Propositional Logic: Semantics

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*

- **Model (i.e. possible world):**
  - Assignment of truth values to symbols
  - Example: m={P=True , Q=False}
    - Note: Often called "assignment" instead of "model", and "model" is used for an assignment that evaluates to true.
- **Validity:**
  - A sentence α is valid, if it is true in every model.
- **Satisfiability:**
  - A sentence α is satisfiable, if it is true in at least one model.
- **Entailment:**
  - $\alpha \vDash \beta$ if and only if, in every model in which α is true, β is also true.

Model Checking

- **Idea:**
  - To test whether $\alpha \vDash \beta$, enumerate all models and check truth of α and β.
  - α entails β if no model exists in which α is true and β is false (i.e. $(\alpha \wedge \neg\beta)$ is unsatisfiable)
- **Proof by Contradiction:**
  $\alpha \vDash \beta$ if and only if the sentence $(\alpha \wedge \neg\beta)$ is unsatisfiable.
- **Model Checking:**
  - Variables: One for each propositional symbol
  - Domains: {true, false}
  - Objective Function: $(\alpha \wedge \neg\beta)$
  - Which search algorithm works best?

Propositional Logic: Some Inference Rules

Modus Ponens:

Know:	$\alpha \Rightarrow \beta$	If raining, then soggy courts.
and	α	It is raining.
Then:	β	Soggy Courts.

Modus Tollens:

Know:	$\alpha \Rightarrow \beta$	If raining, then soggy courts.
And	$\neg \beta$	No soggy courts.
Then:	$\neg \alpha$	It is not raining.

And-Elimination:

Know:	$\alpha \wedge \beta$	It is raining and soggy courts.
Then:	α	It is raining.

Example: Forward Chaining

Knowledge-base describing when the car should brake?

(PersonInFrontOfCar ⇒ Brake)
∧ (((YellowLight ∧ Policeman) ∧ (¬Slippery)) ⇒ Brake)
∧ (Policecar ⇒ Policeman)
∧ (Snow ⇒ Slippery)
∧ (Slippery ⇒ ¬Dry)
∧ (RedLight ⇒ Brake)
∧ (Winter ⇒ Snow)

Observation from sensors:

YellowLight ∧ ¬RedLight ∧ ¬Snow ∧ Dry ∧ Policecar ∧ ¬PersonInFrontOfCar

What can we infer?

- And-elimination: Policecar
- Modus Ponens: Policeman
- And-elimination: Dry
- Modus Tollens: ¬Slippery
- And-elimination: YellowLight ∧ Policeman ∧ ¬Slippery
- Modus Ponens: Brake
- And-elimination: ¬Snow
- Modus Tollens: ¬Winter

Inference Strategy: Forward Chaining

Idea:

- Infer everything (?) that can be inferred.
- Notation: In implication $\alpha \Rightarrow \beta$, α (or its compontents) are called premises, β is called consequent/conclusion.

Forward Chaining:

Given a fact *p* to be added to the KB,

1. Find all implications *I* that have *p* as a premise
2. For each *i* in *I*, if the other premises in *i* are already known to hold
   a) Add the consequent in *i* to the KB

Continue until no more facts can be inferred.