Towards a hybrid approach for intra-daily recourse strategies computation

E. Rachelson

COPI'11, Nov. 24th, 2011

Foreword: a few thoughts

- Often, conference presentations =
  1. hard pb
  2. smart model/algo
  3. outstanding results
  4. discussion at coffee break

- Once things work, long road to the real-life problem
  ...along the way: interesting questions
  ...most conveniently discussed at the coffee break.

Previously on “Intra-daily recourse strategies”

- Daily electricity production planning
- Intra-daily recourse strategies

Making the best out of two worlds

- Extracting domain knowledge to simplify resolution
- A few results
- An abstraction of the problem

Scaling up to the real problems

- The representation issue
- The Learning method
- The conditional dependency problem

Coming up next on “Intra-daily recourse strategies”

Daily production planning

EDF's electricity network:

- 58 nuclear reactors in 20 locations
- 40 thermal plants
- 450 hydraulic p., 640 dams, 50 valleys
- wind, solar, biomass energy $\nearrow$
- Contractual requirements: the reserves

Daily planning problem:

- balance supply/demand
- ensure network stability

Mixed Integer Linear Program formulation

$$\min_{\mathbf{P}_u \in X_u} \sum_{u=1}^{n} C_u(\mathbf{P}_u) + C_0(\mathbf{P}_0)$$

subject to

- supply failure $\mathbf{P}_0 = \mathbf{D}_{ref} - \sum_{u=1}^{n} \mathbf{P}_u$
- initial state of the production park,
- capacity constraints / flow conservation (hydro),
- minimal production/rest periods (on/off),
- gradient constraints,
- max starts and extreme changes,
- min times between schedule changes, etc.

$\Rightarrow \sim 10^6$ variables and constraints.

Lagrangian relaxation and price decomposition for resolution.

Required computation time: $\sim$ 15 minutes.

Intra-daily recourse strategies

Weather, consumers, market prices, etc. $\rightarrow$ hard to predict.
The computed plan can be very suboptimal.

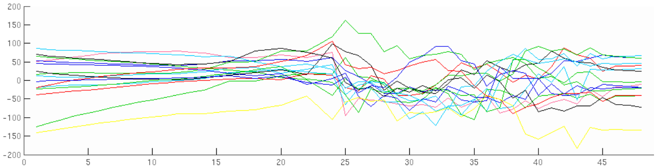


Figure: Variations on $\mathbf{D}_{ref}$

Can EDF change its production plan during the day?

Yes, but $\begin{cases} \text{no more than 30 modified plants} \\ \text{re-declarations every hour} \end{cases}$

(contract with the network manager).

Intra-daily recourse strategies

Weather, consumers, market prices, etc. $\rightarrow$ hard to predict.
The computed plan can be very suboptimal.

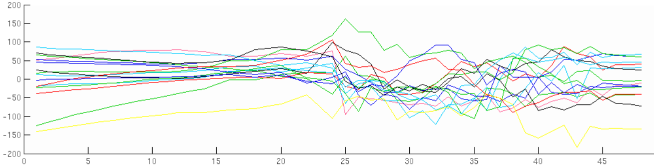


Figure: Variations on $\mathbf{D}_{ref}$

Can EDF change its production plan during the day?

Yes, but $\begin{cases} \text{no more than 30 modified plants} \\ \text{re-declarations every hour} \end{cases}$

(contract with the network manager).

Intra-daily recourse strategies

Original LP +
"max 30 modified plants" $\rightarrow$ linear boolean constraints.

= larger Mixed Integer (Boolean) Program

Required computation time: $\sim$ 40+ minutes.
Resolution time window: $\sim$ <10 minutes.

Current resolution method:
experts make quick adjustments during the day.

Intra-daily recourse strategies

Original LP +
"max 30 modified plants" $\rightarrow$ linear boolean constraints.

= larger Mixed Integer (Boolean) Program

Required computation time: $\sim$ 40+ minutes.
Resolution time window: $\sim$ <10 minutes.

Current resolution method:
experts make quick adjustments during the day.

Intra-daily recourse strategies

Original LP +
"max 30 modified plants" $\rightarrow$ linear boolean constraints.

= larger Mixed Integer (Boolean) Program

Required computation time: $\sim$ 40+ minutes.
Resolution time window: $\sim$ <10 minutes.

Current resolution method:
experts make quick adjustments during the day.

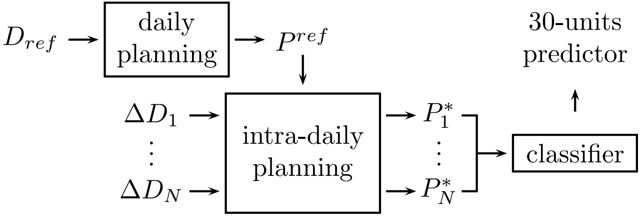
In a nutshell

Intra-daily recourse strategies:

- Very large MILP
- Short resolution time window
- Part of the information is already known the day before

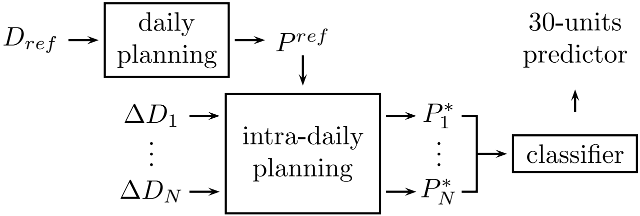
Predicting part of the solution

Exploiting offline resolution time…

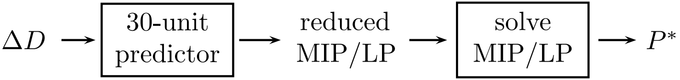


Predicting part of the solution

Exploiting offline resolution time...



...to facilitate online resolution.



A few results

Benchmark problem

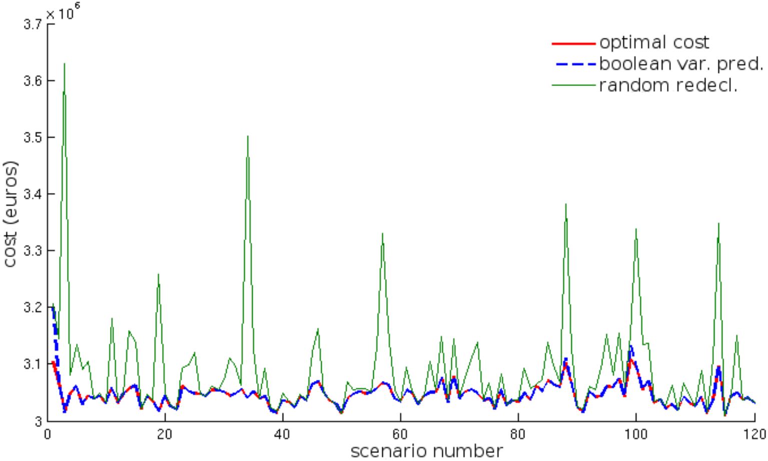
- 27 plants
- $N_{max} = 9$
- 96219 variables
- 61455 constraints
- 120 historical variations ΔD

Supervised Learning method

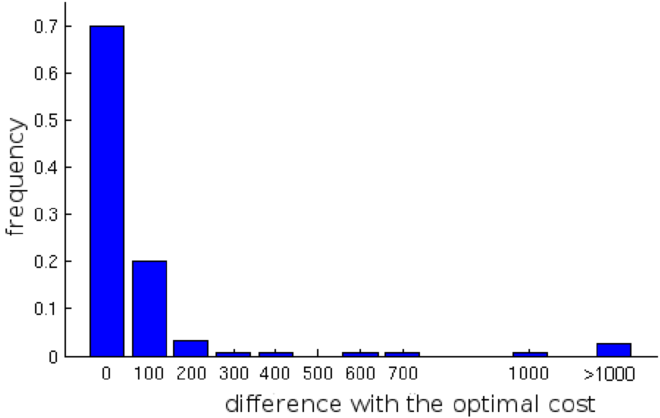
- Boosting + {classif. trees, SVM, rules of thumb}
- prediction: $(\Delta, \text{power plant}) \mapsto b \in \{0; 1\}$

Rachelson E., Ben Abbes A., Diemer S. (2010).
Combining Mixed Integer Programming and Supervised Learning for Fast Re-planning
22nd Int. Conf. on Tools with Artificial Intelligence.

A few results



A few results



A few results

- Multiple local minima, mainly due to equivalent plants.
  - – we did not really find the correct minimum.
  - + we actually found a robust, explainable, quasi-optimal solution.
- The classifier's prediction weakness is not a handicap.

  $\rightarrow$ Global redeclaration structure well captured.

  $\rightarrow$ LP optimization takes care of local optimization.
- Average computation time gain vs. optimality loss.

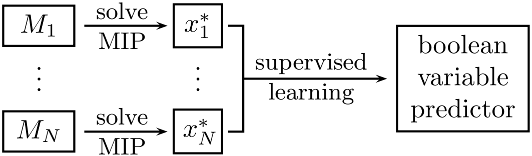
Exact optimization of M	1h
Power plant selection	0.24s
Reduced optimization of M'	128.07s

$\Rightarrow$ 30 times faster with less than 0.1% optimality loss.

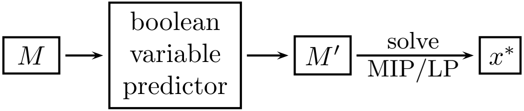
An abstraction of the problem

A Machine Learning point of view on re-planning.

Generalizing experience...



...for online resolution.



An abstraction of the problem

$$\text{prediction: } (\Delta, \text{power plant}) \mapsto b \in \{0;1\}$$
$$\leftrightarrow$$
$$\text{prediction: } (M, x_i) \mapsto b_i \in \{0;1\}$$

Does the previous analysis and modeling hold when confronted to the real-life problem?

Scaling up — the representation issue

Learning a mapping $(M, x_i) \mapsto b_i \in \{0; 1\}$.
but $M \in ?$

E.g.:

- variations in demand $\rightarrow$ change in the r.h.s.
- unit outage/failure $\rightarrow$ more/less constraints

Comparing M and M' cannot be based only on the coeffs.

$\rightarrow$ need to define a *metric over problems*.

E.g.:

- coeffs variation
- geometry on the admissible set

Scaling up — the algorithmic learning issue

Issue: few samples, large dimension.

- Many good mappings are possible.
  $\rightarrow$ Real-life problems need dimension reduction.
- Full problem space coverage is not practical.
  $\rightarrow$ Real-life problems need margins / confidence bounds.

Scaling up — the conditional dependency issue

$$(M, x_i) \mapsto b_i \in \{0;1\} \Leftrightarrow (M) \mapsto b \in \{0;1\}^n?$$

Scaling up — the conditional dependency issue

$$(M, x_i) \mapsto b_i \in \{0;1\} \Leftrightarrow (M) \mapsto b \in \{0;1\}^n?$$

practical version:

30×"predict a unit" ⇔ "predict 30 units"?

Scaling up — the conditional dependency issue

$$(M, x_i) \mapsto b_i \in \{0; 1\} \Leftrightarrow (M) \mapsto b \in \{0; 1\}^n?$$

practical version:
30×“predict a unit” ⇔ “predict 30 units”?

abstract (ML) version:

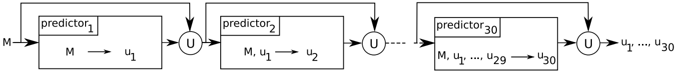
$$\max_{b_i, i \in [1; n_b]} Pr\left(x_i^* = b_i, i \in [1; n_b]\right) \Leftrightarrow \forall i \in [1; n_b], \max_{b_i, i \in [1; n_b]} Pr\left(x_i^* = b_i\right)?$$

Scaling up — the conditional dependency issue

→ Taking the 30 best scores out of the predictor is not necesarily taking the best redeclaration.

→ set prediction problem

→ sequence prediction problem: 30-stage control problem



→ closed-loop formulation: $s = (M, h)$

Any hope left for MIP/ML?

- MILP resolution from a ML perspective
- Predicting discrete values in MILP = promising idea
  - Showed good results
  - Makes the best out of Optimisation (global optimum on a restricted pb) and Inference (capitalizing on previous experience).
- However, this prediction is not a trivial ML problem
- A first method (and some related other ones) proved successful
- But this first approach's grounding is somewhat imperfect
  - representation issue
  - algorithmic issue
  - conditional dependency issue

This presentation in a nutshell:

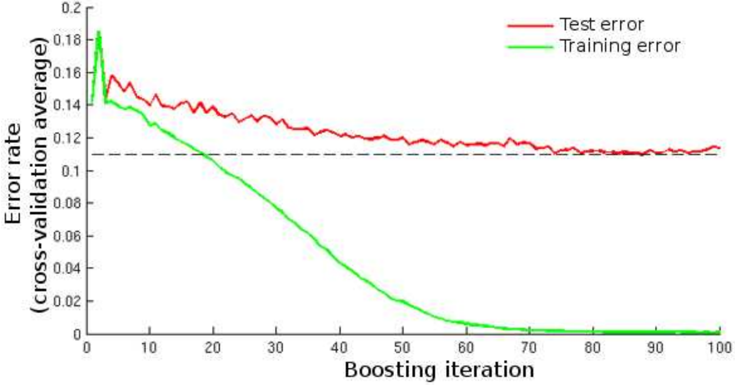
- Experience feedback seemed relevant, for the community, in order to work out a better problem statement.

Many thanks to:

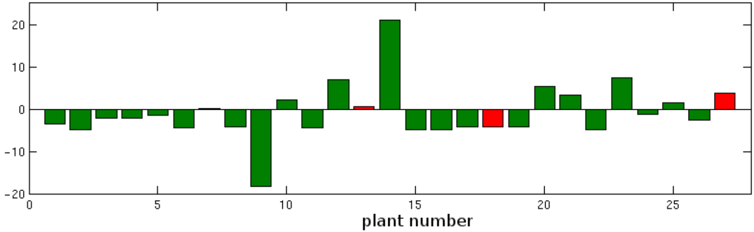
- Ala Ben-Abbes
- Grace Doukopoulos
- Arnaud Lenoir
- Jérome Quenu

Any questions?

Training/testing error



Margins



Trajectories