Towards Global Solution of Semi-infinite Programs

Global Optimization Theory Institute,
Argonne National Laboratory

8th September 2003

Paul I. Barton and Binita Bhattacharjee
Department of Chemical Engineering, MIT

Outline

- Mathematical formulation of a semi-infinite program (SIP)
- Examples and engineering applications
- Overview of lower-bounding methods
  - Discretization-based approaches
  - Reduction-based approaches
- The inclusion-constrained reformulation approach
- Global optimization of semi-infinite programs
- Conclusions

General Form of a Semi-infinite Program (SIP)

An objective function which is expressed in terms of a *finite number of optimization variables*, $\boldsymbol{x}$, is minimized subject to an *infinite number of constraints*, which are expressed over a compact set P of infinite cardinality:

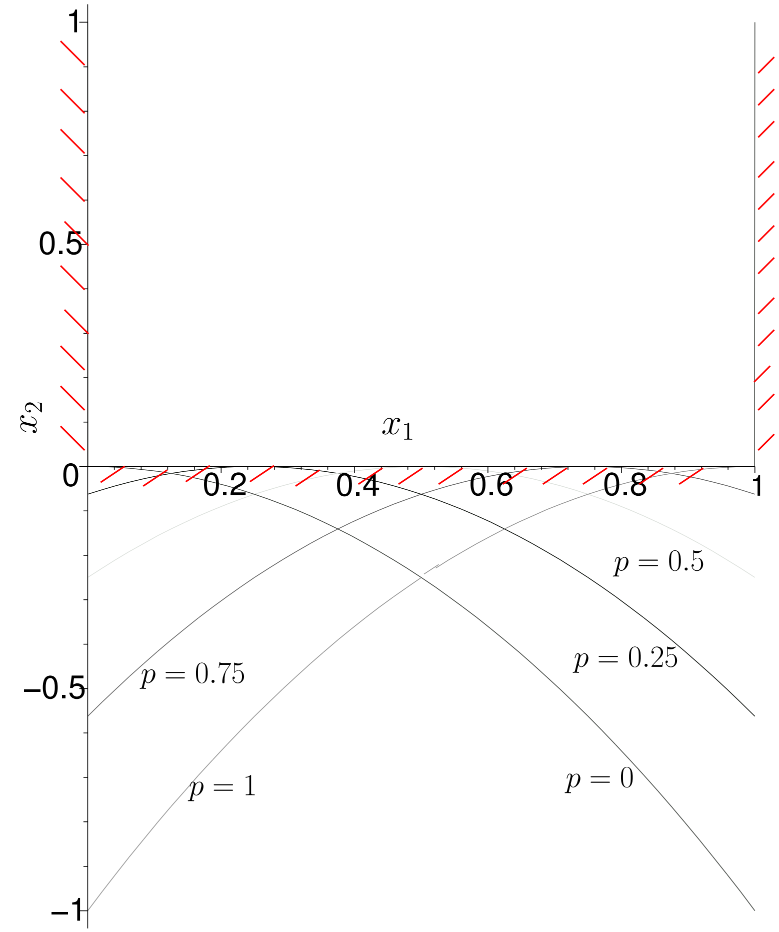
$$
\begin{aligned}
&\min_{\boldsymbol{x} \in X} f(\boldsymbol{x}) \\
&g(\boldsymbol{x}, \boldsymbol{p}) \leq 0 \qquad \forall \boldsymbol{p} \in P \subset \mathbb{R}^{n_p} \\
&|P| = \infty, \quad X \subset \mathbb{R}^{n_x}
\end{aligned}
$$

The global SIP algorithm makes additional mild assumptions

- P and X are Cartesian products of intervals
- $f(\boldsymbol{x})$ is once-continuously differentiable in $\boldsymbol{x}$
- $g(\boldsymbol{x}, \boldsymbol{p})$ is continuous in $\boldsymbol{p}$ and once-continuously differentiable in $\boldsymbol{x}$

SIP Example

$$\min_{\boldsymbol{x}} x_2$$
$$-(x_1 - p)^2 - x_2 \leq 0 \quad \forall p \in [0, 1]$$
$$0 \leq x_1 \leq 1^{a}$$



[a] Hettich, R. and Kortanek, K.O., Semi-infinite Programming: Theory, Methods and Applications, *SIAM Review*, **35**:380-429, 1993.

Engineering Applications

- Robotic trajectory planning
- Design and operation under uncertainty, robust solutions
- Material stress modeling
- Rigorous ranges of validity for (kinetic) models with parametric uncertainty

General Form of a SIP

$$\min_{\boldsymbol{x} \in X} f(\boldsymbol{x})$$

$$g(\boldsymbol{x}, \boldsymbol{p}) \leq 0 \qquad \forall \boldsymbol{p} \in P \subset \mathbb{R}^{n_p}$$

$$|P| = \infty, \quad X \subset \mathbb{R}^{n_x}$$

Exact Finite Reformulation

Numerical solution techniques for SIPs generally rely on constructing a finite reformulation to which known results and algorithms from nonlinear programming (NLP) can be applied. However, in the general case, the exact finite reformulation is nonsmooth:

$$\min_{\boldsymbol{x}\in X} f(\boldsymbol{x})$$
$$\tilde{g}(\boldsymbol{x}) \equiv \max_{\boldsymbol{p}\in P} g(\boldsymbol{x},\boldsymbol{p}) \leq 0$$

When $f(\boldsymbol{x})$, and/or $g(\boldsymbol{x},\boldsymbol{p})$ are nonconvex, this problem:

- Cannot be solved to global optimality using traditional nonsmooth optimization methods.
- May be solved to global optimality using bilevel programming techniques - such an approach does not exploit the special structure of the SIP.

Existing Numerical Methods for SIPs

Instead of solving the exact finite reformulation, an iterative algorithm is used to generate a convergent sequence of upper or lower bounds on the SIP solution.

- Lower-bounding approaches:
  - Discretization
  - Reduction
- Upper-bounding approach:
  - Inclusion-constrained reformulation

Lower-Bounding Algorithms for SIPs

At each iteration, k,

- Select a *finite* subset of points $D_k \subset P$
- Formulate the following finitely-constrained subproblem:

$$\min_{\boldsymbol{x} \in X} f(\boldsymbol{x})$$
$$g(\boldsymbol{x}, \boldsymbol{p}) \le 0 \quad \forall \boldsymbol{p} \in D_k$$

- Solving the subproblem to global optimality yields a rigorous lower bound on the SIP minimum f^{SIP}:

$$\{\boldsymbol{x} \in X : g(\boldsymbol{x}, \boldsymbol{p}) \le 0 \quad \forall \boldsymbol{p} \in D_k\} \supset \{\boldsymbol{x} \in X : g(\boldsymbol{x}, \boldsymbol{p}) \le 0 \quad \forall \boldsymbol{p} \in P\}$$
$$\Downarrow$$
$$f^{SIP} \ge f_k^D$$

Convergence of Lower-Bounding Approaches

- Under appropriate assumptions:
  - $\lim_{k \to \infty} f_k^D = f^{SIP}$
  - Any accumulation point of the sequence $\{\boldsymbol{x}^k\}$ 'solves' the SIP, i.e., the algorithm converges to the 'type' of point (global min/stationary point/KKT point) for which each subproblem is solved.
- The feasibility of the solution cannot be guaranteed at finite termination, even when subproblems are solved to global optimality.
- The feasibility of an incumbent solution $\boldsymbol{x}^k$ can be tested by solving a global maximization problem:

$$\max_{\boldsymbol{p} \in P} g(\boldsymbol{x}^k, \boldsymbol{p})$$

Discretization-based Methods

- Require relatively mild assumptions on problem structure
- Each member set in the sequence $\{D_k\}$ either postulated a priori, or updated adaptively, e.g.

$$D_{k+1} = D_k \cup \{\boldsymbol{p} : \boldsymbol{p} = \arg\max_{\boldsymbol{p} \in S} g(\boldsymbol{x}^k, \boldsymbol{p})\}$$
$$S \subset P, \quad |S| < \infty$$

- Computational cost increases rapidly with the dimensionality of P and the number of iterations, k, since $\lim_{k\to\infty} \sup_{\boldsymbol{p}_1 \in P} \inf_{\boldsymbol{p}_2 \in D_k} ||\boldsymbol{p}_1 - \boldsymbol{p}_2|| = 0$ is required to guarantee convergence of the method.
- In practice, global optimization methods are ignored, and subproblems are solved only for stationary/KKT points $\Rightarrow$ accumulation points of $\{\boldsymbol{x}^k\}$ are stationary/KKT points of the SIP, not global minima.

Reduction-based Methods

- Index set $D_{k+1} = \{p_l\}^k$ where $\{p_l\}^k$ is the set of local maximizers of $g(\boldsymbol{x}^k, \boldsymbol{p})$ on P.
- At each iteration, k, solve

$$\min_{\boldsymbol{x} \in X^*} f(\boldsymbol{x})$$
$$g(\boldsymbol{x}, \boldsymbol{p}_l(\boldsymbol{x})) \leq 0 \quad \forall l = 1, \dots, r_l$$

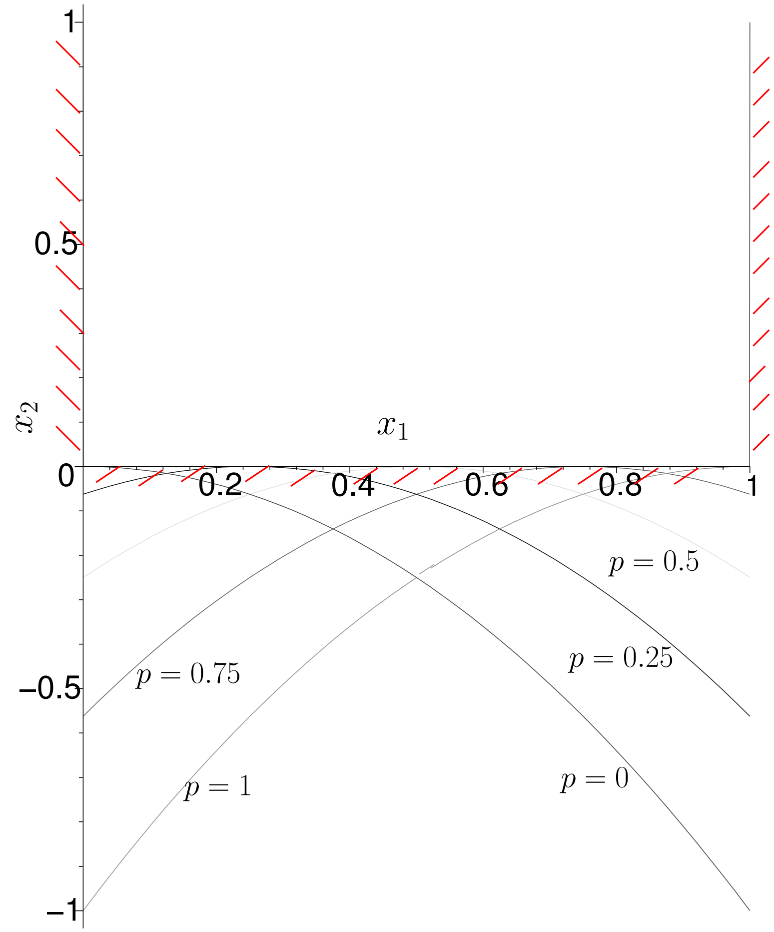
  where $X^* \subset X$ is a neighborhood of a SIP solution. Typically neither the 'valid' neighborhood X^*, nor the number of local maximizers, r_l, are known explicitly.
- Convergence requires strong regularity conditions to be satisfied
- 'Local' reduction methods require an initial starting point in the vicinity X^* of the SIP solution. Convergent 'globalized' reduction methods make even stronger assumptions.
- Computationally cheaper than discretization methods since $|D_k| = r_l \quad \forall k$.

Example: Pathological Case

The feasible set cannot be represented by a finite number of constraints from P

$$\min_{\boldsymbol{x}} x_2$$
$$-(x_1-p)^2 - x_2 \le 0 \quad \forall p \in [0,1]$$
$$0 \le x_1 \le 1$$

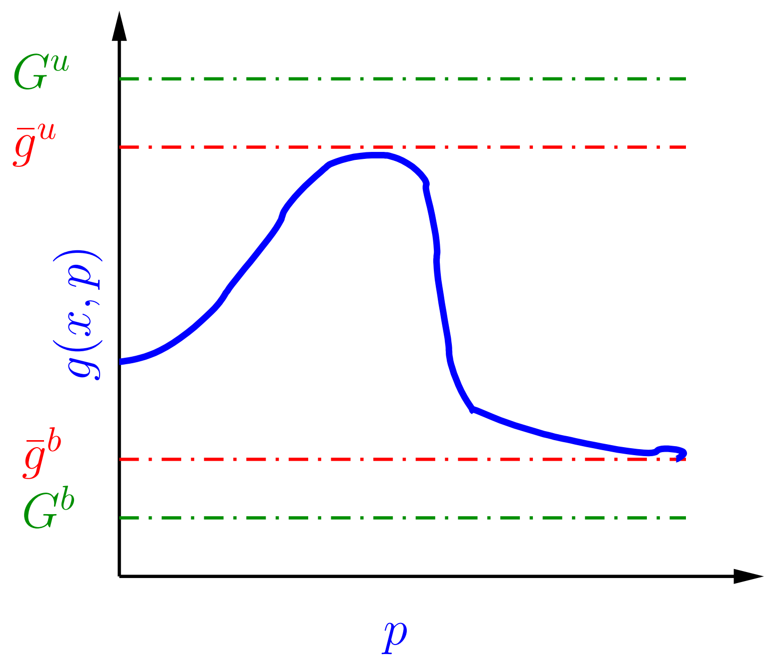
$\Rightarrow$ An upper bounding approach is required to identify feasible solutions to such problems.



Inclusion Functions

An inclusion for a function $g(\boldsymbol{x}, \boldsymbol{p})$ on an interval P can be calculated using interval analysis techniques such that this inclusion $G(\boldsymbol{x}, P)$ is a superset of the true image of the function g on P, i.e.,

$$\{g(\boldsymbol{x}, \boldsymbol{p}) : \boldsymbol{p} \in P\} = [\bar{g}^b,\ \bar{g}^u] \subset [G^b,\ G^u] = G(\boldsymbol{x}, P)$$



The natural interval extension is the simplest inclusion that can be calculated for a continuous, real-valued function.

Upper-bounding Problem for the SIP

A subset of the SIP-feasible set may be represented using an inclusion of $g(\boldsymbol{x}, \boldsymbol{p})$ on P:

$$\{\boldsymbol{x} \in X : \max_{\boldsymbol{p} \in P} g(\boldsymbol{x},\ \boldsymbol{p}) \leq 0\} \supset \{\boldsymbol{x} \in X : G^u(\boldsymbol{x},\ P) \leq 0\}$$

This relation suggests the following finite, inclusion-constrained reformulation (ICR), which may be solved for an upper bound $f^{ICR} \geq f^{SIP}$:

$$\begin{gathered}\min_{\boldsymbol{x} \in X} f(\boldsymbol{x}) \\ G^u(\boldsymbol{x}, P) \leq 0\end{gathered}$$

Any local solution of this problem will be a SIP-feasible upper bound.

Example

$$\min_{\boldsymbol{x}\in X} \tfrac{1}{3}x_1^2 + x_2^2 + \tfrac{1}{2}x_1$$

$$\left(1 - x_1^2 p^2\right)^2 - x_1 p^2 - x_2^2 + x_2 \leq 0 \quad \forall p \in [0, 1]$$

Nonsmooth Reformulation

Min/Max terms which appear in the natural interval extension of $g(\boldsymbol{x}, \boldsymbol{p})$ result in a nondifferentiable optimization problem (which is nonetheless much easier to solve than the exact bilevel programming formulation).

$$\begin{gathered}
\min_{\boldsymbol{x} \in X, \boldsymbol{p}^b \in P^b, \boldsymbol{p}^u \in P^u} \tfrac{1}{3}x_1^2 + x_2^2 + \tfrac{1}{2}x_1 \\
p_2^b = (p_1^b)^2 \\
p_2^u = (p_1^u)^2 \\
p_3^b = -x_1 - 2x_1^2 + x_1^4 \cdot p_2^b \\
p_3^u = -x_1 - 2x_1^2 + x_1^4 \cdot p_2^u \\
p_4^u = \max\left(p_2^u \cdot p_3^u,\ p_2^b \cdot p_3^u,\ p_2^b \cdot p_3^b,\ p_2^u \cdot p_3^b\right) \\
1 + x_2 - x_2^2 + p_4^u \leq 0 \\
p_1^b = 0,\ p_1^u = 1
\end{gathered}$$

- Solve the nonsmooth problem to local optimality using nondifferentiable optimization techniques, or
- Reformulate the nonsmooth problem as an equivalent NLP/MINLP which may be solved to global optimality for a (potentially) tighter upper bound on the SIP minimum value.

Solving the Inclusion-constrained Reformulation to Global Optimality

Reformulation as equivalent smooth NLP

- No additional nonlinearities due to reformulation
- Problem size (number of constraints) grows exponentially with the complexity of the constraint expression.

Reformulation as equivalent MINLP with smooth relaxations

- Binary variables introduce additional nonlinearities
- Problem size (number of binary variables) grows polynomially with the complexity of the constraint expression.

Results from Literature Examples

Problem	f^{PCW}	$\max_{\boldsymbol{p}} g(\boldsymbol{x}^{PCW}, \boldsymbol{p})$	f^{ICR}	$\max_{\boldsymbol{p}} g(\boldsymbol{x}^{ICR}, \boldsymbol{p})$	G^u	CPU
1^b	-0.25	0	-0.25	0	0	0.03
2^b	0.1945	$-2.5 \cdot 10^{-8}$	0.1945	$-2.5 \cdot 10^{-8}$	0	0.42
3^b	5.3347	$5.3 \cdot 10^{-6}$	39.6287	-0.1233	0	0.06
$4^b(n_x{=}3)$	0.6490	$-2.7 \cdot 10^{-7}$	1.5574	-0.6505	0	0.02
$4^b(n_x{=}6)$	0.6161	0.	1.5574	0	0	0.03
$4^b(n_x{=}8)$	0.6156	0	1.5574	0	0	0.03
5^b	4.3012	$1.5 \cdot 10^{-8}$	4.7183	0	0	0.05
6^b	97.1588	$-5.9 \cdot 10^{-7}$	97.1588	$5.7 \cdot 10^{-6}$	0	0.09
7^b	1	0	1	0	0	0.02
8^b	2.4356	$9.9 \cdot 10^{-8}$	7.3891	$-3.9 \cdot 10^{-6}$	0	0.01
9^b	-12	0	-12	0	0	0.02
K^c	-3	0	-3	0	0	0.02
L^c	0.3431	$9.6 \cdot 10^{-6}$	1	-0.2929	0	0.03
M^c	1	0	1	0	0	0.01
N^c	0	0	0	0	0	0.02
$S^c(n_p = 3)$	-3.6743	-1.1640	-3.6406	-2.9997	0	0.33
$S^c(n_p = 4)$	-4.0871	-1.1997	-4.0451	-0.7076	0	0.33
$S^c(n_p = 5)$	-4.6986	-2.1733	-4.4496	-0.7619	0	0.27
$S^c(n_p = 6)$	-5.1351	-2.6513	-4.8541	-2.6833	0	0.28
U^c	-3.4831	$2.4 \cdot 10^{-8}$	-3.4822	-0.0002	0	0.03

Convergence Property of Inclusion Functions

In the general case, the inclusion-constrained reformulation underestimates the feasible set of the SIP such that $f^{SIP} < f^{ICR}$. A better approximation of the SIP-feasible set is necessary to calculate a tighter upper bound for f^{SIP}. The properties of convergent inclusion functions can be exploited to derive tighter inclusion bounds $G^u(\boldsymbol{x}, P)$:

$$G^u(\boldsymbol{x}, P) - \bar{g}^u(\boldsymbol{x}, P) \leq \gamma w(P)^\beta$$

where $w(P) = p^u - p^b$, $\beta \geq 1$, and $0 \leq \gamma < \infty$.

Since $G^u \rightarrow g^u$ as $w(P) \downarrow$ and $\beta \uparrow$, tighter inclusions for the constraint set are obtained using:

- Subdivision: $G^u(\boldsymbol{x}, P) \geq G^u_k(\boldsymbol{x}, P) \geq \bar{g}^u(\boldsymbol{x}, P)$ where

$$G_k = \bigcup_{m \in I_k} G(\boldsymbol{x}, P_k), \qquad \bigcup_{m \in I_k} P_k = P$$

- Higher order inclusion function, e.g. $\beta = 2$ for Taylor models

Convergence Results

Problem	n_x	n_p	$ndiv_{TM}$	CPU_{TM}	$ndiv_{IE}$	CPU_{IE}
3[b]	3	1	16	172	512	291
4[b]	3	1	4	0.1	256	0.42
5[b]	3	1	2	0.40	16	0.16
L[c]	2	1	16	0.68	512	60.48

- Higher-order Taylor models result in convergence over much fewer iterations than natural interval extensions
- Fewer iterations (and correspondingly smaller NLP subproblems) do not necessarily result in lower solution times for the Taylor model formulations
- Reported CPU times do not reflect computational effort required to generate Taylor coefficients.

[b] G.A. Watson, Numerical Experiments with Globally Convergent Methods forSemi-infinite Programming Problems, in *Semi-Infinite Programming and Applications, Proceedings of an International Symposium*, Springer-Verlag, Heidelberg, Germany, Eds. A.V. Fiacco and K.O. Kortanek, 1983.

[c] C.J. Price and I.D. Coope, Numerical Experiments in Semi-infinite Programming, *Computational Optimization and Applications*, **6:**169-189, 1996.

Global Optimization of SIPs

Existing lower and upper-bounding methods can be combined in a branch-and-bound framework to solve SIPs to guaranteed global optimality. The convergence of the branch-and-bound alogorithm rests on two key results:

- $G_k^u(\boldsymbol{x}, P) \rightarrow \bar{g}^u(\boldsymbol{x}, P)$ as $\max\limits_{m \in I_k} w(P_m) \rightarrow 0$
- $f_k^D \rightarrow f^{SIP}$ as $\sup\limits_{\boldsymbol{p}_1 \in P} \inf\limits_{\boldsymbol{p}_2 \in D_k} ||\boldsymbol{p}_1 - \boldsymbol{p}_2|| \rightarrow 0$

Branch-and-Bound Framework

At each node solve

$$\min_{\boldsymbol{x}\in X_i} f_c(\boldsymbol{x})$$
$$g_c(\boldsymbol{x},\,\boldsymbol{p}) \leq 0 \quad \forall \boldsymbol{p} \in D_q$$

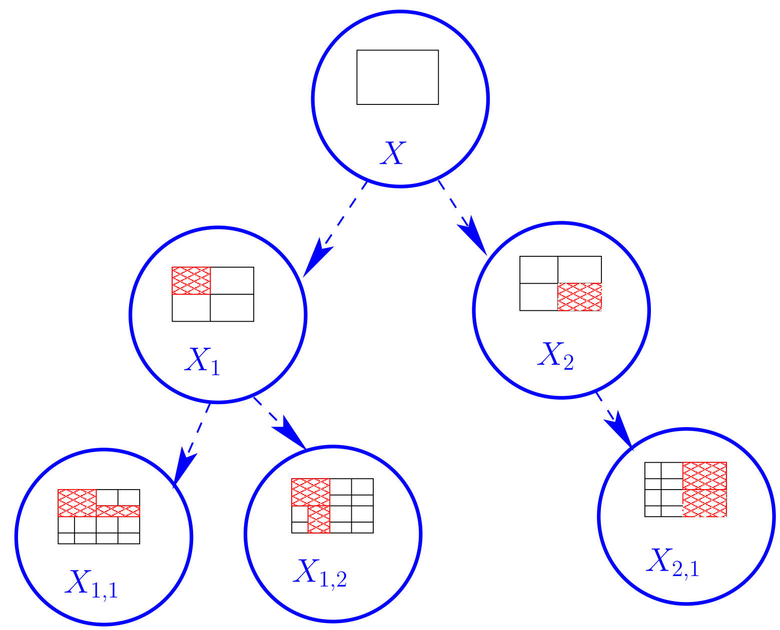
$$\min_{\boldsymbol{x}\in X_i} f(\boldsymbol{x})$$
$$G^u(\boldsymbol{x}, P_m) \leq 0 \quad \forall m \in I_q$$

- f_c, g_c are convex relaxations of f and g respectively
- q is the level of the branch-and-bound tree at which the node $X_i \subset X$ occurs
- D_q is the discretization grid used to define the lower-bounding problem for all nodes which occur at level q, $D_q \subset D_{q+1}\ \forall q$ and $\lim_{q\to\infty} \sup_{\boldsymbol{p}_1\in P} \inf_{\boldsymbol{p}_2\in D_q} ||\boldsymbol{p}_1 - \boldsymbol{p}_2|| = 0$
- $\{P_m\}$ is the partition of P used to define the upper-bounding problem for all nodes which occur at level q, $\max_{m\in I_q} w(P_m) > \max_{m\in I_{q+1}} w(P_{m+1})\ \forall q$, $\lim_{q\to\infty} \max_{m\in I_q} w(P_m) = 0$

Exclusion Heuristic

Upper-bounding problem:
Exclude subintervals P_m, $m \in I_q$ which generate inactive constraints at a node $X_i \subset X$ and its child nodes, i.e., those which satisfy

$$G^u(X_i, P_m) < 0$$



Lower-bounding problem:
Exclude points $\boldsymbol{p} \in D_q$ which generate inactive constraints at a node $X_i \subset X$ and its child nodes, i.e., those which satisfy

$$G^u_c(X_i, \boldsymbol{p}) < 0$$

Conclusions

- The inclusion-constrained reformulation can be used to identify feasible upper bound to the SIP solution value by solving a finite number of NLPs to local optimality (usually one). In many applications feasibility is more important than optimality.
- The inclusion-constrained reformulation yields a convergent sequence of upper bounds on the SIP solution value.
- When multiple iterations are required, the convergence rate of the inclusion-constrained reformulation is significantly improved by the use of higher-order inclusion functions.
- The SIP branch-and-bound framework enables the solution of general, nonlinear SIPs to finite ϵ-optimality by combining existing uppper and lower-bounding approaches for SIPs.