Chapter 4: Planning.

- The task of coming up with sequence of action that will achieve a goal is called PLANNING.
- Planning comes into picture when agent cannot perform 2 task at the same time, if task have same importance level so in such case agent have to put task in a sequence to accomplish the target.
- In every planning problem we have information about the initial state of agent, goal condition of agent and set of action of an agent.
- Aim of an agent is to find the proper sequence of action to be followed from Starting state to Goal state in order to produce an efficient solution.

- Using Searching Agent can find path to reach goal, but if in between agent has multiple option of doing multiple action but he can do only one of them at a time, then planning comes into picture.
- Example: A Automatic Car on Sensing Petrol Pump in path can either stop or bypass Petrol Pump, which action it will perform will depend on some condition ? [Level of Petrol in tank].
- Example : In Wampus World Agent at particular block,agent can move in all direction, but which direction to move will depend on some condition.
- In short a plan is feed into agent during agent construction, which allow him to take decision based on condition.
- PLANNING IS TYPE OF REASOING BEHIND ACTION SELECTION.

Explain planning problem OR problem faced by planning agent?(Univ Q)

- It is defined as problem faced by ordinary Problem Solving Agent using algorithm like DFS,A* etc in large real world problem i.e problem faced by agent due to lack of planning.
- These problem are of 3 types.

1. *Irrelevant Action.*
2. *Finding Heuristic Function.*
3. *Lacking Problem Decomposition.*

1.Irrelevant Action.

- The irrelevant action can put agent in infinite loops while searching for required action.
- Irrelevant action are action which are logically correct but these action make no sense i.e useless action.

Example : Suppose in a store, all Greeting Cards are stored alphabetically on rack & agent want to find a greeting card having label DV43, so agent should no go and check for each card as this irrelevant action will make search process long.

BETTER WILL BE AGENT HAS INFORMATION OF RACK THEN HE CAN PLAN TO VISIT RACK “D” DIRECTLY.

2. *Finding Heuristic function:*

- Heuristic function helps agent to achieve goal efficiently and helps in saving time and effort.
- Suppose agent want to search particular 4 books from collection of 1 Million book, so here Heuristic function will be “Number of book remaining to be brought”
- But if agent given a task of getting 4 particular book then defining Heuristic function needs planning.

- Here heuristic function will take care of which book and how many books need to be brought.
- Problem comes here is that human need to define heuristic function for each of the 4 particular book as well as overall heuristic.
- Thus defining heuristic function where goal itself is conjunction of many subgoals is very difficult.

3.Lacking problem decomposition.

- The problem solver might be inefficient because it cannot take advantage of problem decomposition.
- Problem decomposition helps to solve problem by divide and conquer approach.

Example: Consider the problem of delivering a set of packages (luggage) to their destination across India.

- Here planner will divide entire luggage in many small small group of luggage area-wise and then these packets will be delivered to their destination in batch.
- Here problem comes is, decomposition possible for a problem or not.

Explain how planning problem is represented with suitable example?

- The planning problem is represented using 3 parameters called as STATE,GOAL and ACTION.

A>Representation of State:

- Planner decompose the world into logical condition and represent a state as a conjunction of positive literals(proposition/sentences)
- Example: At(Parcel1,Plane1,Mumbai,Time1) ^ At(Parcel1,Plane1,Delhi,Time2) might be representation of state in Package Delivery Problem

B> Representation of goals.

- A goal(desired final state) is a partially satisfied state represented as a conjunction of positive literals.
- A propositional state(S) satisfies a Goal(G) if S contains all atoms/states which G has.
- Example: A state(S)={Rich^ Famous^ Successful ^ Happy} satisfies Goal(S)={Rich^ Famous)}

C>Representation of Action:

- An action is specified in terms of PRE-CONDITION that must hold true before its execution.
- Every action has EFFECT, when it is executed

Example:- Action[FLY(Passanger, Mumbai, Goa),

PRECOND:-At (Passanger, Mumbai)^ Plane(Passenger)^Airport_S(Mumbai)^Airport_D(Goa)

Effect:- ~At(P, Mumbai)^At(P, Goa)]

Planning problem for Spare Tyre Problem(UNIV Q)

- Consider a problem of changing a Flat(damaged) tyre.
- The goal is to fit good tyre onto axle and spare tyre back to trunk.

Solution:-

- Init(At(Flat,Axle) ^ At(Spare,Trunk))
- Goal(At(Spare,Axle))

1. Action (Remove(Spare,Trunk)
    - PRECOND :- At(Spare_T,Trunk)
    - EFFECT:- ~At(Spare_T,trunk)^At(Spare_T, Ground)
2. Action(Remove(Flat,Axle),
    - PRECOND:- At(Flat_T,Axle)
    - EFFECT :- ~At(Flat_T,Axle) ^ At(Flat_T, Ground)
3. Action(PutOn(Spare,Axle)
    - PRECOND:-At(Spare_T,Ground) ^ ~At(Flat_T,Axle)
    - EFFECT:- ~At(Spare_T,Ground) ^¬At(Spare_T,Axle)

Explain planning with search space?(Univ Q)[Ways to Plan]

- Mostly planning algorithm uses state space search for finding solution.
- In every planning problem action is described using PRE-CONDITION and EFFECT, so it is possible to search(make plan) from both direction.

A: **Forward state search space(Progression Planning)**:-Search(Plan) from initial state to final state.

B: **Backward state search space(Regression Planning)**:- Search(Plan) from final state to initial state.

A:**Forward state search space OR Progression Planning**

- In this, search process starts from problem Initial state and consider sequence of action until we find a sequence that reaches the goal state.
- As here, searching process moves in forward direction so it is also called as Progression Planning.
- Problem formulation is as below:-

a: Initial state: It is defined using positive literals(sentences).

b:Actions: Actions are applicable to a state only if its preconditions are satisfied & successor of state is generated in form of effect of action.

c:Goal test : It checks weather current achieved state satisfies the goal of planning problem.

d:Step Cost: It is cost of each state, which is generally One in most of planning problem.

Example:- ONE NOTE

Drawback:-

- It does not handle irrelevant action as it consider all applicable action.
- Without good heuristic, this search speed slows down.

B. Backward state search space(Regression Planning)

- It is similar to bidirectional search as it starts with goal state and keep on generation possible predecessors.

- This search is also called as Regression Planning.
- So basic idea here is to backtrack the scenario and find out best possibility in order to achieve that goal.
- Here we see what might be correct action at previous state to achieve this goal. Ask Question what might has been done to reach this gaol state.
- This approach is not feasible when there is large number of states which satisfies goal.

- The main advantage of this search is it allows us to consider only relevant actions.
- Here any search algorithm can be used and algorithm terminates when predecessor states becomes same as initial state.
- Example:- For College Exam you can use Backward Planning but for CAT/GRE not

Note: Without good heuristic neither of above search is efficient.

TYPES OF PLANNING

- Total order Planning(TOP)
- Partial Order Planning(POP)
- Hierarchical Task Planning
- Conditional Planning

Partial Order planning(UNIV Q)

- Problem with “total order planning” is that they follow linear sequence of action connected from start state to goal state so they cannot take advantage of decomposition.
- There is need for an approach where we can work on several subgoals independently when we combine all these subgoals we will achieve our main goal.
- Here planner can divide main goal into some subgoals and execute them, provided these subgoals are independent.
- Example: Putting a pair of shoes, the planning can be as below

- Any planning algorithm that can place 2 or more action in different set of action without specifying which action set to be executed first is called "Partial Order planning".
- Components of partial order planning:-

a. A SET OF ACTION THAT MAKE UP THE SET OF PLAN(SUB-GOAL): These are taken from set of action in planning problem. Every plan has START with no precondition and FINISH with no effect.

b. THE SET OF ORDERING CONSTRAIN: Ordering constrain specify what action must be taken before some other action. It is written as A { B and read as “A before B”. Example: SENDING MESSAGE { Reading Message

c. A SET OF CASUAL LINK: It is written as ( ) which indicate A achieve precondition(p) for B. Example: Right_Sock ON ----p-→Right Shoe ON.

d. A SET OF OPEN PRE-CONDITION: A precondition is said to be open it is not compulsory to be achieved by some action in a plan. Example: Carrying Umbrella in rainy Seasons.

Hierarchical Task network planning

- One of the best way to deal with complex problem is hierarchical decomposition.
- The key benefit of this technique is that computational task is reduced into smaller action at each level.
- In this planning initial plan is viewed at very high level and at each level decomposition of action take place till we reach primitive action.
- Primitive action are those action which cannot be further decomposed and agent can execute them automatically.

- Example:- BUILDING A HOUSE, here entire plan can be decomposed into 4 sub-action as

Obtain_Permission,Hire_Contractor,Do_Construction and Pay_Builder.

- The START action, provide all precondition to decomposition plan which are called as External precondition. For Example land & money are **External precondition.**
- **External effects** are preconditions of finish and they are effects of plan execution. For example House and ~Money are external effects.

Conditional planning ?

- It is a way to deal with uncertainty by checking what actually can happen in environment at predetermined points in plan.

A> Conditional planning in fully observable environment:

- Due to uncertainty some times agent cannot predict outcome of its action
- A conditional planning agent handle non-determinism(uncertainty) by building conditional steps into plan at planning time.
- These conditional steps will check state of environment at execution time to decide what to do next.
- Example:

- Vacuum Cleaner:-

1.Action[Move Left, Precondition:- At_Right,
Effect :- At_Left]

2. Action[Move Left, Precondition:- At_Right,
Effect :- (When Functional:At_Left) ^(When Non-Functional:At_Right)]

3:Action[ON LIGHT, Precondition:- Light off,
Effect:- (When electricity ^ bulb working : Light ON) ^ (when no electricity: light off)

B> Conditional planning in Partial observable environment:

- In real world partially observable environment is much more in common.
- Here agent knows only certain amount of information about the actual state.
- So agent make use of “Belief state” to decide what to do next.
- Belief state is kind of assumption that agents considers to handle hidden information of the environment.
- Belief state is constructed using AND & OR clause with the help of probability.

- In Partially Observable Environment where, agent cannot keep a track on every state.
- Example:-In vacuum cleaner e.g. if the dirt is at Right and agent knows about Right, but not about Left. Then, in such cases Dirt might be left behind when the agent is at right, so agent based on some assumption/belief/probability moves to Left to clean dirt on Left.