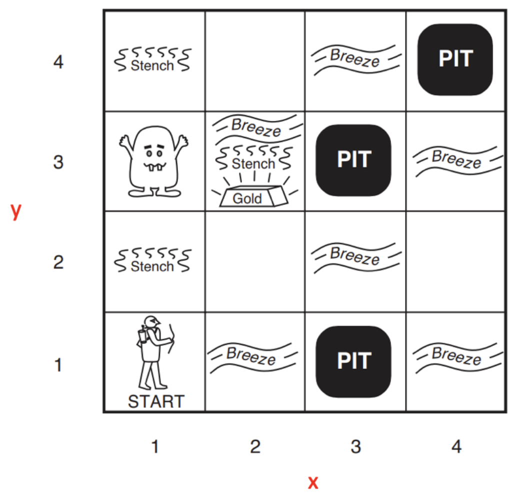
CAS CS 440
Lec 13

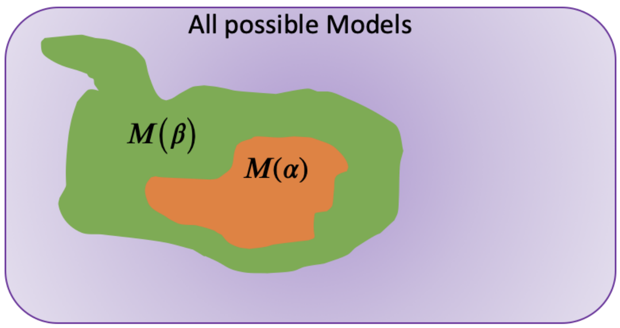
Logic II

1. Review
    a. Example World
        i. 

        ii. Performance Measure:
            1. +1000 when escaping w/ gold
            2. -1000 for dying (falling into pit / eaten)
            3. -1 for each action taken
            4. -10 using your only arrow
        iii. Environment:
            1. 4x4 grid
            2. Agent always starts in (1,1)
            3. Gold + Wumpus chosen randomly (uniformly) from squares not (1,1)
            4. Pit appears in each square with prob 0.2
        iv. Actions:
            1. Go forward 1 square
            2. Turn left/right by 90 degrees
        v. Sensing:
            1. Squares adjacent to Wumpus have stenches
            2. Squared adjacent to pits have breezes
            3. Square with gold in it is glittery
            4. Bump when agent walks into a wall
            5. Wumpus emits a scream when killed (heard anywhere)
        vi. Percept(t) = [Stench, Breeze, Glitter, Bump, Scream]

b. Reasoning
   i. Now we have a notion of “truth”
      1. Can execute the sentence on a hypothetical world (model) → and check whether a sentence is true or false
   ii. Want to relate two sentences to each other
   iii. If sentence a logically follows from sentence B:
      a |= B
      1. For example, if x = 0, then x * y = 0 → true
      2. x = 0 entails x * y = 0
   iv. Every model in which a is true, B is also true

$$\alpha \mid= \beta \; iff \; M(\alpha) \subseteq M(\beta)$$

   v. 

      1. In the example above, M(B) is x = 0 and M(a) is x * y = 0

c. Entailment & Inference
   i. Goal:
      1. Find sentences that are entailed by our KB
      2. Add them to our KB
      3. How?
         a. Algorithm from the previous slide is called model-checking
            i. Enumerate every possible model & check!
            ii. Calculate if $M(KB) \subseteq M(\alpha)$ by brute force
            iii. Brute force guarantees that it finds all possible sentences and also the correctness of those
   ii. Inference algorithm tries to derive sentences that are entailed by KB
      1. Lets differentiate entailment from derivation
         a. If inference algorithm A can derive a from KB:
            $KB \mid-_A \alpha$ now check the quality of the inference algorithm
   iii. Properties of Inference algorithms:
      1. Soundness
         a. Only derive entailed sentences (do not make things up)
      2. Completeness
         a. Can derive every entailed sentences

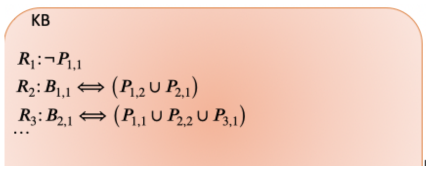
2. Sentence Structure: Prepositional Logic
    a. Two kinds of prepositional logic sentences:
        i. Atomic sentences:
            1. A single prepositional symbol
                a. True/False
                b. Variable
        ii. Complex sentences:
            1. Atomic sentences connected via operators (and paranthases):
                a. Not $\neg$
                b. And $\cap$
                c. Or $\cup$
                d. Implies $\Longrightarrow$
                e. Iff $\Longleftrightarrow$
3. Prepositional Logic
    a. Semantics
        i. Each model fixed truth value (True/False) for every variable
            1. Is there a wumpus at (1,1) → False
            2. Is there a wumpus at (2,2) → Can be True or False → put variable
        ii. Can evaluate truth value of sentences for a model
        iii.

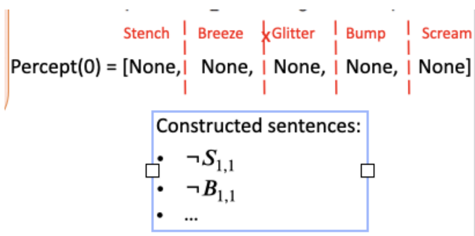
P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
false	*false*	*true*	*false*	*false*	*true*	*true*
false	*true*	*true*	*false*	*true*	*true*	*false*
true	*false*	*false*	*false*	*true*	*false*	*false*
true	*true*	*false*	*true*	*true*	*true*	*true*

4. Prepositional Sentences + Wumpus World
    a. Need lots of variables:
        i. Variables for percepts:
            1. $S_{1,1}, S_{1,2}, \ldots, S_{4,4}$ (is there a stench at (x,y))
            2. $B_{1,1}, B_{1,2}, \ldots, B_{4,4}$ (is there a breeze at (x,y))
            3. $Glitter_{1,1}, Glitter_{1,2}, \ldots, Glitter_{4,4}$ (is there a glitter at (x,y))
            4. … (continue)
        ii. Variables for Unknowns
            1. $W_{1,1}, W_{1,2}, \ldots, W_{4,4}$ (is there a wumpus at (x,y))
            2. $P_{1,1}, P_{1,2}, \ldots, P_{4,4}$ (is there a pit at (x,y))
            3. $Gold_{1,1}, Gold_{1,2}, \ldots, Gold_{4,4}$ (is there a gold at (x,y)

iii. Complex sentences:

1. Axioms:
   a. $B_{1,1} \Leftrightarrow (P_{1,2} \cup P_{2,1})$, …
      i. Breeze is at (1,1) if and only if there is a pit at (1,2) or (2,1)
   b. $S_{1,1} \Leftrightarrow (W_{1,2} \cup W_{2,1})$, …
      i. Stench is at (1,1) if and only if there is a wumpus at (1,2) or (2,1)
   c. … (can make a lot of these)

5. KB + Wumpus World
   a. KB starts off with axioms inside it (axioms are rules not derived, but it is assumed to be true)
   b. Also know $\neg P_{1,1}$ and $\neg W_{1,1}$! → starting point is safe
   c. 

   d. 

      i. Can make many sentences from perceptions → add them to KB

6. Inference
   a. The goal is to derive some new sentences 'a' to add to our KB
      i. Want to derive 'a's that are entailed from our KB!
   b. How?
      i. Model checking = truth table enumeration (we can make a big true table)
      ii. Tree search?
         1. Exponential for prepositional logic (co-NP-complete)
   c. Theorem proving!
      i. Math on logic sentences
      ii. Two sentences a, B are logically equivalent iff true in the same models:

$$\alpha \equiv \beta \Longleftrightarrow \alpha \mid = \beta \cap \beta \mid = \alpha$$

         1. Two sentences are equivalent if they entail each other
      iii. A sentence is valid iff it is true in all models (i.e. tautology)
      iv. A sentence 'a' is satisfiable iff $\exists\, m \in |M(\alpha)| > 0$ (at least one)

7. Theorem Proving
    a. For any a, B:

    $$\alpha \mid = \beta \Longleftrightarrow (a \Longrightarrow \beta) \text{ is valid}$$

        i. Does 'a' imply B?
        ii. We can check if $\alpha \mid = \beta$ by checking that $(a \Longrightarrow \beta)$ is true in every model!
    b. For any a, B:

    $$\alpha \mid = \beta \Longleftrightarrow (a \cap \neg\beta) \text{ is unsatisfiable}$$

        i. We can check if $\alpha \mid = \beta$ by checking that there is no model that satisfies $(a \cap \neg\beta)$
        ii. Proof by contradiction
8. Inference Rules
    a. Modus Ponens:

    Know this to be true   Also know this to be true

    $$\frac{\alpha \Rightarrow \beta, \quad \alpha}{\beta} .$$

    Infer this!

        i. If 'a' implies B and 'a' is true, then B can be implied (B is true)
    b. And-Elimination:

    $$\frac{\alpha \wedge \beta}{\alpha} .$$

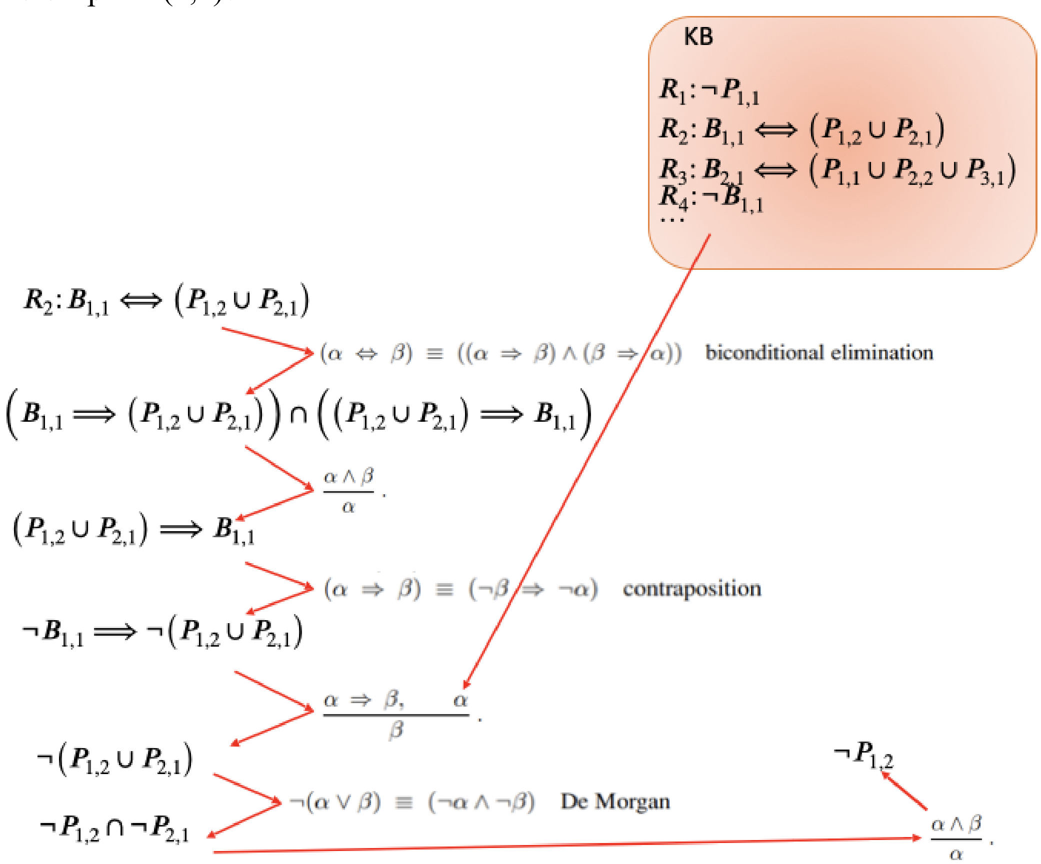
        i. If we know two sentences, one of them is always true
        ii. If 'a' and B is true, then 'a' is true
    c. These two rules are Sound!
        i. If I only know these two rules, I can check entail sentences
        ii. Avoid enumerating models!
            1. Just simply use the mathematical rules
            2. Apply rules (no need to search through all the models)

Logical equivalence

$(\alpha \wedge \beta)$	$\equiv$	$(\beta \wedge \alpha)$	commutativity of $\wedge$
$(\alpha \vee \beta)$	$\equiv$	$(\beta \vee \alpha)$	commutativity of $\vee$
$((\alpha \wedge \beta) \wedge \gamma)$	$\equiv$	$(\alpha \wedge (\beta \wedge \gamma))$	associativity of $\wedge$
$((\alpha \vee \beta) \vee \gamma)$	$\equiv$	$(\alpha \vee (\beta \vee \gamma))$	associativity of $\vee$
$\neg(\neg\alpha)$	$\equiv$	α	double-negation elimination
$(\alpha \Rightarrow \beta)$	$\equiv$	$(\neg\beta \Rightarrow \neg\alpha)$	contraposition
$(\alpha \Rightarrow \beta)$	$\equiv$	$(\neg\alpha \vee \beta)$	implication elimination
$(\alpha \Leftrightarrow \beta)$	$\equiv$	$((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha))$	biconditional elimination
$\neg(\alpha \wedge \beta)$	$\equiv$	$(\neg\alpha \vee \neg\beta)$	De Morgan
$\neg(\alpha \vee \beta)$	$\equiv$	$(\neg\alpha \wedge \neg\beta)$	De Morgan
$(\alpha \wedge (\beta \vee \gamma))$	$\equiv$	$((\alpha \wedge \beta) \vee (\alpha \wedge \gamma))$	distributivity of $\wedge$ over $\vee$
$(\alpha \vee (\beta \wedge \gamma))$	$\equiv$	$((\alpha \vee \beta) \wedge (\alpha \vee \gamma))$	distributivity of $\vee$ over $\wedge$

d.

9. Using Inference Rules
    a. Is there a pit in (1,2)?



b.

c. Round 1: try to derive every sentences I can and continue

d. If you cannot come up with any more sentences, you stop

10. Inference Algorithms
    a. Use any search algorithm
    b. Proof problem:
        i. Initial state = initial KB
        ii. Actions = all inference rules applied to all sentences that match the top half of inference rule
        iii. Result of applying an action = add the sentence in the bottom half of the inference rule to the KB
        iv. Goal: state that contains the sentence we are trying to prove! (stop search)
    c. Searching for proof = enumerating all possible models!
        i. More efficient
            1. Ignoring irrelevant variables
            2. Skipping over the things that are false