Artificial Intelligence

Unit1

Introduction : Basics of problem solving, problem representation (toy problems and real world problems); Structure of agent, rational agent, Specifying task environment, Properties of task environment; measuring problem-solving performance

By

Dr. Shailendra S. Aote

Course Outcomes

On successful completion of the course, students will be able to:

- Represent given problem using state space representation and apply uninformed and informed search techniques on it.
- Solve the fully informed two player games using different AI techniques.
- Solve the AI problems by using logic programming
- Apply uncertainty theory based on techniques like probability theory and fuzzy logic.

Intelligence

The ability to acquire & apply knowledge to solve a problem.

What is AI

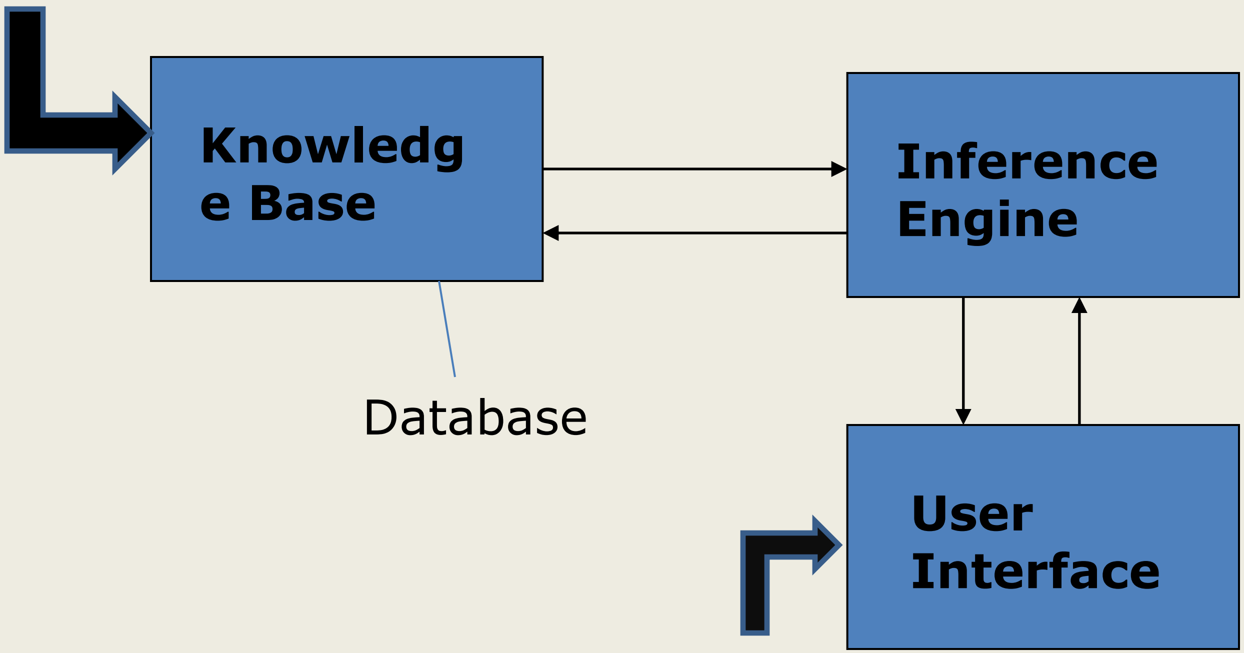
- AI is the study of how to make computers to do things which at the moment , people do better. (Rich & Knight)

- Is a branch of computer Science that is concerned with the automation of intelligent behavior in simple terms.

AI is a way of making a computer thinks intelligently (Mathews)

Block Architecture of AI System

- Collection of facts and rules.
- Place where the domain specific knowledge is stored
- Consist of body of facts axioms, rules and concepts about different entities.



- Means of utilizing facts and rules.
Carries out the search through the Knowledge base
- To prove the hypothesis & to arrive at a conclusion

- Provides the needed facilities for the user to communicate with the system.

Big Questions

- Can machines think?
- And if so, how?
- And if not, why not?
- And what does this say about human beings?

Other possible AI definitions

- AI is a collection of hard problems which can be solved by humans and other living things, but for which we don’t have good algorithms for solving.
  - e. g., understanding spoken natural language, medical diagnosis, learning, self-adaptation, reasoning, chess playing, proving math theories, etc.
- System that Act like human (Turing test)
  - Natural Language processing for communication
  - Knowledge representation to store information
  - Automated Reasoning to use stored information
  - Machine Learning to adapt to new circumstances and to detect and extrapolate (to infer) patterns

Other possible AI definitions

- ❑ **System that Think like human** (human-like patterns of thinking steps)
  - ✓ Through Introspection trying to catch our own thoughts as they go by.
  - ✓ Through psychological experiments
  - ✓ *Once we have a sufficiently precise theory of the mind it becomes possible to express the theory as a computer program.*
- ❑ **System that think rationally** (logically, correctly)

  Aristotle was one of the first to attempt to codify “right Thinking” which is called **Syllogisms**

  <u>Example</u> Socrates is a man

  all men are mortal

  therefore Socrates is mortal

- **Acts Rationally** acting so as to achive one’s goal given one’s beliefs.

What’s easy and what’s hard?

- It’s been easier to mechanize many of the high level cognitive tasks we usually associate with “intelligence” in people
  - e. g., symbolic integration, proving theorems, playing chess, some aspect of medical diagnosis, etc.
- It’s been very hard to mechanize tasks that animals can do easily
  - catching prey and avoiding predators
  - interpreting complex sensory information (visual, aural, …)
  - modeling the internal states of other animals from their behavior
  - working as a team (ants, bees)

History of AI

- AI has roots in a number of scientific disciplines
  - computer science and engineering (hardware and software)
  - philosophy (rules of reasoning)
  - mathematics (logic, algorithms, optimization)
  - cognitive science and psychology (modeling high level human/animal thinking)
  - neural science (model low level human/animal brain activity)
  - Linguistics, economics, etc.
- The birth of AI (1943 – 1956)
  - Pitts and McCulloch (1943): simplified mathematical model of neurons (resting/firing states) can realize all propositional logic primitives (can compute all Turing computable functions)
  - Allen Turing: Turing machine and Turing test (1950)
  - Claude Shannon: information theory; possibility of chess playing computers

- Early enthusiasm (1952 – 1969)
  - 1956 Dartmouth conference

    John McCarthy (Lisp);

    Marvin Minsky (first neural network machine);

    Alan Newell and Herbert Simon (GPS) General Problem Solver;
  - Emphasize on intelligent general problem solving

    GSP (means-ends analysis);

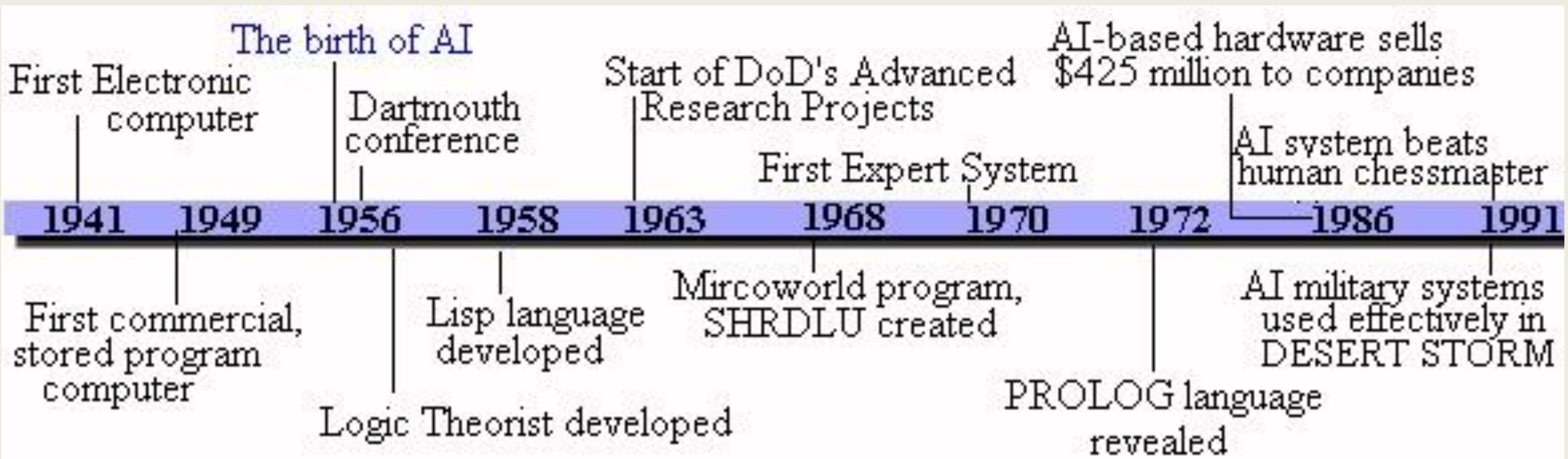
    Lisp (AI programming language);

    Resolution by John Robinson (basis for automatic theorem proving);

    heuristic search (A*, AO*, game tree search)
- Emphasis on knowledge (1966 – 1974)
  - Domain specific knowledge is the key to overcome existing difficulties
  - Knowledge representation (KR) paradigms
  - Declarative vs. procedural representation

- Knowledge-based systems (1969 – 1979)
  - DENDRAL: the first knowledge intensive system (determining 3D structures of complex chemical compounds)
  - MYCIN: first rule-based expert system (containing 450 rules for diagnosing blood infectious diseases)
    EMYCIN: an ES shell
  - PROSPECTOR: first knowledge-based system that made significant profit (geological ES for mineral deposits)
- AI became an industry (1980 – 1989)
  - wide applications in various domains
  - commercially available tools
- Current trends (1990 – present)
  - more realistic goals
  - more practical (application oriented)
  - distributed AI and intelligent software agents
  - rebirth of neural networks and emergence of genetic algos

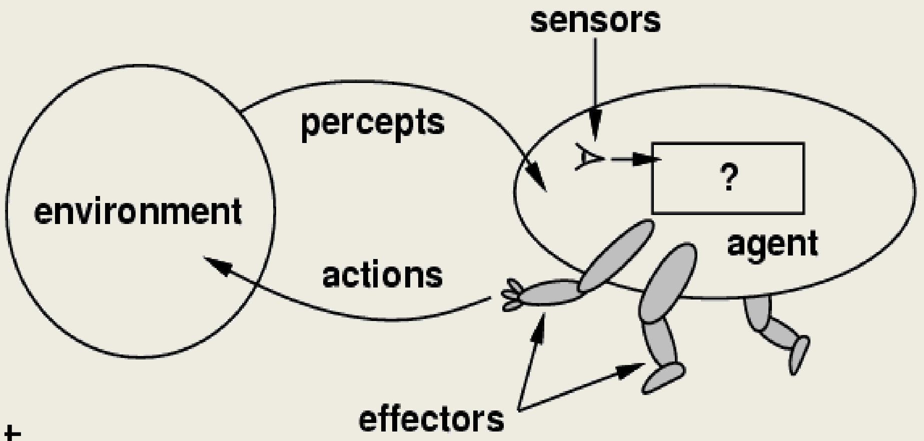
History



The birth of AI
First Electronic computer
Dartmouth conference
Start of DoD's Advanced Research Projects
AI-based hardware sells $425 million to companies
AI system beats human chessmaster
First Expert System
1941
1949
1956
1958
1963
1968
1970
1972
1986
1991
First commercial, stored program computer
Logic Theorist developed
Lisp language developed
Mircoworld program, SHRDLU created
PROLOG language revealed
AI military systems used effectively in DESERT STORM

Intelligent Agents

- Definition: An Intelligent Agent perceives it environment via sensors and acts rationally upon that environment with its effectors.
- Hence, an agent gets percepts one at a time, and maps this percept sequence to actions.
- Properties
  - Autonomous
  - Interacts with other agents plus the environment
  - Reactive to the environment
  - Pro-active (goal- directed) future problems



What do you mean, sensors/percepts and effectors/actions?

- Humans
  - Sensors: Eyes (vision), ears (hearing), skin (touch), tongue (gustation), nose (olfaction).
  - Percepts:
    - At the lowest level – electrical signals from these sensors
    - After preprocessing – objects in the visual field (location, textures, colors, …), auditory streams (pitch, loudness, direction), …
  - Effectors: limbs, digits, eyes, tongue, …
  - Actions: lift a finger, turn left, walk, run, carry an object, …
- The Point: percepts and actions need to be carefully defined, possibly at different levels of abstraction

A more specific example: Automated taxi driving system

- **Percepts**: Video, sonar, speedometer, odometer, engine sensors, keyboard input, microphone, GPS, …
- **Actions**: Steer, accelerate, brake, horn, speak/display, …
- **Goals**: Maintain safety, reach destination, maximize profits (fuel, tire wear), obey laws, provide passenger comfort, …
- **Environment**: U.S. urban streets, freeways, traffic, pedestrians, weather, customers, …
- **Different aspects of driving may require different types of agent programs!**

Rationality

- An ideal **rational agent** should, for each possible percept sequence, do whatever actions that will maximize its performance measure based on

    (1) the percept sequence, and

    (2) its built-in and acquired knowledge**.**

- Hence it includes information gathering, not "rational ignorance."
- Rationality => Need a performance measure to say how well a task has been achieved.

Autonomy

- A system is autonomous to the extent that its own behavior is determined by its own experience and knowledge.
- Therefore, a system is not autonomous if it is guided by its designer according to a priori decisions.
- To survive agents must have:
  - Enough built- in knowledge to survive.
  - Ability to learn.

Examples of Agent Types and their Descriptions

Agent Type	Percepts	Actions	Goals	Environment
Medical diagnosis system	Symptoms, findings, patient's answers	Questions, tests, treatments	Healthy patient, minimize costs	Patient, hospital
Satellite image analysis system	Pixels of varying intensity, color	Print a categorization of scene	Correct categorization	Images from orbiting satellite
Part-picking robot	Pixels of varying intensity	Pick up parts and sort into bins	Place parts in correct bins	Conveyor belt with parts
Refinery controller	Temperature, pressure readings	Open, close valves; adjust temperature	Maximize purity, yield, safety	Refinery
Interactive English tutor	Typed words	Print exercises, suggestions, corrections	Maximize student's score on test	Set of students

State Space Representation

A problem is really a collection of information that the agent will use to decide what to do.

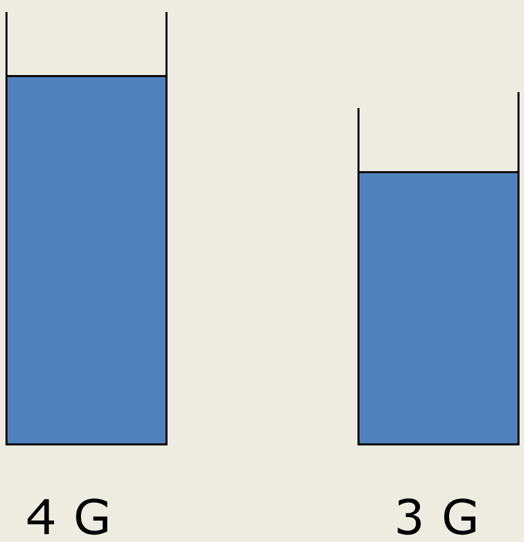
State Space Representation:

- The initial state
- Goal State
- Operator / Successor Function:

  The description of an action in terms of which state will be reached by carrying out the action in a particular state.
- Goal Test
- Path cost

Water Jug Problem

You are given two jugs, a 4-gallon one and 3-gallon one. Neither has any measuring marks on it. There is a pump that can be used to fill the jugs with water. How can you get exactly 2 gallons of water into the 4-gallon jug?



State Space Search: Water Jug Problem

- ❑ **Start state: (0, 0)**
- ❑ **Goal state: (2, n) for any n**
- ❑ **Goal Test : State matching to any configuration of goal state**
- ❑ **Operators:**
  - ✓Fill the 4-gallon jug
  - ✓Fill the 3-gallon jug
  - ✓Pour some water From 4 to 3 gallon jug
  - ✓Pour some water From 3 to 4 gallon jug
  - ✓Empty the 4-gallon jug
  - ✓Empty the 3-gallon jug

A Water Jug solution

4-Gallon Jug	3-Gallon Jug	Rule Applied
0	0	
0	3	2
3	0	4
3	3	2
4	2	4
0	2	5
2	0	4

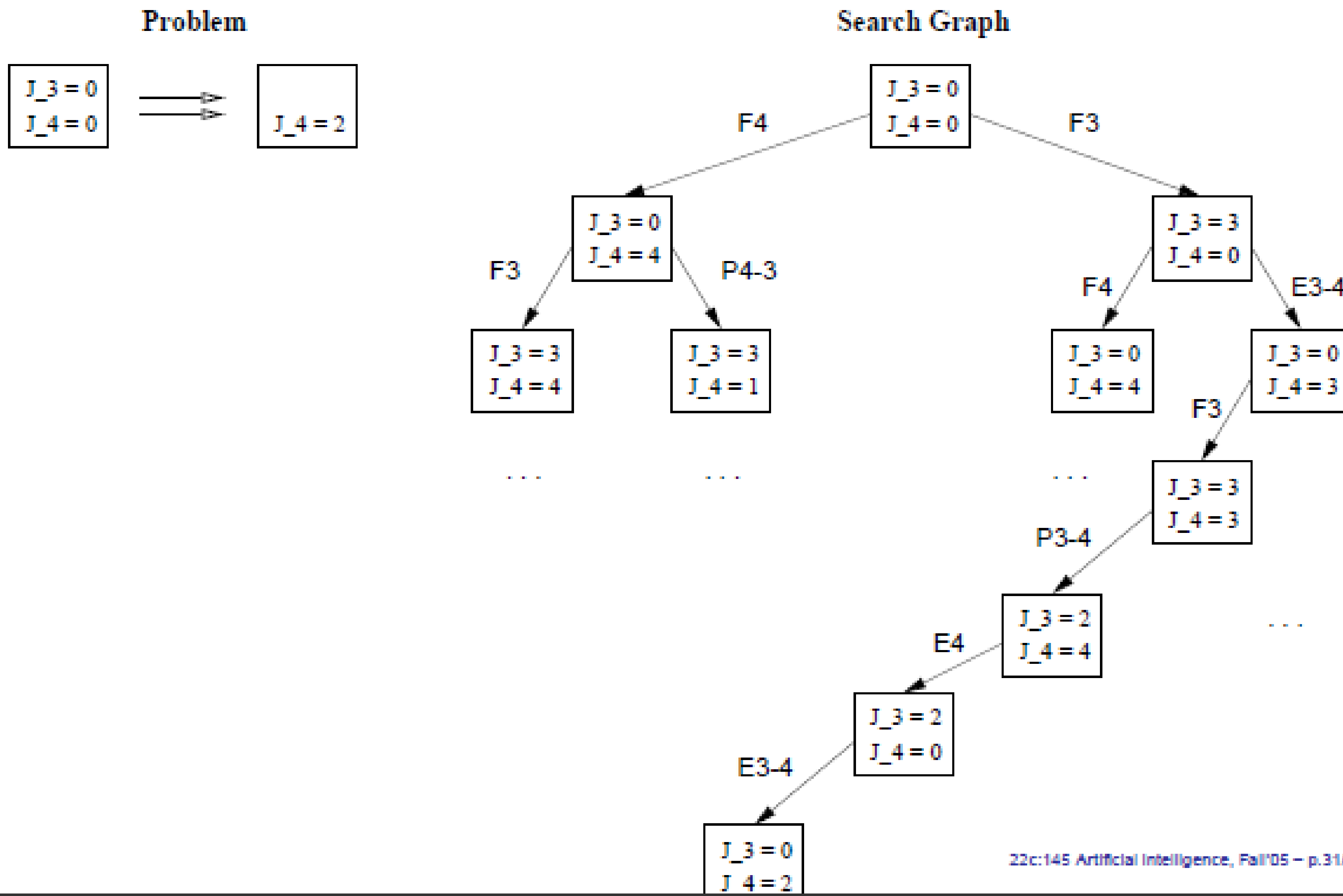
Solution : path cost =6

8,5,3

8,0,0 -> 4,4,0

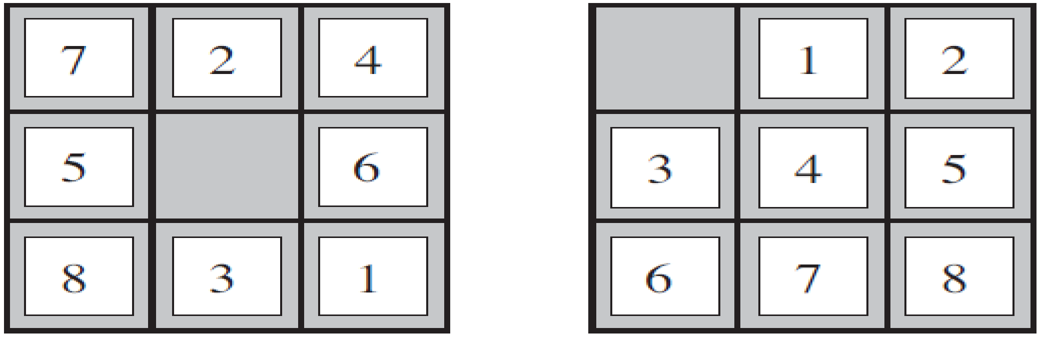
Another way to show the solution ?

The Water Jugs Problem



The 8-puzzle

- **States:** a state description specifies the location of each of the eight tiles in one of the nine squares. For efficiency, it is useful to include the location of the blank.
- **Initial State:** any state
- **Operators:** blank moves left, right, up, or down
- **Transition model:** Given a state and action, this returns the resulting state; for example, if we apply *Left to the start state in Figure the resulting state has the 5 and the blank* switched.
- **Goal test:** state matches the goal configuration shown in Figure
- **Path cost:** each step costs 1, so the path cost is just the length of the path.



Start State Goal State

The 8-queens problem

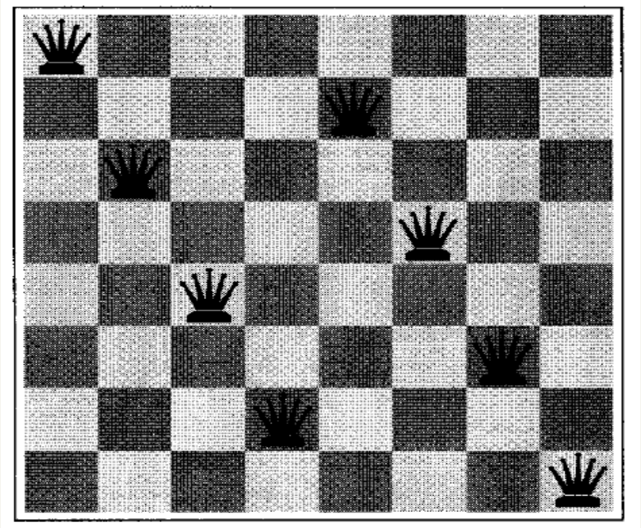
States: Any arrangement of 0 to 8 queens on the board is a state.

Initial state: No queens on the board.

Actions/Operators: Add a queen to any empty square.

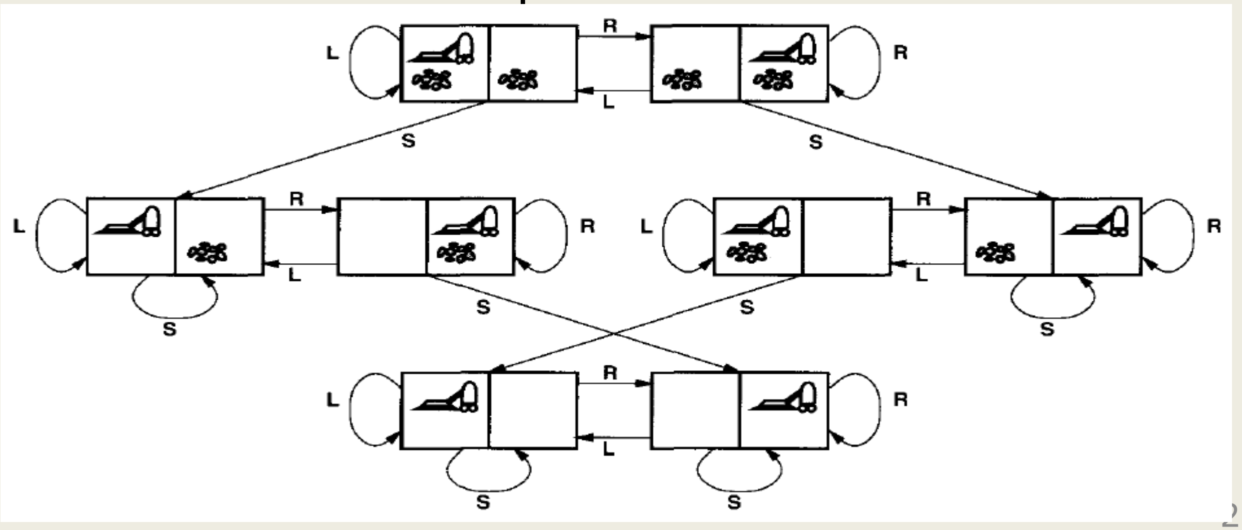
Transition model: Returns the board with a queen added to the specified square.

Goal test: 8 queens are on the board, none attacked



The vacuum world

- **States:** one of the eight states shown in Figure
- **Initial state:** Any state can be designated as the initial state.
- **Operators/Actions:** move left, move right, suck.
- **Transition model:** The actions have their expected effects, except that moving Left in the leftmost square, moving *Right in the rightmost square, and Sucking in a clean square* have no effect.
- **Goal test:** This checks whether all the squares are clean.
- **Path cost:** each action costs 1. so the path cost is the number of steps in the path.



Airline travel problems

- **States**: Each state obviously includes a location (e.g., an airport) and the current time. Furthermore, because the cost of an action (a flight segment) may depend on previous segments, their fare bases, and their status as domestic or international
- **Initial state**: This is specified by the user's query.
- **Actions**: Take any flight from the current location, in any seat class, leaving after the current time, leaving enough time for within-airport transfer if needed.
- **Transition model**: The state resulting from taking a flight will have the flight's destination as the current location and the flight's arrival time as the current time.
- **Goal test**: Are we at the final destination specified by the user?
- **Path cost**: This depends on monetary cost, waiting time, flight time, customs and immigration procedures, seat quality, time of day, type of airplane, frequent-flyer mileage awards, and so on.

Properties of Environments

- **Accessible/ Inaccessible.**
  - If an agent's sensors give it access to the complete state of the environment needed to choose an action, the environment is accessible.
  - Such environments are convenient, since the agent is freed from the task of keeping track of the changes in the environment.
- **Deterministic/ Non-deterministi**c.
  - An environment is deterministic if the next state of the environment is completely determined by the current state of the environment and the action of the agent.
  - In an accessible and deterministic environment the agent need not deal with uncertainty.
- **Episodic/ Nonepisodi**c.
  - An episodic environment means that subsequent episodes do not depend on what actions occurred in previous episodes.
  - Such environments do not require the agent to plan

Properties of Environments

- **Static/ Dynamic.**
  - A static environment does not change while the agent is thinking.
  - In a static environment the agent need not worry about the passage of time while he is thinking, nor does he have to observe the world while he is thinking.
  - In static environments the time it takes to compute a good strategy does not matter.
- **Discrete/ Continuous.**
  - If the number of distinct percepts and actions is limited the environment is discrete, otherwise it is continuous.
-

Characteristics of environments

	Accessible	Deterministic	Episodic	Static	Discrete
Solitaire					
Backgammon					
Taxi driving					
Internet shopping					
Medical diagnosis					

Characteristics of environments

	Accessible	Deterministic	Episodic	Static	Discrete
Solitaire	**No**	**Yes**	**Yes**	**Yes**	**Yes**
Backgammon					
Taxi driving					
Internet shopping					
Medical diagnosis					

Characteristics of environments

	Accessible	Deterministic	Episodic	Static	Discrete
Solitaire	No	Yes	Yes	Yes	Yes
Backgammon	**Yes**	**No**	**No**	**Yes**	**Yes**
Taxi driving					
Internet shopping					
Medical diagnosis					

Characteristics of environments

	Accessible	Deterministic	Episodic	Static	Discrete
Solitaire	No	Yes	Yes	Yes	Yes
Backgammon	Yes	No	No	Yes	Yes
Taxi driving	**No**	**No**	**No**	**No**	**No**
Internet shopping					
Medical diagnosis					

Characteristics of environments

	Accessible	Deterministic	Episodic	Static	Discrete
Solitaire	No	Yes	Yes	Yes	Yes
Backgammon	Yes	No	No	Yes	Yes
Taxi driving	No	No	No	No	No
Internet shopping	**No**	**No**	**No**	**No**	**No**
Medical diagnosis					

Characteristics of environments

	Accessible	Deterministic	Episodic	Static	Discrete
Solitaire	No	Yes	Yes	Yes	Yes
Backgammon	Yes	No	No	Yes	Yes
Taxi driving	No	No	No	No	No
Internet shopping	No	No	No	No	No
Medical diagnosis	**No**	**No**	**No**	**No**	**No**

→ Lots of real-world domains fall into the hardest case!

Summary

- An **agent** perceives and acts in an environment, has an architecture and is implemented by an agent program.
- An **ideal agent** always chooses the action which maximizes its expected performance, given percept sequence received so far.
- An **autonomous agent** uses its own experience rather than built- in knowledge of the environment by the designer.
- An **agent program** maps from percept to action & updates its internal state.
  - **Reflex agents** respond immediately to percpets.
  - **Goal-based agents** act in order to achieve their goal(s).
  - **Utility-based** agents maximize their own utility function.
- **Representing knowledge** is important for