# Multistage Optimization with the help of Quantified Linear Programming

TECHNISCHE UNIVERSITÄT DARMSTADT

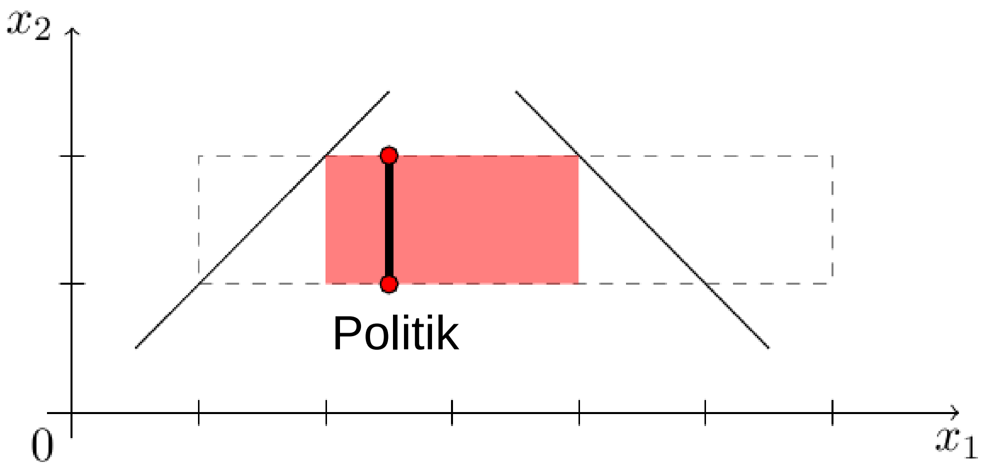
PD Dr. Ulf Lorenz

# Quantified Linear Programs [1)]

$$\exists\, x_1 \in [0,1]\ \forall\, x_2 \in [0,1]\ \exists\, x_3 \in [0,1] :$$

$$\begin{pmatrix} 0 & -1 & -1 \\ -1 & 1 & 1 \\ 2 & 2 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \leq \begin{pmatrix} -1 \\ 1 \\ 3 \end{pmatrix}$$

- Union of all winning policies forms a polytope..




$$\exists x_1 \in [1,6]\ \forall x_2 \in [1,2] : x_1 + x_2 \leq 6 \ \wedge\ x_2 - x_1 \leq 0$$

1) K. Subramani. Analyzing selected quantified integer programs. Springer, LNAI 3097, pages 342–356, 2004.

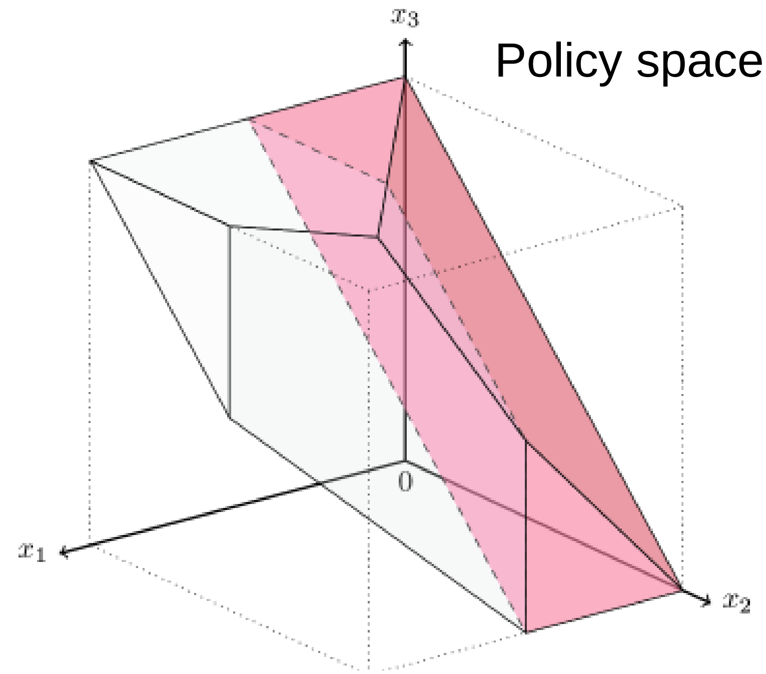
# Quantified Linear Programs

TECHNISCHE UNIVERSITÄT DARMSTADT

Quantified Linear Program (QLP)

- continuous variables
- polyhedral solution space
- relaxation of QIP

Complexity is unknown in general.




- There is a winning policy against the universal player if and only if there is a winning policy against a universal player who is restricted to {0,1}.
- Vertex complexity stays good-natured as long as the number of quantifier changes is limited by a constant.

# Quantified Linear Integer Programs

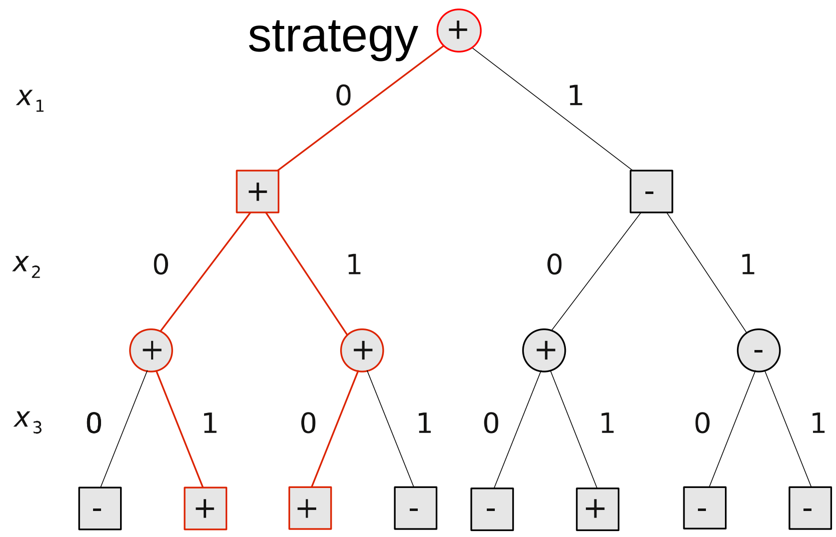
TECHNISCHE UNIVERSITÄT DARMSTADT

$$\exists x_1 \in [0,1]\ \forall x_2 \in [0,1]\ \exists x_3 \in [0,1]\ \text{, x integer:}$$

$$\begin{pmatrix} 0 & -1 & -1 \\ -1 & 1 & 1 \\ 2 & 2 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \leq \begin{pmatrix} -1 \\ 1 \\ 3 \end{pmatrix}$$

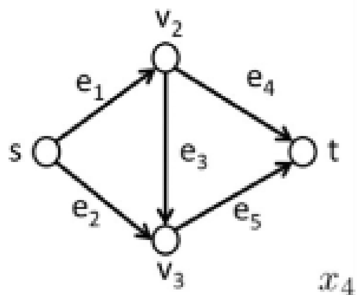
Quantified Integer Program (QIP)

- integral variables
- can be visualized as a game-tree, strategy
- PSPACE-complete



# Example problem: Dynamic Graph Reliability

TECHNISCHE UNIVERSITÄT DARMSTADT




$\exists x_1, x_2 \forall y_{2,4}, y_{2,5} \exists x_3 \forall y_{3,4}, y_{3,5} \exists x_4, x_5, x_\Delta$ (all binary)

$$\left.\begin{aligned} x_1 &= x_4 + x_3 \\ x_2 + x_3 &= x_5 \\ x_1 + x_2 &= 1 \\ x_4 + x_5 &= 1 \end{aligned}\right\}$$ Flow constraints

$$\left.\begin{aligned} x_4 &\le (1 - y_{2,4}) + (1 - x_1) + x_\Delta \\ x_4 &\le (1 - y_{3,4}) + (1 - x_2 - x_3) + x_\Delta \\ x_5 &\le (1 - y_{2,5}) + (1 - x_1) + x_\Delta \\ x_5 &\le (1 - y_{3,5}) + (1 - x_2 - x_3) + x_\Delta \end{aligned}\right\}$$ constraints for existential player

$$2x_\Delta \le y_{3,4} + y_{3,5} + y_{2,4} + y_{2,5} \}$$ Critical constraint for the universal player

Further example problems

- Connect6
- system synthesis of a pump system
- two stage jobshop
- two stage car-sequencing



FLUID SYSTEM TECHNIK

# Quantified Jobshop

TECHNISCHE UNIVERSITÄT DARMSTADT

machines (blue, green, yellow)

Papers 1, 2, 3

Table 1: Jobshop model notation.

| | |
|---|---|
| $J$ | set of jobs |
| $M$ | set of machines |
| $T$ | set of tasks, $T \subseteq J \times M$ |
| $O$ | taskorder, $O \subseteq T \times T$ |
| $s_{j,m}$ | start time (integer) of task $(j,m)$ |
| $d_{j,m}$ | duration of task $(j,m)$ |
| $\delta_{j,m}$ | additional duration of task $(j,m)$ in case of delay |
| $e_{j,m}$ | earliness of task $(j,m)$, i.e., $e = \max\{d^1 - d^2, 0\}$ |
| $\bar{e}$ | mean earliness |
| $m$ | makespan |
| $r_{u,j,m}$ | indicator of unary encoding of retarded task |
| $\tilde{r}_b$ | indicator of binary encoding retarded task |
| $w$ | wrapping indicator for binary to unary translation |

Table 2: Jobshop tasks.

| job | machine | duration | extra |
|---|---|---|---|
| Paper1 | Blue | 45 | 5 |
| Paper1 | Yellow | 10 | 0 |
| Paper2 | Blue | 20 | 5 |
| Paper2 | Green | 10 | 10 |
| Paper2 | Yellow | 34 | 0 |
| Paper3 | Blue | 12 | 0 |
| Paper3 | Green | 17 | 0 |
| Paper3 | Yellow | 28 | 20 |

Table 3: Jobshop order.

| prior task | later task |
|---|---|
| Paper1 Blue | Paper1 Yellow |
| Paper2 Green | Paper2 Blue |
| Paper2 Blue | Paper2 Yellow |
| Paper3 Yellow | Paper3 Blue |
| Paper3 Blue | Paper3 Green |



FLUID SYSTEM TECHNIK

# Quantified Jobshop

TECHNISCHE UNIVERSITÄT DARMSTADT

$$\min \quad m^2 + k\cdot\bar{e} + \frac{1}{M}\cdot m^1 \qquad \text{s.t.} \qquad \exists\, s^1\, y^1\, m^1\, \forall\, \tilde{r}\, \exists\, r\, w,\, s^2\, y^2\, m^2,\, e: \qquad (1)$$

bin. encoding of scenario-index

4 → 01000

$$s^1_{j,m} + d_{j,m} \leq m^1 \qquad \forall\,(j,m)\in T \qquad (2)$$

$$s^1_{i,m} + d_{i,m} \leq s^1_{j,n} \qquad \forall\,(i,m,j,n)\in O \qquad (3)$$

$$s^1_{i,m} + d_{i,m} \leq s^1_{j,m} + M\cdot(1 - y^1_{i,m,j}) \qquad \forall\,(i,m)\in T,\ (j,m)\in T \qquad (4)$$

$$s^1_{j,m} + d_{j,m} \leq s^1_{i,m} + M\cdot y^1_{i,m,j} \qquad \forall\,(i,m)\in T,\ (j,m)\in T \qquad (5)$$

$$\sum_{(u,j,m)\in U} u\cdot r_{j,m} = \sum_{b\in B} 2^b\cdot\tilde{r}_b - |T|\cdot w \quad \wedge \quad \sum_{\substack{u\in U\\(j,m)=T_u}} r_{j,m} \leq 1 \qquad (6)$$

$$s^2_{j,m} + d_{j,m} + \delta_{j,m}\cdot r_{j,m} \leq m^2 \qquad \forall\,(j,m)\in T \qquad (7)$$

$$s^2_{i,m} + d_{i,m} + \delta_{i,m}\cdot r_{i,m} \leq s^2_{j,n} \qquad \forall\,(i,m,j,n)\in O \qquad (8)$$

$$s^2_{i,m} + d_{i,m} + \delta_{i,m}\cdot r_{i,m} \leq s^2_{j,m} + M\cdot(1 - y^2_{i,m,j}) \qquad \forall\,(i,m)\in T,\ (j,m)\in T \qquad (9)$$

$$s^2_{j,m} + d_{j,m} + \delta_{i,m}\cdot r_{i,m} \leq s^2_{i,m} + M\cdot y^2_{i,m,j} \qquad \forall\,(i,m)\in T,\ (j,m)\in T \qquad (10)$$

$$e_{i,m} \geq s^1_{i,m} - s^2_{i,m} \qquad \forall\,(i,m)\in T \qquad (11)$$

$$e_{i,m} \geq 0 \qquad (12)$$

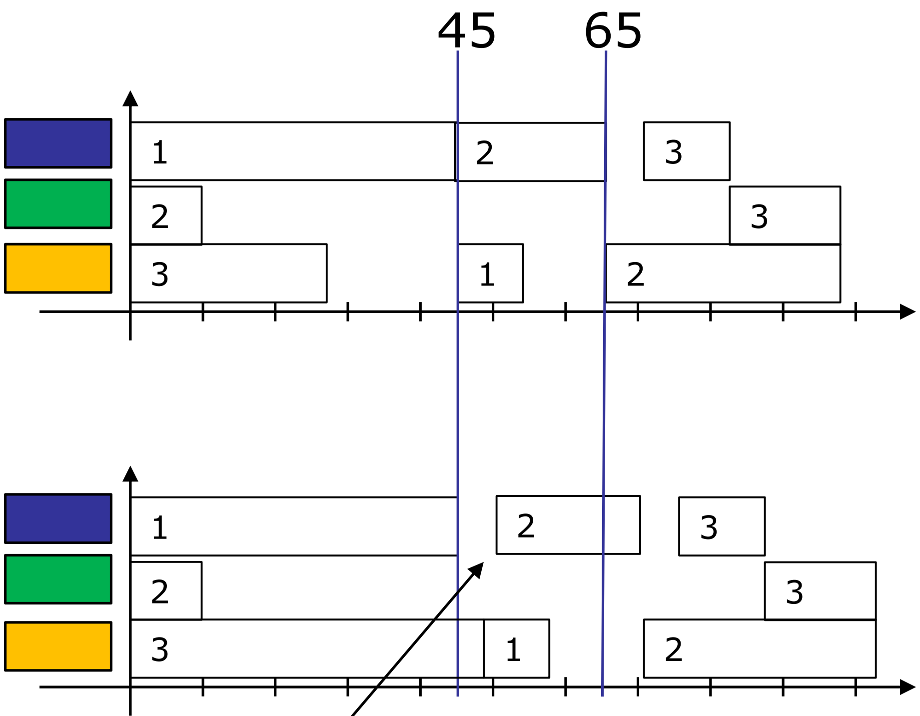
$$\bar{e} = \frac{1}{|T|}\cdot\sum_{(i,m)\in T} e_{i,m} \qquad (13)$$

# Quantified Jobshop

TECHNISCHE UNIVERSITÄT DARMSTADT

Table 4: Solution of the Jobshop Example.

| sc. | 1B | 1Y | 2G | 2B | 2Y | 3Y | 3B | 3G | m.s. |
|---|---|---|---|---|---|---|---|---|---|
| | *first stage solution* | | | | | | | | |
| | 0 | 45 | 0 | 45 | 65 | 0 | 70 | 82 | 99 |
| 1 | 0 | 50 | 0 | 50 | 70 | 0 | 70 | 82 | 104 |
| 2 | 0 | 45 | 0 | 45 | 65 | 0 | 65 | 82 | 99 |
| 3 | 0 | 45 | 0 | 45 | 70 | 0 | 70 | 82 | 104 |
| 4 | 0 | 45 | 0 | 45 | 65 | 0 | 65 | 82 | 99 |
| 5 | 0 | 45 | 0 | 45 | 65 | 0 | 65 | 82 | 99 |
| 6 | 0 | 45 | 0 | 45 | 65 | 0 | 65 | 82 | 99 |
| 7 | 0 | 45 | 0 | 45 | 65 | 0 | 70 | 82 | 99 |
| 8 | 0 | 48 | 0 | 50 | 70 | 0 | 75 | 87 | 104 |

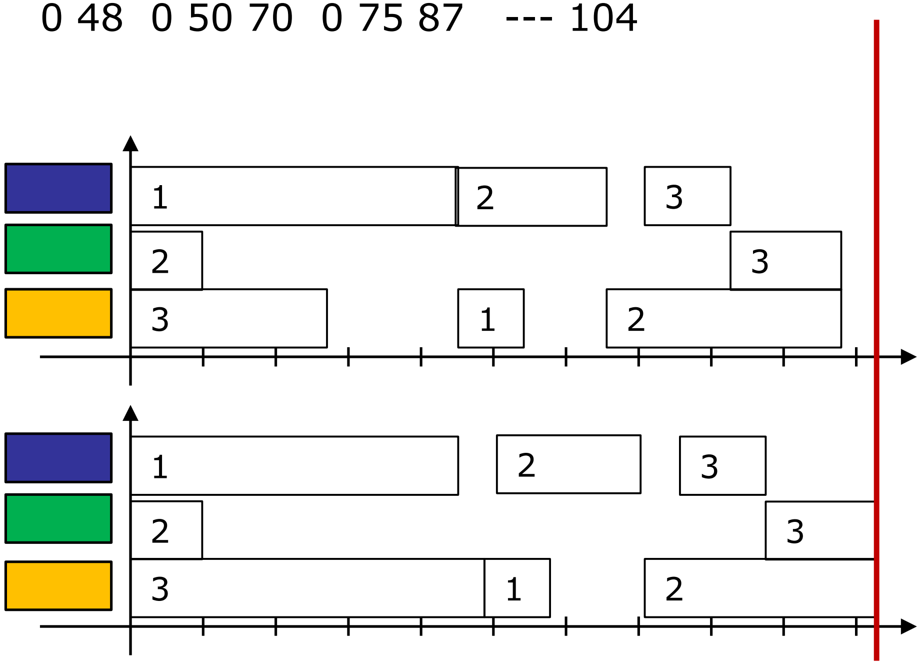



A degree of freedom, i.e. this is no worst-case scenario.

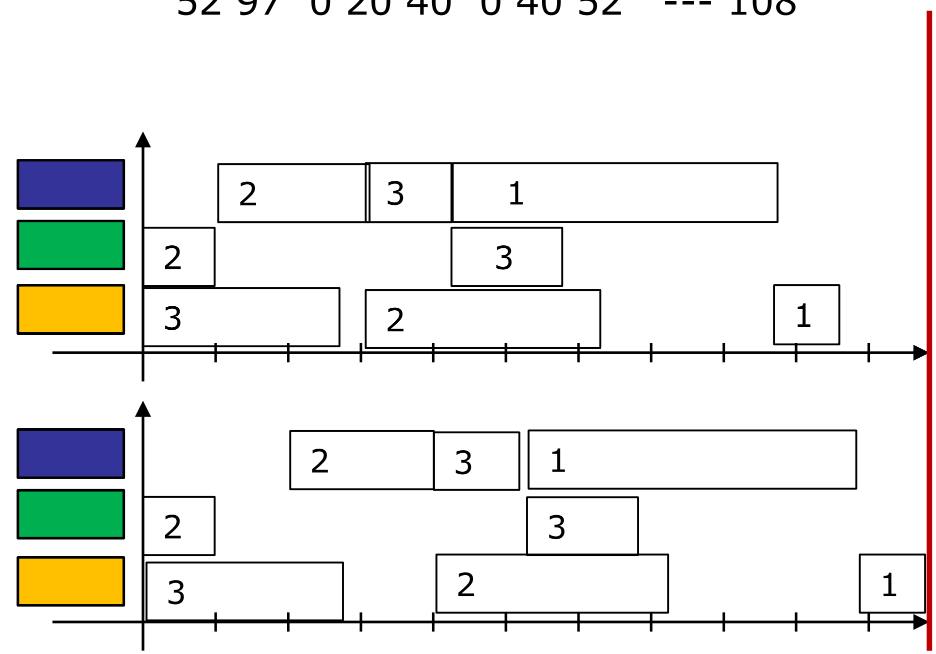
# Quantified Jobshop

Optimal first-stage solution
0 45  0 45 65  0 70 82  --- 99
Scenario 8:
0 48  0 50 70  0 75 87  --- 104

Optimal deterministic solution
42 87  0 10 30  0 30 42  --- 97
Scenario 4:
52 97  0 20 40  0 40 52  --- 108




No scenario is worse than: 104



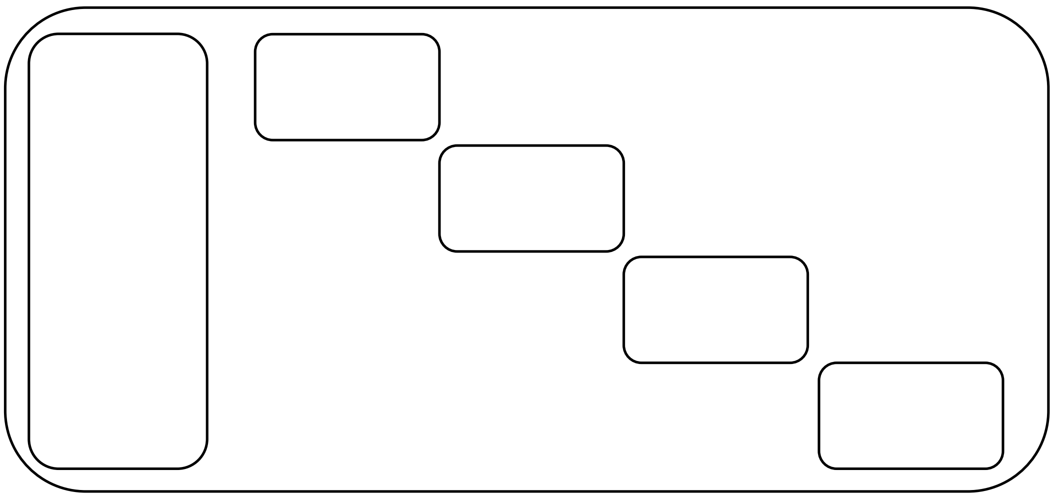

Without disruption awareness: 108

# Algorithms

TECHNISCHE
UNIVERSITÄT
DARMSTADT

## Opportunity 1: Deterministic equivalent

fst-stage | scenarion variables



n universal variables

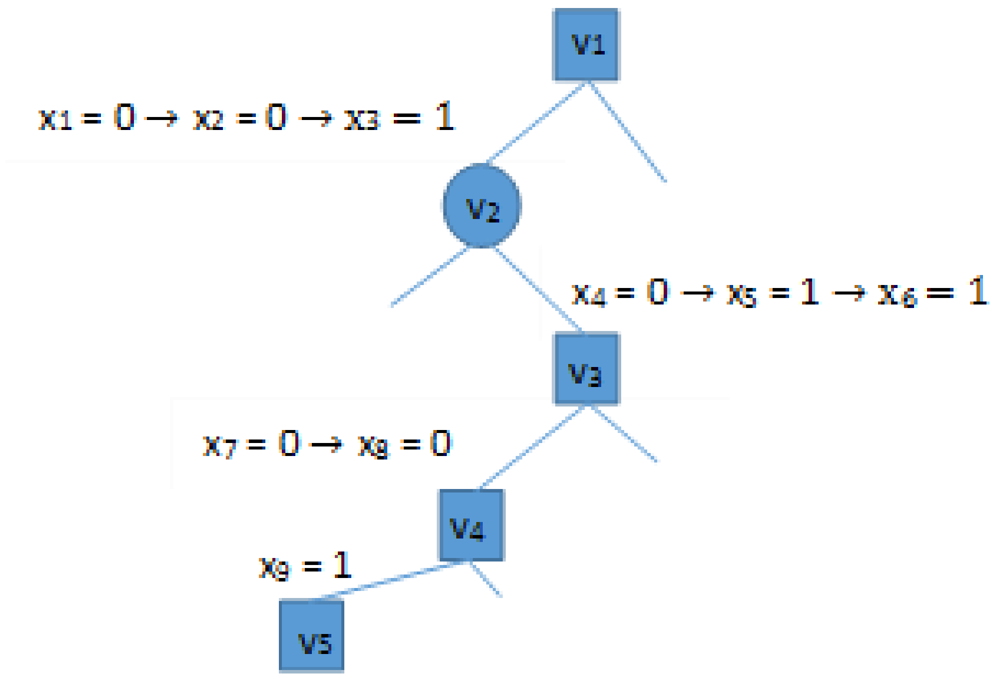
>> n „uncertainty bits“

>> $2^n$ many scenarios

# Algorithms

## Opportunity 2: Tree search and cut algorithm

starting with 0/1-QIPs, run into tree from root downwards

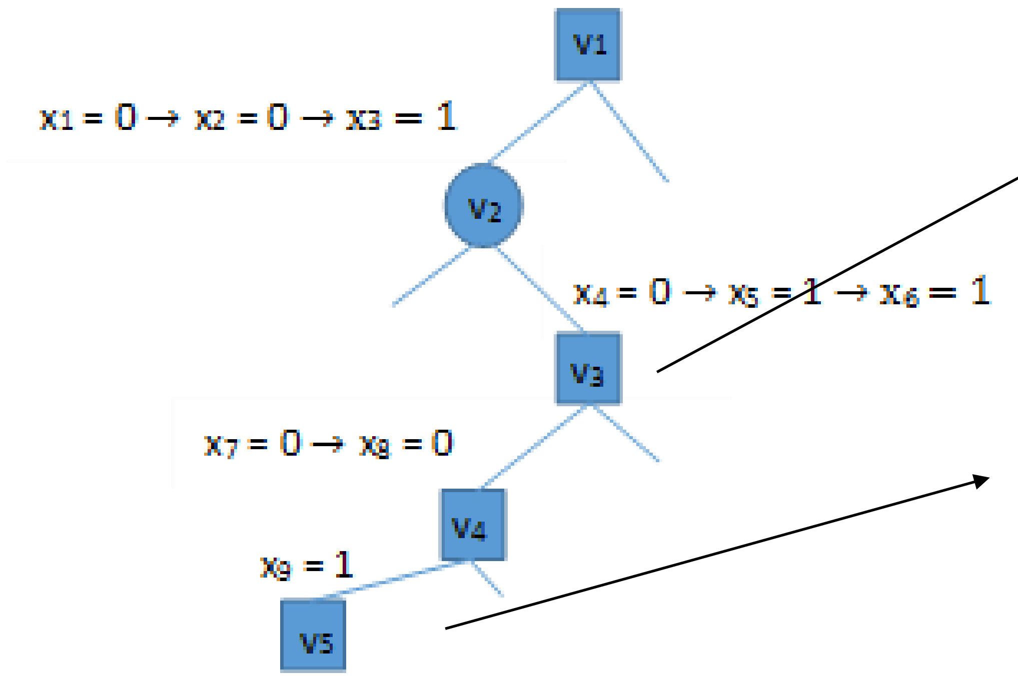


# Algorithms

TECHNISCHE UNIVERSITÄT DARMSTADT

## Opportunity 2: Tree search and cut algorithm

starting with 0/1-QIPs, run into tree from root downwards




solve LP-relaxation
if feasible:
  strengthen with cuts
if infeasible:
  compute backjump

if contradiction detected:
  compute backjump and
  learn cover-cut
  discover earlier implication



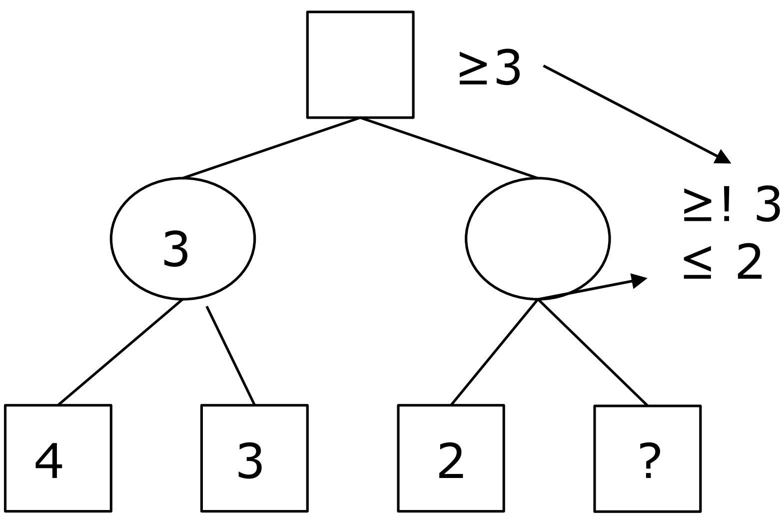
FLUID SYSTEM TECHNIK

# Algorithms

TECHNISCHE
UNIVERSITÄT
DARMSTADT

## Opportunity 2: Tree search and cut algorithm

starting with 0/1-QIPs, run into tree from root downwards




Best first search algorithm?

Curently depth first search
-> Alphabeta algorithm



FLUID SYSTEM TECHNIK

# Algorithms

**Algorithm 1:** A basic alphabeta(int d, int a, int b) routine, sketched

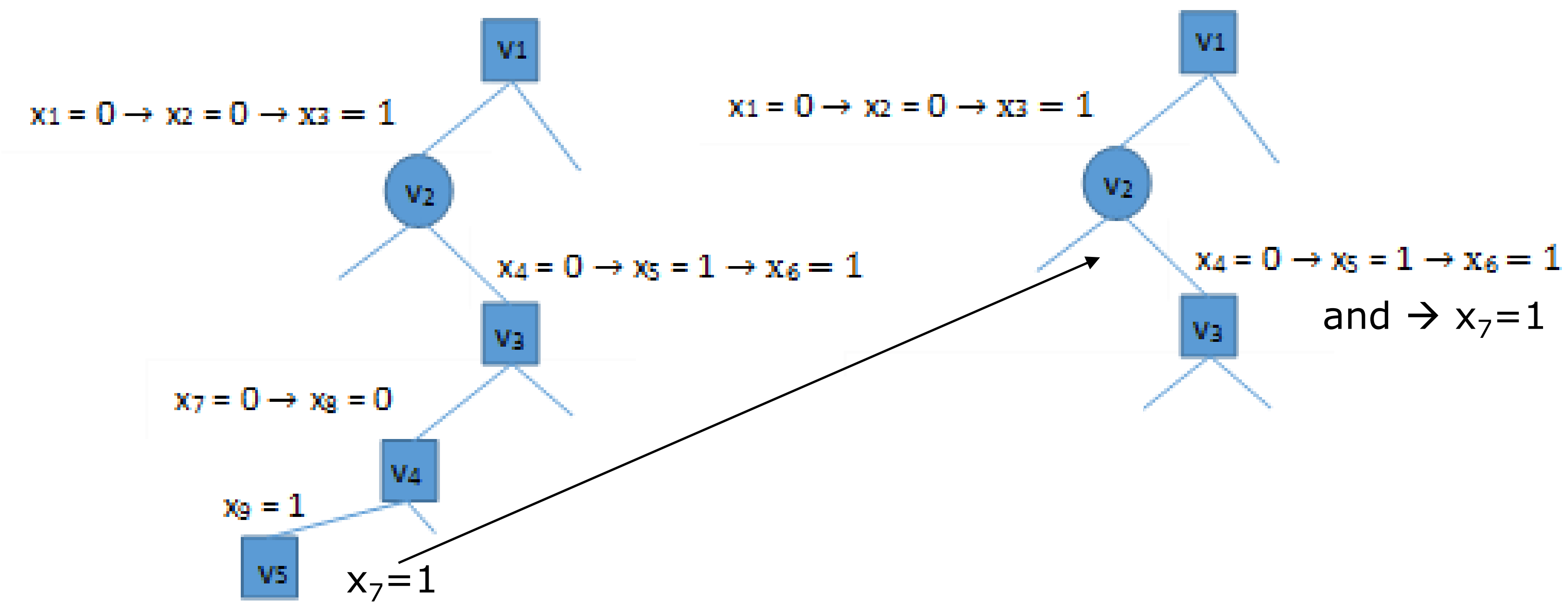
```
compute LP-relaxation, solve it, extract branching variable or cut;
if integer solution found then return objective value ; // leaf reached
if x_i is an existential variable then score := −∞; else score := +∞;
for val_ix from 0 to 1 do // search actually begins ...
    if level_finished(t) then // leave marked recursion levels
        if x_i is an extistential variable then return score ;
        else return −∞ ;
    end
    assign(x_i, val[val_ix], ... );
    v := alphabeta(d-1, fmax(a, score), b);
    unassign(x_i);
    if x_i is an existential variable then
        if v > score then score := v; // existential player maximizes
        if score ≥ b then return score ;
    else
        ... anallogously ...;
    end
end
```

# Algorithms

TECHNISCHE UNIVERSITÄT DARMSTADT

## Opportunity 2: Tree search and cut algorithm

starting with 0/1-QIPs, run into tree from root downwards



# Algorithms

**Algorithm 2:** A local repetition loop extends the basic algorithm with conflict analysis and learning; replaces line 9 in Algorithm 1

```
repeat
    if the current level is marked as finished then leave the recursion level;
    if propagate(confl, confl_var, confl_partner, false) // unit prop. [11] then
        if x_i is an existential variable then
            v = alphabeta(t+1, lsd-1, fmax(a, score), b);
            if v > score then  score := v;
            if score ≥ b then  break;
        else
            ... analogously ...;
        end
    else
        add reason, i.e. a constraint, to the database ; // [11]
        returnUntil(out_target_dec_level) ; // set level_finished(...)
        if  x_i is an existential variable then  return score;  else  return  −∞ ;
    end
until there is no backward implied variable;
```

# Yasol results

| # univ. var. | | Depqbf | DEPCplex | Yasol |
|---|---|---|---|---|
| 1-5 UV | Time | 39185 | 41133s | 73732 |
| | #solved | 312/373 | 315/373 | 259/373 |
| 6-10 UV | Time | 1805s | 1351s | 2146s |
| | #solved | 38/41 | 39/41 | 38/41 |
| 11-15 UV | Time | 7961s | 25297s | 11023s |
| | #solved | 84/97 | 65/97 | 79/97 |
| 16-20 UV | Time | 2609s | 84685s | 5929s |
| | #solved | 166/170 | 37/170 | 167/170 |
| 21+ UV | Time | 16856s | 69600s[3] | 39620s |
| | #solved | 96/116 | 0/116 | 60/116 |
| Σ | Time | 68416s | 194128s | 132350s |
| Σ | #solved | 696/797 | 456/797 | 603/797 |

A collection of 797 QBF instances, grouped concerning the instances' number of universal variables.

# Yasol results

TECHNISCHE UNIVERSITÄT DARMSTADT

| No. UV | DEPCplex | Yasol | DEPScip | DEPCbc |
|---|---|---|---|---|
| 0 (max 1h) | 59 / 59 19520s | 24 / 59 132129s | 48 / 59 55062s | 34 / 59 112958s |
| 1 to 4 (max 900s) | 138 / 138 42s | 103 / 138 39984s | 137 / 138 1072s | 138 / 138 354s |
| 10 to 14 (max 1200s) | 46 / 46 1854s | 46 / 46 972s | 34 / 46 23440s | 23 / 46 38867s |

A collection of 0/1-IPs and artifically constructed 0/1-QIPs, grouped concerning the instances' number of universal variables.



FLUID SYSTEM TECHNIK

# Future research

TECHNISCHE UNIVERSITÄT DARMSTADT

- existential integer variables
- continuous variables in the last stage
- more sophisticated cutting planes
- more models
- best-first search

THANKS FOR YOUR ATTENTION



FLUID SYSTEM TECHNIK