Exact and Heuristic Methods for Energy Pricing Problem

Sezin Afsar, Luce Brotcorne, Patrice Marcotte, Gilles Savard

Bilevel Modelling and Heuristic Methods for Energy Peak Minimization Problem

Sezin Afşar[a,*], Luce Brotcorne[a], Patrice Marcotte[b], Gilles Savard[c]

[a] *Inria Lille-Nord Europe, 40 Avenue Halley 59650 Villeneuve d'Ascq, France*
[b] *DIRO, Université de Montréal, C.P. 6128, Succursale Centre-Ville, Montréal (QC) H3C 3J7, Canada*
[c] *Département de Mathématique et Génie Industriel, École Polytechnique, C.P. 6079, Succursale Centre-Ville, Montréal (QC) H3C 3A7, Canada*

Abstract

The dynamics between a peak-minimizing and revenue-maximizing energy provider and its disutility-minimizing customers are analyzed in a bilevel setting in this paper. Customers' demand is delivered to a smart grid which acts as the follower and the provider decides on the prices as the leader. Different models that are based on the types of appliances that customers own are presented and solved by exact and heuristic methods that are developed.

Keywords: demand response, smart grid, day-ahead pricing, bilevel programming, peak minimization, heuristic algorithms

1. Introduction

Energy demand tends to grow rapidly as a result of technological developments and energy generation increases as a result of it. Managing the supply-demand balance of energy is one of the most challenging problems due to mutual escalation of supply and demand. When this balance is disturbed, it has a huge impact on the economy and society. Energy provider firms usually prefer to keep large capacities which is an expensive solution to avoid these consequences. Another way to deal with instability problem is

*Corresponding author. Tel: +33 59358628

Email addresses: sezin.afsar@inria.fr (Sezin Afşar), luce.brotcorne@inria.fr (Luce Brotcorne), marcotte@iro.umontreal.ca (Patrice Marcotte), gilles.savard@polymtl.ca (Gilles Savard)

Preprint submitted to Elsevier *November 19, 2015*

employing demand side management (DSM) methods that involves control mechanisms at the user side to use the installed capacity more efficiently [26, 30]. Peak clipping, valley filling, load shifting, strategic conservation, strategic load growth and flexible load shape are certain tools in order to form a more appealing load curve [16]. Conservation and energy efficiency programs, fuel substitution programs, demand response programs, and residential or commercial load management programs are applied to reach these curve shaping goals [26, 17, 24]. More precisely, the problem regarding the load curve adjustment via pricing is discussed in several articles in the literature. A thorough literature review concerning dynamic pricing, as well as analyses of real cases, can be found in [4].

In general terms, energy demand curve can be analyzed in two parts: base load and peak load. Base load is typically provided by the facilities that have low unit cost and high start-up cost such as coal, nuclear or natural gas power plants. Peak load is generated by the facilities that have more flexible generation capacity, in return for a high unit cost and low start-up cost like solar, wind or geothermal power plants. Therefore, electricity production and consumption is more expensive during peak periods. Moreover, the energy provider is obliged to keep more energy generation capacity than the peak load to provide a stable power supply.

Demand curve tends to fluctuate substantially over different time periods. For instance, it is observed in the United Kingdom that the minimum load of summer is around 30% of the winter peak and the average load is approximately 55% of the existing generation capacity which leaves almost half of the capacity idle on average [30]. Besides, plug-in electric vehicles (PHEV) are expected to proliferate in near future. Extensive use of PHEVs may even double residential electric load [28].

Decreasing the peak load by load shifting to smooth out supply curve is the main concern of this paper and the problem that is tackled is called Energy Peak Minimization Problem (EPMP). In this framework, there are two decision makers, namely an energy provider who maximizes revenue and its customers who minimize total cost. Many articles approach the problem from a Nash game point of view [28, 31, 29, 20, 21]. However, the hierarchical game between the firm and its customers can be modeled as a bilevel program where the follower's (customers') optimization problem is integrated into the leader's (energy provider's) decision-making process. Bilevel programming has been used to model many problems in the literature such as toll pricing [23, 7], freight tariff setting [6, 10], network design and pricing [5, 25, 8, 9], electric utility planning [19, 3]. It is important to note that bilevel programs are difficult (NP-hard) in general terms [22, 18]. The feasible region of the leader is usually nonconvex and it can be disconnected or empty [11].

In [28], the authors proposed an "incentive-based energy consumption scheduling algorithm in order to minimize the cost of energy" while minimizing peak-to-average load ratio. The customers play a Nash game and the strategies are appliances and loads. In [27], the authors proposed an energy consumption scheduling framework in order to find a trade-off between electricity payment and waiting time of customers while targeting for a low peak-to-average ratio. The modeling approach of these two papers is the basis of customer behavior modeling in this paper. However, customers do not compete with each other, the smart grid aims to find a system optimal schedule in our setting.

The aim of this paper is to pursue the previous work [1] while developing new models and efficient solution methods. A optimal trade-off between revenue and peak cost is aimed while respecting customer choices. The models assume day-ahead and time-of-use pricing. Customers own only nonpreemptive devices in the first model whereas the second model has both preemptive and nonpreemptive appliances. The models with preemptive devices can be found in [1]. We propose two heuristic methods that scale better than the exact method.

The bilevel model formulations are presented in the next section. A theorem that shows a new way of reformulating assignment problem and its use for our models are explained Section 3. It is followed by the description of the heuristic procedures. In Section 5, experimental results of exact and heuristic methods are presented under different parameters and instances. The paper is concluded with a review and future remarks.

2. Models

Mathematical formulation of a general bilevel program is expressed as:

$$\begin{aligned} \max_{x\in X} \quad & f(x,y) \\ \text{s.t.} \quad & y \in \arg\min_{y'\in Y(x)} g(x,y'). \end{aligned}$$

In the above model, to all possible vector selection x of the leader, the follower responds with a vector y that is optimal with respect to the fixed x. The leader chooses x to maximize its objective while considering the follower's reaction. In this paper, we tackle the problem where the electricity supplier firm picks the prices taking optimal schedule response of the follower into account. In most bilevel programs, lower level problem is convex for a given upper level solution. It is also the case for the models in this paper. Note that vertical representation is used for our bilevel programs for convenience of reading.

In bilevel programming, there might be multiple optimal solutions for the follower for a fixed upper level solution. Depending on the implemented solution being most or least favorable for the leader, the modeling approach is called optimistic or pessimistic, respectively. Optimistic approach is used in this work, and more detailed information on pessimistic modeling can be found in [14, 12].

In this section, two bilevel models are presented. Both models assume a power-sharing system of customers where all customers own a smart metering device. Each customer has a set of home appliances which are assumed to be nonpreemptive in Section 2.1, and both preemptive and nonpreemptive in Section 2.2. The previous work of authors [1] can be consulted for a detailed study on preemptive EPMP modeling.

Once a nonpreemptive device starts working, it cannot be stopped until the job is finished and the device's power consumption cannot be changed. Washing and drying machines and dishwashers may be counted as nonpreemptive appliances among many others. On the other hand, preemptive devices can be interrupted, restarted and adjusted, like water and pool heaters, radiators, refrigerators, air conditioners etc.

The scheduling horizon consists of 24 hours. Each appliance is identified with a demand and a time window. It is important to note that the time windows are strict, i.e., it is not possible to operate a device outside of its window. Demand and time windows are provided by customers to the grid. For nonpreemptive devices, demand is satisfied by starting it since once a job starts, it has to be completed without interruption. However, demand for a preemptive device is fulfilled when total power consumption within its time window matches the requested amount.

All information smart meters receive from customers is shared and transmitted, and hence a grid is formed. The grid is responsible of scheduling the power consumption of customers with respect to the prices it receives from the power source every 24 hours. Customers are supposed to be residential users, they may have different habits and power consumption behavior such that some of them are willing to postpone their load for a reduced bill

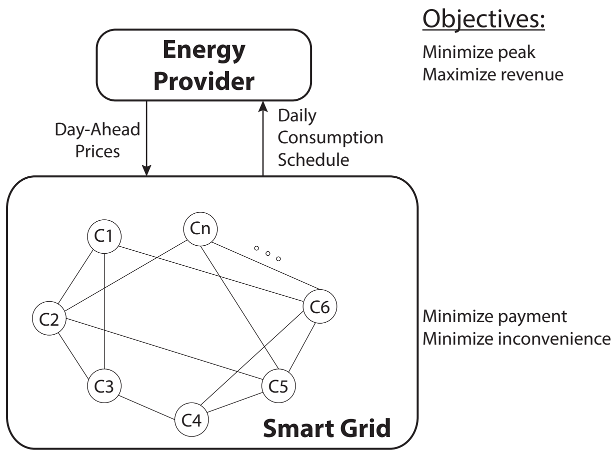


Figure 1: Bilevel Structure

whereas others might be less patient about perturbations. Behavioral differences among customers are represented by an *inconvenience factor* that is specific to each customer. The energy prices are the same for all users for the same time slot since they are all residential users.

A *day-ahead* pricing system is considered in this paper, that is, the provider decides on the prices of the next day and delivers them to the grid. Then, the grid schedules all jobs with respect to these prices and customer preferences. At this point it is important to emphasize the role of grid, without which the system would not function properly since the customers do not have enough time and patience to monitor daily changing prices in real life. The general modeling scheme can be seen in Figure 1.

2.1. Nonpreemptive Monopolist Pricing

Consider a power sharing system with N customers who own A_n nonpreemptive devices (follower) and a monopolist power supplier (leader). Leader decides on the prices p^h of time slots $h \in H$, then smart grid schedules the jobs, or in other words, chooses starting time $x^h_{n,a}$ of each appliance $a \in A_n$ of each customer $n \in N$.

Leader's objective is to find a trade-off between two conflicting objectives: revenue and peak cost. Peak load is represented with Γ in the model and penalty of peak is κ. Peak consumption is penalized to have a smoother power supply. The trade-off between the revenue and peak cost depends on the value of κ with respect to prices.

Each appliance has a fixed power consumption per time slot, $k_{n,a}$, during a fixed period, $l_{n,a}$. Besides, every appliance has a time window chosen by customers which is defined by beginning and end times, $T_{n,a} = [TW^b_{n,a}, TW^e_{n,a}]$, and it cannot be started out of its time window.

Follower's objective function is minimizing total cost. It consists of two parts: electricity bill and inconvenience cost. For a given price vector set by the leader, smart grid minimizes the total electricity payment, as well as minimizing the total inconvenience cost. In other words, customers would like to buy electricity at the cheapest possible price without delaying their consumption too much.

It is supposed that the first time slot of an appliance is the most preferable starting time for the customer. However, a job can be started at any slot within the time window in exchange for a penalty cost. We consider the inconvenience cost of a job to be a linear penalty function that is directly proportional to length of the delay. The penalty is inversely proportional to the width of the desired time window, reflecting the fact that customers that specify narrow time windows are likely to be sensitive to the delay of their tasks by the smart grid. The inconvenience cost of a job, $C_{n,a}(h)$, is computed as follows where $\lambda_{n,a}$ is inconvenience coefficient of an appliance a of customer n. A higher $\lambda_{n,a}$ value represents a lower tolerance to delay.

$$C_{n,a}(h) := \lambda_{n,a} \times k_{n,a} \times l_{n,a} \times \frac{(h - TW^b_{n,a})}{(TW^e_{n,a} - TW^b_{n,a})} \qquad \forall n \in N, \forall a \in A_n, \forall h \in T_{n,a}.$$

Now we present the bilevel mathematical model of EPMP with nonpreemptive appliances and monopolistic pricing:

$$\text{(NMP):}\ \max_{p,\Gamma} \sum_{n\in N}\sum_{a\in A_n}\sum_{h\in T_{n,a}} k_{n,a}x^h_{n,a}\sum_{h'=h}^{h+l_{n,a}} p^{h'} \quad -\kappa\Gamma$$

s.t.

$$\Gamma \geq \sum_{n\in N}\sum_{a\in A_n}\sum_{\substack{h'=h-l_{n,a}\\ s.t. h'\in T_{n,a}}}^{h} k_{n,a}x^{h'}_{n,a} \qquad \forall h\in H \tag{1}$$

$$0 \leq p^h \leq p^h_{\max} \qquad \forall h\in H \tag{2}$$

$$\min_{x} \sum_{n\in N}\sum_{a\in A_n}\sum_{h\in T_{n,a}} k_{n,a}x^h_{n,a}\sum_{h'=h}^{h+l_{n,a}} p^{h'} \quad + \sum_{n\in N}\sum_{a\in A_n}\sum_{h\in T_{n,a}} C_{n,a}(h)x^h_{n,a}$$

s.t.

$$\sum_{h\in T_{n,a}} x^h_{n,a} = 1 \qquad \forall n\in N, \forall a\in A_n \tag{3}$$

$$x^h_{n,a}\in\{0,1\} \qquad \forall n\in N, \forall a\in A_n, \forall h\in T_{n,a} \tag{4}$$

Constraint (1) guarantees that peak load Γ is greater than or equal to all loads for each $h \in H$. When $x^h_{n,a}$ takes value 1 for a time slot h, it brings a power load of $k_{n,a}$ during $l_{n,a}$ time slots. Constraint (2) sets upper and lower limits for prices. Demand satisfaction is ensured by constraint (3).

As mentioned above, all customers receive the same prices. The EPMP with customer-specific pricing is an easier problem since it gives leader the flexibility of shifting only one job at a time by changing one price. However, a price change for one slot can affect whole schedule and load distribution in our model.

2.2. Mixed Monopolist Pricing

In this section, a mixed model involving both preemptive and nonpreemptive appliances is presented in order to have a more realistic view. Leader maximizes net revenue by deciding on prices and follower minimizes total cost by scheduling the use of each appliance.

In the lower level there are N customers, each of them owning two sets of appliances, preemptive (A^1_n) and nonpreemptive (A^2_n). Each appliance has an adequate time window T_{n,a_1} and T_{n,a_2} for type 1 and type 2 devices respectively, as explained in the previous sections. Nonpreemptive devices have fixed power load k_{n,a_2} and operating time l_{n,a_2} whereas preemptive devices have a device power limit $\beta_{n,a_1}^{\max}$ and total power demand E_{n,a_1}.

The objective function of the lower level is sum of total billing and inconvenience cost. Both terms consist of two parts dedicated to different types of appliances. The inconvenience parameter is $C_{n,a_1}(h)$ and $C_{n,a_2}(h)$, the decision variables are x_{n,a_1}^h and y_{n,a_2}^h for for type 1 and type 2 appliances, respectively.

Similar to the previous models, the objective function of upper level is maximizing total revenue while minimizing peak cost. Total revenue term has two parts that come from different types of appliances. Price p^h is the decision variable of upper level and it is defined for each time period $h \in H$. Other decision variable is the peak load (Γ) which is the highest amount of power consumption throughout the scheduling period.

$$\text{(MMP):} \max_{p,\Gamma} \sum_{\substack{n\in N\\ a_2\in A_n^2\\ h\in T_{n,a_2}}} k_{n,a_2} y^h_{n,a_2} \sum_{h'=h}^{h+l_{n,a_2}} p^{h'} + \sum_{\substack{n\in N\\ a_1\in A_n^1\\ h\in T_{n,a_1}}} p^h x^h_{n,a_1} \qquad - \kappa\Gamma$$

s.t.

$$\Gamma \geq \sum_{\substack{n\in N\\ a_2\in A_n^2}} \sum_{\substack{h'=h-l_{n,a_2}\\ \text{s.t.} h'\in T_{n,a_2}}}^{h} k_{n,a_2} y^{h'}_{n,a_2} + \sum_{\substack{n\in N\\ a_1\in A_n^1\\ \text{s.t.} h\in T_{n,a_1}}} x^h_{n,a_1} \qquad \forall h \in H \tag{5}$$

$$0 \leq p^h \leq p^h_{\max} \qquad \forall h \in H \tag{6}$$

$$\min_x \sum_{\substack{n\in N\\ a_2\in A_n^2\\ h\in T_{n,a_2}}} \left(k_{n,a_2} \sum_{h'=h}^{h+l_{n,a_2}} p^{h'} + C_{n,a_2}(h)\right) y^h_{n,a_2} + \sum_{\substack{n\in N\\ a_1\in A_n^1\\ h\in T_{n,a_1}}} \left(p^h + C_{n,a_1}(h)\right) x^h_{n,a_1}$$

s.t.

$$0 \leq x^h_{n,a_1} \leq \beta^{\max}_{n,a_1} \qquad \forall n \in N, a_1 \in A_n^1, h \in T_{n,a_1} \tag{7}$$

$$\sum_{h\in T_{n,a_1}} x^h_{n,a_1} = E_{n,a_1} \qquad \forall n \in N, a_1 \in A_n^1. \tag{8}$$

$$\sum_{h\in T_{n,a_2}} y^h_{n,a_2} = 1 \qquad \forall n \in N, a_2 \in A_n^2 \tag{9}$$

$$y^h_{n,a_2} \in \{0,1\} \qquad \forall n \in N, a_2 \in A_n^2, h \in T_{n,a_2} \tag{10}$$

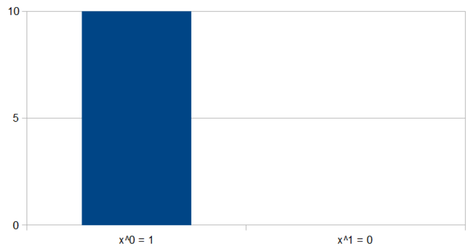
Constraint (5) defines the peak load which is greater than or equal to the total power consumption in a time slot. Constraint (6) represents the price ceiling and stays the same as previous models. Constraints (7) and (8) are the same as the lower level constraints of NMP whereas constraints (9) and (10) are identical to the lower level constraints of PMP.

3. Single Level Formulation

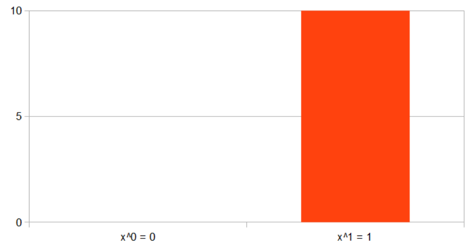
The bilevel program NMP has binary variables in the lower level. Moreover, it belongs to a special class, namely assignment problem. In our case, every device must be assigned to a time slot within its time window. However, there might be time slots without any devices. This property makes the problem even easier than assignment problem. Therefore, for fixed prices, lower level problem can be considered as an assignment problem and hence the integrality constraints can be relaxed. Since it becomes an LP, we can arrange it as a single level problem by adding the primal, dual and complementary slackness constraints (CSC) of lower level to the upper level. Later, this single level model is rearranged as a MIP by linearizing CSC using binary variables [23].

There might be several identical lower level solutions corresponding to the same price vector. As a result of optimistic approach, the solution that gives the highest net revenue for the leader is selected. When we solve the aforementioned MIP, the solution that is favored by the leader might be noninteger. For instance, in our problem, a noninteger solution might provide a lower peak load and hence be more preferable for the energy provider. It is explained on a small example in Figure 2. For the sake of simplicity, there is only one job to be assigned in this example with $k = 10$ and $l = 1$ with time window $[0, 1]$. Price vector of these two hours is $p = (10, 8)$ and inconvenience cost is $C = (0, 20)$. Peak weight is $\kappa = 5$. It means that if $x = (1, 0)$, the load curve appears to be as in Figure 2(a), total cost for the follower is 10*10 + 0 = 100 and net revenue of the leader is 10*10 - 5*10 = 50. If $x = (0, 1)$, the load curve is as in Figure 2(b), total cost for the follower is 8*10 + 20*1 = 100 and net revenue of the leader is 8*10 - 5*10 = 30. And finally, if $x = (0.5, 0.5)$, the load curve is as in Figure 2(c), total cost for the follower is 5*10 + 5*8 + 20*0.5 = 100 and net revenue of the leader is 10*5 + 8*5 - 5*5 = 65. In all three cases, the total cost is the same for the follower, however third solution is the best for the leader which is noninteger.

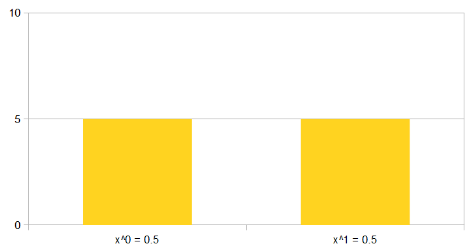
In order to avoid noninteger solutions, another way of writing a MIP model is developed. An integer program (assignment problem) is given in (P) to demonstrate the method. Linear program (R1) is the linear relaxation of (P).



(a) Load Curve of $x = (1, 0)$



(b) Load Curve of $x = (0, 1)$



(c) Load Curve of $x = (0.5, 0.5)$

Figure 2: Toy Example

$$
\begin{aligned}
(\mathrm{P})\quad & \min \sum_{i,j} c_{i,j} x_{i,j} & & \\
& \sum_{i} x_{i,j} = 1 & \forall j \quad & (11) \\
& x_{i,j} \in \{0,1\} & \forall i,j \quad & (12)
\end{aligned}
$$

$$
\begin{aligned}
(\mathrm{R1})\quad & \min \sum_{i,j} c_{i,j} x_{i,j} & & \\
& \sum_{i} x_{i,j} = 1 & \forall j \quad & (13) \\
& x_{i,j} \in [0,1] & \forall i,j \quad & (14)
\end{aligned}
$$

By using (R1), we can write another system of equations (R2), which consists of the primal and dual constraints along with CSCs. The dual variables corresponding to constraints (13) and (14) are v_j and $w_{i,j}$, respectively.

$$
\begin{aligned}
(\mathrm{R2})\quad & \sum_{i} x_{i,j} = 1 & \forall j \quad & (15) \\
& x_{i,j} \le 1 & \forall i,j \quad & (16) \\
& u_j - w_{i,j} \le c_{i,j} & \forall i,j \quad & (17) \\
& x_{i,j}(c_{i,j} - u_j + w_{i,j}) = 0 & \forall i,j \quad & (18) \\
& w_{i,j}(1 - x_{i,j}) = 0 & \forall i,j \quad & (19) \\
& x_{i,j}, w_{i,j} \ge 0 & \forall i,j \quad & (20)
\end{aligned}
$$

Lastly, the nonlinear CSCs (18) and (19) are linearized by adding the integrality constraint of variable $x_{i,j}$ where M_1 and M_2 are sufficiently large constants and hence system of equations (R3) is built:

$$
\begin{aligned}
\text{(R3)}\quad & \sum_i x_{i,j} = 1 && \forall j && (21)\\
& u_j - w_{i,j} \leq c_{i,j} && \forall i,j && (22)\\
& c_{i,j} - u_j + w_{i,j} \leq M_1(1 - x_{i,j}) && \forall i,j && (23)\\
& w_{i,j} \leq M_2 x_{i,j} && \forall i,j && (24)\\
& x_{i,j} \in \{0,1\} && \forall i,j && (25)\\
& w_{i,j} \geq 0 && \forall i,j && (26)
\end{aligned}
$$

Theorem 1. *(R1) has an integer optimal solution* $\iff$ *it is feasible for (R3).*

Proof. Suppose that (R1) has an integer optimal solution, denoted as x^*_{int}. Then, it means x^*_{int} is feasible with respect to constraint 13 and hence, is optimal for (P) as well.

(R2) consists of the optimality conditions of (R1). Therefore, by definition, the feasible region of (R2) has only the optimal solution(s) of (R1). Besides, the feasible region of (R3) has only the integer optimal solution(s) of (R1). Hence, x^*_{int} would be feasible for (R2) and (R3).

Assume that (R1) only has a noninteger optimal solution, x^*_{real}. Then, it is not the optimal solution of (P). Feasible space defined by (R2) consists of only x^*_{real}. Then, x^*_{real} is infeasible for (R3). In fact, the feasible region defined by (R3) is empty, or in other word, (R3) is infeasible. □

Based on Theorem 1, we can write a MIP that is derived from NMP by formulating lower level as (R3). If it is feasible, then all feasible solutions are optimal for NMP as well. Here comes the MIP formulation:

$$\max_{x,u,w,\Gamma} \sum_{\substack{n\in N\\ a\in A_n}} u_{n,a} - \sum_{\substack{n\in N\\ a\in A_n\\ h\in T_{n,a}}} w^h_{n,a} - \sum_{\substack{n\in N\\ a\in A_n\\ h\in T_{n,a}}} C^h_{n,a} x^h_{n,a} - \kappa\Gamma$$

s.t.

$$\Gamma \geq \sum_{n\in N}\sum_{a\in A_n}\sum_{\substack{h'=h-l_{n,a}\\ s.t. h'\in T_{n,a}}}^{h} k_{n,a} x^{h'}_{n,a} \qquad \forall h\in H \tag{27}$$

$$0 \leq p^h \leq p^h_{\max} \qquad \forall h\in H \tag{28}$$

$$\sum_{h\in T_{n,a}} x^h_{n,a} = 1 \qquad \forall n\in N, \forall a\in A_n \tag{29}$$

$$u_{n,a} - w^h_{n,a} \leq k_{n,a}\sum_{h'=h}^{h+l_{n,a}} p^{h'} + C_{n,a}(h) \qquad \forall n\in N, \forall a\in A_n, \forall h\in T_{n,a} \tag{30}$$

$$k_{n,a}\sum_{h'=h}^{h+l_{n,a}} p^{h'} + C_{n,a}(h) - u_{n,a} + w^h_{n,a} \leq M_1(1-x^h_{n,a}) \quad \forall n\in N, \forall a\in A_n, \forall h\in T_{n,a} \tag{31}$$

$$w^h_{n,a} \leq M_2 x^h_{n,a} \qquad \forall n\in N, \forall a\in A_n, \forall h\in T_{n,a} \tag{32}$$

$$x^h_{n,a}\in\{0,1\}, w^h_{n,a}\geq 0 \qquad \forall n\in N, \forall a\in A_n, \forall h\in T_{n,a} \tag{33}$$

Dual variables u and w correspond to primal lower level constraints (3) and (4), respectively. Constraints (27) and (28) come from upper level of NMP and (29) is a lower level primal constraint of NMP. Constraint (30) is the dual constraint corresponding to lower level variable x of NMP. Constraints (31) and (32) are linearized CSCs of NMP's lower level using the integrality constraint (33) of x.

The objective function of NMP contains a bilinear term which means it is linear when one of the variables is fixed. However, it is quadratic when NMP is expressed as a MIP. Therefore, using the objective function of lower level's dual problem, the quadratic expression is replaced with a linear equivalent.

It is important to underline the fact that MMP can be reformulated as a MIP using the above arguments. The lower level problem of MMP has two disjoint parts and they can be treated separately. Hence, we can repeat the same procedure as NMP for the nonpreemptive part. Single level formulation of the preemptive part was given in [1].

4. Heuristic Methods

In this section, two efficient heuristic methods are presented since the classical exact method that allows to reformulate bilevel models as MIP is highly time consuming to solve large instances. These methods are developed by exploiting the intrinsic properties of models and as shown in Section 5, they provide good results in much shorter time.

In all models, there are 3 decision variables: prices, peak load and schedule. The main idea behind our algorithms is fixing one of these variables and compute the value of the other two. Heuristic I generates different price vectors, finds corresponding schedules and updates the price vector accordingly. Heuristic II is based on generating different peak load values. It aims to find lower level solutions under a fixed peak, or in other words, a capacity.

When a bilevel program is reformulated as a MIP as explained in Section 3 and solved using a commercial solver, its solution is optimistic. However, if the lower level model is solved separately, the optimal solution may not be the same. For instance, for the same prices, the solutions in Example 2(a) and 2(b) are equivalent for the follower but the former is better for the leader. In order to obtain optimistic solutions, the output of both heuristics is given to the corresponding MIP as an initial solution with a time limit. Thanks to this last step, we are able to find an optimistic solution that is close to the algorithms' output.

Before explaining the heuristic methods in more detail, we present three subproblems that are used in the methods: *inverse optimization*, to find the optimal prices of a schedule, *minimum peak*, to compute a lower bound on peak load and *fixed peak*, to find a schedule under limited peak load constraint. All subproblems are given with respect to NMP as an example.

4.1. Inverse Optimization

Inverse optimization (IO) [2] (or inverse linear programming [13]) is an approach that allows to compute the optimal upper level decision corresponding to any feasible lower level solution [23, 15]. The most difficult problem in EPMP is the indirect minimization of the peak load since the leader can manipulate the peak only through pricing. However, once the schedule and hence the peak is fixed, computing the corresponding prices is relatively easy. This property is exploited for both heuristic algorithms.

When prices are fixed, lower level becomes an LP which can be solved quickly. Besides, any price between 0 and p_{max} is feasible in both models. It means that any price vector is capable of imposing a feasible lower level solution, i.e., a corresponding schedule. However, the opposite is not true. There may not be a price vector corresponding to every feasible schedule.

The IO formulation of NMP model is given as an example where $\tilde{x}^h_{n,a}$ stands for the fixed lower level decision:

$$\text{(NMIO)}: \max_p \sum_{n\in N}\sum_{a\in A_n}\sum_{h\in T_{n,a}} k_{n,a}\tilde{x}^h_{n,a}\sum_{h'=h}^{h+l_{n,a}} p^{h'}$$

$$\text{s.t. } 0 \le p^h \le p^h_{\max} \qquad \forall h \in H \tag{34}$$

$$k_{n,a}\sum_{h'=h}^{h+l_{n,a}} p^{h'} + C_{n,a}(h) - u_{n,a} + w^h_{n,a} \ge 0 \quad \forall n \in N, \forall a \in A_n, \forall h \in T_{n,a} \mid \tilde{x}^h_{n,a} = 0 \tag{35}$$

$$k_{n,a}\sum_{h'=h}^{h+l_{n,a}} p^{h'} + C_{n,a}(h) - u_{n,a} + w^h_{n,a} = 0 \quad \forall n \in N, \forall a \in A_n, \forall h \in T_{n,a} \mid \tilde{x}^h_{n,a} = 1 \tag{36}$$

$$w^h_{n,a} = 0 \qquad \forall n \in N, \forall a \in A_n, \forall h \in T_{n,a} \mid \tilde{x}^h_{n,a} = 0 \tag{37}$$

$$w^h_{n,a} \ge 0 \qquad \forall n \in N, \forall a \in A_n, \forall h \in T_{n,a} \mid \tilde{x}^h_{n,a} = 1 \tag{38}$$

The problem NMIO is an LP with a single variable vector, p. Since $\tilde{x}^h_{n,a}$ is fixed, the objective function is linear. Besides, it is possible to reformulate the complementarity as linear constraints based on the values of $\tilde{x}^h_{n,a}$.

4.2. Minimum Peak

This subproblem is developed to obtain a lower bound on peak load. Peak definition constraint is included in addition to the lower level constraints and the objective concerns only minimizing peak. Note that the pricing or inconvenience cost are not part of this model:

$$
\begin{aligned}
\text{(MinPeak)}\ \min_{\Gamma,x} \quad & \Gamma \\
\text{s.t.}\quad & \Gamma \geq \sum_{\substack{n\in N\\ a\in A_n}} \sum_{\substack{h'=h-l_{n,a}\\ s.t. h'\in T_{n,a}}}^{h} k_{n,a} x_{n,a}^{h'} && \forall h \in H \\
& \sum_{h\in T_{n,a}} x_{n,a}^{h} = 1 && \forall n \in N, \forall a \in A_n \\
& x_{n,a}^{h} \in \{0,1\} && \forall n \in N, \forall a \in A_n, \forall h \in T_{n,a}
\end{aligned}
$$

The aim is to find a lower bound but it is not necessarily a tight bound. In fact, the optimal schedule computed by this model might be price infeasible since customer is choice is neglected. Nevertheless, it gives an idea about the range of peak load value. More detail about how and why it is used can be found in Section 4.

4.3. Follower's Problem with Limited Capacity

The bilevel models minimize peak load implicitly through pricing. However, one could look at this problem from a different point of view. The problem of finding a feasible schedule under a fixed load capacity is addressed in this subsection. In other words, it is assumed that a fixed peak load value is dictated to the lower level. The model is presented as follows:

$$
\begin{aligned}
\text{(FixedPeak):}\ \min_{p,\Gamma} & \sum_{n\in N} \sum_{a\in A_n} \sum_{h\in T_{n,a}} C_{n,a}(h) x_{n,a}^{h} \\
\text{s.t.}\quad & \Gamma' \leq \sum_{n\in N} \sum_{a\in A_n} \sum_{\substack{h'=h-l_{n,a}\\ s.t. h'\in T_{n,a}}}^{h} k_{n,a} x_{n,a}^{h} \leq \Gamma'' && \forall h \in H && (39)\\
& \sum_{h\in T_{n,a}} x_{n,a}^{h} = 1 && \forall n \in N, \forall a \in A_n \\
& x_{n,a}^{h} \in \{0,1\} && \forall n \in N, \forall a \in A_n, \forall h \in T_{n,a}
\end{aligned}
$$

Note that it is possible to fix peak load to a single value in the preemptive case. However, in non-preemptive and mixed scenarios, it is a fixed interval rather than a single value. Therefore, a lower (Γ') and upper (Γ'') limits are given in Constraint (39). The integrality constraint is kept since it is not an assignment problem anymore. Therefore, the model is a MIP for nonpreemptive and mixed models, whereas it is an LP for preemptive model.

Pricing is not a part of this model. However, it is possible to compute the prices that induce this model's optimal schedule using IO, if they exist. The role of this model and its implications are discussed in more detail in Section 4.5.

4.4. Heuristic I

This heuristic method is based on generating price neighborhoods and finding corresponding optimal schedules. Note that every price within $[0, p_max]$ interval is feasible and there exists a feasible schedule or in other words, a customer response.

At each iteration, the peak load and the time slot that peak occurs is found *(Finding Peak)*. Then, the price of k many time slots following the peak slot are decreased by some discount factor d *(Price Update)*. Lower level problem is solved with this updated price vector:

$$(p^0, p^1, \ldots, (1-d)p^{i+1}, (1-d)p^{i+2}, \ldots, (1-d)p^{i+k}, p^{i+k+1} \ldots)$$

The corresponding schedule is obtained *(Schedule Computation)*. The optimal prices corresponding to this schedule is computed by *Inverse Optimization*. The objective function value of this price-schedule pair is calculated and compared to the incumbent solution. If it is better, incumbent is updated and the procedure is repeated. If it is worse but *close enough* to the incumbent, then it is accepted as a temporary solution but the incumbent is not updated. The algorithm stops when it cannot improve any more.

As mentioned above, the incumbent solution is set to be an initial solution of the MIP (corresponding MIP to each model). Then, MIP is solved for a limited time using a commercial solver and an optimistic solution is obtained.

Base Case (BC) is the solution where all prices are set to maximum and all jobs are scheduled to the beginning of their time windows. The BC is set as the initial solution. Then, a random process *(Random Initialization)* is applied in order to find a different solution than BC since it is observed that sometimes it can be difficult for the rest of the algorithm to move on to a different solution from BC. In other words, BC may be a strong initial solution for Heuristic I. In this process, random price vectors are generated and the lower level problem is solved with these prices. Afterwards, the optimal prices corresponding to the lower level solution is computed by Inverse

Optimization. It is important to note that during this process, only prices *after* the peak slot are randomly generated whereas all prices before peak slot (including peak slot) are set to the upper bound, p_{max}. At the end of the process, the solution with the best net revenue (leader's objective) is taken as the initial solution.

- Step 1. Finding Peak: Two important values of a given solution are computed at this step; **peak load** (Γ) and the time slot that peak occurs (i), or in other words, **peak slot**.

- Step 2. Price Update: The price vector is updated with respect to the peak slot. The prices of k many time slots that follows peak slot that is found in the previous step are decreased by a certain discount factor d.

- Step 3. Schedule Computation: A new schedule x' that corresponds to the updated prices from the previous step is obtained by solving the lower level problem.

- Step 4. Inverse Optimization: Although the updated prices imply the schedule that is computed in the previous step, there can be better prices for the leader that correspond to the same schedule. Therefore, the optimal prices of the new schedule is obtained by solving the inverse optimization model.

- Step 5. Comparison: The objective function values of the price-schedule pair that is computed in the two previous steps is calculated. Then, it is compared to the incumbent solution. If it is better, incumbent is updated and the procedure is repeated. If it is worse but *close enough* to the incumbent, then it is accepted as a temporary solution but the incumbent is not updated.

- Step 6. Stopping Criterion: The algorithm stops when it cannot improve any more.

- Step 7. MIP Procedure: The incumbent solution is set to be an initial solution of the single level MIP formulation (corresponding MIP to each model). Then, MIP is solved for a limited time using a commercial solver and an optimistic solution is obtained.

4.5. Heuristic II

This method aims to find a price-schedule pair that has a high net revenue by fixing the peak load. It achieves this aim by performing a local search on peak load value. For each value, it computes a feasible schedule and corresponding optimal prices.

In order to perform a local search, it is necessary to find upper and lower bounds of peak load. The upper bound is provided by BC (Γ_U). In order to find the lower bound, MinPeak model is solved. The peak load value that is obtained by solving this problem *(Limiting Peak Load)* is the lower bound on peak load (Γ_L).

After computing the limits of peak load, an initial combing procedure is performed *(Combing Peak Interval)* within $[\Gamma_L, \Gamma_U]$ in order to obtain a smaller interval and ease the local search. The interval is divided into equal subintervals and f many values are obtained. With every fixed value Γ_i, the corresponding FixedPeak model is solved which gives a schedule with minimal inconvenience cost and a fixed peak load *(Schedule with Fixed Peak)*. Once there is a feasible schedule, it is possible to compute the corresponding optimal price vector with IO *(Inverse Optimization)*- if it exists. If a Γ_i is price infeasible, then any value below that is discarded. This procedure is repeated for all Γ_i. Then, the two Γ_i values that result in the two highest net revenue are chosen, and called Γ_a and Γ_b. The aim of *(Combing Peak Interval)* step is to narrow the initial interval $[\Gamma_L, \Gamma_U]$ to $[\Gamma_a, \Gamma_b]$ which would be the same in the worst case.

Next, a *(Binary Search)* is performed between Γ_a and Γ_b by solving FixedPeak model and obtaining corresponding optimal prices with IO.

While applying *(Binary Search)*, it is important which side is chosen at every iteration. Although the shape of net revenue curve is roughly (but not exactly) concave, staying on the correct side of the interval remains to be a challenge. Therefore, at each iteration j, two more peak values are evaluated, Γ_j^L and Γ_j^R, that are $c\%$ to the left and to the right of Γ_j, respectively. In other words, at every iteration j, for three fixed peak values (Γ_j^L, Γ_j, Γ_j^R), FixedPeak model is solved, the optimal prices are found by IO, and then net revenue is computed. The search interval and incumbent solution are updated accordingly.

The algorithm consists of the following steps:

- Step 1. <u>Limiting Peak Load</u>: At this step, upper and lower bounds of the peak load (Γ) is calculated. It is clear that BC easily provides an upper bound, and it is denoted as Γ_U. The lower bound (Γ_L) is obtained by solving the MinPeak model which is explained in Section 4.2.

- Step 2. Combing Peak Interval: A combing procedure is performed between Γ_L and Γ_U that are computed in the previous step to obtain a narrower interval for peak load. The interval is divided into equal subintervals and f many peak values (Γ_i) are obtained.

  - Step 2.1. Schedule with FixedPeak: With every fixed value Γ_i that are computed in the previous step, the corresponding FixedPeak model (Section 4.3) is solved in descending Γ_i order. The solution provides a schedule with minimal inconvenience cost and a fixed peak load.

  - Step 2.2. Inverse Optimization: The optimal prices corresponding to the schedule that is computed in the previous step are found by solving the inverse optimization model (Section 4.1), if they exist. If a Γ_i is price infeasible, then any value below that is discarded. This procedure is repeated for all Γ_i.

- Step 3. Narrowing the Interval: At the end of step 2, the two Γ_i values that result in the two highest net revenue are chosen, and called Γ_a and Γ_b.

- Step 4. Binary Search: A binary search is performed between Γ_a and Γ_b by solving FixedPeak model and obtaining corresponding optimal prices with IO. Incumbent solution is updated every time a better solution is found.

- Step 5. Stopping Criterion: The algorithm stops when it cannot improve any more.

- Step 6. MIP Procedure: The incumbent solution is set to be an initial solution of the single level MIP formulation (corresponding MIP to each model). Then, MIP is solved for a limited time using a commercial solver and an optimistic solution is obtained.

5. Experimental Results

In this section, the performance of heuristic procedures are presented in comparison to the classical exact method (CEM) for NMP, PMP and MMP models in terms of peak load, net revenue and computation time.

Since the difficulty levels of models are not the same, the instance sizes vary. For each scenario and parameter set, 10 instances are randomly generated, tested and averaged. For the experiments of NMP, there are 12 customers and each owns 5 nonpreemptive appliances. For PMP, there are 6 customers who owns 5 preemptive devices each. Finally, for MMP there are 7 customers and each one has 3 preemptive and 2 nonpreemptive appliances. All models are tested with peak penalty parameter (κ) for 5 different values: 200, 400, 600, 800 and 1000.

Although all customers are residential users, they may have different levels of sensitivity to delay and hence, they may behave differently. Therefore, a random inconvenience coefficient (λ_n) is generated for each customer n. As mentioned before, the inconvenience penalty function ($C_{n,a}(h)$) is directly proportional with λ_n and demand ($E_{n,a}$ for preemptive and $k_{n,a}l_{n,a}$ for non-preemptive) and inversely proportional with time window length. Hence, when λ_n takes a low value, it means customers are less sensitive to delays which gives the model more flexibility to find a better schedule. Mathematically, it means the solution space is larger when λ_n is small.

All problems are solved with CPLEX version 12.3 on a computer with 2.66 GHz Intel Xeon CPU and 4 GB ram, running on Windows 7 operating system. Time limit is set to 4 hours for CEM. For the instances that are not solved to optimality within this limit, the heuristic solutions are compared to the last found integer solution.

Table 1: Comparison of Heuristic Results of PMP to BC and CEM

	Av Peak Load				Av Net Revenue			
(κ)	BC	CEM	HI	HII	BC	CEM	HI	HII
200	65.4	56.1	55.8	56.1	51540	52358	52329	52358
400	65.4	51.0	51.0	51.0	38460	41612	41612	41612
600	65.4	47.1	47.5	47.1	25380	31898	31860	31898
800	65.4	45.0	45.0	45.5	12300	22786	22787	22758
1000	65.4	44.6	45.1	44.3	-780	13838	13648	13822

The average peak load and net revenue values of 10 randomly generated instances for PMP can be found in Table 1. The heuristic solutions are compared to CEM and BC. When we take the average of the averages for all values in Table 1, the heuristics' solutions are 0.31% and 0.05% away from the optimal net revenue for HI and HII, respectively. Besides, the peak load value is 0.22% and 0.07% away from optimal peak for HI and HII, respectively. It can be observed that both peak load and net revenue tend to decrease as κ increases.

Table 2: Computation Time of CEM and Heuristics for PMP (sec)

(κ)	CEM	HI	HII
200	31.9	21.7	13.5
400	410.4	102.9	74.7
600	2022.7	141.9	122.1
800	4244.2	154.0	152.2
1000	8717.7	153.6	154.0

The comparison of CEM and heuristic methods in terms of computation time is given in Table 2. The time limit on the exact method is 14400 seconds whereas it is 150 seconds for the MIP part of HI and HII. As κ increases, the problem takes more time to solve for all three methods. However, it is clear that the heuristics manage to produce considerably good solutions in much shorter time.

In order to highlight the performance of the algorithms, optimality gap of CEM after 150 seconds is compared to HI and HII. The gap of net revenue with respect to the optimal solution for different κ values can be found in Figure 3. Gap value increases as κ increases in all three curves. Note that at the end of both heuristics, the incumbent solution is given to CPLEX as an initial solution of MIP. This graphic shows that the algorithms make a difference by finding good initial solutions in a few seconds.

The performance comparison of heuristic methods to BC and CEM in terms of average net revenue and average peak load can be found in Table 3. When we take the average of the averages for all values in Table 3, the heuristics' solutions are 0.28% and 0.50% away from the optimal net revenue for HI and HII, respectively. Besides, the peak load value is 1.64% and 1.73% away from optimal peak for HI and HII, respectively. Both peak load and net revenue decrease as κ increases. However, NMP is not as sensitive to κ as PMP due to nonpreemptive property.

Computation time comparison of heuristics to CEM is given for NMP in Table 4. It can be observed that there is no direct relation between compu-

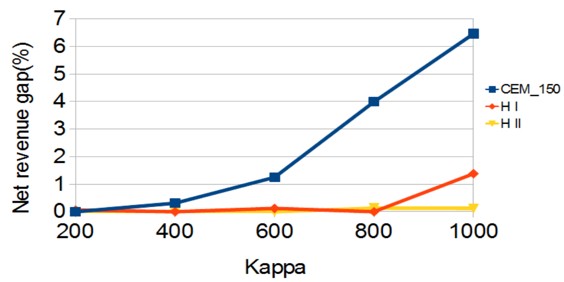


Figure 3: PMP model: Gap Comparison of CEM_{150}, HI and HII to OPT(%)

Table 3: Comparison of Heuristic Results of NMP to BC and CEM

	Av Peak Load				Av Net Revenue			
(κ)	BC	CEM	HI	HII	BC	CEM	HI	HII
200	263.2	151.6	158.8	158.0	217180	231872	231469	231459
400	263.2	146.4	148.1	148.7	164540	202232	201831	201735
600	263.2	145.1	146.6	146.6	111900	173140	172443	171638
800	263.2	143.6	144.7	145.5	59260	144212	143807	143722
1000	263.2	143.5	144.2	144.2	6620	115512	115106	114526

tation time and κ for NMP since manipulating a schedule of nonpreemptive devices is a difficult task i.e., even a small change might create a big perturbation.

Table 4: Computation Time of CEM and Heuristics for NMP (sec)

(κ)	CEM	HI	HII
200	9145.8	153.9	164.9
400	7745.8	153.7	165.2
600	4589.5	153.8	165.4
800	7118.8	153.7	166.5
1000	6182.9	153.7	162.6

The performance of heuristics in comparison to the solution that CEM finds within 150 seconds for NMP is given in Figure 4. Heuristic I provides

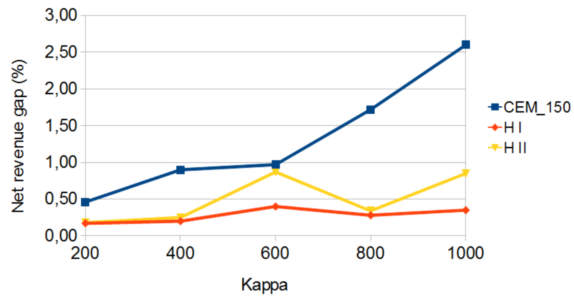


Figure 4: NMP model: Gap Comparison of CEM_{150}, HI and HII to OPT(%)

the best solutions for NMP, whereas both methods perform much better than CEM. This graph demonstrates that the heuristics manage to find solutions that are much closer to optimality within the same time limit.

Table 5: Comparison of Heuristic Results of MMP to BC and CEM

	Av Peak Load				Av Net Revenue			
(κ)	BC	CEM	HI	HII	BC	CEM	HI	HII
200	127.00	83.10	83.30	83.30	99230.00	106161.34	106148.71	106148.71
400	127.00	79.40	79.40	79.90	73830.00	89985.32	89967.83	89894.34
600	127.00	76.35	77.15	77.35	48430.00	74511.72	74403.48	74338.92
800	127.00	75.75	76.70	76.30	23030.00	59262.69	58767.87	59019.60
1000	127.00	74.95	75.35	75.35	-2370.00	44193.59	44050.94	44080.80

The average peak load and net revenue values of 10 random instances for MMP are given in Table 5. There are 7 customers with 3 preemptive and 2 nonpreemptive appliances (35 jobs in total) in these instances. The results of heuristics are compared to CEM and BC in terms of average peak load and average net revenue for different values of κ. When we take the average of the averages for all values in Table 5, the heuristics' solutions are 0.13% and 0.20% away from the optimal net revenue for HI and HII, respectively. Besides, the peak load value is 0.44% and 0.69% away from optimal peak for HI and HII, respectively. Similar to previous models, both peak load and net revenue decrease as κ increases.

Table 6: Computation Time of CEM and Heuristics for MMP (sec)

(κ)	CEM	HI	HII
200	105.60	69.70	82.00
400	884.40	120.50	132.90
600	2844.90	142.20	146.70
800	3487.40	143.40	147.10
1000	8348.30	150.40	148.20

Comparison of heuristics to CEM in terms of computation time is given in Table 6. Computation time tends to increase as κ increases for both heuristics and CEM. However, it can be observed that there is a large difference between these methods and CEM.

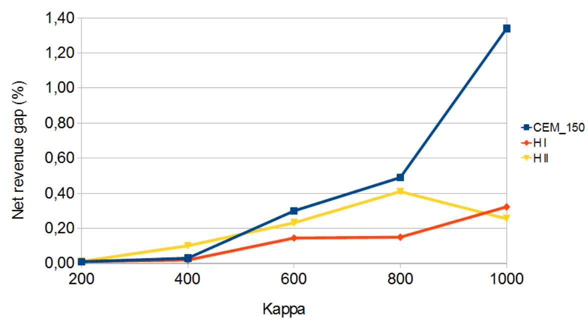


Figure 5: MMP model: Gap Comparison CEM_{150}, HI and HII to OPT(%)

In order to demonstrate the quality of the initial solutions that are provided by heuristics, optimality gap comparison of these solutions to CEM after 150 seconds for different κ values is given in Figure 5. Heuristic I provides best solutions for all κ values, and the performance of Heuristic II improves as κ increases.

In order to analyze the scalability of the algorithms, 10 larger instances are generated and tested. For PMP and MMP, these instances concern 20 customers each owning 5 appliances. In other words, the smart grid deals with 100 jobs that should be scheduled. For NMP, 10 random instances are generated with 40 customers each owning 5 appliances, which corresponds to 200 jobs. All instances are solved using CEM and two heuristics. Since it is not possible to solve these large instances to optimality within reasonable time, the same time limit as the heuristics (150 seconds) is applied to CEM. It is observed that CEM cannot even find one feasible solution whereas both heuristics provide solutions for PMP and MMP. Therefore, the heuristic results are compared to BC for these models. It is important to note that both heuristics provide a good starting point for the MIP and make it possible to find a solution within 150 seconds. It is further observed that CEM finds feasible solutions within 150 seconds for NMP, however it is observed that heuristic methods manage to find much better solutions within the same time limit.

The average net revenue and peak load comparisons of Heuristic I and II to the BC for PMP model are presented in Figure 6 and 7, respectively. Using heuristic methods, it is possible to increase the net revenue upto 50-60% in comparison to BC. As a general rule, Heuristic II provides considerably higher net revenue and lower peak load than Heuristic I for large instances within 150 seconds. Besides, both methods succeed at providing a feasible solution that constitutes a good starting point for the MIP.

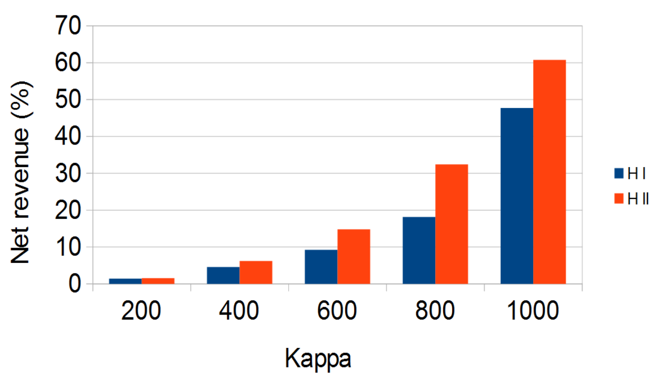


Figure 6: PMP model: Net Revenue Comparison of HI and HII to BC(%) for 100 jobs

The average net revenue and peak load comparisons of Heuristic I and II to the BC for MMP model can be found in Figure 8 and 9, respectively. Both heuristics manage to reach good solutions within 150 seconds, i.e., it is

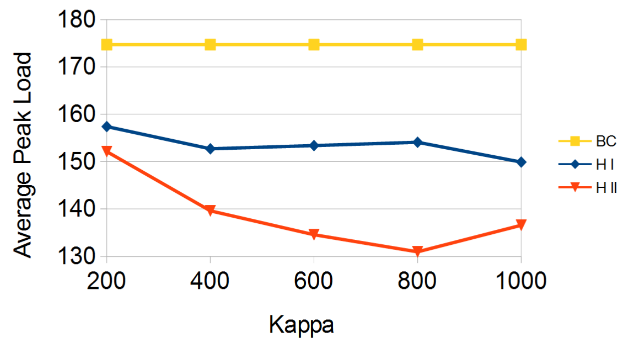


Figure 7: PMP model: Peak Load Comparison of HI and HII to BC for 100 jobs

possible to increase the net revenue upto 150% with respect to BC. Besides, their performances are very close to each other for MMP model. Heuristic II performs slightly better for higher κ values, whereas Heuristic I is better for lower ones. Again, both methods are useful for large instances where CEM takes very long time even to reach a feasible solution.

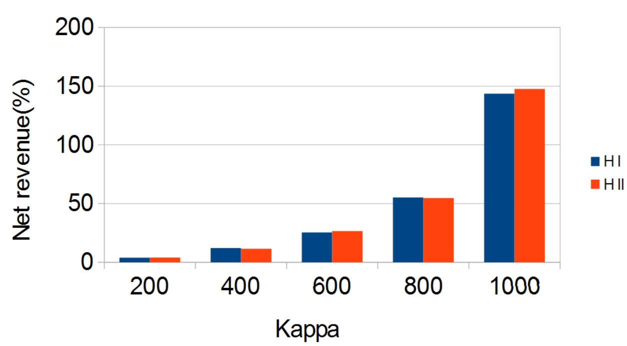


Figure 8: MMP model: Net Revenue Comparison of HI and HII to BC(%) for 100 jobs

For NMP, CEM can reach a feasible solution within 150 seconds, therefore the comparison of heuristics to CEM with respect to net revenue (%) and

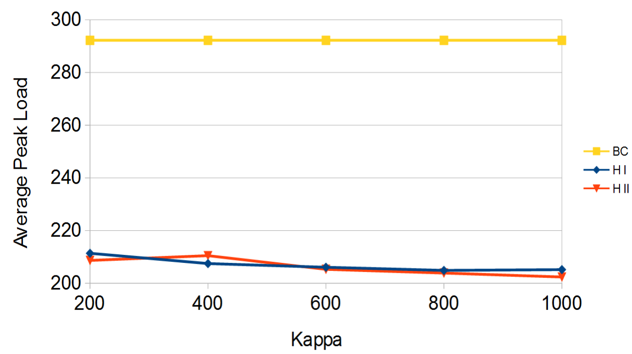


Figure 9: MMP model: Peak Load Comparison of HI and HII to BC for 100 jobs

peak load is given in Figure 10 and 11, respectively. It can be observed that heuristic results are far better than CEM in terms of both values. Heuristic I performs better for lower κ values whereas Heuristic II gives better results for highest value.

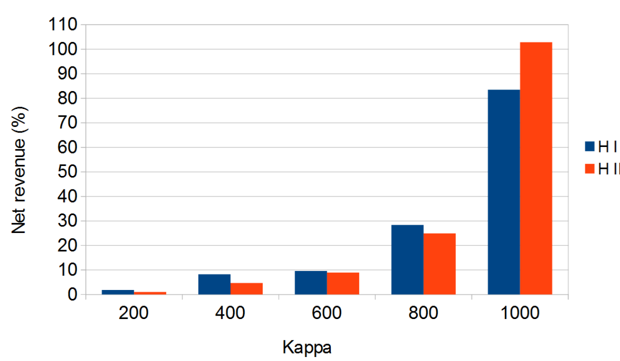


Figure 10: NMP model: Net Revenue Comparison of HI and HII to CEM_{150}(%) for 200 jobs

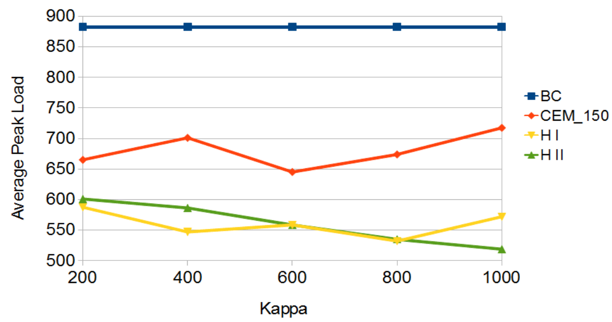


Figure 11: NMP model: Peak Load Comparison of HI and HII to BC and CEM_{150} for 200 jobs

6. Conclusion

In today's and future's energy market, smoothing out the supply curve and sustaining a supply-demand balance is an important problem. In order to provide a reliable energy supply, enterprises are usually compelled to keep large capacities which is an expensive and inefficient solution. In this paper, we focused on a residential energy peak minimization problem. Customers who own preemptive, nonpreemptive or both types of appliances, are interconnected via smart meters and they aim to minimize their total cost whereas the energy provider tries to find an optimal trade-off between revenue and peak cost in a hierarchical setting. Since it is highly time consuming to solve these bilevel models using classical methods, two efficient heuristics methods are developed and tested. Furthermore, it is shown that the bilevel models that are introduced cannot be reformulated as a MIP using the classical method. A more general theorem that suggests a reformulation of the assignment problem to address this issue is proved.

It is displayed that bilevel programming, when combined with day-ahead pricing strategy and smart meter technology, is a powerful way of modeling EPMP. It helps the firm to obtain a smooth supply curve and to keep a smaller generation capacity without forcing the customers to change their habits substantially. Moreover, it is demonstrated that high quality solutions can be found within a very short time for both small and large instances using the heuristic methods that are presented.

References

[1] Afşar, S., Brotcorne, L., Marcotte, P., Savard, G., 2015. Achieving an optimal trade-off between revenue and energy peak within a smart grid environment. Manuscript submitted for publication.

[2] Ahuja, R. K., Orlin, J. B., 2011. Inverse optimization. Operations Research 49 (35), 771–783.

[3] Bjørndal, M., Jørnsten, K., Nov. 2005. The Deregulated Electricity Market Viewed as a Bilevel Programming Problem. Journal of Global Optimization 33 (3), 465–475.

[4] Borenstein, S., Jaske, M., Rosenfeld, A., 2002. Dynamic pricing, advanced metering, and demand response in electricity markets. Center for the Study of Energy Markets.

[5] Brotcorne, L., Cirinei, F., Marcotte, P., Savard, G., May 2011. An exact algorithm for the network pricing problem. Discrete Optimization 8 (2), 246–258.

[6] Brotcorne, L., Labbé, M., Marcotte, P., Savard, G., 2000. A bilevel model and solution algorithm for a freight tariff-setting problem. Transportation Science 34 (3), 289–302.

[7] Brotcorne, L., Labbé, M., Marcotte, P., Savard, G., Nov. 2001. A Bilevel Model for Toll Optimization on a Multicommodity Transportation Network. Transportation Science 35 (4), 345–358.

[8] Brotcorne, L., Labbé, M., Marcotte, P., Savard, G., 2008. Joint design and pricing on a network. Operations Research 56, 1104–1115.

[9] Brotcorne, L., Marcotte, P., Savard, G., Nov. 2008. Bilevel Programming: The Montreal School. INFOR: Information Systems and Operational Research 46 (4), 231–246.

[10] Castelli, L., Labbé, M., Violin, A., Aug. 2013. A Network Pricing Formulation for the revenue maximization of European Air Navigation Service Providers. Transportation Research Part C: Emerging Technologies 33, 214–226.

[11] Colson, B., Marcotte, P., Savard, G., Jun. 2005. Bilevel programming: A survey. 4or 3 (2), 87–107.

[12] Dempe, S., Kalashnikov, V., Kalashnykova, N., 2006. Optimality conditions for bilevel programming problems. Vol. 2 of Springer Optimization and Its Applications. Springer US.

[13] Dempe, S., Lohse, S., 2006. Inverse linear programming. Lecture Notes in Economics and Mathematical Systems (563), 19–28.

[14] Dempe, S., Mordukhovich, B., Zemkoho, A., 2014. Necessary optimality conditions in pessimistic bilevel programming. A Journal of Mathematical Programming and Operations Research 63 (4), 505–533.

[15] Didi-Biha, M., Marcotte, P., Savard, G., 2006. Path-based formulations of a bilevel toll setting. Springer.

[16] Gellings, C. W., 1985. The concept of demand-side management for electric utilities. Proceedings of the IEEE 73 (10), 1468–1470.

[17] Gellings, C. W., Chamberlin, J. H., 1993. Demand Side Management: Concepts and Methods, 2nd Edition. PennWell Books, Tulsa, OK.

[18] Hansen, P., Jaumard, B., Savard, G., 1992. New branch-and-bound rules for linear bilevel programming. SIAM Journal on Scientific and Statistical Computing 13 (5), 1194–1217.

[19] Hobbs, B., Nelson, S., 1992. A nonlinear bilevel model for analysis of electric utility demand-side planning issues. Annals of Operations Research 34, 255–274.

[20] Hobbs, B. F., Metzler, C. B., Pang, J.-S., 2000. Strategic gaming analysis for electric power systems: an mpec approach. IEEE Transactions on Power Systems 15 (2), 638–645.

[21] Ibars, C., Giupponi, L., Carl, A., Gauss, F., 2010. Distributed demand management in smart grid with a congestion game. First IEEE International Conference on Smart Grid Communications (SmartGridComm), 495–500.

[22] Jeroslow, R., 1985. The polynomial hierarchy and a simple model for competitive analysis. Mathematical Programming 32, 146–164.

[23] Labbé, M., Marcotte, P., Savard, G., Dec. 1998. A Bilevel Model of Taxation and Its Application to Optimal Highway Pricing. Management Science 44 (12-Part-1), 1608–1622.

[24] Lujano-Rojas, J. M., Monteiro, C., Dufo-López, R., Bernal-Agustín, J. L., 2012. Optimum load management strategy for wind/diesel/battery hybrid power systems. Renewable Energy 44, 288–295.

[25] Marcotte, P., 1986. Network design problem with congestion effects: A case of bilevel programming. Mathematical Programming 74, 141–157.

[26] Masters, G. M., 2004. Renewable and Efficient Electric Power Systems. Wiley, Hoboken, NJ.

[27] Mohsenian-Rad, A.-H., Leon-Garcia, A., September 2010. Optimal residential load control with price prediction in real-time electricity pricing environments. IEEE Transactions on Smart Grid 1 (2), 120–133.

[28] Mohsenian-Rad, A.-H., Wong, V., Jatskevich, J., Schober, R., Leon-Garcia, A., 2010. Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid. IEEE Transactions on Smart Grid 1 (3), 320–331.

[29] Overbye, J., Weber, J. D., 1999. A two-level optimization problem for analysis of market bidding strategies. IEEE Power Engineering Society Summer Meeting 2, 682–687.

[30] Strbac, G., 2008. Demand side management: Benefits and challenges. Energy Policy 36 (12), 4419–4426.

[31] Weber, J. D., Overbye, T. J., 2002. An individual welfare maximization algorithm for electricity markets. Power 17 (3), 590–596.