Mixed Monotonic Programming for Fast Global Optimization

— published in IEEE Transactions on Signal Processing —

Bho Matthiesen[1] Christoph Hellings[2,3] Eduard A. Jorswieck[4] Wolfgang Utschick[2]

[1] University of Bremen, UB Excellence Chair, Dept. of Communications Engineering, Germany
[2] TU Munich, Methods of Signal Processing, Germany
[3] ETH Zürich, Department of Physics, Switzerland
[4] TU Braunschweig, Department of Information Theory and Communication Systemes, Germany

matthiesen@uni-bremen.de

IEEE ICASSP 2021, June 6, 2021

University of Bremen
Technische Universität München
Technische Universität Braunschweig
DFG Deutsche Forschungsgemeinschaft
German Research Foundation

Introduction

Mixed Monotonic Programming

- Global optimization framework
- Generalizes and replaces Monotonic Optimization [Tuy00]
- Several orders of magnitude faster than SoA

References

[Tuy00] H. Tuy. “Monotonic Optimization: Problems and Solution Approaches.” In: *SIAM J. Optim.* 11.2 (Feb. 2000), pp. 464–494.

Global Optimization

$$\max_{\boldsymbol{x} \in \mathcal{D}} f(\boldsymbol{x})$$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous

$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty

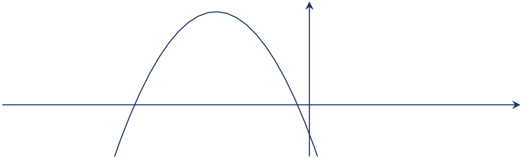
Global Optimization

$$\max_{\boldsymbol{x} \in \mathcal{D}} f(\boldsymbol{x})$$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous

$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty

$f(x)$ concave

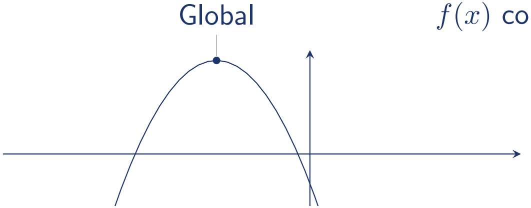


- Convex Problem

Global Optimization

$$\max_{\boldsymbol{x}\in\mathcal{D}} f(\boldsymbol{x})$$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous
$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty



$f(x)$ concave

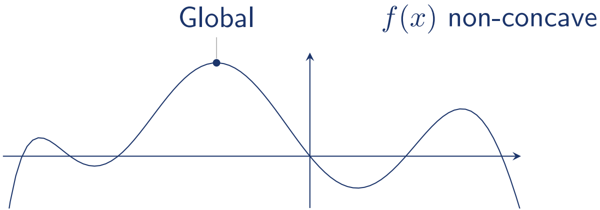
- Convex Problem
  - Local $=$ Global
  - Solvable in Polynomial Time

Global Optimization

$$\max_{\boldsymbol{x} \in \mathcal{D}} f(\boldsymbol{x})$$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous

$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty



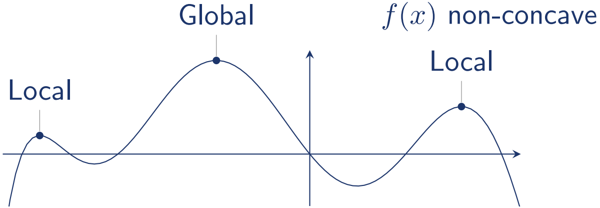
- Convex Problem
  - Local = Global
  - Solvable in Polynomial Time
- Multiextremal Problem

Global Optimization

$$\max_{\boldsymbol{x}\in\mathcal{D}} f(\boldsymbol{x})$$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous

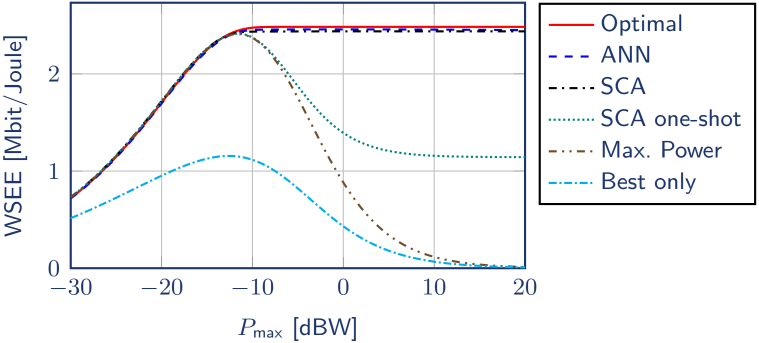
$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty



- Convex Problem
  - Local = Global
  - Solvable in Polynomial Time
- Multiextremal Problem
  - P-Time Algoritms: Guarantee Local Optimality at most
  - Completely Different Approach
  - NP-hard → Exponential Complexity

Global Optimization: Why?

- Ultimate Performance Limits
- Benchmark for Heuristics
- Label Training Data for ML



References

[Mat+20a] B. Matthiesen, A. Zappone, K.-L. Besser, E. A. Jorswieck, and M. Debbah. "A Globally Optimal Energy-Efficient Power Control Framework and its Efficient Implementation in Wireless Interference Networks." In: *IEEE Trans. Signal Process.* 68 (June 2020), pp. 3887–3902.

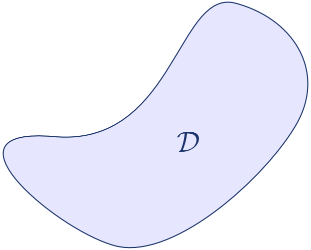
Branch-and-Bound Algorithm

Solve $\max_{\boldsymbol{x}\in\mathcal{D}} f(\boldsymbol{x})$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous
$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty

Prototype Branch-and-Bound Algorithm

Step 1 Initialize outer box $\mathcal{M}_0 \supseteq \mathcal{D}$, set best known solution $\bar{\boldsymbol{x}}^0$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^0)$
Step 2 Select box $\mathcal{M}_k$ for branching
Step 3 Bisect $\mathcal{M}_k$
Step 4 Reduce new Boxes (optional)
Step 5 Compute Bound $\beta(\mathcal{M}) \geq \sup_{\boldsymbol{x}\in\mathcal{M}\cap\mathcal{D}} f(\boldsymbol{x})$ for all new boxes $\mathcal{M}$
Step 6 Update best known solution $\bar{\boldsymbol{x}}^k$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^k)$
Step 7 Delete infeasible ($\mathcal{M} \cap \mathcal{D} = \emptyset$) and suboptimal ($\beta(\mathcal{M}) \leq \gamma_k + \eta$) new boxes
Step 8 Terminate if no box left or $\max_{\mathcal{M}} \beta(\mathcal{M}) \leq \gamma_k$: $\bar{\boldsymbol{x}}^k$ *is global η-optimal solution*



References

[Mat+20b] B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. “Mixed Monotonic Programming for Fast Global Optimization.” In: *IEEE Trans. Signal Process.* 68 (Mar. 2020), pp. 2529–2544.

[HT96] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. 3rd, rev. and enl. ed. Springer, 1996.

[Tuy16] H. Tuy. *Convex Analysis and Global Optimization*. 2nd ed. Springer, 2016.

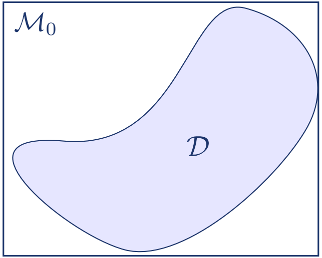
Branch-and-Bound Algorithm

Solve $\max_{\boldsymbol{x} \in \mathcal{D}} f(\boldsymbol{x})$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous
$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty

Prototype Branch-and-Bound Algorithm

Step 1 Initialize outer box $\mathcal{M}_0 \supseteq \mathcal{D}$, set best known solution $\bar{\boldsymbol{x}}^0$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^0)$
Step 2 Select box $\mathcal{M}_k$ for branching
Step 3 Bisect $\mathcal{M}_k$
Step 4 Reduce new Boxes (optional)
Step 5 Compute Bound $\beta(\mathcal{M}) \geq \sup_{\boldsymbol{x} \in \mathcal{M} \cap \mathcal{D}} f(\boldsymbol{x})$ for all new boxes $\mathcal{M}$
Step 6 Update best known solution $\bar{\boldsymbol{x}}^k$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^k)$
Step 7 Delete infeasible ($\mathcal{M} \cap \mathcal{D} = \emptyset$) and suboptimal ($\beta(\mathcal{M}) \leq \gamma_k + \eta$) new boxes
Step 8 Terminate if no box left or $\max_{\mathcal{M}} \beta(\mathcal{M}) \leq \gamma_k$: $\bar{\boldsymbol{x}}^k$ *is global η-optimal solution*



References

[Mat+20b] B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. “Mixed Monotonic Programming for Fast Global Optimization.” In: *IEEE Trans. Signal Process.* 68 (Mar. 2020), pp. 2529–2544.

[HT96] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. 3rd, rev. and enl. ed. Springer, 1996.

[Tuy16] H. Tuy. *Convex Analysis and Global Optimization*. 2nd ed. Springer, 2016.

Branch-and-Bound Algorithm

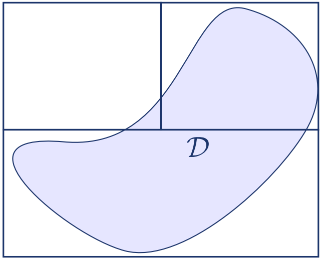
Solve $\max_{\boldsymbol{x} \in \mathcal{D}} f(\boldsymbol{x})$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous
$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty

Prototype Branch-and-Bound Algorithm

Step 1 Initialize outer box $\mathcal{M}_0 \supseteq \mathcal{D}$, set best known solution $\bar{\boldsymbol{x}}^0$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^0)$
Step 2 Select box $\mathcal{M}_k$ for branching
Step 3 Bisect $\mathcal{M}_k$
Step 4 Reduce new Boxes (optional)
Step 5 Compute Bound $\beta(\mathcal{M}) \geq \sup_{\boldsymbol{x} \in \mathcal{M} \cap \mathcal{D}} f(\boldsymbol{x})$ for all new boxes $\mathcal{M}$
Step 6 Update best known solution $\bar{\boldsymbol{x}}^k$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^k)$
Step 7 Delete infeasible ($\mathcal{M} \cap \mathcal{D} = \emptyset$) and suboptimal ($\beta(\mathcal{M}) \leq \gamma_k + \eta$) new boxes
Step 8 Terminate if no box left or $\max_{\mathcal{M}} \beta(\mathcal{M}) \leq \gamma_k$: $\bar{\boldsymbol{x}}^k$ *is global η-optimal solution*

after 2 iterations



References

[Mat+20b] B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. “Mixed Monotonic Programming for Fast Global Optimization.” In: *IEEE Trans. Signal Process.* 68 (Mar. 2020), pp. 2529–2544.

[HT96] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. 3rd, rev. and enl. ed. Springer, 1996.

[Tuy16] H. Tuy. *Convex Analysis and Global Optimization*. 2nd ed. Springer, 2016.

Branch-and-Bound Algorithm

Solve $\max_{\boldsymbol{x} \in \mathcal{D}} f(\boldsymbol{x})$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous

$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty

Prototype Branch-and-Bound Algorithm

Step 1 Initialize outer box $\mathcal{M}_0 \supseteq \mathcal{D}$, set best known solution $\bar{\boldsymbol{x}}^0$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^0)$

Step 2 Select box $\mathcal{M}_k$ for branching

Step 3 Bisect $\mathcal{M}_k$

Step 4 Reduce new Boxes (optional)

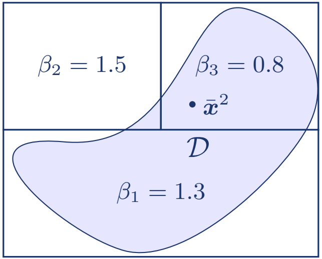
Step 5 Compute Bound $\beta(\mathcal{M}) \geq \sup_{\boldsymbol{x} \in \mathcal{M} \cap \mathcal{D}} f(\boldsymbol{x})$ for all new boxes $\mathcal{M}$

Step 6 Update best known solution $\bar{\boldsymbol{x}}^k$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^k)$

Step 7 Delete infeasible ($\mathcal{M} \cap \mathcal{D} = \emptyset$) and suboptimal ($\beta(\mathcal{M}) \leq \gamma_k + \eta$) new boxes

Step 8 Terminate if no box left or $\max_{\mathcal{M}} \beta(\mathcal{M}) \leq \gamma_k$: $\bar{\boldsymbol{x}}^k$ *is global η-optimal solution*

after 2 iterations



References

[Mat+20b] B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. “Mixed Monotonic Programming for Fast Global Optimization.” In: *IEEE Trans. Signal Process.* 68 (Mar. 2020), pp. 2529–2544.

[HT96] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. 3rd, rev. and enl. ed. Springer, 1996.

[Tuy16] H. Tuy. *Convex Analysis and Global Optimization*. 2nd ed. Springer, 2016.

Branch-and-Bound Algorithm

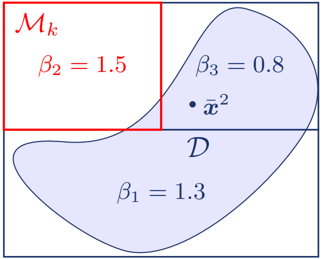
Solve $\max_{\boldsymbol{x}\in\mathcal{D}} f(\boldsymbol{x})$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous
$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty

Prototype Branch-and-Bound Algorithm

Step 1 Initialize outer box $\mathcal{M}_0 \supseteq \mathcal{D}$, set best known solution $\bar{\boldsymbol{x}}^0$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^0)$
Step 2 Select box $\mathcal{M}_k$ for branching: $\arg\max \beta$
Step 3 Bisect $\mathcal{M}_k$
Step 4 Reduce new Boxes (optional)
Step 5 Compute Bound $\beta(\mathcal{M}) \geq \sup_{\boldsymbol{x}\in\mathcal{M}\cap\mathcal{D}} f(\boldsymbol{x})$ for all new boxes $\mathcal{M}$
Step 6 Update best known solution $\bar{\boldsymbol{x}}^k$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^k)$
Step 7 Delete infeasible ($\mathcal{M} \cap \mathcal{D} = \emptyset$) and suboptimal ($\beta(\mathcal{M}) \leq \gamma_k + \eta$) new boxes
Step 8 Terminate if no box left or $\max_{\mathcal{M}} \beta(\mathcal{M}) \leq \gamma_k$: $\bar{\boldsymbol{x}}^k$ *is global η-optimal solution*

in iteration 3



References

[Mat+20b] B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. “Mixed Monotonic Programming for Fast Global Optimization.” In: *IEEE Trans. Signal Process.* 68 (Mar. 2020), pp. 2529–2544.

[HT96] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. 3rd, rev. and enl. ed. Springer, 1996.

[Tuy16] H. Tuy. *Convex Analysis and Global Optimization*. 2nd ed. Springer, 2016.

Branch-and-Bound Algorithm

Solve $\max\limits_{\boldsymbol{x}\in\mathcal{D}} f(\boldsymbol{x})$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous

$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty

Prototype Branch-and-Bound Algorithm

Step 1 Initialize outer box $\mathcal{M}_0 \supseteq \mathcal{D}$, set best known solution $\bar{\boldsymbol{x}}^0$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^0)$

Step 2 Select box $\mathcal{M}_k$ for branching

Step 3 Bisect $\mathcal{M}_k$

Step 4 Reduce new Boxes (optional)

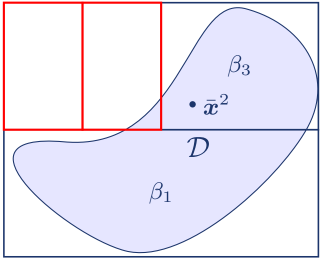
Step 5 Compute Bound $\beta(\mathcal{M}) \geq \sup_{\boldsymbol{x}\in\mathcal{M}\cap\mathcal{D}} f(\boldsymbol{x})$ for all new boxes $\mathcal{M}$

Step 6 Update best known solution $\bar{\boldsymbol{x}}^k$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^k)$

Step 7 Delete infeasible ($\mathcal{M} \cap \mathcal{D} = \emptyset$) and suboptimal ($\beta(\mathcal{M}) \leq \gamma_k + \eta$) new boxes

Step 8 Terminate if no box left or $\max_{\mathcal{M}} \beta(\mathcal{M}) \leq \gamma_k$: $\bar{\boldsymbol{x}}^k$ *is global η-optimal solution*

in iteration 3



References

[Mat+20b] B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. "Mixed Monotonic Programming for Fast Global Optimization." In: *IEEE Trans. Signal Process.* 68 (Mar. 2020), pp. 2529–2544.

[HT96] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. 3rd, rev. and enl. ed. Springer, 1996.

[Tuy16] H. Tuy. *Convex Analysis and Global Optimization*. 2nd ed. Springer, 2016.

Branch-and-Bound Algorithm

Solve $\max_{\boldsymbol{x} \in \mathcal{D}} f(\boldsymbol{x})$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous
$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty

Prototype Branch-and-Bound Algorithm

Step 1 Initialize outer box $\mathcal{M}_0 \supseteq \mathcal{D}$, set best known solution $\bar{\boldsymbol{x}}^0$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^0)$

Step 2 Select box $\mathcal{M}_k$ for branching

Step 3 Bisect $\mathcal{M}_k$

Step 4 Reduce new Boxes (optional)

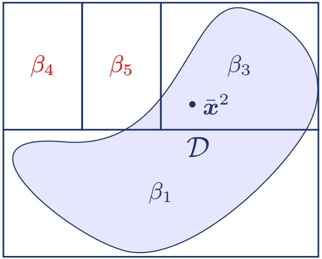
Step 5 Compute Bound $\beta(\mathcal{M}) \geq \sup_{\boldsymbol{x} \in \mathcal{M} \cap \mathcal{D}} f(\boldsymbol{x})$ for all new boxes $\mathcal{M}$

Step 6 Update best known solution $\bar{\boldsymbol{x}}^k$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^k)$

Step 7 Delete infeasible ($\mathcal{M} \cap \mathcal{D} = \emptyset$) and suboptimal ($\beta(\mathcal{M}) \leq \gamma_k + \eta$) new boxes

Step 8 Terminate if no box left or $\max_{\mathcal{M}} \beta(\mathcal{M}) \leq \gamma_k$: $\bar{\boldsymbol{x}}^k$ *is global η-optimal solution*

in iteration 3



References

[Mat+20b] B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. “Mixed Monotonic Programming for Fast Global Optimization.” In: *IEEE Trans. Signal Process.* 68 (Mar. 2020), pp. 2529–2544.

[HT96] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. 3rd, rev. and enl. ed. Springer, 1996.

[Tuy16] H. Tuy. *Convex Analysis and Global Optimization*. 2nd ed. Springer, 2016.

Branch-and-Bound Algorithm

Solve $\max_{\boldsymbol{x} \in \mathcal{D}} f(\boldsymbol{x})$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous

$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty

Prototype Branch-and-Bound Algorithm

Step 1 Initialize outer box $\mathcal{M}_0 \supseteq \mathcal{D}$, set best known solution $\bar{\boldsymbol{x}}^0$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^0)$

Step 2 Select box $\mathcal{M}_k$ for branching

Step 3 Bisect $\mathcal{M}_k$

Step 4 Reduce new Boxes (optional)

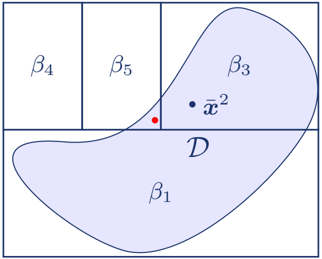
Step 5 Compute Bound $\beta(\mathcal{M}) \geq \sup_{\boldsymbol{x} \in \mathcal{M} \cap \mathcal{D}} f(\boldsymbol{x})$ for all new boxes $\mathcal{M}$

Step 6 Update best known solution $\bar{\boldsymbol{x}}^k$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^k)$

Step 7 Delete infeasible ($\mathcal{M} \cap \mathcal{D} = \emptyset$) and suboptimal ($\beta(\mathcal{M}) \leq \gamma_k + \eta$) new boxes

Step 8 Terminate if no box left or $\max_{\mathcal{M}} \beta(\mathcal{M}) \leq \gamma_k$: $\bar{\boldsymbol{x}}^k$ *is global* η*-optimal solution*

in iteration 3



References

[Mat+20b] B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. "Mixed Monotonic Programming for Fast Global Optimization." In: *IEEE Trans. Signal Process.* 68 (Mar. 2020), pp. 2529–2544.

[HT96] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. 3rd, rev. and enl. ed. Springer, 1996.

[Tuy16] H. Tuy. *Convex Analysis and Global Optimization*. 2nd ed. Springer, 2016.

Branch-and-Bound Algorithm

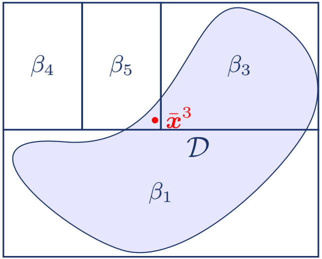
Solve $\max_{\boldsymbol{x} \in \mathcal{D}} f(\boldsymbol{x})$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous
$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty

Prototype Branch-and-Bound Algorithm

Step 1 Initialize outer box $\mathcal{M}_0 \supseteq \mathcal{D}$, set best known solution $\bar{\boldsymbol{x}}^0$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^0)$
Step 2 Select box $\mathcal{M}_k$ for branching
Step 3 Bisect $\mathcal{M}_k$
Step 4 Reduce new Boxes (optional)
Step 5 Compute Bound $\beta(\mathcal{M}) \geq \sup_{\boldsymbol{x} \in \mathcal{M} \cap \mathcal{D}} f(\boldsymbol{x})$ for all new boxes $\mathcal{M}$
Step 6 Update best known solution $\bar{\boldsymbol{x}}^k$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^k)$
Step 7 Delete infeasible ($\mathcal{M} \cap \mathcal{D} = \emptyset$) and suboptimal ($\beta(\mathcal{M}) \leq \gamma_k + \eta$) new boxes
Step 8 Terminate if no box left or $\max_{\mathcal{M}} \beta(\mathcal{M}) \leq \gamma_k$: $\bar{\boldsymbol{x}}^k$ *is global η-optimal solution*

in iteration 3



References

[Mat+20b] B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. "Mixed Monotonic Programming for Fast Global Optimization." In: *IEEE Trans. Signal Process.* 68 (Mar. 2020), pp. 2529–2544.

[HT96] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches.* 3rd, rev. and enl. ed. Springer, 1996.

[Tuy16] H. Tuy. *Convex Analysis and Global Optimization.* 2nd ed. Springer, 2016.

Branch-and-Bound Algorithm

Solve $\max_{\boldsymbol{x}\in\mathcal{D}} f(\boldsymbol{x})$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous

$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty

Prototype Branch-and-Bound Algorithm

Step 1 Initialize outer box $\mathcal{M}_0 \supseteq \mathcal{D}$, set best known solution $\bar{\boldsymbol{x}}^0$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^0)$

Step 2 Select box $\mathcal{M}_k$ for branching

Step 3 Bisect $\mathcal{M}_k$

Step 4 Reduce new Boxes (optional)

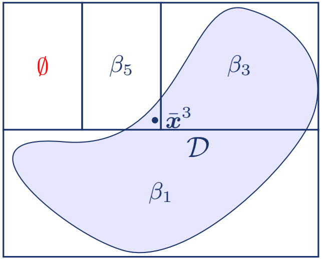
Step 5 Compute Bound $\beta(\mathcal{M}) \geq \sup_{\boldsymbol{x}\in\mathcal{M}\cap\mathcal{D}} f(\boldsymbol{x})$ for all new boxes $\mathcal{M}$

Step 6 Update best known solution $\bar{\boldsymbol{x}}^k$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^k)$

Step 7 Delete infeasible ($\mathcal{M} \cap \mathcal{D} = \emptyset$) and suboptimal ($\beta(\mathcal{M}) \leq \gamma_k + \eta$) new boxes

Step 8 Terminate if no box left or $\max_{\mathcal{M}} \beta(\mathcal{M}) \leq \gamma_k$: $\bar{\boldsymbol{x}}^k$ *is global η-optimal solution*

in iteration 3



References

[Mat+20b] B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. "Mixed Monotonic Programming for Fast Global Optimization." In: *IEEE Trans. Signal Process.* 68 (Mar. 2020), pp. 2529–2544.

[HT96] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. 3rd, rev. and enl. ed. Springer, 1996.

[Tuy16] H. Tuy. *Convex Analysis and Global Optimization*. 2nd ed. Springer, 2016.

Branch-and-Bound Algorithm

Solve $\max\limits_{\boldsymbol{x}\in\mathcal{D}} f(\boldsymbol{x})$

$f : \mathcal{D} \mapsto \mathbb{R}$ continuous

$\mathcal{D} \subseteq \mathbb{R}^n$ compact, non-empty

Prototype Branch-and-Bound Algorithm

Step 1 Initialize outer box $\mathcal{M}_0 \supseteq \mathcal{D}$, set best known solution $\bar{\boldsymbol{x}}^0$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^0)$

Step 2 Select box $\mathcal{M}_k$ for branching

Step 3 Bisect $\mathcal{M}_k$

Step 4 Reduce new Boxes (optional)

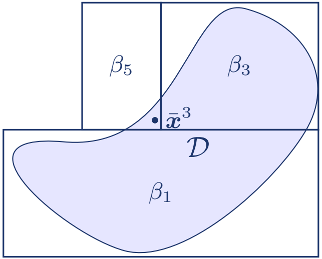
Step 5 Compute Bound $\beta(\mathcal{M}) \geq \sup_{\boldsymbol{x}\in\mathcal{M}\cap\mathcal{D}} f(\boldsymbol{x})$ for all new boxes $\mathcal{M}$

Step 6 Update best known solution $\bar{\boldsymbol{x}}^k$ and value $\gamma_0 = f(\bar{\boldsymbol{x}}^k)$

Step 7 Delete infeasible ($\mathcal{M} \cap \mathcal{D} = \emptyset$) and suboptimal ($\beta(\mathcal{M}) \leq \gamma_k + \eta$) new boxes

Step 8 Terminate if no box left or $\max_{\mathcal{M}} \beta(\mathcal{M}) \leq \gamma_k$: $\bar{\boldsymbol{x}}^k$ *is global η-optimal solution*

after iteration 3



References

[Mat+20b] B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. "Mixed Monotonic Programming for Fast Global Optimization." In: *IEEE Trans. Signal Process.* 68 (Mar. 2020), pp. 2529–2544.

[HT96] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. 3rd, rev. and enl. ed. Springer, 1996.

[Tuy16] H. Tuy. *Convex Analysis and Global Optimization*. 2nd ed. Springer, 2016.

BB: Bounding

- State-of-the-Art: Monotonic Optimization

$$\max_{\boldsymbol{x} \in \mathcal{G} \cap \mathcal{H}} f^+(\boldsymbol{x}) - f^-(\boldsymbol{x}) \qquad f^+, f^- : \text{ increasing functions}$$

H. Tuy. “Monotonic Optimization: Problems and Solution Approaches.” In: *SIAM J. Optim.* 11.2 (Feb. 2000), pp. 464–494

BB: Bounding

- State-of-the-Art: Monotonic Optimization

$$\max_{\boldsymbol{x} \in \mathcal{G} \cap \mathcal{H}} f^{+}(\boldsymbol{x}) - f^{-}(\boldsymbol{x}) \qquad\qquad f^{+}, f^{-} : \text{ increasing functions}$$

- Bounding over $\mathcal{M} = [\boldsymbol{r}, \boldsymbol{s}]$:

$$\max_{\boldsymbol{x} \in \mathcal{G} \cap \mathcal{H} \cap \mathcal{M}} f^{+}(\boldsymbol{x}) - f^{-}(\boldsymbol{x}) \leq \max_{\boldsymbol{x} \in \mathcal{M}} f^{+}(\boldsymbol{x}) - \min_{\boldsymbol{x} \in \mathcal{M}} f^{-}(\boldsymbol{x}) = f^{+}(\boldsymbol{s}) - f^{-}(\boldsymbol{r})$$

H. Tuy. “Monotonic Optimization: Problems and Solution Approaches.” In: *SIAM J. Optim.* 11.2 (Feb. 2000), pp. 464–494

BB: Bounding

- State-of-the-Art: Monotonic Optimization

$$\max_{\boldsymbol{x}\in\mathcal{G}\cap\mathcal{H}} f^+(\boldsymbol{x}) - f^-(\boldsymbol{x}) \qquad\qquad f^+, f^- : \text{ increasing functions}$$

- Bounding over $\mathcal{M} = [\boldsymbol{r}, \boldsymbol{s}]$:

$$\max_{\boldsymbol{x}\in\mathcal{G}\cap\mathcal{H}\cap\mathcal{M}} f^+(\boldsymbol{x}) - f^-(\boldsymbol{x}) \leq \max_{\boldsymbol{x}\in\mathcal{M}} f^+(\boldsymbol{x}) - \min_{\boldsymbol{x}\in\mathcal{M}} f^-(\boldsymbol{x}) = f^+(\boldsymbol{s}) - f^-(\boldsymbol{r})$$

- Example: $\log\left(1 + \frac{\boldsymbol{b}_k^T \boldsymbol{p}}{\boldsymbol{c}_k^T \boldsymbol{p} + \sigma_k^2}\right) = \log\left((\boldsymbol{b}_k + \boldsymbol{c}_k)^T \boldsymbol{p} + \sigma_k^2\right) - \log\left(\boldsymbol{c}_k^T \boldsymbol{p} + \sigma_k^2\right)$

H. Tuy. “Monotonic Optimization: Problems and Solution Approaches.” In: *SIAM J. Optim.* 11.2 (Feb. 2000), pp. 464–494

BB: Bounding

- State-of-the-Art: Monotonic Optimization

$$\max_{\boldsymbol{x}\in\mathcal{G}\cap\mathcal{H}} f^+(\boldsymbol{x}) - f^-(\boldsymbol{x}) \qquad\qquad f^+, f^- : \text{ increasing functions}$$

- Bounding over $\mathcal{M} = [\boldsymbol{r}, \boldsymbol{s}]$:

$$\max_{\boldsymbol{x}\in\mathcal{G}\cap\mathcal{H}\cap\mathcal{M}} f^+(\boldsymbol{x}) - f^-(\boldsymbol{x}) \leq \max_{\boldsymbol{x}\in\mathcal{M}} f^+(\boldsymbol{x}) - \min_{\boldsymbol{x}\in\mathcal{M}} f^-(\boldsymbol{x}) = f^+(\boldsymbol{s}) - f^-(\boldsymbol{r})$$

- Example: $\log\left(1 + \frac{\boldsymbol{b}_k^T\boldsymbol{p}}{\boldsymbol{c}_k^T\boldsymbol{p}+\sigma_k^2}\right) = \log\left((\boldsymbol{b}_k + \boldsymbol{c}_k)^T\boldsymbol{p} + \sigma_k^2\right) - \log\left(\boldsymbol{c}_k^T\boldsymbol{p} + \sigma_k^2\right)$
- BUT: $\frac{\sum_k \log\left(1+\frac{\boldsymbol{b}_k^T\boldsymbol{p}}{\boldsymbol{c}_k^T\boldsymbol{p}+\sigma_k^2}\right)}{\boldsymbol{\phi}^T\boldsymbol{p}+P_c} \overset{???}{=} f^+(\boldsymbol{x}) - f^-(\boldsymbol{x})$

→ Fractional Programs require Dinkelbach's Algorithm

H. Tuy. “Monotonic Optimization: Problems and Solution Approaches.” In: *SIAM J. Optim.* 11.2 (Feb. 2000), pp. 464–494
A. Zappone, et al. ”Globally Optimal Energy-Efficient Power Control and Receiver Design in Wireless Networks,” *IEEE TSP*, 2017.

BB: Bounding

Can we do better?

$$\frac{\sum_k \log\left(1 + \frac{\boldsymbol{b}_k^T \boldsymbol{p}}{\boldsymbol{c}_k^T \boldsymbol{p} + \sigma_k^2}\right)}{\boldsymbol{\phi}^T \boldsymbol{p} + P_c}$$

BB: Bounding

Can we do better?

$$\max_{\boldsymbol{p}\in[\boldsymbol{r},\boldsymbol{s}]} \frac{\sum_k \log\left(1+\frac{\boldsymbol{b}_k^T\boldsymbol{p}}{\boldsymbol{c}_k^T\boldsymbol{p}+\sigma_k^2}\right)}{\boldsymbol{\phi}^T\boldsymbol{p}+P_c} \le \frac{\sum_k \log\left(1+\frac{\boldsymbol{b}_k^T\phantom{\boldsymbol{p}}}{\boldsymbol{c}_k^T\phantom{\boldsymbol{p}}+\sigma_k^2}\right)}{\boldsymbol{\phi}^T\phantom{\boldsymbol{p}}+P_c}$$

BB: Bounding

Can we do better?

increasing

$$\max_{\boldsymbol{p}\in[\boldsymbol{r},\boldsymbol{s}]} \frac{\sum_k \log\left(1+\frac{\boldsymbol{b}_k^T\boldsymbol{p}}{\boldsymbol{c}_k^T\boldsymbol{p}+\sigma_k^2}\right)}{\boldsymbol{\phi}^T\boldsymbol{p}+P_c} \leq \frac{\sum_k \log\left(1+\frac{\boldsymbol{b}_k^T\boldsymbol{s}}{\boldsymbol{c}_k^T\quad+\sigma_k^2}\right)}{\boldsymbol{\phi}^T\quad+P_c}$$

BB: Bounding

Can we do better?

increasing

$$\max_{\boldsymbol{p}\in[\boldsymbol{r},\boldsymbol{s}]} \frac{\sum_k \log\left(1+\frac{\boldsymbol{b}_k^T\boldsymbol{p}}{\boldsymbol{c}_k^T\boldsymbol{p}+\sigma_k^2}\right)}{\boldsymbol{\phi}^T\boldsymbol{p}+P_c} \leq \frac{\sum_k \log\left(1+\frac{\boldsymbol{b}_k^T\boldsymbol{s}}{\boldsymbol{c}_k^T\boldsymbol{r}+\sigma_k^2}\right)}{\boldsymbol{\phi}^T\quad+P_c}$$

decreasing

BB: Bounding

Can we do better?

$$\max_{\boldsymbol{p}\in[\boldsymbol{r},\boldsymbol{s}]} \frac{\sum_k \log\left(1 + \frac{\boldsymbol{b}_k^T \overset{\text{increasing}}{\boldsymbol{p}}}{\boldsymbol{c}_k^T \underset{\text{decreasing}}{\boldsymbol{p}} + \sigma_k^2}\right)}{\boldsymbol{\phi}^T \underset{\text{decreasing}}{\boldsymbol{p}} + P_c} \leq \frac{\sum_k \log\left(1 + \frac{\boldsymbol{b}_k^T \boldsymbol{s}}{\boldsymbol{c}_k^T \boldsymbol{r} + \sigma_k^2}\right)}{\boldsymbol{\phi}^T \boldsymbol{r} + P_c}$$

BB: Bounding

Can we do better?

$$\max_{\boldsymbol{p}\in[\boldsymbol{r},\boldsymbol{s}]} \frac{\sum_k \log\left(1+\frac{\boldsymbol{b}_k^T \overset{\text{increasing}}{\boldsymbol{p}}}{\boldsymbol{c}_k^T \underset{\text{decreasing}}{\boldsymbol{p}}+\sigma_k^2}\right)}{\boldsymbol{\phi}^T \underset{\text{decreasing}}{\boldsymbol{p}} + P_c} \le \frac{\sum_k \log\left(1+\frac{\boldsymbol{b}_k^T\boldsymbol{s}}{\boldsymbol{c}_k^T\boldsymbol{r}+\sigma_k^2}\right)}{\boldsymbol{\phi}^T\boldsymbol{r}+P_c} = F(\boldsymbol{s},\boldsymbol{r})$$

Definition

Continuous function $F : \mathbb{R}^n \times \mathbb{R}^n \mapsto \mathbb{R}$ such that

$$\begin{aligned} F(\boldsymbol{x},\boldsymbol{y}) &\le F(\boldsymbol{x}',\boldsymbol{y}) \quad \text{if } \boldsymbol{x} \le \boldsymbol{x}' && \text{(Increasing in } \boldsymbol{x}\text{)} \\ F(\boldsymbol{x},\boldsymbol{y}) &\ge F(\boldsymbol{x},\boldsymbol{y}') \quad \text{if } \boldsymbol{y} \le \boldsymbol{y}' && \text{(Decreasing in } \boldsymbol{y}\text{)} \end{aligned}$$

Then, F is called Mixed Monotonic (MM) Function.

Mixed Monotonic Programming

$$\max_{\boldsymbol{x} \in \mathcal{D}} \; f(\boldsymbol{x}) \qquad \text{(MMP)}$$

$$f : \mathcal{D} \mapsto \mathbb{R} \text{ continuous} \qquad \mathcal{D} \subseteq \mathbb{R}^n \text{ compact, non-empty}$$

Definition

Let $\mathcal{M}_0 = [\boldsymbol{r}^0, \boldsymbol{s}^0] \supseteq \mathcal{D}$ be a box enclosing $\mathcal{D}$. Assume there exists a mixed monotonic function F on $\mathcal{M}_0$ such that

$$F(\boldsymbol{x}, \boldsymbol{x}) = f(\boldsymbol{x})$$

for all $\boldsymbol{x} \in \mathcal{M}_0$. Then, (MMP) is a mixed monotonic program.

Key Observations:

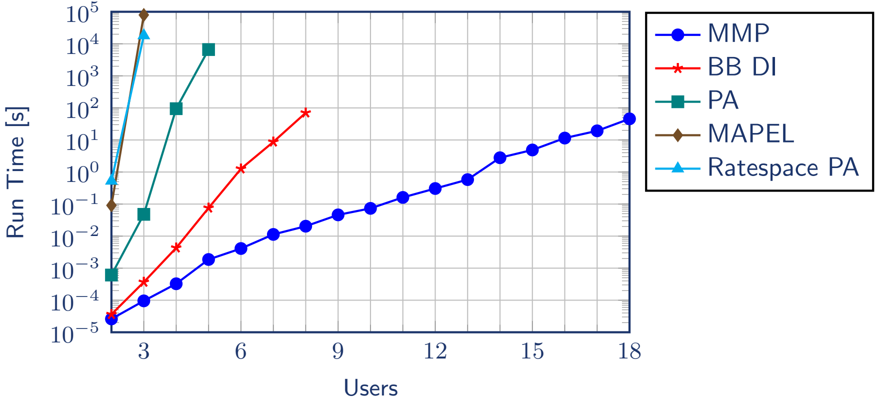
$$\operatorname*{arg\,max}_{\boldsymbol{x} \in \mathcal{D}} \; f(\boldsymbol{x}) = \operatorname*{arg\,max}_{\boldsymbol{x} \in \mathcal{D}} \; F(\boldsymbol{x}, \boldsymbol{x})$$

$$\max_{\boldsymbol{x} \in \mathcal{M} \cap \mathcal{D}} f(\boldsymbol{x}) \leq \max_{\boldsymbol{x} \in \mathcal{M}} F(\boldsymbol{x}, \boldsymbol{x}) \leq \max_{\boldsymbol{x}, \boldsymbol{y} \in \mathcal{M}} F(\boldsymbol{x}, \boldsymbol{y}) = F(\boldsymbol{s}, \boldsymbol{r}). \qquad (\mathcal{M} = [\boldsymbol{r}, \boldsymbol{s}])$$

B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. "Mixed Monotonic Programming for Fast Global Optimization," *IEEE TSP*, Mar. 2020.

BB: Bounding

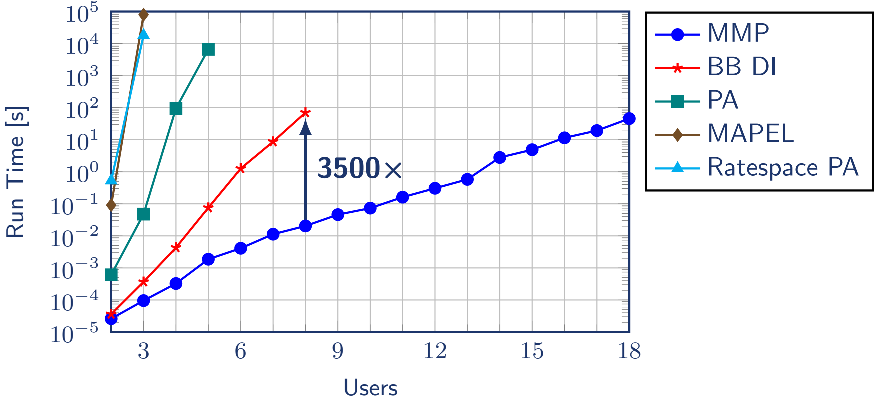
TIN WSR: $$\max_{\boldsymbol{p}\in[\boldsymbol{0},\boldsymbol{P}]} \sum_k w_i \log\left(1 + \frac{\boldsymbol{b}_k^T \boldsymbol{p}}{\boldsymbol{c}_k^T \boldsymbol{p} + \sigma_k^2}\right)$$



B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. “Mixed Monotonic Programming for Fast Global Optimization,” *IEEE TSP*, Mar. 2020.
B. Matthiesen and C. Hellings. *Mixed Monotonic Programming Source Code*. 2019. URL: `https://github.com/bmatthiesen/mixed-monotonic`

BB: Bounding

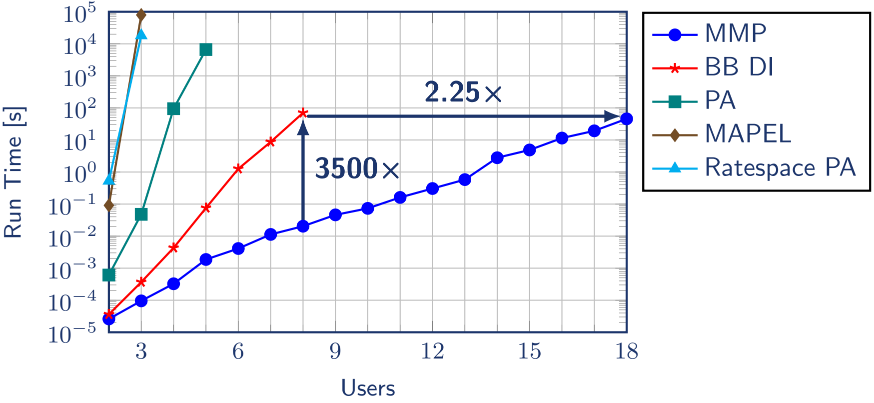
TIN WSR: $$\max_{\boldsymbol{p}\in[\boldsymbol{0},\boldsymbol{P}]} \sum_k w_i \log\left(1 + \frac{\boldsymbol{b}_k^T \boldsymbol{p}}{\boldsymbol{c}_k^T \boldsymbol{p} + \sigma_k^2}\right)$$



B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. "Mixed Monotonic Programming for Fast Global Optimization," *IEEE TSP*, Mar. 2020.
B. Matthiesen and C. Hellings. *Mixed Monotonic Programming Source Code*. 2019. URL: `https://github.com/bmatthiesen/mixed-monotonic`

BB: Bounding

TIN WSR: $$\max_{\boldsymbol{p}\in[\boldsymbol{0},\boldsymbol{P}]} \sum_k w_i \log\left(1 + \frac{\boldsymbol{b}_k^T \boldsymbol{p}}{\boldsymbol{c}_k^T \boldsymbol{p} + \sigma_k^2}\right)$$



B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. “Mixed Monotonic Programming for Fast Global Optimization,” *IEEE TSP*, Mar. 2020.
B. Matthiesen and C. Hellings. *Mixed Monotonic Programming Source Code*. 2019. URL: `https://github.com/bmatthiesen/mixed-monotonic`

Conclusions

Mixed Monotonic Programming

- Global optimization framework
- Versatile and widely applicable → Replaces classical monotonic programming
- Several orders of magnitude faster than SoA

Tutorial *Efficient Global Optimization and its Application to Wireless Interference Networks*

- In-depth treatment of algorithmic foundations
- Several application examples
- At ICASSP 2021 and ICC 2021

matthiesen@uni-bremen.de

`https://github.com/bmatthiesen`

References

[Mat+20b] B. Matthiesen, C. Hellings, E. A. Jorswieck, and W. Utschick. "Mixed Monotonic Programming for Fast Global Optimization." In: *IEEE Trans. Signal Process.* 68 (Mar. 2020), pp. 2529–2544.

[Mat19] B. Matthiesen. "Efficient Globally Optimal Resource Allocation in Wireless Interference Networks." Ph.D. Thesis. Dresden, Germany: Technische Universität Dresden, Nov. 2019. URL: `https://theses.eurasip.org/theses/853/efficient-globally-optimal-resource-allocation-in`.