Décidabilité de l'arithmétique de Presburger

Réf: Carton: LFCC

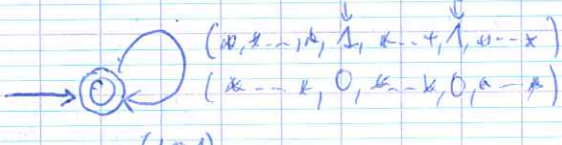
Th: La théorie au premier ordre des entiers munis de l'addition est décidable

Démo • Soit φ une formule close sous forme prénexe : elle s'écrit $\varphi = Q_1 x_1 \dots Q_n x_n \psi$
Pour tout entier $k \in [\![1;n]\!]$, on définit $\varphi_k = Q_{k+1} x_{k+1} \dots Q_n x_n \psi$ avec comme convention $\varphi_0 = \varphi$, $\varphi_n = \psi$. Pour $k \in [\![0;n]\!]$, φ_k a donc k variables libres et s'écrit $\varphi_k(x_1, \dots, x_k)$. On va montrer par récurrence sur $n-k$ que l'ensemble des k-uplets qui satisfont φ_k est un langage rationnel.

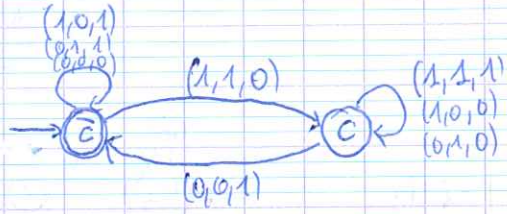
• On définit donc un codage sur les k-uplet : chaque entier est écrit en binaire et un k-uplet est donc un mot sur l'alphabet $\Sigma_k = \{0,1\}^k$ où on force les entiers à avoir la même longueur en ajoutant des 0 si besoin. à la fin.
On définit donc $X_k = \{(n_1, \dots, n_k) \mid \varphi_k(n_1, \dots, n_k) \text{ est vraie}\}$ et on construit un automate A_k qui reconnait les écritures sur Σ_k des éléments de X_k. Ainsi l'automate A_0 accepte au moins un mot ssi φ est vraie.

• Construction de l'automate A_n qui accepte les n-uplets qui satisfont φ : comme φ est une combinaison booléenne de formules atomiques de la forme $x_i = x_j$ ou $x_i + x_j = x_k$, et que Rat est stable par les opérations booléenne, il suffit de construire un automate pour chacune des formule atomique.

* $x_i = x_j$:



* $x_i + x_j = x_k$:



où c est l'état avec retenue.

Ainsi, on a bien construit un automate A_n qui reconnait X_n

• Passage de A_{k+1} à A_k : on écrit $\varphi_k = Q_{k+1} x_{k+1} \varphi_{k+1}$ et on a deux cas suivant si Q_{k+1} est universel ou existentiel

* Si $Q_{k+1} = \forall$, on se ramène au cas $\exists$ en écrivant $\varphi_k = \neg \exists x_{k+1} \neg \varphi_{k+1}$ en utilisant que Rat est stable par complémentation

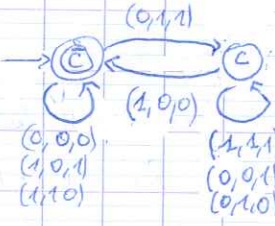
* Si $Q_{k+1} = \exists$: l'automate A_k est obtenu en oubliant la dernière composante de l'alphabet, càd en projetant Σ_{k+1} sur Σ_k via $\pi_k(b_1, \dots, b_k, b_{k+1}) = (b_1, \dots, b_k)$. L'automate A_k a les même états que A_{k+1}, les transitions sont les mêmes que A_k mais étiquetés par $\pi_k(a)$ au lieu de a, les états initiaux sont les mêmes, les états finaux sont tous ceux accessibles depuis les états finaux de A_k en lisant $(0, \dots, 0)$. (cas où x_{k+1} serait plus long)

Ainsi, on a bien A_k accepte $(x_1, \dots, x_k)$ ssi il existe un chemin d'un état q_0 initial vers un état final q_f ssi il existe un chemin dans A_{k+1} avec x_{k+1} qui correspond à la composante oubliée

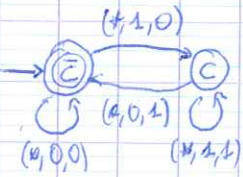
ie A_k accepte $(x_1, \dots, x_k) \Leftrightarrow \exists x_{k+1}$ tq A_{k+1} accepte $(x_1, \dots, x_{k+1})$ □

Exemple : $x = 0[3] \Leftrightarrow \exists y \exists z,\ x = y+z \wedge z = y+y$

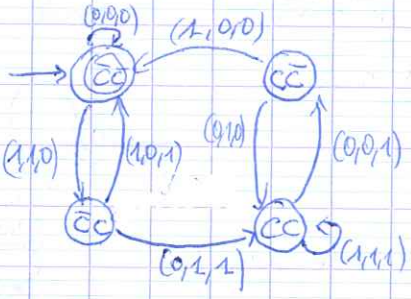
* $x = y+z$:



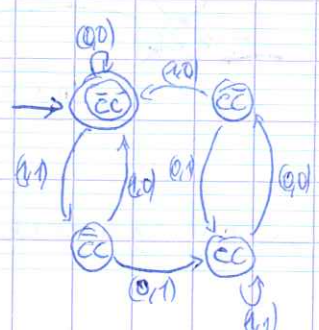
* $z = y+y$:



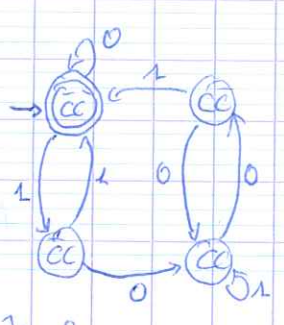
* $x = y+z \wedge z = y+y$:



* $\exists z, \dots$:



* $\exists y \exists z$:



multiples :