Le Coq et l'Hydre

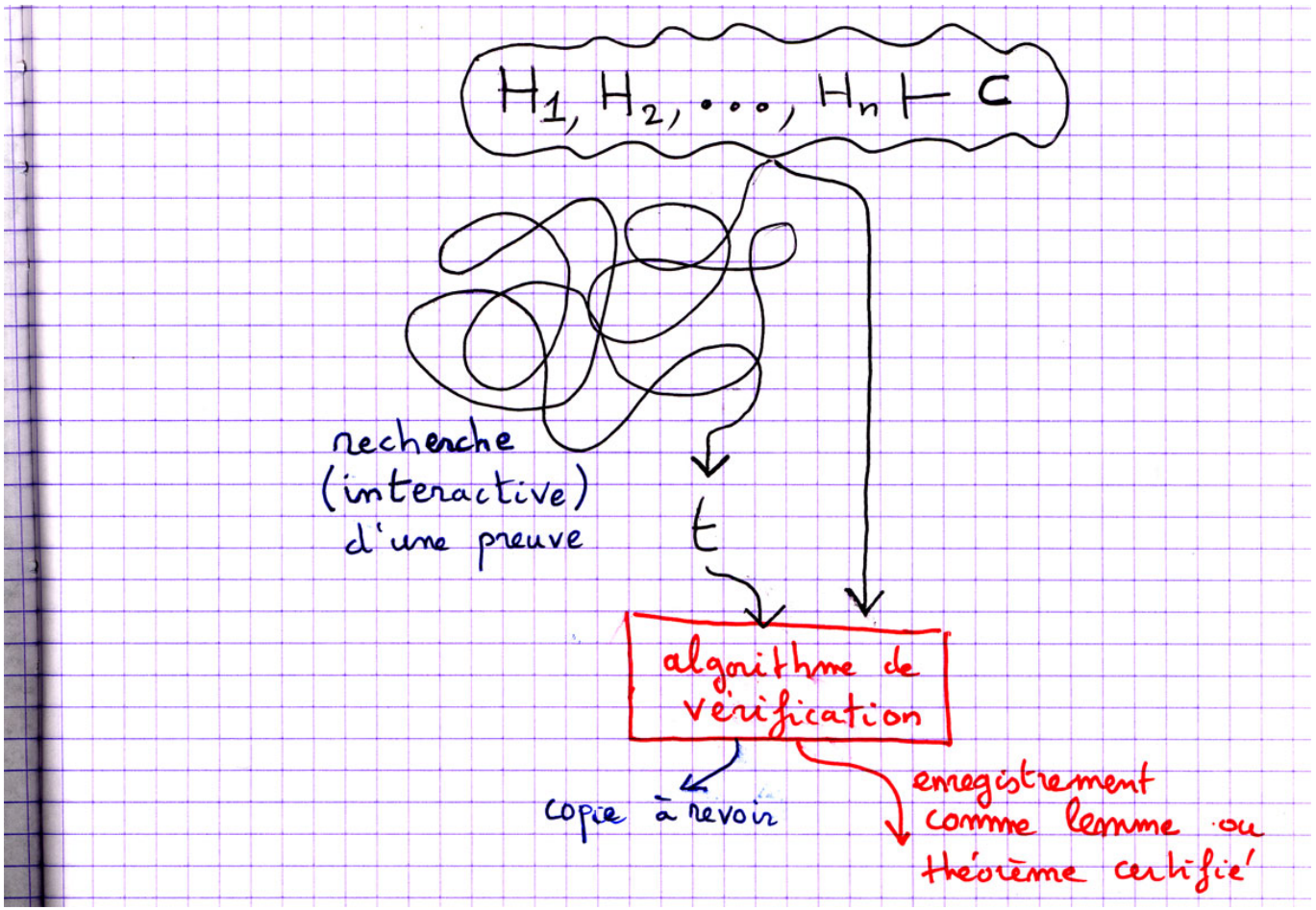
Pierre Castéran, LaBRI (Méthodes Formelles)

18 avril 2007

Assistants de preuves

« En informatique (ou en mathématiques assistées par informatique), un assistant de preuve est un logiciel permettant l'écriture et la vérification de preuves mathématiques, soit sur des théorèmes au sens usuel des mathématiques, soit sur des assertions relatives à l'exécution de programmes informatiques. »(Wikipedia)
Exemples : LCF, HOL, Coq, Isabelle, Nqthm, etc.

Qu'est-ce qu'une preuve formelle ?



Variations sur le schéma précédent

- L'énoncé peut varier au cours de la recherche de preuve :
  - Ajout d'hypothèses non prévues
  - Présence d'inconnues dans l'énoncé, instanciées durant la démonstration :

    `card(E/∼) = n?`

    `...`

    `card(E/∼) = 42`
- Calcul de l'énoncé : *Exemple* : Schémas de preuves par récurrence (structurelle ou bien fondée).

Pourquoi de tels outils ?

- Vérification de programmes
- Preuves (très) calculatoires
- Confiance en une démonstration (souvent incomplète)
- Intérêt pour le discours mathématique
- Intérêt esthétique : l'infini dans une machine finie

Remarques

- Écrire la preuve complète d'un programme ou d'un théorème est un travail assez long, mais à ne faire qu'une fois (en principe.)
- Cet effort peut être justifié dans les cas suivants :
  - Logiciel critique (énergie, transport, toute forme de sécurité)
  - Langages de description : automates, grammaires, langages de programmation, protocoles de télécommunication, protocoles cryptographiques,etc.
  - Démonstrations comportant un très grand nombre de cas à étudier, calculs à effectuer (quatre couleurs, conjecture de Kepler),
  - Intérêt pour la notion de démonstration en soi, enseignement de la logique.

Suites de Goodstein, hydres, et ordinaux

Les suites de Goodstein et problèmes similaires (batailles d'Hydre) présentent les caractéristiques suivantes :

- Propriétés faciles à énoncer, parfois assez contre-intuitives :
- Les preuves de ces propriétés mêlent inférences et calculs

Suites de Goodstein

- On prend un nombre naturel, exprimé sous forme d'une *décomposition héréditaire* en base b.

$$1026 = 2^{2^{2^1+1}+2^1} + 2^1$$

Suites de Goodstein

- On prend un nombre naturel, exprimé sous forme d'une *décomposition héréditaire* en base b.

$$1026 = 2^{2^{2^1+1}+2^1} + 2^1$$

- A chaque étape, on incrémente la base, et on enlève 1.

$$\begin{aligned} & 3^{3^{3^1+1}+3^1} + 3^1 - 1 = 3^{3^{3^1+1}+3^1} + 2 \\ = \quad & 119725151825620197886027400267170471056 83 \end{aligned}$$

Suites de Goodstein

- On prend un nombre naturel, exprimé sous forme d'une *décomposition héréditaire* en base b.

$$1026 = 2^{2^{2^1+1}+2^1} + 2^1$$

- A chaque étape, on incrémente la base, et on enlève 1.

$$\begin{aligned} & 3^{3^{3^1+1}+3^1} + 3^1 - 1 = 3^{3^{3^1+1}+3^1} + 2 \\ = \quad & 119725151825620197886027400267170471056 83 \end{aligned}$$

- On continue : $4^{4^{4^1+1}+4^1} + 1$, $5^{5^{5^1+1}+5^1}$, etc.

Brève Histoire de $G_{8,2}$

$$
\begin{aligned}
8 &= 2^{2+1} \\
80 &= 3^{3+1} - 1 \\
&= 3^3 \times 2 + 3^2 \times 2 + 3 \times 2 + 2 \\
169 &= 4^4 \times 2 + 4^2 \times 2 + 4 \times 2 + 1 \\
6310 &= 5^5 \times 2 + 5^2 \times 2 + 5 \times 2 + 0 \\
93395 &= 6^6 \times 2 + 6^2 \times 2 + 6 + 5 \\
&\dots \\
570623341475 &= 11^{11} \times 2 + 11^2 \times 2 + 11 \\
&\dots \\
4,176 \times 10^{31} &\sim 23^{23} \times 2 + 23^2 \times 2
\end{aligned}
$$

Brève Histoire de $G_{8,2}$ (suite)

$$b^b \times 2 + b^2 \qquad (b = 3 \times 2^{27} - 1)$$

$$\cdots$$

$$b^b \times 2 \qquad (b = N = 3 \times 2^{402653211} - 1)$$

$$b^b + \sum_{i=N}^{0} b^i \times N \qquad (b = N + 1)$$

$$\cdots$$

$$b^b \qquad (b = M)$$

$$\sum_{i=M}^{0} b^i \times M \qquad (b = M + 1)$$

Brève Histoire de $G_{8,2}$ (suite et fin)

$$
\begin{array}{rl}
b & (b = P) \\
P & (b > P) \\
\cdots & \\
4 & \\
3 & \\
2 & \\
1 & \\
0 &
\end{array}
$$

Résultats (Kirby et Paris)

1. La suite de Goodstein issue de 4 en base 2 atteint 0 après $3 \times 2^{402653211} - 1$ étapes,
2. Toute suite de Goodstein finit par atteindre 0,
3. Ce résultat ne peut pas être montré dans l'arithmétique de Peano (premier ordre + récurrence).

Calcul de la longueur de G_4 en *Coq*

Choix d'une structure de donnée appropriée :

$$
\begin{aligned}
4 &= 2^2 \\
&\bullet \quad 3^2 \times 2 + 3 \times 2 + 2 \\
&\bullet \quad 4^2 \times 2 + 4 \times 2 + 1 \\
&\bullet \quad 5^2 \times 2 + 5 \times 2 + 0 \\
&\bullet \quad 6^2 \times 2 + 6 \times 1 + 5 \\
&\quad \ldots
\end{aligned}
$$

Un item de la suite sera représenté par un quadruplet (b, a_2, a_1, a_0)

Calcul de quelques items de la suite (parmi les premiers)

```
nth_item 2 ⇝ (5 2 2 0)
nth_item 3 ⇝ (6 2 1 5)
nth_item 8 ⇝ (11 2 1 0)
nth_item 20 ⇝ (23 2 0 0)
nth_item 21 ⇝ (24 1 23 23)
nth_item 44 ⇝ (47 1 23 0)
nth_item 92 ⇝ (95 1 22 0)
nth_item 188 ⇝ (191 1 21 0)
```

Détection de régularités $(3 \times 2^n - 1, i, j, 0)$.

Preuve d'invariants (par récurrence)

$$
\begin{aligned}
(3 \times 2^n - 1, i, j+1, 0) &\xrightarrow{*} (3 \times 2^{n+1} - 1, i, j, 0) \\
(3 \times 2^n - 1, i, j, 0) &\xrightarrow{*} (3 \times 2^{n+j} - 1, i, 0, 0) \\
(3 \times 2^n - 1, i+1, 0, 0) &\xrightarrow{*} (3 \times 2^{n+3\times 2^n} - 1, i, 0, 0)
\end{aligned}
$$

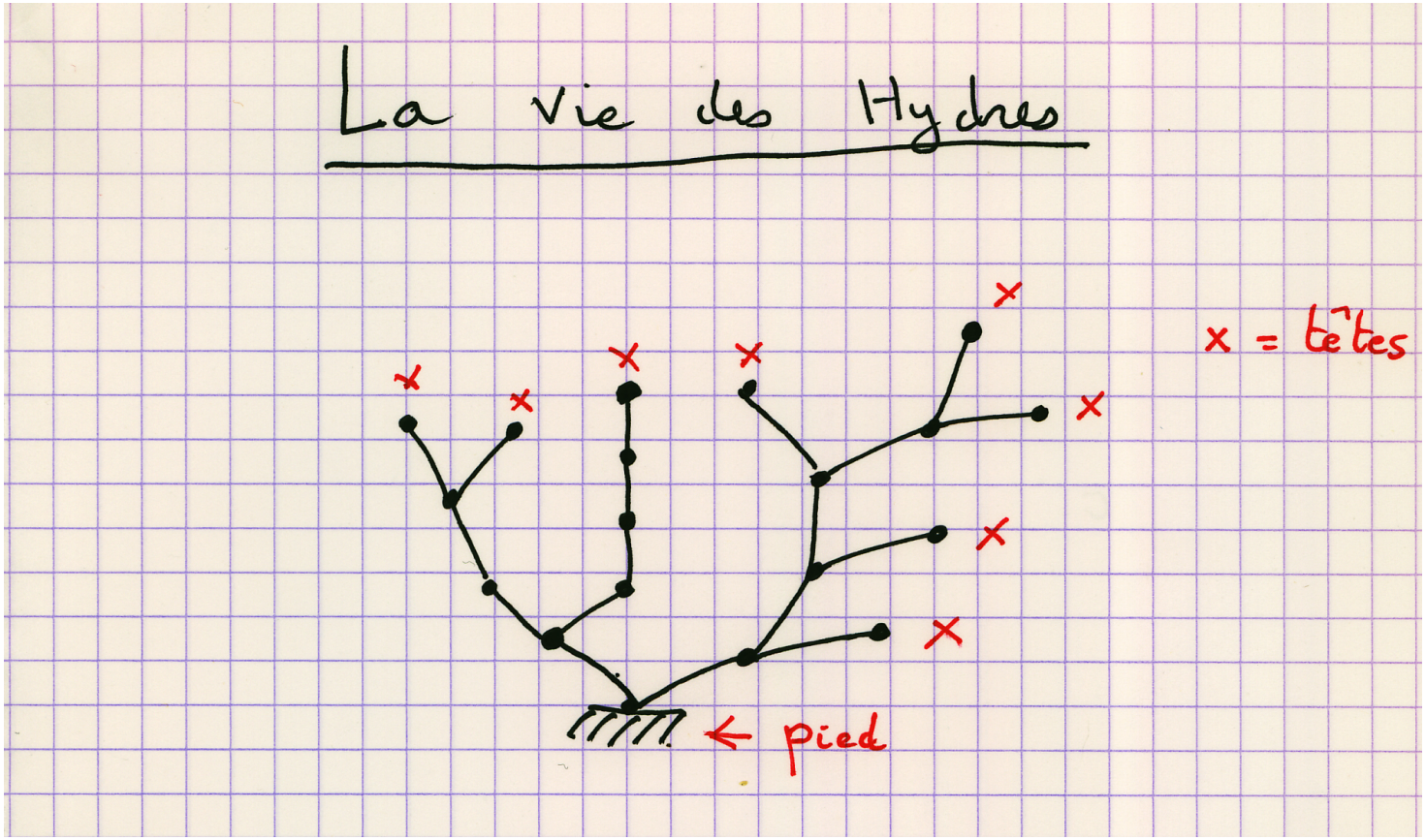
En utilisant ces lemmes et quelques calculs, on obtient :

$$
\begin{aligned}
(3 \times 2^1 - 1, 2, 2, 0) &\xrightarrow{*} (3 \times 2^3 - 1, 2, 0, 0) \\
&\xrightarrow{*} (3 \times 2^{27} - 1, 1, 0, 0) \\
&\xrightarrow{*} (3 \times 2^{27+3\times 2^{27}} - 1, 0, 0, 0)
\end{aligned}
$$

Preuves de terminaison

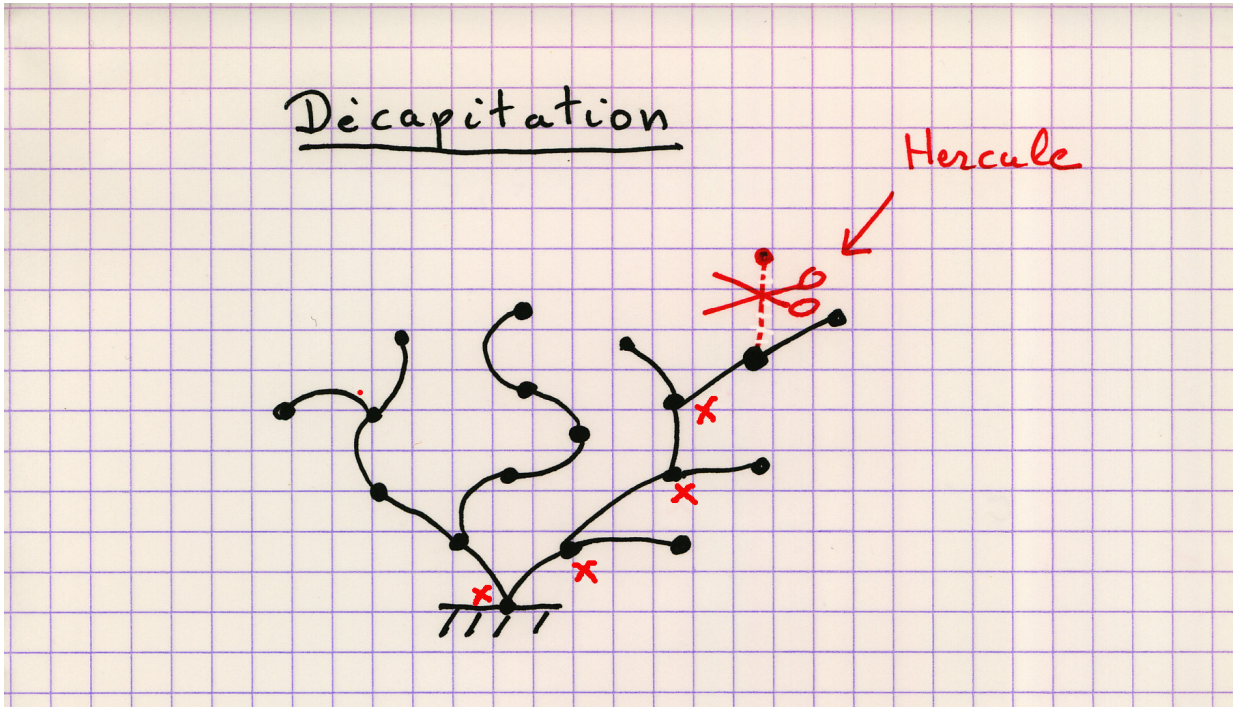
- Une technique classique pour prouver la terminaison d'un processus est d'associer à tout état une *mesure*. Si cette mesure décroît à chaque étape dans un ordre bien fondé, la terminaison est assurée.
- Nous illustrons cette technique sur un problème similaire aux suites de Goodstein : les *batailles d'Hydre*

Batailles d'Hydre



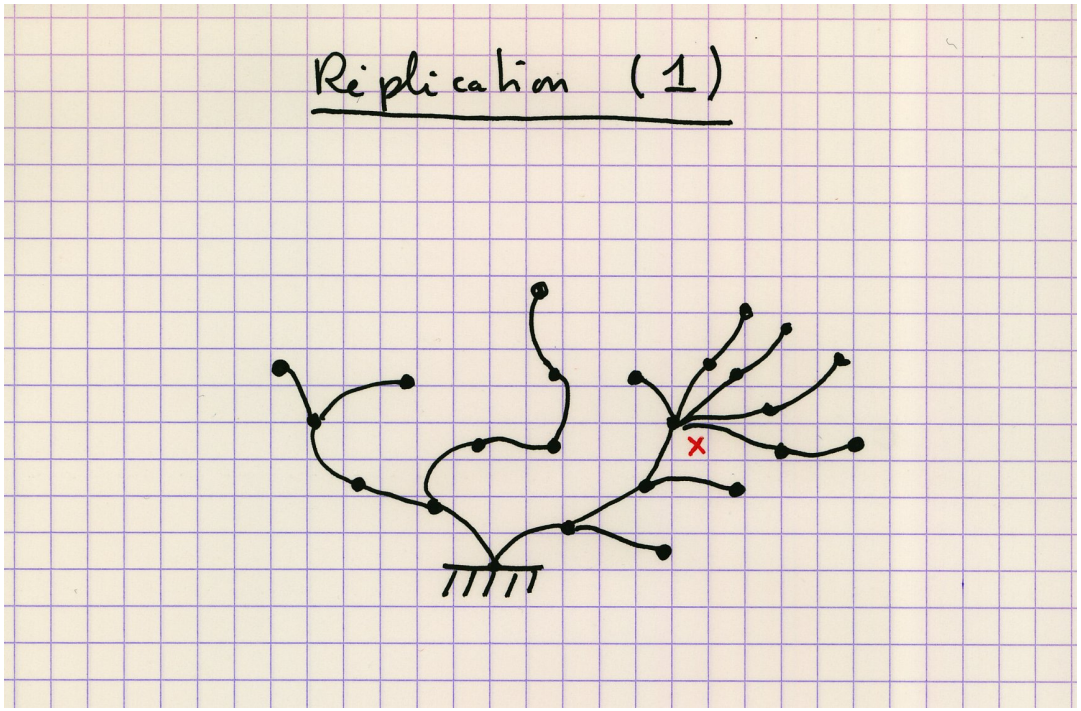
Définition : Une *hydre* est un animal mythique en forme d'arbre fini ; ses feuilles sont appelées *têtes*.

Batailles d'Hydre



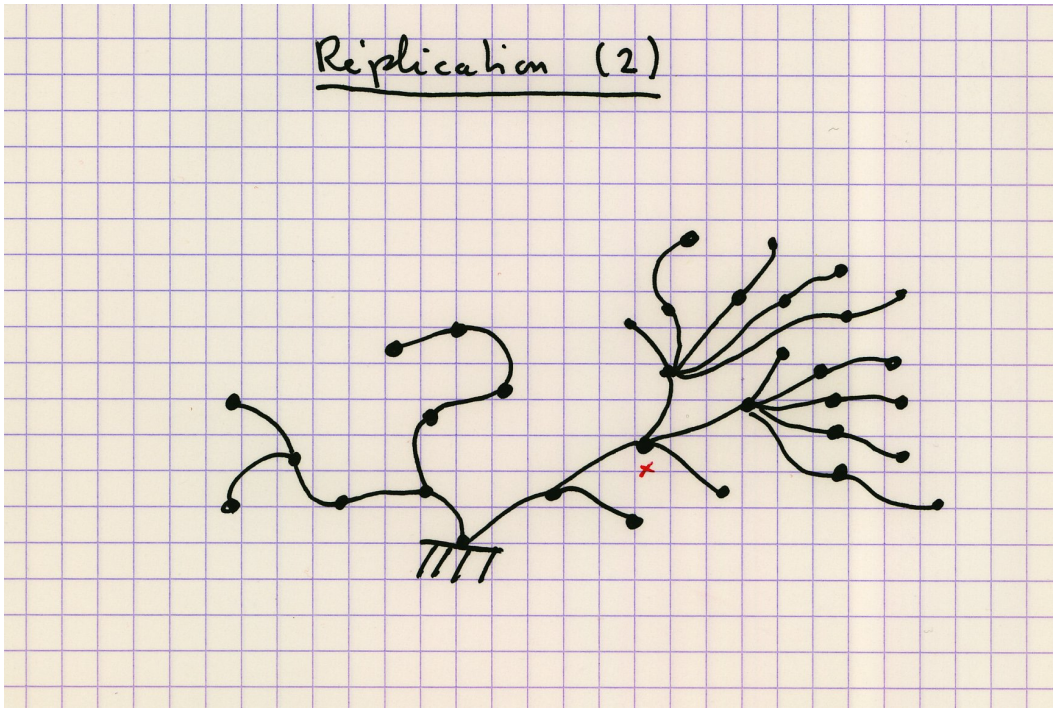
- À chaque tour, Hercule choisit une tête et la coupe
- L'hydre réagit en se répliquant autour des sommets marqués X un nombre fini (quelconque) de fois (des têtes vers le pied)

Batailles d'Hydre



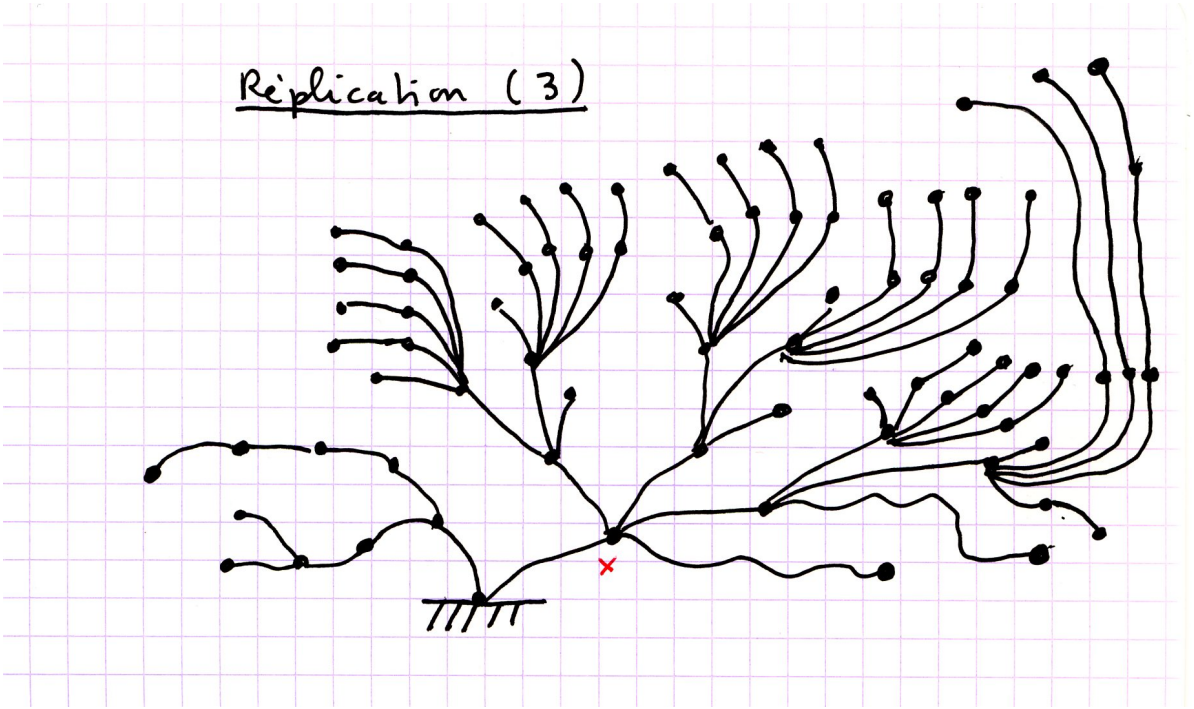
- À chaque tour, Hercule choisit une tête et la coupe
- L'hydre réagit en se répliquant autour des sommets marqués X un nombre fini (quelconque) de fois (des têtes vers le pied)

Batailles d'Hydre



- À chaque tour, Hercule choisit une tête et la coupe
- L'hydre réagit en se répliquant autour des sommets marqués X un nombre fini (quelconque) de fois (des têtes vers le pied)

Batailles d'Hydre



- À chaque tour, Hercule choisit une tête et la coupe
- L'hydre réagit en se répliquant autour des sommets marqués X un nombre fini (quelconque) de fois (des têtes vers le pied)

Théorème : Quelles que soient les stratégies (récursives) d'Hercule (choix d'une tête) et de l'Hydre (nombre de réplications), Hercule finit par gagner (voir Kirby et Paris).

Théorème : Quelles que soient les stratégies (récursives) d'Hercule (choix d'une tête) et de l'Hydre (nombre de réplications), Hercule finit par gagner (voir Kirby et Paris).

Preuve ? Des récurrences sur la taille et/ou la hauteur de l'Hydre sont inenvisageables.

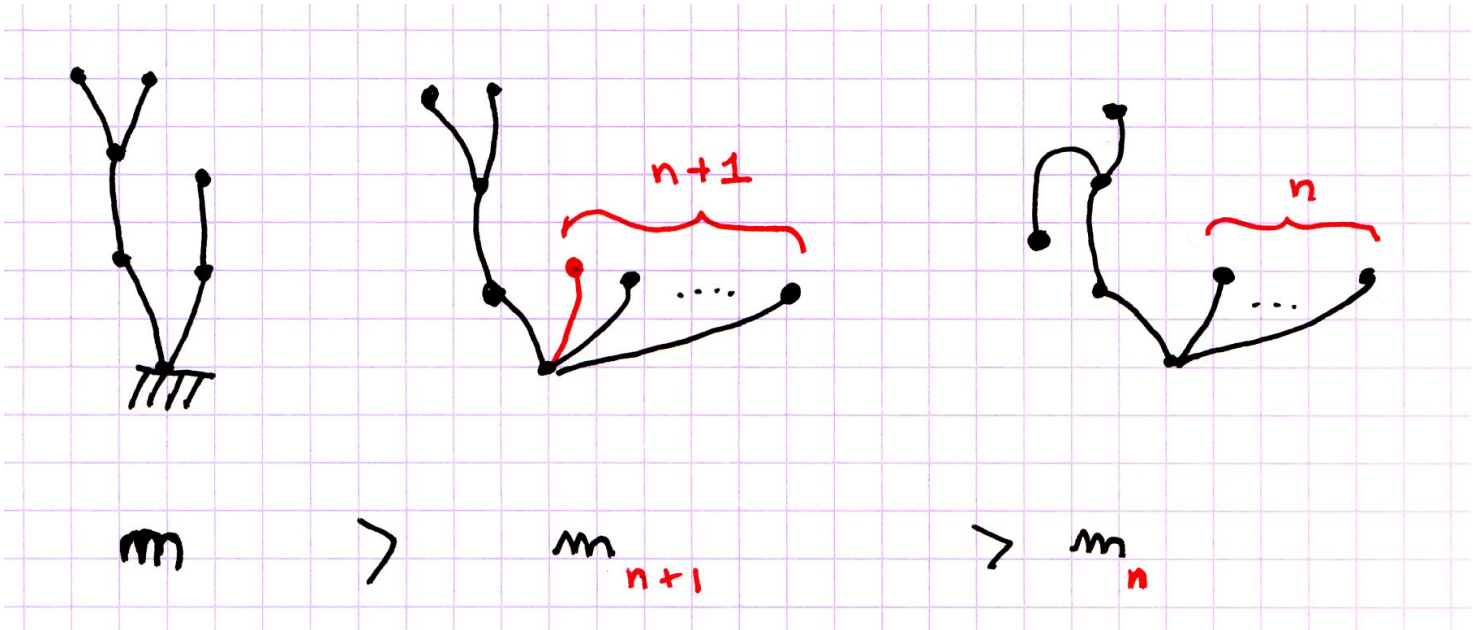
Techniques classiques de terminaison

- Soit une relation $<$ sur un ensemble E. Un élément x de E est *accessible* s'il n'existe pas de suite infinie $x = s_0 > s_1 > \cdots > s_i > s_{i+1} \ldots$. Le *principe de récurrence bien fondée* permet de montrer qu'une propriété P est vérifiée pour tout x accessible.
- $(E, <)$ est *bien fondé* si tout élément de E est accessible pour $<$.

Techniques classiques de terminaison

- Soit une relation $<$ sur un ensemble E. Un élément x de E est *accessible* s'il n'existe pas de suite infinie $x = s_0 > s_1 > \cdots > s_i > s_{i+1} \ldots$. Le *principe de récurrence bien fondée* permet de montrer qu'une propriété P est vérifiée pour tout x accessible.
- $(E, <)$ est *bien fondé* si tout élément de E est accessible pour $<$.
- Pour prouver que toute bataille d'hydre se termine, il suffit d'associer à toute hydre h une mesure $m(h)$ sur un ensemble bien fondé $(E, <)$ telle que si h se transforme en h', alors $m(h') < m(h)$.

E ne peut pas être $\mathbb{N}$, muni de l'ordre naturel $<$.



Un catalogue d'ensembles bien ordonnés

- Un *ordinal* est un ensemble bien ordonné
- On identifie deux ordinaux isomorphes

$$\begin{aligned} 0 &= \emptyset \\ 1 &= \{0\} \\ n+1 &= \{0, 1, \ldots n\} \\ \\ \omega &= \{0, 1, \ldots n, \ldots\} \end{aligned}$$

Arithmétique des ordinaux : la somme

$$
\begin{aligned}
\omega + 1 &= \{0, 1, \ldots n, \ldots \to 0\} \\
\omega + \omega &= \{0, 1, \ldots n, \ldots \to 0, 1, \ldots n, \ldots\} \\
\alpha + \beta &= \alpha \textit{ suivi de } \beta \\
1 + \omega &= \{0, 0, 1, \ldots n, \ldots\} \\
&= \omega \\
&\neq \omega + 1
\end{aligned}
$$

Arithmétique des ordinaux : le produit

$$
\begin{aligned}
\omega \times \omega &= (\mathbb{N} \times \mathbb{N}, <_{\text{lex}}) \\
\alpha \times \beta &= \{(\beta', \alpha') | \beta' \in \beta, \alpha' \in \alpha\} \\
2 \times \omega &= \{(0,0), (0,1), \ldots, (n,0), (n,1), (n+1,0), (n+1,1), \ldots\} \\
&= \omega \\
\omega \times 2 &= \{(0,0), (0,1), \ldots, (0,n), \ldots \longrightarrow (1,0), (1,1), \ldots\} \\
&= \omega + \omega
\end{aligned}
$$

Arithmétique des ordinaux : l'exponentiation de base ω

- Si α est un ordinal, ω^α est l'ensemble des suites finies décroissantes $\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_n$ ordonné par ordre lexicographique.
  Exemple (dans ω^ω) :

$$(12, 10, 10, 9, 9, 9, 8, 8, 8, 8, 8, 1, 1) < (12, 12, 1)$$

Arithmétique des ordinaux : l'exponentiation de base ω

- Si α est un ordinal, ω^α est l'ensemble des suites finies décroissantes $\alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_n$ ordonné par ordre lexicographique.
  Exemple (dans ω^ω) :

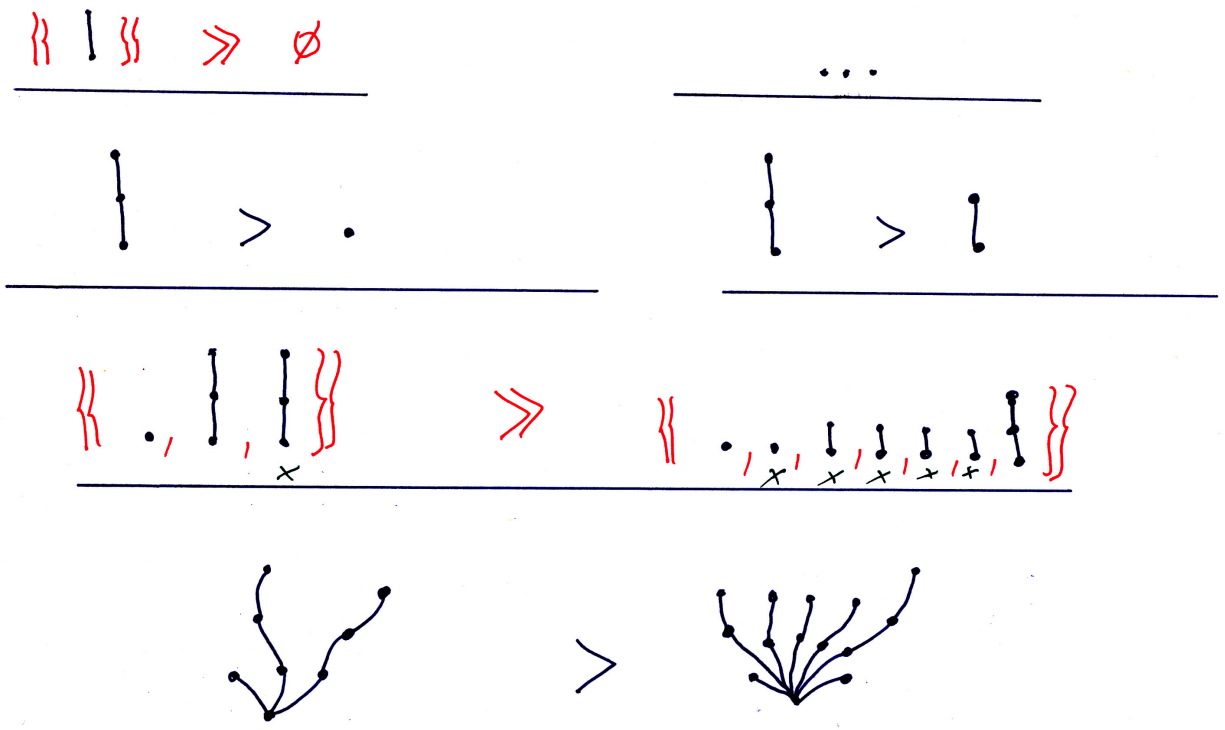
$$(12, 10, 10, 9, 9, 9, 8, 8, 8, 8, 8, 1, 1) < (12, 12, 1)$$

- De façon équivalente, ω^α est l'ensemble des *multiensembles* finis d'éléments de α.
  Exemple :

$$\{\{1, 1, 8, 8, 8, 8, 8, 9, 9, 9, 10, 10, 12\}\} << \{\{1, 12, 12\}\}$$

Hydres et multiensembles

La podoctomie est un isomorphisme d'ordre bien fondé des hydres vers les multiensembles finis d'hydres.



- Un ordinal α mesurant les hydres (et compatible avec leur transformation) doit vérifier l'égalité $\alpha = \omega^\alpha$.

- Un ordinal α mesurant les hydres (et compatible avec leur transformation) doit vérifier l'égalité $\alpha = \omega^\alpha$.
- On appelle ϵ_0 le plus petit ordinal vérifiant l'égalité ci-dessus. ϵ_0 est la borne supérieure de la suite des tours exponentielles ω, ω^ω, ω^{ω^ω}, $\omega^{\omega^{\omega^\omega}}$, etc.

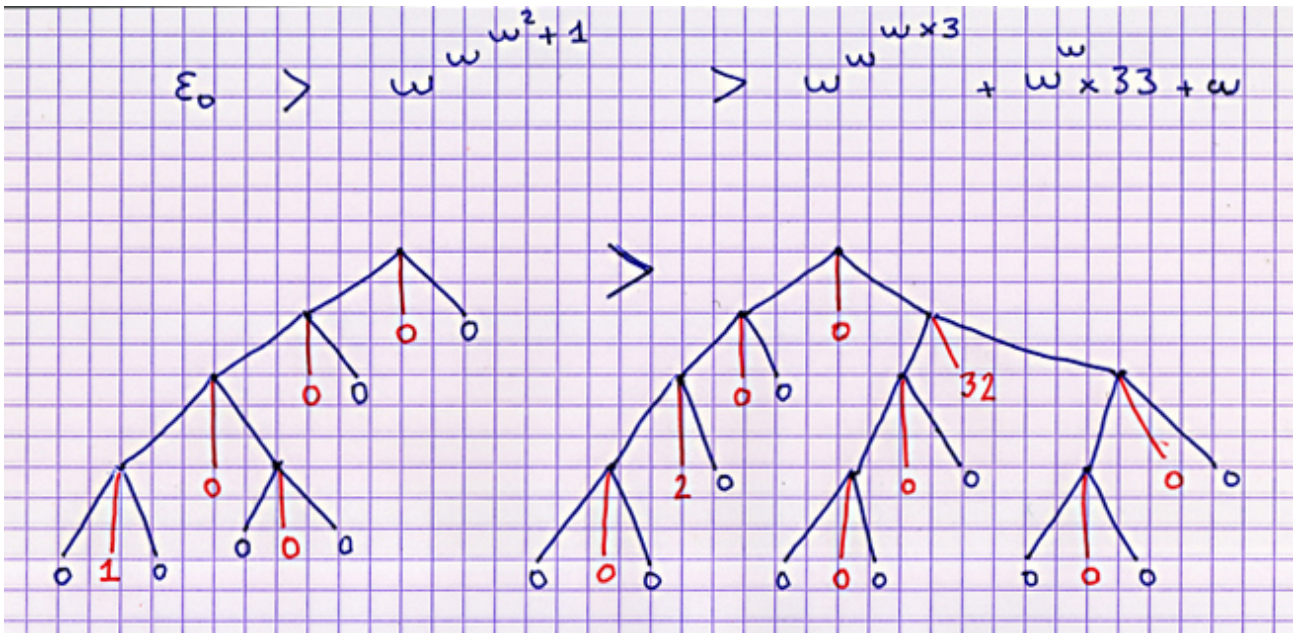
- Un ordinal α mesurant les hydres (et compatible avec leur transformation) doit vérifier l'égalité $\alpha = \omega^\alpha$.
- On appelle ϵ_0 le plus petit ordinal vérifiant l'égalité ci-dessus. ϵ_0 est la borne supérieure de la suite des tours exponentielles ω, ω^ω, ω^{ω^ω}, $\omega^{\omega^{\omega^\omega}}$, etc.
- Pour prouver la terminaison des batailles d'hydre, il suffit d'attribuer un ordinal inférieur à ϵ_0 à toute hydre ; on procède de même pour les items des suites de Goodstein.

Une implantation effective des ordinaux (inférieurs à ϵ_0)

- Bibliothèques sur les ordinaux (permettant de représenter ϵ_0, Γ_0), contenant :
  - Une structure de donnée appropriée,
  - Un ordre linéaire $<$
  - Des capacités de calcul : arithmétique, procédures de décision
  - Une preuve de bonne fondation de $<$.
  - « Validation » par une théorie axiomatique des ordinaux dénombrables (Schütte).

On peut alors prouver que la suite des ordinaux associés aux termes d'une suite de Goodstein est strictement décroissante. De même pour les batailles d'hydre.

```
Inductive T1:Set :=
|  zero
|  cons(alpha:T1)(n:nat)(beta:T1).
```
$\omega^{\alpha} \times (n+1) + \beta$



Preuve de bonne fondation (difficulté)

$$\omega^{\omega^{\omega^{\omega}}} \times 2 > \omega^{\omega^{\omega^{\omega}}} + \omega^{\omega^{\omega}} \times 42 + \omega^{\omega^3} + \omega^{\omega} + 33$$

- L'induction structurelle ne peut suffire
- L'induction sur la hauteur (en ω) non plus

Une preuve complexe

- Lemme : Pour tout α accessible, pour tout n et tout $\beta < \omega^\alpha$, $\omega^\alpha \times (n+1) + \beta$ est accessible.
- preuve : triple induction enchevêtrée :
  1. induction bien fondée sur α,
  2. récurrence sur n,
  3. on prouve que β est accessible
  4. induction bien fondée sur β.

Fin de la preuve

- Tout α est accessible. Preuve par récurrence sur la structure de α (en appliquant le lemme précédent) :
  1. 0 est accessible (car minimal)
  2. Soint α et β accessibles, tels que $\beta < \omega^\alpha$; soit $n \in \mathbb{N}$. Par le lemme précédent, $\omega^\alpha \times (n+1) + \beta$ est accessible.
- Ce type de preuve se retrouve dans plusieurs champs de l'informatique théorique.

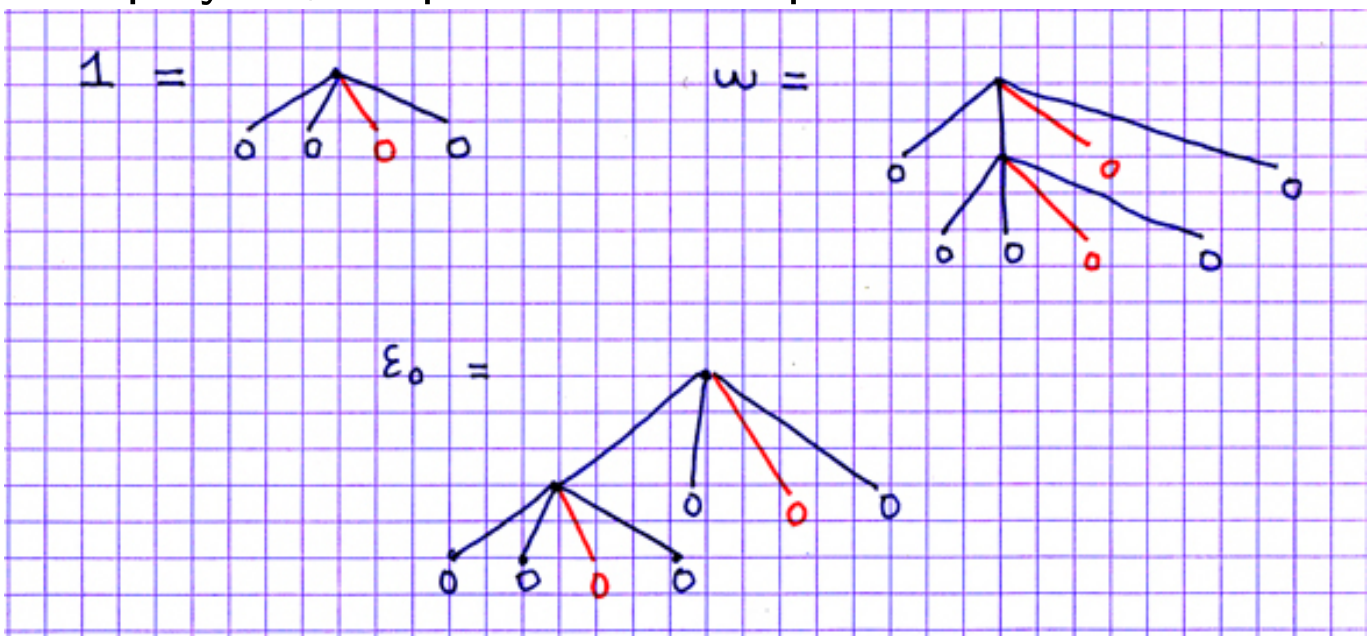
Remarques provisoires

- Nous avons une preuve complète, *sans axiome* de la terminaison des batailles d'hydre et des suites de Goodstein,

```
Theorem Legend :
  forall (str1 : Hercules_strategy)
         (str2 : Hydra_strategy)
         (date : nat) (h : Hydra),
             Hercules_wins str1 str2 date h
```

- Qu'en est-il de notre implantation des ordinaux ?
  - Suffisante comme outil de preuve de terminaison,
  - Limitation ? Conformité avec la théorie ?

On peut pallier la limitation à ϵ_0 en complexifiant les structures de données employées, au prix d'un manque de lisibilité :



Quelques difficultés

- Représentation compacte, mais peu intuitive

```
Inductive T2: Set :=
  | zero : T2
  | cons : T2 → T2 → nat → T2.
```

cons a b n c == $\psi(a, b) \times (n + 1) + c$

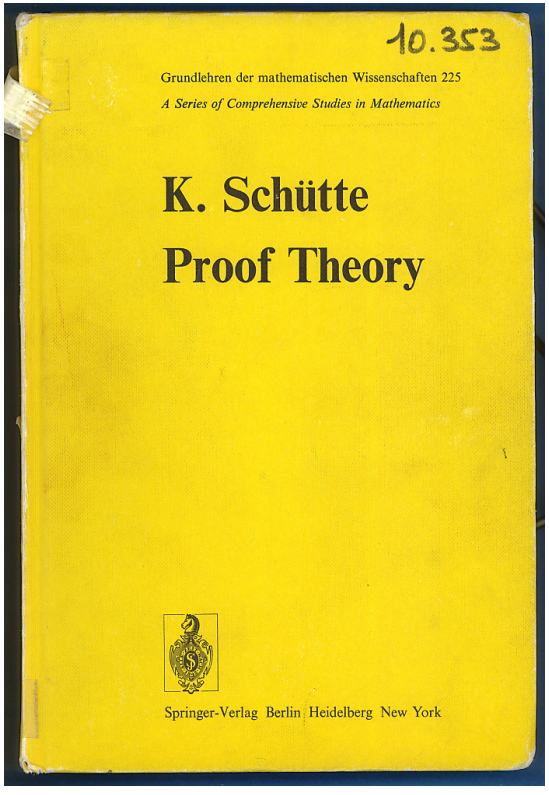
- la relation d'ordre total associée est définie de façon inductive par 7 cas, genre :

```
lt_4 : ∀ α1 α2 β1 β2 n1 n2 γ1 γ2,
       α2 < α1 →
       cons α1 β1 0 zero < β2 →
       cons α1 β1 n1 γ1 <  cons α2 β2 n2 γ2
```

L'exemple de l'addition

```
Fixpoint plus (alpha beta : T2) {struct alpha}:T2 :=
 match alpha,beta with
 |  zero, y  => y
 |  x, zero => x
 |  cons a b n c, cons a' b' n' c' =>
     (match compare (cons a b 0 zero)
                    (cons a' b' 0 zero)
      with | Lt => cons a' b' n' c'
           | Gt => (cons a b n  (c + (cons a' b' n' c')))
           | Eq  => (cons a b (S(n+n')) c')
      end)
 end
where "alpha + beta" := (plus alpha beta): g0_scope.
```

Afin de valider une telle représentation, il est nécessaire de prendre une référence mathématique :



Une Définition axiomatique des ordinaux dénombrables

Ax. I. $\mathbb{O}$ *is a set well-ordered by a relation* $<$.

Ax. II. *Every bounded subset of* $\mathbb{O}$ *is denumerable.* That is: if, given $M \subset \mathbb{O}$ there exists $\alpha \in \mathbb{O}$ such that $\xi < \alpha$ for all $\xi \in M$, then M is a finite or denumerably infinite set.

Ax. III. *Every denumerable subset of* $\mathbb{O}$ *is bounded.* That is: for each finite or denumerably infinite set $M \subset \mathbb{O}$ there exists $\alpha \in \mathbb{O}$ such that $\xi < \alpha$ for all $\xi \in M$.

Corollary: $\mathbb{O}$ is an infinite, but not denumerable set.

Validation de la représentation des ordinaux

- Établir un morphisme injectif des représentations en forme normale de Cantor ou Veblen (ou autres) vers l'ensemble des ordinaux dénombrables (par exemple selon Schütte)
- L'ensemble $\mathbb{O}$ n'étant pas dénombrable, on ne pourra pas *construire* ses éléments :
- Utilisation d'axiomes (traduction des axiomes de Schütte)
- Logique classique, et descriptions non constructives : « un y tel que $f(y) = x$ »

Conclusion

- Preuves fiables, assistance de l'ordinateur (calculs, cas nombreux, vérification de preuves)
- Possibilités de mathématiques classiques (non constructives)
- Interface calcul/raisonnement.

Thèmes de discussion possibles

- Champs d'application
- Semi-automaticité
- Évolution

Annexes

- À quoi ressemblent les règles de typage ?
- Un exemple de défi pour les assistants de preuve

Quelques exemples de règles

$$\textbf{App}\quad \frac{\Gamma \vdash t_1 : \forall\, v : A,\, B \qquad \Gamma \vdash t_2 : A}{\Gamma \vdash t_1\ \ t_2 : B\{v/t_2\}}$$

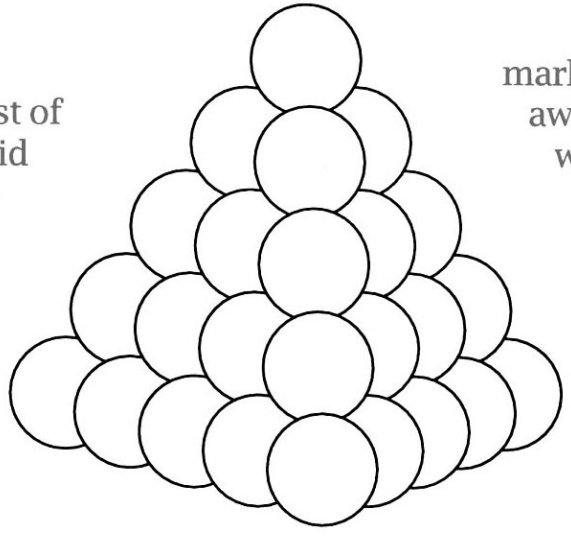
$$\textbf{Lam}\quad \frac{\Gamma :: (v : A) \vdash t : B}{\Gamma \vdash \texttt{fun}\ \ v : A\ \Rightarrow t : \forall\, v : A,\, B}$$

$$\textbf{Conv}\quad \frac{\Gamma \vdash t : A \qquad A \equiv B}{\Gamma \vdash t : B}$$

Conjecture de Kepler

- Conjecture de Kepler (1611) : *Il n'existe pas de façon de ranger des sphères de même diamètre dont la densité soit supérieure à celle du rangement cubique à faces centrées (empilements d'oranges ou de boulets de canon) :* $\frac{\pi}{\sqrt{18}}$
- Contributions de Gauss, Thue (en 2D), Fejes Tóth.

When Hilbert introduced his famous list of 23 problems, he said a test of the perfection of a mathematical problem is whether it can be explained to the first person in the street. Even after a full century, Hilbert's problems have never been thoroughly tested. Who has ever chatted with a telemarketer about the Riemann hypothesis or discussed general reciprocity laws with the family physician?



market. "We need you down here right away. We can stack the oranges, but we're having trouble with the artichokes."

To me as a discrete geometer there is a serious question behind the flippancy. Why is the gulf so large between intuition and proof? Geometry taunts and defies us. For example, what about stacking tin cans? Can anyone doubt that parallel rows of upright cans give the best arrangement? Could some disordered heap of cans

- Preuve par Thomas Hales (1998), dont certaines parties utilisent des calculs sur machine :
  - Énumération de graphes (planaires) pouvant être des contre-exemples à la conjecture. Ces contre-exemples éventuels sont caractérisés par 8 contraintes portant sur les cycles, degrés des sommets, affectation de poids aux faces. Le programme *Java* de Hales énumère 5128 graphes satisfaisant ces contraintes.
  - Programmation linéaire, destinée à vérifier qu'aucun de ces graphes ne constitue un réel contre-exemple.

Le projet Flyspeck

- Vérification de la preuve de Hales par une équipe de 12 arbitres, et conférence consacrée à la preuve : résultat : certitude à 99%,
- Projet Flyspeck : construire une version formelle de la preuve de 1998, principalement à l'aide de *HOL/Light* : mais aussi en *Isabelle/HOL* et *Coq*.

Contribution de Bauer et Nipkow (2005)

- Preuve de la complétude de l'énumération des graphes par le programme de Hales,
- Détection de redondances (2771 graphes au lieu des 5128 énumérés par le programme *Java* de Hales,)
- Détection d'une contrainte absente de la preuve de 98, mais traitée dans le programme *Java*,
- Bilan : la mécanisation de cette preuve a permis de nombreuses simplification dans le calcul de l'énumération des graphes,
- 17000 lignes d'*Isabelle*, 165 minutes de vérification, 2300000 graphes engendrés durant la preuve
- Reste à faire : vérifier que les 2771 graphes ne sont pas des contre-exemples réels à la conjecture de Kepler.