Rapport de stage

Représentation et algorithmique des ensembles semi-linéaires

Fabrice Chevalier et Jérémie Chalopin

10 Juillet 2001

stage réalisé au LSV - ENS CACHAN
sous la direction d'Alain FINKEL

Introduction

Généralités et motivation

Quelques échecs retentissants (comme par exemple la panne du réseau téléphonique aux USA en 1989 ou la destruction du premier exemplaire de la fusée Ariane 5 en 1996) ont achevé de convaincre l'impérieuse nécessité de vérifier certains logiciels critiques.
Le Model Checking est actuellement une des techniques les plus répandues pour vérifier ces applications critiques. La complexité de ces applications ne permet pas toujours de les modéliser par un système fini ; il est donc important de pouvoir représenter et manipuler des ensembles infinis.
En particulier il est important de pouvoir réaliser des opérations ensemblistes sur ces ensembles infinis comme le montre le pseudo-algorithme suivant :

S_0 : ensemble d'états initiaux
$succ$: une fonction successeur
Faire
  $S' \leftarrow S$
  $S \leftarrow S \cup succ(S)$
tant que $\neg(S \subseteq S')$
Renvoyer S

Cet algorithme calcule l'ensemble des états accessibles d'un système. Il peut être appliqué à la vérification par exemple en testant si parmi les états accessibles aucun n'est un état «mauvais» (par exemple l'état «fusée en panne»).

Les ensembles semi-linéaires sont des ensembles infinis de vecteurs d'entiers ayant une représentation finie et des propriétés de clôture intéressantes : union, intersection, complémentaire ... Ils permettent donc de modéliser des systèmes ayant une infinité d'états et d'élaborer des algorithmes intéressants les manipulant (comme le calcul des états accessibles d'un système).

Objectifs

Le but de ce stage était de faire une synthèse des articles publiés sur les ensembles semi-linéaires. La diversité des articles étudiés reflètent l'histoire de l'étude de ces ensembles : découverts dans les années 60, Ginsburg et Spanier ont établi leur principales propriétés mathématiques en relation avec l'étude des langages algébriques. Avec le Model Checking, on «découvre» l'utilité des ensembles semi-linéaires et on cherche des algorithmes efficaces d'où les travaux récents de Huynh(82), Boudet et Common(96), Wolper et Boigelot(2000).

Nous avons mis l'accent sur les opérations réalisables sur ces ensembles et sur leur différents modes de représentation (expressions rationnelles, bases et périodes, formules de Presburger, automates).
Nous avons présenté les principales preuves des propriétés de clôture et des passages d'une représentation à une autre. Le plus souvent la complexité de ces constructions n'était pas explicitement donnée, nous l'avons donc évaluée. Nous avons parfois proposé des algorithmes plus efficaces que ceux trouvés dans la littérature.
Remarque : Toutes les résultats de complexité donnés dans ce rapport sont des complexités d'algorithmes, et non la complexité des problèmes.

Structure du rapport

Nous avons structuré notre étude selon les différents modes de représentations des ensembles semi-linéaires. A chaque fois que nous avons introduit une nouvelle représentation, nous avons explicité ses rapports avec les représentations déjà étudiées, en particulier nous avons donné les constructions permettant de passer d'une représentation à une autre.

Sur la lecture de ce rapport

Voici quelques remarques pouvant guider le lecteur dans sa lecture du rapport :

- La preuve de la stabilité par intersection des ensembles semi-linéaires est assez intéressante car elle illustre bien, sans être excessivement compliquée les principales idées utilisées dans les preuves sur les ensembles semi-linéaires.
- La preuve de la stabilité par complémentaire par contre est très technique et peu agréable.
- La partie 2 sur l'arithmétique de Presburger est assez élégante et c'est surtout un prérequis pour la compréhension de l'utilisation des automates.
- La construction d'un automate pour une équation est très astucieuse et esthétique. Par sa simplicité et son utilité pratique, c'est peut-être le point le plus important.
- Les éléments originaux élaborés dans ce rapport et qu'on ne trouve donc pas dans la littérature sont les passages des rationnels aux formules de Presburger(§3.2), des rationnels aux automates (§4.6) et des bases-périodes aux automates (§4.5)

Table des matières

1 Bases et périodes

Dans cette partie, on définit les ensembles semi-linéaires sur $\mathbb{N}^k$ et leur représentation sous forme de bases et de périodes. On donne ensuite les premières propriétés de clôture de ces ensembles. Enfin, on démontre les deux principaux résultats de cette partie : la stabilité des ensembles semi-linéaires par intersection et complémentaire.

1.1 Définitions et premières propriétés

Définition 1 *Pour toutes parties finies C, P de $\mathbb{N}^k$ on définit $L(C,P) = \{c + \sum_{i=1}^{n} \lambda_i p_i \ ; \ c \in C, \ \lambda_i \in \mathbb{N}\}$ où $P = \{p_1, \ ..., \ p_n\}$.*
$L(\{c\}, P)$ est noté $L(c,P)$, ou $c + P^*$.

Définition 2 *Une partie de $\mathbb{N}^k$ est dite linéaire si elle est de la forme $u + U^* = L(u,U)$.*
Un ensemble semi-linéaire est une union finie de parties linéaires : $\bigcup_{i=1}^{n} (u_i + U_i^*) = \bigcup_{i=1}^{n} L(u_i, U_i)$

Exemples :

- L'ensemble $E = \{(x,y) \in \mathbb{N}^2 \ ; \ x \geq 1\}$ est un ensemble linéaire donc semi-linéaire de $\mathbb{N}^2$: en effet $E = (1,0) + \{(0,1), \ (1,0)\}^*$.
- L'ensemble $F = \{2^n; \ n \in \mathbb{N}\}$ n'est pas un ensemble semi-linéaire de $\mathbb{N}$: en effet supposons cet ensemble semi-linéaire, alors comme celui-ci est infini, il contient au moins un ensemble linéaire de la forme $u+V^*$ tel que V contient un nombre non nul v, donc en particulier il contient $u, u+v, u+2v$, on aurait alors $\exists i \ \exists j \ \exists k \ / \ u = 2^i, \ u + v = 2^j, \ u + 2v = 2^k$, on a $2^i + 2^k = 2^{j+1}$ d'où $i = j = k$ d'où $v = 0$ ce qui est absurde car on a supposé v non nul.

Proposition 1 *Les ensembles semi-linéaires sont clos par produit cartésien, union, somme, étoile et application linéaire.*

Preuve : La clôture par union est évidente.
Les semi-linéaires sont stables par produit cartésien :

$$\begin{aligned}
\left(\bigcup_{i=1}^{n} u_i + U_i^*\right) \times \left(\bigcup_{j=1}^{m} v_j + V_j^*\right) &= \bigcup_{i=1}^{n}\bigcup_{j=1}^{m} (u_i + U_i^*) \times (v_j + V_j^*) \\
&= \bigcup_{i=1}^{n}\bigcup_{j=1}^{m} (u_i \times v_j) + ((U_i \times \{0\})^* + (\{0\} \times V_j)^*) \\
&= \bigcup_{i=1}^{n}\bigcup_{j=1}^{m} (u_i \times v_j) + (U_i \times \{0\} \cup \{0\} \times V_j)^*
\end{aligned}$$

Les semi-linéaires sont stables par somme :

$$\bigcup_i (u_i + U_i^*) + \bigcup_j (v_j + V_j^*) = \bigcup_{i,j} (u_i + v_j + U_i^* + V_j^*) = \bigcup_{i,j} (u_i + v_j + (U_i \cup V_j)^*)$$

Les semi-linéaires sont stables par étoile :

$$(\bigcup_{i=1}^{n} (u_i + U_i^*))^* = \sum_{i=1}^{n} (u_i + U_i^*)^* = \sum_{i=1}^{n} \{0\} \cup (u_i + (\{u_i\} \cup U_i)^*)$$

Et enfin la stabilité par application linéaire : soit f une application linéaire, alors

$$f(\bigcup_i (u_i + U_i^*)) = \bigcup_i (f(u_i) + f(U_i)^*)$$

□

Complexité : L'union et l'application linéaire sont des opération réalisées en temps et en espace linéaires. La somme et le produit cartésien se font en temps et en espace quadratiques.
Par contre, l'étoile requiert un temps et un espace exponentiels :

$$(\sum_i u_i + U_i^*)^* = \sum_i \{0\} \cup (u_i + (\{u_i\} \cup U_i)^*) = \bigcup_{B \subseteq \{1..n\}} \Sigma_{i \in B} u_i + (\cup_{i \in B}(\{u_i\} \cup U_i))^*$$

1.2 La stabilité par intersection

Pour montrer la stabilité par intersection, on montre tout d'abord que l'ensemble des solutions d'un système d'équations linéaires à coefficients dans $\mathbb{Z}$ est semi-linéaire ; puis on exprime l'intersection de deux ensembles semi-linéaires en fonction d'un tel système.
Un corollaire important de la stabilité par intersection des ensembles semi-linéaires est leur stabilité par morphisme inverse, résultat qui nous servira pour montrer la stabilité par complémentaire.

Lemme 1 *Tout ensemble de $\mathbb{N}^k$ formé d'éléments deux à deux incomparables est fini.*

Preuve : Supposons qu'une suite infinie d'éléments deux à deux incomparables existe, alors en extrayant tour à tour selon chaque coordonnée on pourrait extraire une sous-suite croissante, ce qui contredit le fait que les éléments soient incomparables. □

En considérant l'ordre partiel usuel sur $\mathbb{N}^k$ et pour $A \subseteq \mathbb{N}^k$ on définit
$min(A) = \{x \in A; \forall y \in A, \neg(y < x)\}$

Proposition 2 *$min(A)$ est fini.*

Preuve : $min(A)$ est constitué d'éléments deux à deux incomparables, donc est fini d'après le lemme. □

Théorème 1 ([Huy82]) *Soient $\mathbf{a_1}, \ldots, \mathbf{a_n}, \mathbf{b} \in \mathbb{Z}^m$, et soit S l'ensemble des solutions dans $\mathbb{N}^n$ de l'équation*

$$x_1\mathbf{a_1} + \cdots + x_n\mathbf{a_n} = \mathbf{b}$$

Alors $\exists d\ \forall c \in min(S), taille(c) \leq d \cdot 2^{taille(\mathbf{a_1},\ldots,\mathbf{a_n},\mathbf{b})^2}$

Preuve : Voir [Huy82] □

Proposition 3 ([Reu89]) *Soit $\mathbf{a_1}, \ldots, \mathbf{a_n}, \mathbf{b} \in \mathbb{Z}^m$, et soit S l'ensemble des solutions dans $\mathbb{N}^n$ de l'équation $x_1\mathbf{a_1} + \cdots + x_n\mathbf{a_n} = \mathbf{b}$*
Alors S est semi-linéaire.

Preuve : Soit S_0 l'ensemble des solutions de l'équation $x_1\mathbf{a_1} + \cdots + x_n\mathbf{a_n} = \mathbf{0}$
On note $U = min(S)$ et $V = min(S_0 \setminus \{0\})$, on veut montrer que $S = U + V^*$.

Montrons tout d'abord que $S_0 = V^*$. L'inclusion $V^* \subseteq S_0$ est évidente.
On montre l'inclusion inverse par récurrence sur la taille des éléments : soit $\mathbf{x} \in S_0$, si $\mathbf{x} = \mathbf{0}$ ou $\mathbf{x} \in V$ alors $\mathbf{x} \in V^*$; sinon $\exists \mathbf{y} \in V$; $\mathbf{y} < \mathbf{x}$ alors par hypothèse de récurrence $\mathbf{x} - \mathbf{y},\ \mathbf{y} \in V^*$ et donc $\mathbf{x} \in V^*$.
Montrons maintenant que $S = U + V^*$.
$U + V^* \subseteq S$: un élément de U est solution de $\mathbf{a_1}x1 + \cdots + \mathbf{a_n}x_n = \mathbf{b}$ et un élément de V^* est solution de $\mathbf{a_1}x1 + \cdots + \mathbf{a_n}x_n = \mathbf{0}$ donc un élément de $U + V^*$ est dans S.
Montrons que $S \subseteq U + V^*$: si $\mathbf{x} \in U$ alors $\mathbf{x} \in U + V^*$, sinon $\exists \mathbf{y} \in U,\ \mathbf{y} < \mathbf{x}$ alors $\mathbf{x} - \mathbf{y}$ est solution de $\mathbf{a_1}x1 + \cdots + \mathbf{a_n}x_n = \mathbf{0}$ donc $\mathbf{x} - \mathbf{y} \in S_0 = V^*$, d'où $\mathbf{x} \in U + V^*$. □

Complexité : D'après le théorème 1, pour trouver $min(S)$ et $min(S_0)$, il suffit de tester tous les éléments jusqu'à $d \cdot 2^{taille(\mathbf{a_1},\ldots,\mathbf{a_n},\mathbf{b})^2}$ et de prendre les minimaux. Le coût est donc exponentiel en temps en fonction de la taille du système.
Le coût en espace est aussi exponentiel : la taille des bases et des périodes obtenues est bien polynomiale mais il peut y avoir un nombre exponentiel de telles bases et périodes.

Notation : Pour deux vecteurs $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{N}^n$ et $\mathbf{y} = (y_1, \ldots, y_m) \in \mathbb{N}^m$, on note $\mathbf{x} \times \mathbf{y}$ le vecteur de N^{n+m} égal à $(x_1, \ldots, x_n, y_1, \ldots, y_m)$.

Théorème 2 ([GS64]) *Les ensembles semi-linéaires sont clos par intersection.*

Preuve : il suffit de montrer que l'intersection de 2 ensembles linéaires est semi-linéaire.
Soit donc $U, V \subseteq \mathbb{N}^k$, $U = \mathbf{u} + \{\mathbf{u_1}, \ldots, \mathbf{u_n}\}^*$, $V = \mathbf{v} + \{\mathbf{v_1}, \ldots, \mathbf{v_m}\}^*$.

Soit $S = \{\mathbf{x} \times \mathbf{y} / \mathbf{u} + \Sigma_{i=1}^{n} x_i \mathbf{u_i} = \mathbf{v} + \Sigma_{j=1}^{m} y_j \mathbf{v_j}\}$, d'après la proposition 3 , S est semi-linéaire.
Soit f définie par $f(\mathbf{x} \times \mathbf{y}) = \Sigma x_i \mathbf{u_i}$; par la propriété 1, $\mathbf{u} + f(S)$ est semi-linéaire. Il suffit donc de montrer que $\mathbf{u} + f(S) = U \cap V$; or

$$\begin{aligned}
\mathbf{w} \in \mathbf{u} + f(S) &\iff \exists (\mathbf{x}, \mathbf{y}) \in \mathbb{N}^n \times \mathbb{N}^m / \mathbf{w} = \mathbf{u} + f(\mathbf{x} \times \mathbf{y}), \mathbf{u} + \Sigma_{i=1}^{n} x_i \mathbf{u_i} = \mathbf{v} + \Sigma_{j=1}^{m} y_j \mathbf{v_j} \\
&\iff \exists (\mathbf{x}, \mathbf{y}) \in \mathbb{N}^n \times \mathbb{N}^m / \mathbf{w} = \mathbf{u} + \Sigma_{i=1}^{n} x_i \mathbf{u_i} = \mathbf{v} + \Sigma_{j=1}^{m} y_j \mathbf{v_j} \\
&\iff \mathbf{w} \in U \cap V
\end{aligned}$$

□

Complexité : On a $U \cap V = \mathbf{u} + f(S)$, le coût de l'intersection est donc celui du calcul de S. Or S est l'ensemble des solutions du système d'équations $\mathbf{u} + \Sigma_{i=1}^{n} x_i \mathbf{u_i} = \mathbf{v} + \Sigma_{j=1}^{m} y_j \mathbf{v_j}$, et se calcule d'après la proposition 3 en temps et en espace exponentiel en fonction de la taille du système; la taille du système étant $taille(U, V)$, le coût de l'intersection de deux ensembles semi-linéaires est exponentiel.

Corollaire 3 ([GS64]) *Soit f une application linéaire de $\mathbb{N}^k$ dans $\mathbb{N}^m$ et S un ensemble semi-linéaire de $\mathbb{N}^m$; alors $f^{-1}(S)$ est un ensemble semi-linéaire de $\mathbb{N}^k$.*

Preuve : Soit μ l'application de $\mathbb{N}^k$ dans $\mathbb{N}^k \times \mathbb{N}^m$ définie par $\mu(\mathbf{x}) = \mathbf{x} \times f(\mathbf{x})$. Par la proposition 1 et par le théorème 2, comme μ est linéaire, $\mu(\mathbb{N}^k) \cap (\mathbb{N}^k \times S)$ est semi-linéaire. On considère alors l'application linéaire π de $\mathbb{N}^k \times \mathbb{N}^m$ dans $\mathbb{N}^k$ telle que $\pi(\mathbf{x} \times \mathbf{y}) = \mathbf{x}$, on a $\pi(\mu(\mathbb{N}^k) \cap (\mathbb{N}^k \times S)) = f^{-1}(S)$, donc $f^{-1}(S)$ est semi-linéaire. □

Complexité : Le coût du calcul de $f^{-1}(S)$ est donc celui du calcul de $\pi(\mu(\mathbb{N}^k) \cap (\mathbb{N}^k \times S))$. D'après la proposition 1, le coût du calcul de $\mu(\mathbb{N}^k)$ est quadratique et par conséquent, le coût de $\mu(\mathbb{N}^k) \cap (\mathbb{N}^k \times S)$ est celui de l'intersection, qui par le théorème 2 est exponentiel. Ainsi, puisque le calcul de π se fait en temps et espace quadratiques, calculer $f^{-1}(S)$ a donc un coût exponentiel.

1.3 La stabilité par complémentaire

Dans cette partie, principalement inspirée de [GS64], on veut montrer que le complémentaire d'un ensemble semi-linéaire est lui aussi semi-linéaire. La preuve fournie est constructive mais a malheureusement une complexité très coûteuse, puisque l'algorithme proposé est non élémentaire.
Nous avons essayé de rendre plus claire la démonstration proposée par Ginsburg et Spanier, et nous avons explicité la complexité de la construction.
La démonstration se fait par étapes :

- on montre d'abord que tout ensemble semi-linéaire est égal à une réunion d'ensembles linéaires dont les périodes sont linéairement indépendantes.
- on montre ensuite qu'un ensemble linéaire avec des périodes indépendantes et une constante nulle a un complémentaire qui est semi-linéaire.
- on montre pour finir que le complémentaire d'un ensemble semi-linéaire avec des périodes indépendantes est semi-linéaire ; ce qui nous permettra de conclure.

Lemme 2 ([GS64]) *Tout ensemble semi-linéaire peut effectivement être mis sous la forme d'une union finie d'ensembles linéaires, chacun ayant des périodes linéairement indépendantes.*

Preuve : Il suffit de montrer qu'un ensemble linéaire peut être mis sous la forme d'une union finie d'ensembles linéaires, chacun ayant un ensemble de périodes linéairement indépendantes.

Raisonnons par récurrence sur le nombre de périodes : si l'ensemble linéaire n'a qu'une période, celle-ci est indépendante.

Supposons la propriété vraie pour tout ensemble linéaire avec au plus $m-1$ périodes , $m \geq 2$. Soit $X = L(c, \{p_1, ..., p_m\})$ et supposons que $p_1, ..., p_m$ sont dépendantes. Quitte à renommer $p_1, ..., p_m$, il existe k, $1 \leq k < m$ et $a_1, ..., a_m$ positifs ou nuls tels que $\Sigma_{i=1}^{k} a_i p_i = \Sigma_{i>k} a_i p_i$. Pour tout entier $j > k$ définissons $C_j = \{c + i p_j,\ 0 \leq i \leq a_j - 1\}$ si $a_j \geq 2$ et $C_j = \{c\}$ si $a_j = 0, 1$.

D'autre part si $j > k$ on définit $P_j = \{p_1, ..., p_{j-1}, p_{j+1}, ..., p_m\}$. On considère alors les ensembles semi-linéaires $L(C_j, P_j)$. Par hypothèse de récurrence, comme $card(P_j) \leq m$, il suffit donc de montrer que $X = \bigcup_{j>k} L(C_j, P_j)$.
On a clairement $C_j \subseteq X$ et comme $P_j \subseteq P$ on a $L(C_j, P_j) \subseteq X$ donc $\bigcup_{j>k} L(C_j, P_j) \subseteq X$.
Montrons maintenant que $X \subseteq \bigcup_{j>k} L(C_j, P_j)$: soit $y \in X$, alors il existe $b_1, ..., b_m$ positifs ou nuls tels que $y = c + \Sigma_{i=1}^{m} b_i x_i$. On cherche à montrer que l'on peut supposer qu'il existe $j_0 > k$ tel que $b_{j_0} < a_{j_0}$ dans le but d'avoir $y \in L(C_{j_0}, P_{j_0})$.

Supposons donc que $\forall j > k \ b_j \geq a_j$, alors

$$y = c + \sum_{i \leq k} b_i p_i + \sum_{i>k} b_i p_i + \sum_{i \leq k} a_i p_i - \sum_{i>k} a_i p_i$$

$$y = c + \sum_{i \leq k} (b_i + a_i) p_i + \sum_{i>k} (b_i - a_i) p_i$$

En réitérant le processus, on voit que l'on peut supposer l'existence d'un indice $j_0 > k$ tel que $0 \leq b_{j_0} < a_{j_0}$. On a alors $y = c + b_{j_0} p_{j_0} + \sum_{i \neq j_0} b_i p_i$, on a donc $y \in L(C_{j_0}, P_{j_0})$.

Finalement $X = \bigcup_{j>k} L(C_j, P_j)$ et par hypothèse de récurrence, chaque $L(C_j, P_j)$ peut être mis sous la forme d'une union finie d'ensembles linéaires avec des périodes linéairement indépendantes. □

Complexité : La complexité de cet algorithme est non-élémentaire; en effet, on a $X = \bigcup_{j>k} L(C_j, P_j) = \bigcup_{j>k} \bigcup_{i \in C_j} L(c + ip_j, P_j)$. Combien y a-t-il d'ensembles linéaires dans cette décomposition ? Cela dépend des cardinaux des C_j et donc des a_j. A priori, on ne sait rien sur les a_j et on ne peut donc pas évaluer la complexité.

Néanmoins, si on veut faire une analyse un peu plus fine, on peut utiliser le théorème 1 : comme les a_j sont solution d'un système d'équations linéaires dont la taille est $taille(X)$, on peut choisir les a_j tels que leur valeur soit exponentielle en la taille de X. La complexité reste malheureusement non-élémentaire car on applique l'hypothèse de récurrence à ce nombre exponentiel d'ensembles linéaires. Chaque période supplémentaire dans l'ensemble X lève donc un facteur exponentiel, on ne pourra pas borner la hauteur d'exponentielles par ce moyen.

Lemme 3 ([GS64]) *Soit $X = (x_i)_{i \in \mathbb{N}_n}$ un ensemble de vecteurs indépendants de $\mathbb{N}^n$.*

1. *Il existe un entier strictement positif k_0 tel que*

$$\forall y \in \mathbb{N}^n, \exists (k, a_1, \ldots, a_n) \in \mathbb{N}_{k_0} \times \mathbb{Z}^n; ky = \sum_{i=1}^{n} a_i x_i$$

2. *Pour tout $y \in \mathbb{N}^n$, on note $k_y = min\{k \in \mathbb{N}; \exists (a_1, \ldots, a_n) \in \mathbb{Z}^n; ky = \sum_{i=1}^{n} a_i x_i\}$ et $(a_1^y, \ldots, a_n^y)$ les entiers vérifiant $k_y y = \sum_{i=1}^{n} a_i^y x_i$. Pour tout entier positif k tel qu'il existe $(a_1, \ldots, a_n) \in \mathbb{Z}^n$ vérifiant $ky = \sum_{i=1}^{n} a_i x_i$, il existe $p \in \mathbb{N}$ tel que $\forall i \in \mathbb{N}^n, a_i = p a_i^y$ et donc $k = p k_y$.*

Preuve :

1. Puisque les $(x_i)_{i \in \mathbb{N}_n}$ sont indépendants, ils forment une base du $\mathbb{Q}$-espace vectoriel $\mathbb{Q}^n$. Par conséquent, pour tout $y \in \mathbb{N}^n$, il existe $(a_1, \ldots, a_n, b_1, \ldots, b_n) \in \mathbb{Z}^n \times (\mathbb{N} \setminus \{0\})^n$ tel que $y = \sum_{i=1}^{n} (a_i / b_i) x_i$. On pose alors $k = \prod_{i=1}^{n} b_i$ et on obtient

$$\exists (k, a_1, \ldots, a_n) \in \mathbb{N} \setminus \{0\} \times \mathbb{Z}^n; ky = \sum_{i=1}^{n} a_i x_i$$

Quitte à factoriser le plus petit commun multiple des entiers $k, a_1, \ldots, a_n$, on peut supposer que $k, a_1, \ldots, a_n$ sont premiers entre eux.
En montrant qu'il existe seulement un nombre fini de tels k, il nous suffira de poser k_0 égal au maximum de ces k.

Supposons avoir des entiers $k, a_1, \ldots, a_n$ premiers entre eux qui vérifient $ky = \sum_{i=1}^{n} a_i x_i$. On peut exprimer cette relation par le système d'equations suivant.

$$\begin{aligned} ky_1 &= \sum_{i=1}^{n} a_i x_{i1} \\ &\cdots \\ ky_n &= \sum_{i=1}^{n} a_i x_{in} \end{aligned}$$

On définit alors les déterminants Δ et Δ_i, pour $i \in \mathbb{N}_n$ de la manière suivante.

$$\Delta = \begin{vmatrix} x_{11} & \ldots & x_{1n} \\ \vdots & & \vdots \\ x_{n1} & \ldots & x_{nn} \end{vmatrix} \text{ et } \Delta_i = \begin{vmatrix} x_{11} & \ldots & x_{1n} \\ \vdots & & \vdots \\ x_{i-11} & \ldots & x_{i-1n} \\ y_1 & \ldots & y_n \\ x_{i+11} & \ldots & x_{i+1n} \\ \vdots & & \vdots \\ x_{n1} & \ldots & x_{nn} \end{vmatrix}$$

Puisqu'on connaît une solution $(a_i)_{i\in\mathbb{N}_n}$ du système étudié, on sait que $a_i = k\Delta_i/\Delta$ (on sait que $\Delta \neq 0$ par indépendance des x_i) ; ainsi k divise chacun des entiers Δa_i. On note k_1 le plus grand commun diviseur de k et Δ, et on définit par ailleurs k_2 et k_3 ainsi : $k_2 = k/k_1$ et $k_3 = \Delta/k_1$.
Puisque pour tout $i \in \mathbb{N}_n$, $k = k_1 k_2$ divise $a_i\Delta = k_1 k_3 a_i$ et que $k_2 \wedge k_3 = 1$, k_2 divise a_i pour tout $i \in \mathbb{N}_n$; par ailleurs k_2 divise k, et puisque $k, a_1, \ldots, a_n$ sont premiers entre eux, $k_2 = 1$. Par conséquent, k est un diviseur de Δ et donc on a bien un nombre fini de k et alors $k_0 = max\{k \in \mathbb{N}; k \mid \Delta\}$ convient.

2. Pour tout $k \geq k_y$, tel qu'il existe $(a_i)_{i\in\mathbb{N}_n} \in \mathbb{N}^n$ vérifiant $ky = \sum_{i=1}^{n} a_i x_i$, il existe $(p, r) \in \mathbb{N}^2$ tel que $p \geq 1$, $r \in [\![0; k_y - 1]\!]$ et $k = k_y p + r$. Par conséquent $ry = ky - k_y p y = \sum_{i=1}^{n} (a_i - p a_i^y) x_i$; on a donc $r = 0$ par minimalité de k_y et puisque les $(x_i)_{i\in\mathbb{N}_n}$ sont indépendants, $\forall i \in \mathbb{N}_n, a_i - p a_i^y = 0$.

□

Complexité : On remarque que pour déterminer k_0, il suffit de savoir calculer Δ, puisqu'on aura plus qu'a poser $k_0 = \Delta$. Or Δ est un déterminant qui peut être calculé en un temps majoré par $O(taille(X)^n.n!)$. Mais n est fixé, et par conséquent, on peut calculer Δ en un temps polynomial en $taille(X)$.

Proposition 4 ([GS64]) *Soit $X = L(\mathbf{0}, P)$ un ensemble linéaire de $\mathbb{N}^n$, tels que les périodes de X soient indépendantes. Alors $\mathbb{N}^n \setminus X$ est un ensemble semi-linéaire.*

Preuve :

Notons j_0 le cardinal de P et $p_1, p_2, \ldots, p_{j_0}$ les éléments de P ; les $(p_i)_{i\in\mathbb{N}_{j_0}}$ sont donc des vecteurs indépendants. On peut donc ajouter $n - j_0$ vecteurs $p_{n-j_0+1}, \ldots, p_n$ choisis parmi la base canonique de $\mathbb{N}^n$ notée $(\epsilon_1^n, \epsilon_2^n, \ldots, \epsilon_n^n)$ de telle sorte que les vecteurs de la famille $(p_i)_{i\in\mathbb{N}_n}$ soient indépendants.
Par ailleurs, d'après le lemme 3, il existe un entier k_0 tel que pour tout $y \in \mathbb{N}^n$, il existe $(k, a_1, \ldots, a_n) \in \mathbb{N}_{k_0} \times \mathbb{Z}^n$ tel que $ky = \sum_{i=1}^{n} a_i p_i$; on note k_y et $(a_1^y, \ldots, a_n^y)$ les entiers vérifiant cette équation avec un k minimum.
On remarque que tous les éléments $x \in X$ ont un $k_x = 1$ et par ailleurs, $\forall i \in \mathbb{N}_{j_0}, a_i^x \geq 0$ et $\forall i \in [\![j_0 + 1; n]\!], a_i^x = 0$.

La démonstration se fait en plusieurs étapes :

- On va d'abord caractériser G_1 l'ensemble des éléments y de $\mathbb{N}^n$ tels qu'au moins un des a_i^y soit négatif : ces y ne peuvent donc pas être dans X. On montrera alors la semi-linéarité de G_1.
- Par la suite, on travaille donc avec un ensemble de y tels que $\forall i \in \mathbb{N}_n, a_i^y \geq 0$, noté H_1. On caractérise alors l'ensemble G_2 des y vérifiant $\exists i \in [\![j_0 + 1; n]\!]; a_i^y > 0$. Ces éléments ne peuvent pas non plus être dans X. On verra alors que G_2 est lui aussi semi-linéaire.

– On se placera alors dans $H_2 = H_1 \setminus G_2$ où tous les éléments y vérifient $\forall i \in \mathbb{N}_{j_0}, a_i^y \geq 0$ et $\forall i \in [\![j_0+1; n]\!], a_i^y = 0$. Et on montrera que le complémetaire de X dans H_2 est semi-linéaire.

Pour tout $k \in \mathbb{N}_{k_0}$ et toute partie I de $\mathbb{N}^n$, soit τ_{kI} la fonction de $\mathbb{N}^n \times \mathbb{N}^n$ dans $\mathbb{N}^n \times \mathbb{N}^n$ telle que

$$\tau_{kI} : y \times (a_1, \ldots, a_n) \mapsto (ky + \sum_{i \in I} a_i p_i) \times (\sum_{i \notin I} a_i p_i)$$

On appelle K l'ensemble $\{y \times y; y \in \mathbb{N}^n\}$ et on s'intéresse alors à $\tau_{kI}^{-1}(K)$ qui est l'ensemble

$$\{y \times (a_1, \ldots, a_n) \in \mathbb{N}^n \times \mathbb{N}^n; ky + \sum_{i \in I} a_i p_i = \sum_{i \notin I} a_i p_i\}$$

Cet ensemble est semi-linéaire puisque K est l'image du semi-linéaire $\mathbb{N}^n$ par le morphisme μ de $\mathbb{N}^n$ dans $\mathbb{N}^n \times \mathbb{N}^n$ défini ainsi $\mu(y) = y \times y$ et puisque les ensemble semi-linéaires sont stables par morphisme inverse.
On pose $A_I = \{y \times (a_1, \ldots, a_n) \in \mathbb{N}^n \times \mathbb{N}^n; \forall i \in I, a_i > 0\}$. On remarque que $A_I = L(c_0^I, B_{2n})$ où B_{2n} est la base canonique de $\mathbb{N}^{2n}$ et où c_0^I est telle que $\forall i \in \mathbb{N}_n, (c_0^I)_i = 0$, $\forall i \in I, (c_0^I)_{n+i} = 1$ et $\forall i \notin I, (c_0^I)_{n+i} = 0$. Donc A_I est un ensemble linéaire.
On note π la projection de $\mathbb{N}^n \times \mathbb{N}^n$ dans $\mathbb{N}^n$ qui à $(y \times x)$ associe y. On s'intéresse alors à $\pi(A_I \cap \tau_{kI}^{-1}(K))$ qui est semi-linéaire puisque $\tau_{kI}^{-1}(K)$ et A_I le sont et que les ensembles semi-linéaires sont stables par intersection et morphisme.
Ainsi $G_1 = \bigcup_{k \in \mathbb{N}_{k_0}, I \neq \emptyset} \pi(A_I \cap \tau_{kI}^{-1}(K))$ est semi-linéaire. Or $y \in G_1$ signifie qu'il existe $k \in \mathbb{N}$, $(a_1, \ldots, a_n) \in \mathbb{N}^n$ et $I \neq \emptyset$ tel que $ky + \sum_{i \in I} a_i p_i = \sum_{i \notin I} a_i p_i$, on a donc $ky = \sum_{i \notin I} a_i p_i - \sum_{i \in I} a_i p_i$ et d'après le lemme 3 au moins un des a_i^y est négatif.
Réciproquement, pour tout $y \in \mathbb{N}^n$ tel qu'au moins un des a_i^y soit négatif, on note $I = \{i \in \mathbb{N}_n; a_i^y < 0\}$ qui est donc non vide, et on remarque que $k_y y + \sum_{i \in I} -a_i^y p_i = \sum_{i \notin I} a_i^y p_i : y \in G_1$.
Par conséquent, $X \subseteq \mathbb{N}^n \setminus G_1$ et donc $G_1 \subseteq \mathbb{N}^n \setminus X$. On note $H_1 = \mathbb{N}^n \setminus G_1$ et alors $\mathbb{N}^n \setminus X = G_1 \cup H_1 \setminus X$.

Pour tout $i \in N_n$, on note $D_i = \{y \times (a_1, \ldots, a_n); a_i > 0\}$; cet ensemble est linéaire puisqu'il est égal à $L(\epsilon_{2n}^{n+i}, \{\epsilon_1^{2n}, \ldots, \epsilon_{2n}^{2n}\})$. Par conséquent, pour toute partie I de $\mathbb{N}_n$ et pour tout $i \in \mathbb{N}^n$, $D_i \cap A_I \cap \tau_{kI}^{-1}(K)$ est semi-linéaire. On définit alors le sous-ensemble G_2 de H_1 de la manière suivante :

$$G_2 = \bigcup_{i \in [\![j_0+1; n]\!]} \bigcup_{I \subseteq \mathbb{N}_n \setminus \{i\}, k \in \mathbb{N}_{k_0}} \pi(D_i \cap A_I \cap \tau_{kI}^{-1}(K))$$

Cet ensemble est semi-linéaire et correspond aux $y \in H_1$ tels qu'il existe $i \in [\![j_0 + 1; n]\!]$ tel que $a_i^y > 0$. Par conséquent, $G_2 \subseteq \mathbb{N}^n \setminus X$ et en notant $H_2 = H_1 \setminus G_2$, on a $\mathbb{N}^n \setminus X = G_1 \cup G_2 \cup (H_2 \setminus X)$. Plus concrètement, l'ensemble H_2 correspond aux $y \in \mathbb{N}^n$ tels que $\forall j \in [\![1; j_0]\!], a_j^y \geq 0$ et $\forall j \in [\![j_0 + 1; n]\!], a_j^y = 0$.

On va désormais montrer que l'ensemble $H_2 \setminus X$ est semi-linéaire. Pour tout $k \in \mathbb{N}_{k_0}$ et tout $j \in \mathbb{N}_{j_0}$, on pose

$$B_{kj} = \{y \times (a_1, \ldots, a_{j_0}) \in \mathbb{N}^n \times \mathbb{N}^{j_0}; (ky = \sum_{i=1}^{j_0} a_i p_i) \ \wedge \ \neg(k \mid a_j)\}$$

Par ailleurs, on pose π' le morphisme de $\mathbb{N}^n \times \mathbb{N}^{j_0}$ défini de la sorte $\pi'(y \times (a_1, \ldots, a_{j_0})) = y$.

On va montrer que chaque B_{kj} est semi-linéaire et que $H_2 \setminus X = \bigcup_{j \in \mathbb{N}_{j_0}, k \in \mathbb{N}_{k_0}} \pi'(B_{kj})$. Ainsi grâce aux propriétes des ensembles semi-linéaires, on aura prouvé que $H_2 \setminus X$ est semi-linéaire.
Soit $E_k = \{y \times (a_1, \ldots, a_{j_0}) \in \mathbb{N}^n \times \mathbb{N}^{j_0}; ky = \sum_{i=1}^{j_0} a_i p_i\}$. D'après la proposition 3, on peut trouver P_k l'ensemble des éléments minimaux de $E_k \setminus \mathbf{0}$, et puisque $E_k = L(\mathbf{0}; P_k)$, E_k est semi-linéaire.
Soit F_{kj} l'ensemble $\{y \times (a_1, \ldots, a_{j_0}) \in \mathbb{N}^n \times \mathbb{N}^{j_0}; \neg(k \mid a_j)\}$. On pose $C_{kj} = \{0^{n+j-1} \times u \times 0^{j_0 - j}; u \in \mathbb{N}_{k-1}\}$ et $P_{kj} = \{\epsilon_1^{n+j_0}, \ldots, \epsilon_{n+j-1}^{n+j_0}, k\epsilon_{n+j}^{n+j_0}, \epsilon_{n+j+1}^{n+j_0}, \ldots, \epsilon_{j_0}^{n+j_0}\}$; on voit que $F_{kj} = L(C_{kj}, P_{kj})$. Par conséquent, F_{kj} est semi-linéaire et donc, puisque $B_{kj} = E_k \cap F_{kj}$, B_{kj} l'est aussi.

On va maintenant montrer que $H_2 \setminus X$ est égal à l'ensemble suivant $J = \bigcup_{j\in\mathbb{N}_{j_0}, k\in\mathbb{N}_{k_0}} \pi'(B_{kj})$ qui est semi-linéaire.

Pour tout $y \in J$, $\exists(a_1,\ldots,a_{j_0}) \in \mathbb{N}^{j_0}; (ky = \sum_{j=1}^{j_0} a_j p_j)$. Donc puisque $\forall j \in \mathbb{N}_{j_0}, a_j = (k/k_y)a_j^y$, on a bien $\forall j \in [\![1; j_0]\!], a_j^y \geq 0$ et $\forall j \in [\![j_0+1; n]\!], a_j^y = 0$; par conséquent $y \in H_2$ et donc $J \subseteq H_2$.

Pour tout $y \in J$, il existe $k \in \mathbb{N}_{k_0}$ et $(a_1,\ldots,a_{j_0}) \in \mathbb{N}^{j_0}$ tels que $\exists j \in \mathbb{N}_{j_0}, \neg(k \mid a_j)$ et $ky = \sum_{j=1}^{j_0} a_j p_j$. Or si $y \in X$, $k_y = 1$ et $\forall j \in \mathbb{N}_{k_0}, a_j = (k/k_y)a_j^y = ka_j^y$ donc k divise chacun des a_j ; ce qui est absurde. On a donc $J \subseteq H_2 \setminus X$.

Pour tout $y \in H_2 \setminus X$, $\exists(a_1,\ldots,a_{j_0}) \in \mathbb{N}^{j_0}; k_y y = \sum_{j=1}^{j_0} a_j^y p_j$. Si $k_y = 1$, $y \in X$; par conséquent, $k_y > 1$. Or k_y est minimal ; donc il existe $i_0 \in \mathbb{N}_{j_0}$ tel que k_y ne divise pas a_i^y. Par conséquent, $y \in \pi'(B_{k_y i_0})$: y appartient à J. Ainsi, on a montré que $H_2 \setminus X \subseteq J$ et donc que $H_2 \setminus X = J$ est un ensemble semi-linéaire.

Par conséquent, puisque $\mathbb{N}^n \setminus X = G_1 \cup G_2 \cup J$ et que G_1, G_2 et Y sont semi-linéaires, $\mathbb{N}^n \setminus X$ est lui aussi semi-linéaire.

□

Complexité :

On va étudier la complexité des différentes étapes de l'algorithme.

- D'abord, il faut ajouter des vecteurs aux $(p_j)_{j\in\mathbb{N}_{j_0}}$ de telle sorte que les $(p_j)_{j\in\mathbb{N}_n}$ soient toujours indépendants et calculer k_0 ; cela peut se faire en temps polynomial.
- Pour calculer G_1, il faut calculer $\pi(A_I \cap \tau_{kI}^{-1}(K))$ pour toute partie I non vide de $\mathbb{N}_n$ et pour tout $k \in \mathbb{N}_{k_0}$. Calculer $A_I \cap \tau_{kI}^{-1}(K)$ a un coût doublement exponentiel, puisque calculer $\tau_{kI}^{-1}(K)$ prend un temps et un espace exponentiel, et que le coût de l'intersection est lui aussi exponentiel. On effectue ce calcul $k_0(2^n - 1)$ fois et par conséquent calculer G_1 se fait en un temps doublement exponentiel.
- Pour calculer G_2, il nous faut calculer des ensembles assez proches de ceux calculés précedement : $\pi(A_I \cap \tau_{kI}^{-1}(K) \cap D_i)$. Si on commence par calculer $A_I \cap D_i$ en un temps et un espace exponentiel et qu'on intersecte ensuite le résultat avec $\tau_{kI}^{-1}(K)$ qui est lui-même exponentiel en espace, on peut calculer $\pi(A_I \cap \tau_{kI}^{-1}(K) \cap D_i)$ en un temps et un espace doublement exponentiel. Par conséquent, on peut calculer G_2 en un temps doublement exponentiel.
- Le dernier ensemble calculé est $H_2 \setminus X$ et il est la réunion de $k_0 j_0$ ensembles qui sont les B_{kj} ; on obtient ceux-ci en faisant l'intersection des ensembles linéaires F_{kj} et E_k. Les E_k sont constructibles en temps exponentiel et les F_{kj} ne dépendent pas de X : on les construit en temps constant. Par conséquent, $H_2 \setminus X$ est calculé en un temps exponentiel.

Construire le complémentaire d'un ensemble linéaire dont les périodes sont indépendantes et dont la constante est le vecteur nul a donc une complexité double-exponentielle.

Proposition 5 ([GS64]) *Si X est un ensemble linéaire de $\mathbb{N}^n$ avec des périodes indépendantes, alors $\mathbb{N}^n \setminus X$ est un ensemble semi-linéaire de $\mathbb{N}^n$.*

Preuve :

On peut écrire X sous la forme $L(c, P)$ avec $P = \{p_1,\ldots,p_j\}$ $(j \in \mathbb{N}_n)$. Pour tout $i \in \mathbb{N}_n$ tel que $c_i > 0$, on pose $C_i = \{(0,\ldots,0,u_i,0,\ldots,0); u_i \in [\![0; c_{i-1}]\!]\}$ et $P_i = \{\epsilon_1^n,\ldots,\epsilon_{n-1}^n,\epsilon_{n+1}^n,\ldots,\epsilon_n^n\}$. Puisque chaque $L(C_i, P_i)$ est semi-linéaire, $G = \bigcup_{i\in\mathbb{N}_n; c_i>0} L(C_i, P_i)$ est aussi semi-linéaire. Pour tout $y \in G$, on a la propriété $\neg(c \leq y)$; par conséquent, G est inclus dans $\mathbb{N}^n \setminus X$.
On pose $Y = \mathbb{N}^n \setminus G = \{y \in \mathbb{N}^n; c \leq y\}$, et on a donc $\mathbb{N}^n \setminus X = G \cup (Y \setminus X)$. On va désormais montrer la semi-linéarité de $Y \setminus X$.
Soit f la bijection de $\mathbb{N}^n$ dans Y telle que $f(y) = y + c$. Pour toutes parties C et Q de $\mathbb{N}^n$, $f(L(C,Q)) = L(f(C), Q)$. Ainsi, une partie Z de $\mathbb{N}^n$ est semi-linéaire si et seulement si $f(Z)$ l'est. Par conséquent $Y \setminus X$ est semi-linéaire si et seulement si $f^{-1}(Y \setminus X)$ l'est. Or $f^{-1}(Y \setminus X) = f^{-1}(Y) \setminus f^{-1}(X) = \mathbb{N}^n \setminus f^{-1}(X)$ et $f^{-1}(X) = L(\mathbf{0}, \{p_1,\ldots,p_j\})$; d'après la proposition 4, $\mathbb{N}^n \setminus f^{-1}(X)$ est semi-linéaire. Par conséquent $f^{-1}(Y \setminus X)$ et donc $Y \setminus X$ est semi-linéaire.
Ainsi $\mathbb{N}^n \setminus X = G \cup (Y \setminus X)$ étant la réunion de deux ensembles semi-linéaires est lui même semi-linéaire.

□

Complexité : Pour calculer l'ensemble G, il nous faut faire calculer au plus n ensembles $L(C_i, P_i)$. Par ailleurs, on peut majorer la taille des C_i à calculer par $taille(X)$, puisque $taille(c) \leq taille(X)$. Ainsi G se calcule en temps linéaire en la taille de X.
On calcule $Y \setminus X$, en calculant $c + \mathbb{N}^n \setminus f^{-1}(X)$; ce qui a la même complexité que le calcul du complémentaire de $f^{-1}(X)$: un coût doublement exponentiel.
Par conséquent, calculer le complémentaire d'un ensemble linéaire avec des périodes indépendantes se fait en un temps doublement exponentiel.

Théorème 4 ([GS64]) *Si X est un ensemble semi-linéaire de $\mathbb{N}^n$, $\mathbb{N}^n \setminus X$ est aussi semi-linéaire.*

Preuve : D'après le lemme 2, il existe une famille $(Z_i)_{i \in \mathbb{N}_m}$ d'ensembles linéaires ayant chacun des périodes indépendantes telle que $X = \bigcup_{i=1}^{m} Z_i$.
Par conséquent, $\mathbb{N}^n \setminus X = \mathbb{N}^n \setminus \bigcup_{i=1}^{m} Z_i = \bigcap_{i=1}^{m} (\mathbb{N}^n \setminus Z_i)$ est semi-linéaire. □

Complexité : Puisqu'obtenir des périodes indépendantes a une complexité non-élémentaire, obtenir le complémentaire d'un ensemble semi-linéaire quelconque a donc aussi une complexité non-élémentaire.

Corollaire 5 *Si X et Y sont des ensembles semi-linéaires de $\mathbb{N}^n$, $X \setminus Y$ est aussi semi-linéaire.*

Preuve : Il suffit de voir que $X \setminus Y = X \cap (\mathbb{N}^n \setminus Y)$ et puisqu'on a la stabilité par intersection et par passage au complémentaire, on a le résultat souhaité. □

2 L'Arithmétique de Presburger

Dans cette partie inspirée de [GS66], on expose une autre facon de représenter les ensembles semi-linéaire : les formules de Presburger. On donne d'abord leurs propriétés de clôture qui sont donc les mêmes que pour les ensembles semi-linéaires mais beaucoup plus faciles à établir. Ensuite, on montre effectivement que les ensembles exprimables par une formule de Presburger sont les ensembles semi-linéaires et on étudie les coûts des passages d'une représentation à l'autre.

2.1 Définitions

Définition 3 *L'ensemble des formules de Presburger, noté $\mathcal{P}$, est le plus petit ensemble de formules vérifiant :*

1. *Pour tout $n \in \mathbb{N}$ et pour tous $(t_i, t'_i)_{i \in [\![0;n]\!]} \in \mathbb{N}^{2^{n+2}}$,*

$$t_0 + \sum_{i=1}^{n} t_i x_i = t'_0 + \sum_{i=1}^{n} t'_i x_i$$

   est une formule de $\mathcal{P}$.
2. *Si P_1 et P_2 sont dans $\mathcal{P}$, alors $P_1 \wedge P_2$ l'est aussi.*
3. *Si P_1 et P_2 sont dans $\mathcal{P}$, alors $P_1 \vee P_2$ l'est aussi.*
4. *Si P est dans $\mathcal{P}$, $\neg P$ l'est aussi.*
5. *Si P est dans $\mathcal{P}$, et si x est une variable libre de P, $\exists x\ P$ est dans $\mathcal{P}$.*

Remarques :

1. La formule

$$t_0 + \sum_{i=1}^{n} t_i x_i \leq t'_0 + \sum_{i=1}^{n} t'_i x_i$$

   est une formule de Presburger ; elle est équivalente à :

$$\exists y\ t_0 + \sum_{i=1}^{n} t_i x_i = t'_0 + \sum_{i=1}^{n} t'_i x_i + y$$

2. De même si P est une formule de Presburger, $\forall x\ P$ est une formule de Presburger. En effet, $\forall x\ P$ est équivalente à $\neg(\exists x\ \neg P)$.
3. On généralise les formules de Presburger aux vecteurs d'entiers : une équation dont les solutions sont des vecteurs dans $\mathbb{N}^m$, correspond à m équations dont les vecteurs sont dans $\mathbb{N}$.

A chaque formule P de l'arithmétique de Presburger à n variables libres, on associe un sous-ensemble de $\mathbb{N}^n$ contenant les n-uplets $(x_1, x_2, \ldots, x_n)$ vérifiant P. Ces ensembles sont appelés ensembles de Presburger.

2.2 Propriétés

Proposition 6 *Les ensembles de Presburger sont stables par intersection, par réunion, par somme, par passage au complémentaire, par morphisme de monoides et par morphisme inverse.*

Preuve : Il est clair que les ensembles de Presburger sont stables par intersection et par union (l'intersection se traduit par un $\wedge$ et l'union par un $\vee$ dans les formules de Presburger). De même, le passage au complémentaire se traduit tout simplement par le connecteur $\neg$.

Si on se donne deux formules P_1 et P_2 dénotant deux ensembles L_1 et L_2, alors l'appartenance à $L_1 + L_2$ se traduit par la formule $\exists \mathbf{y}\ \exists \mathbf{z}\ (\mathbf{x} = \mathbf{y} + \mathbf{z}) \wedge P_1(\mathbf{y}) \wedge P_2(\mathbf{z})$.

Si on se donne une formule P qui dénote un ensemble L et des morphismes de monoïdes ϕ et ψ, les ensembles $\phi(L)$ et $\psi^{-1}(L)$ sont dénotés respectivement par les formules $\exists \mathbf{z}\ (\mathbf{x} = \phi(\mathbf{z})) \wedge P(\mathbf{z})$ et $\exists \mathbf{z}\ (\mathbf{z} = \psi(\mathbf{x})) \wedge P(\mathbf{z})$. □

Proposition 7 *On peut décider si un ensemble de Presburger donné par sa formule est vide en temps triple exponentiel, et on ne peut pas faire mieux dans le pire des cas.*

Preuve : Voir [FR74]. □

2.3 Des ensembles semi-linéaires aux formules de l'arithmétique de Presburger...

Théorème 6 ([GS66]) *La famille des ensembles de Presburger de $\mathbb{N}^n$ est égale à la famille des ensembles semi-linéaires de $\mathbb{N}^n$. Par ailleurs, étant donné un ensemble dans l'une de ces représentations, on peut obtenir de manière effective son expression dans l'autre modèle.*

Preuve :

- Soit L un ensemble semi-linéaire de $\mathbb{N}^n$; alors on peut écrire L sous la forme $\bigcup_{i=1}^{m} L(\mathbf{c_i}, P_i)$, où P_i est de la forme $P_i = \{\mathbf{p_{i1}}, \mathbf{p_{i2}}, \ldots, \mathbf{p_{ik_i}}\}$. Chaque $L(\mathbf{c_i}, P_i)$ est dénoté par la formule

$$Q_i = \exists y_1\ \exists y_2 \ldots \exists y_{k_i}\ \mathbf{x} = \mathbf{c_i} + \sum_{j=1}^{k_i} y_j \mathbf{p_{ij}}$$

  et par conséquent L est l'ensemble des éléments de $\mathbb{N}^n$ vérifiant $\bigvee_{i=1}^{m} Q_i$.

- D'après la proposition 3, l'ensemble des solutions positives d'une équation

$$t_0 + \sum_{1}^{n} t_i x_i = t'_0 + \sum_{1}^{n} t'_i x'_i$$

  est un ensemble semi-linéaire.

  Par ailleurs, puisque les ensembles semi-linéaires sont stables par union, par intersection, par passage au complémentaire et par projection, on voit aisément que les ensembles de Presburger sont des ensembles semi-linéaires. En effet, les connecteurs $\wedge$, $\vee$, $\neg$ et $\exists$ dans les formules de Presburger se traduisent respectivement par des intersections, des unions, des passages au complémentaire et par des projections sur les ensembles de Presburger associés.

□

Complexité :

- Le passage de la représentation d'un ensemble semi-linéaire par des ensembles de bases et de périodes à une formule de Presburger dénotant le même langage est linéaire.
- La complexité du passage d'un ensemble de Presburger à un ensemble semi-linéaire sous forme de bases et de périodes dépend de la complexité des opérations sur les ensembles semi-linéaires ; or la complexité des algorithmes donnés dans la première partie fait intervenir des coûts exponentiels pour une négation par exemple. La complexité de l'algorithme proposé est donc non-élémentaire.
  On peut noter que la complexité de ce passage est au moins triple exponentielle en temps étant donné qu'il permet de décider si un ensemble de Presburger est vide.

3 Expressions rationnelles

Dans cette partie, on s'intéresse aux expressions rationnelles sur le monoïde commutatif $\mathbb{N}^k$, en utilisant les résultats de la partie 1, on montre tout d'abord que les ensembles de vecteurs exprimables à partir d'expressions rationnelles sont les ensembles semi-linéaire. Ensuite, on utilise des résultats de théorie des graphes pour construire directement une formule de Presburger correspondant à une expression rationnelle.

Définition 4 *Soit M un monoïde commutatif et $\cdot$ sa loi, l'ensemble des parties rationnelles de M est le plus petit sous-ensemble de parties de M vérifiant :*

- *Les parties finies sont rationnelles*
- *Si A et B sont des parties rationnelles alors $A \cup B$ et $A \cdot B$ sont des parties rationnelles*
- *Si A est une partie rationnelle alors A^* est une partie rationelle*

Remarque : Si $M = \mathbb{N}^k$, la loi $\cdot$ est l'addition des vecteurs.

3.1 Equivalence entre expressions rationnelles et ensembles semi-linéaires

Proposition 8 *Les parties rationnelles de $\mathbb{N}^k$ sont les parties semi-linéaires.*
On peut passer effectivement de la représentation sous forme d'expression rationnelle à celle sous forme de bases et de périodes et réciproquement.

Preuve : L'ensemble des parties semi-linéaires est inclus de facon évidente dans l'ensemble des parties rationnelles.
D'après la propostion 1 l'ensemble des parties semi-linéaires est stable par union, somme et étoile ; celui-ci contenant également les parties finies, il contient les parties rationnelles. □

Complexité : Le passage d'un ensemble semi-linéaire sous forme de bases et périodes à une expression rationnelle est immédiat.
Par contre l'algorithme proposé de passage d'une expression rationnelle à une forme équivalente en bases et périodes est non-élémentaire car chaque étoile peut nécessiter un coût exponentiel et qu'on ne borne pas la hauteur d'étoile à priori dans les expressions rationnelles.

3.2 Passage d'une expression rationnelle à une formule de Presburger

Théorème 7 ([Reu89]) *On peut passer d'une expression rationnelle sur $\mathbb{N}^k$ à une formule de Presburger dénotant le même ensemble de vecteurs en temps et en espace exponentiels.*

Tout d'abord, notons que l'on peut associer naturellement à une expression rationnelle un automate sur l'alphabet $\mathbb{N}^k$; cette opération a un coût linéaire.

Passer d'une expression rationnelle à une formule de Presburger dénotant le même ensemble de vecteurs, revient donc à passer d'un automate $\mathcal{A}$ sur l'alphabet $\mathbb{N}^k$ à une formule de Presburger.

Pour cela, nous allons associer naturellement à $\mathcal{A}$ un graphe orienté $G_{\mathcal{A}}$. Ensuite nous allons caractériser les vecteurs qui sont l'image commutative de $G_{\mathcal{A}}$. Puis nous traduirons cette caractérisation par des formules de Presburger avant de conclure.

Définition 5 *Un $\mathbb{N}^k$-automate $\mathcal{A}$ est un quintuplet $\mathcal{A} = (Q, k, \Delta, Q_0, F)$ où Q est un ensemble fini d'états, $k \geq 1$ est un entier, Δ est un ensemble fini d'arêtes étiquetées tel que pour tout $e \in \Delta$, les états $s(e), t(e) \in Q$ sont respectivement son origine et son extrémité, et le vecteur $l(e) \in \mathbb{N}^k$ son étiquette, $Q_0 \subseteq Q$ est l'ensemble des états initiaux et $F \subseteq Q$ l'ensemble des états finaux.*

Un chemin P de $\mathcal{A}$ est une suite finie d'arêtes $P = (e_1, \ldots, e_n)$ telle que $s(e_{i+1}) = t(e_i)$ pour tout i, $1 \leq i \leq n-1$. L'origine et l'extrémité du chemin P sont notés $s(P) = s(e_1)$ et $t(P) = t(e_n)$.

Un vecteur v est accepté par $\mathcal{A}$ s'il existe un chemin $P = (e_1, \ldots, e_n)$ tel que $s(P) \in Q_0$, $t(p) \in F$ et $v = l(e_1) + \cdots + l(e_n)$. L'ensemble des vecteurs acceptés par $\mathcal{A}$ est noté $L(\mathcal{A})$.

Image commutative d'un graphe

Nous considérons ici un graphe orienté fini $G = (N, E)$, où N est l'ensemble des sommets et E l'ensemble des arcs. Pour tout arc $e \in E$, on note $s(e)$ et $t(e)$ son origine et son extrémité. Un chemin P dans G est une séquence finie d'arcs $P = (e_1, \ldots, e_n)$ telle que $\forall i \in \mathbb{N}_{n-1}, t(e_i) = s(e_{i+1})$; on note l'origine de P, $s(P) = s(e_1)$ et l'extrémité $t(P) = t(e_n)$. On note aussi le chemin P de la sorte $P : s(P) \to t(P)$ ou $s(P) \xrightarrow{P} t(P)$.

Le graphe G est fortement connexe si pour tout $(q, q') \in N^2$, on peut trouver un chemin P tel que $q \xrightarrow{P} q'$. De même, le graphe est dit faiblement connexe si le graphe non orienté sous jacent à G est connexe.

Nous notons $\mathbb{Z}^E$ (respectivement $\mathbb{N}^E$) l'ensemble des fonctions de E dans $\mathbb{Z}$ (respectivement $\mathbb{N}$). La fonction qui prend la valeur 1 en e_0 et 0 en tout autre élément de E est simplement notée e_0. Par conséquent toute fonction $f \in \mathbb{Z}^E$ (respectivement $\mathbb{N}^E$) s'écrit de manière unique comme une combinaison linéaire des e avec des coefficients f_e dans $\mathbb{Z}$ (dans $\mathbb{N}$).

$$f = \sum_{e \in E} f_e e$$

Le bilan d'un chemin P dans le graphe G est défini ainsi : $b(P) = t(P) - s(P)$. On remarque que $b(P)$ est nul si et seulement si P est un chemin fermé.

L'image commutative du chemin sert à déterminer le nombre de fois que chaque arc e du graphe apparaît dans P. On pose donc $|P|_e$ le nombre d'occurences de l'arc e dans le chemin P et l'image commutative de P est $\overline{P} \in \mathbb{N}^E$,

$$\overline{P} = \sum_{e \in E} |P|_e e \in \mathbb{N}^E$$

On définit le bilan d'une fonction f de $\mathbb{N}^E$, $f = \sum_{e \in E} f_e e$ de la manière suivante :

$$\begin{aligned} b(f) &= \sum_{e \in E} f_e (t(e) - s(e)) \\ &= \sum_{e \in E} f_e b(e) \end{aligned}$$

On peut alors prouver que $b(P) = b(\overline{P})$ pour tout chemin P dans G. Par ailleurs, pour toute fonction $f \in \mathbb{N}^E$, on définit la restriction de G à f comme le graphe

$$G|f = (N', E')$$

où $E' = \{e \in E \mid f_e > 0\}$ et $N' = \{q \in N \mid \exists e \in E', s(e) = q \text{ ou } t(e) = q\}$. Autrement dit, $G|f$ est le graphe obtenu en retirant à G tous les arcs n'intervenant pas dans f.

Proposition 9 ([Reu89]) *Soit $G = (S, A)$ un graphe orienté et v dans $\mathbb{N}^A$, alors v est l'image commutative d'un chemin si et seulement si les deux conditions suivantes sont réalisées :*

1. *Le graphe $G|f$ est connexe*
2. *Le bilan de f est la différence de deux sommets du graphe*

Preuve : Ces deux conditions sont évidemment nécessaires, montrons qu'elles sont suffisantes. L'idée de la preuve est de trouver un ensemble de chemins dont la somme des images commutatives est f, puis de les relier tour à tour jusqu'à ce qu'il ne reste plus qu'un seul chemin.

Soit $f \neq 0$, Il existe un ensemble de chemins non vides $\{c_1, \cdots, c_n\}$ tels que $\sum_{i=1}^{n} \overline{c_i} = f$ (il suffit de prendre la suite formée de v_a fois l'arc a, v_b fois l'arc b , ...). Choisissons un tel ensemble minimal et montrons que $n = 1$.

On peut supposer que $c_1, \cdots, c_p$ ne sont pas fermés et que $c_{p+1}, \cdots, c_n$ sont fermés. Si $p \geq 2$, comme $b(c_{p+1}) = \cdots = b(c_n) = 0$, on a

$$\begin{aligned} t - s &= b(f) = \sum_{i=1}^{p} b(c_i) \\ &= \sum_{i=1}^{p} (t(c_i) - s(c_i)) \end{aligned}$$

Comme la dernière somme a $p \geq 2$ termes et qu'il s'agit de fonctions caratéristiques, il existe $j \neq k$, $1 \leq j, k \leq p$ tels que $t(c_j) = s(c_k)$. On peut alors relier les chemins j et k, ce qui contredit la minimalité de n ; on a donc $p = 0$ ou $p = 1$.

Supposons maintenant $n \geq 2$, comme $G|f$ est connexe, deux des chemins de $\{c_1, \cdots, c_n\}$ possèdent un sommet commun. Mais puisque $p \leq 1$, au moins un de ces deux chemins est fermé, donc on peut relier ces deux chemins, ce qui contredit la minimalité de n. On a donc montré que nécessairement $n = 1$ □

On note $\mathbf{G}_{q,q'}$ l'ensemble des sous-graphes G' de G tels qu'il existe un chemin P de q à q' dans G et que $G' = G|\overline{P}$. On remarque que $G' \in \mathbf{G}_{q,q'}$ si et seulement si G' est connexe et que $(q, q') \in G'^2$. D'après la proposition 9, $f \in \mathbb{N}^E$ est l'image commutative d'un chemin $P : q \to q'$ si et seulement si il existe un graphe $G' \in \mathbf{G}_{q,q'}$ tel que $G' = G|f$ et $b(f) = q' - q$.

Formule «$G' = G|f$» : considérons un sous-graphe $G' = (N', A')$ de G. Pour toute arête $e \in E$, on définit la formule de Presburger $\Psi^e_{G'}(f)$ par

$$\Psi^e_{G'}(f) := \begin{cases} f_e \geq 1 & \text{si } e \in E' \\ f_e = 0 & \text{sinon} \end{cases}$$

Cette formule correspond à la propriété «Si l'arête e est utilisée dans G' alors elle est utilisée par f». On peut dès lors définir la formule $\Psi_{G'}(f)$:

$$\Psi_{G'}(f) := \bigwedge_{e \in E} \Psi^e_{G'}(f)$$

On a donc $G' = G|f$ si et seulement si $\Psi_{G'}(f)$ est vrai.

Formule «$b(f) = q' - q$» : soient deux sommets $q, q' \in N$. Pour tout sommet $q'' \in N$, on définit la formule de Presburger $\Psi^{q''}_{q,q'}(f)$ par

$$\Psi^{q''}_{q,q'}(f) := \sum_{\substack{e \in E \\ t(e) = q''}} f_e - \sum_{\substack{e \in E \\ s(e) = q''}} f_e = \delta_{q'q''} - \delta_{qq''}$$

où δ est le symbole de Kronecker. Cette formule correspond à la propriété "Le bilan de q'' a bien la bonne valeur". La bonne valeur signifie 0 si $q'' \neq q, q'$; 1 si $q'' = q'$ et -1 si $q'' = q$. On peut alors définir la formule $\Psi_{q,q'}(f)$:

$$\Psi_{q,q'}(f) := \bigwedge_{q'' \in N} \Psi^{q''}_{q,q'}(f)$$

On a donc $b(f) = q' - q$ si et seulement si $\Psi_{q,q'}(f)$ est vrai.

Formule «f est l'image commutative d'un chemin de q à q'» : soit deux sommets q, q'. On définit la formule de Presburger $\Phi^G_{q,q'}(f)$ par

$$\Phi^G_{q,q'}(f) := \bigvee_{G' \in \mathbf{G}_{q,q'}} (\Psi_{q,q'}(f) \bigwedge \Psi_{G'}(f))$$

On a f est l'image commutative d'un chemin de q à q' si et seulement si $\Phi^G_{q,q'}(f)$ est vrai ; c'est en effet la caractérisation de la proposition 9.

Soit un automate $\mathcal{A} = (Q, k, \Delta, Q_0, F)$ sur $\mathbb{N}^k$. On associe naturellement à $\mathcal{A}$ un graphe $G_\mathcal{A} = (N_\mathcal{A}, E_\mathcal{A})$ par $N_\mathcal{A} = Q$ and $E_\mathcal{A} = \Delta$. On considère la formule de Presburger $\Phi_\mathcal{A}(v)$ pour $v \in \mathbb{N}^k$ définie par

$$\Phi_\mathcal{A}(v) = \exists f \in \mathbb{N}^E \bigvee_{(q,q') \in Q_0 \times F} (\Phi^{G_\mathcal{A}}_{q,q'}(f) \bigwedge v = \sum_{e \in E_\mathcal{A}} f_e.l(e))$$

$\Phi^{G_\mathcal{A}}_{q,q'}(f)$ nous assure que f est l'image commutative d'un chemin d'un état initial à un état final et $v = \sum_{e \in E_\mathcal{A}} f_e.l(e)$ nous assure que v est le vecteur correspondant au chemin f. On a donc $v \in L(\mathcal{A})$ si et seulement si $\Phi_\mathcal{A}(v)$ est vrai. On a construit la formule de Presburger correspondant à l'automate $\mathcal{A}$.

Complexité : Intéressons-nous au coût de la construction de cette formule de Presburger.

- Construire la formule $\Psi_{G'}(f)$ se fait en un temps et un espace linéaire en fonction du nombre d'arêtes du graphe G ; et donc dans le cas qui nous intéresse, on le fait en $O(|E|)$.
- De même, construire la formule $\Psi^{q''}_{q,q'}(f)$ se fait aussi en un temps et un espace linéaires en cardinal de E. Ainsi, pour construire $\Psi q, q'(f)$, cela nous coûte un temps et un espace proportionnel au nombre de sommets du graphe multiplié par le nombre d'arcs : on le fait en $O(|N||E|)$.
- Pour construire la formule $\Psi_{q,q'}(f) \wedge \Psi_{G'}(f)$, il nous faut un temps et un espace proportionnel à $(|N||E|)$. Et donc pour construire $\Phi^G_{q,q'}(f)$, il nous faut un temps et un espace qui est en $O(|N||E||\mathbf{G}_{q,q'}|)$, mais on peut majorer $|\mathbf{G}_{q,q'}|$ par le nombre de sous graphes de G, c'est à dire par le nombre de partie de E : cette formule se calcule donc en un espace qui est un $O(2^{|E|}|N||E|)$. Pour déterminer $\mathbf{G}_{q,q'}$, il suffit à partir de tout graphe partiel H de vérifier sa connexité et l'appartenance de q et q' à H ; on peut donc calculer $\mathbf{G}_{q,q'}$ en $O(|E|2^{|E|})$ et ainsi, la formule $\Phi^G_{q,q'}(f)$ en $O(2^{|E|}|N||E|^2)$.
- Lorsqu'on travaille sur les automates, N devient Q et E devient Δ. Calculer la formule $v = \sum_{e \in E_\mathcal{A}} f_e.l(e)$ nécessite un temps et un espace en $O(|\Delta| max\{l(e); e \in \Delta\})$; et par conséquent le calcul de chaque $\Phi^{G_\mathcal{A}}_{q,q'}(f) \wedge v = \sum_{e \in E_\mathcal{A}} f_e.l(e)$ se fait en espace en $O(|\Delta| max\{l(e)\} + 2^{|\Delta|}|Q||\Delta|)$ et en temps en $O(|\Delta| max\{l(e)\} + 2^{|\Delta|}|Q||\Delta|^2)$. Ainsi, le calcul d'une formule de Presburger à partir d'une expression rationnelle nécessite un espace en $O(|\Delta| max\{l(e)\} + 2^{|\Delta|}|Q||\Delta|^2|Q_0||F|)$ et un temps en $O(|\Delta| max\{l(e)\} + 2^{|\Delta|}|Q||\Delta||Q_0||F|)$.

On peut donc calculer la formule de Presburger en un espace qui est un $O(|\Delta| max\{l(e)\} + 2^{|\Delta|}|Q|^3|\Delta|)$ et en un temps majoré par $O(|\Delta| max\{l(e)\} + 2^{|\Delta|}|Q|^3|\Delta|^2)$.

4 Ensembles semi-linéaires et automates

Mise en garde : dans cette partie, nous considérons des automates sur l'alphabet $\{0,1\}^n$, il ne faut surtout pas les confondre avec les automates associés à une expression rationnelle de la partie précédente. En effet, comme nous le verrons, ces deux types d'automates ne représentent pas les mêmes ensembles.
Les parties 4.1 et le début de la partie 4.2 sont inspirées de [WB00]. La fin de la partie 4.2 et les parties 4.3 à 4.7 sont personnelles.

Idée de l'utilisation des automates :

Un ensemble semi-linéaire S est un ensemble de vecteurs. On cherche à associer à S un automate ; il est alors assez naturel de coder les vecteurs sur un alphabet et d'essayer de trouver un automate qui reconnaît exactement les codages des vecteurs de S.
On cherche à coder un vecteur $\mathbf{x} = (x_1, ..., x_k)$ de $\mathbb{N}^k$, pour cela on convertit chaque x_i en base 2 et on rajoute des 0 au début des composantes les plus courtes pour que toutes les composantes aient la même longueur. On obtient alors un mot sur l'alphabet $\{0,1\}^k$ en considérant les premiers chiffres de chaque composante, puis les seconds ...
Exemple : on code le vecteur $(4,3)$ par $(100,11)$ puis en rajoutant des 0 on obtient le vecteur $(100,011)$ qui correspond au mot $(1,0)(0,1)(0,1)$ sur l'alphabet $\{0,1\}^2$.

Remarque : Un même vecteur peut donc avoir différentes représentations (différant par des 0 au début) ; on essaiera par la suite de construire des automates qui acceptent les vecteurs indépendamment de leurs représentations.

4.1 Construction d'un automate pour une équation linéaire

On considère une équation linéaire à coefficients dans $\mathbb{Z}$ et on cherche à construire un automate qui reconnaît les codages des solutions de cette équation (par la suite s'il n'y a pas d'ambiguïté on confondra les vecteurs et leur codage).

Considérons un vecteur $\mathbf{x}$ de longueur n, $\mathbf{x} = (x_1, ..., x_k)$, chaque x_i étant le codage en base 2 de la $i$ème composante de $\mathbf{x}$. Si on concatène les bits de $\mathbf{b} = (b_1, ..., b_k)$ à ceux de $\mathbf{x}$ on obtient le vecteur $\mathbf{x}'$ de longueur $n+1$ dont la valeur vérifie $\mathbf{x}' = 2\mathbf{x} + \mathbf{b}$: le facteur 2 apparaît du fait qu'on a décalé d'une unité vers la droite un nombre écrit en base 2, et on remplace les bits les plus à droite par ceux de $\mathbf{b}$.

Construisons maintenant l'automate reconnaissant les vecteurs solutions de l'équation $a_1x_1+...+a_nx_n = c$ noté $a.\mathbf{x} = c$. L'état où on se trouve va correspondre à la valeur de $a_1x_1 + ... + a_nx_n$ si on a lu le vecteur $\mathbf{x}$ jusque ici. Par exemple, l'état initial est étiqueté 0.

On a vu plus haut que si on rajoutait le caractère $\mathbf{b}$ à $\mathbf{x}$ on obtenait le vecteur $\mathbf{x}'$ dont la valeur vérifie $\mathbf{x}' = 2\mathbf{x}+\mathbf{b}$, on a donc $a.\mathbf{x}' = 2a.\mathbf{x}+a.\mathbf{b}$. Si on était dans l'état α et qu'on lit la lettre $\mathbf{b}$ on se retrouve dans l'état $2\alpha + a.\mathbf{b}$

Prenons un exemple : considérons l'équation $x - y = 2$, supposons qu'on ait lu jusque là le mot $(10, 01)$ qui correspond au couple $(2, 1)$, on se trouve donc normalement dans l'état 1 , supposons que l'on lise le mot $(1, 1)$, les nouvelles valeurs de x et de y sont donc $x' = 101 = 5, y' = 011 = 3$, on doit donc se rendre dans l'état $2(x - y) + a.\mathbf{b} = 2 \cdot 1 + (1 - 1) = 2$, et on peut vérifier que 2 est bien la valeur de $x' - y'$.

Construisons formellement l'automate correspondant à une équation à n variables $a.\mathbf{x} = c$: il s'agit de l'automate sur l'alphabet $\{0, 1\}^n$ $\mathcal{A} = (\mathbb{Z}, \delta, 0, c)$

La fonction de transition δ est définie comme indiqué ci-dessus : $\forall\alpha \in \mathbb{Z}\ \forall\mathbf{b} \in \{0, 1\}^n\ \delta(\alpha, \mathbf{b}) = 2\alpha + a.\mathbf{b}$

Cet automate a un nombre infini d'états mais il n'a qu'un nombre fini d'états co-accessibles, c'est à dire d'états à partir desquels on peut atteindre l'état final. En effet soit $||\mathbf{a}||_1 = \Sigma_{i=1}^n |a_i|$ la norme du vecteur $\mathbf{a}$. Si un état α vérifie $|\alpha| > ||\mathbf{a}||_1$ alors toute transition partant de α mènera à des états α' tels que $|\alpha'| > |\alpha|$ donc en particulier tout état α vérifiant $|\alpha| > ||\mathbf{a}||_1$ et $|\alpha| > |c|$ n'est pas co-accessible.

Remarque : Les états vérifiant $|\alpha| > ||\mathbf{a}||_1$ et $|\alpha| > |c|$ ne sont pas co-accessibles, mais ce ne sont pas les seuls ; néanmoins l'algorithme suivant permet de ne construire que les états à la fois accessibles et co-accessibles.

S est l'ensemble des états de l'automate et A l'ensemble des états à traiter :

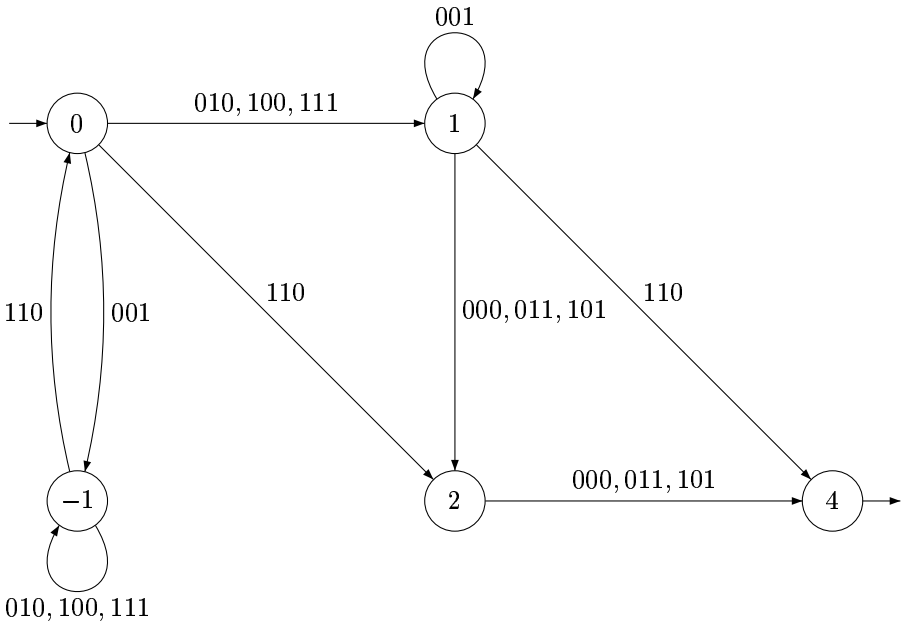
1. $S := c,\ A := c$
2. Tant que $A \neq \emptyset$ faire
3. Enlever un état α de A et pour tout $\mathbf{b}$ de $\{0, 1\}^n$:
   Si $\alpha_0 = (\alpha - a.\mathbf{b})/2$ est entier alors
   - Si α_0 n'est pas dans S, l'ajouter à S et A
   - Ajouter une transition étiquetée $\mathbf{b}$ de α_0 à α
4. Supprimer les états non-accessibles
5. Rajouter un état puits pour compléter l'automate

FIG. 1 - Algorithme de construction de l'automate correspondant à l'équation $a.\mathbf{x} = c$

Théorème 8 ([WB00]) *L'automate construit par l'algorithme précédent est l'automate minimal reconnaissant les solutions de l'équation $a.\mathbf{x} = c$ et peut être construit en temps et en espace exponentiel.*

Preuve : L'automate ainsi construit est déterministe et complet. Il est minimal car les ensembles de mots acceptés à partir de 2 états différents ne peuvent être les mêmes. En effet, le langage accepté par l'état puit est le langage vide ; les langages acceptés par les autres états sont non vides donc en particulier contiennent un mot ; et un mot accepté par un état ne peut l'être par un autre ; on pourrait le démontrer formellement mais cela se voit mieux si on prend un exemple : dans le cas de l'équation $x + y - z = 4$, un mot accepté

en partant de l'état -1 vérifie $x+y-z=5$, s'il était accepté également à partir de l'état 1, il vérifierait $x+y-z=3$; ce qui est absurde. □



Pour raison de lisibilité, on n'a pas représenté l'état puits.

FIG. 2 - Automate pour l'équation $x+y-z=4$

Complexité : Pour évaluer la complexité de l'algorithme, il faut déterminer le nombre d'états de l'automate construit : tout état de l'automate est obtenu en appliquant la transformation $T(\alpha)=(\alpha-a.\mathbf{b})/2$. On commence à appliquer cette transformation à c donc après la première étape de l'algorithme, tous les états trouvés sont étiquetés dans l'intervalle $[\frac{c}{2}-\frac{||\mathbf{a}||_1}{2},\frac{c}{2}+\frac{||\mathbf{a}||_1}{2}]$. On montre par récurrence qu'à la $i$ème étape de l'algorithme, les états trouvés sont dans l'intervalle $[\frac{c}{2^i}-\sum_{j=1}^{i}\frac{||\mathbf{a}||_1}{2^j},\frac{c}{2^i}+\sum_{j=1}^{i}\frac{||\mathbf{a}||_1}{2^j}]$ qui est inclus dans $[\frac{c}{2^i}-||\mathbf{a}||_1,\frac{c}{2^i}+||\mathbf{a}||_1]$. Pour $i>log_2\ c$, cet intervalle est inclus dans $[-||\mathbf{a}||_1,||\mathbf{a}||_1]$. Les états de l'automate sont donc localisés dans au plus $log_2 c+1$ intervalles de taille au plus $2||\mathbf{a}||_1+1$. Le nombre total d'états de l'automate est en $O(log_2\ c\times||\mathbf{a}||_1)$. Le coût de l'algorithme est donc linéaire en la taille du codage de la constante et exponentiel en la taille du codage des coefficients de l'équation.

4.2 Construction d'un automate pour une formule de Presburger

Dans le paragraphe précédent on a montré comment construire un automate reconnaissant les solutions d'une équation linéaire. On construit par induction un automate reconnaissant toute formule de Presburger :

- $\mathcal{A}_{\phi_1\wedge\phi_2}=\mathcal{A}_{\phi_1}\cap\mathcal{A}_{\phi_2}$, c'est à dire l'automate reconnaissant le language intersection construit en temps quadratique.
- $\mathcal{A}_{\phi_1\vee\phi_2}=\mathcal{A}_{\phi_1}\cup\mathcal{A}_{\phi_2}$, c'est à dire l'automate reconnaissant le language union construit en temps linéaire. Cette opération fait perdre son caratère déterministe à l'automate.
- $\mathcal{A}_{\neg\phi}$ est l'automate reconnaissant le language complémentaire de celui accepté par $\mathcal{A}_\phi$. Il peut être nécessaire de déterminiser l'automate $\mathcal{A}_\phi$ pour cette opération ce qui peut engendrer un coût exponentiel.
- $\mathcal{A}_{\exists x\ \phi}$ est l'automate correspondant à la projection de la coordonnée correspondant à x, opération faite en temps linéaire, mais qui peut faire perdre son caractère déterministe à l'automate.

Remarque : Lorsqu'on fait une projection, il est indiqué dans [WB00] qu'on reconnaîtra des mots qui seront solution, mais pas forcément les plus courts. Par exemple, si on prend l'automate construit de la manière

décrite précédemment reconnaissant l'ensemble des x tels que $\exists z\ z = 2x$, alors on ne reconnaîtra pas le mot 1, alors que le mot 01 sera accepté.
Pour remédier à ce problème, il suffit d'augmenter l'ensemble des états initiaux en y ajoutant tous les états atteignables depuis un état initial par une suite finie de 0.

Complexité : On a vu plus haut qu'une négation pouvait engendrer un coût exponentiel ; comme on ne peut pas borner la hauteur des négations à priori, l'algorithme proposé est non-élémentaire.
On peut noter que la complexité de ce passage est au moins triple exponentielle en temps étant donné qu'il permet de décider en temps linéaire si un ensemble de Presburger est vide.

Le coût de l'algorithme est certes non-élémentaire, mais il est bien moindre dans certains cas particuliers, notamment le suivant qui nous servira pour passer d'une représentation sous forme de bases-périodes ou sous forme d'expression rationnelle à l'automate associé (paragraphes 4.5 et 4.6).

Proposition 10 *Soit P une formule de Presburger sans négation et dont on peut borner la hauteur de $\wedge$; alors on peut construire l'automate associé à P en temps et en espace exponentiel.*

Preuve : Tout d'abord, remarquons que pour une telle formule nous n'aurons jamais à déterminiser les automates construits puisque les opérations d'union, intersection et projection ne requièrent pas un automate déterministe.
D'après le théorème 8, on peut construire les automates $(\mathcal{A}_i)_{i\in I}$ associés aux équations de base avec un coût exponentiel. Remarquons alors que les opérations d'union et de projection ont un coût linéaire. Seule l'intersection a un coût quadratique. Mais on peut borner la hauteur d'utilisation de ces intersections, l'algorithme proposé est donc polynomial en la taille des automates $(\mathcal{A}_i)_{i\in I}$, donc exponentiel en la taille de P. □

4.3 Autres opérations sur les automates

Les opérations d'union, intersection et complémentaire se font en utilisant les méthodes classiques sur les automates. Mais comment réaliser les opérations de somme, morphisme, morphisme inverse et étoile ?

Proposition 11 *Soient $\mathcal{A}_1$ et $\mathcal{A}_2$ deux automates dénotant les ensembles de vecteurs L_1 et L_2, ϕ et ψ des morphismes de monoïde. Alors on sait construire des automates reconnaissant $L_1 + L_2$, $\phi(L_1)$, $\psi^{-1}(L_1)$.*

Preuve : Dans la partie 2, on a vu que si deux formules de Presburger P_1 et P_2 dénotaient deux ensembles L_1 et L_2, alors l'appartenance à $L_1 + L_2$ se traduisait par la formule $\exists \mathbf{y}\ \exists \mathbf{z}\ (\mathbf{x} = \mathbf{y} + \mathbf{z}) \wedge P_1(\mathbf{y}) \wedge P_2(\mathbf{z})$.
On construit donc en appliquant la méthode du paragraphe 4.2 l'automate $\mathcal{A}'_0$ reconnaissant $\{\mathbf{x}\times\mathbf{y}\times\mathbf{z}\ ;\ \mathbf{x} = \mathbf{y} + \mathbf{z}\}$. En modifiant un peu les automates $\mathcal{A}_1$ et $\mathcal{A}_2$ on obtient les automates $\mathcal{A}'_1$ et $\mathcal{A}'_2$ reconnaissant respectivement $\{\mathbf{x}\times\mathbf{y}\times\mathbf{z}\ ;\ \mathbf{y}\in L_1\}$ et $\{\mathbf{x}\times\mathbf{y}\times\mathbf{z}\ ;\ \mathbf{z}\in L_2\}$. On construit alors naturellement l'automate $\mathcal{A} = \exists\mathbf{y}\ \exists\mathbf{z}\ \mathcal{A}'_0 \cap \mathcal{A}'_1 \cap \mathcal{A}'_2$ qui reconnaît $\{\mathbf{x}\ ;\ \exists\mathbf{y}\ \exists\mathbf{z}\ (\mathbf{x} = \mathbf{y}+\mathbf{z}) \wedge (\mathbf{y}\in L_1) \wedge (\mathbf{z}\in L_2)\}$. $\mathcal{A}$ est donc l'automate reconnaissant $L_1 + L_2$.
On utilise la même méthode pour construire les automates reconnaissant $\phi(L_1)$ et $\psi^{-1}(L_1)$ en se souvenant des formules de Presburger : $\exists\mathbf{z}\ (\mathbf{x} = \phi(\mathbf{z})) \wedge P(\mathbf{z})$ et $\exists\mathbf{z}\ (\mathbf{z} = \psi(\mathbf{x})) \wedge P(\mathbf{z})$. □

Complexité : Le coût de ces opérations est linéaire en la taille des automates.

Remarque : On ne peut pas construire l'automate correspondant à L^* par la méthode précédente car on ne connaît pas de formule explicite donnant l'étoile d'une formule de Presburger.

Automates pour des systèmes d'équations linéaires
Pour construire un automate reconnaissant les solutions d'un système d'équations linéaires, on peut donc faire un automate pour chaque équation linéaire puis calculer l'intersection de tous ces automates. Néanmoins, on peut modifier l'algorithme utilisé pour une équation pour pouvoir en traiter plusieurs d'un coup. Le principe est le même, mais chaque état contient une valeur pour chaque équation. Ainsi, pour les mêmes raisons qu'auparavant, on est sur d'obtenir l'automate minimal reconnaissant les solutions du système.

4.4 Utilité de la déterminisation

Il peut parfois être intéressant de déterminiser puis de minimiser l'automate suite à une projection (qui correspond à un connecteur $\exists$ dans la formule de Presburger). En effet, intuitivement, cela correspond au fait

que la formule $P(x_1, \ldots, x_n)$ semble plus difficile à satisfaire que $\exists x_n \ P(x_1, \ldots, x_n)$ et que par conséquent l'automate reconnaissant la première formule serait plus gros que le second. Et si par la suite, on travaille de nouveau sur cet automate, en réduisant sa taille, on augmente l'efficacité de l'algorithme.

Prenons par exemple la formule $P(x) : \exists y \ \exists z \ (x - y = 2) \wedge (x + y - z = 4)$; l'automate non déterministe obtenu par l'algorithme décrit précedemment a 7 états et une fois déterminisé et minimisé, il n'en a plus que 4.

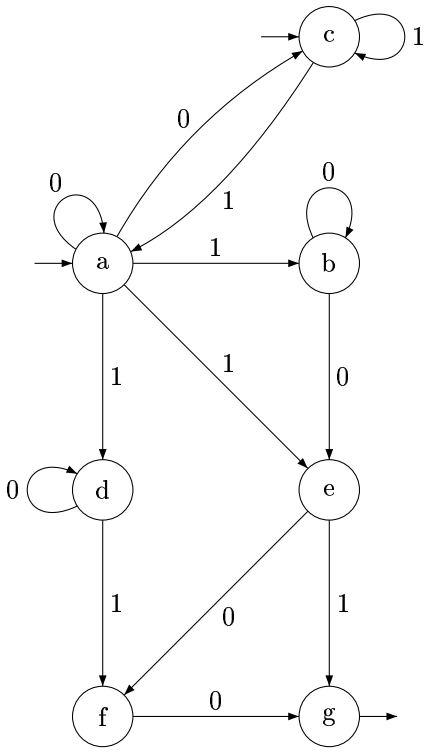


FIG. 3 - Automate non déterministe reconnaissant les solutions de $P(x) : \exists y \ \exists z \ (x - y = 2) \wedge (x + y - z = 4)$

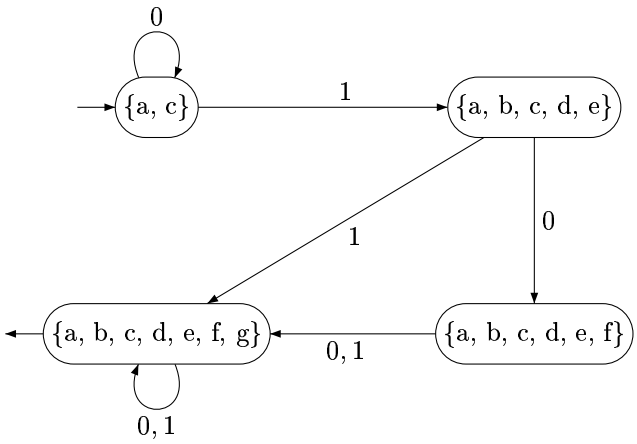


FIG. 4 - Automate déterministe reconnaissant le même langage

Complexité : La complexité de la déterminisation est linéaire en le nombre d'états de l'automate obtenu ; et par conséquent, elle peut être exponentielle en la taille de l'automate de départ. Mais en fait, on essaie d'obtenir des automates plus simples, par conséquent si l'automate déterministe obtenu est plus petit que l'automate de départ, on peut avoir une complexité linéaire.
Mais on ne sait pas décider quels vont être les cas où la déterminisation va être intéressante.

4.5 Passage d'un ensemble semi-linéaire aux automates

Théorème 9 *On peut construire l'automate associé à un semi-linéaire donné sous forme de bases et périodes en temps et en espace exponentiels.*

Preuve : A partir d'un ensemble semi-linéaire donné sous forme de bases et de périodes, comment passer à un automate reconnaissant cet ensemble ? On peut calculer la formule de Presburger correspondante puis utiliser le paragraphe 4.1 pour obtenir l'automate.
Le coût d'une telle opération semble à priori être non-élémentaire étant donné que le passage de l'arithmétique de Presburger aux automates l'est. Mais les formules de Presburger obtenues à partir d'une expression sous forme de base-périodes sont de la forme $\bigvee_{i=1}^{m} Q_i$ où

$$Q_i = \exists y_1\ \exists y_2 \ldots \exists y_{k_i}\ \mathbf{x} = \mathbf{c_i} + \sum_{j=1}^{k_i} y_j \mathbf{p_{ij}}$$

La formule correspondante ne fait donc pas intervenir de négation et la hauteur de $\wedge$ est de 1, donc d'après la proposition 10 le coût de création de l'automate est exponentiel. □

4.6 Passage d'une expression rationnelle aux automates

Théorème 10 *On peut construire l'automate associé à un semi-linéaire donné sous forme d'expression rationnelle en temps et en espace exponentiels.*

Preuve : Le principe est le même qu'au paragraphe précédent : on construit l'automate correspondant à une expression rationnelle en passant par la formule de Presburger correspondante, à partir de laquelle on construit l'automate, mais en se souvenant de la forme d'une telle formule : il s'agit de

$$\exists f \in \mathbb{N}^E \bigvee_{(q,q') \in Q_0 \times F} (\Phi_{q,q'}^{G_{\mathcal{A}}}(f) \bigwedge v = \sum_{e \in E_{\mathcal{A}}} f_e.l(e))$$

$$= \exists f \in \mathbb{N}^E \bigvee_{(q,q') \in Q_0 \times F} \bigvee_{G' \in \mathbf{G}_{q,q'}} (\Psi_{q,q'}(f) \bigwedge \Psi_{G'}(f)) \bigwedge v = \sum_{e \in E_{\mathcal{A}}} f_e.l(e))$$

Chaque expression $\Psi_{q,q'}(f) \bigwedge \Psi_{G'}(f)) \bigwedge v = \sum_{e \in E_{\mathcal{A}}} f_e.l(e)$ a une taille polynomiale en la taille de l'expression rationnelle, ne contient pas de négation, et a une hauteur de $\wedge$ bornée par 1 donc d'après la proposition 10, on peut construire l'automate associé à une telle formule en temps et en espace exponentiels.
Il ne reste plus qu'à faire l'union pour $(q, q') \in Q_0 \times F$, $G' \in \mathbf{G}_{q,q'}$ de tous ces automates, il y en a un nombre exponentiel ; le coût total reste donc exponentiel. □

4.7 Remarque importante sur les automates

On pourrait être tenté de vouloir convertir un automate en une formule de Presburger, une expression rationnelle ou un ensemble de bases et de périodes. Cela est impossible car certains automates ne correspondent pas à une ensemble semi-linéaire :

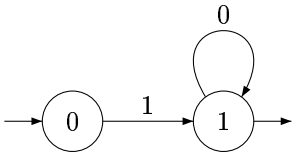


FIG. 5 - automate reconnaissant un langage non semi-linéaire

Cet automate reconnaît le langage $\{2^n; n \in \mathbb{N}\}$ qui n'est pas un ensemble semi-linéaire.

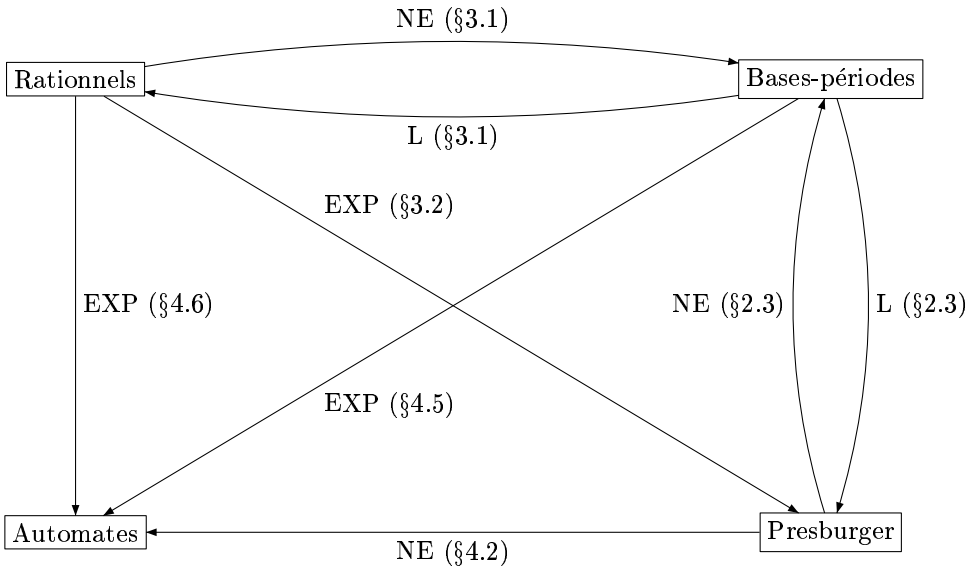
5 Récaptitulatif de la complexité

Légende :	L	Linéaire
	P	Polynomial
	EXP	Exponentiel
	3-EXP	Triple Exponentiel
	NE	Non-élémentaire

Représentation	$\bigcup$	$\bigcap$	Compl.	$+$	$*$	φ	φ^{-1}	Test $\emptyset$	$\subseteq$
Rationnels	L	NE	NE	L	L	L	NE	L	2-EXP
Bases-périodes	L	EXP	NE	P	EXP	L	EXP	L	2-EXP
Presburger	L	L	L	L	NE	L	L	3-EXP	3-EXP
Automates	L	P	EXP/L	L	?	L	L	L	EXP/P

Remarques :

- Lorsqu'il y a deux complexités pour les automates, celle de gauche concerne les automates non-déterministes et celle de droite les automates déterministes.
- Le ? pour l'étoile dans les automates signifie qu'on ne sait pas faire cette opération sous cette représentation.



La section traitant chaque passage est indiqué sur l'arc correpondant.

FIG. 6 - Graphe récapitulatif des passages entre les diverses représentations

Conclusion

Ce stage avait un double objectif : faire une synthèse des articles publiés sur les ensembles semi-linéaires et évaluer la complexité de leur utilisation, ce qui n'avait pas été fait jusque ici.

Pour ce qui est du premier objectif, nous avons présenté les principaux résultats sur ces ensembles, nous avons en particulier présenté les différentes opérations réalisables sur ces ensembles et leurs différents types de représentation.

En ce qui concerne le second objectif, nous avons évalué la complexité de tous les algorithmes trouvés dans la littérature. De plus, nous avons trouvé de nouveaux algorithmes moins complexes de changement de représentation qui n'étaient pas présents dans la littérature.
A titre d'exemple, on peut noter que l'on peut maintenant tester l'inclusion de deux ensembles semi-linéaires sous forme de bases et périodes en temps double exponentiel en utilisant notre passage aux automates, alors que sans ce passage, il semble qu'il fallait passer aux formules de Presburger, ce qui aurait eu un coût triple exponentiel.

Références

[WB00] P. Wolper et B. Boigelot. *On the construction of automata from linear arithmetic constraints*, Proc. 6th International Conference on Tools and Algorithms for the Construction and Analysis of Systems, volume 1785 de Lecture Notes in Computer Science, pages 1-19, Berlin, Mars 2000. Springer-Verlag.

[GS64] S. Ginsburg et E. Spanier. *Bounded ALGOL-Like Languages*, Trans. Amer. Math. Soc. 113 (1964). 333-368

[GS66] S. Ginsburg et E. Spanier. *Semigroups, Presburger formulas and languages* , Pacific Journal of mathematics, Vol. 16, N° 2, 1966

[Reu89] C. Reutenauer. *Aspects mathématiques des réseaux de Petri*, chapitres 2 et 3, Masson, Paris, 1989.

[BC96] A. Boudet et H. Comon. *Diophantine equations, Presburger arithmetic and finite automata*, Proceedings of CAAP'96, volume 1059 de Lecture Notes in Computer Science, pages 30-43. Springer-Verlag.

[Huy82] T. Huynh. *The complexity of semilinear sets*, Elektr. Inform. Kybern. E.I.K 18, 291-338

[FR74] M. J. Fischer et M. O. Rabin. *Super-Exponential Complexity of Presburger Arithmetic*, SIAM-AMS Proceedings Volume VII, 1974.