Propositional logic

Readings:

• CHAPTER 1 [Rosen]

ADDITIONAL BIBLIOGRAPHY:

• Chapters 13,14 Nilsson
• Chapter 7 Russell + Norvig
• A. Deaño "Introducción a la Lógica Formal",
• E. Paniagua Arís, J. L. Sánchez González, F. Martín Rubio, "Lógica computacional", Thomson
• Melvin Fitting "First-order logic and automated theorem proving", Springer-Verlag (New-York 1990)
• Gödel, Escher, Bach: An Eternal Golden Braid Douglas R. Hofstadter

Goal: To automate reasoning

Example: Robot (knowledge-based agent) endowed with a mechanical arm trying to lift an object.

The robot has **general knowledge**:

» Rule: "If the battery is loaded and the object is liftable (its weight is not too large, it is not screwed or anchored to the floor, etc.) then the object moves when the mechanical arm is activated"

The robot has mechanisms (for instance, sensors, internal states, etc.) by means of which the following facts can **be directly checked**

» Whether the battery is loaded.

» Whether the mechanical arm has been activated.

» Whether, after having activated the mechanical arm the object has actually moved.

Specific situation (situation 1):

The robot determines that the battery is loaded, but that, after having activated the mechanical arm the object has not moved.

The robot arrives to the conclusion:

"the object is not liftable"

Robot in situation 1

About what are we speaking?

A: "The mechanical arm has been activated"

B: "The battery is loaded"

D: "The object moves"

P: "The object is liftable"

Atomic propositions (their truth value – either "True" or "False") can only be determined by directly checking in the specific situation in the real world.

What does the robot know?

- "**If** the **battery is loaded and the object is liftable then**, **the object moves after the mechanical arm has been activated**".

  w_1 : $(A \wedge B \wedge P) \Rightarrow D$ ["If A and B and P, then D"]

- "**The mechanical arm has been activated**"

  w_2 : A ["A"]

- **"The battery is loaded"**

  w_3 : B ["B"]

- **"The object has not moved"**

  w_4 : $\neg D$ ["Not D"]

What does the robot conclude? w: $\neg P$ ["Not P"]

Robot in situation 2

Another specific situation (situation 2):

The battery sensor of robot in the previous example is now out of order and it is not possible to determine whether the battery is loaded or not. The movement sensor indicates that, after having activated the mechanical arm, the object has not moved.

The robot reaches the conclusion:

"Either the battery is empty or the object is not movable (or both)"

What does the robot know?

- "**If the battery is loaded and the object is liftable, then, the object moves after the mechanical arm has been activated**".

  $w_1 : (A \wedge B \wedge P) \Rightarrow D$ ["If A and B and P, then D"]

- **"The mechanical arm has been activated"**

  $w_2 : A$ ["A"]

- **"The object has not moved"**

  $w_3 : \neg D$ ["Not D"]

What does the robot conclude?

$w : \neg B \vee \neg P$ ["Not B or not P"]

A knowledge-based agent

A knowledge-based agent ought to be endowed with

» **A knowledge base:** Collection of well-formed formulas (WFFs) that represent true statements about the actual situation of the agent in the real world.

» **A mechanism to determine the truth value of propositions:**
  - Innate knowledge (e.g. rules, restrictions)
  - Sensors that allow to directly check the truth value of a proposition.

» **A lookup mechanism** giving access to the knowledge stored in the knowledge base.

» **A correct inference mechanism** that makes it possible to derive new WFF's from the ones that the agent has collected and to incorporate them into the knowledge base.

If the WFFs in the knowledge base have truth value "*True (T)*" in the actual situation of the agent in the real world, the WFFs derived from them by means of correct inference rules also have truth value "*True*" in that situation.

» **A complete inference mechanism**

It is desirable, but not strictly necessary

[Gödel: It is not always possible]

The game of logic

Deduction (Inference)

Knowledge base (Collection of WFF's) ⟶ Extended knowledge base

Representation ↑ [World of logic]

[Real world] Semantics ↓ Semantics

Propositions about the real world — Propositions about the real world

Forget meaning!

– Consider logic as a game with well-defined rules.

– To play the game you need to know and correctly apply these rules.

PLAYERS BEWARE: When we humans try to carry out a reasoning in formal logic, it is easy (and often misleading) to cheat and resort to informal reasoning, which is based on meaning.

Propositions

- **Proposition (or sentence):** Declarative sentence about some aspect of the world, which has a definite truth value (it is either true or false).

  » **Atomic proposition:** Proposition whose truth value can only be determined by directly checking in the real world.

  "The battery is loaded"

  » **Compound proposition (not atomic):**
  - Proposition composed of atomic propositions articulated by means of logical connectives.
  - Its truth value can be determined by the truth values of the atomic propositions of which it is composed and the truth tables of the logical connectives.

  "If the battery is loaded and the object is liftable, then, the object moves after the mechanical arm has been activated".

THESE ARE NOT PROPOSITIONS

"Is the battery loaded?" [not declarative]

"Activate the mechanical arm" [not declarative]

"This sentence is false" [no definite truth value]

Knowledge base

- **Knowledge base:** Collection of propositions that describe a specific situation in the real world.
  The agent collects in her knowledge base WFFs corresponding to propositions whose truth value is “True” in her actual situation in the real world.
  » **Robot’s knowledge base in situation 1:**
  Proposition 1: “If the battery is loaded and the object is liftable then, the object moves after the mechanical arm has been activated“
  Proposition 2: “The mechanical arm has been activated”
  Proposition 3: “The battery is loaded”
  Proposition 4: “The object moves”
  » **Robot’s knowledge base in situation 2: (battery sensor out of order):**
  Proposition 1: “If the battery is loaded and the object is liftable then, the object moves after the mechanical arm has been activated“.
  Proposition 2: “The mechanical arm has been activated”
  Proposition 3: “The object has not moved”

Symbolic representation: atoms

- **Atoms**
  » **Literal atoms:** T (True) , F (False)
  » **Symbolic atoms**
  Symbols that represent atomic propositions.
  The **denotation** of a symbolic atom is the declarative in natural language to which the atom makes reference

Atoms needed to describe the robot’s world:

Symbolic Atom	Denotation
A	“The mechanical arm has been activated”
B	“The battery is loaded”
D	“The object moves”
P	“The object is liftable”

Symbolic representation: logical connectives

- **Logical connectives:**
  » Logical connectives are used to articulate atomic propositions so as to form compound propositions.
  » The truth value of a compound proposition formed by atomic propositions articulated by means of a logical connective is defined by a **truth table** that is specific to that logical connective.
  » Types of logical connectives
  – Unary: $\neg$ (“not”)
  $\neg D$ [“the object has not moved”]
  – n-ary: $\wedge$ (“and”), $\vee$ (“or”)
  $A \wedge B \wedge P$
  [“The mechanical arm has been activated, the battery is loaded, and the object is liftable”]
  – binary: $\Rightarrow$ (“implies”), $\Leftrightarrow$ (“if and only if”)
  $(A \wedge B \wedge P) \Rightarrow D$
  [“ If the mechanical arm has been activated, the battery is loaded, and the object is liftable then the object moves”]

The elements of logic

- **A formal language**: Symbols + syntactic rules that allow the combination of the symbols in sentences that are grammatically correct (WFF: well-formed formula).
  $(A \wedge B \wedge P) \Rightarrow D$
  is a WFF; i.e. a grammatically correct sentence in propositional logic.
- **Semantics**: Association between WFFs in the formal language and sentences in natural language about the domain of which we are speaking.
- **Inference rules:** Typographical rules (i.e. they involve only symbol manipulations; therefore, they do not rely on meaning) used to generate new WFF’s from a given set of WFFs
  E.g. $\{R,\ R \Rightarrow S\} \vdash_{MODUS\ PONENS} S$

Languages

- **Elements in language**
  - » Syntax (form)
  - » Semantics (meaning)
- **Types of language**
  - » Natural language
    - – Standard means of communication among humans.
    - – It is difficult to specify a systematic syntax for it.

    E.g. Spanish, Urdu, English
  - » Formal language
    - – Syntax consists of well defined rules

      E.g. Programming languages
      (LISP, ADA, PROLOG, Java, HTML, XML)
    - – Propositional logic, predicate logic
- **Grammar of a language**

  Set of rules that allow to
  (i) Determine whether a sentence is correct
  (ii) Generate syntactically correct sentences within that language.

Grammars

- A subset of English can be generated using the following grammar in **Backus-Naur form**

  - » <Sentence> → <Nominal syntagm> <Verbal syntagm>
  - » <Nominal s.> → <article> <adjective> <name> | <article> <name>
  - » <Verbal s.> → <verb> <adverb> | <verb>
  - » <article> → a | the
  - » <adjective> → big | hungry
  - » <name> → rabbit | mathematician
  - » <verb> → eats | jumps | reasons
  - » <adverb> → happily | logically

GRAMMATICALLY CORRECT SENTENCES
“The big rabbit jumps”
“The rabbit eats happily”
“A hungry mathematician jumps logically”

The language of propositional logic

- **Atoms**: `T , F`
  `P,Q,P1,P2,P3,...,Pn,...`
- **Parenthesis:** (,)
- **Logical connectives**

  $\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$ (in order of precedence)
- **Well-formed formulas (WFFs** or **sentences)**
  - » **An atom is a WFF.**
  - » If $\mathbf{w}_1$ and $\mathbf{w}_2$ are WFFs, the following expressions are also WFFs

    $\neg \mathbf{w}_1$ **[negation of w_1]**
    $(\mathbf{w}_1 \wedge \mathbf{w}_2)$ **[conjunction between w_1 and w_2]**
    $(\mathbf{w}_1 \vee \mathbf{w}_2)$ **[disjunction between w_1 and w_2]**
    $(\mathbf{w}_1 \Rightarrow \mathbf{w}_2)$ **[implication, conditional, if-then rule**
    w_1 is the premise or antecedent,
    w_2 is the conclusion or consequent**]**
    $(\mathbf{w}_1 \Leftrightarrow \mathbf{w}_2)$ **[biconditional, if and only if]**

Grammar of propositional logic

In Backus-Naur form

- » < WFF> → < Atomic WFF> | <Compound WFF>
- » < Atomic WFF> → `V|F|` <Symbol>
- » <Symbol> → `P|Q|P1|P2|P3|...|Pn,...`
- » <Compound WFF> → $\neg$ `<WFF>`
  | `(<WFF>` $\wedge$ `<WFF>)`
  | `(<WFF>` $\vee$ `<WFF>)`
  | `(<WFF>` $\Rightarrow$ `<WFF>)`
  | `(<WFF>` $\Leftrightarrow$ `<WFF>)`

Parentheses

- The formal language that we have defined is very strict with parentheses.
- In some expressions, we will make use of the precedence and associative rules and will not write them down explicitly.
- We will use parentheses only to specify a particular order of evaluation or for the sake of clarity

Examples: $((P \wedge Q) \Rightarrow \neg P)$, or $P \wedge Q \Rightarrow \neg P$
$(P \Rightarrow \neg P)$, or $P \Rightarrow \neg P$
$((P \vee P) \Rightarrow \neg P)$, or $P \vee P \Rightarrow \neg P$

Truth table: logical connective "not"

$\neg w_1$: **"Not w_1"**

$\neg w_1$ has the truth value ***True*** if w_1 has the value ***False***.
$\neg w_1$ has the truth value ***False*** if w_1 has the value ***True***.

w_1	$\neg w_1$
True	*False*
False	*True*

Example:
D "The object moves"
$\neg$D "The object **does not** move"

Truth tables: logical connective "and"

$w_1 \wedge w_2$: **"w_1 and w_2"**

$w_1 \wedge w_2$ has the truth value ***True*** only if both w_1 and w_2 have the truth el valor ***True***.
Otherwise it has the truth value ***False***.

w_1	w_2	$w_1 \wedge w_2$
True	*True*	*True*
True	*False*	*False*
False	*True*	*False*
False	*False*	*False*

Example:
A: "The mechanical arm has been activated"
P: "The object is liftable"

A $\wedge$ P: "The mechanical arm has been activated **and** the object is liftable"

Truth tables: logical connective "or"

$w_1 \vee w_2$: **"w_1 or w_2"**

$w_1 \vee w_2$ has the truth value ***True*** if at least one of the WFF's w_1 or w_2 has the truth value ***True***.
Otherwise it has the truth value ***False***.

w_1	w_2	$w_1 \vee w_2$
True	*True*	*True*
True	*False*	*True*
False	*True*	*True*
False	*False*	*False*

Example:
¬ P : "The object is **not** liftable"
¬ B : "The battery is **not** loaded"

¬ P ∨ ¬ B: "The object is **not** liftable **or** the battery is **not** loaded (or both)"

Truth tables: logical connective "implies"

$w_1 \Rightarrow w_2$: "w_1 implies w_2", "If w_1, then w_2"

$w_1 \Rightarrow w_2$ has the same truth value as $(\neg w_1 \vee w_2)$

w_1	w_2		$w_1 \Rightarrow w_2$
True	*True*		*True*
True	*False*		*False*
False	*True*		*True*
False	*False*		*True*

Example:

A ∧ B ∧ P: "The mechanical arm has been activated, the battery is loaded, and the object is liftable"

D: "The object moves"

(A ∧ B ∧ P) ⇒ D: "If the mechanical arm has been activated, the battery is loaded, and the object is liftable, then the object moves"

Truth tables: logical connective "if and only if"

$w_1 \Leftrightarrow w_2$: "if and only if w_1, then w_2"

$w_1 \Leftrightarrow w_2$ has the same truth value as $(w_1 \Rightarrow w_2) \wedge (w_2 \Rightarrow w_1)$

w_1	w_2		$w_1 \Leftrightarrow w_2$
True	*True*		*True*
True	*False*		*False*
False	*True*		*False*
False	*False*		*True*

Example:

A ∧ B ∧ P: "The mechanical arm has been activated, the battery is loaded, and the object is liftable"

D: "the object moves"

(A ∧ B ∧ P) ⇔ D: "the object moves if and only if the mechanical arm has been activated, the battery is loaded, and the object is liftable"

Truth tables: Summary

w_1	w_2		$\neg w_1$	$w_1 \wedge w_2$	$w_1 \vee w_2$	$w_1 \Rightarrow w_2$	$w_1 \Leftrightarrow w_2$
T	*T*		*F*	*T*	*T*	*T*	*T*
T	*F*		*F*	*F*	*T*	*F*	*F*
F	*T*		*T*	*F*	*T*	*T*	*F*
F	*F*		*T*	*F*	*F*	*T*	*T*

Equivalence

Two **different WFF's** w_1, w_2 are **equivalent** $(w_1 \equiv w_2)$ when they have **the same truth table.**

- **Neutral element:** $(w_1 \wedge T) \equiv w_1$; $(w_1 \vee F) \equiv w_1$
- **Absorption:** $(w_1 \vee (w_1 \wedge w_2)) \equiv w_1$
  $(w_1 \wedge (w_1 \vee w_2)) \equiv w_1$
- **Contradiction / excluded middle laws :**
  $(w_1 \wedge \neg w_1) \equiv F$; $(w_1 \vee \neg w_1) \equiv T$
- **Domination laws:**
  $(w_1 \wedge F) \equiv F$; $(w_1 \vee T) \equiv T$
- **Idempotency:** $(w_1 \wedge w_1) \equiv w_1$; $(w_1 \vee w_1) \equiv w_1$
- **Elimination of double negation:** $\neg\neg w_1 \equiv w_1$
- **De Morgan's laws:**
  $\neg(w_1 \vee w_2) \equiv \neg w_1 \wedge \neg w_2$; $\neg(w_1 \wedge w_2) \equiv \neg w_1 \vee \neg w_2$
- **Commutative laws:** $w_1 \vee w_2 \equiv w_2 \vee w_1$; $w_1 \wedge w_2 \equiv w_2 \wedge w_1$
- **Associative laws:**
  $(w_1 \wedge w_2) \wedge w_3 \equiv w_1 \wedge (w_2 \wedge w_3) \equiv w_1 \wedge w_2 \wedge w_3$ [conjunction]
  $(w_1 \vee w_2) \vee w_3 \equiv w_1 \vee (w_2 \vee w_3) \equiv w_1 \vee w_2 \vee w_3$ [disjunction]
- **Distributive laws:**
  $w_1 \wedge (w_2 \vee w_3) \equiv (w_1 \wedge w_2) \vee (w_1 \wedge w_3)$
  $w_1 \vee (w_2 \wedge w_3) \equiv (w_1 \vee w_2) \wedge (w_1 \vee w_3)$.
- **Definition of the conditional:** $w_1 \Rightarrow w_2 \equiv \neg w_1 \vee w_2$
- **Contraposition:** $w_1 \Rightarrow w_2 \equiv \neg w_2 \Rightarrow \neg w_1$
- **Def. of biconditional:** $w_1 \Leftrightarrow w_2 \equiv (w_1 \Rightarrow w_2) \wedge (w_2 \Rightarrow w_1)$
  $\equiv (w_1 \wedge w_2) \vee (\neg w_1 \wedge \neg w_2)$

Interpretations / models

- **Interpretation:** An interpretation is an assignment of a set of truth values ("True" or "False") to the atoms used in the WFF's of a knowledge base.
  - » in a knowledge base whose WFF's involve n different symbolic atoms, the number of possible different interpretations is 2^n
- **Model:** An interpretation is a model of a given knowledge base if all WFF's of the knowledge base have a truth value "True" for that interpretation.

Knowledge base of the robot (situation 1)

» "If the battery is loaded and the object is liftable then the object moves when the mechanical arm has been activated ".

$w_1 : (A \wedge B \wedge P) \Rightarrow D$

» "The mechanical arm has been activated" $w_2 :$ A

» "The battery is loaded" $w_3 :$ B

» "The object has not moved" $w_4 :$ ¬ D

	Atoms				Robot's knowledge base			
	A	B	D	P	$w_1 : (A \wedge B \wedge P) \Rightarrow D$	$w_2 :$ A	$w_3 :$ B	$w_4 :$ ¬D
I_1	T	T	T	T	T	T	T	F
I_2	T	T	T	F	T	T	T	F
I_3	T	T	F	T	F	T	T	T
I_4	T	T	F	F	T	T	T	T
I_5	T	F	T	T	T	T	F	F
I_6	T	F	T	F	T	T	F	F
I_7	T	F	F	T	T	T	F	T
I_8	T	F	F	F	T	T	F	T
I_9	F	T	T	T	T	F	T	F
I_{10}	F	T	T	F	T	F	T	F
I_{11}	F	T	F	T	T	F	T	T
I_{12}	F	T	F	F	T	F	T	T
I_{13}	F	F	T	T	T	F	F	F
I_{14}	F	F	T	F	T	F	F	F
I_{15}	F	F	F	T	T	F	F	T
I_{16}	F	F	F	F	T	F	F	T

Satisfiability

- **Satisfiable (SAT):** A knowledge base is satisfiable if there is at least one interpretation that is a model of the knowledge base.

  `Example: {P,P⇒Q} is SAT`
- **Unsatisfiable (UNSAT):** A knowledge base is unsatisfiable (a contradiction) if no interpretation is a model of the knowledge base.

```
Examples: {F} is UNSAT
          {F, P⇒Q}    is UNSAT
          {P, ¬P}     is UNSAT
          {P∧¬P}      is UNSAT
          {¬Q, P⇒Q, P} is UNSAT
```

- **Tautology:** A WFF is a tautology all interpretations are models of the WFF.

```
Examples: {T}          is a TAUTOLOGY
          {P∨¬P}       is a TAUTOLOGY
          {P⇒P}        is a TAUTOLOGY
          {P⇒(Q⇒P)}    is a TAUTOLOGY
```

Note: The concepts SAT, UNSAT, TAUTOLOGY can be applied to sets of WFF's or to a single WFF's.

Reasoning by means of truth tables

- The WFF `w` is a logical consequence of the knowledge base Δ if `w` has the truth value "True" for all the interpretations that are models of Δ

$$\Delta \models w$$

```
Example 1: {P}     ⊨ P
Example 2: {P,P⇒Q} ⊨ Q
Example 3: {F}     ⊨ w
           (w any WFF)
```

- **Model checking**: To determine whether **w is a logical consequence** of Δ ($\Delta \models w$), build the corresponding **truth tables** and check whether all the models of Δ are also models of `w`.

Reasoning by means of truth tables: Situation 1

- n = 4 atoms
- 2^4 = 16 interpretations (possible alternative situations)
- 1 model (I_4) ➔ The knowledge base **Δ** is SAT
- The WFF **w** = ¬ P can be included in the knowledge base because all the models of **Δ** (I_4) are also models of **w**.

	Atoms				Knowledge base Δ				w
	A	B	D	P	A∧B∧P⇒ D	A	B	¬ D	¬ P
I_1	T	T	T	T	T	T	T	F	
I_2	T	T	T	F	T	T	T	F	
I_3	T	T	F	T	F	T	T	T	
I_4	T	T	F	F	T	T	T	T	T
I_5	T	F	T	T	T	T	F	F	
I_6	T	F	T	F	T	T	F	F	
I_7	T	F	F	T	T	T	F	T	
I_8	T	F	F	F	T	T	F	T	
I_9	F	T	T	T	T	F	T	F	
I_{10}	F	T	T	F	T	F	T	F	
I_{11}	F	T	F	T	T	F	T	T	
I_{12}	F	T	F	F	T	F	T	T	
I_{13}	F	F	T	T	T	F	F	F	
I_{14}	F	F	T	F	T	F	F	F	
I_{15}	F	F	F	T	T	F	F	T	
I_{16}	F	F	F	F	T	F	F	T	

Knowledge base of the robot (situation 2)

- "If the battery is loaded and the object is liftable, then, the object moves when the mechanical arm has been activated ".
  w_1: (A ∧ B ∧ P) ⇒ D
- "The mechanical arm has been activated" w_2: A
- "The object has not moved" w_3 : ¬ D

	Atoms				Robot's knowledge base		
	A	B	D	P	w_1 : (A∧B∧P) ⇒ D	w_2 : A	w_3: ¬D
I_1	T	T	T	T	T	T	F
I_2	T	T	T	F	T	T	F
I_3	T	T	F	T	F	T	T
I_4	T	T	F	F	T	T	T
I_5	T	F	T	T	T	T	F
I_6	T	F	T	F	T	T	F
I_7	T	F	F	T	T	T	T
I_8	T	F	F	F	T	T	T
I_9	F	T	T	T	T	F	F
I_{10}	F	T	T	F	T	F	F
I_{11}	F	T	F	T	T	F	T
I_{12}	F	T	F	F	T	F	T
I_{13}	F	F	T	T	T	F	F
I_{14}	F	F	T	F	T	F	F
I_{15}	F	F	F	T	T	F	T
I_{16}	F	F	F	F	T	F	T

Reasoning by means of truth tables: Situation 2

- n = 4 atoms
- 2^4 = 16 interpretations (possible alternative situations)
- 3 models (I_4, I_7, I_8) ➔ The knowledge base **Δ** is SAT
- The WFF **w = ¬B∨¬P** can be included in the knowledge base because all the models of **Δ** (I_4, I_7, I_8) are models of **w**.

	Atoms				Knowledge base Δ			w
	A	B	D	P	(A∧B∧P) ⇒ D	A	¬ D	¬B∨¬P
I_1	T	T	T	T	T	T	F	
I_2	T	T	T	F	T	T	F	
I_3	T	T	F	T	F	T	T	
I_4	T	T	F	F	T	T	T	T
I_5	T	F	T	T	T	T	F	
I_6	T	F	T	F	T	T	F	
I_7	T	F	F	T	T	T	T	T
I_8	T	F	F	F	T	T	T	T
I_9	F	T	T	T	T	F	F	
I_{10}	F	T	T	F	T	F	F	
I_{11}	F	T	F	T	T	F	T	
I_{12}	F	T	F	F	T	F	T	
I_{13}	F	F	T	T	T	F	F	
I_{14}	F	F	T	F	T	F	F	
I_{15}	F	F	F	T	T	F	T	
I_{16}	F	F	F	F	T	F	T	

Inference rules

- **Inference rules:** Typographical rules that manipulate only symbols (in consequence, they do not use the meaning of these symbols or the truth tables) and that can be used to generate new WFF's from a given set of WFF's.
- **Correct inference rules:** Inference rules such that all the models of the original WFF's are also models of the resulting WFF's.
- **Equivalence rules** are **correct inference rules**.

(Note: not every inference rule is an equivalence rule)

Set of inference rules :

Let w_1, w_2 be two WFF's

(1) *Modus ponens*: $\{w_1, w_1 \Rightarrow w_2\} \vdash_{M.P.} w_2$

(2) *Modus tollens*: $\{\neg w_2, w_1 \Rightarrow w_2\} \vdash_{M.T.} \neg w_1$

(3) ∧ introduction: $\{w_1, w_2\} \vdash_{\wedge\ INTRO} w_1 \wedge w_2$

(4) Commutativity of ∧: $\{w_1 \wedge w_2\} \vdash_{\wedge\ CONMUTA} w_2 \wedge w_1$

(5) Elimination of ∧: $\{w_1 \wedge w_2\} \vdash_{\wedge\ ELIMIN} w_1$

(6) ∨ introduction: $\{w_1\} \vdash_{\vee\ INTRO} w_1 \vee w_2$

$\{w_2\} \vdash_{\vee\ INTRO} w_1 \vee w_2$

(7) Elimination of ¬¬ : $\{\neg\neg w_1\} \vdash_{\neg\neg ELIMIN} w_1$

This set of inference rules is correct, but not complete.

Reasoning by inference

- **Proof:** The sequence of WFF's

  $\{w_1, w_2, ..., w_{n-1}, w_n\}$

  is a proof (or deduction) of w_n from the set if WFF's Δ with the set of inference rules *R* if and only if each and every one of the formulas $w_i, i=1,2,...,n$ is either in Δ or can be deduced from $\{w_1, w_2, ..., w_{i-1}\}$ by application of some inference rule in *R*.

- **Theorem**:

  w_n is a theorem of Δ with the set of inference rules *R* if there is a proof of w_n from Δ with the set of inference rules *R*

  $$\Delta \vdash_R w_n$$

Reasoning by inference: Situation 1

Knowledge base

» "If the battery is loaded and the object is liftable, then, the object moves when the mechanical arm has been activated ".
$w_1: (A \wedge B \wedge P) \Rightarrow D$

» "The mechanical arm has been activated" $w_2: A$

» "The battery is loaded" $w_3: B$

» "the object has not moved" $w_4: \neg D$

Inference

$\{A, B, \neg D\} \vdash_{\wedge INTRO} A \wedge B \wedge \neg D$

$(A \wedge B \wedge P) \Rightarrow D$

$\equiv \neg(A \wedge B \wedge P) \vee D$ [def. of $\Rightarrow$]

$\equiv \neg A \vee \neg B \vee \neg P \vee D$ [De Morgan's law]

$\equiv \neg A \vee \neg B \vee \neg\neg D \vee \neg P$ [commutative + double neg.]

$\equiv \neg(A \wedge B \wedge \neg D) \vee \neg P$ [associative + De Morgan's]

$\equiv (A \wedge B \wedge \neg D) \Rightarrow \neg P$ [def. of $\Rightarrow$]

$\{(A \wedge B \wedge \neg D), (A \wedge B \wedge \neg D) \Rightarrow \neg P\} \vdash_{M.P.} \neg P$

» $\neg P$ is a theorem of the robot's knowledge base in situation 1.

» We have built a proof of $\neg P$ ("the object is **not** liftable") from the knowledge base that describes situation 1 by inference.

Reasoning by inference: Situation 2

Knowledge base

» "If the battery is loaded and the object is liftable, then, the object moves when the mechanical arm has been activated ".
$w_1: (A \wedge B \wedge P) \Rightarrow D$

» "The mechanical arm has been activated" $w_2: A$

» "the object has not moved" $w_3: \neg D$

Inference

$\{A, \neg D\} \vdash_{\wedge INTRO} A \wedge \neg D$

$(A \wedge B \wedge P) \Rightarrow D$

$\equiv \neg(A \wedge B \wedge P) \vee D$ [def. of $\Rightarrow$]

$\equiv \neg A \vee \neg B \vee \neg P \vee D$ [De Morgan's law]

$\equiv \neg A \vee \neg\neg D \vee \neg B \vee \neg P$ [commutative + double neg.]

$\equiv \neg(A \wedge \neg D) \vee \neg B \vee \neg P$ [associative + De Morgan]

$\equiv (A \wedge \neg D) \Rightarrow (\neg B \vee \neg P)$ [def. de $\Rightarrow$]

$\{(A \wedge \neg D), (A \wedge \neg D) \Rightarrow (\neg B \vee \neg P)\} \vdash_{M.P.} (\neg B \vee \neg P)$

» $(\neg B \vee \neg P)$ is a theorem of the robot's knowledge base in situation 2.

» We have built a proof of $(\neg B \vee \neg P)$ ("The battery is not loaded or the object is not liftable") from the knowledge base that describes situation 2 by inference.

Correctness and completeness

- **Correctness**

  A set of inference rules R is said to be correct if for every set of WFF's Δ and WFF w

  $\Delta \vdash_R w$ implies $\Delta \models w$

- **Completeness:**

  A set of inference rules R is said to be complete if for every set of WFF's Δ and WFF w

  $\Delta \models w$ implies $\Delta \vdash_R w$

- To determine whether **w is a logical consequence** of Δ $(\Delta \models w)$
  - **Model checking**: Build **truth tables** and check whether all the models of Δ are also models of w
  - Assume we have a set of **correct and complete** inference rules ***R***. We can **use a complete search method (e.g. BFS) to find a proof** $\Delta \vdash_R w$
    - Node: Knowledge base.
    - Adjacent nodes: Knowledge base extended by the inclusion of WFF's derived by application of some inference rule in R.

Tricks

- The following substitutions transform the evaluation of the truth values of a WFF in propositional logic into an arithmetic problem

Logic	Arithmetic
F	0
T	1
$\wedge$	$\cdot$
$\vee$	$+$

- The notation $\overline{P}$ is used in some texts as an alternative to $\neg$P
- In US books, one typically finds $\supset$ instead of $\Rightarrow$ to denote implication.
- Construction of truth tables

P	$\Rightarrow$ [3]	((P	$\vee$ [1]	Q)	$\Leftrightarrow$ [2]	Q)
F	T	F	F	F	T	F
F	T	F	T	T	T	T
T	F	T	T	F	F	F
T	T	T	T	T	T	T

Knowledge-based agent

» **A knowledge base:** Collection of well-formed formulas (WFFs) that represent true statements about the actual situation of the agent in the real world.

» **A mechanism to determine the truth value of propositions:**
- Innate knowledge (e.g. rules, restrictions)
- Sensors that allow to directly check the truth value of a proposition.

» **A lookup mechanism** giving access to the knowledge stored in the knowledge base.

» **A correct inference mechanism** that makes it possible to derive new WFF's from the ones that the agent has collected and to incorporate them into the knowledge base.

If the WFFs in the knowledge base have truth value "*True (T)*" in the actual situation of the agent in the real world, the WFFs derived from them by means of correct inference rules also have truth value "*True*" in that situation.

» **A complete inference mechanism**

It is desirable, but not strictly necessary

[Gödel: It is not always possible]

Meta-linguistics

It is important not to mix up:

- **Lingüistic symbols,** which are used to form WFF's **in** propositional logic:

  T, F, P, Q, P1, P2…, $\vee$, $\wedge$, $\neg$, $\Rightarrow$, $\Leftrightarrow$

- **Meta-lingüistic symbols,** which are used to speak **about** propositional logic:

  $\models$ (logical consequence), $\vdash_R$ (inference)

These are different:

- **Theorems in** propositional logic: WFF generated by the application of correct inference rules to a set of WFF's.
- **Meta-theorems:** Assertions **about** propositional logic.

  Question: Consider the following (correct) assertion

  "If $\Delta = \{w_1, w_2, ..., w_n\} \models w$ then $(w_1 \wedge w_2 \wedge ... \wedge w_n) \Rightarrow w$ is a tautology"

  is it a theorem or a meta-theorem?

Automated reasoning

Given a set of WFF's $\Delta = \{w_1, w_2, ..., w_n\}$

and a WFF w

How does one show $\Delta \models w$?

(i) Checking that all the models of Δ are also models of w [truth tables]

(ii) Showing that w is a theorem of Δ with the set of correct inference rules R

$\Delta \vdash_R w$ [inference]

(iii) Checking that WFF $(w_1 \wedge w_2 \wedge ... \wedge w_n) \Rightarrow w$ is a tautology [inference / truth tables]

(iv) Proof by contradiction (reductio ad absurdum, refutation)

Show that the set of WFF's that includes the WFF's of Δ and the negation of the goal ($\neg w$)

$\alpha = \{w_1, w_2, ..., w_n, \neg w\}$ is UNSAT

[inference / truth tables]

Normal forms

- **A literal** is an atom (positive literal) or an atom preceded by the logical connective ¬ (negative literal).
  E.g. <u>Positive literal</u>: P
  <u>Negative literal</u>: ¬P
- **Clause**: Disjunction of literals
  E.g. P∨Q∨¬R
  - » **Unit clause**: A clause composed of a single literal. E.g. ¬R
  - » **Empty clause**: A clause with no literals.
    - – It is represented by the symbol □.
    - – Its truth value is F
- **Conjunctive normal form (CNF):** A WFF that consists of a conjunction of clauses (i.e. a conjunction of disjunction of literals) is in conjunctive normal form.
  - » <u>**Meta-theorem:**</u> There is an algorithm to transform any WFF in propositional logic to an equivalent CNF.
- **Disjunctive normal form (DNF):** A WFF that consists of a disjunction of conjunctions of literals is in disjunctive normal form.
  - » <u>**Meta-theorem (dual):**</u> There is an algorithm to transform any WFF in propositional logic to an equivalent CNF.

Transformation to CNF

Algorithm to transform a WFF in propositional login into an **equivalent CNF:**

1. **Eliminate the double implications**
   $w_1 \Leftrightarrow w_2 \equiv (w_1 \Rightarrow w_2) \wedge (w_2 \Rightarrow w_1)$
2. **Eliminate the implications**
   $w_1 \Rightarrow w_2 \equiv \neg w_1 \vee w_2$
3. **Reduce the scope of negations**
   - » De Morgan's laws
   - » Elimination of double negation

   These rules are recursively applied until all the logical connectives for negation (¬) appear immediately before atoms.

4. Transform to CNF using **associative and distributive laws.**

5. Simplify the resulting expressions using **equivalence rules (idempotency, absorption, etc.)**.

Example: Transformation to CNF, I

- **Transform** to an **equivalent CNF** the WFF
  ((P⇔Q)⇒(R⇒S))∧(Q⇒¬(P∧R))

1. Eliminate the double implications
   (((P⇒Q)∧(Q⇒P))⇒(R⇒S))∧(Q⇒¬(P∧R))
2. Eliminate the implications
   (¬((¬P∨Q)∧(¬Q∨P))∨(¬R∨S))∧(¬Q∨¬(P∧R))
3. Reduce the scope of negation
   ((¬(¬P∨Q)∨¬(¬Q∨P))∨(¬R∨S))∧(¬Q∨¬P∨¬R)
   (((¬¬P∧¬Q)∨(¬¬Q∧¬P)))∨(¬R∨S))∧(¬Q∨¬P∨¬R)
   ((P∧¬Q)∨(Q∧¬P)∨(¬R∨S))∧(¬Q∨¬P∨¬R)
4. Apply associative / distributive laws
   ((P∧¬Q)∨(Q∧¬P)∨(¬R∨S))∧(¬Q∨¬P∨¬R)

   Consider ((P∧¬Q)∨(Q∧¬P)∨¬R∨S):
   [matrix with disjunctions implicit by row
   conjunctions implicit among the elements in the same row]

P	¬Q	(generate from this matrix conjunctions of
Q	¬P	disjunctions so that each disjunction contains
¬R		exactly one element from each row)
S		

   (P∨Q∨¬R∨S)∧(P∨¬P∨¬R∨S)∧(¬Q∨Q∨¬R∨S)∧
   (¬Q∨¬P∨¬R∨S)∧(¬Q∨¬P∨¬R)
4. Simplify the expression using equivalence rules
   (P∨Q∨¬R∨S)∧(¬Q∨¬P∨¬R∨S)∧(¬Q∨¬P∨¬R)

Example: Transformation to CNF, II

- **Transform** to an **equivalent CNF** the WFF
  ((P⇔Q)⇒(¬R∧S))∧(Q⇒¬(P∧R))

1. Elimination of the double implications
   (((P⇒Q)∧(Q⇒P))⇒(¬R∧S))∧(Q⇒¬(P∧R))
2. Elimination of the implication
   (¬((¬P∨Q)∧(¬Q∨P))∨(¬R∧S))∧(¬Q∨¬(P∧R))
2. Reduce the scope of ¬
   ((¬(¬P∨Q)∨¬(¬Q∨P))∨(¬R∧S))∧(¬Q∨¬P∨¬R)
   ((¬¬P∧¬Q)∨(¬¬Q∧¬P))∨(¬R∧S))∧(¬Q∨¬P∨¬R)
   (((P∧¬Q)∨(Q∧¬P))∨(¬R∧S))∧(¬Q∨¬P∨¬R)
3. Apply associative / distributive laws
   ((P∧¬Q)∨(Q∧¬P)∨(¬R∧S))∧(¬Q∨¬P∨¬R)

P	¬Q	(generate from this matrix a conjunction of
Q	¬P	8 clauses. Each clause contains exactly
¬R	S	one element from each row)

   (P∨Q∨¬R)∧(P∨Q∨S)∧(P∨¬P∨¬R)∧(P∨¬P∨S)∧
   (¬Q∨Q∨¬R)∧(¬Q∨Q∨S)∧(¬Q∨¬P∨¬R)∧(¬Q∨¬P∨S)∧
   (¬Q∨¬P∨¬R)

4. Simplify the expression using equivalence rules
   (P∨Q∨¬R)∧(P∨Q∨S)∧(¬Q∨¬P∨¬R)∧(¬Q∨¬P∨S)

Reasoning with CNF

Notation

K_n : n-th clause
λ_i : i-th literal

- **Clauses**
  - **The empty clause (□) is UNSAT.**
  - **A non-empty clause is always SAT.**
  - **A clause is a tautology if it is of the form** $K = (\lambda_1 \vee \neg\lambda_1 \vee \lambda_2 \ldots \vee \lambda_i)$, **where λ_1 is a positive literal.**

- **CNF's**
  - **An empty CNF is a tautology.**
  - **A non-empty CNF ($K_1 \wedge K_2 \wedge \ldots \wedge K_n$) is a tautology** if and only if each of the clauses of which it is composed is a tautology.
  - **A CNF ($K_1 \wedge K_2 \wedge \ldots \wedge K_n \wedge$ □) that contains the empty clause is UNSAT.**
  - There are UNSAT CNF's that do not contain the empty clause.

Simplification of CNF's

- Because the logical connectives $\wedge$ and $\vee$ are commutative and associative, it is not necessary to specify the order of the literals in a clause, or of clauses in a CNF.

  E.g. $(A \vee \neg B) \wedge (\neg B \vee C) \equiv (C \vee \neg B) \wedge (A \vee \neg B)$
- The WFF $(K_1 \wedge K_2 \wedge \ldots \wedge K_n)$ and the set of WFF's $\{K_1, K_2, \ldots, K_n\}$ have the same models.

  E.g. $(A \vee \neg B) \wedge (\neg B \vee C) \equiv \{(A \vee \neg B), (\neg B \vee C)\}$
- Repeated literals can be eliminated from a clause.

  E.g. $(A \vee \neg B \vee C \vee \neg B \vee C) \equiv (A \vee \neg B \vee C)$
- Some clauses can be eliminated from a CNF
  - Repeated clauses

    E.g. $(A \vee \neg B) \wedge (A \vee \neg B) \equiv (A \vee \neg B)$
  - Tautological clauses

    E.g. $(A \vee \neg A) \wedge (A \vee \neg B) \equiv T \wedge (A \vee \neg B) \equiv (A \vee \neg B)$
  - Subsumed clauses (absorption)

    E.g. $(A \vee \neg B) \wedge (A \vee \neg B \vee C) \equiv (A \vee \neg B)$

From this moment on, we assume that repetitions and tautologies have been removed from the knowledge base in CNF.

Resolution between clauses

- **Resolution** between clauses

Let λ be a positive literal.

let K_1 and K_2 be two clauses of the form

$K_1 = (\lambda \vee \lambda_{11} \vee \lambda_{21} \ldots \vee \lambda_{i1})$
$K_2 = (\neg\lambda \vee \lambda_{12} \vee \lambda_{22} \ldots \vee \lambda_{j2})$,

The derivation

$\{K_1, K_2\} \vdash_{[\text{RES on } \lambda]} \lambda_{11} \vee \lambda_{21} \ldots \vee \lambda_{i1} \vee \lambda_{12} \vee \lambda_{22} \ldots \vee \lambda_{j2}$

is a correct inference rule

Examples of resolutions

$\{\neg A \vee B, A\} \vdash_{[\text{RES On } A]} B$
$\{\neg A \vee B \vee C, C \vee D \vee A\} \vdash_{[\text{RES On } A]} B \vee C \vee D$
$\{\neg A \vee B \vee C, \neg C \vee A\} \vdash_{[\text{RES On } A]} B \vee C \vee \neg C \equiv T$
$\{\neg A \vee B \vee C, \neg C \vee A\} \vdash_{[\text{RES On } C]} \neg A \vee B \vee A \equiv T$

$\{\neg A, A\} \vdash_{[\text{RES en } A]} \equiv$ □ $\equiv F$ (UNSAT!)
□ is the empty clause (truth value "False")

Resolution between clauses is correct

$K_1 = \lambda \vee \Sigma_1$, $\Sigma_1 = \lambda_{11} \vee \lambda_{21} \ldots \vee \lambda_{i1}$
$K_2 = \neg\lambda \vee \Sigma_2$, $\Sigma_2 = \lambda_{12} \vee \lambda_{22} \ldots \vee \lambda_{j2}$

$\{K_1, K_2\} \vdash_{[\text{RES on } \lambda]} (\Sigma_1 \vee \Sigma_2)$
implies
$\{K_1, K_2\} \models (\Sigma_1 \vee \Sigma_2)$

			$\Delta = \{K_1, K_2\}$		w
λ	Σ_1	Σ_2	$\lambda \vee \Sigma_1$	$\neg\lambda \vee \Sigma_2$	$\Sigma_1 \vee \Sigma_2$
T	***T***	***T***	***T***	***T***	***T***
T	*T*	*F*	*T*	*F*	
T	***F***	***T***	***T***	***T***	***T***
T	*F*	*F*	*T*	*F*	
F	***T***	***T***	***T***	***T***	***T***
F	***T***	***F***	***T***	***T***	***T***
F	*F*	*T*	*F*	*T*	
F	*F*	*F*	*F*	*T*	

Resolution + refutation is complete

- **Resolution by itself is not complete:**
  There are WFF's that cannot be obtained by resolution from a set of WFF's in CNF.

  Example: $\{P\} \models P \vee Q$
  cannot be derived by resolution from the set of clauses $\{P\}$

  In contrast, by refutation

  $\Delta = \{P\}$

  $w = (P \vee Q)$; $\neg w = \neg(P \vee Q) \equiv \neg P \wedge \neg Q$

  $\alpha = \{P, \neg(P \vee Q)\} \equiv \{P, \neg P, \neg Q\}$

  Resolving on P between the first and the second clauses of α

  $\{P, \neg P\} \vdash_{R[RES\ on\ P]} \square$ (empty clause)

  we conclude that α is UNSAT, and, in consequence

  $\{P\} \models (P \vee Q)$;

Resolution + refutation

Given a set of WFF's Δ, and a WFF $\mathbf{w}$

1. Transform the WFF's of Δ into an equivalent set of clauses (transform to CNF + $\wedge$ elimination).

   $\Delta \equiv \Delta_{CNF}$
2. Transform $\neg\mathbf{w}$ into an equivalent set of clauses.

   $\neg\mathbf{w} \equiv (\neg\mathbf{w})_{CNF}$
3. Combine the clauses that result from steps 1 and 2 into a single set of clauses $\alpha_{CNF} \equiv \{\Delta_{CNF}, (\neg\mathbf{w})_{CNF}\}$
4. Repeat

   4.1 Eliminate repetitions and tautologies from α_{CNF}.

   4.2 Perform all possible resolutions in α_{CNF}, incorporating the resulting clauses to α.

   until either the empty clause ($\square$) is derived

   α_{CNF} **is UNSAT, therefore** $\Delta \models \mathbf{w}$

   or no new clauses are generated

   α_{CNF} **is SAT, therefore** $\Delta \not\models \mathbf{w}$

Resolution + refutation is complete and decidable

- **Resolution among clauses+refutation is complete:**
  If, as a result of applying resolution + refutation to a set of WFF's, Δ and to WFF w the empty clause is derived, then $\Delta \models w$.
- **Propositional calculus is decidable by resolution + refutation:**
  Consider a a set of WFF's Δ and a WFF w, such that $\Delta \not\models w$, the procedure resolution + refutation comes to a halt without having generated the empty clause.

  EXAMPLE 1:

  $\Delta \equiv \{A \vee B, A \Rightarrow C\}$; $w \equiv C$

  $\alpha_{CNF} \equiv \{A \vee B, \neg A \vee C, \neg C\}$

  $\{A \vee B, \neg A \vee C\} \vdash_{[RES\ on\ A]} B \vee C$

  $\{\neg C, \neg A \vee C\} \vdash_{[RES\ on\ C]} \neg A$

  $\{\neg A, A \vee B\} \vdash_{[RES\ on\ A]} B$

  α_{CNF} is SAT, therefore $\Delta \not\models w$

  EXAMPLE 2:

  $\Delta \equiv \{A, (A \vee B) \Rightarrow C\}$; $w \equiv C$

  $\alpha_{CNF} \equiv \{A, \neg A \vee C, \neg B \vee C, \neg C\}$

  $\{\neg A \vee C, \neg C\} \vdash_{[RES\ on\ C]} \neg A$

  $\{\neg A, A\} \vdash_{[RES\ on\ A]} \square$

  α_{CNF} is UNSAT, therefore $\Delta \models w$

Common errors

- From the WFF $\{A \vee B\}$, it is not possible to infer $\{A, B\}$ separately.
- Consider the clauses $\{\neg A \vee B \vee C, A \vee \neg B\}$
  The only possible resolutions are tautologies

  $\neg B \vee B \vee C \equiv T$ [resolution on A]

  $\neg A \vee A \vee C \equiv T$ [resolution on B]

  It is not correct to resolve simultaneously on 2 atoms (A and B) to get C.
- Consider the knowledge base Δ, and the WFF w
  - » If w is a logical consequence of Δ, it is not possible to conclude that there are models of Δ (it could be UNSAT)

    $\Delta \models w$ [Δ can be SAT or UNSAT]
  - » If w is not a logical consequence of Δ, there is at least a model of Δ (which is not a model of w)

    $\Delta \not\models w$ [Δ is SAT]

Exercises (Nilsson, chap. 13)

- Show using truth tables the equivalence
  $\neg(P \wedge Q) \equiv (\neg P \vee \neg Q)$

- Show that if Δ, a set of WFF's, is unsatisfiable, then
  $\Delta \models w$, for any WFF w.

- Using truth tables, show that *modus ponens* is a correct inference rule.

- Given a set of WFF's Δ and a WFF w all these alternatives are possible

S1:	$\Delta \not\models w$	$\Delta \not\models \neg w$
S2:	$\Delta \models w$	$\Delta \not\models \neg w$
S3:	$\Delta \not\models w$	$\Delta \models \neg w$
S4:	$\Delta \models w$	$\Delta \models \neg w$ (Δ is UNSAT)