Logique classique

Cours 3 : Raisonnement en logique propositionnelle

Odile PAPINI

POLYTECH
Université d'Aix-Marseille
odile.papini@univ-amu.fr
http://odile.papini.perso.luminy.univ-amu.fr/sources/LOG.html

Plan du cours

Intodution

logique et informatique

- représentation des connaissances
- raisonnement sur les connaissances
- mettre en œuvre le raisonnement
- implantation informatique

Differentes formes de raisonnement

3 formes de raisonnement :

raisonnement déductif

raisonnement abductif

raisonnement inductif

Differentes formes de raisonnement

Exemple

$$\text{si} \quad \models A, \quad \text{et} \quad \models A \to B \quad \text{alors} \quad \models B$$

raisonnement déductif :

connaissant $\models A$ et $\models A \to B$ on peut inférer $\models B$

raisonnement abductif :

connaissant $\models A \to B$ et $\models B$ on peut inférer $\models A$

raisonnement inductif :

connaissant $\models A$ et $\models B$ on peut inférer $\models A \to B$

Differentes formes de raisonnement

Raisonnement déductif :

lois générales $H_1, H_2, \cdots, H_n \models C$

hypothèses $H_1, H_2, \cdots, H_n$

conclusion C

Differentes formes de raisonnement

définition (rappel)

$\mathcal{F}$ un ensemble de formules de $\mathcal{L}$ et I une interprétation,

$\mathcal{F}$ est insatisfaisable ou incohérent ssi $\forall I$ telle que $\exists F \in \mathcal{F}$, $I(F) = 0$.

proposition

$F \in \mathcal{L}$, $\quad \mathcal{F} \models F$ ssi $\mathcal{F} \cup \{\neg F\}$ est insatisfaisable ou incohérent.

on ramène le problème de conséquence logique à celui de la cohérence ou de la satisfaisabilité (SAT).

réfutation : on cherche à montrer l'incohérence (UNSAT)

Méthodes énumératives

Tables de vérité

$F \in \mathcal{L}$ avec $p_1, p_2, \cdots, p_n \in \mathcal{P}$ apparaissant dans F

p_1	p_2	p_3	$\cdots$	p_n	F
0	0	0	$\cdots$	0	
0	0	0	$\cdots$	1	
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	
1	1	1	$\cdots$	1	

limites

pour n variables 2^n lignes : complexité $O(2^n)$

Tables de vérité

Exemple

Table de vérité pour

$$F = ((p \rightarrow q) \land p) \rightarrow q$$

p	q	$p \rightarrow q$	$(p \rightarrow q) \land p$	F
0	0			
0	1			
1	0			
1	1			

Tables de vérité

Exercices

- $S = \{p \vee q,\ p \vee r,\ \neg q \vee \neg r,\ \neg p\}$ est-il satisfaisable ?
- En utilisant les tables de vérité, montrer que

$$\{p \rightarrow r,\ q \rightarrow r\} \quad \models \quad (p \vee q) \rightarrow r$$

Méthodes énumératives

Arbres sémantiques

S : ensemble de clauses correspondant à la forme CNF de $F \in \mathcal{L}$, et $p_1, p_2, \cdots, p_n \in \mathcal{P}$ apparaissant dans S

principe : construction d'un arbre

- **chaque arc est étiqueté par un littéral p_i ou $\neg p_i$**
- **les littéraux étiquetant les deux arcs issus d'un même sommet sont opposés**
- **aucune branche ne comporte plus d'une occurrence de chaque atome**

Méthodes énumératives

Arbres sémantiques

- un arbre est **complet** ssi chaque feuille de l'arbre correspond à une interprétation de $p_1, p_2, \cdots, p_n$.
- une branche est **fermée** ssi il existe un nœud n tel que l'interprétation partielle correspondante est un contre-modèle d'une clause de S et tel que les interprétations partielles de tout noeud ancêtre de n ne sont pas des contre-modèles d'une clause de S.
- un arbre sémantique est **fermée** ssi toutes ses branches sont fermées.

Méthodes énumératives

Arbres sémantiques

proposition :

S un ensemble de clauses est incohérent ssi pour tout arbre sémantique complet correspondant à S, il existe une arbre fini fermé correspondant à S.

proposition :

S un ensemble de clauses est incohérent ssi il existe un sous-ensemble de clauses S', $S' \subset S$ incohérent.

limites

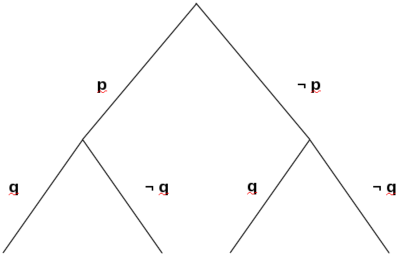
L'arbre sémantique contient 2^n feuilles : complexité $O(2^n)$

Arbres sémantiques

Exemple

Arbre sémantique pour

$$F = ((p \rightarrow q) \wedge p) \rightarrow q$$



Arbres sémantiques

Exercices

- $S = \{p \vee q,\ p \vee r,\ \neg q \vee \neg r,\ \neg p\}$ est-il satisfaisable ?
- En utilisant les les arbres sémantiques, montrer que

$$\{p \rightarrow r,\ q \rightarrow r\} \quad \models \quad (p \vee q) \rightarrow r$$

Méthodes énumératives

Méthode de Quine

amélioration de la méthode des arbres sémantiques : élagage

principe :

coupure dans l'arbre sémantique sur un nœud si toutes les branches issues de ce nœud donnent la même valeur de vérité

S : ensemble de clauses (CNF), p : proposition, $S = S_p \cup S_{\neg p} \cup S''$

S_p : ensemble des clauses où p apparaît

$S_{\neg p}$: ensemble des clauses où $\neg p$ apparaît, $S'' = S \backslash (S_p \cup S_{\neg p})$.

S'_p : ensemble des clauses de S_p privées de p

$S'_{\neg p}$: ensemble des clauses $S_{\neg p}$ privées de $\neg p$.

proposition :

S incohérent ssi $S'_p \cup S''$ et $S'_{\neg p} \cup S''$ incohérents.

Méthodes énumératives

Algorithme de Quine

quine(S : un ensemble de clauses) : (coherent)
données : S_p, $S_{\neg p}$, S", S'_p, $S'_{\neg p}$, *coherent*
début
si $S = \emptyset$ **alors** $coherent \leftarrow vrai$
si $S = \{\square\}$ **alors** $coherent \leftarrow faux$
sinon
choisir un atome $p \in S$
calculer S_p, $S_{\neg p}$ et $S" = S \backslash (S_p \cup S_{\neg p})$
calculer S'_p et $S'_{\neg p}$
quine($S'_p \cup S"$)
quine($S'_{\neg p} \cup S"$)
finsi
finsi
fin

Méthodes énumératives

Méthode de Davis et Putnam (DPLL)

DPLL (Davis, Putnam, Logemann, Loveland) : amélioration de la méthode de Quine

principe :

Tirer parti de la forme CNF pour sélectionner au mieux les propositions afin d'éviter la construction de branches inutiles.

- clause monolittérale : clause contenant seulement p, (respectivement $\neg p$).
- présence d'un seul des littéraux p ou $\neg p$

Méthodes énumératives

Méthode de Davis et Putnam

Une proposition p est sélectionnée en priorité si :

- S contient une clause monolittérale,
  S'_p (respectivement $S'_{\neg p}$ contient la clause vide, donc $S'_p \cup S"$ (respectivement $S'_{\neg p} \cup S"$) est incohérent,
  la recherche d'incohérence se ramène seulement à celle de $S'_{\neg p} \cup S"$ (respectivement $S'_p \cup S"$).
- Si un seul des littéraux p ou $\neg p$ apparaît dans S.
  S_p ou $S_{\neg p}$ est vide,
  la recherche d'incohérence se ramène seulement à celle de $S'_{\neg p} \cup S"$ ou à celle de $S'_p \cup S"$.

Méthodes énumératives

Méthode de Davis et Putnam : Exemple

$$S = \{\neg a \vee b,\ a,\ \neg b\}$$

choix du littéral a :
$S_a = \{a\}$, $S_{\neg a} = \{\neg a \vee b\}$ et $S" = S \backslash (S_a \cup S_{\neg a}) = \{\neg b\}$
$S'_a = \{\square\}$ et $S'_{\neg a} = \{b\}$
S incohérent ssi $S'_a \cup S" = \{\square, \neg b\}$ et $S'_{\neg a} \cup S" = \{b, \neg b\}$ le sont aussi.
$S'_a \cup S" = \{\square, \neg b\}$: il est incohérent, soit $S'_{\neg a} \cup S" = \{b, \neg b\} = T$
choix du littéral b :
$T_b = \{b\}$, $T_{\neg b} = \{\neg b\}$ et $T" = \emptyset$
$T'_b = \{\square\}$ et $T'_b \cup T" = \{\square\}$: incohérent
$T'_{\neg b} = \{\square\}$ et $T'_{\neg b} \cup T" = \{\square\}$: incohérent
T incohérent ssi $T'_b \cup T"$ et $T'_{\neg b} \cup T"$ le sont aussi

Méthode de Davis et Putnam

Exercice

- $S = \{p \vee q,\ p \vee r,\ \neg q \vee \neg r,\ \neg p\}$ est-il satisfaisable ?
- En utilisant la méthode de Davis et Putnam, montrer que

$$\{p \to r,\ q \to r\} \quad \models \quad (p \vee q) \to r$$

Méthodes énumératives

problème SAT

Etant donne une formule booléenne mise sous forme CNF , existe t-il une affectation de valeurs (0 ou 1) des variables rendant la formule vraie ?

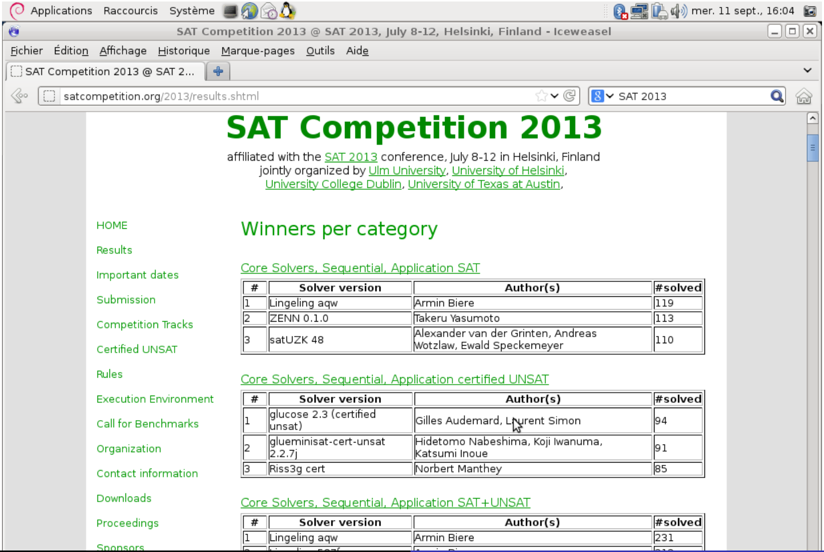
En informatique théorique : complexité
SAT est un problème NP-Complet de référence [Cook 1971]

En pratique : solveurs SAT très efficaces pour résoudre des problèmes combinatoires

- Posit, Satz, zchaff, LySat,
- MiniSat, Sat4J, glucose, Satzilla $\cdots$

http ://www.SATlive.org
http ://www.satcompetition.org/

SAT : compétition SAT 2013



Applications Raccourcis Système mer. 11 sept., 16:04

SAT Competition 2013 @ SAT 2013, July 8-12, Helsinki, Finland - Iceweasel

Fichier Édition Affichage Historique Marque-pages Outils Aide

SAT Competition 2013 @ SAT 2...

satcompetition.org/2013/results.shtml

SAT 2013

SAT Competition 2013

affiliated with the SAT 2013 conference, July 8-12 in Helsinki, Finland
jointly organized by Ulm University, University of Helsinki,
University College Dublin, University of Texas at Austin,

HOME
Results
Important dates
Submission
Competition Tracks
Certified UNSAT
Rules
Execution Environment
Call for Benchmarks
Organization
Contact information
Downloads
Proceedings
Sponsors

Winners per category

Core Solvers, Sequential, Application SAT

#	Solver version	Author(s)	#solved
1	Lingeling aqw	Armin Biere	119
2	ZENN 0.1.0	Takeru Yasumoto	113
3	satUZK 48	Alexander van der Grinten, Andreas Wotzlaw, Ewald Speckemeyer	110

Core Solvers, Sequential, Application certified UNSAT

#	Solver version	Author(s)	#solved
1	glucose 2.3 (certified unsat)	Gilles Audemard, Laurent Simon	94
2	glueminisat-cert-unsat 2.2.7j	Hidetomo Nabeshima, Koji Iwanuma, Katsumi Inoue	91
3	Riss3g cert	Norbert Manthey	85

Core Solvers, Sequential, Application SAT+UNSAT

#	Solver version	Author(s)	#solved
1	Lingeling aqw	Armin Biere	231

SAT : compétition SAT 2013

Applications Raccourcis Système mer. 11 sept., 16:07

SAT Competition 2013 @ SAT 2013, July 8-12, Helsinki, Finland - Iceweasel

Fichier Édition Affichage Historique Marque-pages Outils Aide

satcompetition.org/2013/results.shtml

Full Results

Solvers can have one of the following tags:

- disqualified: the solver produced a wrong answer
- not competiting: the solver was not eligible to win a medal in the track

Core Solvers, Sequential, Application SAT

#	Solver version	#solved	% of all	% of VBS	cumulated CPU time	median CPU time
0	VBS	149	99.33	100.0	67390.7993	452.2872
1	Lingeling aqw	119	79.33	79.87	250711.9549	1671.413
2	ZENN 0.1.0	113	75.33	75.84	285593.1346	1903.9542
3	Doug Hains (disqualified) 1.1	111	74.0	74.5	273179.8937	1821.1993
4	satUZK 48	110	73.33	73.83	288204.0953	1921.3606
5	Riss3g 3g	108	72.0	72.48	298632.0382	1990.8803
6	Lingeling 587f	107	71.33	71.81	306967.9938	2046.4533
7	CSHCapplLG	106	70.67	71.14	309761.1326	2065.0742
8	gluebit_lgl 1.0	105	70.0	70.47	291967.1943	1946.448
9	MIPSat	105	70.0	70.47	307557.2278	2050.3815
10	forl nodrup	104	69.33	69.8	316878.728	2112.5249
11	MIPSat	104	69.33	69.8	325119.698	2167.4647
12	glucose 2.3	103	68.67	69.13	306865.187	2045.7679
13	glue_bit 1.0	102	68.0	68.46	291927.8903	1946.1859
14	Solver version43b b	102	68.0	68.46	315139.4093	2100.9294
15	Riss3g	102	68.0	68.46	322501.0507	2150.007
	strangenight					

Résolution en logique propositionnelle

principe de la résolution J. -A. Robinson

Soit A, B, X des formules propositionnelles
On suppose que $A \vee X$ et $B \vee \neg X$ sont cohérentes

- si X est cohérente alors B est cohérente
- $\neg X$ est cohérente alors A est cohérente

dans les 2 cas $A \vee B$ est cohérente

plus formellement :

$$\{A \vee X, B \vee \neg X\} \models A \vee B$$

Résolution en logique propositionnel

proposition (résolution)

S : un ensemble de clauses, c_1, $c_2 \in S$

si l est un littéral tel que :

$c_1 = l_{i_1} \vee l_{i_2} \vee \cdots \vee l_{i_n} \vee l$ et $c_2 = l_{j_1} \vee l_{j_2} \vee \cdots \vee l_{j_m} \vee \neg l$

alors $S \models r$ avec $r = l_{i_1} \vee l_{i_2} \vee \cdots \vee l_{i_n} \vee l_{j_1} \vee l_{j_2} \vee \cdots \vee l_{j_m}$

On appelle r la **résolvante**

Résolution en logique propositionnelle

proposition :

S : une ensemble de clauses (CNF correspondant à $\mathcal{F}$ ensemble de formules propositionnelles) S est incohérent ssi $S \models \square$

Algorithme de résolution

<u>début</u>
<u>tant que</u> $\square \notin S$ <u>faire</u>
choisir l, c_1, c_2 tels que $l \in c_1$ et $\neg l \in c_2$
calculer la résolvante r
remplacer S par $S \cup \{r\}$
<u>fin tant que</u>
<u>fin</u>

Résolution en logique propositionnelle

Exercice

- $S = \{p \vee q,\ p \vee r,\ \neg q \vee \neg r,\ \neg p\}$ est-il satisfaisable ?

- En utilisant la résolution, montrer que

$$\{p \rightarrow r,\ q \rightarrow r\} \quad \models \quad (p \vee q) \rightarrow r$$

propriétés de la résolution

proposition :

S : un ensemble de clauses, S' : ensemble de clauses après résolution.
Si S est satisfaisable alors S' est satisfaisable

proposition :

F : une formule propositionnelle.
Si F a une preuve par résolution alors F est une tautologie

théorème :

F : une formule propositionnelle.
Si F est une tautologie alors F a une preuve par résolution

Résolution en logique propositionnelle

Exercice

$\mathcal{F} = \{a \vee b \rightarrow c,\ c \rightarrow d\}$,
$F = \neg d \rightarrow \neg a$,
$G = d \rightarrow b$.

En utilisant la résolution, est-ce que

- $\mathcal{F} \models F$?
- $\mathcal{F} \models G$?

Graphes de résolution

principe :

résolution graphique par un graphe acyclique orienté (DAG en anglais)

- **chaque noeud est soit une racine soit a deux parents,**
  - **un noeud racine est étiqueté par une clause initiale**
  - **un noeud qui a 2 parents est étiqueté par la résolvante produite à partir des 2 parents**

C_i est un ancêtre de C_j si il existe un chemin direct de C_i à C_j dans le graphe

Résolution en logique propositionnelle

Exercice

$\mathcal{F} = \{a \vee b \rightarrow c,\ c \rightarrow d\}$,
$F = \neg d \rightarrow \neg a$,

Utiliser un graphe de résolution pour répondre à la question

est-ce que $\mathcal{F} \models F$?

Strategies de résolution : Résolution linéaire

principe :

résolution à partir de C_i et C_j ssi dans le graphe de résolution si l'une des clauses est

- **soit une racine du graphe (clause initiale)**
- **soit C_i est un ancêtre de C_j dans le graphe**

Résolution en logique propositionnelle

Exercice

$\mathcal{F} = \{a \vee b \to c,\ c \to d\}$,
$F = \neg d \to \neg a$,

Utiliser une stratégie de résolution linéaire pour répondre à la question est-ce que $\mathcal{F} \models F$?

Strategies de résolution : Résolution monolittérale

clause monolittérale :

clause ne contenant qu'un littéral

principe :

résolution à partir de C_i et C_j ssi l'une des clauses est une clause monolittérale

adaptée pour **les clauses de Horn :** contenant **au plus un littéral positif**

- résolution complète pour les clauses de Horn
- efficacité, compexité en temps : linéaire en taille de la forme clausale
- opération de simplification

Résolution en logique propositionnelle

Exercice

$\mathcal{F} = \{a,\ b,\ e \rightarrow c,\ \neg a \vee \neg b \vee c,\ d,\ \neg d \vee \neg c \vee g,\}$
$F = g$

Utiliser une stratégie de résolution monolittérale pour répondre à la question est-ce que $\mathcal{F} \models F$?