Mixed-integer programming in signal processing and communications

Tutorial at ICASSP 2015, Brisbane, Australia

TECHNISCHE UNIVERSITÄT DARMSTADT

Yong Cheng[1], Marius Pesavento[2], Marc Pfetsch[3]

[1] Bell Labs, Alcatel-Lucent, Stuttgart, Germany

[2] Communication Systems Group, TU Darmstadt

[3] Discrete Optimization Group, TU Darmstadt

Key goals of the tutorial

TECHNISCHE UNIVERSITÄT DARMSTADT

To learn . . .

- . . . about **applications** in signal processing and communications in which mixed-integer programming is important.
- . . . **modelling problems** in a mixed-integer framework.
- . . . the basic techniques and strategies for computing **optimal solutions**.
- . . . **customizing solution strategies** for applications in signal processing and communications.
- . . . about **software tools** and solvers available.
- . . . examples of fast heuristic algorithms.

What this course cannot provide:

- a general introduction to mathematical optimization.
- an exhaustive overview over the field of mixed-integer programming.

Outline and schedule

TECHNISCHE UNIVERSITÄT DARMSTADT

Part I. [1.30pm] **Basic concepts** *(Marius Pesavento)*

- Overview and applications
- Introduction: Basic concepts (Examples 1 and 2)
  - branch-and-bound, continuous relaxation, . . .
  - cuts, Big-M, branch-and-cut, . . .
  - branching priorities, branching directions, . . .

Coffee break [3.00pm]

Part II. [3.30pm] **Software tools** *(Yong Cheng)*

Part III. [4.00pm] **Application examples**

- Example 3: Admission control and downlink beamforming
- Example 4: Discrete rate adaptation
- Example 5: Codebook-based beamforming

End [5.00pm]

Part I

Basic concepts

Outline

TECHNISCHE UNIVERSITÄT DARMSTADT

What is mixed-integer programming?

TECHNISCHE UNIVERSITÄT DARMSTADT

Mixed-integer (nonlinear) programming (MINLP) deals with optimization problems in which some variables are required to attain only discrete (binary or integer) values:

$$\begin{aligned} \min_{\mathbf{x}} \ & f(\mathbf{x}) \\ \text{s.t. } & \mathbf{g}(\mathbf{x}) \leq 0 \\ & \mathbf{x} \in \mathbb{Z}^p \times \mathbb{R}^{n-p}. \end{aligned}$$

Special case: Mixed-integer linear programming (MILP):

$$\begin{aligned} \min_{\mathbf{x}} \ & \mathbf{c}^\top \mathbf{x} \\ \text{s.t. } & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \in \mathbb{Z}^p \times \mathbb{R}^{n-p}. \end{aligned}$$

Motivation:

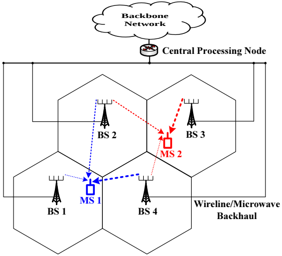
Important applications of mixed-integer programming

Practical optimization problems in signal processing and communication involve both continuous and discrete optimization variables. Resource optimization for communication networks naturally involves integer decision making.

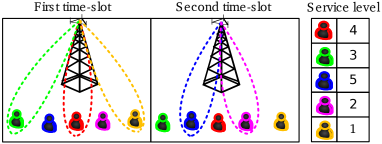
By problem nature:

- Selection problems of undividable quantities: served users, Tx/Rx antenna, CoMP clusters, network topologies, routing paths, . . .

cooperative base stations



user scheduling



Motivation:

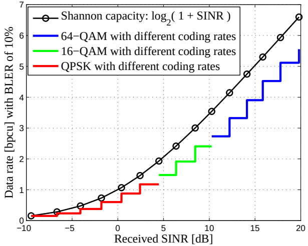
Important applications of mixed-integer programming

TECHNISCHE UNIVERSITÄT DARMSTADT

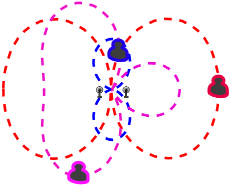
Other are home-made, e.g., imposed by standards:

- Allocation problems: adaptive coding and modulation, codebook based precoding, resource block (time-frequency) allocation, . . .
- Transmission modes: format (open-loop / closed loop spatial MUX, STBC, port-5 beamforming, etc.), number of layers, transmission/decoding strategies in MU systems (single user, SIC, ordering), Tx power, report generation (K-best frequency + layers + MCS + precoder, etc.)

rate adaptation

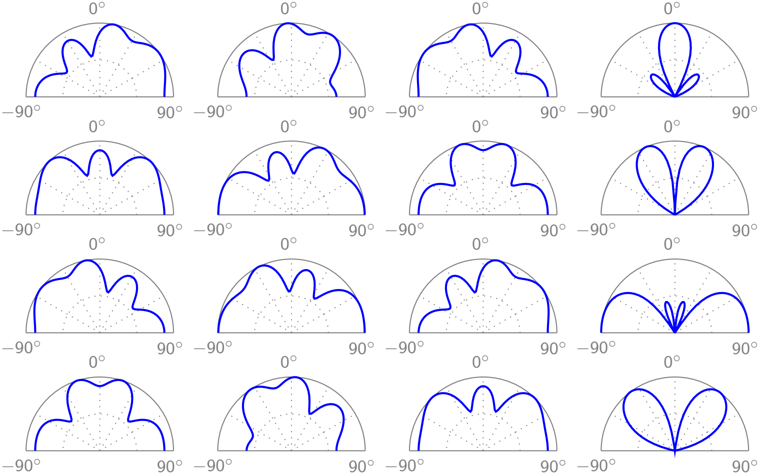


codebook based beamforming



LTE precoder (beamformer) codebook

for 4 Tx antennas (as defined in the standard)

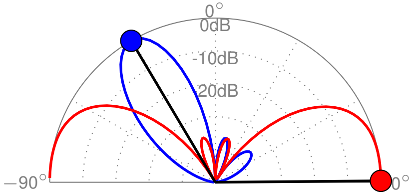


Adaptive Beamforming

Continuous vs. Optimal Codebook-Based

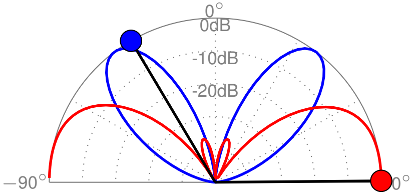
Optimal Continuous

Sum-Power: 0.091 [dB]



Optimal Codebook-Based

Sum-Power: 1.786 [dB]



🔴: User 1

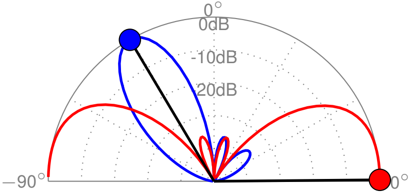
🔵: User 2

Adaptive Beamforming

Continuous vs. Optimal Codebook-Based

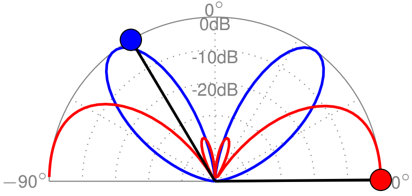
Optimal Continuous

Sum-Power: 0.091 [dB]



Optimal Codebook-Based

Sum-Power: 1.786 [dB]



: User 1

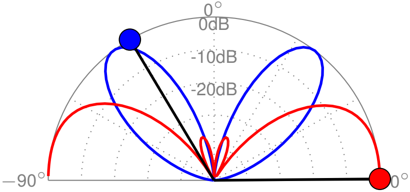
: User 2

Adaptive Beamforming

Projection vs. Optimal Codebook-Based

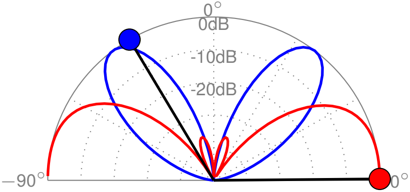
Projection on Codebook

Sum-Power: 1.786 [dB]



Optimal Codebook-Based

Sum-Power: 1.786 [dB]



: User 1

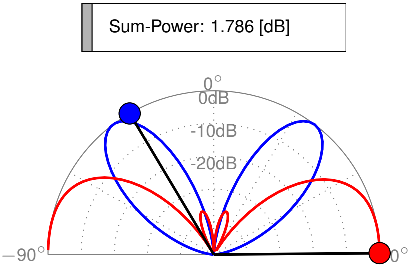
: User 2

Adaptive Beamforming

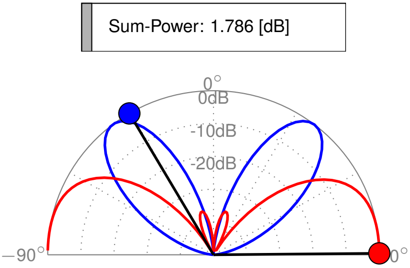
Projection vs. Optimal Codebook-Based

TECHNISCHE UNIVERSITÄT DARMSTADT

Projection on Codebook



Optimal Codebook-Based



●: User 1

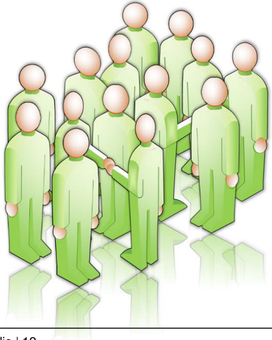
●: User 2

Why are mixed-integer programs difficult?

Combinatorial nature of the problems

TECHNISCHE UNIVERSITÄT DARMSTADT

- Each discrete variable may belong to a finite or discrete set.
  Examples: $\{0, 1\}$, $\{0, 1, 2, \ldots, k\}$, $\mathbb{Z}_+$, $\mathbb{Z}$.
- The number of combinations is exponential, e.g., $|\{\mathbf{x} \in \{0, 1\}^n\}| = 2^n$.
- Example: Handshakes



Why are mixed-integer programs difficult?

Combinatorial nature of the problems

- Each discrete variable may belong to a finite or discrete set.
  Examples: $\{0, 1\}$, $\{0, 1, 2, \dots, k\}$, $\mathbb{Z}_+$, $\mathbb{Z}$.
- The number of combinations is exponential, e.g., $|\{\mathbf{x} \in \{0, 1\}^n\}| = 2^n$.
- Example: Handshakes

Why are mixed-integer programs difficult?

Combinatorial nature of the problems

TECHNISCHE UNIVERSITÄT DARMSTADT

- Each discrete variable may belong to a finite or discrete set.
  Examples: $\{0, 1\}$, $\{0, 1, 2, \ldots, k\}$, $\mathbb{Z}_+$, $\mathbb{Z}$.
- The number of combinations is exponential, e.g., $|\{\mathbf{x} \in \{0, 1\}^n\}| = 2^n$.
- Example: Handshakes

Why are mixed-integer programs difficult?

Combinatorial nature of the problems

TECHNISCHE UNIVERSITÄT DARMSTADT

- Each discrete variable may belong to a finite or discrete set. Examples: $\{0,1\}$, $\{0,1,2,\dots,k\}$, $\mathbb{Z}_+$, $\mathbb{Z}$.
- The number of combinations is exponential, e.g., $|\{\mathbf{x} \in \{0,1\}^n\}| = 2^n$.
- Example: Handshakes

Why are mixed-integer programs difficult?

Examples

TECHNISCHE UNIVERSITÄT DARMSTADT

1. Example

$$2\,x_1 + 2\,x_2 + 3\,x_3 + 5\,x_4 + 7\,x_5 + 7\,x_6 = 18, \qquad x_1, \dots, x_6 \in \{0, 1\}.$$

Why are mixed-integer programs difficult?

Examples

TECHNISCHE UNIVERSITÄT DARMSTADT

1. Example

$$2\,x_1 + 2\,x_2 + 3\,x_3 + 5\,x_4 + 7\,x_5 + 7\,x_6 = 18, \qquad x_1, \dots, x_6 \in \{0, 1\}.$$

Solution: $18 = 2 + 2 + 7 + 7 \quad \Rightarrow \quad \mathbf{x} = (1, 1, 0, 0, 1, 1)^\top.$

Why are mixed-integer programs difficult?

Examples

TECHNISCHE UNIVERSITÄT DARMSTADT

1. Example

$$2\,x_1 + 2\,x_2 + 3\,x_3 + 5\,x_4 + 7\,x_5 + 7\,x_6 = 18, \qquad x_1, \ldots, x_6 \in \{0, 1\}.$$

Solution: $18 = 2 + 2 + 7 + 7 \quad \Rightarrow \quad \mathbf{x} = (1, 1, 0, 0, 1, 1)^\top.$

2. Example

$$2\,x_1 + 2\,x_2 + 3\,x_3 + 5\,x_4 + 7\,x_5 + 7\,x_6 = 13, \qquad x_1, \ldots, x_6 \in \{0, 1\}.$$

Why are mixed-integer programs difficult?

Examples

TECHNISCHE UNIVERSITÄT DARMSTADT

1. Example

$$2\,x_1 + 2\,x_2 + 3\,x_3 + 5\,x_4 + 7\,x_5 + 7\,x_6 = 18, \qquad x_1, \ldots, x_6 \in \{0, 1\}.$$

Solution: $18 = 2 + 2 + 7 + 7 \quad \Rightarrow \quad \mathbf{x} = (1, 1, 0, 0, 1, 1)^\top.$

2. Example

$$2\,x_1 + 2\,x_2 + 3\,x_3 + 5\,x_4 + 7\,x_5 + 7\,x_6 = 13, \qquad x_1, \ldots, x_6 \in \{0, 1\}.$$

Does not have a solution!

Why are mixed-integer programs difficult?

Examples

TECHNISCHE UNIVERSITÄT DARMSTADT

1. Example

$$2\,x_1 + 2\,x_2 + 3\,x_3 + 5\,x_4 + 7\,x_5 + 7\,x_6 = 18, \qquad x_1, \dots, x_6 \in \{0, 1\}.$$

Solution: $18 = 2 + 2 + 7 + 7 \quad \Rightarrow \quad \mathbf{x} = (1, 1, 0, 0, 1, 1)^\top$.

2. Example

$$2\,x_1 + 2\,x_2 + 3\,x_3 + 5\,x_4 + 7\,x_5 + 7\,x_6 = 13, \qquad x_1, \dots, x_6 \in \{0, 1\}.$$

Does not have a solution!
Why? $\Rightarrow$ Enumeration …

Why are mixed-integer programs difficult?

Examples

TECHNISCHE UNIVERSITÄT DARMSTADT

1. Example

$$2\,x_1 + 2\,x_2 + 3\,x_3 + 5\,x_4 + 7\,x_5 + 7\,x_6 = 18, \qquad x_1, \dots, x_6 \in \{0, 1\}.$$

Solution: $18 = 2 + 2 + 7 + 7 \quad \Rightarrow \quad \mathbf{x} = (1, 1, 0, 0, 1, 1)^\top.$

2. Example

$$2\,x_1 + 2\,x_2 + 3\,x_3 + 5\,x_4 + 7\,x_5 + 7\,x_6 = 13, \qquad x_1, \dots, x_6 \in \{0, 1\}.$$

Does not have a solution!
Why? $\Rightarrow$ Enumeration ...
Much more complicated for many more variables ...

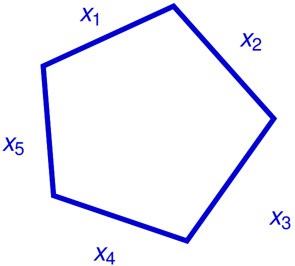
Why are mixed-integer programs difficult?

Mathematical structure

Example: n odd

$$
\begin{array}{llllll}
\max & x_1 + \cdots + x_n & & & & \\
& x_1 + x_2 & & & \leq & 1 \\
& & x_2 + x_3 & & \leq & 1 \\
& & \vdots & & \vdots & \\
& & & x_{n-1} + x_n & \leq & 1 \\
& x_1 + & & x_n & \leq & 1 \\
& x_1, \ldots, x_n \in \{0, 1\} & & & &
\end{array}
$$

- The feasible set is non-convex.
- They are NP-hard.
- Much different from continuous problems.



Solution of relaxation
(ignore integrality conditions):

$$x_1 = \cdots = x_n = \frac{1}{2}.$$

Does not tell anything about integer program.

Why are mixed-integer programs difficult?

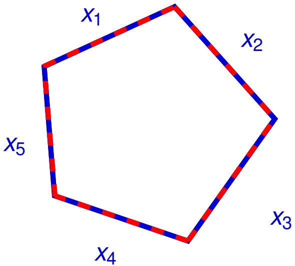
Mathematical structure

TECHNISCHE UNIVERSITÄT DARMSTADT

Example: n odd

$$
\begin{aligned}
\max\ & x_1 + \cdots + x_n \\
& x_1 + x_2 \leq 1 \\
& x_2 + x_3 \leq 1 \\
& \vdots \\
& x_{n-1} + x_n \leq 1 \\
& x_1 + x_n \leq 1 \\
& x_1, \ldots, x_n \in \{0, 1\}
\end{aligned}
$$

- The feasible set is non-convex.
- They are NP-hard.
- Much different from continuous problems.



Solution of relaxation (ignore integrality conditions):

$$x_1 = \cdots = x_n = \frac{1}{2}.$$

Does not tell anything about integer program.

Why are mixed-integer programs difficult?

Mathematical structure

TECHNISCHE UNIVERSITÄT DARMSTADT

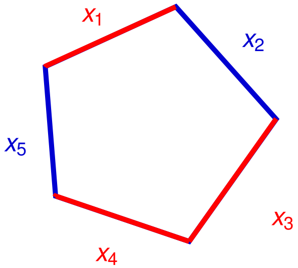
Example: n odd

$$\max\ x_1 + \cdots + x_n$$

$$\begin{array}{llll} x_1 + x_2 & & & \leq 1 \\ & x_2 + x_3 & & \leq 1 \\ & \vdots & & \vdots \\ & & x_{n-1} + x_n & \leq 1 \\ x_1 + & & x_n & \leq 1 \end{array}$$

$$x_1, \ldots, x_n \in \{0, 1\}$$

- The feasible set is non-convex.
- They are NP-hard.
- Much different from continuous problems.



Solution of relaxation (ignore integrality conditions):

$$x_1 = \cdots = x_n = \tfrac{1}{2}.$$

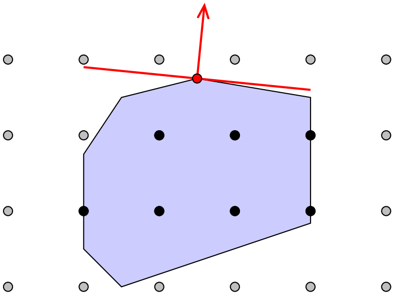
Does not tell anything about integer program.

Outline

TECHNISCHE UNIVERSITÄT DARMSTADT

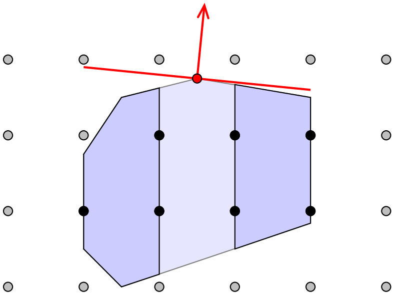
Idea of branch-and-cut

Branch-and-bound



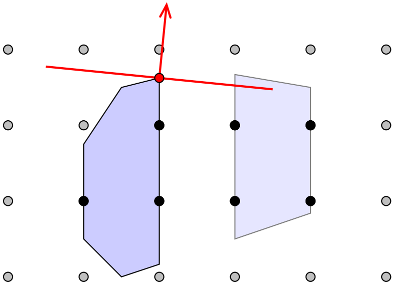
Idea of branch-and-cut

Branch-and-bound



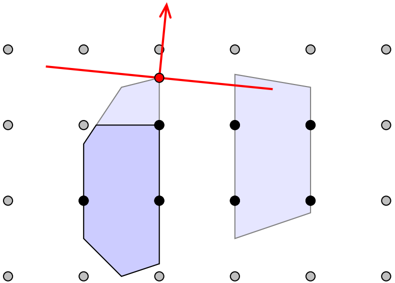
Idea of branch-and-cut

Branch-and-bound



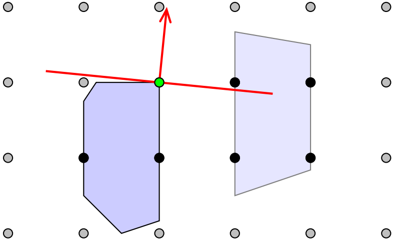
Idea of branch-and-cut

Branch-and-bound



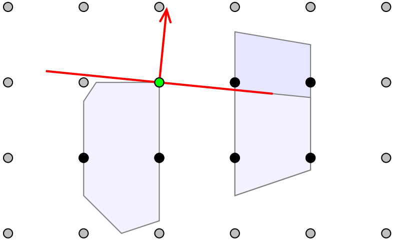
Idea of branch-and-cut

Branch-and-bound



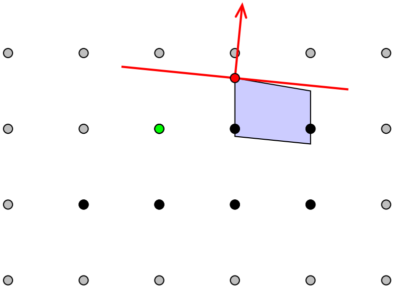
Idea of branch-and-cut

Branch-and-bound



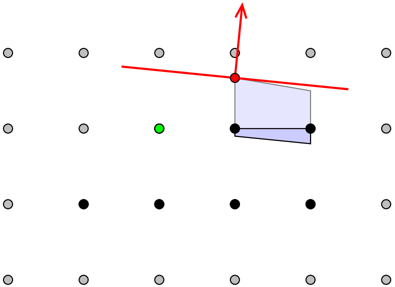
Idea of branch-and-cut

Branch-and-bound



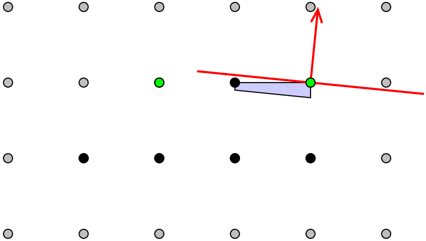
Idea of branch-and-cut

Branch-and-bound



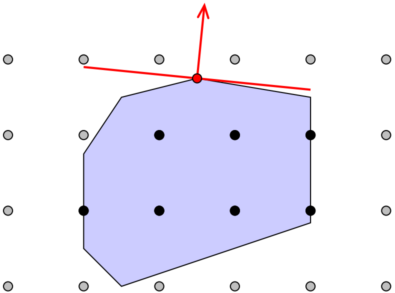
Idea of branch-and-cut

Branch-and-bound



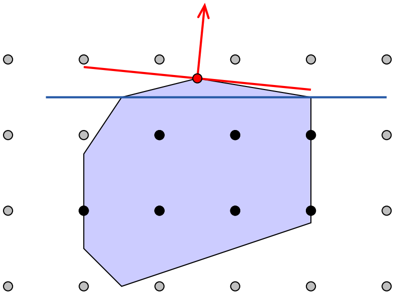
Idea of branch-and-cut

Cutting planes



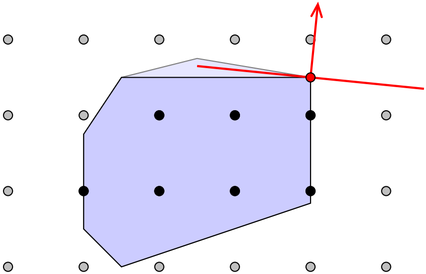
Idea of branch-and-cut

Cutting planes



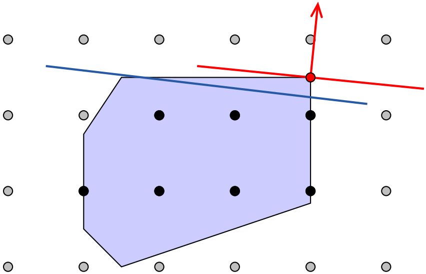
Idea of branch-and-cut

Cutting planes



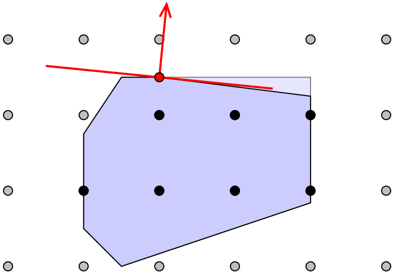
Idea of branch-and-cut

Cutting planes



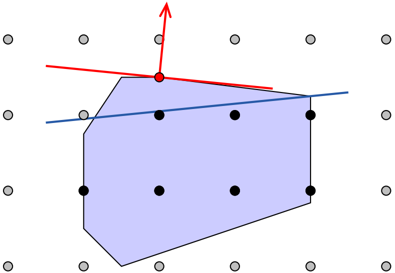
Idea of branch-and-cut

Cutting planes



Idea of branch-and-cut

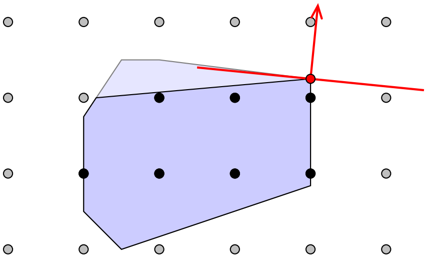
Cutting planes



Idea of branch-and-cut

Cutting planes

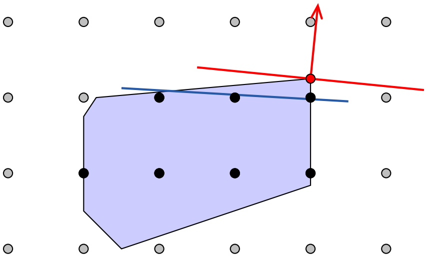
TECHNISCHE UNIVERSITÄT DARMSTADT



Idea of branch-and-cut

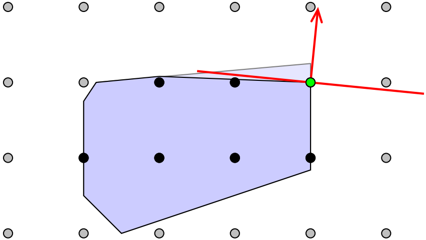
Cutting planes

TECHNISCHE UNIVERSITÄT DARMSTADT



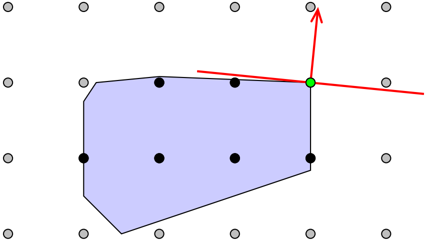
Idea of branch-and-cut

Cutting planes



Idea of branch-and-cut

Cutting planes



Outline

TECHNISCHE UNIVERSITÄT DARMSTADT

Example 1:

Maximum Likelihood (ML) MIMO detector

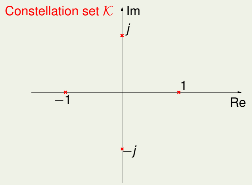
TECHNISCHE UNIVERSITÄT DARMSTADT

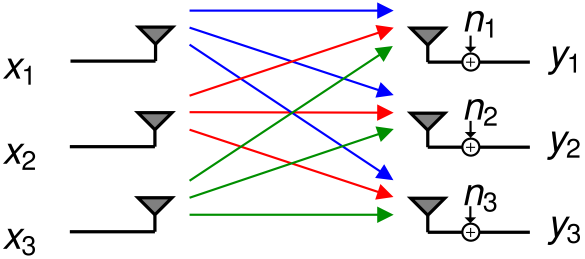
Signal model (3 $\times$ 3) MIMO system

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}; \quad \mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}; \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}; \quad \mathbf{n} = \begin{bmatrix} n_1 \\ n_2 \\ n_3 \end{bmatrix};$$

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}$$

Constellation symbols:



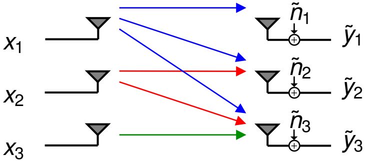


Pre-Processing

TECHNISCHE UNIVERSITÄT DARMSTADT

QR-decomposition: $\mathbf{H} = \mathbf{QL}$ with lower triangular $\mathbf{L}$ and unitary $\mathbf{Q}$,

$$\tilde{\mathbf{y}} = \mathbf{Lx} + \tilde{\mathbf{n}}$$



where $\tilde{\mathbf{y}} = \mathbf{Q}^H\mathbf{y}$ and $\tilde{\mathbf{n}} = \mathbf{Q}^H\mathbf{n}$.

Performance metric:

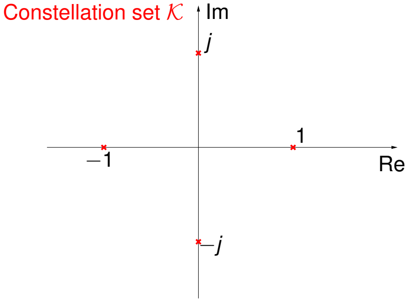
$$M_{\mathrm{ML}}(\mathbf{x}) = \|\mathbf{y} - \mathbf{Hx}\|^2 = \|\tilde{\mathbf{y}} - \mathbf{Lx}\|^2 = \sum_{\ell=1}^{M} \underbrace{\left\| \tilde{y}_\ell - \sum_{k=1}^{\ell} [\mathbf{L}]_{\ell,k} x_k \right\|^2}_{\ell\text{th summand is non-negative and depends only on } x_1,\ldots,x_\ell.}$$

Maximum Likelihood (ML) MIMO detector

Vector detection

$$M^{\star}_{\text{ML}} = \min_{\mathbf{x}\in\mathcal{K}^M} \left\|\mathbf{y} - \mathbf{H}\mathbf{x}\right\|^2 = \min_{\mathbf{x}\in\mathcal{K}^M} \left\|\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}\right\|^2 = \min_{\{x_i\in\mathcal{K}\}_{i=1}^M} \sum_{\ell=1}^{M} \left\|\tilde{y}_\ell - \sum_{k=1}^{\ell}[\mathbf{L}]_{\ell,k}x_k\right\|^2$$

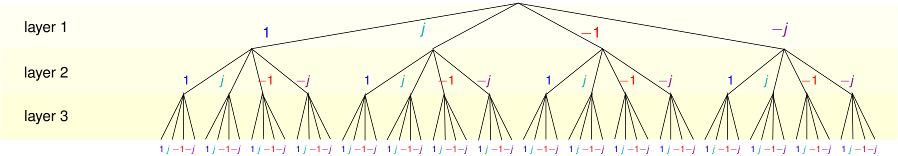
Rotated 4-QAM constellation: $x_k \in \mathcal{K} \coloneqq \{e^{j\frac{\pi}{2}}, e^{j\pi}, e^{j\frac{3\pi}{2}}, e^{j2\pi}\}$



Full tree-search

TECHNISCHE UNIVERSITÄT DARMSTADT

Brute force search: $|\mathcal{K}|^M$ leaf nodes to be visited.



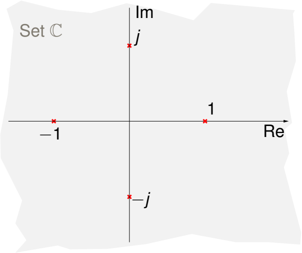
Zero-forcing detector

Continuous relaxation

TECHNISCHE UNIVERSITÄT DARMSTADT

Replace symbol vector constellation $\mathcal{K}^M = \{e^{j\frac{\pi}{2}}, e^{j\pi}, e^{j\frac{3\pi}{2}}, e^{j2\pi}\}^M$ by $\mathbb{C}^M$.

$$M_{\mathrm{ZF}}^{\star} = \min_{\mathbf{x}\in\mathbb{C}^M} \|\mathbf{y} - \mathbf{Hx}\|^2 = \min_{\mathbf{x}\in\mathbb{C}^M} \|\tilde{\mathbf{y}} - \mathbf{Lx}\|^2 = \min_{\{x_i\in\mathbb{C}\}_{i=1}^M} \sum_{\ell=1}^{M} \left\|\tilde{y}_\ell - \sum_{k=1}^{\ell}[\mathbf{L}]_{\ell,k} x_k\right\|^2$$



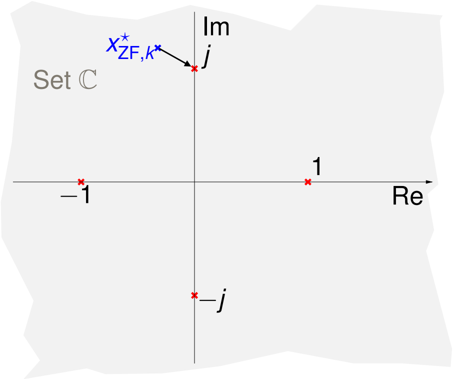
Optimal solution: $\mathbf{x}_{\mathrm{ZF}}^{*} = (\mathbf{H}^H\mathbf{H})^{-1}\mathbf{H}^H\mathbf{y}$.

Zero-forcing detector

Continuous relaxation

Hard-decision demodulation: Optimal ZF solution vector $\mathbf{x}^{\star}_{\mathrm{ZF}} \in \mathbb{C}^M$ must be mapped back to $\mathcal{K}^M$ using decision operator $\lceil \cdot \rfloor$.

$M^{\star}_{\mathrm{ZF}} \leq M^{\star}_{\mathrm{ML}}$



Zero-forcing detector

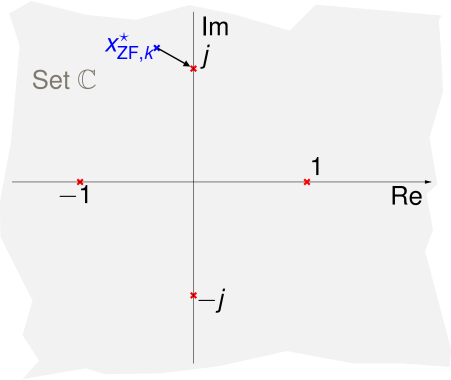
Continuous relaxation

Hard-decision demodulation: Optimal ZF solution vector $\mathbf{x}^\star_{\text{ZF}} \in \mathbb{C}^M$ must be mapped back to $\mathcal{K}^M$ using decision operator $\lceil \cdot \rfloor$.

$$M^\star_{\text{ZF}} \leq M^\star_{\text{ML}}$$

$$M_{\text{ML}}(\lceil \mathbf{x}^\star_{\text{ZF}} \rfloor) \geq M^\star_{\text{ML}}$$

Equality holds if $\lceil \mathbf{x}^\star_{\text{ZF}} \rfloor = \mathbf{x}^\star_{\text{ML}}$.



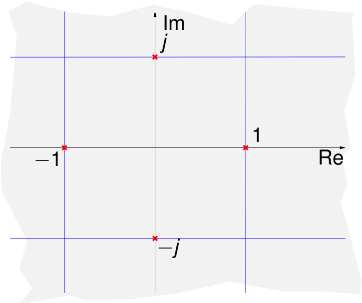
Continuous relaxation

Confine solution to the set:

$$x_k \in \square := \{x_k \,|\, -1 \leq \mathrm{Re}(x_k) \leq 1 \quad \text{and} \quad -1 \leq \mathrm{Im}(x_k) \leq 1\}$$

$$M^\star_\square = \min_{\substack{\{|\mathrm{Re}(x_i)| \leq 1 \,\wedge \\ |\mathrm{Im}(x_i)| \leq 1\}_{i=1}^M}} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 = \min_{\substack{\{|\mathrm{Re}(x_i)| \leq 1 \,\wedge \\ |\mathrm{Im}(x_i)| \leq 1\}_{i=1}^M}} \sum_{\ell=1}^{M} \left\| \tilde{y}_\ell - \sum_{k=1}^{\ell} [\mathbf{L}]_{\ell,k} x_k \right\|^2$$

$$M^\star_{\mathrm{ZF}} \leq M^\star_\square \leq M^\star_{\mathrm{ML}}$$



Continuous relaxation

TECHNISCHE UNIVERSITÄT DARMSTADT

Confined solution to the set:

$$x_k \in \square = \{x_k \mid -1 \leq \mathrm{Re}(x_k) \leq 1 \quad \text{and} \quad -1 \leq \mathrm{Im}(x_k) \leq 1\}$$

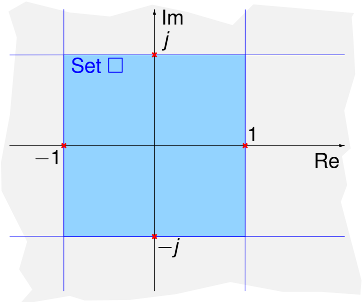
$$M_{\square}^{\star} = \min_{\substack{\{|\mathrm{Re}(x_i)| \leq 1 \wedge \\ |\mathrm{Im}(x_i)| \leq 1\}_{i=1}^{M}}} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 = \min_{\substack{\{|\mathrm{Re}(x_i)| \leq 1 \wedge \\ |\mathrm{Im}(x_i)| \leq 1\}_{i=1}^{M}}} \sum_{\ell=1}^{M} \left\| \tilde{y}_\ell - \sum_{k=1}^{\ell} [\mathbf{L}]_{\ell,k} x_k \right\|^2$$

$$M_{\mathrm{ZF}}^{\star} \leq M_{\square}^{\star} \leq M_{\mathrm{ML}}^{\star}$$

$$M_{\mathrm{ML}}(\lceil \mathbf{x}_{\square}^{\star} \rfloor) \geq M_{\mathrm{ML}}^{\star}$$

Equality holds if $\lceil \mathbf{x}_{\square}^{\star} \rfloor = \mathbf{x}_{\mathrm{ML}}^{\star}$.

Can we do better?



Tightened continuous relaxation

TECHNISCHE UNIVERSITÄT DARMSTADT

Confined solution to the set:

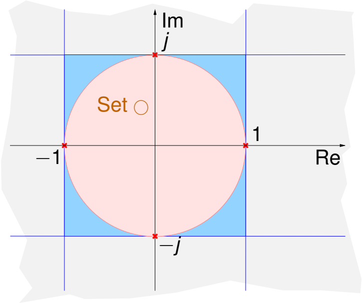
$$x_k \in \bigcirc := \{x_k \mid |x_k| \leq 1\}$$

$$M^\star_\bigcirc = \min_{\{|x_i| \leq 1\}_{i=1}^M} \|\mathbf{y} - \mathbf{Hx}\|^2 = \min_{\{|x_i| \leq 1\}_{i=1}^M} \sum_{\ell=1}^{M} \left\| \tilde{y}_\ell - \sum_{k=1}^{\ell} [\mathbf{L}]_{\ell,k} x_k \right\|^2$$

$$M^\star_{ZF} \leq M^\star_\square \leq M^\star_\bigcirc \leq M^\star_{ML}$$

$$M_{ML}(\lceil \mathbf{x}^\star_\bigcirc \rfloor) \geq M^\star_{ML}$$

Equality holds if $\lceil \mathbf{x}^\star_\bigcirc \rfloor = \mathbf{x}^\star_{ML}$.



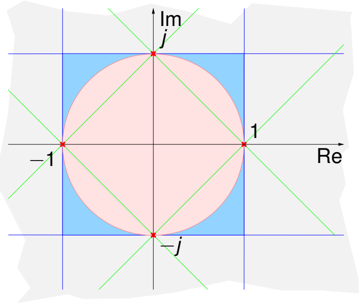
Further tightened continuous relaxation

Confined solution to the set:

$$x_k \in \lozenge \coloneqq \left\{ x_k \;\middle|\; |\mathrm{Re}(x_i)| + |\mathrm{Im}(x_i)| \leq 1 \right\}$$

$$M_{\lozenge}^{\star} = \min_{\{|\mathrm{Re}(x_i)|+|\mathrm{Im}(x_i)|\leq 1\}_{i=1}^{M}} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 = \min_{\{|\mathrm{Re}(x_i)|+|\mathrm{Im}(x_i)|\leq 1\}_{i=1}^{M}} \sum_{\ell=1}^{M} \left\| \tilde{y}_\ell - \sum_{k=1}^{\ell} [\mathbf{L}]_{\ell,k} x_k \right\|^2$$

$$M_{\mathrm{ZF}}^{\star} \leq M_{\square}^{\star} \leq M_{\bigcirc}^{\star} \leq M_{\lozenge}^{\star} \leq M_{\mathrm{ML}}^{\star}$$



Further tightened continuous relaxation

Confined solution to the set:

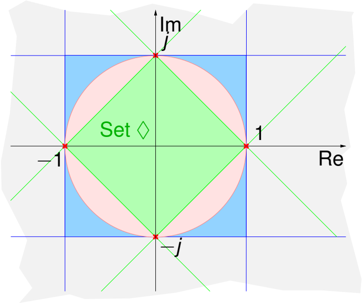
$$x_k \in \Diamond = \{x_k \mid |\mathrm{Re}(x_i)| + |\mathrm{Im}(x_i)| \leq 1\} \qquad \text{(convex hull of } \mathcal{K}\text{)}$$

$$M_{\Diamond}^{\star} = \min_{\{|\mathrm{Re}(x_i)|+|\mathrm{Im}(x_i)|\leq 1\}_{i=1}^{M}} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 = \min_{\{|\mathrm{Re}(x_i)|+|\mathrm{Im}(x_i)|\leq 1\}_{i=1}^{M}} \sum_{\ell=1}^{M} \left\| \tilde{y}_\ell - \sum_{k=1}^{\ell} [\mathbf{L}]_{\ell,k} x_k \right\|^2$$

$$M_{\mathrm{ZF}}^{\star} \leq M_{\square}^{\star} \leq M_{\bigcirc}^{\star} \leq M_{\Diamond}^{\star} \leq M_{\mathrm{ML}}^{\star}$$

$$M_{\mathrm{ML}}(\lceil \mathbf{x}_{\Diamond}^{\star} \rfloor) \geq M_{\mathrm{ML}}^{\star}$$

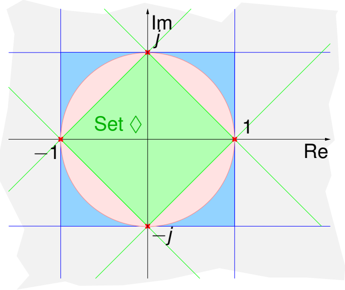
Equality holds if $\lceil \mathbf{x}_{\Diamond}^{\star} \rfloor = \mathbf{x}_{\mathrm{ML}}^{\star}$.



Cuts

TECHNISCHE UNIVERSITÄT DARMSTADT

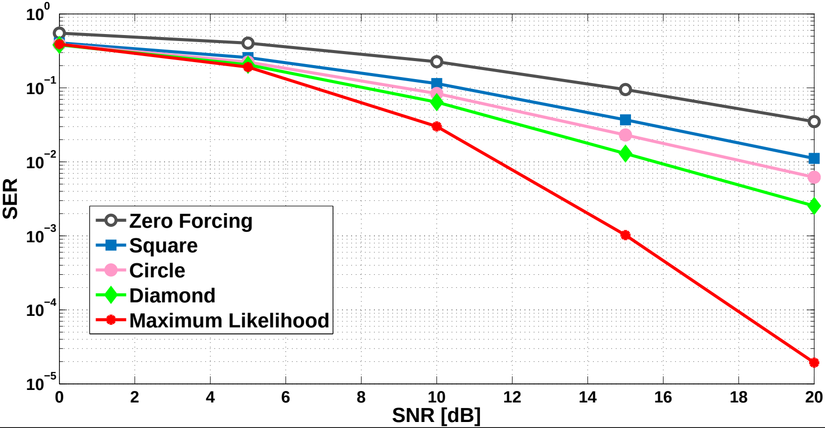
- The constraints $|\mathrm{Re}(x_i)| + |\mathrm{Im}(x_i)| \leq 1$ for $m = 1, \ldots, M$ are also referred to as "cuts" (cutting planes).
- Cuts are additional convex constraints added to the problem that are redundant for the original (mixed-integer) problem.
- However, these constraints reduce the feasible set of the continuous relaxation.



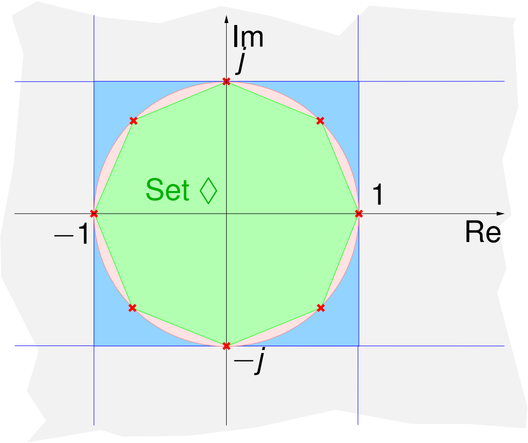
Simulation Results

Symbol Error Rate (SER) vs. Signal-to-Noise Ratio (SNR)

- 4 × 4 MIMO, rotated 4-QAM



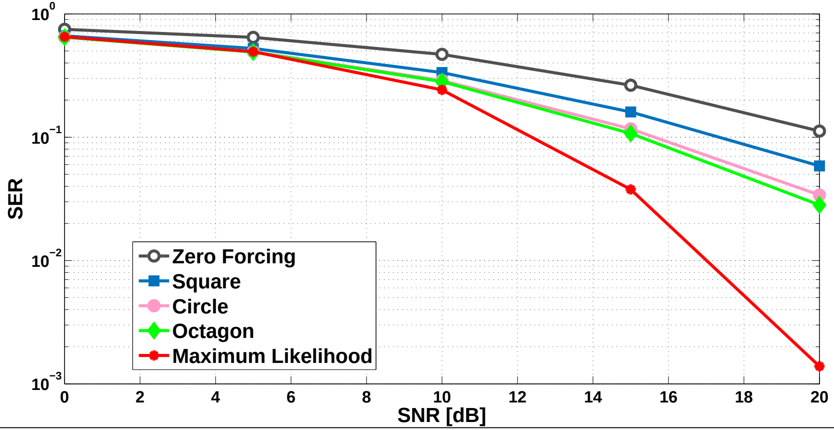
Extension to 8-PSK Modulation



Simulation Results

SER vs. SNR

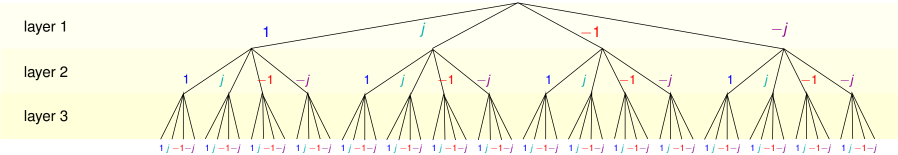
- 4×4 MIMO, 8-PSK

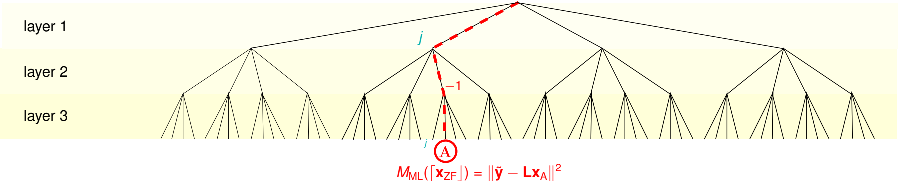


Maximum Likelihood (ML) MIMO detector

Search tree

Zero-forcing solution





Maximum Likelihood (ML) MIMO detector

Branch-and-bound

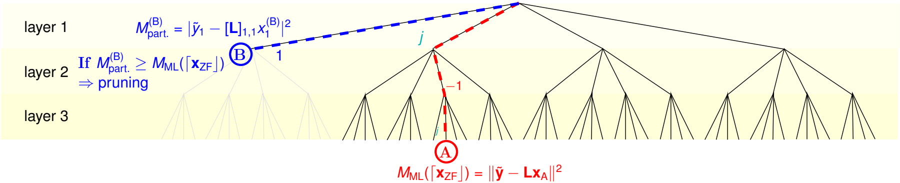
TECHNISCHE UNIVERSITÄT DARMSTADT

Sphere decoder: Transverse through the tree, use partial metric to prune tree.

$$M_{\mathrm{ML}}(\mathbf{x}) = \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 = \|\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}\|^2 = \sum_{\ell=1}^{M} \left\| \tilde{y}_\ell - \sum_{k=1}^{\ell} [\mathbf{L}]_{\ell,k} x_k \right\|^2$$

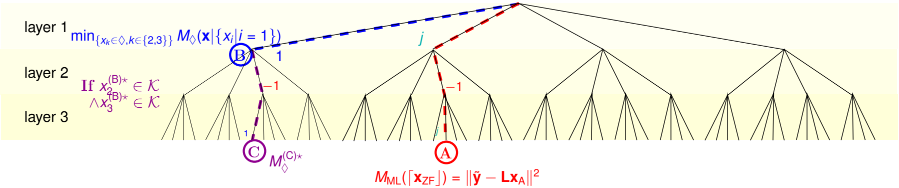
Partial metric for fixed components $\mathcal{S}_d$; w.l.o.g. $\mathcal{S}_d = \{1, \ldots, d\}$:

$$M_{\mathrm{part.}}(\mathbf{x} | \{x_i | i \in \mathcal{S}_d\}) = \underbrace{\sum_{\ell=1}^{d} \left\| \tilde{y}_\ell - \sum_{k=1}^{\ell} [\mathbf{L}]_{\ell,k} x_k \right\|^2}_{\text{partial metric}} \leq \sum_{\ell=1}^{M} \left\| \tilde{y}_\ell - \sum_{k=1}^{\ell} [\mathbf{L}]_{\ell,k} x_k \right\|^2 = M_{\mathrm{ML}}(\mathbf{x})$$



Partial continuous relaxation

Branch-and-bound: Transverse through the tree, fixing part of the variables to elements in $\mathcal{K}$ and solve continuous relaxation on remaining variables.



Local lower bound from continuous relaxation of variables not in index set $\mathcal{S}_d$:

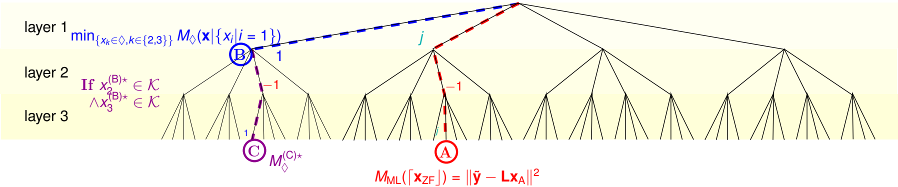
$$M_\Diamond^\star \big|_{\{x_i|i\notin\mathcal{S}_d\}} = \min_{\{x_k\in\Diamond,k\notin\mathcal{S}_d\}} M_\Diamond(\mathbf{x}|\{x_i|i\in\mathcal{S}_d\})$$

$$= \min_{\{x_k\in\Diamond,k\notin\mathcal{S}_d\}} \sum_{\ell=1}^{M} \left\| \tilde{y}_\ell - \sum_{k=1}^{\ell} [\mathbf{L}]_{\ell,k} x_k \right\|^2$$

Pruning rules

TECHNISCHE UNIVERSITÄT DARMSTADT

Branch-and-bound: Transverse through the tree, fixing part of the variables to elements in $\mathcal{K}$ and solve continuous relaxation on remaining variables.



Pruning rules obtained from relaxation: Descendent branches at a node are pruned if the continuous relaxation ...

Infeasibility: ... is infeasible (delete node). (Does not apply in this example.)
Integrality: ... yields integer-feasible solution (terminate sub-branch, save solution).
Dominance: ... yields a metric larger than best known integer-feasible solution (delete node and descendants).

Customizing branching rules

Branching variable and node selection

TECHNISCHE UNIVERSITÄT DARMSTADT

Branching variable selection:

On which variable should we branch? (branching priority, rearranging the tree!)

- generic:
  - minimum integer infeasibility (terminate sub-branches fast)
  - maximum integer infeasibility (try to improve on lower bounds)
  - infer degeneration (increase in the lower bound achieved after branching, strong branching)
- customized strategies: in MIMO example, e.g., first branch on “strongest” symbols with largest detection probability (sorted QR decomposition).

Node selection:

Which node in the tree should be treated next?

- Depth-first search (try to improve fast on global upper bound)
- Breadth-first search (try to improve on lower bounds)
- Best-first search

Customizing branching rules

Branching variable and node selection

Objectives of branching rules:

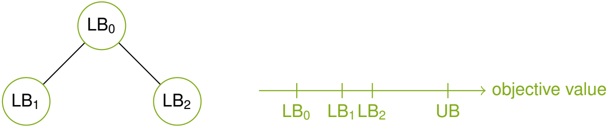
- quickly improve (increase) on the (local/global) lower bound (in minimization problems) obtained from continuous relaxation.
- quickly improve (decrease) on the global upper bound (in minimization problems).
- quickly improve number of variables that take integer values in continuous relaxation solution (integrality).
- early pruning of branches (infeasibility, integrality, dominance).

Mixed-integer programming

Lower/upper bounds

For minimization problems:

- Upper (primal) bounds arise from feasible integral solutions.
- Lower (dual) bounds arise from local relaxations.



- $LB_0 \leq LB_i$, $i = 1, 2$.
- Optimal solution lies in feasible region of active nodes.
- Global lower bound = minimal value of all local lower bounds of active nodes.

Mixed-integer programming

The integrality gap

The **integrality gap** is defined as the relative distance between the best known upper bound and the global lower bound.

Mixed-integer programming

The integrality gap

The **integrality gap** is defined as the relative distance between the best known upper bound and the global lower bound.

For minimization problems: $$\eta \coloneqq \frac{\mathrm{UB} - \mathrm{LB}_{\mathrm{global}}}{\mathrm{UB}} = 1 - \frac{\mathrm{LB}_{\mathrm{global}}}{\mathrm{UB}}.$$

For a given relative gap tolerance, e.g., $\eta_0 = 10^{-3}$, integer-feasible solution declared as optimal solution if $\eta < \eta_0$.

Mixed-integer programming

The integrality gap

The **integrality gap** is defined as the relative distance between the best known upper bound and the global lower bound.

Local lower bound obtained from continuous relaxation of variables in index set $\mathcal{S}_d$:

$$
\begin{aligned}
M^\star_\Diamond \big|_{\{x_i | i \notin \mathcal{S}_d\}} &= \min_{\{x_k \in \Diamond, k \notin \mathcal{S}_d\}} M_\Diamond(\mathbf{x} | \{x_i | i \in \mathcal{S}_d\}) \\
&= \min_{\{x_k \in \Diamond, k \notin \mathcal{S}_d\}} \sum_{\ell=1}^{M} \left\| \tilde{y}_\ell - \sum_{k=1}^{\ell} [\mathbf{L}]_{\ell,k} x_k \right\|^2 .
\end{aligned}
$$

Branch-and-cut terminates:

- if integrality gap falls below predefined threshold (optimal solution).
- if all nodes are pruned without finding a feasible solution (infeasible).
- if runtime exceeds given limit (infeasible or suboptimal solution).

Outline

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Example 2: D-sparse covariance matching

TECHNISCHE UNIVERSITÄT DARMSTADT

System model: Let $\mathbf{R} = \bar{\mathbf{A}}\bar{\mathbf{S}}\bar{\mathbf{A}}^H + q_0\mathbf{I}_K$ with $\bar{\mathbf{S}} = \mathrm{diag}(\bar{s}_1, \ldots, \bar{s}_D) \succeq \bar{s}_0\mathbf{I}_D$, where $\bar{\mathbf{A}} \in \mathbb{C}^{K\times D}$ is a given manifold matrix and $\bar{s}_0$ a pre-defined detection threshold. Let $\hat{\mathbf{R}}$ denote a finite sample estimate of $\mathbf{R}$.

Problem formulation:

$$\begin{aligned} \min_{\mathbf{p}\in\mathbb{R}_+^N, q\in\mathbb{R}_+} \quad & \mathrm{Tr}\left(\hat{\mathbf{R}} - \mathbf{A}\mathbf{P}\mathbf{A}^H - q\,\mathbf{I}\right) \\ \text{s.t.} \quad & \hat{\mathbf{R}} - \mathbf{A}\mathbf{P}\mathbf{A}^H - q\,\mathbf{I} \succeq 0 && \text{positive semi-definiteness} \\ & p_k = 0 \ \vee p_k \geq \bar{s}_0 && \text{on-off constraint} \\ & \|\mathbf{p}\|_0 = D && D\text{-sparsity} \end{aligned}$$

where $\mathbf{A} \in \mathbb{C}^{K\times N}$ is a “fat” sensing matrix with $N \gg D$ and

$$\begin{aligned} \mathbf{p} &= [p_1, p_2, \ldots, p_N]^\top \\ \mathbf{P} &= \mathrm{diag}(p_1, p_2, \ldots, p_N) \\ p_i &\geq 0; \qquad q \geq 0. \end{aligned}$$

D-sparse covariance matching

TECHNISCHE UNIVERSITÄT DARMSTADT

Problem formulation:

$$\begin{aligned} \min_{\mathbf{p}\in\mathbb{R}_+^N, q\in\mathbb{R}_+} \quad & \mathrm{Tr}\left(\hat{\mathbf{R}} - \mathbf{A}\mathbf{P}\mathbf{A}^H - q\,\mathbf{I}\right) & \\ \text{s.t.} \quad & \hat{\mathbf{R}} - \mathbf{A}\mathbf{P}\mathbf{A}^H - q\,\mathbf{I} \succeq 0 & \text{positive semi-definiteness} \\ & p_k = 0 \ \vee p_k \geq \bar{s}_0 & \text{on-off constraint} \\ & \|\mathbf{p}\|_0 = D & D\text{-sparsity} \end{aligned}$$

Introduce auxiliary variables (extended formulation)

$$s_i \geq \bar{s}_0; \qquad b_i = \begin{cases} 1, & \text{for } p_i \geq \bar{s}_0; \\ 0, & \text{for } p_i = 0. \end{cases}$$

D-sparse covariance matching

On-off constraint:

$$b_i \in \{0,1\}; \qquad s_i \geq \bar{s}_0; \qquad p_i = \begin{cases} 0, & \text{for } b_i = 0; \\ s_i, & \text{for } b_i = 1. \end{cases}$$

Mixed-integer semi-definite programming reformulation:

$$\begin{aligned}
\min_{\{(b_i,s_i,p_i)\}_{i=1}^N,q} \quad & \operatorname{Tr}\left(\hat{\mathbf{R}} - \mathbf{APA}^H - q\,\mathbf{I}\right) & \\
\text{s.t.} \quad & \hat{\mathbf{R}} - \mathbf{APA}^H - q\,\mathbf{I} \succeq 0 & \text{positive semi-definiteness} \\
& \sum_{k=1}^{N} b_k = D, & D\text{-sparsity} \\
& p_i = b_i s_i, \qquad b_i \in \{0,1\}, & \text{on-off constraint} \\
& s_i \geq \bar{s}_0, \qquad q \geq 0 & i = 1, \ldots, N.
\end{aligned}$$

Challenge: The bilinear term $b_i s_i$ is non-convex even after continuous relaxation.

The BIG-M

TECHNISCHE UNIVERSITÄT DARMSTADT

Reformulation of on-off constraints ($p_i = b_i s_i$):

$$I: \quad (b_i - 1)M_i + s_i \leq p_i \leq s_i$$
$$II: \quad 0 \leq p_i \leq b_i M_i$$

for sufficiently large constant M_i which upper-bounds s_i.

Case 1: $b_i = 0 \Rightarrow p_i = 0$

$$I: \quad \underbrace{-M_i + s_i}_{<0} \leq p_i \leq s_i \text{ (automatic)}$$

$$II: \quad 0 \leq p_i \leq 0 \Rightarrow p_i = 0$$

Case 2: $b_i = 1 \Rightarrow p_i = s_i$

$$I: \quad s_i \leq p_i \leq s_i \Rightarrow p_i = s_i$$

$$II: \quad 0 \leq p_i \leq M_i \text{ (automatic)}$$

The BIG-M

TECHNISCHE UNIVERSITÄT DARMSTADT

Mixed-integer reformulation:

$$\min_{\{(b_i,s_i,p_i)\}_{i=1}^N,q} \quad \mathrm{Tr}\left(\hat{\mathbf{R}} - \mathbf{APA}^H - q\,\mathbf{I}\right)$$

$$\text{s.t.} \quad \hat{\mathbf{R}} - \mathbf{APA}^H - q\,\mathbf{I} \succeq 0 \qquad \text{positive semi-definiteness}$$

$$\sum_{k=1}^{N} b_k = D, \qquad D\text{-sparsity}$$

$$(b_i - 1)M_i + s_i \leq p_i \leq s_i, \qquad \text{big-}M$$

$$0 \leq p_i \leq b_i M_i, \qquad \text{big-}M$$

$$b_i \in \{0, 1\}, \qquad s_i \geq \bar{s}_0, \qquad q \geq 0 \qquad i = 1, \ldots, N.$$

Ready to be solved using branch-and-cut.

The BIG-M

Choosing the M

Important – Choose constants M_i as small as possible:

- based on a-priori knowledge (problem specific).
- $\mathbf{R} \succeq \mathbf{APA}^H$

$$\Rightarrow \quad \mathrm{Tr}(\mathbf{R}) \geq \mathrm{Tr}\left(\mathbf{APA}^H\right) = \sum_{i=1}^{N} p_i \mathbf{a}_i^H \mathbf{a}_i \geq p_k \mathbf{a}_k^H \mathbf{a}_k, \qquad k = 1, \ldots, K.$$

$$\Rightarrow \quad \text{choose } M_k \geq \frac{\mathrm{Tr}(\mathbf{R})}{\mathbf{a}_k^H \mathbf{a}_k}.$$

- For unitary sensing matrix $\mathbf{A}$:

$$\mathbf{R} \succeq \mathbf{APA}^H$$

$$\Rightarrow \quad \mathbf{P} \preceq \mathbf{A}^H \mathbf{RA}$$

$$\Rightarrow \quad \text{choose } M_1 = M_2 = \ldots = M_N = M \geq \max_{i \in \{1,\ldots,N\}} \lambda(\mathbf{R}).$$

Outline and schedule

TECHNISCHE UNIVERSITÄT DARMSTADT

Part I. [1.30pm] Basic concepts *(Marius Pesavento)*

- Overview and applications
- Introduction: Basic concepts (Examples 1 and 2)
  - branch-and-bound, continuous relaxation,...
  - cuts, Big-M, branch-and-cut,...
  - branching priorities, branching directions,...

Coffee break [3.00pm]

Part II. [3.30pm] Software tools *(Yong Cheng)*

Part III. [4.00pm] Application examples

- Example 3: Admission control and downlink beamforming
- Example 4: Discrete rate adaptation
- Example 5: Codebook-based beamforming

End [5.00pm]

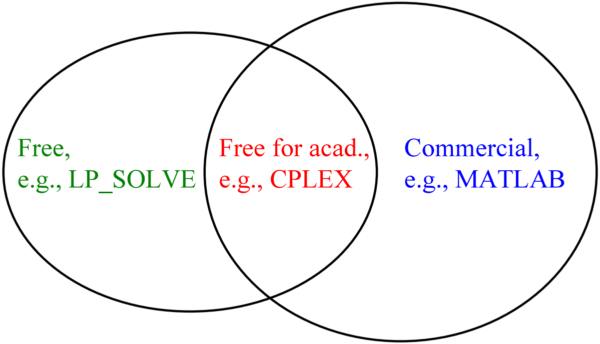
Part II

Software Tools

Software tools

Classification according to charging

- Wikipedia: List of optimization software
  http://en.wikipedia.org/wiki/List_of_optimization_software
- Hans Mittelmann: "Decision Tree for Optimization Software"
  http://plato.asu.edu/guide.html



Common MINLP solvers

Global MILP, MISOCP, and MISDP solvers

TECHNISCHE UNIVERSITÄT DARMSTADT

MILP Solvers $\{\mathbf{x} \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}; \mathbf{x} \in \mathbb{Z}^p \times \mathbb{R}^{n-p}\}$

Free:	CBC, GLPK, LP_SOLVE
Free for acad.:	CPLEX, GUROBI, MOSEK, SCIP, XPRESS
Commercial:	BARON, MATLAB (Optimization Toolbox)

MISOCP Solvers $\{(\mathbf{x}, \mathbf{y}) \mid \|\mathbf{C}_i\mathbf{x} - \mathbf{b}\|_2 \leq y_i, \forall i; \mathbf{x} \in \mathbb{Z}^p \times \mathbb{R}^{n-p}\}$

Free for acad.:	CPLEX, GUROBI, MOSEK, SCIP
Commercial:	BARON, TOMLAB (MATLAB)

MISDP Solvers $\left\{\mathbf{x} \mid \sum_{j=1}^{n} \mathbf{D}_{i,j} x_j \succeq \mathbf{0}, \forall i; \mathbf{x} \in \mathbb{Z}^p \times \mathbb{R}^{n-p}\right\}$

Free for acad.:	SCIP
Commercial:	BARON, TOMLAB (MATLAB)

Algorithms implemented in the solvers

For globally-optimal solutions

- Commonly with parallel implementations
- For dealing with integer variables
  - Branch-and-bound
  - Branch-and-cut
  - Branch-and-price
  - Branch-and-reduce
  - Branch-and-cut-and-price
- For solving continuous relaxations
  - Simplex algorithm and its variations
  - Interior-point method and its variations
- Node heuristics for generating integer-feasible solutions
  - Rounding
  - Relaxation induced neighborhood search (RINS)
  - Feasibility pump

Common parsers/modeling languages

Tools/interfaces for modeling problems

- Call MIP solvers directly
  - Examples: CPLEX, GUROBI, SCIP
- Via third-party tools/parsers:

Free for acad.:	CVX, YALMIP, AMPL, GAMS, AIMMS, MPL
Commercial:	CVX (MIP), TOMLAB (MATLAB), EXCEL

Common programming languages

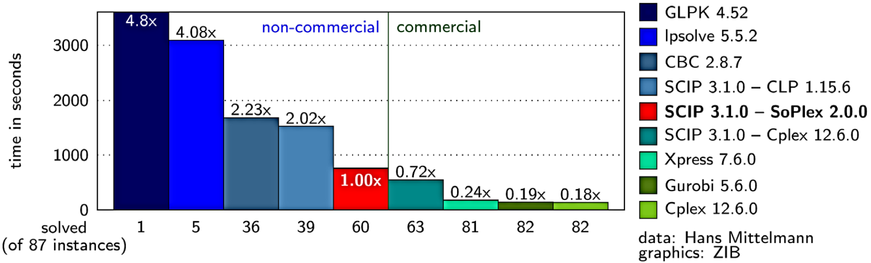
Languages for calling solvers directly

TECHNISCHE UNIVERSITÄT DARMSTADT

- There exist connectors for calling solvers directly using the following programming languages:
  - C/C++
  - .NET
  - JAVA
  - Python
  - R
  - MATLAB (Mathematica, Maple)
- Examples:
  - C++, CPLEX (w/ C++ connector)
  - MATLAB + YALMIP + CPLEX (w/ MATLAB connector)
  - MATLAB + CPLEX (w/ MATLAB connector)
  - MATLAB + YALMIP + LP_SOLVE (w/ MATLAB connector)
  - JAVA + LP_SOLVE (w/ JAVA connector)
  - MATLAB (Optimization Toolbox) + SCIP

Comparison of solvers

- MIP solver benchmark (1 thread)
  From `http://scip.zib.de/`, with 87 test problems:



- New comparison with CPLEX 12.6.1 on `http://scip.zib.de/`
- More comparisons: `http://plato.asu.edu/ftp/milpc.html`

Summary on software tools

- Select "Solver + Language + Parser" based on specific conditions/requirements:
  - Commercial vs. academic,
  - Control of solution process (e.g., adding cuts) vs. black-box,
  - Online (realtime) vs. offline.
- For easier implementation, employ parsers (modeling in math language).
- For better performance, call solvers directly (avoid introducing unnecessary optimization variables).
- Be cautious with using a large number of CPUs/threads.
- When none of the solvers working, customized implementations of the branch-and-X procedure.

Part III

Further Examples

Outline

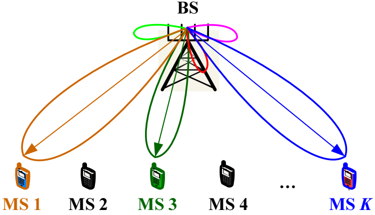
TECHNISCHE UNIVERSITÄT DARMSTADT

Example 3: Admission control and downlink beamforming

TECHNISCHE UNIVERSITÄT DARMSTADT

Motivation

- Single transmitter with N antenna elements
- K single antenna receivers
- Frequency-flat quasi-static channel $\mathbf{h}_k$, $k = 1, \ldots, K$



Reference:

E. Matskani, N.D. Sidiropoulos, Z.-Q. Luo, and L. Tassiulas, "Convex approximation techniques for joint multiuser downlink beamforming and admission control," IEEE Trans. Wireless Communications, vol. 7, no. 7, pp. 2682–2693, Jul. 2008.

Example 3: Motivation

TECHNISCHE UNIVERSITÄT DARMSTADT

- Typical joint multiuser transmit beamforming problem:

$$
\begin{aligned}
\min_{\{\mathbf{w}_k \in \mathbb{C}^N\}_{k=1}^K} \quad & \sum_{k=1}^{K} \|\mathbf{w}_k\|_2^2 \\
\text{s.t.} \quad & \sum_{k=1}^{K} \|\mathbf{w}_k\|_2^2 \leq P, \\
& \frac{|\mathbf{w}_k^H \mathbf{h}_k|^2}{\sum_{\ell \neq k} |\mathbf{w}_\ell^H \mathbf{h}_k|^2 + \sigma_k^2} \geq c_k, \qquad k = 1, \ldots, K.
\end{aligned}
$$

- Can be reformulated as a convex problem (second-order cone program)
- Easily becomes infeasible, e.g., for large number of users, high SINR target c_k, highly correlated channels, etc.

Example 3: User admission control

TECHNISCHE UNIVERSITÄT DARMSTADT

- Introduce admission control, i.e., drop some users and reformulate the problem
- Stage 1 – find the largest set of users which could be served:

$$S_o = \underset{S \subseteq \{1,\ldots,K\},\{\mathbf{w}_k \in \mathbb{C}^N\}_{k \in S}}{\operatorname{argmax}} |S|$$

$$\text{s.t.} \quad \sum_{k \in S} \|\mathbf{w}_k\|_2^2 \leq P,$$

$$\frac{|\mathbf{w}_k^H \mathbf{h}_k|^2}{\sum_{\ell \neq k, \ell \in S} |\mathbf{w}_\ell^H \mathbf{h}_k|^2 + \sigma_k^2} \geq c_k, \qquad \forall k \in S.$$

Example 3: Optimum beamformer design

TECHNISCHE UNIVERSITÄT DARMSTADT

- Stage 2 – Find optimum beamforming configuration:

$$\begin{aligned}
\min_{\{\mathbf{w}_k \in \mathbb{C}^N\}_{k \in S_o}} \quad & \sum_{k \in S_o} \|\mathbf{w}_k\|_2^2 \\
\text{s.t.} \quad & \sum_{k \in S_o} \|\mathbf{w}_k\|_2^2 \leq P, \\
& \frac{|\mathbf{w}_k^H \mathbf{h}_k|^2}{\sum_{\ell \neq k, \ell \in S_o} |\mathbf{w}_\ell^H \mathbf{h}_k|^2 + \sigma_k^2} \geq c_k, \qquad \forall k \in S_o.
\end{aligned}$$

Example 3: Joint optimization

TECHNISCHE UNIVERSITÄT DARMSTADT

- Perform admission control and optimum beamforming jointly to enhance the performance:

$$\min_{\{\mathbf{w}_k \in \mathbb{C}^N, s_k \in \{-1,+1\}\}_{k=1}^K} \quad \epsilon \sum_{k=1}^{K} \|\mathbf{w}_k\|_2^2 + (1-\epsilon) \sum_{k=1}^{K} \lambda_k (s_k+1)^2$$

$$\text{s.t.} \quad \sum_{k=1}^{K} \|\mathbf{w}_k\|_2^2 \leq P,$$

$$\frac{|\mathbf{w}_k^H \mathbf{h}_k|^2 + \delta^{-1}(s_k+1)^2}{\sum_{\ell \neq k} |\mathbf{w}_\ell^H \mathbf{h}_k|^2 + \sigma_k^2} \geq c_k, \qquad k = 1, \ldots, K,$$

- where δ is a constant (big-M) with $\delta \leq \min_k \dfrac{4c_k^{-1}}{P \max_m \|h_m\|_2^2 + \sigma_k^2}$; s_k are auxiliary variables.

Example 3: Equivalent matrix form

- Define $\mathbf{s}_k := [s_k \ 1]^T, \quad \mathbf{S}_k := \mathbf{s}_k \mathbf{s}_k^T, \quad \mathbf{W}_k := \mathbf{w}_k \mathbf{w}_k^H, \quad \mathbf{H}_k := \mathbf{h}_k \mathbf{h}_k^H$
- The previous problem can be rewritten as

$$
\begin{aligned}
\min_{\{\mathbf{W}_k, \mathbf{S}_k\}_{k=1}^K} \quad & \epsilon \sum_{k=1}^K \mathrm{Tr}(\mathbf{W}_k) + (1-\epsilon) \sum_{k=1}^K \lambda_k \mathrm{Tr}(\mathbf{1}_{2\times 2} \mathbf{S}_k) \\
\text{s.t.} \quad & \sum_{k=1}^K \mathrm{Tr}(\mathbf{W}_k) \leq P, \\
& \frac{\mathrm{Tr}(\mathbf{H}_k \mathbf{W}_k) + \delta^{-1} \mathrm{Tr}(\mathbf{1}_{2\times 2} \mathbf{S}_k)}{\sum_{\ell \neq k} \mathrm{Tr}(\mathbf{H}_k \mathbf{W}_\ell) + \sigma_k^2} \geq c_k, && \forall k \\
& \mathbf{W}_k \succeq 0, \quad \mathrm{rank}(\mathbf{W}_k) = 1, && \forall k \\
& \mathbf{S}_k \succeq 0, \quad \mathrm{rank}(\mathbf{S}_k) = 1, \quad \mathbf{S}_k(1,1) = \mathbf{S}_k(2,2) = 1, && \forall k
\end{aligned}
$$

Example 3: Semidefinate relaxation (SDR)

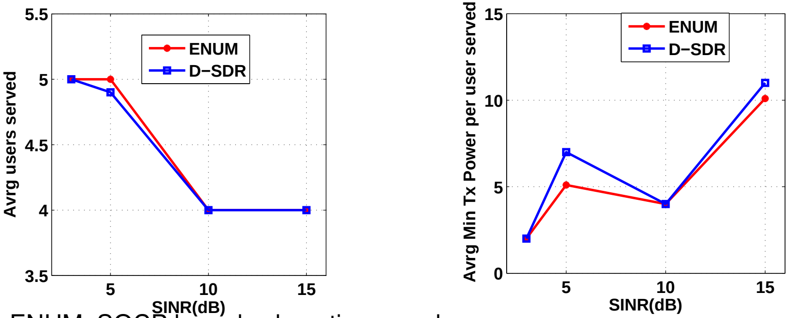
TECHNISCHE UNIVERSITÄT DARMSTADT

- Only rank-one constraints are non-convex.
- Dropping the rank-one constraints, we can reformulate the problem as

$$
\begin{aligned}
\min_{\{\mathbf{W}_k,\mathbf{S}_k\}_{k=1}^K} \quad & \epsilon \sum_{k=1}^{K} \mathrm{Tr}(\mathbf{W}_k) + (1-\epsilon) \sum_{k=1}^{K} \lambda_k \mathrm{Tr}(\mathbf{1}_{2\times 2}\mathbf{S}_k) \\
\text{s.t.} \quad & \sum_{k=1}^{K} \mathrm{Tr}(\mathbf{W}_k) \leq P, \\
& \mathrm{Tr}(\mathbf{H}_k\mathbf{W}_k) + \delta^{-1}\mathrm{Tr}(\mathbf{1}_{2\times 2}\mathbf{S}_k) \geq c_k \sum_{\ell \neq k} \mathrm{Tr}(\mathbf{H}_k\mathbf{W}_\ell) + \sigma_k^2, && \forall k \\
& \mathbf{W}_k \succeq 0, && \forall k \\
& \mathbf{S}_k \succeq 0, \quad \mathbf{S}_k(1,1) = \mathbf{S}_k(2,2) = 1, && \forall k
\end{aligned}
$$

Example 3: Simulation results

- # transmit antennas $N = 4$; # users $K = 14$; TX power $P = 100$ watts.
- Rayleigh channels with $\sigma_k^2 = 1, \forall k$; 30 Monte-Carlo runs.



ENUM: SOCP based exhaustive search

D-SDR: semidefinite relaxation based deflation (greedy algorithm)

Example 3: Further references on SDR based approach

TECHNISCHE UNIVERSITÄT DARMSTADT

- E. Matskani, N.D. Sidiropoulos, Z.-Q. Luo, and L. Tassiulas, “Convex approximation techniques for joint multiuser downlink beamforming and admission control,” IEEE Trans. Wireless Communications, vol. 7, no. 7, pp. 2682–2693, Jul. 2008.
- E. Matskani, N.D. Sidiropoulos, Z.-Q. Luo, and L. Tassiulas, “Efficient Batch and Adaptive Approximation Algorithms for Joint Multicast Beamforming and Admission Control,” IEEE Trans. Signal Processing, vol. 57, no. 12, pp. 4882–4894, Dec. 2009
- I. Mitliagkas, N.D. Sidiropoulos, A. Swami, “Joint Power and Admission Control for Ad-Hoc and Cognitive Underlay Networks: Convex Approximation and Distributed Implementation,” IEEE Trans. Wireless Communications, vol. 10, no. 12, pp. 4110–4121, December 2011
- Z. Xu, M. Hong, Z.Q. Luo, “Semidefinite Approximation for Mixed Binary Quadratically Constrained Quadratic Programs,” SIAM Journal on Optimization, vol. 24, no. 3, pp. 1265–1293, 2014

Outline

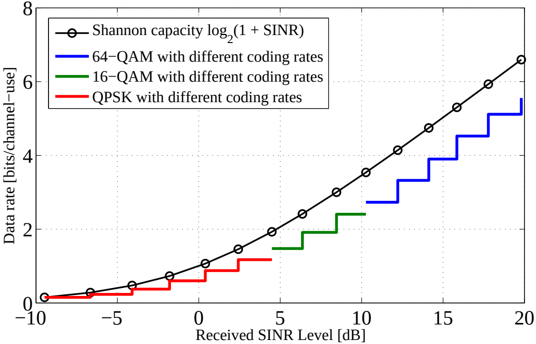
TECHNISCHE UNIVERSITÄT DARMSTADT

Example 4: Discrete rate adaptation

Motivation

TECHNISCHE UNIVERSITÄT DARMSTADT

- Adaptive modulation and coding in practical wireless systems
- Data rates determined by modulation and coding schemes (MCSs).



Note: quadrature amplitude modulation (QAM), quadrature phase-shift keying (QPSK)

Example 4: Discrete rate adaptation

MCSs defined in LTE (BLER of 10%)

TECHNISCHE UNIVERSITÄT DARMSTADT

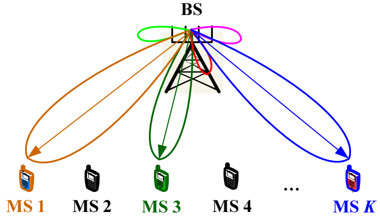
Mod. Orders	Code Rates ($\times 1024$)	Data Rates R_ℓ [bit/symbol]	SINR Thresholds Γ_ℓ [dB]
...	...	...	...
16QAM	378	1.4766	4.489
16QAM	490	1.9141	6.367
16QAM	616	2.4063	8.456
64QAM	466	2.7305	10.266
64QAM	567	3.3223	12.218
64QAM	666	3.9023	14.122
...	...	...	...

Joint discrete rate adaptation and multiuser downlink beamforming

Example 4: Discrete rate adaptation

Scenario

- One BS with M antennas, K single-antenna MSs
- L candidate MCSs, i.e., L candidate data rates



Users	MCSs
MS 1	16-QAM
MS 2	—
MS 3	QPSK
MS 4	—
⋮	⋮
MS K	16-QAM

Discrete rate adaptation $\Longleftrightarrow$ MCS assignment

- $\mathbf{h}_k \in \mathbb{C}^M$: channel vector of kth MS, known at kth MS and BS
- $\mathbf{w}_k \in \mathbb{C}^M$: beamformer of kth MS, computed at BS

Example 4: Discrete rate adaptation

System model

- BS transmitting $\sum_{j=1}^{K} \mathbf{w}_j x_j$
  - $x_j \in \mathbb{C}$: data symbol of jth MS, $\mathrm{E}(|x_j|^2) = 1$
- Received signal $y_k \in \mathbb{C}$ at kth MS:

$$y_k = \underbrace{\mathbf{h}_k^H \mathbf{w}_k x_k}_{\text{desired signal}} + \underbrace{\sum_{j=1, j\neq k}^{K} \mathbf{h}_k^H \mathbf{w}_j x_j}_{\text{interference}} + \underbrace{z_k}_{\text{noise}} .$$

- Assumptions: (i) uncorrelated data symbols and noise, (ii) single-user detection, i.e., interference treated as noise.
- Received SINR at kth MS:

$$\mathrm{SINR}_k^{(\mathrm{DL})} := \frac{\text{desired signal power}}{\text{interference power} + \text{noise power}} = \frac{|\mathbf{h}_k^H \mathbf{w}_k|^2}{\sum_{j=1, j\neq k}^{K} |\mathbf{h}_k^H \mathbf{w}_j|^2 + \sigma_k^2} .$$

Example 4: Discrete rate adaptation

Modeling discrete rate adaptation

TECHNISCHE UNIVERSITÄT DARMSTADT

- Binary variable $a_{k,\ell} \in \{0, 1\}$, $k = 1, \dots, K$, $\ell = 1, \dots, L$

$$a_{k,\ell} = \begin{cases} 1 & \ell\text{th candidate MCS assigned to } k\text{th MS} \\ 0 & \text{otherwise} \end{cases}$$

- R_ℓ: data rate corresponding to ℓth MCS

	MCS_1, R_1	MCS_2, R_2	$\cdots$	MCS_L, R_L
MS 1	$a_{1,1}$	$a_{1,2}$	$\cdots$	$a_{1,L}$
MS 2	$a_{2,1}$	$a_{2,2}$	$\cdots$	$a_{2,L}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
MS K	$a_{K,1}$	$a_{K,2}$	$\cdots$	$a_{K,L}$

At most one MCS for each MS: $\sum_{\ell=1}^{L} a_{k,\ell} \leq 1 \Longleftarrow$ admission control

Example 4: Discrete rate adaptation

Problem formulation

TECHNISCHE UNIVERSITÄT DARMSTADT

MINLP formulation (combinatorial program):

$$\max_{\{a_{k,\ell},\mathbf{w}_k\}} \quad \sum_{k=1}^{K} \sum_{\ell=1}^{L} a_{k,\ell} R_\ell - \rho \sum_{k=1}^{K} \|\mathbf{w}_k\|_2^2 \qquad \text{(system utility function)}$$

$$\text{s.t.} \quad \sum_{k=1}^{K} \|\mathbf{w}_k\|_2^2 \leq P^{(\text{MAX})} \qquad \text{(per-BS sum-power constraint)}$$

$$\sum_{\ell=1}^{L} a_{k,\ell} \leq 1, \forall k \qquad \text{(multiple-choice, admission control)}$$

$$\text{SINR}_k = \frac{|\mathbf{h}_k^H \mathbf{w}_k|^2}{\sum_{j=1, j\neq k}^{K} |\mathbf{h}_k^H \mathbf{w}_j|^2 + \sigma_k^2} \geq \sum_{\ell=1}^{L} a_{k,\ell} \Gamma_\ell, \forall k \qquad \text{(SINR constraint)}$$

$$\sum_{\ell=1}^{L} a_{k,\ell} R_\ell \geq \sum_{\ell=1}^{L} a_{k,\ell} R_k^{(\text{MIN})}, \forall k \qquad \text{(rate requirement when admitted)}$$

$$a_{k,\ell} \in \{0, 1\}, \forall k, \ell \qquad \text{(integer constraint)}$$

- Constant ρ: weighting factor; Constant $P^{(\text{MAX})}$: TX power budget of BS; Constant $R_k^{(\text{MIN})}$: minimum rate requirement of kth MS when admitted.

Example 4: Discrete rate adaptation

Reformulating the SINR constraints

- The SINR constraints:

$$\sum_{\ell=1}^{L} a_{k,\ell}\Gamma_\ell \leq \mathrm{SINR}_k = \frac{\left|\mathbf{h}_k^H \mathbf{w}_k\right|^2}{\sum_{j=1,j\neq k}^{K} \left|\mathbf{h}_k^H \mathbf{w}_j\right|^2 + \sigma_k^2}, \forall k, \text{ are equivalent to}$$

$$\left(\sum_{j=1,j\neq k}^{K} \left|\mathbf{h}_k^H \mathbf{w}_j\right|^2 + \sigma_k^2\right) \sum_{\ell=1}^{L} a_{k,\ell}\Gamma_\ell \leq \left|\mathbf{h}_k^H \mathbf{w}_k\right|^2, \forall k.$$

- Introduce the big-M constant $U_k > 0$:

$$U_k \coloneqq \sqrt{P^{(\mathrm{MAX})} \|\mathbf{h}_k\|_2^2 + \sigma_k^2}, \quad \text{such that} \quad U_k^2 \geq \sum_{j=1}^{K} \left|\mathbf{h}_k^H \mathbf{w}_j\right|^2 + \sigma_k^2.$$

- Since $a_{k,\ell} \in \{0, 1\}$, $\sum_{\ell=1}^{L} a_{k,\ell} \leq 1$, **equivalent** SINR constraints:

$$\sum_{j=1}^{K} \left|\mathbf{h}_k^H \mathbf{w}_j\right|^2 + \sigma_k^2 \leq \left(1 - a_{k,\ell}\right) U_k^2 + \gamma_\ell^2 \left|\mathbf{h}_k^H \mathbf{w}_k\right|^2, \forall k, \forall \ell,$$

with the constant $\gamma_\ell \coloneqq \sqrt{1 + 1/\Gamma_\ell}$.

Example 4: Discrete rate adaptation

Reformulating the SINR constraints

TECHNISCHE UNIVERSITÄT DARMSTADT

- The SINR constraints are now in the form:

$$\sum_{j=1}^{K} \left|\mathbf{h}_k^H \mathbf{w}_j\right|^2 + \sigma_k^2 \leq \left(1 - a_{k,\ell}\right) U_k^2 + \gamma_\ell^2 \left|\mathbf{h}_k^H \mathbf{w}_k\right|^2, \forall k, \forall \ell.$$

- Choose the phase of $\mathbf{w}_k$ to make $\mathbf{h}_k^H \mathbf{w}_k$ real and non-negative [Bengtsson'01].
- Since $a_{k,\ell} \in \{0, 1\}$, the SINR constraints can be equivalently reformulated as

$$\mathrm{Im}(\mathbf{h}_k^H \mathbf{w}_k) = 0, \;\; \mathrm{Re}(\mathbf{h}_k^H \mathbf{w}_k) \geq 0, \; \forall k$$

$$\left\| \left[\mathbf{h}_k^H \mathbf{W}, \; \sigma_k \right] \right\|_2 \leq \left(1 - a_{k,\ell}\right) U_k + \gamma_\ell \, \mathrm{Re}(\mathbf{h}_k^H \mathbf{w}_k), \forall k, \forall \ell$$

$$\mathbf{W} \coloneqq [\mathbf{w}_1, \; \mathbf{w}_2, \; \cdots, \; \mathbf{w}_K] \in \mathbb{C}^{M \times K}$$

which become convex second-order cone constraints when $\{a_{k,\ell}\}$ relaxed to be continuous in [0, 1].

Example 4: Discrete rate adaptation

Standard mixed-integer second-order cone program (MISOCP)

TECHNISCHE UNIVERSITÄT DARMSTADT

Standard MISOCP formulation:

$$\max_{\{a_{k,\ell},\mathbf{w}_k\}} \quad \sum_{k=1}^{K}\sum_{\ell=1}^{L} a_{k,\ell}R_\ell - \rho\sum_{k=1}^{K}\|\mathbf{w}_k\|_2^2$$

$$\text{s.t.} \quad \sum_{k=1}^{K}\|\mathbf{w}_k\|_2^2 \le P^{(\mathrm{MAX})}; \quad \sum_{\ell=1}^{L} a_{k,\ell} \le 1, \forall k; \quad a_{k,\ell} \in \{0,1\}, \forall k,\ell$$

$$\sum_{\ell=1}^{L} a_{k,\ell}R_\ell \ge \sum_{\ell=1}^{L} a_{k,\ell}R_k^{(\mathrm{MIN})}, \forall k$$

$$\mathrm{Im}(\mathbf{h}_k^H\mathbf{w}_k) = 0, \forall k; \quad \mathrm{Re}(\mathbf{h}_k^H\mathbf{w}_k) \ge 0, \forall k$$

$$\left\|\left[\mathbf{h}_k^H\mathbf{W},\ \sigma_k\right]\right\|_2 \le (1 - a_{k,\ell})U_k + \gamma_\ell\,\mathrm{Re}(\mathbf{h}_k^H\mathbf{w}_k), \forall k,\ell \qquad \text{(SINR cons.)}$$

$$\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_K]$$

- When $\{a_{k,\ell}\}$ relaxed into the interval [0, 1], the formulation becomes a convex SOCP, i.e., the associated continuous relaxation is a convex SOCP.
- Globally-optimal solutions via the branch-and-X method

Example 4: Discrete rate adaptation

Extended formulation

TECHNISCHE UNIVERSITÄT DARMSTADT

- Introduce virtual beamformer $\mathbf{v}_{k,\ell} \in \mathbb{C}^M$ for the case that ℓth data rate assigned to kth MS.
- Introduce virtual transmission power $\phi_{k,\ell} \geq 0$ for the virtual beamformer $\mathbf{v}_{k,\ell}$, i.e., $\phi_{k,\ell} = \|\mathbf{v}_{k,\ell}\|_2^2$.
- Since $a_{k,\ell} \in \{0, 1\}$, $\sum_{\ell=1}^{L} a_{k,\ell} \leq 1$, relate $\{\mathbf{v}_{k,\ell}, \forall \ell\}$ to $\mathbf{w}_k$ according to

$$\mathbf{w}_k = \sum_{\ell=1}^{L} \mathbf{v}_{k,\ell}, \forall k.$$

- To make sure at most one of $\{\mathbf{v}_{k,\ell}, \forall \ell\}$ non-zero (rate selection), impose

$$\|\mathbf{v}_{k,\ell}\|_2^2 \leq a_{k,\ell}\phi_{k,\ell} \Longleftrightarrow \left\|[2\mathbf{v}_{k,\ell}^T,\ (a_{k,\ell} - \phi_{k,\ell})]\right\|_2 \leq a_{k,\ell} + \phi_{k,\ell}, \forall k, \forall \ell$$

$$0 \leq \phi_{k,\ell} \leq a_{k,\ell} P^{(\mathrm{MAX})}, \forall k, \forall \ell.$$

Extended formulation: solving the optimization problem in an extended optimization space (i.e., with more optimization variables).

Example 4: Discrete rate adaptation

Extended (improved) MISOCP formulation

TECHNISCHE UNIVERSITÄT DARMSTADT

Extended MISOCP formulation:

$$
\begin{aligned}
\max_{\{a_{k,\ell},\mathbf{v}_{k,\ell},\phi_{k,\ell}\}} \quad & \sum_{k=1}^{K}\sum_{\ell=1}^{L} a_{k,\ell}R_\ell - \rho\sum_{k=1}^{K}\sum_{\ell=1}^{L}\phi_{k,\ell} \\
\text{s.t.} \quad & \sum_{k=1}^{K}\sum_{\ell=1}^{L}\phi_{k,\ell} \le P^{(\mathrm{MAX})}; \;\; \sum_{\ell=1}^{L} a_{k,\ell} \le 1, \forall k; \;\; a_{k,\ell} \in \{0,1\}, \forall k, \forall \ell \\
& \sum_{\ell=1}^{L} a_{k,\ell}R_\ell \ge \sum_{\ell=1}^{L} a_{k,\ell}R_k^{(\mathrm{MIN})}, \forall k \\
& \mathbf{w}_k = \sum_{\ell=1}^{L}\mathbf{v}_{k,\ell}, \forall k; \;\; \mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_K] \\
& \mathrm{Im}(\mathbf{h}_k^H\mathbf{v}_{k,\ell}) = 0, \forall k, \forall \ell; \;\; \mathrm{Re}(\mathbf{h}_k^H\mathbf{v}_{k,\ell}) \ge 0, \forall k, \forall \ell \\
& \left\|[\mathbf{h}_k^H\mathbf{W}, \; \sigma_k]\right\|_2 \le \left(1 - \sum_{\ell=1}^{L} a_{k,\ell}\right)U_k + \sum_{\ell=1}^{L}\gamma_\ell \,\mathrm{Re}(\mathbf{h}_k^H\mathbf{v}_{k,\ell}), \forall k \\
& \left\|[2\mathbf{v}_{k,\ell}^T, \; (a_{k,\ell} - \phi_{k,\ell})]\right\|_2 \le a_{k,\ell} + \phi_{k,\ell}, \forall k, \forall \ell \\
& 0 \le \phi_{k,\ell} \le a_{k,\ell}P^{(\mathrm{MAX})}, \forall k, \forall \ell
\end{aligned}
$$

Example 4: Discrete rate adaptation

Low-complexity heuristics

TECHNISCHE UNIVERSITÄT DARMSTADT

- For large-scale problems (e.g., with large K), pursue high-quality solutions, rather than optimality (complexity-performance tradeoff):
  - Inflation procedure (greedily assign data rates),
  - Deflation procedure (greedily de-assign data rates),
  - Mixture of inflation and deflation procedures,
  - Genetic algorithm (randomly combine the integer-feasible solutions),
  - Any other heuristics
- Solution quality: relative MIP gap η:

$$\eta := \frac{\Phi^{(\text{UB})} - \Phi^{(\text{INT})}}{\Phi^{(\text{INT})}} = \frac{\Phi^{(\text{UB})}}{\Phi^{(\text{INT})}} - 1.$$

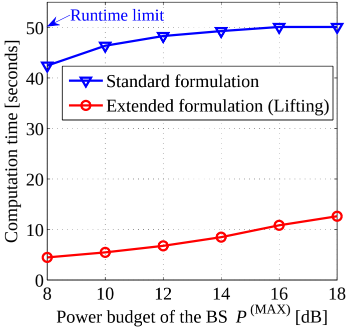
For a given relative gap tolerance, e.g., $\eta_0 = 10^{-3}$, integer-feasible solution declared as optimal solution if $\eta < \eta_0$.

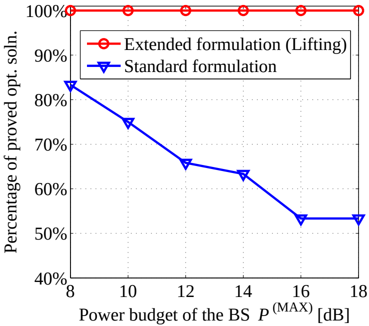
Example 4: Discrete rate adaptation

Simulation results

TECHNISCHE UNIVERSITÄT DARMSTADT

- System parameters $(M, K, L) = (4, 10, 15)$, optimality tolerance $\eta_0 = 10^{-3}$
- $\sigma_k^2 = -143$ dB, 3GPP channel model, random MS drops
- Runtime limit of CPLEX set as $T = 50$ seconds, 600 Monte Carlo runs





- Customizing strategies for the solver CPLEX (see the references)

Outline

TECHNISCHE UNIVERSITÄT DARMSTADT

Example 5: Codebook-based beamforming

Motivation

- In multiuser downlink beamforming: received signal $y_k \in \mathbb{C}$ at kth MS:

$$y_k = \mathbf{h}_k^H \mathbf{w}_k x_k + \sum_{j=1, j\neq k}^{K} \mathbf{h}_k^H \mathbf{w}_j x_j + z_k$$

  - $\mathbf{h}_k^H$ and $\mathbf{w}_k$: channel vector and beamformer of kth MS, resp.
  - Interference treated as noise.
  - Both $\{\mathbf{h}_k^H\}$ and $\{\mathbf{w}_k\}$ known at BS, only $\mathbf{h}_k^H$ known at kth MS.

- Effective channel $\mathbf{h}_k^H \mathbf{w}_k$ required for symbol detection, how to signal $\mathbf{h}_k^H \mathbf{w}_k$?

- In standards, e.g., LTE, two methods are defined:
  - in non-codebook-based beamforming, BS transmitting user-specific reference signals, and kth MS estimating $\mathbf{h}_k^H \mathbf{w}_k$,
  - employing codebook-based beamforming.

Example 5: Codebook-based beamforming

System model

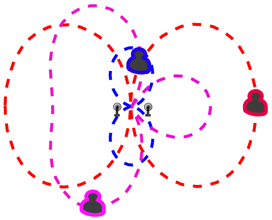
TECHNISCHE UNIVERSITÄT DARMSTADT

- Codebook-based beamforming:

$$\mathbf{w}_k = \sqrt{p_k}\mathbf{u}_k, \ \ \mathbf{u}_k \in \{\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_L\}$$

  - $\mathbf{f}_\ell \in \mathbb{C}^M$: predefined, with $\|\mathbf{f}_\ell\|_2 = 1$, $\ell = 1, 2, \ldots, L$

Beam pattern selection



- Received signal $y_k \in \mathbb{C}$ at kth MS:

$$y_k = \mathbf{h}_k^H \mathbf{u}_k \sqrt{p_k} x_k + \sum_{j=1, j\neq k}^{K} \mathbf{h}_k^H \mathbf{u}_j \sqrt{p_j} x_j + z_k.$$

- When $\mathbf{u}_k = \mathbf{f}_{\ell_k}$, BS signalling ℓ_k and p_k to kth MS
- Reconstructing $\mathbf{h}_k^H \mathbf{f}_{\ell_k} \sqrt{p_k}$ at kth MS
- No user-specific reference signals $\Longrightarrow$ simpler implementation

Example 5: Codebook-based beamforming

Problem formulation

TECHNISCHE UNIVERSITÄT DARMSTADT

Power minimization under SINR requirements:

$$\min_{\{\mathbf{u}_k, p_k\}} \quad \sum_{k=1}^{K} p_k$$

$$\text{s.t.} \quad \sum_{k=1}^{K} p_k \leq P^{(\mathrm{MAX})}; \quad p_k \geq 0, \forall k$$

$$\mathbf{u}_k \in \{\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_L\}, \forall k \qquad \text{(multiple-choice)}$$

$$\mathrm{SINR}_k^{(\mathrm{DL})} = \frac{p_k |\mathbf{h}_k^H \mathbf{u}_k|^2}{\sum_{j=1, j\neq k}^{K} p_j |\mathbf{h}_k^H \mathbf{u}_j|^2 + \sigma_k^2} \geq \Gamma_k^{(\mathrm{MIN})}, \forall k \qquad \text{(SINR cons.)}$$

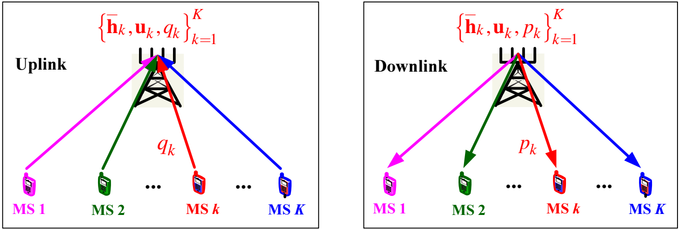
- Combinatorial program
- Reformulation as a mixed-integer linear program
- Commercial solver, e.g., CPLEX, based approach
- Polynomial-time OPTIMAL scheme built on uplink-downlink duality

Example 5: Codebook-based beamforming

Uplink-downlink duality

TECHNISCHE UNIVERSITÄT DARMSTADT

- Considering $\overline{\mathbf{h}}_k := \mathbf{h}_k / \sigma_k$, uplink (UL) and DL systems achieving same SINR region with:
  - Same beamformers & **total** transmitted BS power,
  - Different transmission powers.



- Originally proposed for non-codebook-based beamforming.
- Valid for codebook-based beamforming (see the references).

Example 5: Codebook-based beamforming

Equivalence of uplink & downlink formulations

TECHNISCHE UNIVERSITÄT DARMSTADT

Uplink problem:

$$Q^{(\mathrm{UL})} := \min_{\{\mathbf{u}_k, q_k\}} \sum_{k=1}^{K} q_k$$

$$\text{s.t.} \quad \sum_{k=1}^{K} q_k \leq P^{(\mathrm{MAX})}, \quad q_k \geq 0$$

$$\mathbf{u}_k \in \{\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_L\}, \forall k$$

$$\frac{q_k \left|\overline{\mathbf{h}}_k^H \mathbf{u}_k\right|^2}{\sum_{j=1, j\neq k}^{K} q_j \left|\overline{\mathbf{h}}_j^H \mathbf{u}_k\right|^2 + 1} \geq \Gamma_k^{(\mathrm{MIN})}, \forall k$$

Downlink problem:

$$P^{(\mathrm{DL})} := \min_{\{\mathbf{u}_k, p_k\}} \sum_{k=1}^{K} p_k$$

$$\text{s.t.} \quad \sum_{k=1}^{K} p_k \leq P^{(\mathrm{MAX})}, \quad p_k \geq 0$$

$$\mathbf{u}_k \in \{\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_L\}, \forall k$$

$$\frac{p_k \left|\overline{\mathbf{h}}_k^H \mathbf{u}_k\right|^2}{\sum_{j=1, j\neq k}^{K} p_j \left|\overline{\mathbf{h}}_k^H \mathbf{u}_j\right|^2 + 1} \geq \Gamma_k^{(\mathrm{MIN})}, \forall k$$

- Feasible uplink problem if and only if feasible downlink problem.
- When uplink problem feasible:
  - $Q^{(\mathrm{UL})} = P^{(\mathrm{DL})}$
  - An optimal soln. of UL problem $\overset{\text{closed-form}}{\Longleftrightarrow}$ an optimal soln. of DL problem.

Example 5: Codebook-based beamforming

Low-complexity power iteration method (PIM)

TECHNISCHE UNIVERSITÄT DARMSTADT

Uplink problem:

$$Q^{(\mathrm{UL})} := \min_{\{\mathbf{u}_k, q_k\}} \sum_{k=1}^{K} q_k$$

$$\text{s.t.} \quad \sum_{k=1}^{K} q_k \leq P^{(\mathrm{MAX})}, \quad q_k \geq 0$$

$$\mathbf{u}_k \in \{\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_L\}, \forall k$$

$$\frac{q_k |\overline{\mathbf{h}}_k^H \mathbf{u}_k|^2}{\sum_{j=1, j\neq k}^{K} q_j |\overline{\mathbf{h}}_j^H \mathbf{u}_k|^2 + 1} \geq \Gamma_k^{(\mathrm{MIN})}, \forall k$$

Adapted PIM:

Init.: $q_k^{(0)} = 0$, $k = 1, \ldots, K$.

1. Given $\{q_k^{(n)}\}$, select optimal beamformer $\mathbf{u}_k^{(n)} \in \{\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_L\}$.
2. Given $\{\mathbf{u}_k^{(n)}\}$, update power $q_k^{(n+1)}$.
3. Check $\sum_{k=1}^{K} q_k^{(n+1)} \leq P^{(\mathrm{MAX})}$. If violated, terminate (infeasible).

- For fixed uplink powers $\{q_k\}$, beamformers $\{\mathbf{u}_k\}$ decoupled.

- Adapted PIM optimally yielding:
  - Infeasibility certificates, or
  - Optimal solutions.

SCBF problem

Numerical results

- One BS with $M = 4$ antennas, $K = 4$ single-antenna MSs, LTE-A codebook with $L = 16$ beamformers
- $\sigma_k^2 = -143$ dB, 3GPP channel model, random MS drops
- Identical SINR target for all MSs
- With CPLEX as benchmark, 5000 Monte Carlo runs:

Average computation time [seconds] vs. SINR target $\Gamma_k^{(\mathrm{MIN})}$ [dB]

$\Gamma_k^{(\mathrm{MIN})}$	-6	-4	-2	0	2	4
CPLEX	0.3586	0.3601	0.3620	0.3644	0.3725	0.3775
PIM	0.0010 (0.28%)	0.0012 (0.33%)	0.0018 (0.50%)	0.0045 (1.23%)	0.0042 (1.13%)	0.0012 (0.32%)

TECHNISCHE UNIVERSITÄT DARMSTADT

Part IV

Summary and Concluding Remarks

Summary

TECHNISCHE UNIVERSITÄT DARMSTADT

- Mixed-integer programming (MIP): a powerful tool for network optimization and resource allocation
  - Basics and general applications of MIP
  - Software tools for MIP
  - Practical applications
- More applications in design and optimization of cellular networks
  - Load balancing in heterogenous networks
  - Uplink joint transmit-receive beamforming
  - Decoding delay selection in asynchronous relay networks
  - Topology optimization of optical fiber networks
  - Backhaul network resource allocation (routing)
  - Dynamic BBUs and RRHs mapping in C-RAN

Acknowledgement

TECHNISCHE UNIVERSITÄT DARMSTADT

Special thanks go to

- Prof. Stefan Ulbrich, Dr. Sarah Drewes, and Dipl. Math. Anne Philipp, Dept. of Mathematics, TU Darmstadt.
- M.Sc. Ganapati Hegde and Dipl.-Ing. Christian Steffens, Communication Systems Group, TU Darmstadt.

References

Basic literature

TECHNISCHE UNIVERSITÄT DARMSTADT

General mixed-integer programming:

- L.A. Wolsey and G.L. Nemhauser: “Integer and Combinatorial Optimization”, Wiley, 1999.
- D.-S. Chen, R.G. Batson, and Y. Dang, “Applied Integer Programming: Modeling and Solution”, Wiley, 2010.
- L.A. Wolsey, “Production Planning by Mixed Integer Programming”, Springer, 2006.

Convex optimization:

- S. Boyd and L. Vandenberghe: “Convex Optimization,” Cambridge University Press, 2004.

Global mixed-integer nonlinear optimization:

- J. Lee and S. Leyffer (editors), “Mixed Integer Nonlinear Programming,” Springer, Dec. 2011.
- S. Vigerske: “Decomposition of Multistage Stochastic Programs and a Constraint Integer Programming Approach to Mixed-Integer Nonlinear Programming,” PhD thesis, Humboldt-University Berlin, 2013.
- R. Horst and H. Tuy: “Global optimization,” Springer-Verlag, Berlin, second edition, 1993.
- M. Locatelli and F. Schoen: “Global optimization,” vol. 15 of MOS-SIAM Series on Optimization. Philadelphia, 2013.

References

Software

- J. Löfberg, “YALMIP : A Toolbox for Modeling and Optimization in MATLAB,” In Proceedings of the CACSD Conference, Taipei, Taiwan, 2004.
- M. Grant and S. Boyd, “CVX: Matlab software for disciplined convex programming, version 2.0 beta,” `http://cvxr.com/cvx`, September 2013.
- M.R. Bussieck and S. Vigerske: “MINLP solver software,” in J.J. Cochran, Jr. L.A. Cox, P. Keskinocak, J.P. Kharoufeh, and J.C. Smith, editors, Wiley Encyclopedia of Operations Research and Management Science, Wiley & Sons, Inc., 2010.
- T. Achterberg: “SCIP: solving constraint integer programs,” Mathematical Programming Computation, Volume 1, Number 1, pp. 1–41, 2009
- T. Koch, A. Martin, and M. E. Pfetsch, “Progress in Academic Computational Integer Programming,” in *Facets of Combinatorial Optimization*, chapt. 14, Michael Jünger and Gerhard Reinelt, Eds. Springer-Verlag, 2013, pp. 483–506.

References

Details for examples

TECHNISCHE UNIVERSITÄT DARMSTADT

- Y. Cheng and M. Pesavento, "An Optimal Iterative Algorithm for Codebook-Based Downlink Beamforming," IEEE Signal Process. Letters, vol. 20, no. 8, pp. 775–778, Aug. 2013.
- Y. Cheng, M. Pesavento, and A. Philipp, "Joint Network Optimization and Downlink Beamforming for CoMP Transmissions using Mixed Integer Conic Programming," IEEE Trans. on Signal Processing, vol. 61, no. 16, pp. 3972–3987, Aug. 2013.
- Y. Cheng and M. Pesavento, "Joint Optimization of Rate Adaptation and Downlink Beamforming using Mixed Integer Conic Programming," IEEE Trans. on Signal Processing, vol. 63, no. 7, pp. 1750–1764, Apr. 2015.
- N. Bornhorst and M. Pesavento, "Filter-and-forward beamforming with adaptive decoding delays in asynchronous multi-user relay networks," Signal Processing, vol. 109, pp. 132–147, April 2015.
- A. Philipp, S. Ulbrich, Y. Cheng, and M. Pesavento, "Multiuser Downlink Beamforming with Interference Cancellation using an SDP-Based Branch-and-Bound Algorithm," IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), May 2014, Florence, Italy.
- Y. Cheng and M. Pesavento, "Predistortion and Precoding Vector Assignment in Codebook-Based Downlink Beamforming," in Proc. IEEE Int. Workshop on Signal Processing Advances for Wireless Communications (SPAWC), Darmstadt, Germany, Jun. 2013, pp. 440–444.

References

Details for examples

- Y. Cheng and M. Pesavento, “Robust Codebook-based Downlink Beamforming using Mixed Integer Conic Programming,” in Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), Vancouver, Canada, May 2013, pp. 4187–4191.
- Y. Cheng, A. Philipp, and M. Pesavento, “Dynamic Rate Adaptation and Multiuser Downlink Beamforming using Mixed Integer Conic Programming,” in Proc. European Signal Processing Conference (EUSIPCO), Bucharest, Romania, Aug. 2012, pp. 824–828.
- Y. Cheng, S. Drewes, A. Philipp, and M. Pesavento, “Joint Network Optimization and Beamforming for Coordinated Multi-Point Transmission using Mixed Integer Programming,” in Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), Kyoto, Japan, Mar. 2012, pp. 3217–3220.
- Y. Cheng, S. Drewes, A. Philipp, and M. Pesavento, “Joint Network Topology Optimization and Multicell Beamforming using Mixed Integer Programming,” in Proc. Int. ITG Workshop on Smart Antennas (WSA), Dresden, Germany, Mar. 2012, pp. 187–192.

References

Further interesting literature

- S. Jokar, V. Mehrmann, M. E. Pfetsch, and H. Yserentant, “Sparse Approximate Solution of Partial Differential Equations,” Appl. Numer. Math., vol. 60, no. 4, pp. 452–472, 2010.
- S. Jokar and M.E. Pfetsch, “Exact and Approximate Sparse Solutions of Underdetermined Linear Equations,” SIAM J. Sci. Comput, vol. 31, no. 1, pp. 23–44, 2008.
- J. Lessmann, Y. Cheng, S. Pfeiffer, and X.P. Costa, “Time-varying Routing in Realistic Mobile Backhaul Networks,” In Proc. IEEE Workshop on Next Generation Backhaul/Fronthaul Networks (BackNets’15), ICC’15, London, UK, June 8-12, 2015.
- L.-W. Lu, and R.-S. Fan, “Simulated annealing algorithm in solving frequency assignment problem,” Proceedings of the 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE’10), Chengdu, China, Aug. 20-22, 2010.
- R. Madan, S. Cui, S. Lall, and A.J. Goldsmith, “Mixed Integer-Linear Programming for Link Scheduling in Interference-Limited Networks,” in Proc. of 1st Workshop on Resource Allocation in Wireless Networks, Trento, Italy, Apr. 2005.
- I. Pitas, “Optimization and adaptation of discrete-valued digital filter parameters by simulated annealing,” IEEE Trans. on Signal Processing, vol. 42, no. 4, pp. 860–866, Apr. 1994.

References

Further interesting literature

- A.M. Tillmann and M.E. Pfetsch, "The Computational Complexity of the Restricted Isometry Property, the Nullspace Property, and Related Concepts in Compressed Sensing," IEEE Trans. on Information Theory, vol. 60, no. 2, pp. 1248–1259, 2014.
- D. Yuan, V. Angelakis, L. Chen, E. Karipidis, and E. G. Larsson, "On optimal link activation with interference cancellation in wireless networking," IEEE Trans. on Vehicular Technology, vol. 62, pp. 939–945, Feb. 2013.
- L. Yu, E. Karipidis and E.G. Larsson, "Coordinated scheduling and beamforming for multicell spectrum sharing networks using branch and bound," in Proc. of European Signal Processing Conference (EUSIPCO), Aug. 2012.
- D.J. Xu, and M.L. Daley, "Design of optimal digital filter using a parallel genetic algorithm," IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing, vol. 42, no. 10, pp. 673–675, Oct. 1995.