Symbolic Bernstein expansion over a convex polytope

Philippe Clauss

Abstract

The objective is to find accurate symbolic bounds of a parametrized multi-variate polynomial whose variables are defined over a parametrized convex polytope.

1 Introduction

It is well-known that any convex polytope can be defined as the set of all convex combinations of its vertices: let $v_1, v_2, ..., v_{nv}$ be the vertices of P. Then any point x in P can be defined as a linear combination of the vertices, called a *convex combination*, $x = a_1 v_1 + a_2 v_2 + ... + a_{nv} v_{nv}$ where $\forall\ a_i, i = 1..nv, 0 \leq a_i \leq 1$ and $\sum_{i=1}^{nv} a_i = 1$.

While considering a parametrized polytope, the same definition of its point can be used. However, the polytope has first to be decomposed into polytopes of constant shapes defined over adjacent and convex domains of the parameters. For such parameter domain, the parametrized vertices coordinates and the number of vertices are constant. Those parameter domains and parametrized coordinates can be computed using the polyhedral library *PolyLib*.

2 Bernstein expansion

The Bernstein expansion applies for variables defined over a "box", that is to say, a cartesian product of intervals. However when considering parametrized intervals, the mapping of these intervals $[\alpha, \beta]$ onto the unit box $[0, 1]$ is only valid if $\beta - \alpha \neq 0$. Hence parameter values that are roots of $\beta - \alpha$ have to be excluded.

It is also possible to handle intervals whose bounds depend on some variables, each interval being defined from variables defined from previous intervals, in the same way as loop bounds defined from outer loops indices. It is also necessary in this case to exclude values that are roots of the equations $\beta - \alpha$.

The polynomial has finally to be considered on a union of intervals excluding those roots and then to be evaluated for them. Unfortunately it is not always possible to find those roots since $\beta - \alpha$ can be a high degree multi-variate polynomial.

The polynomial variables are defined over boxes. It can then be difficult to consider more general sets of variables. For example, considering variables over convex domains could be useful for loop analysis. It is possible to transform a polytope into a cartesian product of boxes. However it would need to first compute a Fourier-Motzkin decomposition of the initial polytope. Moreover some roots of equations $\beta - \alpha$ have still to be excluded.

The Bernstein polynomials constitute a basis for polynomials of a given degree and number of variables. They can be seen in a geometrical point of view as the coefficients $a_1, a_2, ..., a_{nv}$ of a convex combination, since their values are between 0 and 1 and their sum equals 1. Hence the Bernstein coefficients can be seen as the vertices of all the possible values of the polynomial. However all the convex combinations of these vertices do not correspond to values effectively taken by the polynomial. That is why the specific form of the Bernstein polynomials introduces a constraint on their possible values. Anyway since the Bernstein coefficients can be seen as vertices, they naturally provide upper and lower bounds of the polynomial values.

3 A Bernstein basis from convex combination coefficients

Consider a polynomial whose variables are defined over a convex polytope P. The vector of the variables x can then be defined as a convex combination of the vertices of P, $v_1, v_2, ..., v_{nv}$: $x = a_1 v_1 + a_2 v_2 + ... + a_{nv} v_{nv}$ where $\forall\ a_i, i = 1..nv, 0 \leq a_i \leq 1$ and $\sum_{i=1}^{nv} a_i = 1$.

Then x can be replaced into the polynomial with its expanded form $a_1 v_1 + a_2 v_2 + ... + a_{nv} v_{nv}$. A new polynomial is obtained, whose variables are $a_1, a_2, ..., a_{nv}$ defined over the unit box $[0, 1]^{nv}$. Some upper and lower bounds of this polynomial could be obtained by computing its Bernstein coefficients. However these bounds would not be very accurate, since the property saying that $\sum_{i=1}^{nv} a_i = 1$ and which links the variables is not taken into account. For example, the case where all variables are equal to 1 would be abusively considered.

By looking further to the Bernstein basis, it is observed that the Bernstein polynomials are all the monomials obtained from the expansion of $(x + (1 - x))^n$, where n is the polynomial degree. These monomials are such that:

1. Their sum equals 1 ;
2. Each monomials varies between 0 and 1 ;
3. All monomials constitute a basis for the considered polynomials for a given degree and number of variables.

Hence any set of polynomials with those 3 properties constitutes a basis whose coefficients in the basis provide lower and upper bounds of the polynomial values.

Consider again a polynomial whose variables are convex combination coefficients. For such a polynomial, a first idea would be to build a basis from the expansion of $(a_1 + a_2 + ... + a_v)^n$. However the obtained monomials do not seem constituting a basis. For example constant terms cannot be defined in this basis.

The property saying that $\sum_{i=1}^{nv} a_i = 1$ allows to rewrite the polynomial as another polynomial whose monomials are all of degree n. Such a polynomial can then be generated from the basis built with $(a_1 + a_2 + ... + a_v)^n$. The polynomial transformation is done in the following way: any monomial M of degree $i < n$ can be rewritten as a sum of monomials of degree n by expanding $M = M\,(a_1 + a_2 + ... + a_v)^{n-i}$. Since $a_1 + a_2 + ... + a_v$ is equal to 1, the equality holds.

4 Example

Consider the polynomial $\frac{1}{2}i^2 + \frac{1}{2}i + j$ over the box $[0, N] \times [0, i]$. This box defines a convex polytope whose vertices are: $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$, $\begin{pmatrix} N \\ 0 \end{pmatrix}$ and $\begin{pmatrix} N \\ N \end{pmatrix}$. Hence any point $\begin{pmatrix} i \\ j \end{pmatrix}$ is a convex combination of the vertices: $\begin{pmatrix} i \\ j \end{pmatrix} = a_1 \begin{pmatrix} 0 \\ 0 \end{pmatrix} + a_2 \begin{pmatrix} N \\ 0 \end{pmatrix} + a_3 \begin{pmatrix} N \\ N \end{pmatrix}, 0 \leq a_i \leq 1, \sum_{i=1}^{3} a_i = 1$.

By replacing $\begin{pmatrix} i \\ j \end{pmatrix}$ with its convex combination, a new polynomial is obtained whose variables are a_1, a_2, a_3:

$$\frac{1}{2}N^2 a_2^2 + N^2 a_2 a_3 + \frac{1}{2}N^2 a_3^2 + \frac{1}{2}N a_2 + \frac{3}{2}N a_3$$

Monomials whose degrees are less than 2 are transformed into sums of monomials of degree 2:

- $\frac{1}{2}Na_2 = \frac{1}{2}Na_2(a_1 + a_2 + a_3)$
- $\frac{3}{2}Na_3 = \frac{3}{2}Na_3(a_1 + a_2 + a_3)$

The final polynomial is:

$$(\frac{1}{2}N^2 + \frac{1}{2}N)a_2^2 + (\frac{1}{2}N^2 + \frac{3}{2}N)a_3^2 + \frac{1}{2}Na_1a_2 + \frac{3}{2}Na_1a_3 + (N^2 + 2N)a_2a_3$$

The basis is built from the expansion of $(a_1 + a_2 + a_3)^2$ providing the following monomials:

$$\begin{aligned} P1 =& \quad a_1^2 \\ P2 =& \quad a_2^2 \\ P3 =& \quad a_3^2 \\ P4 =& \quad 2a_1a_2 \\ P5 =& \quad 2a_1a_3 \\ P6 =& \quad 2a_2a_3 \end{aligned}$$

The coefficients of these polynomials are used in the matrix T allowing to change the basis from:

$$\{a_1^2, a_2^2, a_3^2, a_1a_2, a_1a_3, a_2a_3\}$$

to the basis:

$$\{P1, P2, P3, P4, P5, P6\}$$

Let B be the column vector of the monomials defining the new basis and let C be the row vector of the coefficients into the polynomial. The polynomial in the new basis is obtained by computing $(C \cdot T^{-1} \cdot B)$:

$$0P1 + (\frac{1}{2}N^2 + \frac{1}{2}N)P2 + (\frac{1}{2}N^2 + \frac{3}{2}N)P3 + \frac{1}{4}NP4 + \frac{3}{4}NP5 + (\frac{1}{2}N^2 + N)P6$$

It can then be concluded that the polynomial varies between 0 and $\frac{1}{2}N^2 + \frac{3}{2}N$. It can be observed that this bounds are the exact bounds.

5 About the Implementation

A program computing the Bernstein coefficients of a multi-variate polynomial defined over a convex polytope has to consider 2 kinds of information:

- the definition of the parametrized convex polytope on which the variables are defined ;
- the definition of the multi-variate polynomial.

The input definition of the parametrized polytope has to be of the same format as for several PolyLib functions as the one computing parametric vertices or the one computing Ehrhart polynomials. It consists in 2 matrix, the first one defining the constraints on the variables and the second one defining the constraints uniquely on the parameters. Note that these last kind of constraints can also be entered in the first matrix (and so no more in the second one). Each matrix has to be preceded by 2 integer values giving the number of lines and columns of the following matrix. The first column of each matrix contains either a 0 or a 1. A 0 means that the constraint in the row is an equality. A 1 means that the constraint in the row is an inequality of the form *linear_equation* ≥ 0. The next colums concern the variables, then the parameters and finally the constant values in the constraints.

The definition of the multi-variate polynomial can have the following form. A matrix can define all the monomials in the following way. Each line is associated to a monomial. The first columns concern variables where each of their degree is entered. The last 2 colums concern the coefficient in the monomial. Since this coefficient can be rational, the first column is used to enter the numerator and the last one the denominator. Values in this last column must be greater or equal to 1. Finally, the matrix has to be preceded by 2 values providing the number of rows and colums of the matrix. Notice that the number of columns must be the number of variables plus 2, and have to be coherent with the number of variables in the parametrized polytope definition.

For example, the input file for the above example could be the following:

```
# 1) Definition of the polytope

4 5

#  i  j  N  cst
1  1  0  0  0
1 -1  0  1  0
1  0  1  0  0
1  1 -1  0  0

0 3

# 2) Definition of the polynomial

3 4

# i  j  num  den
  2  0  1    2
  1  0  1    2
  0  1  1    1
```

From such an input, the program first calls the parametric vertices finding function with the first part of the input, that is to say, the definition of the polytope. The function returns adjacent parameters domains each being associated to parametric vertices coordinates.

The next step of the program is to consider each parameter domain one by one. For each parameter domain, the corresponding vertices are used to transform the initial polynomial into a new one whose variables are the coefficients in the convex combination of the vertices. Then this polynomial is transformed again in order to be a sum of monomials of maximal degree n. Finally, this last polynomial is expressed into the basis built from the expansion of $(a_1 + a_2 + ... + a_v)^n$. Notice that matrix T used to change the basis can be built directly from the knowledge of the number of vertices and the maximal degree n. Notice also that n is the maximum of the sums of the degrees in each row defining a monomial in the initial matrix.

The next step is to analyze the coefficients in this last basis, in order to get lower and upper bounds. If only one parameter is considered, then the minimum and maximum coefficients are easy to determine. The maximum coefficient is the one which has the greatest coefficients from its highest to its lowest degree monomial. It is the one whose monomial coefficients, from the highest to the lowest monomial degree, are lexicographically greater than monomial coefficients of any other coefficient. For example, $\frac{1}{2}N^2 + \frac{3}{2}N$ has the monomial coefficients $(\frac{1}{2}, \frac{3}{2})$ and $\frac{1}{2}N^2 + N$ has the monomial coefficients $(\frac{1}{2}, 1)$. $\frac{1}{2}N^2 + \frac{3}{2}N$ is the maximum since $(\frac{1}{2}, \frac{3}{2}) \succ (\frac{1}{2}, 1)$.

In the case of several parameters, several cases have to be considered since a given coefficient can be the maximum, subject to some parameters values. For example, consider two parameters N and M and consider that we have two coefficients exactly equal to N and M. Each of them can be the maximum depending on the condition $N > M$. However since some parameter constraints could have been entered in the initial input, and since we are considering a specific parameter domain, it is possible to determine that $N > M$ never holds over this domain.

Two cases have to be considered whether the bernstein coefficients are linear or non-linear (polynomial) functions over the parameters.

If the bernstein coefficients are linear functions then a solution can be the following. Let nc be the number of coefficients. A coefficient c_i is the maximum if it is greater than any other coefficient: $\forall j \neq i, c_i > c_j$. All these inequations define linear constraints that have to be combined with the parameter constraints. If the so defined domain is empty, then c_i can never be the maximum. Otherwise, more concise and non redundant constraints for which c_i is maximum can be obtained using the PolyLib. Hence nc cases have to be tested whether any c_i can be the maximum or not. Finally, upper bounds are obtained on several adjacent parameter domains. Getting the lower bounds is similar.

If the bernstein coefficients are not linear, that is to say, if at least one coefficient is not linear, a system of polynomial inequations has to be solved over the definition domain of the parameters (if it is known). An approach could be to apply again bernstein expansion to the bernstein coefficients, in order to get their respective minimum and maximum values allowing to sort them and find the greatest between them. Notice that these last bernstein coefficients (the bernstein coefficients of the bernstein coefficients) would be constants, and no more functions over parameters, if the parameters are defined over non parametrized polytopes. However, some coefficients can be simultaneously maximum depending on the values of the parameters. If the coefficients are linear functions, then the domains on which each coefficient is maximum can be determined in the same way as in the previous case of linear bernstein coefficients. The general approach could consist in a recursive application of the bernstein expansion until constant bounds or bounds over the highest level parameters on different domains are obtained.