Application of the Level Method for Computing Locational Convex Hull Prices

Nicolas Stevens & Anthony Papavasiliou

July 2022

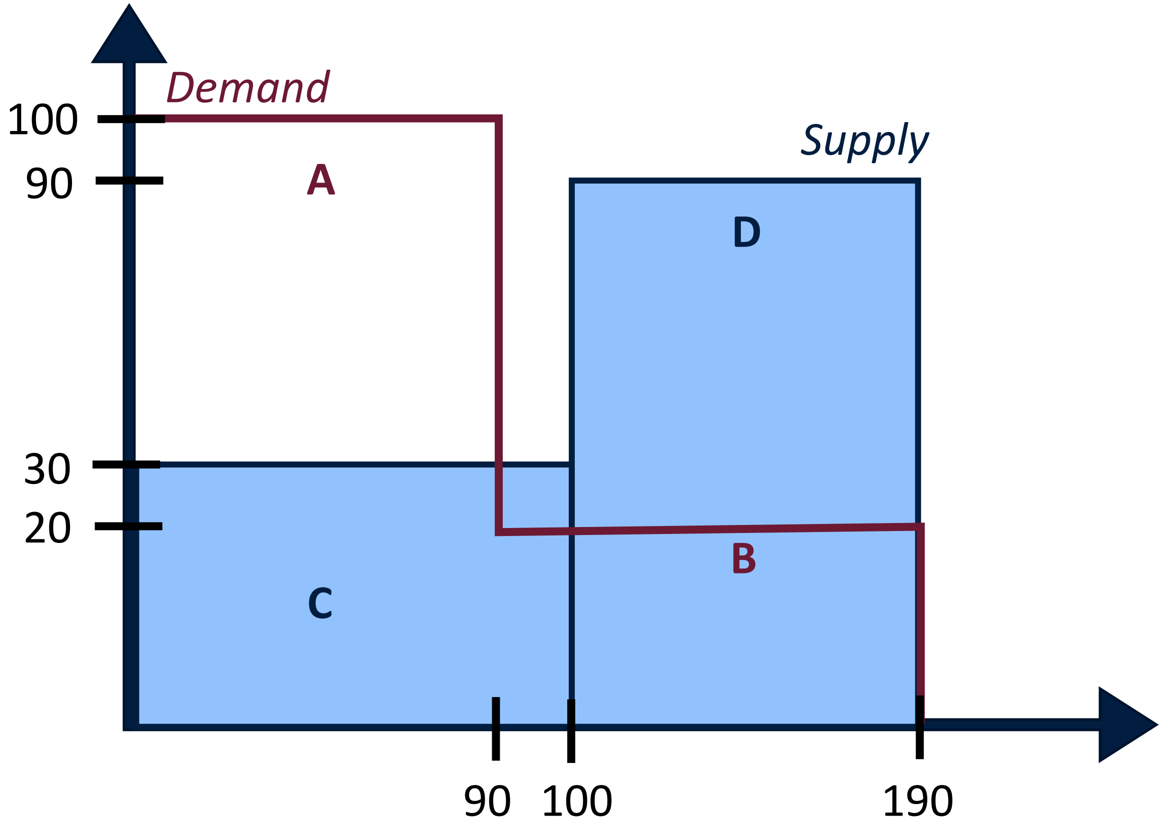
3-6 JULY EURO 2022
AALTO UNIVERSITY
ESPOO FINLAND

EURO2022

UCLouvain

LIDAM
CORE | Louvain Institute of Data Analysis and Modeling in economics and statistics

What is the right price?



- **C** is *indivisible* (all-or-nothing)
- Welfare optimum solution is to clear **A**, **C** and a fraction of **D**
- What is the right price?
  - At 20€/MWh C is not willing to produce
  - At 30€/MWh, B is not willing to consume
- Idea: Combine the **uniform price** with **discriminatory payments** (**uplifts**) to restore the proper incentives
- This let open the question of what is the **right uniform price?**

Example inspired from (Madani et al., 2018)

Plan

Pricing Schemes	• Mapping of the different pricing schemes • Main concepts related to Convex Hull Pricing (CHP)
CHP & the Level Method	• Algorithmic schemes to compute CHP: subgradient and Kelley's algorithm • Level stabilization • Adaptation of the basic algorithm to CHP specificities
Numerical Results	• Size of the EU day-ahead market problem • Comparison of the Level Method and the Dantzig Wolfe algorithm on EU instances
Discussion	• Comparison of the different pricing schemes results • Properties of the pricing schemes
Conclusion	• Perspective for future research

Plan

Pricing Schemes

- Mapping of the different pricing schemes
- Main concepts related to Convex Hull Pricing (CHP)

CHP & the Level Method

- Algorithmic schemes to compute CHP: subgradient and Kelley's algorithm
- Level stabilization
- Adaptation of the basic algorithm to CHP specificities

Numerical Results

- Size of the EU day-ahead market problem
- Comparison of the Level Method and the Dantzig Wolfe algorithm on EU instances

Discussion

- Comparison of the different pricing schemes results
- Properties of the pricing schemes

Conclusion

- Perspective for future research

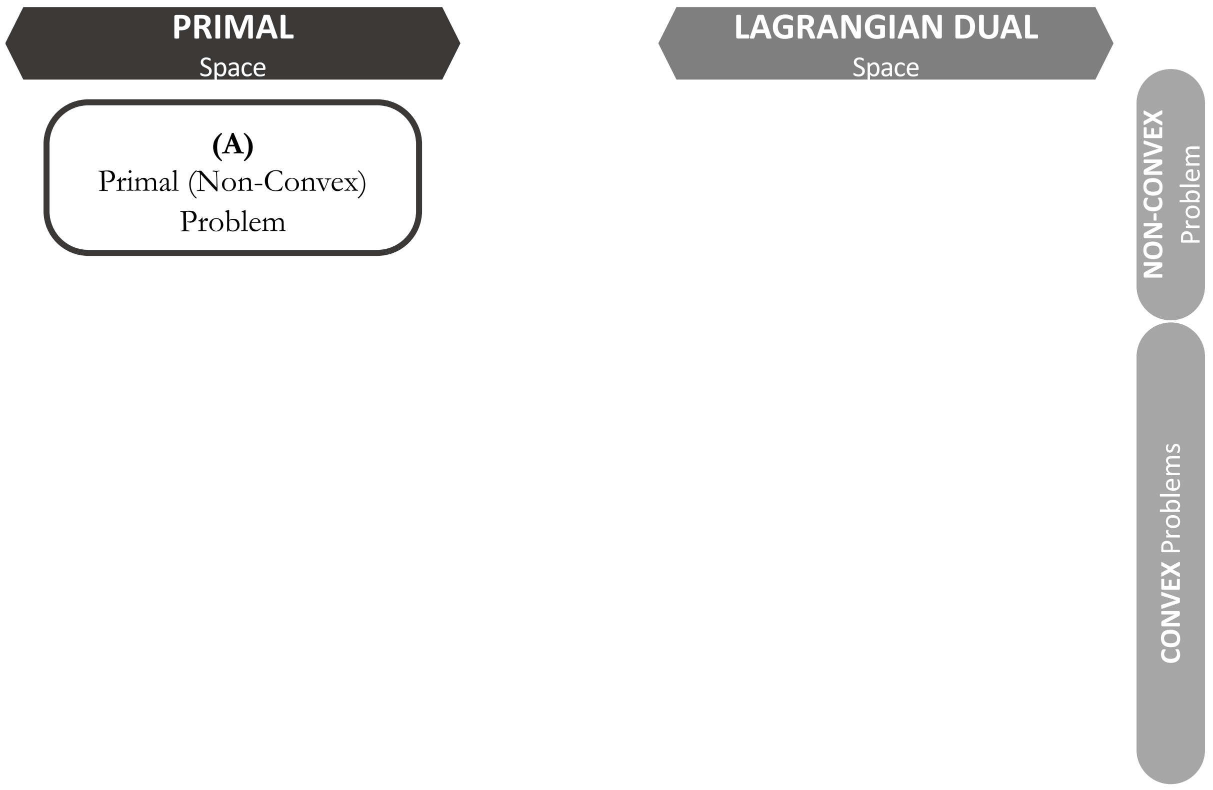
Non-convex pricing schemes

$$\min_{c,p,u,f} \sum_{g \in \mathcal{G}} c_g$$

$$(\pi_t^i) \quad \sum_{g \in \mathcal{G}_i} p_{g,t} - D_t^i = \sum_{\substack{l \in \\ from(i)}} f_{l,t} - \sum_{\substack{l \in \\ to(i)}} f_{l,t} \quad \forall i, t$$

$$(c_g, p_{g,t}, u_{g,t}) \in \mathcal{X}_g \quad \forall g \in \mathcal{G}$$

$$f \in \mathcal{F}$$



Non-convex pricing schemes

CORE | Louvain Institute of Data Analysis and Modeling in economics and statistics

$$\min_{c,p,u,f} \sum_{g\in\mathcal{G}} c_g$$

$$(\pi_t^i) \quad \sum_{g\in\mathcal{G}_i} p_{g,t} - D_t^i = \sum_{\substack{l\in\\ from(i)}} f_{l,t} - \sum_{\substack{l\in\\ to(i)}} f_{l,t} \quad \forall i,t$$

$$(c_g, p_{g,t}, u_{g,t}) \in \mathcal{X}_g \quad \forall g\in\mathcal{G}$$

$$f \in \mathcal{F}$$

PRIMAL Space

(A) Primal (Non-Convex) Problem

LAGRANGIAN DUAL Space

NON-CONVEX Problem

(Γ) Partial Lag. Relaxation (only MC constraints)

CONVEX Problems

$$\pi^{CHP} = \arg\max_{\pi} L(\pi)$$

$$L(\pi) = \sum_{i,t} \pi_t^i D_t^i - \sum_{g\in\mathcal{G}} \max_{(c,p,u)_g\in\mathcal{X}_g} \left\{ \sum_t p_{g,t}\pi_t^{i(g)} - c_g \right\} + \min_{f\in\mathcal{F}} \left\{ \sum_{i,t} \pi_t^i \left(\sum_{\substack{l\in\\ from(i)}} f_{l,t} - \sum_{l\in to(i)} f_{l,t} \right) \right\}$$

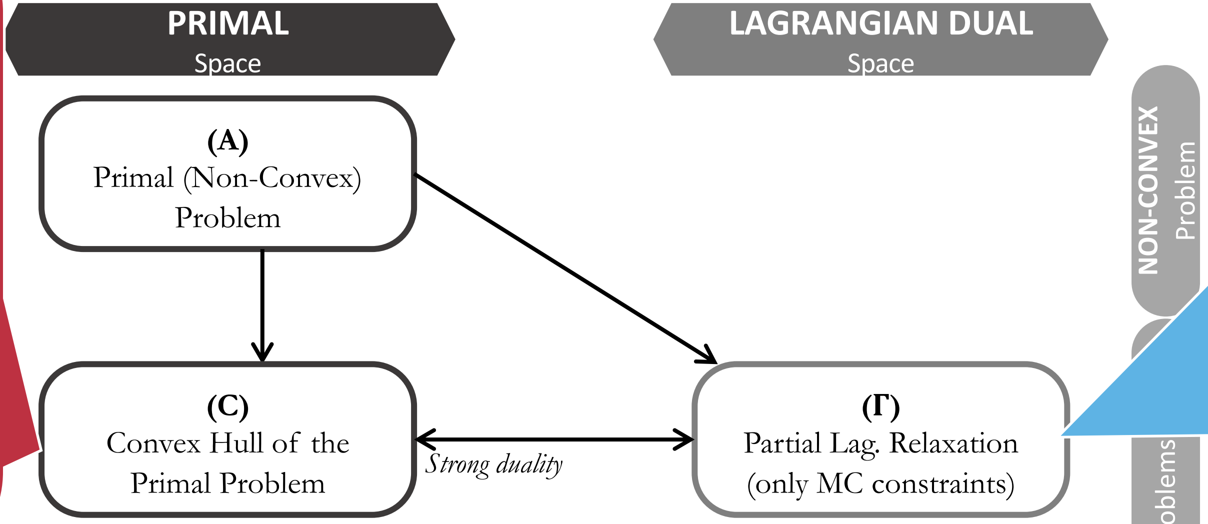
DUAL approaches

- Convex Hull Pricing
- Hogan and Ring, 2003
- **Uniform** prices minimizing the side payments
- PRICES = Lag. multipliers of the Lagrangian relaxation
- This is a **convex** and **non-smooth** problem with a **first-order oracle**
  - Wang et al., 2013
  - Andrianesis et al., 2021

Non-convex pricing schemes

PRIMAL approaches

- In the **primal space**: solving this Lagrangian relaxation amounts to shaping the CH of the primal constraints
- Approach: develop tight — **BUT custom** — formulation specific to the targeted problem
  - Hua and R. Baldick, 2017
  - Yu et al., 2020
  - Álvarez et al., 2020
  - Madani et al., 2018



$$\min_{c,p,u,f} \sum_{g\in\mathcal{G}} c_g$$

$$(\pi_{i,t}^{CHP}) \quad \sum_{g\in\mathcal{G}_i} p_{g,t} - D_t^i = \sum_{\substack{l\in\\ from(i)}} f_{l,t} - \sum_{\substack{l\in\\ to(i)}} f_{l,t} \quad \forall i,t$$

$$(c_g, p_{g,t}, u_{g,t}) \in \text{conv}(\mathcal{X}_g)$$

$$f \in \mathcal{F}$$

$$\pi^{CHP} = \arg\max_{\pi} L(\pi)$$

$$L(\pi) = \sum_{i,t} \pi_t^i D_t^i - \sum_{g\in\mathcal{G}} \max_{(c,p,u)_g\in\mathcal{X}_g} \left\{ \sum_t p_{g,t}\pi_t^{i(g)} - c_g \right\} + \min_{f\in\mathcal{F}} \left\{ \sum_{i,t} \pi_t^i \left(\sum_{\substack{l\in\\ from(i)}} f_{l,t} - \sum_{l\in to(i)} f_{l,t} \right) \right\}$$

DUAL approaches

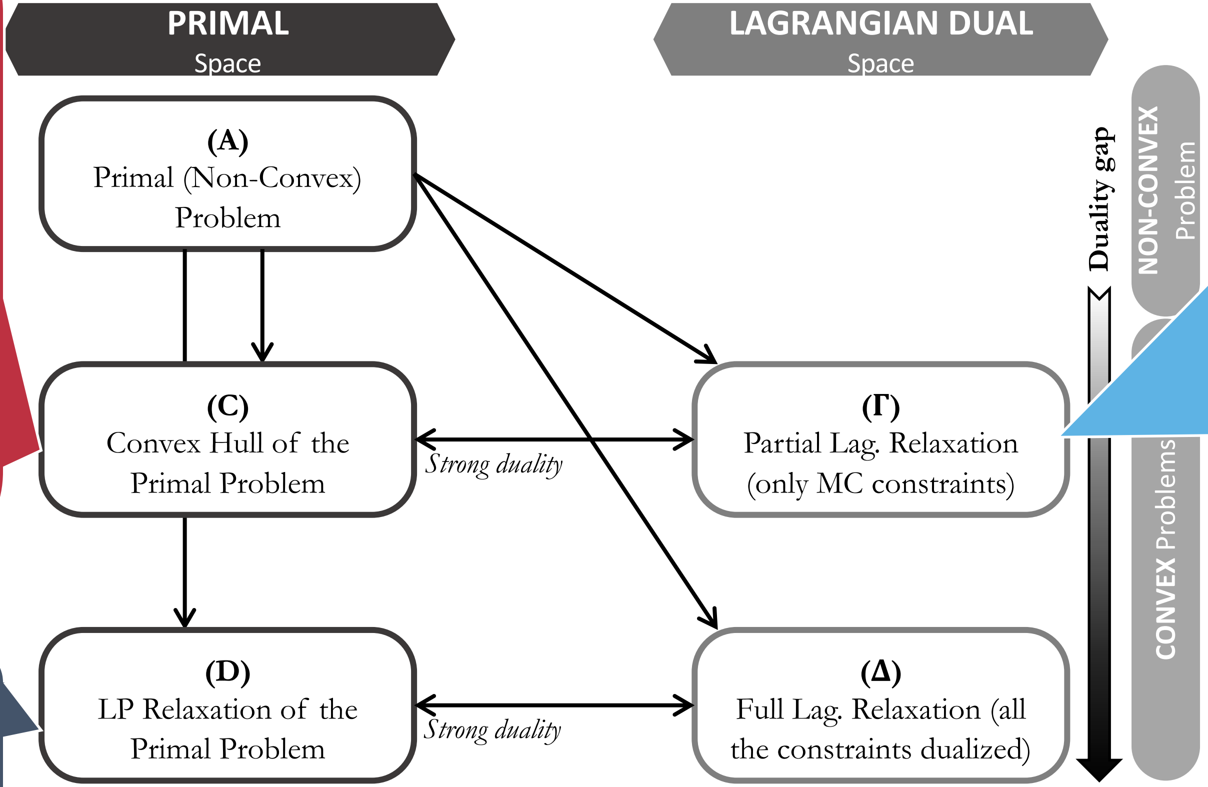
- Convex Hull Pricing
- Hogan and Ring, 2003
- **Uniform** prices minimizing the side payments
- PRICES = Lag. multipliers of the Lagrangian relaxation
- This is a **convex** and **non-smooth** problem with a **first-order oracle**
  - Wang et al., 2013
  - Andrianesis et al., 2021

Non-convex pricing schemes

PRIMAL approaches

- In the **primal space**: solving this Lagrangian relaxation amounts to shaping the CH of the primal constraints
- Approach: develop tight — **BUT custom** — formulation specific to the targeted problem
  - Hua and R. Baldick, 2017
  - Yu et al., 2020
  - Álvarez et al., 2020
  - Madani et al., 2018

- Scalable **approximation**: in some cases = CHP
- Considered by PJM (PJM Interconnection, 2017)



DUAL approaches

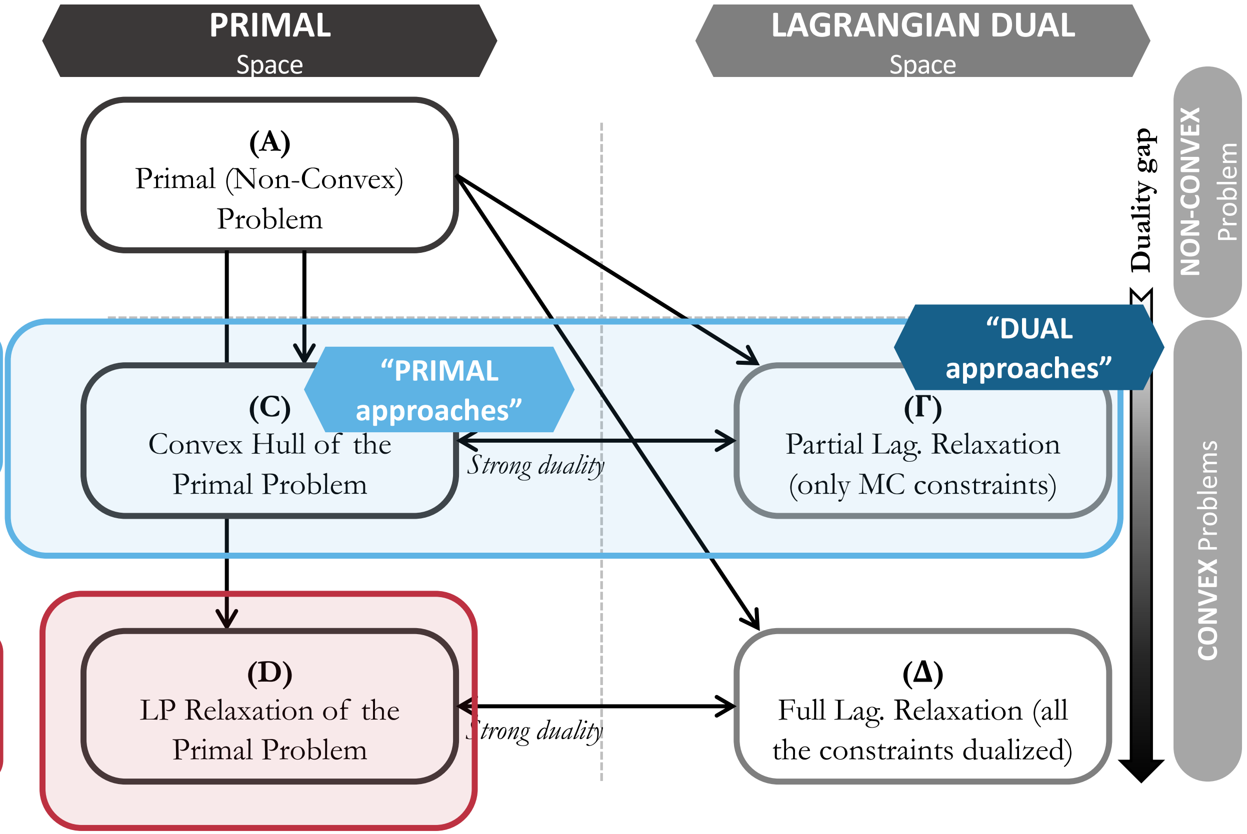
- Convex Hull Pricing
- Hogan and Ring (2003)
- **Uniform** prices minimizing the side payments
- PRICES = Lag. multipliers of the Lagrangian relaxation
- This is a **convex** and **non-smooth** problem with a **first-order oracle**
  - Wang et al. (2013)
  - Andrianesis et al. (2021)

Non-convex pricing schemes

1 IP pricing (“O’Neil pricing”)*: marginal pricing (with binary variable fixed) + side payments

2 Convex Hull Pricing (CHP)

3 Extended LMP (ELMP)



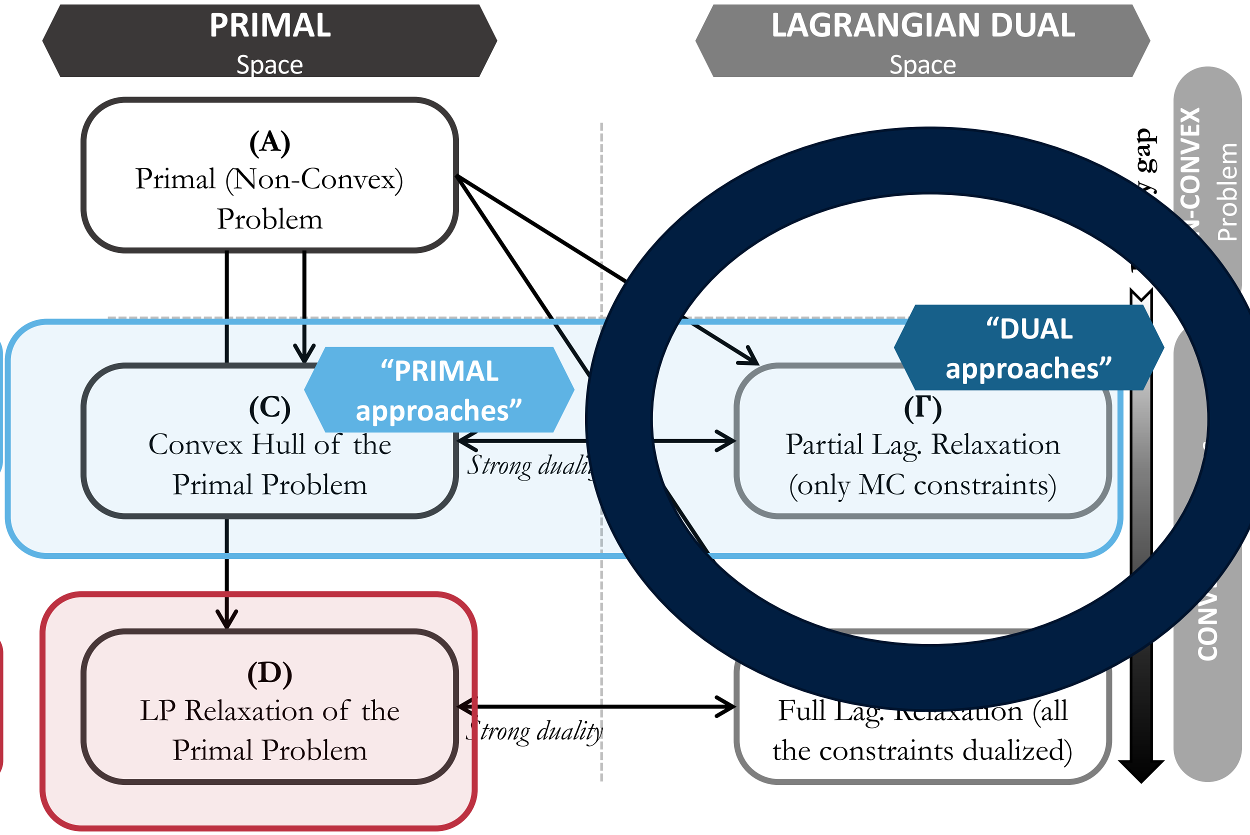
* O’Neill et al., 2005

Non-convex pricing schemes

1 **IP pricing ("O'Neil pricing")*: marginal pricing (with binary variable fixed) + side payments**

2 **Convex Hull Pricing (CHP)**

3 **Extended LMP (ELMP)**



FOCUS of our paper: develop a scalable algorithm for solving the Lagrangian relaxation

- Advantages of **dual approaches**: **Generic** (work for any generator model)
  - US models
  - Or EU model
- While a **primal approach**:
  - Either is limited to an **approximation** of CHP
  - Or need to write a new paper for each new market model

* O'Neill et al., 2005

Plan

Pricing Schemes	• Mapping of the different pricing schemes • Main concepts related to Convex Hull Pricing (CHP)
CHP & the Level Method	• **Algorithmic schemes to compute CHP: subgradient and Kelley's algorithm** • **Level stabilization** • **Adaptation of the basic algorithm to CHP specificities**
Numerical Results	• Size of the EU day-ahead market problem • Comparison of the Level Method and the Dantzig Wolfe algorithm on EU instances
Discussion	• Comparison of the different pricing schemes results • Properties of the pricing schemes
Conclusion	• Perspective for future research

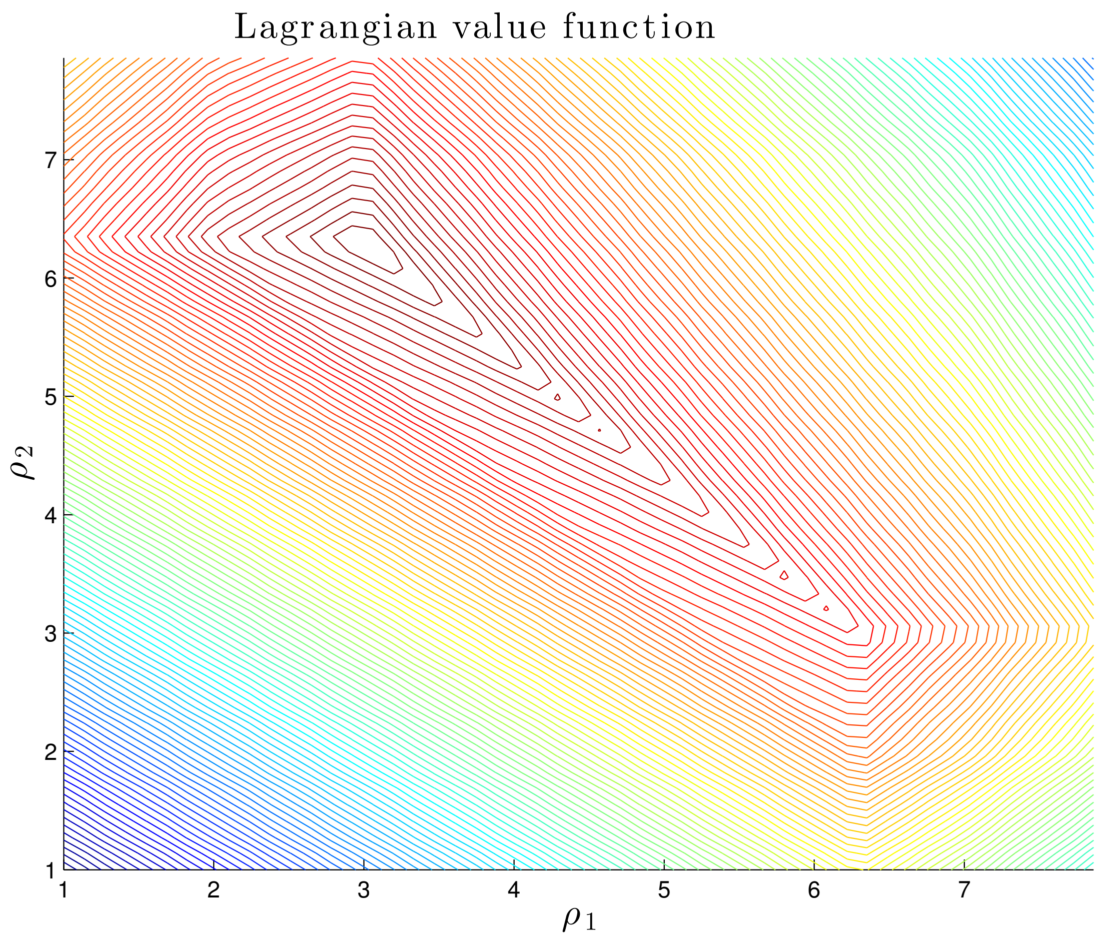
An example: the subgradient algorithm

Toy example (Ruiz et al., 2012)	
Demand: [30, 40] *MW*	*Two-periods, single node*

G	C_g^P	C_g^{SU}	$P_g^{\min}$	$P_g^{\max}$
SMOKESTACK01	*3*	*53*	*0*	*16*
SMOKESTACK02	*3*	*53*	*0*	*16*
SMOKESTACK03	*3*	*53*	*0*	*16*
HIGH_TECH01	*2*	*30*	*0*	*7*
HIGH_TECH02	*2*	*30*	*0*	*7*
MED_TECH01	*7*	*0*	*2*	*6*

GENERIC ALGORITHMIC SCHEME

1. Given a price π_k, $L(\pi_k)$ and $\partial L(\pi_k)$ are evaluated
2. Given this information, a new price π_{k+1} is generated
3. If stopping criterion, stop. Otherwise, go to 1



An example: the subgradient algorithm

Toy example (Ruiz et al., 2012)

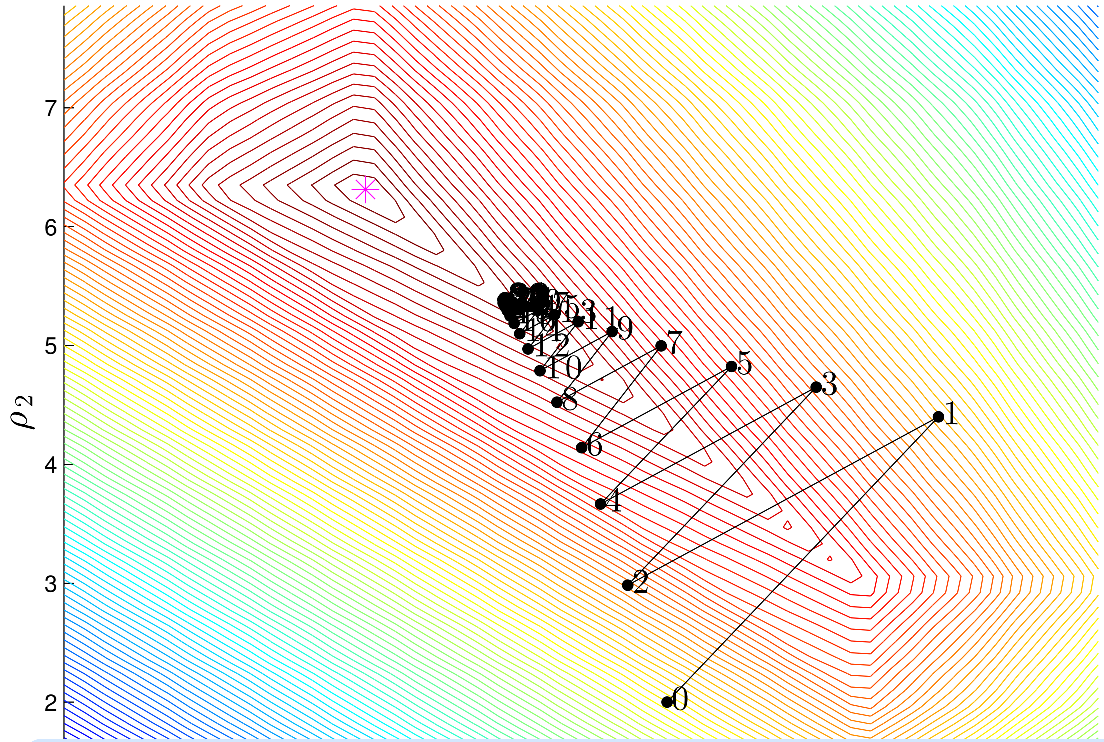
Demand: [30, 40] *MW*	*Two-periods, single node*

G	C_g^P	C_g^{SU}	$P_g^{\min}$	$P_g^{\max}$
SMOKESTACK01	3	53	0	16
SMOKESTACK02	3	53	0	16
SMOKESTACK03	3	53	0	16
HIGH_TECH01	2	30	0	7
HIGH_TECH02	2	30	0	7
MED_TECH01	7	0	2	6

GENERIC ALGORITHMIC SCHEME

1. Given a price π_k, $L(\pi_k)$ and $\partial L(\pi_k)$ are evaluated
2. Given this information, a new price π_{k+1} is generated
3. If stopping criterion, stop. Otherwise, go to 1

Lagrangian value function and Subgradient path



- **Supgradient** algorithm is **memoryless**
- Typical oscillation behavior

→ In moderate dimension, such as for our CHP problem, there are more optimistic algorithmic schemes

The Kelley's algorithm (Nesterov, 2004)

- Basis for the Level Method
- Based on the idea of iteratively constructing a **model**: the piecewise linear function $L(\pi)$ is **upper-approximated** at each iterate by a model function $\hat{L}(\pi, k)$ consisting of **supporting hyperplanes**

Model function

$$\hat{L}(\pi, k) = \min_{j=0..k} [\langle g_j, \pi - \pi_j \rangle + L(\pi_j)] \quad (5)$$

Master program

$$\max_{\pi \in Q, \theta} \quad \theta \qquad s.t. \quad \theta \leq \langle g_j, \pi \rangle + b_j \quad \forall j = 0..k \quad (6)$$

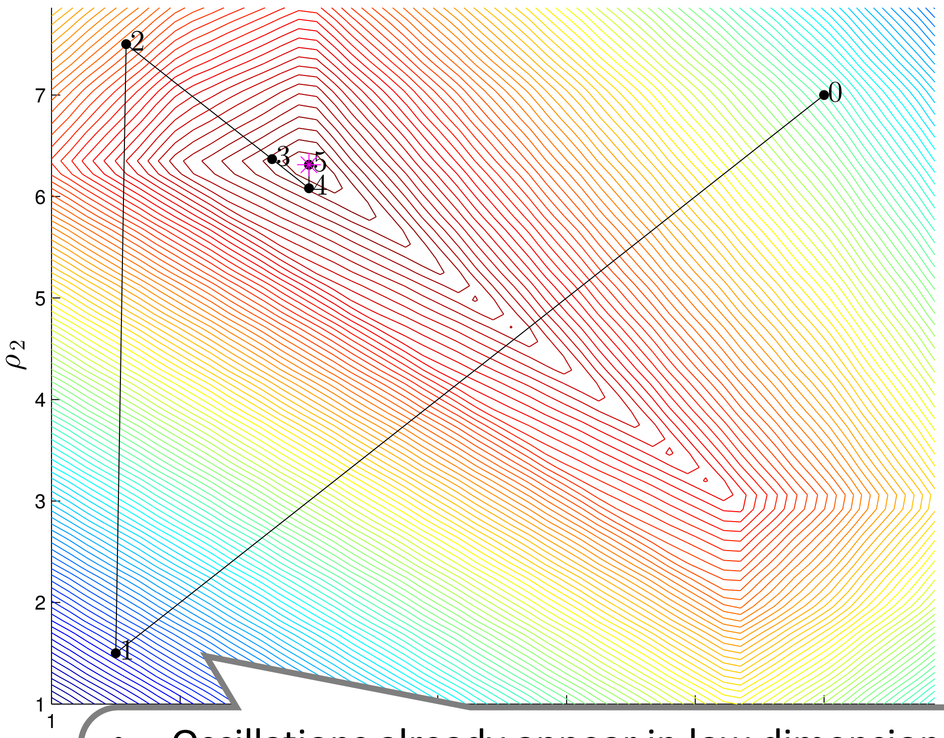
Update rule

$$\pi_{k+1} = \arg\max_{\pi} \hat{L}(\pi, k). \quad (7)$$

Stopping criterion

$$\frac{UB_k - LB_k}{|UB_k|} \leq \epsilon \quad (8)$$

Lagrangian value function and Kelley path



- Oscillations already appear in low dimension
- **Unstable** in high dimension: adding a new supporting hyperplane can move the optimum far from the previous point

The Level stabilization (Nesterov, 2004)

- The underlying idea of the Level Method is to **update prices more smoothly**
- Select the next iterate π_{k+1} so that it is better than π_k (as evaluated by $\hat{L}(\pi, k)$), without being optimal at all costs

Same **Model function** as Kelley

Same **Master program** as Kelley

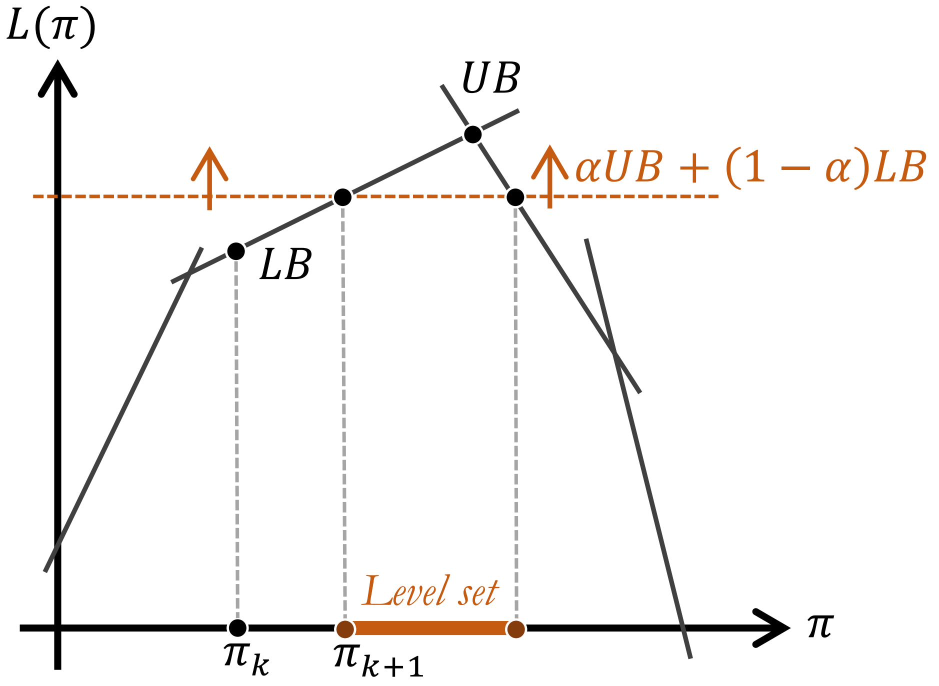
NEW **Update rule** Projection prog.

$$\min_{\pi \in Q} \quad ||\pi - \pi_k||_2^2$$
$$s.t. \quad \langle g_j, \pi \rangle + b_j \geq \alpha UB_k + (1-\alpha) LB_k \quad \forall j = 0..k \qquad (9)$$

α is the **projection parameter**

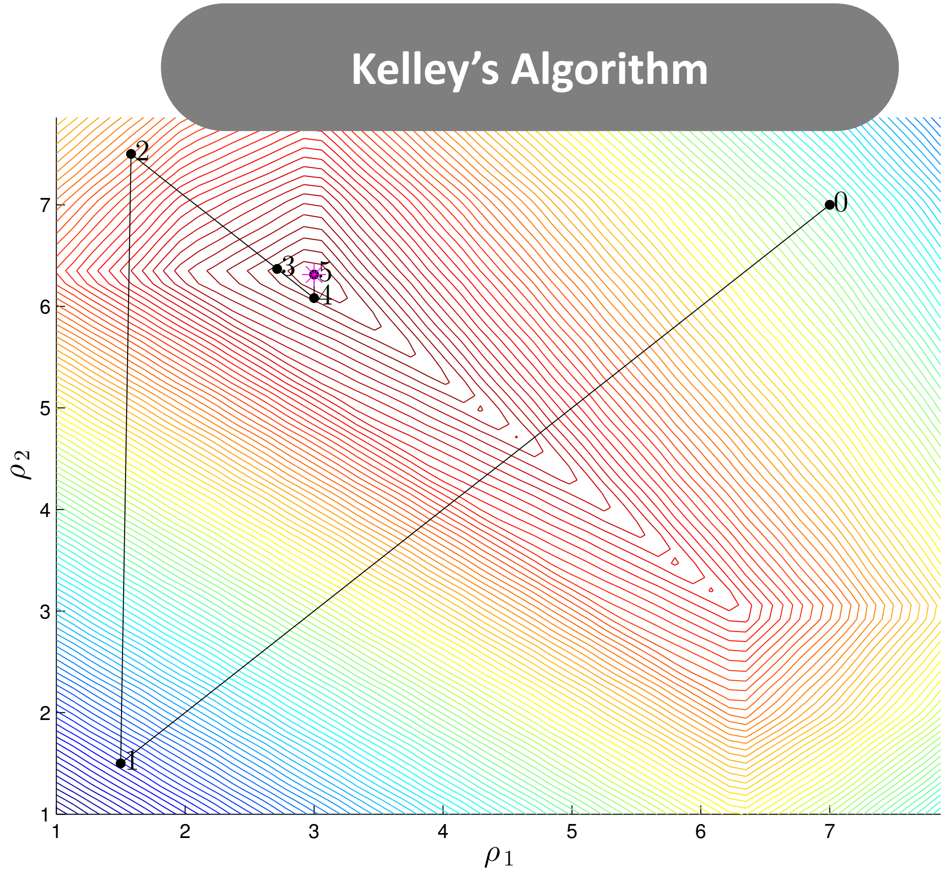
- $\alpha = 1$: Kelley's algorithm
- $\alpha = 0$: the iterate does not move

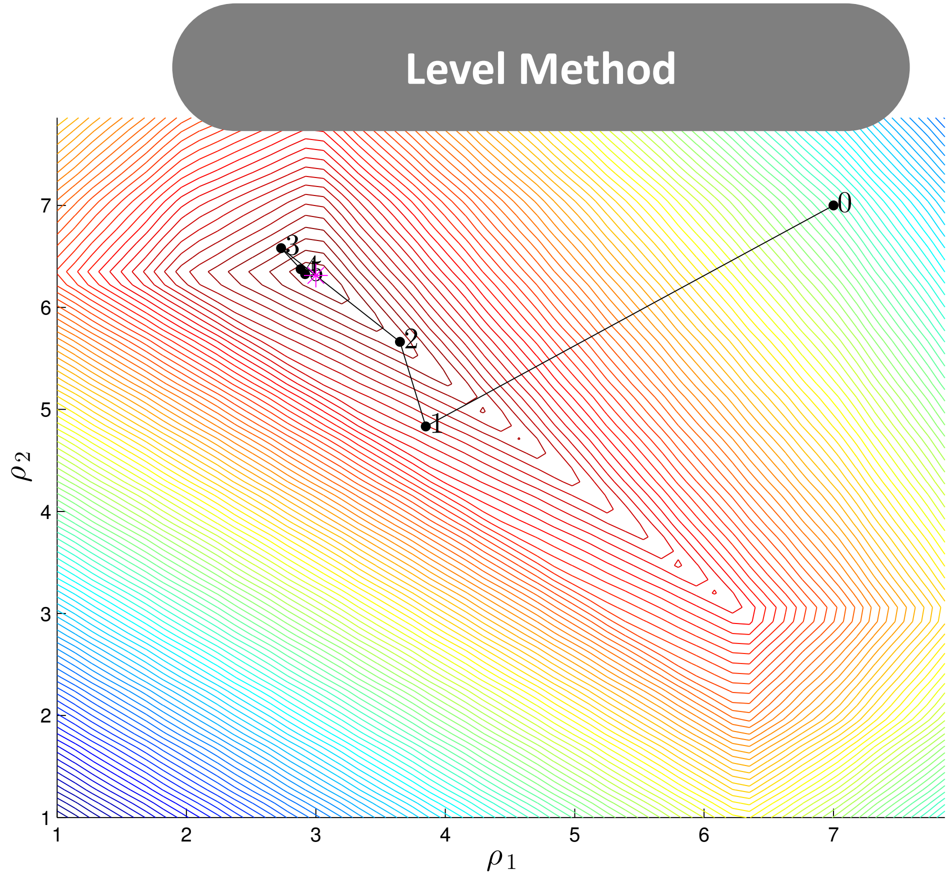
Same **Stopping criterion** as Kelley



The Level Method stabilizes Kelley's path – illustration on 2D example

CORE | Louvain Institute of Data Analysis and Modeling in economics and statistics





Adaptation of the Level Method to the CHP specificities

- Adaptations of the Level Method to the specifies of the Convex Hull Pricing problem
  - Dualization of the **network** to include it explicitly in the master program
  - Separability of the subproblems → **multi-cut Level Method**

- The Level Method implies a parameter α
  - It is **NOT a "heuristic"**
  - The Method is **robust** towards the exact value of α

TABLE III
SENSITIVITY OF THE LEVEL METHOD WITH RESPECT TO PARAMETER α ON THE BE 96-PERIOD CASE (AVERAGE OVER 6 INSTANCES)

α	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Level iter	54	44	45	43	41	43	45	48	60

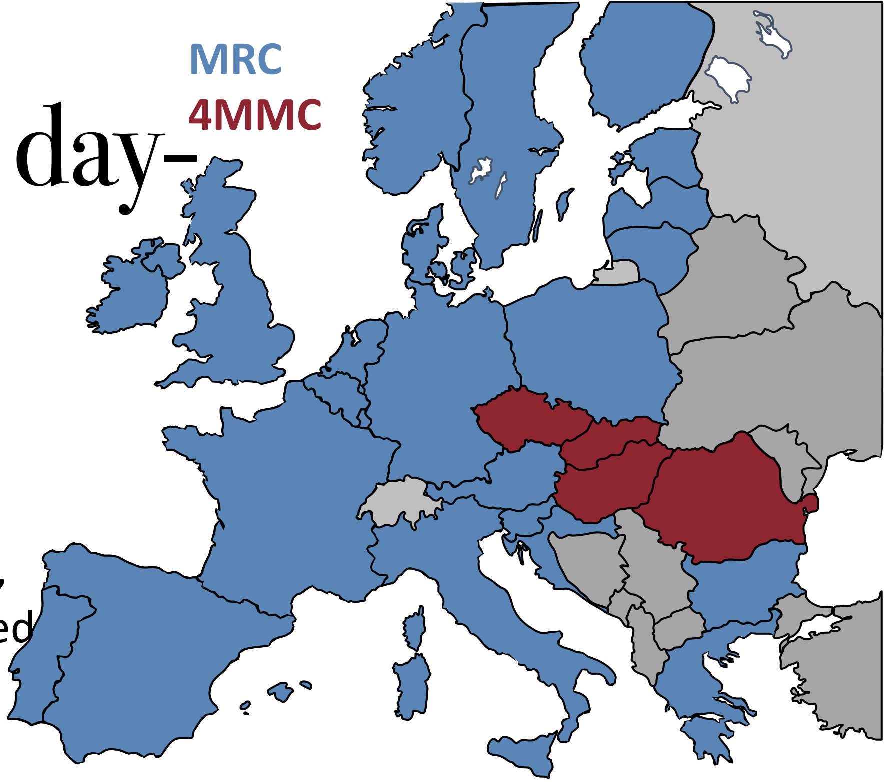
Plan

Pricing Schemes	• Mapping of the different pricing schemes • Main concepts related to Convex Hull Pricing (CHP)
CHP & the Level Method	• Algorithmic schemes to compute CHP: subgradient and Kelley's algorithm • Level stabilization • Adaptation of the basic algorithm to CHP specificities
Numerical Results	• Size of the EU day-ahead market problem • Comparison of the Level Method and the Dantzig Wolfe algorithm on EU instances
Discussion	• Comparison of the different pricing schemes results • Properties of the pricing schemes
Conclusion	• Perspective for future research

Computing CHP in the European day-ahead auction

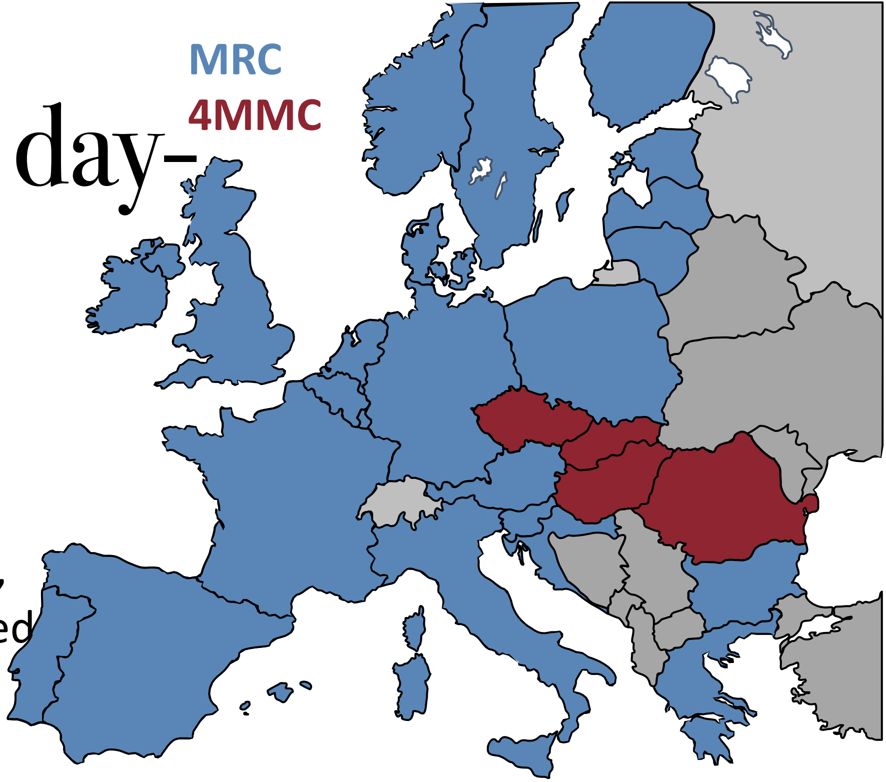
Requirements*

- Euphemia is afforded **12 minutes** of run time
- The market model includes a network of ~ **40 bidding zones**, and its geographic footprint is expected to be further enlarged
- The market model is expected to move towards **15-minute granularity** by 2022 (a horizon of **96 periods**)



* (Nemo Committee, 2019)

Computing CHP in the European day-ahead auction



Requirements*

- Euphemia is afforded **12 minutes** of run time
- The market model includes a network of ~ **40 bidding zones**, and its geographic footprint is expected to be further enlarged
- The market model is expected to move towards **15-minute granularity** by 2022 (a horizon of **96 periods**)

TABLE II
DESCRIPTION OF THE SIZE OF THE EU INSTANCES.

Test case	Bidding Zones	Lines	Generators
BE	30	30	74
BE-NL	59	63	145

Test cases

- Tested on 2 sets of cases
  - **FERC data** (11 instances, no network, >900 generators) (Krall et al. 2012)
  - **Central Europe data** (6 times series, 2 different networks) (Aravena and Papavasiliou, 2016)
- The Level Method is benchmarked against a **Dantzig-Wolfe algorithm** (Andrianesis et al. 2021)
- Implemented in Julia (JuMP), Run on a personal computer (Intel Core i5, 2.6 GHz, 8 GB of RAM) , Gurobi 9.1.1, Stopping criterion: 0.01%

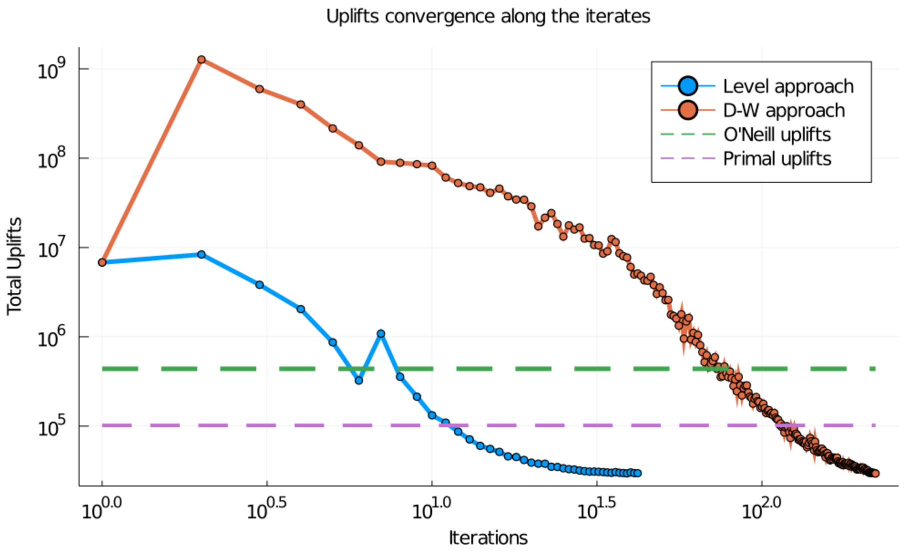
* (Nemo Committee, 2019)

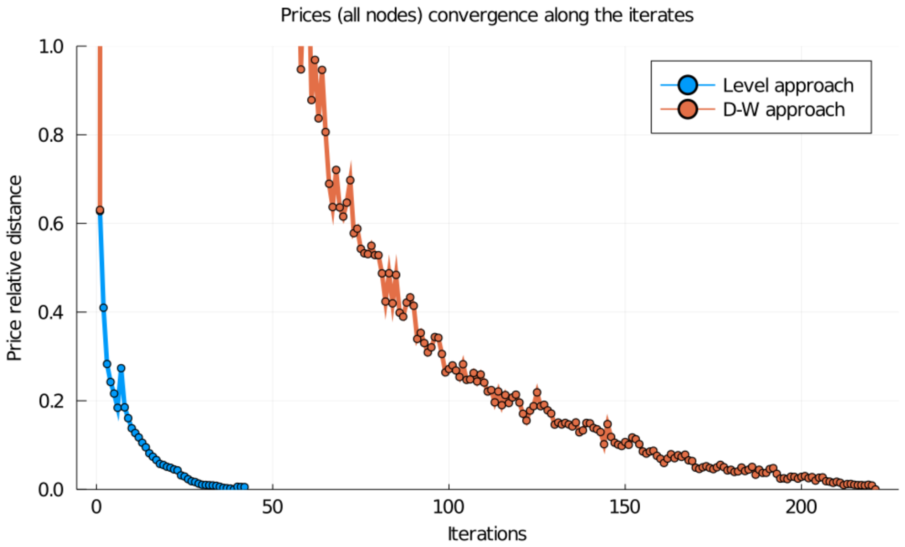
Comparison of the Level Method and Dantzig-Wolfe algorithm

TABLE IV
RESULTS OF THE LEVEL METHOD AND THE DANTZIG-WOLFE ALGORITHM ON THE BE TEST CASE (AVERAGE OVER 6 INSTANCES)

horizon	12	24	48	96
Dispatch Cost [€]	2,759,706	4,956,513	11,328,351	24,097,373
O'Neill Uplifts [€]	377,528	146,167	281,649	2,617,852
Primal Meth. Uplifts [€]	50,871	64,323	83,172	98,391
CHP Uplifts [€]	7,237	11,905	21,745	31,403
Level iter	19	26	32	44
Level av. time/iter[a][s]	0.5 (0.05)	0.8 (0.1)	2.0 (0.4)	5.8 (1.6)
Level total run time [s]	10	21	65	255
D-W iter	19	40	77	236
D-W av. time/iter[a][s]	0.4 (0.02)	0.7 (0.1)	1.9 (0.3)	6.9 (2.1)
D-W total run time [s]	7	27	146	1622

[a] $(\cdot)$ denotes the average time per iterate for solving the "master programs" (i.e. master plus projection in the case of the Level Method).





Comparison of the Level Method and Dantzig-Wolfe algorithm

LiDAM
CORE | Louvain Institute of Data Analysis and Modeling in economics and statistics

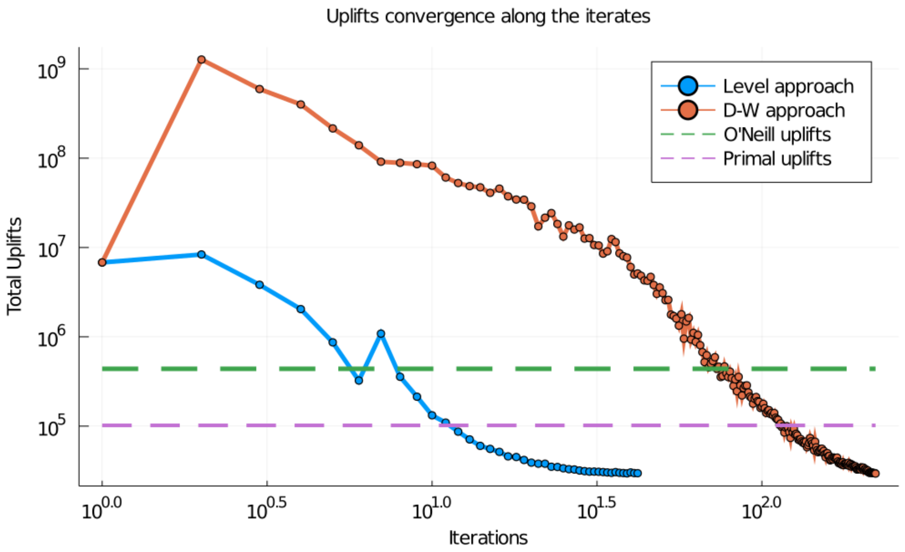
TABLE IV
RESULTS OF THE LEVEL METHOD AND THE DANTZIG-WOLFE ALGORITHM ON THE BE TEST CASE (AVERAGE OVER 6 INSTANCES)

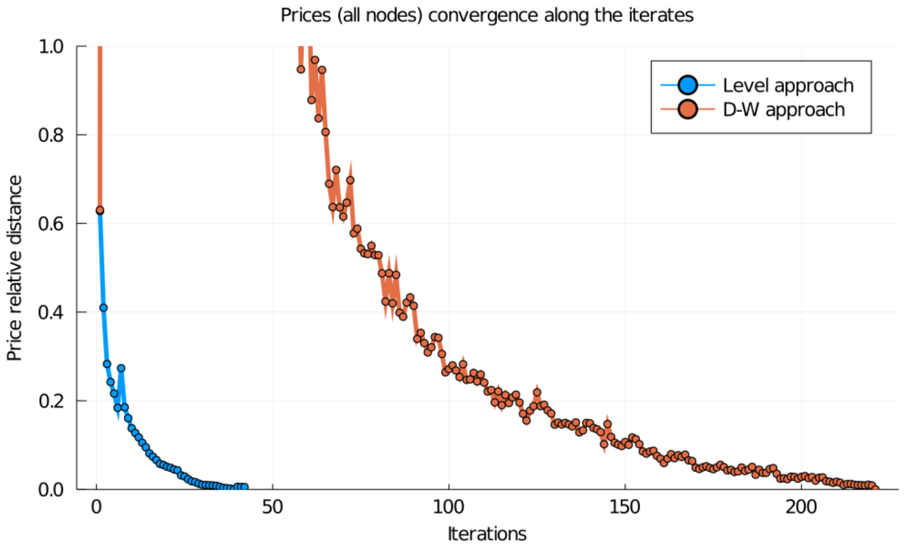
1

The Level Method scale well with the dimension
Better than the D-W

horizon	12	24	48	96
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]7,373
[illegible]	[illegible]	[illegible]	[illegible]	[illegible],852
Pr[illegible]	[illegible]	[illegible]	[illegible]	[illegible]391
[illegible]	[illegible]	[illegible]	[illegible]	[illegible],403
Level iter	19	26	32	44
Level av. time/iter[a][s]	0.5 (0.05)	0.8 (0.1)	2.0 (0.4)	5.8 (1.6)
Level total run time [s]	10	21	65	255
D-W iter	19	40	77	236
D-W av. time/iter[a][s]	0.4 (0.02)	0.7 (0.1)	1.9 (0.3)	6.9 (2.1)
D-W total run time [s]	7	27	146	1622

[a] $(\cdot)$ denotes the average time per iterate for solving the "master programs" (i.e. master plus projection in the case of the Level Method).





Comparison of the Level Method and Dantzig-Wolfe algorithm

CORE | Louvain Institute of Data Analysis and Modeling in economics and statistics

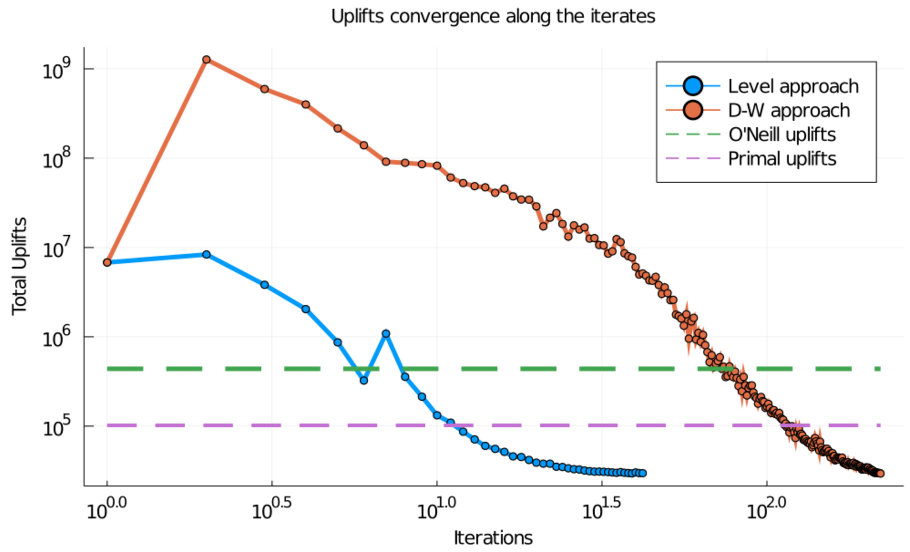
TABLE IV
RESULTS OF THE LEVEL METHOD AND THE DANTZIG-WOLFE ALGORITHM ON THE BE TEST CASE (AVERAGE OVER 6 INSTANCES)

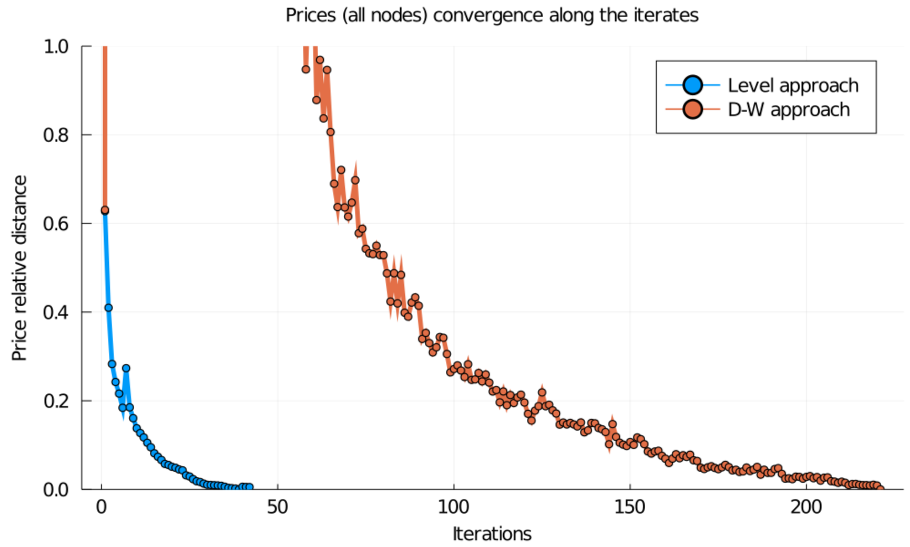
2

On a test case of the EU-like dimension, the Level Method retrieve a solution in a reasonable time

horizon	12	24	48	96
				7,373
				,852
Pr				391
				,403
Level iter	19	26	32	44
Level av. time/iter^a[s]	0.5 (0.05)	0.8 (0.1)	2.0 (0.4)	5.8 (1.6)
Level total run time [s]	10	21	65	255
D-W iter	19	40	77	236
D-W av. time/iter^a[s]	0.4 (0.02)	0.7 (0.1)	1.9 (0.3)	6.9 (2.1)
D-W total run time [s]	7	27	146	1622

^{a} (·) denotes the average time per iterate for solving the "master programs" (i.e. master plus projection in the case of the Level Method).





Comparison of the Level Method and Dantzig-Wolfe algorithm

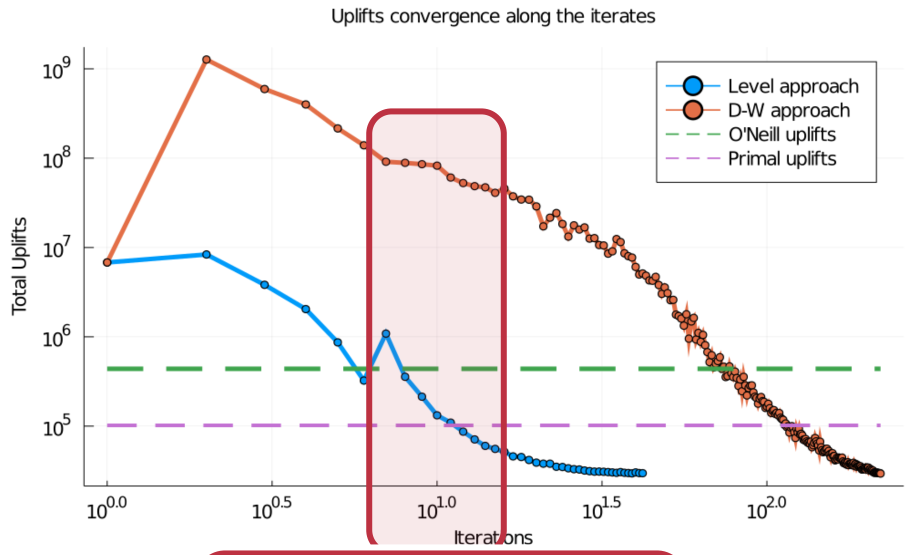
TABLE IV
RESULTS OF THE LEVEL METHOD AND THE DANTZIG-WOLFE ALGORITHM ON THE BE TEST CASE (AVERAGE OVER 6 INSTANCES)

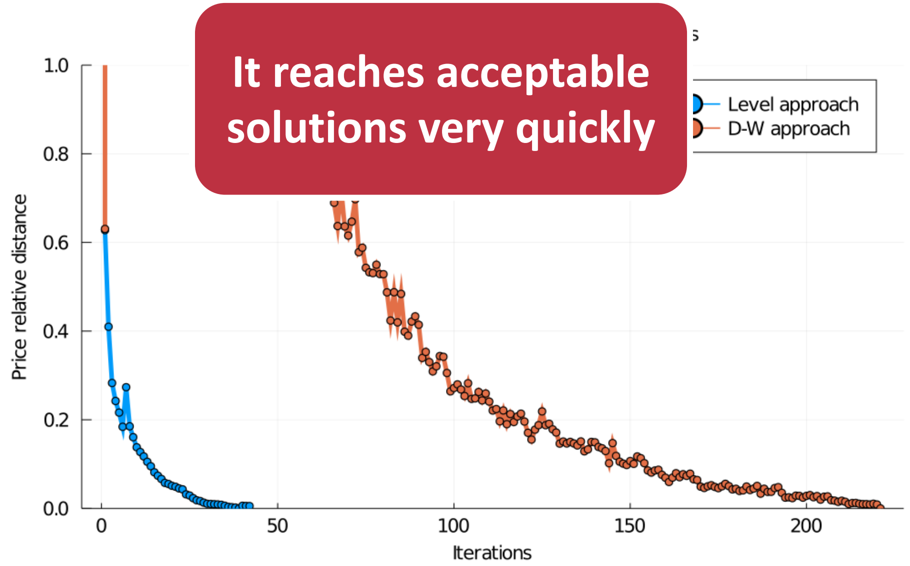
3

Robustness: the Level Method reaches quickly good price candidates (valuable since clearing time limited to 12 min)

horizon	12	24	48	96
[illegible]	[illegible]	[illegible]	[illegible]	7,373
[illegible]	[illegible]	[illegible]	[illegible]	,852
Pr[illegible]	[illegible]	[illegible]	[illegible]	391
[illegible]	[illegible]	[illegible]	[illegible]	,403
Level iter	19	26	32	44
Level av. time/iter[a][s]	0.5 (0.05)	0.8 (0.1)	2.0 (0.4)	5.8 (1.6)
Level total run time [s]	10	21	65	255
D-W iter	19	40	77	236
D-W av. time/iter[a][s]	0.4 (0.02)	0.7 (0.1)	1.9 (0.3)	6.9 (2.1)
D-W total run time [s]	7	27	146	1622

[a] (·) denotes the average time per iterate for solving the "master programs" (i.e. master plus projection in the case of the Level Method).





It reaches acceptable solutions very quickly

Plan

LiDAM
CORE | Louvain Institute of Data Analysis and Modeling in economics and statistics

Pricing Schemes	• Mapping of the different pricing schemes • Main concepts related to Convex Hull Pricing (CHP)
CHP & the Level Method	• Algorithmic schemes to compute CHP: subgradient and Kelley's algorithm • Level stabilization • Adaptation of the basic algorithm to CHP specificities
Numerical Results	• Size of the EU day-ahead market problem • Comparison of the Level Method and the Dantzig Wolfe algorithm on EU instances
Discussion	• **Comparison of the different pricing schemes results** • **Properties of the pricing schemes**
Conclusion	• Perspective for future research

Comparison with other pricing schemes
Uplifts (LOC) magnitude

TABLE IV
RESULTS OF THE LEVEL METHOD AND THE DANTZIG-WOLFE ALGORITHM ON THE BE TEST CASE (AVERAGE OVER 6 INSTANCES)

	horizon	12	24	48	96
	Dispatch Cost [€]	2,759,706	4,956,513	11,328,351	24,097,373
(IP pricing)	O'Neill Uplifts [€]	377,528	146,167	281,649	2,617,852
(ELMP pricing)	Primal Meth. Uplifts [€]	50,871	64,323	83,172	98,391
(CHP pricing)	CHP Uplifts [€]	7,237	11,905	21,745	31,403
	Level iter	19	26	32	44
	Level av. time/iter[a][s]	0.5 (0.05)	0.8 (0.1)	2.0 (0.4)	5.8 (1.6)
	Level total run time [s]	10	21	65	255
	D-W iter	19	40	77	236
	D-W av. time/iter[a][s]	0.4 (0.02)	0.7 (0.1)	1.9 (0.3)	6.9 (2.1)
	D-W total run time [s]	7	27	146	1622

[a] (·) denotes the average time per iterate for solving the "master programs" (i.e. master plus projection in the case of the Level Method).

4

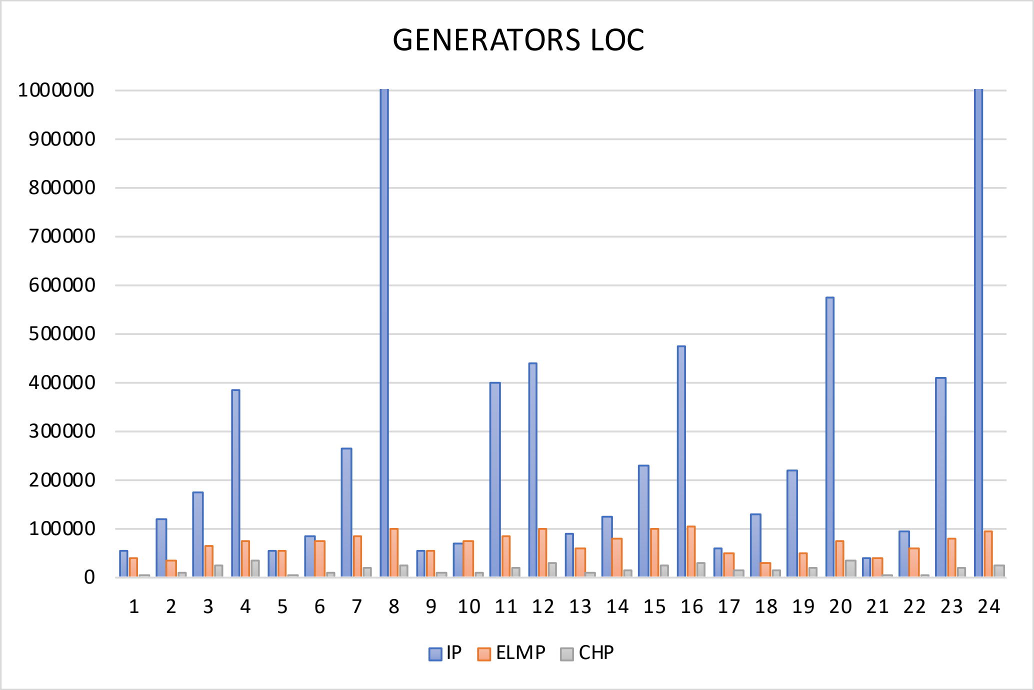
Computing the exact Convex Hull Prices is worth compared to ELMP (the "scalable" approximation)

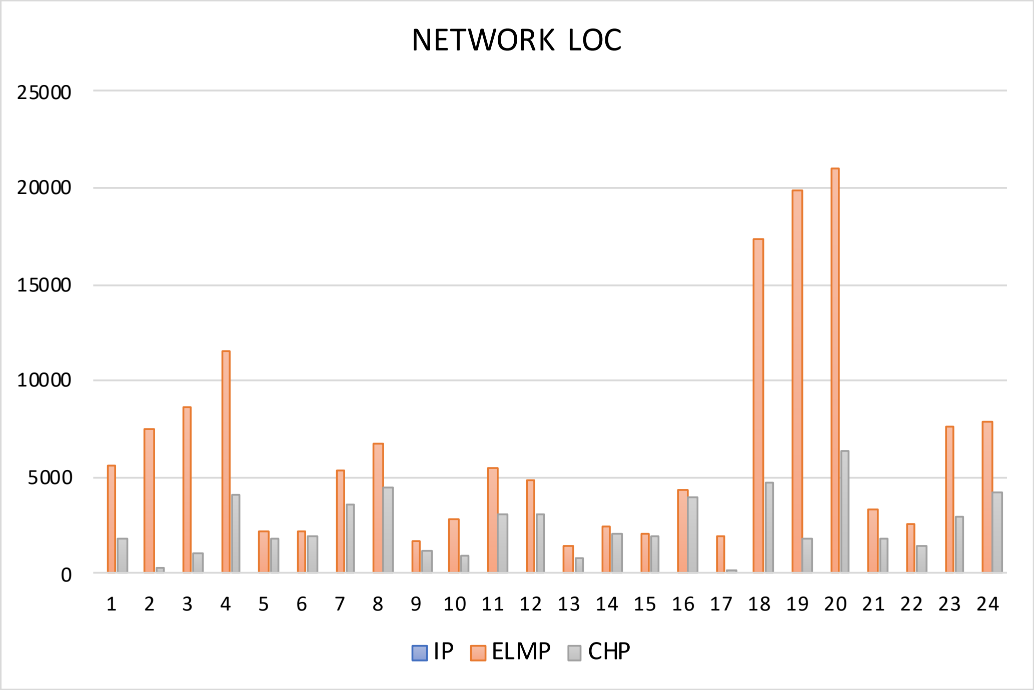
Furthermore, CHP has some more advantages and interesting properties... (next slides)

Disclaimer: I focus on the side payments understood as lost opportunity costs (**LOC**), but there are other metrics, and this is controversial...

Comparison with other pricing schemes

LOC split between network and suppliers





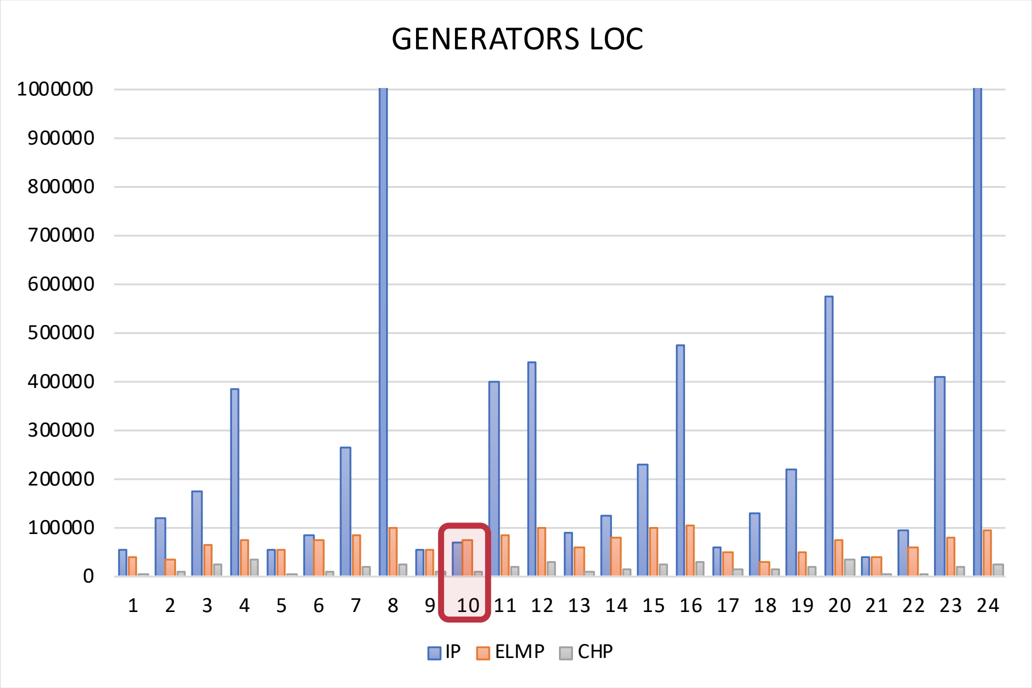
Property 1 (IP LOC Network). *A convex network model faces a zero Network LOC under IP pricing.*

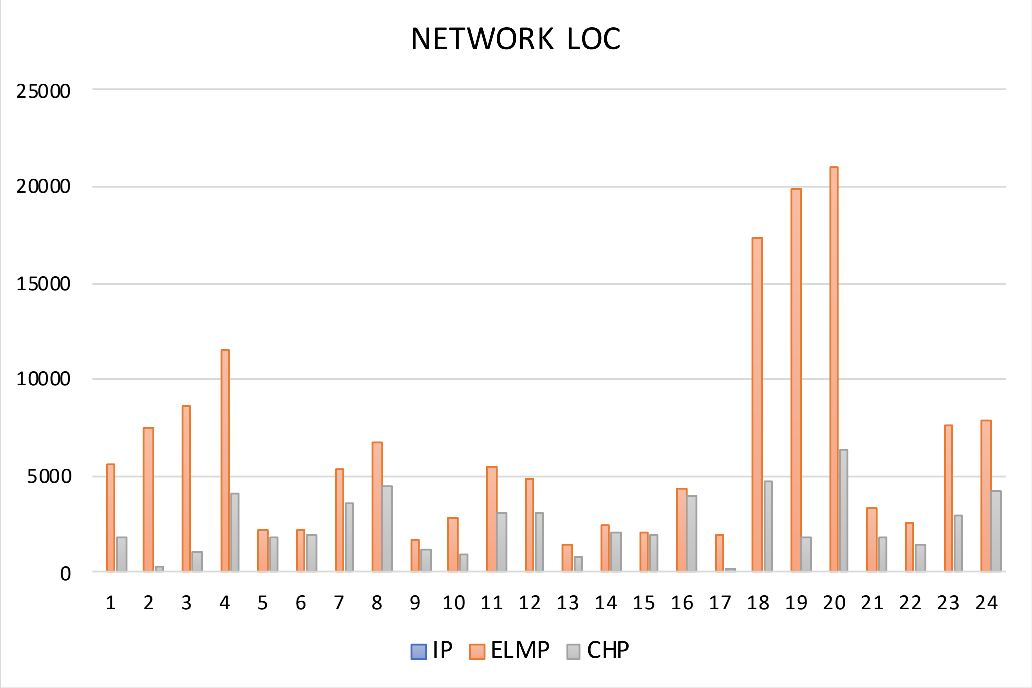
Property 2 (IP LOC Generators). *A convex generator faces a zero generator LOC under IP pricing.*

Property 3 (CHP Convex Uplifts). *Under CHP, both convex generators and convex network can face a non-zero LOC (generator LOC or Network LOC).*

Comparison with other pricing schemes

IP vs ELMP uplifts (LOC)





Property 4 (ELMP vs IP Uplifts). *Given a feasible primal solution, ELMP does not guarantee a lower LOC than IP pricing.*

Comparison with other pricing schemes

Primal solution robustness

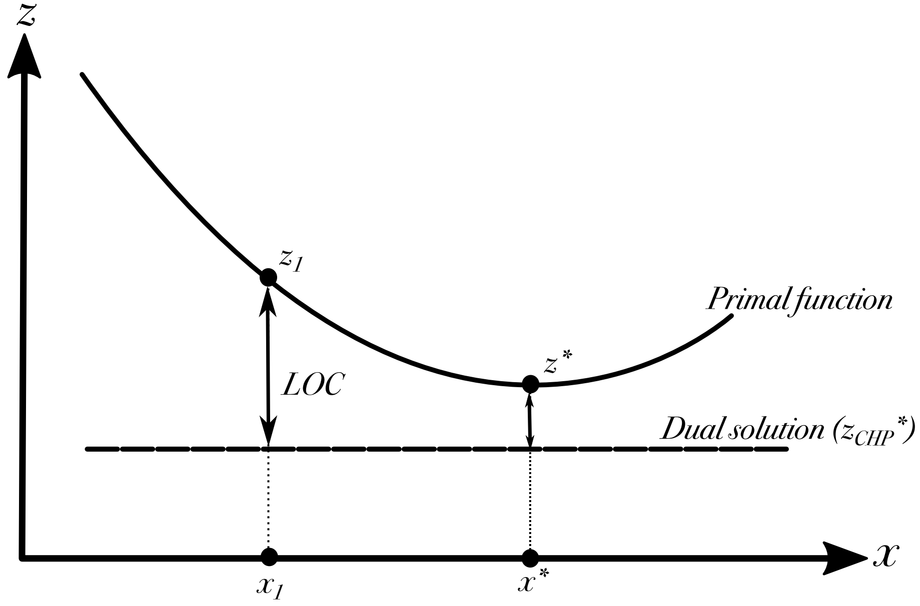
(Opt. gap)	0.1%	0.09%	0.08%	0.07%	0.06%	0.05%	0.04%	0.03%	0.02%	0.01%
Primal Cost	5213357	5212947	5212121	5212121	5211690	5211057	5210885	5210743	5210685	5210685
IP LOC	115043	101212	194521	194521	129455	119929	119579	119360	119351	119351
ELMP LOC	43346	42937	42111	42111	41680	41047	40875	40733	40675	40675
CHP LOC	12742	12332	11506	11506	11075	10443	10271	10128	10070	10070
Δ Primal Cost		409	826	0	431	633	172	142	58	0
Δ IP LOC		13831	-93309	0	65066	9525	350	219	9	0
Δ ELMP LOC		409	826	0	431	633	172	142	58	0
Δ CHP LOC		409	826	0	431	633	172	142	58	0

Property 5 (IP LOC Primal Sensitiveness). *Under IP Pricing, the relationship between the* LOC *and the optimality gap of the primal solution is not monotone: the* LOC *does not necessarily grow with the optimality gap of the primal solution.*

Property 6 (CHP LOC Primal Sensitiveness). *Under Convex Hull Pricing, the total* LOC *grows monotonically with the optimality gap of the primal solution:*

$$|LOC(\pi_{CHP}, [p,u,c]_1) - LOC(\pi_{CHP}, [p,u,c]_1)| = |z_1^* - z_2^*|$$

where $[p,u,c]_1$ *and* $[p,u,c]_2$ *denote two primal feasible solutions and* $LOC(\pi, [p,u,c])$ *denotes the lost opportunity costs associated to price* π *and primal solution* $[p,u,c]$.



Plan

Pricing Schemes

- Mapping of the different pricing schemes
- Main concepts related to Convex Hull Pricing (CHP)

CHP & the Level Method

- Algorithmic schemes to compute CHP: subgradient and Kelley's algorithm
- Level stabilization
- Adaptation of the basic algorithm to CHP specificities

Numerical Results

- Size of the EU day-ahead market problem
- Comparison of the Level Method and the Dantzig Wolfe algorithm on EU instances

Discussion

- Comparison of the different pricing schemes results
- Properties of the pricing schemes

Conclusion

- Perspective for future research

Conclusion

- The Level Method exhibits **favorable empirical performances** to solve Convex Hull Pricing problem

- It is capable to **compute Convex Hull Prices on large instances** including network and a horizon of 96 periods (which anticipates future EU DA market evolution)

- Further promising paths of research:
  1. **Empirical research**: Expanding tests on realistic instances of Euphemia + Examining the **effects of non-uniform pricing on enhancing welfare** in the EU day-ahead market
  2. **Theoretical research**: Further understand the properties of the different pricing schemes

Thank you!

Nicolas Stevens
nicolas.stevens@uclouvain.be

CORE | Louvain Institute of Data Analysis and Modeling in economics and statistics

UCLouvain

References

- M. Madani, C. Ruiz, S. Siddiqui, and M. Van Vyve, “Convex hull, ip and european electricity pricing in a european power exchanges setting with efficient computation of convex hull prices,” 2018, arXiv:1804.00048.
- W. W. Hogan and B. J. Ring, “On minimum-uplift pricing for electricity markets”, *Electricity Policy Group*, pp. 1–30, 2003.
- P. R. Gribik, W. W. Hogan, S. L. Pope *et al.*, “Market-clearing electricity prices and energy uplift”, *Cambridge, MA*, 2007.
- P. Andrianesis, D. J. Bertsimas, M. Caramanis, and W. Hogan, “Computation of convex hull prices in electricity markets with non-convexities using dantzig-wolfe decomposition,” IEEE Trans. Power Syst., to be published, doi: 10.1109/TPWRS.2021.3122000.
- G. Wang, U. V. Shanbhag, T. Zheng, E. Litvinov, and S. Meyn, “An extreme-point subdifferential method for convex hull pricing in energy and reserve markets–Part I: Algorithm structure,” IEEE Trans. Power Syst., vol. 28, no. 3, pp. 2111–2120, Aug. 2013.
- G. Wang, U. V. Shanbhag, T. Zheng, E. Litvinov, and S. Meyn, “An extreme-point subdifferential method for convex hull pricing in energy and reserve markets–Part II: Convergence analysis and numerical per- formance,” IEEE Trans. Power Syst., vol. 28, no. 3, pp. 2121–2127, Aug. 2013.
- B. Hua and R. Baldick, “A convex primal formulation for convex hull pric- ing,” IEEE Trans. Power Syst., vol. 32, no. 5, pp. 3814–3823, Sep. 2017.
- Y. Yu, Y. Guan, and Y. Chen, “An extended integral unit commitment formulation and an iterative algorithm for convex hull pricing,” IEEE Trans. Power Syst., vol. 35, no. 6, pp. 4335–4346, Nov. 2020.
- C. Álvarez, F. Mancilla-David, P. Escalona, and A. Angulo, “A Bienstock- Zuckerberg-based algorithm for solving a network-flow formulation of the convex hull pricing problem,” IEEE Trans. Power Syst., vol. 35, no. 3, pp. 2108–2119, May 2020.
- PJM Interconnection, “Proposed enhancements to energy price formation”, 2017.
- R. P. O’Neill, P. M. Sotkiewicz, B. F. Hobbs, M. H. Rothkopf, and W. R. Stewart Jr, “Efficient market-clearing prices in markets with nonconvexities”, *European journal of operational research*, vol. 164, no. 1, pp. 269–285, 2005.
- Carlos Ruiz, Antonio J Conejo, and Steven A Gabriel. Pricing non-convexities in an electricity pool. *Power Systems, IEEE Transactions on*, 27(3):1334–1342, 2012.
- Y. Nesterov, *Introductory lectures on convex optimization: a basic course*. Springer, 2004.
- Nemo Committee, “CACM annual report 2019”, July 2020. [Online]. Available: http://www.nemo- committee.eu/assets/files/cacm- annual-report-2019.pdf
- E. Krall, M. Higgins, and R. P. ONeill, “Rto unit commitment test system”, *Federal Energy Regulatory Commission*, vol. 98, 2012.
- I. Aravena and A. Papavasiliou, “Renewable energy integration in zonal markets”, *IEEE Transactions on Power Systems*, vol. 32, no. 2, pp. 1334– 1349, 2016