RESEARCHERID
THOMSON REUTERS

[COTII- 2018]

ISSN 2348 – 8034
Impact Factor- 5.070

GLOBAL JOURNAL OF ENGINEERING SCIENCE AND RESEARCHES

INTELLIGENT AGENTS: PROBLEM SOLVING TECHNIQUES

Urvi Mitra & Garima Srivastava
Amity School of Engineering and Technology
Amity University, Lucknow

ABSTRACT

Artificial Intelligence is the recent tsunami of technological innovation that the world has been hit with. It's impact on our life has been tremendous and will be path breaking in the future. AI is not something esoteric or something to be scared of, it is just making computers more intelligent, capable of solving a wider variety of problems than it already did, undertaking tasks that they never have.
It is still used, even though we fail to notice it. When we make image searches on Google and it fetches us pictures of sunset, it is AI at work. Also, auto correct predicting words for you and voice typing are all instances of AI.
At the core, all the complicated systems are based on intelligent agents, systems that take inputs from their environment and map them into outputs to achieve certain predefined goals, basically solve problems, autonomously. These intelligent agents are designed after studying and modeling human brains to simulate how they work and solve problems. This paper focuses on the problem solving techniques used by these intelligent agents

Keywords: *Intelligent Agents, Environment, Goal, Problem Solving Techniques in Intelligent Agents, Problem Solving Agents*

I. INTRODUCTION

Artificial Intelligence is a field of Computer Science, mentioned first in 1956 but evolving till date that attempts to study human intelligence and simulate it in Computer Systems. We have witnessed the far-reaching impact this field has had in our life till now and can imagine how massively it can change our lifestyles in the coming decade. Be it driverless cars, robots that automate mundane, repetitive tasks and home assistants that take voice commands, it is changing the face of computers from machines that just follow instructions to machines that take their own decisions, much like humans do.
This paper attempts to study how intelligent agents that carry out tasks automatically for us, the unit of artificial intelligence are built. It also tries to look at how human cognition is studied as a basis and subsequently modeled to design such agents.

1. Introduction to agents

Agents
An agent is an entity that senses or perceives its environment through **sensors** and responds to it i.e takes action on its environment through **actuators.**

A **rational agent** is one that acts in a way that achieves the goal it is designed to accomplish. How successful is an agent depends on what it is designed for. A **performance measure** is determined for each agent. If it meets the performance measure and achieves its **goal**, it is said to be successful.

An **ideal rational agent** takes the appropriate action necessary to maximize its performance measure and reach it on the basis of its previous perceptual experience and its built-in knowledge and code.

An **autonomous agent** has built-in knowledge and also learns or acquires knowledge through its percepts and experience. It uses the two to take appropriate action. These agents are called **intelligent agents.**

Real life examples of agents include humans, their sensors being eyes, ears, skin, taste buds, etc. i.e. their sense organs through which they perceive their environment. Their effectors include hands, fingers, legs, mouth, etc. that enable them to take action. Another example of an agent is a robot, whose sensors are its camera, infrared, bumper, etc. and effectors are grippers, wheels, lights, speakers, etc.

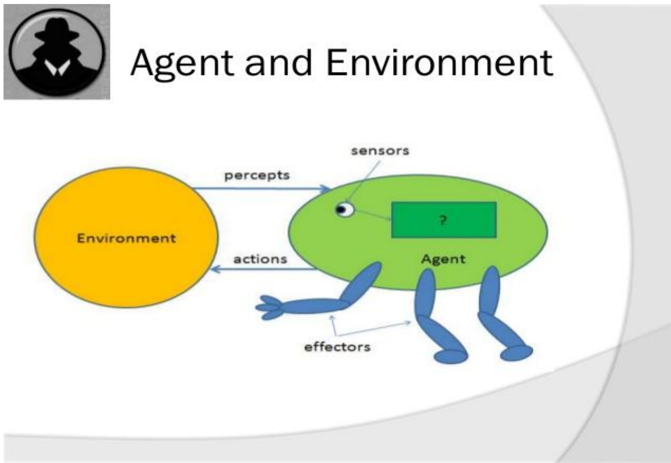


Fig 1: How an agent operates

Components of an agent

An agent comprises of an **architecture** i.e. how its parts are structured and designed and a **program** that makes the architecture do stuff.

From the previous discussion, it is clear that an agent is comprised of the following components:

- **Environment**, that an agent acts on and interacts with
- **Percepts,** the set of inputs that the agent has perceived from its environment through-
- **Sensors**
- **Action,** the appropriate output of the system generated on the bases of percept sequence and the program, through-
- **Actuators or Effectors**
- **Goals,** that the agent is expected to accomplish

Agent= Architecture + Program
Agent Architecture= Sensors + Actuators
An **agent function** maps all possible input percept sequences to the output actions of an agent. It can be expressed mathematically as [f: P* → A]. And agent program's task is to produce the agent function.

AGENT DESCRIPTORS

An agent is described/specified using two shorthand notations- PAGE descriptor and PEAS descriptor. These stand for the following-

- **PAGE DESCRIPTOR**
  P- Percepts
  A- Actions
  G-Goals
  E- Environment

- **PEAS DESCRIPTOR**
  P- Performance
  E- Environment

A- Actuators
S- Sensors

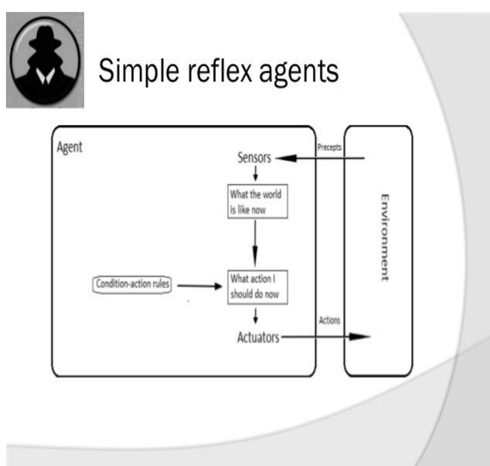
2. **Types Of environment**

- **Accessible v Inaccessible**: An accessible environment is one whose complete state can be determined using the agent's sensory apparatus. The agent would have to maintain an internal state corresponding to its perception of the environment. On the other hand, inaccessible environments can't be determined using an agent's sensors.
- **Deterministic v Non-deterministic:** A deterministic environment is one whose next state is determined by its current state and its agent's actions. A deterministic and inaccessible environment may look like non-deterministic to the agent. Most real environments are complex and have to be treated as non-deterministic in practice.
- **Discrete v Continuous**: A discrete environment is one which has a limited or finite number of distinctive, clearly defined percepts as well as actions.
- **Episodic v Non-episodic**: In environments termed as episodic the agent's experience can be divided and subdivided into "episodes" containing the agent perceiving and producing actions depending only on that episode. Subsequent episodes don't depend on previous episodes, and so the agent can limit how further it does need to think.
- **Static v Dynamic**: A static environment is unchangeable while an agent is deliberating. The agent will have no need then to keep checking the outside world while it is deciding on an action and does not need to worry about the passage of time
- An environment program is used to simulate various types of environments during testing of agents.

3. **Types of agents**

SIMPLE REFLEX AGENTS

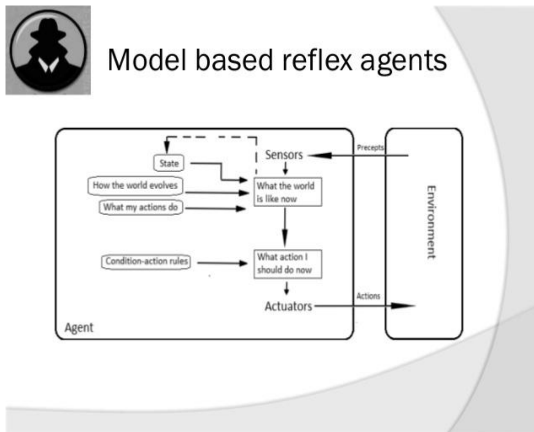
- Simple reflex agents act on the basis of the current percept, ignoring the rest. They are based on the condition- action rule: if condition then action.
- It would be impossible, except for the most trivial of situations, to find the appropriate action for a given set of percepts because of the large number of entries such a table would require. However, one can often summaries and enlist certain parts of the look-up table by noting commonly occurring input/output associations which can be written as condition-action rules.
-



ISSN 2348 – 8034
Impact Factor- 5.070

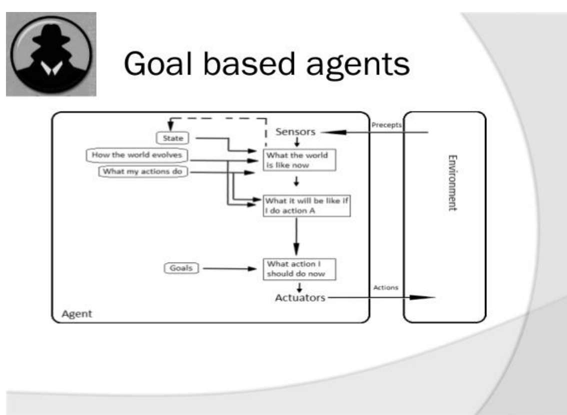
MODEL BASED REFLEX AGENTS

- Reflex agents also needed to keep a track of all its previous states. Along with the knowledge of how the world evolves, the knowledge of its previous history would give it more information about its current state. Thus, it would be able to better decide on its actions. This was made possible by model based reflex agents, an improvement on simple reflex agents.
- Such reflex agents have an internal state. They also have access to information and not currently available to their sensory systems. This makes their actions more "intelligent" than simple agents.



GOAL BASED AGENTS

- The right action to be taken by the agent not only depends on its current state and the state of the world, but also on the goal it needs to achieve. Goal based agents are designed such that they need to be given goal information.
- These agents can determine the consequences of their actions using the knowledge of how the world evolves and how their actions affect the world, their current state, etc. It can compare each of these consequences against its stated goal to determine which action(s) achieves the goal, and hence which action(s) is the most appropriate and should be taken.

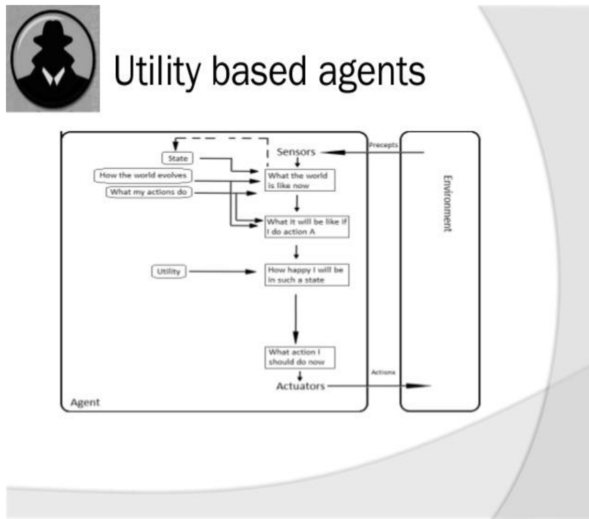


UTILITY BASED AGENTS

- A utility based agent is described as having a utility function that maps a state, or a sequence thereof, on to a real number which represents its utility or usefulness. This utility can be used to choose among the alternate sequences of actions or states that would achieve the goal.
- If one or more goals are conflicting i.e. only some, not all, can be achieved, the utility can be used to measure the appropriate trade-offs. When there are several goals, among which none can be achieved with

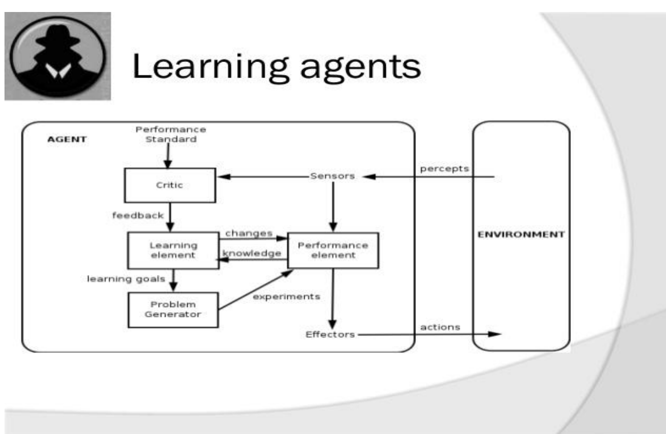
ISSN 2348 – 8034
Impact Factor- 5.070

full probability, the utility is used to measure the probability or likelihood of the fulfillment of these goals and relate and compare it with their importance.



LEARNING AGENTS

- The most powerful out of all these agents are learning agents. This is because they have the capability to learn by actively exploring, discovering and experimenting with the outside environment.They are able to decide appropriate actions using knowledge of the environment which they gained themselves. This distinguishes them from the others. A generic learning agents has the following four components
- **The Performance Element**: It takes percepts as inputs and decides appropriate actions in a way a non-learning agent would do.
- **The Critic** : It uses a definite, fixed standard of performance. This measure of performance is conveyed to the learning element so it knows how the agent is performing
- **The Learning Element** : This component receives information about performance from the critic and makes the desired improvements to the performance element.
- **The Problem Generator:** This component suggests actions leading to new, better and more informative experiences that'd help the agent learn (e.g. like carrying out experiments).



4. Problem solving in intelligent agents

Formulation of Problems

There are certain features to consider in order to formulate problems.

- The possible world states
- **Accessibility**: the agent can determine via its sensors in which state it is
- **consequences of actions:** the agent knows the results of its actions
- **levels**: problems and actions can be specified at various levels
- **constraints** i.e. theconditions that influence the problem-solving process
- **performance measures** to be applied
- **costs** and the utilization of resources

Example: vacuum world problem

Problem

Problems encountered by intelligent agents are broadly classified into four types.

- single-state problem
- multiple-state problem
- contingency problem
- exploration problem

Single-State Problem

Single state problems have the following features

- exact prediction is possible in these problems
- After any sequence of actions, the **state** is precisely known.
- **Accessibility:** All essential information about the world can be obtained through sensors.
- **consequences of actions** are known to the agent
- **goal:** for each known initial state, there is a unique goal state that is guaranteed to be reachable via an action sequence
- This is the simplest case, but it is severely restricted
- Example: Vacuum world problem
- Limitations: It can't deal with incomplete accessibility and incomplete knowledge about consequences I.e indeterminism

Multiple-State Problem

- In such problems, semi-exact prediction is possible
- **state** is not known exactly, but they are limited to a set of possible states after each action
- **accessibility of the world**: Not all essential information can be obtained through sensors. Reasoning can be used to determine the set of possible states .
- **consequences of actions** are not always or completely known to the agent; actions or the environment might exhibit randomness.
- **Goal**: Due to ignorance, there may be no fixed action sequence that leads to the goal.
- It is less restricted, but more complex
- Example: Vacuum world problem
- Limitations: Can't deal with changes in the world during execution ("contingencies")

Contingency Problem

- In such problems, exact prediction is impossible
- **state** is unknown in advance, it may depend on the outcome of actions and changes in the environment.
- **Accessibility of the world:** Some essential information may be obtained through sensors only at execution time.
- **consequences of actions** may not be known at planning time
- **Goal**: instead of single action sequences, there are trees of actions
- **agent design** is different from the previous two cases: the agent must act on incomplete plans

- Search and execution phases are interleaved
- Example: Vacuum world
- Limitations: It can't deal with situations in which the environment or effects of action are unknown.

Exploration Problem

- The effects of actions are unknown.
- **State:** the set of possible states may be unknown
- **Accessibility of the world**: Some essential information may be obtained through sensors only at execution time.
- **Consequences of actions** may not be known at planning time
- **Goal** can't be completely formulated in advance because states and
- **Consequences** may not be known at planning time
- **Discovery**: what states exist
- **Experimentation:** what are the outcomes of actions
- **Learning**: remember and evaluate experiments
- **agent design:** Different from the previous cases: the agent must experiment
- **search**: It requires search in the real world, not in an abstract model.
- These are realistic problems and very hard.

Well-Defined Problems

- There is exact formulation of problems and solutions in such problems.
- **initial state**: current state / set of states, or the state at the beginning of the problem-solving process must be known to the agent
- **state space**: The set of all states reachable from the initial state by a possible sequence of actions.
- **Path:** the sequence of actions between two states
- **goal test** determines if the agent has reached a goal state
- **path cost** A function that assigns a cost to a path usually the sum of the costs of actions along the path
- **components**: Initial-State, Operators, Goal-Test, Path-Cost
- **solution**: Path from the initial state to a state that satisfies the goal test
- **search algorithm** takes the problem data type and computes a solution

Performance Measuring

Parameters to measure the performance of the agent's problem solving.

- **Success**: Has a solution been found?
- **Quality:** Is it a good solution? What are the criteria?
- **optimal solution** may be difficult to find and not necessary
- **cost** i.e. the sum of
  - search cost (time, resources to find a solution)
  - path cost (as defined above)

Example Problems

Toy Problems

There are certain sample toy problems solved by intelligent agents.

- vacuum world
- 8-queens
- 8-puzzle
- missionaries and cannibals

Real-World Problems

Intelligent agents find application in solving problems in the real world, for instance

- **route finding:** travel advisory, computer networks, airline travel

- **travelling salesperson problem:** each city must be visited exactly once. It is more complex than route finding
- **VLSI**: layout ie positioning of gates and connections that are too complex for humans and crucial for successful operation and costs
- **robot navigation:** The generalization of route finding to continuous space, possibly multi-dimensional (actions involving arms).

Search

The mapping of a problem to a solution action sequence called search involves the following steps.

- Examine possible sequences of actions
- Define input problem description and initial state
- Define output solution as an action sequence
- Define search space I.e. set of all possible action sequences

Search in AI

- search of a problem space for a solution to a problem is through data structures
- The basic idea is to find a path from the initial description of a problem to a description of the solved problem
- The problem space is created incrementally, not predefined and already in existence

Problem Space

Representation of a problem space is in the form of

- **Network graph** with nodes as states and arcs as possible steps
- **Tree** multiple representations of states

Search Methods in AI

- **Depth First Search** systematically expands each path to the end, backtracking when a dead end is encountered
- **Breadth First Search**: all nodes at one level are expanded and it finds the shortest path
- **Beam Search** is a directed, heuristic variation of breadth-first where only a limited number of nodes are expanded and all successor nodes are evaluated, the best ones are selected for expansion
- **Hill Climbing** is a directed variation of depth-first successor node with the greatest progress towards the goal

  Branch and Bound is a directed search where the most promising node in the tree is selected
- **Best-First** is directed, heuristic search algorithm that requires estimate of the distance to the solution and selects the node with the smallest estimate
- **A* (A-Star)**

➢ It is the combination of best-first and branch and bound technique
➢ requires estimate of the distance to the solution
➢ It uses estimate and previous path length to calculate the cost
➢ If estimates are always greater than zero but never greater than the actual cost, the lowest cost path will be found
➢ It reduces the number of nodes expanded by best-first

Solution

Solution can be defined by the following parameters.

- **Action sequence** that satisfies the goal
- **Validation:** Does the solution really achieve the goal?
- **Verification**: Is the sequence of actions admissible?
- **Feasibility:** With the available resources, can the sequence of actions be carried out?
- **Execution:** actions are carried out

5. **What do agents have to offer?**

- the ability to solve problems that have hitherto been beyond the scope of automation – either because no existing technology could be used to solve the problem, or because it was considered too expensive (difficult, time-consuming, risky) to develop solutions using existing technology; or
- the ability to solve problems that can already be solved in a significantly better (cheaper, more natural, easier, more efficient, or faster) way.

Applications of intelligent agents

Intelligent agents (broadly speaking, artificial intelligence) has made a foray into the following areas-

- Manufacturing
- Air Traffic Control
- Information Management
- Ecommerce
- Business Process Management
- Health
- Entertainment
- Games
- Interactive Theatre and Cinema

II. CONCLUSION

Intelligent agents and multi-agent systems emerge as a key enabling technology in building open, large-scale, and distributed systems for operation in complex dynamic environments. It includes a wide range of application areas in today's knowledge society and networked economy such as dynamic e-commerce, collaborative e-business, virtual enterprises, smart environments, and web/grid service-oriented applications. Intelligent software agents with their abilities of autonomous operations, learning, adaptation, interactions and cooperation offer many benefits to individual users and organizations. It includes the automation of routine tasks and business processes, adaptive decision and collaboration support, and coordinated service and resource sharing within and across organizations, leading to new efficiencies, productivity gains and better services for individuals and businesses.

The current research on intelligent agents focuses on autonomous negotiation, decision-making, adaptation and coordination mechanisms for agent systems situated in open dynamic environments characterized by the presence of changing, incomplete and uncertain information.

The popular research areas include-

- Automated negotiations and collective decision-making
- Distributed learning and adaptation in multi-agent systems
- Dynamic interactions and organizational mechanisms
- Intelligent social agents and service-oriented agents

III. ACKNOWLEDGMENT

I wish to expresss my most sincere and profound gratitude to**, Ms. Garima Srivastava, Assisstant Professor, Computer Science and Engineering Department, ASET, Amity University Lucknow,** for her guidance throughout the work. I I am also grateful to her for cooperation in providing with all required resources and for giving me inspiration and a chance to show my capabilities. I extend special thanks to my friends and family for their constant support.

REFERENCES

1. http://www.mind.ilstu.edu/curriculum/ants_nasa/intelligent_agents.php
2. https://www.researchgate.net/publication/298082655_Bottom_Up_Approach_of_Modeling_Human_Decision_Making_for_Building_Intelligent_Agents

ISSN 2348 – 8034
Impact Factor- 5.070

3. Stuart J. Russell, Peter Norvig Artificial Intelligence: A Modern Approach (International Edition)
4. GF Luger - 2005 - Pearson education, Artificial intelligence: structures and strategies for complex problem solving