

Sur les extensions et les utilisations en Informatique d'un résultat Mathématique : Les polynômes d'Ehrhart

Philippe Clauss

Les polynômes d'Ehrhart constituent un résultat mathématique important, permettant d'exprimer le nombre de solutions entières d'un système d'équations et d'inéquations rationnelles et paramétriques. L'informatique trouve ici un outil permettant une analyse symbolique exacte des programmes. Nous montrons dans cet article que le mathématicien Eugène Ehrhart a posé les bases d'un vaste domaine d'investigations, tant en mathématique, permettant d'intéressantes extensions, qu'en informatique, où les applications de ses résultats augmentent encore les possibilités des ordinateurs.

Au début des années soixante, Eugène Ehrhart, mathématicien strasbourgeois bien connu des lecteurs de ce bulletin, a mis en évidence des résultats concernant le domaine de la géométrie algébrique. Plus précisément, ses résultats portent sur le nombre de solutions entières d'un système d'égalités et d'inégalités linéaires rationnelles, dépendant d'un paramètre entier positif [5, 6, 7, 9, 8, 10, 12, 13]. Par exemple, si l'on considère le système d'inégalités suivant :

$$\begin{cases} 0 & \leq x & \leq n \\ x + y & \leq 2n \\ 0 & \leq y & \leq 2n \end{cases}$$

où n est un paramètre entier positif. Le nombre de solutions entières de ce système est donné par son polynôme d'Ehrhart qui est $\frac{3}{2}n^2 + \frac{5}{2}n + 1$.

Ce problème étant bien cerné, E. Ehrhart l'a tout de suite posé en terme de problème géométrique. En effet, tout système d'équations linéaires définit un polyèdre convexe.

Ses résultats sont notamment d'un grand intérêt pour la recherche en informatique, et plus particulièrement en *informatique parallèle*, domaine de recherche dédié à la conception et à l'utilisation des ordinateurs les plus puissants du monde, appelés communément *super-calculateurs*. Nous montrons dans cet article, comment ils sont utilisés à résoudre de nombreux problèmes liés à l'utilisation de ces machines.

Ces ordinateurs sont caractérisés par une architecture matérielle composée de plusieurs processeurs, contrairement aux micro-ordinateurs que l'on trouve aujourd'hui sur nos bureaux, qui n'en possèdent qu'un seul. Les processeurs sont interconnectés afin de pouvoir se transmettre données et résultats de calculs, et s'occupent simultanément à résoudre le même problème général, comme par exemple le calcul des éléments d'une matrice de taille 10000×10000 .

Dans une première partie, nous rappelons quelques notions géométriques ainsi que les principaux résultats d'E. Ehrhart. Nous présentons ensuite nos extensions de ses résultats, à savoir, les polynômes d'Ehrhart à plusieurs variables, puis les polynômes d'Ehrhart de la projection affine d'un polytope. Nous décrivons, au paragraphe 4, comment sont calculés automatiquement ces polynômes par ordinateur. Au paragraphe 5, après avoir introduit le contexte informatique, nous montrons quelques utilisations à des problèmes d'analyse et de transformation de programmes.

I.- Géométrie et polynômes d'Ehrhart

Rappelons tout d'abord la définition d'un polyèdre convexe :

Definition 1 *Un polyèdre convexe P est un sous-ensemble de \mathbb{R}^d qui est l'intersection d'un nombre fini de demi-espaces fermés. Il peut être défini par un système d'inégalités linéaires de la forme $P = \{x \in \mathbb{R}^d \mid A \cdot x \leq B\}$, où A est une matrice de taille $m \times d$ de réels et B un vecteur de taille m de réels, m étant le nombre d'inégalités linéaires.*

Les équations composant les systèmes linéaires paramétriques considérés par E. Ehrhart sont de la forme générale suivante :

$$\sum_i a_i x_i < bn + c \quad \sum_i a_i x_i = bn + c \quad \sum_i a_i x_i \leq bn + c$$

où les a_i , b et c sont des constantes entières, les x_i sont les variables du système, et n est un paramètre entier positif.

Ces systèmes dépendant d'un paramètre entier positif n , les polyèdres ainsi définis sont donc également *paramétriques*. D'un point de vue géométrique, les solutions entières du système initial sont les points de coordonnées entières appartenant au polyèdre. Compter le nombre de solutions entières du système initial revient donc à compter le nombre de points entiers du polyèdre.

Le dénombrement des solutions entières n'a bien sûr de sens que si ce nombre est fini. Les polyèdres considérés sont donc eux aussi finis, ou *bornés*. On les appelle alors des *polytopes*. Ils sont l'enveloppe convexe d'un nombre fini de points de l'espace. On distingue les *polytopes entiers*, dont leurs sommets possèdent tous des coordonnées entières, et les *polytopes rationnels*, pour lesquels au moins un sommet possède des coordonnées rationnelles.

Le résultat majeur d'E. Ehrhart est d'avoir démontré que les nombres de points entiers de tels polytopes sont des polynômes en n , de degré égal à la dimension du plus petit espace contenant le polytope, si le polytope est entier, et sont des *pseudo-polynômes* de mêmes caractéristiques si le polytope est rationnel [11].

Les *pseudo-polynômes* sont des polynômes particuliers : leurs coefficients sont des *nombres périodiques*, c'est à dire des listes finies de valeurs rationnelles, où la valeur prise par le coefficient est déterminée par le rang de la valeur dans la liste, donné par la valeur du modulo de n par rapport au nombre de valeurs de cette liste. Pour être plus clair, prenons un exemple simple :

Exemple 1 Soit le pseudo-polynôme $f(n) = [0, 1, 2]n^2 + [1, \frac{1}{2}]n + [0, \frac{1}{4}]$. Selon les valeurs de n modulo 3 et n modulo 2, $f(n)$ prendra les valeurs suivantes :

$$\begin{aligned} \text{Si } n \text{ modulo } 3 = 1 \text{ et } n \text{ modulo } 2 = 1, & \quad f(n) = n \\ \text{Si } n \text{ modulo } 3 = 1 \text{ et } n \text{ modulo } 2 = 0, & \quad f(n) = \frac{1}{2}n + \frac{1}{4} \\ \text{Si } n \text{ modulo } 3 = 2 \text{ et } n \text{ modulo } 2 = 1, & \quad f(n) = n^2 + n \\ \text{Si } n \text{ modulo } 3 = 2 \text{ et } n \text{ modulo } 2 = 0, & \quad f(n) = n^2 + \frac{1}{2}n + \frac{1}{4} \\ \text{Si } n \text{ modulo } 3 = 0 \text{ et } n \text{ modulo } 2 = 1, & \quad f(n) = 2n^2 + n \\ \text{Si } n \text{ modulo } 3 = 0 \text{ et } n \text{ modulo } 2 = 0, & \quad f(n) = 2n^2 + \frac{1}{2}n + \frac{1}{4} \end{aligned}$$

□

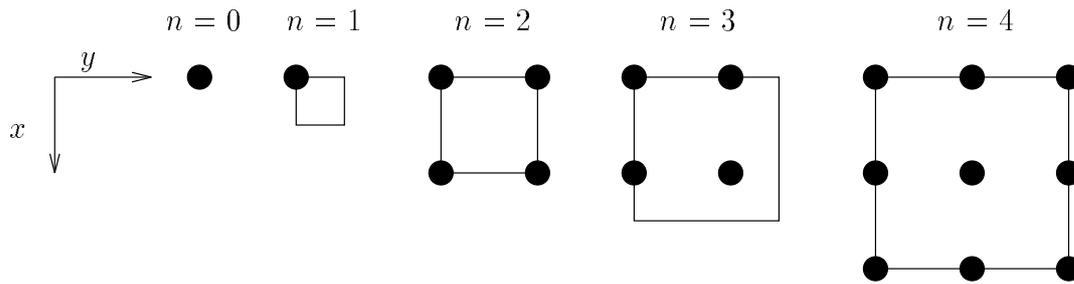
Le nombre maximum de valeurs prises par les coefficients périodiques est donné par le *dénominateur du polytope*, c'est à dire par le plus petit commun multiple des dénominateurs des coordonnées des sommets du polytope. Un autre résultat assure que le coefficient du terme du plus haut degré est constant (non périodique), si l'on est sûr que le système possède au moins une solution entière.

Voyons un exemple simple afin d'illustrer ces rappels.

Exemple 2 Soit S_n le système linéaire suivant :

$$(S_n) \begin{cases} x & \geq 0 \\ x & \leq \frac{1}{2}n \\ y & \geq 0 \\ y & \leq \frac{1}{2}n \\ n & \geq 0 \end{cases}$$

Ce système S_n définit un polytope rationnel paramétrique P_n et plus précisément, un carré dont les sommets ont pour coordonnées $(0, 0)$, $(\frac{1}{2}n, 0)$, $(0, \frac{1}{2}n)$ et $(\frac{1}{2}n, \frac{1}{2}n)$. Le carré et ses points entiers sont représentés sur la figure suivante pour n de 0 à 4 :



Le dénominateur de P_n est 2, la dimension de l'espace le contenant est 2, et on est sûr que S_n possède au moins une solution entière. Par conséquent, le pseudo-polynôme d'Ehrhart de P_n a donc la forme générale suivante : $pe(n) = c_1n^2 + [c_2, c_3]n + [c_4, c_5]$. Afin de déterminer les coefficients c_i , 5 valeurs numériques de $pe(n)$, $n \in [0, 4]$, déterminées en comptant les points entiers de la figure ci-dessus, permettent de poser un système d'égalités linéaires dont les solutions sont leurs valeurs numériques :

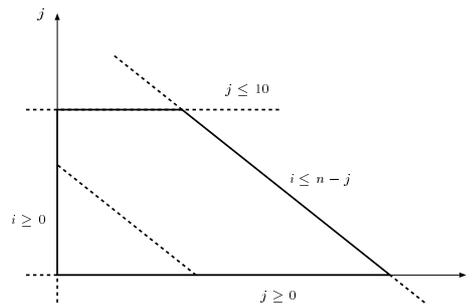
$$\left. \begin{array}{l} pe(0) = c_5 = 1 \\ pe(1) = c_1 + c_2 + c_4 = 1 \\ pe(2) = 4c_1 + 2c_3 + c_5 = 4 \\ pe(3) = 9c_1 + 3c_2 + c_4 = 4 \\ pe(4) = 16c_1 + 4c_3 + c_5 = 9 \end{array} \right\} \Rightarrow pe(n) = \frac{1}{4}n^2 + [\frac{1}{2}, 1]n + [\frac{1}{4}, 1]$$

Ainsi, le carré (ou système d'inéquations) défini avec $n = 9\,999\,999$ possède $100\,000\,009\,999\,998$ points entiers (ou solutions entières).

□

Plus généralement, la "forme" du polytope P_n varie selon des intervalles de valeurs de n . Ainsi, déterminer une expression paramétrique du nombre de points entiers contenus dans n'importe quel polytope paramétrique P_n , consiste tout d'abord, à décomposer P_n en une union de polytopes P_{n_1}, P_{n_2}, \dots , définis sur des domaines adjacents de valeurs du paramètre n , et pour lesquels les coordonnées des sommets sont des expressions affines en n , puis à calculer le polynôme d'Ehrhart pour chacun de ces polytopes.

Exemple 3 Dans la figure ci-dessous, le polytope défini par $P_n = \{(x, y) \mid x \geq 0, y \geq 0, x + y \leq n, y \leq 10\}$ est représenté, montrant le changement de sa forme par rapport au paramètre n .



Lorsque n est compris entre 0 et 10, P_n est un triangle, et lorsque n est supérieur à 10, P_n est un quadrilatère. Ainsi, P_n est décomposé en 2 polytopes :

$$P_n = \begin{cases} P_{n_1} & \text{si } 0 \leq n \leq 10, \text{ de sommets } (0, 0), (n, 0) \text{ et } (0, n) \\ P_{n_2} & \text{si } 10 \leq n, \text{ de sommets } (0, 0), (n, 0), (0, 10) \text{ et } (n - 10, 10) \end{cases}$$

□

II.- Les polynômes d'Ehrhart à plusieurs variables

Dans notre contexte d'utilisation de ces résultats, ceux-ci doivent être étendus à un nombre quelconque de paramètres. Les systèmes considérés plus généraux sont alors de la forme :

$$\sum_i a_i x_i < \sum_j b_j n_j + c \quad \sum_i a_i x_i = \sum_j b_j n_j + c \quad \sum_i a_i x_i \leq \sum_j b_j n_j + c$$

où les a_i , b_j et c sont des constantes entières, les x_i sont les variables du système, et les n_j sont des paramètres entiers positifs. E. Ehrhart a entamé une étude de ces systèmes. Cette étude se termine par la conjecture suivante [10, p. 139] :

Conjecture 1 (conjecture d'Ehrhart) *Pour tout système linéaire diophantien de dimension quelconque, dépendant linéairement de plusieurs paramètres entiers positifs, le nombre symbolique de solutions entières exprimé en fonction de ces paramètres s'exprime sur plusieurs domaines de valeurs de ces paramètres, par différents pseudo-polynômes.*

Nous démontrons cette conjecture dans [2]. Cette extension passe également par une re-définition des nombres périodiques en tableaux de valeurs rationnelles possibles de dimension égale au nombre de paramètres. Par exemple, $(-1)^{n-m}$ peut être représenté par le tableau à deux dimensions : $\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}_{n,m}$. Le résultat final de notre extension se traduit par le théorème suivant :

Théorème 1 (théorème fondamental d'Ehrhart étendu) *Soit P_N , $N = (n_1, n_2, \dots, n_p)$, un polytope dans un espace de dimension k , et tel que les coordonnées de ses sommets sont des expressions affines en N . Le nombre de points entiers de P_N est un polynôme à plusieurs variables n_1, n_2, \dots, n_p , de degré k , si P_N est entier, et est un pseudo-polynôme à plusieurs variables n_1, n_2, \dots, n_p , de degré k , et dont la pseudo-période est le dénominateur de P_N , si P_N est rationnel. Ce pseudo-polynôme est de la forme :*

$$pe(n_1, n_2, \dots, n_p) = \sum_{i_1=0}^k \sum_{i_2=0}^{k-i_1} \dots \sum_{i_p=0}^{k-i_1-i_2-\dots-i_{p-1}} c_{i_1, i_2, \dots, i_p} n_1^{i_1} n_2^{i_2} \dots n_p^{i_p}$$

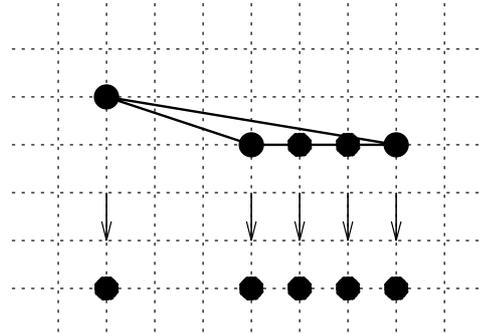
où les c_{i_1, i_2, \dots, i_p} sont des nombres périodiques définis par des tableaux de valeurs rationnelles de dimension p .

Notons que bien que les polynômes d'Ehrhart soient une expression paramétrique du nombre de solutions entières, ils servent également au dénombrement des solutions entières de systèmes d'inéquations instanciés, ou non-paramétriques, par simple introduction d'un paramètre "artificiel" dans les inéquations.

III.- Le polynôme d'Ehrhart de la projection affine d'un polytope

La projection affine d'un polytope pose une difficulté supplémentaire. Les points entiers contenus dans un polytope P_N sont organisés en un réseau régulier. Si l'on applique une projection affine à un tel polytope, l'objet géométrique résultant n'est plus, en général, un réseau régulier de points entiers. On ne peut donc plus modéliser ces points par leur enveloppe convexe, comme il est montré

sur l'exemple de la figure ci-dessous.



Le dénombrement des points résultants d'une telle projection affine ne peut donc plus s'effectuer par la méthode décrite dans les paragraphes précédents. Toutefois, là aussi, les polynômes d'Ehrhart nous apportent l'information nécessaire au calcul du polynôme d'Ehrhart d'une projection affine d'un polytope (paramétrique ou non). Dans le cas d'une projection d'un polytope paramétrique, nous obtenons le résultat suivant démontré dans [3] :

Théorème 2 *Le nombre de points entiers, résultants d'une projection affine des points entiers contenus dans un polytope paramétrique, est défini sur plusieurs domaines adjacents de valeurs des paramètres, par différents polynômes d'Ehrhart.*

Un exemple d'application informatique de ces résultats à un problème d'accès en mémoire locale (ou mémoire cache) d'un processeur par un programme est décrit dans la suite. Voyons à présent comment le calcul de ces polynômes s'effectue de manière automatique par un ordinateur.

IV.- Le calcul des polynômes d'Ehrhart par ordinateur

Nous avons écrit un programme en langage *C* calculant les polynômes d'Ehrhart (à une ou plusieurs variables) pour n'importe quel polytope entier ou rationnel défini par un système d'inégalités linéaires paramétriques de la forme décrite plus haut. Ce programme se décompose en plusieurs étapes principales de calculs :

1. Lecture du système d'inégalités paramétriques S_N .
2. Calcul des coordonnées paramétriques des sommets de P_N défini par S_N , et des domaines d'existence de ces sommets définis par q contraintes linéaires sur les paramètres, de la forme $C \cdot N \leq D$, où C est une matrice de taille $q \times p$ et D un vecteur de taille q . Pour cette étape, nous utilisons un programme écrit par V. Loechner et D.K. Wilde décrit dans [17].

Chaque domaine de définition des sommets paramétriques correspond à un polytope associé à un unique polynôme d'Ehrhart : les coordonnées des sommets sur un domaine donné sont des combinaisons affines des paramètres.

3. Pour chacun de ces domaines d'existence, calcul du polynôme d'Ehrhart correspondant, par résolutions successives de systèmes d'équations linéaires symboliques.

Les valeurs initiales numériques nécessaires à la résolution des systèmes s'obtiennent par la détermination préalable de *boucles de parcours* du polytope P_N . Pour le lecteur non familier du jargon informatique, il s'agit de structures de contrôle d'un programme permettant d'effectuer un même traitement sur un domaine convexe de points entiers, par parcours de ce domaine selon l'ordre *lexicographique* des coordonnées. Dans notre cas, ce traitement consiste tout simplement en un dénombrement des éléments de ce domaine d'indices, pour quelques petites valeurs des paramètres.

Ces boucles de parcours s'obtiennent notamment par la méthode de Fourier-Motzkin [19], dont nous utilisons une implémentation logicielle.

Exemple 4 Considérons le système d'inégalités linéaires suivant :

$$H_{n,m} \begin{cases} 0 \leq x \leq \frac{m-1}{2} \\ 0 \leq y \leq \frac{m-2x-1}{2} \\ x + y \geq m - n - 1 \\ 1 \leq m \leq 2n + 1 \end{cases}$$

A partir de cette définition du polytope $P_{n,m}$, le programme calcule tout d'abord les coordonnées paramétriques de ses sommets ainsi que leurs domaines d'existence :

- Si $1 \leq m \leq n + 1$ (**domaine 1**), alors les coordonnées des sommets de $P_{n,m}$ sont $(\frac{m-1}{2}, 0)$, $(0, \frac{m-1}{2})$ et $(0, 0)$.
- Si $n + 1 \leq m \leq 2n + 1$ (**domaine 2**), alors les coordonnées des sommets de $P_{n,m}$ sont $(m - n - 1, 0)$, $(\frac{m-1}{2}, 0)$, $(0, m - n - 1)$ et $(0, \frac{m-1}{2})$.

Sur chacun des domaines 1 et 2, un polynôme d'Ehrhart est calculé. Voyons cela en détail pour le domaine 1.

Le polynôme d'Ehrhart $pe(n, m)$ est exprimé d'abord par un polynôme à une variable (un des paramètres). Son degré est donné par la dimension du polytope $P_{n,m}$, ici 2, et la période des coefficients par son dénominateur, ici 2. D'où $pe(n, m) = c_1 m^2 + [c_2, c_3] m + [c_4, c_5]$, où les c_i sont des pseudo-polynômes en n de mêmes caractéristiques. Le calcul des 5 coefficients c_1, \dots, c_5 s'effectue par le système linéaire de 5 équations suivant, utilisant 5 valeurs initiales symboliques $pe(n, 1), \dots, pe(n, 5)$:

$$\begin{cases} c_1 + c_2 + c_4 & = & pe(n, 1) \\ 4c_1 + 2c_3 + c_5 & = & pe(n, 2) \\ 9c_1 + 3c_2 + c_4 & = & pe(n, 3) \\ 16c_1 + 4c_3 + c_5 & = & pe(n, 4) \\ 25c_1 + 5c_2 + c_4 & = & pe(n, 5) \end{cases}$$

La résolution de ce système nécessite le calcul préalable de $pe(n, 1), \dots, pe(n, 5)$, qui possèdent tous la même forme générale suivante : $c_1 n^2 + [c_2, c_3] n + [c_4, c_5]$. Nous avons donc 5 systèmes linéaires de 5 équations à résoudre, nécessitant chacun la détermination de 5 valeurs numériques initiales. Pour cela, notre programme détermine une boucle de parcours de $P_{n,m}$:

```
init_val := 0 ;
pour x de 0 à (m-1)/2 faire
    pour y de max(0, m-n-x-1) à (m-2*x-1)/2 faire
        init_val := init_val + 1 ;
```

Par exemple, pour le calcul de $pe(n, 1)$, nous exécutons 5 fois cette boucle de parcours avec $m = 1$, et en faisant varier la valeur de n à chaque fois, de 1 à 5. Les valeurs numériques ainsi obtenues permettent alors de résoudre le système suivant :

$$\begin{cases} c_1 + c_2 + c_4 & = & pe(1, 1) & = & 1 \\ 4c_1 + 2c_3 + c_5 & = & pe(2, 1) & = & 1 \\ 9c_1 + 3c_2 + c_4 & = & pe(3, 1) & = & 1 \\ 16c_1 + 4c_3 + c_5 & = & pe(4, 1) & = & 1 \\ 25c_1 + 5c_2 + c_4 & = & pe(5, 1) & = & 1 \end{cases} \Rightarrow pe(n, 1) = 1$$

De la même manière, le programme calcule $pe(n, 2) = 1$, $pe(n, 3) = 3$, $pe(n, 4) = 4$, $pe(n, 5) = 6$. Le tout premier système peut donc maintenant être résolu :

$$\forall m \in [1..n + 1], pe(n, m) = \frac{1}{8} m^2 + \left[\frac{1}{2}, \frac{1}{4} \right]_m + \left[\frac{3}{8}, 0 \right]_m$$

L'application des mêmes opérations sur le domaine 2 donne le résultat suivant :

$$\forall m \in [n + 1..2n + 1], pe(n, m) = nm - \frac{1}{2} n^2 - \frac{1}{2} n - \frac{3}{8} m^2 + \left[1, \frac{3}{4} \right]_m m + \left[\frac{3}{8}, 0 \right]_m$$

□

Voyons à présent quelques cas d'utilisation de ces mathématiques à l'informatique parallèle.

V.- Polynômes d'Ehrhart et Informatique Parallèle

Nous nous intéressons plus particulièrement au domaine de la parallélisation automatique de programmes séquentiels en informatique parallèle. Son objectif est de déterminer des méthodes efficaces permettant de détecter et d'exploiter le parallélisme inhérent à une séquence d'opérations d'un programme séquentiel "classique". Afin d'introduire l'idée générale de ces méthodes, prenons un exemple évident d'une suite d'additions :

```
X := A + B ;
Y := C + D ;
Z := X + Y ;
```

Sur un ordinateur possédant un seul processeur, l'ordre de calcul sera d'abord le calcul de **X**, puis le calcul de **Y**, et finalement le calcul de **Z**. Mais pour accélérer le calcul, on peut remarquer que le calcul de **Y** peut s'effectuer indépendamment du calcul de **X**. Par conséquent, sur un ordinateur possédant au moins deux processeurs, **X** et **Y** peuvent être calculés simultanément sur deux processeurs différents. Par contre, le calcul de **Z** ne peut évidemment pas s'effectuer avant ceux de **X** ou de **Y**. Par conséquent, **Z** doit être calculé après **X** et **Y** sur n'importe quel processeur.

La parallélisation automatique possède son plus large champ d'applications dans les programmes séquentiels de types scientifiques : calculs numériques, simulations physiques, ... Ces programmes, écrits le plus souvent dans le langage de programmation *fortran*, ont un parallélisme inhérent situé à 80 % dans des structures de contrôle appelées communément *boucles*. Ces boucles permettent d'exprimer la répétition d'une suite d'opérations relativement à une variation de valeurs d'indices. Elles peuvent être schématisées de la manière suivante :

```
Pour i de inf(i) a sup(i) faire
  Pour j de inf(j) a sup(j) faire
    .....
    operation 1 ;
    operation 2 ;
    .....
    operation n ;
    .....
  Fin Pour
Fin Pour
```

Ainsi, l'indice de la boucle la plus interne, c'est à dire la plus proche de la suite d'opération est incrémenté de 1 après l'exécution de l'**operation n**, puis la suite d'opérations est ré-exécutée. Sa valeur initiale est donnée par la valeur de **inf(..)** et sa valeur maximum par **sup(..)**. Dès que sa valeur maximum est atteinte, la suite d'opérations est exécutée une dernière fois puis l'indice de la boucle de niveau juste supérieur est incrémenté. Si sa valeur maximum n'est pas atteinte, la boucle la plus interne est à nouveau exécutée pour toutes les valeurs de son indice. Par exemple, un programme permettant d'afficher tous les éléments d'une matrice *A* de taille $N \times N$ pourrait être le suivant :

```
Pour i de 1 a N faire
  Pour j de 1 a N faire
    Afficher[A(i,j)]
  Fin Pour
Fin Pour
```

Un ensemble de structures de boucle de ce type, incluses les unes dans les autres est appelé communément un *nid de boucles*.

De plus, les bornes d'indices **inf(..)** et **sup(..)** sont en général des fonctions affines des indices des boucles de niveaux supérieurs, et les opérations portent souvent sur des éléments de tableaux, ou de matrices, dont les indices sont eux-mêmes exprimés par des fonctions affines des indices de boucles.

Un exemple simple et bien connu est le produit matriciel. Tout mathématicien connaît la formule $c_{ij} = \sum_{k=1}^N a_{ik}b_{kj}$, exprimant le produit de deux matrices carrées A et B de tailles $N \times N$ en une matrice C . Ce calcul peut être programmé en un nid de boucles :

```

Pour i de 1 a N faire
  Pour j de 1 a N faire
    Pour k de 1 a N faire
      c[i,j] := c[i,j] + a[i,k] * b[k,j]
    Fin Pour
  Fin Pour
Fin Pour

```

où le calcul d'un élément $c(i, j)$ est effectué en N pas par cumul dans $c(i, j)$.

Nos méthodes de parallélisation automatique passent par une modélisation géométrique d'un tel nid de boucles. Chaque pas de calcul de tous les éléments $c(i, j)$ est représenté dans l'espace par un point de coordonnées entières (i, j, k) . Ainsi, le produit matriciel est représenté par un cube défini par les bornes inférieures et supérieures de tous les indices de boucles i, j et k , dans lequel seuls les points à coordonnées entières sont significatifs : ce sont les solutions entières du système défini par :

$$\begin{cases} 1 \leq i \leq N \\ 1 \leq j \leq N \\ 1 \leq k \leq N \end{cases}$$

Dans ce cas, leur nombre, ou le polynôme d'Ehrhart du cube, est facile à déterminer, mais permet de vérifier la méthode : $pe(n) = N^3$. Ce nombre nous permet dans ce cas d'exprimer le nombre de calculs élémentaires effectués par notre programme. Cette information est utile notamment pour évaluer le temps d'exécution du programme, ou pour réaliser l'équilibrage de charge entre les processeurs, c'est à dire allouer les calculs élémentaires en nombre égal pour chaque processeur de l'ordinateur.

Voyons maintenant quelques applications particulières des polynômes d'Ehrhart à l'analyse et à la transformation de programmes parallèles.

Le parallélisme potentiel

La parallélisation d'un programme consiste non seulement à une allocation des calculs élémentaires à des processeurs mais également à un ordonnancement dans le temps de ces calculs. Une méthode aujourd'hui bien connue définit une fonction linéaire d'allocation des calculs de la forme :

$$\begin{aligned} \mathbb{Z}^n &\longrightarrow \mathbb{Z}^{n-1} \\ alloc : X &\longmapsto D \cdot X \end{aligned}$$

où X est un point à coordonnées entières associé à un calcul, et D est une matrice de taille $(n-1) \times n$. Cette fonction revient à projeter le polyèdre convexe défini par les points de calculs, que l'on appelle le *domaine de calcul*, selon un vecteur \vec{p} . La matrice D est donc telle que la matrice $\begin{pmatrix} D \\ \vec{p} \end{pmatrix}$ définit une base de \mathbb{Z}^n . Ainsi, $alloc(X)$ nous donne les coordonnées du processeur effectuant le calcul X .

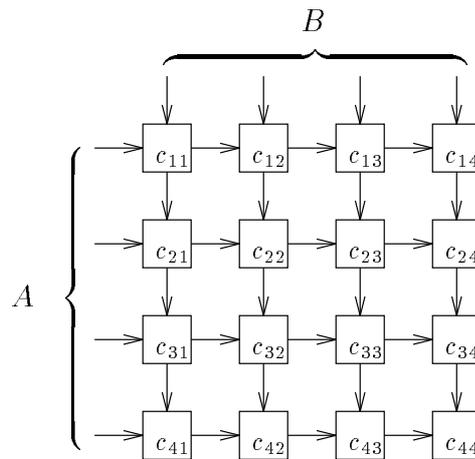
L'ordonnancement temporel des calculs s'exprime également par une fonction linéaire, exprimant une modélisation discrète du temps, et considérant que tous les calculs élémentaires prennent une même durée, c'est à dire 1 :

$$\begin{aligned} \mathbb{Z}^n &\longrightarrow \mathbb{Z} \\ pas : X &\longmapsto \vec{t} \cdot X + c \end{aligned}$$

Les vecteurs \vec{t} et \vec{p} doivent vérifier la relation $\vec{p} \cdot \vec{t} \neq 0$, exprimant que deux calculs X et Y de mêmes instants de calculs, c'est à dire tels que $pas(X) = pas(Y)$, ne peuvent pas être effectués par le même processeur. De plus, lorsque le calcul d'un élément utilise un autre élément calculé par

le programme, ce calcul ne peut se faire qu'après le calcul de l'élément utilisé. On parle alors de dépendances entre les calculs, modélisées par des *vecteurs de dépendances* \vec{d} entre les points entiers du domaine de calcul. Le respect de ces dépendances est réalisé si et seulement si $\vec{d} \cdot \vec{t} > 0$ pour tous vecteurs \vec{d} .

Pour notre exemple du produit matriciel, un ordonnancement temporel et une allocation valides sont définis respectivement par $pas(i, j, k) = i + j + k - 2$ et $alloc(i, j, k) = (i, j)$. Ces choix correspondent à un programme parallèle pouvant être représenté par un réseau carré de taille $N \times N$ de processeurs interconnectés et d'exécution rythmée par une horloge globale, où chaque processeur de coordonnées (i, j) est dédié au calcul d'un élément c_{ij} de la matrice C . De plus, les connections entre les processeurs servent à la circulation des éléments des matrices A et B dans un ordre et une direction bien définis, correspondant aux besoins en données de chaque processeur pour son calcul. Un tel réseau est représenté ci-dessous pour $N = 4$.



Les calculs simultanés d'un instant donné t_0 , tels que $pas(X) = t_0$, sont par conséquent définis géométriquement par les points entiers appartenant à l'intersection entre un hyperplan orthogonal au vecteur \vec{t} et le domaine de calcul. Leur nombre définit le nombre de calculs simultanés à l'instant t_0 , ou *parallélisme potentiel* à t_0 .

Pour notre exemple du produit matriciel avec $pas(i, j, k) = i + j + k - 2$, ces calculs simultanés à t_0 sont définis par :

$$\begin{cases} 1 \leq i \leq N \\ 1 \leq j \leq N \\ 1 \leq k \leq N \\ i + j + k - 2 = t_0 \end{cases}$$

Le nombre de points entiers du polytope paramétrique P_{N,t_0} ainsi défini correspond donc au parallélisme potentiel de l'instant t_0 . Le calcul du polynôme d'Ehrhart de P_{N,t_0} donne le résultat suivant :

$$\begin{aligned} \text{Si } 1 \leq t_0 \leq N, & \quad pe(N, t_0) = \frac{1}{2}t_0^2 + \frac{1}{2}t_0 \\ \text{Si } N \leq t_0 \leq 2N - 1, & \quad pe(N, t_0) = -\frac{3}{2}N^2 - t_0^2 + 3Nt_0 - t_0 \\ \text{Si } 2N - 1 \leq t_0 \leq 3N - 2, & \quad pe(N, t_0) = \frac{9}{2}N^2 + \frac{1}{2}t_0^2 - 3Nt_0 - \frac{3}{2}N + \frac{1}{2}t_0 \end{aligned}$$

Il est intéressant de maximiser ces polynômes, afin d'obtenir le nombre maximum symbolique de calculs simultanées, $maxpar(N)$, atteint lors de l'exécution du programme parallèle. Cette information peut être nécessaire pour déterminer le nombre maximum de processeurs simultanément actifs, et donc le nombre minimum de processeurs utiles à l'exécution de ce programme. Cette maximisation ne posant pas de problème mathématique particulier, nous en donnons juste le résultat final :

$$\begin{aligned} \text{Si } N \leq 7, & \quad maxpar(N) = \frac{N(N+1)}{2} \\ \text{Si } N > 7, & \quad maxpar(N) = \frac{3}{4}N^2 - \frac{3}{2}N + \left[\frac{1}{4}, 0\right] \end{aligned}$$

Ces quantités peuvent être comparées au nombre de processeurs, np , résultant de la fonction d'allocation *alloc*, afin de déterminer le taux d'utilisation du réseau de processeurs. Dans notre exemple, np est facile à déterminer ($np(N) = N^2$). Mais dans le cas plus général d'une fonction *alloc* quelconque, la détermination de np nécessite l'utilisation de la méthode présentée au paragraphe 4.

Pour l'exemple, le calcul du ratio $\frac{maxpar(N)}{np(N)}$ nous informe qu'environ 3/4 des processeurs au maximum sont utilisés simultanément pour N grand.

La taille des mémoires caches

Nous utilisons ici l'extension des polynômes d'Ehrhart aux projections affines de polytopes décrite au paragraphe 4.

Considérons le programme suivant :

```

Pour i de 1 a 8 faire
  Pour j de 1 a 5 faire
    a(6i+9j-7) := a(6i+9j-7) + 5
  Fin Pour
Fin Pour

```

Un objectif classique en analyse de programmes est de déterminer le nombre de données différentes accédées lors de l'exécution. Ce nombre sert notamment à évaluer la taille mémoire nécessaire au stockage des données utiles au programme. Si celui-ci est exécuté sur un seul processeur d'une machine parallèle à mémoire distribuée, c'est-à-dire où chaque processeur possède une mémoire locale (ou cache), cette taille concernera uniquement la mémoire de ce processeur.

Dans notre exemple, les données accédées sont des éléments d'un tableau à une dimension a . Ces éléments sont référencés par une combinaison affine des indices de boucle du programme, $a(6i + 9j - 7)$. Déterminer le nombre de données accédées par le programme revient à déterminer le nombre de valeurs différentes de $6i + 9j - 7$ atteintes pour $1 \leq i \leq 8$ et $1 \leq j \leq 5$. Ce problème revient donc à dénombrer les points entiers résultant de l'application de la projection affine définie par $Proj_Aff(i, j) = 6i + 9j - 7$, aux points entiers du polytope défini par $P = \{(i, j) \in \mathbb{Z}^2 \mid 1 \leq i \leq 8, 1 \leq j \leq 5\}$.

Ce problème relève bien de la méthode du paragraphe 4. Le polytope paramétré, caractérisant les points de P qui résultent en une même valeur $y = 6i + 9j - 7$, est défini par $P_y = \{(i, j) \in \mathbb{Z}^2 \mid 1 \leq i \leq 8, 1 \leq j \leq 5, y = 6i + 9j - 7\}$. On calcule les polynômes d'Ehrhart de P_y . Le résultat consiste en 3 domaines adjacents D_1 , D_2 et D_3 de valeurs de y , chacun étant associé à un polynôme d'Ehrhart $pe_1(y)$, $pe_2(y)$ et $pe_3(y)$:

- $D_1 = \{y \in \mathbb{Z} \mid 8 \leq y \leq 44\}$,
 $pe_1(y) = [0, \frac{1}{18}, 0]y + [0, -\frac{1}{9}, 0, 0, -\frac{5}{18}, 0, 0, \frac{5}{9}, 0, 0, -\frac{11}{18}, 0, 0, \frac{2}{9}, 0, 0, \frac{1}{18}, 0]$
- $D_2 = \{y \in \mathbb{Z} \mid 44 \leq y \leq 50\}$,
 $pe_2(y) = [0, 3, 0, 0, 2, 0]$
- $D_3 = \{y \in \mathbb{Z} \mid 50 \leq y \leq 86\}$,
 $pe_3(y) = [0, -\frac{1}{18}, 0]y + [0, \frac{46}{9}, 0, 0, \frac{95}{18}, 0, 0, \frac{49}{9}, 0, 0, \frac{83}{18}, 0, 0, \frac{52}{9}, 0, 0, \frac{89}{18}, 0]$

Par exemple, l'élément de tableau $a(41)$ est référencé 2 fois par le programme : puisque $41 \text{ modulo } 3 = 2$ (la 2ème valeur du 1er coefficient périodique est prise en compte) et $41 \text{ modulo } 18 = 5$ (la 5ème valeur du 2ème coefficient périodique est prise en compte), on obtient $pe_1(41) = \frac{41}{18} - \frac{5}{18} = 2$.

Les domaines de valeurs de y sont transformés en domaines disjoints : $D_1 = \{y \in \mathbb{Z} \mid 8 \leq y \leq 44\}$, $D_2 = \{y \in \mathbb{Z} \mid 44 < y < 50\}$ et $D_3 = \{y \in \mathbb{Z} \mid 50 \leq y \leq 86\}$. Sur chaque domaine D_q et pour tout polynôme $pe_v(y)$ extrait de $pe_q(y)$, tel qu'il n'est pas égal à zéro pour toute valeur de y , on définit un polytope pour lequel les points entiers doivent être dénombrés. On obtient les polytopes suivants

:

$$\begin{array}{l}
 D_1 \left\{ \begin{array}{l} \{8 \leq y \leq 44, y \text{ modulo } 18 = 2\} \\ \{8 \leq y \leq 44, y \text{ modulo } 18 = 5\} \\ \{8 \leq y \leq 44, y \text{ modulo } 18 = 8\} \\ \{8 \leq y \leq 44, y \text{ modulo } 18 = 11\} \\ \{8 \leq y \leq 44, y \text{ modulo } 18 = 14\} \\ \{8 \leq y \leq 44, y \text{ modulo } 18 = 17\} \end{array} \right. \quad D_2 \left\{ \begin{array}{l} \{44 < y < 50, y \text{ modulo } 6 = 2\} \\ \{44 < y < 50, y \text{ modulo } 6 = 5\} \end{array} \right. \\
 \\
 D_3 \left\{ \begin{array}{l} \{50 \leq y \leq 86, y \text{ modulo } 18 = 2\} \\ \{50 \leq y \leq 86, y \text{ modulo } 18 = 5\} \\ \{50 \leq y \leq 86, y \text{ modulo } 18 = 8\} \\ \{50 \leq y \leq 86, y \text{ modulo } 18 = 11\} \\ \{50 \leq y \leq 86, y \text{ modulo } 18 = 14\} \\ \{50 \leq y \leq 86, y \text{ modulo } 18 = 17\} \end{array} \right.
 \end{array}$$

L'application de la transformation décrite dans la proposition 4 donne les polytopes suivants :

$$D_1 \left\{ \begin{array}{l} \{1 \leq z \leq 2\} \\ \{1 \leq z \leq 2\} \\ \{0 \leq z \leq 2\} \\ \{0 \leq z \leq 1\} \\ \{0 \leq z \leq 1\} \\ \{0 \leq z \leq 1\} \end{array} \right. \quad D_2 \left\{ \begin{array}{l} \{7 < z < 8\} \\ \{7 \leq z \leq 7\} \end{array} \right. \quad D_3 \left\{ \begin{array}{l} \{3 \leq z \leq 4\} \\ \{2 \leq z \leq 4\} \\ \{2 \leq z \leq 3\} \end{array} \right.$$

Le dénombrement des points entiers de ces polytopes est facile et résulte en $2 + 2 + 3 + 2 + 2 + 2 + 0 + 1 + 2 + 2 + 2 + 2 + 3 + 2 = 27$.

Pour tous les polynômes $pe_v(y)$ non toujours nuls, on calcule leurs racines afin de déterminer N_r . Par exemple, on a les polynômes $pe_v(y)$ suivants sur le domaine D_1 :

$$D_1 \left\{ \begin{array}{l} Si \ y \text{ modulo } 18 = 2, \quad pe_2(y) = \frac{1}{18}y - \frac{1}{9} \\ Si \ y \text{ modulo } 18 = 5, \quad pe_5(y) = \frac{1}{18}y - \frac{5}{18} \\ Si \ y \text{ modulo } 18 = 8, \quad pe_8(y) = \frac{1}{18}y + \frac{5}{9} \\ Si \ y \text{ modulo } 18 = 11, \quad pe_{11}(y) = \frac{1}{18}y - \frac{11}{18} \\ Si \ y \text{ modulo } 18 = 14, \quad pe_{14}(y) = \frac{1}{18}y + \frac{2}{9} \\ Si \ y \text{ modulo } 18 = 17, \quad pe_{17}(y) = \frac{1}{18}y + \frac{1}{18} \end{array} \right.$$

Le calcul de leurs racines résulte en une seule racine valide 11. Pour les autres domaines, on détermine une seule autre racine valide sur D_3 qui est 83. Par conséquent, l'on doit soustraire 2 au décompte précédent.

En conclusion, le nombre de données différentes accédées par le programme, ou la taille mémoire nécessaire au programme, est de 25.

VI.- Conclusion

Les extensions et applications des polynômes d'Ehrhart nous ont amenés à des publications dans des conférences et revues d'audience internationales spécialisées en informatique [1, 2, 3, 4], et bien d'autres sont à prévoir. Ces résultats sont aujourd'hui reconnus et utilisés dans d'autres travaux [14, 15, 18, 16]. Beaucoup d'autres utilisations auraient pu être présentées ici, et peuvent être espérées moyennant une modélisation opportune des problèmes posés. L'analyse symbolique permise par les travaux d'Eugène Ehrhart, est non seulement intéressante pour résoudre des problèmes de nature paramétrée, mais l'est aussi pour tout autre type de problèmes grâce à des modélisations constructives. Pour les lecteurs ayant accès à internet, une page régulièrement mise à jour, et donnant toutes les informations et références de nos travaux sur le sujet est accessible à l'adresse suivante : <http://icps.u-strasbg.fr/Ehrhart/Ehrhart.html>

Les polynômes d'Ehrhart n'intéressent pas uniquement l'informatique. Les mathématiques s'y intéressent également beaucoup aujourd'hui, se posant des questions telles que le sens géométrique des coefficients de ces polynômes.

Rappelons qu'E. Ehrhart a obtenu deux prix de l'Académie des Sciences pour ses travaux arithmo-géométriques.

Remerciements

Je tiens à remercier Monsieur Eugène Ehrhart pour l'honneur qu'il me fait de nos rencontres, de son aide et son soutien à mes travaux.

Bibliographie

- [1] Ph. Clauss. Counting Solutions to Linear and Nonlinear Constraints through Ehrhart polynomials: Applications to Analyze and Transform Scientific Programs. *10th ACM Int. Conf. on Supercomputing*, Philadelphia, May 1996. Egalement accessible comme rapport de recherches ICPS 96-03. <http://icps.u-strasbg.fr/pub-96/pub-96-03.ps.gz>
- [2] Ph. Clauss et V. Loechner. Parametric Analysis of Polyhedral Iteration Spaces. *IEEE Int. Conf. on Application Specific Array Processors, ASAP'96*, Chicago, Illinois, August 1996. Version étendue à paraître dans *Journal of VLSI Signal Processing*, Kluwer Academic Publishers, Boston, USA, 1997. Egalement accessible comme rapport de recherches ICPS 96-04. <http://icps.u-strasbg.fr/pub-96/pub-96-04.ps.gz>
- [3] Ph. Clauss. Handling Memory Cache Policy with Integer Points Countings. *Third International Conference Euro-par'97*, Passau, Allemagne, Août 1997. Lecture Notes in Computer Science 1300, Springer, pages 285-293. Egalement accessible comme rapport de recherches ICPS 97-02. <http://icps.u-strasbg.fr/pub-97/pub-97-02.ps.gz>
- [4] Ph. Clauss, V. Loechner et D. Wilde. Deriving Formulae to Count Solutions to Parameterized Linear Systems using Ehrhart Polynomials: Applications to the Analysis of Nested-Loop Programs. Soumis à *Journal of Symbolic Computation*. Egalement accessible comme rapport de recherches ICPS 97-05. <http://icps.u-strasbg.fr/pub-97/pub-97-05.ps.gz>
- [5] E. Ehrhart. Sur les polyèdres rationnels homothétiques à n dimensions. *C.R. Acad. Sci. Paris*, 254:616-618, 1962.
- [6] E. Ehrhart. Sur un problème de géométrie diophantienne linéaire I. *J. Reine Angew. Math.*, 226:1-29, 1967.
- [7] E. Ehrhart. Sur un problème de géométrie diophantienne linéaire II. *J. Reine Angew. Math.*, 227:25-49, 1967.
- [8] E. Ehrhart. Une méthode géométrique en combinatoire, *l'Ouvert*, bulletin de la régionale alsacienne de l'A.P.M.E.P., déc. 1976.
- [9] E. Ehrhart. Sur les systèmes d'inéquations diophantiennes linéaires, *J. Reine Angew. Math.*, 262/263, 45-57, 1973.
- [10] E. Ehrhart. *Polynômes arithmétiques et Méthode des Polyèdres en Combinatoire*. International Series of Numerical Mathematics, vol.35, Birkhäuser Verlag, Basel/Stuttgart, 1977.
- [11] E. Ehrhart. Sur les polygones entiers ou rationnels, *l'Ouvert*, bulletin de la régionale alsacienne de l'A.P.M.E.P., sept. 1993.
- [12] E. Ehrhart. Un théorème arithmo-géométrique et ses généralisations, *l'Ouvert*, bulletin de la régionale alsacienne de l'A.P.M.E.P., déc. 1994.
- [13] E. Ehrhart. La loi de réciprocité diophantienne, *l'Ouvert*, bulletin de la régionale alsacienne de l'A.P.M.E.P., n. 394, 1994.
- [14] S. Ghosh, M. Martonosi et S. Malik. Cache Miss Equations: An Analytical Representation of Cache Misses. *11th ACM Int. Conf. on Supercomputing*, Vienne, Autriche, juillet 1997.
- [15] D.-C. R. Ju, J.-F. Collard and K. Oukbir: Probabilistic Memory Disambiguation and its Application to Data Speculation. Rapport de recherche PRiSM 96-010, Université de Versailles, 1996. [ftp://ftp.prism.uvsq.fr/pub/reports/1996/1996.010.ps.gz](http://ftp.prism.uvsq.fr/pub/reports/1996/1996.010.ps.gz)

- [16] V. Loechner. Contribution à l'étude des polyèdres paramétrés et applications en parallélisation automatique. Thèse d'informatique de l'Université Louis Pasteur, décembre 1997.
- [17] V. Loechner and D. K. Wilde. Parameterized polyhedra and their vertices. *International Journal of Parallel Programming*, vol. 25, no. 6, décembre 1997. Egalement rapport de recherche ICPS 96-09, Université Louis Pasteur, 1996. <http://icps.u-strasbg.fr/pub-96/pub-96-09.ps.gz>
- [18] R. Sakellariou, J. R. Gurd. Compile-Time Minimisation of Load Imbalance in Loop Nests. *11th ACM Int. Conf. on Supercomputing*, Vienne, Autriche, juillet 1997.
- [19] G. M. Ziegler. *Lectures on Polytopes*. Graduate Texts in Mathematics, Springer-Verlag, New-York, 1995.