Intelligent Systems: Reasoning and Recognition

James L. Crowley

Ensimag 2 Second Semester 2017/2018

Lesson 10 16 March 2018

Knowledge, Planning and Problem Solving

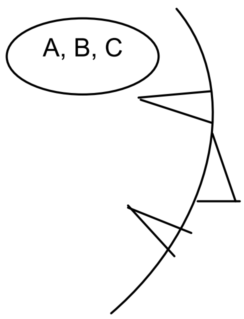
Knowledge and Meta-Knowledge

Intelligence is a description of the interaction between an agent and an environment. This was the basis of the Turing Test (social interaction that could not be distinguished from human).

For Turing, human-level performance at social interaction provides a standard for considering a system to be capable of intelligent interaction. In many domains, human-level performance at interaction is considered the standard for whether we consider a system to be intelligent. The subject of this course is techniques to build systems with human-level performance (or better).

The Intelligent Agent

In 1970s, Nils Nilsson (and others) proposed the Intelligent Agent as a fundamental concept for formalizing intelligence.



The Intelligent Agent is an abstract concept composed of 3 pars: (A, B, C)
A) Actions: The ability to act; The ability to change the state of the universe.
B) Goals. Desired states.
C) Knowledge; The ability to choose actions to accomplish goals.

Nilsson proposed that to be considered as intelligent, three things are required:
1) The agent must be able to act; able to change the state of the universe (embodied)
2) The agent must have goals, (desired states of the universe).
3) The agent must be able to choose its actions to accomplish goals (be rational).

This was formulated as the principal of Rationality, and widely used in Economics.

The "Principal of Rationality:

The intelligent agent is assumed to be rational.
The intelligent agent chooses its actions to accomplish its goals.

Unfortunately, most human behavior is not rational.

Knowledge is what enables the intelligent agent to interact.

Knowledge = Competence = Cognitive Abilities.
    The ability to act to bring the world to a desired state.

There are many forms of cognitive abilities. In the next 9 lectures we will look a many (but not all).

Meta-Knowledge

Meta Knowledge: The ability to acquire new knowledge. For example:
- Learning (there are MANY forms of learning).
- Practice (repetition).
- Communication (acquiring knowledge from others)
- Planning

Meta-knowledge concerns techniques that provide new cognitive abilities.

Theories about human knowledge have been developed in Cognitive Psychology. Much progress has been made over the last 50 years.

Cognitive Psychology tells us what to compute. Informatics (including neural networks) tell us how to compute. You can see Cognitive Psychology as providing a roadmap for research on Intelligence.

Kinds of Knowledge (categories of cognitive abilities).

Up to now, the techniques we have studied have concerned recognition.
The primary applications are for Visual Recognition and Auditory Recognition.
This is a form of Perceptual knowledge.

Perception: Abilities to construct an internal description (model) of the environment.

Humans use visual, auditory, tactile, olfactory, and gustative senses to detect, locate and recognize phenomena in order to compose a description of the environment.
We also use proprioception (feedback from muscles) to measure how the environment constrains motion.

Conceptual Knowledge: Recognized phenomena are represented as concepts.

Concepts are the fundamental building blocks of cognition.
Concepts are created as abstractions or generalizations from experience;
A concept is instantiated (Reified) by memories of sensory experiences and associated with possible actions (affordances) or dangers.

Concepts are abstract mental representations that associate perceived phenomena with actions, ideas and experience. Collections of concepts can be assembled into sets and represented as a new concept. This is called chunking.

Essential elements for a concept include:
1) A symbolic name
2) Recognition abilities for the concept.
3) Memory of sensorial experiences.
4) Actions that are enabled or inhibited.

In the coming lectures we will explore techniques to represent and reason with concepts.

Action: The ability to change the state of the environment.

Actions can sometimes be preplanned, but most human action “situated”.
That is it is determined dynamically based on the current state of the world, as provided by perception.
Forms of action knowledge include manipulation, navigation and communication.
(communicating through gesture or motion).

Tightly coupled perception and action provide "Operational" knowledge.

Operational Knowledge: Situated interaction using perception and action

The Subsumption Principal

Complex knowledge (cognitive abilities) subsume (include, builds on)
more primitive forms of knowledge.

We say that operational knowledge subsumes perception and action knowledge.

Operational knowledge is necessarily situated. It is determined from the perceived state of the environment. It cannot (generally) be pre-determined.

Procedural knowledge: a series of operations to bring the environment to a desired state.

Procedural knowledge subsumes Operational knowledge. You must know HOW to perform each operation in the procedure.

Today we will look at planning as a means to generate operational knowledge and procedural knowledge.

Planning

Problem Spaces

Planning is formalized using a Problem space.

A problem space is defined as
1) A set of states {U} (the Universe),
2) A set of operators for changing states {A} (Operations or Actions).

A state is defined using a conjunction of predicates.

A problem is {U}, {A} plus
an initial state $i \in \{U\}$
a set of Goal States $\{G\} \subset \{U\}$

A plan creates a sequence of actions A_1, A_2, A_3, A_4,... that lead from the state S to one of the states $g \in \{G\}$

With this approach, problem solving is formalized as Planning: The search for a sequence of actions leading to a goal.

The core concept is the notion of "State".
A state is a "partial" description of the environment represented as a conjunction of predicates.

State: A conjunction of predicates (truth-valued functions) over entities.

A state is defined as a conjunction of predicates whose arguments are entities.
Entities represent perceived phenomena.

Predicates express relations between entities (spatial, temporal, part-of, category inclusion, etc) or properties of entities.
Predicates can be negated. The negation of a predicate is a predicate.

Blocks World

Blocks world is an abstract, toy world for exploring problems of reasoning and intelligence. We will use Blocks world to illustrate different principals and techniques concerning knowledge representation.

Blocks world is composed of:

- A table
- A set of blocks
- An agent (robot hand) that can act on (move) the blocks

The blocks, tables and hand are the primitive concepts that make up a blocks world. They are primitive because they are directly perceivable.

Classic Definition:
1) A universe composed of a set of cubic blocks and a table
2) Blocks are mobile, the table is immobile
3) The agent is a mobile hand,
4) A block can sit on a table, on another block, or in the hand.
5) There cannot be more than one block on another block
6) The table is large enough for all blocks to be on the table.
7) The hand can move only one block at a time.

The state of the universe is formalized using predicates.
Blocks are represented by Capital Letters {A, B, C, …}
Variables (lower case letters) represent sets of blocks
This are specified by Quantifiers: for-all x (∀x), There-exists y: (∃y)

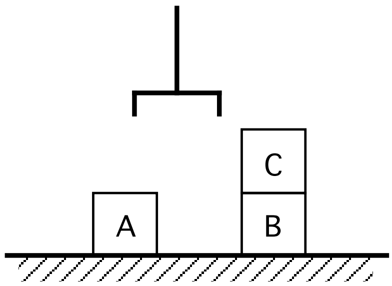
Predicates

Typical predicates used to define states in blocks world are

On(y, x)	O(y,x)	Block x is on Block y
OnTable(x)	OT(x)	Block x is on the table.
Held(x)	H(x)	Block x is in the hand.
Free(x)	F(x)	No block is On x : ¬∃y: (On(y, x)) or ∀y: (¬On(y, x))
HandFree()	HF()	The hand is empty, or ¬∃x (H(x))

Predicates are complex concepts that build on the primitive concepts.

For example: HF ∧ OT(A) ∧ OT(B) ∧ On(B,C) ∧ F(C) ∧ F(A)



This is an example of a situation. We will look schema for representing and reasoning about situations in the coming weeks.

Actions

The system can move from one state to another by performing actions.
Actions are typically represented as condition-action rules.
Actions can be defined by arguments, Preconditions, and Post-conditions.

Action (<blocks>)
Preconditions: Predicates that must be true to execute the action
Post-conditions: Predicates that are made be true or false by the action

Nilsson defined four actions for blocks world.

Grasp(x):
Precondition: HF() ∧ F(x) ∧ OT(x)
Postcondition: ¬HF() ∧ ¬F(x) ∧ ¬OT(x) ∧ H(x)

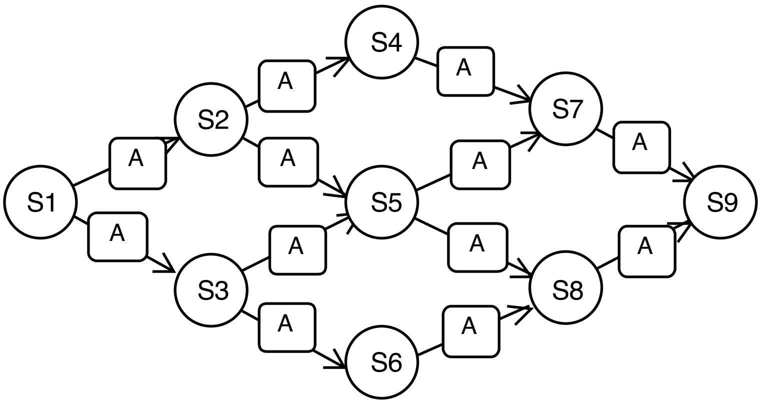
Pose(x):
Precondition: H(x)
Postcondition: ¬H(x) ∧ HF() ∧ F(x) ∧ OT(x)

Stack(x, y) ;;; Stack block x on block y
Precondition: H(x) ∧ F(y)
Postcondition: ¬H(x) ∧ ¬F(y) ∧ F(x) ∧ O(y, x) ∧ HF()

Unstack (x, y)
Precondition: F(x) ∧ O(y, x) ∧ HF()
Postcondition: ¬F(x) ∧ ¬O(y, x) ∧ ¬HF() ∧ H(x) ∧ F(y)

As defined by Nilsson, in Blocks World all precondition predicates are rendered false by execution of an action. However, this is not true of most problem domains.

To solve a problem we search for a path through the state space from an initial state to a target state.



We will use blocks world in the coming lectures to illustrate various techniques.

Comments on Blocks World and Search

Note that blocks world is a "Closed" world. It has a finite number of states.
Real problems tend to be open, with a very large branching factor (possible actions) and an infinite number of states.