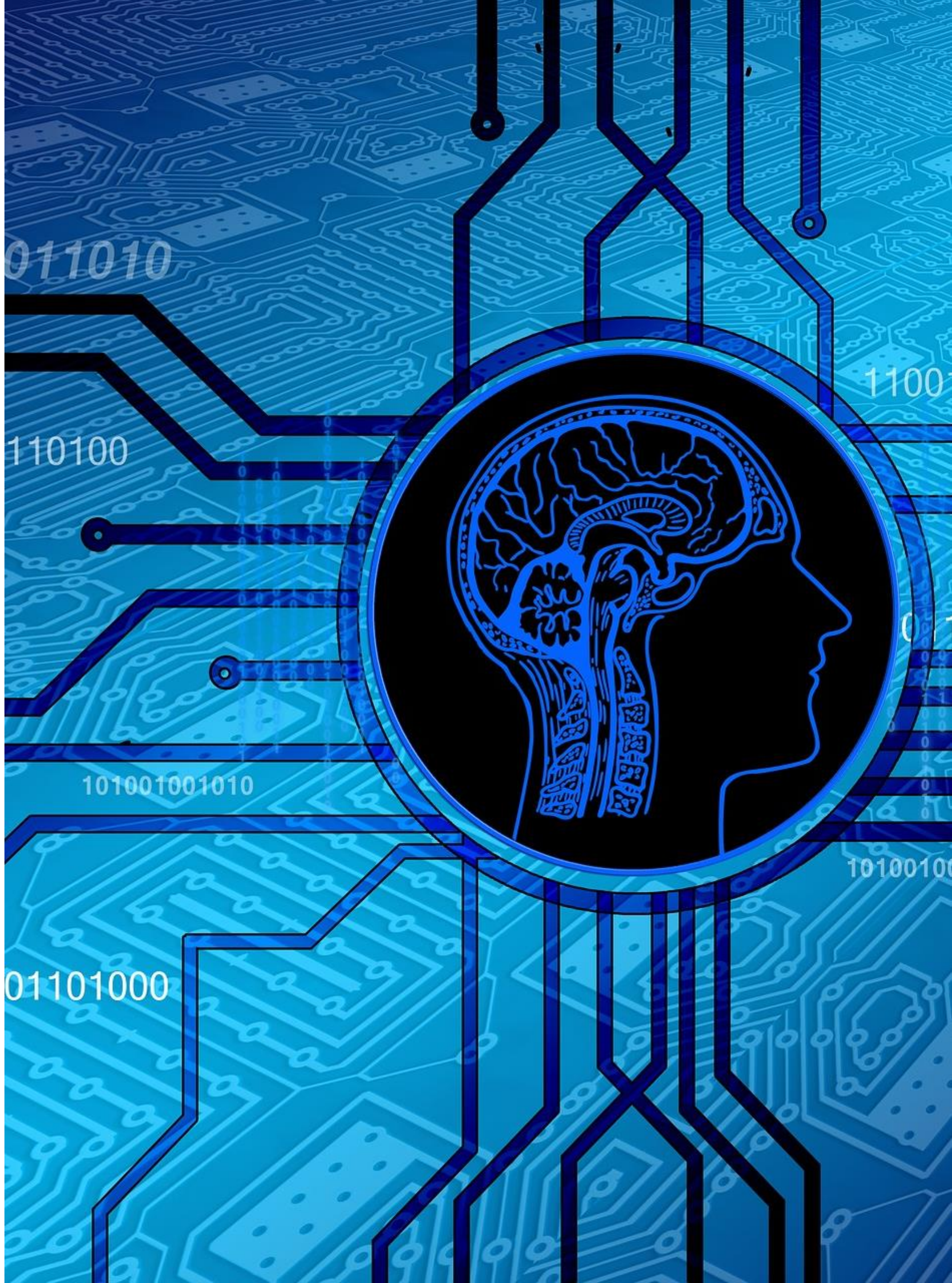
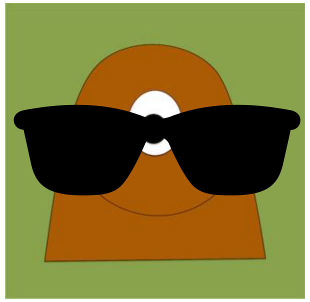
Revision (Logic)

Informatics 2D: Reasoning and Agents



Logical Agents: KBs



- 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 with respect to 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
- Propositional logic solves many problems but lacks expressive power.

Logical equivalence

Two sentences are logically equivalent iff true in the same models:

$$\alpha \equiv \beta \text{ iff } \alpha \models \beta \text{ and } \beta \models \alpha$$

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

Validity and Satisfiability

A sentence is **valid** if it is true in *all models*

- *true*, $A \lor \neg A$, $A \Rightarrow A$, $(A \land (A \Rightarrow B)) \Rightarrow B$

Validity is connected to inference via the **Deduction Theorem**

- $KB \vDash \alpha$ if and only if $(KB \Rightarrow \alpha)$ is valid

A sentence is **satisfiable** if it is true in *some model*

- e.g., $A \lor B$, C

A sentence is **unsatisfiable** if it is true in *no models*

- e.g., $A \land \neg A$

Satisfiability is connected to inference via the following:

- $KB \vDash \alpha$ if and only if $(KB \land \neg\alpha)$ is unsatisfiable
- prove α by *reductio ad absurdum*

Effective Propositional Inference

- Two algorithms: DPLL & WalkSAT
  - DPLL is a decision procedure, i.e., it will return true (yes) or false (no) for a set of propositional clauses (cf. complete algorithm)
  - They work with Conjunctive Normal Form (CNF)

The DPLL algorithm

Determine if an input propositional logic sentence (in CNF) is satisfiable.

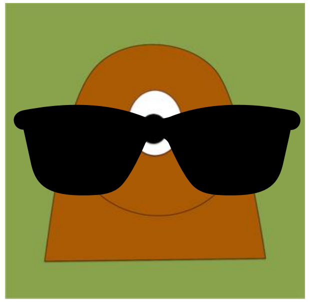
Improvements over truth table enumeration:

1. Early termination
2. Pure symbol heuristic
3. Unit clause heuristic

The WalkSAT algorithm

- Incomplete, local search algorithm
- Evaluation function:
  - The min-conflict heuristic of minimizing the number of unsatisfied clauses
- Algorithm checks for satisfiability by randomly flipping the values of variables
- Balance between greediness and randomness

First-Order Logic



- objects and relations are semantic primitives
- syntax: constants, functions, predicates, equality, quantifiers.
- Predicates (applied to terms) have a truth value (i.e., true or false)
  - e.g., < (less than) is a predicate so, $x < 3 + 5$ is either true or false
- Functions just construct new terms out of other terms
- Increased expressive power: sufficient to define Wumpus world.
  - Quantifiers, Equality …

Unification and Generalised Modus Ponens

- Rules for quantifiers — UI, EI etc.
- Reducing FOL to PL
- Unification
- Generalised modus ponens (GMP)

Most General Unifier (MGU)

Unifying *Knows(John, x)* and *Knows(y, z)*

$$\theta = \{y/John, x/z\} \quad \text{or} \quad \theta = \{y/John, x/John, z/John\}$$

The first unifier is more general than the second.

FOL: There is a **single** most general unifier (MGU) that is unique up to renaming of variables.

$$\text{MGU} = \{y/John, x/z\}$$

Can be viewed as an equation solving problem.

- i.e. solve $Knows(John, x) \stackrel{?}{=} Knows(y, z)$

MGU Exercises

- P(x,A) =?= P(f(y),y)
- P(x,g(x)) =?= P(f(y),y)

- Decomposition
- Conflict
- Eliminate
- Delete
- Switch
- Coalesce
- Occurs Check

Inference

- Forward chaining
  - Informally: Unify all the assumptions in an implication rule with facts in KB to discharge them. If successful, add instantiated conclusion to KB.
    This is like a discovery process.
- Backward chaining
  - Informally: Unify conclusion of an implication rule with some fact in KB. If successful, add instantiated assumptions to KB as new goals.
    This is like a decomposition into sub-problems.

Resolution

- Formal statements for various types of resolution.
- Refutation method works by looking for a contradiction i.e., tries to derive falsity (empty clause).
- Need to negate the goal before converting it to clausal form.

Example

9. Which of the following clauses is the result of resolving clause $P(G(y)) \vee \neg Q(G(y))$ with clause $\neg P(F(x)) \vee Q(x)$, assuming they can be resolved?

   (a) $P(G(y)) \vee \neg P(F(x))$.

   (b) $Q(y) \vee \neg Q(F(y))$.

   (c) Resolution fails.

   (d) $P(G(y)) \vee P(F(G(y))) \vee \neg Q(G(y))$.

   (e) $P(G(y)) \vee \neg P(F(G(y)))$.

Example

9. Which of the following clauses is the result of resolving clause $P(G(y)) \vee \neg Q(G(y))$ with clause $\neg P(F(x)) \vee Q(x)$, assuming they can be resolved?

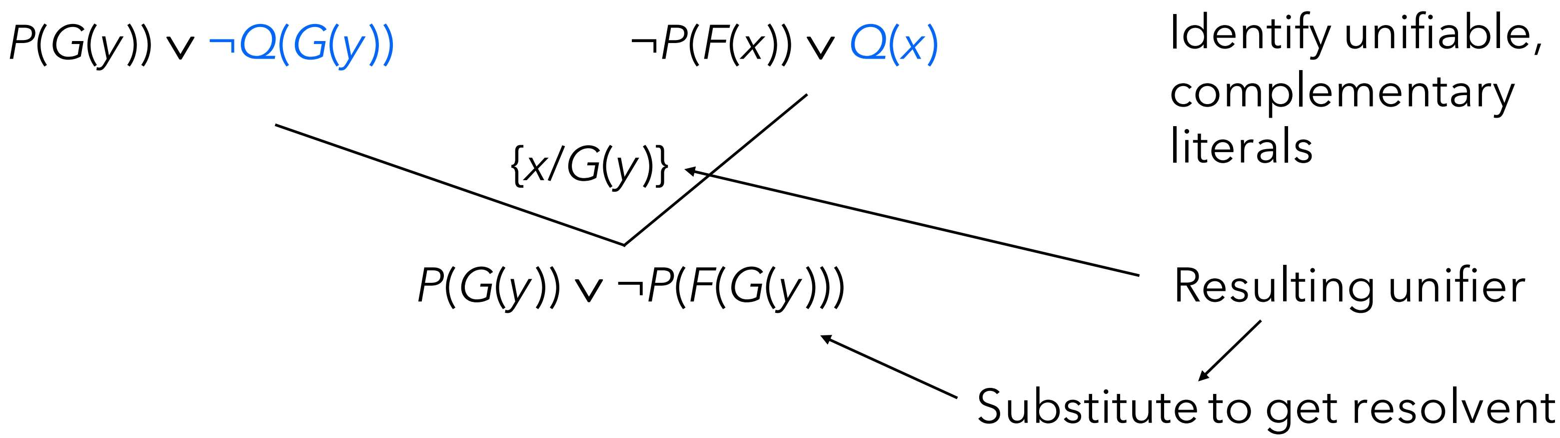
(a) $P(G(y)) \vee \neg P(F(x))$.

(b) $Q(y) \vee \neg Q(F(y))$.

(c) Resolution fails.

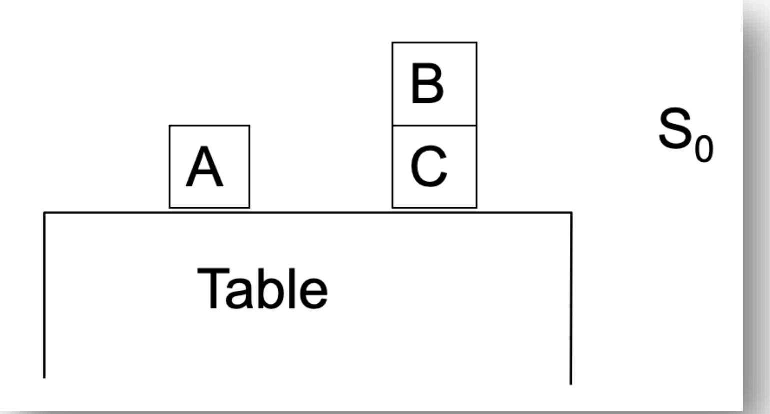
(d) $P(G(y)) \vee P(F(G(y))) \vee \neg Q(G(y))$.

(e) $P(G(y)) \vee \neg P(F(G(y)))$.



Situation Calculus

- We axiomatize all our actions, then use a general theorem prover to prove that a situation exists in which our goal is true.
  - Effect Axioms
  - Frame Axioms
- The actions in the proof would comprise our plan.



Goal: $On(A,B,s') \wedge On(B,C,s')$

- **Effect Axiom:**

  $\forall x,y,s.$

  $Clear(x,s) \wedge Clear(y,s) \Rightarrow On(x,y,Result(Move(x,y),s))$

- **Frame Axiom:**

  $\forall x,y,s,a.$

  $On(x,y,s) \wedge (\forall z.\ a=Move(x,z) \Rightarrow y=z)$

  $\Rightarrow On(x,y,Result(a,s))$

- **Other axioms:**

  Unique action axioms, unique name axioms

Initial State

$On(A,Table,S0)$

$On(B,C,S0)$

$On(C,Table,S0)$

$Clear(A,S0)$

$Clear(B,S0)$

Constants: A, B, C, S0
Variables: a, x, y, s

Convert this KB into CNF Form

- Eliminate implications
- Move negations inwards
- Standardize variables apart
- Skolemize
- Drop Universal Quantifiers
- Create clauses

Goal: On(A,B,s') ∧ On(B,C,s')

- **Effect Axiom:**
  ∀x,y,s.
  Clear(x,s) ∧ Clear(y,s) ⇒ On(x,y,Result(Move(x,y),s))
- **Frame Axiom:**
  ∀x,y,s,a.
  On(x,y,s) ∧ (∀z. a=Move(x,z) ⇒ y=z)
  ⇒ On(x,y,Result(a,s))
- **Other axioms:**
  Unique action axioms, unique name axioms

Initial State

On(A,Table,S0)
On(B,C,S0)
On(C,Table,S0)
Clear(A,S0)
Clear(B,S0)

Constants: A, B, C, S0
Variables: a, x, y, s

Convert this KB into CNF Form

KB and Axioms as Clauses

Constants: A, B, C, S0
Variables: a, x, y, s

Initial State

$On(A,Table,S0)$

$On(B,C,S0)$

$On(C,Table,S0)$

$Clear(A,S0)$

$Clear(B,S0)$

Goal

$\neg On(A,B,s') \vee \neg On(B,C,s')$

Effect Axiom

$\neg Clear(x,s) \vee \neg Clear(y,s) \vee On(x,y, Result(Move(x,y),s))$

Frame Axioms

Skolem function

$\neg On(x,y,s) \vee a=Move(x,Z(x,y,s,a)) \vee On(x,y,Result(a,s))$

$\neg On(x,y,s) \vee \neg y=Z(x,y,s,a) \vee On(x,y,Result(a,s))$

Unique Action Axioms: $\neg\, Move(A,B)=Move(B,z)$, etc

Unique Name Axiom: disequality for every pair of constants in KB

Edinburgh University **Students' Association**

Nominations for the 2024 Teaching Awards are now open! Submit your nomination here.

https://www.eusa.ed.ac.uk/whatson/awards/teachingawards

8 years ago today... #AAAI-16

Thank you!