ARTIFICIAL INTELLIGENCE

HANDWRITTEN NOTES

- MODULES COVERED

(1) Intro to Artificial Intelligence (AI)
(2) Intelligent Agents
(3) Solving problem by searching
(4) Knowledge & Reasoning
(5) Reasoning Under Uncertainty
(6) planning & learning.

By - [in] - Osama Khan

Page No.
Date

1. Intro to Artificial Intelligence (AI):

* (AI)

→ Artificial intelligence is simulation of human intelligence processes by machines, espesially computer systems. Specific applications of AI includes expert systems, natural language processing, speech recognition and machine vision

Advantage

- → Good at detail-oriented jobs;
- → Reduced time for data-heavy tasks
- → Delivers consistent results
- → AI-powered virtual agents are always available

Disadvantage

- → Expensive
- → Requires deep technical expertise
- → Lack of availiblity to generalize from one task to another.

* parts of AI

Artificial: Non natural
Intelligence:- Ability to learn understand and think

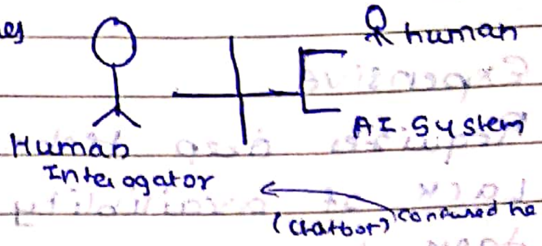
* Categories of AI

System that acts like human
" " think like human
" " think rationally
" " act rationally

* AI Approaches / perspective

(1) Acting humanly (The turing test approach)

→ The art of creating machines that perform functions that requires Intelligence when performed by people.



Q1 Thinking humanly: (The cognitive modelling approach)

insano ki tarah sochna

→ The exciting new effort to make computers think machine with minds, in full of literal sense.

Human brain ⟺ Computer system

(3) Acting Rationally: (Rational agent approach)

Sensor, actuator

→ The branch of computer Science that is concerned with the automation of intelligent behaviour.

- Doing behaving Rightly
- Generalized approach
- Maximum expected performance

(4) Thinking Rationally (The law of thought approach)

→ The study of computations that makes it possible to perceive reason and act.

hamesha sahi Rehna (100)

- Right thinking
- Requires 100% knowledge
- To many computations required

History of AI

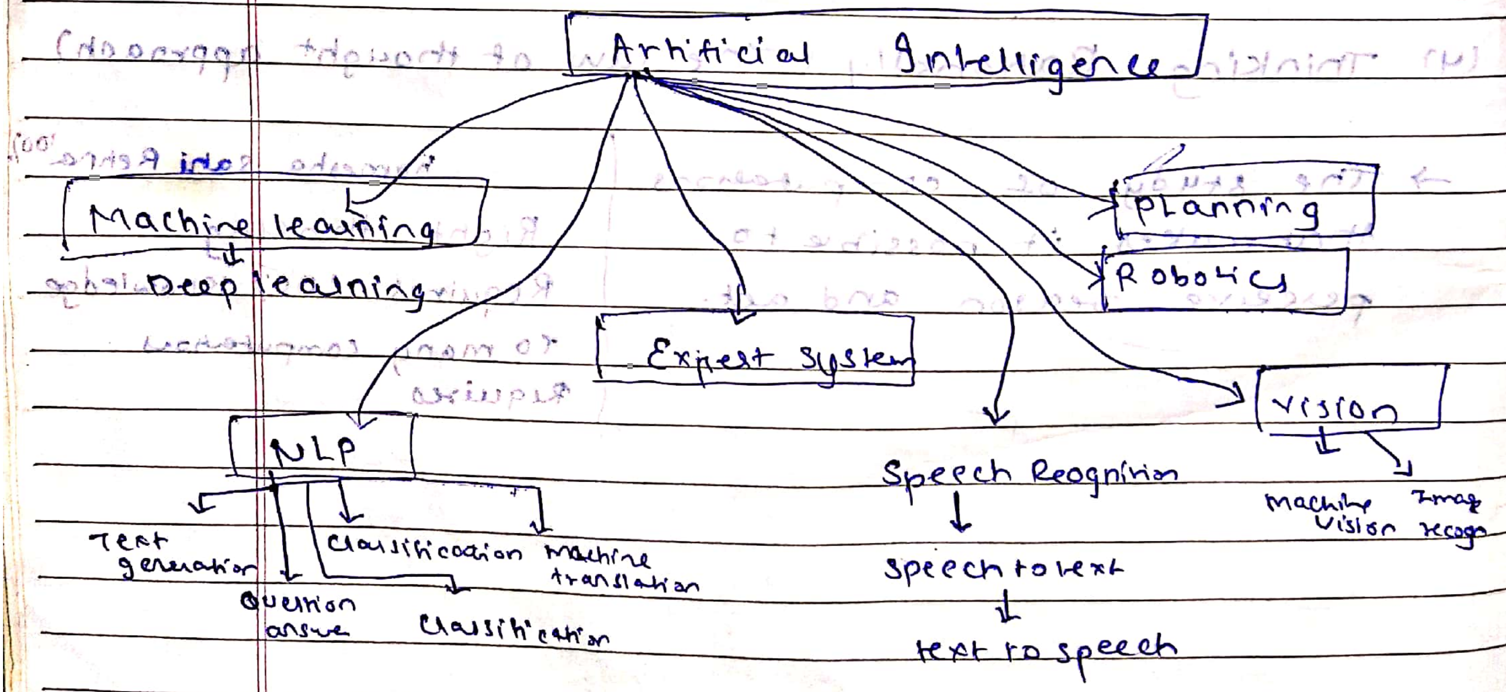
1950 In 1950 Alan Turing proposed Turing test and same year Issac Asimov proposed three laws of robotics

1951 The first AI program is written

1955 The first self learning game playing program created

1955 MIT AI Lab is set up

continued in ppt. - - - -



* present state of AI

Knowledge base
control strategy
Inference Mechanism

* Applications of AI

(1) AI in transport & Tourism ⟶ website, hotels suggest
Chatbot in website
(2) AI in self Driving car ⟶ koi samne aj age to break
(3) AI in E-commerce ⟶ link to e-commerce multiple chize
(4) AI in Gaming ⟶ chess, X&O playing with computer
(5) AI in Entertainment ⟶ Netflix me koi movies suggest karna aur series

* Ethics in AI

AI ethics is a system of moral principles and techniques intended to inform the development and responsible use of artificial intelligence technology.

1) Transparency
2) Respect for human values
3) fairness
4) safety
5) Accountability
6) privacy

Environment

2. Intelligent Agents

* Introduction to Agent

→ An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators

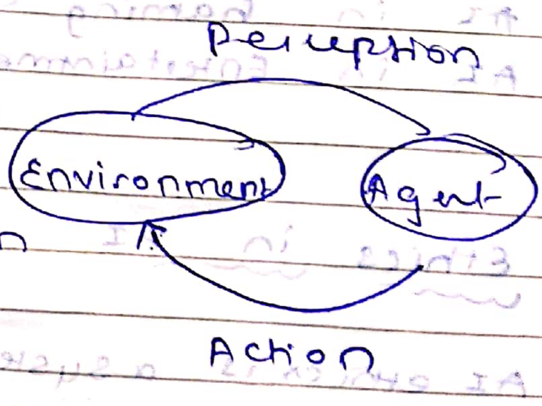
Human agent! sensor:- Eyes, ears, nose, skin
Actuator: legs, hands

- operates in an environment
- perceive its environment through sensors
- Acts upon its environment through actuators/effectors
- Has Goals.

* Structure of Intelligent Agent

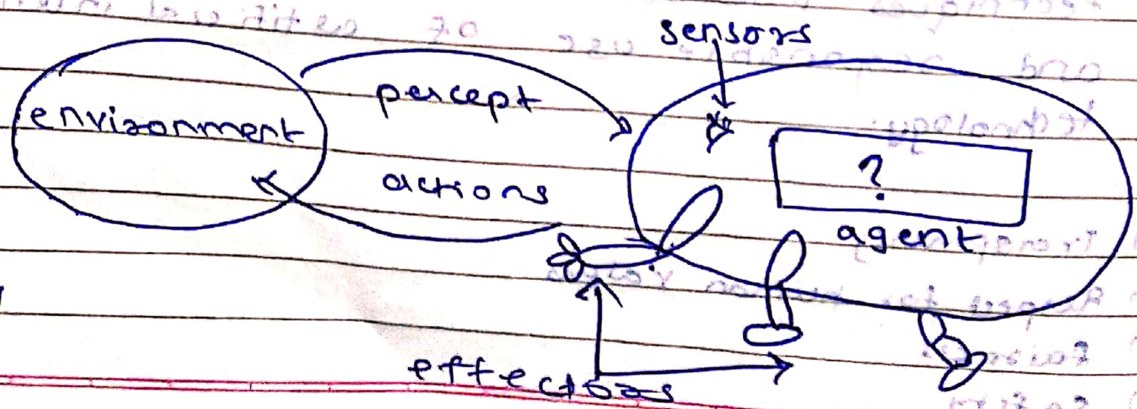
* Must sense
Must act

An agent perceives its environment through sensor and acts upon that environment through actuator.



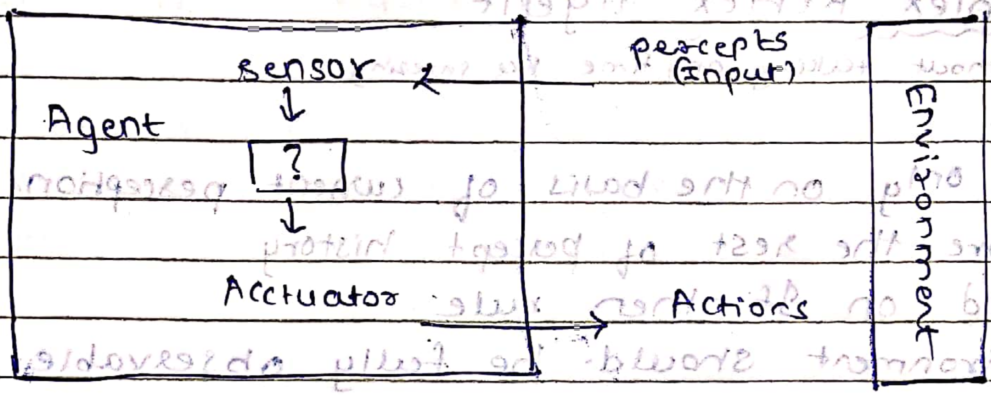
Agent Diagram Architecture

Imp



Imp

Diagram/Architecture Agent



* Characteristics of Intelligent Agent

- learning / Reasoning
- reactivity
- autonomy
- Goal - oriented

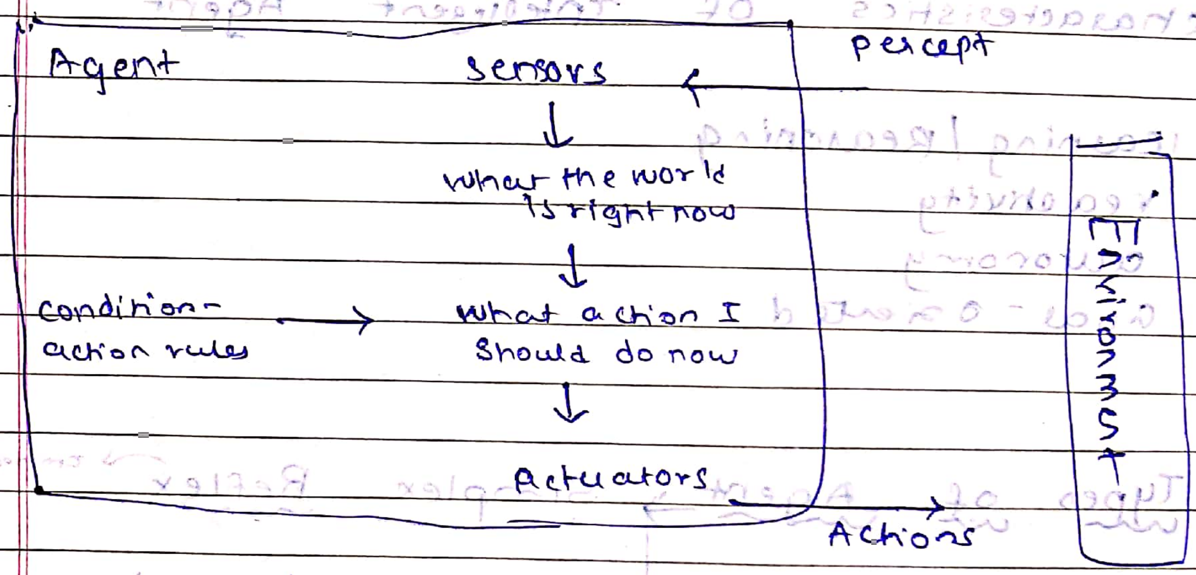
* Types of Agent

- Simplex Reflex → Immediately
- Model based
- goal based
- utility based

(1) Simplex Reflex Agent

without telling any time like sneezing

- Acts only on the basis of current perception.
- Ignore the rest of percept history
- Based on If - then rule
- Environment should be fully observable.

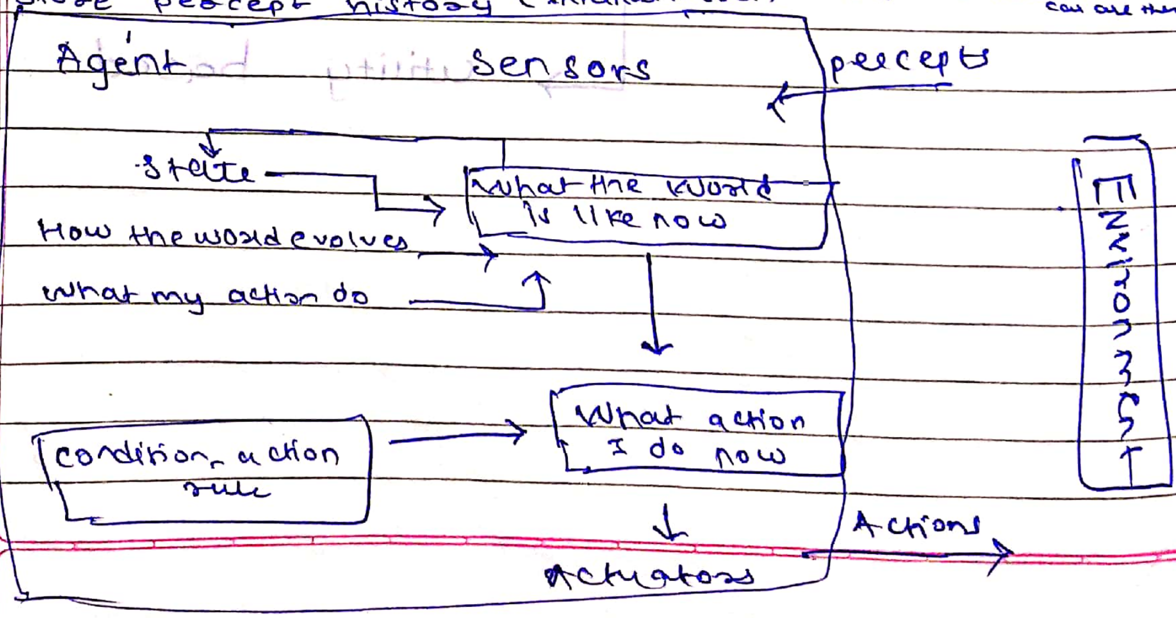


(2) Model Based reflex Agents.

Knowledge based

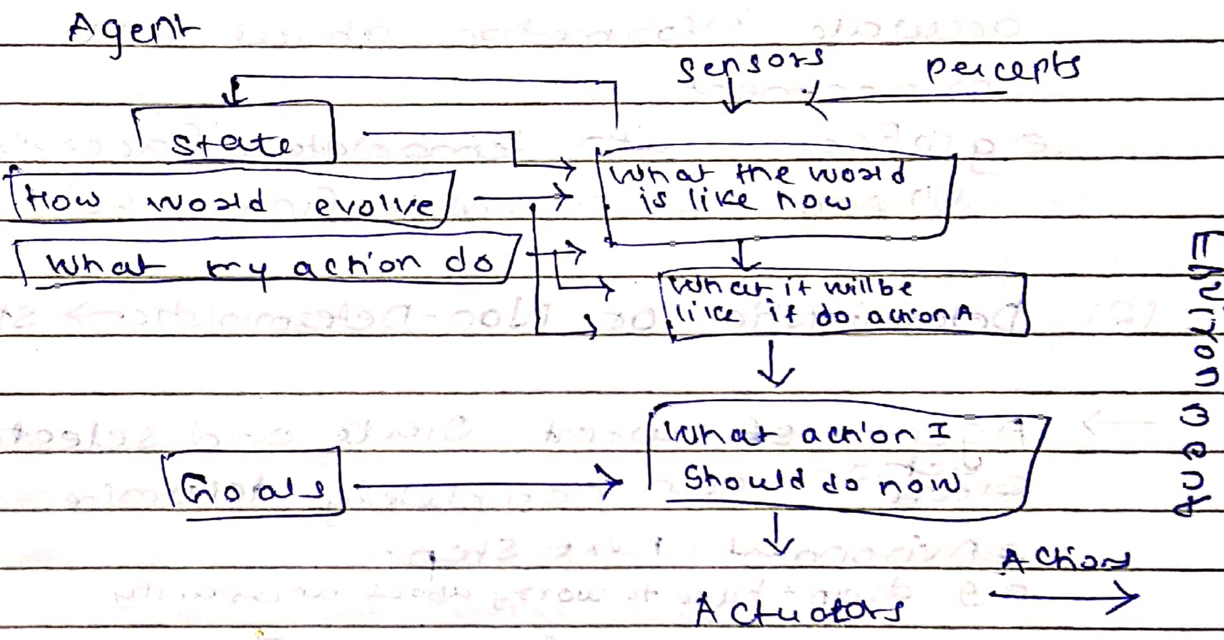
- partially observable Environment
- store percept history (Internal model)

e.g like self driving does not know how many car are there in road.



(3) goal based Agent.

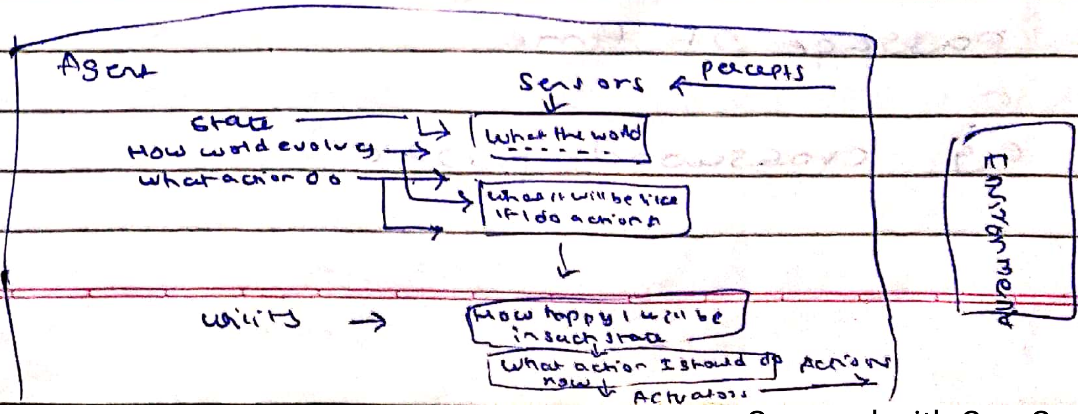
- Expansion of Model based Reflex Agent
- Desirable situation (Goal)
- Searching and planning e.g best route find karna pehle se pata ho destination
  e.g ~~Robot~~



(4) utility Based Agent

- Focus on utility not goal
- utility function
  Deals with happy and unhappy state

e.g GPS take short Route to happy the Agent.



* Environment

Environment is part of the universe that surrounds intelligent system

(1) Accessible

→ Agent can obtain complete and accurate information about state environment

e.g (i) Room with temperature [accesible]
(ii) Any event on earth [inaccessible]

(2) Deterministic or Non-Deterministic → stochastic

→ Agent ~~of~~ current state and selected action ~~state~~ can completely determine environment next step.

e.g doesn't have to worry about uncertanity

* Non-deterministor [stochastic]

→ nature of environment cannot be decided by agent alone.

e.g Random in nature

(3) Static

→ Env. doesn't change its state with passage of time

e.g crossword puzzle

Page No.
Date

(4) Dynamic

→ State of env. changes ~~with~~ w.r.t time

e.g car driving → traffic zyada mile ya kam mile?

(5) Discrete continuous

→ finite no. of percept and action
e.g chess game

Agent can perceive and make observations from the env. continuously without any lag

e.g [self driving]

(6) Observable

↳ Agent sensor can access complete state of env. at each point of time
(fully or completely)

(7) partially observable

↳ In this some part of env is not in the reach of agent.

(8) Single

If only one agent is involved in an env. & operating by itself then such env is called single agent

(9) Multi

→ If multi agent are operating in an environment then such env. called multi-agent environment

3. Solving problem by searching

* problem solving

AI most popular 1960 to most research on problem solving like Tic-Tac Toe, waterjug problem, 8-Queen, that the game solve by human and machine both are called problem solving area

* State space search

S: { S, A, $\underbrace{\text{Action (S)}}_{\text{what action taken}}$, $\underbrace{\text{Result (S,a)}}_{\text{path of direction}}$, $\underbrace{\text{cost (S,a)}}_{\text{min cost}}$ }

precise → Represent the knowledge of game
Analyze → Easy to take step

1	2	3
4	5	
6	7	8

Initial State

→

1	2	3
4	5	6
7	8	

goal State

→ Two type of searching
(1) uninformed → Blind search (More time)
(2) Informed → Quickly solve (less time)

* problem formulation

→ It is one of the core step of problem-solvi. which decidees what action should be taken to acheive the goal.
Initial State
Goal state
Action
Transition
path costing.

* Solving problem by searching

5	4	
6	1	8
7	3	2

Initial state

→

1	2	3
4	5	6
7	8	

Goal state

* performance evaluation

→ In AI, complexity is expresed by three factor
b → branching factor
d → depth
m → max. length

* completness :- Is it guaranteed that our algorithm always finds a solution when there is one.

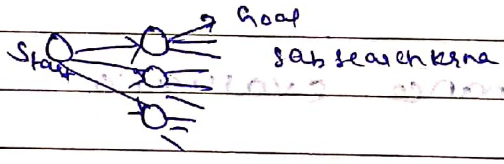
* Optamality :- Optimal solution

* Time complexity :- How much time it takes

* Space complexity :- How much memory required

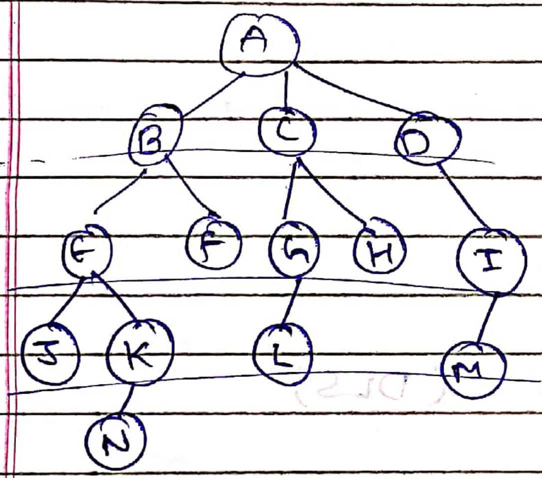
* Uninformed Search

→ Uniformed Search (i.e brute force, blind)
Takes more time
No additional information about no. of step
Time consuming (Time, space).
DFS bfs etc



Breadth first search (BFS)

- BFS is an uninformed search
  no knowledge of specific domain
- FIFO (Queue)
- Shallowest Node → level by level
- optimal → shortest Result
- Time complexity = $O(b^d)$



level by level

← A ←

~~B~~ C D

~~C~~ D E F

~~D~~ E F G H.

~~E~~ F G H I

~~F~~ G H I J K

~~G~~ H I J K

~~H~~ I J K L

~~I~~ J K L M

~~J~~ K L M

~~K~~ L M

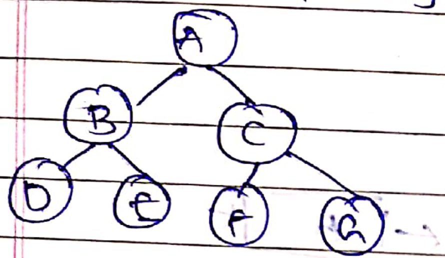
~~L~~ M N

~~M~~ N

N

* Depth first search (DFS)

- uninformed search technique
- stack (LIFO)
- Deepest Node
- Incomplete → NO solution
- non optimal
- Time complexity $O(b^d)$

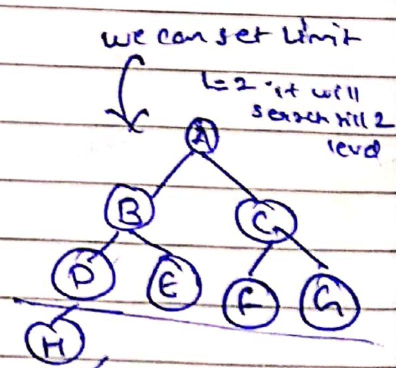


A B D E C F G

IMP

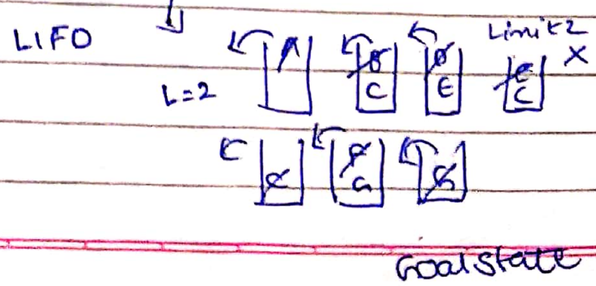
* Depth Limited Search (DLS)

We can set the limit
Time complexity $O(b^d)$
work on LIFO



Advantage
- Solve problem of infinitely deep path with no solution of DFS
- Memory efficient

Disadvantage
- It can be incomplete if solution is below depth limit

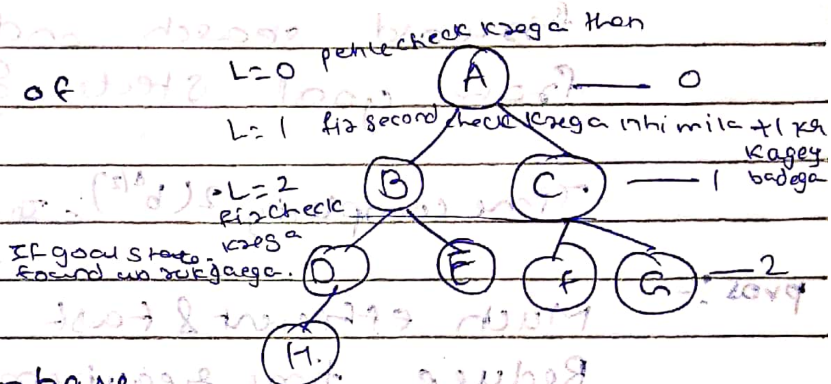


* Iterative Deepening Search (DFID) *

- It works on LIFO
combination of DFS & BFS

Advantage

- It inherits Advantage of both DFS and DLS

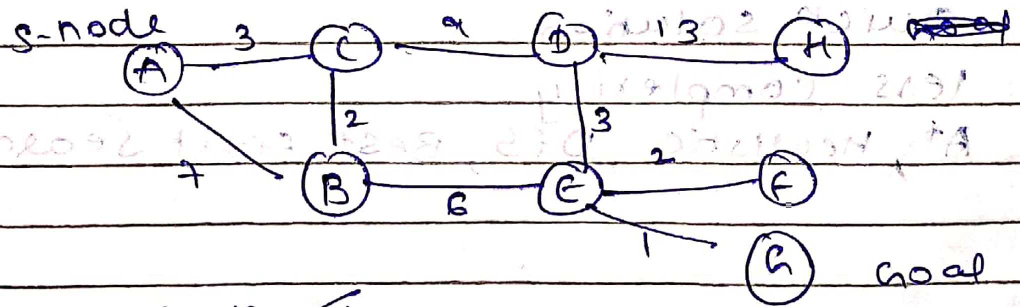


Disadvantage

- Keeps repeating process of previous phase

* Uniformed cost search (UCS)

Uniformed cost search can be achieved by implementing the stack as priority queue ordered by path cost
It check the shortest path to any node



~~A - C - D - H~~
~~3 + 9 + 13 = 25~~

A - C - B - E - G
= 12

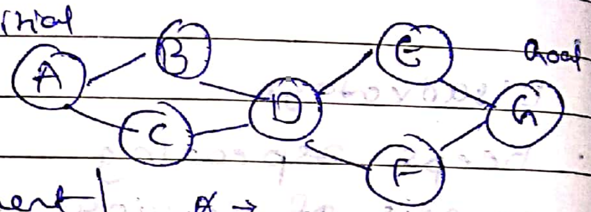
* Feature
Always Reach to goal node
use priority queue
backtracking is possible

Advantage
optimal solution
disadvantage
can get stuck in infinite loop

* Bidirectional Search

In bidirectional search two simultaneous searches are run.
One start from initial state called ~~and~~ forward search and other start from goal state called ~~go~~ backward search.

Time complexity $2(b^{d/2})$



pros:- Much efficient & fast
Reduce time requirement

~~A~~ →
C ~~B~~ →
D ~~F~~ →
~~B~~ →

~~G~~ E
~~E~~ F ←
~~F~~ D ←
D ←

cons:- Implementation is difficult
Goal must be known in advance

A B C D F E G answer

* Informed search → Have proper guide

- Search with information
- Use knowledge to find step to solution
- Quick solution
- less complexity
- A*, Heuristic, DFS, Best first search

Imp

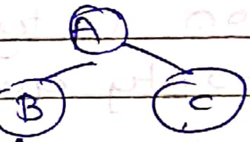
* Heuristic search & function

It is a function that gives an Estimation of cost of getting from node n to the goal node

e.g (i) A → B → C → D

$h(n)=3$ $h(n)=2$ $h(n)=1$

ii)



path cost 4 h(n=4)

h(n=3)
path cost 3
so I will choose this
∴ The path which gives me minimum cost.

Types of Heuristic function

(i) Admissible : $(h(n) \leq h'(n))$
path cost ko underestimate karta he

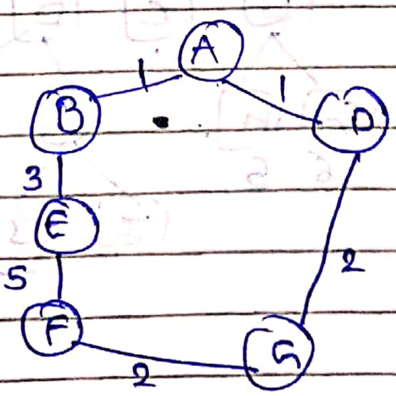
(ii) Non-Admissible $(h(n) > h'(n))$
path cost ko overestimate karta he

$h(B)=3, h(D)=5$

h(B)=3

h(E)=2

h(F)=3



h(B)=3
actual = 11 → It takes 11
3 < 11

h(D) = 5
Actual = 3

5 > 3

e.g 8 puzzle
Route from one distance to other

Scanned with CamScanner

IMP

* Best first search (BFS)

It is a combination of Breadth first search and depth first search.

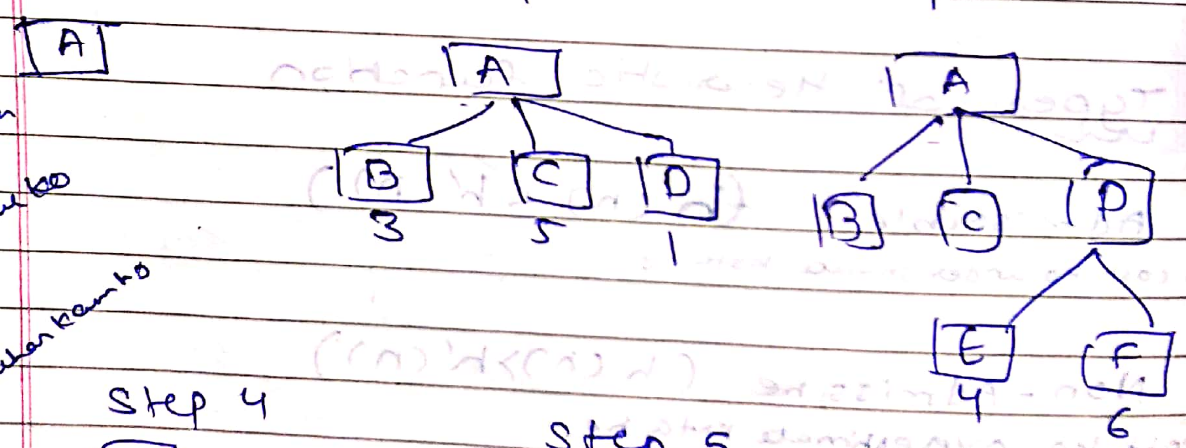
Best first search is algorithm which explore the search tree by expanding the most promising node chosen according to the heuristic value of node.

Efficient selection of the current best candidate for extension is typically implemented using priority queue.

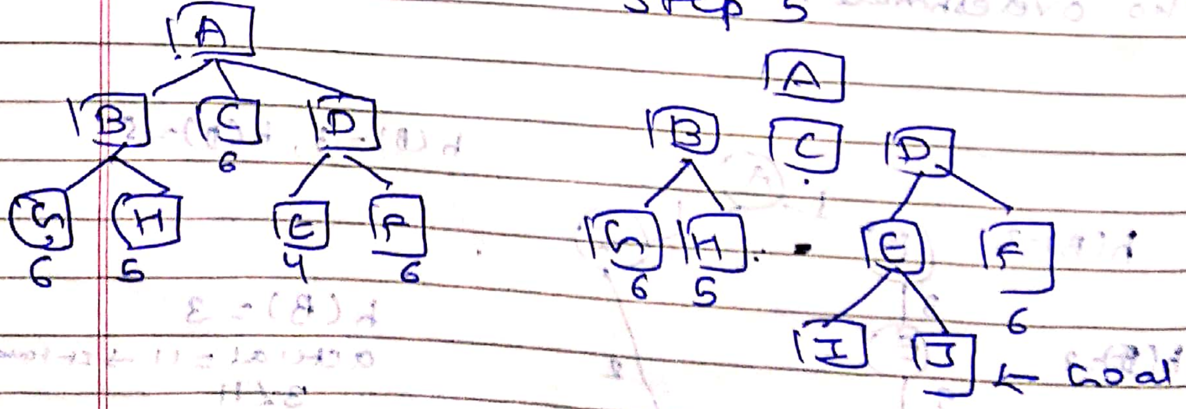
sab se kam path ko choose krega. compare krega sab se kam heuristic value ko

ek path se dusre path me ja sakta jahan kam ho

Step 1 Step 2 Step 3



Step 4 Step 5



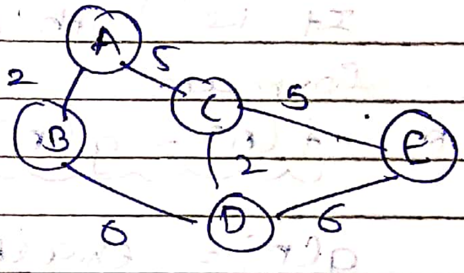
Greedy Best first search

A greedy algorithm is an algorithm that follows the heuristic of making the locally optimal choice at each stage with the hope of finding a global optimum.

Evaluation function $f(n) = h(n)$ (heuristic).

$f(n) = h(n)$

Actual cost = 10
heuristic $f^n \to 10$



No extra nodes are expended
Optimal soln found

$h(n) <$ actual cost
$h(n) \leq h^*(n)$

* A* Search

- It is the combination of Dijkstra's Algo and Best first search
- A* search find the shortest path through a search space to goal state using heuristic function.
- A* algo finds the lowest cost path that passes through particular state
- It is both complete and optimal.

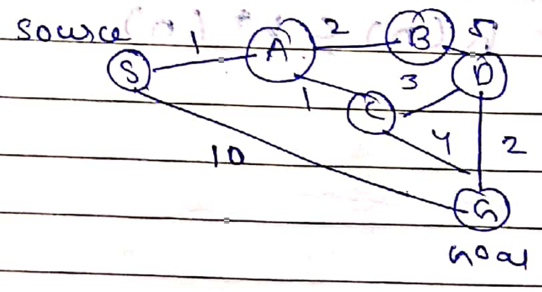
formula

* $f(n) = g(n) + h(n)$

$g(n)$ = exact cost
$h(n)$ = estimation of remaining cost

Numerical

e.g



State	h(n)
S	5
A	3
B	4
C	2
D	6
G	0

→ 1st iteration:- S-A → $f(n) = g(n) + h(n)$
$= 1 + 3 = 4$ → Ex
S-G → $f(n) = g(n) + h(n)$
$= 10 + 0 = 10$ → hold

2nd Iteration SA-B → $= 3 + 4 = 7$ → hold
SA-C → $= 2 + 2 = 4$ → ex

3rd Iteration :- SAB-D = 5+6=11 - hold
SAC-G = ~~5~~6+0=6 - ex

path = S - A - C - G
cost = 6

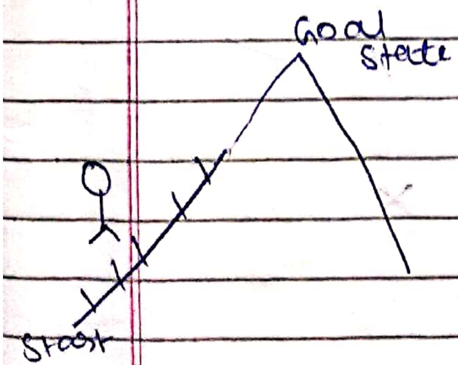
* Local search → Hill climbing search
→ Simulated Annealing search

local search method work on complete state formulations. They keep only a smaller number of nodes in memory. It is useful for optimization
e.g N-Queen

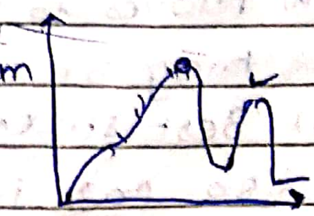
Hill climbing (local search Algo, greedy Approach, No backtrack)

1. Evaluate the initial state
2. Loop until a solution is found or there are no operators left
{If better then current state then it is new current state}

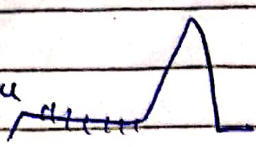
limitations:



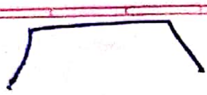
1) local maximum



2) plateau



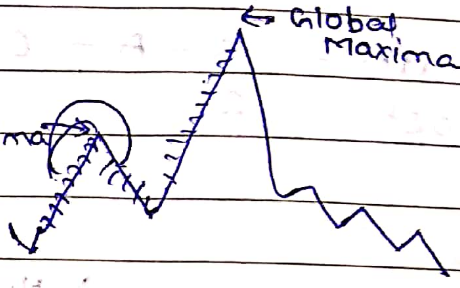
3) ridge



Simulated Annealing search

← pick aa ke check krega kahan zyada height he

- checks all neighbous
- It allows downward step



- Annealing is a process in metallurgy where metals are slowly cooled to make them reach a state of low energy where they are very strong

Advantage	Disadvantage
(1) Easy to code for complex problem	(1) slow process
(2) Gives Good soln	(2) can't tell whether an optimal soln is found
(3) Statistically guarantees finding optimal soln.	(3) some other method also required.

Difference

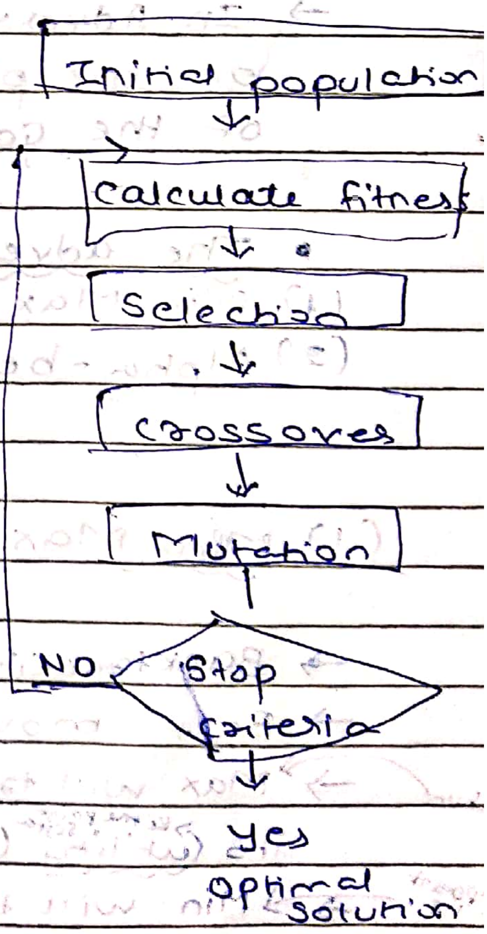
	Simulated Annealing	Hill climb
(1)	annealing schedule is maintained	×
(2)	moves to worst state may be accept	×
(3)	Best state found so far is also maintained	×

* Genetic Algorithm (John Holland)

works on Genes.

→ Abstraction of Real biological Evolution
→ Solve complex problems (like N-P hard)
→ focus on Optimization
→ population of possible solutions for a given problem
→ from a group of individuals, the best will survive

$$\text{phenotype} \underset{\text{Decode}}{\overset{\text{Encode}}{\rightleftarrows}} \text{Genotype}$$



e.g cake
make
Same ne. are 3

a b c d e
f g h i

* **Game playing** → Adversarial Search Technique
→ Min-Max Search
→ Alpha-beta pruning

→ It is a game-playing technique where agents are surrounded by a competitive environment.

→ Game playing mean where human-intelligence & logical factor is used.
e.g. tic tac toe, chess, ludo.

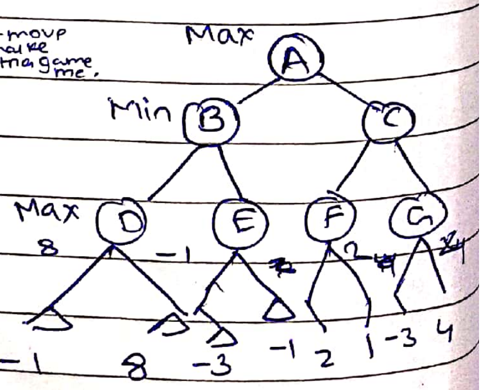
→ To complete game one has to win and other looses automatically.

→ In Adversarial search the result depends on the player which will decide the result of the game.

• The adversarial search are
(1) Min-Max Search
(2) Alpha-beta pruning.

(1) Min-Max Search

→ Backtracking algorithm
→ Best move strategy used → Best move chalke jitna game me.
→ Max will try to maximize its utility (Best move) → ki tme point sejta
method
→ Min will try to minimize utility (worst move)
opponent he



Time complexity $O(b^d)$

max min

Alpha - beta pruning ($\alpha - \beta$)

cutoff search by exploring less no. of nodes

Traverse ~~in order~~ the search tree in depth-first order

prune whenever $\alpha \geq \beta$

Max node update alpha

Min node update beta

Scanned with CamScanner

4. Knowledge & Reasoning

KBA (Knowledge Based Agent)

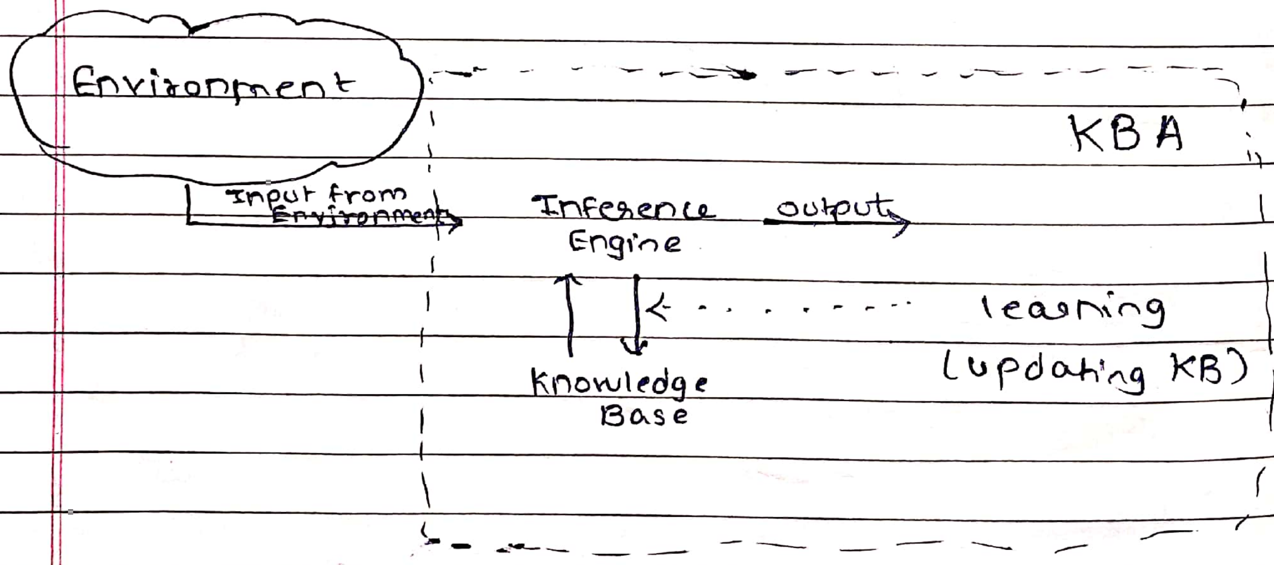
- An intelligent agent needs knowledge about the real world for taking decisions and reasoning to act efficiently.

Two main parts

(1) Knowledge ~~System~~ base.

(2) Inference System.

(1) * The Architecture of Knowledge-based agent.



Page No.
Date

Inference System

Inference system: deriving new sentences from old
Inference system allows us to add a new sentences to knowledge base.
There are two Rules:-
Forward chaining
Backward chaining

Various knowledge - based agent

Knowledge level
logical level
Implementation level

Properties of knowledge based agent

Represent states, actions etc.
Incorporate new percept
update internal representation of the world
Deduce hidden properties of the world
Deduce appropriate actions

Knowledge Representation Rules

Knowledge Representation is part of AI which concerned with AI agents thinking

Two levels
Knowledge level
Symbol level

FOL
FOPL others Rules
fuzzy (1) logical representation
If then (2) production rule represen
(3) semantic networks
when task becomes complex it use (4) frame representation

double sentence

* propositional Logic (Either True or false)

would like our AI to have knowledge about world and logically draw conclusion from it.

logical constant : True, false
propositional symbol : P, Q, S . . . (atomic sentences)
Wrapping parenthesis (. . . . . .)
connectives

	$\wedge$	.... and	[conjunction]
	$\vee$	... or	[disjunction]
if then	$\Rightarrow$	implies	[implication]
iff	$\Leftrightarrow$	is equivalent	[biconditional]
	$\neg$	not	[negation]

* (1) Syntax → Atomic 1+1=2 = T
→ complex

* (2) Semantics

A model specifies which of proposition symbol are true and which are false

a	b	NOT(a)
F	f	True
F	T	True
T	F	False
T	T	False

Tautologies

A sentence is tautology if it is true for any setting of its propositional symbols.

P	Q	P OR Q	NOT (P) AND NOT (Q)	(P OR Q) OR (NOT(P) and NOT(Q))
F	F	F	T	T
F	T	T	F	T
T	F	T	F	T
T	T	T	F	T

Modal Logic	Temporal Logic
Higher Order Logic	
first order predicate Logic	
propositional Logic	

First order
* predicate Logic — single sentence

First order logic does not only assume that the world contains facts like propositional logical but also assume the follg thing in the world.

Object, Relations, function

* Two main part
(1) Syntax (2) semantics

- Atomic e.g Ravi and Ajay are brother => Brothers(Ravi, Ajay).
- complex
  - subject :- main part of sentence.
  - predicate : relation

e.g x is an integer.
$\underbrace{x}_{\text{sub}}\ \underbrace{\text{is an integer}}_{\text{predicate}}$

* Quantifier in First-order Logic

A universal quantifier (for all, everyone, everything) ($\forall$)
E Existential Quantifier (for some, atleast one) ($\exists$)

Page No.
Date

(1) universal Quantifier $(\forall x)$ → sentence me ye ana chahiye

→ universal Quantifier is a symbol of logical representation, which specifies that the statement within its range is true for everything

→ It is represented as $\forall$

→ If x is a variable, then $\forall x$ is ~~reddas~~ read as

- for all x
- for each x
- for every x

e.g $\forall x$ man(x) → drink (x, coffee).

→ There are all x where x is a man who drinks coffee

(2) Existential Quantifier $(\exists x)$ $\cdot \wedge$ sentence me ye ana chahiye

→ Are types of quantifier, which express that statement within its scope is true for atleast one instance of something.

→ It is represented as $\exists$

- There exists a 'x'
- for some 'x'
- for atleast one 'x'

e.g some boy play cricket :

$\exists x$ boys (x) → plays (x, cricket)

Page No.
Date

* Inference in FOL

Inference in first-Order Logic is used to deduce new facts or sentences from existing sentence

* Some basic Inference Rule

(1) Universal Generalization :- It states that if premise p(c) is true for any arbitrary element then $\forall x\, p(x)$

It can be represented as $\dfrac{p(c)}{\forall x\, p(x)}$

e.g A byte contain 8 bit

$\forall x\, p(x)$ it is true.

(2) Universal Instantation:- It can applied multiple time to add new sentence

It can represented as $\dfrac{\forall x\, p(x)}{p(c)}$

e.g All King who are greedy are Evil

→ $\forall x\ King(x) \wedge greedy(x) \rightarrow Evil(x)$.

(3) Existential Instantiation:- applied only once to replace existential sentences.

It can represented as $\dfrac{\exists x\, P(x)}{P(c)}$

(4) Existential Introduction:-

$\dfrac{P(c)}{\exists x\, P(x)}$

Wumpus

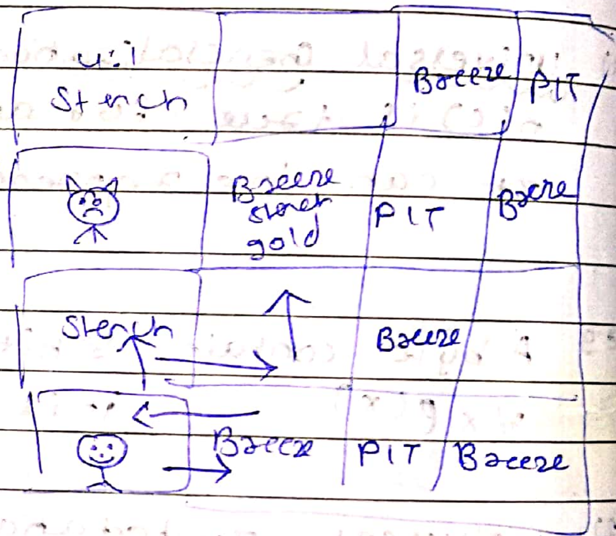
Examples

Elements

Wumpus, strench, Agent, pit, Breeze, gold
Arrow.

Awards

+100 pts if agent comes out with gold
-1 point on agent every action
-10 point if arrow used
-200 point if agent dies



PEAS properties

Environment

Empty Room
Room with wumpus
stenchy Room
Breeze Rooms
Room with gold
Arrow

Sensor:
camera, odor sensor, Audio

Effector:
(i) Motor to move R, L
(ii) Robotic Arm to grab gol
(iii) Robotic mechanic to shoot Ar

Page No.
Date

(1) forward chaining (Question se answer tak jana)
(2) Backward chaining (Answer se Question tak jana)

* Forward chaining
When Based on the available data a decision taken then the process is called forward chaining
It works from a Initial state and

ect to fol

(1) It is crime for an American to sell weapon to enemy of America
→ American (x) ∧ Weapon (y) ∧ sell (x, y, z) ∧ enemy (z, America) → Criminal (x)

- country nono is an enemy of America
→ Enemy (Nono, America)

- Nono has some missile
→ owns (Nono, x) Missile (x)

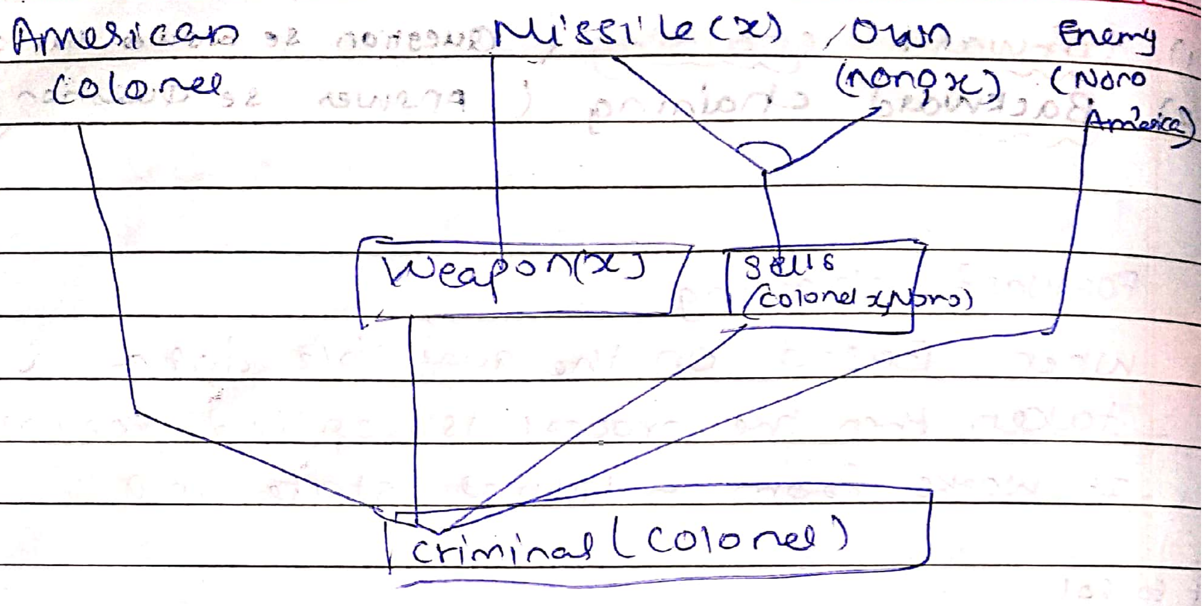
- All missile were sold to Nono by colonel
→ ∀x Missile (x) ∧ Own (Nono, x) ⇒ sell (colonel, x, Nono)

- Missile is a weapon
→ missile (x) ⇒ weapon (x)

- colonel is American
→ American (colonel)

Scanned with CamScanner

Page No.
Date



American colonel
Missile(x)
Own (Nono,x)
Enemy (Nono America)
Weapon(x)
Sells (colonel x,Nono)
criminal (colonel)

Page No.
Date

Resolution

Resolution procedure is a sound and compl inference procedure for FOL.

* Steps to convert FOL to CNF → Conjuctive Normal Form

(1) Eliminate implication
(2) Standardize Varible
(3) Move negation inwards
(4) Skolemization
(5) Drop universal Qualifier

(1) Eliminate implication

$\alpha \rightarrow B = \neg\alpha \vee B$

$\alpha \leftrightarrow B = (\alpha \rightarrow B) \wedge (B \rightarrow \alpha)$

(2) Standardize variable

$\exists(x)$ simile(x)	$\exists(x)$ simile (x)
$\exists(x)$ Graduating(x)	$\exists(y)$ Graduating (y)
$\forall(x)$ happy(x)	$\forall(z)$ happy(z)

(3) Move Negation inwards

$\neg(\forall(x)\ P(x)) \equiv \exists(x)\ \neg P(x)$

$\neg(\exists(x)\ P(x)) \equiv \forall(x)\ \neg P(x)$

$\neg(\alpha \vee \beta) \quad \neg\alpha \wedge \neg\beta$

$\neg(\alpha \wedge \beta) \quad \neg\alpha \vee \neg\beta$

$\neg(\neg\alpha) \quad \alpha$

Page No.
Date

(4) Skolemization (Remove Existential Quantifier)

(1) $\exists (x)$ simile (x)
(2) $\exists (y)$ Graduating (y)

(1) Ans Smile (A)
(2) Ans Graduating (B)

(5) Drop Universal Quantifier.

(1) $\forall x$ (smile (x))
(2) $\forall y$ Graduating (y)

After dropping
Smile(x)
Graduating (y)

IMP
IMT

Q. • All people who are graduating are happy
• All happy people smile
• Some one is Graduating

(i) convert to fol
(ii) convert FOL to CNF
(iii) Prove "is someone smiling?" using resolution
(iv) Draw Resolution tree

• $\forall x$ (Graduating$(x) \rightarrow$ happy (x))

• $\forall x$ (happy $(x) \rightarrow$ smile (x))

• $\exists x$ (Graduating) (x)

always take ∃ (negation) when prove ask?

We need to prove that is someone smiling

$\neg \exists x \; smile(x)$

Convert FOL to CNF

Eliminate Implication $\alpha \to \beta \equiv \neg\alpha \vee \beta$

$\forall x [\cancel{\;}\; Graduating(x) \vee Happy(x)]$

$\forall x [\neg happy(x) \vee smile(x)]$

$\exists x \; Graduating(x)$

$\neg \exists x \; smile(x)$

standardize

$\forall x [\neg Graduading(x) \vee Happ(x)]$

$\forall x [\neg \cancel{Grad}\,happy(y) \vee smile(y)]$

$\forall x \; Graduating(z)$

$\neg \exists w \cdot smile(w)$

Move Negation inward

same

same

same

$\forall w \; \neg smile(w)$

skolemization

same

same

Graduating (A)

same $\forall w \; \neg smile(w)$

Drop universal Quantifiers

$\neg Graduating(x) \vee Happy(x)$

$\neg happy(y) \vee smile(y)$

Graduating (A)

$\neg smile(w)$ (1)

Now the Sentence is in CNF

Resolution tree

¬ smile (w) | ¬happy(y) ∨ smile(y)

(θ : y/w)

¬ happy (w) | Graduating(x) ∨ ¬Happy(x)

(θ : x/w)

¬ Graduating (w) | Graduating (A)

Null

Hence someone is smiling.

Page No.
Date

5. Reasoning under Uncertai

* uncertainity

→ Many time in complex world theory agent and events in environment contradicted to each other, and this in reduction of performance measure.

e.g Accident happen during Journey this called as uncertainty.

More e.g

Toothache → cavity

Not all have cavity

Toothache → cavity V gum problem - - - -

* probability

probability provides a way of summariz ~~date~~ the uncertainty that comes from our and ignorance, there by solving the qualification problem

The priori probability

The probability of the event comput before the collection of new data.

e.g If 0.01% population has ... then

Page No.
Date

posterior probability

Adjust probability accordingly is called posterior probability.

Bayes Theorem

Bayes theorem considered both the prior probabi of an event and diagnostic value of a test to determine the posterior probability.

$$P(D/T) = \frac{P(T|D) * P(D)}{P(T|D) * P(D) + P(T|\bar{D}) * P(\bar{D})}$$

From PPT.
Q.

Given:-
→ $P(M_1) = 30/50 = 0.6$
$P(M_2) = 20/50 = 0.4$
$P(D) = 1\% = 0.01$

As a chance both machine produce defective bulb is equal.

$\therefore P(M_1/D) = 50\% = 0.5$
$P(M_2/D) = ?$

$$P(D|M_2) = \frac{P(M_2|D) * P(D)}{P(M_2)}$$

$$= \frac{0.5 \times 0.01}{0.4}$$

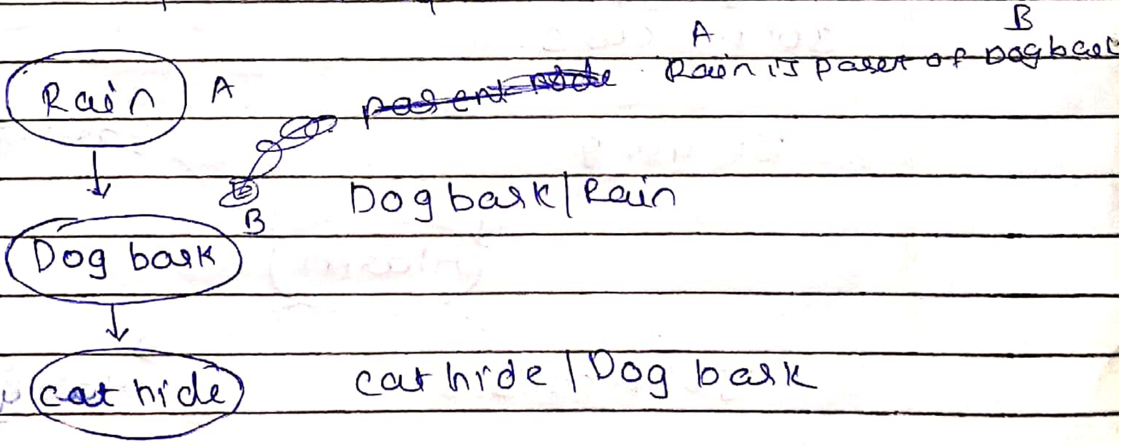
$$= 0.0125$$

* Bayesian Belief Network.

bayesian belief network is key computer technology for dealing with probablistic event and to solve a problem which has uncertainty we can define a Bayesian Network as:-

It is Made up of
1. Directed Acyclic Graph
2. conditional probability Table

(I) Directed Acyclic Graph



(2) conditional probability Table

~B not bark
B = Bark
R = Rain
~ not Rain

	R	~R
B	9/48	18/48
~B	3/48	18/48

$(B = T \,\&\, R = T) = 0.19$

$(B = T \,\&\, R = F) = 0.375$

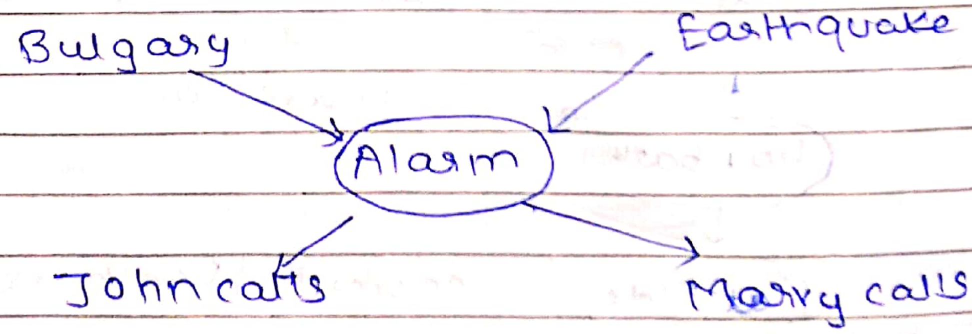
$(B = f \,\&\, R = T) = 0.06$

$(B = f \,\&\, R = F) = 0.375$

Example of Bayesian Belief Network.

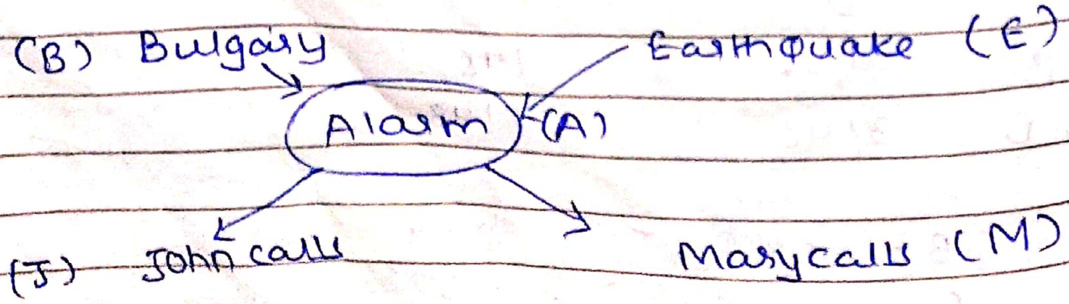
Assume your house has an alarm system against bulgarary. you live in the sesimically active area and the alarm system can occasionally set of by an earthquake. you have two neighbours Mary and john. who don't know each other. If they hear the alarm they call you but this is not guaranteed.

we want to represent probability distribution of events.
Bulgary, Earthquake, Alarm, Mary calls
John calls.



Directed Acyclic graph

Nodes = Random variable
Links = direct dependencies



~~P(B=T) = 0.001~~ ~~P(E=T) = 0.002~~
~~P(B=F) = 0.999~~ ~~P(E=F)~~

local conditional distribution

$p(B=T) = 0.001$ $\quad$ $p(E=T)$ 0.002

$p(B=f)$ 0.999 $\quad$ $p(E=f)$ 0.998

B	E	p(A=T)	p(A=f)
T	T	0.95	0.05
T	f	0.99	0.06
f	T	0.29	0.71
f	f	0.001	0.999

A	p(J=T)	p(J=f)
T	0.90	0.10
f	0.50	0.50

A	P(M=T)	p(M=f)
T	0.70	0.30
f	0.01	0.99

$p(J, M, \sim B, \sim E)$

$p(J|A) \cdot p(M|A) \cdot p(A|\sim B, \sim E) \cdot P(\sim B) \, P(\sim E)$

$= 0.90 \times 0.70 \times 0.999 \times 0.998$

$= 0.00062$

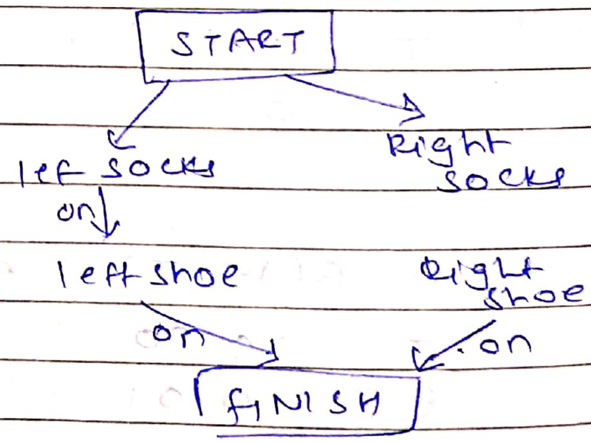
6. planning & Learning

6.1 partial order problem / Total order problem

partial order planning

- we take one path
  work on several sub-goal independently
  solve them with sub plan
  combine the sub plan
  Least commitment strategy

delaying a choice during search



1 2 3
3 2 1
2 1 3 → we can delay search

Total order planning

- we take multiple path
  Forward backward state space search are form of total order planning
  cannot take advantage of problem decomposition

Start	Start	Start	Start	Start	Start
Right sock	Right sock	left sock	left sock	Right sock	left sock
left sock	left sock	Right sock			
Right shoe	left shoe	Right shoe			
left shoe	Right shoe	left shoe ↓			
finish	finish	finish			

6.2 learning AI, supervised, unsupervised, semi, Reinforcement Ensemble learning.

* Supervised : We have Input and output on this basis we try new model of Input for validity.
E.g Exists pole

All age group → tell who can win
take data

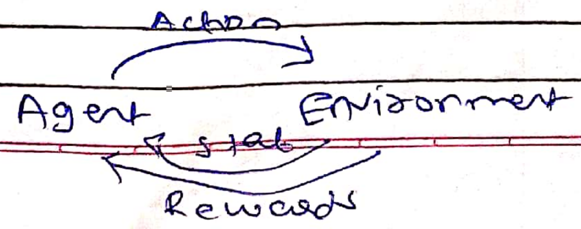
- Both Input & output
- classification who can win

* unsupervised: Not having a knowledge in Real world.
e.g p.k movie hero doesn't know about Earth and sensing he make decision and clustering between men & women & child.

- only Input knowledge
- clustering K-Mean

* Reinforcement: Reinforcement learning is based on rewarding desired ~~bated~~ behaviours or undesired ones
e.g. trial & error

- Reward
- R learning



Page No.
Date

Semi-supervised learning

comibnation of supervised & unsupervised learning. pickup large unlabeled dataset label a small portion of dataset put unlabeled dataset into clustering using unseperuised learning Algo.

E.g.

Google photo take all photo and cluster Each image by looking same face.

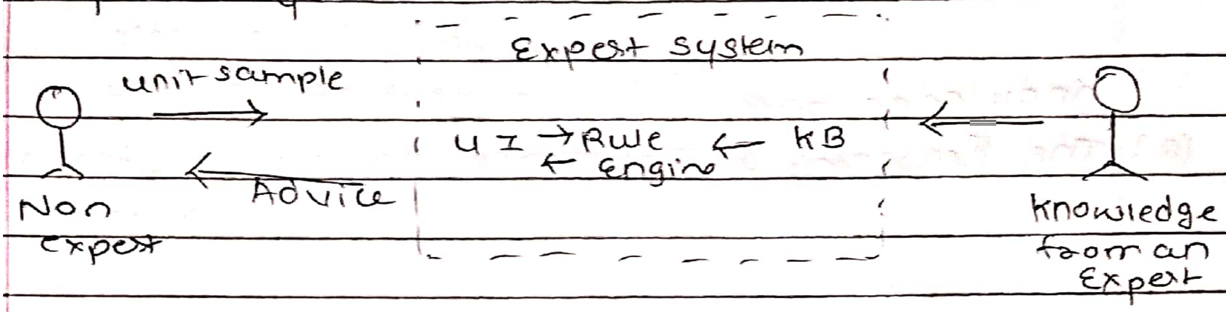
learning :- It can Improves its performance on future tasks after making observations about world.

forms of learning

(1) component: which component need to be improved
(2) knowledge: what prior knowledge agent already have
(3) Representation: what Rep... used for data & components
(4) feedback: what feedback is available to learn from.

Scanned with CamScanner

Expert system



Components

- UI
- Inference Engine
- Knowledge base

Characteristic

- Simulation of human Reasoning
- Representation of human knowledge
- Use approximation
- provide ~~e~~ explanations
- Expertise
- Symbolic Reasoning
- Deep Knowledge
- Self Knowledge

ES-shell

Knowledge Acquisition subsystem — Knowledge base (Facts, Heuristics) — Inference Mechanism (Reasoning with Uncertainty) — Explanation subsystem → uses Inter face — U S ...

Knowledge Engineer