ==>     1 _ 4 ==>      1 8 4
 7 6 5             7 6 5          7 6 5

       We must choose between
       left and up because there
       is a tie. Choose up arbitrarily      ↓

 1 2 3 (0)         1 2 3 (-1)     _ 2 3 (-2)
 8 _ 4     <==     _ 8 4 <==      1 8 4
 7 6 5             7 6 5          7 6 5
```

EEL5840: Elements of Machine Intelligence

MILECE

Tentative Control Strategy - BACKTRACKING

```
 ①               ②              ③
2 8 3           2 8 3          2 8 3
1 6 4    ==>    1 6 4 ==>      _ 6 4
7 _ 5           _ 7 5          1 7 5

_ 8 3 ④         8 _ 3 ⑤        _ 8 3 ⑥
2 6 4    ==>    2 6 4 ==>      2 6 4
1 7 5           1 7 5          1 7 5
```

Here the configuration
is identical to No. 4,
∴ backtrack

EEL5840: Elements of Machine Intelligence

MILECE

Tentative Control Strategy - BACKTRACKING

```
 ⑤               ⑥              ⑦
8 _ 3           8 3 _          8 _ 3
2 6 4    ==>    2 6 4 ==>      2 6 4
1 7 5           1 7 5          1 7 5

8 3 _ ⑥         8 3 4 ⑦        8 _ 3 ⑤
2 6 4    ==>    2 6 _ ==>      2 6 4
1 7 5           1 7 5          1 7 5
```

Here the
configuration
is identical to No. 5,
∴ backtrack

Here we have
applied ≥ 6 rules
∴backtrack

EEL5840: Elements of Machine Intelligence

MILECE

The End!