Solving linearly constrained nonlinear minimax problems using cutting plane techniques

MSc Jan Kronqvist PhD Andreas Lundell

Prof. Tapio Westerlund

PROCESS DESIGN AND SYSTEMS ENGINEERING LABORATORY

OPTIMIZATION AND SYSTEMS ENGINEERING

Åbo Akademi University

Contents of the talk

- Brief introduction to the Extended Supporting Hyperplane (ESH) algorithm.

Contents of the talk

- Brief introduction to the Extended Supporting Hyperplane (ESH) algorithm.
  - Subproblem in the ESH algorithm.

Contents of the talk

- Brief introduction to the Extended Supporting Hyperplane (ESH) algorithm.
  - Subproblem in the ESH algorithm.
- This subproblem can be solved as a continuous nonlinear minimax problem.

Contents of the talk

- Brief introduction to the Extended Supporting Hyperplane (ESH) algorithm.
  - Subproblem in the ESH algorithm.
- This subproblem can be solved as a continuous nonlinear minimax problem.
- How to solve linearly constrained nonlinear minimax problems by cutting plane techniques.

Contents of the talk

- Brief introduction to the Extended Supporting Hyperplane (ESH) algorithm.
  - Subproblem in the ESH algorithm.
- This subproblem can be solved as a continuous nonlinear minimax problem.
- How to solve linearly constrained nonlinear minimax problems by cutting plane techniques.
- Numerical comparison of different methods for a test set of minimax problems.

- ESH is an algorithm intended for solving convex Mixed-Integer Nonlinear Programming (MINLP) optimization problems.[1]

[1] The extended supporting hyperplane algorithm for convex mixed-integer nonlinear programming, Kronqvist, J., Lundell, A., Westerlund, T., Journal of Global Optimization (2015), Accepted

- ESH is an algorithm intended for solving convex Mixed-Integer Nonlinear Programming (MINLP) optimization problems.[1]
- A subproblem in the ESH algorithm can be solved as a linearly constrained nonlinear minimax problem.
  - Finding a suitable interior point of a convex set.

[1] The extended supporting hyperplane algorithm for convex mixed-integer nonlinear programming, Kronqvist, J., Lundell, A., Westerlund, T., Journal of Global Optimization (2015), Accepted

- The MINLP problem

$$\text{find } x^* \in \underset{x \in L \cap C \cap Y}{\arg\min}\, c^T x, \tag{P-MINLP}$$

$$\begin{aligned}
X &= \left\{x \,\middle|\, \underline{x}_i \le x_i \le \overline{x}_i,\ i = 1, \ldots, N\ , x \in \mathbb{R}^n \right\}, \\
L &= \{x \,|\, Ax \le a,\ Bx = b,\ x \in X\}, \\
C &= \{x \,|\, g_m(x) \le 0,\ m = 1, \ldots, M,\ x \in X\}, \\
Y &= \{x \,|\, x_i \in \mathbb{Z},\ \forall i \in I_{\mathbb{Z}},\ x \in X\}.
\end{aligned}$$

- The MINLP problem

$$\text{find } x^* \in \underset{x \in L \cap C \cap Y}{\arg\min}\, c^T x, \qquad \text{(P-MINLP)}$$

$$\begin{aligned}
X &= \left\{x \,\middle|\, \underline{x}_i \leq x_i \leq \overline{x}_i,\ i = 1, \ldots, N,\ x \in \mathbb{R}^n \right\}, \\
L &= \{x \,|\, Ax \leq a,\ Bx = b,\ x \in X\}, \\
C &= \{x \,|\, g_m(x) \leq 0,\ m = 1, \ldots, M,\ x \in X\}, \\
Y &= \{x \,|\, x_i \in \mathbb{Z},\ \forall i \in I_{\mathbb{Z}},\ x \in X\}.
\end{aligned}$$

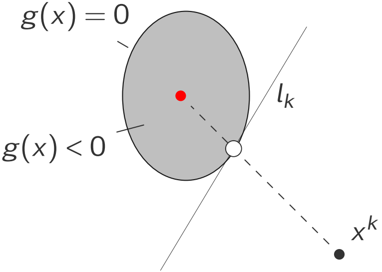
- The ESH algorithm solves (P-MINLP) by solving a sequence of linearly relaxed subproblems.

- In each iteration the objective function is minimized within a linearly overestimated set Ω of the feasible region defined by $L \cap C \cap Y$

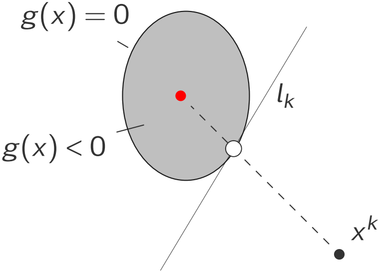
- In each iteration the objective function is minimized within a linearly overestimated set Ω of the feasible region defined by $L \cap C \cap Y$
- In case the current solution point x_k is not within the set C (the nonlinear constraints are not satisfied).
  - A supporting hyperplane to the set C is generated and added to set Ω.
  - The supporting hyperplane improves the linear relaxation of (P-MINLP) and excludes the current solution point x_k from Ω.

- In each iteration the objective function is minimized within a linearly overestimated set Ω of the feasible region defined by $L \cap C \cap Y$
- In case the current solution point x_k is not within the set C (the nonlinear constraints are not satisfied).
  - A supporting hyperplane to the set C is generated and added to set Ω.
  - The supporting hyperplane improves the linear relaxation of (P-MINLP) and excludes the current solution point x_k from Ω.
- In order to find the generation point for the supporting hyperplane a point within the interior of C is needed.
  - This point is used for line searches for the boundary of the set C

- A sketch of the main principle of the ESH algorithm, a line search is conducted between the interior point and the current solution x^k and a supporting hyperplane is generated.



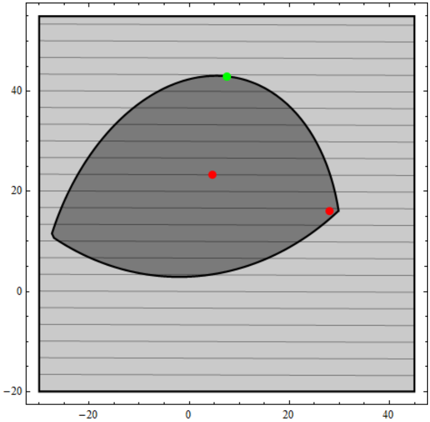
- A sketch of the main principle of the ESH algorithm, a line search is conducted between the interior point and the current solution x^k and a supporting hyperplane is generated.



- How does the choice of interior point affect the ESH algorithm?
- How to obtain an interior point efficiently?

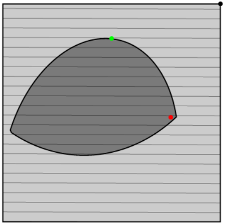
- To illustrate how the choice of interior point affects the ESH algorithm, consider the following MINLP problem:

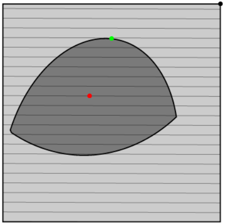
$$\begin{aligned}
\text{minimize} \quad & -0.05x_1 - 15x_2 \\
\text{subject to} \quad & g_1(x_1,x_2) = 0.1x_1^2 + 0.05x_2^2 - 0.025x_1x_2 - 90 \le 0 \\
& g_2(x_1,x_2) = 0.5x_1^2 + 0.35x_2^2 + 2x_1 - 45x_2 + 130 \le 0 \\
& -30 \le x_1 \le 45, -20 \le x_2 \le 55, \\
& x_1 \in \mathbb{R}, \quad x_2 \in \mathbb{Z}.
\end{aligned} \tag{EX1}$$



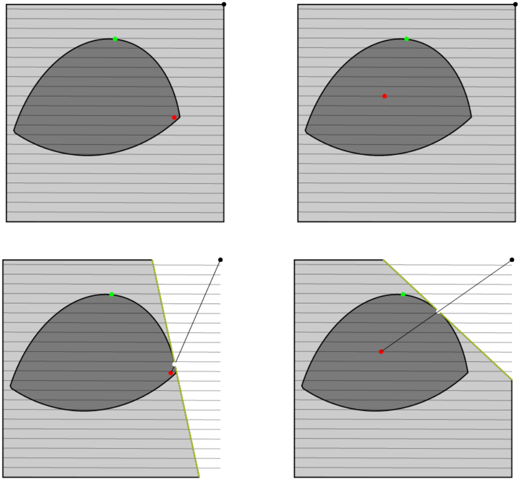
Contours of the objective function, the feasible region defined by the nonlinear constraints and the region defined by the linear constraints (variable bounds).

The figures shows the first iteration of the ESH algorithm with two different interior points

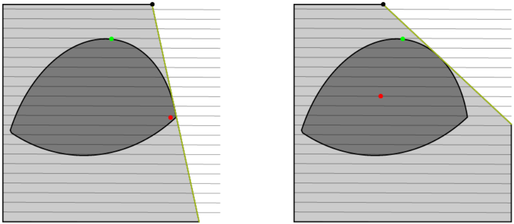




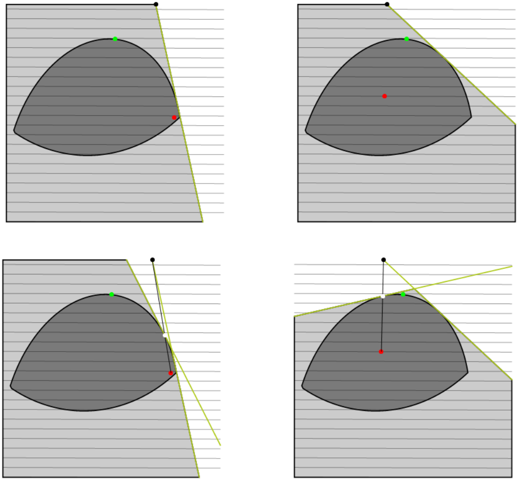
The figures shows the first iteration of the ESH algorithm with two different interior points



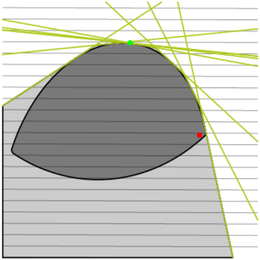
The figures shows the second iteration of the ESH algorithm with two different interior points



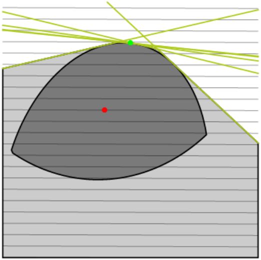
The figures shows the second iteration of the ESH algorithm with two different interior points



- Depending on the interior point it either takes 9 or 6 iterations with ESH algorithm to solve the MINLP problem



9 iterations



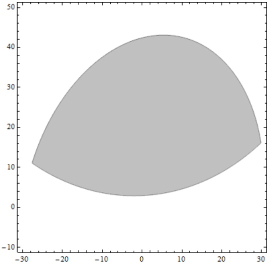
6 iterations

- Desired properties of the interior point:

- Desired properties of the interior point:
  - Force the relaxed solution into the feasible set with as few supporting hyperplanes as possible.
  - Easy to obtain!

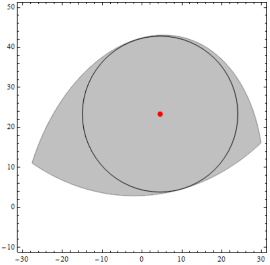
- Desired properties of the interior point:
  - Force the relaxed solution into the feasible set with as few supporting hyperplanes as possible.
  - Easy to obtain!
- What about the Chebyschev center?

- A Chebychev center of the convex set C is a point inside the set farthest from the exterior of C, *i.e.*, the deepest point of the set C. [2]
  - Lets consider a case where the set C is defined as
    $C = \{x \mid g_1(x) \leq 0,\ g_2(x) \leq 0\}$
    $g_1(x) = 0.1x_1^2 + 0.05x_2^2 - 0.025x_1x_2 - 90$
    $g_2(x) = 0.5x_1^2 + 0.35x_2^2 + 2x_1 - 45x_2 + 130$



2 Convex Optimization, Boyd, S., and Vandenberghe, L., Cambridge University Press

- A Chebychev center of the convex set C is a point inside the set farthest from the exterior of C, *i.e.,* the deepest point of the set C. [3]
  - Lets consider a case where the set C is defined as
    $C = \{x \mid g_1(x) \leq 0,\ g_2(x) \leq 0\}$
    $g_1(x) = 0.1x_1^2 + 0.05x_2^2 - 0.025x_1x_2 - 90$
    $g_2(x) = 0.5x_1^2 + 0.35x_2^2 + 2x_1 - 45x_2 + 130$



3 Convex Optimization, Boyd, S., and Vandenberghe, L., Cambridge University Press

- The Chebyshev center of the set $C = \{x \mid g_m(x) \leq 0 \quad \forall m,\ x \in \mathbb{R}^n\}$ can be obtained by solving the following convex optimization problem [4]

[4] Convex Optimization, Boyd, S., and Vandenberghe, L., Cambridge University Press

- The Chebyshev center of the set $C = \{x \mid g_m(x) \leq 0 \quad \forall m,\ x \in \mathbb{R}^n\}$ can be obtained by solving the following convex optimization problem [4]

$$
\begin{aligned}
&\text{maximize} && R \\
&\text{subject to} && \tilde{g}_m(x,R) \leq 0 \quad \forall m \\
&\text{where} && \\
& && \tilde{g}_m(x,R) := \sup_{\|u\| \leq 1} g_m(x + Ru) \\
& && u \in \mathbb{R}^n.
\end{aligned}
\tag{1}
$$

[4] Convex Optimization, Boyd, S., and Vandenberghe, L., Cambridge University Press

- The Chebyshev center of the set $C = \{x \mid g_m(x) \leq 0 \quad \forall m,\ x \in \mathbb{R}^n\}$ can be obtained by solving the following convex optimization problem [4]

$$\begin{aligned} &\text{maximize} && R \\ &\text{subject to} && \tilde{g}_m(x,R) \leq 0 \quad \forall m \\ &\text{where} && \\ & && \tilde{g}_m(x,R) := \sup_{\|u\| \leq 1} g_m(x + Ru) \\ & && u \in \mathbb{R}^n. \end{aligned} \tag{1}$$

- However, even if this is a convex NLP problem it is difficult to solve unless the constraint functions $\tilde{g}_m$ can be determined analytically.

[4] Convex Optimization, Boyd, S., and Vandenberghe, L., Cambridge University Press

- The Chebyshev center of the set $C = \{x \mid g_m(x) \leq 0 \quad \forall m,\ x \in \mathbb{R}^n\}$ can be obtained by solving the following convex optimization problem [4]

$$\begin{aligned} &\text{maximize} && R \\ &\text{subject to} && \tilde{g}_m(x,R) \leq 0 \quad \forall m \\ &\text{where} && \\ & && \tilde{g}_m(x,R) := \sup_{\|u\|\leq 1} g_m(x+Ru) \\ & && u \in \mathbb{R}^n. \end{aligned} \tag{1}$$

- However, even if this is a convex NLP problem it is difficult to solve unless the constraint functions $\tilde{g}_m$ can be determined analytically.

Finding the Chebyshev center on the previous slide required the solution of <u>266 + 1 convex NLP problems!</u>

[4] Convex Optimization, Boyd, S., and Vandenberghe, L., Cambridge University Press

- The problem of finding the Chebychev center of a convex set is generally too difficult.

- The problem of finding the Chebychev center of a convex set is generally too difficult.
- However, the only requirement for an interior point $\bar{x}$ with the ESH algorithm is:
  $max\{g_1(\bar{x}), g_2(\bar{x}), .., g_M(\bar{x})\} < 0.$

- The problem of finding the Chebychev center of a convex set is generally too difficult.
- However, the only requirement for an interior point $\bar{x}$ with the ESH algorithm is: $max\{g_1(\bar{x}), g_2(\bar{x}), .., g_M(\bar{x})\} < 0$.
- An interior point for the ESH algorithm can therefore be found by solving the following problem:

$$\text{find } \bar{x} \in \arg\min_{x \in L} F(x), \tag{2}$$

where $F(x) := max\{g_1(x), g_2(x), .., g_M(x)\}$.

- The problem of finding the Chebychev center of a convex set is generally too difficult.
- However, the only requirement for an interior point $\bar{x}$ with the ESH algorithm is: $max\{g_1(\bar{x}), g_2(\bar{x}), .., g_M(\bar{x})\} < 0$.
- An interior point for the ESH algorithm can therefore be found by solving the following problem:

$$\text{find } \bar{x} \in \arg\min_{x \in L} F(x), \tag{2}$$

where $F(x) := max\{g_1(x), g_2(x), .., g_M(x)\}$.

- The linearly constrained minimax problem

$$\text{find } \bar{x} \in \arg\min_{x \in L} F(x), \tag{3}$$

$$\begin{aligned} F(x) &:= \max\{g_1(x), g_2(x), .., g_M(x)\}, \\ X &= \left\{x \,\middle|\, \underline{x}_i \leq x_i \leq \overline{x}_i,\ i = 1, \ldots, N\ , x \in \mathbb{R}^n \right\}, \\ L &= \{x \,|\, Ax \leq a,\ Bx = b,\ x \in X\}. \end{aligned}$$

- If all the functions g_m are convex, then the max function F is also a convex function.

- The linearly constrained minimax problem

$$\text{find } \bar{x} \in \arg\min_{x \in L} F(x), \tag{3}$$

$$\begin{aligned} F(x) &:= \max\{g_1(x), g_2(x), .., g_M(x)\}, \\ X &= \left\{x \,\middle|\, \underline{x}_i \leq x_i \leq \overline{x}_i,\ i = 1, \dots, N\ , x \in \mathbb{R}^n\right\}, \\ L &= \{x \,|\, Ax \leq a,\ Bx = b,\ x \in X\}. \end{aligned}$$

- If all the functions g_m are convex, then the max function F is also a convex function.
  - Simply a convex NLP problem!

- The linearly constrained minimax problem

$$\text{find } \bar{x} \in \arg\min_{x \in L} F(x), \tag{3}$$

$$\begin{aligned} F(x) &:= \max\{g_1(x), g_2(x), .., g_M(x)\}, \\ X &= \left\{x \,\middle|\, \underline{x}_i \le x_i \le \overline{x}_i,\ i = 1, \ldots, N, x \in \mathbb{R}^n\right\}, \\ L &= \{x \,|\, Ax \le a,\ Bx = b,\ x \in X\}. \end{aligned}$$

- If all the functions g_m are convex, then the max function F is also a convex function.
  - Simply a convex NLP problem!
- However, the function F may not be a smooth function.

- The linearly constrained minimax problem

$$\text{find } \bar{x} \in \arg\min_{x \in L} F(x), \tag{3}$$

$$\begin{aligned} F(x) &:= \max\{g_1(x), g_2(x), .., g_M(x)\}, \\ X &= \left\{x \,\middle|\, \underline{x}_i \leq x_i \leq \overline{x}_i,\ i = 1, \ldots, N, x \in \mathbb{R}^n\right\}, \\ L &= \{x \,|\, Ax \leq a,\ Bx = b,\ x \in X\}. \end{aligned}$$

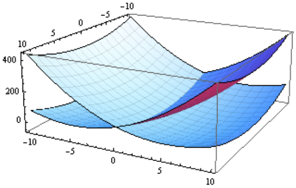
- If all the functions g_m are convex, then the max function F is also a convex function.
  - Simply a convex NLP problem!
- However, the function F may not be a smooth function.
  - In the ESH subproblem the function F is rarely a smooth function!

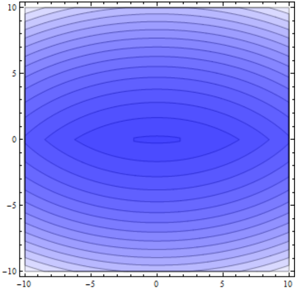
- Since F is a nonsmooth function, standard gradient based methods may fail to solve the minimax problem

- Since F is a nonsmooth function, standard gradient based methods may fail to solve the minimax problem
- To illustrate this, let's examine the following example:

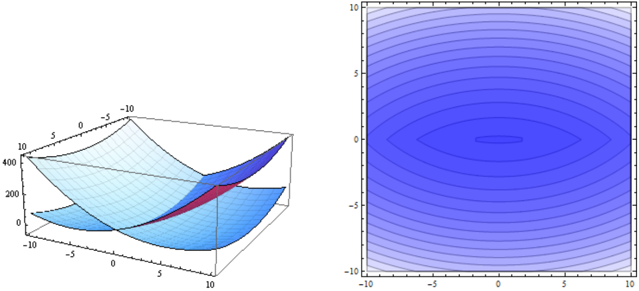
$$\text{find } \bar{x} \in \arg\min_{x \in L} F(x),$$

$$\begin{aligned} F(x) &= \max\{g_1(x), g_2(x)\} \\ g_1(x) &= 0.9x_1^2 + 1.8(x_2 - 5)^2 - 60 \\ g_2(x) &= 0.9x_1^2 + 1.8(x_2 + 5)^2 - 60 \\ L &= \{x \,|\, -10 \le x_1 \le 10,\ -10 \le x_2 \le 10\}. \end{aligned}$$

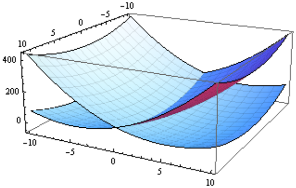


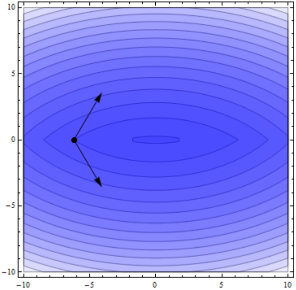


- The functions g_1, g_2 and a contour plot of the max function F.



- The functions g_1, g_2 and a contour plot of the max function F.
- Suppose we start at the point $x_1 = -6.2, x_2 = 0$





- The functions g_1, g_2 and a contour plot of the max function F.
- Suppose we start at the point $x_1 = -6.2, x_2 = 0$
- At this point F is not differentiable and neither $-\nabla g_1$ or $-\nabla g_2$ gives a descent direction

- To solve the minimax problem we can use a nonsmooth method,

- To solve the minimax problem we can use a nonsmooth method,
- or rewrite it as a smooth problem by adding an auxillary variable μ

$$\begin{aligned} \text{minimize} \quad & \mu \\ \text{subject to} \quad & g_m(x) \le \mu \quad \forall m \\ & Ax \le a \\ & Bx = b \\ & \mu \in \mathbb{R}, x \in X \end{aligned} \tag{3}$$

- To solve the minimax problem we can use a nonsmooth method,
- or rewrite it as a smooth problem by adding an auxillary variable μ

$$
\begin{aligned}
\text{minimize} \quad & \mu \\
\text{subject to} \quad & g_m(x) \leq \mu \quad \forall m \\
& Ax \leq a \\
& Bx = b \\
& \mu \in \mathbb{R}, x \in X
\end{aligned}
\tag{3}
$$

- Resulting in a standard NLP problem.

- To solve the minimax problem we can use a nonsmooth method,
- or rewrite it as a smooth problem by adding an auxillary variable μ

$$\begin{aligned} \text{minimize} \quad & \mu \\ \text{subject to} \quad & g_m(x) \leq \mu \quad \forall m \\ & Ax \leq a \\ & Bx = b \\ & \mu \in \mathbb{R}, x \in X \end{aligned} \tag{3}$$

- Resulting in a standard NLP problem.
- This gave us an idea:
  - By utilizing the special problem structure it could possible to solve the problem efficiently by a modified version of Kelley's cutting plane method.

Basic steps of Kelley's cutting plane method[5]

[5] The cutting-plane method for solving convex programs, Kelley, J. E., Journal of the Society for Industrial & Applied Mathematics 8,703-712 (1960)

Basic steps of Kelley's cutting plane method[5]

0. First define $\Omega_0 = \{x, \mu \,|x \in L, \mu_{\min} \leq \mu \leq \mu_{\max}\}$, set $k = 1$ and specify the accepted tolerance ϵ for the nonlinear constraints

[5] The cutting-plane method for solving convex programs, Kelley, J. E., Journal of the Society for Industrial & Applied Mathematics 8,703-712 (1960)

Basic steps of Kelley's cutting plane method[5]

0. First define $\Omega_0 = \{x, \mu \,| x \in L, \mu_{\min} \leq \mu \leq \mu_{\max}\}$, set $k = 1$ and specify the accepted tolerance ϵ for the nonlinear constraints

Repeat until $g_m(\hat{x}_k) - \mu_k < \epsilon \quad \forall m$
1. Find minimum of the objective within the linear set

$$[\hat{x}_k, \mu_k] \in \underset{x,\mu \in \Omega_{k-1}}{\arg\min}\, \mu$$

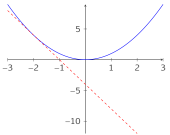
2. Generate a cutting plane for the most violated constraint at $[\hat{x}_k, \mu_k]$ and update the set Ω

$$l_k(x, \mu) = g_m(\hat{x}_k) + \nabla g_m(\hat{x}_k)^T(x - \hat{x}_k) - \mu$$

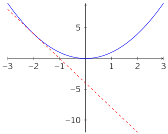
$$\Omega_k = \{x, \mu \mid l_k(x, \mu) \leq 0, \quad x, \mu \in \Omega_{k-1}\}, \quad k = k + 1$$

[5] The cutting-plane method for solving convex programs, Kelley, J. E., Journal of the Society for Industrial & Applied Mathematics 8,703-712 (1960)

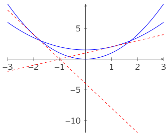
- Since the functions g_m are all convex, all linearizations $g_m(\hat{x}_k) + \nabla g_m(\hat{x}_k)^T(x - \hat{x}_k)$ underestimates the function g_m.



- Since the functions g_m are all convex, all linearizations $g_m(\hat{x}_k) + \nabla g_m(\hat{x}_k)^T(x - \hat{x}_k)$ underestimates the function g_m.

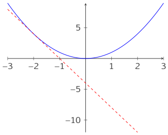


- Note that the linearizations also underestimates the max function F. [6]

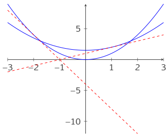


6 Rockafellar, R.T.: Convex analysis. Princeton University Press (1970)

- Since the functions g_m are all convex, all linearizations $g_m(\hat{x}_k) + \nabla g_m(\hat{x}_k)^T(x - \hat{x}_k)$ underestimates the function g_m.



- Note that the linearizations also underestimates the max function F. [6]

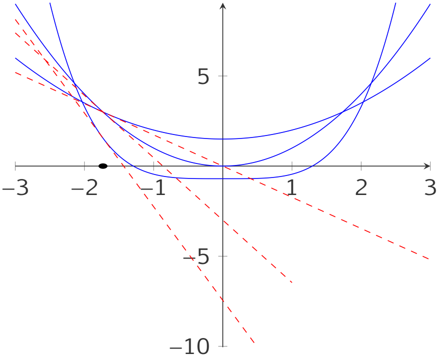


- Hence μ_k gives a lower bound of the optimal value of the maxfunction and an upper bound is given by $F(\hat{x}_k)$.
  - The optimality gap is hence given by $F(\hat{x}_k) - \mu_k$

[6] Rockafellar, R.T.: Convex analysis. Princeton University Press (1970)

- At a point $[\hat{x}_k, \mu_k]$ several of the nonlinear constraints might be violated.
  - At such a point it is possible to generate several cutting planes.
  - How to choose which cutting planes?

- At a point $[\hat{x}_k, \mu_k]$ several of the nonlinear constraints might be violated.
  - At such a point it is possible to generate several cutting planes.
  - How to choose which cutting planes?
  - Functions such that $g_m(\hat{x}_k) = F(\hat{x}_k)$ describes the function F within the neighborhood of $\hat{x}_k$.

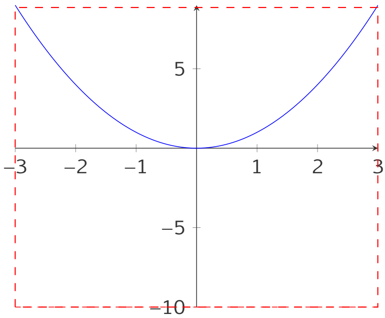


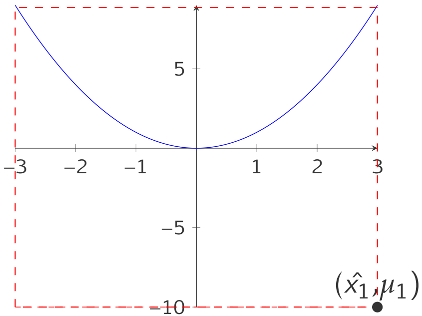
- Kelley's cutting plane method often takes to long steps and "jumps" over the optimal solution.

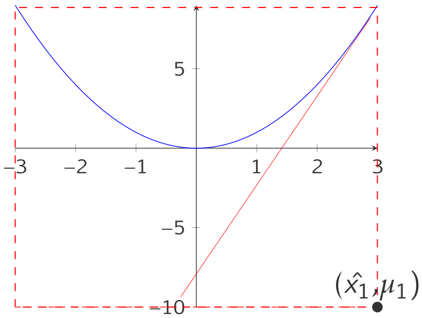
- Kelley's cutting plane method often takes to long steps and "jumps" over the optimal solution.
  - Lets consider the following example:
    minimize $F(x) = x^2, \quad x \in [-2, 2]$

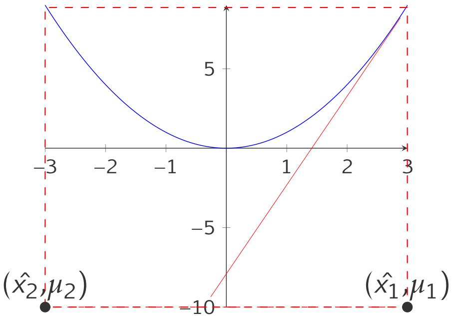
- Kelley's cutting plane method often takes to long steps and "jumps" over the optimal solution.
  - Lets consider the following example:
    minimize $F(x) = x^2, \quad x \in [-2, 2]$

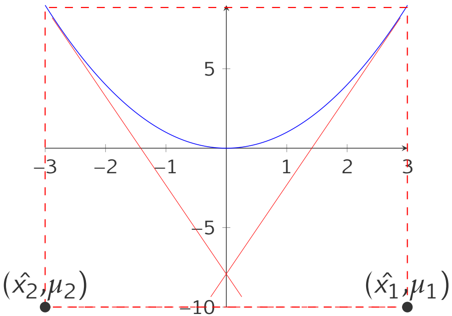
$$\begin{aligned} \text{minimize} \quad & \mu \\ \text{subject to} \quad & x^2 \le \mu \\ & -2 \le x \le 2 \\ & -10 \le \mu \le 10 \\ & x, \mu \in \mathbb{R} \end{aligned} \tag{4}$$

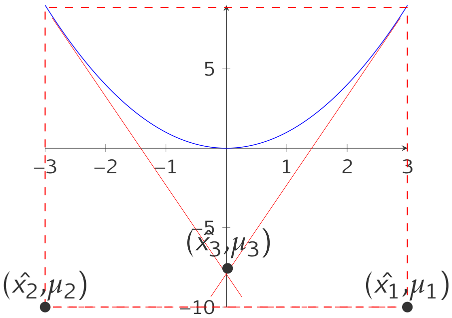

5
−3
−2
−1
1
2
3
−5
−10

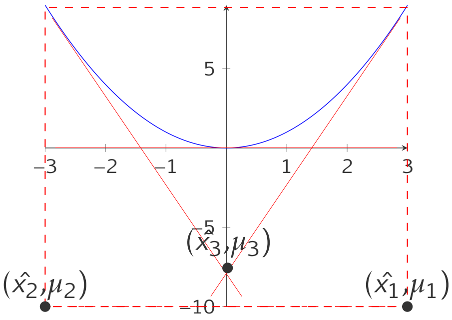

5
−5
−10
−3
−2
−1
1
2
3
$(\hat{x}_1, \mu_1)$

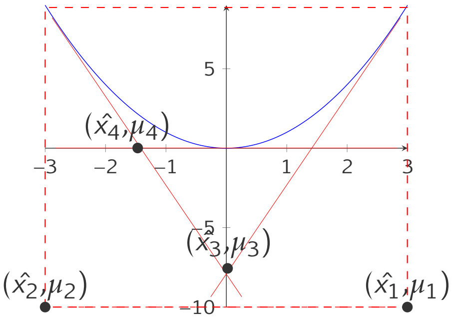

5
−5
−10
−3
−2
−1
1
2
3
$(\hat{x}_1, \mu_1)$

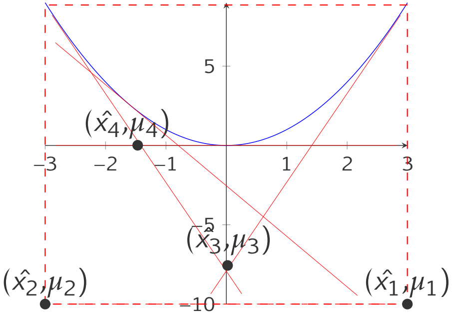

5
−5
−10
−3
−2
−1
1
2
3
$(\hat{x}_2, \mu_2)$
$(\hat{x}_1, \mu_1)$

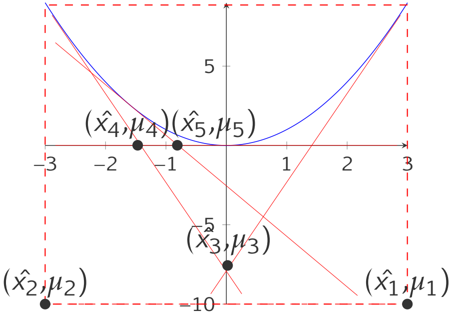

5
−3
−2
−1
1
2
3
−5
−10
$(\hat{x}_2, \mu_2)$
$(\hat{x}_1, \mu_1)$

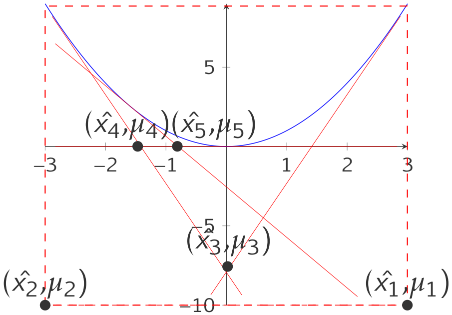

5
−5
−10
−3
−2
−1
1
2
3
$(\hat{x}_3, \mu_3)$
$(\hat{x}_2, \mu_2)$
$(\hat{x}_1, \mu_1)$


5
−5
−10
−3
−2
−1
1
2
3
$(\hat{x}_3, \mu_3)$
$(\hat{x}_2, \mu_2)$
$(\hat{x}_1, \mu_1)$


5
−5
−10
−3
−2
−1
1
2
3
$(\hat{x}_4, \mu_4)$
$(\hat{x}_3, \mu_3)$
$(\hat{x}_2, \mu_2)$
$(\hat{x}_1, \mu_1)$


5
−5
−10
−3
−2
−1
1
2
3
$(\hat{x}_4, \mu_4)$
$(\hat{x}_3, \mu_3)$
$(\hat{x}_2, \mu_2)$
$(\hat{x}_1, \mu_1)$


5
$(\hat{x}_4, \mu_4)$
$(\hat{x}_5, \mu_5)$
−3
−2
−1
1
2
3
−5
$(\hat{x}_3, \mu_3)$
$(\hat{x}_2, \mu_2)$
$(\hat{x}_1, \mu_1)$
−10



- However, by taking advantage of the problem structure we can stop in the 4:th iteration and verify that the optimal solution was found already in iteration 3
  - $F(x_3) - \mu_4 = 0$.

- In case cutting planes are generated for all function g_m such that $g_m(\hat{x}_{k-1}) = F(\hat{x}_{k-1})$ and $\hat{x}_{k-1}$ is not the optimal solution, it can easily be shown $(\hat{x}_k - \hat{x}_{k-1})$ gives a descent direction for the max function at $\hat{x}_{k-1}$.

- In case cutting planes are generated for all function g_m such that $g_m(\hat{x}_{k-1}) = F(\hat{x}_{k-1})$ and $\hat{x}_{k-1}$ is not the optimal solution, it can easily be shown $(\hat{x}_k - \hat{x}_{k-1})$ gives a descent direction for the max function at $\hat{x}_{k-1}$.
- Since both $\hat{x}_k$ and $\hat{x}_{k-1}$ fulfills all constraints of the minimax problem we can preform a line search between these points for the minimum of the max function F,

$$\lambda = \arg\min_{\lambda \in [0,1]} F\left(\lambda \hat{x_k} + (1-\lambda)\hat{x_{k-1}}\right). \tag{5}$$

- If $\lambda \neq 1$ a better solution is obtained at $x_k = \lambda \hat{x_k} + (1-\lambda)\hat{x}_{k-1}$.
  - A new upper bound for the objective is given by $F(x_k)$ and additional cutting planes can be generated at x_k.

Basic steps of the modified method

Basic steps of the modified method

0. First define $\Omega_0 = \{x, \mu \,|x \in L, \mu_{\text{min}} \leq \mu \leq \mu_{\text{max}}\}$, set $k = 1$ and specify the accepted optimality gap ϵ

Basic steps of the modified method

0. First define $\Omega_0 = \{x, \mu \,| x \in L, \mu_{\min} \leq \mu \leq \mu_{\max}\}$, set $k = 1$ and specify the accepted optimality gap ϵ

Repeat until $F(x_k) - \mu_k < \epsilon$
1. Find minimum of the objective within the linear set

$$[\hat{x}_k, \mu_k] \in \arg\min_{x,\mu \in \Omega_{k-1}} \mu$$

2. If k>1,

$$\text{find } \lambda = \arg\min_{\lambda \in [0,1]} F\left(\lambda \hat{x}_k + (1-\lambda)\hat{x}_{k-1}\right)$$

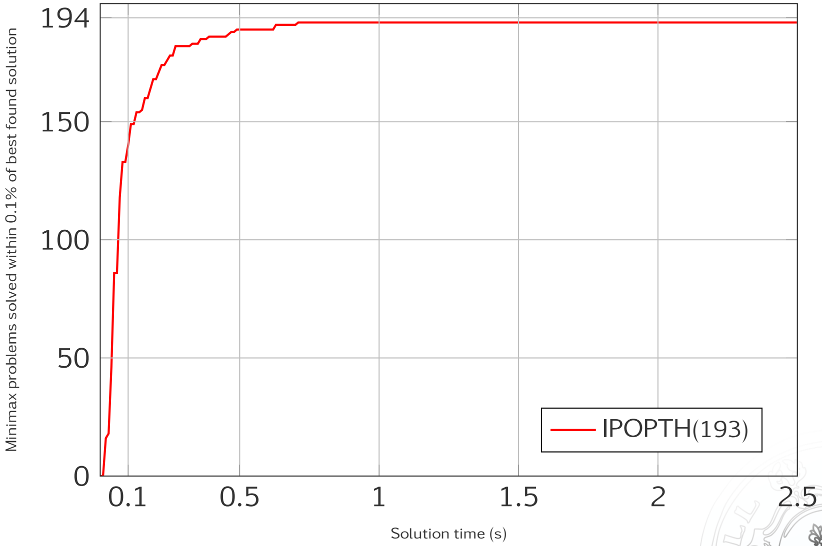
and set $x_k = \lambda \hat{x}_k + (1-\lambda)\hat{x}_{k-1}$.
3. Generate cutting planes $l_k(x, \mu)$ for all functions g_m such that $g_m(x) = F(x)$ at the both x_k and $\hat{x}_k$ and update the set Ω

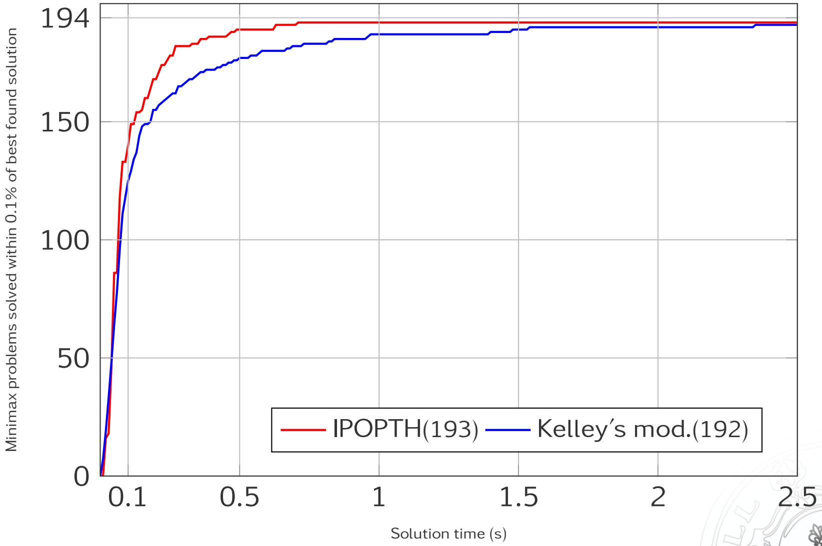
$$\Omega_k = \{x, \mu \,|\, l_k(x, \mu) \leq 0, \quad x, \mu \in \Omega_{k-1}\} \quad k = k + 1$$

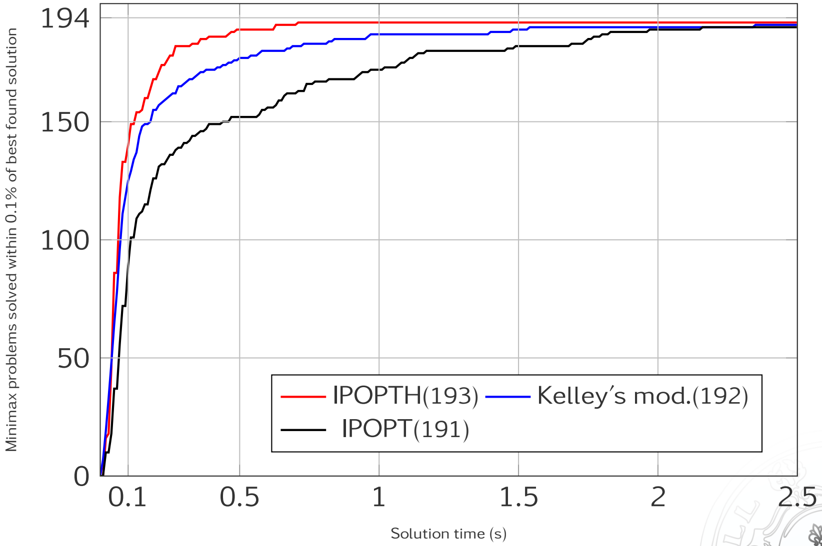
- The method was implemented in Matlab 2013.
  - Gurobi 6.0.3 was used as a subsolver for the LP subproblems.
  - For the line search we used Matlab's fminbnd.
  - Unconstrained variables are given limits of -10^{20} and 10^{20}.

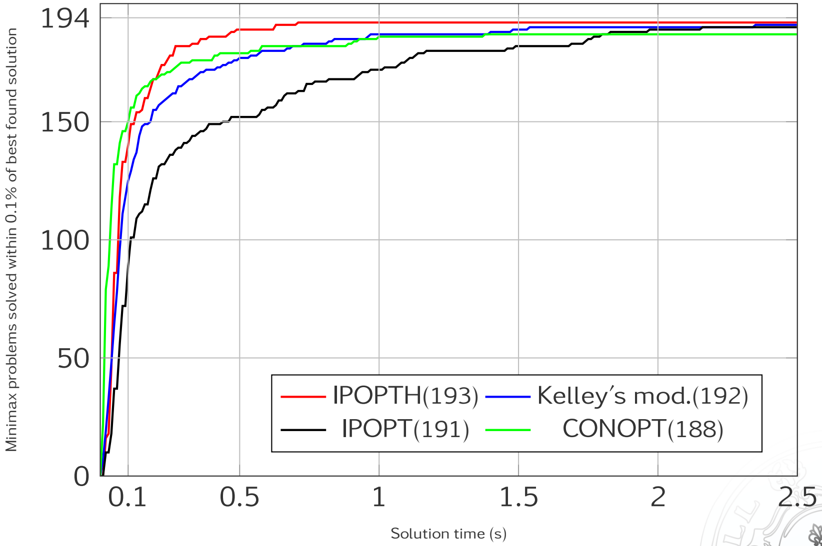
- The method was implemented in Matlab 2013.
  - Gurobi 6.0.3 was used as a subsolver for the LP subproblems.
  - For the line search we used Matlab's fminbnd.
  - Unconstrained variables are given limits of -10^{20} and 10^{20}.
- As a test set we have used all MINLP problems classified as convex and containing at least two nonlinear constraints within MINLPlib 2.
  - These have been rewritten as nonlinear minimax problems, where the objective is to minimize the pointwise maximum of the nonlinear constraint functions and obtain an interior point.

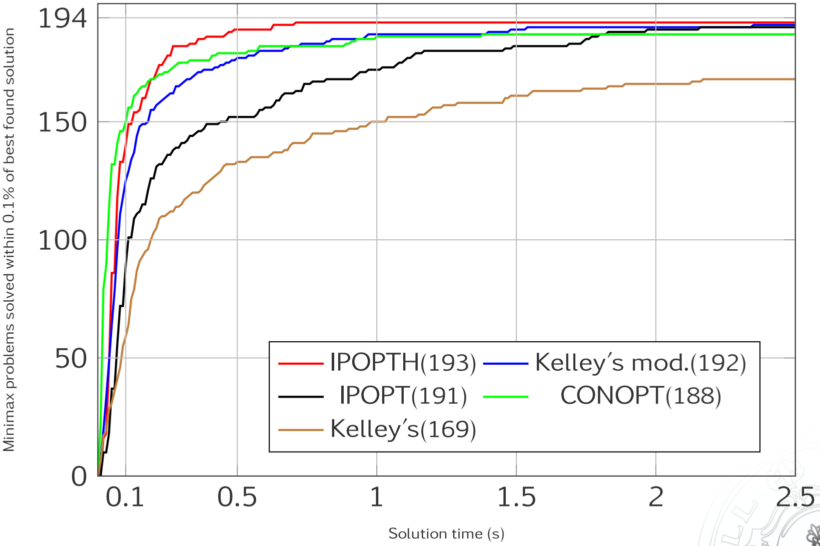
- The method was implemented in Matlab 2013.
  - Gurobi 6.0.3 was used as a subsolver for the LP subproblems.
  - For the line search we used Matlab's fminbnd.
  - Unconstrained variables are given limits of -10^{20} and 10^{20}.
- As a test set we have used all MINLP problems classified as convex and containing at least two nonlinear constraints within MINLPlib 2.
  - These have been rewritten as nonlinear minimax problems, where the objective is to minimize the pointwise maximum of the nonlinear constraint functions and obtain an interior point.
- To evaluate the performance we have compared the Matlab implementation against the following NLP solvers in GAMS: CONOPT, IPOPT, IPOPTH, SNOPT and MINOS.

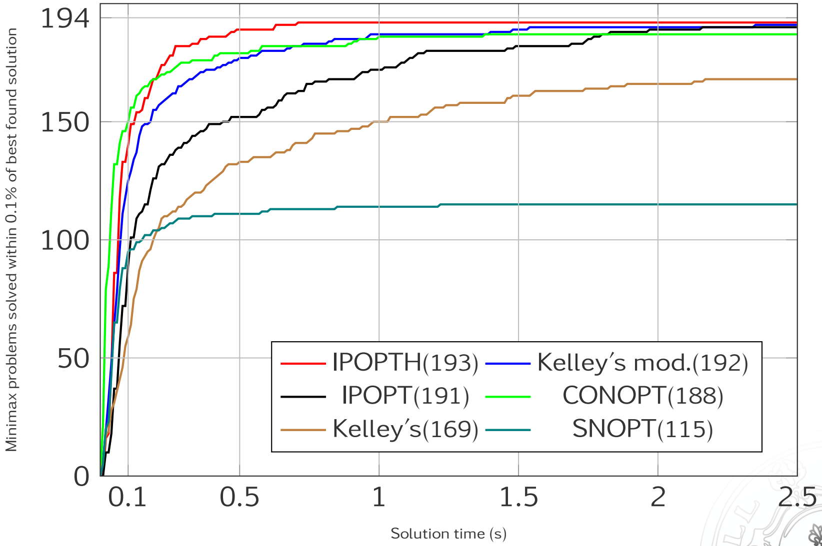

194
150
100
50
0
0.1
0.5
1
1.5
2
2.5
Minimax problems solved within 0.1% of best found solution
Solution time (s)
IPOPTH(193)

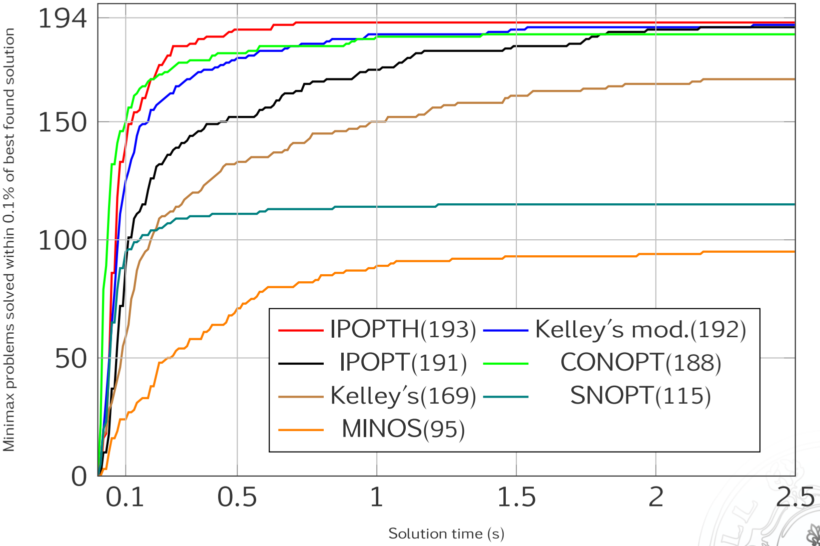

194
150
100
50
0
Minimax problems solved within 0.1% of best found solution
0.1
0.5
1
1.5
2
2.5
Solution time (s)
IPOPTH(193)
Kelley's mod.(192)

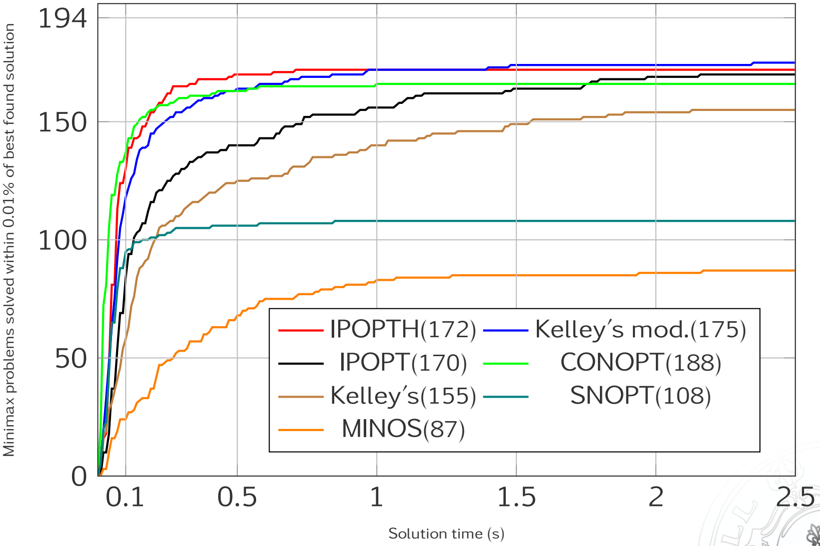

194
150
100
50
0
0.1
0.5
1
1.5
2
2.5
Minimax problems solved within 0.1% of best found solution
Solution time (s)
IPOPTH(193)
Kelley's mod.(192)
IPOPT(191)


194
150
100
50
0
0.1
0.5
1
1.5
2
2.5
Minimax problems solved within 0.1% of best found solution
Solution time (s)
IPOPTH(193)
Kelley's mod.(192)
IPOPT(191)
CONOPT(188)

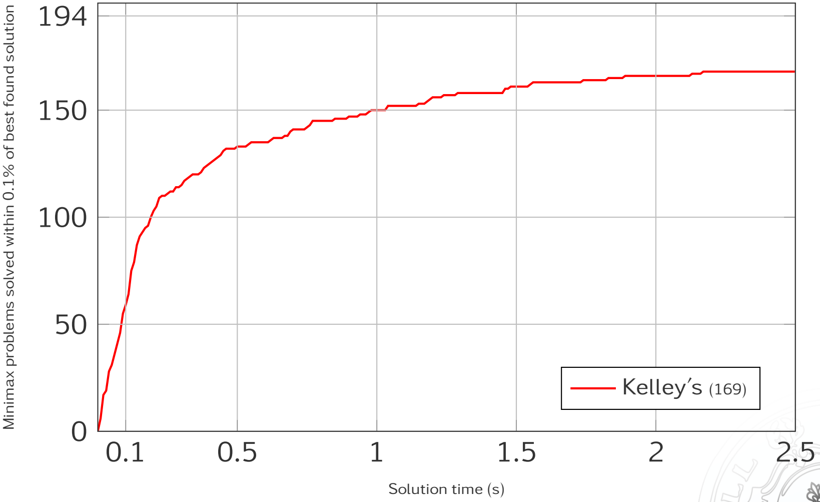

194
150
100
50
0
Minimax problems solved within 0.1% of best found solution
0.1
0.5
1
1.5
2
2.5
Solution time (s)
IPOPTH(193)
Kelley's mod.(192)
IPOPT(191)
CONOPT(188)
Kelley's(169)


194
150
100
50
0
Minimax problems solved within 0.1% of best found solution
0.1
0.5
1
1.5
2
2.5
Solution time (s)
IPOPTH(193)
Kelley's mod.(192)
IPOPT(191)
CONOPT(188)
Kelley's(169)
SNOPT(115)

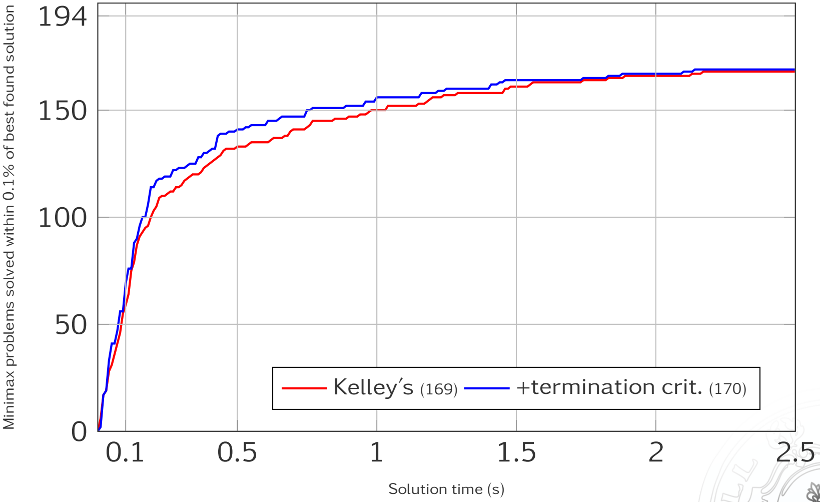

194
150
100
50
0
0.1
0.5
1
1.5
2
2.5
Minimax problems solved within 0.1% of best found solution
Solution time (s)
IPOPTH(193)
Kelley's mod.(192)
IPOPT(191)
CONOPT(188)
Kelley's(169)
SNOPT(115)
MINOS(95)


194
150
100
50
0
Minimax problems solved within 0.01% of best found solution
0.1
0.5
1
1.5
2
2.5
Solution time (s)
IPOPTH(172)
Kelley's mod.(175)
IPOPT(170)
CONOPT(188)
Kelley's(155)
SNOPT(108)
MINOS(87)

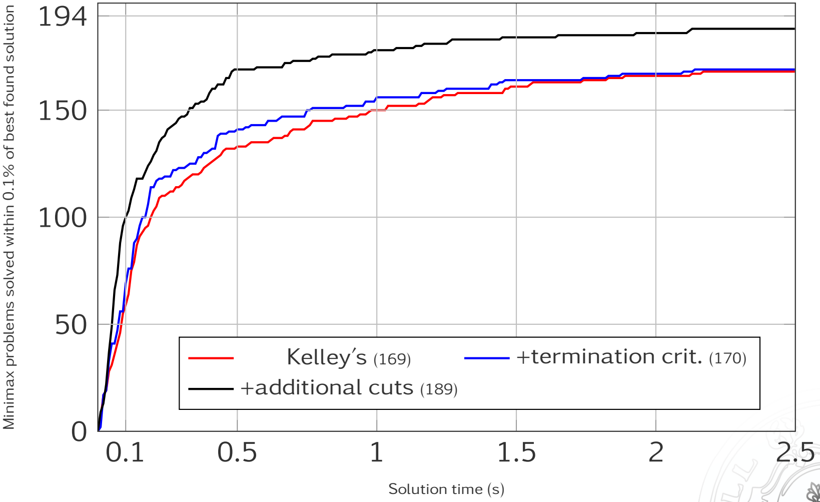
How does the modifications affect the performance?



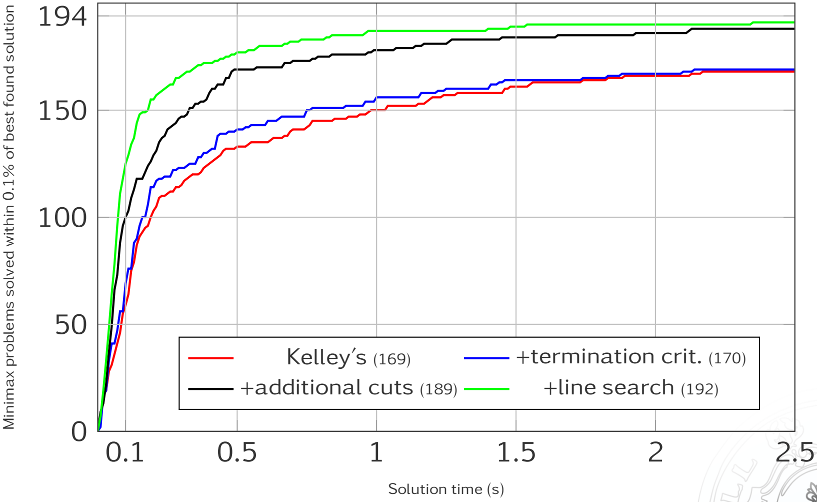
How does the modifications affect the performance?



How does the modifications affect the performance?

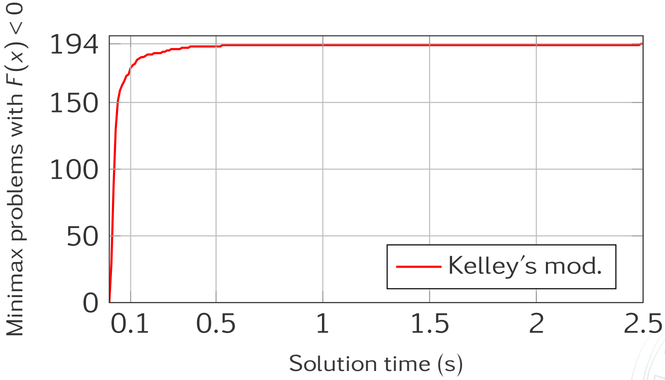


How does the modifications affect the performance?



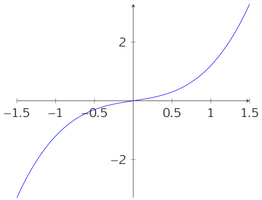
- However, in the ESH algorithm we do not need an optimal solution of the minimax problems. The solution $\bar{x}$ only has be within the set C, *i.e.*, $F(\bar{x}) < 0$.

- However, in the ESH algorithm we do not need an optimal solution of the minimax problems. The solution $\bar{x}$ only has be within the set C, *i.e.*, $F(\bar{x}) < 0$.
- Such a solution can be found much faster.

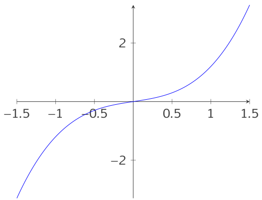


- How to efficiently find an interior point of the set $C = \{x \mid g_m(x) \leq 0 \quad \forall m,\ x \in \mathbb{R}^n\}$, when the functions g_m are pseudo convex?
  - Rewriting the problem results in the following nonconvex constraints: $g_m(x) - \mu \leq 0$.

- How to efficiently find an interior point of the set $C = \{x \mid g_m(x) \leq 0 \quad \forall m,\ x \in \mathbb{R}^n\}$, when the functions g_m are pseudo convex?
  - Rewriting the problem results in the following nonconvex constraints: $g_m(x) - \mu \leq 0$.



- How to efficiently find an interior point of the set $C = \{x \mid g_m(x) \leq 0 \quad \forall m,\ x \in \mathbb{R}^n\}$, when the functions g_m are pseudo convex?
  - Rewriting the problem results in the following nonconvex constraints: $g_m(x) - \mu \leq 0$.



- These minimax problems can be solved with a method similar to αECP, which is a method for solving pseudoconvex MINLP problems.[7]

[7] Solving pseudo-convex mixed integer optimization problems by cutting plane techniques, Westerlund, T., Pörn, R., Optimization and Engineering 3, 253-280 (2002).

Thank you for your attention!

Any questions?