Le démonstrateur de théorèmes MUSCADET

Dominique Pastre
Université René Descartes (Paris 5)

Principales références
et
support de cours du DEA MIASH - 2002

Principales références

MUSCADET : Un système de démonstration automatique de théorèmes utilisant connaissances et métaconnaissances en mathématiques, thèse d'état, Paris VI, 1984

MUSCADET : An automatic theorem proving system using knowledge and metaknowledge in mathematics, *Artificial Intelligence Journal,* vol. 38 n°3, 1989, p. 257-318

Automated theorem proving in mathematics, *Annals on Artificial Intelligence and Mathematics*, vol. 8, n ° 3-4, 1993, 425-447

Le manuel

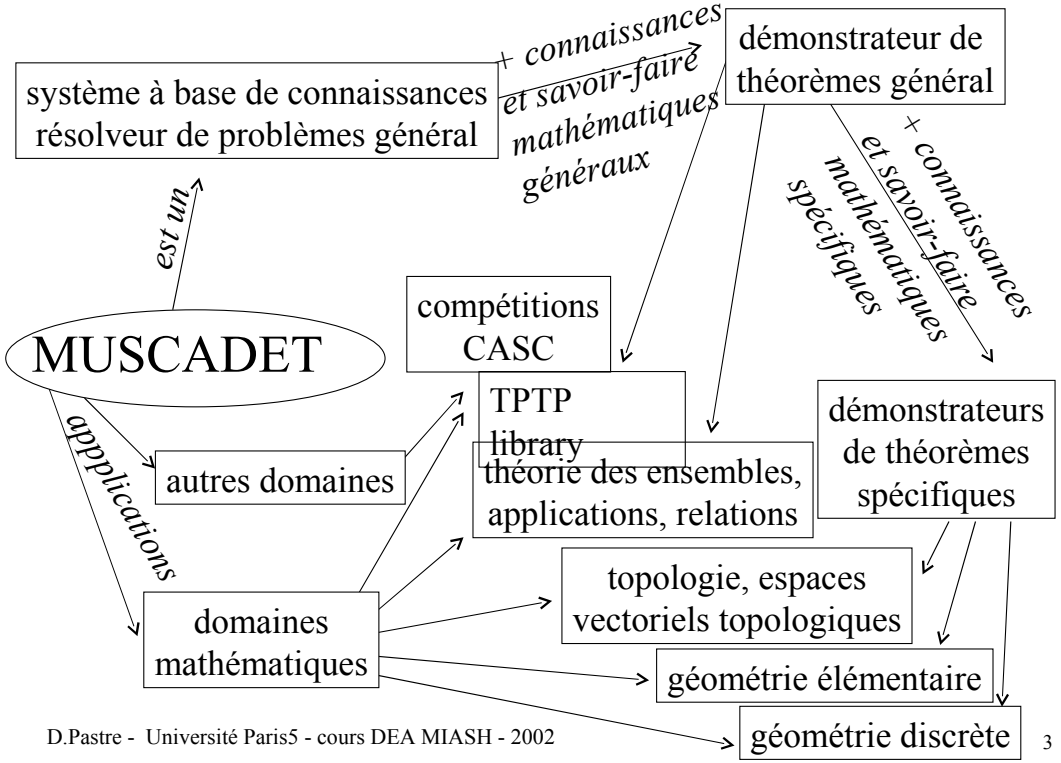
Muscadet version 2.3 : Manuel de l'utilisateur, 2001, 16p (http://www.math-info.univ-paris5.fr/~pastre/muscadet/manuel-fr.ps)

Plus courts avec d'autres exemples

Muscadet2.3 : A knowledge-based theorem prover based on natural deduction, International Joint Conference on Automated Reasoning IJCAR 2001 (Conference on Automated Deduction CADE-JC), 685-689

Implementation of knowledge bases for natural deduction, 8th International Conference on Logic for Programming, Artificial Intelligence and Reasoning, 2nd International Workshop on Implementation of Logics, Cuba, 2001, 49-68

Strong and weak points of the MUSCADET theorem prover, AI Communications, 15, 2002, 147-160 (http://www.math-info.univ-paris5.fr/~pastre/AICom/AIC263.pdf)



Règles d'inférence

Règle "∀" : **pour montrer** ∀x P(x)

(c'est-à-dire **si** la conclusion du théorème en cours de démonstration est ∀x P(x))

prendre x1 quelconque

(c'est-à-dire **créer** un objet quelconque x1)

et **montrer** P(x1)

(c'est-à-dire **remplacer** la conclusion par P(x1))

Règle "⇒" : **pour** montrer A⇒B, supposer A et montrer B

("supposer A" consiste à ajouter A comme nouvelle hypothèse, en la décomposant si c'est une conjonction, et en lui faisant subir certains traitement particuliers dans quelques autres cas)

Règle "∧" : **pour** montrer $A_1 \wedge A_2 \wedge ... \wedge A_n$
démontrer successivement tous les A_i

Règle "stop" : **si** on a obtenu, comme nouvelle hypothèse,
la conclusion que l'on veut démontrer,
alors le théorème est démontré

Règle "stop_∨" : **si** la conclusion est une disjonction $A_1 \vee A_2 \vee ... \vee A_n$
et **si** on a obtenu comme nouvelle hypothèse l'un des A_i
alors le théorème est démontré

Règle "hyp_∨" : **si** on a A∨B, parmi d'autres hypothèses
et **si** on veut montrer C
alors montrer (A⇒C)∧(B⇒C)

Règle "hyp_∃" : **si** on a l'hypothèse ∃x P(x)
et **si** l'on n'a pas encore d'hypothèse de la forme P(y)
alors créer x1 et supposer P(x1)

Règle "concl_∧" : **pour** montrer ∃x P(x),
chercher x tel que P(x)

Plus précisément :

Pour montrer $\exists x\ (C_1(x) \wedge C_2(x) \wedge ... \wedge C_n(x))$

chercher un objet y tel que, compte tenu des hypothèses actuelles, pour tout i compris entre 1 et n, $C_i(y)$ soit vérifié (cas simple) ou démontré (par appel récursif au démonstrateur)

Règle "elim_fonc" : **si** on a l'expression P(F(A))
où F est un symbole fonctionnel
alors créer un objet B
supposer B:F(A)
et remplacer P(F(A)) par P(B)

Règle "def_concl_1" : **si** on veut montrer P(x)
et **si** on connait une définition du prédicat P
alors remplacer P(x) par cette définition

Règle "def_concl_2" : **si** on a l'hypothèse A:F(B)
où F est un symbole fonctionnel
qui a une définition F(B) = {y | P(y)}
et **si** on veut montrer x∈A
alors remplacer la conclusion x∈A par P(x)

Règles relatives aux concepts définis par l'utilisateur

Le *prédicat* P donne des règles de la forme

Règle "Pi" : **si** on a l'hypothèse P(...)
alors ...

Ceci est est fait automatiquement par des métarègles

exemple :

définition formelle :

$$A \subset B \Leftrightarrow \forall x\ (x \in A \Rightarrow x \in B)$$

règle :

Règle "⊂" : **si** on a les hypothèses A⊂B et x∈A
alors ajouter l'hypothèse x∈B

Le *symbole fonctionnel* F donne des règles de la forme

Règle "Fi" : si on a les hypothèses Y:F(...) et X∈Y
alors ...

exemple :

définition formelle :

$\mathcal{P}(A) = \{ X \mid X \subset A \}$

règle :

Règle "$\mathcal{P}$" : si on a les hypothèses B:$\mathcal{P}$(A) et x∈B
alors ajouter l'hypothèse x⊂A

autre exemple

définition formelle : $A \cap B = \{x \mid x \in A \wedge x \in B\}$

règles :

Règle "∩11" : si on a les hypothèses C:A∩B et x∈C
alors ajouter l'hypothèse x∈A

Règle "∩12" : si on a les hypothèses C:A∩B et x∈C
alors ajouter l'hypothèse x∈B

Règle "∩2" : si on a les hypothèses C:A∩B, x∈A et x∈B
alors ajouter l'hypothèse x∈C

Remarque : la règle "∩2" n'est pas de la forme

si on a les hypothèses x∈A et x∈B
alors ajouter l'hypothèse x∈A∩B
qui serait expansive

Démonstration du théorème
$\forall A \forall A' \forall B \forall B' (A \subset A' \wedge B \subset B' \Rightarrow A \cap B \subset A' \cap B')$

règles	objets	hypothèses	conclusion
			$\forall A \forall A' \forall B \forall B' (A \subset A' \wedge B \subset B' \Rightarrow A \cap B \subset A' \cap B')$
∀	a,a',b,b'		$a \subset a' \wedge b \subset b' \Rightarrow a \cap b \subset a' \cap b'$
elim_fonc	c, c'	c:a∩b, c':a'∩b'	$a \subset a' \wedge b \subset b' \Rightarrow c \subset c'$
⇒		a⊂a', b⊂b'	c⊂c'
defconcl_1			$\forall X (X \in c \Rightarrow X \in c')$
∀ et ⇒	x1	x1∈c	x1∈c'
∩11		x1∈a	
∩12		x1∈b	
⊂		x1∈a', x1∈b'	
∩2		x1∈c'	
stop			Théorème démontré

Ensemble des parties de l'intersection de deux ensembles
Théorème à démontrer $\forall A \forall B (\mathcal{P}(A \cap B) =_{ens} \mathcal{P}(A) \cap \mathcal{P}(B))$

Définition de l'intersection

$A \cap B = \{X \mid X \in A \wedge X \in B\}$

Définition de l'ensemble des parties d'un ensemble

$\mathcal{P}(A) = \{ X \mid X \subset A \}$

Définition de l'égalité d'ensembles

$A =_{ens} B \Leftrightarrow A \subset B \wedge B \subset A$

Définition de l'inclusion

$A \subset B \Leftrightarrow \forall X (X \in A \Rightarrow X \in B)$

règles	objets	hypothèses	conclusion
			∀A∀B($\mathcal{P}$(A∩B) =ens $\mathcal{P}$(A)∩$\mathcal{P}$(B))
∀	a, b		$\mathcal{P}$(a∩b) =ens $\mathcal{P}$(a)∩$\mathcal{P}$(b)
elim_fonc	c, pc	c:a∩b, pc:$\mathcal{P}$(c)	
	pa, pb	pa:$\mathcal{P}$(a), pb:$\mathcal{P}$(b)	
	pd	pd:pa∩pb	pc =ens pd
def_concl1			pc ⊂ pd ∧ pd ⊂ pc
∧	donne Théorème 1 et Théorème 2		
Théorème 1			pc ⊂ pd
def_concl1			∀X (X∈ pc ⇒ X∈ pd)
∀	x		x∈ pc ⇒ x∈ pd
⇒		x∈ pc	x∈ pd
$\mathcal{P}$		x ⊂ c	
def_concl2			x ∈ pa ∧ x∈ pb
∧	donne Théorème 11 et Théorème 12		

Théorème 11

règle	objets	hypothèses	conclusion
...	...		x∈ pa
defconcl2			x ⊂ a
defconcl1			∀X (X∈ x ⇒ X∈ a)
∀ et ⇒	t	t∈ x	t∈ a
⊂		t∈ c	
∩11		t∈ a	Théorème 11 démontré

Théorème 12

	...	...	x∈ pb
			...
			Théorème 12 démontré
remontée			Théorème 1 démontré

Théorème 2

règle	objets	hypothèses	conclusion
...	...		pd ⊂ pc
defconcl1			∀X (X∈ pd ⇒ X∈ pc)
∀ et ⇒	x	x∈ pd	x∈ pc
∩1 et 2		x∈ pa, x∈ pb	
$\mathcal{P}$(2 fois)		x ⊂ a, x ⊂ b	
defconcl1			x ⊂ c
(2 fois)			∀X (X∈ x ⇒ X∈ c)
∀ et ⇒	t	t∈ x	t∈ c
⊂(2 fois)		t∈ a, t∈ b	
∩2		t∈ c	
stop			Théorème 2 démontré
remontée			Théorème 0 démontré

Exemple de super-action :"ajouter une hypothèse"

L'appel à cette super-action est récursif

Pour ajouter-hyp H

si H est déjà une hypothèse ou si H est de la forme X=X
alors ne rien faire

si H est de la forme A∧B alors ajouter-hyp A et ajouter-hyp B

si H est de la forme ∀X P ou A⇒B
alors créer des règles locales à ce (sous)-théorème

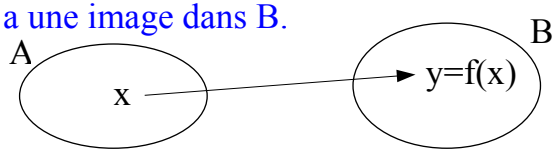
si H est de la forme *pour le seul* Y *tel que* Y:F(X), P(X))
et si il n'y a pas encore d'hypothèse de la forme Y:F(X)
alors créer un nouvel objet Y1 et ajouter-hyp Y1:F(X)

sinon ajouter H comme nouvelle hypothèse

Traitement des hypothèses existentielles

Créer systématiquement des objets pourrait être expansif. Le traitement des hypothèses existentielles a donc une faible priorité et celles-ci sont traitées une par une, dans l'ordre où elles sont apparues, et toutes les autres règles sont réessayées après chaque ajout.

Exemple : Si f est une application de A dans B, alors tout élément de A a une image dans B.

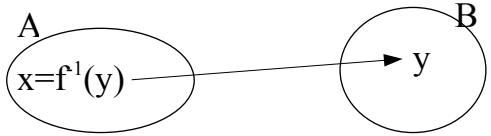


Cas particulier si f est une application de A dans A :

$a \rightarrow a_1=f(a) \rightarrow a_2=f(a_1) \rightarrow a_3=f(a_2) \rightarrow \ldots$

On déduit tout ce qu'il est possible de déduire de l'hypothèse $a_i=f(a_{i-1})$ avant de créer a_{i+1}.

Si f est de plus surjective, tout élément de B a un antécédant dans A.



Cas particulier si f est une application surjective de A dans A :

$\ldots \rightarrow a_4=f^{-1}(a_2) \rightarrow a_2=f^{-1}(a) \rightarrow a \rightarrow a_1=f(a) \rightarrow a_3=f(a_1) \rightarrow \ldots$

On crée alternativement une image et un antécédant.

De plus s'il y a plusieurs applications, les images et antécédents sont créés alternativement pour toutes les applications.

Changement de l'ordre des règles

On doit donner la priorité aux règles créant les objets les plus spécifiques.

Métarègle : si la règle R est susceptible de créer a tel que P
la règle R' est susceptible de créer b tel que Q
P est plus général que Q

alors R' doit être appliquée avant R

Plus précisément, on a la métarègle suivante, qui en est une restriction suffisante :

si la règle R comporte comme action ajouter-hyp $\exists x \in A\ C$
la règle R' comporte comme action ajouter-hyp $\exists x' \in A\ C'$
C est une conjonction de termes dont l'un est égal à C'
modulo x et x'

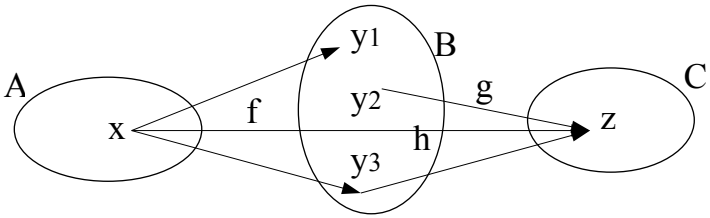
alors échanger R et R'

Exemple

Si f est une application de A dans B, alors tout élément de A a une image dans B.

Si f est surjective de A dans B, tout élément de B a un antécédant dans A.

Si h est la composée (de A dans C) de f, application de A dans B, et de g, application de B dans C, et si z=h(x), alors il existe un élément y de B tel que y=f(x) et z=g(y)



On a alors $y_1=y_3$ et, si g est injective $y_2=y_3$.
Plutôt que de créer y_1, puis y_2 et y_3, il est préférable de créer uniquement y_3 qui a les trois propriétés.

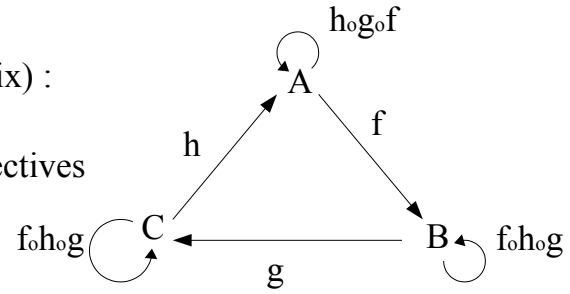
Exemple en Théorie des ensembles

Théorème : Soient f, g, h trois applications de A dans B, B dans C, C dans A. Si parmi les trois applications $h \circ g \circ f$, $g \circ f \circ h$, $f \circ h \circ g$, deux sont injectives (resp. surjectives) et la troisième est surjective (resp. injective), alors f, g et h sont bijectives

Soit (un cas parmi six) :

$h \circ g \circ f$ injective

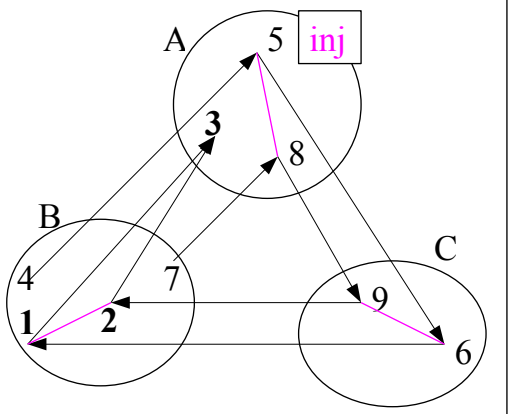
$g \circ f \circ h$ et $f \circ h \circ g$ surjectives



Cas $h \circ g \circ f$ injective, $g \circ f \circ h$ et $f \circ h \circ g$ surjectives (un cas parmi six)

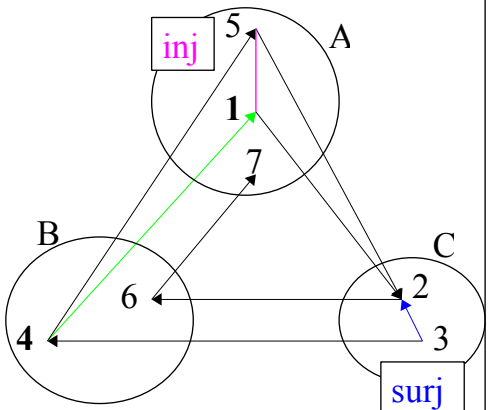
injectivité de h

si "1" et "2" ont même image "3" alors ils sont égaux



surjectivité de h

"4" est un antécédent de "1" car "1" est égal à son image "5"



Espaces vectoriels topologiques (EVT)

Quelques définitions

Un **EVT** est un espace vectoriel dans lequel on a défini une **topologie** et deux applications "+" et "×" **continues**.

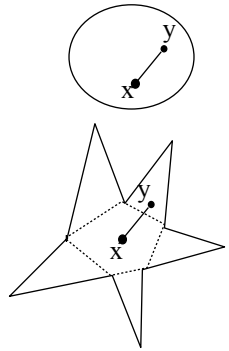
segment $[x\ y] = \{ \lambda*x + \mu*y \mid \lambda+\mu = 1\}$

A **convexe** $\Leftrightarrow \forall x \in A\ \forall y \in A\ [x\ y] \subset A$

A **étoilé** par rapport à x $\Leftrightarrow \forall y \in A\ [x\ y] \subset A$

A **symétrique** $\Leftrightarrow \forall x \in A\ -x \in A$

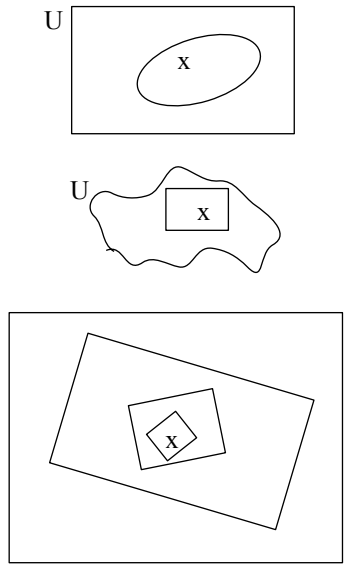
noyau(A) = $\{x \in A \mid \forall y \in A\ [x\ y] \subset A\}$



U est un **voisinage** de x s'il contient un ouvert contenant x.

Une **base de voisinages** est une famille d'ensembles telle que quel que soit le voisinage U d'un point x, il existe un ensemble de la famille, inclus dans U, contenant x.

On peut se rapprocher de plus en plus de x en ne prenant **que** des voisinages de la base.



Pour démontrer ces théorèmes, MUSCADET dispose de *connaissances* sur les espaces topologiques et sur les espaces vectoriels, élaborées par *auto-observation* et *observation* de mathématiciens ayant accepté de raisonner "à voix haute" puis ayant justifié leurs choix a posteriori.

L'importance des *voisinages* et des *bases de voisinages* a été mise en évidence, permettant de travailler avec des voisinages particuliers permettant de déduire les propriétés cherchées.

Propriété connue : une union d'ouverts (resp. voisinages) est un ouvert (resp. voisinage)
Savoir-faire associé : pour montrer qu'un ensemble est ouvert (un voisinage), on peut essayer de le décomposer en une union d'ouverts (de voisinages)

Exemples de propriétés et savoir-faire

Propriété connue : l'image réciproque d'un ouvert (resp. un voisinage) par une application continue est un ouvert (resp. un voisinage)

Savoir-faire associé : pour montrer qu'un ensemble est un ouvert, chercher un ouvert dont il soit l'image par une application continue (+ ou × pour les EVT)

Propriété connue : si on a deux bases de voisinages $\mathcal{V}$ et $\mathcal{W}$ dans les espaces E et F, l'ensemble $\{V\times W \mid V\in \mathcal{V} \text{ et } W\in \mathcal{W}\}$ est une base de voisinages de l'espace E×F

Savoir-faire associé : si U est un voisinage du point (x,y), créer V et W voisinages respectivement de x et y tels que U = V×W

Cas particulier (et prioritaire) pour les points de la "diagonale" : si U est un voisinage de (x,x), créer V voisinage de x tel que U = V×V

Voisinages dans $\mathbb{R}$: les intervalles ouverts symétriques par rapport à $x\in\mathbb{R}$ constituent une base de voisinages de x

Cas particulier, voisinages de 0 : les intervalles]-a,a[forment une base de voisinages de 0

Théorème 1 : Si U est un ouvert, alors son translaté a+U est un ouvert.

Démonstration :
Soit fa l'application unaire qui à x associe a+x, c'est une projection par rapport à la deuxième variable de l'application binaire "+".Cette application est continue (connaissance sur les ET). Elle est aussi bijective et son inverse est l'application f_b qui à x associe b+x avec b=-a (connaissances sur les EV).

On cherche une application g et un ouvert X tel que $a+U=g^{-1}(X)$.

$g=f_b$ et X=U conviennent

MUSCADET *démontre ce théorème après 72 applications de règles en l'ayant décomposé en 3 sous-théorèmes.*

Théorème 2 : Si U est un ouvert, alors son homothétique λ*U est un ouvert.

Démonstration analogue

Théorème A. Si U est un voisinage de l'origine O, alors il existe un voisinage V de O, inclus dans U, étoilé et symétrique par rapport à O.

Démonstration

1. appliquer la continuité de "×" en O = 0×O
   il existe un voisinage W de O et un intervalle]-α α[de R
   tels que]-α α[× W ⊂ U
   V =]-α α[× W convient
2. avoir l'idée de l'essayer
3. vérifier qu'il convient, il est
   - voisinage de O
   - inclus dans U
   - étoilé par rapport à O
   - symétrique

Pour le point 1.

- décomposer V en une union V = ∪{ β×W | β∈]-α α[}
- appliquer deux théorèmes "connus":
  - l'homothétique d'un voisinage est un voisinage
    (c'est le Théorème 2)
  - une union de voisinages est un voisinage

Pour les points 3. et 4.

- connaissances sur les EV, par exemple λ×(β×t) = (λ.β)×t

MUSCADET *démontre ce théorème après 305 applications de règles en l'ayant décomposé en 14 sous-théorèmes.*

Théorème B. Si U est un voisinage de l'origine O, alors il existe un voisinage V de O, inclus dans U, étoilé et symétrique par rapport à O et tels que x∈ V et y∈ V impliquent [x y]⊂ U

Démonstration

1. appliquer la continuité de + en O = O+O
   il existe un voisinage W de O tel que W+W ⊂ U
2. appliquer le théorème A à W, ce qui donne V, voisinage de O, inclus dans V, étoilé et symétrique
3. il reste à vérifier la dernière propriété
   x∈ V∧ y∈ V ⇒ [x y]⊂ U
   pour déduire que V convient

Muscadet démontre ce théorème après 422 applications de règles en l'ayant décomposé en 20 sous-théorèmes.

The TPTP Problem Library (Geoff Sutcliffe et Christian Suttner)

http://www.cs.miami.edu/~tptp

base internationale de problèmes dans les domaines suivants

en gras : problèmes énoncés comme formules du **premier ordre** *et* ensembles de clauses
les autres : uniquement sous forme de clauses

Logique	Logique combinatoire **Calcul logique**	Modèles de Henkin
Mathématiques	**Théorie des ensembles (SET)** Théorie des graphes **Algèbre : Groupes** + (Boole, Robbins, Distr, Treillis, Anneaux, Algèbre générale) Théorie des nombres	Topologie Analyse **Géométrie** Théorie des corps Théorie des catégories

Informatique	**Informatique théorique (COM)** Représentation des connaissances **Traitement du langage naturel**	**Planification** **Conception de programmes** **Vérification de programmes**
Engineering	Conception de circuits	Vérification de circuits
Sciences sociales	**Management (MGT)**	
Autres	**Syntaxique (SYN)**	**Puzzles**

Exemples de théorèmes du domaine **SET**

Il n'existe pas d'ensemble formé des éléments qui n'appartiennent pas à eux-mêmes.

$$\neg\ \exists X\ \forall Y\ (Y \in X \Leftrightarrow Y \notin Y)$$

S'il existe un ensemble formé des éléments qui appartiennent à eux-mêmes, alors il n'est pas vrai que tout ensemble a un complément.

$$\exists Y\ \forall X\ (X \in Y \Leftrightarrow X \in X) \Rightarrow \neg \forall X1\ \exists Y1\ \forall Z\ (Z \in Y1 \Leftrightarrow Z \notin X1)$$

Si on a l'axiome *pour tout Z il existe un ensemble formé des éléments qui appartiennent à Z et n'appartiennent pas à eux-mêmes*, alors on a le théorème *il n'existe pas d'ensemble universel.*

axiome : $\forall Z\ \exists Y\ \forall X\ (X \in Y \Leftrightarrow (X \in Z \wedge X \notin X)$

théorème : $\neg\ \exists Z\ \forall X\ (X \in Z)$

Il n'existe pas d'ensemble formé des éléments X tels qu'il n'existe pas de chaine $X \in Z \in X$.

$$\neg\ \exists Y\ \forall X\ (X \in Y \Leftrightarrow \neg\ \exists Z\ (X \in Z \wedge Z \in X)$$

L'égalité d'ensembles est une relation symétrique.

axiome : $\forall X\ \forall Y\ (X =_{ens} Y \Leftrightarrow \forall Z\ (Z \in X \Leftrightarrow Z \in Y)$

théorème : $\forall X\ \forall Y\ (X =_{ens} Y \Leftrightarrow Y =_{ens} X)$

Démonstration du théorème $\neg\ \exists X\ \forall Y\ (Y \in X \Leftrightarrow Y \notin Y)$
($X = \{ Y \mid Y \notin Y\}$ n'est pas un ensemble)

par le principe de Résolution : clauses $\left|\begin{array}{l} Y \notin a \vee Y \notin Y \\ Y \in Y \vee Y \in a \end{array}\right. \rangle\!\!-\ \square$

par Muscadet :

- hyp : $\exists X\ \forall Y\ (Y \in X \Leftrightarrow Y \notin Y)$ | concl : $\neg\ \exists X\ \forall Y\ (Y \in X \Leftrightarrow Y \notin Y)$
- hyp : $\exists X\ \forall Y\ (Y \in X \Leftrightarrow Y \notin Y)$ | concl : *faux*
- objet : a
- règles locales : r0 : si $Y \in a$ et $Y \in Y$ alors faux
  r1 : si $Y \notin Y$ alors $Y \in a$
  r2 : pour tout objet Y on a $Y \in Y \vee Y \in a$
- hyp : $a \in a \vee a \in a$ (règle r2)
  $a \in a$ (règle "$\vee$")
  faux (règle r0) théorème démontré (par l'absurde)