Principles of Artificial Intelligence

Vasant Honavar
Artificial Intelligence Research Laboratory
Department of Computer Science
Bioinformatics and Computational Biology Program
Center for Computational Intelligence, Learning, & Discovery
Iowa State University
honavar@cs.iastate.edu
www.cs.iastate.edu/~honavar/
www.cild.iastate.edu/
www.bcb.iastate.edu/
www.igert.iastate.edu

Course Overview

- Foundations
- Intelligent agents
- Problem-solving as State Space Search
- Knowledge Representation and Reasoning
- Planning
- Representing and Reasoning with Uncertain Knowledge
- Decision-Making
- Learning
- Sample Applications of AI

Course Staff

- Vasant Honavar
  Professor of Computer Science
  211 Atanasoff Hall
  http://www.cs.iastate.edu/~honavar/
  honavar@cs.iastate.edu
  Office hours: Mon, Wed: 2:00 – 3:00pm
- Oksana Yakhnenko
  Ph.D. Student, Computer Science
  Artificial Intelligence Research Lab
  http://www.cs.iastate.edu/~oksayakh/
  oksayakh@cs.iastate.edu
  Office hours: TBA

Iowa State University

Department of Computer Science
Artificial Intelligence Research Laboratory

Prerequisites

- Programming
- Mathematics
  - Set theory, logic, probability, calculus
- Data structures
  - Lists, trees, graphs
- Algorithms
- Technical writing and presentation

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Iowa State University

Department of Computer Science
Artificial Intelligence Research Laboratory

Course Materials

- Primary Text:
  - S. Russell and P. Norvig, Artificial Intelligence – A Modern Approach, 2nd edition, 2003
- Additional materials
  http://www.cs.iastate.edu/~cs572/
  - Lecture notes
  - Readings
  - Programming resources

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Iowa State University

Department of Computer Science
Artificial Intelligence Research Laboratory

Course Mechanics

- Grading
  - Problem Sets
  - Laboratory assignments
  - 2 Exams
  - Term project
  - Participation in class
- Late assignments
- Academic Honesty
  - University policy on academic dishonesty
  - Problem sets, labs, term project, collaboration
- Disability

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

What is intelligence?

Intelligent entities

- Perceive
- Reason
- Choose
- Act
- Achieve goals
- Learn
- Reflect
- Communicate
- Organize

What is Artificial Intelligence?

Computation: Cognition : : Calculus : Physics (Artificial Intelligence, Cognitive Science)

- What is the algorithmic basis of learning?
- What is the algorithmic basis of rational decision making?
- Can we automate scientific discovery?
- Can we automate creativity?

What is Artificial Intelligence?

Computation: Society : : Calculus : Physics

- What is the informational and algorithmic basis of inter-agent interaction, communication, and coordination?
- Under what conditions can self-interested rational agents cooperate to achieve a common good?
- How do groups and coalitions form?
- How do different social organizations (democracies, economies, etc.) differ in terms of how they process information?

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

What is artificial intelligence?

Computation : cognition :: calculus : physics
Algorithms as theories

We will have a theory of intelligence when we have computer programs (information processing models) that display intelligence

AI is about

- Study of computational models of intelligence
- Falsifiable hypotheses about intelligent behavior
- Construction of intelligent artifacts (agents)
- Mechanization of tasks requiring intelligence
- Exploring the design space of intelligent systems

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

The Road to Computer Science and Artificial Intelligence

- History of computer science is really a history of human attempts to understand nous (rational mind)
- Aristotle (384-322 BC) distinguishes matter from form thereby laying the foundations of representation
- Panini (350 BC) develops a formal grammar for Sanskrit
- Al Khowarizmi (825) introduces algorithms in his text which was translated to Latin as Algorithmi de numero Indorum
- Descartes (1556-1650) – Cogito ergo sum!

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

The Road to Computer Science and Artificial Intelligence

- Hobbs (1650) suggests that thinking is a rule-based process analogous to arithmetic
- Leibnitz (1646-1716) seeks a general method for reducing all truths to a kind of calculation
- Boole (1815-1864) proposes logic and probability as the basis of laws of thought
- Peirce (1867-1879) introduces theory of signs
- Russell and Whitehead (1910-1915) reduce large portions of mathematics to logic
- Frege (1848-1925) further develops first order logic

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Iowa State University

Department of Computer Science
Artificial Intelligence Research Laboratory

The Road to Computer Science and Artificial Intelligence

- Tarski (1902-1983) introduces a theory of reference for relating objects in a logic to objects in the world
- Ramon Cajal (1910) and others lay the foundations of neuroscience
- Pavlov, Watson, Thorndike, Hull and Skinner (early 1900s) develop behaviorist psychology

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Iowa State University

Department of Computer Science
Artificial Intelligence Research Laboratory

The Road to Computer Science and Artificial Intelligence

- Hilbert (1862-1943) presents the decision problem – Is there an effective procedure for determining whether or not a given theorem logically follows from a given set of axioms?
- Turing (1912-1954) invents the Turing Machine to formalize the notion of an effective procedure
- Godel (1906-1978) shows the existence of an effective procedure to prove any theorem in Frege’s logic and proves the incompleteness theorem

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Iowa State University

Department of Computer Science
Artificial Intelligence Research Laboratory

The Road to Computer Science and Artificial Intelligence

- Church, Kleene, Post, Markov (1930-1950) develop other models of computation based on alternative formalizations of effective procedures
- Turing and Church put forth the Church-Turing thesis that Turing machines are universal computers.
- Several special purpose analog and digital computers are built (including the Atanasoff-Berry Computer)

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

The Road to Computer Science and Artificial Intelligence

- Wiener, Ashby and introduce Cybernetics – the science of communication and control in humans and machines
- Shannon (1948) develops information theory – offering a means to quantify information
- Von Neumann (1956) works out a detailed design for a stored program digital computer
- Several digital computers are constructed and universal languages for programming are developed – Lisp, Snobol, Fortran…

The Road to Computer Science and Artificial Intelligence

- Von Neumann, McCulloch, Rashevsky (1940-1956) investigate the relationship between the brain and the computer
- Von Neumann and Morgenstern (1942) develop a formal framework for rational decision making under uncertainty
- Donald Hebb publishes Organization of Behavior
- Skinner (1950) publishes Verbal Behavior
- Chomsky (1956) develops a theory of syntactic structures
- Wittgenstein (1950s) challenges the rationalist tradition
- Von Neumann (1956) develops a theory of self-reproducing automata

The Road to Computer Science and Artificial Intelligence

- McCarthy, Minsky, Selfridge, Simon, Newell, Uhr et al (1956) begin to investigate the possibility of artificial intelligence
- Rosenblatt and Novikoff (1958) introduce a simple learning machine
- Early demonstrations of artificial intelligence and the publication of Computers and Thought (1959)

The Road to Computer Science and Artificial Intelligence

- Dantzig and Edmunds (1960-62) introduce reduction – a general transformation from one class of problems to another
- Cobham and Edmunds (1964-65) introduce polynomial and exponential complexity
- Cook and Karp (1971-72) develop the theory of NP-completeness which helps recognize problems that are intractable
- Newell and Simon (1970) put forth the physical symbol system hypothesis
- Husserl and Heidegger (1970) advocate a situated view of cognition

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

AI and other disciplines

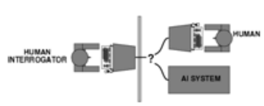
- Much of AI can be viewed as a subfield of CS
- Much of CS can be viewed as a subfield of AI
- AI shares many of the concerns of cognitive science, psychology, epistemology, neuroscience, linguistics, and social sciences
- AI has been (until recently) unique in its use of algorithms as theories
- The domain of enquiry for AI is the entire range of human and non-human intellectual enterprise – Cogito ergo sum!

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

The Turing Test – A measure of AI's success

- When does a system behave intelligently?
  - Turing (1950) Computing Machinery and Intelligence
  - Operational test of intelligence: imitation game



  - Anticipated 50 years of debate on AI
  - Requires all major components of AI: knowledge, reasoning, language understanding, learning, …
  - More sophisticated tests exist – robotic version

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

AI and Intelligent Agents

Intelligent agents must

- Perceive
- Reason
- Choose
- Act
- Achieve goals
- Learn
- Reflect
- Communicate
- Organize

Agents and Environments

An agent can

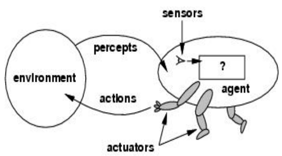
- perceive its environment through its sensors
- represent aspects of its environment and reason with the representation to predict consequences of its actions
- act on its environment through its effectors
- make rational choices
- learn from experience
- communicate with other agents – signals, signs, language
- interact with other agents – cooperation, competition
- display autonomous, purposeful behavior

I will know an agent when I see one!

Which of the following is an agent?

- A spell checker
- A payroll program
- An email program
- A web portal
- A thermostat
- A unix daemon
- A physician's diagnostic assistant
- An electronic stock trader
- An autonomous vehicle
- A personal information assistant
- Yours AI professor

Agents and environments



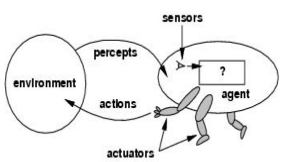
- Agents include humans, robots, softbots, thermostats, etc.
- The *agent function* maps percept sequence to actions

$$f : P^* \rightarrow A$$

- An agent can perceive its own actions, but not always it effects

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

Agents and environments

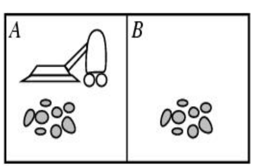


- The *agent function* will internally be represented by the *agent program*.
- The agent program runs on the physical *architecture* to produce *f*.

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

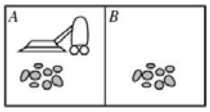
The vacuum-cleaner world



- Environment: square A and B
- Percepts: [location and content] e.g. [A, Dirty]
- Actions: left, right, suck, and no-op

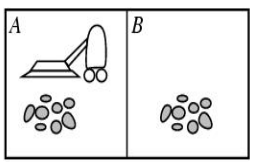
These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

The vacuum-cleaner world



Percept sequence	Action
[A,Clean]	Right
[A, Dirty]	Suck
[B, Clean]	Left
[B, Dirty]	Suck
[A, Clean],[A, Clean]	Right
[A, Clean],[A, Dirty]	Suck
…	…

The vacuum-cleaner world



function REFLEX-VACUUM-AGENT ([*location, status*]) return an action
if *status* == *Dirty* then return *Suck*
else if *location* == *A* then return *Right*
else if *location* == *B* then return *Left*

Rational Agents

- A rational agent is one that does the right thing
- What does it mean to do the right thing?
  - Depends on the performance measure
- Examples of performance measures
  - E.g. the amount of dirt cleaned within a certain time
  - E.g. how clean the floor is
  - …
- What is your performance measure?
- Who decides what the performance measure should be?
  - Internal drives
  - External rewards

Rationality

- What is rational at a given time depends on:
  - Performance measure
  - Knowledge of the environment
  - Actions
  - Percept sequence to date (sensors)
- Rationality ≠ omniscience
  - An omniscient agent knows the actual outcome of its actions
- Rationality ≠ perfection
  - Rational agent maximizes *expected* performance
  - Perfect agent maximizes *actual* performance

Environments

- To design a rational agent we must specify PEAS
  - Performance
  - Environment
  - Actions
  - Sensors

Environments

PEAS for a fully automated taxi:

- Performance
  - Safety, reaching destination, profits, comfort
- Environment
  - Streets, other traffic, pedestrians, weather,, …
- Actuators
  - Steering, accelerating, brake, horn, speaker, display,…
- Sensors
  - Video, sonar, speedometer, engine sensors, keyboard, GPS, …

Environment types

- Observable: can the agent sense the state of the environment?
- Controllable: Can the agent perform actions that lead to desired state change?
- Predictable: Deterministic, Stochastic
- Episodic vs non-episodic
- Static vs dynamic
- Discrete versus continuous
- Purposeful: are there other agents?
- Open versus closed

Environment types

	Solitaire	Backgammon	Internet shopping	Taxi
Observable??				
Deterministic??				
Episodic??				
Static??				
Discrete??				
Single-agent??				

Environment types

Fully vs. partially observable: an environment is fully observable when the sensors can detect all aspects that are *relevant* to the choice of action

	Solitaire	Backgammon	Internet shopping	Taxi
Observable??				
Deterministic??				
Episodic??				
Static??				
Discrete??				
Single-agent??				

Environment types

Fully vs. partially observable: an environment is fully observable when the sensors can detect all aspects that are *relevant* to the choice of action.

	Solitaire	Backgammon	Internet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??				
Episodic??				
Static??				
Discrete??				
Single-agent??				

Environment types

Deterministic vs. stochastic: if the next environment state is completely determined by the current state the executed action then the environment is deterministic.

	Solitaire	Backgammon	Internet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??				
Episodic??				
Static??				
Discrete??				
Single-agent??				

Environment types

Deterministic vs. stochastic: if the next environment state is completely determined by the current state the executed action then the environment is deterministic

	Solitaire	Backgammon	Internet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??	YES	NO	YES	NO
Episodic??				
Static??				
Discrete??				
Single-agent??				

Environment types

Episodic vs. non episodic: In an episodic environment the agent's experience can be divided into episodes – agent's actions in consecutive episodes are independent

	Solitaire	Backgammon	Internet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??	YES	NO	YES	NO
Episodic??				
Static??				
Discrete??				
Single-agent??				

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

Environment types

Episodic vs. non episodic: In an episodic environment agent's experience can be divided into episodes – agent's actions in consecutive episodes are independent

	Solitaire	Backgammon	Internet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??	YES	NO	YES	NO
Episodic??	NO	NO	NO	NO
Static??				
Discrete??				
Single-agent??				

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

Environment types

Static vs. dynamic: If the environment can change while the agent is choosing an action, the environment is dynamic

	Solitaire	Backgammon	Internet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??	YES	NO	YES	NO
Episodic??	NO	NO	NO	NO
Static??				
Discrete??				
Single-agent??				

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Environment types

Static vs. dynamic: If the environment can change while the agent is choosing an action, the environment is dynamic

	Solitaire	Backgammon	Internet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??	YES	NO	YES	NO
Episodic??	NO	NO	NO	NO
Static??	YES	YES	SEMI	NO
Discrete??				
Single-agent??				

Environment types

Discrete vs. continuous

	Solitaire	Backgammon	Internet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??	YES	NO	YES	NO
Episodic??	NO	NO	NO	NO
Static??	YES	YES	SEMI	NO
Discrete??				
Single-agent??				

Environment types

Discrete vs. continuous

	Solitaire	Backgammon	Internet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??	YES	NO	YES	NO
Episodic??	NO	NO	NO	NO
Static??	YES	YES	SEMI	NO
Discrete??	YES	YES	YES	NO
Single-agent??				

Environment types

Single vs. multi-agent: Does the environment contain other agents?

	Solitaire	Backgammon	Internet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??	YES	NO	YES	NO
Episodic??	NO	NO	NO	NO
Static??	YES	YES	SEMI	NO
Discrete??	YES	YES	YES	NO
Single-agent??				

Environment types

Single vs. multi-agent: Does the environment contain other agents?

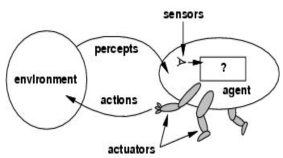
	Solitaire	Backgammon	Internet shopping	Taxi
Observable??	FULL	FULL	PARTIAL	PARTIAL
Deterministic??	YES	NO	YES	NO
Episodic??	NO	NO	NO	NO
Static??	YES	YES	SEMI	NO
Discrete??	YES	YES	YES	NO
Single-agent??	YES	NO	NO	NO

Environment types

- The simplest environment is
  - Fully observable, deterministic, episodic, static, discrete and single-agent
- Many real world environments can be:
  - Partially observable, stochastic, non-episodic dynamic, continuous and multi-agent

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

Agent types



All agents have:

- Input = current percept
- Output = action
- Program = processes input to produce output

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

Agent types

Function TABLE-DRIVEN_AGENT(*percept*) **returns** an action

static: *percepts*, a sequence initially empty
table, a table of actions, indexed by percept sequence

append *percept* to the end of *percepts*
action ← LOOKUP(*percepts*, *table*)
return *action*

This approach is doomed to failure in all but the simplest cases
Why?

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

Agent types

Basic agent types

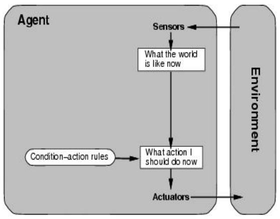
- Simple reflex agents or reactive agents
- Model-based reflex agents
- Deliberative agents (often goal-based)
- Utility-based agents

All these agent types can be

- Learning
- Communicative
- Interactive (competitive, collaborative)

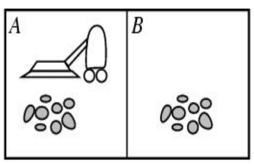
These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Simple Reflex Agents



- Select action on the basis of *only the current* percept
  - E.g. the vacuum-agent
- Implemented through *condition-action rules*
  - If dirty then suck

The vacuum-cleaner world



function REFLEX-VACUUM-AGENT ([*location, status*]) return an action
if *status* == *Dirty* then return *Suck*
else if *location* == *A* then return *Right*
else if *location* == *B* then return *Left*

Simple reflex agents

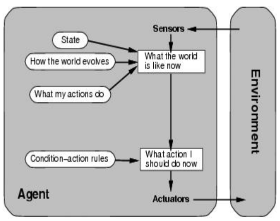
function SIMPLE-REFLEX-AGENT(*percept*) **returns** an action

static: *rules*, a set of condition-action rules

state ← INTERPRET-INPUT(*percept*)
rule ← RULE-MATCH(*state*, *rule*)
action ← RULE-ACTION[*rule*]
return *action*

May fail if the environment is not fully observable
Why?

Reflex agents with state



- Can handle partially observable environments.
  - Maintain internal state
- Over time update state using world knowledge
  - How does the world change
  - How do actions affect world
  - ⇒ Model of World

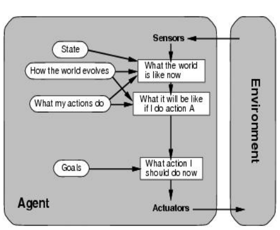
Reflex agents with state

function REFLEX-AGENT-WITH-STATE(*percept*) **returns** an action

static: *rules*, a set of condition-action rules
state, a description of the current world state
action, the most recent action.

state ← UPDATE-STATE(*state*, *action*, *percept*)
rule ← RULE-MATCH(*state*, *rule*)
action ← RULE-ACTION[*rule*]
return *action*

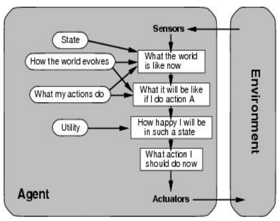
Goal-based agents



- The agent seeks to achieve a specified goal
- Attaining a goal may require a long sequence of actions
- Needs a model (representation) of the world

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

Utility-based agents

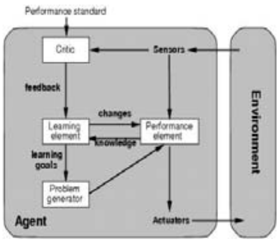


- Different goals have different utility to the agent
- Utility function maps a (sequence of) state(s) onto a real number (utility)
- Goal-based agents are a special case

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

Learning agents

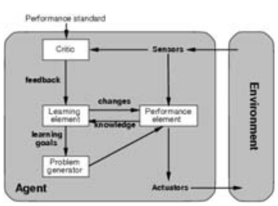


- All previous agent-programs use knowledge (e.g., model of the environment)
- Where do these models come from?
  - Learning

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Iowa State University
Department of Computer Science
Artificial Intelligence Research Laboratory

Learning agents



- *Learning element*
- introduce improvements in performance element.
  - Critic provides feedback on agent's performance
- *Performance element*:
  - Corresponds to the previous agent programs
- *Problem generator*: suggests actions that lead to new and informative experiences.

These slides are based on material from Russell and Norvig, with updates by Vasant Honavar, 2006.

Multi-agent Systems

Beyond the scope of this course

- Inter-agent communication
- Multi-agent Interaction
  - Cooperation
  - Competition
- Multi-agent organizations