OSE SEMINAR 2011

A reformulation framework for global optimization

ANDREAS LUNDELL

CENTER OF EXCELLENCE IN
OPTIMIZATION AND SYSTEMS ENGINEERING
ÅBO AKADEMI UNIVERSITY

ÅBO, DECEMBER 8 2011

Main topics of the talk

- Want to introduce an algorithm for solving nonconvex mixed integer nonlinear problems (MINLPs) to global optimality.

Main topics of the talk

- Want to introduce an algorithm for solving nonconvex mixed integer nonlinear problems (MINLPs) to global optimality.
- The framework is an extension to that used in the signomial global optimization algorithm.

Main topics of the talk

- Want to introduce an algorithm for solving nonconvex mixed integer nonlinear problems (MINLPs) to global optimality.
- The framework is an extension to that used in the signomial global optimization algorithm.
- Convex reformulations in an extended variable space using variants of the αBB quadratic convex underestimator.

The signomial global optimization algorithm

- The SGO algorithm can solve mixed integer signomial problems to global optimality as a converging sequence of convex subproblems.

The signomial global optimization algorithm

- The SGO algorithm can solve mixed integer signomial problems to global optimality as a converging sequence of convex subproblems.
- Convex underestimators for signomial terms obtained through power and exponential transformations approximated by piecewise linear functions (PLFs).

The signomial global optimization algorithm

- The SGO algorithm can solve mixed integer signomial problems to global optimality as a converging sequence of convex subproblems.
- Convex underestimators for signomial terms obtained through power and exponential transformations approximated by piecewise linear functions (PLFs).
  - The transformations convexifying the problem is obtained by solving a MILP problem.

The signomial global optimization algorithm

- The SGO algorithm can solve mixed integer signomial problems to global optimality as a converging sequence of convex subproblems.
- Convex underestimators for signomial terms obtained through power and exponential transformations approximated by piecewise linear functions (PLFs).
  - The transformations convexifying the problem is obtained by solving a MILP problem.
  - The reformulated and overestimated convex problem is solved using a convex MINLP solver.

The signomial global optimization algorithm

- The SGO algorithm can solve mixed integer signomial problems to global optimality as a converging sequence of convex subproblems.
- Convex underestimators for signomial terms obtained through power and exponential transformations approximated by piecewise linear functions (PLFs).
  - The transformations convexifying the problem is obtained by solving a MILP problem.
  - The reformulated and overestimated convex problem is solved using a convex MINLP solver.
  - The convex relaxation is made tighter by iteratively including more breakpoints in the PLFs.

The signomial global optimization algorithm

- The SGO algorithm can solve mixed integer signomial problems to global optimality as a converging sequence of convex subproblems.
- Convex underestimators for signomial terms obtained through power and exponential transformations approximated by piecewise linear functions (PLFs).
  - The transformations convexifying the problem is obtained by solving a MILP problem.
  - The reformulated and overestimated convex problem is solved using a convex MINLP solver.
  - The convex relaxation is made tighter by iteratively including more breakpoints in the PLFs.
- The goal is to extend the problem scope to also include twice-differentiable functions.

The αGO/SGO algorithm

- The algorithm proposed here is based on the SGO algorithm together with quadratic convex underestimators for nonconvex C^2-functions
  - the original αBB underestimator
  - the spline αBB underestimator

The αGO/SGO algorithm

- The algorithm proposed here is based on the SGO algorithm together with quadratic convex underestimators for nonconvex C^2-functions
  - the original αBB underestimator
  - the spline αBB underestimator
- The signomial functions can be handled as either signomials or C^2-functions or using a mixture of the strategies.

The αGO/SGO algorithm

- The algorithm proposed here is based on the SGO algorithm together with quadratic convex underestimators for nonconvex C^2-functions
  - the original αBB underestimator
  - the spline αBB underestimator
- The signomial functions can be handled as either signomials or C^2-functions or using a mixture of the strategies.
- The result is a method able to solve nonconvex MINLP problems containing twice-differentiable functions to global optimality.

The considered class of MINLP problems

$$\begin{array}{lll} \text{minimize} & f(\mathbf{x}) & \mathbf{x} = (x_1, x_2, \ldots, x_N) \\ \text{subject to} & \mathbf{A}\mathbf{x} = \mathbf{a} \quad \mathbf{B}\mathbf{x} \leq \mathbf{b} & \\ & g_n(\mathbf{x}) \leq 0 & n = 1, 2, \ldots, J_n \\ & \underbrace{q_m(\mathbf{x})}_{\text{convex}} + \underbrace{\sigma_m(\mathbf{x})}_{\text{signomial}} + \underbrace{h_m(\mathbf{x})}_{\text{nonconvex}} \leq 0 & m = 1, 2, \ldots, J_m \end{array}$$

- The vector $\mathbf{x}$ can contain both continuous and integer-valued variables.
- The differentiable real functions f and g are (pseudo)convex, and the functions q, σ and h are convex, signomial and twice-differentiable (C^2) nonconvex respectively.
- A signomial function is of the form

$$\sigma(\mathbf{x}) = \sum_j c_j \prod_i x_i^{p_{ji}}.$$

Convex underestimation of C^2-functions

- A convex underestimator for twice-differentiable functions in a box-domain from, *e.g.,* Floudas (2000).

Convex underestimation of C^2-functions

- A convex underestimator for twice-differentiable functions in a box-domain from, *e.g.,* Floudas (2000).

Theorem

A function $g(\mathbf{x}) \in C^2$ *has the convex underestimator*

$$\boxed{L(\mathbf{x}) = g(\mathbf{x}) + \sum_i \alpha(\underline{x}_i - x_i)(\overline{x}_i - x_i)}$$

for $x_i \in [\underline{x}_i, \overline{x}_i]$ $\forall i$ *if and only if the parameter* α *fulfills*

$$\alpha \geq \max\left\{0, -\frac{1}{2}\min_i \lambda_i\right\}$$

where the λ_i*'s are the eigenvalues of the Hessian matrix of* $g(\mathbf{x})$ *on the interval* $[\underline{x}_i, \overline{x}_i]$.

- Several methods for calculating the α-values are available.

Example: a univariate nonconvex function

- The αBB underestimator for the function

$$h(x) = x \cdot \sin x + x/10$$

on the interval $0 \leq x \leq 8$ becomes

$$L(x) = h(x) + \underbrace{4.10293}_{\alpha} \cdot (0 - x)(8 - x).$$

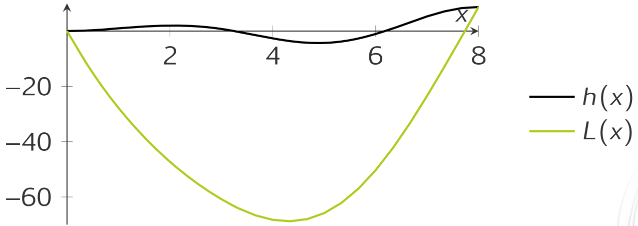
Example: a univariate nonconvex function

- The αBB underestimator for the function

$$h(x) = x \cdot \sin x + x/10$$

on the interval $0 \leq x \leq 8$ becomes

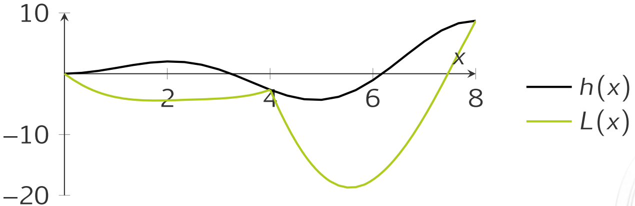
$$L(x) = h(x) + \underbrace{4.10293}_{\alpha} \cdot (0 - x)(8 - x).$$



Example: The αBB underestimator

- If the domain is partitioned, new αBB underestimators (with different α-values) can be obtained in each partition, *e.g.,*

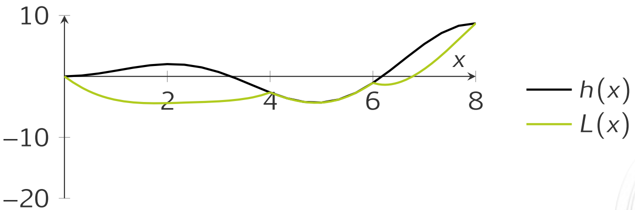
$$L(x) = \begin{cases} h(x) + 1.6 \cdot (0-x)(4-x) & \text{if } 0 \leq x \leq 4 \\ h(x) + 4.10293 \cdot (4-x)(8-x) & \text{if } 4 \leq x \leq 8 \end{cases}$$



Example: The αBB underestimator

- If the domain is partitioned, new αBB underestimators (with different α-values) can be obtained in each partition, *e.g.*,

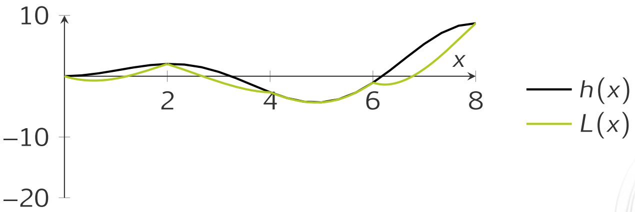
$$L(x) = \begin{cases} h(x) + 1.6 \cdot (0-x)(4-x) & \text{if } 0 \leq x \leq 4 \\ h(x) & \text{if } 4 \leq x \leq 6 \\ h(x) + 4.10293 \cdot (6-x)(8-x) & \text{if } 6 \leq x \leq 8 \end{cases}$$



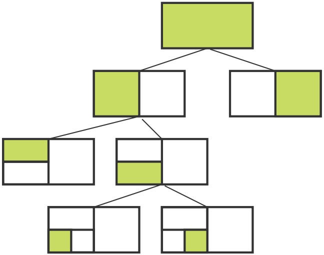
Example: The αBB underestimator

- If the domain is partitioned, new αBB underestimators (with different α-values) can be obtained in each partition, *e.g.*,

$$L(x) = \begin{cases} h(x) + 1.32544 \cdot (0-x)(4-x) & \text{if } 0 \le x \le 2 \\ h(x) + 1.6 \cdot (0-x)(4-x) & \text{if } 2 \le x \le 4 \\ h(x) & \text{if } 4 \le x \le 6 \\ h(x) + 4.10293 \cdot (6-x)(8-x) & \text{if } 6 \le x \le 8 \end{cases}$$

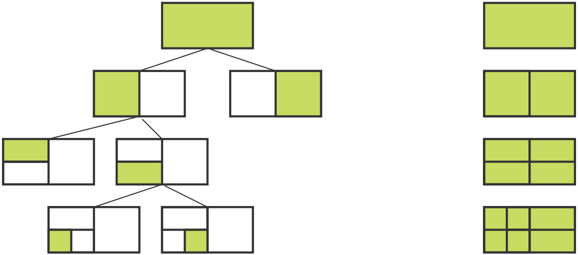


Branching vs reformulation



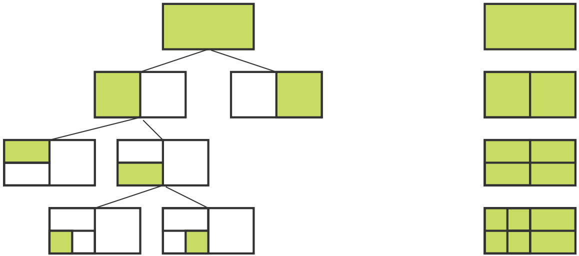
- **Branching:** *n* convex subproblems (the subproblem with the green domains are solved in each node)
- **Reformulation:** one convex MINLP problem, no branching strategy (the whole domain is considered in each iteration)

Branching vs reformulation



- **Branching:** *n* convex subproblems (the subproblem with the green domains are solved in each node)

Branching vs reformulation



- **Branching:** *n* convex subproblems (the subproblem with the green domains are solved in each node)
- **Reformulation:** one convex MINLP problem, no branching strategy (the whole domain is considered in each iteration)

Including the αBB underestimator into the framework

- In the SGO algorithm (for signomial constraints) additional binary and continuous variables are used (in the PLFs) to partition the domain of each variable.

Including the αBB underestimator into the framework

- In the SGO algorithm (for signomial constraints) additional binary and continuous variables are used (in the PLFs) to partition the domain of each variable.
- In the extended algorithm (for C^2-constraints), a convex quadratic function αx^2 is added and in addition a variable $\widehat{W}$ subtracted to be able to reformulate in the subdomains without branching, *i.e.,*

$$\underbrace{h(x) + \alpha x^2 - \widehat{W}}_{\text{convex}} \leq 0.$$

Including the αBB underestimator into the framework

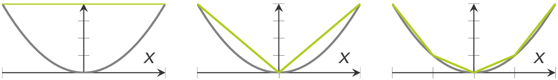
- In the SGO algorithm (for signomial constraints) additional binary and continuous variables are used (in the PLFs) to partition the domain of each variable.
- In the extended algorithm (for C^2-constraints), a convex quadratic function αx^2 is added and in addition a variable $\widehat{W}$ subtracted to be able to reformulate in the subdomains without branching, *i.e.,*

$$\underbrace{h(x) + \alpha x^2 - \widehat{W}}_{\text{convex}} \leq 0.$$

- The function αx^2 must be convex enough to overpower any nonconvexities in the variable range.

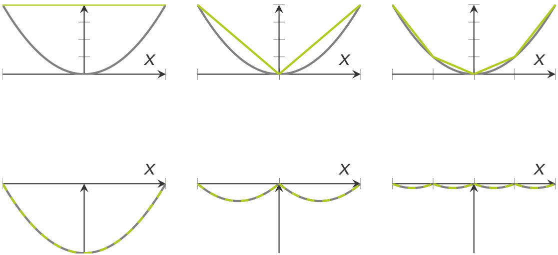
Convex reformulation of the constraints

$$h(x) + \alpha x^2 - \widehat{W} \leq 0$$



Convex reformulation of the constraints

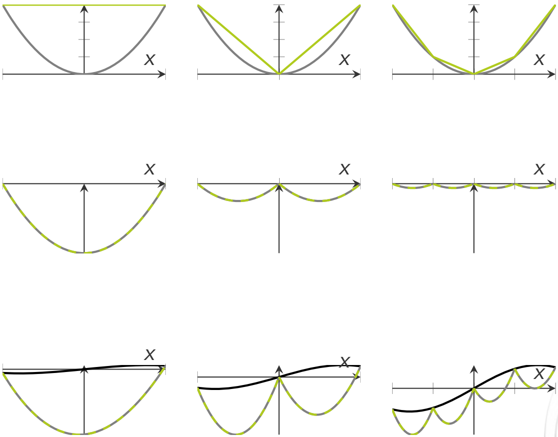
$$h(x) + \alpha x^2 - \widehat{W} \leq 0$$



- If α in αx^2 is large enough then $h(x) + \alpha x^2 - \widehat{W}$ will be convex.
- If $\widehat{W}$ is a PLF of αx^2 then $h(x)$ is underestimated since $\alpha x^2 - \widehat{W} \leq 0$.

Convex reformulation of the constraints

$$h(x) + \alpha x^2 - \widehat{W} \leq 0$$



Pros and cons for the reformulation technique

+ No explicit branching is needed in the reformulation framework.

Pros and cons for the reformulation technique

+ No explicit branching is needed in the reformulation framework.
+ The nonconvex problem can be solved simply as a sequence of convex problems in the reformulation framework.

Pros and cons for the reformulation technique

+ No explicit branching is needed in the reformulation framework.
+ The nonconvex problem can be solved simply as a sequence of convex problems in the reformulation framework.

– If the αBB underestimator is applied in this form, the same α-value for each variable must be used in all the different intervals in the reformulation framework. For example:

$$\alpha = \max_k \alpha_k,$$

where α_k is the minimal α-value needed for convexification in the k-th breakpoint interval of the PLFs.

Pros and cons for the reformulation technique

+ No explicit branching is needed in the reformulation framework.
+ The nonconvex problem can be solved simply as a sequence of convex problems in the reformulation framework.

– If the αBB underestimator is applied in this form, the same α-value for each variable must be used in all the different intervals in the reformulation framework. For example:

$$\alpha = \max_k \alpha_k,$$

where α_k is the minimal α-value needed for convexification in the k-th breakpoint interval of the PLFs.

– If the αBB underestimator is used in this form in the reformulation framework, α cannot be zero in intervals where the function is convex.

The spline αBB underestimator

- In order to overcome the drawbacks in the reformulation framework when using the traditional αBB underestimator, **a quadratic spline underestimator** can be applied. Then:

The spline αBB underestimator

- In order to overcome the drawbacks in the reformulation framework when using the traditional αBB underestimator, **a quadratic spline underestimator** can be applied. Then:
  - different α-values in different subintervals can be used

The spline αBB underestimator

- In order to overcome the drawbacks in the reformulation framework when using the traditional αBB underestimator, **a quadratic spline underestimator** can be applied. Then:
  - different α-values in different subintervals can be used
  - α can be zero in intervals where the function is convex

The spline αBB underestimator

- In order to overcome the drawbacks in the reformulation framework when using the traditional αBB underestimator, **a quadratic spline underestimator** can be applied. Then:
  - different α-values in different subintervals can be used
  - α can be zero in intervals where the function is convex

- The spline αBB underestimator is a smooth (continuously differentiable) piecewise quadratic convex function.

The spline αBB underestimator

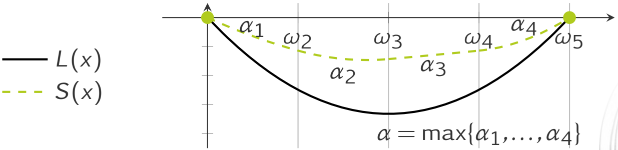
- In order to overcome the drawbacks in the reformulation framework when using the traditional αBB underestimator, **a quadratic spline underestimator** can be applied. Then:
  - different α-values in different subintervals can be used
  - α can be zero in intervals where the function is convex
- The spline αBB underestimator is a smooth (continuously differentiable) piecewise quadratic convex function.
- Originally the spline underestimator was developed for improving the αBB underestimators in subsubdomains (Meyer and Floudas, 2005).
  - This property of the spline can be utilized in the reformulation framework as well.

The spline αBB underestimator

- The spline αBB-underestimator is a smooth convex piecewise polynomial expression

$$S(x) = \begin{cases} \alpha_1 x^2 + \beta_1 x + \gamma_1 & \text{if } x \in [\omega_1, \omega_2] \\ \alpha_2 x^2 + \beta_2 x + \gamma_2 & \text{if } x \in [\omega_2, \omega_3] \\ \vdots & \vdots \\ \alpha_{K-1} x^2 + \beta_{K-1} x + \gamma_{K-1} & \text{if } x \in [\omega_{K-1}, \omega_K], \end{cases}$$

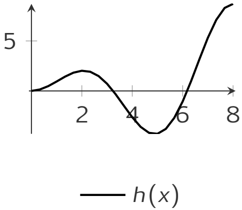
- The α_k's ensure convexity. The β_k and γ_k for $k \in \{2, \ldots, K-1\}$ ensure smoothness and continuity, and β_1, γ_1 gives $S(\omega_1) = S(\omega_K) = 0$.



An illustrative example

- Consider now the same function as before

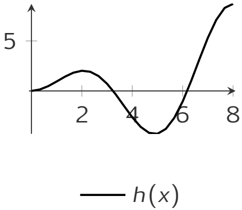
$$h(x) = x \cdot \sin x + x/10.$$



An illustrative example

- Consider now the same function as before

$$h(x) = x \cdot \sin x + x/10.$$



- The convex underestimators are then

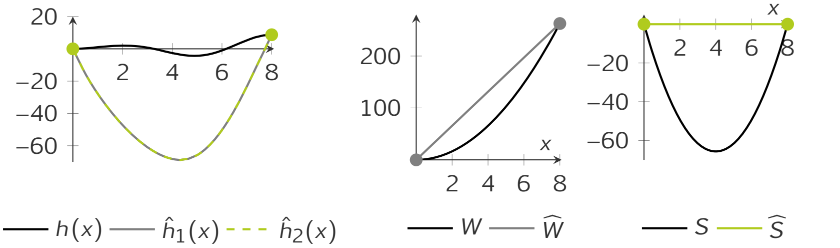
$$\hat{h}_1(x) = x \cdot \sin x + x/10 + \alpha x^2 - \widehat{W}$$

for the reformulated original αBB understimator and

$$\hat{h}_2(x) = x \cdot \sin x + x/10 + S(x) - \widehat{S}$$

for the reformulated spline αBB underestimator, where $\widehat{W}$ is the PLF of $W = \alpha x^2$ and $\widehat{S}$ is the PLF of the spline function $S(x)$.

An illustrative example

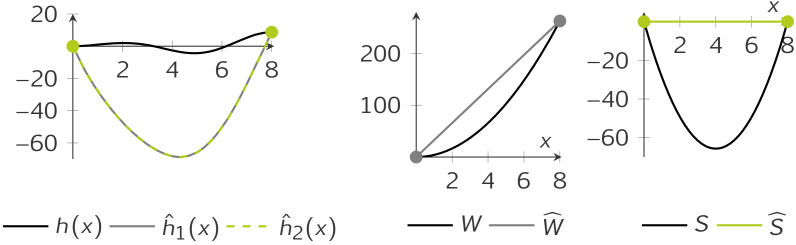


$$\hat{h}_1(x) = x \cdot \sin x + x/10 + \boxed{\alpha x^2 - \widehat{W}} \qquad \alpha = 4.10293$$

$$\hat{h}_2(x) = x \cdot \sin x + x/10 + \boxed{S(x) - \widehat{S}}$$

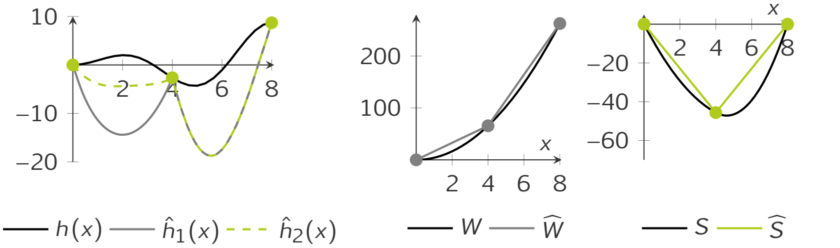
$\widehat{W}$ is the PLF of $W = \alpha x^2$ and $\widehat{S}$ is the PLF of $S(x)$

An illustrative example



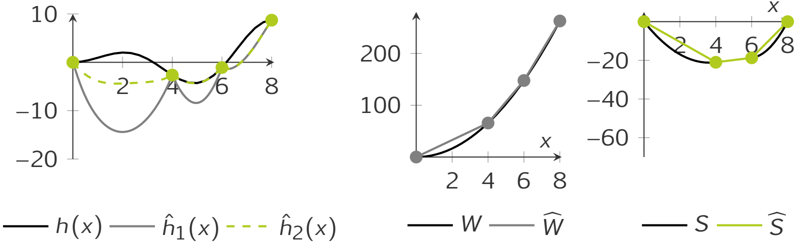
$$W(x) = 4.1x^2 \qquad S(x) = 4.1x^2 - 32.8x, 0 \leq x \leq 8$$

An illustrative example



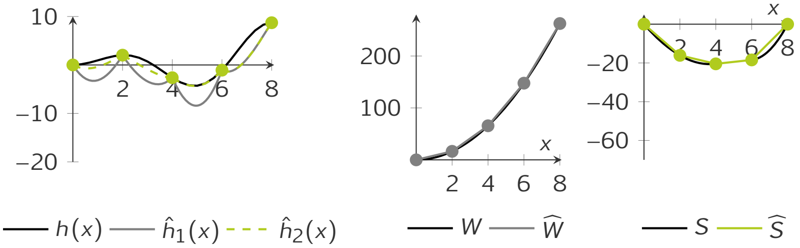
$$W(x) = 4.1x^2 \qquad S(x) = \begin{cases} 1.6x^2 - 17.8x & 0 \leq x \leq 4 \\ 4.1x^2 - 37.8x + 40.0 & 4 \leq x \leq 8 \end{cases}$$

An illustrative example



$$W(x) = 4.1x^2 \qquad S(x) = \begin{cases} 1.6x^2 - 11.7x & 0 \le x \le 4 \\ 1.1x - 25.6 & 4 \le x \le 6 \\ 4.1x^2 - 48.1x + 122.1 & 6 \le x \le 8 \end{cases}$$

An illustrative example



$$W(x) = 4.1x^2 \qquad S(x) = \begin{cases} 1.3x^2 - 10.7x & 0 \leq x \leq 2 \\ 1.6x^2 - 11.8x + 1.1 & 2 \leq x \leq 4 \\ 1.1x - 24.5 & 4 \leq x \leq 6 \\ 4.1x^2 - 48.2x + 123.2 & 6 \leq x \leq 8 \end{cases}$$

Generalization to N dimensions

- The formulation can easily be extended from one to N dimensions by using the underestimators

$$h(\mathbf{x}) + \sum_{i=1}^{N} \left(\alpha_i x_i^2 - \widehat{W}_i \right) \leq 0, \quad \mathbf{x} = (x_1, x_2, \ldots, x_N), \quad \text{or}$$

$$h(\mathbf{x}) + \sum_{i=1}^{N} \left(S_i(x_i) - \widehat{S}_i \right) \leq 0, \quad \mathbf{x} = (x_1, x_2, \ldots, x_N).$$

when using the reformulated versions of the original αBB and spline αBB underestimators respectively.

Generalization to N dimensions

- The formulation can easily be extended from one to N dimensions by using the underestimators

$$h(\mathbf{x}) + \sum_{i=1}^{N} \left(\alpha_i x_i^2 - \widehat{W}_i \right) \leq 0, \quad \mathbf{x} = (x_1, x_2, \ldots, x_N), \quad \text{or}$$

$$h(\mathbf{x}) + \sum_{i=1}^{N} \left(S_i(x_i) - \widehat{S}_i \right) \leq 0, \quad \mathbf{x} = (x_1, x_2, \ldots, x_N).$$

  when using the reformulated versions of the original αBB and spline αBB underestimators respectively.

- Here $\widehat{W}_i$ is the PLF of $W_i = \alpha_i x_i^2$ and $\widehat{S}_i$ is the PLF of S_i.

Implementation in a global optimization algorithm

- The underestimator is implemented in the algorithm αGO for solving nonconvex MINLP problems with C^2-constraints

Implementation in a global optimization algorithm

- The underestimator is implemented in the algorithm αGO for solving nonconvex MINLP problems with C^2-constraints
    - Uses the same framework as the SGO algorithm.

Implementation in a global optimization algorithm

- The underestimator is implemented in the algorithm αGO for solving nonconvex MINLP problems with C^2-constraints
  - Uses the same framework as the SGO algorithm.
  - A mixed version αSGO is also possible (power and exponential transformations are used for signomials).

Implementation in a global optimization algorithm

- The underestimator is implemented in the algorithm αGO for solving nonconvex MINLP problems with C^2-constraints
  - Uses the same framework as the SGO algorithm.
  - A mixed version αSGO is also possible (power and exponential transformations are used for signomials).
- A sequence of overestimated convex MINLP problems is solved until the solution fulfills the constraints in the original nonconvex problem.

Implementation in a global optimization algorithm

- The underestimator is implemented in the algorithm αGO for solving nonconvex MINLP problems with C^2-constraints
  - Uses the same framework as the SGO algorithm.
  - A mixed version αSGO is also possible (power and exponential transformations are used for signomials).
- A sequence of overestimated convex MINLP problems is solved until the solution fulfills the constraints in the original nonconvex problem.
- The overestimated feasible region of the convexified problems (in the original variables) is reduced in each iteration by improving the PLFs of $W = \alpha_i x_i^2$ or $S(x)$.

The original nonconvex MINLP problem

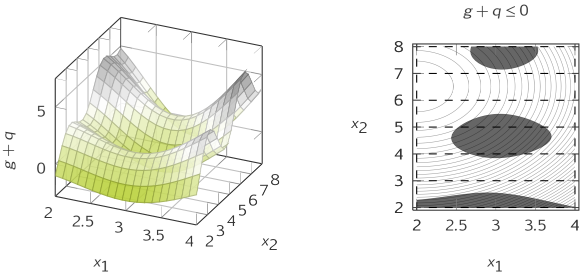
$$\begin{aligned} \text{minimize} \quad & f(x_1, x_2) = (2x_1 - 4)^2 + (x_2 - 13/2)^2 \\ \text{subject to} \quad & \underbrace{x_1 \cos^2 x_2 + x_2 \sin^2 x_1 - 3/x_2}_{g(x_1,x_2)} + \underbrace{x_1/2 - 5/2}_{q(x_1)} \leq 0, \\ & 2 \leq x_1 \leq 4, \quad 2 \leq x_2 \leq 8, \quad x_1 \in \mathbb{R}, \quad x_2 \in \mathbb{Z}. \end{aligned}$$

The original nonconvex MINLP problem

minimize $f(x_1, x_2) = (2x_1 - 4)^2 + (x_2 - 13/2)^2$

subject to $\underbrace{x_1 \cos^2 x_2 + x_2 \sin^2 x_1 - 3/x_2}_{g(x_1,x_2)} + \underbrace{x_1/2 - 5/2}_{q(x_1)} \leq 0,$

$2 \leq x_1 \leq 4, \quad 2 \leq x_2 \leq 8, \quad x_1 \in \mathbb{R}, \quad x_2 \in \mathbb{Z}.$



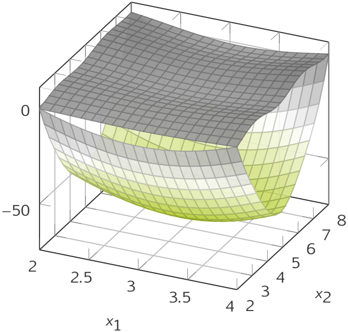
The reformulated MINLP problem

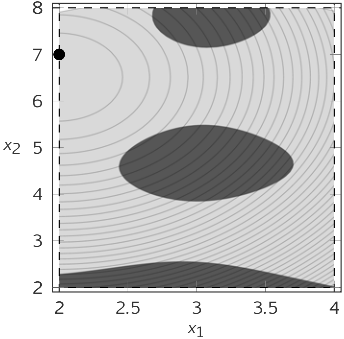
$$\begin{aligned} &\text{minimize} && f(x_1,x_2) = (2x_1-4)^2 + (x_2-13/2)^2 \\ &\text{subject to} && x_1\cos^2 x_2 + x_2 \sin^2 x_1 - 3/x_2 + x_1/2 - 5/2 \\ & && \qquad + S_1(x_1) + S_2(x_2) - \widehat{S}_1 - \widehat{S}_2 \leq 0, \\ & && \widehat{S}_1 = \mathrm{PLF}(S_1(x_2), V_1;\, \Omega_1),\ \widehat{S}_2 = \mathrm{PLF}(S_2(x_2), V_2;\, \Omega_2), \\ & && 2 \leq x_1 \leq 4, \quad 2 \leq x_2 \leq 8, \quad x_1 \in \mathbb{R}, \quad x_2 \in \mathbb{Z}, \end{aligned}$$

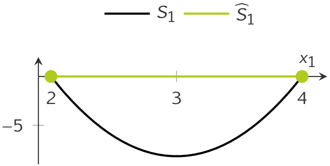
V_i and Ω_i are sets including the variables and breakpoints in PLF_i of $S_i(x_1)$

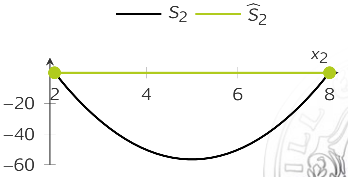
- This reformulated problem is convex in the extended variable space consisting of the original variables x_1 and x_2, as well as, those needed for the PLFs in V_1 and V_2.

αGO iteration 1

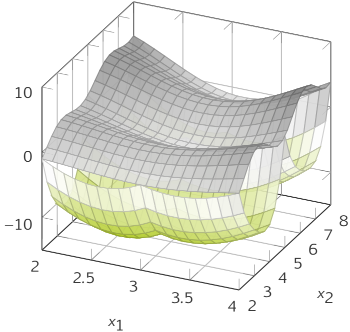


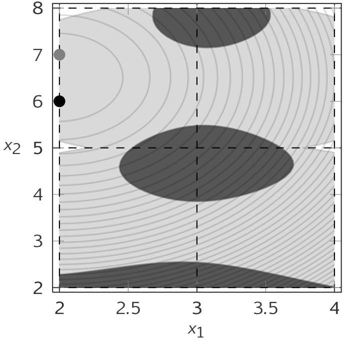


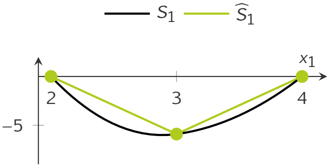


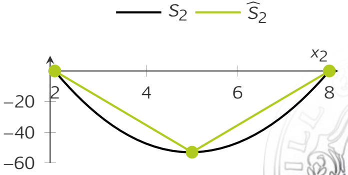


αGO iteration 2

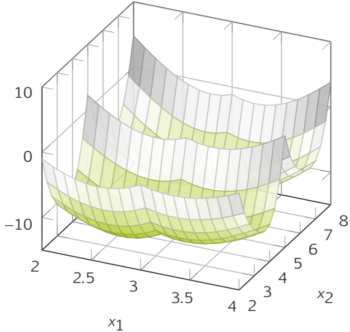


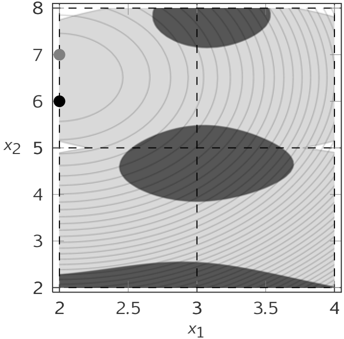


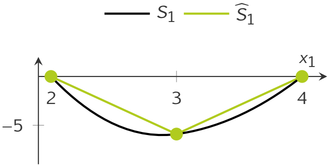


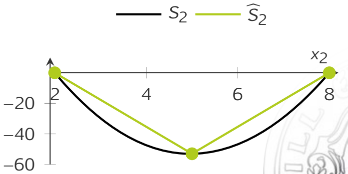


αGO iteration 2

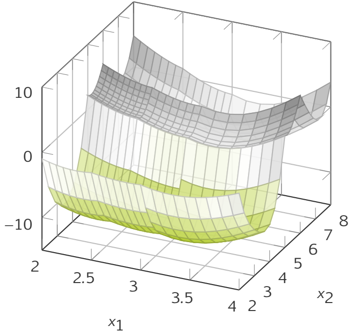


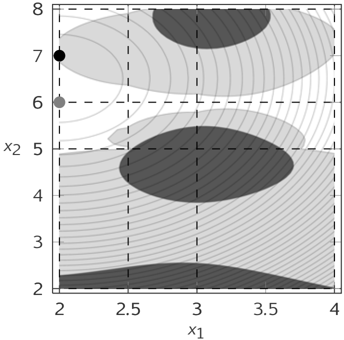


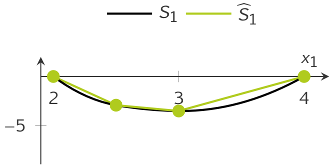


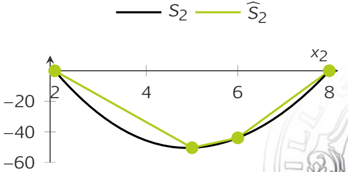


αGO iteration 3

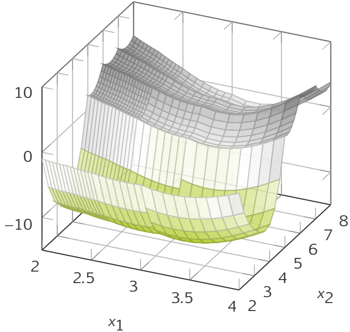


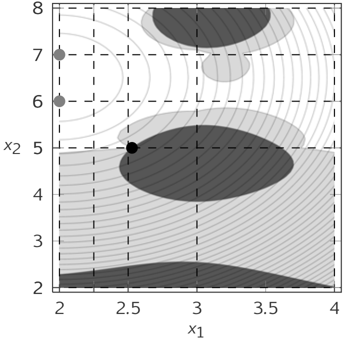


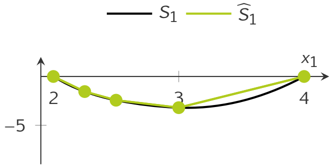


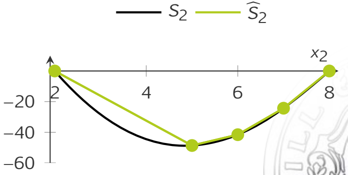


αGO iteration 4

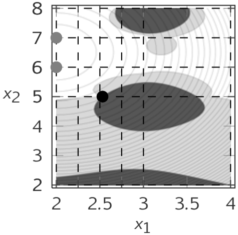


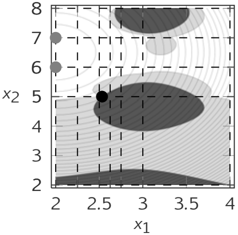






αGO iteration 5 and 6





Iter.	Regions	$f(x_1, x_2)$	x_1	x_2	$g(x_1, x_2) + q(x_1)$
1	1	0.2500	2.0	7	4.9959
2	4	0.2500	2.0	6	4.9959
3	9	0.2500	2.0	7	4.9959
4	16	3.3630	2.52749	5	0.0273
5	20	3.3767	2.53074	5	0.0139
6	24	3.3848	2.53263	5	0.0061

On-going and future work

- Exclusion of infeasible regions by, *e.g.,* interval arithmetic and integer cuts

On-going and future work

- Exclusion of infeasible regions by, *e.g.,* interval arithmetic and integer cuts
- Improving the splines by calculating them on smaller intervals

On-going and future work

- Exclusion of infeasible regions by, *e.g.,* interval arithmetic and integer cuts
- Improving the splines by calculating them on smaller intervals
- Numerical studies

Summary

- αGO/SGO is a global optimization algorithm for nonconvex twice-differentiable MINLP problems.
  - The αBB underestimator is used to handle the nonconvex C^2-functions.

Summary

- αGO/SGO is a global optimization algorithm for nonconvex twice-differentiable MINLP problems.
  - The αBB underestimator is used to handle the nonconvex C^2-functions.

- In the algorithm no direct branching strategy is needed for handling the nonconvexities, instead a reformulated and convexified MINLP problem is solved iteratively until the global solution is found.

Summary

- αGO/SGO is a global optimization algorithm for nonconvex twice-differentiable MINLP problems.
  - The αBB underestimator is used to handle the nonconvex C^2-functions.
- In the algorithm no direct branching strategy is needed for handling the nonconvexities, instead a reformulated and convexified MINLP problem is solved iteratively until the global solution is found.
- Easy to implement, since only a method for calculating the α-values and a convex MINLP solver is needed.

Some references

A. Lundell, A. Skjäl, and T. Westerlund.
A reformulation framework for global optimization.
Journal of Global Optimization, submitted, 2011.

A. Lundell, J. Westerlund, and T. Westerlund.
Some transformation techniques with applications in global optimization.
Journal of Global Optimization, 43(2):391–405, 2009.

C. A. Floudas.
Deterministic Global Optimization. Theory, Methods and Applications.
Number 37 in Nonconvex Optimization and Its Applications. Kluwer Academic Publishers, 2000.

C. A. Meyer and C. A. Floudas.
Convex underestimation of twice continuously differentiable functions by piecewise quadratic perturbation: Spline αBB underestimators.
J. Glob. Optim., 32(2):221–258, 2005.

The end of the presentation

Thank you for listening!

The end of the presentation

Thank you for listening!

Questions?