AIR

MCQ's

- https://www.sanfoundry.com/artificial-intelligence-mcqs-state-space-search/#:~:text=5.,available in state-space search%3F&text=Explanation%3A There are four states,goal test and step cost.
- https://letsfindcourse.com/ai-mcq-questions/artificial-intelligence-semantic-networks-mcq

Links

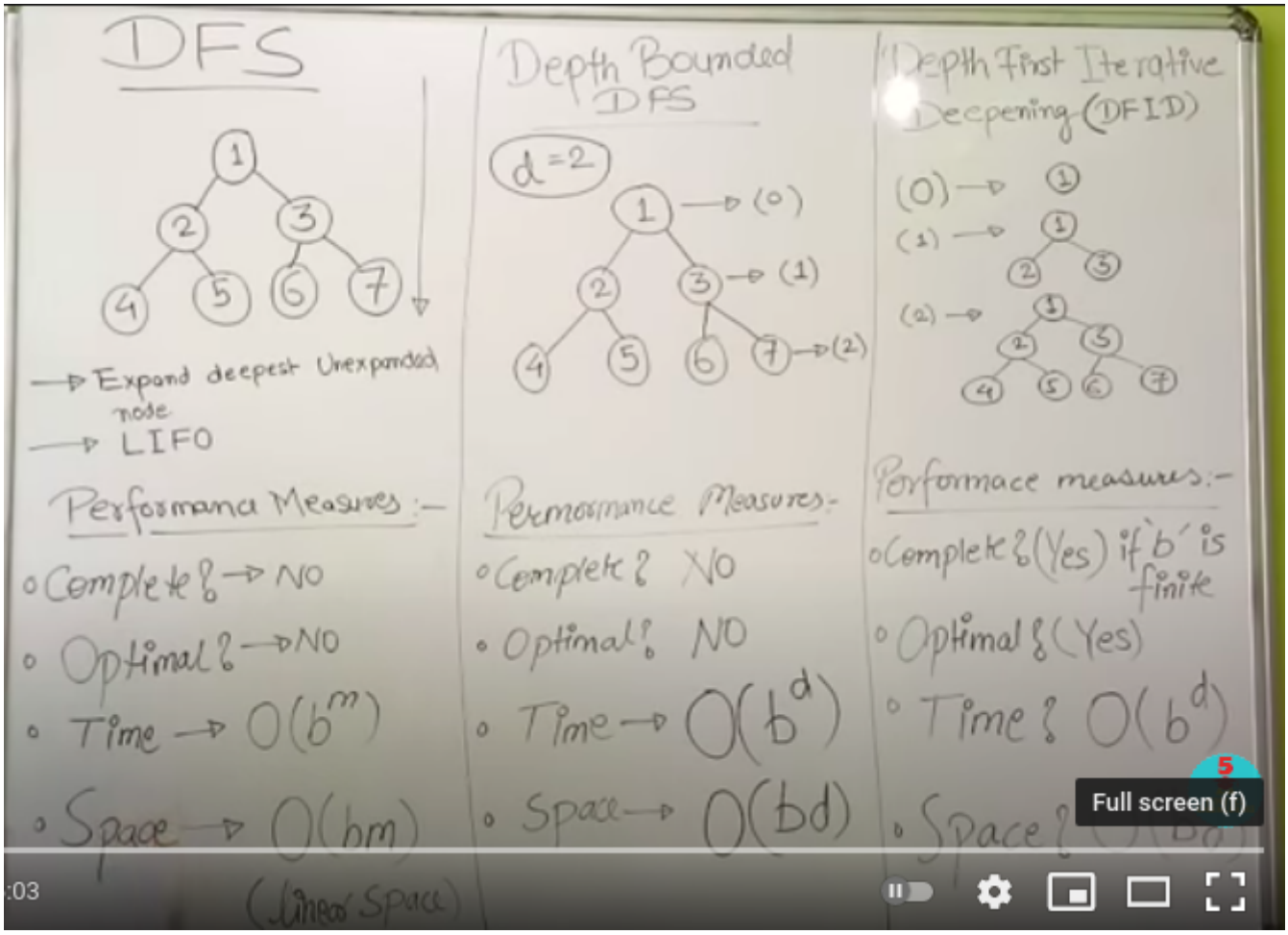
- https://sopantalekar.files.wordpress.com/2019/06/robotics.pdf
- https://www.way2techin.com/2020/07/be-comp-air-thoery-notes.html?m=1

Unit 1

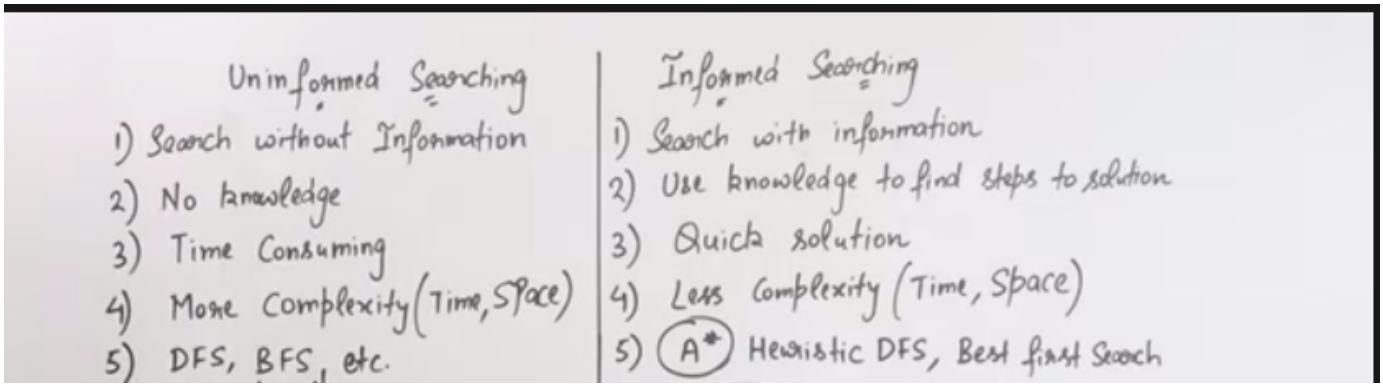
Aritificial Intellignece

- AI is defined by John McCarthy in 1956. (Dartmouth Conference)
- John defined AI as the science and engineering of making intelligent machines.
- AI sytems worls like a human brain, where a machine shows inteelligence while performing tasks, such systems are called as intelligent system or expert systems.

DFS, Depth bounded DFS, Depth first iterative deepning (DFID)

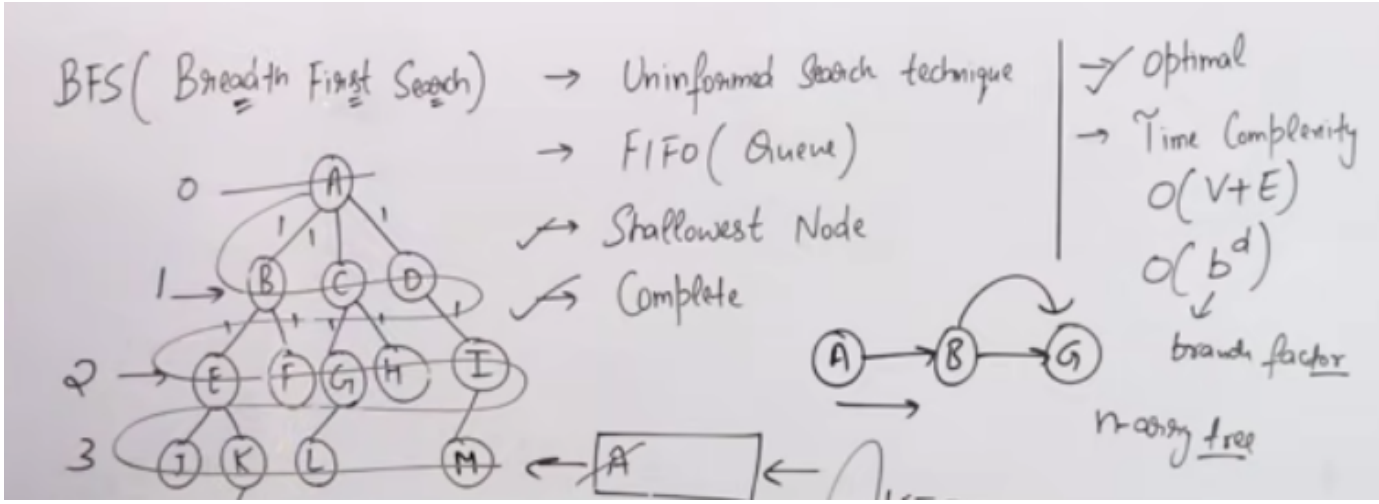


Informed vs Uninformed Search



- uninformed search gurantees the optimal solution
- informed search does not gurantee optimal solution (gives good solution)

Bread First Search



Completenes: Yes

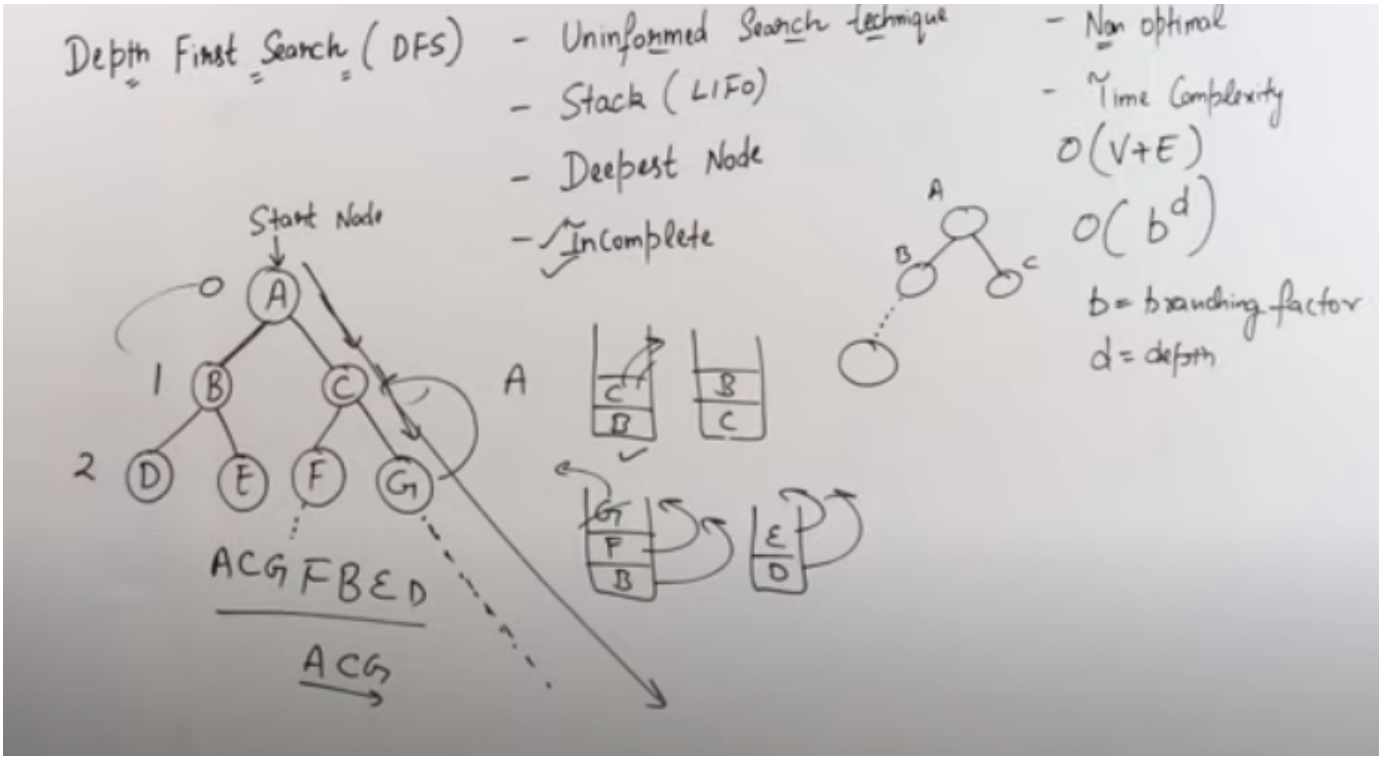
Optimal: Yes

Time: O(b^d)

where b - branching factor

d - depth

Depth First Search

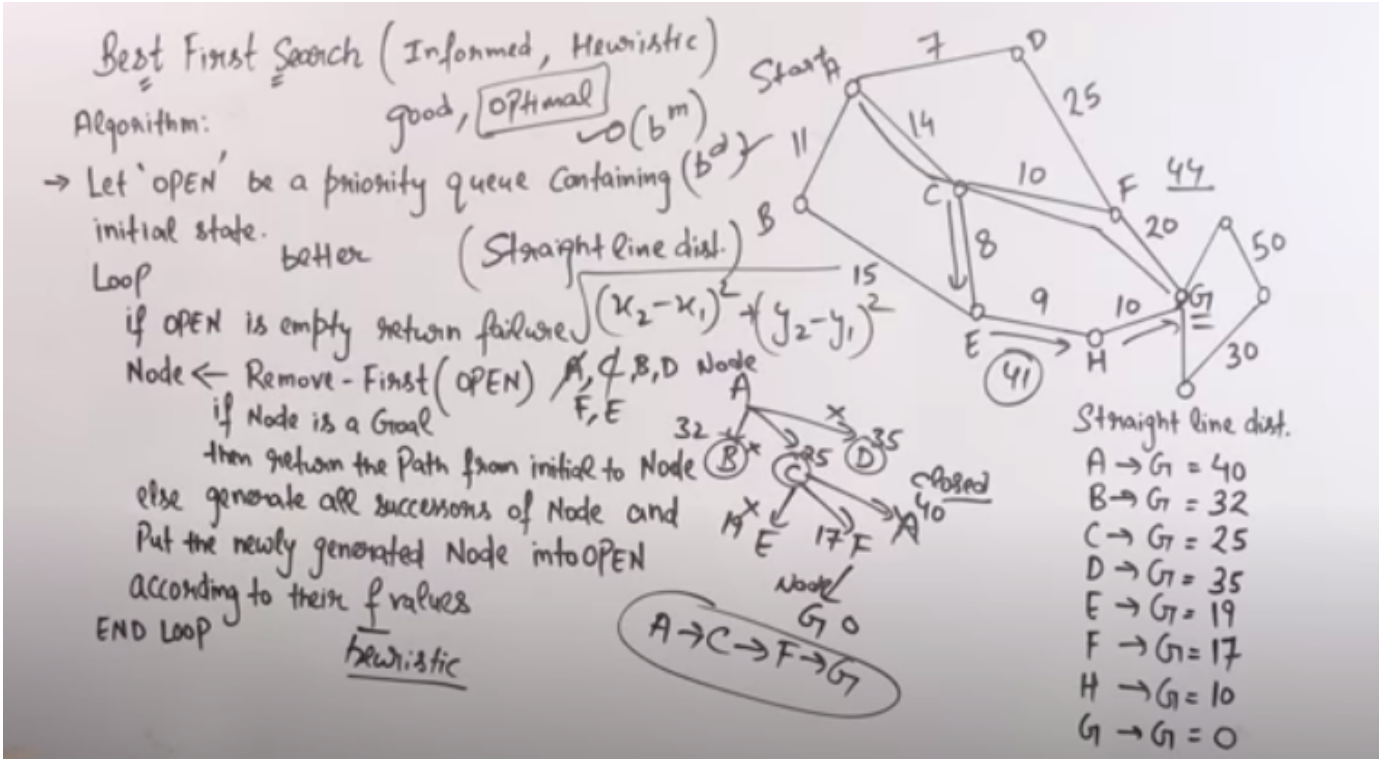


Completenes: No

Optimal: No

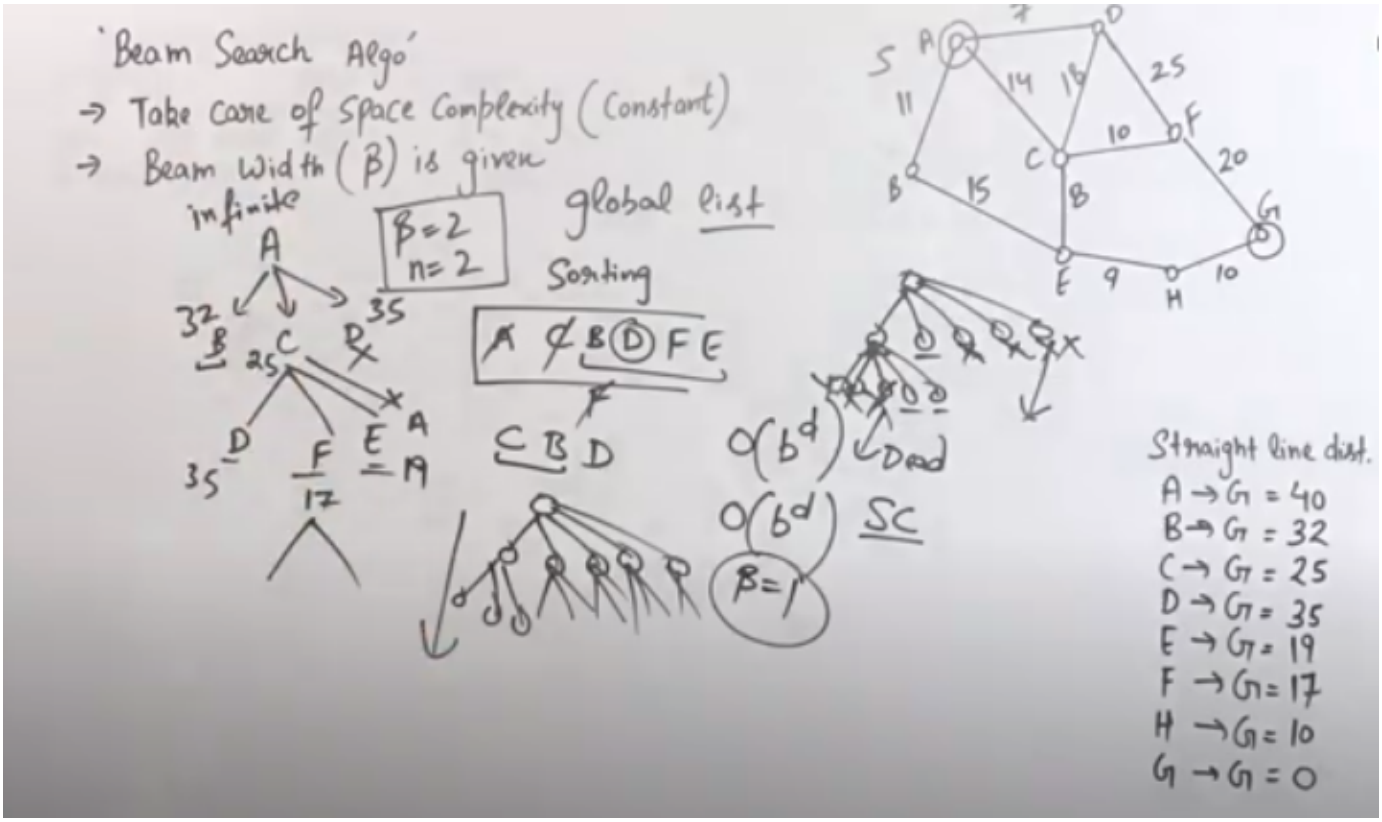
Time: O(b^d)

Best First Search



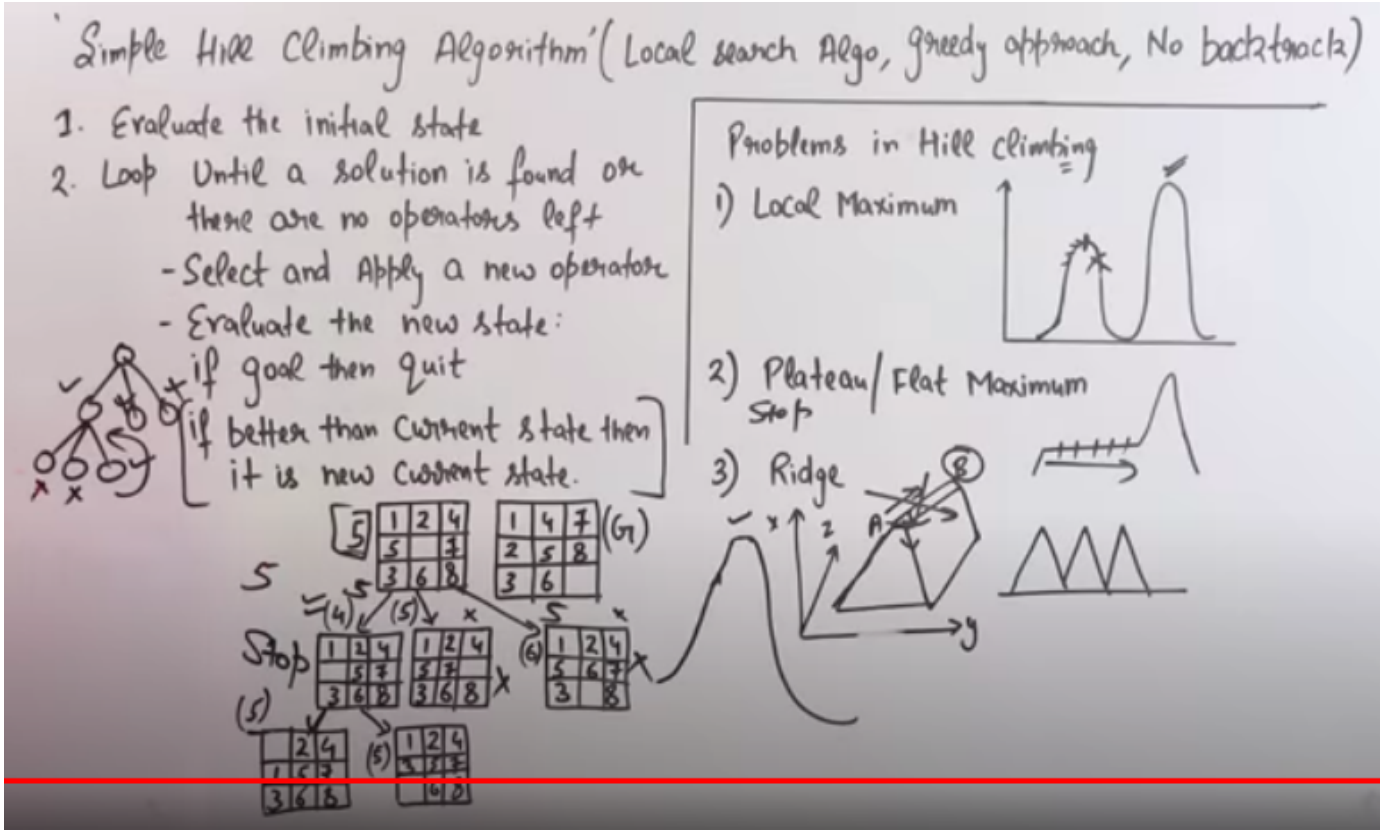
- It does not gurrantee optimal solution but it gives good solution
- Time and Space - O(b^d)

Beam Search Algo



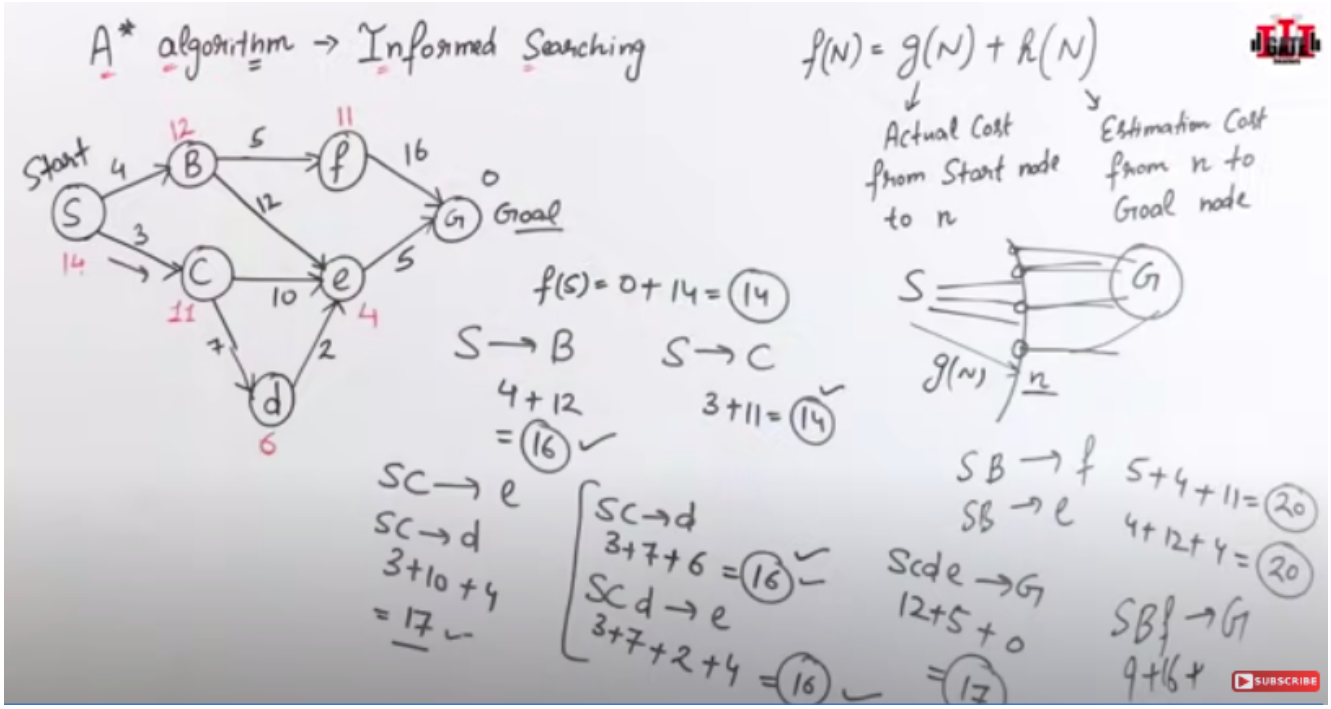
- Extension of Best first search
- In best serach we are keeping the unexplored nodes also in the priority queue and those nodes are participating while sorting by heuristic value so time complexity is large in best firs search.
- Here Beam search algorithm B value is given in the problem which indicates that only that no of best nodes we keep in the priority queue.
- Improved space complexity. Space complexity is constant.
- Cannot gurantee the completeness

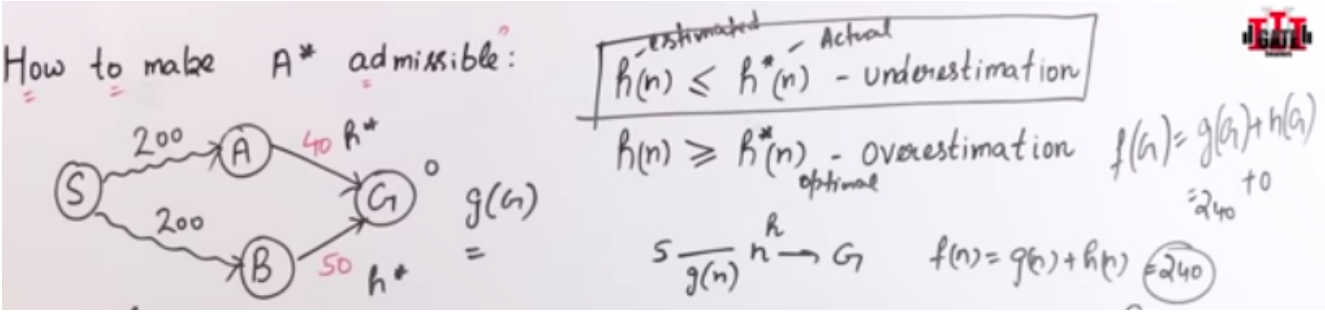
Hill climbing algorithm



- Beam value is 1

*A Algorithm**

- 

- 

- In case of overestimation we cannot gurantee the optimal solution but if we underestimate then we guranteed get optimal solution

Important Points

- States Space Search
  - Forward - Progression Planning
  - Backward - Regression Planning
- greedy best serach - pure heuristic search
- best first serach using huristic - Judea Pearl
- A* algorithm - Peter Hart, Nils Nilsson and Bertram Raphael
- B* algorithm - Hans Berliner
- admissible gurante to return an optimal solution if exists - A* algorithm is admissible
- BFS used for: shortest path between two nodes, bipartitteness, GPS navigation
- DFS used for:path finding

Unit 2

Problem Decomposition

- Problem decompostion is dividing the problem into sub problems and then finding the solution for sub problems and from the solution of the subproblems we get the solution to our main problem by combining the sub problem solutions.
- Problem decomposition is problem solving stretegy
- Ex: Necklace Problem

Goal trees:

- A goal tree is a tree with nodes represents the goals or subgoals and the links represents the set of actions to be performed in order to reach that goal.

Rule based Expert systems:

- Expert system has two main components:
  - Knowledge Base
  - Reasoning
- Every expert system is developed for a specific task domain

- taks is nothing but the some goal oriented activity
- domain is the space in whcih that task or activity is performed.
- Knowledge base in the experst sytems is the heart of expert system and reasonign is the brain
- Two types of knowledge system:
  - Factual Knowledge: factual knowledge is the knowledge which is accepted globally as a standard. This knowledge is verified by researchers and experts.
  - Heuristic Knowledge: heuristic knowledge is experimental and not approved publically.
- Knowledge base of expert system increases over time period. It learns from the experirces over time.
- Inference engine uses chaining of IF...THEN rules for the reasoning.
  - Forward chaining: In forward chaining the system is given with some conditions and by following them it moves towards the conclusion. It is data driven approach.
  - Backward chaining: Here initial state and goal state is given and the system has to find the path between initial and goal state. It is goal driven approach.
- Knowledge is most of times incomplete and uncertain.
- Simplest techinue to deal with uncertain knowledge is to associate the weight or confidence with each rule.
- The set of methods used to represent the uncertain knowldege in combination with uncertain data in the reasoning process is called as reasoning with uncertainty.
- Fuzzy logic provides methods for reasoning with uncertainty and the system that uses it are called as fuzzy systems.

Planning

- Planning involves the decision making technique to reach the target.
- Planning in an acivity where intelligent system come up with some sequecene of actions to reach the target.
- planning language is called as action language.

STRIPS:

- Standford Reasearch Institute Problem Solver

ADL

- Action Description Language
- It is an advenacement over STRIPS.

Difference between STRIPS and ADL

☞ Comparison between STRIPS and ADL

Sr. No.	STRIPS language	ADL
1.	Only allows positive literals in the states, **For example :** A valid sentence in STRIPS is expressed as ⇒ Intelligent^ Beautiful.	Can support both positive and negative literals. **For example :** Same sentence is expressed as ⇒ ¬Stupid ∧ ¬Ugly
2.	Makes use of closed-world assumption (i.e. Unmentioned literals are false)	Makes use of Open World Assumption (i.e. unmentioned literals are unknown)
3.	We only can find ground literals in goals. **For example :** Intelligent ∧ Beautiful.	We can find quantified variables in goals. **For example :** ∃x At (P1, x) ∧ At(P2, x) is the goal of having P1 and P2 in the same place in the example of the blocks.
4.	Goals are conjunctions **For example :** Intelligent ∧ Beautiful	Goals may involve conjunctions and disjunctions **For example :** Intelligent ∧ (Beautiful ∨ Rich).
5.	Effects are conjunctions.	Conditional effects are allowed : when P:E means E is an effect only if P is satisfied.
6.	Does not support equality.	Equality predicate (x = y) is built in.
7.	Does not have support for types.	Supported for types **For example :** The variable p : Person

State Space Planning

Forward State Space Planning (FSSP)

Applicable actions: Given a state S an action *a* is *applicable* in the state if its preconditions are satisfied in the state. That is,

$$pre(a) \subseteq S$$

Progression: If an applicable action *a* is applied in a state *S* then the state *transitions* or *progresses* to a new state S', defined as,

$$S' = \gamma(S, a)$$
$$= \{S \cup effects^{+}(a)\} \setminus effects^{-}(a)$$

Plan: A plan π is a sequence of actions $<a_1, a_2, ..., a_n>$. A plan π is *applicable* in a state S_0 if there are states $S_1,...,S_n$ such that $\gamma(S_{i-1},a_i)=S_i$ for $i=1,...,n$.

The final state is $S_n = \gamma(S_0,\pi)$.

Valid plan: Let G be a goal description.
Then a plan π is a valid plan in a state state S_0 if,

$$G \subseteq \gamma(S_0,\pi)$$

Forward vs. Backward State Space Planning

FSSP has a *high branching factor*.

This is because the given state is *completely described* and *many actions* are applicable.

Without the guidance of a good heuristic function search may be computationally expensive.

On the other hand the *goal description is often quite small*
- for example G = (on(G,A) ∧ on(B,J)) in the example

Will working backwards from the goal be more efficient?
- only the *relevant* actions will be considered

Backward State Space Planning (BSSP)

Relevant actions: Given a goal G an action *a* is *relevant to the goal* if it produces some positive effect in the goal, and deletes none. That is,

$$\{effect^+(a) \cap G\} \neq \phi \ \wedge \{effects^-(a) \cap G\} = \phi$$

Regression: If a relevant action *a* is applied in a goal G then the goal *regresses* to a new goal G', defined as,

$$G' = \gamma^{-1}(G, a)$$
$$= \{G \setminus effects^+(a)\} \cup pre(a)$$

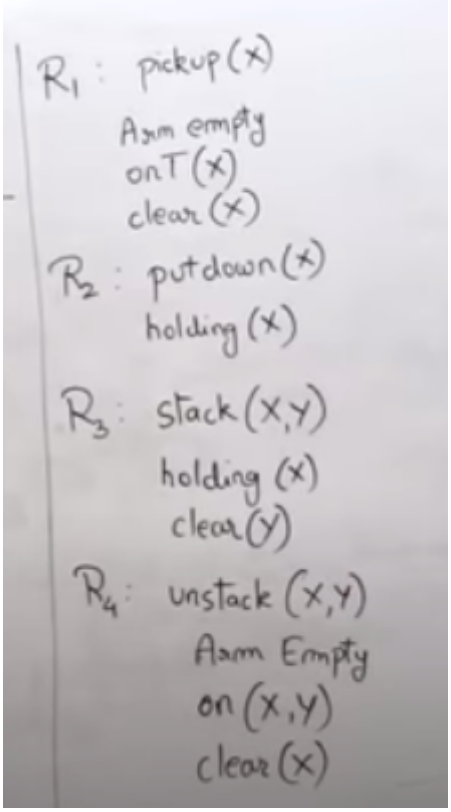
Plan: A plan π is a sequence of actions $<a_1, a_2, ..., a_n>$. A plan π is *relevant* in to a goal G_n if there are goals $G_1,...,G_{n-1}$ such that $G_{i-1} = \gamma^{-1}(G_i, a_i)$ for $i = 1,...,n$. The final goal is $G_1 = \gamma^{-1}(G_n, \pi)$

Valid plan: Let S_0 be the start state. Regression ends when $G_1 \subseteq S_0$
The plan π is a valid plan if $G_n \subseteq \gamma(S_0, \pi)$

note: validity still checked by progression

	FSSP	BSSP
Start at	Start State	Goal Description
Moves	*a* is *applicable* $pre(a) \subseteq S$	*a* is relevant $\{effect^+(a) \cap G\} \neq \phi \wedge \{effects^-(a) \cap G\} = \phi$
Transition $S' = \{S \cup effects^+(a)\} \setminus effects^-(a)$	Progression Sound $\pi \leftarrow \pi \circ a_1 a_2 \cdots$	Regression $G' = \{G \setminus effects^+(a)\} \cup pre(a)$ not Sound $\pi \leftarrow a \circ \pi$
GoalTest	$G \subseteq \gamma(S_0, \pi)$	$\gamma^{-1}(G_n, \pi) \subseteq S_0$ followed by validity check

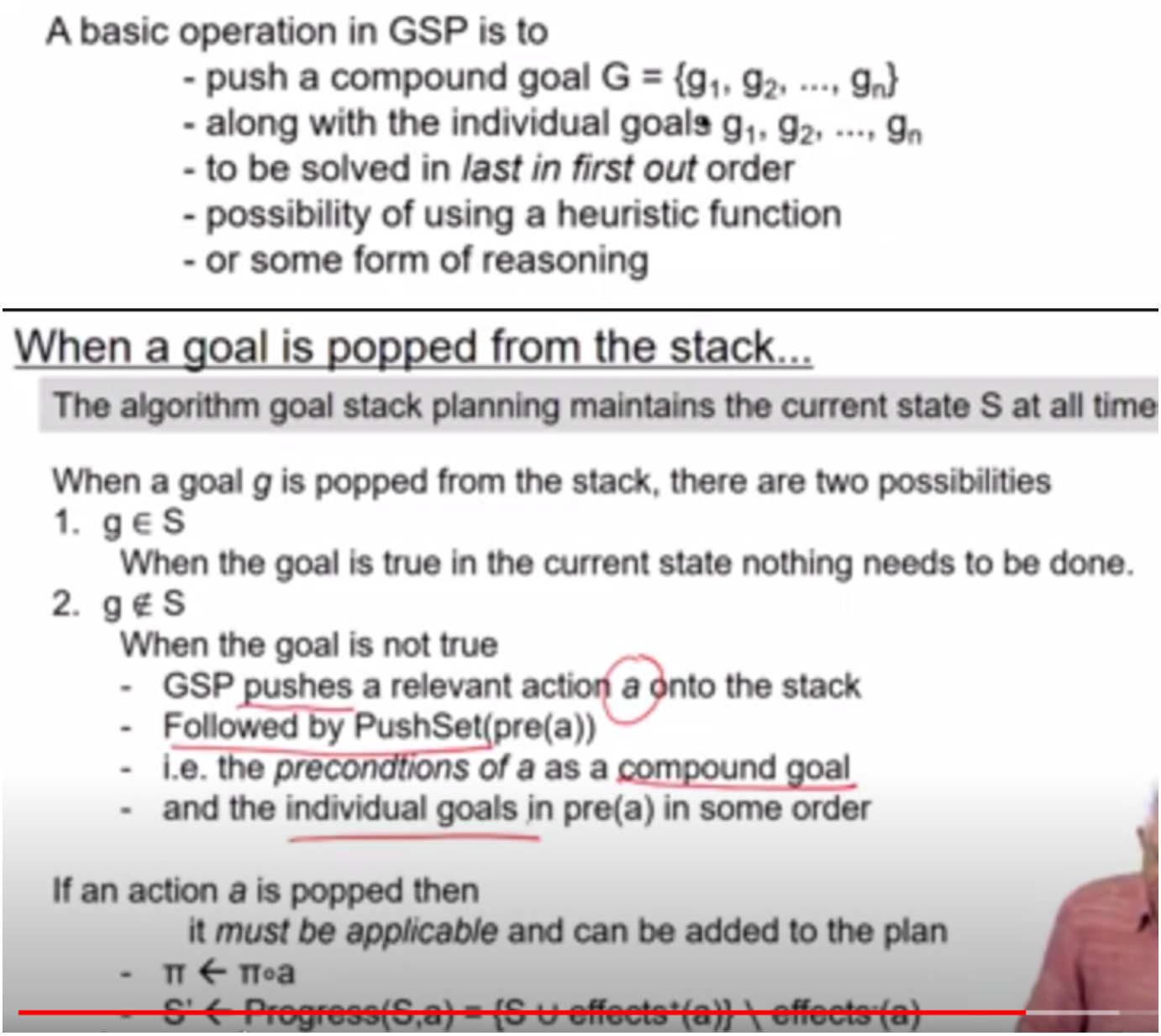
Goal Stack Planning



FSSP
- searches for *applicable actions* from the start state S
- constructs the plan from the start state
- the *first action found* is the *first action in the plan*
- suffers from *high branching* since S is a full description

BSSP
- searches for *relevant actions* from the goal description G
- construct the plan from the the goal description
- the *first action found* is the *last one in the plan*
- suffers from *spurious sub-goals*

Goal Stack Planning is an algorithm designed to
- *search* in a *goal directed backward* manner
- to take advantage of low branching
- but *construct plans in a forward* manner
- adding actions that are *applicable*



- Goal stack planning is linear planning
- linearly means we break down the compound goal into individual goals and then we solve individual goals one by one serially.

Plan Space Planning

- It is also called as Non linear planning / Parital order planning / least commitment planning
- In the other planning approaches like goal stack plannign we find the set of actions and then add it into the plan but in plan space planning we have the space of all possible plans and we have to search in this space for a plan.

Partial plans

The search space in plans space planning (PSP) constitutes of *partial plans*

A partial plan π is a 4-tuple
$\pi = \langle A, O, L, B \rangle$ where

- A is the set of partially instantiated operators in the plan,
- O is the set of ordering links or relations of the form $(A_i < A_j)$,
- L is the set of causal links of the form (A_i, P, A_j),
- B is the set of binding constraints of the form -
  $(?X = ?Y)^*$, $(?X \neq ?Y)$, $(?X = A)$ or $(?X \neq B)$ or $(?X \in D_X)$
  where D_X is a subset of the domain of ?X.

The Initial Plan

Given the planning problem
$\langle S=\{s_1, s_2, \ldots, s_2\}, G = \{g_1, g_2, \ldots, g_k\}, O\rangle$

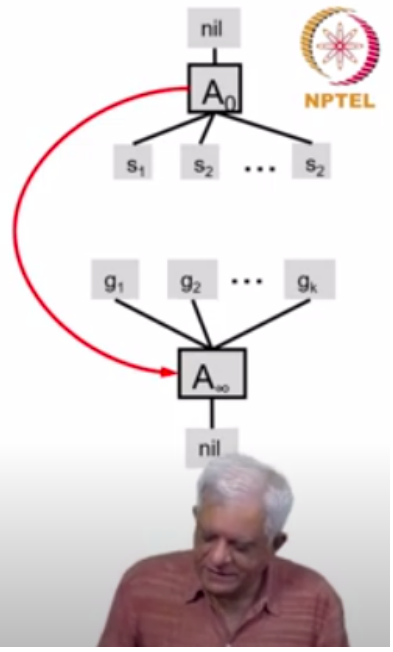
Planning in *plan space planning*
always begins with an *initial plan*
$\pi_0 = \langle\{A_0, A_\infty\}, \{(A_0 < A_\infty)\}, \{\}, \{\}\rangle$ where

A_0 is the *initial action* with *no preconditions* and
the *effects* $s_1, s_2, \ldots, s_2$

A_∞ is the *final action* with *preconditions* $g_1, g_2, \ldots, g_k$ and
no effects

$(A_0 < A_\infty)$ says that A_0 *precedes* A_∞

There are no *causal links* and *binding constraints*



Flaws

A partial plan may have two kinds of *flaws*

1. *Open goals* – any precondition of any action that is *not* supported by a causal link

2. *Threats* – A causal link (A_i, P, A_j) is said to have a *threat* if there exists another action A_t in the plan that potentially deletes *P*

- ## Resolving a threat

  To eliminate the threat one needs to ensure that *at least one of the three conditions* for the threat is *not* met. This can be done by, respectively,

  - *Separation*: Ensure that P and Q *cannot unify*.
    - This can be done by adding an *appropriate binding constraint* to the set B in the partial plan.
  - *Promotion*: Advance the action A_{threat} to happen *before* it can disrupt the causal link.
    - Add an ordering link ($A_{threat} \prec A_i$) to the set O in the partial plar
  - *Demotion*: Delay the action A_{threat} to happen *after* both the causal link actions.
    - Add an ordering link ($A_j \prec A_{threat}$) to the set O

Constraints

- Constraints is logical relation among the variables
- Constraint satisfaction is the process of finding the solution to a set of constraints
- popular examples of CSP are n-queens, sudoku, map coloring, latin square, etc.
- Types of constraints:
  - Unary Constraints: involve single variable
  - Binary: involve pair of variables
  - High order constraints: involve 3 or more variables Ex: cript arithmatic column constraint
- Varities of CSPs:
  - Discrete varibles and finite domain: Ex boolean CSP
  - Discreet variebles and infinite domain Ex; job scheduling
  - continuous variables Ex: start/end time for hubble space telescope observations
  - Linear constraints

Backtracking and Look ahead stretegies:

- By using these techniques we can foresee the impace of next move and based on that we can decide the assignment of current variable.

Strategies to improve backtracking efficiency

- Constraint Propagation
- Forward checking

- Most Costrained variable
- Most constraining variable
- Least constraingin variable
- Higher order Directional Consistency

Important Points

- STRIPS - Richard Fikes and Nils Nilsson 1971
- ADL (Action Description Language) - Pednault 1987
- horn clause: A clause that has at most one positive literal

Unit 3

Declarative approach to build an agent
TELL and ASK mechanism

Types of knowledge based agents

- Knowledge Level
- Logical Level
- Imlementation Level

The WUMPUS World Environment

- WUMPUS is an early computer game also known as Hunt the Whompus
- Developed by Gregory Yob in 1972/1973
- Written in BASIC (Begineer's All-purpose Symbolic Instruction Code)

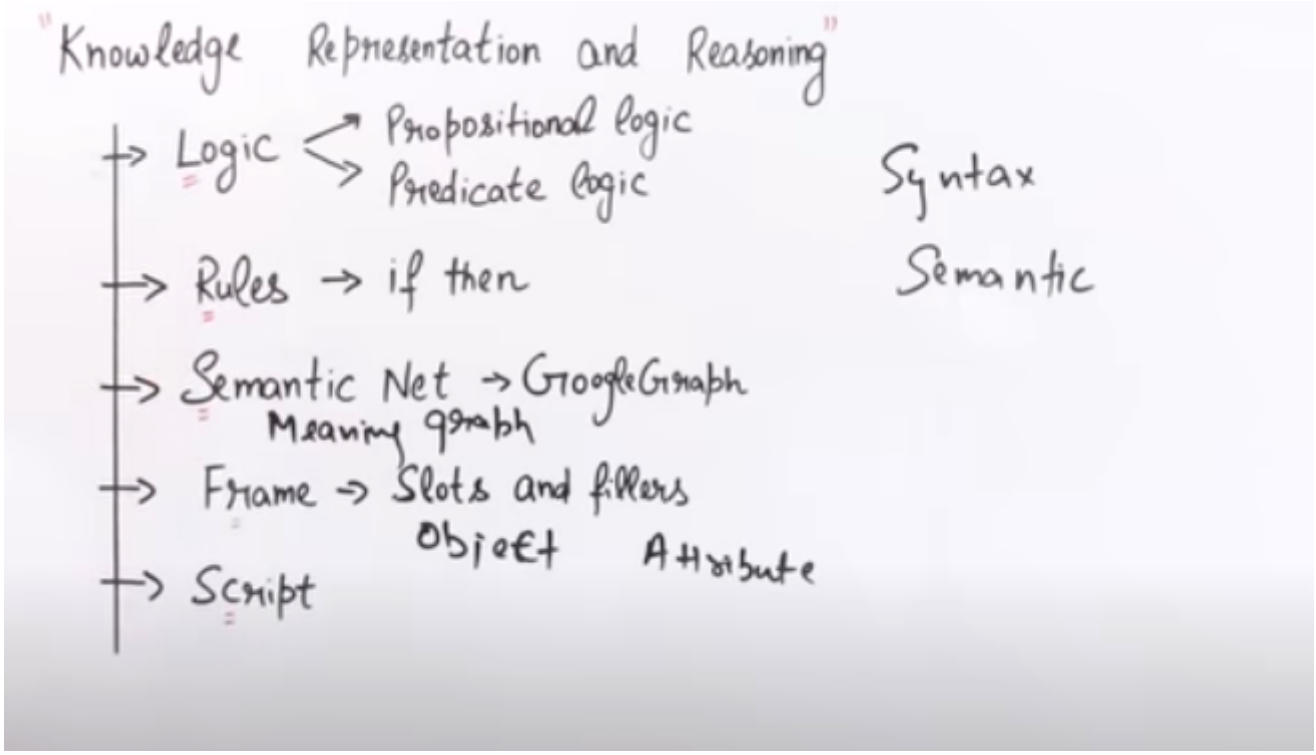
Facets of Knowledge

- Facets means viewpoint or aspect
- A basic facet consists of a relation and set of units as domain.
- In case of binary relation the units are considered as co-domain.
- Facets of knowledge:
    - Explaination
    - Interpretation
    - Application
    - Perspective
    - Empathy
    - Self Knowledge

Knowledge Representation

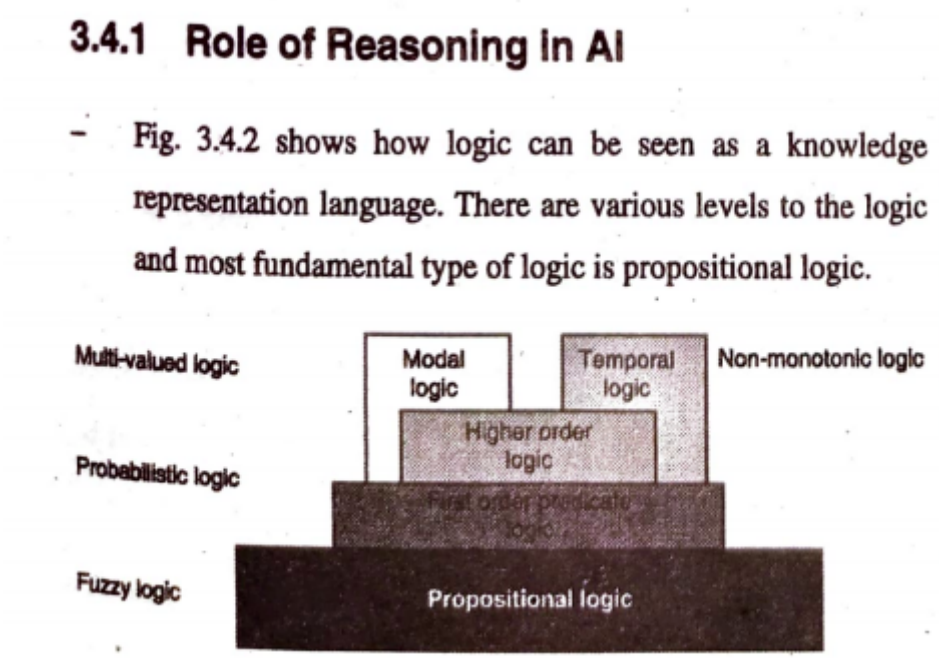
- In order to take the decisions and to perform any action, machine needs some sort of knowledge so that by using that machine can take the decision.

- In order to take actions properly the knowledge should be represented very well.
- There are differnet types of knowledge representation



- Logic means representing the english sentences in the form mathematical form so that a machine can understand it properly.
- In preposition logic we have only 2 values true or false
- In predicate logic we have quantifiable values and values such for all, etc.

Role of Reasoning in AI

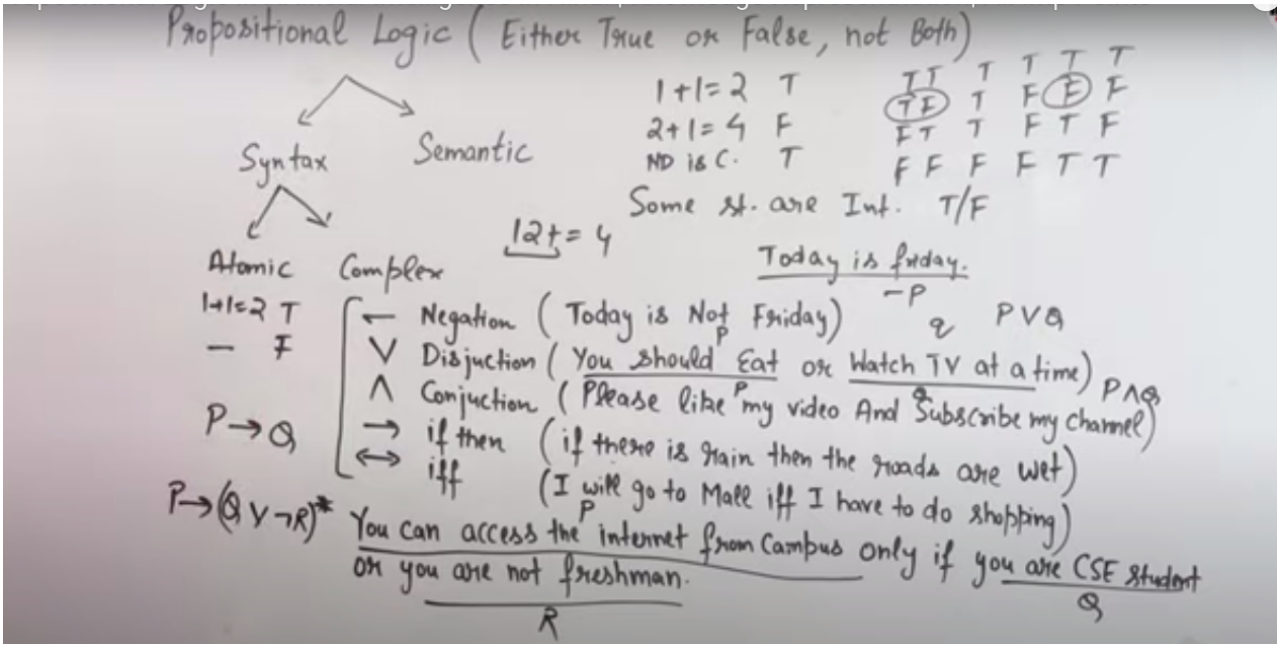


3.4.1 Role of Reasoning in AI

- Fig. 3.4.2 shows how logic can be seen as a knowledge representation language. There are various levels to the logic and most fundamental type of logic is propositional logic.

Prepositional Logic

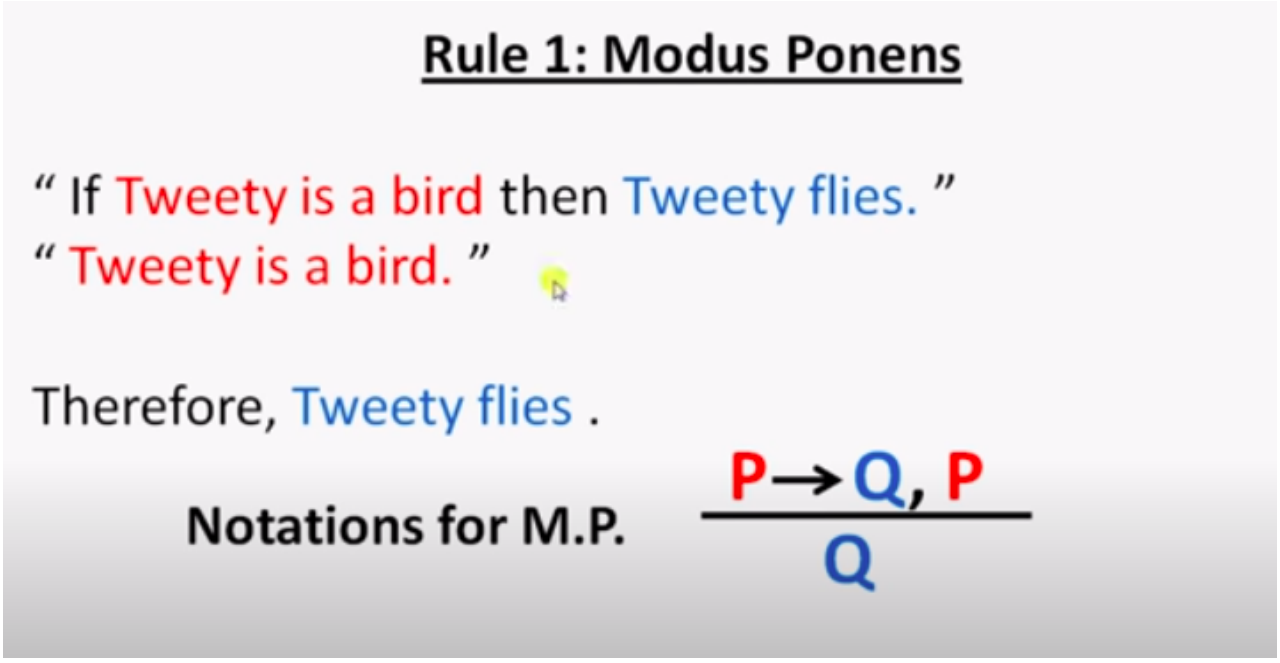
- Prepositional logic is simple but powerful for some AI problems

- Atomic formula is formula that has no strict sub-formulas. Atomic logic formulas are called as prepositions.
- Syntax based prepositional logic
  - 

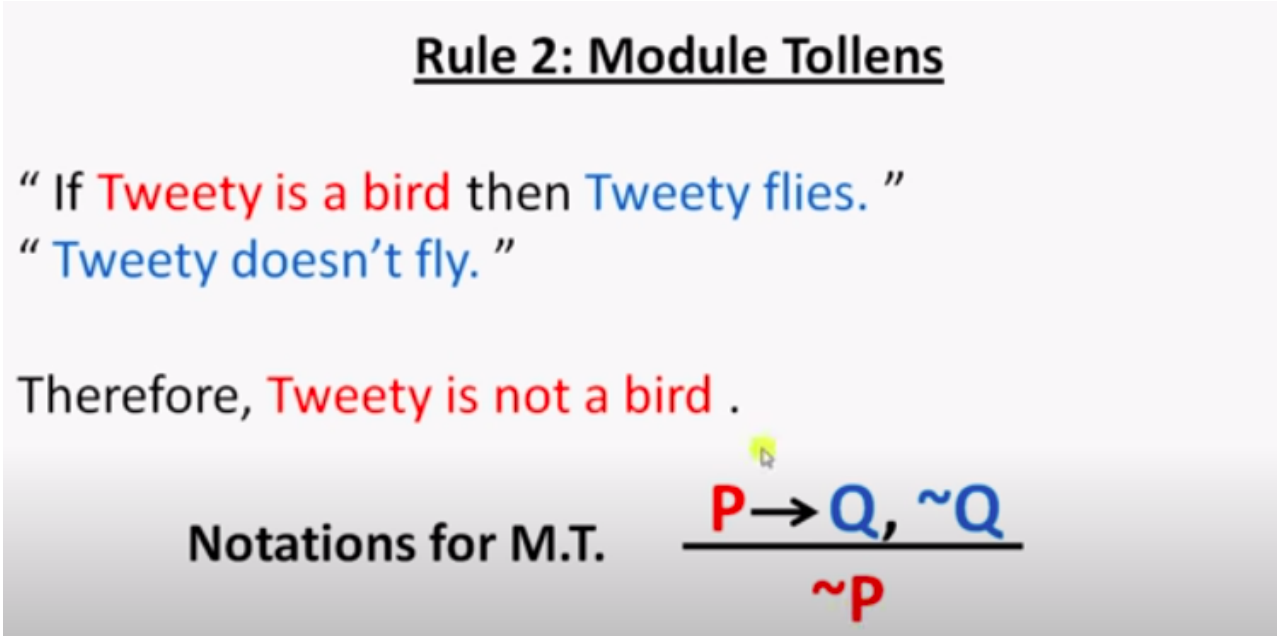
- Semantics based preposional logic
  - semantics of sentence is meaning of a sentence
  - In semantic based PL world facts are repsented using PL symbols and each symbol mapped to real world feature.
  - we can define semantics of each prepositional symbol in following manner:
    - A means "It is hot"
    - B means "It is humid"
  - Sentence is cnsidered true when its interpretation in the real world is true.
- Examples
  - Ram and shyam is a student
    - P: Ram is a student Q: Shyam is a student
    - P ^ Q
  - Today is sunday or monday
    - P: today is monday Q:today is sunday
    - P V Q
  - Ram teaches shyam or pappu but not teaches at a same time
    - P: ram teacher pappu Q: ram teaches shyam
    - (P V Q) ^ ¬(P^Q)
  - It is is hot then it shall rain

        - P: It is hot Q: It shall rain
        - P→Q
    - He requires a doctoe and also a lawyer if and only if he is sick and also injured.
        - P: He requires a doctor Q: He requires a lawyer R: he is sick S: He is injured
        - (P ^ Q)↔(R ^ S)
- Advantages:
    - Simple and easy to implements
    - Suitable for some AI problems
    - foundation for FOL (First Order Logic)
- Disadvantgaes:
    - Cannot solve complex problems
    - cannot express the individual entitiy properties and the realtion between the individual entities. Ex: rahul is tall
    - cannot express gneralization and specailization

Inference Rules:

- Modus Ponens
    - 

    - Here in the above image denominator Q is the conclusion
- Module Tollens

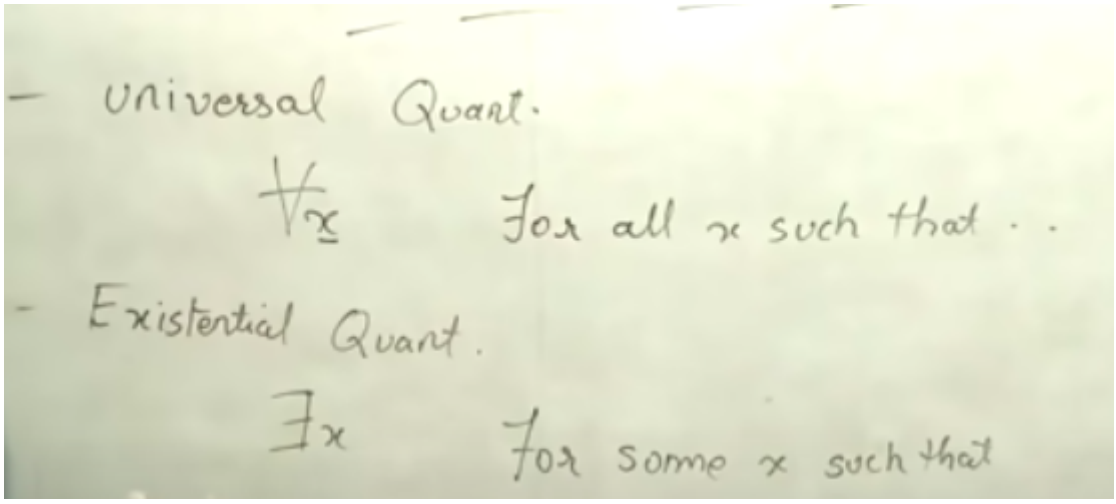
- 

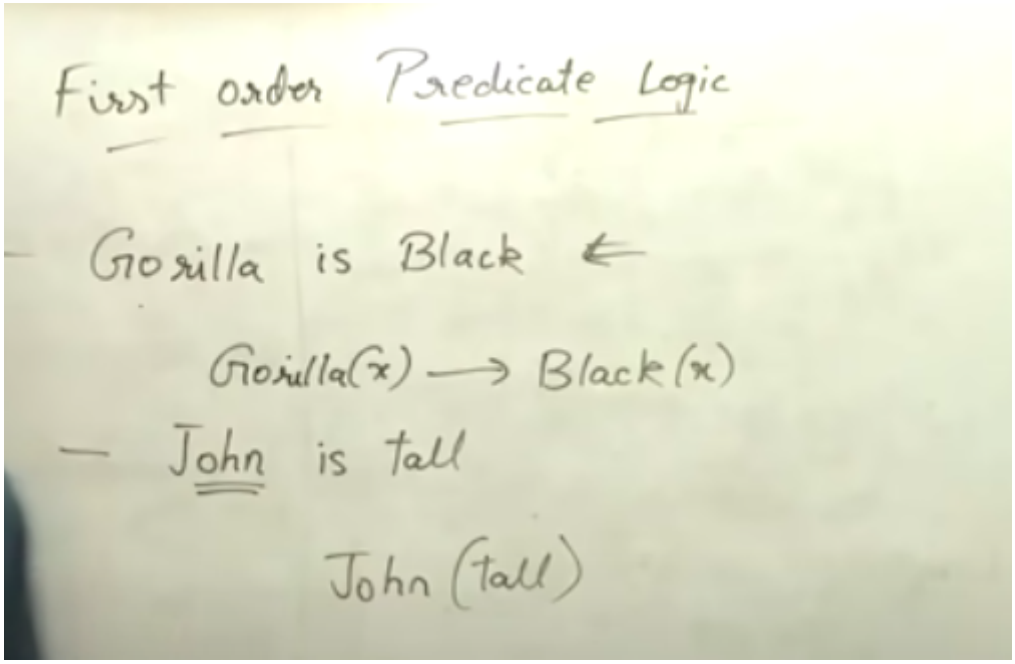
- **Table 3.5.3 : Inference Rules**

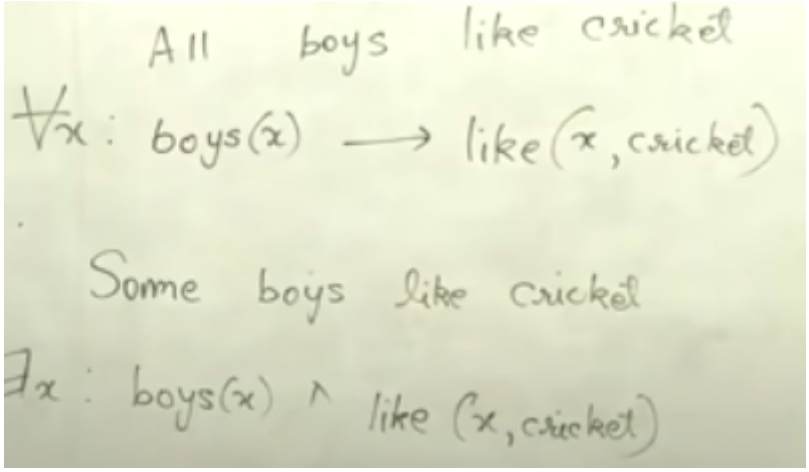
Inference Rules	Premise (KB)	Conclusion
Modus Ponens	X, X→Y	Y
Substitution	X →Z &Y → Z	A→B
Chain rule	X → Y, Y→Z	X →Z
AND introduction	X, Y	X ^ Y
Transposition	X →Y	~X →~Y

- There are two types or categories of inference rules:
  - Sound Inference:
    - Sound inference property says that if "X is derived from the knowledge base then X is enatailed by knowledge base".
    - Soundness property represented by
    - "If KB |- X then KB | = X".
  - Complete Inference
    - Converse of Soundness
    - "If X is enatailed by KB then X is derived form the KB"
      - Completeness property can be represented as :
      - " If KB |= Q then KB |- Q".

First Order Logic (FOL):

- It is also called as Predicate logic or FOPL (First Order Predicate Logic)
- It uses variables, relations and quantifiers to represent the knowledge.
- Two types of qunatifiers
- 

- In universal quant we use -> and in existential quant we use ^
- 

- 

- Examples:
  - Ram is tall

    - tall(Ram)
- Rahul loves neha
    - love(Rahul,Neha)
- Ram teaches either math or cs
    - teaches(Ram, math) V teaches(Ram, cs)
- All students like football
    - $(\forall x)(\text{Student}(x) \rightarrow \text{like}(x, \text{football}))$
- Some students like football
    - $\exists x(\text{Student}(x) \wedge \text{like}(x, \text{football}))$
- All studetns are happy
    - $\forall x(\text{Student}(x) \rightarrow \text{happy}(x))$
- Every rational number is a real number
    - $\forall x(\text{RN}(x) \rightarrow \text{REN}(x))$

      RN(x): x is a rational number.

      REN(x): x is a real number

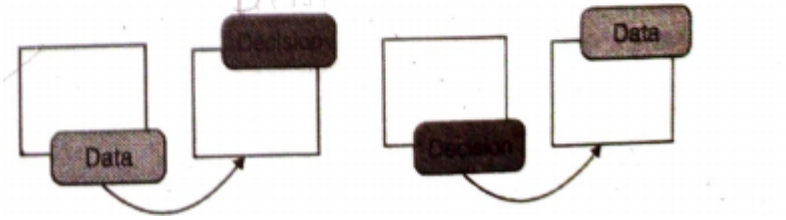
Deductive Retrieval

Forward and Backward Chaining

- For any type of inferencing we have to find a path from start to goal. When we have some data and we make a decision based on this data then the process is called as the **forward channing**, whereas if we have a decision and based on the decision if we fetch the initial data then it is called as **backward chaining.**

- Take an example, "If it is raining then, we will take umbrella". Here, "it is raining" is the data and "we will take umbrella" is a decision. This means that we knew already that it's raining that's why we are going to take umbrella. This process is **forward chaining.**

- Now, take another example, if you are going out and you make a decision of taking umbrella. Then based on this decision it can be guessed that it is raining. Here, "taking umbrella" is a decision based on which we guessed that the data can be "it's raining". This process is **backward chaining.**



Second Order Logic

- Second order logic is the extension to FOL.
- It FOL we have quantifier "for every object" and here we have extra quantifier called "for every property of objects".
- This increases the expressive strength of SOL as compared of FOL
- For classical extensional logic, properties can be identified with sets, so that the SOL provides us with the quantifier "for every set of objects".

Important Points

- Semantic Relations
  - meronymy: a is a part of b
  - holonymy: part of itself
  - hyponymy: subordinate
  - hypernymy: superordinate
  - synonymy: same as this entry
  - antonymy: opposite

Unit 4

Natural Language Processing (NLP)

- NLP is a form of AI which allow machins to read the text by simulating the human ability to understand the language.
- It allows machine to udnerstand and process the language that human speak.
- text and verbal input and output
- components of NLP:
  - NLU (Natural Language Understanding):
    - In NLU the the natuaral input is converted into some inermediate representation.
    - NLU is a complicated process because one sentence can have many meaninggs and many sentences mean the same.
    - Various types of ambiguities such as lexical, referntial and syntactical.
    - Lexical: a word can be noun, verb, etc
    - Syntactical issues
    - Referntial: In some sentence there is ambiguity like to whom we are referencing.
  - NLG (Natural Language Generation)
    - Converts intermediate generation back into natural lang output
    - sub processes of NLG
      - Text Plannnig: extracting relavant text from KB
      - Sentence Planning: selecting correct words, forming meaningful sentence, grammer, etc.
      - Text Realization: process of mapping the planned sentence into a structure.

Stages of NLP

- Lexical Analysis:
  - It deals with recognization and identification of structure of the sentence.
  - It divides paragraph into sentences, phrases and words.
- Syntactic Analysis
  - the sentences are parsed as noun, verbs, etc
  - grammer check
- Semantic Analysis
  - meaning of the sentence
- Disclosure integration
  - context of the sentence

- Pragmatic Analysis
  - Here the sentences are reinerpreted to verifiy the meaning in the given context.
  - It requires real world knowledge of the language

Important Application of NLP

- Machine Translation
  - machine translation is the process of converting the text from one human language to another human language.
  - IBM demonsrated the first machine translation model which is based on word to word translation
  - Various approaches of machine translation:
    - word for word
    - syntactic transfer
    - inter-lingual approaches
    - example-based
    - statistical
- Information Retrieval
- Big Data Information Retrieval

General Model of Learning Agents:

- Learning Element
  - learning element helps the agent to learn new things
- Processing Element
  - process element is the main component of agent
  - it is like neuron in the human body
  - it generate actions that need to be performed for the given input from the environment.
- Problem Generator
  - problem generator helps the processing elements by suggesting actions which can generate new experices.
  - this is usedufl for training an agenet
- Critic
  - critic gives feeback from the environment to the learning element
  - it helps the model to learn the new things from the environment feedback

Types of Learning or Feedback

- Supervised

- Unsupervised
- Reinforcement

Supervised learning

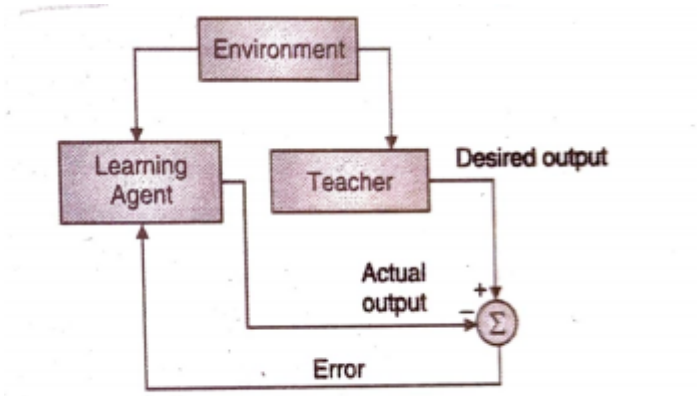
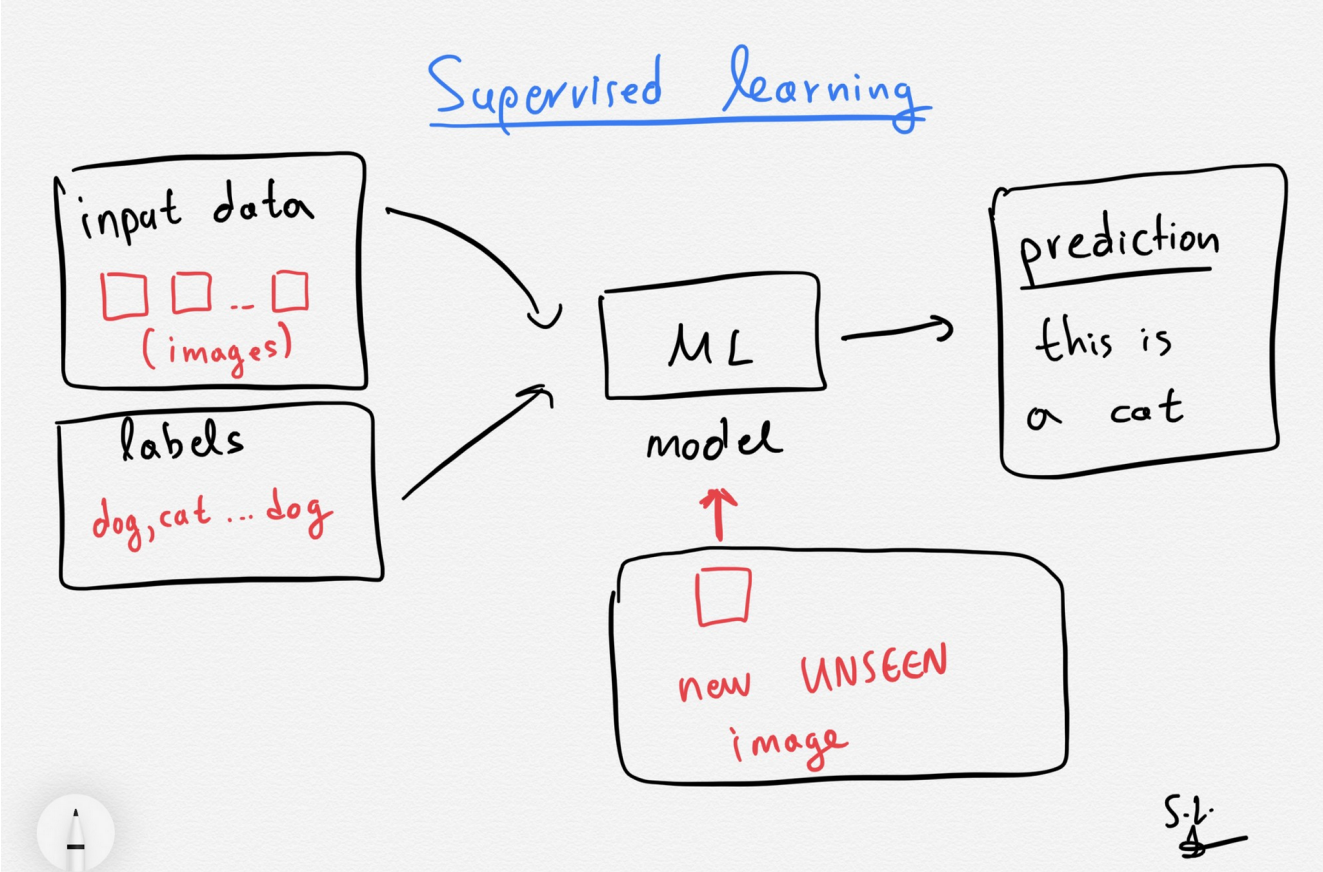
- In supervised learning, learning agent and teacher both take input from the environment.
- teacher has the desired answer for a given inuput
- learning agent process the input and find actual output.
- the adder then subtract the actual output from the desired output and calculates the error which is sent back to the learning agent so that the larning agent can improve next time.
- 

Fig. 4.3.5 : Supervised Learning
- Here we have labelled data
- 

- Supervised models can be further grouped into regression and classification cases :

Unsupervised Learning

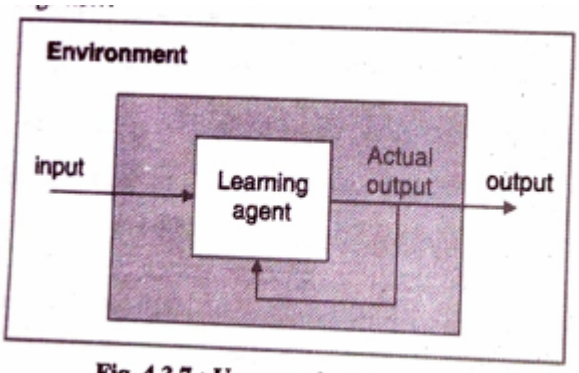
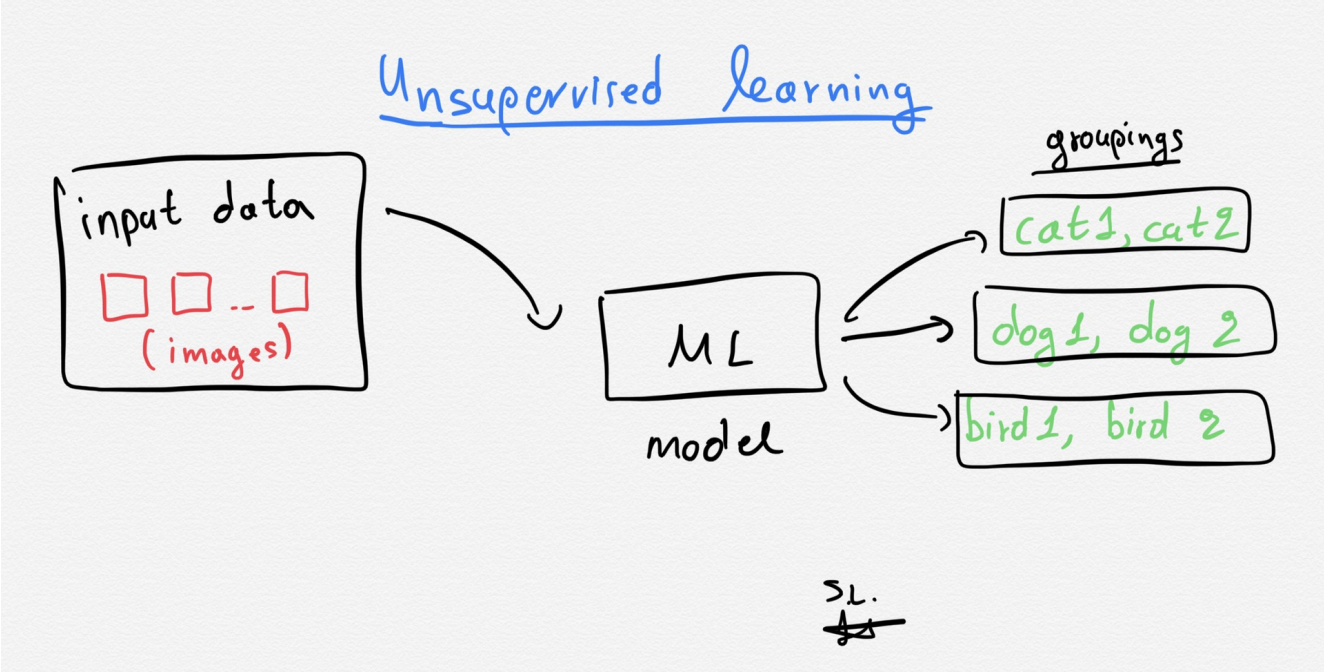
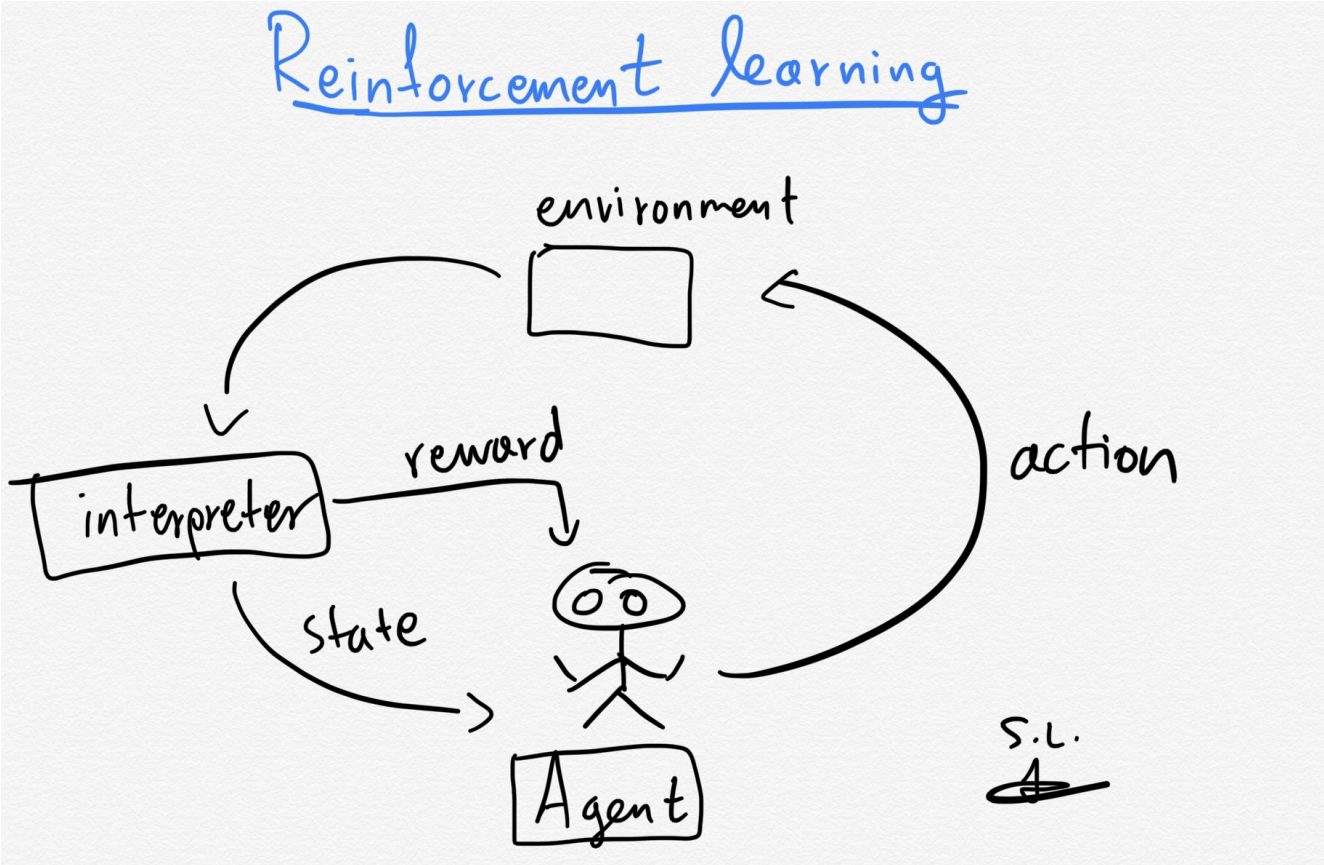
- Unsupervised learning does not have any teacher which means that correc answer is not known to the learning agent.
- Here the learning agent try to find out the pattern in the data by own and prdict the output.
- 

Fig. 4.3.7 : Unsupervised Learning
- 

- Unsupervised models can be further grouped into clustering and association cases.

Reinforcement Learning:

- 

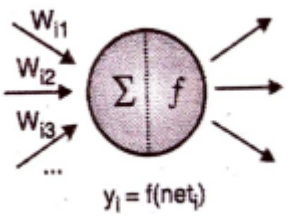
- This family of models consists of algorithms that use the estimated errors as rewards or penalties.
- If the error is big, then the penalty is high and the reward low. If the error is small, then the penalty is low and the reward high.

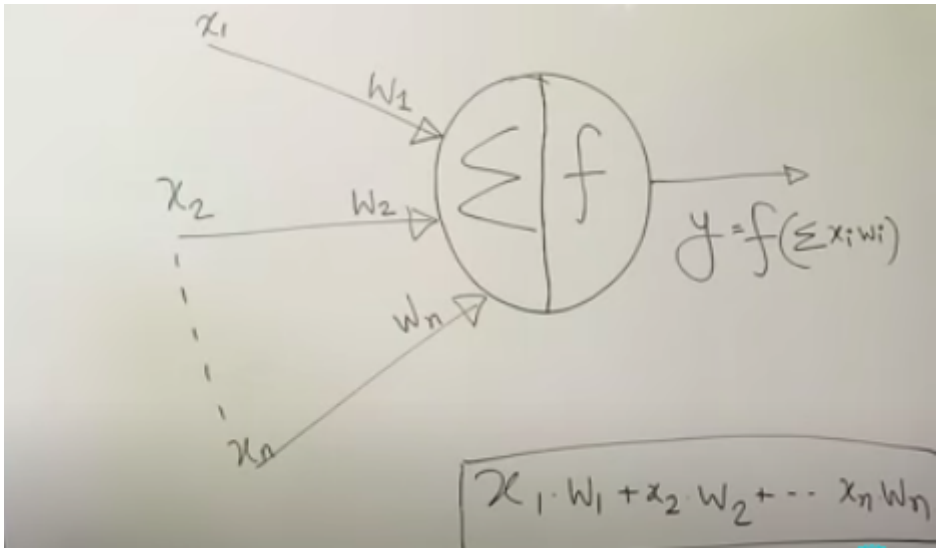
Neural Network

- the basic computational unit is neuron.
- human body has 10^11 neurons and each neuron is connected to 10^4 others.
- A neuron has:
  - Dendrites (inputs)
  - Axon (output)
  - Cell body
- checmical reaction happends in synapses
- receptors of neuron - synapses
- synapses types:
  - Exciatory
  - Inhabitory
- When neuron sends the informaition, it is neuron fires and also called as depolarization.

Artifical Neural Network (ANN)

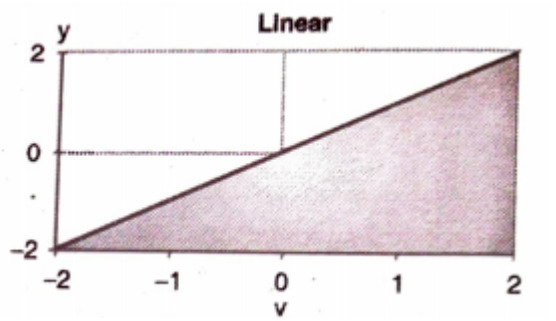
- ANN consists of many simple processing units, which are wired together in a complex communication network as human brain.

- processing units are same as human brain neurons
- 

- Each input assigned with some weight and it is given to the processing unit.
- 

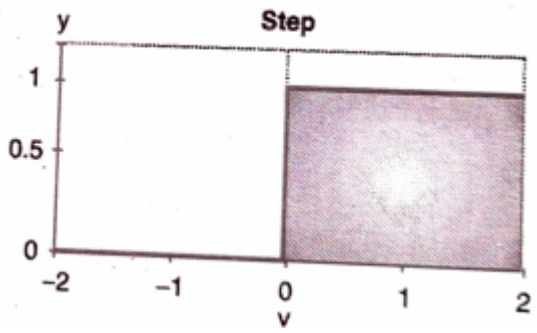
- Here in the above image there are two parts of processing units one is summation and other is activation function.
- First it calculate the weighted sum by multiplying each input with the its weight and add it to the weighted sum.
- Now this weighted sum is sent to activation function which generate approprite output accoring to the weighted sum.

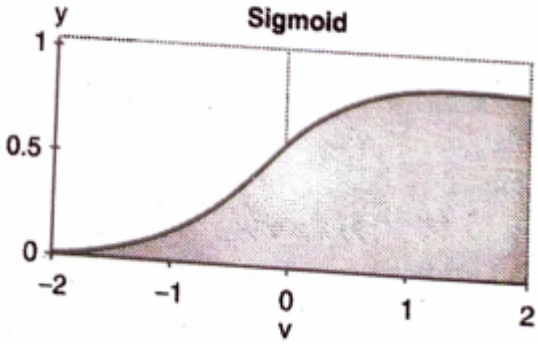
Activation functions:

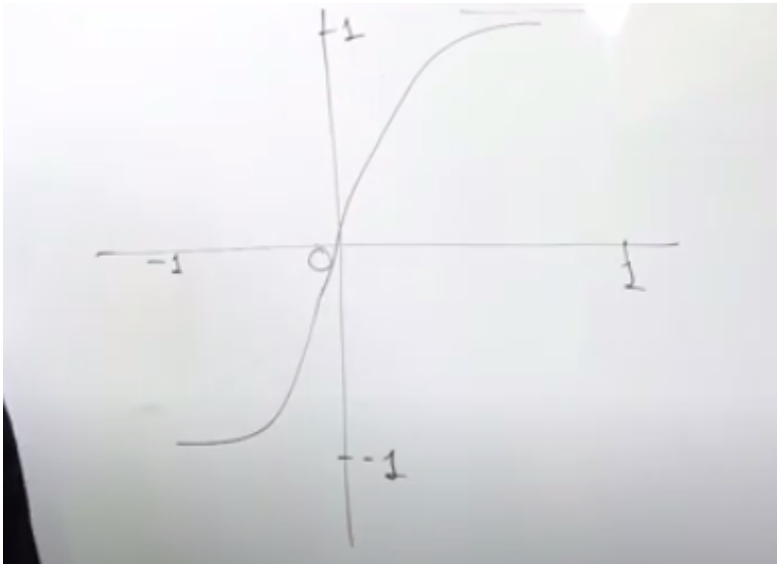
- Linear function
  - $$f(v) = a + v = a + \sum w_i x_i$$
    where parameter a is called bias.
  - 

- Heviside step function

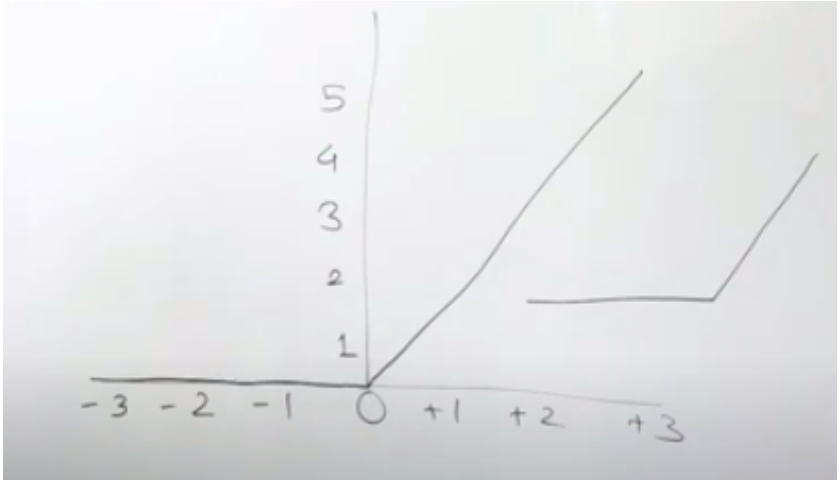
- $$f(v) = \begin{cases} 1 & \text{if } v \geq a \\ 0 & \text{otherwise} \end{cases}$$

  Here a is called the threshold.
- 

- Sigmoid function
  - $$f(v) = \frac{1}{1 + e^{-v}}$$
  - 

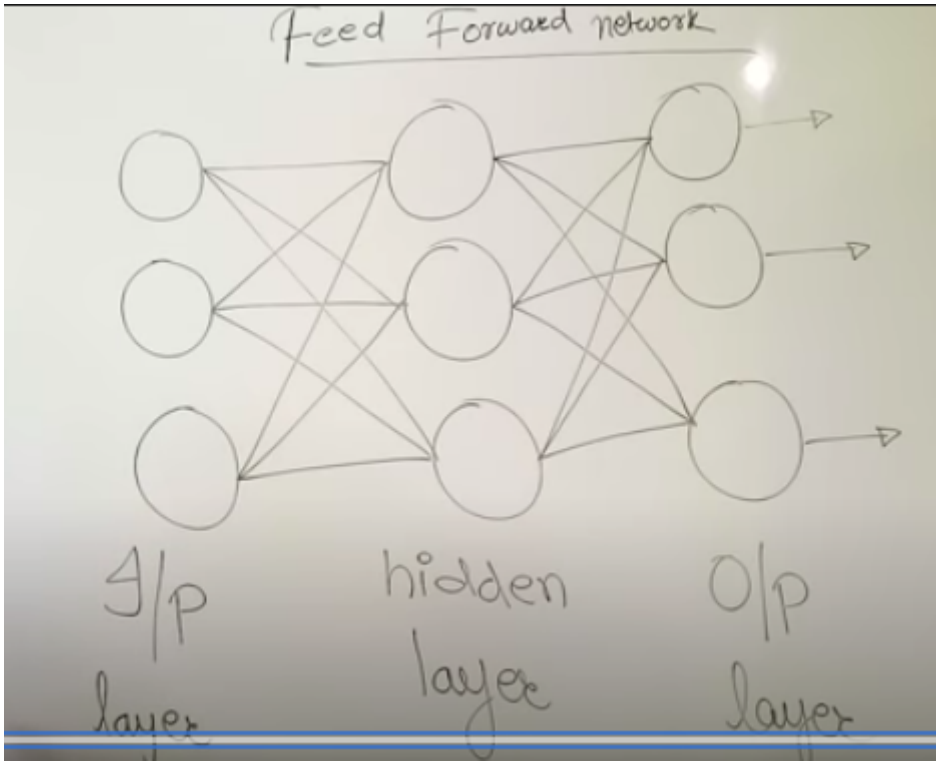
  - sigmoid function is not zero centric i.e. on y axis the range is from o to 1
- tanh function
  - hyperbolic tangent function
  - 

  - zero centric function
  - range is from -1 to 1
- ReLU

- Rectified linear unit
- most popular and most widely used activation function
- 

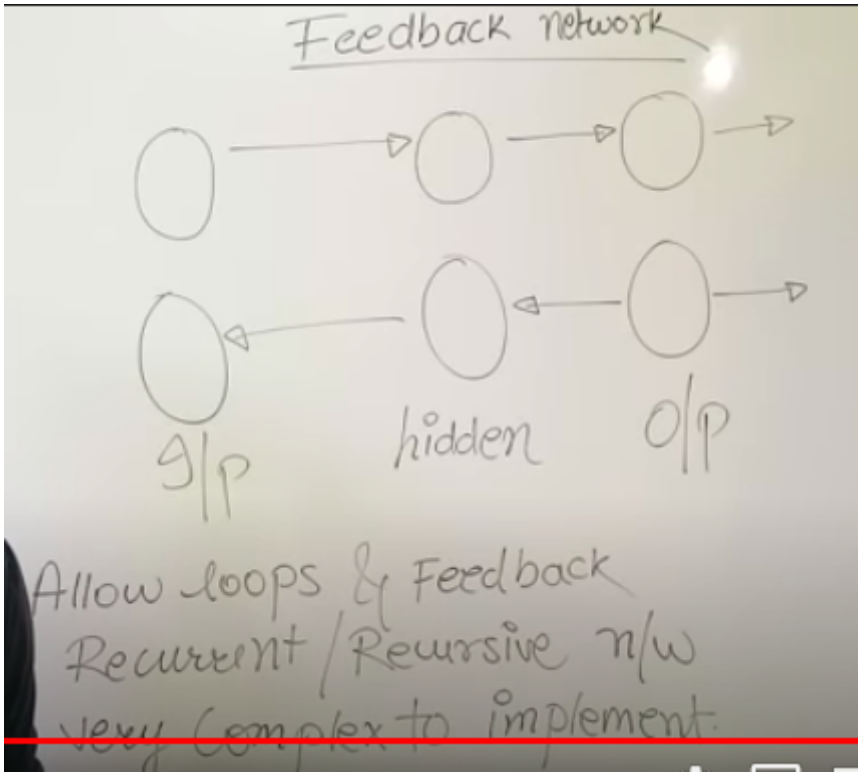
- for positive values the output is linear in nature
- and for negative values output is zero

Feed forward network

- These are the networks without cycles or feeback loops hence are called as feed-forward networkd
- also called as perceptron
- In feed-forward networks the signal only flows in forward direction i.e. if the layer1 receives the signal and then it passes two layer2 neron then it cannot sent back to again neuron1.
- Mutliple hidden layers are present between input and output layers
- 

- We can use heviside step function on the output layer as an activation function because it has only two output states and for the hidden and input layer we can use other activation functions.

Feedback network

- 

Important Points

- First machine transaltion - IBM 1954 word to word translation
- Inductive learning is the techniue of supervised learning. it is also called as concept learning
- statistical processing of natural language involves two techniques: detecting n-gram and stopwords lists
- document preprocessing has 3 steps
- two types of erroes in nn. first error is at output layer and second layer is at hidden layer
- The Hopfield Network is one good way to implement an associative memory

Unit 5

Introduction: Robots and Robotics

- Isaac Asimov introduced the term robotics first time in the 1940s
- Robotics is a domain in artificial intelligence that deals with the study of creating intelligent and efficient robots.
- Robotics is a branch of AI, which is composed of Electrical Engineering, Mechanical Engineering, and Computer Science for designing, construction, and application of robots.
- Robots are the artificial agents acting in real world environment

Point Robot

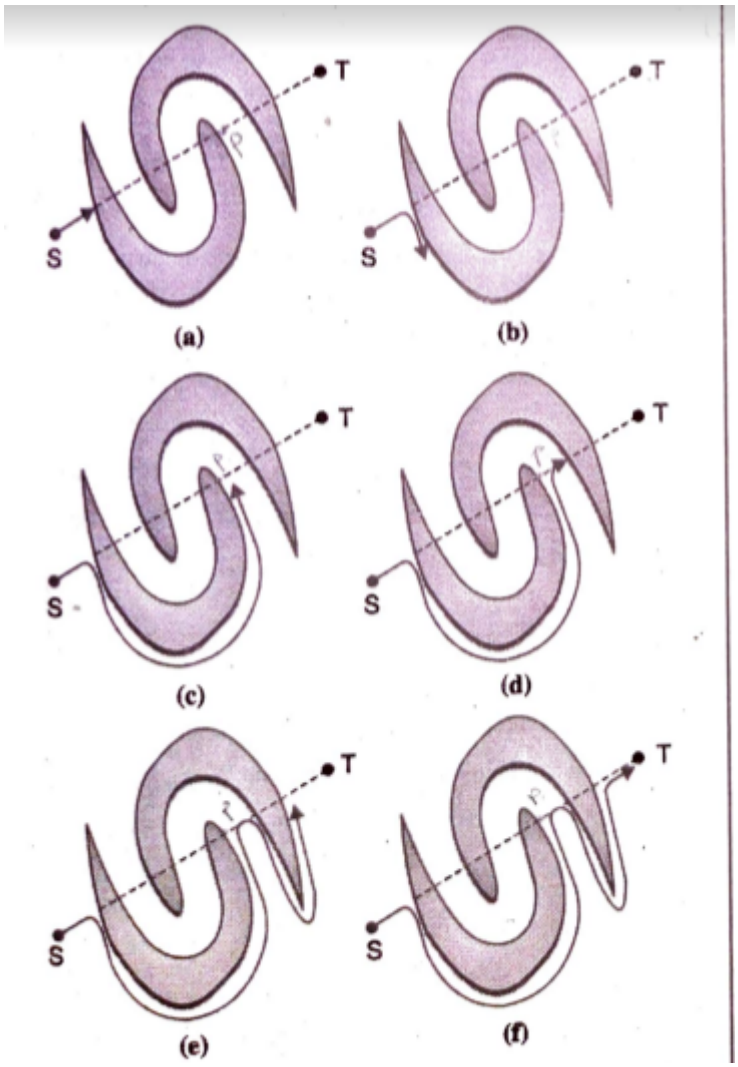
- The abstraction of autonomous robot in a simple form is a point robot.
- Plane is the operation domain of robot R^2
- robot is represented by point (x,y) on the plain. It is called as pose or configuration of robot
- In the process of moving the robot its state is updated from (a,b) to (c,d)

- The operation of robot is based on the plain but whole plain is not available for the robot. The group of poses which are valid is known as free space of robot.
- Free Space - cfree
- Occupied Space R^2-cfree

Point robot:

- Path planning
- Sensing
- Mapping

Path Planning for point robot

- Here it is important that the start and end (goal) must be in cfree for the existence of path.
- Simplest algorithm to find the path is Bug2 algorithm
- 

- In bug2 algorithm the robot start with the starting point and follows the startight line unitl it hits any obstacle. Once the robot hits the obstacle it follows the path of obstacle until it gets back to the point where the robot can again follow the straight line. If it does not reach to this poin then there is no path from start to goal.
- If robot finds the straight line then again robot follows the straight line path until it reach to goal.

- Bug2 algorithm gurrantees to return the path if it is exist and return failure if the path does not exists.
- Also bug2 algorithm does not gurrantees to return the optimal path

Sensing for Point Robot

- There is a need of some standard mechanism for robot to sense its surrounding envrionment which helps it to handle the navigation.
- laser point sensor is a sensor which return the distance to first encountered object in the path from the robot and in case if there is no obstacle in the path then it returns infinity.
- This sensor can be used by the robot which is using bug2 algorithm for determining the local shape of the obstacle to decide further movement along the boundary of the obstacle.

Mobile Robot Hardware

Structure of a mobile robot:
two pricipal design have been adopted:

- functional: top down
- behavioural: bottom up

Components of Mobile Robot

- Locomotion: how robot moves in the environment
- Sensing: how robot sense the properties in the environment
- Reasoning: how mapping of environment properties into actions
- Communication: how robot communicate to the outside operator.

Non-Visual Sensors

Contact Sensors

- most common type of sensors in the field of robotics
- Contact sensors are used to detech the physical contatc of robot with the other objects.
- Transducer is used by the contact sensors for the sensing operations.
- Number of contact sensors are there such as push button switch, limit switch or tactile bumper switch, potentiometer, strain gauge, etc.
- Main two types are bumper and tactile.
- If we want very little information related to contact then we can use bumper. It is very generic and return the result in YES/NO options i.e. whether the robot is touching to the object or not.
- If we want more detail information like to which object it is touching, intensity of the hardness of the contact, etc.
- Contact sensors is mainly used for obstacle avoidance for the robots.

Inertial Sensors

- Inertial sensors are considered as the type of external sensors which avoid making any reference with the external world.

- The term inertial sensor used to denote the combination of three axis accelerometer and three axis gyroscope.
- Devices containing these sensors are referred as inertial measurement units (IMUs)
- A gyroscope measure the sensor angular veloctiy i.e. the rate of chnage of the sensor orientation.
- An accelormeter measures the external specific force acting on the sensor.
- Now a days many gyroscope and accelerometers are based on Micro Electro Mechanical System (MEMS) technology.
- inertial sensors are used for tracking human motion and for pose estimation.

Infrared Sensors

- IR sensors uses the infrared light to detech the object in front of them and calculate the distance from the obstacle.
- There are basically two components in the IR: emitter and detector.
- Emitter emits the IR light in the large ARC form and if there os a object in the path then the light will be reflected from the object to the detector. So the detctor can measure the angle and the distance of the object. Here traingle is formed between emitter, detector and the object.
- There are three wires in the IR sensor: red, black and yellow
- red is used for power, black is for ground and yellow is for getting IR output.
- There are some problems with IR
    - Less perfection because range is quite small i.e. between 10cm to 1m
    - Because of sublight and other flames the detector may confuse bcause sunligh and flames also emits the IR light.

SONAR

- SONAR stands for Sound Navigation and Ranging
- SONAR uses sound propogation for navigation, commnication and to detect the object under water.
- SONAR used in submariens for deteciton of other object under the water.
- Two types of technilogies share the name SONAR:
    - Passive sonar: passive sonar listens for the sound made by vessels
    - Active sonar: active sonar generates the sound pulses and listening for the echos.
- The acoustic frequencies used in SONAR systems vary from very low (infrasonic) to extremely high (ultrasonic).
- Before the intodiction of RADAR, SONAR can also used for detection of the objects in air. That sonar is called as SODAR.
- The typical SONAR module consumers 100 mA in standby mode, meaning no sound pulses are being emitted.

RADAR

- RADAR stands for Radio detecting and Ranging.
- The orinciple of operation of RADAR is same as SONAR.
- In RADAR the radio waves are emitted and then when they hit a particular surface the waves reflected back and the distance of the object is calculated.
- In sonar acoustic waves are used but here in RADAR radio waves are used.
- RADAR can go through surface layer of any specific object.
- Similar to SONAR systems, RADAR systems also experience problems with secular reflections.
- RADAR systems are the example of external senesors and used to identify range, altitude, direction or speed.

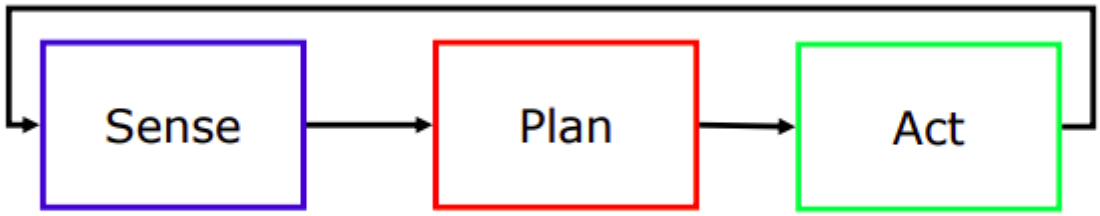
Laser Rangefinders

- It is used to detech the objec in the path and to get measurement like how far the object is from the laser.
- laster rangefinder are based on some methodologies:
  - Traingulation: The use of geometric relationships between the outgoing light beam, the incoming ray, and its position on the film plain.
  - Time of flight: The measurement of the time delay for an outgoing light ray to hit a target whose distance is being measured and returned.
  - Phase-based: based on the diff between the phase of the emitted and reflected signals.

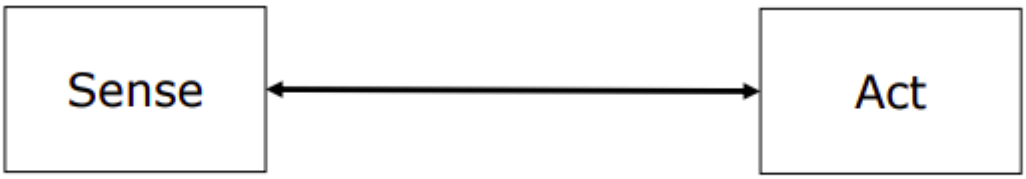
Bilogical Sensing

- Bilogy is a vast stream which provides inspiration for the most of aritificial intelligence as well as robotics.
- Sensory stimuli in animals are geenrally used for the purpose of diverse objectives, the behavious is classifie into two basic classes: tropism and taxis.
- The term topotaxis was developed to refer to reactions that are directed with respect to a stimulus.
- Further it leads to a familiy of some more behavioural categries:
  - Telotaxis: repersents a turning response
  - Astrotaxis: represents alignment to celestial objects
  - Polamtaxis: represents orientation effect
  - Mnemotaxis: represents a learned direction change

Robot System Control

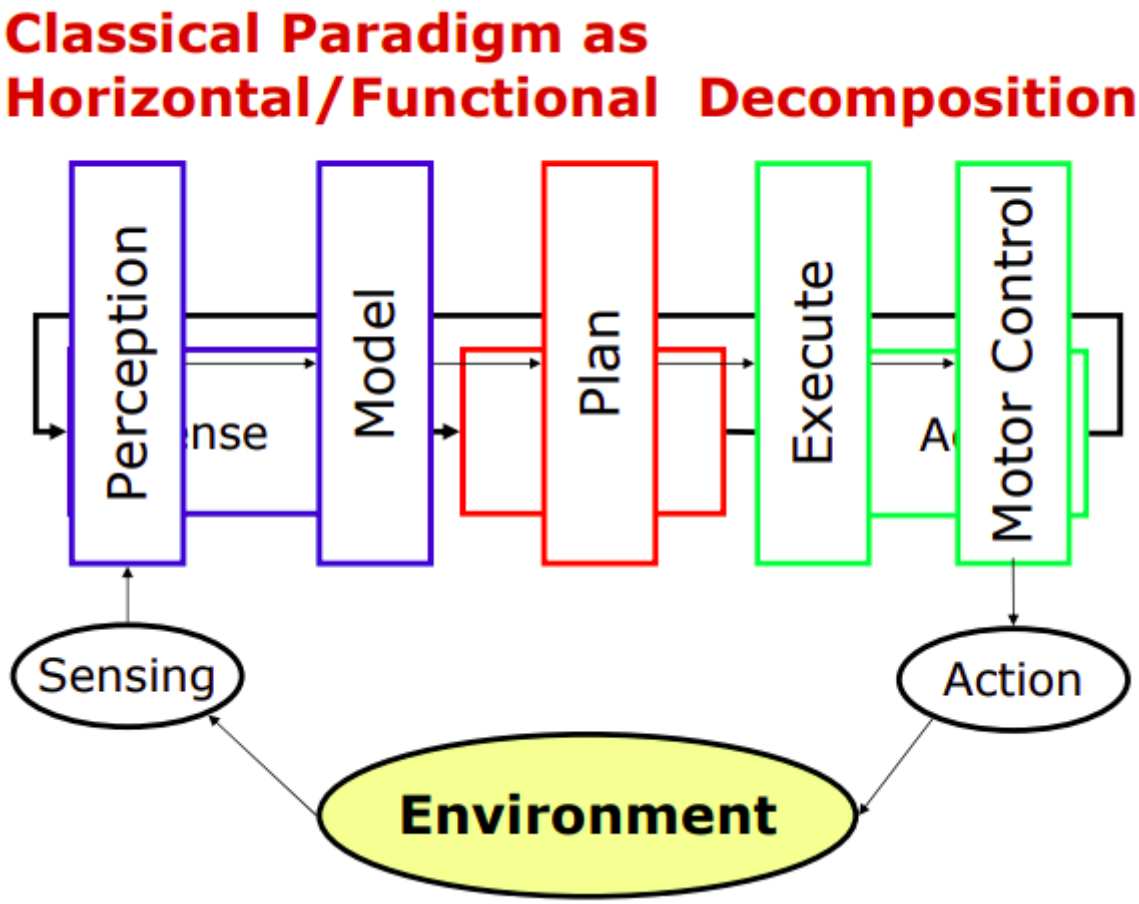


Reactive / Behavior-based Paradigm



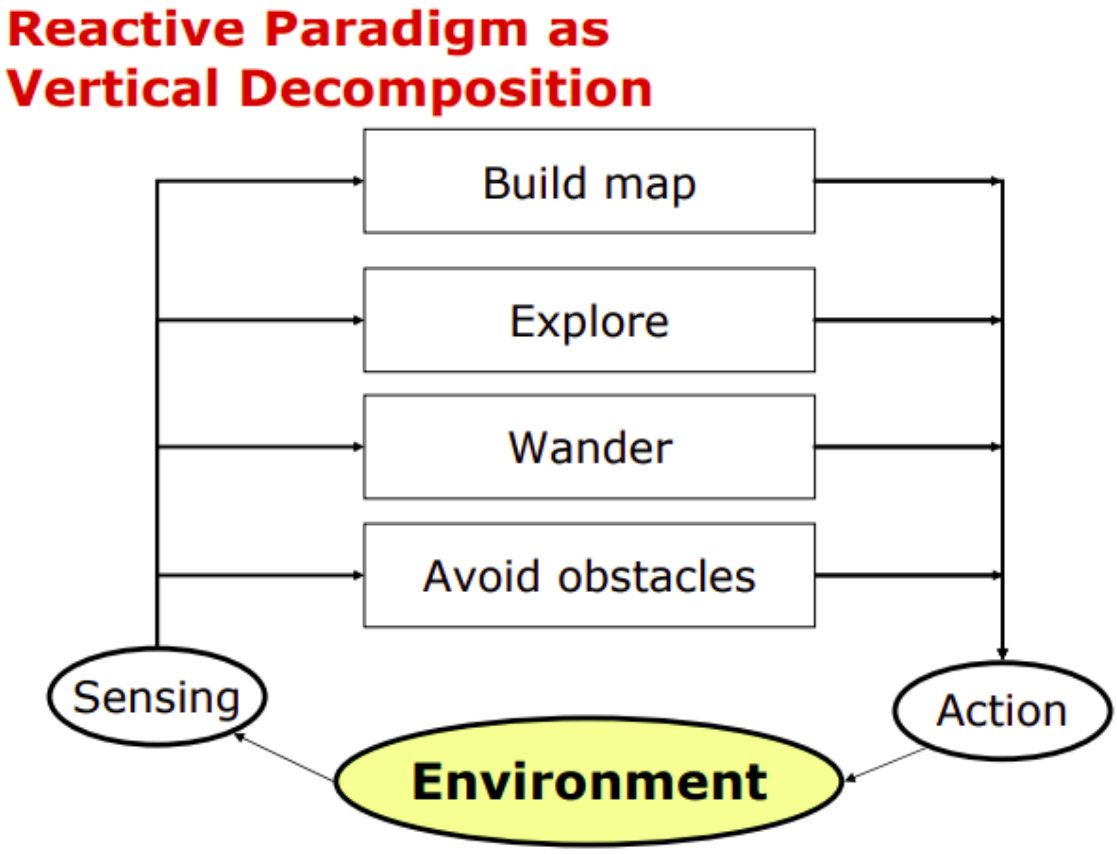
Horizontoal Decomposition

- Also called as functional decomposition



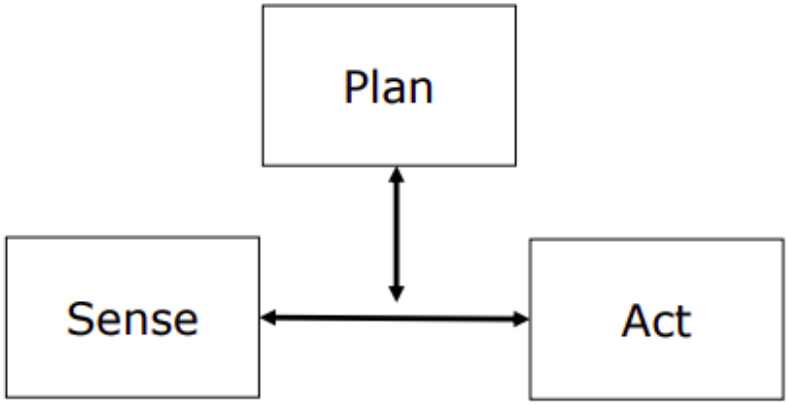
Vertical Decomposition

- also known as behavioral decomposition
- There are some problems with classifical planner:
  - Multiple goals
  - Multiple sensors
  - Extensibility
  - Robustness

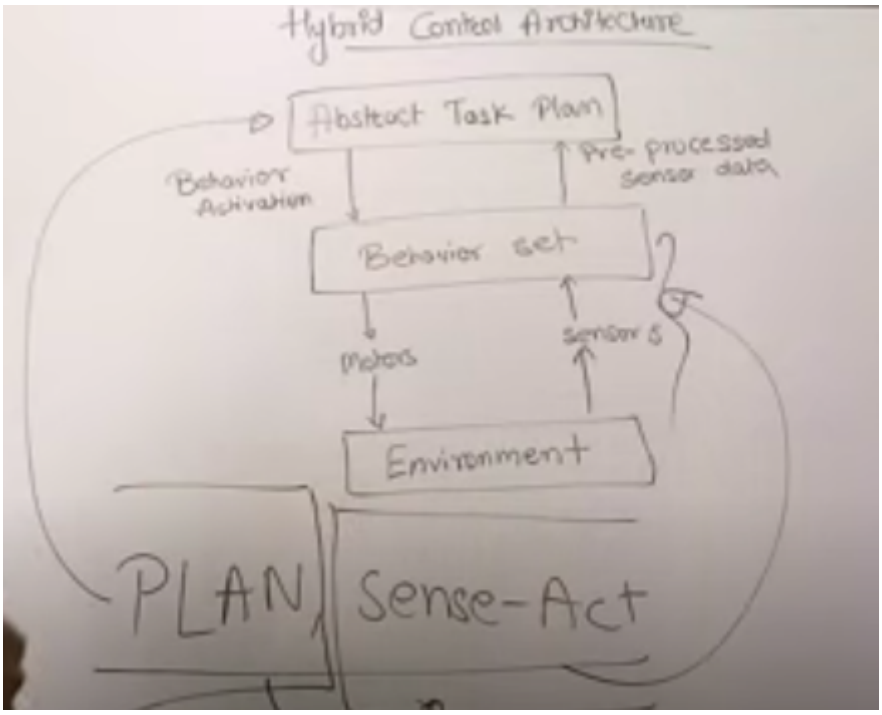


- Behavious are the direct mapping of sensory inputs to motor outputs.

Hybrid Control Architecture



- Combines advantages of previous paradigms
  - World model used for planning
  - Closed loop, reactive control



- Hybrid architecture combines the reactive control with the abstract task planning.
- Abstract task planning layer makes the decisions and plans and the reactive behavior layer provides the reactive actions and handles the sensores and actuators.

Middleware

- Middleware is a computer sofware that is used to connect software components or applications.
- Middleware is nothing but the collection of services which allow multiple process running on one or more machien to interact.
- In robot there are diff harware and software components and generally one software is running in robot which handle all the tasks like reading sensot data, perception, modelling, planning, controlling motor for actions, etc.

- So this makes code reuse, code sharing and maintainence of the code very difficult.
- In middleware whole application is divided or structured into many little tasks and then these tasks communicate with each other throught the communication channel provided by middleware.
- A wide range of projects for robotics middleware exist:
  - Player Project
    - The player project originally known as Player/Stage Project.
    - It is used to create sofware for research into robotics and sensor systems.
    - Its components include player network server and stage robot platform simulators.
  - RT-middleware
    - RT-middleware is a common platform standard for robots based on distributed oject technology.
  - Urbi
    - open-source cross platform software created in c++
    - Used to develop the application for robotics and complex systems.
    - It is based on UObject distributed c++ component architecture.
    - It also includes urbiscript which ia a parallel and event driven script language.
  - MIRO
    - distributed object oriented framework for mobile robot control based on CORBA (Common Object Request Broker Architecure).
  - Orca
  - OpenRDK
    - OpenRDK is an open-source software framework for robotics for developing lossy coupled modules.
  - ROS
    - Robot Operating System
    - It is a collection of sofwtare framework for robots, providing OS like functionlity.
  - YARP
    - open-source software package written in c++

AIR
Unit 6

Page :
Date : / /

Contents

1. Localization
2. Types of localization
3. Simple landmark
4. Mapping
5. Levels of mapping
6. Robots in practice
   a. Space Robots
   b. Agriculture / Forestry
   c. Domestic
   d. Mining
   e. Delivery
   f. Intelligent
   g. Autonomous Aircraft

☆ Localization. Classification.

1. Localization → A mobile robot must be aware of its own location so that it can perform various tasks / fun^n on the basis of particular requirment.
2. This is also suitable for specific ~~plans~~ paths or to identify whether the location is suitable or not
4. Robot localization indicates the capability to set its own location and direction within framework.
5. There are diff types of localization as follows

1. **Relative L** → In this robot attempts to find out the location based on its position from the last point. If it can locate the start position then eventually it can also track last/end position. In this ~~sem~~ with the help of the sensors the data can be taken as input.

2. **Absolute L** → Completely opposite to relative L., it does not uses the previous position rather with the help of external sensors it tries to approximate the location.

3. **Local L** → In this it follows the location of a robot from an initial point. with the help of previous location it decides the new location. with respect to adjacent locations of its last position.

Page:
Date: / /

4. Global L → It does not require any previous data to decide the location it can decide its on its own

5 Possive L → when robot is finding its location with the help of sensors and you cannot manipulate the robots direction, sensors or performance then its Passe L.

6 Active L → Opposite to passive, when robot is finding its location with the help of sensors and you can also control the robot's performance, sensors, direction then it is called as active L.

7. Known Environment → If the robot is kept in known environment then it is very easy to locate itself as it will be knowing the coordinates and can execute the task more precisely and efficiently in less time.

8 Unknown E. → whereas if robot is in unknown E. then it has to calculate the position / location and then carry out the task, so in this situation it becomes a bit difficult for him to conc.

9. **Static Environment** → ~~The~~ ~~area~~ Objects which are not moving in environment i.e which is static in nature and only moving obj is your robot which is following a defined path.

10. **Dynamic E.** → In environment where objects are also moving with robot is called as Dynamic E. In this it becomes a little bit complex to perform tasks/actions.

11. **Natural E.** → Natural robots are not particularly organized or skilled for robots. It remains unaffected with the presence of robot. So the computation task is a little bit complex in natural E.

12. **Artificial E.** → Unlike natural ~~case~~ environment, the artificial E is made according to the robot. They are specially made for robots so in these environments robots can perform tasks better than natural E.

13. **Strong L.** → The identification of exact location of the robot has a lot of speical implication that involves compucation etc and finds the ~~exact~~ ~~exact~~ exact location without approximation.

14. **Weak L.** → The localization in which robots are familiar with the existing location that has been followed previously is called weak L.

Page:
Date: / /

Simple Lankmark Measurement.

1. Landmarks are nothing but features corresponding to distinct objects in physical world.
2. Landmarks are used for navigation purposes in robotics.
3. There are different types of classes of landmark.

1. **Active** → This landmark acts like sender, it sends the signals which comprises of information about robot & environment

2. **Inactive** → opposite to active it does not give more information about environment.

3. **Natural** → To take this type of landmark into consideration, a mobile robot must be capable of computing high & complex tasks, as there is a lot of diversity in nature a single tree can be seen through diff perspective, so to compute this it needs high computation power.

4. **Artificial** → It is kind of info. providing type. These are the landmarks made for robots. There are active and passive types. Active means the components who take active participation in forwading the information about environment & passive is opposite of this

☆ Mapping.

1. Mapping is nothing but a spatial model of the environment.
2. It estimates the landmark given the robot pose.
3. i.e according the robots location it creates maps according to the landmarks. so that specific tasks can be Performed.
4. Maps generally represent structural aspects in few abstract areas may be with meaningful labels.
5. There are 2 types of maps

1. Metric maps.

1. These kind of maps are based on coordinates
2. and are basically 2D in nature
3. These maps are sensitive to noise.
4. And it is difficult to calculate the distance precisely.

2. Topological maps.

1. They are referred as relational maps.
2. Because it shows the places & relationship betn them.
3. It includes very clear relation to tasks & meaning of problem.

Page:
Date: / /

Levels of Maps.

1. **Sensorial** → Signals which represent raw data or domain conversation are referred as sensorial graph.

2. **Geometric** → Signals which represent 2D & 3D objects gathered from sensor data.

3. **Local Relation** → Signals which represent operational, structural meaningfullness relations among geometric objects.

4. **Topological** → Extensive relational association that attach objects & positions accross the complete environment.

5. **Semantic** → operational tags related with the elements of the map. It is in either 2D or 3D. It is also called as perceptual maps. Map discovery is imp. for a lot of sensible mobile robotics functions

Robots in Practice.

1. Space Robotics

1. The external space is a perfect environment to use the autonomous robotic system.
2. When it comes to space exploration, human life is very precious so why take risk instead we can send robots to do the same work.
3. It is desirable to discover our nearbly planet with the help of ~~ar~~ robotics & AI.
4. Some of the successful missions conducted by space Robots are
   1. Mangalyan (2) NASA/JPL (3) Soviet.
5. Whenever humans life is in danger we can replace that work with robots.

2 Agriculture/Forestry.

1. Tasks which are carried out in agriculture will be in natural environment, so the robots have to localize and mark landmarks accordingly.
2. As the natural environment is tough to understand robot ~~of~~ which can solve or have capability of solving complex tasks can be used.
3. Also, due to this cost of the operation increases.

a. Forest maintainence →
   1) For weeding (2) Sowing seeds (3) costly opn
   (explain these points) ↗

Page :
Date : / /

b Lawn cutting Robots.

1. Useful for rigorous agriculture.
2. Made for huge golf, football & cricket ground
3. It drives on vehicle & uses GPS & gyroscope for specific patterns.

c Harvesting Robots.

1. It requires large amount of time & money.
2. height problem in harvesting.
   eg. fruits etc.

Domestic Robots.

concept

1. Efficient domestic robots does not exist in realit, but a large no. of cleaning robots are present for performing various applications.
2. It can be used floor polishing, street cleaning & vaccuming
3. Best example of domestic Robots are Robosoft & Roomba.
4. Robosoft is a independent vaccum cleaner & Roomba is used for household work.

Mining Automation

1. A portable benefit to the mining automation is
1. Provides better protection for workers.
2. superior production from the mine.
3. Less equipment maintence cost.
4. 2 types of mining
   4.1) Open pit mining (4.2) underground mining.

4.1 a) open pit mining is categorized by huge, bulky & costly equipment known as dragline
b. Automatic dragline system most likely represent the biggest robots in the world.
c. It is used to move bulky material over draglines

4.2 a) Underground Mining is categorized by the condition which are required for performing hazardous tasks in a rigid environment.
b Due to risk of rockfall, isolated underground mining is generally performed in recent times.
c. Also in underground oxygen is less so small ~~robotic~~ robots can be used instead of humans.

Page:
Date: / /

Delivery Robots.

1. As the name suggests these robots are used to deliver the objects.
2. There are 2 types of delivery
(1) Light work (2) Heavy duty.
3. Light work is the tasks where the tasks are easy to complete & privacy is also maintained eg. delivering medicine & ~~patients~~ patient's file.
4. Working in random / dynamic environment is one of the biggest challenges in delivery.
5. Although some scientists have developed "Helpmate Robot" which is used in Hospitals for delivering files & medicine
6. with the help of sonar & video sensor for the movements
7. Another examples are automated guided vehicle & cargo handling.
8. These types of robots can be very useful and they can perform much more better than humans in some conditions.

Intelligent Vehicle.

1. Smart / Intelligent cares. with driving assistance
   eg. ABS, GPS, blinker etc
2. next level - drivrless car with all sensors
   GPS et

(explain this points in exam)

Autonomous Aircraft.

1. support to militory appⁿ -> by using good type ~~appⁿ~~ appⁿ to figure out the info about enemy's database and found & destroy
2. support to civilian appⁿ with autonomous appⁿ for analysis, assesment in city
   eg. for checking the traffic, pollution etc.

(expand these points)

Unit 6

Servo Control

- The easiest method for using sensor information and particularly vision to associate a robot's pose with a landmark
  is defined as image-based servoing or visual servoing or sensor-based servoing.
- Servoing is generally used to eneable a robot to shift to an exact target location with the help of practical sensor dimensions which is refferes as **homing**.

Recursive Filtering

Techniques of recursive filtering:

- Classification of data points
- Adaptive localization at multiple scales
- Weighted voting of correction vectors
- convergence of the estimate
- Multi-scale local correction results
- Comparative behaviour
- Quality Measures

MCQ's Important Poins

- If a robot has k legs, then the number of possible events is N = (2k-1)!
- For a robot unit to be considered a functionalindustrial robot, typically, how many degrees offreedom would the robot have? 6
- MLP (Multi Layer Perception) is a feed forward network.
- L293D is a motor driver IC
- Principles of cybernetics - Norbert wiener
- Robot with catesian coordinate system - 3 linear movements
- Robot with catesian cylindrical system - 2 linear movements and 1 rotational
- Robot with catesian polar system - 2 rotational and 1 linear movements
- ZMP stands for Zero moment point
- Robot is derived from Czech word Robota
- leading robotics intitute in America - CMU
- drive for lighter class of robot - pneumatic drive
- not a programming language for computer controlled robot - AMU
- unmanned aerial vehicle (UAV) or uncrewed aerial vehicle
- Types of hill climbing algorithms
  - Simple hill Climbing:

- Steepest-Ascent hill-climbing:
- Stochastic hill Climbing: