Today

See Russell and Norvig, chapter, 2 and 7

- Kinds of Agents
- Logical Agents
- Propositional Logic

References

Russell and Norvig

D. Dennett. **Kinds of Minds**. Weidenfeld and Nicolson, London, 1996.

Michael Wooldridge. **An introduction to Multi-Agent Systems**. Wiley, 2002

R. Fagin, J.Y. Halpern, Y. Moses, and M. Vardi. **Reasoning about Knowledge**. MIT Press, 1995.

The Springer series of volumes on intelligent agents: see `dis.cs.umass.edu/atal/books/`.

Properties of Agents

- **autonomy**: the agent can evolve on its own, without being directly controlled from outside.
- **social interaction**: agents usually interact with other agents, sometimes in cooperation, and sometimes in competition.
- **reaction**: a **reactive agent** is one that takes account of its environment, and responds to changes in the environment.
- **goal-directed**: the agent has its own goals, and takes initiatives in order to meet these goals.

The Intentional Stance

Daniel Dennett has proposed that:

> The intentional stance is the strategy of interpreting the behavior of an entity (person, animal, artifact, whatever) by treating it **as if** it were a rational agent who governed its "choice" of "action" by a "consideration" of its "beliefs" and "desires".
>
> Kinds of Minds, p 27

We get a stronger notion of agent if we follow this up, and design agents with extra properties.

Intelligent Agents

Here we attribute to agents the mental attitudes suggested by Dennett – beliefs, motivations, obligations, etc.

In addition, they may have:

- **mobility**: the agent is able to displace itself physically (eg around the Net).
- **rationality**: the agent will always act so as to work towards achieving its goals (with respect to its beliefs).
- **distribution**: various agents are physically separate (eg hosted by different processors).

Intentional Systems

Here "Intentional" means the property of mental attitudes like belief, desire etc. whereby they link up to things in the world about which we have beliefs, desires etc.

> **Intentional systems** are, by definition, all and only those entities whose behavior is predictable/explicable from the intentional stance.

Kinds of Minds, p 34

Examples thermostats, amoebas, bats, people, and chess-playing computers, . . . (Dennett)

Example

Take for vacuum cleaner:

- Percepts: location and contents, e.g., $[A, Dirty]$
- Actions: $Left$, $Right$, $Suck$, $NoOp$

What is the **right** way to organise the actions dependent on the percept history?

Rationality

Fixed performance measure evaluates the environment sequence
– one point per square cleaned up in time T?
– one point per clean square per time step, minus one per move?
– penalize for $> k$ dirty squares?
A rational agent chooses whichever action maximizes the expected value of the performance measure given the percept sequence to date

Rational $\neq$ omniscient
Rational $\neq$ clairvoyant
Rational $\neq$ successful

But: Rationality leads to exploration, learning, autonomy . . .

PEAS

To design a rational agent, we must specify the task environment

Consider, e.g., the task of designing an automated taxi:
what are: Performance measure, Environment, Actuators, Sensors?

Performance measure eg safety, destination, profits, legality, comfort,

Environment eg streets/motorways, traffic, pedestrians, weather, ...

Actuators eg steering, accelerator, brake, horn, speaker/display, ...

Sensors eg video, accelerometers, gauges, engine sensors, GPS, ...

Environment types

	Solitaire	Backgammon	e-shopping	Taxi
Observable	Yes	Yes	No	No
Deterministic	Yes	No	Partly	No
Episodic	No	No	No	No
Static	Yes	Semi	Semi	No
Discrete	Yes	Yes	Yes	No
Single-agent	Yes	No	Yes (except auctions)	No

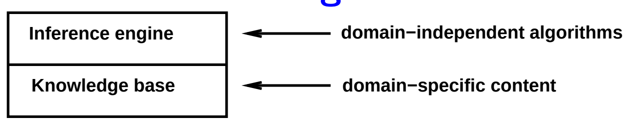
The environment type largely determines the agent design

The real world is (of course) partially observable, stochastic, sequential, dynamic, continuous, multi-agent

Agent types

Four basic types in order of increasing generality:
– simple reflex agents
– reflex agents with state
– goal-based agents
– utility-based agents
All these can be turned into learning agents

Knowledge bases



Knowledge base = set of sentences in a formal language

Declarative approach to building an agent (or other system):
TELL it what it needs to know

Then it can ASK itself what to do—answers should follow from the KB

Agents can be viewed at the knowledge level
i.e., what they know, regardless of how implemented

Or at the implementation level
i.e., data structures in KB and algorithms that manipulate them

A simple knowledge-based agent

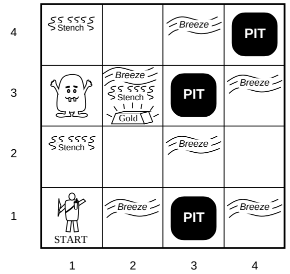
```
function KB-AGENT( percept) returns an action
   static: KB, a knowledge base
           t, a counter, initially 0, indicating time

   TELL(KB, MAKE-PERCEPT-SENTENCE( percept, t))
   action ← ASK(KB, MAKE-ACTION-QUERY(t))
   TELL(KB, MAKE-ACTION-SENTENCE(action, t))
   t ← t + 1
   return action
```

The agent must be able to:
- Represent states, actions, etc.; Incorporate new percepts
- Update internal representations of the world
- Deduce hidden properties of the world, and appropriate actions

Wumpus World PEAS description

Performance measure
gold +1000, death -1000
-1 per step, -10 for using the arrow

Environment
Squares adjacent to wumpus are smelly
Squares adjacent to pit are breezy
Glitter iff gold is in the same square
Shooting kills wumpus if you are facing it
Shooting uses up the only arrow
Grabbing picks up gold if in same square
Releasing drops the gold in same square



Sensors Breeze, Glitter, Smell

Actuators Left turn, Right turn, Forward, Grab, Release, Shoot

Wumpus world characterisation

Observable?? No—only local perception

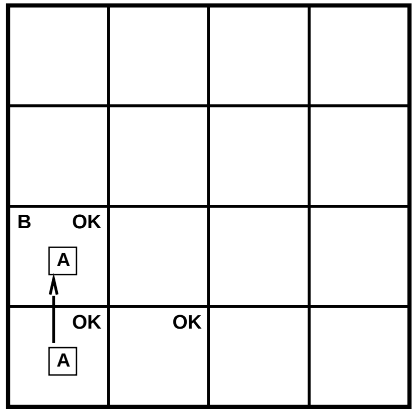
Deterministic?? Yes—outcomes exactly specified

Static?? Yes—Wumpus and Pits do not move

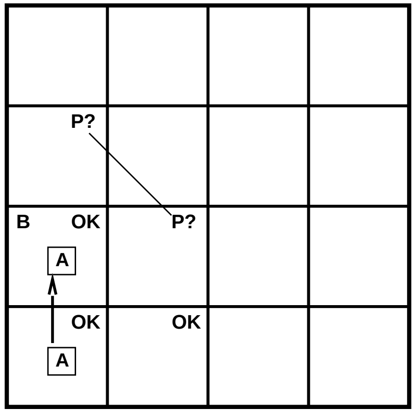
Discrete?? Yes

Single-agent?? Yes—Wumpus is essentially a natural feature

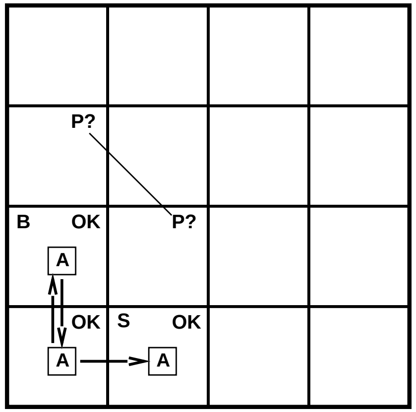
Exploring a wumpus world



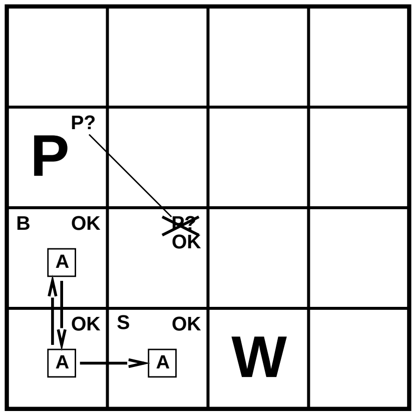
Exploring a wumpus world



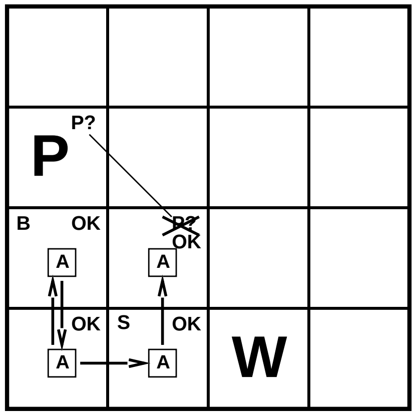
Exploring a wumpus world



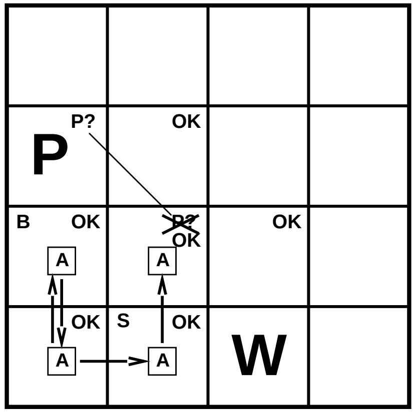
Exploring a wumpus world



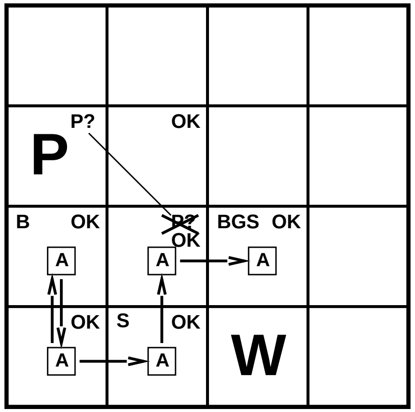
Exploring a wumpus world

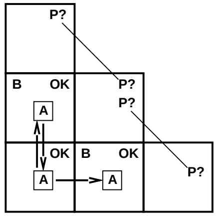


Exploring a wumpus world



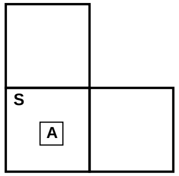
Exploring a wumpus world





Breeze in (1,2) and (2,1)
⇒ no safe actions

Assuming pits uniformly distributed,
(2,2) has pit w/ prob 0.86, vs. 0.31



Smell in (1,1)
⇒ cannot move
Can use a strategy of coercion:
shoot straight ahead
wumpus was there ⇒ dead ⇒ safe
wumpus wasn't there ⇒ safe

Logic in general

Logics are formal languages for representing information
such that conclusions can be drawn

Syntax defines the sentences in the language

Semantics define the "meaning" of sentences;
i.e., define truth of a sentence in a world

E.g., the language of arithmetic

$x+2 \geq y$ is a sentence; $x2+y>$ is not a sentence

$x+2 \geq y$ is true iff the number $x+2$ is not less than the number y

$x+2 \geq y$ is true in a world where $x=7,\ y=1$
$x+2 \geq y$ is false in a world where $x=0,\ y=6$

Entailment

Entailment means that one thing *follows from* another:

$$KB \models \alpha$$

Knowledge base KB entails sentence α if and only if
α is true in all worlds where KB is true

E.g., the KB containing "Schalke won" and "Hibs won" entails "Either Schalke won or Hibs won"

E.g., $x+y=4$ entails $4=x+y$

Entailment is a relationship between sentences (i.e., *syntax*) that is based on *semantics*

Note: brains process *syntax* (of some sort)

Models

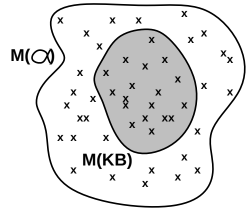
Logicians typically think in terms of models, which are formally structured worlds with respect to which truth can be evaluated

We say m is a model of a sentence α if α is true in m

$M(\alpha)$ is the set of all models of α

Then $KB \models \alpha$ if and only if every model of KB is also a model of α,

KB = Schalke won and Hibs won
α = Schalke won

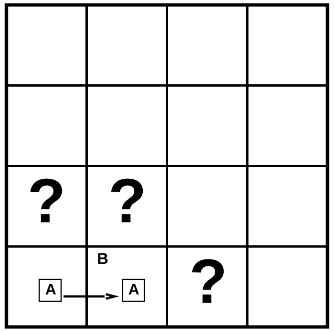


Entailment in the wumpus world

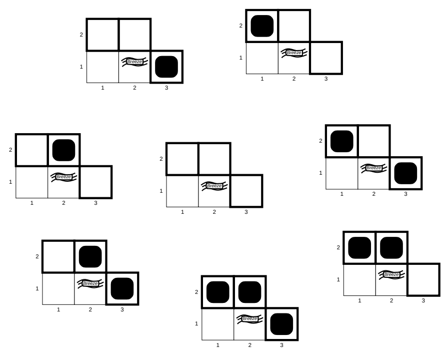
Situation after detecting nothing in [1,1],
moving right, breeze in [2,1]

Consider possible models for ?s assuming only pits

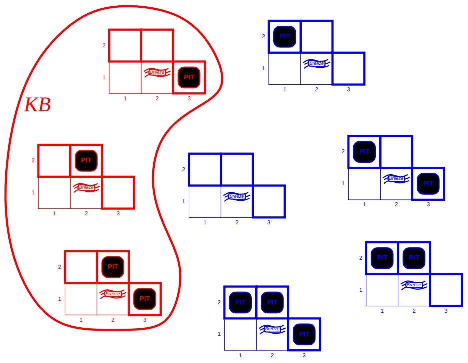
3 Boolean choices $\Rightarrow$ 8 possible models



Wumpus models

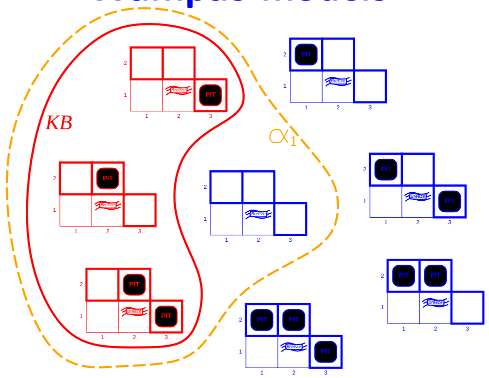


Wumpus models



KB = wumpus-world rules + observations

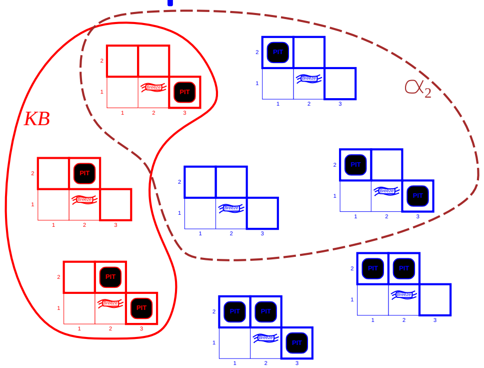
Wumpus models



KB = wumpus-world rules + observations

α_1 = "[1,2] is safe", $KB \models \alpha_1$, proved by model checking

Wumpus models



KB = wumpus-world rules + observations

α_2 = "[2,2] is safe", $KB \not\models \alpha_2$

Inference

$KB \vdash_i \alpha$ = sentence α can be derived from KB by procedure i

Consequences of KB are a haystack; α is a needle.
Entailment = needle in haystack; inference = finding it

Soundness: i is sound if
whenever $KB \vdash_i \alpha$, it is also true that $KB \models \alpha$

Completeness: i is complete if
whenever $KB \models \alpha$, it is also true that $KB \vdash_i \alpha$

Propositional logic: Syntax

Propositional logic is the simplest logic—illustrates basic ideas

The proposition symbols P_1, P_2 etc are sentences

If S is a sentence, $\neg S$ is a sentence (negation)

If S_1 and S_2 are sentences, $S_1 \wedge S_2$ is a sentence (conjunction)

If S_1 and S_2 are sentences, $S_1 \vee S_2$ is a sentence (disjunction)

If S_1 and S_2 are sentences, $S_1 \Rightarrow S_2$ is a sentence (implication)

If S_1 and S_2 are sentences, $S_1 \Leftrightarrow S_2$ is a sentence (biconditional)

Propositional logic: Semantics

Each model specifies true/false for each proposition symbol

E.g. $P_{1,2}$ $P_{2,2}$ $P_{3,1}$
$true$ $true$ $false$

Rules for evaluating truth with respect to a model m:

$\neg S$	is true iff	S	is false		
$S_1 \wedge S_2$	is true iff	S_1	is true and	S_2	is true
$S_1 \vee S_2$	is true iff	S_1	is true or	S_2	is true
$S_1 \Rightarrow S_2$	is true iff	S_1	is false or	S_2	is true
i.e.,	is false iff	S_1	is true and	S_2	is false
$S_1 \Leftrightarrow S_2$	is true iff	$S_1 \Rightarrow S_2$	is true and	$S_2 \Rightarrow S_1$	is true

Simple recursive process evaluates an arbitrary sentence.

Truth tables for connectives

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
$false$	$false$	$true$	$false$	$false$	$true$	$true$
$false$	$true$	$true$	$false$	$true$	$true$	$false$
$true$	$false$	$false$	$false$	$true$	$false$	$false$
$true$	$true$	$false$	$true$	$true$	$true$	$true$

Wumpus world sentences

Let $P_{i,j}$ be true if there is a pit in $[i,j]$.
Let $B_{i,j}$ be true if there is a breeze in $[i,j]$.

$$\neg P_{1,1}$$
$$\neg B_{1,1}$$
$$B_{2,1}$$

"Pits cause breezes in adjacent squares"

Wumpus world sentences

Let $P_{i,j}$ be true if there is a pit in $[i,j]$.
Let $B_{i,j}$ be true if there is a breeze in $[i,j]$.

$$\neg P_{1,1}$$
$$\neg B_{1,1}$$
$$B_{2,1}$$

"Pits cause breezes in adjacent squares"

$$B_{1,1} \quad \Leftrightarrow \quad (P_{1,2} \lor P_{2,1})$$
$$B_{2,1} \quad \Leftrightarrow \quad (P_{1,1} \lor P_{2,2} \lor P_{3,1})$$

"A square is breezy *if and only if* there is an adjacent pit"

Truth tables for inference

$B_{1,1}$	$B_{2,1}$	$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$P_{2,2}$	$P_{3,1}$	KB	α_1
f	f	f	f	f	f	f	f	t
f	f	f	f	f	f	t	f	t
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
f	t	f	f	f	f	f	f	t
f	t	f	f	f	f	t	$\underline{t}$	$\underline{t}$
f	t	f	f	f	t	f	$\underline{t}$	$\underline{t}$
f	t	f	f	f	t	t	$\underline{t}$	$\underline{t}$
f	t	f	f	t	f	f	f	t
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
t	t	t	t	t	t	t	f	f

Inference by enumeration

Depth-first enumeration of all models is sound and complete

function TT-ENTAILS?(KB, α) **returns** *true* or *false*
  symbols ← a list of the proposition symbols in KB and α
  return TT-CHECK-ALL(KB, α, *symbols*, [ ])

function TT-CHECK-ALL(KB, α, *symbols*, *model*) **returns** *true* or *false*
  if EMPTY?(*symbols*) **then**
    if PL-TRUE?(KB, *model*) **then return** PL-TRUE?(α, *model*)
    else return *true*
  else do
    P ← FIRST(*symbols*); *rest* ← REST(*symbols*)
    return TT-CHECK-ALL(KB, α, *rest*, EXTEND(P, *true*, *model*)) **and**
      TT-CHECK-ALL(KB, α, *rest*, EXTEND(P, *false*, *model*))

Summary

PEAS classification.

Logical agents apply inference to a knowledge base
to derive new information and make decisions

Basic concepts of logic:
- syntax: formal structure of sentences
- semantics: truth of sentences wrt models
- entailment: necessary truth of one sentence given another
- inference: deriving sentences from other sentences
- soundness: derivations produce only entailed sentences
- completeness: derivations can produce all entailed sentences

Wumpus world requires the ability to represent partial and negated information, reason by cases, etc.