Agent program

types:

① Simple reflex agent

② model based reflex agent

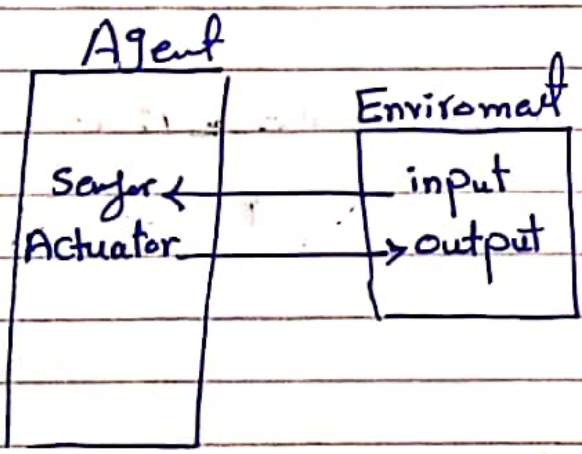
③ goal based agent

④ utility based

طريقة انك تبعد عن فكرة ال table

بنعتمد بناءاً على ال Enviroment properties

① Simple Reflex Agent



الى بيحصل جوه ال Agent انه بياخد ال Input، وبيشوف ايه ال Action المناسب بناءاً على condition Action Rules وبطلع ال Action.

- Simplest kinds of agents [Based on condition Rules]
- Selects action Based on only Current precept (ignoring History)

Interpret_input ←

Function generates an abstracted Description of the Current state from the percept.

لازم فانكشن ترجم الـ Input عشان الـ Agent يقدر يفهمه ويحلله

Rule_Match ←

function returns the first rule in the set of rules that matches the given State Description.

فانكشن ترجع الـ Rule اللي بتناسب الـ Description بتاع الـ Input اللي جالها

- Works only in Full observable Enviroment

ميتفعش يخش اى حاجة لازم ياخد كل الـ Inputs

Function SimpleReflexAgent (percept) returs Action.

(Input) (output)

static : rules (Set of Condition Action Rules) ← الشروط الثابتة

state ← InterpretInput (percept)
rule ← RuleMatch(state, rules)
action ← RuleAction [rule]

return action

Subject : .... Date: / /

[2] Model Based reflex Agent

(هو بردو مازال reflex
يعني معتمد على ال conditions
بس معقد شوية عن ال simple

- Handle partial observability →

ممكن يعكس ال full observable
عشان بيركز على ال History مش على ال current بس

- Maintains internal state that depends on percept History

- 2 types of knowledge must be encoded to update the internal space

1. information about how the world evolves Independently
   ازاى البيئة بتتغير بعيدًا عن ال Agent
2. informations about how the agent's own actions affect the world.
   ازاى البيئة هتتغير في وجود ال Agent

Important:

Description of current state = Current percepts + old internal state

function: Input output

Model-based-reflex-agent (percept) returns action.

static: state, الحالة قبل ما يدخل الـ Agent
model, الحالة بعد ما الـ Agent يتدخل
rules (set of conditions)

state ← update-state(state, action, percept, model)
Rule ← Rule match(state, rules)
action ← Rule.Action(rule)
return action.

من هنا للآخر هيفرق معاك قات

Note • Uncertainty about the current state may be Unavoidable
ممكن دايمًا مكون واثق من الـ Action ده 100% بس الـ Agent لازم يتعرف

ده هيخلي الـ Agent بتاعي مجرد رد فعل للـ Environment وبعد ده اللي إحنا عايزينه بعد

goal based هتشتغل ازاي الـ Agent

[3] Goal-based Agents

ده مش جاي يرد على الـ Enviroment بس ، ده جاي يحقق goal معين ، وعنده stept محددة عايز يعملها . فهياخد اعتباره بعد كده خطوات الـ goal ، وهو بيختار الـ Action

- the Agent program can combine the goal with the model.

← في الحالة دي هيحصل كذا rule Match ، هنشوف اللي يحقق الـ goal ونختارها

Function goal_Based_Agent (percept) returns Action

Static: state → description of current state
rules → condition action rules
action → most recent action
goal

state ← updateState (state, action, Percept, model)

rules ← ruleMatch (state, rules)

actions ← rule action [rules]

action ← Choose - Action (goal, actions)

بدل كل الـ matched بختار منهم الأقرب للـ goal المناسب

return action.

- Searching and planning are used to achieve the agent's goal.

Decision Making

is different from condition action rules

involves consideration of the future

- This Agent is more flexible because the knowledge that supports its decision is represented explicitly and can be modified

توفير كل المعلومات زي انا فين، موقفي ايه
مساله ال goal مش سهل، عشان كدة
ثم النوع الرابع

4 Utility Based Agents

دكمان عنده goal بس عنده
technique معين بيقيس مدى قربه
او بعده من ال goal

Utility Function

internally measures its performance among different states of the world.

وظيفتها اي اديها ال goals وتوصلها ال Action
المختلفة، هي تحدد اي انسب Action ممكن
نعملها

Subject : .... Input .... Date: .... output

function UtilityBased Agent (percept) returns Action

static: state, rules, action,
utility → the function that measures performance

state ← updateState
rules ← RuleMatch (state, rules)
actions ← RuleAction [rules]
action ← chooseAction (Utility, actions)

return action.

Goals

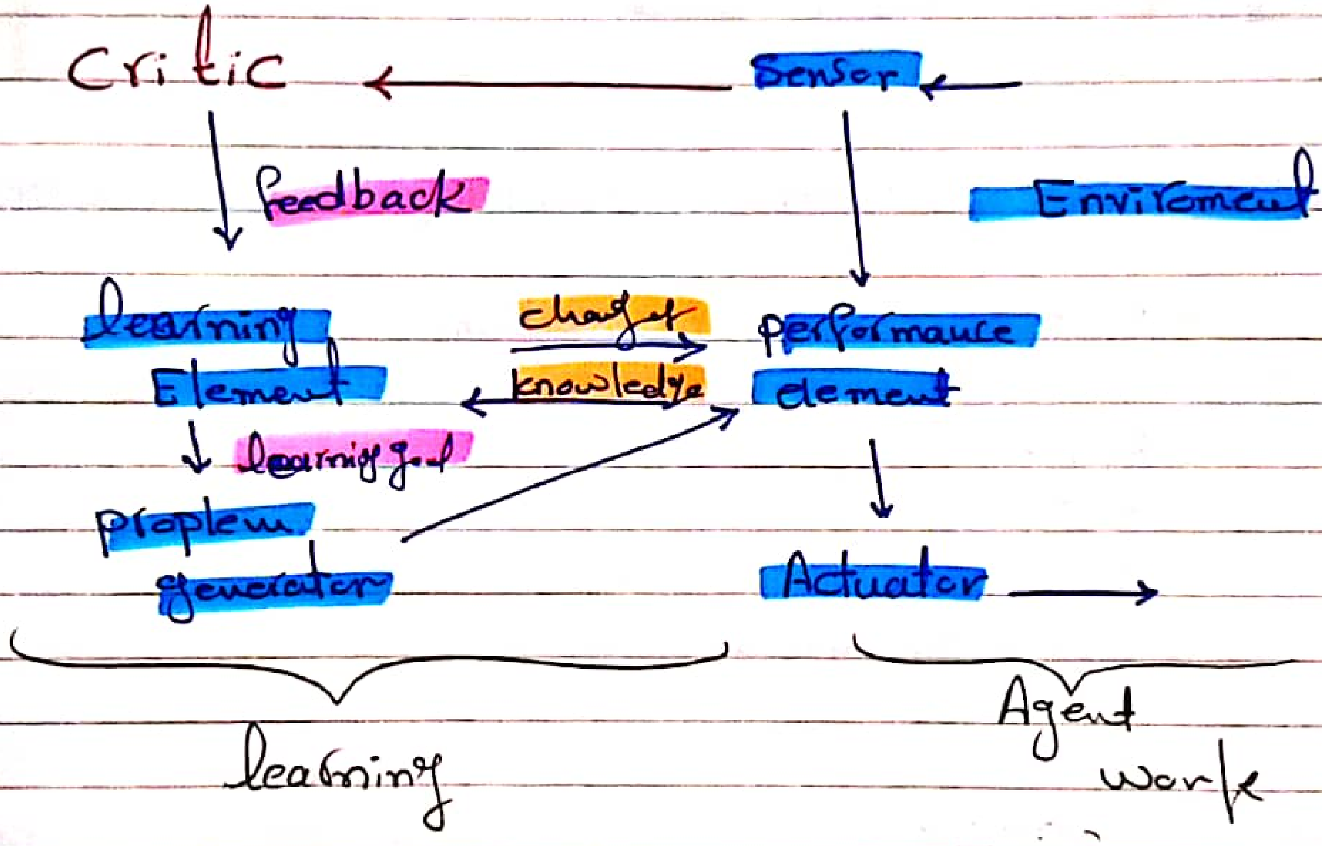
Conflicting goals	several goals
بيعارضوا بعض ، وساعتها لازم اختار واحد فيهم على التاني (trade off)	كتر ال goal وساعتها ال Utility هتحكم على اساسه ال Importance

- this agent chooses the action that maximizes expected Utility.

كل ما كانت الخبرة أكبر
كل ما كان التوقع احسن

learning Agent

Performance Element → All types of Agents



بفرض احكم على ال Action لحد ما يعيي المشاكل
يعني زي ال Actuator

Note →

جزء ال learning ده برة ال
scope بتاع الكورس
(مش علينا)

Subject: Chapter 3 Date: / /

Solving problems by Searching

كيفية الـ search انه عندك problem وبتدور على الـ solution المناسب ليه

Example:
- theorem proving
- Solving Puzzles
- playing games

Classic AI Searching problems

- Map Searching (navigation)
(اختيار أحسن طريقة من مكان واحد لـ مكان (مسافة / زمان / تكلفة))
- Rubik's Cube
- 8-Puzzle
- N-Queens (لعبة شطرنج بطريقة مختلفة)
- the River problem

Map Searching (navigation)

- Currently in Arad
- Flight leaves from Bucharest
عايز يوصل المطار في وقت معين
هيأخد أقصر طريقة
- Non-refundable ticket
- Goal: Bucharest in time
ممكن انه يكون هناك أكثر من وقت

البحث على الـ Map في البرمجيات

- formulate problem :
  - states : various cities
    مشكلات ان عندي مدن كتير وطرق كتير
  - Action: drive between cities
    اللي هحله عشان ازحل المشكلة اني هسير من مدينة للتانية

- find solution: (action sequence)
  - sequence of cities
    سلسلة المدن اللي هسير فيها

problem solving Agents

- Goal based Agent decides based on sequence of actions that will lead to desirable state.
  هيحقق المشكلة عن طريق خطوات توصل لل goal في الاخر

- Goals help to organize behaviour by limiting the objectives that the agent achieves