Catching Cheats: Detecting Strategic Manipulation in Distributed Optimisation of Electric Vehicle Aggregators (Extended Abstract)*

Alvaro Perez-Diaz[1] , **Enrico H. Gerding**[1] and **Frank McGroarty**[2]
[1]Electronics and Computer Science, University of Southampton
[2]Southampton Business School, University of Southampton
{a.perez-diaz, eg, f.j.mcgroarty}@soton.ac.uk

Abstract

We consider a scenario where self-interested Electric Vehicle (EV) aggregators compete in the day-ahead electricity market in order to purchase the electricity needed to meet EV requirements. We propose a novel decentralised bidding coordination algorithm based on the Alternating Direction Method of Multipliers (ADMM). Our simulations using real market and driver data from Spain show that the algorithm is able to significantly reduce energy costs for all participants. Furthermore, we postulate that strategic manipulation by deviating agents is possible in decentralised algorithms like ADMM. Hence, we describe and analyse different possible attack vectors and propose a mathematical framework to quantify and detect manipulation. Our simulations show that our ADMM-based algorithm can be effectively disrupted by manipulative attacks achieving convergence to a different non-optimal solution which benefits the attacker. At the same time, our proposed manipulation detection algorithm achieves very high accuracy.

1 Introduction

To date, there exists a world-wide fleet of well over five million electric vehicles (EVs) and EV sales are growing exponentially [International Energy Agency, 2019]. Despite their great environmental benefits, EVs present a novel and heavy strain to electricity networks and markets, which will need to accommodate a new type of consumer with high demand [Rigas *et al.*, 2015]. This challenge is addressed by using EV aggregators: intermediaries between a fleet of EVs and the electricity grid and markets which are able to make informed charging decisions [Kempton *et al.*, 2001].

In this paper we focus on the interaction of a number of self-interested EV aggregators with the electricity day-ahead market, where increased electricity demand means increased prices, the so-called *price impact*. Here, reduced overall costs can be achieved by inter-aggregator coordination, producing more informed and optimised collective bidding. A centralised solution has been studied in [Perez-Diaz *et al.*, 2018a; Perez-Diaz *et al.*, 2018b]. However, these and other centralised approaches require a trusted environment and sharing private information with the central coordinator. In order to address these shortcomings, we propose a decentralised coordination algorithm using the Alternating Direction Method of Multipliers (ADMM) [Boyd *et al.*, 2010].

However, although this novel decentralised algorithm tackles the shortcoming described above, it introduces the possibility of strategic manipulation, where an aggregator deviates from the *vanilla* ADMM algorithm with the aim to decrease its energy costs, in detriment of other aggregators. We explore this issue by defining several attack vectors which seek to improve an aggregator's own energy allocation. Furthermore, in order to address this problem, we propose a manipulation detection algorithm that monitors the behaviour of the aggregators to identify deviations. Note that this issue exists in any ADMM decentralised optimisation scenario with rational and self-interested agents, and is not limited to our EV setting.

2 The Day-Ahead Market Model

Day-ahead markets divide each day into 24 hourly slots, each running a separate uniform-priced double-sided auction. Before closure time (usually noon) on a given day, bids and offers for each hourly slot of the next day must be submitted to the market. Then, a matching algorithm determines the accepted bids and offers, and establishes an hourly uniform price using marginal pricing, this is, the price of the intersection between supply and demand.

As a consequence, the arrival of a new buy order pushes the clearing price up if it gets accepted, the so-called price impact. This highlights the importance of eliminating unnecessary bid overlapping, *i.e.* spreading energy orders over time to avoid a high price impact at any given hour. Formally, if the bids an offers for a given hour t are known, we can compute the price impact function $\mathcal{P}_t$ relating an amount of extra energy big, E, with the resulting price: $\mathcal{P}_t(E)$.

3 The EV Aggregator Model

In our model, EVs arrive and depart dynamically over time. When an EV arrives to the charging point, it communicates the desired departure time and desired state of charge at departure to the aggregator. We assume that arrival time and

[*] This paper is an extended abstract of the following article published in JAIR: [Perez-Diaz *et al.*, 2020]

state of charge and can be automatically inferred by the aggregator. Each EV has a maximum charging speed in kW, which depends on two factors: the available physical infrastructure, and the EV's battery. These requirements will fully characterise the EV and will be used to constrain the aggregator bidding. The charging schedule of the EV is then left at the aggregator's discretion, which can choose when to perform the charging while guaranteeing the desired state of charge by departure time. This flexibility allows charging the battery in an informed way, rather than randomly, or at arrival, providing cheaper electricity costs.

The aim of the aggregator is to charge its EVs while minimising electricity costs. Recall that, due to the nature of the day-ahead market, electricity bids need to be placed in advance, before knowing one's real requirements. This requires the aggregators, and all market participants, to forecast their electricity needs and the hourly prices, and bid accordingly (we will denote forecasts by a hat). The aggregator's optimisation problem is then defined as follows: given an EV aggregator's forecasted energy requirements and price impact functions, find the optimal distribution of energy quantities to bid across the 24 hourly slots of the next day, $\mathbf{E} = (E_0, \ldots, E_{23})$, in order to satisfy its clients' charging needs while minimising the total cost of the purchased energy:

$$\min_{\{E_t\}} \sum_t \hat{\mathcal{P}}_t(E_t) \cdot E_t \tag{1}$$

$$\text{subject to: forecasted requirement constraints} \tag{2}$$

We refer to the full paper [Perez-Diaz *et al.*, 2020] for details.

4 The Decentralised Coordination Algorithm

The bidding algorithm detailed in the previous section for a single aggregator is the basis for the novel ADMM-based decentralised coordination algorithm proposed next. In essence, the goal is to produce an iterative decentralised algorithm, where each EV aggregator solves a local optimisation problem using only their own private information. The solutions to each local problem are coordinated by a global *consensus* step, and this procedure is iterated. *Consensus* refers to the fact that, asymptotically, all the local variables will coincide.

Following the model from Section 3, let $\mathbf{E^i} = \left(E_0^i, \ldots, E_{23}^i\right)$ be the energy schedule for aggregator i. Moreover, let $\mathbf{E} = \left(\mathbf{E}^1, \ldots, \mathbf{E}^n\right)$ be the joint vector encapsulating each individual energy schedule. We can now rewrite Eq. (1) as:

$$\min_{\mathbf{E}} \sum_{t=0}^{23} \left[\hat{\mathcal{P}}_t \left(\sum_{i=1}^{n} E_t^i \right) \cdot \sum_{i=1}^{n} E_t^i \right] =$$

$$= \min_{\mathbf{E}} \sum_{i=1}^{n} \left[\sum_{t=0}^{23} \left(E_t^i \cdot \hat{\mathcal{P}}_t \left(\sum_{j=1}^{n} E_t^j \right) \right) \right] \tag{3}$$

This way, the objective function is expressed as a sum of n terms, as required by the ADMM formulation. Note that, given that the price impact of each aggregator affects everybody else, we cannot separate Eq. (3) in the variable i, *i.e.* the equation is coupled and the sum's terms cannot be independently distributed among the aggregators. This type of problem is suited to be formulated as a *global variable consensus problem* [Boyd *et al.*, 2010], which works as follows. Consider a minimisation problem in the following form:

$$\min_{\mathbf{x}} \sum_{i=1}^{n} f_i(\mathbf{x})$$

where the goal is that each term in the sum can be handled independently. In the cases where the variable $\mathbf{x}$ is not separable in i, *local* variables $\mathbf{x}^i$ and a *global* variable $\mathbf{z}$ can be introduced, rewriting the problem as:

$$\min_{\{\mathbf{x}^i\}} \sum_{i=1}^{n} f_i(\mathbf{x}^i)$$

$$\text{subject to: } \mathbf{x}^i - \mathbf{z} = 0, \ \forall = 1, \ldots, n$$

As mentioned above, the problem constraints require all local variables to agree with each other and with the global variable. This way, global consensus on the solution is achieved. Also, note that f_i uses only aggregator i's individual constraints, which can be embedded into the function f_i itself.

In a similar vein and focusing on our scenario, let $\mathbf{E}$ and $\mathbf{E}^{(i)}$ be the global and local variables respectively, each of which comprises a vector with dimension $24n$ *i.e.* $\mathbf{E}^{(i)} = \left(\mathbf{E}^{(i),1}, \ldots, \mathbf{E}^{(i),n}\right)$ and $\mathbf{E}^{(i),j} = \left(E_0^{(i),j}, \ldots, E_{23}^{(i),j}\right)$. Following Eq. (3), the functions f_i are given by: $f_i\left(\mathbf{E}^{(i)}\right) = \sum_{t=0}^{23} \left[E_t^{(i),i} \cdot \hat{\mathcal{P}}_t \left(\sum_{j=1}^{n} E_t^{(i),j} \right) \right]$ if constraints (2) are met by $\mathbf{E}^{(i),i}$, and $f_i\left(\mathbf{E}^{(i)}\right) = \infty$ otherwise. This poses the problem in a way that can be iteratively solved by using the ADMM algorithm [Boyd *et al.*, 2010].

Given this, the iterative algorithm works as follows: at iteration k, each EV aggregator solves their *local* problem and updates their local copy of the energy schedule, $\mathbf{E}^{(\mathbf{i})}_{[\mathbf{k}]}$. Then, an aggregation step collects all the local solutions proposed by each aggregator and updates the global energy schedule, $\mathbf{E}_{[k]}$, reporting this vector back to all the aggregators. Lastly, each aggregator updates their local copy of the dual variable and proceeds to the new iteration. This procedure is iterated until every aggregator's local solutions are close enough.

5 Strategic Manipulation of the ADMM Algorithm

The ADMM-based algorithm described in the previous section asymptotically reaches the global optimum if every participating agent runs the algorithm faithfully. In our case, where agents are assumed to be self-interested, an aggregator could deviate from their assigned local algorithm and/or misreport their local solutions with the aim of improving their allocation. More specifically, in our scenario we assume a potential attacker aims to reduce its energy costs (*i.e.* increase its utility). Note that we do not look at all possible manipulation vectors, as this is not feasible, but instead focus on several intuitive and specific types of manipulation that are beneficial for the attacker in our setting.

Formally, the electricity costs incurred by aggregator i when a global allocation $\mathbf{E} = \left(\mathbf{E}^1, \ldots, \mathbf{E}^n\right)$ is reached are given by:

$$\text{cost}_i = \sum_{t=0}^{23} \left[E_t^i \cdot \hat{\mathcal{P}}_t \left(\sum_{j=1}^{n} E_t^j \right) \right] \qquad (4)$$

In order to reduce these costs, the attacker aims to minimise the price impact on their desired hours, which in turn can be achieved by moving other aggregators' overlapping allocations to different hours. To this end, we consider different attack vectors, namely *Shift*, *Proportional*, *Freeze*, *FreezeShift*, *FreezeProp* and *Adversarial* attacks, as summarised in Table 1. Each attack has an attack *strength* that can be varied. For *FreezeProp*, it is denoted by λ. This will be discussed empirically in Section 7. Also, we consider *All-* variants for the *Shift*, *Proportional*, *FreezeShift* and *FreezeProp* attacks, where all the other aggregators—not just one—are attacked. Finally, note that we assume that an attacker performs a given attack vector with a given intensity in every round.

6 Detecting Manipulation

In this section, we detail a mathematical framework for quantifying the influence of a given ADMM participant, *i.e.* an EV aggregator, onto the rest of participants. The aim is to be able to detect outliers that are symptom of strategic manipulation in the system. Although this framework is general, and can be applied to any ADMM (or variant) scenario, we focus on our particular case for ease of exposition.

The basic idea is that any group of aggregators with overlapping energy requirements should influence each other's schedules with *similar* intensity. If a particular aggregator i is self-interested and wants to improve its allocation by deviating from the ADMM algorithm, it will exert a heavier influence onto its competitors' allocations. Conversely, as happens in the adversarial attack, an aggregator that tries to wrongly flag another benign aggregator as deviator would exert too little influence.

A key point is that each aggregator i produces a (local) proposed schedule for all the n participating aggregators. Formally, focusing on the algorithm's iteration number k:

$$\mathbf{E}_{[k]}^{(i)} = \left(\mathbf{E}_{[k]}^{(i),1}, \dots, \mathbf{E}_{[k]}^{(i),n} \right)$$

Hence, this local solution proposed by aggregator i at iteration k contains its own schedule, $\mathbf{E}_{[k]}^{(i),i}$, and all the schedules for all the other participants, $\mathbf{E}_{[k]}^{(i),j}$ for $j \neq i$. We assume that each aggregator, benign or deviator, is truthful about their own allocations in their proposed local solutions and that the deviating behaviour starts from the second ADMM round, when every aggregator has seen the proposals from each other aggregator. This allows us to focus on the first two iterations ($k = 0, 1$) for ease of exposition.

Attack name	Short description
Shift	Shift the proposed allocation for one attacked aggregator to more expensive hours
Proportional	Scale down the proposed allocation for the attacked aggregator
Freeze	Propose its individually optimal allocation for itself (without considering the competitor aggregators)
FreezeShift	Freeze + Shift
FreezeProp	Freeze + Proportional
Adversarial	Attempt to incriminate a benign aggregator as deviator by artifically favouring their allocations

Table 1: Summary of the proposed attack vectors. In addition, *Shift*, *Proportional*, *FreezeShift* and *FreezeProp* have *All-* variants where all the other aggregators—not just one—are attacked.

Now, in order to study the influences among different aggregators, we define the *difference matrix*, a square matrix of dimension n storing how much each aggregator affects its competitors' self-proposed allocations:

$$d^{i,j} = \| E_{[1]}^{(i),j} - E_{[0]}^{(j),j} \|$$

Moreover, we normalise this matrix to account for the natural differences due to the size of the different aggregators, using the total amount of energy allocated by each aggregator to itself as a proxy to potentially unknown aggregator size. Formally, we can write: $\text{size}_i = \sum_{t=0}^{23} E_{[0]}^{(i),i}$ and the proportion of the size of aggregator i among the whole group of aggregators is given by: $p_i = \dfrac{\text{size}_i}{\sum_j \text{size}_j}$. Then, the *normalised difference matrix*, $\bar{d}$, is given by:

$$\bar{d}^{i,j} = \| E_{[1]}^{(i),j} - E_{[0]}^{(j),j} \| \cdot \frac{\sqrt{p_i}}{\text{size}_i + \text{size}_j} \qquad (5)$$

This scaling function was chosen as it empirically *flattens* the entries of the matrix $\bar{d}$ corresponding to benign aggregators, eliminating most of the dependence on aggregator size.

Lastly, we assume that $n - 1$ aggregators are benign and only one of them can potentially be a deviator. This is motivated by the fact that, with a perfect detection algorithm, there exists a Nash equilibrium in which no-one wants to deviate. Note that the proposed detection algorithm, which we are now ready to introduce, could be extended to deal with the more general case of having any number of deviators.

We are now ready to introduce our threshold-based manipulation detection algorithm, which makes use of the normalised difference matrix. In more detail, the algorithm looks at the difference matrix $\bar{d}$, computes the median of the matrix entries, $\mu_{1/2}$, and then finds the entry that deviates the most from the median. This is done separately for off- and on-diagonal elements (as there are intrinsic magnitude differences between $\bar{d}^{i,i}$ and $\bar{d}^{i,j}$ even when all aggregators are benign) and only the highest deviation of the two is taken as final candidate. Lastly, this candidate is classified as deviator if its deviation from the median is greater than the user-defined threshold α.

The choice of threshold α is critical and will be studied empirically in Section 7. Also, although the presented algorithm is designed to work in scenarios with at most one manipulating agent, by selecting the aggregator that deviates the

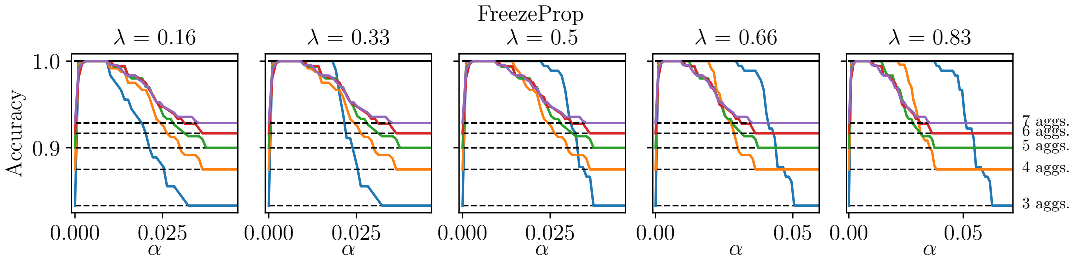

Figure 1: Accuracy analysis for the *FreezeProp* attack vector under different attack strengths (λ) and number of aggregators. Results averaged over every day of November 2016. All aggregators have capacity for 150 000 EVs. Dashed lines represent the naive benchmark for each scenario, which considers every aggregator to be benign.

most, it can be easily adapted to a general scenario. The most straightforward way would be to simply classify as deviator any aggregator i with $|\mu_{1/2} - \bar{d}^{i,j}| > \alpha$ for some j. Finally, we would like to note that the proposed algorithm could be used in conjunction with other detection methods in order to provide better results.

7 Empirical Evaluation

In this section we present a summarised analysis of the performance of the different attack vectors (Section 5) and the manipulation detection framework (Section 6). This empirical evaluation uses real market and vehicle usage data from Spain. Full details and extra results are available in [Perez-Diaz *et al.*, 2020].

As discussed throughout the paper, a deviating aggregator would try own allocation, that is, artificially reduce its energy costs, by manipulating the algorithm in a subtle way that goes unnoticed. Consequently, we identify two key quantities to analyse in order to assess the efficacy of a given attack vector: the effectiveness of the considered attack, in terms of utility increase (*i.e.* energy cost reduction) for the attacker, and the convergence of the algorithm under attack. More specifically, by convergence we refer to whether the iterative ADMM algorithm converges to a global solution.

The performance of the different attack vectors considered differs is quite varied. For example, *FreezeProp* present very good results, consistently providing reduced costs for the attacker and close to 100% convergence rates for all attack strengths. On the other hand *Shift* presents very large cost reductions but fails to provide algorithm convergence. These results are depicted in Fig. 2. We would like to note that even a 1% cost reduction in the scenarios considered in this work represents savings in the order of hundreds of thousands of Euros per year.

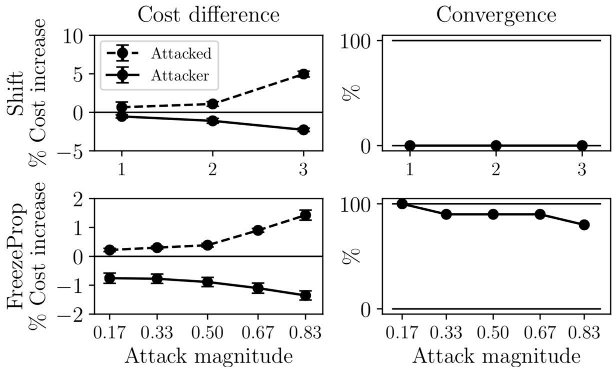

Figure 2: Cost and convergence analysis for the *Shift* and *FreezeProp* attack vectors. Scenarios with four aggregators and averaged over the ten first days of November 2016. All aggregators have capacity for 150 000 EVs.

Finally, in order to assess the efficacy of the manipulation detection algorithm, we turn our attention to its *accuracy* [Metz, 1978]. In more detail, accuracies 0 and 1 correspond to detectors which are always wrong and always right, respectively. Moreover, we will compare the accuracy of our proposed algorithm with that of a naive detector that classifies all the aggregators as benign. This is motivated by the fact that our dataset is unbalanced given that we consider at most one deviator per simulation. A well performing algorithm should present increased accuracy from this naive benchmark.

Here, we will focus on the *FreezeProp* attack, which has shown to be effective at successfully manipulating the coordination algorithm. The detection results are shown in Fig. 1. We can see that scenarios with larger number of aggregators are more difficult and that stronger attacks are easier to detect. The proposed algorithm significantly outperforms the naive benchmark and is able to achieve very high accuracy for a range of α values.

8 Conclusion

In this paper, we have presented a decentralised coordination mechanism for multi-EV aggregator bidding in the day-ahead market which employs the Alternating Direction Method of Multipliers (ADMM) algorithm. This proposed algorithm extends previous works in the literature with a focus on privacy.

Moreover, we present the first study of strategic manipulation of ADMM algorithms by self-interested internal agents. Furthermore, working towards resilient decentralised optimisation, we study how deviating behaviour can be detected and propose a mathematical framework for detecting strategic manipulation.

Empirical results in a realistic setting show that a deviating aggregator is able to successfully manipulate the coordination mechanism, significantly increasing their utility. Finally, our proposed detection algorithm clearly outperforms a naive benchmark and is able to detect manipulation with very high accuracies.

References

[Boyd *et al.*, 2010] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2010.

[International Energy Agency, 2019] International Energy Agency. Global EV Outlook 2019: Scaling-up the transition to electric mobility. https://www.iea.org/reports/global-ev-outlook-2019. 2019. Last accessed on 05/05/2020.

[Kempton *et al.*, 2001] Willett Kempton, Jasna Tomic, Steven Letendre, Alec Brooks, and Timothy Lipman. Vehicle-to-Grid Power: Battery, Hybrid, and Fuel Cell Vehicles as Resources for Distributed Electric Power in California. *Fuel Cell*, IUCD-ITS-R(June):95, 2001.

[Metz, 1978] C E Metz. Basic principles of ROC analysis. *Seminars in Nuclear Medicine*, 8(4):283–298, 1978.

[Perez-Diaz *et al.*, 2018a] Alvaro Perez-Diaz, Enrico Gerding, and Frank McGroarty. Coordination and Payment Mechanisms for Electric Vehicle Aggregators. *Applied Energy*, 212:185–195, 2018.

[Perez-Diaz *et al.*, 2018b] Alvaro Perez-Diaz, Enrico Gerding, and Frank McGroarty. Coordination of Electric Vehicle Aggregators: A Coalitional Approach. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018)*, pages 676–684, 2018.

[Perez-Diaz *et al.*, 2020] Alvaro Perez-Diaz, Enrico Gerding, and Frank McGroarty. Catching Cheats: Detecting Strategic Manipulation in Distributed Optimisation of Electric Vehicle Aggregators. *Journal of Artificial Intelligence Research*, 67:437–470, 2020.

[Rigas *et al.*, 2015] Emmanouil S. Rigas, Sarvapali D. Ramchurn, and Nick Bassiliades. Managing Electric Vehicles in the Smart Grid Using Artificial Intelligence: A Survey. *IEEE Transactions on Intelligent Transportation Systems*, 16(4):1619–1635, 2015.