Projecting Periodic Polyhedra for Loop Nest Analysis

Benoît Meister
ICPS - LSIIT
Pôle API
Bd Sebastian Brant
F-67400 Illkirch
meister@icps.u-strasbg.fr

4-30-2004

Abstract

The polytope model is a powerful framework to manipulate, parallelize and optimize nested *for* loops in compilers. This paper, based on an extension of this theory to the concept of periodic polyhedra, shows precise methods analysis methods to achieve dead code elimination in parameterized loop nests, and to compute the exact volume of data accessed by loop nests through integer multidimensional affine reference functions. We determine the values of the parameters for which a considered loop nest is actually executed, allowing dead code elimination in loop nests. We also determine the number of points with integer coordinates belonging to the integer projection of a Z-polyhedron, which indicates the exact number of data accessed by a loop nest through (especially) non-invertible array access functions. These methods integrate the formalism used in the polyhedral library *Polylib*.

1 Motivation

For many years, compiler writers have focused on parameterized loop nests, mainly because of their importance in scientific and multimedia programs. The *polytope model* (with e.g. [1, 18, 9, 16, 19]) allows to manipulate loop nests whose bounds are affine functions of the loop indices with integer-valued parameters in the constant part by modeling them as parameterized rational polytopes P. As the considered loop indices are incremented by a constant integer value, the values taken by the n-vector of indices belong to a subset of $\mathbb{Z}^n$: an integer lattice L. So the values taken by the index vector $I \in \mathbb{Z}^n$, where n defines the number of nested loops, are given by the so-called $\mathbb{Z}$-*polytope* $P \cap L$.

Example 1 The iterations of the following Gaussian elimination code:

```
for(i=1;i<=n; i++)
  for(j=i+1;j<=n;j++)
    for(k=i+1; k<=n; k++)
      a[j][k]=a[j][k]-a[j][i]*a[i][k]/a[i][i];
```

are modeled by the parameterized $\mathbb{Z}$-polytope $P \cap \mathbb{Z}^3$, where

$$P = \{1 \le i \le n; i+1 \le j \le n; i+1 \le k \le n\}$$

and n is an integer parameter. $\square$

This model has shown his power as a model for loop analysis. Several optimization and parallelization methods are also obtained by transforming polyhedra (and thus their associated loop nest) by affine functions. But a major weakness of the model shows up when the transforming function is non-invertible: the result is not a $\mathbb{Z}$-polyhedron anymore. It is important to note that, throughout the paper, we will consider $\mathbb{Z}$-polyhedra whose lattice is $\mathbb{Z}^n$, as it is always possible to transform a $\mathbb{Z}$-polyhedron to this simpler form.

Example 2 The following example has been presented by Clauss [4]. It is a parametric version of an example initially presented by Ferrante *et al* [8] and also treated by Pugh [15].

```
for i = 1 to 8 do
  for j = 1 to p do
    a(6i+9j-7) = ...
```

The elements of the array a reached by this loop are defined by :

$$E = \{k \in \mathbb{Z} \mid \exists i, j, p \in \mathbb{Z}^3 : 1 \le i \le 8; 1 \le j \le p; k = 6i + 9j - 7\}$$

Although these points are defined by a set of affine rational constraints (*i.e.*, equalities and inequalities with integer coefficients), the set E of integer values of k, which is the image of a $\mathbb{Z}$-polyhedron by a non-invertible affine function, cannot be described by a $\mathbb{Z}$-polyhedron. E is the $\mathbb{Z}$-polyhedron $\{k = 3x + 2, 8 \le k \le 9p + 41, x \in \mathbb{Z}\}$ *minus* the two points $\{k = 11\}$ and $\{k = 9p + 38\}$. These two points that are not accessed by the loop nest will be called *data holes* from now on. $\square$

This kind of problem occurs particularly when looking for integer values or value ranges taken by some variables or parameters while having only rational constraints. Several methods have been proposed to deal with it, each of them being specialized into solving some particular problems. For instance, Pugh firstly focused on the existence of data dependences (by solving the integer linear programming problem) with omega [14]. Feautrier's PIP computes the integer lexicographic extrema of a parametric polyhedron. This result was firstly aimed around computing dependence functions and (re-)scheduling loop nests [7]. As solving this problem also implies solving the parametric integer linear programming problem, it also has other applications as for instance code generation [5, 2]. The most general method for counting the computation and data volumes of a loop nest has been proposed by Clauss [4], which uses Ehrhart polynomials [6]. But when data is accessed through non-invertible affine functions, the data holes are determined by integer roots of some polynomials. So the feasibility of this technique is limited by the possibility of finding these roots. Our global objective is to show that all these problems with integer points in rational polyhedra can be handled by using the periodic polyhedron model, which derives from the polytope model. It is shown in [13] that this model allows to compute the integer hull of a parametric polyhedron P, and also its extrema w.r.t. a set of linear objective functions (these function define an order among the integer points of P). Lexicographic extrema in P are just given by considering a particular order. This model is presented in section 2. Then, we show in section 3 how to determine the exact set of values accessed by a loop nest through a non-invertible reference function, and discuss its use for eliminating dead code. This result gives directly an alternative method, developed in section 4, for computing the number of integer points in the image of a $\mathbb{Z}$-polyhedron by a singular affine function. Some concluding remarks are made and future works are discussed in section 5.

2 Periodic Polyhedra

Let P be a polyhedron defined by a minimal set of equalities $Eq(I, N) = 0$ and inequalities $In(I, N) \geq 0$, over a set of variables $I \in \mathbb{Z}^n$ and parameters $N \in \mathbb{Z}^p$. Focusing on a given variable i_k, the inequalities define upper and lower rational bounds for i_k in function of the other variables and parameters. The bound derived from a given inequatility $In_x(I, N) \geq 0$ is the solution in i_k of the equation $In_x(I, N) = 0$.

Example 3 Consider the polyhedron

$$P_2 = \begin{cases} 3i + 2k - 1 \geq 0 \\ -3i - 2k + 8 \geq 0 \\ -2i - k \geq 0 \\ 2i + k + p - 1 \geq 0 \end{cases}$$

The inequality $2i + k + p - 1 \geq 0$ defines a rational lower bound for i: $i_{max} = (-k - p + 1)/2$. The values of k and p for which the value of i is bounded by this inequality is given by projecting i by using the latter equality, giving :

$$\begin{cases} k - 3p + 1 \geq 0 \\ -k + 3p + 13 \geq 0 \\ p - 1 \geq 0 \qquad \square \end{cases}$$

The values P' of the other variables and parameters for which i_k is actually bounded by $In_x(I, N) = 0$ is given by projecting P along i_k using $In_x(I, N) = 0$. The equation $In_x(I, N) = 0$ defines a hyperplane: we will call it *supporting hyperplane*. Its intersection with P' is known to be a *facet* of P. Thus, we can call P' a *projected facet*.

But we are interested in integer-valued variables. Similarly to rational bounds, it has been shown in [13] that integer bounds of a variable i_k can be derived from the inequalities defining the polyhedron. The integer bound derived from the inequality $In_x(I, N) \geq 0$ is the solution in i_k to the equation

$$In_x(I, N) - (In_x(I, N)) \; mod \; a_k = 0, \tag{1}$$

where a_k is the coefficient for i_k in $In_x(I, N)$. $(In_x(I, N)) \; mod \; a_k$ is a periodic function of I and N: the integer bound given by equation (1) has then a periodic definition.

Example 4 The inequality $3i + 2k - 1 \geq 0$ of P_2 defines an integer bound for i, given by

$$3i + 2k - 1 - (3i + 2k - 1) \; mod \; 3 = 0 = 3i + 2k - 1 - (2k - 1) \; mod \; 3,$$

which can also be written $i = (-2k + 1 + (2k-1)\ mod\ 3)/3$. $(2k-1)\ mod\ 3$ is a periodic function of k, with a period of 3:

- $(2k-1)\ mod\ 3 = 2$ if $k\ mod\ 3 = 0$
- $(2k-1)\ mod\ 3 = 1$ if $k\ mod\ 3 = 1$
- $(2k-1)\ mod\ 3 = 0$ if $k\ mod\ 3 = 2$

Then, the definition of the integer bound given by equation (1) is also periodic:

- $i = (-2k+3)/3$ if $k\ mod\ 3 = 0$
- $i = (-2k+2)/3$ if $k\ mod\ 3 = 1$
- $i = (-2k+1)/3$ if $k\ mod\ 3 = 2$

It can be written in a short form by using *periodic numbers*, which represent periodic rational-valued functions of integer variables by a multidimensional array: $i = (-2k + \begin{bmatrix}3 & 2 & 1\end{bmatrix}_k)/3$ Notice that here, as i is periodic only in function of one variable, the periodic number is one-dimensional. $\square$

The corresponding geometric object, defined by equation (1), is then a *periodic polyhedron*. Similarly to the rational case, the values P' of the other variables and parameters for which i_k is actually bounded by equation (1) is given by projecting P along i_k using equation (1). Equation (1) defines a periodic hyperplane, which will be called *supporting (pseudo-)hyperplane*. The periodic polyhedron P' is called *projected pseudo-facet*, as its intersection with the supporting pseudo-hyperplane is called a *pseudo-facet* w.r.t. i_k.

Example 5 The projected pseudo-facet corresponding to the pseudo-hyperplane $i = (-2k + \begin{bmatrix}3 & 2 & 1\end{bmatrix}_k)/3$ is given by projecting P_2 along i using this equation.

$$P_2' = \begin{cases} \begin{bmatrix}5 & 6 & 7\end{bmatrix}_k \geq 0 \\ k - \begin{bmatrix}6 & 4 & 2\end{bmatrix}_k \geq 0 \\ -k + 3p + \begin{bmatrix}1 & 0 & -1\end{bmatrix}_k \geq 0 \end{cases}$$

The corresponding pseudo-facet is then given by: $\begin{cases} 3i + 2k - \begin{bmatrix}3 & 2 & 1\end{bmatrix}_k = 0 \\ \begin{cases} \begin{bmatrix}5 & 6 & 7\end{bmatrix}_k \geq 0 \\ k - \begin{bmatrix}6 & 4 & 2\end{bmatrix}_k \geq 0 \\ -k + 3p + \begin{bmatrix}1 & 0 & -1\end{bmatrix}_k \geq 0 \end{cases} \end{cases}$ $\square$

More details on periodic polyhedra and some of their applications can be found in [11, 13]. Here, we just need to know that integer points in a rational polyhedron P are bounded by their pseudo-facets, which are lower-dimensional periodic polyhedra. The integer points of a pseudo-facet are also bounded by their own pseudo-facets, and so on, leading to 0-dimensional pseudo-facets, which are particular integer points. These integer points are *extremal* points of P, as all the integer points of P belong to the convex hull of these points. These points are called the *pseudo-vertices* of P.

We see in next section how to use these extremal points to determine the exact data accessed by a loop nest.

3 Domain of existence of an integer point in a polyhedron

In the polytope model, iterations are modeled by the integer points of a $\mathbb{Z}$-polytope $P(I, N)$: $\mathcal{P}.\begin{pmatrix} I \\ N \\ 1 \end{pmatrix} \geq 0 \cap \mathbb{Z}^{n+p+1}$, while data accessed through (multi-dimensional) access functions are the image of these points by the corresponding mapping. This mapping can be represented by an access matrix. It has been shown [4] that when this matrix is non-invertible, the image is not a $\mathbb{Z}$-polyhedron in the general case, but a union of $\mathbb{Z}$-polyhedra. Consider a $m \times (n + p)$ access matrix A, through which a loop with n indices and p parameters accesses a m-dimensional array. Let $X \in \mathbb{Z}^m$ be the coordinates of an accessed array element. The access of an element X by an iteration I of P is solution to:

$$\{\mathcal{P}.\begin{pmatrix} I \\ N \\ 1 \end{pmatrix} \geq 0; A.\begin{pmatrix} I \\ N \\ 1 \end{pmatrix} = X\}$$

Describing the accessed data reduces to look for the set of values of X for which the set of iterations I accessing X through A is not empty. In other words, we seek the domain of existence in $P(I, N)$ of an integer point in the inverse image (also called *preimage*) of X by A:

$$\{X \in \mathbb{Z}^m \mid preimage(X, A) \cap P(I, N) \neq \emptyset\}$$

First, there must exist integer points in $preimage(X, A)$. Then, we must see for which values of X these points are in $P(I, N)$.

3.1 Existence of an integer point in the preimage

For simplicity, assume there is no equality in the definition of P (we can always come to such a case). We look for integer values of X such that there exists an integer solution I to $A.\begin{pmatrix} I \\ N \\ 1 \end{pmatrix} = \begin{pmatrix} V & & R & C \\ 0 & \dots & 0 & 1 \end{pmatrix}.\begin{pmatrix} I \\ N \\ 1 \end{pmatrix} = X$, where V (respectively R and C) is the part of A corresponding to I (respectively N and the constants).

It is known [17] that there is an integer solution I to this equation for any integer value of X if and only if

$$H^{-1}.\begin{pmatrix} \mathbb{I}_m & -C \end{pmatrix}.\begin{pmatrix} X \\ 1 \end{pmatrix} \in \mathbb{Z}^m, \tag{2}$$

where $V = [H\ 0] \cdot U$ is the left Hermite normal form of V, and $\mathbb{I}_m$ is the m-dimensional identity matrix. H is an integer lower triangular matrix, so its inverse is rational in the general case. Let d_r be the common denominator of the elements of the r^{th} row of H^{-1}, and H'^{-1} the matrix H^{-1} whose r^{th} row has been multiplied by d_r. Equation (2) can then be written:

$$(H'^{-1}.X - H'^{-1}.C) \ mod\ d = 0, \tag{3}$$

where $d = \begin{pmatrix} d_r \end{pmatrix}, r \in [1..m]$, and mod is element-wise.

Example 6 The polyhedron presented in example 2 has only one equation: $\{6i + 9j - 7 = k\}$ There is an integer solution in (i, j) for any value of k if and only if $\frac{1}{3}.(k + 7) \in \mathbb{Z}$, that is to say $k + 7 \; mod \; 3 = 0$. $\square$

The general solution of this equation is found by considering that:

- $H'^{-1}.X - H'^{-1}.C = 0 \Longrightarrow (H'^{-1}.X - H'^{-1}.C) \; mod \; d = 0$. A first set of solutions is then obtained by solving a linear equation (i.e., it is the lattice of integer points given by the integer kernel of H'^{-1}).
- Each row of (3) is of the form: $f(X) \; mod \; d_r = 0$, where $f(X)$ is an affine function of X. $f(X) \; mod \; d_r$ is a periodic function of $X = (x_k)$ of period $S = (s_k), k \in [1..m]$. Hence, if X is a solution to $f(X) \; mod \; d_r$ = 0, then $X + Y.S$ is also a solution for any $Y \in \mathbb{Z}^m$. So the solutions reached by this way form a lattice of integer points, which is the intersection of the lattices given by the different $f(X) \; mod \; d_r$.

All the solutions can be reached by these two lattices, so the general solution to (3) is given by a particular integer solution to (3) and the combination of the two lattices:

$$\begin{pmatrix} X \\ 1 \end{pmatrix} = \begin{pmatrix} G & X_0 \\ 0 \ldots 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} X' \\ 1 \end{pmatrix}, X' \in \mathbb{Z}^m,$$

where $[G\,0]$ is the left Hermite normal form of a matrix whose column-vectors are the basis vectors of the two lattices, and X_0 is a particular solution to (3). Then, any integer point X' is reached by integer points I through A:

$$X' \in \mathbb{Z}^m \Longrightarrow I \in \mathbb{Z}^n$$

The interested reader will look for a more detailed and better illustrated presentation in [12].

Example 7 The solution to $k+7 = 0$ is unique: its integer kernel is the null vector. $k+7 \; mod \; 3$ is a periodic function of an integer variable: it is the periodic number $\begin{bmatrix} 1 & 2 & 0 \end{bmatrix}_k$, of period 3. $k = 2$ is a particular integer solution to $k + 7 \; mod \; 3 = 0$. The necessary and sufficient condition over k for the existence of an integer solution (i, j) to $6i + 9j - 7 = k$ is then $k = 3k' + 2, k' \in \mathbb{Z}$. We can make explicit this change of variables from k to k', giving: $6i + 9j - 7 = 3k' + 2 \Leftrightarrow 2i + 3j - k' - 3 = 0$ $\square$

The data accessed by the loop nest belong to the computed lattice, which is a first information. Changing the variables from X to X' corresponds to a compression of the data space: in the new data space, each array element X' can be accessed by an integer iteration point I through A. But this I does not necessarily belong to the iteration domain $\{P(I, N) \geq 0\}$. In next subsection, we look for the array elements X' actually accessed by the loop nest.

3.2 Exact data accessed by the loop nest

Considering the compressed data space, we want to compute the exact set of array elements X' that are accessed by the loop nest through the access function A' such that:

$$A \cdot \begin{pmatrix} I \\ N \\ 1 \end{pmatrix} = X = \begin{pmatrix} G & X_0 \\ 0 \ldots 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} X' \\ 1 \end{pmatrix} \Leftrightarrow A' \cdot \begin{pmatrix} I \\ N \\ 1 \end{pmatrix} = \begin{pmatrix} X' \\ 1 \end{pmatrix}$$

The problem defines a polyhedron with m equalities: $E(I, X', N) = \{P(I, N) \geq 0; A'(I, N) = X'\}$. Hence, considering X' and N as parameters, its geometric dimension is $n - m$ in the n-dimensional variable space.

In the following, we compute particular integer points of $E(I, X', N)$. When the definition of E contains equalities, the values of the points of E are dependent on each other, which is not suitable for our algorithm. We must then eliminate these equations, transforming E from a non-fully-dimensional polyhedron in a n-dimensional space $\mathcal{S}$ into a fully-dimensional polyhedron in an $(n - m)$-dimensional space $\mathcal{S}'$. We can eliminate m equations by eliminating m variables I_2 from E. But we want that an integer point in $\mathcal{S}'$ corresponds to an integer point in $\mathcal{S}$, that is to say: $I_1 \in \mathbb{Z}^{n-m} \Longrightarrow I_2 \in \mathbb{Z}^m$, where I_1 are the remaining variables in $\mathcal{S}'$. This problem is exactly the same as the one treated in the latter subsection, where we wanted that $X \in \mathbb{Z}^m \Longrightarrow I \in \mathbb{Z}^n$. A compression over I_1 can then be computed in the same way: $\begin{pmatrix} I_1 \\ 1 \end{pmatrix} = \begin{pmatrix} G_1 & I_{1,0} \\ 0 \dots 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} I'_1 \\ 1 \end{pmatrix}$

Example 8 The polyhedron $\{1 \leq i \leq 8; 1 \leq j \leq p; 2i + 3j - k' - 3 = 0\}$ has two variables i and j and one equality. Its geometric dimension is then 1. We want to eliminate the equation by eliminating the variable j and obtain a fully-dimensional polyhedron in a space of dimension 1. Thus, j has to be integer for each integer value of the variables of the transformed polyhedron. It is the case when $i = 3i' + 2k'$. This defines a compression of i, which gives the transformed polyhedron:

$$\begin{cases} 1 - 2k' \leq 3i' \leq 8 - 2k' \\ 1 \leq j \leq p \\ 2i' + j + k' - 1 = 0 \end{cases}$$

We can then eliminate the equality, as well as j, giving:

$$P_3 = \begin{cases} 1 - 2k' \leq 3i' \leq 8 - 2k' \\ -k' - p + 1 \leq 2i' \leq -k' \end{cases}$$

Notice that the compressions of X and of I_1 can be combined into a single compression. We look for the values of X' such that there is an integer solution I_1 of $E(I_1, X', N)$. Obviously, there exists an integer point I_1 in a parametric polyhedron $E(I_1, X', N)$ if and only if the (parametric) convex hull of its integer points, i.e., its integer hull, is not empty. It is shown in [13] that the (parametric) integer hull of a polyhedron is the convex hull of some extremal integer points called *pseudo-vertices*, presented in section 2. The integer hull is non-empty if and only if there exists such extremal points. So a straightforward way to obtain the values of X' for which there exists an integer point I_1 in $E(I_1, X', N)$ would be to compute the values of X' for which $E(I_1, X', N)$ has pseudo-vertices. For a given pseudo-vertex, these values are given by the projection of the pseudo-vertex into the parameters space (X', N). By construction, as pseudo-facets are expressed as the intersection of a supporting pseudo-hyperplane and a projected pseudo-facet, a pseudo-vertex is then expressed as the intersection of n equalities, that describe its coordinates in the variable space, and some inequalities, that are its projection into the parameters space. We call this projection the *validity domain* of the pseudo-vertex, as it is very similar to the validity domains of rational vertices of a parametric polyhedron, introduced by Loechner in [10].

Example 9 The pseudo-facet w.r.t. i' of P_3 derived from the inequality $2i' + k' + p - 1 \geq 0$ is:

$$P_{3,1} = \begin{cases} 2i' + k' + p - 1 - (k' + p - 1) \; mod \; 2 = 0 \\ \begin{cases} k' - 3p + 1 + 3(k' + p - 1) \; mod \; 2 \geq 0 \\ -k' + 3p + 13 - 3(k' + p - 1) \; mod \; 2 \geq 0 \\ p - 1 - (k' + p - 1) \; mod \; 2 \geq 0 \end{cases} \end{cases} = \begin{cases} 2i' + k' + p - \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}_{p,k'} \; mod \; 2 = 0 \\ \begin{cases} k' - 3p + \begin{bmatrix} 4 & 3 \\ 3 & 4 \end{bmatrix}_{p,k'} \geq 0 \\ -k' + 3p + \begin{bmatrix} 10 & 13 \\ 13 & 10 \end{bmatrix}_{p,k'} \geq 0 \\ p - \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}_{p,k'} \geq 0 \end{cases} \end{cases}$$

As the value of all its variables are determined by equalities, it is a pseudo-vertex. The values of k' and p for which this pseudo-vertex belongs to P_3 (i.e., its validity domain) are given by the inequalities. $\square$

As there exists an integer point in E as soon as there exists a pseudo-vertex, the values of X and N such that there exists an integer point I_1 in E can be given by the union of the validity domains of the pseudo-vertices of E.

Computing all the pseudo-vertices is not necessary. The existence of an integer point in E do not reduce to the existence of *all* its pseudo-vertices, but to the existence of *at least one* pseudo-vertex. Given a linear order $\mathcal{O}$ on the variables I_1, there is always at least one integer point that is the minimum (we could also take the maximum) w.r.t. this order. Thus, the existence of an integer point in E reduces to the existence of a pseudo-vertex that is minimal w.r.t. $\mathcal{O}$.

Example 10 for P_3, we can for instance minimize i. Two pseudo-vertices minimize i': $P_{3,1}$ and

$$P_{3,2} = \begin{cases} 3i' + 2k' - 1 - (2k' - 1) \; mod \; 3 = 0 \\ \begin{cases} 7 - (2k' - 1) \; mod \; 3 \geq 0 \\ k' - 2 - 2((2k' - 1) \; mod \; 3) \geq 0 \\ -k' + 3p - 1 + 2(2k' - 1) \; mod \; 3 \geq 0 \end{cases} \end{cases} = \begin{cases} 3i + 2k - \begin{bmatrix} 3 & 2 & 1 \end{bmatrix}_{k'} = 0 \\ \begin{cases} \begin{bmatrix} 5 & 6 & 7 \end{bmatrix}_{k'} \geq 0 \\ k - \begin{bmatrix} 6 & 4 & 2 \end{bmatrix}_{k'} \geq 0 \\ -k + 3p + \begin{bmatrix} 3 & 1 & -1 \end{bmatrix}_{k'} k \geq 0 \end{cases} \end{cases}$$

The values of k and p such that there is an integer point in P_3 are then given by the union of the validity domains of $P_{3,1}$ and $P_{3,2}$:

$$P'_{3,1} = \begin{cases} k' - 3p + \begin{bmatrix} 4 & 3 \\ 3 & 4 \end{bmatrix}_{p,k'} \geq 0 \\ -k' + 3p + \begin{bmatrix} 10 & 13 \\ 13 & 10 \end{bmatrix}_{p,k'} \geq 0 \\ p - \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}_{p,k'} \geq 0 \end{cases} \cup P'_{3,2} = \begin{cases} k' - \begin{bmatrix} 6 & 4 & 2 \end{bmatrix}_{k'} \geq 0 \\ -k' + 3p + \begin{bmatrix} 3 & 1 & -1 \end{bmatrix}_{k'} \geq 0 \end{cases}$$

The fact that the union of $P'_{3,1}$ and $P'_{3,2}$ describe the exact set of integer values of k' such that there exists an integer solution in P_3 is not obvious. To convince the reader, let us show how, for instance, we can compute the data holes introduced in section 1. The periodic numbers used in the definition of these validity domains are a short notation. These periodic polyhedra can be written more explicitly as:

$$P'_{3,1} = \begin{cases} \{3p-4 \le k' \le 3p+10; p \ge 2\} \; if \; k' \; mod \; 2 = 0 \; and \; p \; mod \; 2 = 0 \\ \{3p-3 \le k' \le 3p+13; p \ge 1\} \; if \; k' \; mod \; 2 = 1 \; and \; p \; mod \; 2 = 0 \\ \{3p-3 \le k' \le 3p+13; p \ge 1\} \; if \; k' \; mod \; 2 = 0 \; and \; p \; mod \; 2 = 1 \\ \{3p-4 \le k' \le 3p+10; p \ge 2\} \; if \; k' \; mod \; 2 = 1 \; and \; p \; mod \; 2 = 1 \end{cases}$$

$$P'_{3,2} = \begin{cases} \{6 \le k' \le 3p+3\} \; if \; k' \; mod \; 3 = 0 \\ \{4 \le k' \le 3p+1\} \; if \; k' \; mod \; 3 = 1 \\ \{2 \le k' \le 3p-1\} \; if \; k' \; mod \; 3 = 2 \end{cases}$$

Each element of this decomposition is the intersection of a polyhedron and an integer lattice (defined, for instance, by $k' \; mod \; 2 = 0$ and $p \; mod \; 2 = 0$), *i.e.*, a $\mathbb{Z}$-polyhedron. So the data holes can be found in the following way: let $P_{k'}$ be the projection of P_3 on the space of k': $P_{k'} = \{2 \le k' \le 3p+13\}$. The data holes can be determined for each validity domain by scanning the different lattices defined by the periodic numbers. For a given lattice L, the holes generated by the $\mathbb{Z}$-polyhedron P'_L whose lattice is L is given by $(P_{k'} \cap L) \setminus P'_L = (P_{k'} \setminus R) \cap L$. Here, the holes generated by $\{3p-4 \le k' \le 3p+10; p \ge 2, k' \; mod \; 2 = 0, p \; mod \; 2 = 0\}$ are: $P_{k'} \setminus \{3p-4 \le k' \le 3p+10; p \ge 2\} \cap \{k' \; mod \; 2 = 0, p \; mod \; 2 = 0\} = \{3p+11 \le k' \le 3p+13\} \cap \{k' \; mod \; 2 = 0, p \; mod \; 2 = 0\} = \{k' = 3p+12\}$, which is one of the data holes we were looking for ($k = 3k'+2 = 9p+38$). $\square$

Identifying the data holes determine the data computed by a loop nest more precisely that just by projecting along the access function the polyhedron defined by the loop bounds. In the example developed here, if a program analysis tells that the only array elements that are used after the loop nest execution are the elements $a[11]$ and $a[3p+12]$, the whole loop nest is useless and can be eliminated. This analysis can be considered costly when considering a single loop nest whose volume of accessed data is large. In this case, the ratio between the number of data accessed and the number of holes shall tend to zero, as well as the chances for having to eliminate a whole loop nest. But dead code elimination based on the analysis of liveness of array elements have to propagate along the whole analyzed program. The error associated to an approximate analysis propagates as well, and the final error may disallow a powerful dead code elimination.

Identifying the exact set of integer points in the projection of a $\mathbb{Z}$-polytope also allows to enumerate this set. Among many applications (the interested reader can refer to [3] where several applications are shown), this allows for instance to determine the exact number of data accessed by a loop nest through a non-invertible array access function. Next section shows how the number of integer points in such a projection can be derived from the domain of existence of an integer point in a polyhedron.

4 Number of integer points in the projection of a $\mathbb{Z}$-polyhedron

Counting the number of integer points in the image of a $\mathbb{Z}$-polyhedron P by a non-singular integer affine transformation T is trivial, as it equals the number of integer points in P. We have seen that when the transformation is singular, i.e., it can be represented by a non-invertible matrix A, some *holes* appear in the image. After some adequate transformations, we have obtained the definition of a set of points X' such that there exists an integer point I in the projection of a polyhedron $P(I,N)$, depending on parameters N. These integer points X' are the image of the integer points of P. So counting the number of existing X' gives the number of points in the projection of the $\mathbb{Z}$-polyhedron P along A. Ehrhart polynomials [3, 6] give the number of integer points in a rational polyhedron. Thus, the operation is almost straightforward, the only difficulty coming from the fact that the set of points X' are defined by a periodic polyhedron.

Example 11 The set of integer points in E, that is to say in P_3, is defined in section 3 by $P'_{3,1} \cup P'_{3,2}$. $P'_{3,1}$ is periodic in function of k', with a period of 2, and $P'_{3,2}$ is also periodic along k' but with a period of 3. The couple $(P'_{3,1}, P'_{3,2})$ is then also periodic along k' with a period of 6. Similarly, it is periodic along p with a period of 2. We can then write $P'_{3,1} \cup P'_{3,2}$ as:

$$\begin{cases} 3p - \begin{bmatrix} 4 & 3 & 4 & 3 & 4 & 3 \\ 3 & 4 & 3 & 4 & 3 & 4 \end{bmatrix}_{p,k'} \leq k' \leq 3p + \begin{bmatrix} 10 & 13 & 10 & 13 & 10 & 13 \\ 13 & 10 & 13 & 10 & 13 & 10 \end{bmatrix}_{p,k'} \\ p \geq \begin{bmatrix} 2 & 1 & 2 & 1 & 2 & 1 \\ 1 & 2 & 1 & 2 & 1 & 2 \end{bmatrix}_{p,k'} \end{cases}$$

$$\cup \left\{ \begin{bmatrix} 6 & 4 & 2 & 6 & 4 & 2 \\ 6 & 4 & 2 & 6 & 4 & 2 \end{bmatrix}_{p,k'} \leq k' \leq 3p + \begin{bmatrix} 3 & 1 & -1 & 3 & 1 & -1 \\ 3 & 1 & -1 & 3 & 1 & -1 \end{bmatrix}_{p,k'} \right.$$

This system defines twelve $\mathbb{Z}$-polyhedra, each of them having its own Ehrhart polynomial. We can transform the system by considering that k' can always be written $k' = 6k'' + k' \; mod \; 6 = k'' + \begin{bmatrix} 0 & 1 & 2 & 3 & 4 & 5 \end{bmatrix}_{k'}$, and similarly $p = 2p' + p \; mod \; 2 = 2p' + \begin{bmatrix} 0 & 1 \end{bmatrix}_p$:

$$\begin{cases} 6p' - \begin{bmatrix} 4 & 2 & 2 & 0 & 0 & -2 \\ 0 & 0 & -2 & -2 & -4 & -4 \end{bmatrix}_{p,k'} \leq 6k'' \leq 6p' + \begin{bmatrix} 10 & 12 & 8 & 10 & 6 & 8 \\ 16 & 12 & 14 & 10 & 12 & 8 \end{bmatrix}_{p,k'} \\ 2p' \geq \begin{bmatrix} 2 & 1 & 2 & 1 & 2 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}_{p,k'} \end{cases}$$

$$\cup \left\{ \begin{bmatrix} 6 & 3 & 0 & 3 & 0 & -3 \end{bmatrix}_{p,k'} \leq 6k'' \leq 6p' + \begin{bmatrix} 3 & 0 & -3 & 0 & -3 & -6 \\ 6 & 3 & 0 & 3 & 0 & -3 \end{bmatrix}_{p,k'} \right.$$

In this form, the variables over which the polyhedron is enumerated, namely k'' and p', are independent from $k' \; mod \; 6$ and $p \; mod \; 2$. Then, the number of distinct values of k' in $P'_{3,1} \cup P'_{3,2}$ is the sum of the six Ehrhart polynomials given by the distinct combinations of $k' \; mod \; 6$, for each value of $p \; mod \; 2$. Let us see what is the Ehrhart Polynomial $\epsilon(p)$ for $p \; mod \; 2 = 0$:

$$\begin{cases} \{6p' - 4 \leq 6k'' \leq 6p' + 10; 2p' \geq 2\} \cup \{6 \leq 6k'' \leq 6p' + 3\} & , k' \; mod \; 6 = 0 \\ \{6p' - 2 \leq 6k'' \leq 6p' + 12; 2p' \geq 1\} \cup \{3 \leq 6k'' \leq 6p'\} & , k' \; mod \; 6 = 1 \\ \{6p' - 2 \leq 6k'' \leq 6p' + 8; 2p' \geq 2\} \cup \{0 \leq 6k'' \leq 6p' - 3\} & , k' \; mod \; 6 = 2 \\ \{6p' \leq 6k'' \leq 6p' + 10; 2p' \geq 1\} \cup \{3 \leq 6k'' \leq 6p'\} & , k' \; mod \; 6 = 3 \\ \{6p' \leq 6k'' \leq 6p' + 6; 2p' \geq 2\} \cup \{0 \leq 6k'' \leq 6p' - 3\} & , k' \; mod \; 6 = 4 \\ \{6p' + 2 \leq 6k'' \leq 6p' + 8; 2p' \geq 1\} \cup \{-3 \leq 6k'' \leq 6p' - 6\} & , k' \; mod \; 6 = 5 \end{cases}$$

$$
= \begin{cases} \{2 \le 6k'' \le 6p'+10; p'=1\} \cup \{6 \le 6k'' \le 6p'+10; p' \ge 2\} & , k' \; mod \; 6 = 0 \\ \{3 \le 6k'' \le 6p'+12; 2p' \ge 1\} & , k' \; mod \; 6 = 1 \\ \{0 \le 6k'' \le 6p'+8; p' \ge 1\} & , k' \; mod \; 6 = 2 \\ \{3 \le 6k'' \le 6p'+10; 2p' \ge 1\} & , k' \; mod \; 6 = 3 \\ \{0 \le 6k'' \le 6p'+6; p' \ge 1\} & , k' \; mod \; 6 = 4 \\ \{-3 \le 6k'' \le 6p'-6; 2p' \ge 1\} \cup \{6p'+2 \le 6k'' \le 6p'+8; 2p' \ge 1\} & , k' \; mod \; 6 = 5 \end{cases}
$$

Their number of integer points are then:

$$
\begin{cases} 2 \; if \; p'=1, p'+2 \; if \; p' \ge 2 & , k' \; mod \; 6 = 0; \quad p'+2 \; if \; p' \ge 1 \quad , k' \; mod \; 6 = 1 \\ p'+2 \; if \; p' \ge 1 & , k' \; mod \; 6 = 2; \quad p'+1 \; if \; p' \ge 1 \quad , k' \; mod \; 6 = 3 \\ p'+2 \; if \; p' \ge 1 & , k' \; mod \; 6 = 4; \quad p'+1 \; if \; p' \ge 1 \quad , k' \; mod \; 6 = 5 \end{cases}
$$

Summing all these possible values over $k' \; mod \; 6$ gives the number of integer points in $P'_{3,1} \cup P'_{3,2}$ when $p \; mod \; 2 = 0$: $\epsilon(p) = \begin{cases} 16 \; if \; p' = 1 \\ 6p'+10 \; if \; p' \ge 2 \end{cases} = \begin{cases} 16 \; if \; p = 2 \\ 3p+10 \; if \; p \ge 4 \end{cases}$

5 Conclusion

We have shown a new use of periodic polyhedra theory, as a way to define the exact set of integer points in the image of a parametric $\mathbb{Z}$-polyhedron by a singular function. This allows to determine the exact data accessed by a loop nest through a singular array reference, which occurs when the dimension of the array is lower than the number of loop indices. A method for computing the number of such points is derived, allowing to compute relevant data such as the exact volume of data accessed or communicated by a loop nest without having to solve polynomial roots as in [4]. The periodic polyhedron model is an improvement of the polytope model, meaning that it allows more precision. The methods for solving problems with the help of the periodic polyhedron model are being implemented using the polyhedral library *Polylib*. We intend to go further in showing the relevance of this model for loop nest analysis.

References

[1] C. Ancourt and F. Irigoin. Scanning polyhedra with DO loops. In *Proc. ACM SIGPLAN '91*, pages 39–50, june 1991.

[2] P. Boulet and P. Feautrier. Scanning polyhedra without Do-loops. In *Int. Conf of Parallel Architectures and Compilation Techniques PACT'98*. IEEE, October 1998.

[3] Ph. Clauss. Counting solutions to linear and nonlinear constraints through Ehrhart polynomials: Applications to analyze and transform scientific programs. In *10th ACM Int. Conf. on Supercomputing, Philadelphia*, 1996.

[4] Ph. Clauss. Handling memory cache policy with integer points countings. In Lengauer, Griebl, and Gorlatch, editors, *Euro-Par'97, Passau*, pages 285–293. Springer-Verlag, LNCS 1300, August 1997.

[5] J.-F. Collard, P. Feautrier, and T. Risset. Construction of DO loops from systems of affine constraints. *Parallel Processing Letters*, 5(3):421–436, 1995.

[6] E. Ehrhart. *Polynômes arithmétiques et Méthode des Polyèdres en Combinatoire*, volume 35 of *International Series of Numerical Mathematics*. Birkhäuser Verlag, Basel/Stuttgart, 1977.

[7] P. Feautrier. Dataflow analysis of scalar and array references. *Int. J. of Parallel Programming*, 20(1):23–53, February 1991.

[8] J. Ferrante, J. Sarkar, and W. Thrash. On estimating and enhancing cache effectiveness. In *Advances in Languages and Compilers for Parallel Processing*, pages 328–343. MIT Press, 1991.

[9] C. Lengauer. Loop parallelization in the polytope model. In *CONCUR 93*, 1993.

[10] V. Loechner. *Contribution à l'étude des polyèdres paramétrés et applications en parallélisation automatique*. PhD thesis, Université Louis Pasteur, 1997. http://icps.u-strasbg.fr/pub-97/.

[11] B. Meister. Using periodics in integer polyhedral problems. research report, LSIIT - ICPS, 2003. http://icps.u-strasbg.fr/∼meister.

[12] B. Meister. Parametric integer programming with periodic polyhedra. research report, LSIIT - ICPS, 2004.

[13] B. Meister. Periodic polyhedra. In Evelyn Duesterwald, editor, *Compiler Construction, 13th International Conference CC 2004, Part of ETAPS 2004*, pages 134–149. Springer, April 2004.

[14] W. Pugh. The omega test: a fast and practical integer programming algorithm for dependence analysis. *Comm. of the ACM*, 8:102–114, August 1992.

[15] William Pugh. Counting solutions to presburger formulas: how and why. In *Proceedings of the ACM SIGPLAN 1994 conference on Programming language design and implementation*, pages 121–134. ACM Press, 1994.

[16] P. Quinton, S. V. Rajopadhye, and T. Risset. On manipulating z-polyhedra using a canonical representation. *Parallel Processing Letter*, 7(2):181–194, 1997.

[17] A. Schrijver. *Theroy of Linear and Integer Programming*. John Wiley and Sons, 1986.

[18] M.E. Wolf and M.S. Lam. A loop transformation theory and an algorithm to maximize parallelism. *IEEE Transactions on Parallel and Distributed Systems*, 2(4):452–471, October 1991.

[19] J. Xue. Unimodular transformations of non-perfectly nested loops. *Parallel Computing*, 22(12):1621–1645, 1997.