# Quantified Programming applied to Runway Scheduling

## TOR Workshop 2018

Michael Hartisch, Simon Gnad, Ulf Lorenz

Chair of Technology Management, University of Siegen, Germany

Wenden, 12. März 2018

# Outline

Section 1

# Quantified Programming

# Necessity of Quantified Programming

- Robust optimization is crucial
  - uncertain input data
  - optimal deterministic plan is only best-case scenario

# Necessity of Quantified Programming

- Robust optimization is crucial
  - uncertain input data
  - optimal deterministic plan is only best-case scenario
  - replanning is expensive

# Necessity of Quantified Programming

- Robust optimization is crucial
  - uncertain input data
  - optimal deterministic plan is only best-case scenario
  - replanning is expensive
  - optimal *strategy* is required

# Necessity of Quantified Programming

- Robust optimization is crucial
  - uncertain input data
  - optimal deterministic plan is only best-case scenario
  - replanning is expensive
  - optimal *strategy* is required
- Interested in strategy with optimal worst case outcome

  → for each possible data situation we want a "good" solution

# IP vs QIP: Example

$$
\begin{aligned}
\min_{x_1,x_2} \quad & -2x_1 - x_2 \\
\text{s.t.} \quad & 2x_1 - x_2 \leq 1 \\
& -x_1 + 2x_2 \leq 1 \\
& x_1, x_2 \in \{0, 1\}
\end{aligned}
$$

---

# IP vs QIP: Example

$$\begin{aligned} \min_{x_1,x_2} \quad & -2x_1 - x_2 \\ \text{s.t.} \quad & 2x_1 - x_2 \leq 1 \\ & -x_1 + 2x_2 \leq 1 \\ & x_1, x_2 \in \{0,1\} \end{aligned}$$

# IP vs QIP: Example

$$\begin{aligned} \min_{x_1,x_2} \quad & -2x_1 - x_2 \\ \text{s.t.} \quad & 2x_1 - x_2 \leq 1 \\ & -x_1 + 2x_2 \leq 1 \\ & x_1, x_2 \in \{0,1\} \end{aligned}$$

---

$$\begin{aligned} & \min_{x_1} \left\{ -2x_1 + \max_{y} \min_{x_2} \quad -x_2 \right\} \\ & \text{s.t.} \quad \exists x_1 \in \{0,1\} \quad \forall y \in \{0,1\} \quad \exists x_2 \in \{0,1\} : \\ & \qquad 2x_1 - x_2 \leq 1 - y \\ & \qquad -x_1 + 2x_2 \leq 1 + y \end{aligned}$$

# IP vs QIP

UNIVERSITÄT SIEGEN

| | IP | QIP |
|---|---|---|
| variables | basically one type of variables | |
| variable order | no (relevant) variable order | |
| objective function | min or max function | |
| solution format | variable assignment + objective value | |
| game type | single-player game | |

# IP vs QIP

| | IP | QIP |
|---|---|---|
| variables | basically one type of variables | two types of variables: existential (∃) and universal (∀) |
| variable order | no (relevant) variable order | explicit order of the variable blocks |
| objective function | min or max function | min-max objective |
| solution format | variable assignment + objective value | assignment strategy + objective value of optimal play |
| game type | single-player game | two-person zero-sum game |

# QIPs - Formulation

Parameters:

- variable domain $\mathcal{D} = \{x \in \mathbb{Z}^n \mid l \leq x \leq u\}$
- quantification vector $Q \in \{\exists, \forall\}^n$
- matrix $A$, right-hand side vector $b$, objective vector $c$

# QIPs - Formulation

Parameters:

- variable domain $\mathcal{D} = \{x \in \mathbb{Z}^n \mid l \leq x \leq u\}$
- quantification vector $Q \in \{\exists, \forall\}^n$
- matrix $A$, right-hand side vector $b$, objective vector $c$

Problem statement:

$$\min_{x^{(1)}} \left( c^{(1)} x^{(1)} + \max_{x^{(2)}} \left( c^{(2)} x^{(2)} + \min_{x^{(3)}} \left( \ldots + \min_{x^{(\beta)}} c^{(\beta)} x^{(\beta)} \right) \right) \right)$$

$$\text{s.t.} \quad Q \circ x \in \mathcal{D} : \; Ax \leq b$$

Parameters:

- variable domain $\mathcal{D} = \{x \in \mathbb{Z}^n \mid l \leq x \leq u\}$
- quantification vector $Q \in \{\exists, \forall\}^n$
- matrix $A$, right-hand side vector $b$, objective vector $c$

Problem statement:

$$\min_{x^{(1)}} \left( c^{(1)} x^{(1)} + \max_{x^{(2)}} \left( c^{(2)} x^{(2)} + \min_{x^{(3)}} \left( \ldots + \min_{x^{(\beta)}} c^{(\beta)} x^{(\beta)} \right) \right) \right)$$

$$\text{s.t.} \quad Q \circ x \in \mathcal{D} : \ Ax \leq b$$

$$\exists x_1 \in [l_1, u_1]_{\mathbb{Z}} \ \forall x_2 \in [l_2, u_2]_{\mathbb{Z}} \ \exists x_3 \in [l_3, u_3]_{\mathbb{Z}} \ \forall x_4 \ldots \ : Ax \leq b$$

# Robust Runway Scheduling

Motivated by cooperative research with FAU Erlangen (Frauke Liers).

A. Heidt, H. Helmke, F. Liers, and A. Martin. Robust runway scheduling using a time-indexed model. In D. Schäfer, editor, Proceedings of the SESAR Innovation Days (2014) EUROCONTROL, 2014. ISBN 978-2-87497-077-1.

# Basic Runway Scheduling Problem

- Set of time slots $W = \{1, \ldots, T\}$
- Set of airplanes $A = \{1, \ldots, n\}$

# Basic Runway Scheduling Problem

UNIVERSITÄT SIEGEN

- Set of time slots $W = \{1, \dots, T\}$
- Set of airplanes $A = \{1, \dots, n\}$
- Maximum number of allowed airplanes for each time slot $b$

# Basic Runway Scheduling Problem

- Set of time slots $W = \{1, \dots, T\}$
- Set of airplanes $A = \{1, \dots, n\}$
- Maximum number of allowed airplanes for each time slot $b$
- Costs for airplane $i$ when scheduled at time slot $j$: $c_{i,j}$

# Basic Runway Scheduling Problem

- Set of time slots $W = \{1, \ldots, T\}$
- Set of airplanes $A = \{1, \ldots, n\}$
- Maximum number of allowed airplanes for each time slot $b$
- Costs for airplane $i$ when scheduled at time slot $j$: $c_{i,j}$

B-Matching

- Each airplane must be assigned to exactly one time slot
- At most $b$ airplanes can be assigned to each time slot

# Basic Runway Scheduling Problem

- Set of time slots $W = \{1, \ldots, T\}$
- Set of airplanes $A = \{1, \ldots, n\}$
- Maximum number of allowed airplanes for each time slot $b$
- Costs for airplane $i$ when scheduled at time slot $j$: $c_{i,j}$

B-Matching

- Each airplane must be assigned to exactly one time slot
- At most $b$ airplanes can be assigned to each time slot

**Schedule each airplane to a time slot with minimal overall costs.**

# The Basic Model

$$
\begin{aligned}
\min \quad & \sum_{i\in A}\sum_{j\in W} c_{i,j}x_{i,j} & \\
\text{s.t.} \quad & \sum_{i\in A} x_{i,j} \leq b & \forall j \in W \\
& \sum_{j\in W} x_{i,j} = 1 & \forall i \in A \\
& x_{i,j} \in \{0,1\} & \forall i \in A, j \in W
\end{aligned}
$$

# The Basic Model

$$
\begin{aligned}
\min \quad & \sum_{i \in A} \sum_{j \in W} c_{i,j} x_{i,j} & \\
\text{s.t.} \quad & \sum_{i \in A} x_{i,j} \leq b & \forall j \in W \\
& \sum_{j \in W} x_{i,j} = 1 & \forall i \in A \\
& x_{i,j} \in \{0, 1\} & \forall i \in A, j \in W
\end{aligned}
$$

- Earliest and latest landing time via objective function, additional constraints or specific time windows $W_i$ for each airplane $i \in A$:

# The Basic Model

$$
\begin{aligned}
\min \quad & \sum_{i \in A} \sum_{j \in W} c_{i,j} x_{i,j} & \\
\text{s.t.} \quad & \sum_{i \in A} x_{i,j} \leq b & \forall j \in W \\
& \sum_{j \in W} x_{i,j} = 1 & \forall i \in A \\
& x_{i,j} \in \{0, 1\} & \forall i \in A, j \in W
\end{aligned}
$$

- Earliest and latest landing time via objective function, additional constraints or specific time windows $W_i$ for each airplane $i \in A$:

$$
\text{earliest}_i \leq \sum_{j \in W} j \cdot x_{i,j} \leq \text{latest}_i \qquad \forall i \in A
$$

# Adding Uncertainty

- Actual time window of arrival is uncertain

# Adding Uncertainty

- Actual time window of arrival is uncertain

$\Rightarrow$ Schedule obtained using the model might become infeasible

# Adding Uncertainty

UNIVERSITÄT SIEGEN

- Actual time window of arrival is uncertain

⇒ Schedule obtained using the model might become infeasible

- Robust plan is required:

⇒ For "any conceivable" time window the starting plan should be adaptable at moderate costs

# Adding Uncertainty

- Actual time window of arrival is uncertain

$\Rightarrow$ Schedule obtained using the model might become infeasible

- Robust plan is required:

$\Rightarrow$ For "any conceivable" time window the starting plan should be adaptable at moderate costs

$\exists$ a cheap initial plan

# Adding Uncertainty

- Actual time window of arrival is uncertain

$\Rightarrow$ Schedule obtained using the model might become infeasible

- Robust plan is required:

$\Rightarrow$ For "any conceivable" time window the starting plan should be adaptable at moderate costs

$\exists$ a cheap initial plan

$\forall$ possible time windows

# Adding Uncertainty

- Actual time window of arrival is uncertain

⇒ Schedule obtained using the model might become infeasible

- Robust plan is required:

⇒ For "any conceivable" time window the starting plan should be adaptable at moderate costs

∃ a cheap initial plan

∀ possible time windows

∃ a cheap recovery plan

# Adding Uncertainty: Example; $b = 2$

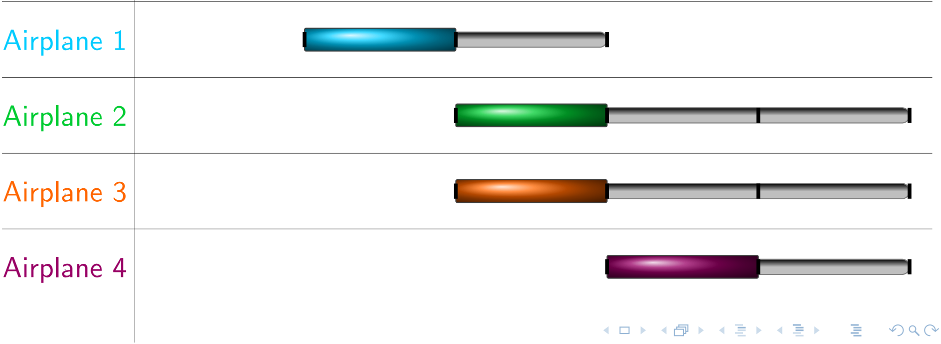
| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| Airplane 1 | **1** | **0** | **1** | **5** | **7** | **9** |
| Airplane 2 | **4** | **1** | **0** | **2** | **5** | **7** |
| Airplane 3 | **4** | **1** | **0** | **3** | **4** | **7** |
| Airplane 4 | **5** | **2** | **1** | **0** | **1** | **5** |

# Adding Uncertainty: Example; $b = 2$

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| Airplane 1 | **1** | **0** | **1** | **5** | **7** | **9** |
| Airplane 2 | **4** | **1** | **0** | **2** | **5** | **7** |
| Airplane 3 | **4** | **1** | **0** | **3** | **4** | **7** |
| Airplane 4 | **5** | **2** | **1** | **0** | **1** | **5** |

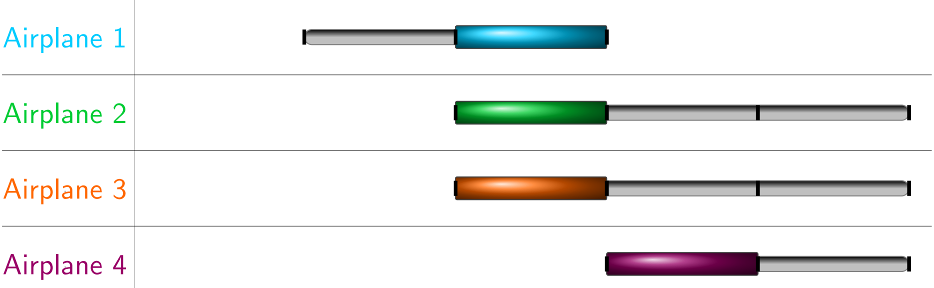
# Adding Uncertainty: Example; $b = 2$

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| Airplane 1 | **1** | **0** | **1** | **5** | **7** | **9** |
| Airplane 2 | **4** | **1** | **0** | **2** | **5** | **7** |
| Airplane 3 | **4** | **1** | **0** | **3** | **4** | **7** |
| Airplane 4 | **5** | **2** | **1** | **0** | **1** | **5** |



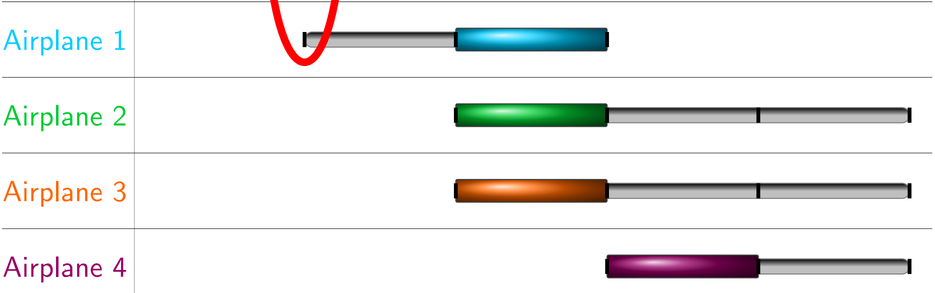
# Adding Uncertainty: Example; $b = 2$

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| Airplane 1 | **1** | **0** | **1** | **5** | **7** | **9** |
| Airplane 2 | **4** | **1** | **0** | **2** | **5** | **7** |
| Airplane 3 | **4** | **1** | **0** | **3** | **4** | **7** |
| Airplane 4 | **5** | **2** | **1** | **0** | **1** | **5** |



# Adding Uncertainty: Example; $b = 2$

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| Airplane 1 | 1 | 0 | 1 | 5 | 7 | 9 |
| Airplane 2 | 4 | 1 | 0 | 2 | 5 | 7 |
| Airplane 3 | 4 | 1 | 0 | 3 | 4 | 7 |
| Airplane 4 | 5 | 2 | 1 | 0 | 1 | 5 |



# Adding Uncertainty: Costs of replanning

Possible costs for replanning:

a) replanning an airplane result in a fixed fee of $\ell$

# Adding Uncertainty: Costs of replanning

Possible costs for replanning:

a) replanning an airplane result in a fixed fee of $\ell$

b) replanning/moving an airplane $k$ time windows costs $k \cdot \ell$

# Adding Uncertainty: Costs of replanning

Possible costs for replanning:

a) replanning an airplane result in a fixed fee of $\ell$

b) replanning/moving an airplane $k$ time windows costs $k \cdot \ell$

c) replanning an airplane $i$ costs $\tilde{c}_{i,j}$ of the selected time slot $j$

d) ... practical knowledge needed

# Adding Uncertainty: The Quantified Program

$$\min \quad \sum_{i\in A}\sum_{j\in W} c_{i,j}x_{i,j} + \max\left\{\min\left\{\sum_{i\in A}\sum_{j\in W} \ell z_{i,j}\right\}\right\}$$

# Adding Uncertainty: The Quantified Program

$$\min \quad \sum_{i\in A}\sum_{j\in W} c_{i,j}x_{i,j} + \max\left\{\min\left\{\sum_{i\in A}\sum_{j\in W} \ell z_{i,j}\right\}\right\}$$

$$\begin{aligned}\text{s.t.} \quad & \exists X \in \{0,1\}^{|A|\times|W|} \\ & \forall S \in \mathcal{S}, \quad L \in \mathcal{L} \\ & \exists Y \in \{0,1\}^{|A|\times|W|}, \quad Z \in \{0,1\}^{|A|\times|W|} :\end{aligned}$$

# Adding Uncertainty: The Quantified Program

$$\min \quad \sum_{i\in A}\sum_{j\in W} c_{i,j}x_{i,j} + \max\left\{\min\left\{\sum_{i\in A}\sum_{j\in W} \ell z_{i,j}\right\}\right\}$$

$$\text{s.t.} \quad \exists X \in \{0,1\}^{|A|\times|W|}$$

$$\forall S \in \mathcal{S}, \quad L \in \mathcal{L}$$

$$\exists Y \in \{0,1\}^{|A|\times|W|}, \quad Z \in \{0,1\}^{|A|\times|W|}:$$

$$\sum_{j\in W} x_{i,j} = 1 \qquad \sum_{j\in W} y_{i,j} = 1 \quad \forall i \in A$$

$$\sum_{i\in A} x_{i,j} \leq b \qquad \sum_{i\in A} y_{i,j} \leq b \quad \forall j \in W$$

$$z_{i,j} \geq y_{i,j} - x_{i,j} \qquad \forall i \in A, \quad j \in W$$

$$s_i \leq \sum_{j\in W} j \cdot y_{i,j} \leq s_i + l_i \qquad \forall i \in A$$

# Adding Uncertainty: The Quantified Program

$$\min \quad \sum_{i\in A}\sum_{j\in W} c_{i,j}x_{i,j} + \max\left\{\min\left\{\sum_{i\in A}\sum_{j\in W} \ell z_{i,j}\right\}\right\}$$

$$\text{s.t.} \quad \exists X \in \{0,1\}^{|A|\times|W|}$$

$$\forall S \in \mathcal{S}, \quad L \in \mathcal{L}$$

$$\exists Y \in \{0,1\}^{|A|\times|W|}, \quad Z \in \{0,1\}^{|A|\times|W|} :$$

$$\sum_{j\in W} x_{i,j} = 1 \qquad \sum_{j\in W} y_{i,j} = 1 \quad \forall i \in A$$

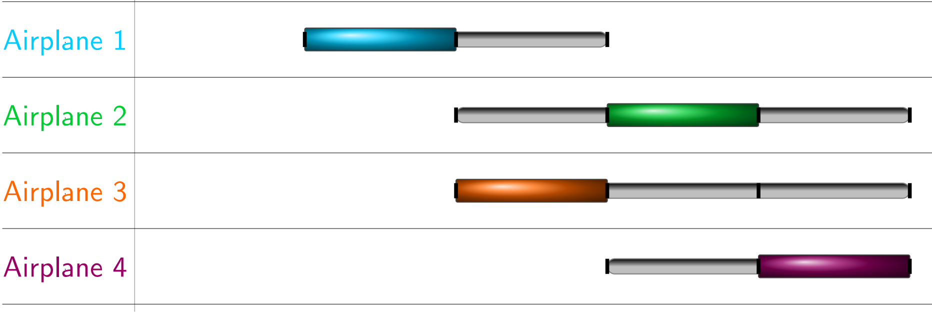
$$\sum_{i\in A} x_{i,j} \le b \qquad \sum_{i\in A} y_{i,j} \le b \quad \forall j \in W$$

$$z_{i,j} \ge y_{i,j} - x_{i,j} \quad \forall i \in A, \quad j \in W$$

$$s_i \le \sum_{j\in W} j \cdot y_{i,j} \le s_i + l_i \quad \forall i \in A$$

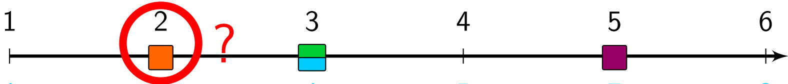
# fixed fee ($\ell = 4,\ b = 2$), costs: 11


| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| Airplane 1 | 1 | 0 | 1 | 5 | 7 | 9 |
| Airplane 2 | 4 | 1 | 0 | 2 | 5 | 7 |
| Airplane 3 | 4 | 1 | 0 | 3 | 4 | 7 |
| Airplane 4 | 5 | 2 | 1 | 0 | 1 | 5 |

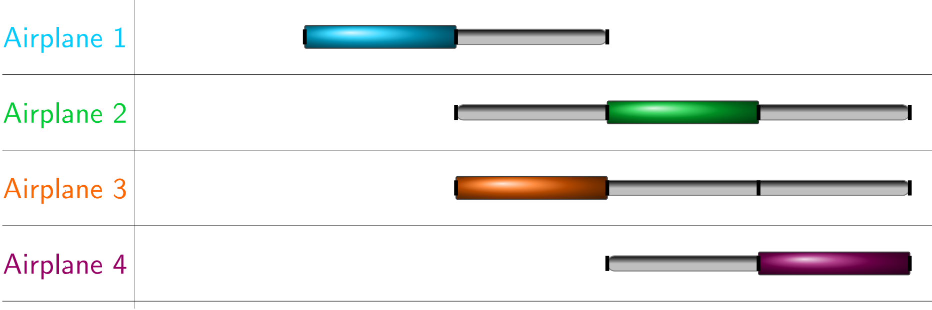


# fixed fee ($\ell = 4$, $b = 2$), costs: 11





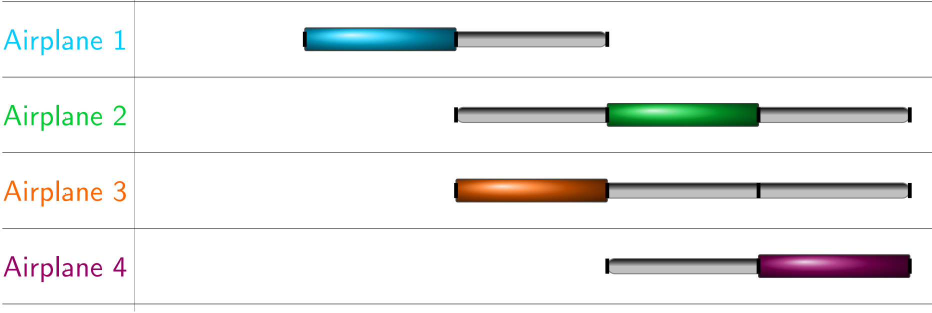

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| Airplane 1 | 1 | 0 | 1 | 5 | 7 | 9 |
| Airplane 2 | 4 | 1 | 0 | 2 | 5 | 7 |
| Airplane 3 | 4 | 1 | 0 | 3 | 4 | 7 |
| Airplane 4 | 5 | 2 | 1 | 0 | 1 | 5 |



# distance fee ($\ell = 3$, $b = 2$), costs: 14

UNIVERSITÄT SIEGEN

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| Airplane 1 | **1** | **0** | **1** | **5** | **7** | **9** |
| Airplane 2 | **4** | **1** | **0** | **2** | **5** | **7** |
| Airplane 3 | **4** | **1** | **0** | **3** | **4** | **7** |
| Airplane 4 | **5** | **2** | **1** | **0** | **1** | **5** |



# Restricting Universal Power

We are able to implicitly add an (independent) universal constraint system $A^{\forall}x \leq b^{\forall}$.

- force the selected interval lengths to be larger than x on average
- forcing the intervals to be „close to“ zero-cost time slot

# Solving QIPs: The DEP

Interested in first stage solution (here: Initial plan $X$).

# Solving QIPs: The DEP

Interested in first stage solution (here: Initial plan $X$).

- build the deterministic equivalent program:
  - for each „scenario" (assignment of the universal variables) the constraint system is stated explicitly
  - An IP is formed with exponentially growing size regarding the number of universal variables

# Solving QIPs: The DEP

Interested in first stage solution (here: Initial plan $X$).

- build the deterministic equivalent program:
  - for each „scenario“ (assignment of the universal variables) the constraint system is stated explicitly
  - An IP is formed with exponentially growing size regarding the number of universal variables
- solve the resulting IP using standard software (CPLEX, Gurobi, etc.)

Problem: rapid growth $\Rightarrow$ insufficient memory + IP is NP-complete

# Solving QIPs: Tree Search

- having an opponent (universal variables) sounds like game playing

# Solving QIPs: Tree Search

- having an opponent (universal variables) sounds like game playing
- combine game playing techniques with linear programming techniques

# Solving QIPs: Tree Search

- having an opponent (universal variables) sounds like game playing
- combine game playing techniques with linear programming techniques
- alphabeta, move ordering, LP relaxation, cutting planes, QIP specific results ...

# Solving QIPs: Tree Search

- having an opponent (universal variables) sounds like game playing
- combine game playing techniques with linear programming techniques
- alphabeta, move ordering, LP relaxation, cutting planes, QIP specific results ...

$\Rightarrow$ QMIP solver Yasol (www.q-mip.org)

# Solving QIPs: Tree Search

- having an opponent (universal variables) sounds like game playing
- combine game playing techniques with linear programming techniques
- alphabeta, move ordering, LP relaxation, cutting planes, QIP specific results ...

$\Rightarrow$ QMIP solver Yasol (www.q-mip.org)

Advantages:

- no problem storing the instance itself (of course the search tree might grow rapidly)

# Solving QIPs: Tree Search

- having an opponent (universal variables) sounds like game playing
- combine game playing techniques with linear programming techniques
- alphabeta, move ordering, LP relaxation, cutting planes, QIP specific results ...

$\Rightarrow$ QMIP solver Yasol (www.q-mip.org)

Advantages:

- no problem storing the instance itself (of course the search tree might grow rapidly)
- insignificant/easy scenarios (branches) can be cut off

| | CPLEX (DEP) | | | Yasol (CPLEX as LP Solver) | | |
|---|---|---|---|---|---|---|
| Instance | Feas. | Value | Time | Feas. | Value | Time |
| FP_A7_W9_b2_T147_ol_S40_R73 | - | - | 3600 | feasible | 40 | 33 |
| FP_A5_W7_b2_T190_l_S25_R96 | - | - | 3600 | feasible | 25 | 64 |
| FP_A90_W120_b4_T_l_S_R75 | - | - | 3600 | - | - | 3600 |
| FP_A60_W100_b2_T_ol | - | - | 3600 | - | - | 3600 |
| FP_A8_W11_b2_T21_l_S44_R88_EAEA | - | - | 3600 | feasible | 44 | 9 |
| FP_A5_W7_b2_T2_ol_S25_R96_EA | - | - | 3600 | feasible | 25 | 1 |
| FP_A60_W110_b2_T_l_S_R96 | - | - | 3600 | infeas. | - | 4 |
| FP_A7_W12_b2_T16_ol_S36_R96_EAEA | feasible | 36 | 15 | feasible | 36 | 3 |
| FP_A9_W11_b3_T3013_S45_EA | - | - | 3600 | feasible | 45 | 753 |
| FP_A5_W12_b2_T2_ol_S18_R96_EAEA | feasible | 18 | 1 | feasible | 18 | 1 |
| FP_A7_W9_b2_T180_ol_S36_R96 | - | - | 3600 | feasible | 36 | 50 |
| FP_A5_W12_b2_T2_ol_S18_R96_EA | feasible | 18 | 0 | feasible | 18 | 1 |
| FP_A5_W14_b2_T360_l_S26_R96_EA | - | - | 3600 | feasible | 26 | 91 |
| FP_A8_W11_b2_T35_l_S44_R88_EA | - | - | 3600 | feasible | 44 | 7 |
| FP_A70_W95_b3_T_l_S_R80 | - | - | 3600 | infeas. | - | 3 |
| FP_A6_W9_b2_T6_ol_S26_R96_EAEA | feasible | 26 | 5 | feasible | 26 | 4 |
| FP_A85_W110_b3_T_l_S_R75 | - | - | 3600 | - | - | 3600 |
| FP_A7_W12_b2_T21_ol_S36_R96_EA | feasible | 36 | 15 | feasible | 36 | 4 |
| FP_A5_W7_b2_T2_ol_S25_R96 | - | - | 3600 | feasible | 25 | 1 |
| FP_A30_W50_b2_T_ol | - | - | 3600 | infeas. | - | 2 |
| FP_A7_W9_b2_T82_ol_S40_R92 | - | - | 3600 | feasible | 40 | 46 |
| FP_A9_W11_b3_T2490_S45_EAEA | - | - | 3600 | feasible | 45 | 1052 |
| FP_A7_W9_b2_T122_ol_S41_R86 | - | - | 3600 | feasible | 41 | 42 |
| FP_A5_W14_b2_T1068_l_S26_R96_EAEA | - | - | 3600 | feasible | 26 | 28 |
| FP_A7_W10_b2_T228_ol_S33_R80 | - | - | 3600 | feasible | 33 | 105 |
| FP_A45_W60_b2_T_l_S_R78 | - | - | 3600 | infeas. | - | 2 |
| FP_A5_W7_b2_T2_ol_S25_R96_EAEA | - | - | 3600 | feasible | 25 | 1 |
| FP_A150_W200_b4_T_l_S_R80 | - | - | 3600 | - | - | 3600 |
| FP_A6_W9_b2_T6_ol_S26_R96_EA | feasible | 26 | 5 | feasible | 26 | 3 |

# Conclusion and Outlook

- Investigated matching problem under uncertainty

# Conclusion and Outlook

UNIVERSITÄT SIEGEN

- Investigated matching problem under uncertainty
- Used quantified programs to model and solve them

# Conclusion and Outlook

- Investigated matching problem under uncertainty
- Used quantified programs to model and solve them
- Our general QMIP solver beats the DEP approach (Cplex) on those instances

# Conclusion and Outlook

- Investigated matching problem under uncertainty
- Used quantified programs to model and solve them
- Our general QMIP solver beats the DEP approach (Cplex) on those instances

- Enhancing and improving the model

# Conclusion and Outlook

- Investigated matching problem under uncertainty
- Used quantified programs to model and solve them
- Our general QMIP solver beats the DEP approach (Cplex) on those instances

- Enhancing and improving the model
- how does a realistic/relevant instance look like?

# Conclusion and Outlook

- Investigated matching problem under uncertainty
- Used quantified programs to model and solve them
- Our general QMIP solver beats the DEP approach (Cplex) on those instances

- Enhancing and improving the model
- how does a realistic/relevant instance look like?
- design problem specific heuristics to attack relevant instances

Thank you!

UNIVERSITÄT SIEGEN

The original QIP

$$
\begin{array}{llllllllll}
\min & 2x_1 & - & 2x_2 & - & 3x_3 & - & 2x_4 & & \\
\text{s.t.} & \exists x_1 & & \forall x_2 & & \exists x_3 & & \forall x_4 & : & \\
& x_1 & + & x_2 & + & x_3 & & & \leq & 2 \\
& -x_1 & & & + & x_3 & - & x_4 & \leq & 0 \\
& & - & x_2 & + & x_3 & - & x_4 & \leq & 0 \\
& -x_1 & + & x_2 & - & x_3 & + & x_4 & \leq & 1
\end{array}
$$

# App: The Deterministic Equivalent Program

## The original QIP

$$
\begin{array}{llllllllll}
\min & 2x_1 & - & 2x_2 & - & 3x_3 & - & 2x_4 & & \\
\text{s.t.} & \exists x_1 & & \forall x_2 & & \exists x_3 & & \forall x_4 & : & \\
& x_1 & + & x_2 & + & x_3 & & & \leq & 2 \\
& -x_1 & & & + & x_3 & - & x_4 & \leq & 0 \\
& & - & x_2 & + & x_3 & - & x_4 & \leq & 0 \\
& -x_1 & + & x_2 & - & x_3 & + & x_4 & \leq & 1
\end{array}
$$

## The corresponding DEP

$$
\begin{array}{llllllll}
\min & k & & & & & & \\
\text{s.t.} & -k & +2x_1 & - & 3x_3^0 & & & \leq & 0 \\
& -k & +2x_1 & & & - & 3x_3^1 & \leq & 2 \\
& -k & +2x_1 & - & 3x_3^0 & & & \leq & 2 \\
& -k & +2x_1 & & & - & 3x_3^1 & \leq & 4 \\
& & x_1 & + & x_3^0 & & & \leq & 2 \\
& & x_1 & & & + & x_3^1 & \leq & 1 \\
& & -x_1 & + & x_3^0 & & & \leq & 0 \\
& & -x_1 & & & + & x_3^1 & \leq & 0 \\
& & -x_1 & + & x_3^0 & & & \leq & 1 \\
& & -x_1 & & & + & x_3^1 & \leq & 1 \\
& & & & x_3^0 & & & \leq & 0 \\
& & & & & & x_3^1 & \leq & 1 \\
& & & & x_3^0 & & & \leq & 1 \\
& & & & & & x_3^1 & \leq & 2 \\
& & -x_1 & - & x_3^0 & & & \leq & 1 \\
& & -x_1 & & & - & x_3^1 & \leq & 0 \\
& & -x_1 & - & x_3^0 & & & \leq & 0 \\
& & -x_1 & & & - & x_3^1 & \leq & -1
\end{array}
$$

# App: The Deterministic Equivalent Program

## The original QIP

$$
\begin{array}{lrcrcrcrcl}
\min & 2x_1 & - & 2x_2 & - & 3x_3 & - & 2x_4 & & \\
\text{s.t.} & \exists x_1 & & \forall x_2 & & \exists x_3 & & \forall x_4 & : & \\
 & x_1 & + & x_2 & + & x_3 & & & \leq & 2 \\
 & -x_1 & & & + & x_3 & - & x_4 & \leq & 0 \\
 & & - & x_2 & + & x_3 & - & x_4 & \leq & 0 \\
 & -x_1 & + & x_2 & - & x_3 & + & x_4 & \leq & 1
\end{array}
$$

## The corresponding DEP

$$
\begin{array}{llrcrcrcl}
\min & k & & & & & & & \\
\text{s.t.} & -k & +2x_1 & - & 3x_3^0 & & & \leq & 0 \\
 & -k & +2x_1 & & & - & 3x_3^1 & \leq & 2 \\
 & -k & +2x_1 & - & 3x_3^0 & & & \leq & 2 \\
 & -k & +2x_1 & & & - & 3x_3^1 & \leq & 4 \\
 & & x_1 & + & x_3^0 & & & \leq & 2 \\
 & & x_1 & & & + & x_3^1 & \leq & 1 \\
 & & -x_1 & + & x_3^0 & & & \leq & 0 \\
 & & -x_1 & & & + & x_3^1 & \leq & 0 \\
 & & -x_1 & + & x_3^0 & & & \leq & 1 \\
 & & -x_1 & & & + & x_3^1 & \leq & 1 \\
 & & & & x_3^0 & & & \leq & 0 \\
 & & & & & & x_3^1 & \leq & 1 \\
 & & & & x_3^0 & & & \leq & 1 \\
 & & & & & & x_3^1 & \leq & 2 \\
 & & -x_1 & - & x_3^0 & & & \leq & 1 \\
 & & -x_1 & & & - & x_3^1 & \leq & 0 \\
 & & -x_1 & - & x_3^0 & & & \leq & 0 \\
 & & -x_1 & & & - & x_3^1 & \leq & -1
\end{array}
$$

Solution of DEP:

$$k = 2 \qquad x_1 = 1 \qquad x_3^0 = 0 \qquad x_3^1 = 0$$