Machine-learning aided operation and planning of power systems

ECE Seminar Series, New York University

Salvador Pineda (spineda.uma.es)

OASYS group, University of Málaga (Spain)

April 30, 2020

About Málaga

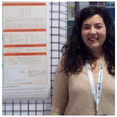
- Over 300 sundays per year (known as Costa del Sol)
- University of Málaga was stablished in 1972 and currently has 40000 students and 2500 faculty members
- Málaga is becoming the Silicon Valley of the south of Spain
- Andalusia Technology Park includes over 600 companies (Oracle, Ericsson, IBM, TDK, Huawei, Microsoft, Cisco), 20.000 employees and a turnover of 2.000 M€ in 2018

About OASYS

- **O**ptimization and **A**nalytics for **S**ustainable energ**Y** **S**ystems
- Established in 2018
- 6 members (Open positions for 1 PhD student and 1 Postdoc)

Juanmi (Professor)

Salva (Professor)



Asun (Postdoc)

Adrián (PhD student)

Miguel (PhD student)

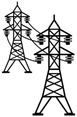
Alvaro (PhD student)

- More info: oasys.uma.es

Power Systems

Producers

Lines



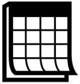
Consumers

Renewables

Storage

Decisions

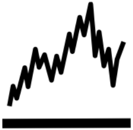
Long-term



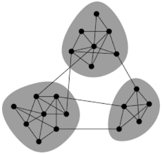
Short-term

Dimension

Temporal

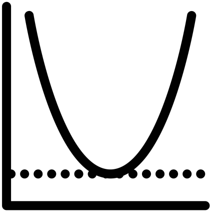


Spatial

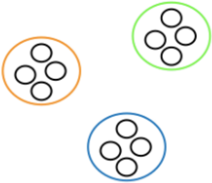


Math Tools

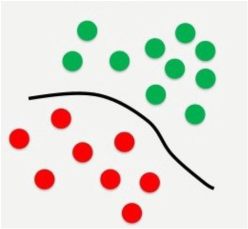
Optimization



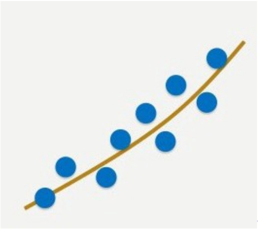
Unsupervised learning (clustering)



Supervised learning (classification)

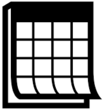


Supervised learning (regression)

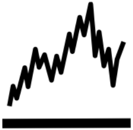


APPLICATION 1

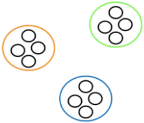
Long-term planning



Temporal dimension



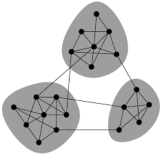
Clustering



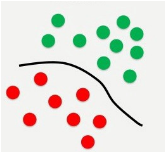
APPLICATION 2

Short-term operation

Spatial dimension



Classification



APPLICATION 1

Chronological Time-Period Clustering for
Optimal Capacity Expansion Planning With Storage

S. Pineda J. M. Morales

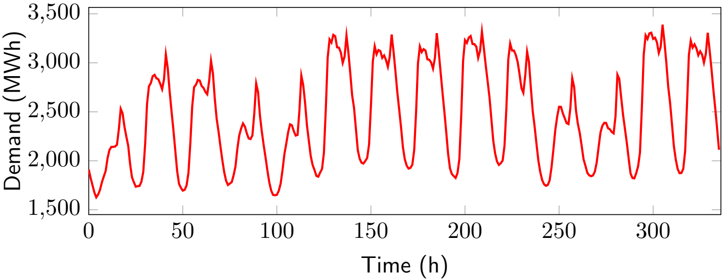
IEEE Trans. in Power Systems, vol. 33, no. 6, pp. 7162-7170, 2018

What problem are we solving?

Expansion Planning Problem	
Horizon	1 year
Decisions	Generation investments Line investments
Objective	Min prod. + inv. cost
Constraints	Generation = Demand Unit technical limits Line technical limits
Comput. burden	Very high

How are planning problems usually solved?

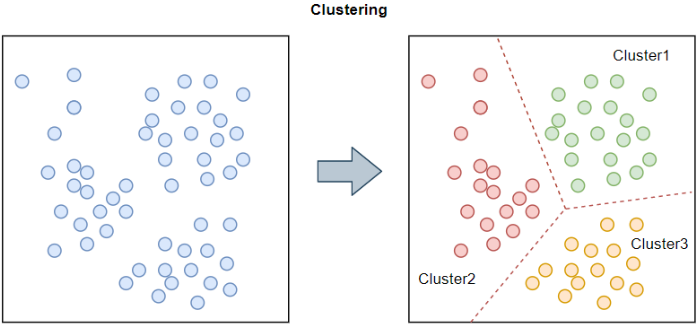
- Taking advantage of the fact that the electrical demand shows strong daily, weekly and annual patterns.



- Using statistical learning techniques such as clustering to group time periods and reduce the computational cost of the planning problem.

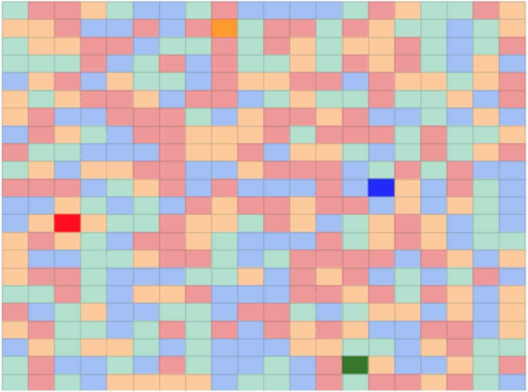
What is clustering?

- The clustering consists in grouping a set of objects so that members of the same group (called cluster) are more similar.



How is clustering used in planning problems?

- The most common approach is to cluster days and solve the optimization problem considering only the **representative days**.



- Day-to-day chronology information is lost

How do I know if two days are similar or not?

- Each day is characterized by normalized demand, wind and solar

$$x_1 = [\underbrace{0.4, 0.6, \ldots, 0.5}_{\text{24 demand values}}, \underbrace{0.1, 0.2, \ldots, 0.4}_{\text{24 wind values}}, \underbrace{0.1, 0.2, \ldots, 0.1}_{\text{24 solar values}}]$$

$$x_2 = [\underbrace{0.6, 0.1, \ldots, 0.3}_{\text{24 demand values}}, \underbrace{0.4, 0.5, \ldots, 0.2}_{\text{24 wind values}}, \underbrace{0.3, 0.4, \ldots, 0.2}_{\text{24 solar values}}]$$

- The similarity between two days is computed using a norm

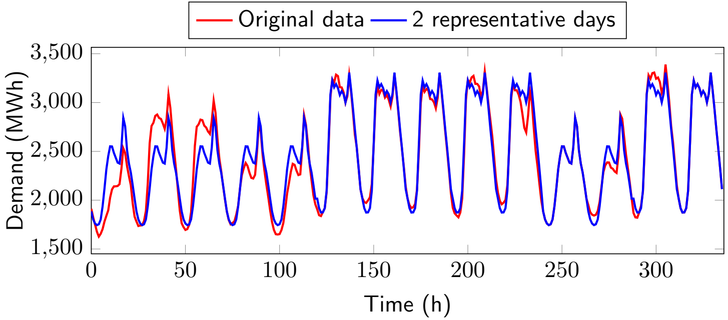
$$d(x_1, x_2) = ||x_1 - x_2||_2$$

How does the clustering algorithm work?

1) Set the initial number of clusters n to the number of days N.
2) Determine cluster centroid as $\overline{\mathbf{x}}_I = \frac{1}{|I|}\sum_{i\in I}\mathbf{x}_i$
3) Compute dissimilarity as $D(I,J) = \frac{2|I||J|}{|I|+|J|}||\overline{\mathbf{x}}_I - \overline{\mathbf{x}}_J||^2$
4) Merge the two closest $(I', J') \in$ argmin $D(I,J)$ s.t. $I \neq J$.
5) Update $n \leftarrow n-1$. If $n = N'$ go to 6). Otherwise, go to 2).
6) Representative days are cluster medoids.
7) Weight of representative days proportional to cluster elements.

How do the representative days approach work?

- As the figure below shows, using representative days works quite well



- Instead of 14 days (336 h), we use 2 representative days (48 h).
- Computational burden is reduced.

And what about current power systems?

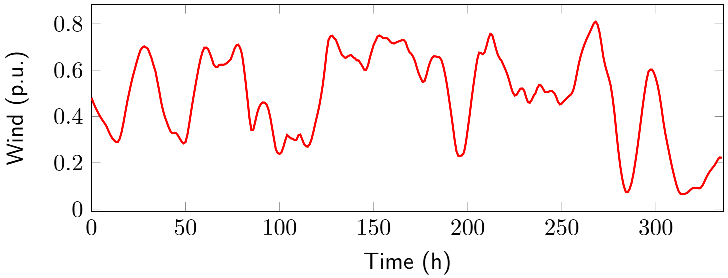
- Current power systems have new actors

Generators Lines Demand Renewables Storage

- Renewable generation is free and reduce CO2 emissions
- Renewable generation may happen at the wrong time
- Storage energy systems are the perfect partner of renewables:
  - If wind blows and demand is low, the battery stores energy
  - If wind does not blow and demand is high, the battery injects energy

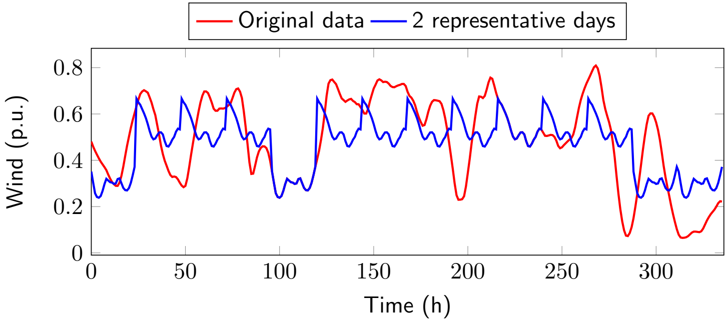
Can we still use representative days?

- Some renewables do not present a strong daily pattern



Can we still use representative days?

- Some renewables do not present a strong daily pattern



- Energy stored can be used several days later (**seasonal storage**)
- Crucial to overcome the “dark calm” periods in central Europe
- Seasonal storage cannot be modeled using representative days

What do we propose?

- Instead of using representative days, we propose a new clustering methodology to group **consecutive hours** and maintain chronology
- By doing so we can capture the **longer dynamics** of power generation from renewable sources such as wind
- In addition, we can model the operation of the batteries more accurately since we maintain the **chronology** of the data

How do I know if two consecutive hours are similar or not?

- Each hour is characterized by normalized demand, wind and solar

$$x_1 = [\underbrace{0.4}_{\text{demand}}, \underbrace{0.1}_{\text{wind}}, \underbrace{0.2}_{\text{solar}}]$$

$$x_2 = [\underbrace{0.6}_{\text{demand}}, \underbrace{0.4}_{\text{wind}}, \underbrace{0.3}_{\text{solar}}]$$

- The similarity between two consecutive hours is computed as

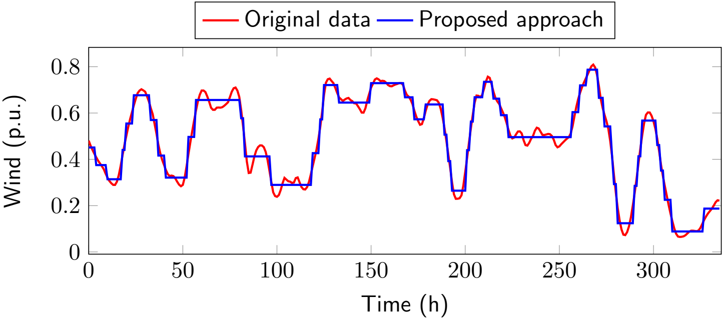
$$d(x_1, x_2) = ||x_1 - x_2||_2$$

How does the proposed clustering algorithm work?

1) Set the initial number of clusters n to the total number of hours N.
2) Determine the centroid of each cluster as $\overline{\mathbf{x}}_I = \frac{1}{|I|}\sum_{i \in I} \mathbf{x}_i$
3) Compute dissimilarity as $D(I,J) = \frac{2|I||J|}{|I|+|J|}||\overline{\mathbf{x}}_I - \overline{\mathbf{x}}_J||^2$
4) Merge two closest **adjacent** clusters.
5) Update $n \leftarrow n-1$. If $n = N'$ go to 6). Otherwise, go to 2).
6) Determine the representative periods as the clusters' centroids $\overline{\mathbf{x}}_I$.
7) Time-period duration proportional to number of elements.

Does the proposed clustering work?

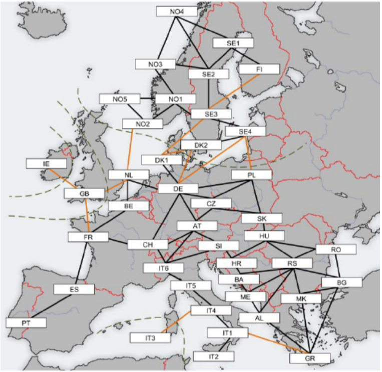
- Wind is approximated more accurately



- Seasonal storage can be properly modeled

Have you tried it on a realistic case study?

- Electric power system (28 countries) for 2030 (single target year)
- Investments in conventional and renewable generation, transmission lines and two storage technologies (intraday and interday)
- Greenfield approach (no initial capacities)
- Given renewable penetration target



Have you tried it on a realistic case study?

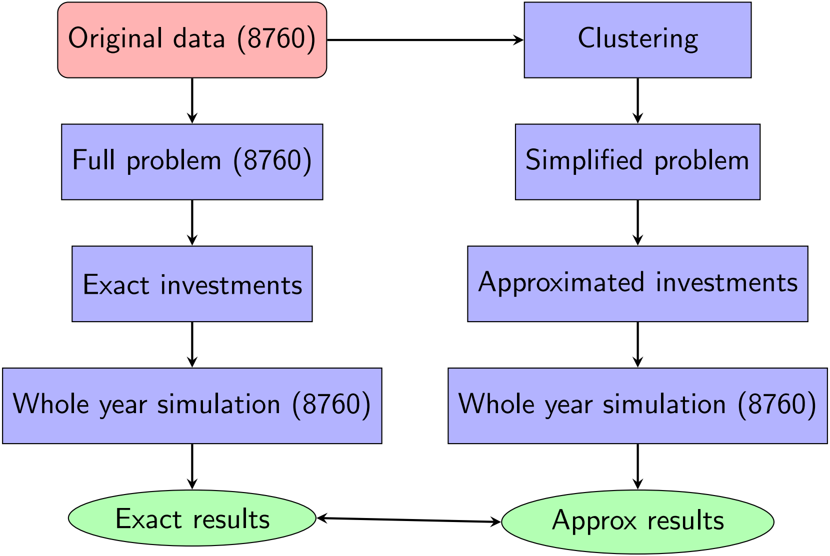
Table: Generation technology data

Technology	i_g^G (€/MW)	y_g^G (years)	c_g (€/MWh)	r_g^+/r_g^- (p.u.)
Base	$4 \cdot 10^6$	60	10	0.1
Peak	$1.5 \cdot 10^6$	40	40	1.0
Wind	$1.5 \cdot 10^6$	25	-	-
Solar	$1 \cdot 10^6$	25	-	-

Table: Storage technology data

Storage	η_s (h)	ξ_s (p.u.)	i_s^S (€/MW)	y_s (years)
intraday	6	0.8	$1.5 \cdot 10^6$	80
interday	48	0.7	$2 \cdot 10^6$	60

What about the results?



What about the results?

- We compare four different models

Approach	Acronym	Number periods
Full year	F	8760
28 representative days	D-28	$28 \times 24 = 672$
4 representative weeks	W-4	$4 \times 168 = 672$
672 chronological periods	C-672	672

What about the results?

Results for a 50% renewable target

	F	C-672	D-28	W-4
Base (GW)	208	206	235	207
Peak (GW)	20	16	41	0
Wind (GW)	772	747	692	790
Solar (GW)	217	255	276	155
Hydro (GW)	160	160	160	160
Intraday (GW)	48	31	135	100
Interday (GW)	144	151	0	69
Network (GW)	23	18	34	19
Cost (10^9€)	95.13	99.31	102.17	133.72
Share (%)	50	49.5	46.1	47.1
Shed (%)	0	0.1	0.2	1.1

What about the results?

Results for a 50% renewable target

	F	C-672	D-28	W-4
Base (GW)	208	206	235	207
Peak (GW)	20	16	41	0
Wind (GW)	772	747	692	790
Solar (GW)	217	255	276	155
Hydro (GW)	160	160	160	160
Intraday (GW)	48	31	135	100
Interday (GW)	144	151	0	69
Network (GW)	23	18	34	19
Cost (10^9€)	95.13	99.31	102.17	133.72
Share (%)	50	49.5	46.1	47.1
Shed (%)	0	0.1	0.2	1.1

Full model meets renewable target at lowest cost and no load shedding

What about the results?

Results for a 50% renewable target

	F	C-672	D-28	W-4
Base (GW)	208	206	235	207
Peak (GW)	20	16	41	0
Wind (GW)	772	747	692	790
Solar (GW)	217	255	276	155
Hydro (GW)	160	160	160	160
Intraday (GW)	48	31	135	100
Interday (GW)	144	151	0	69
Network (GW)	23	18	34	19
Cost (10^9€)	95.13	99.31	102.17	133.72
Share (%)	50	49.5	46.1	47.1
Shed (%)	0	0.1	0.2	1.1

Full model invests both in intraday and interday storage

What about the results?

Results for a 50% renewable target

	F	C-672	D-28	W-4
Base (GW)	208	206	235	207
Peak (GW)	20	16	41	0
Wind (GW)	772	747	692	790
Solar (GW)	217	255	276	155
Hydro (GW)	160	160	160	160
Intraday (GW)	48	31	135	100
Interday (GW)	144	151	0	69
Network (GW)	23	18	34	19
Cost (10^9€)	95.13	99.31	102.17	133.72
Share (%)	50	49.5	46.1	47.1
Shed (%)	0	0.1	0.2	1.1

W-4 underinvests in peak and involves high load shed and cost

What about the results?

Results for a 50% renewable target

	F	C-672	D-28	W-4
Base (GW)	208	206	235	207
Peak (GW)	20	16	41	0
Wind (GW)	772	747	692	790
Solar (GW)	217	255	276	155
Hydro (GW)	160	160	160	160
Intraday (GW)	48	31	135	100
Interday (GW)	144	151	0	69
Network (GW)	23	18	34	19
Cost (10^9€)	95.13	99.31	102.17	133.72
Share (%)	50	49.5	46.1	47.1
Shed (%)	0	0.1	0.2	1.1

D-28 underinvests in wind+interday and overinvests in solar+intraday

What about the results?

Results for a 50% renewable target

	F	C-672	D-28	W-4
Base (GW)	208	206	235	207
Peak (GW)	20	16	41	0
Wind (GW)	772	747	692	790
Solar (GW)	217	255	276	155
Hydro (GW)	160	160	160	160
Intraday (GW)	48	31	135	100
Interday (GW)	144	151	0	69
Network (GW)	23	18	34	19
Cost (10^9€)	95.13	99.31	102.17	133.72
Share (%)	50	49.5	46.1	47.1
Shed (%)	0	0.1	0.2	1.1

Representative weeks and days fail to meet renewable target

What about the results?

Results for a 50% renewable target

	F	C-672	D-28	W-4
Base (GW)	208	206	235	207
Peak (GW)	20	16	41	0
Wind (GW)	772	747	692	790
Solar (GW)	217	255	276	155
Hydro (GW)	160	160	160	160
Intraday (GW)	48	31	135	100
Interday (GW)	144	151	0	69
Network (GW)	23	18	34	19
Cost (10^9€)	95.13	99.31	102.17	133.72
Share (%)	50	49.5	46.1	47.1
Shed (%)	0	0.1	0.2	1.1

C-672 approximates well investments in renewables and storage

What about the results?

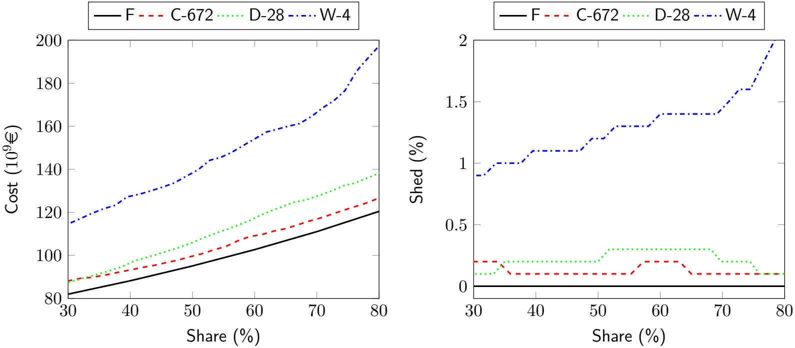
Results for a 50% renewable target

	F	C-672	D-28	W-4
Base (GW)	208	206	235	207
Peak (GW)	20	16	41	0
Wind (GW)	772	747	692	790
Solar (GW)	217	255	276	155
Hydro (GW)	160	160	160	160
Intraday (GW)	48	31	135	100
Interday (GW)	144	151	0	69
Network (GW)	23	18	34	19
Cost (10^9€)	95.13	99.31	102.17	133.72
Share (%)	50	49.5	46.1	47.1
Shed (%)	0	0.1	0.2	1.1

C-672 provides a closer cost, renewable target and load shedding

What about the results?

- Cost and load shed for different renewable targets



- Benefits of proposed approach rise for high renewable shares

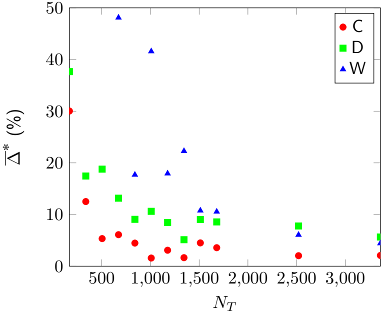
- Average cost increase over all renewable penetration levels.

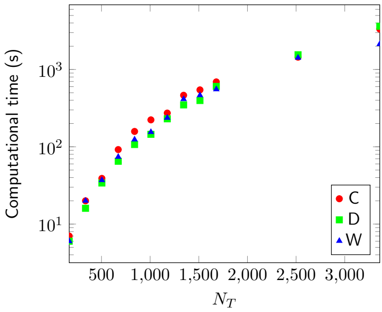
Approach	Number periods	Av. cost increase	Time
F	8760	0 %	$\sim$ 10 h
D-28	$28 \times 24 = 672$	13.1 %	$\sim$ 100 s
W-4	$4 \times 168 = 672$	48.1 %	$\sim$ 100 s
C-672	672	6.1 %	$\sim$ 100 s

- The proposed approach is the closest to the full model

What about the results?

- Results for different number of time periods





- The computational burden of proposed approach equals existing ones

What about the results?

- Finally, we evaluate the proposed method in different scenarios

Scenario	C-672	D-28	W-4
Base	**6.1%**	13.1%	48.1%
Only_wind	**8.6%**	18.3%	31.5%
Only_solar	9.0%	**7.2%**	32.8%

- If only wind, the proposed approach outperforms others methods

- If only solar, representative days works good enough

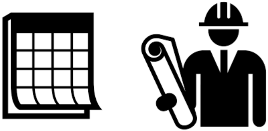
Conclusions

- Existing planning problems do not properly capture mid-term dynamics of renewable generation and fail to model storage operation.
- We propose a time-period clustering that retains the chronology of time-dependent parameters throughout the planning horizon.
- The proposed method determines expansion plans that account for the economic value of pairing seasonal storage and renewables.
- Numerical results show that the proposed method yields significant cost reductions without increasing computational burden.

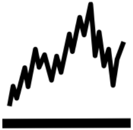
Two applications

APPLICATION 1

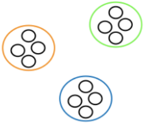
Long-term planning



Temporal dimension



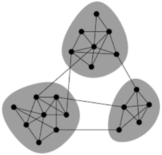
Clustering



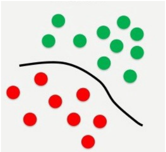
APPLICATION 2

Short-term operation

Spatial dimension



Classification



APPLICATION 2

Data-Driven Screening of Network Constraints for Unit Commitment

S. Pineda J. M. Morales A. Jiménez-Cordero

IEEE Transactions on Power Systems, early access

What problem are we solving?

Unit Commitment Problem	
Horizon	1 day
Decisions	Generation commitment Generation dispach Power flows
Objective	Min production cost
Constraints	Generation = Demand Unit technical limits Line technical limits
Comput. burden	High

How is that problem formulated?

$$\min_{\mathbf{x}\in\mathbb{R}^n,\, \mathbf{y}\in\{0,1\}^m} \quad f(\mathbf{x},\mathbf{y}) \tag{1a}$$

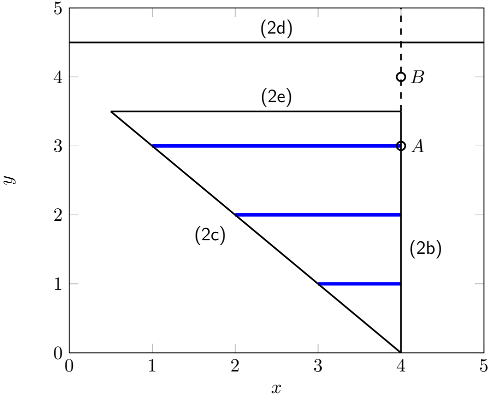
$$g_i(\mathbf{x},\mathbf{y}) \leqslant 0, \quad \forall i \tag{1b}$$

$$h_j(\mathbf{x}) \leqslant 0, \quad \forall j \tag{1c}$$

- Continuous variables $\mathbf{x}$: power dispatches, power flows through lines
- Binary variables $\mathbf{y}$: on/off status of the generating units
- Objective function (1a) minimizes the total generation costs
- Equation (1b): technical constraints of generating units
- Equation (1c): technical constraints of network
- Even if all functions are linear, problem (1a)-(1c) is **NP-hard**

Can we remove constraints to reduce time?

$$
\begin{aligned}
\max_{x\in\mathbb{R},y\in\mathbb{Z}} \quad & x + y && \text{(2a)}\\
\text{s.t.} \quad & x \leqslant 4 && \text{(2b)}\\
& x + y \geqslant 4 && \text{(2c)}\\
& y \leqslant 4.5 && \text{(2d)}\\
& y \leqslant 3.5 && \text{(2e)}
\end{aligned}
$$



- Constraint (2b) is an *active constraint*
- Constraint (2c) is an *inactive constraint*
- Constraint (2d) is a *redundant constraint*
- Constraint (2e) is defined as *quasi-active constraint*

How is the Unit Commitment problem formulated?

- Single-period
- DC power flow
- Thermal units
- Renewable units
- Known demand
- No failures

$$\min_{p_g, u_g, q_n, \epsilon_n} \quad \sum_g c_g p_g + L \sum_n |\epsilon_n| \tag{3a}$$

$$\text{s.t.} \quad q_n + \epsilon_n = \sum_{g: b_g = n} p_g - d_n, \forall n \tag{3b}$$

$$\sum_n q_n = 0 \tag{3c}$$

$$u_g \underline{p}_g \leqslant p_g \leqslant u_g \rho_g \overline{p}_g, \forall g \tag{3d}$$

$$-\overline{f}_l \leqslant \sum_n a_{ln} q_n \leqslant \overline{f}_l, \forall l \tag{3e}$$

$$u_g \in \{0, 1\}, \forall g \tag{3f}$$

We compare 8 different methods to remove constraints (3e)

Which methods can be used to remove constraints?

Benchmark method (BN)

- No network constraints are removed
- Computational burden extremely high
- Useful for comparison purposes

Which methods can be used to remove constraints?

Single-bus method (SB)

- All network constraints are removed
- Fastest method
- Close-to-optimal solutions in low-congested systems
- Highly suboptimal solutions in general

Which methods can be used to remove constraints?

Perfect information method (PI)

- Removes all constraints not binding at the optimum
- It cannot be implemented in practice
- It removes quasi-active constraints

Which methods can be used to remove constraints?

Naive method (NV)

- It removes line constraints that have not been congested in the past
- It requires access to historical data
- Low number of removed constraints

Which methods can be used to remove constraints?

Constraint generation (CG)

- It starts by solving the UC without any network constraint
- Line constraints exceeding their capacity are iteratively added
- It provides the same solution as BN
- High computational burden since the UC is solved at each iteration

Roald method (RO)[1]

- Two optimization problems for each line are solved

$$\min_{p_g, q_n, d_n} / \max_{p_g, q_n, d_n} \quad \sum_n a_{l'n} q_n$$
$$\text{s.t.} \quad (3b), (3c), (3d), (3e)$$
$$\underline{d}_n \leqslant d_n \leqslant \overline{d}_n, \forall n$$

- If the objective functions reach the line limit, then its capacity constraints are kept. Otherwise, such constraints are removed.

- It only removes redundant constraints

[1] Roald and Molzahn 2019.

Which methods can be used to remove constraints?

Data-driven method (DD)

- Line congestion is inferred via statistical learning
- No need for solving additional optimization problems
- It removes not only redundant but also inactive constraints
- K-nearest neighbors is used for its simplicity and interpretability

Which methods can be used to remove constraints?

Data-driven method (DD)

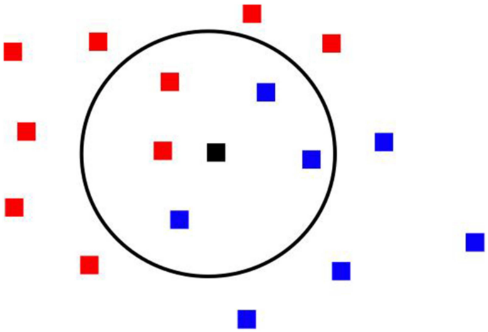
- Data on net demand and congestion status is denoted as $(\tilde{\mathbf{d}}_t, s_{lt}) \forall t$
- For a new time period $\hat{t}$, find the subset of closest K neighbors ($\mathcal{N}_K$) using the distance function below

$$\text{dist}(\tilde{\mathbf{d}}_t, \tilde{\mathbf{d}}_{\hat{t}}) = \|a_l^T(\tilde{\mathbf{d}}_t - \tilde{\mathbf{d}}_{\hat{t}})\|_2 \qquad (5)$$

- Two individuals are close if the net demand of those buses that have a higher impact on the power flow through line l are similar enough
- If $s_{lt} = 0 \; \forall t \in \mathcal{N}_K$, line l is assumed uncongested for $\hat{t}$ and its capacity constraints are removed. Otherwise, such constraints are kept.

Which methods can be used to remove constraints?

Data-driven method (DD)



■ → Line is congested

■ → Line is not congested

■ → Line is assumed to be congested and its constraint is kept

Which methods can be used to remove constraints?

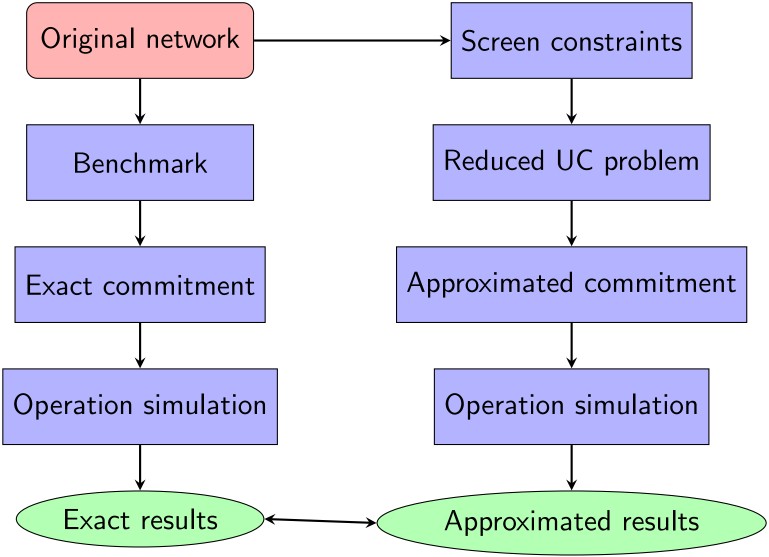
Data-driven + constraint generation (DD+CG)

- Use data to rapidly remove a large number of constraints
- Then iteratively add violated line constraints
- It also provides the same solution as BN
- It requires way less iterations than CG

Have you tried it on a realistic case study?

- IEEE RTS-96 test system modified to accommodate 19 wind farms
- 73 nodes and 120 transmission lines
- 300 training days and 60 test days
- We consider a low- and a high-congested case

What about the results?



What about the results?

Low-congested case

Method	Removed(%)	Δcost(%)	Infes(%)	Time(%)
Benchmark	0	0.00	0.000	100
Single	100	-0.04	0.044	19.8
Perfect	99.9	0.01	0.001	15.9
Naive	99.2	0.00	0.000	16.5
ConGen	99.9	0.00	0.000	27.0
Roald	85.8	0.00	0.000	24.8
Data-driven	99.8	0.00	0.000	17.5
DD+CG	99.8	0.00	0.000	22.6

What about the results?

Low-congested case

Method	Removed(%)	Δcost(%)	Infes(%)	Time(%)
Benchmark	0	0.00	0.000	100
Single	100	-0.04	0.044	19.8
Perfect	99.9	0.01	0.001	15.9
Naive	99.2	0.00	0.000	16.5
ConGen	99.9	0.00	0.000	27.0
Roald	85.8	0.00	0.000	24.8
Data-driven	99.8	0.00	0.000	17.5
DD+CG	99.8	0.00	0.000	22.6

Single-bus method provides acceptable results

What about the results?

Low-congested case

Method	Removed(%)	Δcost(%)	Infes(%)	Time(%)
Benchmark	0	0.00	0.000	100
Single	100	-0.04	0.044	19.8
Perfect	99.9	0.01	0.001	15.9
Naive	99.2	0.00	0.000	16.5
ConGen	99.9	0.00	0.000	27.0
Roald	85.8	0.00	0.000	24.8
Data-driven	99.8	0.00	0.000	17.5
DD+CG	99.8	0.00	0.000	22.6

Five methods provide the same solution than the benchmark

What about the results?

Low-congested case

Method	Removed(%)	Δcost(%)	Infes(%)	Time(%)
Benchmark	0	0.00	0.000	100
Single	100	-0.04	0.044	19.8
Perfect	99.9	0.01	0.001	15.9
Naive	99.2	0.00	0.000	16.5
ConGen	99.9	0.00	0.000	27.0
Roald	85.8	0.00	0.000	24.8
Data-driven	99.8	0.00	0.000	17.5
DD+CG	99.8	0.00	0.000	22.6

Roald is conservative and keeps 14% of line constraints

What about the results?

Low-congested case

Method	Removed(%)	Δcost(%)	Infes(%)	Time(%)
Benchmark	0	0.00	0.000	100
Single	100	-0.04	0.044	19.8
Perfect	99.9	0.01	0.001	15.9
Naive	99.2	0.00	0.000	16.5
ConGen	99.9	0.00	0.000	27.0
Roald	85.8	0.00	0.000	24.8
Data-driven	99.8	0.00	0.000	17.5
DD+CG	99.8	0.00	0.000	22.6

Naive and Data-driven achieve the highest time reduction

What about the results?

High-congested case

Method	Removed(%)	Δcost(%)	Infes(%)	Time(%)
Benchmark	0	0.00	0.000	100.0
Single-bus	100	-20.14	10.557	3.1
Perfect	95.1	1.65	0.435	5.7
Naive	72.5	0.01	0.001	33.3
ConGen	83.2	0.00	0.000	65.4
Roald	21.7	0.00	0.000	80.4
Data-driven	89.4	0.79	0.107	12.6
DD+CG	88.8	0.00	0.000	32.7

What about the results?

High-congested case

Method	Removed(%)	Δcost(%)	Infes(%)	Time(%)
Benchmark	0	0.00	0.000	100.0
Single-bus	100	-20.14	10.557	3.1
Perfect	95.1	1.65	0.435	5.7
Naive	72.5	0.01	0.001	33.3
ConGen	83.2	0.00	0.000	65.4
Roald	21.7	0.00	0.000	80.4
Data-driven	89.4	0.79	0.107	12.6
DD+CG	88.8	0.00	0.000	32.7

Single-bus approach is fast but provides catastrophic results

What about the results?

High-congested case

Method	Removed(%)	Δcost(%)	Infes(%)	Time(%)
Benchmark	0	0.00	0.000	100.0
Single-bus	100	-20.14	10.557	3.1
Perfect	95.1	1.65	0.435	5.7
Naive	72.5	0.01	0.001	33.3
ConGen	83.2	0.00	0.000	65.4
Roald	21.7	0.00	0.000	80.4
Data-driven	89.4	0.79	0.107	12.6
DD+CG	88.8	0.00	0.000	32.7

Perfect provides suboptimal results due to quasi-active constraints

What about the results?

High-congested case

Method	Removed(%)	Δcost(%)	Infes(%)	Time(%)
Benchmark	0	0.00	0.000	100.0
Single-bus	100	-20.14	10.557	3.1
Perfect	95.1	1.65	0.435	5.7
Naive	72.5	0.01	0.001	33.3
ConGen	83.2	0.00	0.000	65.4
Roald	21.7	0.00	0.000	80.4
Data-driven	89.4	0.79	0.107	12.6
DD+CG	88.8	0.00	0.000	32.7

ConGen removes a lot of constraints but requires high time

What about the results?

High-congested case

Method	Removed(%)	Δcost(%)	Infes(%)	Time(%)
Benchmark	0	0.00	0.000	100.0
Single-bus	100	-20.14	10.557	3.1
Perfect	95.1	1.65	0.435	5.7
Naive	72.5	0.01	0.001	33.3
ConGen	83.2	0.00	0.000	65.4
Roald	21.7	0.00	0.000	80.4
Data-driven	89.4	0.79	0.107	12.6
DD+CG	88.8	0.00	0.000	32.7

Roald only removes 22% of constraints and limits time reduction

What about the results?

High-congested case

Method	Removed(%)	Δcost(%)	Infes(%)	Time(%)
Benchmark	0	0.00	0.000	100.0
Single-bus	100	-20.14	10.557	3.1
Perfect	95.1	1.65	0.435	5.7
Naive	72.5	0.01	0.001	33.3
ConGen	83.2	0.00	0.000	65.4
Roald	21.7	0.00	0.000	80.4
Data-driven	89.4	0.79	0.107	12.6
DD+CG	88.8	0.00	0.000	32.7

Data-driven removes most constraints but involves small infeasibilities

What about the results?

High-congested case

Method	Removed(%)	Δcost(%)	Infes(%)	Time(%)
Benchmark	0	0.00	0.000	100.0
Single-bus	100	-20.14	10.557	3.1
Perfect	95.1	1.65	0.435	5.7
Naive	72.5	0.01	0.001	33.3
ConGen	83.2	0.00	0.000	65.4
Roald	21.7	0.00	0.000	80.4
Data-driven	89.4	0.79	0.107	12.6
DD+CG	88.8	0.00	0.000	32.7

DD+CG provides the optimum and significantly reduces time

What about the results?

2000-bus case

Method	Removed(%)	Δcost(%)	Infes(%)	Time(%)
Benchmark	0.0	0.00	0.00	100.0
Single	100.0	-2.17	0.26	0.4
Perfect	99.7	-0.22	0.13	1.0
Naive	92.3	0.00	0.00	10.6
ConGen	98.8	0.00	0.00	8.9
Roald	54.3	0.00	0.00	64.7
Data-driven	98.6	0.04	0.03	2.3
DD+CG	98.5	0.00	0.00	5.3

- Computational burden of Data-driven similar to Single and Perfect
- Solution provided by Data-driven involve tiny inaccuracies
- DD+CG recovers the original solution at lowest time

What about the results?

Method	# Removed	Original solution	Time
Benchmark (BN)	🔴	🟢	⌛ ⌛ ⌛
Single-bus (SB)	🟢	🔴	⌛
Perfect (PI)	🟢	🔴	⌛
Naive (NV)	🟠	🟠	⌛
Constraint gen (CG)	🟢	🟢	⌛ ⌛
Roald (RO)	🟠	🟢	⌛ ⌛
Data-driven (DD)	🟢	🟠	⌛
DD + CG	🟢	🟢	⌛

Conclusions

- Disregarding the capacity constraints of some transmission lines reduces the computational burden of the unit-commitment problem.
- We propose a data-driven approach that takes advantage of historical information to disregard both *redundant* and *inactive* constraints.
- Our method achieves computational savings of 70%-98% depending on the congestion level of the power system and its size.
- Combining our method with a constraint generation procedure guarantees that the solution of the original problem is recovered.

Thanks for the attention!

Questions?

OASYS
Optimization and Analytics for
Sustainable energY Systems

More info: oasys.uma.es
Email: spineda@uma.es