Artificial Intelligence

Agent-Based Systems

MONTANA STATE UNIVERSITY

Administrative

- Dr. John W. Sheppard
  - Email: john.sheppard@cs.montana.edu
  - Phone: (406) 994-4835
  - Address: Department of Computer Science
    EPS 363
    Montana State University
    Bozeman, MT 59715

- Teaching Assistant
  - TBD

Objectives

- Formulate and assess problems in artificial intelligence.
- Assess the strengths and weaknesses of methods for representing knowledge.
- Assess the strengths and weaknesses of AI algorithms.
- Implement software solutions to AI-related problems.

Readings

- S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, Second Edition
- Various papers distributed throughout the course.

MONTANA STATE UNIVERSITY

Mountains Minds

Policies

- All assignments to be original work of individual.
- All students will abide by ethics code.
- All assignments due at *beginning* of indicated class.
- No assignments will be accepted late without prior arrangement with the instructor (except for serious, unavoidable circumstances).
- Class attendance is required.
- No cell phones or pagers in class.

MONTANA STATE UNIVERSITY

Mountains Minds

Policies (continued)

- All algorithms must include pseudo-code and prose-description.
- Any programming language on any machine is acceptable.
- Program must run on department network.
- All programming assignments must include fully commented code and several sample runs.
- The web may be used for background research, but not to solve problems.
- Any web resources used must by cited.

MONTANA STATE UNIVERSITY

Mountains Minds

A Note on Cheating

- All work turned in is expected to be your own work.
- Papers turned in will be photocopied.
- Papers will be checked randomly for similarity to any web-based or other solutions.
- Failure to cite sources used will result in failure of assignment.
- Plagiarism or other modes of cheating will result in failure of the course.
- Detailed examples of academic misconduct provided in syllabus.

Grading

- Four problem assignments (10% each)
- Three programming assignments (10% each)
- Programming assignments have written component
  - Program 1: Programming 70%, Writing 30%
  - Program 2: Programming 50%, Writing 50%
  - Program 3: Programming 30%, Writing 70%
- Class attendance (10%).
- Final examination (20%).

Programming Assignments

- All programming assignments must include a short paper describing the results of an experiment using the implemented algorithms.
- Unless otherwise indicated, all papers must include the following:
  - Clear, testable hypothesis
  - Description of the *algorithm* implemented (not the code)
  - Description of the experimental approach
  - Analysis and discussion of results
  - References (at a minimum, the course text and class notes).
- In addition to the paper, the following must be turned in:
  - Fully commented source code
  - Sample runs tracing execution on a *small* example problem
  - Data sets

Grading Appeals

- Assignments will be returned within one week of being handed in.
- Questions on grading should be directed to the TA within one week of receiving your paper.
- If you are not satisfied with the resolution from the TA, you have two days from receiving that resolution to appeal, in writing, to the instructor.
- The instructor's decision is final.

MONTANA STATE UNIVERSITY

Mountains and Minds

What is Artificial Intelligence?

- What do you think?

MONTANA STATE UNIVERSITY

Mountains and Minds

What is AI?

- Generally speaking, attacking a problem that people seem to solve well:
  - Ill structured (intuitive reasoning)
    - How does an expert reach a diagnosis?
  - Highly complex (combinatorics)
    - How does a grand master structure a chess game?
  - Requires an understanding of the world
    - How do you plan your day to accomplish your goals?
  - May require learning and adaptation
    - How well do you know Bozeman now, as opposed to when you came?

MONTANA STATE UNIVERSITY

Mountains and Minds

Dimensions of AI

	Reasoning	
Behavior	Think like humans	Think rationally
	Act like humans	Act rationally

MONTANA STATE UNIVERSITY
Mountains Minds
13

Acting Humanly

- Turing (1950): "Computing machinery and intelligence."
  - Can machines think? Can machines behave intelligently?
  - Proposed the Turing test--an "imitation test."
  - Predicted that by 2000, a machine might have a 30% chance of fooling a person for 5 minutes.
  - Anticipated all major arguments against AI that arose in the subsequent 50 years.
  - Suggested the major components of AI: knowledge, reasoning, language understanding, learning.
- Problem: The Turing test is not reproducible, constructive, or amenable to mathematical analysis.

MONTANA STATE UNIVERSITY
Mountains Minds
14

Thinking Humanly

- The birth of "cognitive science."
- 1960s "cognitive revolution" : information processing psychology replaced prevailing orthodoxy of behaviorism.
- Requires scientific theories of internal activities of the brain.
  - Level of abstraction (knowledge or circuits)
  - Methods of validation
    - Human testing
    - Neurological testing
- Both cognitive science and cognitive neuroscience are distinct from AI.

MONTANA STATE UNIVERSITY
Mountains Minds
15

Thinking Rationally

- Normative (prescriptive) rather than descriptive.
- Aristotle: What are correct arguments/thought processes?
- Several Greeks schools developed various forms of logic, which represented notations and rules for derivation of thoughts.
- There is a direct line through mathematics and philosophy to modern AI.
- Problems:
  - Not all intelligent behavior is mediated by logical deliberation.
  - Does not explain the purpose of thinking as opposed to reactive behavior.

Acting Rationally

- **Def:** A system behaves *rationally* if it "does the right thing."
  - Does not need to be optimal.
  - Does not need to be human-like.
- Rational behavior does not necessarily require "thinking," but thinking should be in the service of rational action.
- Aristotle: "Every art and every inquiry, and similarly every action and pursuit, is thought to aim at some good."

Rational Agents

- An *agent* is an entity that perceives and acts.
- Abstractly, an agent is a function, mapping from percept histories to actions:
  - $f : P^* \rightarrow A$
- For any given class of environments and tasks, we seek the agent (or class of agents) with the best performance.
- Caveat: Computational limitations make perfect rationality unachievable. Therefore, we seek to design the best program for given machine resources.

A Brief History of the Field

- 1943-1956
  - McCulloch & Pitts/Hebb
    - Simple neural models of processing
  - Shannon/Turing
    - Computers can manipulate symbols
    - Chess as a canonical example
  - Dartmouth Conference (1956)
    - Genesis of the term "AI"
  - Newell & Simon
    - Logic Theorist

MONTANA STATE UNIVERSITY

Mountains Minds

A Brief History of the Field

- 1952-1969
  - General Problem Solver
    - Generalization of Logic Theorist
  - Samuel Checkers Program
    - Used learning to improve performance
  - LISP (1958)/Time Sharing/Advice Taker (McCarthy)
  - Minsky Micro-Worlds
    - The hackers (or scruffies) vs. the neats
  - Resolution Method Invented

MONTANA STATE UNIVERSITY

Mountains Minds

A Brief History of the Field

- 1966-1974
  - Realizing the Limits of "Syntax" (Weak Methods)
    - What is a notion of "meaning" beyond symbols?
  - Ubiquity of the Combinatorics
    - Every problem seems to hit a dead-end
  - The (in)famous Perceptron Paper

MONTANA STATE UNIVERSITY

Mountains Minds

A Brief History of the Field

- 1969-1979
  - Expert Systems
    - Dendral, Prospector, MYCIN
  - Formalisms for “Knowledge”
    - Reaction to the problems of syntax
  - Recognition of Need to Understand Uncertainty
    - MYCIN uncertainty factors as a heuristic
  - Shakey the Robot
    - Highly successful in a limited micro-world

MONTANA STATE UNIVERSITY

Mountains Minds

A Brief History of the Field

- 1980-1988
  - Expert Systems Engineering
    - R1 at DEC
  - LISP Machines Developed
  - Industrial Vision Takes Hold
  - Hilare Robot Performs Mapping and Planning in Real World
  - Japanese 5th Generation Project
  - The Return of Neural Networks

MONTANA STATE UNIVERSITY

Mountains Minds

A Brief History of the Field

- 1990-present
  - Formal Methods to the Fore
    - HMM’s
    - Probability and statistics (graphical models)
  - AI Dilution
    - Many systems need just a “little” AI combined with solid CS
  - AI Maturity
    - Where did all those hackers go?

MONTANA STATE UNIVERSITY

Mountains Minds

Shakey vs. Polly

Shakey, ca. 1969

Polly, ca. 1994

MONTANA STATE UNIVERSITY

25

Mountains & Minds

Shakey vs. Polly

Shakey, ca. 1969

Goal: study *general* methods whereby a robot control system can interact with a complex environment

Problem: Move about and manipulate objects

Environment: Single room painted black and white (micro-world)

- Built at SRI, late '60s
- TV Camera (120x120x4)
- laser range finder
- "whiskers"
- offboard computing with RF link (PDP 10)

MONTANA STATE UNIVERSITY

26

Mountains & Minds

Shakey vs. Polly

Goal: To demonstrate the principled development and use of a "layered" architecture for program development

Problem: Give tours

Environment: MIT AI lab

- "Real World Interface" B12 base with sensors
- C30 32-bit DSP (Pentek 4283)
- B/W Frame-grabber (Data Translation DT1451)
- B/W Video surveillance camera (Chinon CX-101)
- Voice synthesizer (RC Systems)

Polly, ca. 1994

MONTANA STATE UNIVERSITY

27

Mountains & Minds

Shakey vs. Polly

Shakey, ca. 1969

Architecture:

- Low level motion primitives (e.g. move 2 meters)
- Grid-based mapping scheme
- Line-based object recognition
  - adequate for simple polygons
- Problem solving
  - "logical" rules describing the effects of actions
  - Goal phrased as a statement to "prove"
  - Proof yields recipe for action
- "Exceptions" start the process over again

Shakey vs. Polly

Architecture:

- Layers of pre-emptive control
  - low level obstacle avoidance
  - wandering
  - corridor following
  - tour
- Topological mapping scheme
- Context dependent vision for low-level navigation
- Context dependent exception mechanisms for getting "lost"

Polly, ca. 1994

Shakey vs. Polly

Shakey, ca. 1969

The plusses:

A complete robot system

Highly principled, general design

The minuses:

Simple environment

Largely deterministic

Largely accessible

Episodic

Static

Continuous

How do we scale to reality?

Shakey vs. Polly

The plusses:
- Real-time, interactive
- Completely autonomous, robust
- Real world
  - Non-deterministic
  - Inaccessible
  - Episodic
  - Dynamic
  - Continuous

The minuses:
- A simple task
- Context dependent techniques
- How do we scale to complexity?

Polly, ca. 1994

MONTANA STATE UNIVERSITY

Game Playing

- Extremely large state spaces.
- Involve interaction of multiple, competing agents.
- Desire to “win” forces optimization.
- Games studied:
  - Checkers
  - Chess
  - Backgammon
  - Card games
  - War games

MONTANA STATE UNIVERSITY

Automated Reasoning

- Attempts to derive new knowledge from old.
- Prove mathematical theorems about the world.
- Strong logic/mathematical foundation.
- Typically considered “hard” for computers to do.

MONTANA STATE UNIVERSITY

Expert Systems

- Focused on specific problem areas.
- Capture “rules” and “heuristics” used by humans to solve problems.
- Attempted to retain expert knowledge in midst of expert attrition.
- Some famous expert systems:
  - MYCIN (infectious disease diagnosis)
  - DENDRAL (determine organic molecular structure)
  - PROSPECTOR (project location and type of ore deposits)
  - XCON and R1 (configure VAX computers)

Natural Language Processing

- Understand content of written and spoken text.
- Continuous speech processing.
- Dialogue and discourse processing.
- Story interpretation.
- Semantics captured through advanced knowledge representation schemes (e.g., semantic networks and augmented transition networks).
- Statistical language learning yielding significant results.
- Strong relationships between NLP and other problems areas (e.g., image processing, hardware design)

Planning/Robotics

- Given
  - Description of current state
  - Description of goal state
  - Specification of available actions
- Find
  - Sequence of actions taking agent from current state to goal state (often optimizing some cost criterion).
- Types of planners
  - Reactive planners
  - Deliberative planners
  - Hierarchical planners
  - Stochastic planners

Machine Learning

- Supervised learning
  - Agent learns from experience with a teacher providing correct behavior.
- Unsupervised learning
  - Agent learns by looking for self-consistent, internal patterns in data/experience.
- Semi-supervised learning
  - Agent learns from a combination of labeled examples and unlabeled examples
- Reinforcement learning
  - Agent learns from experience by receiving cost/reward feedback from the environment.

Knowledge Representation

- Logical rules
- Mathematical models
  - Structural models
  - Behavioral models
  - Statistical models
- Semantic models
  - Semantic networks
  - Ontologies
- Biological models
  - Neural networks
  - Genetic algorithms
  - Fuzzy logic

Philosophy of AI

- What is intelligence?
- What is thinking?
- Can a machine ever be “truly” intelligent?
- What is “true?”
- Is intelligence innately human? Is it just behavioral?
- Is intelligence “emergent?”

State of the Art

- So which of the following are currently possible?
  - Play a decent game of table tennis.
  - Drive along a curved mountain road.
  - Drive in the center of Cairo.
  - Play a decent game of bridge.
  - Discover and prove a new mathematical theorem.
  - Write an intentionally funny story.
  - Give competent legal advice in a specialized area of law.
  - Translate spoken English into spoken Swedish in real time.

MONTANA STATE UNIVERSITY

Mountains Minds

Lessons

- Making the problem too simple leads to interesting solutions in uninteresting situations
  - simple environment
  - simple task
- Generality is in the eye of the beholder
  - general architecture
  - generalizable approach
- Neat vs. scruffy
  - more important that it's "clean" and we can prove properties
  - more important that it works, even if it has a few "hacks"
- Once understand how it works, it doesn't seem like "AI"

MONTANA STATE UNIVERSITY

Mountains Minds

RoboCup—Medium Size

MONTANA STATE UNIVERSITY

Mountains Minds

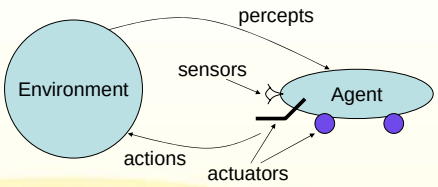
RoboCup—Small Size

MONTANA STATE UNIVERSITY

Mountains Minds

Agents

- **Def:** An *agent* is anything that can *perceive* its environment through *sensors* and *act* upon that environment through *effectors*.



MONTANA STATE UNIVERSITY

Mountains Minds

Rationality

- **Def:** A system behaves *rationally* if it "does the right thing."
  - Does not need to be optimal.
  - Does not need to be human-like.

MONTANA STATE UNIVERSITY

Mountains Minds

Rational Agents

- **Def:** A *rational agent* is an agent that "does the right thing."
- Requirements:
  - Performance Measure.
  - Percept Sequence (history)
  - Knowledge about Environment
  - Actions Available in Environment

MONTANA STATE UNIVERSITY

Mountains Minds

Ideal Rational Agents

- **Def:** An *ideal rational agent* is an agent that, for each possible percept sequence, performs an action that maximizes its expected performance on the basis of the evidence provided by the percepts and built-in knowledge that the agent possesses.
- **Goal:** Map percept sequences to actions to maximize performance.

MONTANA STATE UNIVERSITY

Mountains Minds

Mapping

- **Def:** An *ideal mapping* is the specification of which actions should be performed by an ideal agent in response to any given percept sequence.
- How would *you* create or represent such mappings?

— Lookup Tables
— Computer Programs
— Decision Rules

MONTANA STATE UNIVERSITY

Mountains Minds

Autonomous Agents

- **Def:** An agent is *autonomous* to the extent that its behavior is determined by its own experience.
- The more an agent relies upon internal knowledge and ignores percepts, the less autonomous it is.

MONTANA STATE UNIVERSITY
Mountains Minds
49

PAGE Model

- Must first specify the setting for intelligent agent design.
- Consider the task to be performed (e.g., designing an automated taxi).
- Must identify the following four factors:
  - What are the *percepts* for the agent?
  - What *actions* can the agent perform?
  - What *goals* is the agent trying to achieve?
  - What is the *environment* within which the agent will function?

MONTANA STATE UNIVERSITY
Mountains Minds
50

PAGE Model—Medical Diagnosis

- Percepts
  - Symptoms, findings, patient's answers, test results.
- Actions
  - Questions, tests, procedures, surgery, treatment.
- Goals
  - Healthy patient, minimize cost.
- Environment
  - Patient, clinic, hospital.

MONTANA STATE UNIVERSITY
Mountains Minds
51

PAGE Model—Image Analysis

- Percepts
  - Pixels of varying intensity, color, texture.
- Actions
  - Print a categorized scene, identify an object.
- Goals
  - Correct categorization, correct identification.
- Environment
  - Images from an orbiting satellite.

MONTANA STATE UNIVERSITY
52
Mountains Minds

PAGE Model—Parts Picking

- Percepts
  - Pixels of varying intensities.
- Actions
  - Pick up parts, sort into bins.
- Goals
  - Place parts in correct bins.
- Environment
  - Conveyer belt with parts.

MONTANA STATE UNIVERSITY
53
Mountains Minds

PAGE Model—Refinery Control

- Percepts
  - Temperature, pressure, viscosity, purity.
- Actions
  - Open valves, close valves, adjust temperature, raise/lower pressure.
- Goals
  - Maximize purity, yield, safety.
- Environment
  - Refinery

MONTANA STATE UNIVERSITY
54
Mountains Minds

PAGE Model—English Tutor

- Percepts
  - Typed/spoken words.
- Actions
  - Print exercises, carry on conversations, provide suggestions and corrections.
- Goals
  - Maximize student's score on English test.
- Environment
  - Set of students, classroom.

MONTANA STATE UNIVERSITY
Mountains & Minds
55

PAGE Model—Taxi Driver

- Percepts
  - Cameras, speedometer, odometer, GPS, sonar, microphone.
- Actions
  - Steer, accelerate, brake, talk to passenger.
- Goals
  - Safe, fast, legal, comfortable trip with maximum profit.
- Environment
  - Roads, other traffic, pedestrians, customers.

MONTANA STATE UNIVERSITY
Mountains & Minds
56

Properties of an Environment

- **Accessible vs. Inaccessible**
  - Does the agent's sensors provide sufficient information to fully capture the state of the environment, or is state information only partially captured?

MONTANA STATE UNIVERSITY
Mountains & Minds
57

Properties of an Environment

- **Deterministic vs. Nondeterministic**
  - Are successor states fully determined by the current state and the action taken by the agent, or can successors be selected "at random" from a set of possible next states?
  - Note: Partial observability (i.e., inaccessibility) may make the environment appear to be nondeterministic.

MONTANA STATE UNIVERSITY
Mountains Minds
58

Properties of an Environment

- **Episodic vs. Nonepisodic**
  - Is the utility of the agent's actions dependent only upon the action taken in the current state, or is look-ahead required to determine utility?
  - Immediate vs delayed "reward."

MONTANA STATE UNIVERSITY
Mountains Minds
59

Properties of an Environment

- **Static vs. Dynamic**
  - Does the environment's state remain constant while the agent is deliberating, or can the environment's state change "on its own?"
  - If the environment is static but the utility changes with time, then the environment is *semidynamic*.

MONTANA STATE UNIVERSITY
Mountains Minds
60

Properties of an Environment

- **Discrete vs. Continuous**
  - Are the number of percepts and actions clearly enumerable, or do either percepts or actions lie within a range of possible values?

MONTANA STATE UNIVERSITY

Mountains Minds

Sample Environments

Environment	*Accessible*	*Deterministic*	*Episodic*	*Static*	*Discrete*
Time Chess	Yes	Yes	No	Semi	Yes
Untimed Chess	Yes	Yes	No	Yes	Yes
Poker	No	No	No	Yes	Yes
Backgammon	Yes	No	No	Yes	Yes
Taxi Driving	No	No	No	No	No
Medical Diagnosis	No	No	No	No	No
Image Analysis	Yes	Yes	Yes	Semi	No
Part Picking	No	No	Yes	No	No
Refinery	No	No	No	No	No
English Tutoring	No	No	No	No	Yes

MONTANA STATE UNIVERSITY

Mountains Minds

Understanding the Environment

- Helps to capture difficulty of problem.
- Helps to determine appropriate technique.
- Helps to determine appropriate utility.

MONTANA STATE UNIVERSITY

Mountains Minds

Functioning in the Environment

```
function RunEnvironment(state, UpdateFn, agents,
                        terminatep, EnvFn, PerfFn);
    repeat
        for-each agent in agents do
            percept[agent] := GetPercept(agent, state);
        end-for;
        for-each agent in agents do
            action[agent] ← EnvFn[agent](percept[agent], score);
        end-for;
        state ← UpdateFn(action, agents, score);
        score ← PerfFn(score, agents, state);
    until terminatep(state);
    return(score);
end_function;
```

MONTANA STATE UNIVERSITY

Mountains Minds

Agent Functions

- An agent is completely specified by the *agent function* mapping percept sequences to actions.
- In principle, one can supply each possible sequence to see what it does—a huge lookup table.
- Only one agent function (or an equivalence class) is considered to be "rational."
- Aim: To find a way to implement the rational agent function concisely.

MONTANA STATE UNIVERSITY

Mountains Minds

Basic Agent Structure

```
function SkeletonAgent(percept);
        static memory;

        memory ← UpdateMemory(memory, percept);
        action ← ChooseBestAction(memory);
        memory ← UpdateMemory(memory, action);
        return(action);
end_function;
```

MONTANA STATE UNIVERSITY

Mountains Minds

Lookup Table Approach

```
function TableDrivenAgent(percept);
        static percept_history, table;

        append(percept, percept_history);
        action ← Lookup(percept_history, table);
        return(action);
end_function;
```

What is wrong with this approach?

MONTANA STATE UNIVERSITY

Mountains Minds

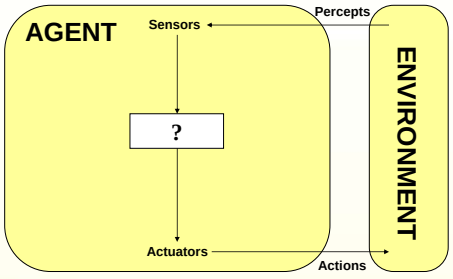
Problems with Lookup Table

- Extremely large table required for moderate-sized problem.
- Excessive time required to build table.
- Eliminates autonomy (and adaptability) in agent.
- Even with learning, determining correct values could take a long time.

MONTANA STATE UNIVERSITY

Mountains Minds

Basic Agent Structure



MONTANA STATE UNIVERSITY

Mountains Minds

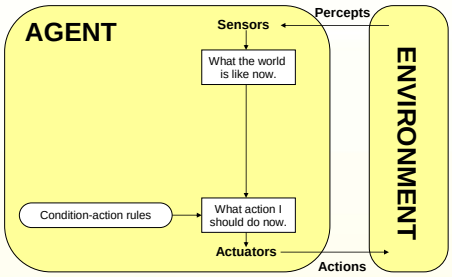
Agent Architectures

- Simple Reflex Agents
- Agents with Internal State
- Goal-Based Agents
- Utility-Based Agents

MONTANA STATE UNIVERSITY

Mountains Minds

Simple Reflex Agents



MONTANA STATE UNIVERSITY

Mountains Minds

Simple Reflex Agents

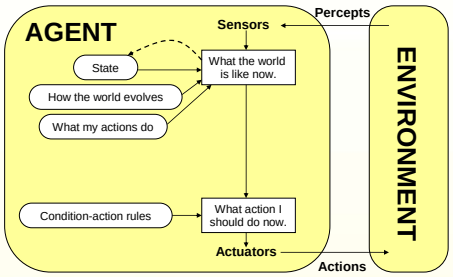
```
function SimpleReflexAgent(percept);
        static rules;    // Condition-action rules

        state ← InterpretInput(percept);
        rule ← RuleMatch(state, rules);
        action ← RuleAction(rule);
        return(action);
end_function;
```

MONTANA STATE UNIVERSITY

Mountains Minds

Agents with Internal State



MONTANA STATE UNIVERSITY

Mountains Minds

Agents with Internal State

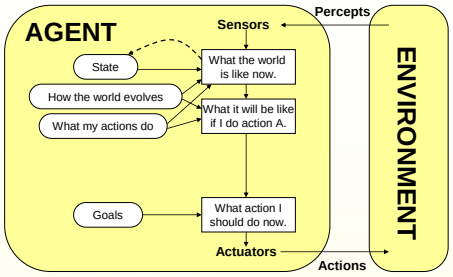
```
function ReflexAgentWithState(percept);
        static state, rules;

        state ← UpdateState(state, percept);
        rule ← RuleMatch(state, rules);
        action ← RuleAction(rule);
        state ← UpdateState(state, action);
        return(action);
end_function;
```

MONTANA STATE UNIVERSITY

Mountains Minds

Goal-Based Agents



MONTANA STATE UNIVERSITY

Mountains Minds

Goal-Based Agents

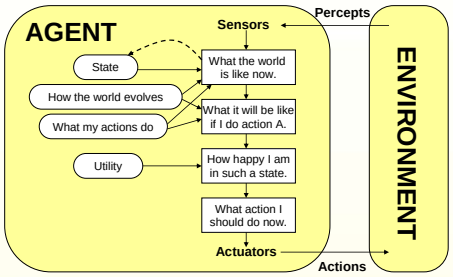
```
function GoalBasedAgent(percept, goal);
        static state, rules;

        state ← UpdateState(state, percept);
        action ← GoalActionSearch(state, goal, rules);
        state ← UpdateState(state, action);
        return(action);
end_function;
```

MONTANA STATE UNIVERSITY

Mountains Minds

Utility-Based Agents



MONTANA STATE UNIVERSITY

Mountains Minds

Utility-Based Agents

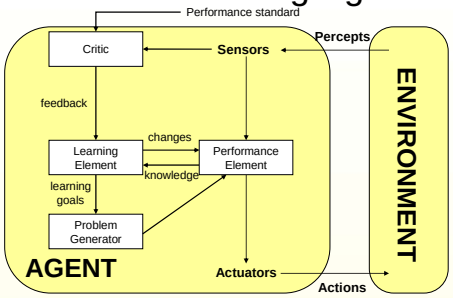
```
function UtilityBasedAgent(percept);
        static state, rules;

        state ← UpdateState(state, percept);
        acts ← GoalActionSearch(state, goal, rules);
        vals ← EvalActs(state, goal, acts, utility);
        action ← FindBestAction(acts, vals);
        state ← UpdateState(state, action);
        return(action);
end_function;
```

MONTANA STATE UNIVERSITY

Mountains Minds

General Learning Agents



MONTANA STATE UNIVERSITY

79

Mountains Minds