Application of the Level Method for Computing Locational Convex Hull Prices

Nicolas Stevens, *Student Member, IEEE* and Anthony Papavasiliou, *Senior Member, IEEE*

***Abstract*—Convex hull pricing is a well-documented method for coping with the non-existence of uniform clearing prices in electricity markets with non-convex costs and constraints. We revisit primal and dual methods for computing convex hull prices, and discuss the positioning of existing approximation methods in this taxonomy. We propose a dual decomposition algorithm known as the Level Method and we adapt the basic algorithm to the specificities of convex hull pricing. We benchmark its performance against a column generation algorithm that has recently been proposed in the literature. We provide empirical evidence about the favorable performance of our algorithm on large test instances based on PJM and Central Europe.**

***Index Terms*—Convex hull pricing, Non-uniform pricing, Level method, Bundle methods.**

I. Introduction

A. Pricing non-convexities

THE classical analysis of an economic dispatch problem, together with its dual, provides a fundamental argument for *uniform pricing* in electricity markets [1] — an optimal dispatch can be supported by a set of competitive equilibrium prices. In other words, even if a central authority cannot effectively control the dispatch of the assets itself, it can provide prices that align the behaviour of selfish profit maximizing agents with social welfare maximization. However, as the argument assumes convexity of the dispatch problem, a *fundamental* challenge for market efficiency is *non-convexity*, as the latter implies that it is not guaranteed that a competitive market equilibrium exists.

Non-convexities are at the heart of power system operations [2], in terms of both the *network model* as well as in the *market orders*: (i) they are present in the alternating current (AC) power flow equations which characterize the physics of the grid and (ii) in the mixed integer programming (MIP) constraints that describe the market offers. As the day-ahead (DA) markets in Europe and in the US rely on a linear direct current (DC) power flow model of the grid, point (i) is not encountered in these markets[1]. On the other hand, point (ii) is a reality in both US markets that rely on solving a unit commitment (UC) problem, as well as in the EU market which includes integer market orders — the so-called "block orders". Throughout this paper, we neglect (i) and rather focus on (ii).

The inexistence of equilibrium prices in electricity auctions has triggered a long-lasting debate on the choice of an appropriate pricing scheme in the presence of non-convexities. Convex hull pricing (CHP) has arisen as one promising alternative: while being so far mainly debated in the US, it has also recently emerged as a possible option for the EU market [4]. A practical concern of CHP is that its computation can be challenging (e.g. see Issue 7 in [5]). Our paper aims at addressing these computational challenges by putting forward a workable algorithm (the *Level Method*) for realistic instances subject to *network constraints*. In the remainder of this section, we sketch the main concepts related to CHP and we discuss the context of non-uniform pricing discussions in the EU. Insofar as the EU market is concerned, we discuss institutional aspects as well as computational issues, which motivate our choice of test instances.

B. Non-uniform pricing schemes

The most widely debated "non-uniform pricing schemes" in the literature include integer programming (IP) pricing proposed by O'Neill [6], convex hull pricing proposed by Gribik and Hogan [7], [8], and "extended LMP" pricing which has been applied early on in the PJM market [9], [10]. They all amount to a *convex relaxation* of the market clearing problem. These strategies consist of combining a *uniform electricity price* with discriminatory payments, called *uplift payments*, which aim at restoring the incentives of market participants for following the market matches. In this framework, the overall market clearing procedure can be described in three steps, which are also followed by our simulations:

1) Solve the *primal problem*, in order to establish the dispatch and commitment instructions ;
2) Solve a *pricing problem* in order to compute uniform electricity prices ;
3) Solve the independent *profit maximization problems* of all market agents (generators and the network operator) in order to establish *uplift payments*.

Regarding step 3, it is worth noting that uplift payments are often categorized as follows in the literature:

- *Potential Congestion Revenue Shortfall* are uplifts associated with the network *congestion revenue* [3].
- *Generator side-payments* are defined as the difference between the maximum profit achieved by self-scheduling given the market prices and the as-cleared profit.

The pricing strategy proposed by O'Neill is a common choice in non-convex settings. We also use it as a benchmark for our simulations. However, it does not attempt to minimize uplift payments, and can therefore possibly lead to high side payments. Uplifts are undesirable, as they can distort the

[1]Note, nevertheless, that the debate on TSO/ISO-DSO integration has recently motivated the consideration of more advanced models for the representation of network constraints in market-clearing platforms [3].

incentives of bidders or create revenue adequacy problems for the market operator that needs to finance them [11].

These concerns motivate *Convex Hull Pricing* (CHP), the main property of which is to *minimize uplifts*. Because it is computationally challenging, PJM (and other US ISOs) has recently implemented a new pricing scheme, referred to as "extended LMP" which is more tractable computationally than CHP. For certain forms of simple market orders, it can also be shown to be a reasonable *approximation* of CHP [9]. We expand on how it relates to the computation of CHP in section II.

C. Uniform pricing in the EU

The EU market landscape presents a number of major institutional differences compared to US markets [12]. One such notable difference is that day-ahead energy auctions (and, in the future, also day-ahead reserve auctions) are operated by *private Nominated Electricity Market Operators* (NEMOs) while, in the US, it is the *public* ISO that operates both the market and the network. One implication of this difference relates to the ability of the market operator to socialize uplift payments. This difference may, in part, justify the currently employed "uniform" pricing scheme that is adopted in Europe, as implemented in Euphemia, the algorithm that clears the pan-European day-ahead auction [13].

In Euphemia parlance, the aforementioned generator side payments can be related to: (i) *paradoxically accepted blocks* (PAB) — cleared bids actually facing losses, i.e. requiring *make-whole payments* (as defined in [5]) — and (ii) *paradoxically rejected blocks* (PRB) — a rejected bid that would have been profitable, i.e. facing a *lost opportunity cost*. The EU day-ahead market "avoids" uplift payments by (i) constraining the problem by not allowing the acceptance of PABs while (ii) allowing PRBs, but not paying their lost opportunity costs. Ultimately, it does not effectively reduce the uplifts to zero, but it guarantees zero *make-whole payments*, while increasing the total *lost opportunity cost* and not paying it. Consequently, this pricing scheme only outputs *uniform prices* while it does not provide the market participants with any discriminatory payments. This justifies why, in EU NEMO parlance, it is referred to as *uniform*, in contrast to the three *non-uniform* pricing schemes that are discussed previously.

This *uniform* pricing scheme involves "primal-dual" constraints that implicate dispatch and price decisions in a single market clearing model. The solution implemented in Euphemia amounts to an iterative algorithm that matches market orders while aiming to find a feasible price (without PAB). If this is not possible, the algorithm generates a cut in the primal model and repeats the process. In contrast to the *non-uniform* pricing schemes that work in three steps (dispatch, price, uplifts), the EU *uniform* pricing scheme works as a single — but iterative — step, and couples dispatch and price problems together.

This makes the problem that Euphemia is called to solve (a mixed integer quadratic program subject to complementarity constraints) computationally challenging. Moreover, the approach deteriorates market welfare, since welfare-enhancing orders can be discarded if no market clearing price can be found to support the aforementioned clearing rule. For these reasons, non-uniform pricing schemes, and in particular *convex hull pricing*, have recently received consideration by the European NEMOs as a possible option for the European DA energy auction [4]. Considering the aforementioned institutional EU structure, as well as the algorithm implemented in Euphemia, this would constitute a disruptive market design evolution.

Computationally speaking, implementing CHP in Europe comes with three paramount requirements [4], [13]:

- 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 in the near future (a horizon of 96 periods).

Forty bidding zones for ninety-six periods implies a 3,840-dimensional price space. These requirements motivate the considered use cases in section IV.

D. Contributions and Structure of the paper

The contribution of the paper is twofold:

1) We propose the *Level Method* [14] for computing CHP and adapting it to the specificities of our problem. We specifically adapt the algorithm in order to exploit the convexity of the *network* model. We further introduce a "multi-cut" variant of the Level Method in order to leverage the separability of the sub-problems. Note that two types of approaches have been envisioned in the literature for solving CHP: *dual approaches* and the *primal approaches* (we define these in section II). The *Level Method* belongs to the former. Primal approaches, and their drawbacks which motivate our choice for a dual approach, are presented in section II. The review of alternative (tested) dual approaches comes in section III and motivates our choice of the Level Method.
2) We efficiently solve CHP, using the Level Method, for large instances *including a network* and a horizon of 96 periods, which anticipates the evolution of the EU market. We conduct a critical comparison of our approach against both *primal* and *dual decomposition* approaches. In particular, we compare it to a notable recent publication by [15], which proposes a Dantzig-Wolfe (D-W) algorithm for computing CHP. The D-W algorithm exhibits favorable performance on a test case without a network and with 24 time periods, as considered in [15]. Given our preoccupation with a market clearing model at the scale of the EU market, the question becomes how the method scales when moving from a 24-dimensional to a 3,840-dimensional price space. When increasing the dimension, the Level Method is empirically shown to attain favorable performance relative to [15].

Our paper is inspired by an older unpublished work [16], and is further motivated by [15]. We describe the mathematical formulation of CHP in section II. We then introduce the Level Method in section III. In section IV, we test the algorithm on multiple large instances and compare the results with D-W. Section V concludes and discusses areas of further research.

II. Mathematical formulation

A. Convex hull pricing program

We define the dispatch problem subject to *network constraints* as follows:

$$\min_{c,p,u,f} \sum_{g\in\mathcal{G}} c_g \tag{1a}$$

$$(\pi_t^i) \quad \sum_{g\in\mathcal{G}_i} p_{g,t} - D_t^i = \sum_{\substack{l\in\\ from(i)}} f_{l,t} - \sum_{\substack{l\in\\ to(i)}} f_{l,t} \quad \forall i,t \tag{1b}$$

$$(c_g, p_{g,t}, u_{g,t}) \in \mathcal{X}_g \quad \forall g\in\mathcal{G} \tag{1c}$$

$$f\in\mathcal{F} \tag{1d}$$

Here, $\mathcal{G}_i$ denotes the set of generators (or market offers) at node i. Each offer is modelled with a total cost c_g, a power output $p_{g,t}$ at time t and a set of non-convex constraints $\mathcal{X}_g$. The generic variables u_g stand for all the binary variables encountered in the generator model. The demand at time t and node i, D_t^i, appears in the market clearing (MC) constraints (1b). Regarding the network, $f_{l,t}$ stands for the flow on line l, while $from(i)$ is the set of lines originating from i and $to(i)$ the ones directed towards i. No assumption is made on the network constraints $\mathcal{F}$, except that it is a *convex* set.

Each generator g is assumed to be a selfish agent that maximizes profit, i.e. solves the following program:

$$\max_{c,p,u} \sum_t p_{g,t}\pi_t^{i(g)} - c_g \tag{2a}$$

$$(c_g, p_{g,t}, u_{g,t}) \in \mathcal{X}_g \tag{2b}$$

Here, $i(g)$ stands for the node of generator g, while $\pi_t^{i(g)}$ represents the market price of node $i(g)$ at time t.

A fundamental result [7], [8] on CHP establishes that minimizing uplifts amounts to solving the following problem:

$$\pi^{CHP} = \arg\max_{\pi} L(\pi) \tag{3}$$

Here, $L(\pi)$ denotes the *Lagrangian dual function*, obtained by relaxing constraints (1b) of problem (1):

$$L(\pi) = \sum_{i,t} \pi_t^i D_t^i \tag{4a}$$

$$-\sum_{g\in\mathcal{G}} \max_{(c,p,u)_g\in\mathcal{X}_g} \left\{ \sum_t p_{g,t}\pi_t^{i(g)} - c_g \right\} \tag{4b}$$

$$+\min_{f\in\mathcal{F}} \left\{ \sum_{i,t} \pi_t^i \left(\sum_{\substack{l\in\\ from(i)}} f_{l,t} - \sum_{l\in to(i)} f_{l,t} \right) \right\} \tag{4c}$$

We recognize in (4b) the profit maximization problems (2) of the generators. As established in [8], using the optimal primal dispatch solution of (1) and injecting it into (4) clarifies why the previous Lagrangian problem does indeed minimize the uplifts. As also pointed out in the literature, the definition (3) of CHP also indicates that the uplifts can be interpreted as the *duality gap* between (1) and (4).

B. US versus EU models

In addition to institutional differences between US and EU markets, another major difference relates to the definition of market products. The US markets follow a *unit-bidding* model, where each unit is represented explicitly in the market, along with its technical characteristics. On the other hand, the EU day-ahead market follows a *portfolio-bidding* model (which cannot be subsumed in the unit commitment formulation), where each agent submits multiple generic market orders that represent the portfolio of its assets in an aggregated way. These market orders [13] include convex hourly orders — *stepwise* and *interpolated curves* — as well as non-convex orders[2] — mainly the family of *block orders*. The latter is a financial order spanning over multiple periods and involving a *binary acceptance* variable.

Model (1) remains general regarding the bid (generator) constraints (1c), which are simply represented as the non-convex set $\mathcal{X}_g$. This implies that the approach outlined in this paper can accommodate all the flavours of unit commitment models as well as the EU-like auctions. This exceeds what a "primal CHP approach" can model.

Finally, model (1) considers a general (but *convex*) set of network constraints $\mathcal{F}$. Our approach can in fact accommodate any convex representation of the network. In both the US and EU market, $\mathcal{F}$ would amount to a set of *linear* constraints, the main difference being that certain US markets are *nodal* (larger number of nodes) while the EU market is *zonal* (roughly one zone per country). We remark in section III on the specific treatment of the network in our proposed Level Method.

C. Primal and dual approaches for computing CHP

In this section, (i) we locate the Level Method in the perspective of the landscape of all the alternative of approaches for solving CHP and (ii) we motivate the choice of a dual approach in the light of the limitations of the primal approaches.

As noted in section I, there are two main approaches envisioned for computing convex hull prices — i.e. solving problem (3): (i) the *Lagrangian dual approaches*, which directly attempt to maximize function $L(\pi)$ using an iterative algorithm, and (ii) the *primal approaches*, understood as methods that seek to describe the CH of the non-relaxed constraints (1c)-(1d) by developing tight formulations. Figure 1 outlines the landscape of approaches for computing convex hull prices. The top problem (A) corresponds to the dispatch problem (1). Below, on the left, we find primal relaxations of (A) while, on the right, we find Lagrangian relaxations of (A) — Lagrangians are indeed a widely employed method for deriving convex relaxations of non-convex programs [17]. The problem (Γ) corresponds to the CHP definition (3), which can be solved by dual decomposition approaches such as the Level Method. Problem (Γ) maps to its primal equivalent in (C). The underlying idea of the primal formulation is that computing the CHP as the Lagrangian multipliers of (3) is

[2]Note that other non-convex (and less standard) products in Euphemia such as the Italian unique national price (PUN) or complex orders [13], are not directly compatible with CHP, because they implicate primal and dual variables in their definition.

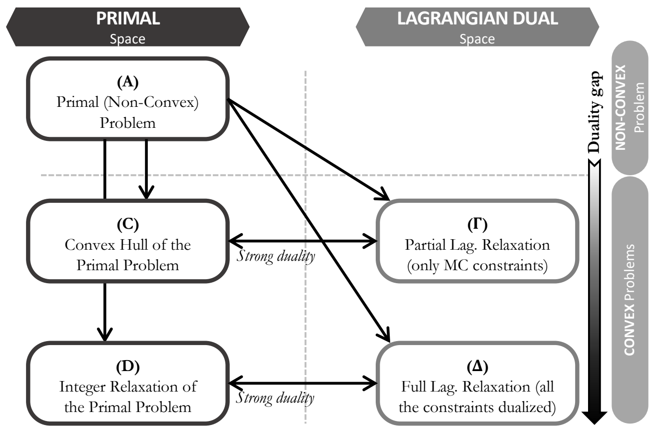

Fig. 1. Landscape of problems for computing / estimating convex hull prices.

equivalent to computing the dual variable π associated to the market clearing constraint (1b) in the primal problem (1), if the latter is expressed on the convex hull of its domain — i.e. $\mathrm{conv}(\mathcal{X}_g)\ \forall g \in \mathcal{G}$ (see [18],[17] for the general result in Lagrangian relaxation theory or [19] for the specific result related to CHP).

Although (C) is the tightest primal relaxation of (A), there exist looser relaxations, such as (D), which amounts to relaxing the integrality constraints $u_{g,t} \in \{0,1\}$ to $u_{g,t} \in [0,1]$. This corresponds to *PJM pricing*, discussed in the introduction. PJM pricing can be interpreted as a computationally efficient *approximation* of CHP [9]. In certain cases, relaxing the integrality constraints in $\mathcal{X}_g$ may provide $\mathrm{conv}(\mathcal{X}_g)$. In this case, problems (C) and (D) are equivalent and PJM pricing effectively corresponds to convex hull pricing. The fact that relaxation (D) is looser than (C) implies that the *duality gap* between (A)–(D) will be greater than or equal to the one between (A)–(C).

Interestingly (and to the best of our knowledge, unnoticed in the literature), one can also relate the primal relaxed problem (D) to its Lagrangian dual counterpart (Δ). While CHP is solving the *partial* Lagrangian dual relaxation (Γ), PJM pricing corresponds to solving the *full* (looser) Lagrangian dual relaxation (Δ), where all the constraints — and not only the market clearing constraints — are dualized[3].

Regarding the primal CHP problem (C), a way to approach it is to develop a *tight* — but *custom* — formulation, specific to the targeted problem (A). Recent researches have embraced this idea: [19] proposes an explicit formulation for the primal model of CHP for classical UC constraints. Madani [24] analyses primal CHP formulations for the constraints of the European day-ahead market clearing model[4]. More recent research further elaborates on the idea, developing tight (custom) formulations for MISO [25] or proposing a network flow model of unit commitment, in order to compute CHP for a broader set of constraints [26]. One value of the primal CHP approaches is to establish the link between *CHP theory* and the literature dedicated to *tight formulations of UC polytopes* such as [27]–[35] (see chapter 2 in [16]). Similarly, when including a non-convex network model, the primal CHP approach [3] also establishes the connection between *CHP theory* and *SDP/SOCP* relaxations of AC power flow [2].

Nevertheless, as also voiced in [15], there are certain constraints for which the convex hull is not tractable in the sense that it may not be possible to characterize the convex hull with a scalable number of constraints. This already holds for simple ramp constraints [19]. This is also acknowledged by [26], where the authors do not account for them in their network flow model. Instead, [25] needs to combine the proposed tight formulation with an iterative algorithm in order to account for the ramp constraints in a *scalable* way. It goes without saying that these modelling limitations also hold for more advanced constraints such as multimode CCGT units, detailed battery models, and so on. Thus, since the pricing mechanism becomes dependent on the quality of the primal formulation, the primal approach can be ruined by adding a new constraint — which is particularly concerning, since electricity market models are constantly subject to changes (e.g. triggered by regulatory requirements such as article 40 of EGBL guidelines). These modelling limitations imply that, if the representation of the convex hull is not tractable, the primal approaches are irremediably left with an *approximation* of convex hull prices, such as the PJM pricing model (D). This is illustrated in our numerical results of section IV, where a primal method benchmark [19] is included. This motivates our choice for a dual approach.

III. The Level Method

A. Review of existing algorithms

The appropriate algorithmic scheme for solving (3) is related to the type of function $L(\pi)$.

Property 1 (Concave)**.** *Function $L(\pi)$ is concave in π.*

Property 2 (Non-smooth)**.** *Function $L(\pi)$ is a non-smooth (piecewise linear) function, i.e. each facet can be seen as corresponding to a set of binary (commitment) decisions u_g.*

Property 3 (First-order oracle)**.** *A first-order oracle is available, i.e. given a price π, both the function value $L(\pi)$ as well as its supergradient $s \in \hat{\partial}L(\pi)$ can be evaluated.*

Property 4 (Supergradient)**.** *Let (c^*, p^*, u^*, f^*) be the optimal reactions to π (solving respectively* (4b) *and* (4c)*). Then*

$$s = D_t^i - \sum_{g\in\mathcal{G}} p_{g,t}^* + \sum_{l\in from(i)} f_{l,t}^* - \sum_{l\in to(i)} f_{l,t}^*$$

is a supergradient of L in π; i.e. $s \in \hat{\partial}L(\pi)$.

[3]Taylor [2], which inspired Fig. 1, proposes an interesting interpretation of CHP by relating it to the semi-definite programming (SDP) relaxation of problem (1). The proposition is motivated by the well-known SDP relaxation of a non-convex quadratically constrained program (QCP) [20]–[22] and the fact that a MIP can be expressed as a QCP. However, the above taxonomy reveals an innacuracy in the reasonning: it mixes (Δ) and (Γ), as it omits the fact that CHP relies on a partial (and not complete) Lagrangian relaxation, where only the market clearing constraints are relaxed (i.e. dualizing fewer constraints can only improve the duality gap [23]).

[4]Note however that [24] focuses on a subset of the market constraints, ignoring e.g. linked blocks and exclusive groups [13].

Each call to the oracle implies solving MIP profit maximization programs (2) for each generator as well as for the network program (4c) — these are thus *slave problems*. We propose later a special treatment of the network and leverage the separability of the profit maximization problems in order to substantially improve the formulation. Any algorithm tackling this problem would work in three steps:

1) Given a price π_k, evaluate $L(\pi_k)$ and $\hat{\partial} L(\pi_k)$;
2) Given this information, generate a new price π_{k+1} ;
3) If the stopping criterion is met, stop. Else, go to step 1.

The main difference between dual decomposition algorithms is in the way that they construct the sequence of iterates $\{\pi_k\}_{k=0}^{\infty}$: (i) some algorithms simply update the prices based on the latest supergradient information — they are *memoryless* —; (ii) while other algorithms will keep memory of the sequence of iterates. We briefly summarize three approaches, which were tested (and compared to the Level Method) by the authors in previous work [16].

A well-known scheme belonging to category (i) is the *subgradient scheme*. Perhaps surprisingly, it is proven to be optimal for general convex non-smooth optimization with arbitrarily high dimension [14]. However, when dealing with problems of "moderate" dimension such as the one presented in our context, there exists more optimistic alternatives.

Indeed, the subgradient scheme for piecewise-linear functions, such as our problem (3), tends to oscillate between the facets of the Lagrangian dual function, around an edge. Therefore, one idea is to "catch the edge" and follow it until the optimum, instead of oscillating from one facet to another, as the subgradient method does. This intuitive reasoning leads to the *Extreme-Point Subdifferential* (EPSD) algorithm, which is specifically applied in [36], [37] to the CHP problem. However, our experiments in [16] reveal that each iterate of the algorithm is costly, as it requires not only to solve the problems (2) for each generator to optimality, but to enumerate *all* the optimal solutions.

Unlike these two *memoryless* schemes, the *Analytic Center Cutting Plane Method* (ACCPM, see [14], [38] for the theory and [37], [39] for its application to CHPs) is based on the principle of iteratively reducing the search domain: the price domain is initially limited to a box and, at each iterate, the supergradient is used for generating a *cut*, which shrinks the search domain. The next testing point is chosen as the analytical center of the updated domain.

Our original investigation of these alternative *dual approaches* (subgradient, EPSD and ACCPM) in [16] concluded that none of them were competitive with the *Level Method* for computing CHPs.

B. Kelley's approach

The Kelley algorithm [14] forms the basis for the proposed Level Method. It is based on the idea of iteratively constructing a *model* (upper approximation) of the Lagrangian function $L(\pi)$, using its supergradients.

Definition 1 (model function)**.** *Let Q be the initial domain of our problem (i.e. a* box *limiting the prices, which can be economically interpreted as* price caps*) and let $\{\pi_k\}_{k=0}^{\infty}$ be a sequence in Q. Let s_k be the supergradient at iterate π_k. Then*

$$\hat{L}(\pi, k) = \min_{j=0..k} \{\langle s_j, \pi - \pi_j\rangle + L(\pi_j)\} \tag{5}$$

is a model for the Lagrangian function $L(\pi)$, such that $\hat{L}(\pi, k) \geq L(\pi)$.

In order words, the piecewise linear function $L(\pi)$ is upper-approximated at each iterate by a model function $\hat{L}(\pi, k)$ consisting of supporting hyperplanes. At iteration 0, this is a single hyperplane. Then, as the iterate count k is increasing, the model function $\hat{L}(\pi, k)$ is becoming increasingly accurate.

Definition 2 (master program)**.** *The maximization of the model function yields the master program at iterate k:*

$$\begin{aligned} \max_{\pi \in Q, \theta} \quad & \theta \\ s.t. \quad & \theta \leq \langle s_j, \pi\rangle + b_j \quad \forall j = 0..k \end{aligned} \tag{6}$$

Here, s_j are the "cut coefficients" (as defined in Property 4) and $b_j = L(\pi_j) - \langle s_j, \pi_j\rangle$ are the "cut constants". This is a computationally tractable linear program.

Having the upper-approximation function $\hat{L}(\pi, k)$ at hand, one needs to decide the rule for building the sequence of iterates $\{\pi_k\}_{k=0}^{\infty}$. The more intuitive way to pick the next iterate is:

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

i.e. the solution of the master program (6). This defines Kelley's cutting plane method. One of its benefits is that it explicitly provides an *upper bound* as well as a *lower bound* at each iterate k: a lower bound is defined as $LB_k = \max_{j=0..k} L(\pi_j)$, while an upper bound is $UB_k = \max_{\pi} \hat{L}(\pi, k)$. Note that the sequence of upper bounds $\{UB_j\}_{j=0}^{k}$ is decreasing, as the definition of the model function implies that $\hat{L}(\pi, k+1) \leq \hat{L}(\pi, k)$. The upper and lower bounds can be combined to define the *relative gap*, which is used as a *stopping criterion* for the Kelley (and Level) Method:

$$\frac{UB_k - LB_k}{|UB_k|} \leq \epsilon \tag{8}$$

C. Level stabilization

Kelley's algorithm is *finite*, because each iterate adds a new hyperplane and the number of hyperplanes supporting the function is finite. Nevertheless, despite its simplicity and its good behaviour in low dimension, it tends to be unstable in higher dimension. This is due to the unstable nature of piecewise linear functions: adding a new supporting hyperplane can move the optimum far from the previous point (i.e. to a corner of the box Q). This well-known drawback [40] justifies why multiple *stabilization approaches* have been proposed in the literature, including the *Level Method* [14], [40].

The underlying idea of the Level Method is to update prices more smoothly: instead of using the optimum of the model function as the next iterate, the algorithm chooses π_{k+1} such that it is "better" than π_k (as evaluated by the model function $\hat{L}(\pi_{k+1}, k)$) without being optimal at all costs. We observe in

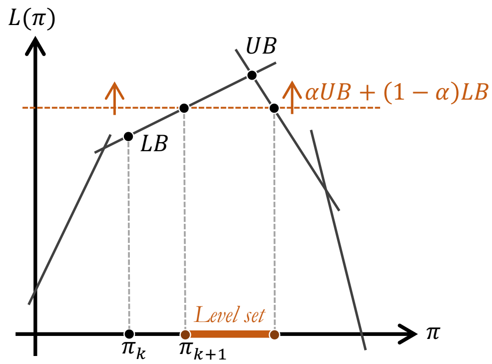

Fig. 2. Illustration of projection on the level set.

section IV that this stabilization has a major influence on the practical performance of the algorithm.

A graphical illustration in 1-D is presented in Fig. 2. The cuts, the LB and the UB are obtained as in Kelley's method, by solving the master program (6). However, unlike in Kelley's method, the next price candidate is selected by solving a projection program.

Definition 3 (projection program)**.** *The iterate π_{k+1} is chosen as the projection of π_k on the "level set" $\hat{L}(\pi, k) \geq \alpha UB_k + (1-\alpha)LB_k$, which amounts to solving:*

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

Here, $\alpha \in [0,1]$ is the projection parameter. This is a computationally tractable quadratic program.

Regarding the calibration of α, $\alpha = 1$ corresponds to the classic Kelley method, while $\alpha = 0$ implies that the iterate simply does not move. We note that a theoretically optimal α exists [14] for *general* convex non-smooth functions, but that a calibration to the *specific* problem can still be meaningful. Our empirical tests on the CHP problem reveal that, for the high-dimensional instances that we are interested in, the approach is largely *insensitive* to the choice of α. This is shown later in Table III, where any value of α between 0.2 and 0.7 exhibits similar performances. Following [16], the value $\alpha = 0.2$ is chosen for all of our experiments in the present work.

Regarding the choice of the box Q, experimental evidences show that the Level Method is not too sensitive to its exact value, despite it impacts the quality of the UB estimate. In all of our experiments, Q is initially set to $\pm 300\$/MWh$ and is then progressively shrunk after 10, 20 and 30 iterates to $\pm 25\$/MWh$ around the latest price candidate. This is justified by an analysis of the volatility of the price iterates, which rapidly reach a price close to the CHP.

D. Refinements of the Level Method in the context of CHP

We now propose adjustments to the basic algorithm which exploit the structure of our problem. We specifically leverage the fact that: (i) the network model is *convex* and (ii) the profit maximization programs of the generators are *separable*.

In our development so far, we have been treating the *convex* network term (4c) identically to the *non-convex* generators, i.e. by solving the network profit maximization given a price π, and generating a supergradient. We illustrate below the treatment of the *convex parts* of the primal program by focusing our discussion on the network. The idea applies identically to *convex* generators (e.g. the convex orders in Euphemia, which are numerous), a *convex* pumped-storage model, etc. (see section 3.6 and appendix A in [16] for a treatment of these cases).

For the sake of illustration, let us assume that the network constraints $\mathcal{F}$ correspond to the *DC (voltage angle) power flow*. Term (4c) then reads as follows:

$$\min_{f,\psi} \sum_{i,t} \pi_t^i \left(\sum_{l \in from(i)} f_{l,t} - \sum_{l \in to(i)} f_{l,t} \right) \tag{10a}$$

$$(\mu_{l,t}) \quad f_{l,t} \leq \overline{F}_l \qquad \forall l, t \tag{10b}$$

$$(\nu_{l,t}) \quad f_{l,t} \geq \underline{F}_l \qquad \forall l, t \tag{10c}$$

$$(\lambda_{l,t}) \quad f_{l,t} = B_l(\psi_{or(l),t} - \psi_{dest(l),t}) \qquad \forall l, t \tag{10d}$$

Here, B_l stands for the susceptance of line l, and $\overline{F}_l$ and $\underline{F}_l$ are its max and min capacity, while $or(l)$ and $dest(l)$ denote the origin and destination nodes of line l. The dual of (10) can be expressed as:

$$\max_{\mu \geq 0, \nu \geq 0, \lambda} \sum_{l,t} \nu_{l,t} \underline{F}_l - \mu_{l,t} \overline{F}_l \tag{11a}$$

$$\pi_t^{or(l)} - \pi_t^{dest(l)} + \mu_{l,t} - \nu_{l,t} + \lambda_{l,t} = 0 \qquad \forall l, t \tag{11b}$$

$$\sum_{l \in to(i)} \lambda_{l,t} B_l - \sum_{l \in from(i)} \lambda_{l,t} B_l = 0 \qquad \forall i, t \tag{11c}$$

Problem (11) can now be injected into (4) as a substitute for (4c), meaning that the network dual variables (μ, ν, λ) would explicitly be variables of the master (and projection) program.

Secondly, the classical Kelley and Level Methods add a *single cut* at each iterate, namely one single cut for all the generators. Nevertheless, the dual function is *separable* with respect to the generators. We therefore propose a *multi-cut Level Method*, whereby we compute one cut (one lower approximation) for each generator profit maximization subproblem. Our experiments reveal that this adaptation can deliver substantial computational benefits. Generating more cuts makes the *model function* more accurate, which enables the algorithm to converge faster. Note that multi-cut versions of other approaches have been applied successfully in different contexts, such as for two-stages stochastic programs [41], [42].

To summarise, after the inclusion of both the network dual and the multi-cut approach, the master program (6) at iterate k becomes:

$$\max_{\substack{\mu \geq 0, \nu \geq 0, \\ \lambda, \pi \in Q, \theta}} \sum_{i,t} \pi_t^i D_t^i + \sum_{l,t} \left(\nu_{l,t} \underline{F}_l - \mu_{l,t} \overline{F}_l \right) - \sum_{g \in \mathcal{G}} \theta_g \tag{12a}$$

$$\theta_g \geq \langle p_{g,\cdot,\cdot}^j, \pi^{i(g)} \rangle - c_g^j \quad \forall g, j = 0..k \tag{12b}$$

$$\pi_t^{or(l)} - \pi_t^{dest(l)} + \mu_{l,t} - \nu_{l,t} + \lambda_{l,t} = 0 \quad \forall l, t \tag{12c}$$

$$\sum_{l \in to(i)} \lambda_{l,t} B_l - \sum_{l \in from(i)} \lambda_{l,t} B_l = 0 \quad \forall i, t \tag{12d}$$

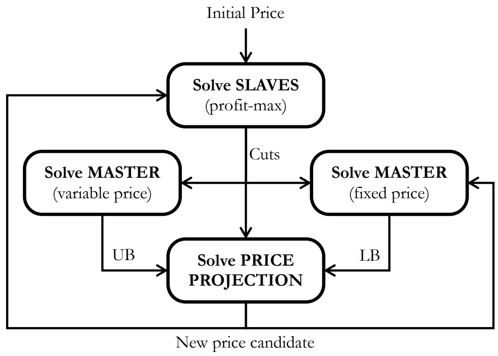

Fig. 3. Our implementation of the Level Method for the computation of CHP.

Here, $\{p_g^j\}_{j=0}^k$, corresponds to the sequence of generator g power output for iterates $j = 0..k$. These parameters are also *cut coefficients* for generator g. On the other hand, $\{c_g^j\}_{j=0}^k$, which corresponds to the sequence of generator g cost for iterates $j = 0..k$, are the *cut constants*. The translation of the projection program (9) is applied as discussed previously.

In the classical Kelley/Level Methods, estimating the lower bound (evaluating (4) at a given π) follows directly from the resolution of the slave subproblems. The inclusion of the network into the master program, as described above, complicates the process. Indeed, the network contribution in the dual function (4c) is not solved explicitly anymore, but now comes in the master objective (12a), together with constraints (12c) and (12d) that should not be violated. Therefore, estimating the value of $L(\pi)$ after having retrieved the cuts from the slaves (for the same π) amounts to solving the master (linear) program (12) with the variables π fixed. The overall procedure is described schematically in Fig. 3. Note that the resolution of the two master programs (with π fixed and variable) can be parallelized.

IV. Simulation Results

This section presents the numerical results of the (multi-cut) Level Method on instances of realistic scale. The Level Method has been benchmarked against other *dual approaches* in earlier work by the authors [16]. It is chosen as the most promising method for computing CHPs among all tested alternatives. In the present section, we therefore focus on its comparison with a recent work [15] which employs a Danzig-Wolfe (D-W) column generation algorithm [43] for iteratively building the convex hull of the dispatch problem, i.e. D-W gradually discovers the corners of the *primal* formulation. As in the case of the Level Method, it can be applied to *any* UC formulation. We use it as a performance benchmark in our analysis, due to its favourable empirical performance. We also include *O'Neill pricing*, discussed in the introduction, as another benchmark in our analysis, as well as *PJM pricing* (discussed in section II-C) as a *primal method* benchmark.

Unlike other computational researches on CHP [15], [37] which are mainly concerned about the number of generators in the problem, we rather focus our investigations on the sensitivity of the algorithms with respect to the dimension of the price space. Indeed, although the number of generators is surely relevant, since the ultimate goal is to compute *prices* by optimizing $L(\pi)$, the price-space dimension is expected to have a significant impact on the performance of any tested method. Therefore, we first present results *without* a network, with a horizon of 24 periods, and then introduce network constraints and extend the time horizon to 96 periods.

TABLE I
Results of the Level Method and the Dantzig-Wolfe algorithm on FERC datasets (average over 11 instances).

Dispatch Cost [$]	29,791,214	Level iter	19
O'Neill Uplifts [$]	652,263	Level av. time/iter[a][s]	8.2 (0.36)
Primal Uplifts [$]	11,400	D-W iter	29
CHP Uplifts [$]	9,746	D-W av. time/iter[a][s]	8.9 (0.34)

[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).

For all our test cases, the comprehensive market procedure for computing the prices and measuring uplifts follows the steps that are described in section I. Concretely, there are three steps: dispatch, price, and uplift computation. The Level Method and D-W differ with respect to the second step. Both approaches have been implemented in Julia (JuMP) and all the tests are run on a personal computer (Intel Core i5, 2.6 GHz with 8 GB of RAM) using Gurobi 9.1.1.

A. FERC (US) test cases

The first test cases in our analysis are based on FERC datasets [34], [44]. The test sets are publicly available, together with the associated UC model, and are also used by [15]. These test cases consist of a detailed UC model. The only adaptations in our work are the removal of reserve and the netting out of renewable supply from the load. The UC model includes, among others, min up and down time constraints, ramp constraints (including start-up and shut-down ramp rates), variable start-up costs which depend on how long a unit has been off, no-load costs, and piecewise linear production costs. The model has *no network*, but gathers > 930 generators. This corresponds to an instance of realistic size, barring for the absence of the network. As in [15], we conduct our analysis on a 24-period horizon with hourly time step.

Table I presents the average results over 11 FERC instances, while Fig. 4 illustrates the convergence behaviour of both approaches on one of the instances. The 11 instances essentially correspond to 11 different load profiles, with slight changes in the production fleet, which varies from 934 to 978 generators. The stopping criterion of the Level Method (equation (8)) is set to 0.01%. The number of iterates reported in Table I for D-W corresponds to the iterations that are required for reaching the same amount of uplifts as the Level Method. Both algorithms are initialized at a uniform price of 20$/MWh.

The results already show the attractive performance of the Level Method, both (i) in terms of *iteration count* and (ii) in terms of *robustness*. Indeed, there is an average improvement of 34% compared to D-W in terms of number of iterates (Table I). It should be noted that this number of iterates is a reasonable measure for comparing the performance of both approaches. Concretely, both methods have to solve the

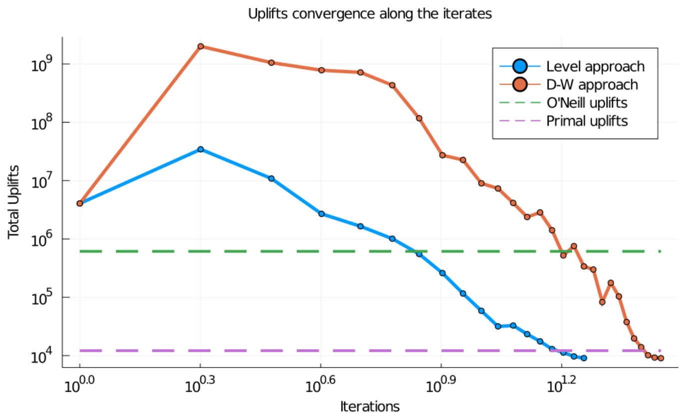

Fig. 4. Convergence of the Level Method and D-W algorithm, measured by the uplifts (*O'Neill pricing* and the primal method are used as benchmark thresholds), on the "FERC 2015-07-01 high wind" instance. Both axes are in logarithmic scale.

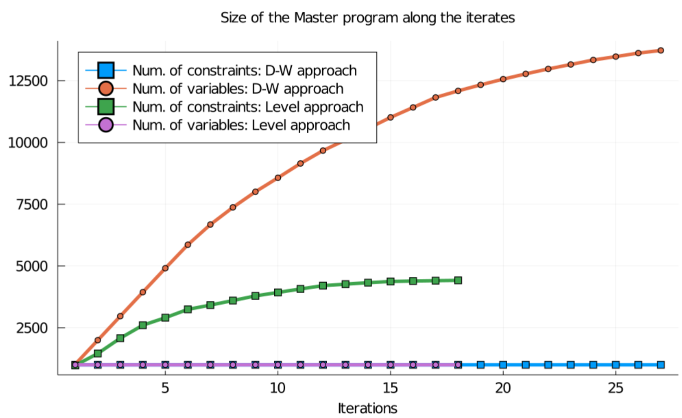

Fig. 5. Size of the Level Method and D-W master programs on the "FERC 2015-07-01 high wind" instance. The Level Method adds cuts, which implies that the number of constraints is growing. On the other hand, D-W adds columns, which implies that the number of variables is growing. The robustness of the Level Method translates in a master program that grows less rapidly than D-W.

same subproblems and mainly differ in the other computations that they are required to perform. Whereas the Level Method has to solve both a linear master and a quadratic projection, D-W is only required to solve the linear (master) extended formulation. On the other hand, the extended formulation solved by D-W is larger than the Level master program, as illustrated in Fig. 5. Overall, this results in a similar run time per iterate, as reported in Table I which shows both the average run time per iterate as well as, between parentheses, the average run time spent in the master programs (master plus projection programs for the Level Method). This implies that the number of iterations (the *analytical complexity*: the number of calls of the oracle to reach a reliability target) is a reasonable measure for comparing performance. It also has the benefit of being less dependent on the specific machine or on the implementation details. Note that, for both approaches, the slave subproblems can be parallelized.

Furthermore, there is a gain in *robustness*: the Level Method exhibits a more stable performance, as observed in Fig. 4. Indeed, Fig. 4 suggests that it does not seem possible to stop the D-W algorithm long before its termination, since uplifts remain high for a large number of iterations (we also refer the reader to Fig. 7 of the next use case, which shows how the convergence of uplift over iterates translates to the distance of prices from CHPs). Instead, the Level Method reaches near-optimal prices in fewer iterations. This is an inherent advantage of the Level Method, which is by design a *stabilization* approach.

Finally, we comment on the *primal method* benchmark. The FERC model exceeds what a primal CHP approach such as [19] can model, since it includes ramp constraints and time-dependent startup costs. The integer relaxation is therefore expected to lead to an *approximation* of CHP. The quality of the primal method largely depends on the tightness of the formulation. In this respect, the FERC model is derived from a careful review of the literature dedicated to tight formulations of the unit commitment model [27], [35]. The quality of the model is discussed in [34], where it is accompanied by computational experiments of its tightness. As observed in Table I, the primal method turns out to provide a close approximation of CHP *on these FERC instances*. Nonetheless, this is not always guaranteed, as we observe in the next test case (Table IV), where the primal method leads to an average uplift which is $\sim 60,000$€ higher than CHP, for a market of comparable dispatch cost.

The test cases analysed so far suggest a promising performance for the Level Method. Nevertheless, even if these FERC instances are of realistic scale insofar as the number of power plants are concerned, we are interested in computing *prices*. This suggests that it is the *dimension of the price space* that matters the most. There are essentially two ways[5] to increase the price dimension: (i) augmenting the *time horizon* — the horizon of future EU markets will be 96 periods of 15 minutes — and (ii) adding a network — which is unavoidable in both the EU and the US markets. This motivates the next test cases.

B. EU test cases

We now extend our analysis to use cases *with a network*. The EU dataset that we utilize is the one used in [45]. The network data is based on [46], and is constructed among others from an ENTSO-E database. The market suppliers are modelled as a slightly simpler version of the UC model than the FERC test case, essentially simplifying the cost structure: there is a single start-up cost, instead of the variable start-up costs of FERC, and the marginal production cost is constant. All the cases are simulated over 6 different load profiles. As we are interested in studying the scalability of the Level Method and D-W algorithm with respect to the *network* and the *time horizon*, the data has been aggregated into two test cases: BE and BE-NL, which are described in Table II. As detailed in section I, Euphemia, the EU market clearing algorithm, currently computes prices for $\sim$ 40 bidding zones, and is

[5]A third way would be the introduction of reserve. The current EU DA market does not co-optimize energy and reserve, which is why it is not considered in our analysis. Nevertheless, art. 40 of EGBL guidelines indicates that this could constitute a future evolution of the EU market.

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

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

expected to move to 15-minute granularity (96 time periods) in the near future. This makes our two tests cases with 96 periods very relevant proxies of the evolving EU context with respect to price dimensionality.

The final results are obtained with the stopping criterion set to 0.01%, as for the FERC cases. Table III shows the sensitivity of the Level Method towards parameter α, previously discussed in section III-C. Table IV presents results for the BE test case with multiple *time horizons*. Figs. 6 and 7 illustrate the convergence of the BE test case with 96 periods. Table V presents a comparison for different *network sizes*. It is worth noting that, in all the test cases (except the 12-period BE test case, which is however less relevant for practical applications), the Level Method turns out to be superior to D-W in terms of iteration count. Furthermore, we observe that the benefits of the Level Method are magnified when increasing the dimension of the price space.

More specifically, insofar as sensitivity with respect to the *time horizon* is concerned, Table IV demonstrates that the Level Method scales well with respect to the horizon of the problem as it increases from 19 to 44 iterates as the horizon grows from 12 to 96 periods. On the other hand, the performance of D-W is seriously harmed by the increase of the horizon: the number of iterates increases from 19 to 236. The stable behavior of the Level Method is corroborated by Fig. 6. We observe that, within 6 iterates, it already reaches a price that achieves lower uplifts than those of O'Neill pricing.

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).

TABLE V
RESULTS OF LEVEL METHOD AND THE DANTZIG-WOLFE ALGORITHM FOR DIFFERENT NETWORK SIZES (AVERAGE OVER 6 INSTANCES).

horizon	**24**		**96**	
test case	**BE**	**BE-NL**	**BE**	**BE-NL**
Level iter	26	21	44	42
Level av. time/iter[a][s]	0.8 (0.1)	1.5 (0.3)	5.8 (1.6)	12.3 (4.8)
Level total run time [s]	21	31	255	514
D-W iter	40	32	236	156
D-W av. time/iter[a][s]	0.7 (0.1)	1.3 (0.2)	6.9 (2.1)	12.7 (3.7)
D-W total run time [s]	27	42	1622	1976

[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).

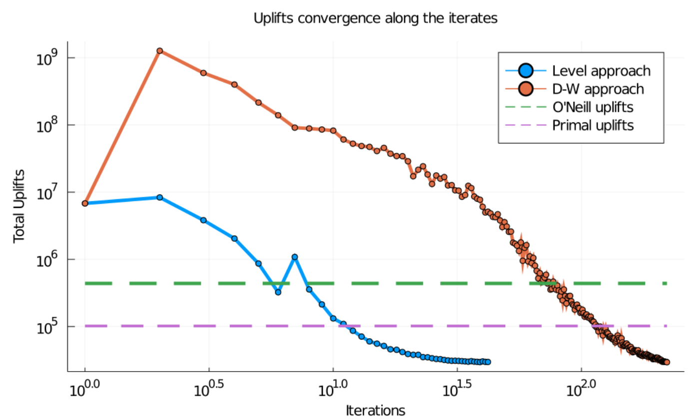

Fig. 6. Convergence of the Level and D-W approaches, measured by the uplifts (*O'Neill pricing* and the primal method are used as benchmark thresholds), on the BE summer weekday 96-periods instance. Both axes are in logarithmic scale.

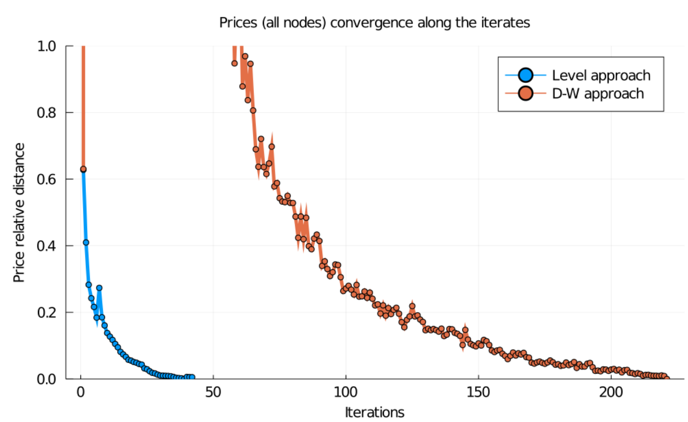

Fig. 7. Convergence of the Level and D-W approaches, measured by the price relative distance to CHP, on the BE summer weekday 96-periods instance.

Fig. 7 also presents the convergence of both algorithms on the same instance in terms of price distance to the optimum. Being capable to reach quickly decent price candidates is an attractive feature for EU implementation of CHP, recalling from section I that Euphemia is currently granted 12 minutes for computing the EU day-ahead market matchings and prices.

As far as the *network size* is concerned, Table V presents the sensitivity with respect to the two use cases. Perhaps surprisingly, neither of the methods is strongly affected by the size of the network, rather the contrary. On the instance with 24 periods, the benefits of the Level Method are similar as in the FERC cases. On the 96-period instances, the Level Method moves from five to four times faster than D-W on the BE and BE-NL cases, in terms of iteration count. Overall, D-W seems much more affected by the increase in the *time horizon* rather than the presence of a *network*, to which the test cases suggest D-W is rather robust. This is possibly due to the fact that D-W is required to explore in the space of promising power plant schedules — and these schedules become more and more numerous when increasing the horizon — while the network size does not affect immediately the number of schedules.

It should be stressed that the aforementioned computational gains can make a difference for the practical implementation of CHP, keeping in mind the 12-minute run time limit of Euphemia. From Table IV, we observe that the Level Method requires less than 5 minutes on average for solving a 96-period instance. The D-W algorithm requires 27 minutes.

The computational times reported in our results may of course not be representative of the implementation of the EU NEMOs, as solving the slaves in parallel and increasing the computational power would reduce the run time. Assuming an idealized parallelization of the slaves — which is very optimistic considering the NEMOs currently run Euphemia on 8 threads [47] —, the run time per iterate would be lower-bounded by the time for solving the master programs (master plus projection programs for the Level Method, as reported between brackets in the tables). As an example, the "most difficult" BE-instance was solved in 266 iterates by D-W, with 2.3 sec/iter for solving the master program. This implies a lower bound of more than 10 minutes for obtaining the CHP. On the same instance, the Level Method required 37 iterates, with 1.4 sec/iter for solving the masters, which amounts to a total of less than 1 minute. Furthermore, whereas the price dimension of our test cases has been selected so as to be comparable to the EU market, the number of generators (or market bids) is well below the value that occurs in practice. As an order of magnitude, Euphemia currently solves instances with around 160,000 hourly orders (convex) and 4,000 block orders (non-convex) [4]. This suggests that the time for solving the master programs would likely be higher on the real instances of Euphemia.

V. Conclusion

Our paper proposes a (known) bundle stabilization approach for efficiently solving convex hull pricing. We demonstrate that the Level Method is able to converge within few iterations to a certain target gap, while exhibiting a stable behaviour, on large instances which, in terms of price space dimension, are comparable to the size of the EU day-ahead auction.

It is likely that the choice of the best algorithm for solving CHP will depend on the specific use-case: the dimension of the network, the time horizon, the complexity of the unit commitment / market orders, the run time that is afforded to the algorithm, etc. Although no method can conceivably provide an ultimate solution for computing CHP in an arbitrarily complex setting, the Level Method indicates the promising behaviour of a family of "bundle approaches". This suggests areas of future research on alternative bundle approaches, such as the *Proximal Stabilization* method, the *Doubly-Stabilized Bundle Method* [40] or the Boxstep method [48], which appear to be well suited for solving the CHP Lagrangian relaxation.

Another question for future research relates to how the proposed approach can be adapted in case one of the following assumptions is relaxed: the convexity of the grid model and the separability of the generators profit maximization problems.

Having scalable algorithms capable to compute CHP on large instances also enables more extensive quantitative analysis of its economical behaviour. As far as the EU market is concerned, we are interested in (i) expanding tests on realistic instances of Euphemia — our preliminary tests show that the Level Method can solve the 4MMC run of Euphemia [13] in ~ 1 minute —, (ii) examining the effects of non-uniform pricing on enhancing welfare in the EU day-ahead market, and (iii) understanding distributional effects of non-uniform pricing as well as gaming effects.

Acknowledgment

The authors thank Gauthier de Maere d'Aertrycke for numerous contributions including the treatment of the network subproblem, Yurii Nesterov for guiding us towards the Level Method as well as Mehdi Madani, William W. Hogan, Panagiotis Andrianesis, Ross Baldick and Alejandro Angulo for their fruitful comments on our work. The work has received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme (grant agreement No. 850540) as well as from the FEVER Project (grant agreement No. 864537).

References

[1] R. E. Bohn, M. C. Caramanis, and F. C. Schweppe, "Optimal pricing in electrical networks over space and time," *The Rand Journal of Economics*, pp. 360–376, 1984.

[2] J. A. Taylor, *Convex optimization of power systems*. Cambridge University Press, 2015.

[3] M. Garcia, H. Nagarajan, and R. Baldick, "Generalized convex hull pricing for the ac optimal power flow problem," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 3, pp. 1500–1510, 2020.

[4] Nemo Committee, "CACM annual report 2019," July 2020. [Online]. Available: http://www.nemo-committee.eu/assets/files/cacm-annual-report-2019.pdf

[5] D. A. Schiro, T. Zheng, F. Zhao, and E. Litvinov, "Convex hull pricing in electricity markets: Formulation, analysis, and implementation challenges," *IEEE Transactions on Power Systems*, vol. 31, no. 5, pp. 4068–4075, 2015.

[6] 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.

[7] W. W. Hogan and B. J. Ring, "On minimum-uplift pricing for electricity markets," *Electricity Policy Group*, pp. 1–30, 2003.
[8] P. R. Gribik, W. W. Hogan, S. L. Pope *et al.*, "Market-clearing electricity prices and energy uplift," *Cambridge, MA*, 2007.
[9] PJM Interconnection, "Proposed enhancements to energy price formation," 2017.
[10] Federal Energy Regulatory Commission, Apr. 2019, docket No. EL18-34-000.
[11] M. Van Vyve *et al.*, "Linear prices for non-convex electricity markets: models and algorithms," CORE, Tech. Rep., 2011.
[12] A. Papavasiliou and G. Bertrand, "Market design options for scarcity pricing in european balancing markets," *IEEE Transactions on Power Systems*, 2021.
[13] Nemo Committee *et al.*, "Euphemia public description, single price coupling algorithm," Tech. Rep., October 2020.
[14] Y. Nesterov, *Introductory lectures on convex optimization: a basic course*. Springer, 2004.
[15] 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 Transactions on Power Systems*, 2021.
[16] N. Stevens, A. Papavasiliou, and G. de Maere d'Aertrycke, "Models and algorithms for pricing electricity in unit commitment," 2016. [Online]. Available: https://ap-rg.eu/wp-content/uploads/2020/06/MasterStevens.pdf
[17] C. Lemaréchal, "Lagrangian relaxation," in *Computational combinatorial optimization*. Springer, 2001, pp. 112–156.
[18] L. A. Wolsey, *Integer programming*. John Wiley & Sons, 2020.
[19] B. Hua and R. Baldick, "A convex primal formulation for convex hull pricing," *IEEE Transactions on Power Systems*, vol. 32, no. 5, 2017.
[20] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM review*, vol. 38, no. 1, pp. 49–95, 1996.
[21] S. Boyd and L. Vandenberghe, "Semidefinite programming relaxations of non-convex problems in control and combinatorial optimization," in *Communications, Computation, Control, and Signal Processing*. Springer, 1997, pp. 279–287.
[22] C. Lemarechal and F. Oustry, "Sdp relaxations in combinatorial optimization from a lagrangian viewpoint," in *Advances in Convex Analysis and Global Optimization*. Springer, 2001, pp. 119–134.
[23] C. Lemaréchal and A. Renaud, "A geometric study of duality gaps, with applications," *Mathematical Programming*, vol. 90, no. 3, pp. 399–427, 2001.
[24] 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," *arXiv preprint arXiv:1804.00048*, 2018.
[25] Y. Yu, Y. Guan, and Y. Chen, "An extended integral unit commitment formulation and an iterative algorithm for convex hull pricing," *IEEE Transactions on Power Systems*, vol. 35, no. 6, pp. 4335–4346, 2020.
[26] 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 Transactions on Power Systems*, vol. 35, no. 3, pp. 2108–2119, 2019.
[27] G. Morales-España, J. M. Latorre, and A. Ramos, "Tight and compact milp formulation for the thermal unit commitment problem," *Power Systems, IEEE Transactions on*, vol. 28, no. 4, pp. 4897–4908, 2013.
[28] G. Morales-España, C. Gentile, and A. Ramos, "Tight mip formulations of the power-based unit commitment problem," *OR Spectrum*, vol. 37, no. 4, pp. 929–950, 2015.
[29] C. Gentile, G. Morales-Espana, and A. Ramos, "A tight mip formulation of the unit commitment problem with start-up and shut-down constraints," *Institute for Research in Technology (IIT), Technical Report IIT-14-040A*, 2014.
[30] D. Rajan and S. Takriti, "Minimum up/down polytopes of the unit commitment problem with start-up costs," *IBM Res. Rep*, 2005.
[31] P. Damcı-Kurt, S. Küçükyavuz, D. Rajan, and A. Atamtürk, "A polyhedral study of ramping in unit commitment," *Univ. California-Berkeley, Res. Rep. BCOL*, vol. 13, 1777.
[32] M. Queyranne, L. Wolsey *et al.*, "Tight mip formulations for bounded up/down times and interval-dependent start-ups," UCL, Tech. Rep., 2015.
[33] M. Silbernagl, M. Huber, and R. Brandenberg, "Improving accuracy and efficiency of start-up cost formulations in mip unit commitment by modeling power plant temperatures," 2014.
[34] B. Knueven, J. Ostrowski, and J.-P. Watson, "On mixed-integer programming formulations for the unit commitment problem," *INFORMS Journal on Computing*, vol. 32, no. 4, pp. 857–876, 2020.
[35] S. Sridhar, J. Linderoth, and J. Luedtke, "Locally ideal formulations for piecewise linear functions with indicator variables," *Operations Research Letters*, vol. 41, no. 6, pp. 627–632, 2013.
[36] 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 marketspart i: Algorithm structure," *Power Systems, IEEE Transactions on*, vol. 28, no. 3, pp. 2111–2120, 2013.
[37] ——, "An extreme-point subdifferential method for convex hull pricing in energy and reserve marketspart ii: Convergence analysis and numerical performance," *Power Systems, IEEE Transactions on*, vol. 28, no. 3, pp. 2121–2127, 2013.
[38] S. Boyd, L. Vandenberghe, and J. Skaf, "Analytic center cutting-plane method," 2008.
[39] C. Wang, P. B. Luh, P. Gribik, L. Zhang, and T. Peng, "A subgradient-based cutting plane method to calculate convex hull market prices," in *Power & Energy Society General Meeting, 2009. PES'09. IEEE*. IEEE, 2009, pp. 1–7.
[40] A. Frangioni, "Standard bundle methods: untrusted models and duality," in *Numerical Nonsmooth Optimization*. Springer, 2020, pp. 61–116.
[41] J. R. Birge and F. Louveaux, *Introduction to stochastic programming*. Springer Science & Business Media, 2011.
[42] J. R. Birge and F. V. Louveaux, "A multicut algorithm for two-stage stochastic linear programs," *European Journal of Operational Research*, vol. 34, no. 3, pp. 384–392, 1988.
[43] F. Vanderbeck and L. A. Wolsey, "Reformulation and decomposition of integer programs," in *50 Years of Integer Programming 1958-2008*. Springer, 2010, pp. 431–502.
[44] E. Krall, M. Higgins, and R. P. ONeill, "Rto unit commitment test system," *Federal Energy Regulatory Commission*, vol. 98, 2012.
[45] I. Aravena and A. Papavasiliou, "Renewable energy integration in zonal markets," *IEEE Transactions on Power Systems*, vol. 32, no. 2, pp. 1334–1349, 2016.
[46] N. Hutcheon and J. W. Bialek, "Updated and validated power flow model of the main continental european transmission network," in *2013 IEEE Grenoble Conference*. IEEE, 2013, pp. 1–5.
[47] Nemo Committee, "Release note euphemia 10.3," April 2019. [Online]. Available: https://www.nordpoolgroup.com/492d61/globalassets/download-center/single-day-ahead-coupling/release-note-euphemia_10-3.pdf
[48] R. E. Marsten, W. Hogan, and J. W. Blankenship, "The boxstep method for large-scale optimization," *Operations Research*, vol. 23, no. 3, pp. 389–405, 1975.

Nicolas Stevens (M'22) obtained his B.Sc. and M.Sc. degrees in Mathematical Engineering from UCLouvain, Louvain-la-Neuve, Belgium. He is since 2020 a Ph.D. student in Applied Mathematics at the Center for Operations Research and Econometrics of UCLouvain. From 2016 to 2020, Nicolas worked as a consultant in the energy sector at N-SIDE, the company that notably developed Euphemia, the clearing algorithm of the EU DA electricity market.

Anthony Papavasiliou (M'06-SM'20) is an Associate Professor at the Center for Operations Research and Econometrics at the Universite catholique de Louvain, where he holds the ENGIE Chair. He works on operations research, electricity market design, and electric power system operations. He is the recipient of the Francqui Foundation research professorship (2018), the ERC Starting Grant (2019), and the Bodossaki Foundation Distinguished Young Scientist award (2021). He has served as an Associate Editor of Operations Research and the IEEE Transactions on Power Systems.