Bernstein Branch-and-Bound Algorithm for Unconstrained Global Optimization of Multivariate Polynomial MINLPs

P. S. V. Nataraj

Systems and Control Engineering Group, IIT Bombay, India

Bhagyesh V. Patil

Laboratoire d'Informatique de Nantes Atlantique, France

Overview

- Introduction
  - Mixed-Integer Non-Linear Programming (MINLP)
- Existing solution approaches
- Background
  - The Bernstein form
- Proposed strategy with enhancements
  - Main algorithm
  - Numerical Experiments
- Concluding remarks

Introduction

- A MINLP is a optimization problem of the following form:

$$
\begin{aligned}
& \min_{x} \; f(x) \\
\text{subject to} \quad & g(x) \le 0 \\
& h(x) = 0 \\
& x_k \in \mathrm{X} \subseteq \mathbb{R}, \quad k = 1, 2, \ldots, l_d \\
& x_k \in \{0, 1, \ldots, q\} \subset \mathbf{Z}, \quad k = l_d + 1, \ldots, l \\
& q \ge 0, \; q \in \mathbf{Z}
\end{aligned}
$$

- We consider *unconstrained* MINLPs wherein f *polynomial* in nature.
- Minimize MINLP *globally.*

Existing solution approaches

- Generalized Benders Decomposition (Geoffrion, *J Optim Theory Appl,* 1972).
- Branch-and-Bound (Gupta et al., *Management Science*, 1985).
- Outer Approximation (Duran et al., *Mathematical Programming*, 1986).
- Branch-and-Cut (Padberg et al., *Operations Research Letters*, 1987).
- An LP/NLP-based-Branch-and-Bound (Grossmann et al., *Comp Chem Engg*, 1992).
- Extended Cutting Plane method (Westerlund et al., *Comp Chem Engg*, 1995).
- Limited to *convex MINLPs*.

cont...

- Two approaches for nonconvex MINLPs:

  - Use of convex underestimators (Tawarmalani et al., *Math Program*, 2003) (α-BB, BARON)

  - Use of equivalent convex formulation (Liberti, *Ph.D. thesis*, 2004) (Bonmin, COUENNE)

- Both approaches limited to only standard forms (*linear*, *bilinear, trilinear terms*).

The Bernstein form

- Consider the n variate polynomial *p* in power form over $x^I \in \mathrm{U} = [0,1]$.

$$p(x) = \sum_{I \le N} a_I x^I, \quad a_I \in \text{i}$$

- The equivalent Bernstein form is

$$p(x) = \sum_{I \le N} b_I(\mathrm{U}) B_{N,I}(x)$$

where $B_{N,I}(x)$ are the Bernstein basis polynomials (C. G. Lorentz, Bernstein Polynomials, 1997).

Cont...

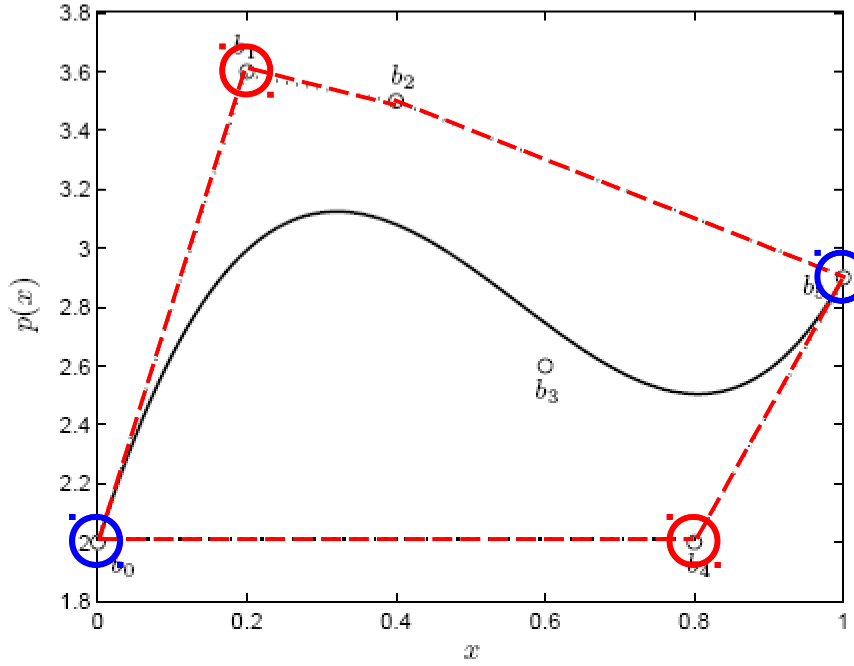
- b_I are the Bernstein coefficients over U, can be computed as follows:

$$b_I(\mathrm{U}) = \sum_{J \le I} \frac{\binom{I}{J}}{\binom{N}{J}} a_J$$

- The unit interval is not really a restriction as any finite interval X can be linearly transformed to it.

- We shall use the *matrix method* to compute the Bernstein coefficients (Shaswati Ray and P.S.V. Nataraj, *J. Glob. Opti.*, 2009).

Properties of the Bernstein form



The polynomial function, its Bernstein coefficients, and the convex hull.

Range Enclosure

Convex Hull

Vertex Property

Convex hull property

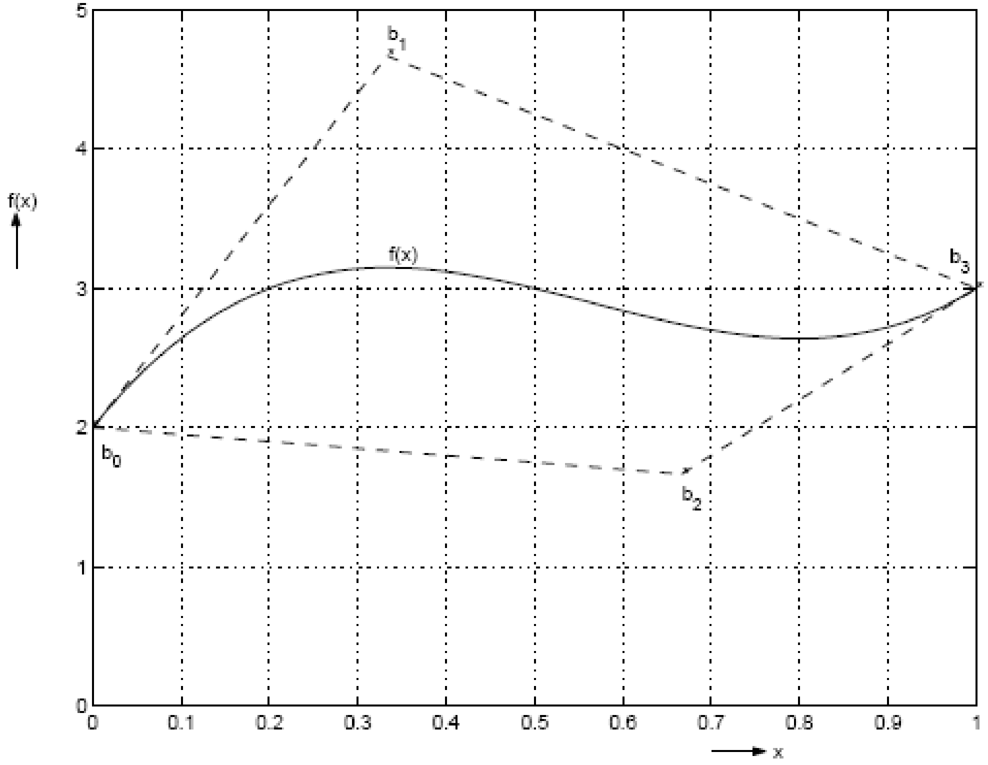


Figure : The polynomial function, its Bernstein coefficients, and the convex hull

Subdivision procedure

- *Tightening* of bounds is possible by subdivision.
- Bernstein coefficients of the subdivided boxes can be computed from the Bernstein coefficients of the original box.
- Thus avoids the repeated computation of Bernstein coefficients of the function.
- Subdivision direction can be selected based on any one of the existing subdivision *direction selection* rules (T. Csendes and D. Ratz, *J. Glob. Optim.*, 1997).
- Simplest one is subdivide along direction of *maximum width*

$$y(r) = \max(w(d))$$

where d box to be subdivided, and r is a direction in which it is subdivided.

Subdivision using the Bernstein form

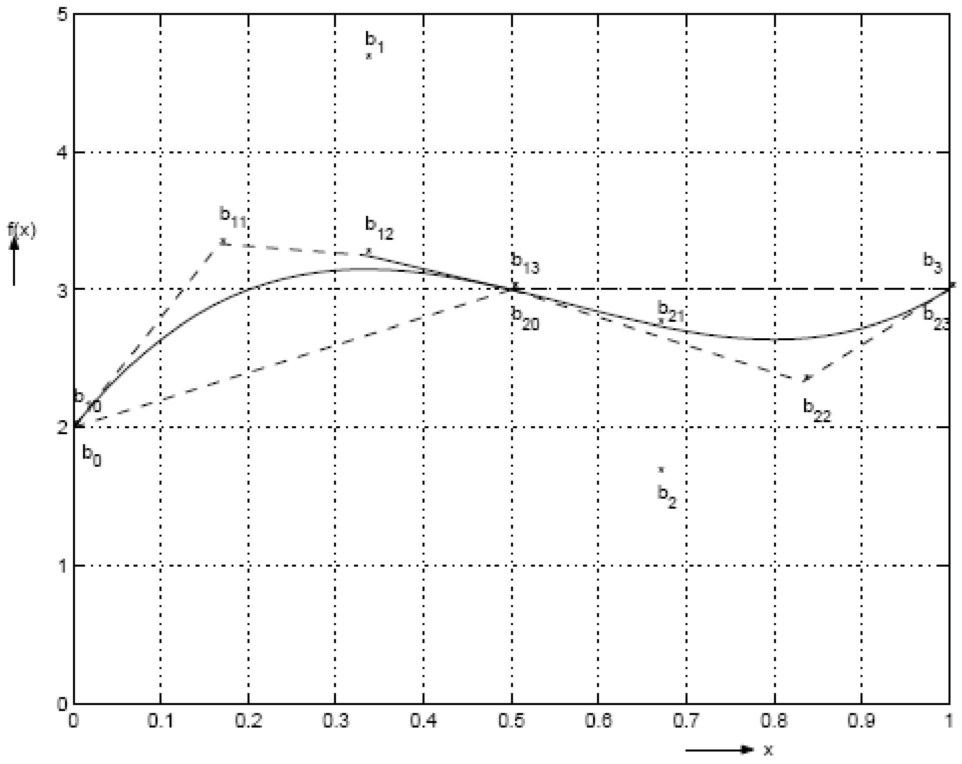


Figure : Improvement in range enclosure with subdivision

Cut-off test

- In global minimization algorithms, cut-off test is used to *delete boxes*, which are sure not to contain the global minimum.

- The usual procedure is to assign the maximum value of Bernstein coefficients of the objective function as initial cut-off test value.

- Any box whose minimum Bernstein coefficients value is greater than this will be deleted.

- If the maximum Bernstein coefficients value of any box is lesser than the present cut-off test value, the cut-off test value is updated using this value.

Unconstrained global optimization algorithm

- We proposed a new Bernstein unconstrained global optimization algorithm

  - a modified subdivision procedure

  - a combinations of different accelerating devices (cut-off, monotonicity, concavity tests)

  - a Bernstein box and hull consistencies algorithms for domain pruning

Modified subdivision procedure

- Continuous decision variables

$$\mathrm{x} = [\underline{\mathrm{x}_1}, \overline{\mathrm{x}_1}] \times \ldots \times [\underline{\mathrm{x}_r}, \overline{\mathrm{x}_r}] \times \ldots \times [\underline{\mathrm{x}_l}, \overline{\mathrm{x}_l}]$$

$$\mathrm{x}_A = [\underline{\mathrm{x}_1}, \overline{\mathrm{x}_1}] \times \ldots \times [\underline{\mathrm{x}_r}, m(\mathrm{x}_r)] \times \ldots \times [\underline{\mathrm{x}_l}, \overline{\mathrm{x}_l}]$$

$$\mathrm{x}_B = [\underline{\mathrm{x}_1}, \overline{\mathrm{x}_1}] \times \ldots \times [m(\mathrm{x}_r), \overline{\mathrm{x}_r}] \times \ldots \times [\underline{\mathrm{x}_l}, \overline{\mathrm{x}_l}]$$

- Integer decision variables

$$\mathrm{x} = [\underline{\mathrm{x}_1}, \overline{\mathrm{x}_1}] \times \ldots \times [\underline{\mathrm{x}_r}, \overline{\mathrm{x}_r}] \times \ldots \times [\underline{\mathrm{x}_l}, \overline{\mathrm{x}_l}]$$

$$\mathrm{x}_A = [\underline{\mathrm{x}_r}, \lfloor m(\mathrm{x}_r) \rfloor]$$

$$\mathrm{x}_B = [\lfloor m(\mathrm{x}_r) + 1 \rfloor, \overline{\mathrm{x}_r}]$$

where, $m(\mathrm{x}_r)$ denotes the midpoint of $[\underline{\mathrm{x}_r}, \overline{\mathrm{x}_r}]$.

Domain pruning

- Generally, subintervals of original box are thrown to isolate the global minimum.
- Classical tools, such as interval function evaluation, cut-off test monotonicity and concavity tests are used.
- Sometimes, gives no information about nonexistence of stationary points.
- Alternative: *consistency techniques*.

cont…

$$\text{Consider } f(x, y) = 0 \quad x \in \mathrm{X},\ y \in \mathrm{Y}$$

$$\text{a) if } \forall x \in \mathrm{X},\ \exists y \in \mathrm{Y}, \text{ such that } f(x, y) = 0$$

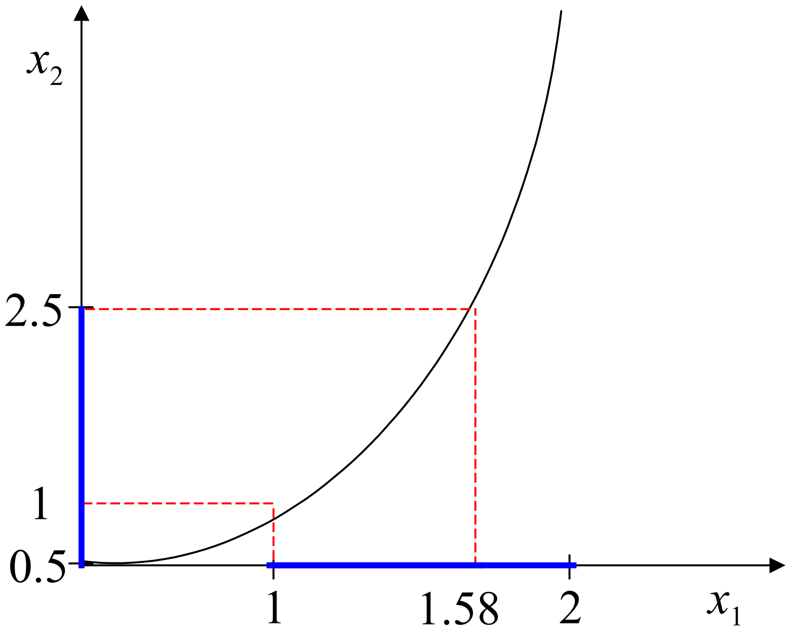
$$\text{b) if } \forall y \in \mathrm{Y},\ \exists x \in \mathrm{X}, \text{ such that } f(x, y) = 0$$

- Box consistency
  - Compute *box consistent* region for an equation by box (width) narrowing operations for a chosen variable.
- Hull consistency
  - Compute *hull consistent* region for an equation by constraint inversion for a selected term and chosen variable.

Hull consistency procedure

$$x_2 = x_1^2$$

$$x_1 \in [1, 2]\ ,\ x_2 \in [0.5, 2.5]$$



$$x_2 = x_2 \cap x_1^2$$
$$= [0.5, 2.5] \cap [1, 4]$$
$$= [1, 2.5]$$

$$x_1 = x_1 \cap x_2^{1/2}$$
$$= [1, 2] \cap [0.7071, 1.5812]$$
$$= [1, 1.5812]$$

Box consistency procedure

$$x_2 - x_1^2 = 0$$

$$x_1 \in [1,2]\ ,\ x_2 \in [0.5, 2.5]$$

Left end point narrowing (a=1)

$$\begin{aligned} \mathrm{h}(\mathrm{X}_a) &= \mathrm{X}_2 - (a)^2 \\ &= [0.5, 2.5] - (1)^2 \\ &= [-0.5, 1.5] \end{aligned}$$

$$0 \in \mathrm{h}(\mathrm{X}_a) \Rightarrow \text{Cannot be increased}$$

Right end point narrowing (b=2)

$$\begin{aligned} \mathrm{h}(\mathrm{X}_b) &= \mathrm{X}_2 - (b)^2 \\ &= [0.5, 2.5] - (2)^2 \\ &= [-3.5, -1.5] \end{aligned}$$

$$0 \notin \mathrm{h}(\mathrm{X}_b) \Rightarrow \text{Can be decreased}$$

Right end point narrowing cont…

$$X_{1,new} = X_1 \cap N(X_1)$$

$$= [1,2] \cap \left(b - \frac{h(X_b)}{h'x_1} \right)$$

$$= [1,2] \cap \left(2 - \frac{[-3.5,-1.5]}{-2 \times [1,2]} \right)$$

$$= [1,2] \cap [1,1.6251]$$

$$= [1,1.6251]$$

$$x_2 - x_1^2 = 0,\ x_1 \in [1,1.6251],\ x_2 \in [0.5,2.5]$$

Possible set of constraints

- At global minimum of the objective function all the first partial derivatives of the objective function should be 0.

$$f'_r(x) = 0, \quad r = 1, 2, \ldots, l$$

- Similarly, we can apply consistency to following relation

$$f(x) \leq \widetilde{p}$$

$\widetilde{p}$ being the upper bound on the global minimum.

Bernstein unconstrained global optimization algorithm

Step 1. Compute the Bernstein coefficient array of the objective polynomial on ***y***.

Step 2. Set $\widetilde{p} := \infty$ and *y* as the min. Bernstein value over Bernstein coefficient array of the objective polynomial.

Step 3. Initialize the lists $\mathcal{L} := \{(\mathrm{y}, (b_o(\mathrm{y})), y)\}$, $\mathcal{L}^{sol} := \{\}$.

Step 4. Pick the first item from list deleting its entry. If list is empty go to step 11.

Step 5. Subdivide the box ***y*** (along longest width direction) into two boxes using modified subdivision procedure. Obtain Bernstein coefficient arrays of objective polynomial on these subboxes.

Step 6. If minimum on subbox greater than $\widetilde{p}$, then discard. Else update $\widetilde{p}$ and stored the item appropriately in list $\mathcal{L}$

Step 7. (Cut-off Test) Discard the items from list $\mathcal{L}$ whose minimum greater than $\widetilde{p}$

Step 8. Apply monotonicity and concavity tests along with **domain contraction steps** based on the Bernstein hull and Bernstein box consistency techniques.

Step 9. Check the vertex condition. If 'true' within specified accuracy, then put the vertex point and domain in the solution list. Go to step 4.

Step 10. If termination criteria is satisfied, store item in the solution list L^{sol}. Else go to step 4.

Step 11. Analyze the solution list and return the global minimum and global minimizer(s).

Numerical tests

We consider 10 test problems#

Dimension - 3 to 9

Integer variables - 1 to 6

- Performance comparison based on
  (a) Use of classical tools (cut-off, monotonicity, concavity) and their combinations
  (b) Use of Bernstein consistency algorithms for domain pruning
  (c) Use of combination of (a) and (b)

- Performance metric considered total number of boxes processed for all 10 test problems to find the global minimum.

All 10 problems taken from PHC pack, the database of polynomial systems, *Technical report*, Mathematics Department, University of Illinois, Chicago, USA, 2001. Problems where modified as MINLPs by restricting some decision variables as Integer.

Use of classical tools

Performance metric	No accelerating device	Only Cut-off	Only Mono.	Only Concavity	Cut-off + Mono.	Cut-off + Mono. + Concavity
Total boxes processed	10,356	**2,290**	8,366	**9,734**	**1,870**	1,870

Performance metric	Only Cut-off	Cut-off + BBC to $f(x) \le \tilde{p}$	Cut-off + BHC to $f(x) \le \tilde{p}$	Cut-off + BBC to $f(x) \le \tilde{p}$ + BHC to $f(x) \le \tilde{p}$
Total boxes processed	2,290	**1,438**	2,004	**1,242**

cont...

Performance metric	Only Mono.	Mono. + BBC to $f(x) \le \tilde{p}$	Mono. + BHC to $f(x) \le \tilde{p}$	Mono. + BHC to $f(x) \le \tilde{p}$ + BBC to $f(x) \le \tilde{p}$
Total boxes processed	8,366	**1,250**	1562	**816**

Performance metric	Only Concavity	Concavity + BBC to $f(x) \le \tilde{p}$	Concavity + BHC to $f(x) \le \tilde{p}$	Concavity + BHC to $f(x) \le \tilde{p}$ + BBC to $f(x) \le \tilde{p}$
Total boxes processed	9,734	**1,430**	1,938	**1,172**

Use of Bernstein box and hull consistency algorithms

Performance metric	Cut-off + Mono. + BBC to $f_r^{'}(x) = 0$	Cut-off + Mono. + BHC to $f_r^{'}(x) = 0$	Cut-off + Mono. + BBC to $f_r^{'}(x) = 0$ + BHC to $f_r^{'}(x) = 0$
Total boxes processed	**803**	1,051	**788**

Concluding remarks

- The proposed strategy has been found to give *guaranteed* global minimum without using any convexification or linearization technique.
- Among the classical tool combinations; cut-off and monotonicity combination found efficient.
- Among the classical tool combinations with Bernstein consistency algorithms to the relation $f(x) \leq \tilde{p}$; monotonicity performance was superior compared to cut-off and concavity.
- Overall, combination of cut-off and monotonicity along with Bernstein consistency algorithms to the relation $f_r'(x) = 0,$ is found to perform best.

Thank You

Qualitative features of the proposed strategy over the existing solvers

- No initial *guess* required, only initial search domain required.
- No function and gradient *evaluations* required.
- Bounds on the global optimum are *guaranteed*.
- No prior knowledge of stationary points required.
- No *convexification* or *linearization* needed.
- No need of multiple trials (like with genetic algorithms).

Views about existing solvers

- Most of the existing MINLP solvers can be classified as *branch and bound* solvers, solvers based on the *linear relaxation* of the functions, or solvers based on the combination of the *relaxation* and *linearization* of functions.

- Branch and bound solvers uses NLP relaxation by relaxing integrality restriction (Bonmin (B-BB mode), *fminconset*, SBB, MILANO, LINDOBB). However, NLP solver used to solve the NLP relaxation usually ensures only local optimal solutions, they work as heuristics in case of a nonconvex MINLP.

- Another class of solvers utilizes linear relaxation of objective and constraint functions (AOA, Bonmin (B-OA mode), DICOPT). In particular, outer approximation uses gradient based linearization, but yields outer approximation only for convex MINLPs. For nonconvex problems, sometimes the outer approximation method may not be able to generate a sufficiently accurate outer approximation to the master problem. In such cases, we may found a large number of near optimal solutions without ever finding an optimal solution.

Cont…

- An alternative way can be the use of cutting plane method (AlphaECP), where an sequence of MILP relaxation is solved and optimal solution for MINLP is obtained by adding the cutting planes. However, generation of cutting plane can be time consuming and we will get a MINLP solution only at the end.

- Since gradient based linearization ensures global solutions only for convex MINLPs, some solvers (BARON) uses an additional convexification step to branch also on continuous variables in nonconvex terms.