Perception

Gal A. Kaminka

Representing Knowledge

Reminder: The basic agent algorithm

```
1  W is knowledge base
2  g is an unsatisfied goal in W
3  B set of actions available
4  P set of percepts available
5
6  while g not satisfied:
7     PERCEIVE() using percepts in P to update W
8     CHOOSE() action b (from B) that advances towards g
9     EXECUTE() action b
```

- Knowledge base W starts with initial knowledge $\Leftarrow$ **the focus**
- PERCEIVE() adds knowledge

Knowledge Representation

- Whole area of AI devoted to knowledge representation
- Commonly known as **KR**
- Own conferences
  - KRR, knowledge representation and reasoning
- Lots of thought on how to reason about knowledge
  - Look up *ontology*, *description logics*, etc.

Simple KR used *here* and *now*

- Keep track of *beliefs*
  - We will discuss procedures later
- Each belief: tuple $\langle k, v \rangle$
  - k is key (can have variable parameters)
  - v is a value

Fluents

- In many areas of AI, logic is dominant representation approach
- *Fluents* used as representation (non-procedural knowledge)
- fluent $f = v$:
  - belief with key f has value v
- This looks suspiciously like predicates... But no.
  - fluents can change values

Fluent name and parameters

- A fluent $f(e_1, ..., e_n)$:
  - f is fluent name
  - n is the fluent arity (including 0, $f()$)
  - e_i entities in a known set E_i, *or* variables
- Ground fluents: no variables, e_i constants

Fluent values

- Fluents can have values from a range V_f
- *fluent literal*: $f(...) = v$, where $v \in V_f$
- Value can be simple (true/false, number)
- Value can be complex (data structure)

Examples

- robot_at_cell_x() = *true*

Examples

- robot_at_cell_x() = *true*
- robot_at(x) = *true*

Examples

- robot_at_cell_x() = *true*
- robot_at(x) = *true*
- robot_at() = x

Examples

- robot_at_cell_x() = *true*
- robot_at(x) = *true*
- robot_at() = x
- at(r,x) = *true*

Examples

- $\mathsf{robot\_at\_cell\_x}() = \mathit{true}$
- $\mathsf{robot\_at}(x) = \mathit{true}$
- $\mathsf{robot\_at}() = x$
- $\mathsf{at}(r,x) = \mathit{true}$
- $\mathsf{at}(r) = x$

Using fluents to represent environment states

- Given set of Fluents F
- Environment: collection of states
- State s is set of fluent literals
  - Each assigned a value (i.e., is *literal*)
  - *Consistent*: no $f = v_1, f = v_2$, s.t. $f \in F, v_1 \neq v_2$
  - *Maximal*: $\forall f \in F, (f = v) \in s$, i.e., $|s| = |F|$
- Partial state – same def, but not maximal
  - Really, a collection of states that share fluent literals.
- This is a **factored state representation**
  - State is composed of multiple factors

Perception in the Agent

Reminder: The basic agent algorithm

```
1  W is knowledge base
2  g is an unsatisfied goal in W
3  B set of actions available
4  P set of percepts available
5
6  while g not satisfied:
7     PERCEIVE() using percepts in P to update W
8     CHOOSE() action b (from B) that advances towards g
9     EXECUTE() action b
```

- Knowledge base W: collection of fluents
- PERCEIVE() adds knowledge $\Leftarrow$ **the focus**

Perception

Naive view

- Update the knowledge-base available to the agent
  - i.e., Transform environment features to knowledge-base
- Information from sensors *overwrites* knowledge-base
- Examples:
  - Current range straight ahead
  - Current time
  - Current location
  - Current map
  - Current objects seen

Perception

Naive view

- Update the knowledge-base available to the agent
  - i.e., Transform environment features to knowledge-base
- Information from sensors *overwrites* knowledge-base
- Examples:
  - Current range straight ahead
  - Current time
  - Current location
  - Current map
  - Current objects seen

What's so complicated?

Perception computationally complex

- Extracting information from current readings
  - e.g., “Current objects seen” requires computer vision
- May also require multiple readings over time
  - “Current map”, “current location” requires SLAM
  - Expensive, and requires memory

Perception conceptually complex: Body

- Requires interpretation (sensors never perfect)
  - At best, have a model of the sensor
- Perception takes time (e.g., for computation)
- Integrating information from multiple sensors
  - **Sensor Fusion**: whole field!
  - e.g., "obstacle near" if one of many sensors active

Will come back to this later.

Perception conceptually complex: Environment

- *Transparent* environment: everything perceivable.
  - May require **attention process**
- *Translucent* environment: Not everything perceivable
  - May require **active perception**

Will come back to this later.

Impact of embodiment on agent algorithm

Perception conceptually complex: Body

- Requires interpretation (sensors never perfect)
  - At best, have a model of the sensor
- Perception takes time (e.g., for computation)
- Integrating information from multiple sensors
  - **Sensor Fusion**: whole field!
  - e.g., "obstacle near" if one of many sensors active

~~Will come back to this later~~. Now.

Integrating and interpreting information from sensors

- From multiple sensors
- From multiple readings (over time)
- Conceptually PERCEIVE() into (at least) two processes:
  - $K \leftarrow \mathit{UPDATE}(W)$ which generates new beliefs K (set)
  - $W_{new} \leftarrow \mathit{REVISE}(K, W)$ which integrates new beliefs with old
- Separates the act of sensing from the integration over time
- Reasoning using sensor models, etc.: in REVISE()

Synchronous perception (Original)

```
1  W is knowledge base
2  g is an unsatisfied goal in W
3  B set of actions available
4  P set of percepts available
5
6  while g not satisfied:
7     PERCEIVE() using percepts in P to update W
8     CHOOSE() action b (from B) that advances towards g
9     EXECUTE() action b
```

- Single perception step after single action step
- OK if **actuation, perception are instantaneous**
- What happens when they are not?

Semi-synchronous algorithm

```
1  W is knowledge base
2  g is an unsatisfied goal in W
3  B set of actions available
4  P set of percepts available
5
6  while g not satisfied:
7     CHOOSE() action b (from B) that advances towards g
8     In parallel:
9        EXECUTE() action b
10       PERCEIVE() using percepts in P to update W
```

- Perception step in parallel to action step
- Now action, perception can have duration
- Still needs both to finish before can continue

Asynchronous algorithm

```
1  W is knowledge base
2  g is an unsatisfied goal in W
3  B set of actions available
4  P set of percepts available
5
6  In parallel:
7    while g not satisfied:
8      CHOOSE() action b (from B) that advances towards g
9      EXECUTE() action b
10   while (true) PERCEIVE() using percepts in P to update W
```

- *Perception process* in parallel to *action selection process*
- Completely independent?
- **No**, will need write-locks accessing W

Impact of environment on agent algorithm

Perception conceptually complex: Environment

- *Transparent* environment: everything perceivable.
  - May require **attention process**
- *Translucent* environment: Not everything perceivable
  - May require **active perception**

~~Will come back to this later.~~ Now

Attention, Active Perception

- How will these algorithms change to allow:
  - attention? (deciding which percepts to update)
  - active perception? (choosing actions that will enable perception?)
- Attention takes place at the UPDATE() stage
  - UPDATE(W) takes knowledge into account. But how?
- Active Perception is even bigger challenge:
  - Take actions to generate new sensing possibilities
  - Takes place in CHOOSE()!

Open areas of research!