

Négations et quantifications dans les contraintes

Alain Colmerauer

JFPLC, Marseille, 28 juin 2000

Laboratoire d'Informatique de Marseille
CNRS, Universités de Provence
et de la Méditerranée

Monsieur P et Madame S

• Soient x et y deux entiers pris dans $2..100$. Monsieur P sait que Madame S connaît la valeur de la somme $x + y$ et Madame S sait que Monsieur P connaît la valeur du produit xy . Chacun veut connaître les nombres x et y et le dialogue suivant s'engage :

1. Monsieur P : je ne connais pas les deux nombres,
2. Madame S : je le savais,
3. Monsieur P : maintenant je les connais,
4. Madame S : et moi aussi.

• Trouver la valeur de la paire $\{x, y\}$.

• La réponse est

$$\{x, y\} = \{4, 13\}.$$

Monsieur P et Madame S, suite

- On se place dans

$$(\mathbf{Z}, =, \leq, +, \times, 0, 1)$$

- x, y sont les solutions de la contrainte $C(x, y)$:

$$C(x, y) := (\exists p)(\exists s) \left[\begin{array}{l} p = xy \wedge s = x + y \wedge \\ D(x) \wedge D(y) \wedge \\ Q_1(p) \wedge Q_2(s) \wedge Q_3(p) \wedge Q_4(s) \end{array} \right]$$

$$Q_4(s) := (\forall p_1)(\forall p_2) \left[\begin{array}{l} R(p_1, s) \wedge Q_3(p_1) \wedge \\ R(p_2, s) \wedge Q_3(p_2) \rightarrow p_1 = p_2 \end{array} \right],$$

$$Q_3(p) := (\forall s_1)(\forall s_2) \left[\begin{array}{l} R(p, s_1) \wedge Q_2(s_1) \wedge \\ R(p, s_2) \wedge Q_2(s_2) \rightarrow s_1 = s_2 \end{array} \right],$$

$$Q_2(s) := (\forall p_1)(R(p_1, s) \rightarrow Q_1(p_1)),$$

$$Q_1(p) := (\exists s_1)(\exists s_2)(\neg s_1 = s_2 \wedge R(p, s_1) \wedge R(p, s_2)),$$

$$R(p, s) := (\exists x)(\exists y)(D(x) \wedge D(y) \wedge p = xy \wedge s = x + y)$$

$$D(x) := 2 \leq x \wedge x \leq 100.$$

Sur l'équation du 2^e degré

- On se place dans

$$(\mathbf{R}, =, \leq, +, \times, 0, 1)$$

- Déterminer la condition pour laquelle l'équation du deuxième degré admet au moins une solution revient à trouver les nombres réels a, b, c qui satisfont à la contrainte

$$(\exists x)(ax^2 + bx + c = 0).$$

- La réponse est

$$(\neg a = 0 \wedge b^2 - 4ac \geq 0) \vee (a = 0 \wedge \neg b = 0) \vee (a = 0 \wedge c = 0).$$

Sur l'intersection de deux cercles

- On se place dans

$$(\mathbf{R}, =, \leq, +, \times, 0, 1)$$

- Deux cercles de centre distincts ne peuvent avoir plus de deux points en commun, s'écrit :

$$\neg \left[\begin{array}{l} \exists p \exists r \exists q \exists s \\ \exists x_1 \exists y_1 \exists x_2 \exists y_2 \exists x_3 \exists y_3 \\ \neg p = q \wedge \\ \neg (x_1 = x_2 \wedge y_1 = y_2) \wedge \\ \neg (x_1 = x_3 \wedge y_1 = y_3) \wedge \\ \neg (x_2 = x_3 \wedge y_2 = y_3) \wedge \\ (x_1 - p)^2 + (y_1)^2 = r^2 \wedge (x_1 - q)^2 + (y_1)^2 = s^2 \wedge \\ (x_2 - p)^2 + (y_2)^2 = r^2 \wedge (x_2 - q)^2 + (y_2)^2 = s^2 \wedge \\ (x_3 - p)^2 + (y_3)^2 = r^2 \wedge (x_3 - q)^2 + (y_3)^2 = s^2 \end{array} \right]$$

Optimisation linéaire classique

- On se place dans

$$(\mathbf{Q}, =, \leq, +, -, 0, 1)$$

ou

$$(\mathbf{R}, =, \leq, +, -, 0, 1)$$

- Trouver le plus grand x qui satisfait une contrainte linéaire, revient à trouver un x tel que

$$p(x) \wedge (\forall u p(u) \rightarrow u \leq x)$$

avec

$$p(x) := \begin{array}{l} \exists y_1 \dots \exists y_n \\ \left[\begin{array}{l} a_{01}y_1 + \dots + a_{0n}y_n = x \wedge \\ a_{11}y_1 + \dots + a_{1n}y_n \leq b_1 \wedge \\ \vdots \\ a_{m1}y_1 + \dots + a_{mn}y_n \leq b_m \end{array} \right] \end{array}$$

et, suivant que a_{ij} est un entier positif ou négatif,

$$a_{ij}y_j := \underbrace{y_j + \dots + y_j}_{a_{ij}}$$

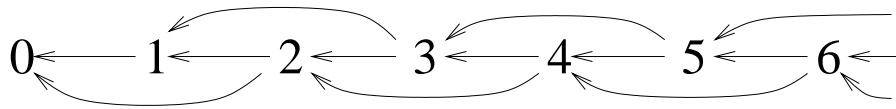
ou

$$a_{ij}y_j := \underbrace{(-y_j) + \dots + (-y_j)}_{-a_{ij}}$$

Positions k -gagnantes dans un jeu

- On se donne un entier positif ou nul i et à tour de rôle on soustrait 1 ou 2 de i sans jamais rendre i strictement négatif. La première personne qui ne peut plus jouer a perdu.

- Graphe (V, E) du jeu



- Soit $x \in V$ un sommet (appelé aussi position) quelconque du graphe orienté (V, E) et supposons que ce soit au tour de la personne A de jouer. La position x est dite k -gagnante si, quelle que soit la façon de jouer de l'autre personne B , il est toujours possible que A gagne en jouant au plus k coups.

- On montre que l'ensemble des positions k -gagnantes du jeu précédent est

$$\{i \in \mathbf{N} \mid i < 3k \text{ et } i \bmod 3 \neq 0\}$$

Positions k -gagnantes dans un jeux, suite 1

- En écrivant $\text{gagnant}_k(x)$, pour x est une position k -gagnante de (V, E) , on montre que

$$\text{gagnant}_k(x) \leftrightarrow \left[\begin{array}{l} \exists y \text{ coup}(x, y) \wedge \neg(\\ \exists x \text{ coup}(y, x) \wedge \neg(\\ \exists y \text{ coup}(x, y) \wedge \neg(\\ \exists x \text{ coup}(y, x) \wedge \neg(\\ \dots \\ \exists y \text{ coup}(x, y) \wedge \neg(\\ \exists x \text{ coup}(y, x) \wedge \neg(\\ \text{faux} \quad \underbrace{\quad \quad \quad}_{2k} \end{array} \right]$$

- avec

$$\text{coup}(x, y) \leftrightarrow (x, y) \in E.$$

- D'une façon plus générale si, $V \subseteq \mathbf{D}$, on peut prendre n'importe quel relation binaire coup dans \mathbf{D} telle que

1. pour tous $x \in V$ et $y \in V$, $\text{coup}(x, y) \leftrightarrow (x, y) \in E$,
2. pour tout $x \in \mathbf{D} - V$ il existe $y \in \mathbf{D} - V$ tel que $\text{coup}(x, y)$.

Littérature : Thi-Bich-Hanh Dao et Alain Colme-rauer, CP2000.

Positions k -gagnantes dans un jeu, suite 2

La relation binaire *coup* du jeu précédent s'écrit,

- dans $(\mathbf{N}, =, +, 0, 1)$, l'arithmétique de Presburger,

$$\text{coup}(x, y) \leftrightarrow x = y + 1 \vee x = y + 1 + 1,$$

- dans $(\mathbf{Z}, =, +, 0, 1)$, les entiers additifs,

$$\text{coup}(x, y) \leftrightarrow \left[\begin{array}{l} (x = y + 1 \wedge \neg x = 0) \vee \\ (x = y + 1 + 1 \wedge \neg x = 0 \wedge \neg x = 1) \end{array} \right],$$

- dans $(\mathbf{Q}, =, +, 0, 1)$, les rationnels additifs,

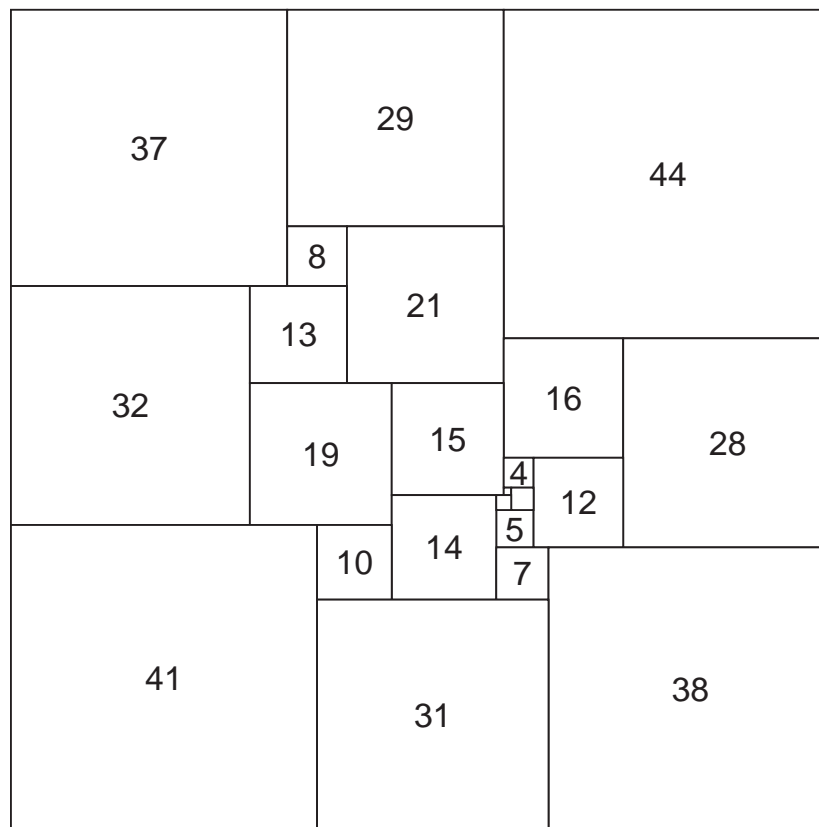
$$\text{coup}(x, y) \leftrightarrow \left[\begin{array}{l} (x = y + 1 \wedge \neg x = 0) \vee \\ (x = y + 1 + 1 \wedge \neg x = 0 \wedge \neg x = 1) \end{array} \right],$$

- dans $(\mathbf{A}, F \cup \{s, 0\})$, l'algèbre des arbres,

$$\text{coup}(x, y) \leftrightarrow \left[\begin{array}{l} x = s(y) \vee \\ x = s(s(y)) \vee \\ (x = y \wedge \neg x = 0 \wedge \neg(\exists z)(x = s(z))) \end{array} \right]$$

Décomposition d'un carré en n carrés de tailles toutes distinctes

Exemple pour $n = 22$.



Littérature : Ian Gambini, thèse et article.

Décomposition d'un carré, suite 1

- On se place dans

$$(\mathbf{R}, =, \leq, +, -, 0, 1)$$

- Pour n fixé, contrainte sur les tailles c_i des $n+1$ carrés :

$$\text{décomposition}(c_0, c_1, \dots, c_n) :=$$

$$0 < c_1 \wedge c_1 < c_2 \wedge \dots \wedge c_{n-1} < c_n \wedge$$

$$\left[\begin{array}{l} \exists x_1 \exists y_1 \dots \exists x_n \exists y_n \\ \left(\bigwedge_{i=1}^n \left[\begin{array}{l} x_i \in [0, c_0 - c_i) \wedge \\ y_i \in [0, c_0 - c_i) \end{array} \right] \wedge \right. \\ \left. \left[\begin{array}{l} \forall x \forall y \forall b_1 \dots \forall b_n \\ \left[\begin{array}{l} x \in [0, c_0) \wedge \\ y \in [0, c_0) \wedge \\ \left(\bigwedge_{i=1}^n [b_i = 0 \vee b_i = 1] \right) \wedge \\ \left(\bigwedge_{i=1}^n \left[\begin{array}{l} x \in [x_i, x_i + c_i) \wedge \\ y \in [y_i, y_i + c_i) \wedge \\ \leftrightarrow b_i = 1 \end{array} \right] \right) \end{array} \right] \right] \\ \rightarrow \sum_{i=1}^n b_i = 1 \end{array} \right] \end{array} \right]$$

Décomposition d'un carré, suite 2

- On se place dans

$$(\mathbf{Z}, =, \leq, +, -, 0, 1)$$

- On décompose un carré de dimension $n \times n$ en n carrés les m premiers de tailles c_i nulles et les $n - m$ derniers de tailles non nulles et toutes distinctes.

$$\text{décomposition entière } e_n(c_1, \dots, c_n) :=$$

$$\left[\begin{array}{l} \exists c_0 \ c_0 = 0 \ \wedge \\ \left(\bigwedge_{i=1}^n c_{i-1} \leq c_i \right) \ \wedge \\ \left(\bigwedge_{i=2}^n c_{i-1} = c_i \ \rightarrow \ c_{i-2} = c_{i-1} \right) \end{array} \right] \wedge$$

$$\left[\begin{array}{l} \exists x_1 \exists y_1 \dots \exists x_n \exists y_n \\ \left(\bigwedge_{i=1}^n \left[\begin{array}{l} x_i \in [0, n-1-c_i] \ \wedge \\ y_i \in [0, n-1-c_i] \end{array} \right] \ \wedge \right) \\ \left[\begin{array}{l} \forall x \forall y \forall b_1 \dots \forall b_n \\ \left[\begin{array}{l} x \in [0, n-1] \ \wedge \\ y \in [0, n-1] \ \wedge \\ \left(\bigwedge_{i=1}^n [b_i = 0 \vee b_i = 1] \right) \ \wedge \\ \left(\bigwedge_{i=1}^n \left[\begin{array}{l} x \in [x_i, x_i + c_i - 1] \ \wedge \\ y \in [y_i, y_i + c_i - 1] \ \wedge \\ \leftrightarrow b_i = 1 \end{array} \right] \right) \end{array} \right] \\ \rightarrow \sum_{i=1}^n b_i = 1 \end{array} \right] \end{array} \right]$$

Nombre premier

On se place dans

$$(\mathbf{Z}, =, \leq, +, \times, 0, 1)$$

- Contrainte évidente

$$\begin{aligned} \text{premier}(\nu) &:= \\ \nu &\geq 2 \wedge \neg(\exists x)(\exists y)(x \geq 2 \wedge y \geq 2 \wedge \nu = xy) \end{aligned}$$

- Contrainte pas évidente

$$\text{premier}(\nu) :=$$

$$\nu \geq 0 \wedge$$

$$(\exists a)(\exists b) \dots (\exists z)(a \geq 0 \wedge b \geq 0 \wedge \dots \wedge z \geq 0 \wedge$$

$$\nu = (k + 2)(1 - (wz + h + j - q)^2$$

$$- ((gk + 2g + k + 1)(h + j) + h - z)^2$$

$$- (2n + p + q + z - e)^2$$

$$- (16(k + 1)^3(k + 2)(n + 1)^2 + 1 - f^2)^2$$

$$- (e^3(e + 2)(a + 1)^2 + 1 - o^2)^2$$

$$- ((a^2 - 1)y^2 + 1 - x^2)^2$$

$$- (16r^2y^4(a^2 - 1) + 1 - u^2)^2$$

$$- (n + l + v - y)^2$$

$$- (((a + u^2(u^2 - a))^2 - 1)(n + 4dy)^2 + 1 - (x + cu)^2)^2$$

$$- ((a^2 - 1)l^2 + 1 - m^2)^2$$

$$- (q + y(a - p - 1) + s(2ap + 2a - p^2 - 2p - 2) - x)^2$$

$$- (z + pl(a - p) + t(2ap - p^2 - 1) - pm)^2$$

$$- (ai + k + 1 - l - i)^2$$

$$- (p + l(a - n - 1) + b(2an + 2a - n^2 - 2n - 2) - m)^2))$$

Littérature : Jones, Sato, Wada et Wiens [1976].

Arbre fini énorme

- On se place dans l'algèbre des arbres finis ou infinis

$$(\mathbf{A}, =, F)$$

- On suppose que l'ensemble d'arbre \mathbf{A} est construit sur un ensemble de symboles fonctionnels comprenant notamment les symboles $0, 1, 2, 3, s, f$, d'ariétés respectives $0, 0, 0, 0, 1, 4$. Il est alors possible de définir une famille de contrainte *énorme* $_k(x)$, avec $k \geq 0$ telle que, pour $k \geq 0$, on ait :

$$\textit{énorme}_k(x) \leftrightarrow x = s \underbrace{2^{2^{\dots^2}}}_k(0)$$

$$|\textit{énorme}_k(x)| = 9 + 158k$$

- A noter que $\underbrace{2^{2^{\dots^2}}}_5 > 10^{20000}$, un nombre probablement bien supérieur au nombre d'atomes constituant l'univers et au nombre de nanosecondes qui se sont écoulées depuis sa création.

Arbre fini énorme, suite 1

- Pour $k \geq 0$,

$$\text{énorme}_k(x) := \exists z \text{ triangle}_k(3, x, z, s(0))$$

- avec toujours pour $k \geq 0$,

$$\text{triangle}_0(t, x, z, y) := z = x \wedge z = y$$

$$\text{triangle}_{k+1}(t, x, z, y) := \left[\left[\begin{array}{l} [\exists u_1 \exists u_2 z = f(x, u_1, u_2, y)] \\ \wedge \\ \left[\forall t' \forall y' \forall z' \right. \\ \left. \left[(t' = 1 \vee t' = 2) \wedge \right. \right. \\ \left. \left. \text{triangle}_k(t', z, z', y') \right] \rightarrow \right. \\ \left. \left[(t' = 1 \wedge \text{forme1}(y')) \vee \right. \right. \\ \left. \left. \left(t' = 2 \wedge \left[\begin{array}{l} \exists u \exists v \text{ forme2}(u, y', v) \wedge \\ (t = 1 \rightarrow \text{trans1}(u, v)) \wedge \\ (t = 2 \rightarrow \text{trans2}(u, v)) \wedge \\ (t = 3 \rightarrow \text{trans3}(u, v)) \end{array} \right] \right) \right] \right] \end{array} \right] \right]$$

- et

$$\text{forme1}(x, z, y) := \exists u_1 \dots \exists u_6 z = f(u_1, f(u_1, u_2, u_3, x), f(y, u_4, u_5, u_6), u_6)$$

$$\text{forme2}(x) := \exists u_1 \dots \exists u_4 x = f(u_1, f(u_2, u_2, u_2, u_2), f(u_3, u_3, u_3, u_3), u_4)$$

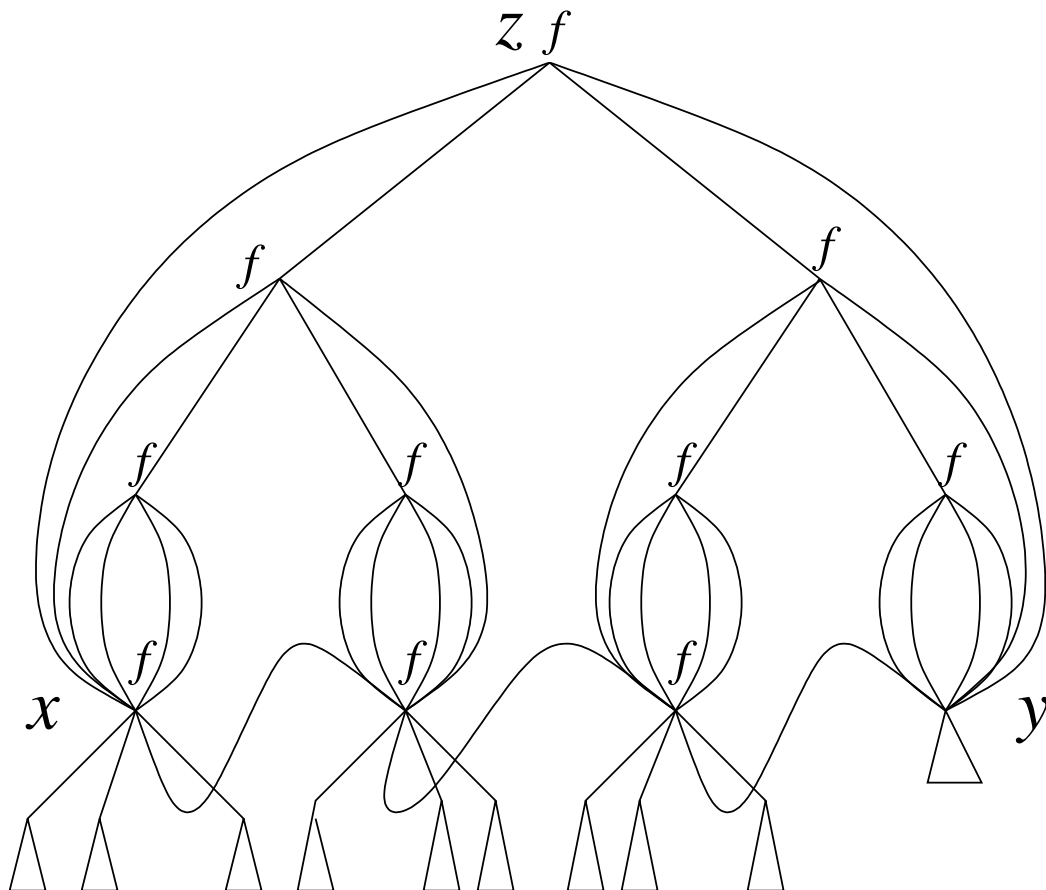
$$\text{trans1}(x, y) := \exists u_1 \dots \exists u_4 x = f(u_1, u_2, u_3, u_4) \wedge (y = u_2 \vee y = u_3)$$

$$\text{trans2}(x, y) := \text{trans2}(x, y) \vee x = y$$

$$\text{trans3}(x, y) := x = s(y)$$

Arbre fini énorme, suite 2

Pour donner une idée de ce que $\text{triangle}_k(t, x, z, y)$ signifie, voici la disposition de trois arbres x, z, y tels que $\text{triangle}_2(1, x, z, y)$:



Littérature : P. Mielniczuk, Thi-Bich-Hanh Dao et Alain Colmerauer CP2000.

Formule normalisée

- En utilisant les équivalences

$$\begin{aligned} \text{faux} &\leftrightarrow \neg \text{vrai}, \\ p \vee q &\leftrightarrow \neg(\neg p \wedge \neg q), \\ p \rightarrow q &\leftrightarrow \neg(p \wedge \neg q), \\ (\forall x)p &\leftrightarrow \neg(\exists x)\neg p, \end{aligned}$$

on transforme une formule construite sur l'alphabet

$$V \cup R \cup F \cup \{\text{vrai}, \text{faux}, \neg, \wedge, \vee, \rightarrow, \exists\}$$

en une formule équivalente construite sur l'alphabet

$$V \cup R \cup F \cup \{\text{vrai}, \neg, \wedge, \exists\}.$$

- En utilisant le fait que

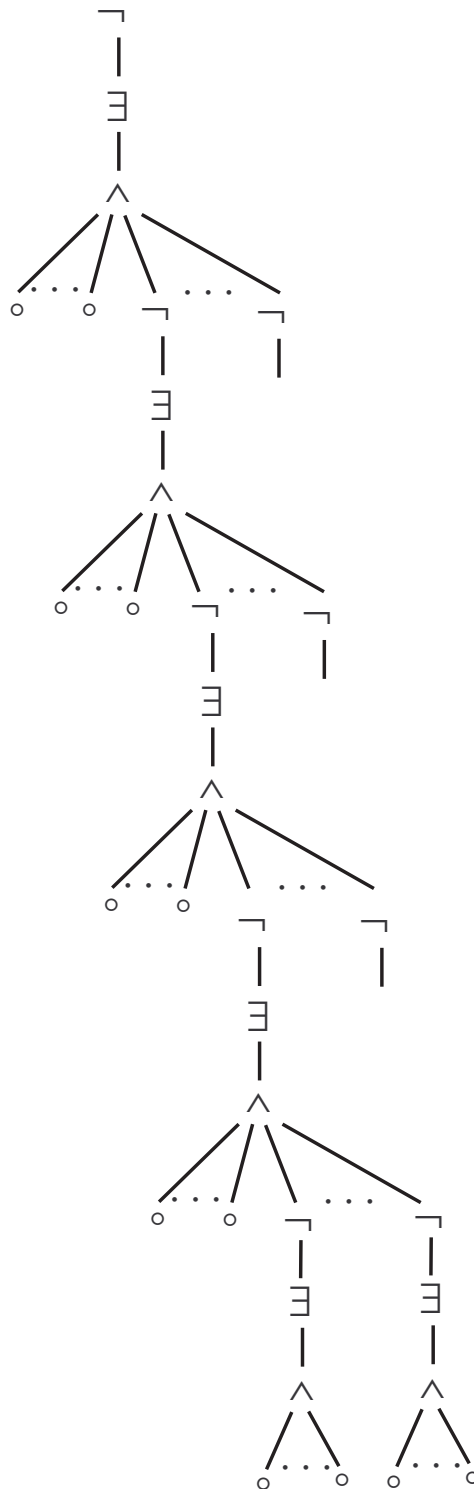
$$p \wedge (\exists x)q(x) \leftrightarrow (\exists y)(p \wedge q(y))$$

on remonte certaines quantifications existentielles pour se ramener à un ensemble P de formules normalisées de la forme

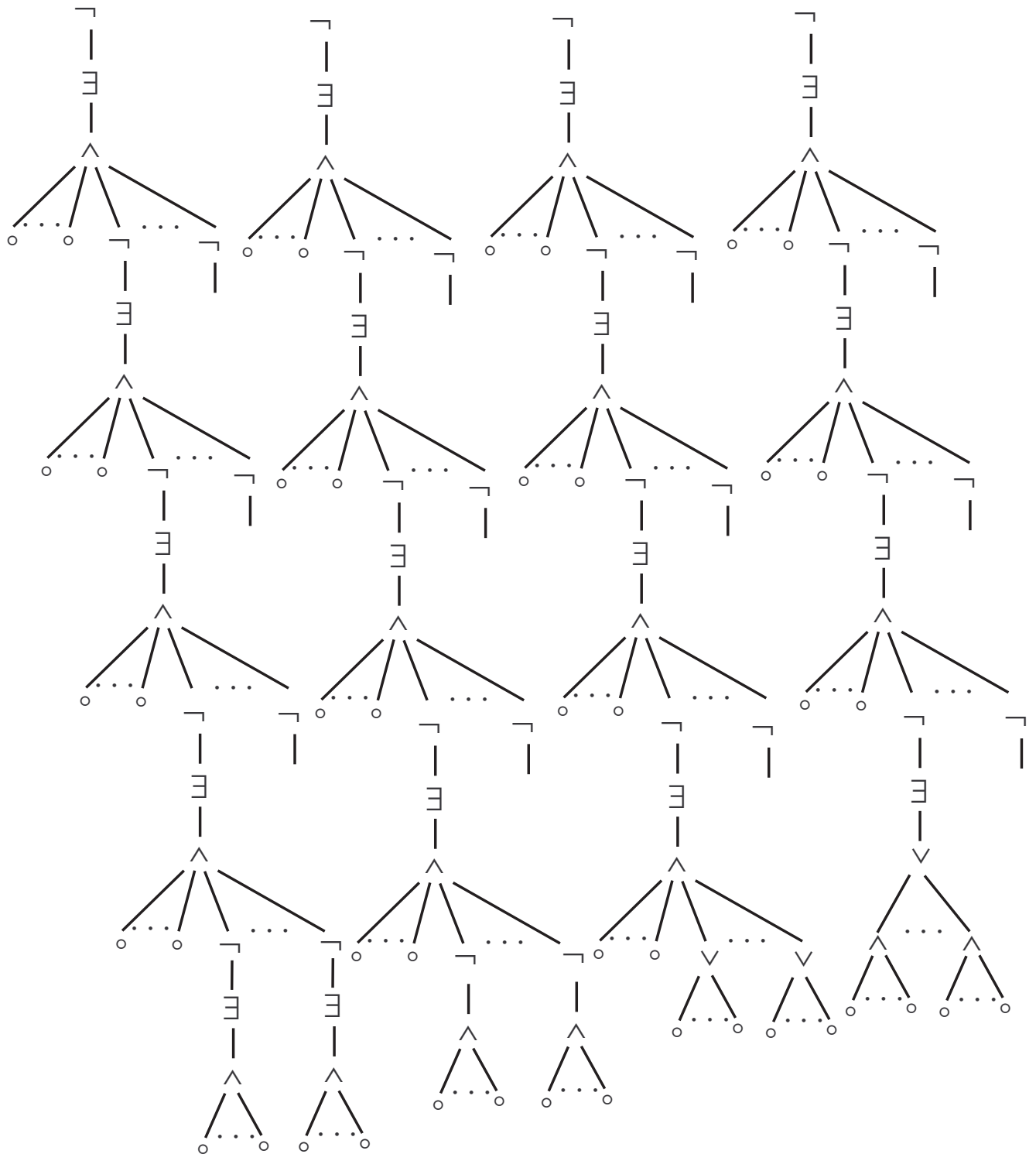
$$\neg(\exists x_1) \dots (\exists x_k)(\text{vrai} \wedge e_1 \wedge \dots \wedge e_m \wedge p_1 \wedge \dots \wedge p_n)$$

où les e_i sont des formules élémentaires et les p_i des éléments de P .

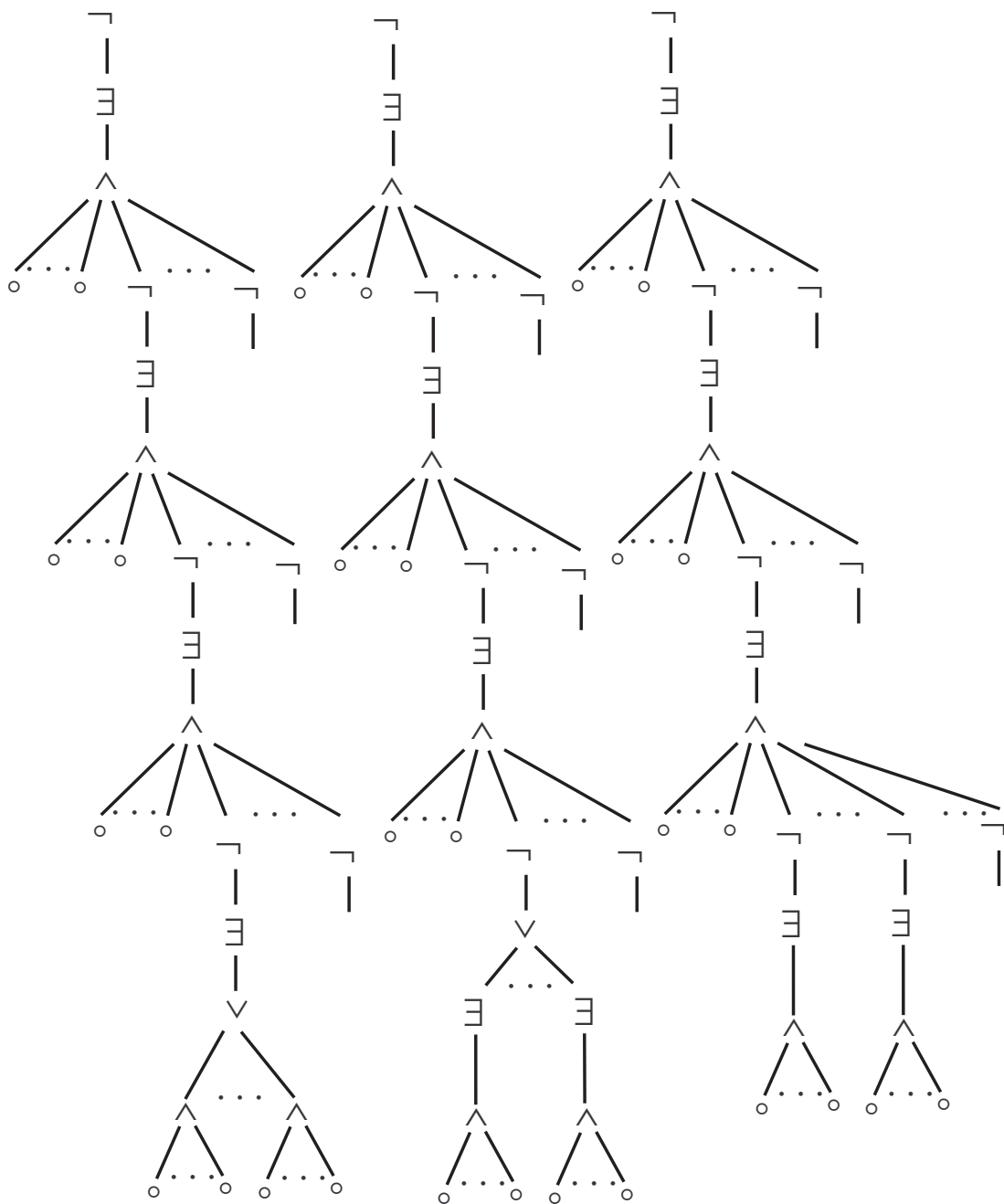
Exemple de formule normalisé



Elimination de quantificateurs, 1



Elimination de quantificateurs, 2



Ordre total dense sans extrêmes

- On est dans

$$(E, =, \leq, <)$$

- Axiomatisation

Lien entre \leq et $<$

$$(0) (\forall x)(\forall y)(x < y \leftrightarrow \neg y \leq x)$$

Ordre

$$(1) (\forall x)(x \leq x)$$

$$(2) (\forall x)(\forall y)(x \leq y \wedge y \leq x \rightarrow x = y)$$

$$(3) (\forall x)(\forall y)(\forall z)(x \leq y \wedge y \leq z \rightarrow x \leq z)$$

Ordre total

$$(4) x \leq y \vee y \leq x$$

Ordre dense

$$(5) (\forall x)(\forall y)(x < z \rightarrow (\exists y)(x < y \wedge y < z))$$

Ordre sans extrêmes

$$(6) (\forall x)(\exists y)(y < x)$$

$$(7) (\forall x)(\exists y)(x < y)$$

Ordre total dense sans extrêmes, suite

Des axiomes précédent on déduit notamment :

$$\bullet x = y \leftrightarrow x \leq y \wedge y \leq x$$

$$\bullet x \leq x \leftrightarrow \text{vrai}$$

$$\bullet \neg x \leq y \leftrightarrow y < x$$

$$\bullet (\exists x) \left(\begin{array}{l} (\wedge_i u_i < x) \wedge \\ (\wedge_i v_i \leq x) \wedge \\ (\wedge_i x < y_j) \wedge \\ (\wedge_i x \leq z_j) \wedge \\ p \end{array} \right) \leftrightarrow \left(\begin{array}{l} (\wedge_{i,j} u_i < y_j) \wedge \\ (\wedge_{i,j} u_i < z_j) \wedge \\ (\wedge_{i,j} v_i < y_j) \wedge \\ (\wedge_{i,j} v_i \leq z_j) \wedge \\ p \end{array} \right)$$

Note : on omet d'écrire les quantifications universelles les plus extérieures.

Rationnels additifs ordonnés

- On est dans

$$(\mathbf{Q}, =, \leq, <, +, -, 0, 1)$$

- Axiomatisation

Relation d'ordre total dense sans extrêmes

(0–7)

Propriétés des constantes

$$(8) \neg 0 = 1$$

Propriétés de l'addition

$$(9) x + (y + z) = (x + y) + z$$

$$(10) x + y = y + x$$

$$(11) x + 0 = x$$

$$(12) x + (-x) = 0$$

$$(13_n) (\exists y) \underbrace{(y + \cdots + y)}_n = x$$

Interférence de l'ordre avec les constantes et l'addition

$$(14) 0 \leq 1$$

$$(15) x \leq y \rightarrow x + z \leq y + z$$

Rationnels additifs ordonnés, suite

Des axiomes précédents on déduit notamment

$$\bullet s = t \leftrightarrow s \leq t \wedge t \leq s$$

$$\bullet 0 \leq 1 + \dots + 1 \leftrightarrow \text{vrai}$$

$$\bullet x \leq x \leftrightarrow \text{vrai}$$

$$\bullet s \leq t \leftrightarrow ks \leq kt$$

$$\bullet \neg s \leq t \leftrightarrow t < s$$

$$\bullet (\exists x) \left(\begin{array}{l} (\wedge_i a_i < kx) \wedge \\ (\wedge_i b_i \leq kx) \wedge \\ (\wedge_i kx < c_i) \wedge \\ (\wedge_i kx \leq d_i) \wedge \\ p \end{array} \right) \leftrightarrow \left(\begin{array}{l} (\wedge_{i,j} a_i < c_j) \wedge \\ (\wedge_{i,j} a_i < d_j) \wedge \\ (\wedge_{i,j} b_i < c_j) \wedge \\ (\wedge_{i,j} b_i \leq d_j) \wedge \\ p \end{array} \right)$$

où les s, t, a_i, b_i, c_i, d_i sont des termes et où on écrit kt pour $\underbrace{t + \dots + t}_k$

Littérature : Algorithme de Jean Baptiste Joseph Fourier.

Ordre total discret sans extrêmes

- On est dans

$$(E, \leq, \text{pred}, \text{succ})$$

- Axiomatisation

Ordre

$$(1) (\forall x)(x \leq x)$$

$$(2) (\forall x)(\forall y)(x \leq y \wedge y \leq x \rightarrow x = y)$$

$$(3) (\forall x)(\forall y)(\forall z)(x \leq y \wedge y \leq z \rightarrow x \leq z)$$

Ordre total

$$(4) (\forall x)(\forall y)(x \leq y \vee y \leq x)$$

Pas d'extrêmes et existence de

prédécesseurs et successeurs

$$(5a) (\forall x)(\neg x \leq \text{pred}(x))$$

$$(5b) (\forall x)(\neg \text{succ}(x) \leq x)$$

$$(6a) (\forall x)(\forall y)(\neg y \leq x \rightarrow x \leq \text{pred}(y))$$

$$(6b) (\forall x)(\forall y)(\neg y \leq x \rightarrow \text{succ}(x) \leq y)$$

Ordre total discret sans extrêmes, suite

Des axiomes on déduit notamment

- $\text{pred}(\text{succ}(x)) = x$
- $\text{succ}(\text{pred}(x)) = x$
- $x = y \leftrightarrow x \leq y \wedge y \leq x$
- $x \leq \text{succ}^n(x) \leftrightarrow \text{vrai}$
- $x \leq \text{pred}^{n+1}(x) \leftrightarrow \neg \text{vrai}$
- $\text{pred}(x) \leq \text{pred}(y) \leftrightarrow x \leq y$
- $\text{succ}(x) \leq \text{succ}(y) \leftrightarrow x \leq y$
- $\neg s \leq t \leftrightarrow \text{succ}(t) \leq s$
- $(\exists x) \left(\begin{array}{l} ((\wedge_i s_i \leq x) \wedge) \\ ((\wedge_i x \leq t_i) \wedge) \\ p \end{array} \right) \leftrightarrow \left(\begin{array}{l} ((\wedge_{i,j} s_i \leq t_j) \wedge) \\ p \end{array} \right)$

Entiers additifs ordonnés

- On est dans

$$(Z, =, \leq, +, -, 0, 1)$$

- Axiomatisation

Propriétés des constantes

$$(1) \neg 0 = 1$$

Propriétés de l'addition

$$(2) x + (y + z) = (x + y) + z$$

$$(3) x + y = y + x$$

$$(4) x + 0 = x$$

$$(5) x + (-x) = 0$$

$$(6_n) (\exists y)(\exists z)(x = \underbrace{y + \dots + y}_n \wedge \left(\begin{array}{l} z = 0 \vee \\ z = 1 \vee \\ z = 1 + 1 \vee \\ \dots \\ z = \underbrace{1 + \dots + 1}_{n-1} \end{array} \right))$$

Relation d'ordre total

$$(7) x \leq x$$

$$(8) x \leq y \wedge y \leq x \rightarrow x = y$$

$$(9) x \leq y \wedge y \leq z \rightarrow x \leq z$$

$$(10) x \leq y \vee y \leq x$$

Interférence de l'ordre avec le reste

$$(11) 0 \leq 1$$

$$(12) \neg x \leq 0 \rightarrow 1 \leq x$$

$$(13) x \leq y \rightarrow x + z \leq y + z$$

Entiers additifs ordonnés, suite

- On écrit

$k|x$ pour $(\exists y)(x = \underbrace{y + \dots + y}_k)$, c'est-à-dire k divise x ,
 $k \nmid x$ pour $\neg(k|x)$

Des axiomes précédents on déduit alors

- $\neg s \leq t \leftrightarrow t \leq s + 1$

- $s \leq t \leftrightarrow ks \leq kt$

- $\gamma|t \leftrightarrow k\gamma|kt$

- $\gamma \nmid t \leftrightarrow k\gamma \nmid kt$

- $(\exists y) \left(\begin{array}{l} (\wedge_i a_i \leq ky) \wedge \\ (\wedge_i ky \leq b_i) \wedge \\ (\wedge_i \gamma_i | ky + c_i) \wedge \\ (\wedge_i \delta_i \nmid ky + d_i) \end{array} \right) \leftrightarrow (\exists x)(F(x) \wedge k|x)$

avec

$$F(x) := \left[\begin{array}{l} (\wedge_i a_i \leq x) \wedge \\ (\wedge_i x \leq b_i) \wedge \\ (\wedge_i \gamma_i | x + c_i) \wedge \\ (\wedge_i \delta_i \nmid x + d_i) \end{array} \right]$$

- $(\exists x)F(x) \leftrightarrow \bigvee_{j=0}^{\delta-1} F_{-\infty}(j) \vee \bigvee_{j=0}^{\delta-1} \bigvee_{a_i} F(a_i + j)$

où δ est le ppcm des δ_i, γ_i .

Littérature : Algorithme de D. C. Cooper

Arithmétique naturelle générale

- On est dans

$$(\mathbf{N}, =, +, \times, 0, 1)$$

- Axiomatisation incomplète de Peano

Propriétés des constantes

$$(1) \quad \neg 0 = 1$$

Propriétés de l'addition

$$(2) \quad x + 1 = y + 1 \rightarrow x = y$$

$$(3) \quad x + 0 = x$$

$$(4) \quad x + (y + 1) = (x + y) + 1$$

Propriétés de la multiplication

$$(5) \quad x \times 0 = 0$$

$$(6) \quad x \times (y + 1) = (x \times y) + x$$

Schéma d'induction

$$(7_p) \quad p(0) \wedge (\forall x)(p(x) \rightarrow p(x + 1)) \rightarrow (\forall x)p(x)$$

- En enlevant (5) et (6) on obtient une axiomatisation complète de

$$(\mathbf{N}, =, +, 0, 1),$$

dite *arithmétique de Presburger*.

Corps ordonné des réels

- On est dans

$$(R, \leq, +, -, \times, 0, 1)$$

- Axiomatisation

Propriétés des constantes

$$(1) \neg 0 = 1$$

Propriétés de l'addition

$$(2) x + (y + z) = (x + y) + z$$

$$(3) x + y = y + x$$

$$(4) x + 0 = x$$

$$(5) x + (-x) = 0$$

Propriétés de la multiplication

$$(6) x \times (y \times z) = (x \times y) \times z$$

$$(7) x \times y = y \times x$$

$$(8) x \times 1 = x$$

$$(9) \neg x = 0 \rightarrow (\exists y)(x \times y = 1)$$

$$(10) x \times y = 0 \rightarrow (x = 0 \vee y = 0)$$

Interférences entre addition et multiplication

$$(11) x \times (y + z) = (x \times y) + (x \times z)$$

$$(12) (\exists y)(y^2 = x \vee y^2 + x = 0)$$

$$(13_n) x_0^2 + x_1^2 + \dots + x_n^2 = 0 \rightarrow x_0 = 0 \wedge x_1 = 0 \wedge \dots \wedge x_n = 0$$

$$(14_n) (\exists y)(x_{2n+1} \times y^{2n+1} + x_{2n} \times y^{2n} + \dots + x_0 = 0) \vee x_{2n+1} = 0$$

Introduction de l'ordre

$$(14) x \leq y \leftrightarrow (\exists z)(x + z^2 = y)$$

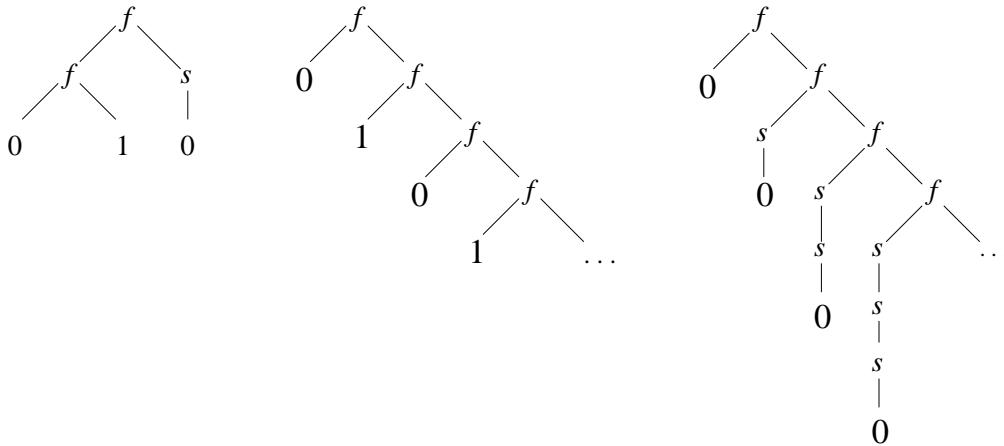
- Littérature : Alfred Tarsky, Georges Collins, James Renegar, Hohn Hong, ...

Arbres munis de constructeurs

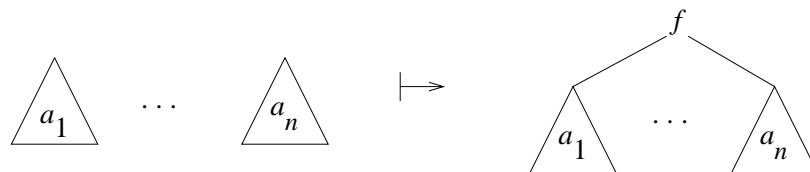
On est dans

$(A, =, F)$

Exemples d'arbres



Opération de construction associée à chaque $f \in F$

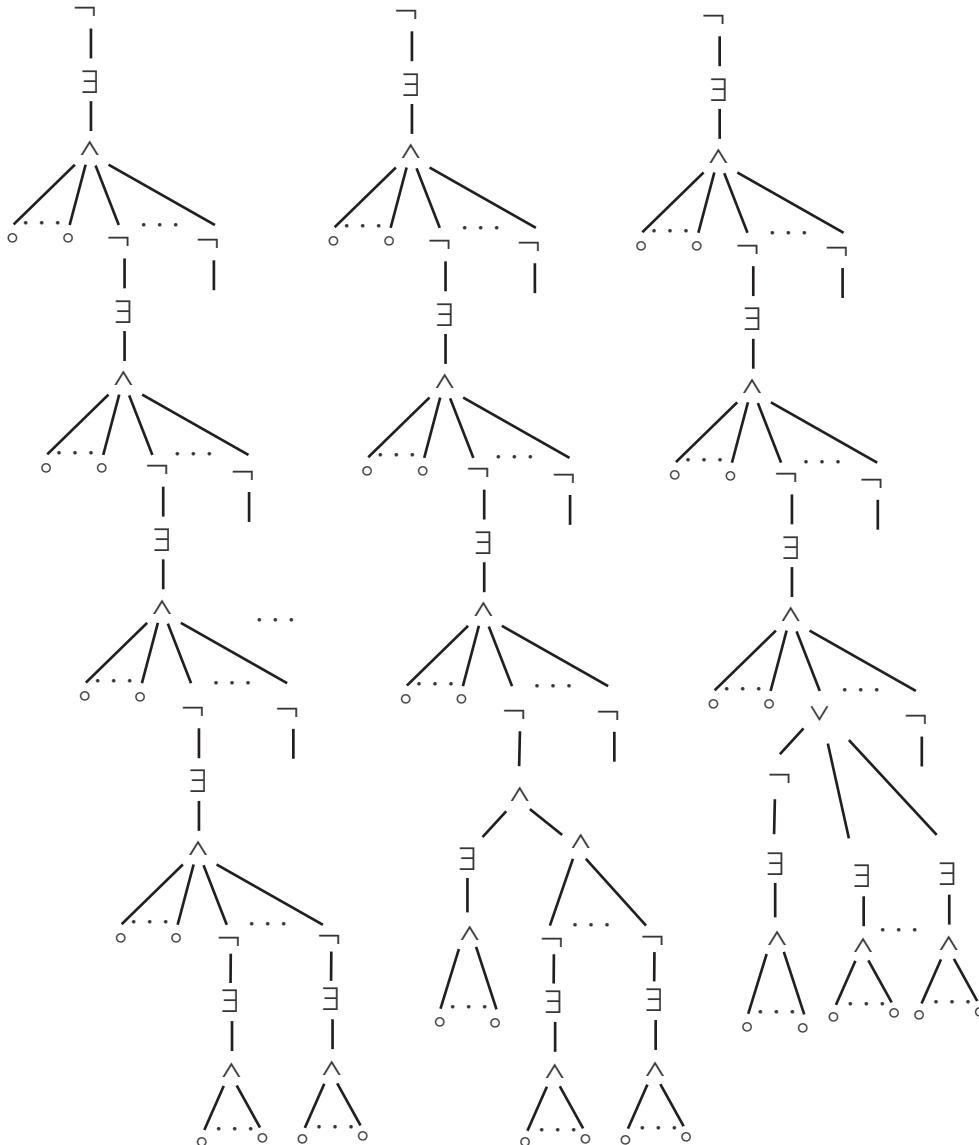


Axiomatisation (de Michael Maher)

- (1) $(\forall \bar{x})(\forall \bar{y})(\neg f(\bar{x}) = g(\bar{y}))$, avec f, g distincts
- (2) $(\forall \bar{x})(\forall \bar{y})(f(\bar{x}) = f(\bar{y}) \rightarrow \bigwedge_i x_i = y_i)$
- (3_t) $(\forall \bar{x})(\exists! \bar{y})(\bigwedge_i y_i = t_i(\bar{x}, \bar{y}))$

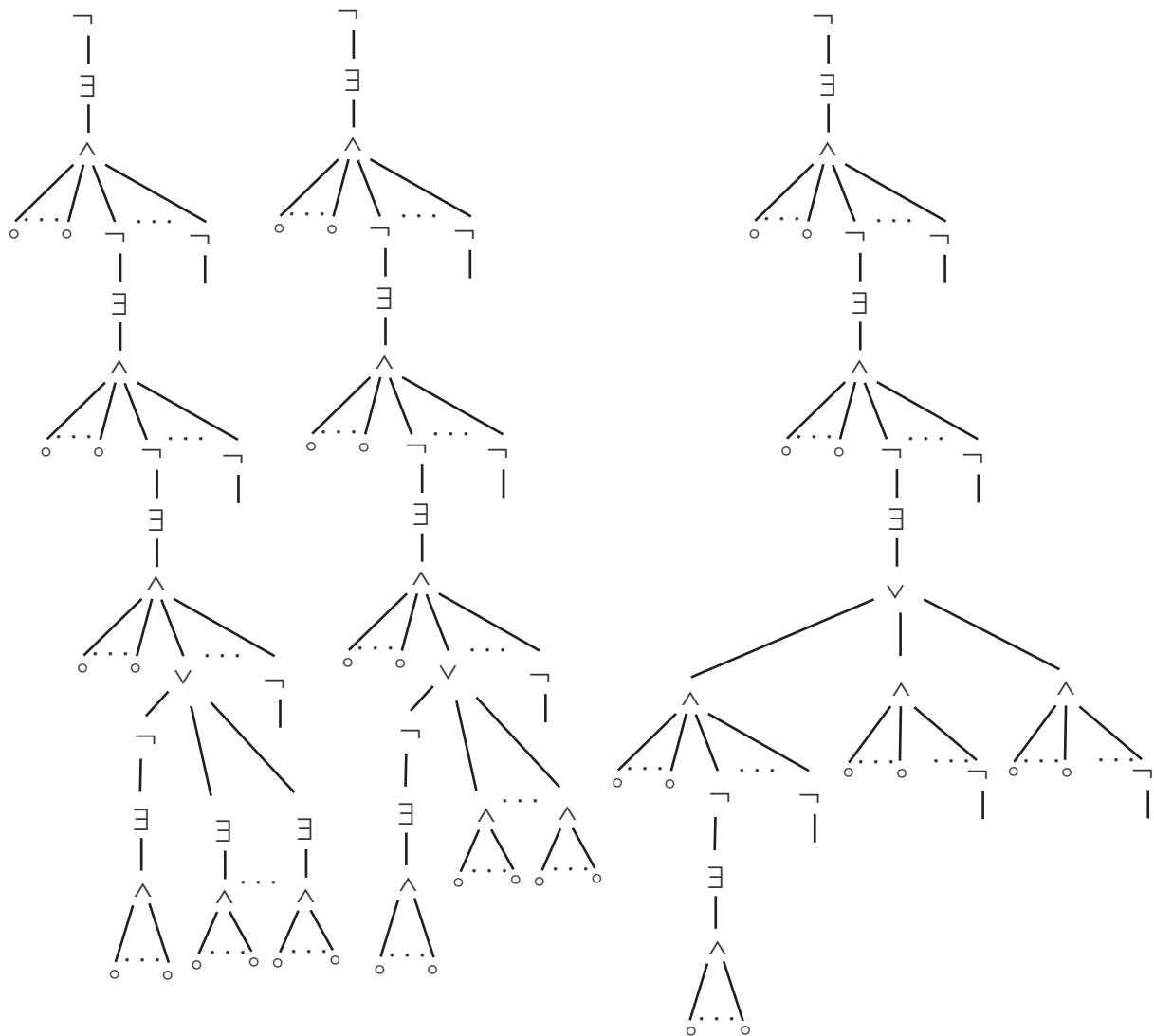
Note : dans (3_t) les y_i sont des variables distinctes et $t_i(\bar{x}, \bar{y})$ est un terme formé d'un symbole fonctionnel suivi de variables prises dans \bar{x} et \bar{y} .

Autre élimination de quantificateurs, 1

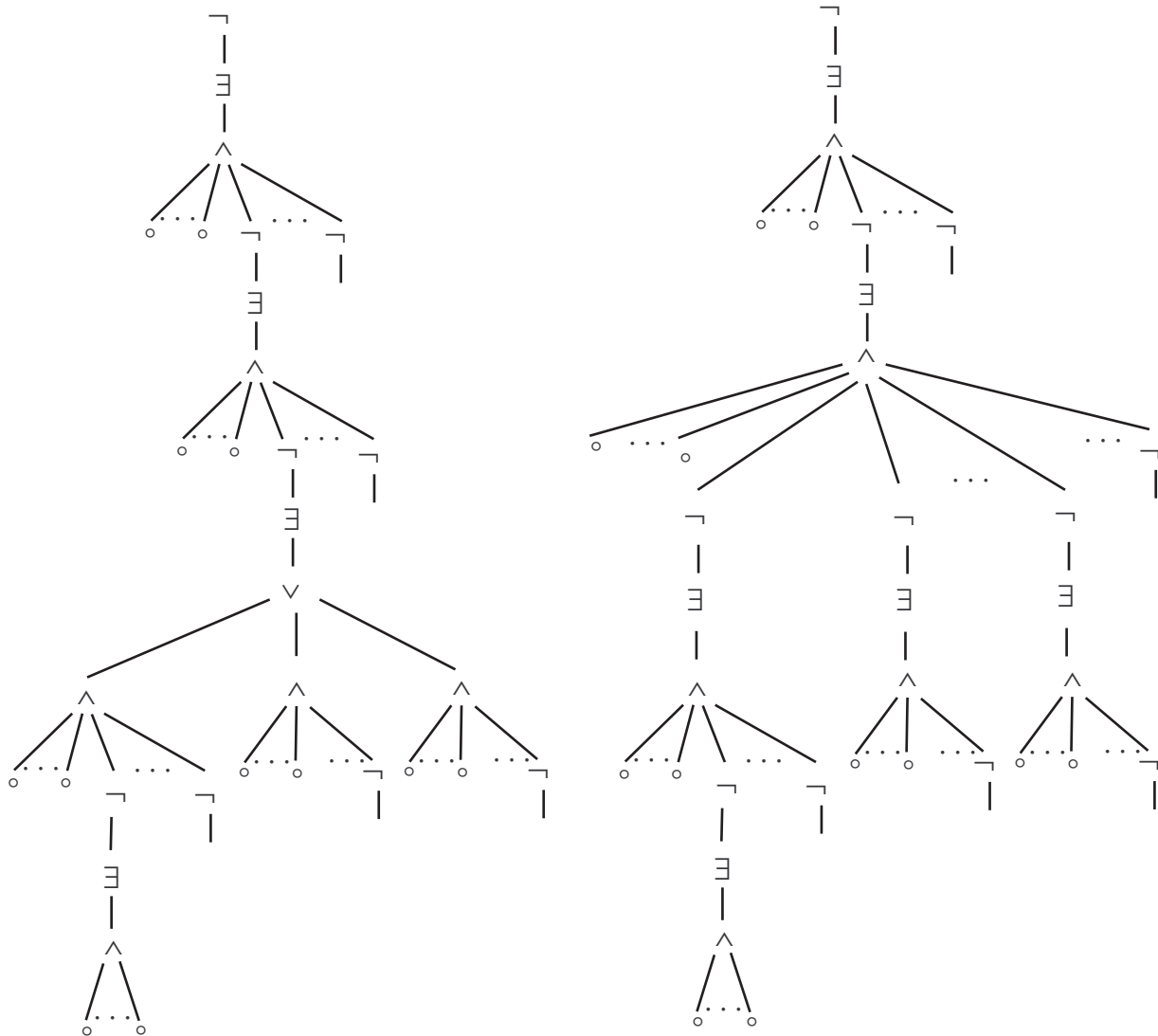


Littérature : Thi-Bich-Hanh Dao , JFPLC2000.

Autre élimination de quantificateurs, 2



Autre élimination de quantificateurs, 3

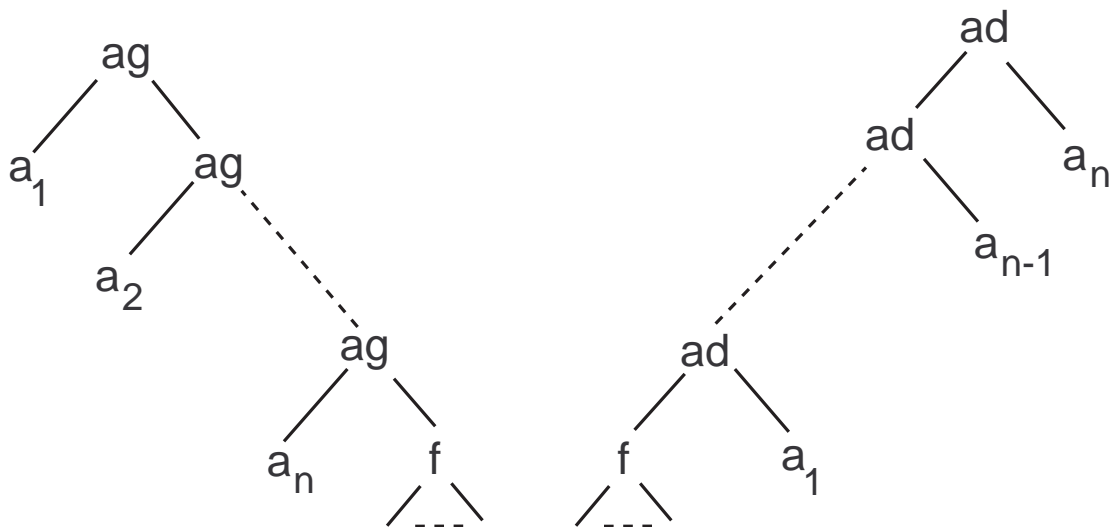


Queues finies ou infinies imbriquées

- On est dans

$$(\mathbf{A}, =, F \cup \{ag, ad\})$$

- Exemples de queues

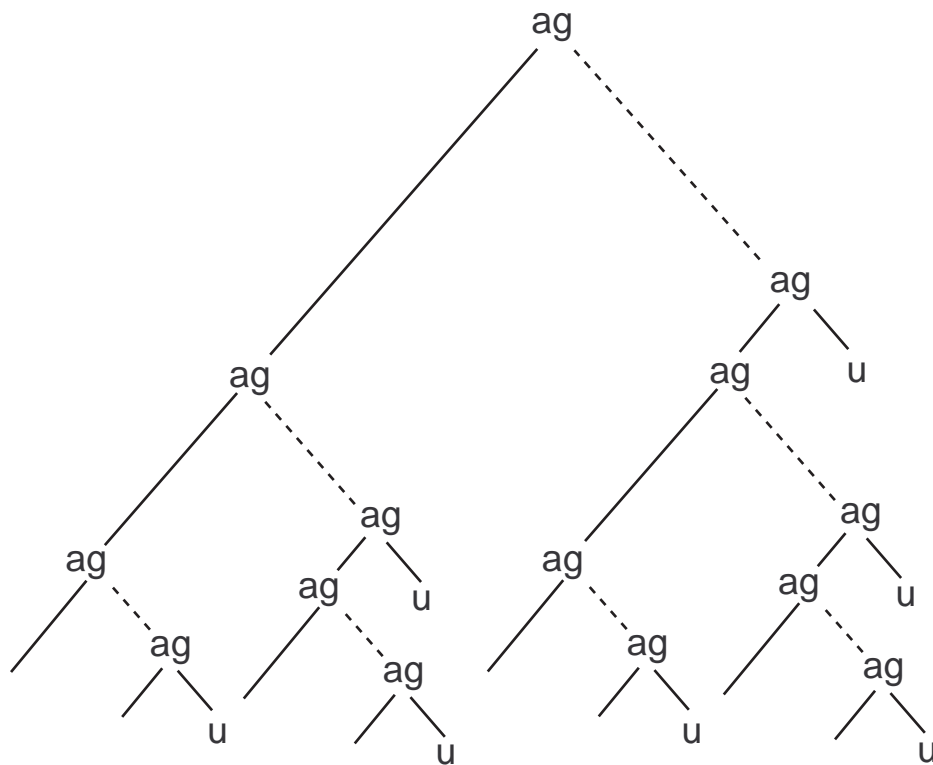


Queues finies ou infinies imbriquées, suite 1

Exemple de contrainte. Un arbre x tel que

$$ag(x, x) = ad(x, x)$$

est de la forme



Queues finies ou infinies imbriquées, suite 2

- On écrit

$$\begin{aligned}x[u] & \text{ pour } ad(x, u), \\ [u]x & \text{ pour } ag(u, x)\end{aligned}$$

- Essai d'axiomatisation

(1) $f(\bar{x}) \neq g(\bar{y})$, avec f, g distincts entre eux et de ag, ad

(2) $f(\bar{x}) = f(\bar{y}) \rightarrow \wedge_i x_i = y_i$

(3_t) $(\exists! \bar{y})(\wedge_i y_i = t_i(\bar{x}, \bar{y}))$

(4) $([u]x)[v] = [u](x[v])$

(5) $\neg(\exists u)(\exists v)(x = [u]v \vee x = [v]u) \rightarrow [y]x = x[y]$

Note : dans le système d'équations de (3_t) on exclu un bouclage traversant uniquement les symboles ad, ag .

Littérature : N. Bjørner, Tatiana Rybina et Andrei Voronkov.

Propagation de contraintes locales

- Résolution locale et propagation sur les \wedge

$$r(x_1, \dots, x_n) \wedge x_1 \in a_1 \wedge \dots \wedge x_n \in a_n$$

$$\leftrightarrow$$

$$r(x_1, \dots, x_n) \wedge x_1 \in b_1 \wedge \dots \wedge x_n \in b_n$$

- Propagation au delà des négations et quantifications

$$p \wedge \neg(\exists x_1) \dots (\exists x_n)(q)$$

$$\leftrightarrow$$

$$p \wedge \neg(\exists x_1) \dots (\exists x_n)(p \wedge q)$$

- En particulier

$$y \in a \wedge \neg(\exists x_1) \dots (\exists x_n)(y \in b \wedge q)$$

$$\leftrightarrow$$

$$y \in a \wedge \neg(\exists x_1) \dots (\exists x_n)(y \in a \cap b \wedge q)$$

Programmer avec des contraintes

- Multiplication orthodoxe mais limitée. Pour tout $n \geq 0$, on définit $fois_n(x, y, z)$ comme $|x| < n$ et $z = x \times y$, par

$$fois_0(x, y, z) := \text{faux}$$

$$fois_{n+1}(x, y, z) := \left(\begin{array}{l} (x = 0 \wedge z = 0) \vee \\ (x < 0 \wedge fois_n(x + 1, y, z + y)) \vee \\ (x > 0 \wedge fois_n(x - 1, y, z - y)) \end{array} \right)$$

- Multiplication non orthodoxe mais illimitée

$$fois(x, y, z) := \left(\begin{array}{l} (x = 0 \wedge z = 0) \vee \\ (x < 0 \wedge fois(x + 1, y, z + y)) \vee \\ (x > 0 \wedge fois(x - 1, y, z - y)) \end{array} \right)$$

Programmer avec des contraintes, suite 1

- Orthodoxe

$conc_0(x, y, z) := \text{faux}$

$$conc_{n+1}(x, y, z) := \left((x = \text{vide} \wedge y = z) \vee \begin{array}{l} (\exists e)(\exists x')(\exists z') \left(\begin{array}{l} x = \text{liste}(e, x') \wedge \\ z = \text{liste}(e, z') \wedge \\ conc_n(x', y, z') \end{array} \right) \end{array} \right)$$

- Non orthodoxe

$$conc(x, y, z) := \left((x = \text{vide} \wedge y = z) \vee \begin{array}{l} (\exists e)(\exists x')(\exists z') \left(\begin{array}{l} x = \text{liste}(e, x') \wedge \\ z = \text{liste}(e, z') \wedge \\ conc(x', y, z') \end{array} \right) \end{array} \right)$$

Programmer avec des contraintes, suite 2

- Orthodoxe

$$\text{gagnant}_0(x) \leftrightarrow \text{faux}$$

$$\text{gagnant}_{k+1}(x) \leftrightarrow \left[\begin{array}{l} \exists y \text{ coup}(x, y) \wedge \neg(\\ \exists z \text{ coup}(y, z) \wedge \neg(\\ \text{gagnant}_k(z)) \end{array} \right]$$

- Non orthodoxe

$$\text{gagnant}(x) \leftrightarrow \left[\begin{array}{l} \exists y \text{ coup}(x, y) \wedge \neg(\\ \exists z \text{ coup}(y, z) \wedge \neg(\\ \text{gagnant}(z)) \end{array} \right]$$

Résolution de Monsieur P et Madame S, 1

• Première étape. On considère le problème de Monsieur P et Madame S. On ne tient compte que de la deuxième phrase du dialogue et on s'intéresse à l'ensemble des valeurs de $s = x + y$ que Madame S pourrait connaître, c'est-à-dire telles que $Q_2(s)$. En sachant que 53 est le plus petit nombre premier qui est plus grand que $100/2$, on montre d'abord que $Q_2(s)$ entraîne $s < 53 + 2$, c'est-à-dire,

$$s < 55$$

En effet si $s \geq 55$, il existerait $x \in 2..100$ et $y \in 2..100$ avec $s = x + y$ et $x = 53$. Il ne peut exister $x' \in 2..100$ et $y' \in 2..100$ tels que $\{x', y'\} \neq \{x, y\}$ et $x'y' = xy$. Si c'était le cas l'un des nombres x', y' serait un multiple de 53 et serait donc strictement plus grand que 100. Donc on n'aurait pas $Q_2(s)$.

Résolution de Monsieur P et Madame S, 2

• Deuxième étape. Toujours en ne tenant compte que de la deuxième phrase du dialogue, on montre que la somme s que Madame P connaît ne peut être la somme de deux nombres premiers.

Sachant que $s \in 4..54$ et que les nombres premiers inférieures à 57 sont 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, on en déduit que

$$s \in \{11, 17, 23, 27, 29, 35, 37, 41, 47, 51, 53\}$$

et donc le couple (s, p) que l'on cherche figure dans la liste :

- (11, {18, 24, 28, 30})
- (17, {30, 42, 52, 60, 66, 70, 72})
- (23, {42, 60, 76, 90, 102, 112, 120, 126, 130, 132})
- (27, {50, 72, 92, 110, 126, 140, 152, 162, 170, 176, 180, 182})
- (29, {54, 78, 100, 120, 138, 154, 168, 180, 190, 198, 204, 208, 210})
- (35, {66, 96, 124, 150, 174, 196, 216, 234, 250, 264, 276, 286, 294, 300, 304, 306})
- (37, {70, 102, 132, 160, 186, 210, 232, 252, 270, 286, 300, 312, 322, 330, 336, 340, 342})
- (41, {78, 114, 148, 180, 210, 238, 264, 288, 310, 330, 348, 364, 378, 390, 400, 408, 414, 418, 420})
- (47, {90, 132, 172, 210, 246, 280, 312, 342, 370, 396, 420, 442, 462, 480, 496, 510, 522, 532, 540, 546, 550, 552})
- (51, {98, 144, 188, 230, 270, 308, 344, 378, 410, 440, 468, 494, 518, 540, 560, 578, 594, 608, 620, 630, 638, 644, 648, 650})
- (53, {102, 150, 196, 240, 282, 322, 360, 396, 430, 462, 492, 520, 546, 570, 592, 612, 630, 646, 660, 672, 682, 690, 696, 700, 702})

Ici on a mis dans un même ensemble les valeurs p correspondant à un même s .

Résolution de Monsieur P et Madame S, 3

- Troisième étape. Si l'on tient compte de la troisième phrase du dialogue alors cette liste devient quelque chose de la forme :

$$\begin{aligned} & (11, \{18, 24, 28\}) \\ & (17, \{52\}) \\ & (23, \{112, 120, \dots\}) \\ & (27, \{50, 92, \dots\}) \\ & (29, \{54, 100, \dots\}) \\ & (35, \{96, 124, \dots\}) \\ & (37, \{102, 160, \dots\}) \\ & (41, \{114, 148, \dots\}) \\ & (47, \{172, 246, \dots\}) \\ & (51, \{98, 144, \dots\}) \\ & (53, \{240, 282, \dots\}) \end{aligned}$$

- Quatrième étape. En tenant compte de la quatrième phrase du dialogue on trouve $(p, s) = (17, 52)$ et donc $\{x, y\} = \{4, 13\}$.