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

Pascal Benchimol[1] Markus Kruber[2] Axel Parmentier[3]

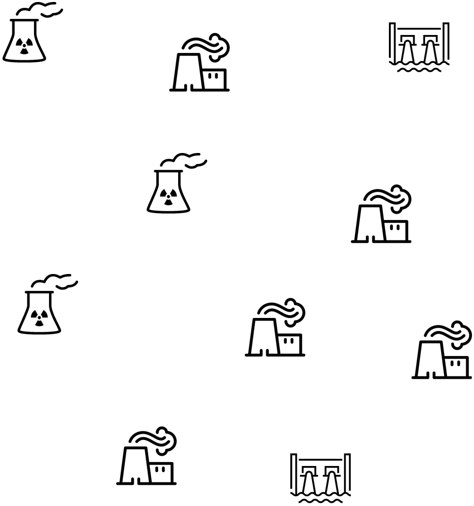
[1]EDF

[2]Chair of Operations Research, RWTH Aachen University

[3]CERMICS, École des Ponts Paristech

March 29, 2017

EDF - Global Problem



Generation units

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

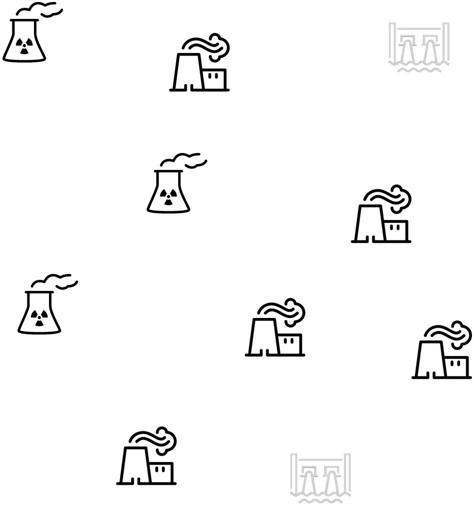
Technically feasible production schedules

Min operating cost

Horizon: $24\text{h} + 24\text{h}$

Time limit: 15m

EDF - Global Problem



Generation units

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

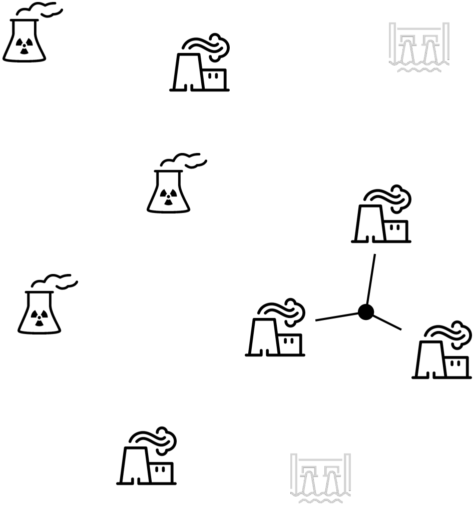
Technically feasible production schedules

Min operating cost

Horizon: $24\text{h} + 24\text{h}$

Time limit: 15m

EDF - Global Problem



Generation units

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

Technically feasible production schedules

Min operating cost

Horizon: $24\text{h} + 24\text{h}$

Time limit: 15m

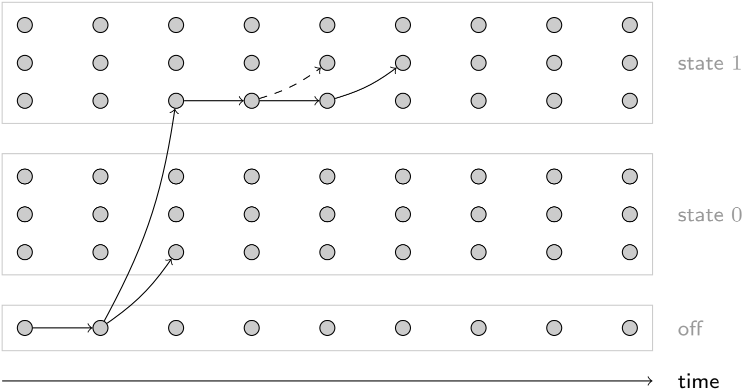
Production Plan - Local Problem

Operations Research | RWTH AACHEN UNIVERSITY

Production Plan - Local Problem

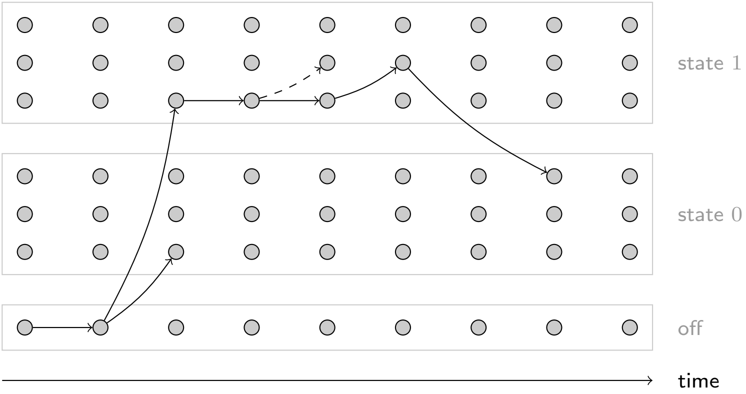
startup · #startups

Production Plan - Local Problem



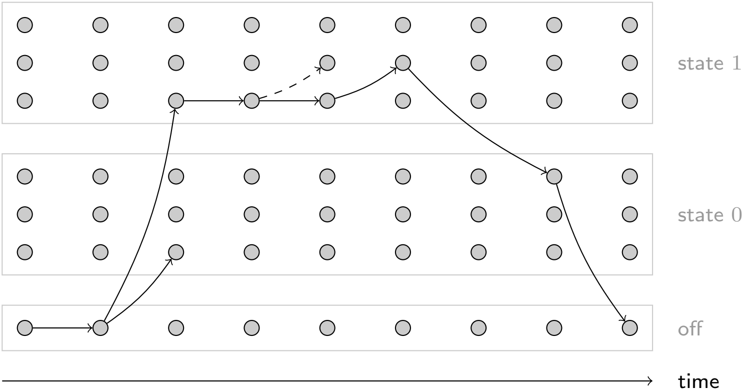
startup · #startups · min_duration_production_level

Production Plan - Local Problem



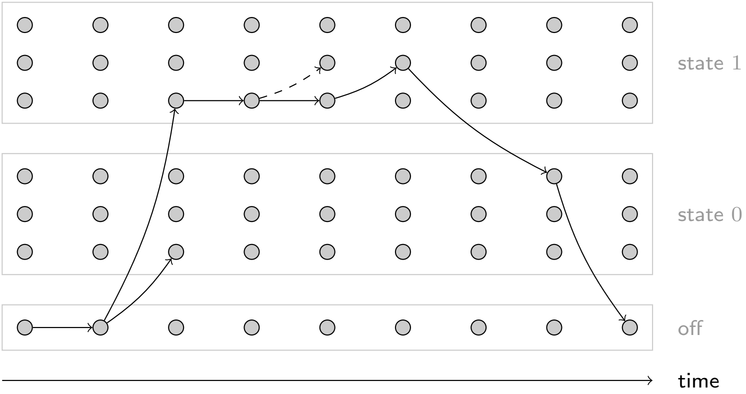
startup · #startups · min_duration_production_level
min_duration_power_state · #modulations

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

The Unit Commitment Problem (1)

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

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

The Unit Commitment Problem (1)

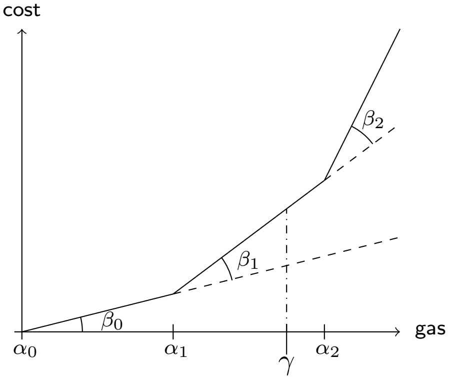
Assume we can generate all technically feasible production plans $\mathcal{P}_i$ for each plant $i \in V$.

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

What makes the problem more difficult?

- Linking constraints: Shared gas stock
- Piecewise linear cost function on gas consumption γ

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.

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

The Unit Commitment Problem (2)

Let j describe a group of plants.

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

The Unit Commitment Problem (2)

Let j describe a group of plants.

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

Why not solving it directly?

Column Generation in General

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

Applied Column Generation

Start with an inital set of feasible paths $\widetilde{\mathcal{P}} \subset P$.

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

Applied Column Generation

Start with an inital set of feasible paths $\widetilde{\mathcal{P}} \subset P$.

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

Dual and Pricing Problem

Dual

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

Dual and Pricing Problem

Dual

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

Pricing

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

Dual and Pricing Problem

Dual

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

Pricing

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

Main Task

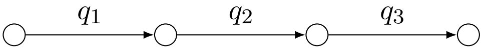
Solve pricing problem fast!

Table of Content

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



$$q_P = q_1 \oplus q_2 \oplus q_3$$

$(\mathcal{R}, \oplus)$ 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 ρ 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 $c : \mathcal{R} \to \mathbb{R}$ $\rho : \mathcal{R} \to \{0, 1\}$

Output:

- An o-d path P such that

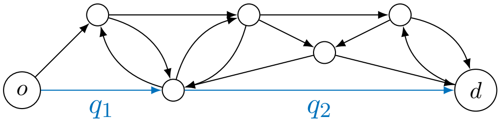
$$\rho\left(\bigoplus_{a \in P} q_a\right) = 0$$

which minimizes

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

Cost and constraint(s):

- non-linear(s)



$q_P = q_1 \oplus q_2$ $\quad\rho(q_P) = 0$

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:

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

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^{start}

Output:

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

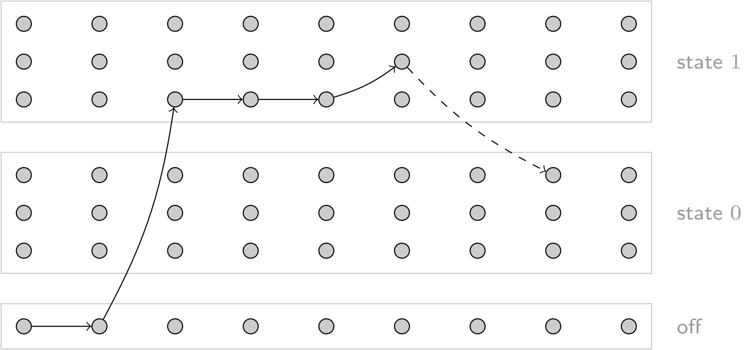
Output:

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

Model:

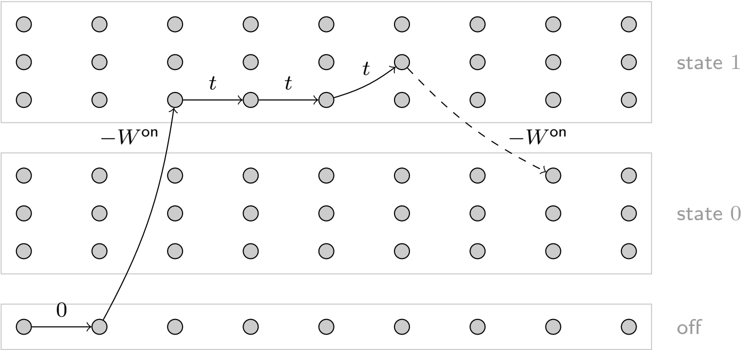
- $\mathcal{R} = \mathbb{R}^2$
- $q_a = (c_a, w_a)$
- $c : ((q^0, q^1)) \mapsto q^0$
- $\rho : ((q^0, q^1) \mapsto \mathbb{1}_{q^1 > W^{\text{start}}}$

Example 3: Minumum Duration in Online State



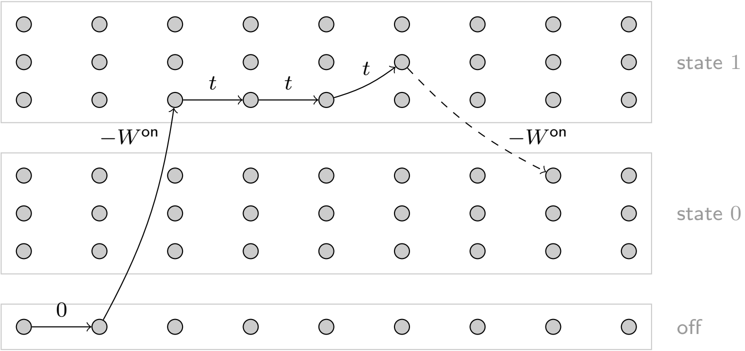
Stay in online state for at least W^{on}.

Example 3: Minumum Duration in Online State



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

Example 3: Minumum Duration in Online State

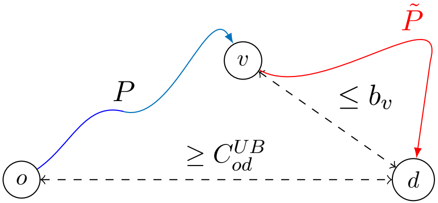


Stay in online state for at least W^{on}.

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

Table of Content

Usual A* algorithm



- $q_P \in \mathbb{R}$
- $C_{od}^{UB} \geq \min_{P \in \mathcal{P}_{o,d}} q_P$
- $b_v \leq q_P, \forall P \in \mathcal{P}_{vd}$

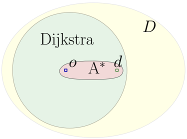
A path $P \in \mathcal{P}_{ov}$ satisfying $q_P + b_v > C_{od}^{UB}$ is not the subpath of an optimal path.

A* algorithm: a Branch & Bound

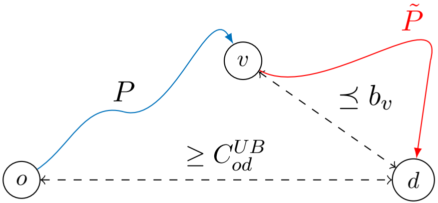
- Generate all the paths satisfying

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

- Update C_{od}^{UB}



Generalized A* algorithm



- $q_P \in \mathcal{R}$
- $C_{od}^{UB} \geq \min_{P|\rho(P)=0} c(q_P)$
- $b_v \preceq q_{\tilde{P}}, \forall \tilde{P} \in \mathcal{P}_{vd}$

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

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

- 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)	$c(q_P \oplus b_v)$

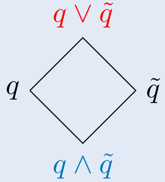
Bound Computation

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 $q \vee \tilde{q}$

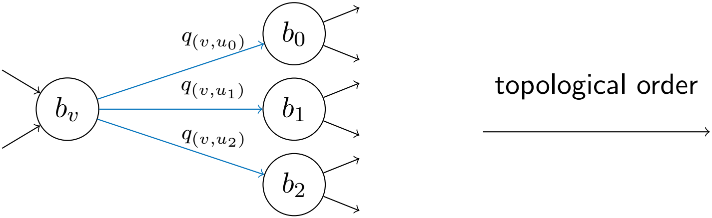
$$\left.\begin{array}{l} b \succeq q \\ b \succeq \tilde{q} \end{array}\right\} \Leftrightarrow b \succeq q \vee \tilde{q}$$

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:

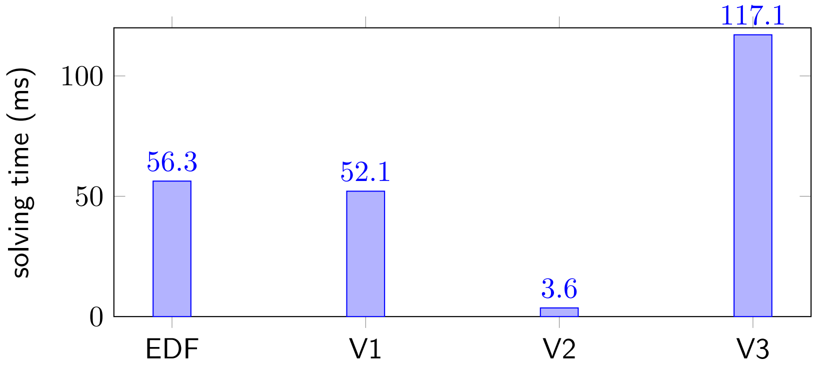
$$b_v = \begin{cases} 0 & \text{,if } v = d \\ \bigwedge \left(b_v, \bigwedge_{u \in N^+(v)} \left(q_{(v,u)} \oplus b_u \right) \right) & \text{,if } v \neq d \end{cases}$$

Table of Content

Some Numbers

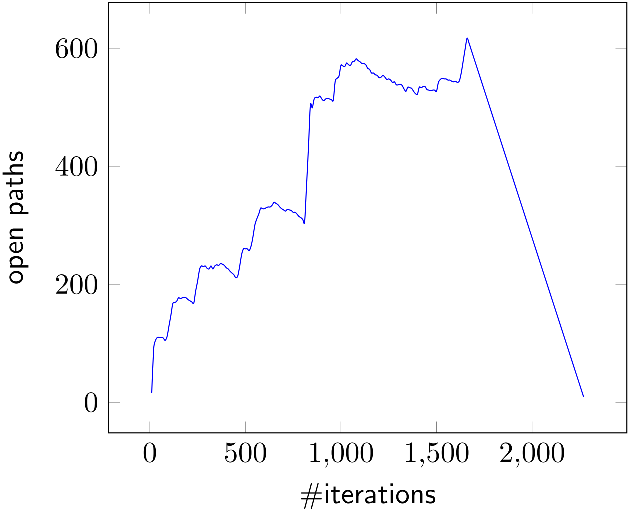
- Plants
  - 8 gas plants $(3, 3, 2)$
  - 97 non-gas plants
- Size of graph (depends on the model)
  - ~ 2.000 nodes
  - ~ 10.000 arcs

Solving Non-Gas Subproblem



	EDF	V1	V2	V3
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



Solving Gas Subproblem

Pricing

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

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

What was the Problem?

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

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

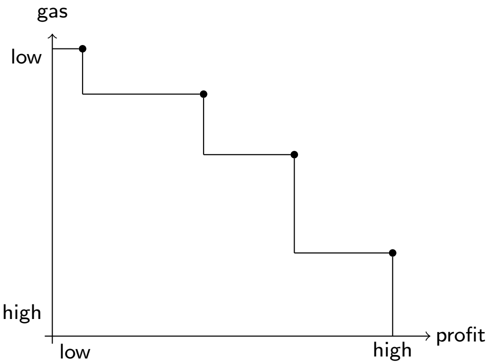
What was the Problem?

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

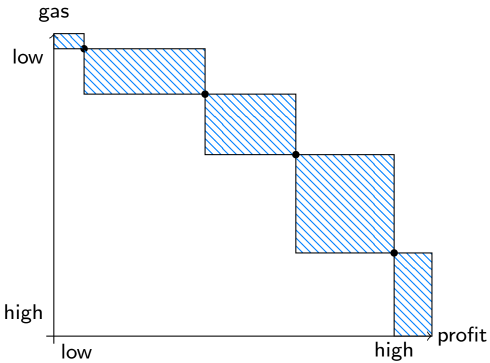
$$\begin{aligned} \text{maximize} \quad & \sum_{p \in \mathcal{P}} c_p x_p - \sum_j \sum_k \beta_{jk} \xi_{jk} \\ \text{subject to} \quad & \qquad \qquad \qquad \dots \end{aligned}$$

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

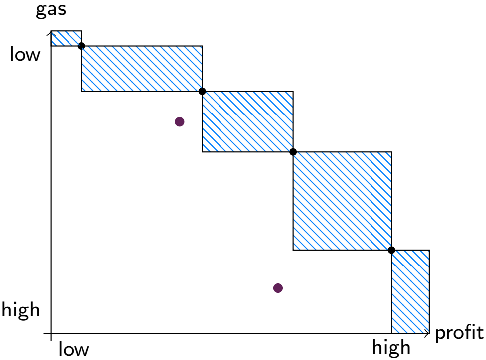
Pareto Frontier



Pareto Frontier

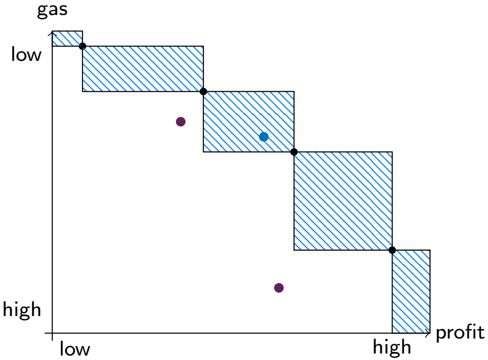


Pareto Frontier



dominated by pareto

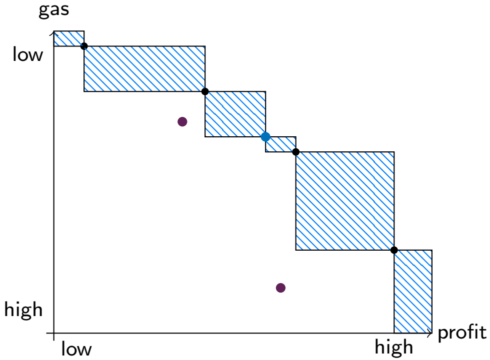
Pareto Frontier



dominated by pareto

update pareto

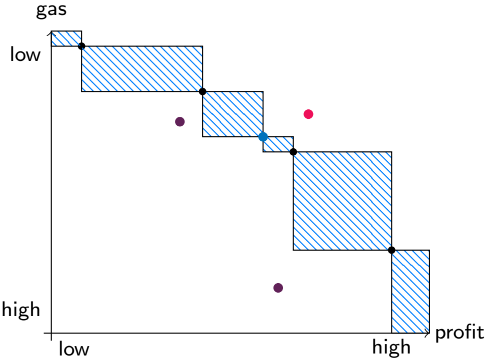
Pareto Frontier



dominated by pareto

update pareto

Pareto Frontier

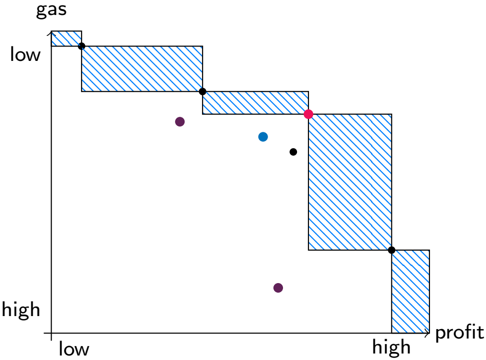


dominated by pareto

update pareto

dominates pareto

Pareto Frontier



dominated by pareto

update pareto

dominates pareto

How to Compute? RCSPP

Talking about failures!

Bound test before:

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

How to Compute? RCSPP

Talking about failures!

Bound test before:

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

Bound test after:

for (*profit*, *gas*) in pareto:
    if $c_{\textit{profit}}(q_P \oplus b_v) \leq$ *profit* and $c_{\textit{gas}}(q_P \oplus b_v) \geq$ *gas*
        discard

How to Compute? RCSPP

Talking about failures!

Bound test before:

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

Bound test after:

for (*profit*, *gas*) in pareto:
    if $c_{\textit{profit}}(q_P \oplus b_v) \leq$ *profit* and $c_{\textit{gas}}(q_P \oplus b_v) \geq$ *gas*
        discard

No solution after 8h...

How to Compute? Linear Combination

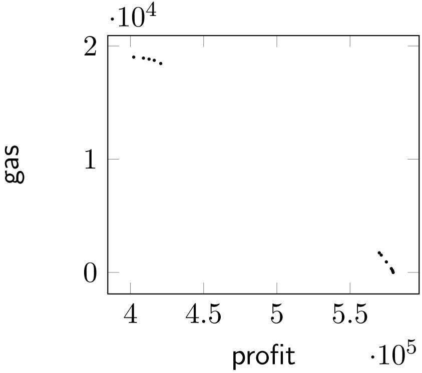
Still talking about failures!

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

How to Compute? Linear Combination

Still talking about failures!

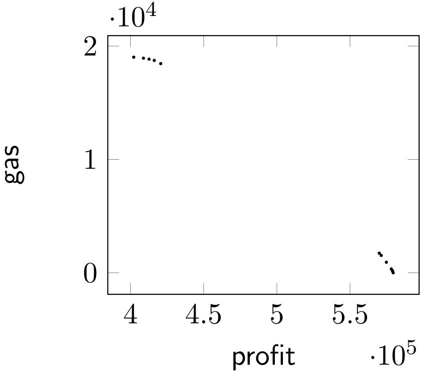
$$c : ((\ldots, \mathit{profit}, \ldots, \mathit{gas}, \ldots)) \mapsto \alpha \cdot \mathit{profit} + (1-\alpha) \cdot \mathit{gas}$$



How to Compute? Linear Combination

Still talking about failures!

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



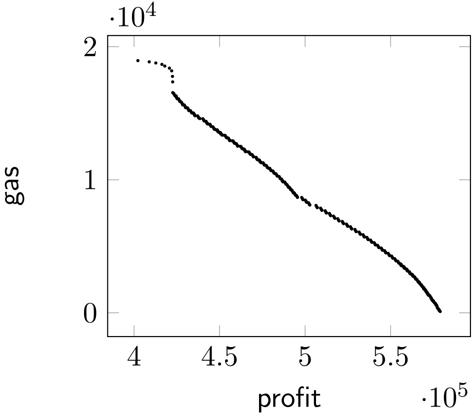
parteo front is non convex!

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 Consumpition



Solving time: ~ 38s
Size pareto: ~ 600

Summary

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