Enclosure of range of multivariate polynomial

Tomasz Strek

A fundamental problem in range methods is computing the ranges of values of real functions or polynomials. Interval methods provide rigorous enclosures of range of expressions, however the limitation of the methods is the overestimation. In this paper we propose to calculate the enclosure of range of multivariate polynomial using Bernstein basis function.

■ Introduction

The problem of reliability of computer computations is one of great concern to specialists in many areas of science and engineering. The notion of computing estimates of numerical error in computer simulations is not new. In recent years, considerable progress has been made in determining theoretical and computational techniques that aid to improve the reliability of results of simulations. An important advance in this area has been the recent discovery of methods to determine upper and lower bounds of local approximation error in any given simulation. It is predicted that computable bounds giving upper and lower limits to computed quantities of interest could be a natural by−product in every simulation [Ode2003].

The idea of validated computation (verified computation) is based on the rigorous estimation of the influences of uncertainties on the calculation. Such uncertainties arise mainly from two reasons. First is associated with computational inaccuracies based on the finite accuracy of the computational environments. Second reason is the fact that there are uncertainties in the variables of the model to be analyzed.

A fundamental problem in range methods is computing the ranges of values of real functions or polynomials. Interval methods provide rigorous enclosures of range of expressions, however the limitation of the methods is the overestimation. The one of tool to control overestimation was proposed by Moore [Moo1966, Moo1979]. Author proposed the tool of subdivision to compute the range of function to a desired accuracy. Moore actually suggested two different strategies for subdivision called uniform subdivision and adaptive subdivision. Both these strategies are well known and are extensively used in various applications of interval analysis.
Nataraj and Sheela in the paper [Nat2002] presented a new subdivision strategy in interval analysis for computing the ranges of functions. The authors showed through several real−world examples that the proposed in the paper parallel−adaptive strategy is more efficient than the widely used uniform and adaptive subdivision strategies. In this work, parallel means simultaneous processing of all boxes present in a subdivision partition. Unfortunately, in case of multivariate functions, the computational expense by simple interval methods increases astronomically, especially with subdivision strategy.

In paper [Usc2005] a *Mathematica* package for affine arithmetic was presented and applied for validated computations of initial value problems of ordinary differential equations. Affine arithmetic was proposed by Stolfi and Comba [Sto2003] in order to tackle the conservativeness problem caused by standard interval arithmetic. Similarly to interval arithmetic, affine arithmetic can be used for manipulating imprecise values and for evaluating functions over intervals. Also, likewise interval arithmetic, it provides guaranteed bounds for computed results, although affine arithmetic also takes into consideration the correlation or dependencies between the error sources.

Another well known method for range analysis is the Bernstein coefficient method based on the Bernstein convex hull property. This method relies on the simple idea that if a polynomial is written in the Bernstein basis, the range of the polynomial is bounded by the values of the minimum and maximum Bernstein coefficients. A modification of this method elevates the degree of the Bernstein polynomials before using the coefficients to find the range. It is known that as the degree is elevated, the bounds become tighter, but an additional computational cost is involved.

In this paper the Bernstein form for multivariate polynomials [Ber2000] is used to construct for approximating the range of the multivariate polynomial ([Lin1995], [Lin1996]). This method produce much tighter and more accurate intervals than interval arithmetic. The paper [Mar2002] compares the performance and efficiency of different function range interval methods for plotting f(x, y)=0 on a rectangular region based on a subdivision scheme, where f (x, y) is a polynomial.

The bounds obtained by the Bernstein coefficients method can be tightened if the unit box is bisected into subboxes and Bernstein expansion is applied to the polynomial on these subboxes [Gar2001]. In paper [Gar2001] authors applied the expansion of a multivariate polynomial into Bernstein polynomial to compute solution of systems of polynomial equations. Systems of polynomial equations appear in a great variety applications, e.g. in geometric intersection computations, chemical equilibrium problems, combustions, and kinematics, to name only a few.

■ Interval arithmetic

Interval arithmetic is the arithmetic defined on sets of intervals, rather than sets of real numbers. A form of interval arithmetic perhaps first appeared in 1924 and 1931 in [Bur1924], [You1931], than later in [Sun1958]. Modern development of interval arithmetic began with Moore's dissertation [Moo1962], [Moo1966]. Since then, thousands of research articles and books have been published on the subject [Ale1974], [Ale1998], [Ale2000]. There is an interesting amount of software support for interval computations.

The primary motivation for using interval analysis in almost all applications is that the interval extension of a function provides bounds for the variation of the function. It means that the interval arithmetic evaluation of real function always contains the range of the function. This comes from the fundamental property of interval arithmetic.

Unfortunately the conventional interval methods do not carry the information on functional dependency. Let's consider simple computation of powers. Note that the computation of powers can be done in several ways.
To see this consider the following examples.

```
xint1 = Interval[{-1, 2}]
xint2 = xint1 * xint1
xint3 = xint1^2
```

```
Interval[{-1, 2}]

Interval[{-2, 4}]

Interval[{0, 4}]
```

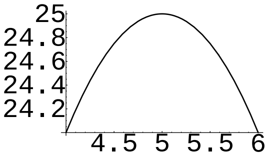
The dependency problem in interval arithmetic is typically caused by cancellation effects. For example, if x=[a,b], then y=x−x, if not recognised to represent the same number, is computed as [a−b]−[a,b]=[a,b]+[−b,−a]=[a−b,b−a], resulting in a width that is not zero, but twice as large as before. Such a situation can not be avoided in conventional interval methods in practice if the cancelling terms appear as a result of preceding complicated science computations.

```
f1[x_] := x * (10 - x); x0 = Interval[{4, 6}]; r1 = f1[x0]
```

```
Interval[{16, 36}]
```

The overestimation of enclosures of range of given polynomial using interval arithmetic is really large. The real range for above function is [24,25].

```
Plot[f1[x], {x, Min[x0], Max[x0]}, ImageSize → 100]
```



```
- Graphics -
```

■ Bernstein form of multivariate polynomial

Let us consider the multivariate polynomial in s real variables with the maximum degree $n_1 + ... + n_s$, that is,

$$p\ (x_1, x_2, ..., x_s) = \sum_{i_1=1}^{n_1} ... \sum_{i_s=1}^{n_s} a_{i_1 .. i_s} * x_1^{\ i_1} ... x_s^{\ i_s} \tag{1}$$

where $(x_1, x_2, ..., x_s) \in [x_{11}, x_{12}]\times[x_{21}, x_{22}]\times...\times[x_{s1}, x_{s2}]$.
We can also assume that $[x_{11}, x_{12}]=[x_{21}, x_{22}]=...=[x_{s1}, x_{s2}]=[0,1]$. Any finite interval [a,b] can be mapped to [0,1] by a linear transformation $\frac{x-a}{b-a}$: $\frac{[a,b]-a}{b-a}$=[0,1].
We introduce the Bernstein basis function

$$B_{j,k}(x) = \binom{k}{j} x^j (1-x)^{k-j} \tag{2}$$

where $x \in [0,1]$.

```
Bernstein[x_, k_, j_] :=
  Module[{}, Binomial[k, j] * Power[x, j] * Power[(1 - x), k - j]];
```

It can be easily shown that

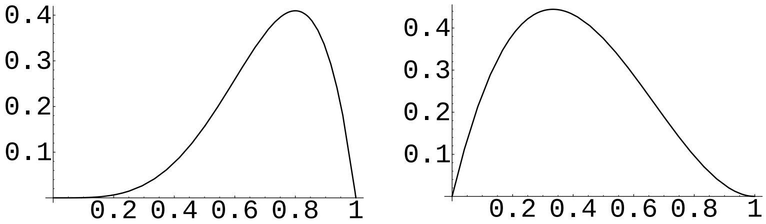
$$B_{j,k}(x)>0 \tag{3}$$

$$\sum_{j=0}^{k} B_{j,k}(x) = 1 \tag{4}$$

$$x^i = \sum_{j=0}^{k} \frac{\binom{j}{i}}{\binom{k}{i}} B_{j,k}(x) \tag{5}$$

for x=[0,1] and $i=0,1,2,...,n \leq k$.

```
p1 = Plot[Bernstein[x, k = 5, j = 4],
   {x, 0, 1}, DisplayFunction → Identity];
p2 = Plot[Bernstein[x, k = 3, j = 1], {x, 0, 1},
   DisplayFunction → Identity];
Show[GraphicsArray[{p1, p2}], ImageSize -> 300];
```



Two examples of Bernstein basis function $B_{4,5}$ (left plot) and $B_{1,3}$ (right plot)

```
Simplify[Sum[Bernstein[x, k, j], {j, 0, k}]]
```

1

```
k0 = 30;
i0 = 10;
Simplify[
 Sum[Binomial[j, i0] / Binomial[k0, i0] * Bernstein[x, k0, j],
  {j, 0, k0}]]
```

x^{10}

We can write polynomial (1) using (2)

$$p(x_1, \ldots, x_s) = \sum_{i_1=0}^{n_1} \ldots \sum_{i_s=0}^{n_s} a_{i_1 \ldots i_s} \times \sum_{j_1=0}^{k_1} \frac{\binom{j_1}{i_1}}{\binom{k_1}{i_1}} B_{j_1,k_1}(x_1) \ldots \sum_{j_s=0}^{k_s} \frac{\binom{j_s}{i_s}}{\binom{k_s}{i_s}} B_{j_s,k_s}(x_s) \quad (6)$$

After some mathematical manipulations we can write polynomial in the Bernstein form

$$p(x_1, \ldots, x_s) = \sum_{j_1=1}^{k_1} \ldots \sum_{j_s=1}^{k_s} b_{i_1 .. i_s} * B_{j_1,k_1}(x_1) \ldots B_{j_s,k_s}(x_s) \quad (7)$$

where

$$b_{j_1 \ldots j_s} = \sum_{i_1=0}^{\min(j_1,n_1)} \ldots \sum_{i_s=0}^{\min(j_s,n_s)} a_{i_1 .. i_s} \frac{\binom{j_1}{i_1}}{\binom{k_1}{i_1}} \ldots \frac{\binom{j_s}{i_s}}{\binom{k_s}{i_s}} \quad (8)$$

and $k_i \geq n_i$ for $i = 1..s$.

The quantities b of Bernstein form for multivariate polynomials (2) can be used to construct for approximating the range p([0,1],...,[0,1]) of the polynomial (1).
It can be proved that ([Lin1995], [Lin1996])

$$p([0, 1], \ldots, [0, 1]) \subseteq \left[\min_{j_1 \ldots j_s} b_{j_1 \ldots j_s}, \max_{j_1 \ldots j_s} b_{j_1 \ldots j_s} \right]. \quad (9)$$

```
powxi[x_, k_, i_] := Module[{j},
   Sum[(Binomial[j, i] / Binomial[k, i]) *
     Subscript[B, j, k][x], {j, i, k}]];

pow2ber[pp_, vars_, k_, prnout_] :=
  Module[{pb, pj, pj1, pj2, epj, blj, lofp, n},
   pb = 0;
   lofp = Length[pp];
   n = Length[k];
   coef = poly2coeffast[pp, vars];
   Do[pj = pp[[j]];
    epj = Exponent[pj, vars];
    blj = Table[
      powxi[Subscript[x, i], k[[i]], epj[[i]]], {i, 1, n}];
    pj1 = coef[[j]] * Apply[Times, blj];
    pj2 = Expand[pj1];
    If[prnout == 1,
     Print["PART No=", j];
     Print[{pj, epj}];
     Print[{blj}];
     Print[{pj1, pj2}]
    ];
    pb += pj2
    , {j, 1, lofp}
   ];
   Return[pb]
  ];

poly2coeffast[pp_, vars_] := Module[{pos1, pos2, pos, coef},
   pos1 = Position[pp, _Rational];
   pos2 = Position[pp, _Integer, 2];
   pos3 = Position[pp, _Real, 2];
   pos = Join[pos1, pos2, pos3];
   coef = Extract[pp, pos];
   Return[coef]
  ];

tosubform[poly_] := Module[{vars, varsnew},
   vars = Variables[poly];
   varsnew = Table[Subscript[x, i], {i, Length[vars]}];
   polynew = poly /. Thread[vars -> varsnew];
   Return[polynew]
  ];

to01form[{xmin_, xmax_}, x_] := (xmax - xmin) * x + xmin;
widthint[int_] := Max[int] - Min[int]
```

■ Numerical results

□ Example 1

Let us consider polynomial

$$p\ (x_1, x_2) = 2 x_1 + 3 x_2 - x_1 x_2 \qquad (10)$$

over domain $(x_1, x_2) \in [4, 6]\times[4.6]$ with $k_1 = k_2 = 2$.

First we choose the power form polynomial which will be transformed to Bernstein form.

```
poly = 2 * x + 3 * y - x * y;
poly0 = tosubform[poly];
vars0 = Variables[poly0];
ndim = Length[vars0];
Print["Polynomial: ", {poly, poly0}];
Print["Variables: ", vars0];
```

Polynomial: $\{2\,x + 3\,y - x\,y,\ 2\,x_1 + 3\,x_2 - x_1\,x_2\}$

Variables: $\{x_1, x_2\}$

We will calculate enclosure of real range of polynomial for following intervals values of variables of polynomial

```
xint = Table[Interval[{Subscript[x, i, 1], Subscript[x, i, 2]}],
  {i, 1, ndim}]
```

$\{\mathrm{Interval}[\{x_{1,1}, x_{1,2}\}], \mathrm{Interval}[\{x_{2,1}, x_{2,2}\}]\}$

Numerical values of intervals are given below

```
vxint = Table[Interval[{4, 6}], {i, 1, ndim}]
```

```
{Interval[{4, 6}], Interval[{4, 6}]}
```

To apply Bernstein form of polynomial we have to first transform any finite interval [a,b] to [0,1].

```
varsnew0 =
  Table[to01form[{Subscript[x, i, 1], Subscript[x, i, 2]},
    Subscript[x, i]], {i, 1, ndim}];
vxran = Table[{Min[vxint[[i]]], Max[vxint[[i]]]}, {i, 1, ndim}];
varsnew1 = Table[to01form[{vxran[[i, 1]], vxran[[i, 2]]},
   Subscript[x, i]], {i, 1, ndim}]
```

$\{4 + 2\,x_1, 4 + 2\,x_2\}$

Next we have to transform polynomial variables to form with $x_i = [0, 1]$. Considered polynomial (10) defined over [0,1] x [0,1] we can write in th following form

```
poly1 = poly0 /. Thread[vars0 -> varsnew1];
poly2 = Expand[poly1];
Print["Polynomial: ", poly2];
```

Polynomial: $4 - 4\,x_1 - 2\,x_2 - 4\,x_1\,x_2$

Now we can derive Bernstein form of polynomial, pb0.

```
klist = Exponent[poly2, vars0];
klist = Max[klist] * Table[1, {ndim}] + 1;
prnout = 0;
pb0 = pow2ber[poly2, vars0, klist, prnout];
vars1 = Variables[pb0];
Length[vars1];
```

The quantities (coefficient) of Bernstein form for multivariate polynomials are used to construct for approximating the range of the polynomial. The width of interval which represent enclosure of range of considered polynomial is calculated too.

```
coeflist = poly2coeffast[pb0, vars1];
nofc = Length[coeflist];
If[nofc < 5, Print[coeflist];
  Table[{pb0[[i]], coeflist[[i]]}, {i, 1, nofc}]
 ];
coefrange = Interval[{Min[coeflist], Max[coeflist]}];
Print["Enclosure of range: ", coefrange];
Print["Width of enclosure: ", widthint[coefrange] // N];
```

```
Enclosure of range: Interval[{-6, 4}]
```

```
Width of enclosure: 10.
```

We can calculate the enclosure of real range of polynomial using interval arithmetic.

```
polyintval = poly0 /. Thread[vars0 -> vxint];
Print["Polynomial and variables range: ", {poly0, vxint}];
Print["Enclosure of range: ", polyintval];
Print["Width of enclosure: ", widthint[polyintval] // N];
```

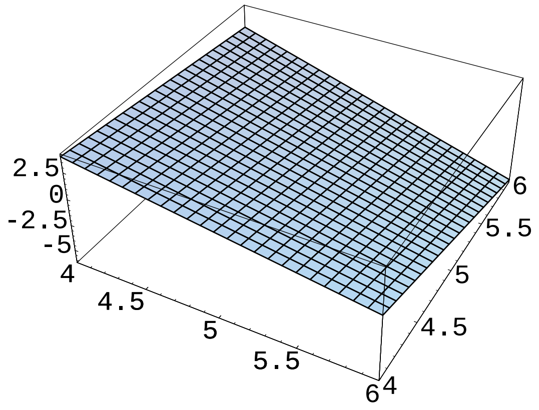
```
Polynomial and variables range:
 {2 x1 + 3 x2 - x1 x2, {Interval[{4, 6}], Interval[{4, 6}]}}
```

```
Enclosure of range: Interval[{-16, 14}]
```

```
Width of enclosure: 30.
```

As we can see in considered case the width of resulting interval is three times less when we calculate the enclosure of range of polynomial using Bernstein form method.
If dimension of polynomial is equal two we can plot considered polynomial

```
If[ndim == 2,
  poly0;
  vxint;
  Plot3D[poly0,
   {Subscript[x, 1], Min[vxint[[1]]], Max[vxint[[1]]]},
   {Subscript[x, 2], Min[vxint[[2]]], Max[vxint[[2]]]},
   ImageSize -> 200];
 ];
```



Plot of real range of polynomial

□ Example 2

We can use mentioned method for calculating enclosure of multivariate polynomial.

$$p\ (x_1, x_2, x_3) = -4\, x_1\, x_2\, x_3 + 2\, x_2^2\, x_3^3 \qquad (11)$$

over domain $(x_1, x_2, x_3) \in [-1, 2]\times[-1,2]\times[-1,2]$ with $k_1 = k_2 = k_3 = 4$.

```
poly = -4 * x * y * z + 2 * y^2 * z^3;
poly0 = tosubform[poly];
vars0 = Variables[poly0];
ndim = Length[vars0];
Print["Polynomial: ", {poly, poly0}];
Print["Variables: ", vars0];
```

Polynomial: $\{-4\, x\, y\, z + 2\, y^2\, z^3,\ -4\, x_1\, x_2\, x_3 + 2\, x_2^2\, x_3^3\}$

Variables: $\{x_1, x_2, x_3\}$

```
xint = Table[Interval[{Subscript[x, i, 1], Subscript[x, i, 2]}],
   {i, 1, ndim}];
vxint = Table[Interval[{-1, 2}], {i, 1, ndim}]
varsnew0 =
  Table[to01form[{Subscript[x, i, 1], Subscript[x, i, 2]},
    Subscript[x, i]], {i, 1, ndim}];
(**)
vxran = Table[{Min[vxint[[i]]], Max[vxint[[i]]]}, {i, 1, ndim}];
varsnew1 = Table[to01form[{vxran[[i, 1]], vxran[[i, 2]]},
   Subscript[x, i]], {i, 1, ndim}]
```

```
{Interval[{-1, 2}], Interval[{-1, 2}], Interval[{-1, 2}]}
```

$\{-1 + 3\, x_1,\ -1 + 3\, x_2,\ -1 + 3\, x_3\}$

Considered polynomial (11) defined over [0,1] x [0,1] x [0,1] we can write in th following form

```
poly1 = poly0 /. Thread[vars0 -> varsnew1];
poly2 = Expand[poly1]
```

$$2 - 12\, x_1 + 36\, x_1\, x_2 - 18\, x_2^2 + 6\, x_3 + 36\, x_1\, x_3 - 72\, x_2\, x_3 - 108\, x_1\, x_2\, x_3 + 162\, x_2^2\, x_3 - 54\, x_3^2 + 324\, x_2\, x_3^2 - 486\, x_2^2\, x_3^2 + 54\, x_3^3 - 324\, x_2\, x_3^3 + 486\, x_2^2\, x_3^3$$

Next we can derive Bernstein form of polynomial, pb1.

```
klist = Exponent[poly2, vars0];
klist1 = Max[klist] * Table[1, {ndim}] + 1;
prnout = 0;
pb1 = pow2ber[poly2, vars0, klist1, prnout];
vars1 = Variables[pb1];
Length[vars1];
```

Now we can calculate the enclosure of range of considered polynomial.

```
coeflist = poly2coeffast[pb1, vars1];
nofc = Length[coeflist];
coefrange = Interval[{Min[coeflist], Max[coeflist]}];
Print["Enclosure of range: ", coefrange];
Print["Width of enclosure: ", widthint[coefrange] // N];
```

```
Enclosure of range: Interval[{-28, 80}]
```

```
Width of enclosure: 108.
```

The bounds obtained by the Bernstein coefficients method can be tightened if the degree of basis Bernstein function is elevated, but an additional computational cost is involved.

```
klist2 = Max[klist] * Table[1, {ndim}] + 4;
pb1 = pow2ber[poly2, vars0, klist2, prnout];
vars1 = Variables[pb1];
coeflist = poly2coeffast[pb1, vars1];
coefrange = Interval[{Min[coeflist], Max[coeflist]}];
Print["Enclosure of range: ", coefrange];
Print["Width of enclosure: ", widthint[coefrange] // N];
```

```
Enclosure of range: Interval[{-16, 80}]
```

```
Width of enclosure: 96.
```

The width of enclosure of range of considered polynomial calculated using interval arithmetic is greater.

```
polyintval = poly0 /. Thread[vars0 -> vxint];
Print["Polynomial and variables range: ", {poly0, vxint}];
Print["Enclosure of range: ", polyintval];
Print["Width of enclosure: ", widthint[polyintval] // N];
```

Polynomial and variables range: $\{-4\, x_1\, x_2\, x_3 + 2\, x_2^2\, x_3^3,$
  {Interval[{-1, 2}], Interval[{-1, 2}], Interval[{-1, 2}]}}

```
Enclosure of range: Interval[{-40, 80}]
```

```
Width of enclosure: 120.
```

□ Example 3

Now we can use presented method for calculating enclosure of fifth–degree polynomial.

```
poly = 2 * y^2 * z^3 - 4 * v - 4 * x * y * z * w * v;
poly0 = tosubform[poly];
vars0 = Variables[poly0];
ndim = Length[vars0];
Print["Polynomial: ", {poly, poly0}];
Print["Variables: ", vars0];
```

Polynomial:
$\{-4\, v - 4\, v\, w\, x\, y\, z + 2\, y^2\, z^3,\ -4\, x_1 - 4\, x_1\, x_2\, x_3\, x_4\, x_5 + 2\, x_4^2\, x_5^3\}$

Variables: $\{x_1, x_2, x_3, x_4, x_5\}$

```
xint = Table[Interval[{Subscript[x, i, 1], Subscript[x, i, 2]}],
   {i, 1, ndim}];
vxint = Table[Interval[{-1, 2}], {i, 1, ndim}]
varsnew0 =
  Table[to01form[{Subscript[x, i, 1], Subscript[x, i, 2]},
    Subscript[x, i]], {i, 1, ndim}];
(**)
vxran = Table[{Min[vxint[[i]]], Max[vxint[[i]]]}, {i, 1, ndim}];
varsnew1 = Table[to01form[{vxran[[i, 1]], vxran[[i, 2]]},
   Subscript[x, i]], {i, 1, ndim}]
```

```
{Interval[{-1, 2}], Interval[{-1, 2}],
 Interval[{-1, 2}], Interval[{-1, 2}], Interval[{-1, 2}]}
```

$\{-1 + 3\, x_1,\ -1 + 3\, x_2,\ -1 + 3\, x_3,\ -1 + 3\, x_4,\ -1 + 3\, x_5\}$

Considered polynomial defined over [0,1] x [0,1] x [0,1] x [0,1] x [0,1] we can write in th following form

```
poly1 = poly0 /. Thread[vars0 -> varsnew1];
poly2 = Expand[poly1]
```

$$\begin{aligned}&6-24\,x_1-12\,x_2+36\,x_1\,x_2-12\,x_3+36\,x_1\,x_3+36\,x_2\,x_3-108\,x_1\,x_2\,x_3+\\&\;36\,x_1\,x_4+36\,x_2\,x_4-108\,x_1\,x_2\,x_4+36\,x_3\,x_4-108\,x_1\,x_3\,x_4-\\&\;108\,x_2\,x_3\,x_4+324\,x_1\,x_2\,x_3\,x_4-18\,x_4^2+6\,x_5+36\,x_1\,x_5+36\,x_2\,x_5-\\&\;108\,x_1\,x_2\,x_5+36\,x_3\,x_5-108\,x_1\,x_3\,x_5-108\,x_2\,x_3\,x_5+324\,x_1\,x_2\,x_3\,x_5-\\&\;72\,x_4\,x_5-108\,x_1\,x_4\,x_5-108\,x_2\,x_4\,x_5+324\,x_1\,x_2\,x_4\,x_5-108\,x_3\,x_4\,x_5+\\&\;324\,x_1\,x_3\,x_4\,x_5+324\,x_2\,x_3\,x_4\,x_5-972\,x_1\,x_2\,x_3\,x_4\,x_5+162\,x_4^2\,x_5-\\&\;54\,x_5^2+324\,x_4\,x_5^2-486\,x_4^2\,x_5^2+54\,x_5^3-324\,x_4\,x_5^3+486\,x_4^2\,x_5^3\end{aligned}$$

Next we can derive Bernstein form of polynomial, pb1.

```
klist = Exponent[poly2, vars0];
klist1 = Max[klist] * Table[1, {ndim}] + 3;
prnout = 0;
pb1 = pow2ber[poly2, vars0, klist1, prnout];
vars1 = Variables[pb1];
Length[vars1];
```

Now we can calculate the enclosure of range of considered polynomial.

```
coeflist = poly2coeffast[pb1, vars1];
nofc = Length[coeflist];
coefrange = Interval[{Min[coeflist], Max[coeflist]}];
Print["Enclosure of range: ", coefrange];
Print["Width of enclosure: ", widthint[coefrange] // N];
```

```
Enclosure of range: Interval[{-88, 132}]
```

```
Width of enclosure: 220.
```

The width of enclosure of range of considered polynomial calculated using interval arithmetic is greater again.

```
polyintval = poly0 /. Thread[vars0 -> vxint];
Print["Polynomial and variables range: ", {poly0, vxint}];
Print["Enclosure of range: ", polyintval];
Print["Width of enclosure: ", widthint[polyintval] // N];
```

```
Polynomial and variables range:
```

$$\begin{aligned}&\{-4\,x_1-4\,x_1\,x_2\,x_3\,x_4\,x_5+2\,x_4^2\,x_5^3,\\&\;\{\text{Interval}[\{-1,2\}],\text{Interval}[\{-1,2\}],\text{Interval}[\{-1,2\}],\\&\;\;\text{Interval}[\{-1,2\}],\text{Interval}[\{-1,2\}]\}\}\end{aligned}$$

```
Enclosure of range: Interval[{-144, 132}]
```

```
Width of enclosure: 276.
```

■ About the Author

Dr. Tomasz Strek works in the Institute of Applied Mechanics at the Poznan University of Technology, Poland. Strek is The Annual Stipends for Young Scientist of The Foundation for Polish Science (2001,2002) and First Prize of Mechanics Committee of Polish Academy of Science for Young Scientist (2005).
He is researcher in three primary areas: computational fluid dynamics, dynamical systems and verified computations. He uses *Mathematica* extensively in research and teaching.

Address
Institute of Applied Mechanics
Poznan University of Technology
Piotrowo 3, 60–965 Poznan, Poland
tomasz.strek@put.poznan.pl
www.tomasz.strek.prv.pl

■ References

[Ale1974] G. Alfeld and J. Herzberger, *Introduction to Interval Computations,* Wissenschaftsverlag, Mannheim, Wien, Zürich (1974).

[Ale1998] G. Alefeld and D. Claudio, The basic properties of interval arithmetic, its software realization and some applications. *Computer and Structures* **67,** pp.3−8 (1998).

[Ale2000] G. Alfeld and G. Mayer, Interval analysis: theory and applications. *Journal of Computational and Applied Mathematics* **121,** pp.421−464 (2000).

[Ber2000] J. Berchtold, I. Voiculescu, A. Bowyer, Interval arithmetic applied to multivariate Bernstein−form polynomials, Technical report Number 31/98, School of Mechanical Engineering, University of Bath, 2000.

[Bur1924] J.C. Burkill, Function of intervals. *Proccedings of the London Mathematical Society* **22,** pp.375−446 (1924).

[Gar2001] J.Garloff, A.p. Smith, Solution of Systems of Polynomial Equations by Using Bernstein Expansion, Symbolic Algebraic Methods and Verification Methods, Editors: G. Alefeld, J. Rohn, S. Rump, T. Yamamoto, Springer Mathematics, Wien, pp.87−97, (2001).

[Mar2002] Ralph Martin, Huahao Shou, Irina Voiculescu, Adrian Bowyer, Guojin Wang. Comparison of interval methods for plotting algebraic curves, Computer Aided Geometric Design 19, pp.553-587 (2002).

[Moo1962] R.E. Moore, *Interval Arithmetic and Automatic Error Analysis in Digital Computing.* PhD thesis, Stanford University (1962).

[Moo1966] R.E. Moore, *Interval Analysis,* Prentice Hall, Englewood Cliffs (1966).

[Moo1979] R.E. Moore, *Methods and applications of interval analysis,* SIAM, Philadelphia (1979).

[Nat2002] P.S.V. Nataraj and S.M. Sheela, A New Subdivision Strategy for Range Computations. *Reliable Computing* **8,** pp.83−92 (2002).

[Lin1995] Q. Lin and J.G. Rokne, Methods for bounding the range of a polynomial. *Journal of Computational and Applied Mathematics* **58,** pp.193−199 (1995).

[Lin1996] Q. Lin and J.G. Rokne, Interval Approximation of Higher Order to the Ranges of Functions, *Computers and Math. Applic.* **31** (7), pp.101−109 (1996).

[Ode2003] J.T. Oden, T. Belytschko, I. Babuska and T.J.R. Hughes, Research directions in computational mechanics, *Comput. Methods Appl. Mech. Engrg.* **192,** pp.913−922 (2003).

[Sto2003] J. Stolfi and L. H. de Figueiredo, “An Introduction to Affine Arithmetic,” *TEMA Tend. Mat. Appl. Comput*, **4**(3), pp. 297-312 (2003).

[Sun1958] T. Sunaga, Theory of an interval algebra and its application to numerical analysis. *RAAG Memories* **2,** pp.29−46 (1958).

[You1931] Y.C. Young, The algebra of many−valued quantities. *Math. Ann.*, **104,** pp. 260−290 (1931).

[Usc2005] A.Uscilowska, A Package for Affine Arithmetic, *Applied Mathematica: Proceedings of the 7th International Mathematica Symposium*, eProceedings of IMS 2005, The University of Western Australia, Perth, Australia, 5−8.08.2005, Editors: Paul Abbott and Shane McCarthy, Wolfram Media Inc., Champaign USA, 2005.