Knowledge Representation and Reasoning. Propositional Logic

Lecture Notes

for

Computer Science Students
Faculty EAIiIB AGH

Antoni Ligęza

Other support material:

`http://home.agh.edu.pl/~ligeza`

`http://ai.ia.agh.edu.pl/wiki/`

References

1. Mordechai Ben-Ari: Mathematical Logic for Computer Science (Logika matematyczna w informatyce). Springer-Verlag, London, 2001 (WN-T, Warszawa, 2005).
2. Kenneth A. Ross i Charles R. B. Wright: Discrete Mathematics (Matematyka dyskretna). WN PWN, 2013.
3. Antoni Ligęza: Logical Foundations for Rule-Based Systems. Springer-Verlag, Berlin, 2006. Wydawnictwo AGH, Kraków, 2005.
4. Michael R. Genesereth, Nils J. Nilsson: Logical Foundations of Artificial Intelligence. Morgan Kaufmann Publishers, Inc., Los Altos, California, 1987.
5. Zbigniew Huzar: Elementy logiki dla informatyków. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 2007.
6. Stuart Russell, Peter Norvig: Artificial Intelligence. A Modern Approach. Pearson, 2010.
7. Marek Wójcik: Zasada rezolucji. Metoda automatycznego wnioskowania. PWN, Warszawa, 1991.
8. C. L. Chang and R. C. T. Lee: Symbolic Logic and Mechanical Theorem Proving. Academic Press, 1973.
9. Ronald J. Brachman and Hector J. Levesque: Knowledge Representation and Reasoning. Morgan Kaufmann, 2004.
10. Frank van Harmelen, Vladimir Lifschitz, Bruce Porter (Eds.): *Handbook of Knowledge Representation*. Elsevier B.V., Amsterdam, 2008. `http://ii.fmph.uniba.sk/~sefranek/kri/handbook/`

On the Net. Stanford: Coursera

Stanford on-line Course:

https://www.coursera.org/course/intrologic

1. Wikipedia-pl: http://pl.wikipedia.org/wiki/Logika_matematyczna
2. Wikipedia-en: http://en.wikipedia.org/wiki/Logic
3. AI-Lab-Prolog: http://ai.ia.agh.edu.pl/wiki/pl:prolog:prolog_lab
4. EIS-KRR: http://ai.ia.agh.edu.pl/wiki/pl:dydaktyka:krr:start
5. ALI-home: home.agh.edu.pl/~ligeza
6. David Poole and Allen Mackworth: Artificial Intelligence. Foundations of Computational Agents. http://artint.info/
7. Ulf Nilsson and Jan Maluszynski: Logic, Programming and Prolog. http://www.ida.liu.se/~ulfni/lpp/

A Problem to Start: Tracking the Murderer

Some knowledge specification — in natural language:

- If Sarah was drunk then either James is the murderer or Sarah lies,
- Either James is the murderer or Sarah was not drunk and the crime took place after midnight,
- If the crime took place after midnight then either James is the murderer or Sarah lies,
- Sarah does not lie when sober.

Introduction symbols and transformation to formal specification:

- A = James is the murderer,
- B = Sarah is drunk,
- C = Sarah lies,
- D = The murder took place after midnight.

$$B \Longrightarrow A \vee C$$

$$A \vee (\neg B \wedge D)$$

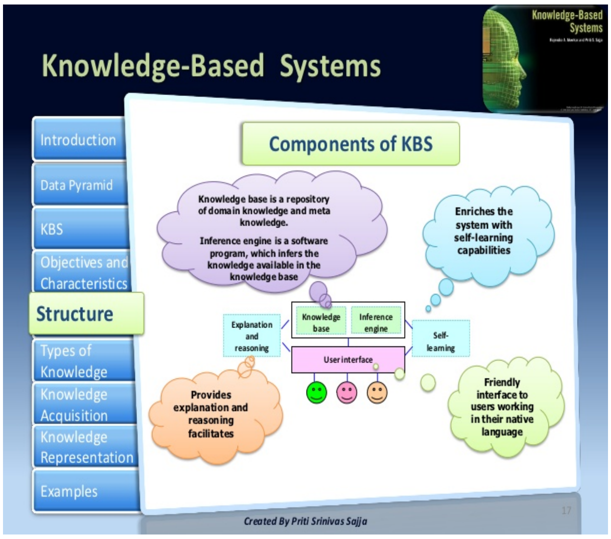
$$D \Longrightarrow (A \vee C)$$

$$C \Longrightarrow B$$

Questions:

Who is the murderer? Which facts are true/false? Is the system consistent? How many models does it have (if consistent)? What are the exact models?

Knowledge-Based Systems: Basic Concepts



We need some tools:

- A formal language for KRR:
  - alphabet
  - syntax,
  - semantics,
  - inference rules;
- various goals — various types of reasoning,
- inference strategies,
- knowledge acquisition,
- knowledge verification,
- minimal knowledge representation (uniform representation),
- internal knowledge structure (knowledge graphs),
- user interface, explanations, learning, adaptation, optimization.

Formal Knowledge Representation Language: Specification of Requirements

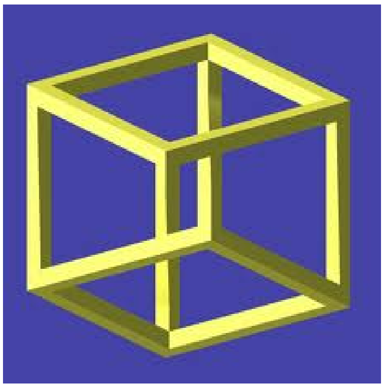
Knowledge Representation Language:

- unique interpretation,
- expressive power and precision, but
- automated processing,
- adequate for the domain,
- readable for man and machine,
- extensible,
- ...

Knowledge Base:

- consistent, (internal consistency)
- complete,
- valid; sound, (external consistency)
- non-redundant,
- efficient in problem solving, (optimal)
- ...

Something has gone wrong...



Logic — how it works?

Logic – formal language:

- alphabet,
- *syntax*,
- *semantics*,
- axiomatization

$$\models p \vee \neg p$$

$$\not\models p \wedge \neg p$$

- equivalency-preserving transformations,
- inference rules (logical consequence),
- inference chain – derivation,
- problem, hypothesis, question,
- answer, solution, proof.

System modeling :

- language selection – for adequate modeling,
- model building (inputs – outputs – internal structure),
- model analysis (verification, validation),
- model exploration – theorem proving, SAT,
- model tuning and adaptation; learning.

Example: Unicorn

Given the following Knowledge Base (KB):

- If the unicorn is mythical, then it is immortal
- If the unicorn is not mythical, then it is a mortal mammal
- If the unicorn is either immortal or a mammal, then it is horned
- The unicorn is magical if it is horned

answer the following questions:

- Is the unicorn mythical? (M)
- Is it magical? (G)
- Is it horned? (H)

In terms of logic:

$$\text{KB} \models G, H, M$$

$$\text{KB} \vdash G, H, M$$

Logic for KRR – Tasks and Tools

- Theorem Proving – Verification of Logical Consequence:

$$\Delta \models H;$$

- Automated Inference – Derivation:

$$\Delta \vdash H;$$

- SAT (checking for models) – satisfiability:

$$\models_I H;$$

- un-SAT verification – unsatisfiability:

$$\not\models_I H \quad \text{for any interpretation I;}$$

- Tautology verification (completeness):

$$\models H$$

- valid inference rules – checking:

$$(\Delta \vdash H) \quad \longrightarrow \quad (\Delta \models H)$$

- complete inference rules – checking:

$$(\Delta \models H) \quad \longrightarrow \quad (\Delta \vdash H)$$

Unicorn - Logical Model

Definition of propositional variables:

- M: The unicorn is mythical
- I: The unicorn is immortal
- L: The unicorn is mammal
- H: The unicorn is horned
- G: The unicorn is magical

Building a **Logical Model** for the puzzle:

- If the unicorn is mythical, then it is immortal:

$$M \longrightarrow I$$

- If the unicorn is not mythical, then it is a mortal mammal:

$$\neg M \longrightarrow (\neg I \wedge L)$$

- If the unicorn is either immortal or a mammal, then it is horned:

$$(I \vee L) \longrightarrow H$$

- The unicorn is magical if it is horned:

$$H \longrightarrow G$$

Resulting Boolean formula (the Knowledge Base):

$$(M \longrightarrow I) \wedge (\neg M \longrightarrow (\neg I \wedge L)) \wedge ((I \vee L) \longrightarrow H) \wedge (H \longrightarrow G)$$

A Solution

$$(M \longrightarrow I) \equiv (\neg M \vee I)$$

$$(\neg M \longrightarrow (\neg I \wedge L)) \equiv (M \vee (\neg I \wedge L))$$

$$(M \vee (\neg I \wedge L)) \equiv ((M \vee \neg I) \wedge (M \vee L))$$

$$\frac{\neg M \vee I, M \vee L}{I \vee L}$$

$$\frac{I \vee L, (I \vee L) \longrightarrow H}{H}$$

$$\frac{H, H \longrightarrow G}{G}$$

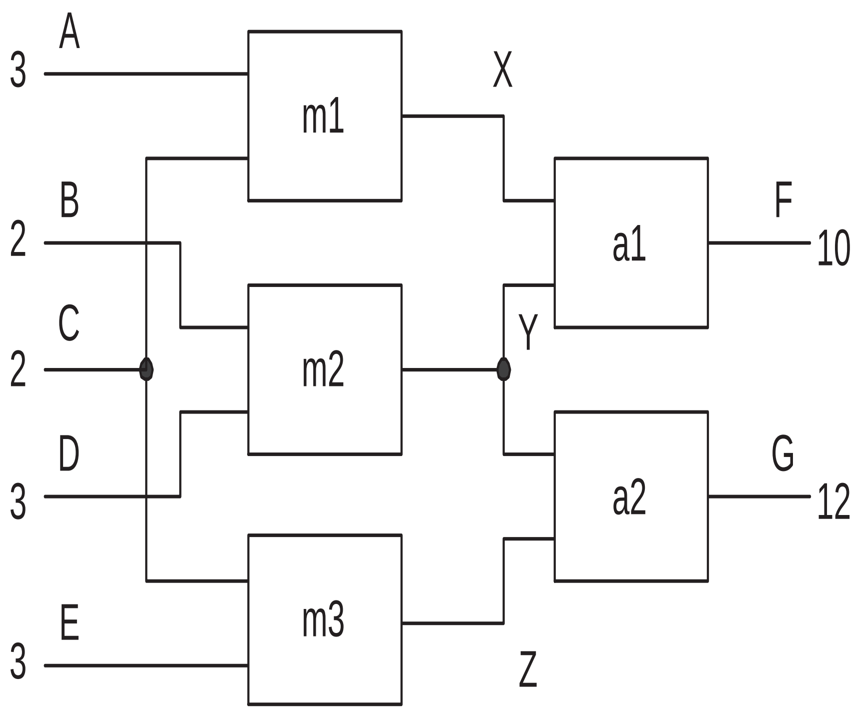
So we have:

$$\mathrm{KB} \vdash H \wedge G$$

Questions:

- What about M (mythical), I (immortal) and L (mammal)?
- What combinations are admissible?
- How many models do we have?

Abductive Inference and Consistency-Based Reasoning



Questions:

- Does the system work OK (Fault Detection)?
- Which component(s) is(are) faulty (Fault Isolation)?
- What are the diagnoses (minimal diagnoses)?

Inference example

A – signal from process,

P – signal added to a queue,

B – signal blocked by process,

D – signal received by process,

S – state of the process saved,

M – signal mask read,

H – signal management procedure activated,

N – procedure executed in normal mode,

R – process restart from context,

I – process must re-create context.

Rules — axiomatization:

$A \longrightarrow P$,

$P \wedge \neg B \longrightarrow D$,

$D \longrightarrow S \wedge M \wedge H$,

$H \wedge N \longrightarrow R$,

$H \wedge \neg R \longrightarrow I$,

Facts:

A, $\neg B$, $\neg R$.

Conclusions

P, D, S, M, H, I, $\neg N$.

Try to draw an *AND/OR/NOT Graph*

How to represent:

- facts?
- implication?
- disjunctive conditions?
- conjunctive conditions?
- negation?
- constraints?

Examine Forward Chaining vs Backward Chaining!

Properties of Formulas

A formula ϕ may be:

true/satisfied — for interpretation I, $\models_I \phi$,

false/unsatisfied — for interpretation I, $\not\models_I \phi$,

satisfiable — there exists interpretation I such that $\models_I \phi$,

unsatisfiable/always false — there does not exists interpretation I, $\models_I \phi$,

tautology — for any interpretation I, $\models_I \phi$; we write:

$$\models \phi$$

always false — for any interpretation I:

$$\not\models \phi$$

Extra problem

Assumptions:

A1. There are 3 houses in a row

A2. The houses are numbered 1, 2 and 3, from left to right

A3. Each house has one of the colors Blue, Green or White

A4. Each house is inhabited by one person with one of the nationalities: Dutch, German and Italian

A5. Each person drinks (exactly one) of the following beverages: Coffee, Tea and Water

Conditions (constraints):

C1 The third house is green

C2 There is one house between the house of the person drinking coffee and the blue house

C3 The person drinking water lives in the blue house

C4 The Italian lives to the left of the coffee drinking person

C5 The German lives in house two

Query:

Who lives in the 1st house? What does the Dutch drink?

Syntax

Definicja 1 *Definition of legal formulas:*

- $\top$ *i* $\bot$ *are formulas,*
- *every* $p \in P$ *is a formula,*
- *if* ϕ, ψ *are formulas, then:*
  - $\neg(\phi)$ *is a formula (also:* $\neg(\psi)$*),*
  - $(\phi \wedge \psi)$ *is a formula,*
  - $(\phi \vee \psi)$ *is a formula,*
  - $(\phi \Rightarrow \psi)$ *is a formula,*
  - $(\phi \Leftrightarrow \psi)$ *is a formula,*
  - *and nothing else.*

Set of formulas == $\mathbf{FOR}$.

Every formula has a *parsing tree*.

Atomic formulas — simple symbols.

Literals: atoms or their negations.

Semantics

Interpretation I maps propositional symbols into $\mathcal{T} = \{\mathbf{T}, \mathbf{F}\}$.

Definicja 2 *Let P be a set of propositional symbols. Interpretations is defined as:*

$$I : P \longrightarrow \{\mathbf{T}, \mathbf{F}\}, \tag{1}$$

Notation: $I(\phi) = \mathbf{T}$ is noted as $\models_I \phi$; $I(\phi) = \mathbf{F}$ is noted as $\not\models_I \phi$

Definicja 3 *The Interpretation I is extended over all formulas ϕ, ψ, φ from* **FOR** *as follows:*

- $I(\top) = \mathbf{T}$ *($\models_I \top$),*
- $I(\bot) = \mathbf{F}$ *($\not\models_I \bot$),*
- $\models_I \neg\phi$ *iff* $\not\models_I \phi$,
- $\models_I \psi \wedge \varphi$ *iff* $\models_I \psi$ *and* $\models_I \varphi$,
- $\models_I \psi \vee \varphi$ *iff* $\models_I \psi$ *or* $\models_I \varphi$,
- $\models_I \psi \Rightarrow \varphi$ *iff* $\models_I \varphi$ *or* $\not\models_I \psi$,
- $\models_I \psi \Leftrightarrow \varphi$ *iff* $\models_I (\psi \Rightarrow \varphi)$ *and* $\models_I (\varphi \Rightarrow \psi)$.

Definicja 4 **Equivalence** *Formulas ϕ and ψ are logically equivalent iff for any I:*

$$\models_I \phi \qquad \textit{iff} \qquad \models_I \psi. \tag{2}$$

Definicja 5 **Logical Implication** *Formula ψ is logical consequence of ϕ iff for any I:*

$$\textit{if} \qquad \models_I \phi \qquad \textit{then} \qquad \models_I \psi. \tag{3}$$

Truth Tables

ϕ	$\neg\phi$
F	**T**
T	**F**

ϕ	φ	$\phi \wedge \varphi$
F	**F**	**F**
F	**T**	**F**
T	**F**	**F**
T	**T**	**T**

ϕ	φ	$\phi \vee \varphi$
F	**F**	**F**
F	**T**	**T**
T	**F**	**T**
T	**T**	**T**

ϕ	φ	$\phi \Rightarrow \varphi$
F	**F**	**T**
F	**T**	**T**
T	**F**	**F**
T	**T**	**T**

ϕ	φ	$\phi \Leftrightarrow \varphi$
F	**F**	**T**
F	**T**	**F**
T	**F**	**F**
T	**T**	**T**

Tabular definitions of logical connectives

ϕ	ψ	$\neg\phi$	$\phi \wedge \psi$	$\phi \vee \psi$	$\phi \Rightarrow \psi$	$\phi \Leftrightarrow \psi$
true	*true*	*false*	*true*	*true*	*true*	*true*
true	*false*	*false*	*false*	*true*	*false*	*false*
false	*true*	*true*	*false*	*true*	*true*	*false*
false	*false*	*true*	*false*	*false*	*true*	*true*

Semantics through equivalent transformation:

- $\phi \Rightarrow \psi \equiv \neg\phi \vee \psi$,
- $\phi \Leftrightarrow \psi \equiv (\phi \Rightarrow \psi) \wedge (\phi \Leftarrow \psi)$,
- $\phi|\psi \equiv \neg(\phi \wedge \psi)$ –Sheffer function or NAND; also noted as $\overline{\phi \wedge \psi}$,
- $\phi \downarrow \psi \equiv \neg(\phi \vee \psi)$ – Pierce function or NOR; other notation $\overline{\phi \vee \psi}$,
- $\phi \bigoplus \psi \equiv (\neg\phi \wedge \psi) \vee (\phi \wedge \neg\psi)$ — EX-OR,

For n arguments there are 2^{2^n} functions, so for $n = 2$ there is 16 different functions.

Functional Completeness

Definicja 6 *A Set of Functions is functionally complete if it allows to express any logical function.*

Some examples:

AND, OR, NOT:

$$\{\neg, \wedge, \vee\}$$

AND, NOT:

$$\{\neg, \wedge\}$$

OR, NOT:

$$\{\neg, \vee\}$$

IMPLICATION, NOT:

$$\{\neg, \Rightarrow\}$$

NAND:

$$\{|\}$$

NOR:

$$\{\downarrow\}$$

Definicja 7 *A functionally complete set of functions is minimal — if it cannot be further reduced.*

For convenience, redundant systems are in use.

Most important equivalent transformations

- $\neg\neg\phi \equiv \phi$ — double negation elimination,
- $\phi \wedge \psi \equiv \psi \wedge \phi$ — conjunction alternation,
- $\phi \vee \psi \equiv \psi \vee \phi$ — disjunction alternation,
- $(\phi \wedge \varphi) \wedge \psi \equiv \phi \wedge (\varphi \wedge \psi)$ — commutativity,
- $(\phi \vee \varphi) \vee \psi \equiv \phi \vee (\varphi \vee \psi)$ — commutativity,
- $(\phi \vee \varphi) \wedge \psi \equiv (\phi \wedge \psi) \vee (\varphi \wedge \psi)$ — distributive law,
- $(\phi \wedge \varphi) \vee \psi \equiv (\phi \vee \psi) \wedge (\varphi \vee \psi)$ — distributive law,
- $\phi \wedge \phi \equiv \phi$ — idempotency,
- $\phi \vee \phi \equiv \phi$ — idempotency,
- $\phi \wedge \bot \equiv \bot$, $\phi \wedge \top \equiv \phi$ — identity,
- $\phi \vee \bot \equiv \phi$, $\phi \vee \top \equiv \top$ — identity,
- $\phi \vee \neg\phi \equiv \top$ — *tertium non datur*; excluded middle,
- $\phi \wedge \neg\phi \equiv \bot$ — falsification,
- $\neg(\phi \wedge \psi) \equiv \neg(\phi) \vee \neg(\psi)$ — De Morgan rule,
- $\neg(\phi \vee \psi) \equiv \neg(\phi) \wedge \neg(\psi)$ — De Morgan rule,
- $\phi \Rightarrow \psi \equiv \neg\psi \Rightarrow \neg\phi$ — contraposition,
- $\phi \Rightarrow \psi \equiv \neg\phi \vee \psi$ — implication elimination.

Example: Tautology Verification

$$\phi = ((p \Rightarrow r) \wedge (q \Rightarrow r)) \Leftrightarrow ((p \vee q) \Rightarrow r).$$

There are (2^3) possible interpretations.

p	q	r	$p \Rightarrow r$	$q \Rightarrow r$	$(p \Rightarrow r) \wedge (q \Rightarrow r)$	$(p \vee q) \Rightarrow r$	Φ
0	0	0	1	1	**1**	**1**	**1**
0	0	1	1	1	**1**	**1**	**1**
0	1	0	1	0	**0**	**0**	**1**
0	1	1	1	1	**1**	**1**	**1**
1	0	0	0	1	**0**	**0**	**1**
1	0	1	1	1	**1**	**1**	**1**
1	1	0	0	0	**0**	**0**	**1**
1	1	1	1	1	**1**	**1**	**1**

Other possibility — through equivalent transformations:

$$\phi \equiv ((\neg p \vee r) \wedge (\neg q \vee r)) \Leftrightarrow (\neg(p \vee q) \vee r).$$

$$\phi \equiv ((\neg p \wedge \neg q) \vee r) \Leftrightarrow (\neg(p \vee q) \vee r).$$

$$\phi \equiv (\neg(p \vee q) \vee r) \Leftrightarrow (\neg(p \vee q) \vee r).$$

Let us put: $\psi = (\neg(p \vee q) \vee r)$; so we see:

$$\phi \equiv \psi \Leftrightarrow \psi,$$

Example: Logical Consequence Verification

$$\frac{(p \Rightarrow q) \wedge (r \Rightarrow s)}{(p \vee r) \Rightarrow (q \vee s)}$$

Put:

$$\phi = (p \Rightarrow q) \wedge (r \Rightarrow s)$$

and

$$\varphi = (p \vee r) \Rightarrow (q \vee s),$$

Now, check if:

$$\phi \models \varphi. \tag{4}$$

p	q	r	s	$p \Rightarrow q$	$r \Rightarrow s$	$(p \Rightarrow q) \wedge (r \Rightarrow s)$	$p \vee r$	$q \vee s$	$(p \vee r) \Rightarrow (q \vee s)$
0	0	0	0	1	1	**1**	0	0	**1**
0	0	0	1	1	1	**1**	0	1	**1**
0	0	1	0	1	0	**0**	1	0	**0**
0	0	1	1	1	1	**1**	1	1	**1**
0	1	0	0	1	1	**1**	0	1	**1**
0	1	0	1	1	1	**1**	0	1	**1**
0	1	1	0	1	0	**0**	1	1	**1**
0	1	1	1	1	1	**1**	1	1	**1**
1	0	0	0	0	1	**0**	1	0	**0**
1	0	0	1	0	1	**0**	1	1	**1**
1	0	1	0	0	0	**0**	1	0	**0**
1	0	1	1	0	1	**0**	1	1	**1**
1	1	0	0	1	1	**1**	1	1	**1**
1	1	0	1	1	1	**1**	1	1	**1**
1	1	1	0	1	0	**0**	1	1	**1**
1	1	1	1	1	1	**1**	1	1	**1**

From analysis of columns 7 and 10 the logical consequence is confirmed (but not equivalence; see rows: 7, 10, 12 and 15).

Minterms

Definicja 8 *Literal A literal is an atomic formula p or its negation $\neg p$.*

Definicja 9 *Let $q_1, q_2, \ldots q_n$ are literals:*

$$\phi = q_1 \wedge q_2 \wedge \ldots \wedge q_n$$

is a minterm, simple conjunction or product.

Lemat 1 *Minterm is satifiable iff it does not contain a pair of complementary literals.*

Lemat 2 *Minterm is unsatisfiable iff it contains a pair of complementary literals.*

Notation:

$$\phi = q_1 \wedge q_2 \wedge \ldots \wedge q_n$$

to

$$[\phi] = \{q_1, q_2, \ldots q_n\}$$

Definicja 10 *Minterm ϕ subsumes minterm ψ iff $[\phi] \subseteq [\psi]$.*

Lemat 3 *Let ϕ and ψ are any minterms; then :*

$$\psi \models \phi \qquad \textit{iff} \qquad [\phi] \subseteq [\psi].$$

Maxterms

Definicja 11 *Let $q_1, q_2, \ldots q_n$ are literals; then:*

$$\phi = q_1 \vee q_2 \vee \ldots \vee q_n$$

is a maxterm, simple disjunction or a clause.

Lemat 4 *Maxterm is falsifiable iff it does not contain a pair of complementary literals.*

Lemat 5 *Maxterm is a tautology iff it contains a pair of complimentary literals.*

Definicja 12 *Maxterm ψ subsumes maxterm ϕ iff*

$$[\psi] \subseteq [\phi]$$

Lemat 6 *Let ϕ and ψ are any maxterms; then:*

$$\psi \models \phi \qquad \textit{iff} \qquad [\psi] \subseteq [\phi].$$

Let us consider a clause:

$$\psi = \neg p_1 \vee \neg p_2 \vee \ldots \vee \neg p_k \vee h_1 \vee h_2 \vee \ldots \vee h_m$$

After applying the de Morgan rule:

$$\neg(p_1 \wedge p_2 \wedge \ldots \wedge p_k) \vee (h_1 \vee h_2 \vee \ldots \vee h_m)$$

This can be put as:

$$p_1 \wedge p_2 \wedge \ldots \wedge p_k \Rightarrow h_1 \vee h_2 \vee \ldots \vee h_m$$

Definicja 13 *A clause of the form:*

$$\psi = \neg p_1 \vee \neg p_2 \vee \ldots \vee \neg p_k \vee h$$

is called a Horn Clause.

Alternative notations:

$$p_1 \wedge p_2 \wedge \ldots \wedge p_k \Rightarrow h.$$

In PROLOG or in DATALOG:

$$h : -p_1, p_2, \ldots, p_k.$$

also:

```
h :- p_1, p_2,..., p_k.
```

```
h if p_1 and p_2 and ... and p_k.
```

Three forms of Horn clauses:

- facts,
- full clauses,
- constraints/calls.

CNF — Conjunctive Normal Form

Definicja 14 *Formula Ψ is in Conjunctive Normal Form (CNF) iff*

$$\Psi = \psi_1 \wedge \psi_2 \wedge \ldots \wedge \psi_n$$

where $\psi_1, \psi_2, \ldots, \psi_n$ are clauses. Notation: $[\Psi] = \{\psi_1, \psi_2, \ldots, \psi_n\}$.

Examples:

Which of the following are in CNF:

1. $(p \vee g \vee \neg r) \wedge (p \vee r) \wedge \neg r$
2. $((p \wedge q) \vee \neg r) \wedge (p \vee r) \wedge \neg r$
3. $\neg(p \vee q) \wedge (p \vee r) \wedge \neg r$

Definicja 15 *Implicent — a clause, that is false then the respective formula is also false.*

Definicja 16 *A formula is in maximal CNF form iff it is composed of all full/maximal clauses:*

$$maxCNF(\Psi) = \psi_1 \wedge \psi_2 \wedge \ldots \wedge \psi_n$$

all $\psi_1, \psi_2, \ldots, \psi_n$ contain all propositional symbols in use.

Definicja 17 *Formula*

$$\Psi = \psi_1 \wedge \psi_2 \wedge \ldots \wedge \psi_n$$

in CNF is minimal iff it cannot be reduced without violating logical equivalence.

CNF — appropriate for inconsistency checking.

DNF — Disjunctive Normal Form

Definicja 18 *Formula Φ is in Disjunctive Normal Form (DNF) iff*

$$\Phi = \phi_1 \vee \phi_2 \vee \ldots \vee \phi_n$$

where $\phi_1, \phi_2, \ldots, \phi_n$ are any minterms. Notation: $[\Phi] = \{\phi_1, \phi_2, \ldots, \phi_n\}$.

Example:
Which of the following are in DNF:

1. $(p \wedge q) \vee ((p \vee \neg q) \wedge (\neg p \vee \neg q)))$
2. $(p \wedge q) \vee ((p \vee q) \vee \neg(p \wedge q)))$
3. $(p \wedge q) \vee ((p \wedge \neg q) \vee (\neg p \wedge \neg q)))$

Definicja 19 *Implicant — a minterm, such that if it is true, then the respective formula is also true.*

Definicja 20 *A maximal DNF form is any formula containing all the possible minterms:*

$$maxDNF(\Phi) = \phi_1 \vee \phi_2 \vee \ldots \vee \phi_n$$

where all the minterms $\phi_1 \vee \phi_2 \vee \ldots \vee \phi_n$ are composed of all the propositional symbols in use.

Definicja 21 *Formula*

$$\Phi = \phi_1 \vee \phi_2 \vee \ldots \vee \phi_n$$

in DNF is minimal iff it cannot be reduced without violating logical equivalency.

DNF — is appropriate for checking satisfiability.

Transformation to CNF/DNF

1. $\Phi \Leftrightarrow \Psi \equiv (\Phi \Rightarrow \Psi) \wedge (\Psi \Rightarrow \Phi)$ – elimination of equivalence,
2. $\Phi \Rightarrow \Psi \equiv \neg\Phi \vee \Psi$ – elimination of implication,
3. $\neg(\neg\Phi) \equiv \Phi$ – elimination of double negations,
4. $\neg(\Phi \vee \Psi) \equiv \neg\Phi \wedge \neg\Psi$ – De Morgan's rule,
5. $\neg(\Phi \wedge \Psi) \equiv \neg\Phi \vee \neg\Psi$ – De Morgan's rule,
6. $\Phi \vee (\Psi \wedge \Upsilon) \equiv (\Phi \vee \Psi) \wedge (\Phi \vee \Upsilon)$ – distributivity rule; towards CNF,
7. $\Phi \wedge (\Psi \vee \Upsilon) \equiv (\Phi \wedge \Psi) \vee (\Phi \wedge \Upsilon)$ – distributivity rule; towards DNF.

Example:

$$(p \wedge (p \Rightarrow q)) \Rightarrow q \equiv \neg(p \wedge (p \Rightarrow q)) \vee q \equiv$$
$$\neg(p \wedge (\neg p \vee q)) \vee q \equiv (\neg p \vee \neg(\neg p \vee q)) \vee q \equiv$$
$$(\neg p \vee (p \wedge \neg q)) \vee q \equiv \neg\mathbf{p} \vee (\mathbf{p} \wedge \neg\mathbf{q}) \vee \mathbf{q} \equiv$$
$$(\neg p \vee p) \wedge (\neg p \vee \neg q) \vee q \equiv \neg p \vee \neg q \vee q \equiv \neg p \vee \top \equiv \top.$$

Example:
Transforming CNF to DNF:

- $\phi = ((p \vee q) \wedge (p \vee r) \wedge (q \vee s) \wedge (r \vee s)), \quad \psi = ((p \wedge s) \vee (q \wedge r))$
- $\phi = ((p \vee q) \wedge (q \vee r) \wedge (r \vee p)), \quad \psi = ((p \wedge q) \vee (q \wedge r) \vee (r \wedge p))$
- $\phi = ((p \vee q \vee r) \wedge (q \vee r \vee s) \wedge (r \vee s \vee p)) \quad \psi = ((p \wedge q) \vee (p \wedge s) \vee (q \wedge s) \vee r).$

Example:

Let us reconsider:

$$\phi = (p \Rightarrow q) \wedge (r \Rightarrow s),$$

$$\varphi = (p \vee r) \Rightarrow (q \vee s).$$

We check for logical implication:

$$\phi \models \varphi.$$

Transform ϕ to DNF:

$$\begin{aligned} \phi &= (p \Rightarrow q) \wedge (r \Rightarrow s) = (\neg p \vee q) \wedge (\neg r \vee s) = \\ &= (\neg p \wedge \neg r) \vee (\neg p \wedge s) \vee (q \wedge \neg r) \vee (q \wedge s). \end{aligned}$$

and next to its maxDNF form:

$$\begin{aligned} maxDNF(\phi) = &(\neg p \wedge \neg q \wedge \neg r \wedge \neg s) \vee (\neg p \wedge \neg q \wedge \neg r \wedge s) \vee (\neg p \wedge \neg q \wedge r \wedge s) \vee \\ &(\neg p \wedge q \wedge \neg r \wedge \neg s) \vee (\neg p \wedge q \wedge \neg r \wedge s) \vee (\neg p \wedge q \wedge r \wedge s) \vee \\ &(p \wedge q \wedge \neg r \wedge \neg s) \vee (p \wedge q \wedge \neg r \wedge s) \vee (p \wedge q \wedge r \wedge s). \end{aligned}$$

Transform φ to DNF:

$$\begin{aligned} \varphi &= (p \vee r) \Rightarrow (q \vee s) = \neg(p \vee r) \vee q \vee s = (\neg p \wedge \neg r) \vee q \vee s = \\ &= (\neg p \wedge \neg r) \vee q \vee s. \end{aligned}$$

and next to its maxDNF form:

$$\begin{aligned} maxDNF(\varphi) = &(\neg p \wedge \neg q \wedge \neg r \wedge \neg s) \vee (\neg p \wedge \neg q \wedge \neg r \wedge s) \vee (\neg p \wedge \neg q \wedge r \wedge s) \vee \\ &(\neg p \wedge q \wedge \neg r \wedge \neg s) \vee (\neg p \wedge q \wedge \neg r \wedge s) \vee (\neg p \wedge q \wedge r \wedge s) \vee \\ &(\neg p \wedge q \wedge r \wedge \neg s) \vee (p \wedge q \wedge \neg r \wedge \neg s) \vee (p \wedge q \wedge \neg r \wedge s) \vee \\ &(p \wedge q \wedge r \wedge s) \vee (p \wedge q \wedge r \wedge \neg s) \vee (p \wedge \neg q \wedge \neg r \wedge s) \vee \\ &(p \wedge \neg q \wedge r \wedge s). \end{aligned}$$

It can be seen that:

$$[maxDNF(\phi)] \subseteq [maxDNF(\varphi)],$$

Logic for KRR — Tasks and Tools

- Theorem Proving — Verification of Logical Consequence:

$$\Delta \models H;$$

- Automated Inference —- Derivation:

$$\Delta \vdash H;$$

- SAT (checking for models) — satisfiability:

$$\models_I H \quad \text{(if such I exists)};$$

- un-SAT verification — unsatisfiability:

$$\not\models_I H \quad \text{(for any I)};$$

- Tautology verification (completeness):

$$\models H$$

- valid inference rules — checking:

$$(\Delta \vdash H) \quad \longrightarrow \quad (\Delta \models H)$$

- complete inference rules — checking:

$$(\Delta \models H) \quad \longrightarrow \quad (\Delta \vdash H)$$

Two basic approaches – reasoning paradigmes:

- systematic evaluation of possible interpretations — the 0-1 method; problem — combinatorial explosion,

- logical inference — derivation — with rules preserving logical consequence.

Notation: formula H is derivable from Δ:

$$\Delta \vdash H$$

Two principal issues in logical knowledge-based systems:

$$\Delta \vdash H \quad \text{versus} \quad \Delta \models H$$

Some more important inference rules

- $\frac{\alpha}{\alpha \vee \beta}$ — Disjunction Introduction,
- $\frac{\alpha,\ \beta}{\alpha \wedge \beta}$ — Conjunction Introduction,
- $\frac{\alpha \wedge \beta}{\alpha}$ — Conjunction Elimination,
- $\frac{\alpha,\ \alpha \Rightarrow \beta}{\beta}$ — Modus Ponens (modus ponendo ponens); implication elimination (EI),
- $\frac{\alpha \Rightarrow \beta,\ \neg\beta}{\neg\alpha}$ — Modus Tollens (modus tollendo tollens),
- $\frac{\alpha \vee \beta,\ \neg\alpha}{\beta}$ — Modus Tollendo Ponens,
- $\frac{\alpha \bigoplus \beta,\ \alpha}{\neg\beta}$ — Modus Ponendo Tollens,
- $\frac{\alpha \Rightarrow \beta,\ \beta \Rightarrow \gamma}{\alpha \Rightarrow \gamma}$ — Transitivity Rule,
- $\frac{\alpha \vee \gamma,\ \neg\gamma \vee \beta}{\alpha \vee \beta}$ — **Resolution Rule**,
- $\frac{\alpha \wedge \gamma,\ \neg\gamma \wedge \beta}{\alpha \wedge \beta}$ — Dual Resolution Rule; (backward) dual resolution (works backwards); also *consolution*
- $\frac{\alpha \Rightarrow \beta,\ \gamma \Rightarrow \delta}{(\alpha \vee \gamma) \Rightarrow (\beta \vee \delta)}$ — Constructive Dilemma I,
- $\frac{\alpha \Rightarrow \beta,\ \gamma \Rightarrow \delta}{(\alpha \wedge \gamma) \Rightarrow (\beta \wedge \delta)}$ — Constructive Dilemma II.

The Deduction Theorems

Twierdzenie 1 *Let $\Delta_1, \Delta_2, \ldots \Delta_n$ and Ω are logical formulas. Ω is their logical consequence iff $\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \Rightarrow \Omega$ is a tautology.*

Twierdzenie 2 *Let $\Delta_1, \Delta_2, \ldots \Delta_n$ and Ω are logical formulas. Ω is their logical consequence iff $\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \wedge \neg\Omega$ is invalid (false under any interpretation).*

Theorem proving: having $\Delta_1, \Delta_2, \ldots \Delta_n$ assumed to be true show that so is Ω. Hence:

$$\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \models \Omega$$

Basic methods for theorem proving:

- evaluation of all possible interpretations (the 0-1 method),
- direct proof (forward chaining) – derivation of Ω from initial axioms,
- search for proof (backward chaining) – search for derivation of Ω from initial axioms; Backtracking Search,
- proving tautology – from the Deduction Theorem 1 we prove that $\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \Rightarrow \Omega$ is a tautology,
- indirect proof – through constraposition:
  $\neg\Omega \Rightarrow \neg(\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n)$.
- Reductio ad Absurdum; basing on Deduction Theorem 2 we show that
  $\Delta_1 \wedge \Delta_2 \wedge \ldots \Delta_n \wedge \neg\Omega$.
  is unsatisfiable

Examples

Direct proof: $(p \Rightarrow r) \wedge (q \Rightarrow s) \wedge (\neg r \vee \neg s) \models (\neg p \vee \neg q)$:

1. $p \Rightarrow r$ assumption,
2. $q \Rightarrow s$ assumption,
3. $\neg r \vee \neg s$ assumption,
4. $s \Rightarrow \neg r$ implication reconstruction; through equivalence to 3,
5. $q \Rightarrow \neg r$ transitivity 2 and 4,
6. $\neg p \vee r$ EI from 1,
7. $\neg q \vee \neg r$ EI from 5
8. $\neg p \vee \neg q$ by resolution rule from 6 and 7.

Proving tautology: $[p \Rightarrow (q \Rightarrow r)] \models [q \Rightarrow (p \Rightarrow r)]$.
We transform the formula $[p \Rightarrow (q \Rightarrow r)] \Rightarrow [q \Rightarrow (p \Rightarrow r)]$ and through elimination of implications we obtain $\alpha \vee \neg\alpha$.

Indirect proof: $p \models \neg q \Rightarrow \neg(p \Rightarrow q)$

1. $\neg(\neg q \Rightarrow \neg(p \Rightarrow q))$ assumption (contraposition),
2. $\neg(q \vee \neg(p \Rightarrow q))$ EI,
3. $(\neg q \wedge (p \Rightarrow q))$ De Morgan rule,
4. $\neg q$ CE,
5. $p \Rightarrow q$ CE from 3,
6. $\neg p \vee q$ EI from 5,

7. $q \vee \neg p$ commutativity from 6,

8. $\neg p$ RR from 4 and 7.

Reductio ad Absurdum: $(p \vee q) \wedge \neg p \models q$

1. $p \vee q$ assumption,

2. $\neg p$ assumption,

3. $\neg q$ assumption (negation of the hypothesis),

4. q RR to 1 and 2

5. $\perp$ from 3 and 4.

Example: Logical Consequence

$$\frac{(p \Rightarrow q) \wedge (r \Rightarrow s)}{(p \vee r) \Rightarrow (q \vee s)}$$

Let us put:

$$\phi = (p \Rightarrow q) \wedge (r \Rightarrow s)$$

and

$$\varphi = (p \vee r) \Rightarrow (q \vee s),$$

So we have to check if:

$$\phi \models \varphi. \tag{5}$$

p	q	r	s	$p \Rightarrow q$	$r \Rightarrow s$	$(p \Rightarrow q) \wedge (r \Rightarrow s)$	$p \vee r$	$q \vee s$	$(p \vee r) \Rightarrow (q \vee s)$
0	0	0	0	1	1	**1**	0	0	**1**
0	0	0	1	1	1	**1**	0	1	**1**
0	0	1	0	1	0	**0**	1	0	**0**
0	0	1	1	1	1	**1**	1	1	**1**
0	1	0	0	1	1	**1**	0	1	**1**
0	1	0	1	1	1	**1**	0	1	**1**
0	1	1	0	1	0	**0**	1	1	**1**
0	1	1	1	1	1	**1**	1	1	**1**
1	0	0	0	0	1	**0**	1	0	**0**
1	0	0	1	0	1	**0**	1	1	**1**
1	0	1	0	0	0	**0**	1	0	**0**
1	0	1	1	0	1	**0**	1	1	**1**
1	1	0	0	1	1	**1**	1	1	**1**
1	1	0	1	1	1	**1**	1	1	**1**
1	1	1	0	1	0	**0**	1	1	**1**
1	1	1	1	1	1	**1**	1	1	**1**

From columns 7 and 10 we conclude that there is logical consequence (but no equivalence — 7, 10, 12 i 15).

The Resolution Method

1. Problem:

$$\Delta \models H$$

2. From Deduction Theorem 2:

$$\Delta \cup \neg H$$

   should be unsatisfiable.

3. Transform $\Delta \cup \neg H$ to CNF.

4. Using the RR derive an empty formula $\bot$.

Example:

1. Problem:

$$(p \Rightarrow q) \wedge (r \Rightarrow s) \models (p \vee r) \Rightarrow (q \vee s)$$

2. From Deduction Theorem 2 — show that:

$$[(p \Rightarrow q) \wedge (r \Rightarrow s)] \cup \neg[(p \vee r) \Rightarrow (q \vee s)]$$

   is unsatisfiable.

3. After transformation to CNF we have:

$$\{\neg p \vee q, \neg r \vee s, p \vee r, \neg q, \neg s\}$$

4. Derive $\bot$.

Dual Resolution Method

1. Problem:

$$\Delta \models H$$

2. From Deduction Theorem 1 show that:

$$\Delta \Rightarrow H$$

   is a tautology.

3. Transform $\Delta \Rightarrow H$ to DNF.

4. Using the DRR derive an empty formula — the always true one $\top$.

Example:

1. Problem:

$$(p \Rightarrow q) \wedge (r \Rightarrow s) \models (p \vee r) \Rightarrow (q \vee s)$$

2. From Deduction Theorem 1 show that:

$$[(p \Rightarrow q) \wedge (r \Rightarrow s)] \Rightarrow [(p \vee r) \Rightarrow (q \vee s)]$$

   is a tautology.

3. After transformation to DNF we have:

$$\{p \wedge \neg q, r \wedge \neg s, \neg p \wedge \neg r, q, s\}$$

4. Using the DRR derive an empty formula — the always true one $\top$.

Example of Resolution Derivation

A – signal from process,

P – signal added to a queue,

B – signal blocked by process,

D – signal received by process,

S – state of the process saved,

M – signal mask read,

H – signal management procedure activated,

N – procedure executed in normal mode,

R – process restart from context,

I – process must re-create context.

Rules — axiomatization:

$A \longrightarrow P$,

$P \wedge \neg B \longrightarrow D$,

$D \longrightarrow S \wedge M \wedge H$,

$H \wedge N \longrightarrow R$,

$H \wedge \neg R \longrightarrow I$,

Facts:

A, $\neg B$, $\neg R$.

Application of RR to CNF:

$\{\neg A \vee P, \neg P \vee B \vee D, \neg D \vee S, \neg D \vee M, \neg D \vee H, \neg H \vee \neg N \vee R, \neg H \vee R \vee I, A, \neg B, \neg R\}$

Conclusions

P, D, S, M, H, I, $\neg N$.

Inference step; derivation

Step of inference: single application of RR.

Example:

Application of RR:

$$\frac{\phi \vee \neg p, p \vee \psi}{\phi \vee \psi}$$

Notation: $\{\phi \vee \neg p, p \vee \psi\} \vdash_R \phi \vee \psi$

Definicja 22 Derivation *A derivation of ϕ from Δ we call a sequence:*

$$\phi_1, \phi_2 \ldots \phi_k$$

such that:

- *formula ϕ_1 is derivable from Δ (in a single step):*

$$\Delta \vdash \phi_1,$$

- *every next formula is derivable from Δ and the earlier-derived formulas:*

$$\{\Delta, \phi 1, \phi_2, \ldots, \phi_i\} \vdash \phi_{i+1}$$

  for $i = 2, 3, \ldots, k-1$,

- *ϕ is the last formula:*

$$\phi = \phi_k$$

Notation: $\Delta \vdash \phi$, and ϕ is called *derivable from* Δ.

Set of Logical Consequences Cn

Definicja 23 *Let Δ be set of formulas. The set of logical consequences is:*

$$Cn(\Delta) = \{\phi\colon \Delta \models \phi\}$$

where every ϕ contains (only) propositional symbols of Δ.

Lemat 7 **Properties of Cn** *There are:*

- $\Delta \subseteq Cn(\Delta)$,
- *monotonicity — if $\Delta_1 \subseteq \Delta_2$, then:*

$$Cn(\Delta_1) \subseteq Cn(\Delta_2)$$

- $Cn(Cn(\Delta)) = Cn(\Delta)$ *(the so-called fixed point).*

Constructive Theorem Proving: The Fitch System

- AND Introduction (AI):

$$\frac{\phi_1, \ldots, \phi_n}{\phi_1 \wedge \ldots \wedge \phi_n}$$

- AND Elimination (AE):

$$\frac{\phi_1 \wedge \ldots \wedge \phi_n}{\phi_i}$$

- OR Introduction (OI):

$$\frac{\phi_i}{\phi_1 \vee \ldots \vee \phi_n}$$

- OR Elimination (OE):

$$\frac{\phi_1 \vee \ldots \vee \phi_n, \phi_1 \Rightarrow \psi, \ldots \phi_n \Rightarrow \psi}{\psi}$$

- Negation Introduction (NI):

$$\frac{\phi \Rightarrow \psi, \phi \Rightarrow \neg\psi}{\neg\phi}$$

- Negation Elimination (NE):

$$\frac{\neg\neg\phi}{\phi}$$

- Implication Introduction (II):

$$\frac{\phi \vdash \psi}{\phi \Rightarrow \psi}$$

- Implication Elimination (IE):

$$\frac{\phi, \phi \Rightarrow \psi}{\psi}$$

- Equivalence Introduction (EI),
- Equivalence Elimination (EE)

SAT by Example: Unicorn

Given the following Knowledge Base (KB):

- If the unicorn is mythical, then it is immortal
- If the unicorn is not mythical, then it is a mortal mammal
- If the unicorn is either immortal or a mammal, then it is horned
- The unicorn is magical if it is horned

answer the following questions:

- Is the unicorn mythical? (M)
- Is it magical? (G)
- Is it horned? (H)

In terms of logic:

$$\text{KB} \models G, H, M$$

$$\text{KB} \vdash G, H, M$$

Unicorn - Logical Model

Definition of propositional variables:

- M: The unicorn is mythical
- I: The unicorn is immortal
- L: The unicorn is mammal
- H: The unicorn is horned
- G: The unicorn is magical

Building a **Logical Model** for the puzzle:

- If the unicorn is mythical, then it is immortal:

$$M \longrightarrow I$$

- If the unicorn is not mythical, then it is a mortal mammal:

$$\neg M \longrightarrow (\neg I \wedge L)$$

- If the unicorn is either immortal or a mammal, then it is horned:

$$(I \vee L) \longrightarrow H$$

- The unicorn is magical if it is horned:

$$H \longrightarrow G$$

Resulting Boolean formula (the Knowledge Base):

$$(M \longrightarrow I) \wedge (\neg M \longrightarrow (\neg I \wedge L)) \wedge ((I \vee L) \longrightarrow H) \wedge (H \longrightarrow G)$$

A Solution

$$(M \longrightarrow I) \equiv (\neg M \vee I)$$

$$(\neg M \longrightarrow (\neg I \wedge L)) \equiv (M \vee (\neg I \wedge L))$$

$$(M \vee (\neg I \wedge L)) \equiv ((M \vee \neg I) \wedge (M \vee L))$$

$$\frac{\neg M \vee I, M \vee L}{I \vee L}$$

$$\frac{I \vee L, (I \vee L) \longrightarrow H}{H}$$

$$\frac{H, H \longrightarrow G}{G}$$

So we have:

$$\mathrm{KB} \vdash H \wedge G$$

Questions:

- What about M (mythical), I (immortal) and L (mammal)?
- What are the exact models? What combinations are admissible?
- How many models do we have?

SAT: Transformation to CNF and Encoding

Introducing enumeration:

1. M: The unicorn is mythical
2. I: The unicorn is immortal
3. L: The unicorn is mammal
4. H: The unicorn is horned
5. G: The unicorn is magical

Building a **Logical Model** for the puzzle:

- If the unicorn is mythical, then it is immortal:

$$M \longrightarrow I$$

- If the unicorn is not mythical, then it is a mortal mammal:

$$\neg M \longrightarrow (\neg I \wedge L)$$

- If the unicorn is either immortal or a mammal, then it is horned:

$$(I \vee L) \longrightarrow H$$

- The unicorn is magical if it is horned:

$$H \longrightarrow G$$

CNF and Encoded File

Resulting CNF:

$$\{\neg M \vee I, M \vee \neg I, M \vee L, \neg I \vee H, \neg L \vee H, \neg H \vee G\}$$

```
-1  2
 1 -2
 1      3
   -2       4
       -3   4
           -4    5
```

Input file in the DIMACS format:

```
p cnf 5 6
-1 2 0
1 -2 0
1 3 0
-2 4 0
-3 4 0
-4 5 0
```

Using Minisat

Page: `http://minisat.se/`

Online: `http://www.msoos.org/2013/09/minisat-in-your-browser/`

Manual: `http://www.dwheeler.com/essays/minisat-user-guide.html`

How to get ALL solutions?

Extra problem

Assumptions:

A1. There are 3 houses in a row

A2. The houses are numbered 1, 2 and 3, from left to right

A3. Each house has one of the colors Blue, Green or White

A4. Each house is inhabited by one person with one of the nationalities: Dutch, German and Italian

A5. Each person drinks (exactly one) of the following beverages: Coffee, Tea and Water

Conditions (constraints):

C1 The third house is green

C2 There is one house between the house of the person drinking coffee and the blue house

C3 The person drinking water lives in the blue house

C4 The Italian lives to the left of the coffee drinking person

C5 The German lives in house two

Query:

Who lives in the 1st house? What does the Dutch drink?

References

1. Mordechai Ben-Ari: Mathematical Logic for Computer Science (Logika matematyczna w informatyce). Springer-Verlag, London, 2001 (WN-T, Warszawa, 2005).
2. Kenneth A. Ross i Charles R. B. Wright: Discrete Mathematics (Matematyka dyskretna). WN PWN, 2013.
3. Antoni Ligęza: Logical Foundations for Rule-Based Systems. Springer-Verlag, Berlin, 2006. Wydawnictwo AGH, Kraków, 2005.
4. Michael R. Genesereth, Nils J. Nilsson: Logical Foundations of Artificial Intelligence. Morgan Kaufmann Publishers, Inc., Los Altos, California, 1987.
5. Zbigniew Huzar: Elementy logiki dla informatyków. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 2007.
6. Stuart Russell, Peter Norvig: Artificial Intelligence. A Modern Approach. Pearson, 2010.
7. Marek Wójcik: Zasada rezolucji. Metoda automatycznego wnioskowania. PWN, Warszawa, 1991.
8. C. L. Chang and R. C. T. Lee: Symbolic Logic and Mechanical Theorem Proving. Academic Press, 1973.
9. Ronald J. Brachman and Hector J. Levesque: Knowledge Representation and Reasoning. Morgan Kaufmann, 2004.
10. Frank van Harmelen, Vladimir Lifschitz, Bruce Porter (Eds.): *Handbook of Knowledge Representation*. Elsevier B.V., Amsterdam, 2008. `http://ii.fmph.uniba.sk/~sefranek/kri/handbook/`

On the Net; Stanford/Coursera

Coursera on-line/Stanford:

`https://www.coursera.org/course/intrologic`

1. Wikipedia-pl: `http://pl.wikipedia.org/wiki/Logika_matematyczna`
2. Wikipedia-en: `http://en.wikipedia.org/wiki/Logic`
3. AI-Lab-Prolog: `http://ai.ia.agh.edu.pl/wiki/pl:prolog:prolog_lab`
4. EIS-KRR: `http://ai.ia.agh.edu.pl/wiki/pl:dydaktyka:krr:start`
5. ALI-home: `home.agh.edu.pl/~ligeza`
6. David Poole and Allen Mackworth: Artificial Intelligence. Foundations of Computational Agents. `http://artint.info/`
7. Ulf Nilsson and Jan Maluszynski: Logic, Programming and Prolog. `http://www.ida.liu.se/~ulfni/lpp/`
8. Inês Lynce: Boolean SATisfiability. Materiały z kursu ACAI, Lille, October 2015.