

A fixed-structure automaton for load management of electric vehicles

S. Stüdli, R. H. Middleton, and J. H. Braslavsky

Abstract—The uncontrolled charging of electric vehicles (EVs) imposes additional stresses on the grid. These stresses are set to increase due to the predicted increase in penetration of EVs. However, EV charging loads offer opportunities for controlling the actual demand to limit the peak demand or track variable generation to support the grid. This also can facilitate the integration of intermittent renewable energy generation in the near future. In this paper a learning automaton is proposed to manage EVs capable of on/off charging. We propose a new algorithm for distributed control of charging based on the broadcast of a congestion signal to regulate the aggregated demand. We show that the proposed algorithm converges to steady state operation, and analyse its implications on the distribution of power demand amongst the EVs. The potential of the proposed algorithm is illustrated by simulations for capping the aggregate demand of the EVs, and for tracking of slowly varying power generation signals.

I. INTRODUCTION

Growing concerns about greenhouse gases drive the deployment of renewable energy generation. In particular, wind and solar power are expected to become a significant part of the energy mix in many countries. For example, Ireland aims to produce 40% of their energy from renewables by 2025, of which 86% are planned to be wind generation [1]. The electricity generated by such plants fluctuates with weather conditions, which imposes new challenges on the grid. Load management is regarded as a key strategy to deal with such variability by controlling demand to match fluctuations in electricity generation, an application referred to as load tracking [1]–[3]. In addition, load management has the potential to reduce generation costs and transmission losses by reducing the demand of controllable loads at peak times, an application referred to as peak shaving [3].

It is well recognised that electric vehicles, among other controllable loads, are able to provide such services to the grid [2], [3]. This capability arises due to their expected high penetration levels [4]–[8] and their high flexibility in charging time. For example, [9] assumes that EVs are parked for 90% of the time. Hence, various algorithms have been developed to manage and control their demand. For instance [4], [5], [10], [11] developed algorithms to schedule the charging of electric vehicles. While [12], [13] went further and allowed the electric vehicles to inject power into the grid during peak demands. [9] also allowed reactive power balancing by the EVs.

S. Stüdli and R. Middleton are with the Centre for Complex Dynamic Systems and Control, School of Electrical Engineering and Computer Science, The University of Newcastle, Callaghan NSW 2308, Australia.

J. H. Braslavsky is with the Commonwealth Scientific and Industrial Research Organisation (CSIRO), Energy technology, 10 Murray Dwyer Circuit, Mayfield West 2304, Australia.

This work was supported in part by SFI under grant 11/PI/1177.

This paper expands on previous work of the authors [14], [15] and suggests a new learning automata game for distributed control of charging demand based on a congestion control algorithm. In previous works, our focus lay on load management using EVs with continuously adaptable charge rates. In the interest of incremental deployment and in regard to other loads, we believe it is important to employ load management with minimal requirements for the electric vehicles. In this context, we limit the abilities of the EV chargers to be only capable of on/off operation, namely, between a maximum charging rate when on, and not charging when off (the effects of on/off constraints on EV charging are investigated in [10]).

While other studies rely on day-ahead forecasts [4], [5], pricing signals [4], [12], [16], or employ a centralised control unit [5], [6], in this paper we suggest a distributed algorithm that shows reasonable performance without requiring any forecasts. We acknowledge that customers need to be encouraged to participate in load management programs, and that this encouragement can take the form of electricity price incentives. The control method proposed in this paper does not rely directly on such pricing signals or incentives. However, it may give indirect financial benefits to the users by constraining the power available to the EVs when price is high during peak demand, shifting charging to off-peak.

The proposed control strategy relies on the broadcast of a congestion signal that places a cap on the aggregate demand of the EVs. The problem considered is defined in Section II. Section III explains the automaton proposed as a distributed algorithm for EV charging. This algorithm is analysed in Section IV, and shown to converge to stationary distribution independent of the chosen starting distribution and the parameters. This analysis is performed by interpreting the algorithm as a Markov Chain and then showing its ergodicity. The Matlab simulation examples in Section V illustrate the promising performance of the approach for controlling aggregate demand to track constant and slowly varying reference signals, as applied in peak shaving and load tracking scenarios. Finally, Section VI presents some concluding remarks and discussion of further research questions.

II. PROBLEM DESCRIPTION

We consider two main scenarios for load management. The first is peak shaving, where the demand of the controlled loads is reduced during periods of high total demand. This in essence encompasses a defined period of time during which the total load should be capped. An example of this scenario is the imposition of a limit to the maximum aggregate power

consumption of EVs during the daily evening demand peak. The second scenario considered is load tracking, where the controllable loads track an externally specified varying power level. This latter scenario could be useful in situations such as where a group of EVs is set to charge their batteries with rates matching the output of a wind power plant.

In both of these scenarios the aim is to share a known limited power $\bar{P}$ among a group of loads. In the case of peak shaving $\bar{P}$ is a constant power cap during the peak time, while for load tracking the available power is the varying output of a variable power output generator. Note also that in the first case the aim is to be as close to the cap as possible to minimise impacts to the customers.

In this initial work we make two simplifying assumptions to focus the analysis on the fundamental properties of the proposed algorithm. Firstly, we neglect any deadlines for the EV charging. Whilst deadlines are not always necessary, there are important cases where neglecting deadlines can lead to undesired consequences, such as customers without enough charge. Secondly, we assume that the available power and the aggregated demand of the EVs are known by the infrastructure operator without errors. The relaxation of these assumptions and the consideration of other practical constraints and robustness issues are left for future work.

In contrast to previous work [14], [15], here the EV chargers have binary (on/off) control. Although in practice the power drawn might not be constant during the complete charging cycle, it will be subject to slow changes and remain constant during a long period. For simplicity therefore, we take ‘on’ as equivalent to a fixed power consumption, whilst off is taken as zero power drain. In practice, there will be a small fractional power overhead when the charger is off.

In the following, $\bar{p}_i$ refers to the power consumption by EV i if turned on and $p_i(t)$ refers to its actual power consumption at time t . Therefore the power limiting constraint can be written

$$\sum_i p_i(t) \leq \bar{P}(t) \quad \forall t, \quad (1)$$

and the objective is set to maximise the power delivered to the participating EVs without violating this constraint.

III. AUTOMATON

Considering the similarities of the above load managing problems and problems encountered in communication networks [17], the authors in [14], [15] previously suggested the use of an AIMD (additive increase/multiplicative decrease) algorithm for managing the load of EVs. Based on this idea, we suggest here an automata game [18], to cope with EVs with discrete charging, in that their demand can assume only one of two possible states: on or off. Each EV repeatedly cycles the execution of a series of commands according to a broadcast signal, by the infrastructure operator, and then waits for the next cycle. Depending on whether a signal has been received or not, we identify two phases: turn-on, when no signal has been received, and turn-off, when a broadcast signal has been received. In comparison with the AIMD algorithm, the turn-on phase corresponds to additive increase.

The agent remains in this mode as long as no capacity signal is received. The turn-off phase is entered upon receipt of a signal and corresponds to the multiplicative decrease phase of the AIMD algorithm. A complete flow chart of the proposed charging algorithm is shown in Figure 1.

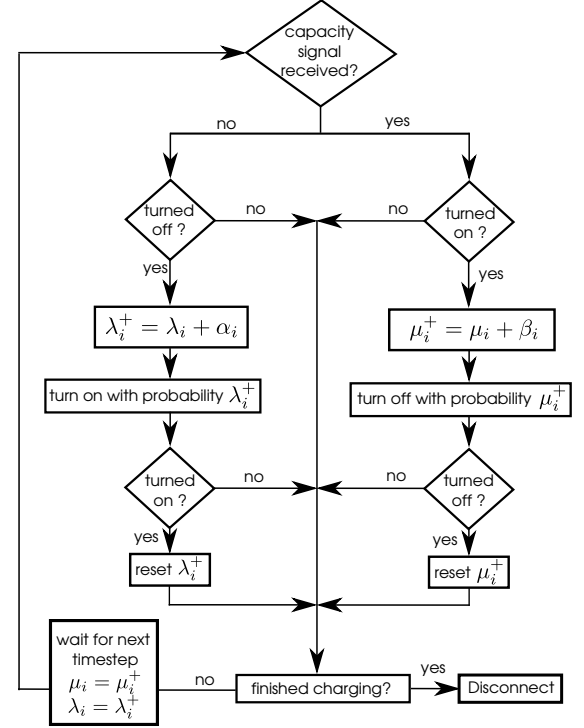


Fig. 1: Diagram depicting the algorithm executed by each agent.

The turn-on phase only affects EVs that are currently off. Two variables are associated with this phase: a probability $\lambda_i \in [0, 1]$, and an additive factor $\alpha_i \in [0, 1]$. During this phase the EV increases its probability of turning on λ_i by adding α_i giving λ_i^+ . Afterwards, it turns on with probability λ_i^+ , or stays off. When the EV turns on, λ_i^+ is reset to 0.

Similarly, the turn-off phase only influences EVs currently turned on. Associated with this phase, there is a probability μ_i and a second additive factor β_i . In the same way as for the turn-on phase, the EV determines stochastically whether to turn off using the increased probability $\mu_i^+ = \mu_i + \beta_i$. In case the agent turns off, it resets its μ_i^+ to 0.

At a final step each EV checks whether it is fully charged. If charging is incomplete, it waits for the next time step for which it sets $\lambda_i = \lambda_i^+$ and $\mu_i = \mu_i^+$.

IV. SYSTEM ANALYSIS

In this section, we first analyse the behaviour of a single agent (EV) during both phases. From this analysis it becomes clear that the algorithm implements a fixed-structure automaton and that the overall system represents a game between multiple automata, as defined in [18]. Using this fact, we are able to interpret the overall system as a Markov Chain for which we show ergodicity. This means that the system converges to a steady state distribution independent of the

initial conditions. This is particularly relevant to examining simulation results, since it is important to know that the simulation represents on average the behaviour of the system and it does not depend on the initial states of the EVs. For example, it obviates any need for Monte Carlo simulations.

A. Behaviour of a single EV

Let n_i and m_i be integer values larger than 1 and choose $\alpha_i = \frac{1}{n_i}$ and $\beta_i = \frac{1}{m_i}$. In this case $\lambda_i = k\alpha_i$, where k is an integer in the interval $[0, n_i - 1]$ and similarly $\mu_i = l\beta_i$, where l is an integer in the interval $[0, m_i - 1]$. Further, it is clear that at all times at least one of the probabilities λ_i or μ_i is 0. In particular, whenever the agent is on, λ_i is set to 0 (that is the probability of changing to the on state is zero). Conversely, when the agent is off, μ_i is set to zero. This means that agent i can occupy $n_i + m_i$ states. In the following, $\beta^{(l)}$ denotes the state where the agent is turned on and its internal values are $\mu_i = l\beta_i$ and $\lambda_i = 0$. Analogously, $\alpha^{(k)}$ denotes the states where the agent is turned off and its internal values are $\lambda_i = k\alpha_i$ and $\mu_i = 0$.

The transitions between the states depend on whether the overall system is *congested*, namely, whether the overall power demand is too high, or not. Hence, the behaviour of the agent is identical to a fixed-structure automaton, as defined in [18], where the input is the congestion/no congestion signal, the state set is the collection of $n_i + m_i$ states above, and the action is the power demanded by the EV. Figure 2 depicts the automaton for the two possible inputs from the environment: congested or not congested. The action of the automaton is $\bar{p}_i$ for the states $\beta^{(l)}$ with $l \in [0, m_i - 1]$ and 0 for the remaining states.

In the following let $\mathbf{I}_n$ denote the identity matrix of size $n \times n$ and $\mathbf{0}_{n \times m}$ the $n \times m$ matrix containing all zeros. Define a vector z such that z_k is the probability that EV i is in state $\alpha^{(k)}$ if $k \leq n_i$ and in state $\beta^{(k-n_i)}$ if $k > n_i$. Then z evolves according to the transition matrix

$$\bar{A}_i = \begin{bmatrix} \bar{B}_i & \bar{C}_i \\ \mathbf{0}_{m_i \times n_i} & \mathbf{I}_{m_i} \end{bmatrix} \quad (2)$$

with $\bar{B}_i$ is a matrix of zeros with the first off diagonal above the main diagonal taking the values $1 - \alpha_i, 1 - 2\alpha_i, \dots, 1 - (n_i - 1)\alpha_i$ and $\bar{C}_i$ a zero matrix with the first column taking the values $\alpha_i, 2\alpha_i, \dots, 1$, if the system is not congested.

In the case of congestion the transition matrix becomes

$$\hat{A}_i = \begin{bmatrix} \mathbf{I}_{n_i} & \mathbf{0}_{n_i \times m_i} \\ \hat{C}_i & \hat{B}_i \end{bmatrix} \quad (3)$$

with the matrices identical as before but now taking the values $1 - \beta_i, 1 - 2\beta_i, \dots, 1 - (m_i - 1)\beta_i$ and $\beta_i, 2\beta_i, \dots, 1$, respectively. Note that the above matrices are both row stochastic.

B. Game between N automata

As the behaviour of the single EV is identical to a fixed-structure automaton, the system with N participating EVs is a game between such automata, where the output of the environment depends deterministically on the actions of each

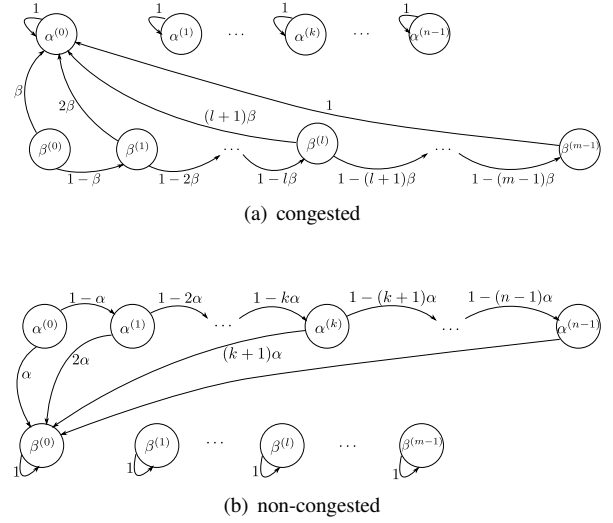


Fig. 2: The transition graph of a single agent's behaviour.

automaton. As in [18], it is therefore possible to describe the complete game by a Markov Chain. First, we consider the simple case where there is only one EV. We assume that whenever this EV is on, the system is congested.

Let I_A be a diagonal matrix of size $(n + m) \times (n + m)$ containing 1 if the state is congested and 0 otherwise, i.e.

$$I_A = \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{I}_m \end{bmatrix}. \quad (4)$$

From that the transition matrix can be constructed by

$$A = (\mathbf{I}_{n+m} - I_A)\bar{A} + I_A\hat{A}, \quad (5)$$

which simplifies to

$$A = \begin{bmatrix} \bar{B} & \bar{C} \\ \hat{C} & \hat{B} \end{bmatrix}. \quad (6)$$

The Markov Chain associated with the single EV system is then

$$x(\tau) = x(\tau - 1)A = x(\tau - 1) \begin{bmatrix} \bar{B} & \bar{C} \\ \hat{C} & \hat{B} \end{bmatrix}. \quad (7)$$

Next, consider the system where N EVs participate. From [18], we know that the states of the Markov Chain are all combinations of the states the single automata can be in, so the total number of states is $\prod_{i=1}^N (n_i + m_i)$. Each of those states has a defined power consumption and hence defines whether the EVs receive a congestion signal or not depending on the available power. For instance $[\alpha^{(0)}, \beta^{(0)}, \alpha^{(0)}]$ consumes in total $\sum_i p_i(t) = \bar{p}_2$, as only agent 2 is on.

Let I_A be again a diagonal matrix, containing 1 if the state is congested and 0 otherwise, then the row stochastic transition matrix of the complete system with N EVs is

$$A = (\mathbf{I} - I_A)(\bar{A}_1 \otimes \dots \otimes \bar{A}_N) + I_A(\hat{A}_1 \otimes \dots \otimes \hat{A}_N), \quad (8)$$

where $\otimes$ denotes the Kronecker product.

Example 1: Assume that two EVs are participating in the load management procedure and that the system is congested only when both EVs are on. Then the transition matrix is

$$A = (\mathbf{I} - I_A)(\bar{A}_1 \otimes \bar{A}_2) + I_A(\hat{A}_1 \otimes \hat{A}_2), \quad (9)$$

where $I_A = v_1 \otimes v_2$ and

$$v_i = \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{I}_{m_i} \end{bmatrix}. \quad (10)$$

C. Ergodicity of the Markov Chain

We now show that the Markov Chain derived above is ergodic independent of the number of participating agents and their parameters.

Theorem 1: If N EVs are participating and the following assumptions are true

- 1) the states $[\beta^{(k_1)}, \beta^{(k_2)}, \dots, \beta^{(k_N)}]$ for any existent k_i give rise to congestion, and
- 2) the states $[\alpha^{(k_1)}, \alpha^{(k_2)}, \dots, \alpha^{(k_N)}]$ for any feasible k_i are not congested,

then the Markov Chain associated with the system defined in (8) is ergodic.

Note that the steady state of the system can be trivially computed given the initial conditions if one of the above assumptions is violated. The proof of Theorem 1 follows directly from Lemma 3, Lemma 4, and Theorem 2, below.

Theorem 2 (Ergodic Theorem [19]): If the stochastic matrix A is regular, then

$$A^\tau \rightarrow \mathbf{1}y \quad (11)$$

where y is the stochastic eigenvector of A belonging to the eigenvalue 1.

Hartfiel [19] defines a stochastic matrix to be *regular* if it corresponds to precisely one essential class of vertices, and the sub-matrix corresponding to this essential class, A_1 in the canonical form, is primitive. This means that there exists a k such that every vertex is reachable from every vertex in exactly τ steps.

Fact 1: The states $[\alpha^{(0)}, \dots, \alpha^{(0)}]$ and $[\beta^{(0)}, \dots, \beta^{(0)}]$ are reachable from any other state.

Proof: Every time an EV turns off it goes into the state $\alpha^{(0)}$, similarly whenever an EV turns on it reaches the state $\beta^{(0)}$. Further, we know that the agent stays in state $\alpha^{(0)}$ as long as congestion events occur, otherwise it enters state $\alpha^{(1)}$. Also, the EV stays in $\beta^{(0)}$ until a congestion event arises, upon which it switches to state $\beta^{(1)}$. Below, the proof of reachability is given for the state $[\alpha^{(0)}, \dots, \alpha^{(0)}]$, as the proof for state $[\beta^{(0)}, \dots, \beta^{(0)}]$ is analogous.

First, choose any congested state j . From this state, there is a non-zero probability that all agents turn off. This state is clearly not congested as all the agents are off. Therefore, there exists a positive probability that all the agents turn on in the next time step. This leads to the state $[\beta^{(0)}, \dots, \beta^{(0)}]$, which is congested, according to our assumption. From this state the probability to reach $[\alpha^{(0)}, \dots, \alpha^{(0)}]$ is positive, showing that this state is reachable from any congested state.

Secondly, choose any non-congested state j . The probability that all remaining states turn on in the next time step is larger than 0. This leads to a congested state, where all agents are turned on. Hence the possibility exists that all agents turn off at the same time, leading to the state $[\alpha^{(0)}, \dots, \alpha^{(0)}]$. Thus the state $[\alpha^{(0)}, \dots, \alpha^{(0)}]$ is reachable from any non congested state and the fact is proven. ■

From the above results we derive the following Lemma.

Lemma 3: The Markov Chain associated with the system from Theorem 1 contains only one essential class.

Proof: Assume there are two essential classes and let's take one vertex from each class, vertex i and j . Both of these vertices are essential and not communicating with each other, i.e. there is no path from vertex i to vertex j . However from Fact 1, it is clear that vertex i communicates with state $[\alpha^{(0)}, \dots, \alpha^{(0)}]$. Further, vertex j also communicates with state $[\alpha^{(0)}, \dots, \alpha^{(0)}]$, because of Fact 1. This means that there exists a path from vertex i to j and from j to i . This contradicts the assumption that they are from two different essential classes. ■

Fact 2: Every path through $[\alpha^{(0)}, \dots, \alpha^{(0)}]$ can be enlarged as often as wanted by two or three steps.

Proof: From $[\alpha^{(0)}, \dots, \alpha^{(0)}]$ the possibility exists to reach state $[\beta^{(0)}, \dots, \beta^{(0)}]$. Then the possibility exists to return to state $[\alpha^{(0)}, \dots, \alpha^{(0)}]$, which corresponds to a cycle of length 2. From $[\beta^{(0)}, \dots, \beta^{(0)}]$ it is also possible to reach first $[\beta^{(1)}, \dots, \beta^{(1)}]$ and then return to $[\alpha^{(0)}, \dots, \alpha^{(0)}]$, which corresponds to a cycle of length 3, which proves the fact above. ■

Lemma 4: The essential class associated with the system in Lemma 3 is primitive.

Proof: Generate a path from vertex i to vertex j through $[\alpha^{(0)}, \dots, \alpha^{(0)}]$ for i, j any essential vertex. It follows from Fact 1 that such a path exists for every essential vertex. Then choose the longest of those paths and set k its length plus 2. As from Fact 2 all paths can be enlarged by 2 or 3 steps, it is possible to reach every essential vertex from any other in k steps. Note that the generated k is not necessarily the smallest step length to reach any other vertex. ■

D. Accumulated Behaviour

Since the Markov Chain is ergodic, the probability distribution of which states are occupied converges to a steady state θy , where y is the Perron eigenvector corresponding to the eigenvalue 1 and θ is a scalar to normalise the vector. This vector can be computed numerically, for example by using Matlab, and contains useful information about the overall power consumption by the agents, which we show next.

From the power consumption of each agent it is simple to construct a matrix containing the total power consumption from each state in its diagonal:

$$P = \bar{p}_1 v_1 \oplus \bar{p}_2 v_2 \oplus \dots \oplus \bar{p}_N v_N, \quad (12)$$

where $\oplus$ stands for the Kronecker sum and v_i is defined as in Example 1. Additionally, the Perron eigenvector contains the steady state probability distribution of the states, which is identical to the expected fraction of time that a state

occurs. Therefore, by summing over the entries of the Perron eigenvector which corresponds to states with the same power consumption defined in P , a histogram of the expected total power consumption distribution can be constructed.

Figure 3 shows such a diagram for five agents, all with a power consumption of $3.7kW$ and a total energy allowance of $10kW$. The dark blue bar chart occurs when the EVs use the parameters $\alpha = \beta = 0.2$. The second largest eigenvalue, which is a measure for the convergence rate, in this case is $0.624 + 0.438i$ with magnitude 0.763 . By changing these parameters, the distribution can be modified. For example, the dark green bars of Figure 3 are computed when the EVs use $\alpha = 0.1$ and $\beta = 0.3$. As the bars are higher for lower power consumption values ($0 - 7kW$), it is expected that the system stays longer in states with low power consumption than in the example where the EVs are using the identical values for α and β . In this example the second largest eigenvalue computes to $0.661 + 0.409i$ with magnitude 0.777 .

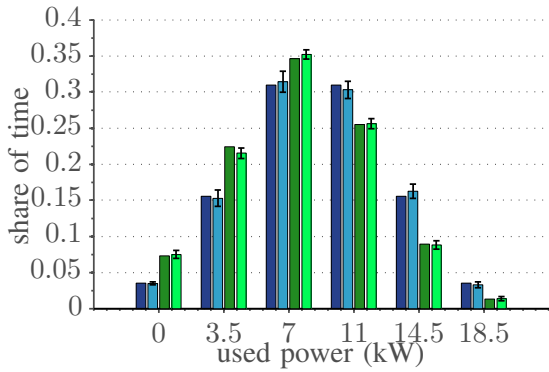


Fig. 3: Histogram of the expected power consumption in dark colors and the averaged power consumption from simulations, see Section V, in light colors. The x -axis shows the power consumed in kW and the y -axis shows the ratio of time that this power is consumed. The blue bar charts is for the parameter values $\alpha = \beta = 0.2$ and the green bars for the parameter values $\alpha = 0.1$ and $\beta = 0.3$. The black lines show the standard deviation for the simulated cases.

V. SIMULATION RESULTS

In this section, the analytic results gained from the previous section are confirmed in a simulation setting using Matlab. Further simulations are conducted using more participating agents, to illustrate the usefulness of the algorithm.

Figure 3 shows the histogram of power usage, light colour, compared to the analytical computed values, dark colour. As for the analytic result, five agents are participating and their parameters are $\alpha = \beta = 0.2$ for the blue graphs and $\alpha = 0.1$ and $\beta = 0.3$ for the green graphs. A comparison of this histogram with Figure 3 shows that the simulated asymptotic power distribution matches well that computed analytically.

However, the results above represent a relatively “bad” load management, since for approximately 38% of the time the demand is lower or higher than the available power, while only in the remaining 62% of time the demanded power

is close to the available power, i.e. one agent’s behaviour determines whether congestion or no congestion occurs. This is due to the small number of participating agents and large parameters α and β , which show the behaviour well, but are not suitable for real world applications.

To avoid variability due to a small number of participating agents and to illustrate the effects of connecting and disconnecting vehicles, we conducted a simulation with 75 agents. The simulation uses a constant available power of $100kW$ to show the usefulness of the algorithm in peak shaving tasks. Note that the total consumption of fifty electric vehicles uncontrolled would be $185kW$, which corresponds to a reduction of 45%. Naturally such a reduction increases the necessary charging time, and could not be imposed for a long period without causing inconvenience for the owner.

The agents disconnect as soon as they finished charging or after a predefined charging time. The energy requirement is randomly selected between $12kW$ and $24kW$ and the charging time between 8 and 14 hours. The connection time is randomly chosen in the first 8 hours of the simulation, which is running for 16 hours, with ten vehicles, which are connected from the start. Further, we assume that all EVs use the same parameters $\bar{p} = 3.7kW$, $\alpha = 10^{-5}$, and $\beta = 10^{-4}$. Those parameters have been chosen manually. We conjecture that those parameters could be selected by solving an optimisation problem to achieve the desired accumulated behaviour. However, this is beyond the scope of this paper.

Figure 4 shows the available power and the aggregate demand of the vehicles. Mostly, the overshoot is caused by one vehicle, which means that by proper selection of the cap the aggregated power can be kept below a certain value with high probability. At the beginning and the end of the simulation the aggregated demand is less than the available power, this behaviour is due to the small number of vehicles connected during those times, as either a lot of them are not yet connected or most are already finished charging, respectively. During those times all connected vehicles are on and no capacity signal is sent.

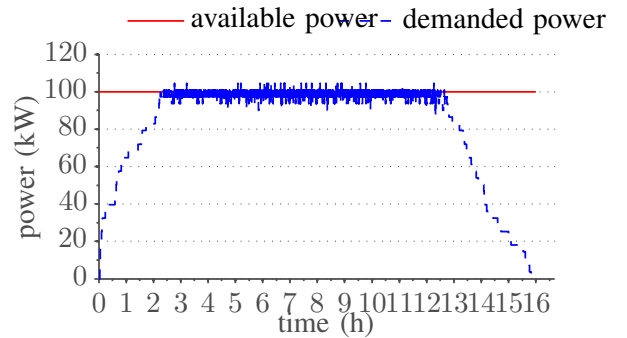


Fig. 4: Available and aggregated power consumption during simulation where a total of 75 vehicles connect and disconnect during the simulation.

The second simulation conducted uses a variable power in time steps of 10 minutes and runs for a total of 3 hours.

We neglect here the connecting, disconnecting, or completing charging throughout the simulation period, due to the short length of the simulated period. In total 50 EVs participate, all of which use identical parameters $\bar{p} = 3.7kW$, $\alpha = 10^{-5}$, and $\beta = 10^{-4}$. Figure 5 shows the available power and the power demanded by the agents. The demanded power is able to follow such a power signal very well. Clever adjustment of the parameters allows also tracking of rapidly changing power outputs as shown in Figure 6, where the available power changes every 10 seconds. The parameters are chosen manually as $\alpha = 0.0003$, and $\beta = 0.0005$. Further tuning of the variables may improve the outcome.

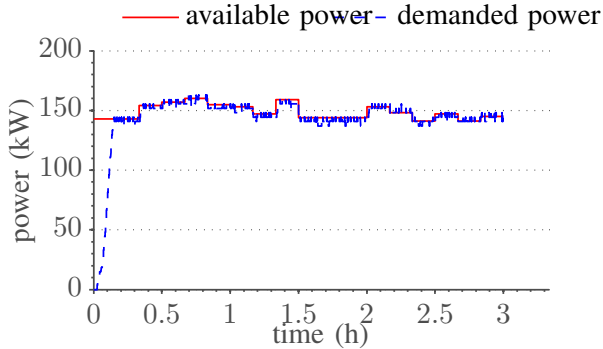


Fig. 5: Available and demanded power during a 3 hours long simulation period, where the power varies every 10minutes.

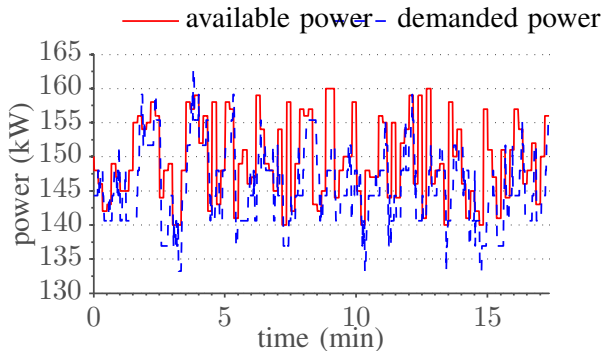


Fig. 6: Available and demanded power during a short period extracted from a 3 hours long simulation period, where the power varies every 10seconds.

VI. CONCLUSION

In this paper, we suggested a game between multiple fixed structure automata for load management of electric vehicles, with binary (on/off) actions available. The algorithm suggested is based on the AIMD algorithm, used in previous papers by the authors [14], [15]. We used Markov theory to show that the game forms an ergodic Markov Chain. Using this it is possible to compute a power histogram to which the system converges independent of the initial conditions. Simulations show the usability of the algorithm for peak shaving tasks, as well as for slow varying load tracking.

Further studies are necessary to find a method for optimal parameter selection, as well as to investigate the possibility to deal with more realistic and different scenarios, similar to the ones in [14]. Current work also applies the algorithm to thermostatically controlled loads, such as fridges, water heaters, and air conditioners. Future work, should then look at the interaction if both algorithms, the one presented here, as well as the one presented in [14], [15], are applied at the same time to different loads.

ACKNOWLEDGEMENT

The authors would like to thank F. Wirth for fruitful discussions, and in particular, the suggestion to analyse the system as a Markov chain and R. Shorten for his support.

REFERENCES

- [1] P. Finn, C. Fitzpatrick, and M. Leahy, "Increased penetration of wind generated electricity using real time pricing; demand side management," in *Sustainable Syst. and Technology (ISSST '09) IEEE Int. Symp. on*, May 2009.
- [2] A. Brooks, E. Lu, D. Reicher, C. Spirakis, and B. Wehl, "Demand dispatch," *Power Energy Mag., IEEE*, vol. 8, no. 3, May - Jun. 2010.
- [3] D. Callaway and I. Hiskens, "Achieving controllability of electric loads," *Proc. IEEE*, vol. 99, no. 1, Jan. 2011.
- [4] L. Gan, U. Topcu, and S. Low, "Optimal decentralized protocol for electric vehicle charging," in *Decision and Control and European Control Conf. (CDC-ECC), 50th IEEE Conf. on*, Dec. 2011.
- [5] S. Deilami, A. Masoum, P. Moses, and M. Masoum, "Real-time coordination of plug-in electric vehicle charging in smart grids to minimize power losses and improve voltage profile," *Smart Grid, IEEE Trans. on*, vol. 2, no. 3, Sep. 2011.
- [6] K. Clement-Nyns, E. Haesen, and J. Driesen, "The impact of charging plug-in hybrid electric vehicles on a residential distribution grid," *Power Syst., IEEE Trans. on*, vol. 25, no. 1, Feb. 2010.
- [7] P. Richardson, D. Flynn, and A. Keane, "Optimal charging of electric vehicles in low-voltage distribution systems," *Power Syst., IEEE Trans. on*, vol. 27, no. 1, Feb. 2012.
- [8] J. A. P. Lopes, F. J. Soares, and P. M. R. Almeida, "Identifying management procedures to deal with connection of electric vehicles in the grid," in *Proc. of IEEE Bucharest Power Tech Conf.*, 2009.
- [9] L. Carradore and R. Turri, "Electric vehicles participation in distribution network voltage regulation," in *Universities Power Eng. Conf. (UPEC), 45th Int., Aug. 31 2010 - Sep. 3 2010*, 2010.
- [10] K. Mets, T. Verschueren, F. De Turck, and C. Devellder, "Evaluation of multiple design options for smart charging algorithms," in *Commun. Workshops (ICC), IEEE Int. Conf. on*, Jun. 2011.
- [11] D. Wu, D. Aliprantis, and L. Ying, "Load scheduling and dispatch for aggregators of plug-in electric vehicles," *Smart Grid, IEEE Trans. on*, vol. 3, no. 1, Mar. 2012.
- [12] W. Shi and V. Wong, "Real-time vehicle-to-grid control algorithm under price uncertainty," in *Smart Grid Commun. (SmartGridComm), IEEE Int. Conf. on*, Oct. 2011.
- [13] J. Lopes, F. Soares, and P. Almeida, "Integration of electric vehicles in the electric power system," *Proc. IEEE*, vol. 99, no. 1, Jan. 2011.
- [14] S. Stüdl, E. Crisostomi, R. Middleton, and R. Shorten, "A flexible distributed framework for realising electric and plug-in hybrid vehicle charging policies," *Int. Journal of Control*, vol. 85, no. 8, 2012.
- [15] —, "AIMD-like algorithms for charging electric and plug-in hybrid vehicles," in *Electric Vehicle Conf. (IEVC), 2012 IEEE Int.*, Mar. 2012.
- [16] Z. Ma, D. Callaway, and I. Hiskens, "Decentralized charging control for large populations of plug-in electric vehicles," in *Proc. IEEE Conf. on Decision and Control*, 2010.
- [17] S. Keshav and C. Rosenberg, "How internet concepts and technologies can help green and smarten the electrical grid," *ACM SIGCOMM Comput. Communication Review*, vol. 41, no. 1, Mar. 2011.
- [18] K. S. Narendra and M. A. L. Thathachar, *Learning automata: an introduction*. Prentice-Hall, Inc., 1989.
- [19] D. J. Hartfiel, *Markov Set-Chains*, ser. Lecture Notes in Mathematics. Springer, 1998.