

# Resource constrained shortest path algorithm for EDF short-term thermal production planning problem

Pascal Benchimol<sup>1</sup>   Markus Kruber<sup>2</sup>   Axel Parmentier<sup>3</sup>

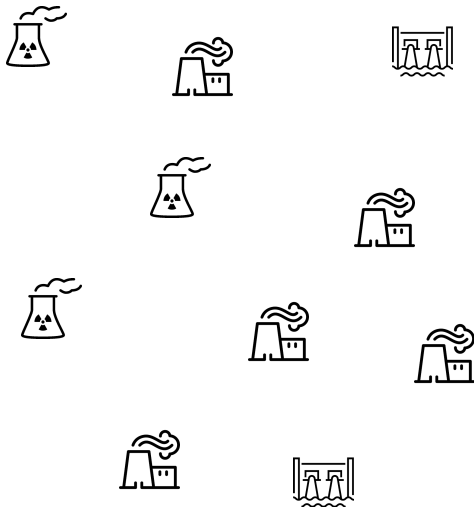
<sup>1</sup>EDF

<sup>2</sup>Chair of Operations Research, RWTH Aachen University

<sup>3</sup>CERMICS, École des Ponts Paristech

March 29, 2017

# EDF - Global Problem



## Generation units

- ▶ ~ 60 nuclear
- ▶ ~ 100 thermal
- ▶ ~ 500 hydraulic

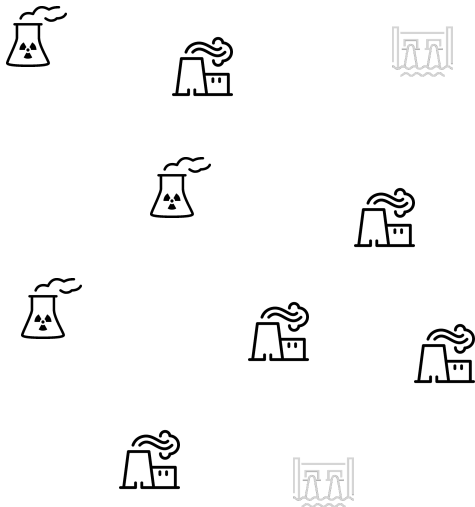
Technically feasible  
production schedules

Min operating cost

Horizon: 24h + 24h

Time limit: 15m

# EDF - Global Problem



## Generation units

- ▶ ~ 60 nuclear
- ▶ ~ 100 thermal
- ▶ ~ 500 hydraulic

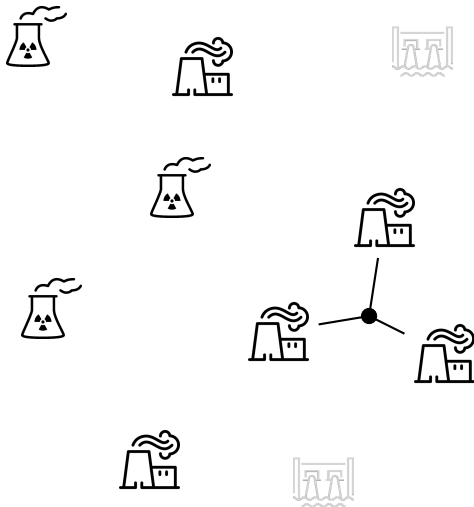
Technically feasible  
production schedules

Min operating cost

Horizon: 24h + 24h

Time limit: 15m

# EDF - Global Problem



## Generation units

- ▶ ~ 60 nuclear
- ▶ ~ 100 thermal
- ▶ ~ 500 hydraulic

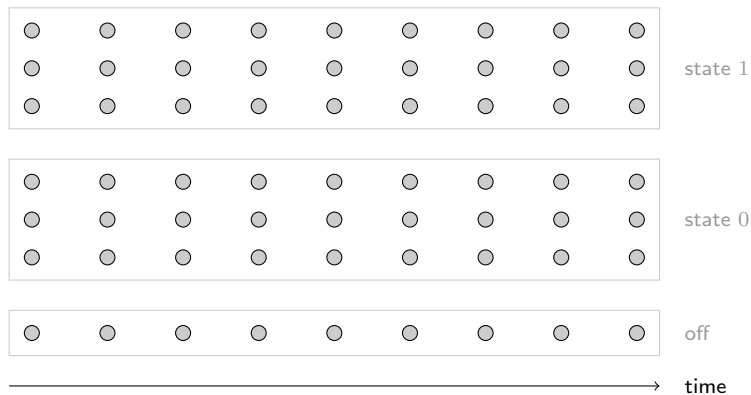
Technically feasible  
production schedules

Min operating cost

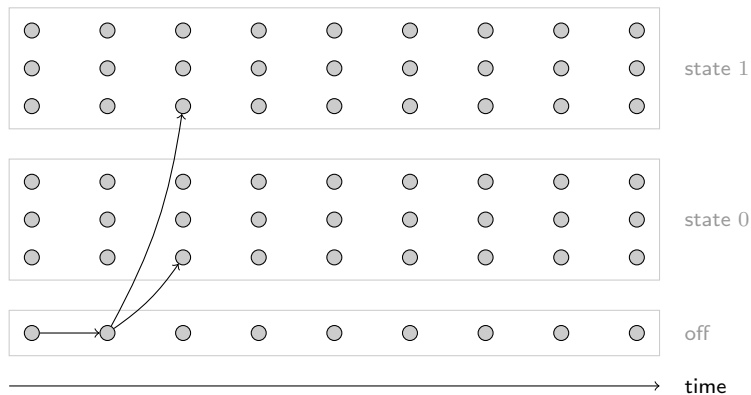
Horizon: 24h + 24h

Time limit: 15m

# Production Plan - Local Problem

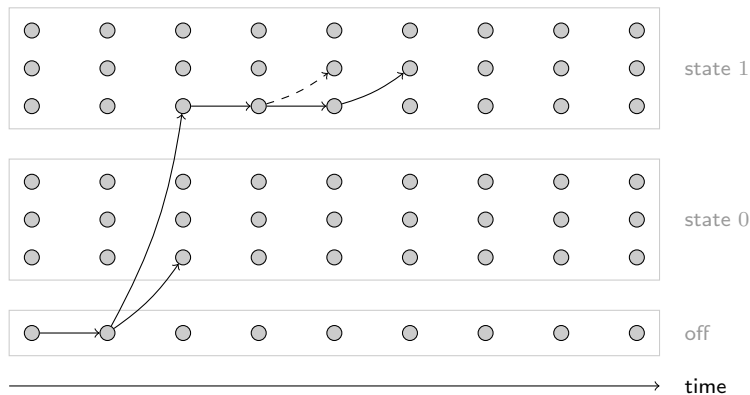


# Production Plan - Local Problem



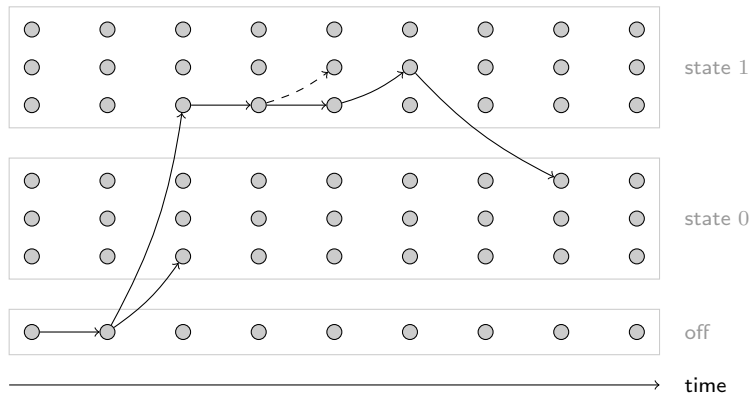
startup · #startups

# Production Plan - Local Problem



startup · #startups · min\_duration\_production\_level

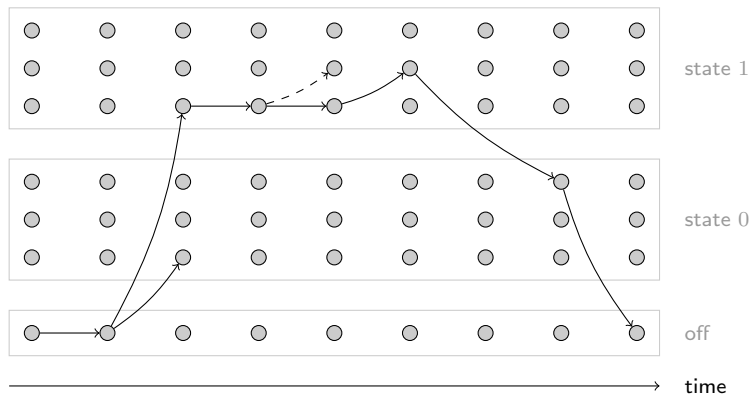
# Production Plan - Local Problem



startup · #startups · min\_duration\_production\_level  
min\_duration\_power\_state · #modulations

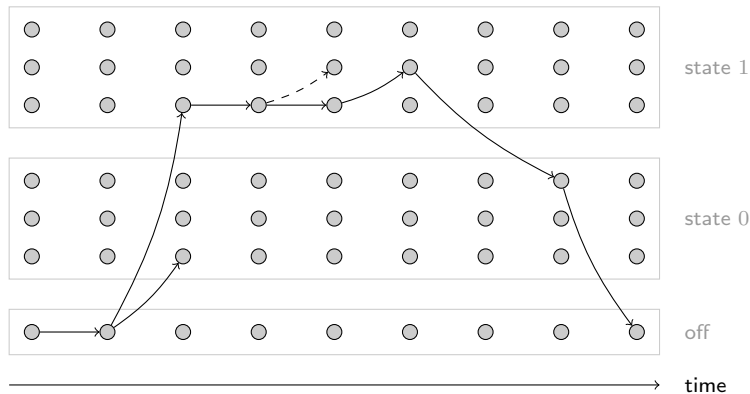


# Production Plan - Local Problem



startup · #startups · min\_duration\_production\_level  
min\_duration\_power\_state · #modulations · shutdown

# Production Plan - Local Problem



startup · #startups · min\_duration\_production\_level  
min\_duration\_power\_state · #modulations · shutdown  
min/max\_power\_state · min/max\_increase/decrease  
#deep\_decreases

# Table of Content

1. The Unit Commitment Problem
2. Resource Constraint Shortest Path Problem
3. Enumeration Algorithms
4. Computational Results
5. Pareto Frontier - An Alternative Approach

# The Unit Commitment Problem (1)

Assume we can generate all technically feasible production plans  $\mathcal{P}_i$  for each plant  $i \in V$ .

$$\begin{array}{ll} \text{maximize} & \sum_{p \in \mathcal{P}} c_p x_p \\ \text{subject to} & \sum_{p \in \mathcal{P}_i} x_p = 1 \quad \forall i \in V \\ & x_p \in \{0, 1\} \quad \forall p \in \mathcal{P} \end{array}$$

# The Unit Commitment Problem (1)

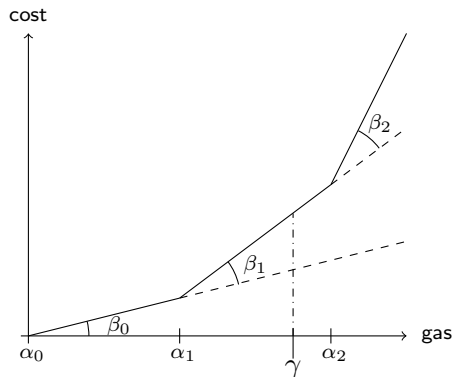
Assume we can generate all technically feasible production plans  $\mathcal{P}_i$  for each plant  $i \in V$ .

$$\begin{aligned} & \text{maximize} && \sum_{p \in \mathcal{P}} c_p x_p \\ & \text{subject to} && \sum_{p \in \mathcal{P}_i} x_p = 1 \quad \forall i \in V \\ & && x_p \in \{0, 1\} \quad \forall p \in \mathcal{P} \end{aligned}$$

What makes the problem more difficult?

- ▶ Linking constraints: Shared gas stock
- ▶ Piecewise linear cost function on gas consumption  $\gamma$

# Excursion: Piecewise Linear Objective Function



$$\begin{aligned} \min \quad & \sum_k \beta_k \xi_k \\ \text{s.t.} \quad & \gamma - \alpha_k \leq \xi_k \quad \forall k \\ & \xi_k \geq 0 \quad \forall k \end{aligned}$$

# The Unit Commitment Problem (2)

Let  $j$  describe a group of plants.

$$\begin{array}{ll} \text{maximize} & \sum_{p \in \mathcal{P}} c_p x_p - \sum_j \sum_k \beta_{jk} \xi_{jk} \\ \text{subject to} & \sum_{p \in \mathcal{P}_i} x_p = 1 \quad \forall i \in V \end{array}$$

$$\begin{array}{ll} \sum_{p \in \mathcal{P}_j} \gamma_p x_p - \alpha_{jk} \leq \xi_{jk} & \forall j, k \\ x_p \in \{0, 1\} & \forall p \in \mathcal{P} \\ \xi_{jk} \geq 0 & \forall j, k \end{array}$$

# The Unit Commitment Problem (2)

Let  $j$  describe a group of plants.

$$\begin{aligned} & \text{maximize} && \sum_{p \in \mathcal{P}} c_p x_p - \sum_j \sum_k \beta_{jk} \xi_{jk} \\ & \text{subject to} && \sum_{p \in \mathcal{P}_i} x_p = 1 && \forall i \in V \\ & && \sum_{p \in \mathcal{P}_j} \gamma_p x_p \geq g_j && \forall j \\ & && \sum_{p \in \mathcal{P}_j} \gamma_p x_p \leq G_j && \forall j \\ & && \sum_{p \in \mathcal{P}_j} \gamma_p x_p - \alpha_{jk} \leq \xi_{jk} && \forall j, k \\ & && x_p \in \{0, 1\} && \forall p \in \mathcal{P} \\ & && \xi_{jk} \geq 0 && \forall j, k \end{aligned}$$



## The Unit Commitment Problem (2)

Let  $j$  describe a group of plants.

$$\begin{aligned} & \text{maximize} && \sum_{p \in \mathcal{P}} c_p x_p - \sum_j \sum_k \beta_{jk} \xi_{jk} \\ & \text{subject to} && \sum_{p \in \mathcal{P}_i} x_p = 1 && \forall i \in V \\ & && \sum_{p \in \mathcal{P}_j} \gamma_p x_p \geq g_j && \forall j \\ & && \sum_{p \in \mathcal{P}_j} \gamma_p x_p \leq G_j && \forall j \\ & && \sum_{p \in \mathcal{P}_j} \gamma_p x_p - \alpha_{jk} \leq \xi_{jk} && \forall j, k \\ & && x_p \in \{0, 1\} && \forall p \in \mathcal{P} \\ & && \xi_{jk} \geq 0 && \forall j, k \end{aligned}$$

Why not solving it directly?

# Column Generation in General

1. Start with subset of variables  $\tilde{\mathcal{P}} \subset \mathcal{P}$
2. Solve relaxed version of primal problem  $x_p \geq 0$
3. Get dual variables for each constraint  $\lambda, \mu, \pi$
4. Find violated constraint in the dual problem
5. Add variable  $x_p$  to the problem  $\tilde{\mathcal{P}} \cup \{p\}$

# Applied Column Generation

Start with an initial set of feasible paths  $\tilde{\mathcal{P}} \subset P$ .

$$\begin{array}{ll} \text{maximize} & \sum_{p \in \tilde{\mathcal{P}}} c_p x_p - \sum_j \sum_k \beta_{jk} \xi_{jk} \\ \text{subject to} & \sum_{p \in \tilde{\mathcal{P}}_i} x_p = 1 \quad \forall i \in V \\ & \sum_{p \in \tilde{\mathcal{P}}_j} \gamma_p x_p \geq g_j \quad \forall j \\ & \sum_{p \in \tilde{\mathcal{P}}_j} \gamma_p x_p \leq G_j \quad \forall j \\ & \sum_{p \in \tilde{\mathcal{P}}_j} \gamma_p x_p - \alpha_{jk} \leq \xi_{jk} \quad \forall j, k \\ & x_p \geq 0 \quad \forall p \in \tilde{\mathcal{P}} \\ & \xi_{jk} \geq 0 \quad \forall j, k \end{array}$$

# Applied Column Generation

Start with an initial set of feasible paths  $\tilde{\mathcal{P}} \subset P$ .

$$\begin{array}{ll} \text{maximize} & \sum_{p \in \tilde{\mathcal{P}}} c_p x_p - \sum_j \sum_k \beta_{jk} \xi_{jk} \\ \text{subject to} & \sum_{p \in \tilde{\mathcal{P}}_i} x_p = 1 \quad \forall i \in V \quad \lambda_i \\ & \sum_{p \in \tilde{\mathcal{P}}_j} \gamma_p x_p \geq g_j \quad \forall j \quad \mu_j^0 \\ & \sum_{p \in \tilde{\mathcal{P}}_j} \gamma_p x_p \leq G_j \quad \forall j \quad \mu_j^1 \\ & \sum_{p \in \tilde{\mathcal{P}}_j} \gamma_p x_p - \alpha_{jk} \leq \xi_{jk} \quad \forall j, k \quad \pi_{jk} \\ & x_p \geq 0 \quad \forall p \in \tilde{\mathcal{P}} \\ & \xi_{jk} \geq 0 \quad \forall j, k \end{array}$$

# Dual and Pricing Problem

## Dual

$$\begin{aligned} \min \quad & - \sum_{i \in V} \lambda_i + \sum_{j \in J} -\mu_j^0 g_j + \mu_j^1 G_j + \sum_k \pi_{jk} \alpha_{jk} \\ \text{s.t.} \quad & -\beta_{jk} + \pi_{jk} \leq 0 \quad \forall j, k \\ & c_p + \lambda_{i(p)} + \gamma_p (\mu_{j(p)}^0 - \mu_{j(p)}^1 - \sum_k \pi_{j(p)k}) \leq 0 \quad \forall p \in \tilde{\mathcal{P}} \end{aligned}$$

# Dual and Pricing Problem

## Dual

$$\begin{aligned} \min \quad & - \sum_{i \in V} \lambda_i + \sum_{j \in J} -\mu_j^0 g_j + \mu_j^1 G_j + \sum_k \pi_{jk} \alpha_{jk} \\ \text{s.t.} \quad & -\beta_{jk} + \pi_{jk} \leq 0 \quad \forall j, k \\ & c_p + \lambda_{i(p)} + \gamma_p (\mu_{j(p)}^0 - \mu_{j(p)}^1 - \sum_k \pi_{j(p)k}) \leq 0 \quad \forall p \in \tilde{P} \end{aligned}$$

## Pricing

$$\max_{p \in \tilde{P}} c_p + \lambda_{i(p)} + \gamma_p (\mu_{j(p)}^0 - \mu_{j(p)}^1 - \sum_k \pi_{j(p)k})$$

# Dual and Pricing Problem

## Dual

$$\begin{aligned} \min \quad & - \sum_{i \in V} \lambda_i + \sum_{j \in J} -\mu_j^0 g_j + \mu_j^1 G_j + \sum_k \pi_{jk} \alpha_{jk} \\ \text{s.t.} \quad & -\beta_{jk} + \pi_{jk} \leq 0 \quad \forall j, k \\ & c_p + \lambda_{i(p)} + \gamma_p (\mu_{j(p)}^0 - \mu_{j(p)}^1 - \sum_k \pi_{j(p)k}) \leq 0 \quad \forall p \in \tilde{P} \end{aligned}$$

## Pricing

$$\max_{p \in \tilde{P}} c_p + \lambda_{i(p)} + \gamma_p (\mu_{j(p)}^0 - \mu_{j(p)}^1 - \sum_k \pi_{j(p)k})$$

## Main Task

Solve pricing problem fast!

# Table of Content

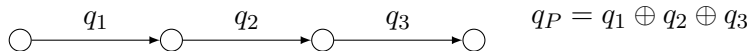
1. The Unit Commitment Problem
  - 1.1 Model
  - 1.2 Column Generation
2. Resource Constraint Shortest Path Problem
3. Enumeration Algorithms
  - 3.1 Enumeration algorithms
  - 3.2 Bounds
4. Computational Results
5. Pareto Frontier - An Alternative Approach



# Shortest Path in an Ordered Monoid

For each arc  $a$  a resource  $q_a \in \mathcal{R}$

- ▶ Associative binary operator  $\oplus$ : path resources
- ▶ Neutral element  $0$ : empty path



$(\mathcal{R}, \oplus)$  is a monoid.

- ▶ An order  $\preceq$  compatible with  $\oplus$ :  $q \preceq \tilde{q} \Rightarrow \begin{cases} r \oplus q \preceq r \oplus \tilde{q} \\ q \oplus r \preceq \tilde{q} \oplus r \end{cases}$

$(\mathcal{R}, \oplus, \preceq)$  is an ordered monoid.

- ▶ Non-decreasing cost  $c$  and constraint  $\rho$  functions.

# Shortest Path with Resources in an Ordered Monoid

Given an ordered monoid  $(\mathcal{R}, \oplus, \preceq)$

Input:

- ▶ Digraph  $D = (V, A)$
- ▶ Two vertices  $o, d \in V$
- ▶ Resources  $q_a \in \mathcal{R}$
- ▶ Two non-decreasing oracles  $c : \mathcal{R} \rightarrow \mathbb{R}$   
 $\rho : \mathcal{R} \rightarrow \{0, 1\}$

Output:

- ▶ An  $o$ - $d$  path  $P$  such that

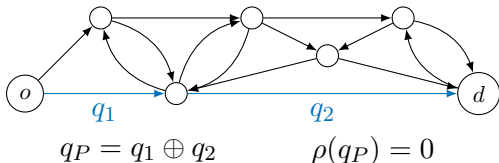
$$\rho\left(\bigoplus_{a \in P} q_a\right) = 0$$

which minimizes

$$c\left(\bigoplus_{a \in P} q_a\right)$$

Cost and constraint(s):

- ▶ non-linear(s)



# Example 1: Usual Resource Constrained Shortest Path

## Input:

- ▶ Digraph  $D = (V, A)$
- ▶ Origin  $o$ , Destination  $d$
- ▶ Costs  $c_a \in \mathbb{R}$
- ▶ Weights  $w_a^i \in \mathbb{R}$  for  $i \in [n]$
- ▶ Thresholds  $W^i \in \mathbb{R}$  for  $i \in [n]$

## Output:

- ▶ An  $o$ - $d$  path  $P$  such that
$$\sum_{a \in P} w_a^i \leq W^i \quad \forall i \in [n]$$
which minimizes 
$$\sum_{a \in P} c_a$$

# Example 1: Usual Resource Constrained Shortest Path

## Input:

- ▶ Digraph  $D = (V, A)$
- ▶ Origin  $o$ , Destination  $d$
- ▶ Costs  $c_a \in \mathbb{R}$
- ▶ Weights  $w_a^i \in \mathbb{R}$  for  $i \in [n]$
- ▶ Thresholds  $W^i \in \mathbb{R}$  for  $i \in [n]$

## Output:

- ▶ An  $o$ - $d$  path  $P$  such that
$$\sum_{a \in P} w_a^i \leq W^i \quad \forall i \in [n]$$
which minimizes 
$$\sum_{a \in P} c_a$$

## Model:

- ▶  $\mathcal{R} = \mathbb{R}^{n+1}$
- ▶  $q_a = (c_a, w_a^1, \dots, w_a^n)$
- ▶  $c : ((q^0, \dots, q^n)) \mapsto q^0$
- ▶  $\rho : ((q^0, \dots, q^n)) \mapsto \max_i \mathbb{1}_{q^i > W^i}$

## Example 2: Restricting Startups

### Input:

- ▶ Digraph  $D = (V, A)$
- ▶ Origin  $o$ , destination  $d$
- ▶  $w_a = \begin{cases} 1, & \text{if startup arc,} \\ 0, & \text{otherwise.} \end{cases}$
- ▶ Max startups  $W^{\text{start}}$

### Output:

- ▶ An  $o$ - $d$  path  $P$  of minimum cost such that the number of startups per plant is not greater than  $W^{\text{start}}$ .

## Example 2: Restricting Startups

### Input:

- ▶ Digraph  $D = (V, A)$
- ▶ Origin  $o$ , destination  $d$
- ▶  $w_a = \begin{cases} 1, & \text{if startup arc,} \\ 0, & \text{otherwise.} \end{cases}$
- ▶ Max startups  $W^{\text{start}}$

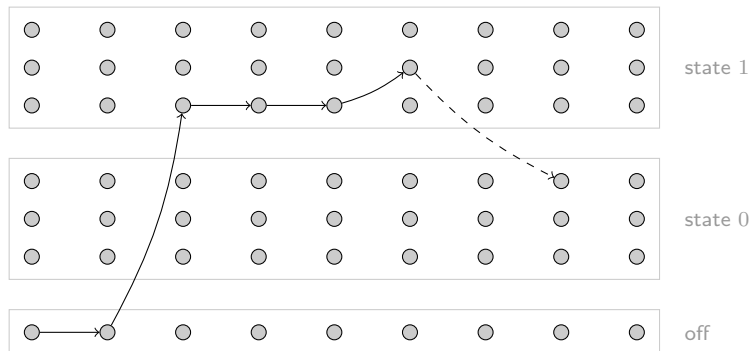
### Output:

- ▶ An  $o$ - $d$  path  $P$  of minimum cost such that the number of startups per plant is not greater than  $W^{\text{start}}$ .

### Model:

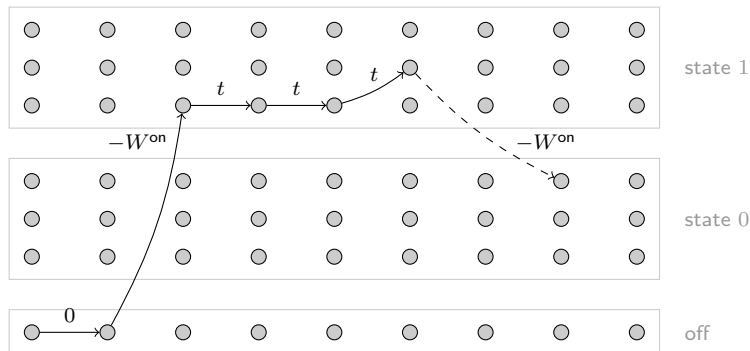
- ▶  $\mathcal{R} = \mathbb{R}^2$
- ▶  $q_a = (c_a, w_a)$
- ▶  $c : ((q^0, q^1)) \mapsto q^0$
- ▶  $\rho : ((q^0, q^1)) \mapsto \mathbb{1}_{q^1 > W^{\text{start}}}$

## Example 3: Minimum Duration in Online State



Stay in online state for at least  $W^{\text{on}}$ .

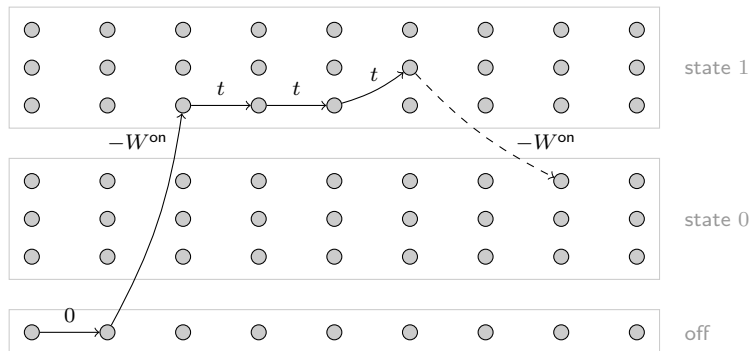
## Example 3: Minimum Duration in Online State



Stay in online state for at least  $W^{\text{on}}$ .



## Example 3: Minimum Duration in Online State



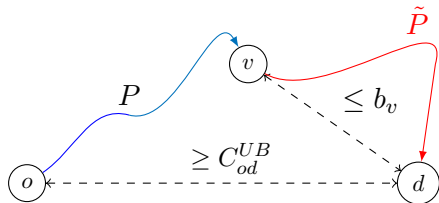
Stay in online state for at least  $W^{\text{on}}$ .

$$(c_a, w_a^1) \oplus (c_{a'}, w_{a'}^1) = \begin{cases} \infty & , \text{if } w_a^1 < 0 \wedge w_{a'}^1 < 0, \\ w_{a'}^1 & , \text{if } w_a^1 \geq 0 \wedge w_{a'}^1 < 0, \\ w_a^1 + w_{a'}^1 & , \text{otherwise} \end{cases}$$

# Table of Content

1. The Unit Commitment Problem
  - 1.1 Model
  - 1.2 Column Generation
2. Resource Constraint Shortest Path Problem
3. Enumeration Algorithms
  - 3.1 Enumeration algorithms
  - 3.2 Bounds
4. Computational Results
5. Pareto Frontier - An Alternative Approach

# Usual A\* algorithm



▶  $q_P \in \mathbb{R}$

▶  $C_{od}^{UB} \geq \min_{P \in \mathcal{P}_{o,d}} q_P$

▶  $b_v \leq q_P, \forall P \in \mathcal{P}_{vd}$

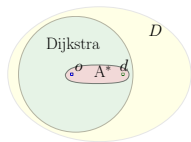
A path  $P \in \mathcal{P}_{ov}$  satisfying  $q_P + b_v > C_{od}^{UB}$  is not the subpath of an optimal path.

## A\* algorithm: a Branch & Bound

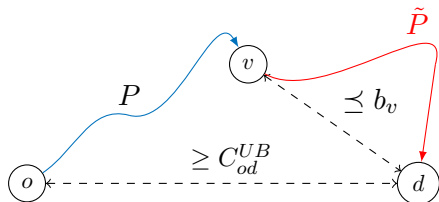
- ▶ Generate all the paths satisfying

$$q_P + b_v \leq C_{od}^{UB}$$

- ▶ Update  $C_{od}^{UB}$



# Generalized A\* algorithm



►  $q_P \in \mathcal{R}$

►  $C_{od}^{UB} \geq \min_{P|\rho(P)=0} c(q_P)$

►  $b_v \leq q_{\tilde{P}}, \forall \tilde{P} \in \mathcal{P}_{vd}$

A path  $P \in \mathcal{P}_{ov}$  satisfying  $c(q_P \oplus b_v) > C_{od}^{UB}$  or  $\rho(q_P \oplus b_v) = 1$  is not the subpath of an optimal path.

## Generalized A\* Algorithm: a Branch & Bound

- Generate all the paths satisfying

$$c(q_P \oplus b_v) \leq C_{od}^{UB} \quad \text{and} \quad \rho(q_P \oplus b_v) = 0 \quad (\text{Low})$$

- Update  $C_{od}^{UB}$

# Generic enumeration algorithm

## Preprocessing.

$L \leftarrow$  empty path in  $o$

$c_{od}^{UB} \leftarrow +\infty$ .

While  $L$  is not empty:

- ▶ Extract from  $L$  a path  $P$  of **minimum key**.
- ▶ If  $v = d$  and  $\rho(P) = 0$ ,  
 $c_{od}^{UB} \leftarrow \min(c_{od}^{UB}, c(P))$ .
- ▶ **Test if  $P$  must be extended**. If yes:
  - ▶ for each  $a \in \delta^+(v)$ , add  $P + a$  to  $L$ .

$L$ : cand. paths list.  
 $c_{od}^{UB}$ : Upper bound on optimal path cost.  
 $v$ : destination of  $P$ .

## Add. structures

$b_v$ : lower bound on  $v$ - $d$  paths  $q_P$   
 $M_v$ : list of non dominated  $o$ - $v$  paths

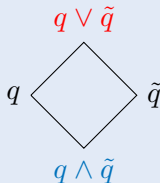
Algorithm	Test	Key
Generalized A*	(Low)	$c(q_P \oplus b_v)$
Label dominance	(Dom)	$c(q_P)$
Label correcting	(Dom), (Low)	$c(q_P \oplus b_v)$

## Definition: *lattice*

A partially ordered set  $(\mathcal{R}, \preceq)$  is a lattice if any pair  $(q, \tilde{q})$  admits:

A greatest lower bound  
or *meet* denoted  $q \wedge \tilde{q}$

$$\left. \begin{array}{l} b \preceq q \\ b \preceq \tilde{q} \end{array} \right\} \Leftrightarrow b \preceq q \wedge \tilde{q}$$



A least upper bound or  
*join* denoted  $q \vee \tilde{q}$

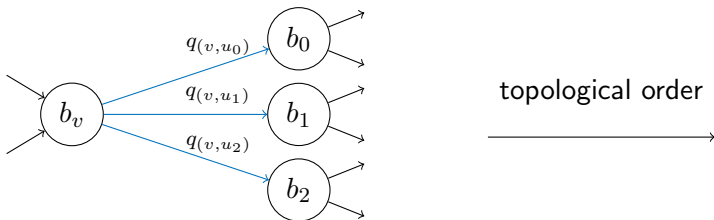
$$\left. \begin{array}{l} b \succeq q \\ b \succeq \tilde{q} \end{array} \right\} \Leftrightarrow b \succeq q \vee \tilde{q}$$

Example:

$(\mathbb{R}^2, \leq)$  endowed  $\leq$  with the product order

- ▶  $q \wedge \tilde{q} = (\min(q_1, \tilde{q}_1), \min(q_2, \tilde{q}_2))$
- ▶  $q \vee \tilde{q} = (\max(q_1, \tilde{q}_1), \max(q_2, \tilde{q}_2))$

# Generalized Ford-Bellman algorithm



If there is no cycles of negative cost,  $b_v$  can be computed by the **generalized dynamic programming equation**:

$$b_v = \begin{cases} 0 & , \text{if } v = d \\ \bigwedge \left( b_v, \bigwedge_{u \in N^+(v)} (q(v, u) \oplus b_u) \right) & , \text{if } v \neq d \end{cases}$$

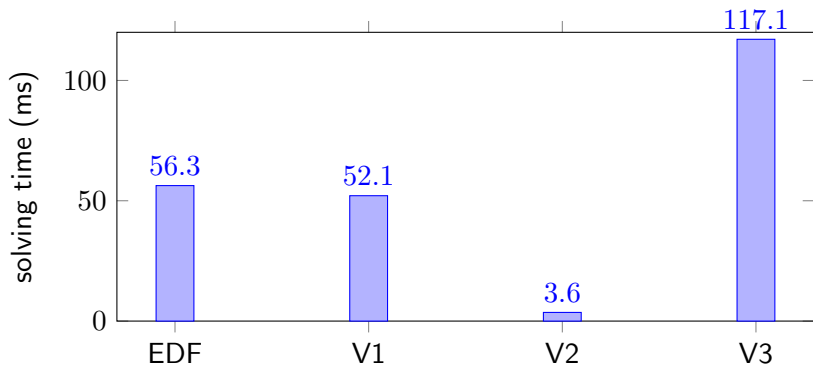
# Table of Content

1. The Unit Commitment Problem
  - 1.1 Model
  - 1.2 Column Generation
2. Resource Constraint Shortest Path Problem
3. Enumeration Algorithms
  - 3.1 Enumeration algorithms
  - 3.2 Bounds
4. Computational Results
5. Pareto Frontier - An Alternative Approach



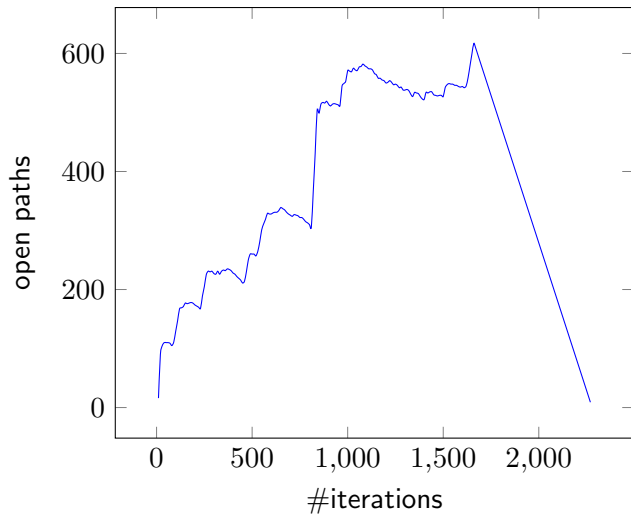
- ▶ Plants
  - ▶ 8 gas plants (3, 3, 2)
  - ▶ 97 non-gas plants
- ▶ Size of graph (depends on the model)
  - ▶ ~ 2.000 nodes
  - ▶ ~ 10.000 arcs

# Solving Non-Gas Subproblem



selection	node	node	path	path
key	early date	early date	$q_p \oplus b_v$	$q_p$
#iter	2.100	1.914	3.596	230.763
#dis dom	190k	137k	9k	201k
#dis bound	0	79k	9k	1.029k
#od paths	327	14	1	899
speedup	1.00x	0.93x	0.06x	2.08x

# An Attempt to Explain



## Pricing

$$\max_{p \in P} c_p + \lambda_{i(p)} + \gamma_p (\mu_{j(p)}^0 - \mu_{j(p)}^1 - \sum_k \pi_{j(p)k})$$

dual	V2 (ms)	EDF (ms)
0	0.15068	44.0543
-1	1.26815	43.0639
-5	5.45315	52.6267
-10	7.33129	40.8962
-50	1.05291	28.4392

# Column Generation of Gas Plants



# Table of Content

1. The Unit Commitment Problem
  - 1.1 Model
  - 1.2 Column Generation
2. Resource Constraint Shortest Path Problem
3. Enumeration Algorithms
  - 3.1 Enumeration algorithms
  - 3.2 Bounds
4. Computational Results
5. Pareto Frontier - An Alternative Approach

# What was the Problem?

Assume we can generate all technically feasible production plans  $\mathcal{P}_i$  for each plant  $i \in V$ .

$$\begin{array}{ll} \text{maximize} & \sum_{p \in \mathcal{P}} c_p x_p - \sum_j \sum_k \beta_{jk} \xi_{jk} \\ \text{subject to} & \dots \end{array}$$

# What was the Problem?

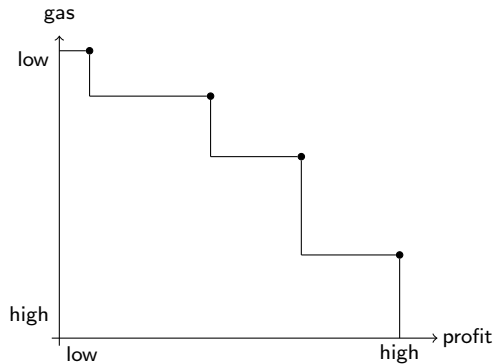
Assume we can generate all technically feasible production plans  $\mathcal{P}_i$  for each plant  $i \in V$ .

$$\begin{aligned} & \text{maximize} && \sum_{p \in \mathcal{P}} c_p x_p - \sum_j \sum_k \beta_{jk} \xi_{jk} \\ & \text{subject to} && \dots \end{aligned}$$

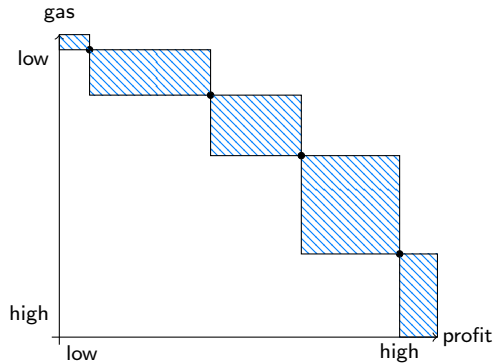
Try to compute all non-dominated paths based on profit and gas consumption.



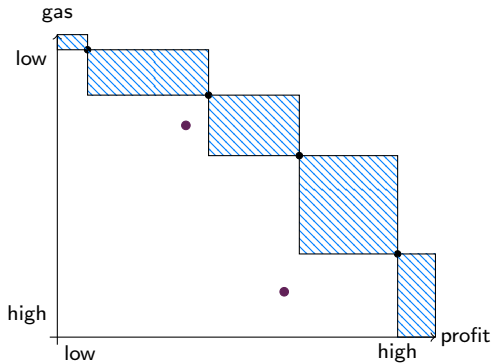
# Pareto Frontier



# Pareto Frontier

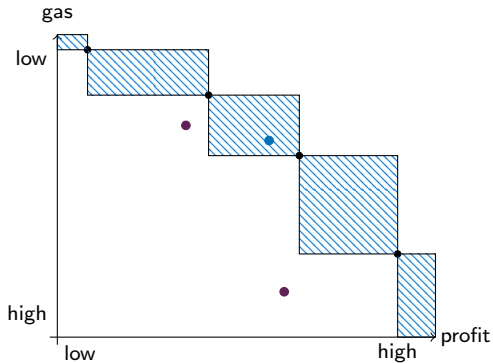


# Pareto Frontier



dominated by pareto

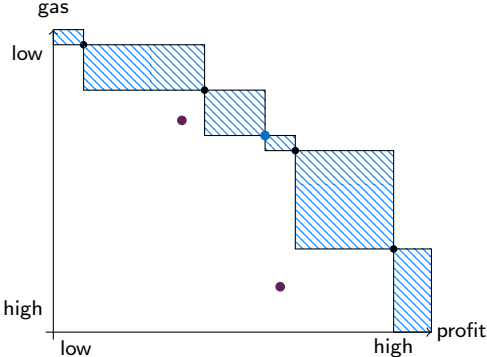
# Pareto Frontier



dominated by pareto

update pareto

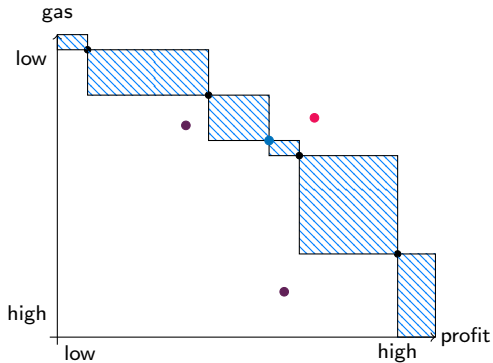
# Pareto Frontier



dominated by pareto

update pareto

# Pareto Frontier

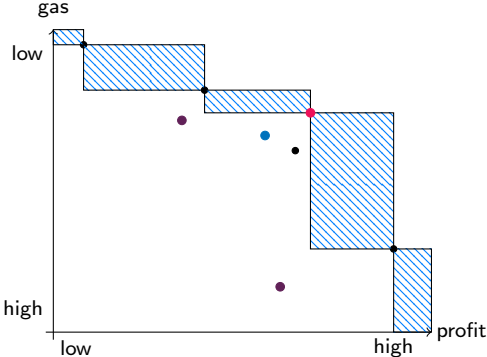


dominated by pareto

update pareto

dominates pareto

# Pareto Frontier



dominated by pareto

update pareto

dominates pareto

# How to Compute? RCSP

Talking about failures!

Bound test before:

$$\text{discard if: } c(q_P \oplus b_v) \leq C_{od}^{LB} \quad (\text{Low})$$



# How to Compute? RCSP

Talking about failures!

Bound test before:

$$\text{discard if: } c(q_P \oplus b_v) \leq C_{od}^{LB} \quad (\text{Low})$$

Bound test after:

for (*profit*, *gas*) in pareto:

if  $c_{\text{profit}}(q_P \oplus b_v) \leq \textit{profit}$  and  $c_{\text{gas}}(q_P \oplus b_v) \geq \textit{gas}$   
discard

# How to Compute? RCSP

Talking about failures!

Bound test before:

$$\text{discard if: } c(q_P \oplus b_v) \leq C_{od}^{LB} \quad (\text{Low})$$

Bound test after:

for (*profit*, *gas*) in pareto:

$$\text{if } c_{\text{profit}}(q_P \oplus b_v) \leq \textit{profit} \text{ and } c_{\text{gas}}(q_P \oplus b_v) \geq \textit{gas} \\ \text{discard}$$

No solution after 8h...

# How to Compute? Linear Combination

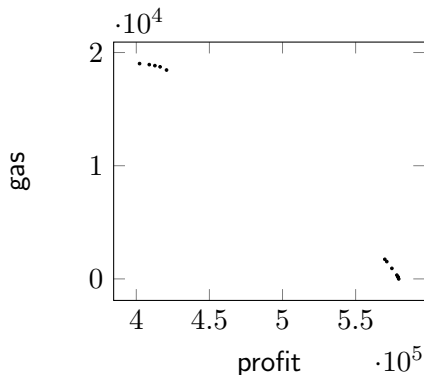
Still talking about failures!

$$c : ((\dots, \mathit{profit}, \dots, \mathit{gas}, \dots)) \mapsto \alpha \cdot \mathit{profit} + (1 - \alpha) \cdot \mathit{gas}$$

# How to Compute? Linear Combination

Still talking about failures!

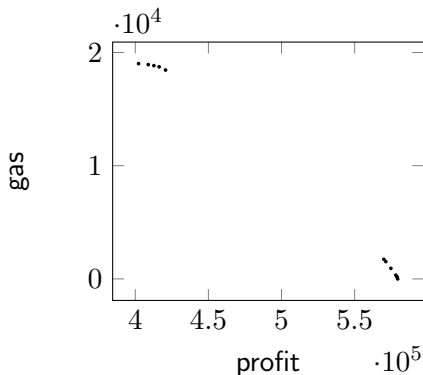
$$c : ((\dots, \textit{profit}, \dots, \textit{gas}, \dots)) \mapsto \alpha \cdot \textit{profit} + (1 - \alpha) \cdot \textit{gas}$$



# How to Compute? Linear Combination

Still talking about failures!

$$c : ((\dots, \textit{profit}, \dots, \textit{gas}, \dots)) \mapsto \alpha \cdot \textit{profit} + (1 - \alpha) \cdot \textit{gas}$$



parteo front is non convex!

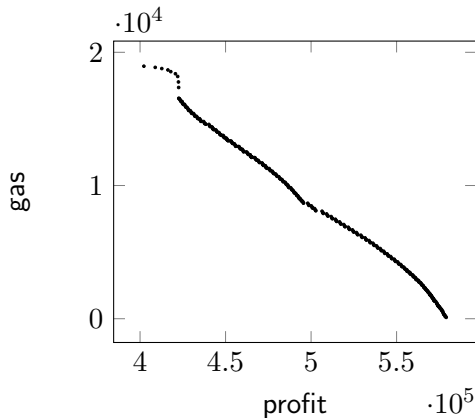
# How to Compute? Upper Bound on Gas Consumption

$gasUpperBound \leftarrow INF$

while  $gasUpperBound \geq 0$

- ▶ solve  $RCSPP(gasUpperBound)$
- ▶ add solution ( $profit, gas$ ) to pareto
- ▶  $gasUpperBound \leftarrow gas - \epsilon$


# How to Compute? Upper Bound on Gas Consumption



Solving time:  $\sim 38s$

Size pareto:  $\sim 600$

# Summary

- ▶ Redesign of the graph
- ▶ Modeling of RCSP in lattice ordered monoid
- ▶ (Conditional) bound computation
- ▶ Solving of non-/gas pricing problems
- ▶ Column generation - Multi Unit Commitment Problem 
- ▶ Pareto Frontier