Demand-side Management: Optimising Through Differential Evolution Plug-in Electric Vehicles to Partially Fulfil Load Demand

Edgar Galván-López[1], Marc Schoenauer[2],
Constantinos Patsakis[3], and Leonardo Trujillo[4]

[1] School of Computer Science & Statistics, Trinity College Dublin
edgar.galvan@scss.tcd.ie.
[2] TAO Project, INRIA Saclay & LRI - Univ. Paris-Sud and CNRS, Orsay, France
marc.schoenauer@inria.fr
[3] Department of Informatics, University of Piraeus
kpatsak@unipi.gr
[4] Doctorado en Ciencias de la Ingeniería, Instituto Tecnológico de Tijuana
leonardo.trujillo@tectijuana.edu.mx

Abstract. In this paper, we investigate the use of an stochastic optimisation bio-inspired algorithm, differential evolution, and proposed two fitness (cost) functions that can automatically create an intelligent scheduling for a demand-side management system so that it can use plug-in electric vehicles's (PEVs) batteries to partially and temporarily fulfil electricity requirements from a set of household units. To do so, we proposed two fitness functions that aim: (a) to use the most amount of energy from the batteries of PEVs while still guaranteeing that they can complete a journey, and (b) to enrich the previous function to reduce peak loads.

Keywords: Differential Evolution, Demand-side Management Systems, Plug-in Electric Vehicles.

1 Introduction

Evolutionary Algorithms (EAs) [1, 2], also known as Evolutionary Computation systems, are influenced by the theory of evolution by natural selection. These algorithms have been with us for some decades and are very popular due to robust theoretical works [3–6] developed around them that have helped us to understand why they work (e.g, representations' properties) and due to their successful application in a variety of different and challenging problems, ranging from the automated design of an antenna carried out by NASA [7], the automated optimisation of game controllers [8], the automated evolution of Java code [9], up to the automated design of combinational logic circuits [10, 11]. EAs can be considered a "black-box", as they do not require any specific knowledge of the fitness function. They work even when, for example, it is not possible to define a gradient on the fitness function or to decompose the fitness function into a sum of per-variable objective functions.

In this work, we are interested in investigating the applicability of EAs in a dynamic and challenging problem in Demand-Side Management (DSM) Systems taken from Smart Grids where, in summary, the goal is to automatically create fine-grained solutions that indicate the amount of energy that can be taken from electric vehicles' (PEVs) batteries to partially satisfy energy demand in residential areas and reducing electricity peaks, whenever possible. The proposed approach and fitness functions used in our work (described in Section 2) is not amenable to analytic solution or simple gradient-based optimisation, hence search algorithms such as EAs are required.

DSM is normally considered as a mechanism or program, implemented by utility companies to control the energy consumption at the customer side [12]. DSM is an important research area in the Smart Grid (SG) community as shown by the increasing number of publications over the years (e.g., more than 2,000 papers have been published in this area where more than two thirds have been published since 2010 [13]).

DSM programs include different approaches (e.g., manual conservation and energy efficiency programs [14], Residential Load Management (RLM) [15, 16]), where RLM programs based on smart pricing are amongst the most popular methods. The idea behind smart pricing is to encourage users to manage their loads, so that they can reduce electricity prices while, at the same time, the utility companies achieve a reduction in the peak-to-average ratio (PAR)[5] in load demand by shifting consumption whenever possible [13, 15, 17].

One of the major limitations of smart pricing is the fact that the electricity price is proportional to the electricity demand (i.e., a high number of appliances/devices connected to the grid results in having high electricity costs). To alleviate this problem, we propose the development of a demand-side *autonomous intelligent* management system that exploit plug-in electric vehicles' (PEVs) batteries. More precisely, our system uses the PEV's batteries to partially and temporarily fulfil the demand of end-use consumers instead of using only the electricity available from a substation transformer. This is possible thanks to the vehicle to grid technology (V2G), which is described as a system in which electric-drive vehicles can feed power to the grid with the appropriate communication/connection technologies acting as mobile generators of limited output [18, 19].

The deployment of such a system implies several significant challenges, e.g. different driving patterns resulting in the amount of energy needed at the time of departure, amount of energy taken from the PEVs' batteries. To tackle this problem, we use an optimisation EA.

Thus, the main contribution of this research is a novel approach to balance the load demand from dozens of household units using both a substation transformer and PEVs' batteries as mobile energy storage units[6] by considering the *automatic* generation of solutions via the use of EAs. To this end, we are interested in maximising, in general, the use of available energy from the PEVs' batteries while ensuring that each of the PEVs can complete a journey to work, where the PEVs can be charged, and in particular,

[5] Peak-to-average ratio is calculated by the maximum load demand for a period of time over the average load demand, so a lower PAR is normally preferred due to e.g. maintenance costs [16].

[6] In this work, we use the terms "substation transformer" and "PEV's batteries" to differentiate between the two sources of energy.

helping in the reduction of peak loads at the transformer level by using the most quantity of energy from the PEVs' batteries during these peak periods. This problem would be simple enough if it was not for the dynamicity associated to the problem and if we would not care about keeping the PAR relatively low.

To achieve this, we allow the DSM system to make fine-grained decisions (i.e., variable amount of energy requested) by using a continuous representation instead of using a discrete representation (i.e., turning a device/appliance on or off resulting in feeding/getting a constant amount of energy) as normally adopted in DSM [20].

To this end, we use a form of EAs, called Differential Evolution (DE) [21], that allows us to achieve this. More specifically, DE uses a vector of real-valued functions and we use them to represent an individual (potential solution) that specifies an energy consumption scheduling vector, which in turn indicates the amount of energy that should be taken from the PEVs' batteries aiming at fulfilling the goals previously described (e.g., maximising the energy consumption available from the batteries while at the same time reducing peak loads at the transformer level with associated constraints such as guaranteeing that each PEV would complete a journey to work). Details on how this algorithm works and its adoption in this research are described in Section 2.

1.1 Significance of this Research

From the 1980s, DSM has been studied extensively by the research community. Analysing the research carried on DSM is difficult if we consider that there are more than 2,000 scientific papers published only in the IEEE Xplore database. Inspired by the work conducted by Poli [22], where the author analysed titles, keywords and abstract of hundreds of papers, we also carried a similarity analysis relationship between hundred of papers[7] that discussed DSM and key terms of these papers for a quick and useful interpretation of the research carried out in this area.

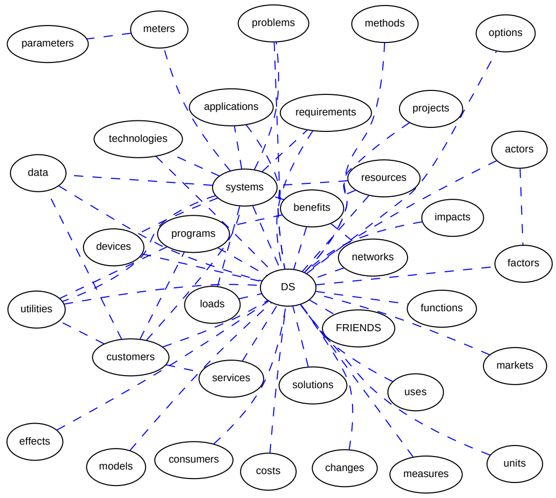
As we will see, the research conducted in DSM over the last decades has evolved significantly, and due to space constraints, we only show the visual representation[8] of the research conducted from 1985 until 2009 (572 papers were analysed) and from 2010 until 2015 (1,841 were analysed), shown in Figure 1.

It is clear to see that some areas remain of vital importance in DSM, such as the benefits that DSM can offer to both customers and utility companies. There are, however, other areas of research emerging in DSM as shown at the bottom of Figure 1 (research conducted over the last five years). Note, for example, the interest of investigating the impact/integration of electric vehicles in DSM. This is shown in the very core of Figure 1 regarding the analysis from 2010 to 2015 (bottom of the figure). Other elements worth observing are data, users, devices.

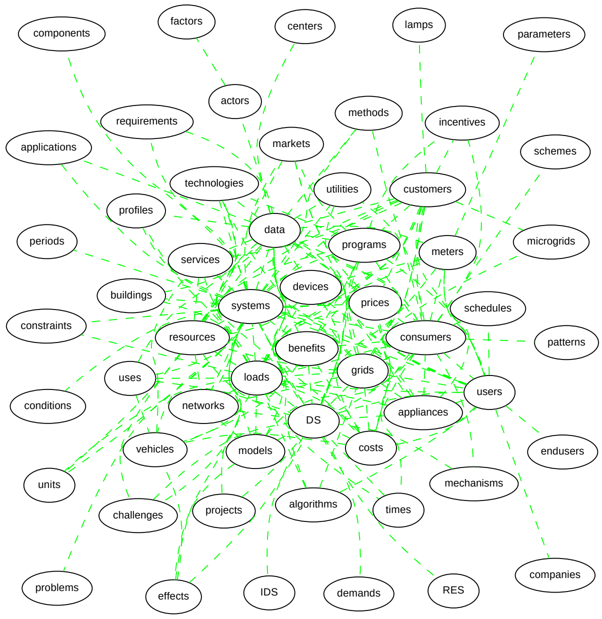
The research presented in this work deals with these elements and shows its importance to DSM. Specifically, as mentioned previously, we are interested in using PEVs' batteries as mobile energy storage units to help the SG by designing an intelligent autonomous DSM.

[7] Source: IEEE Xplore database searching for "Demand-side Management". Last accessed date: 22/01/2015.

[8] Details on how these figures were produced can be found in [22].



(a) Analysis from 1985 – 2009



(b) Analysis from 2010 – 2015

Fig. 1. Analysis of the publications on 'Demand Side Management' Systems from 1985 until 2009 (top) and from 2010 until 2015 (bottom). Only links (similarities) with strength greater than 40 in (a) and 60 (b) were passed to neato. The rest-length for repulsive forces between nodes was set to 9.

Energy storage units, such as pumped hydroelectric energy storage units and compressed air energy storage units, have been with us around for several decades [23] via [24] and they have been used to provide both energy and ancillary services. Their use, however, have not been massively popular mainly because there is a cost associated with their acquisition and their corresponding installation. However, with the emergence of relatively new technologies (e.g., PEVs) and their relatively "easy" integration into the grid, it is necessary to account for autonomous and intelligent algorithms to exploit their capabilities. This in consequence can bring substantial benefits to both end-use consumers and to the grid (e.g., reduction of peak loads, savings in electricity costs; see [25–27] for a more detailed discussion of energy storage units' benefits).

The rest of this paper is organised as follows. In the following section we briefly introduce differential evolution and present our proposed approach. In Section 3, we present the experimental setup used in this work and Section 4 discusses the findings of our approach. Finally, in Section 5 we draw some conclusions.

2 PROPOSED APPROACH

2.1 Background

There are multiple EAs methods, such as Genetic Algorithms (GAs) [28], Genetic Programming (GP) [29], Differential Evolution (DE) [21]. All these methods use evolution as an inspiration to automatically generate potential solutions for a given problem. They differ, mainly, in the representation used (i.e., encoding of a solution). For example, the typical representation used in GAs is fixed bitstrings, GP's typical representation is tree-like structures, DE uses a vector of real-valued functions.

In this work, we use a DE algorithm given its natural representation (i.e., real-valued functions). Other bio-inspired algorithms can also use this type of representation, however, in this work we decided to use a DE given its efficiency for global optimisation over continuous search spaces [21]. By using this type of representation, we can have a more fine-grained action granularity (e.g., in this work, each element in the vector represents how much energy will be taken from the PEVs' batteries to feed electricity to household units), instead of using a more limited representation such as a bitstring representation that could indicate to take a pre-defined amount of energy (i.e., on or off) from PEVs' batteries to partially fulfil energy consumption from household units. We further discuss this later in this section.

The goal of DE is to evolve NP D-dimensional parameter vectors $x_{i,G} = 1, 2, \cdots, NP$, so-called population, which encode the potential solutions (individuals), i.e., $x_{i,G} = \{x^1_{i,G} \cdots, x^D_{i,G}\}, i = 1, \cdots, NP$ towards the global optimum solution (e.g., highest values when maximising a cost function). The initial population is randomly generated and this should be done by spreading the points across the entire search space (e.g., this could be achieved by distributing each parameter on an individual vector with uniform distribution between lower and upper bounds x^l_j and x^u_j). To automatically evolve these potential solutions over generations via the definition of a fitness function, DE uses the most common bio-inspired operators as commonly carried out in EAs: mutation and crossover to find the global optimum solution. Each of these operators is briefly explained in the following lines (refer to [21, 30] for a detailed description on how they work).

The mutation operator generates a mutant vector following one of the following strategies:
DE/rand/1

$$v_{i,G} = x_{r_1^i,G} + F \cdot (x_{r_2^i,G} - x_{r_3^i,G})$$

DE/best/1

$$v_{i,G} = x_{best,G} + F \cdot (x_{r_1^i,G} - x_{r_2^i,G})$$

DE/rand-to-best/1

$$v_{i,G} = x_{i,G} + F \cdot (x_{best,G} - x_{i,G}) + F \cdot (x_{r_1^i,G} - x_{{}_2^i,G})$$

DE/best/2

$$v_{i,G} = x_{best,G} + F \cdot (x_{r_1^i,G} - x_{r_2^i,G}) + F \cdot (x_{r_3^i,G} - x_{r_4^i,G})$$

DE/rand/2

$$v_{i,G} = x_{r_1^i,G} + F \cdot (x_{r_2^i,G} - x_{r_3^i,G}) + F \cdot (x_{r_4^i,G} - x_{r_5^i,G})$$

where indexes r_1, r_2, r_3, $r_4 \in \{1, 2, \cdots, NP\}$ are random and mutually different. F is a real and constant factor $\in [0, 2]$ for scaling differential vectors and $x_{best,G}$ is the individual with best fitness value (e.g., highest value for a maximisation function) in the population at generation G.

The crossover operator increases the diversity of the mutated parameter vectors and is defined by:

$$v_{i,G+1} = (v_{1i,G+1}, v_{2i,G+1}, \cdots, v_{Di,G+1})$$

where:

$$v_{ji,G+1} = \begin{cases} v_{ji,G+1} & \text{if } randb(j) \leq CR \text{ or } j = rnbr(i), \\ x_{ji,G} & \text{otherwise} \end{cases}$$

where $j = 1, \cdots, D$, $randb(j)$ is the jth evaluation of a uniform random number generator with outcome $\in [0, 1]$. CR is the constant crossover rate $\in [0, 1]$. $rnbr(i)$ is a randomly chosen index $\in 1, 2, \cdots, D$ which ensures that $u_{i,G+1}$ receives at least one parameter value from $u_{i,G+1}$.

The performance of the DE algorithm depends on different factors, such as the values associated to the parameters (e.g., population size) as well as the variant of the operator used (e.g., variant of the mutation operator). This, intuitively means, that some preliminary runs would be normally required to determine which variant of an operator performs better on a given problem. We further discuss this in the following section.

2.2 Proposed Representation and Fitness Function

We now extend the natural DE representation to tackle the problem described throughout the paper and proceed to define the fitness (cost) function that allows the algorithm to automatically guide the evolutionary search.

Let N denote the number of household units (users), where the number of household units is $N \triangleq | N |$. For each household $n \in N$, let l_n^t denote the total load at time $t \in T \triangleq \{t_i, \cdots, t_f\}$. Without loss of generality, we assume that time granularity is 15 minutes. The load for household n, from t_i to t_f, is denoted by:

$$l_n \triangleq [l_n^{t_i}, \cdots, l_n^{t_f}] \tag{1}$$

From this, we can calculate the load across all household units N at each time $t \in [t_i, t_f]$ as follows:

$$L_t \triangleq \sum_{n \in N} l_n^t \tag{2}$$

Similarly, let M denote the number of plug-in electric vehicles available in N. For each electric vehicle $m \in M$, let E_m^t denote the energy that can be taken from the PEV at time $t \in T \triangleq \{t_i, \cdots, t_f\}$. Without loss of generality, we assume that time granularity is again 15 minutes. The total energy taken from an PEV from t_i until t_f is denoted by:

$$E_m \triangleq [E_m^{t_i}, \cdots, E_m^{t_f}] \tag{3}$$

We use this as a foundation to represent an individual that specifies an energy consumption scheduling vector. More specifically, an individual is represented by:

$$E_M \triangleq \begin{bmatrix} E_{m_1}^{t_i}, \cdots, E_{m_1}^{t_f} \\ E_{m_2}^{t_i}, \cdots, E_{m_2}^{t_f} \\ \vdots \\ E_{m_M}^{t_i}, \cdots, E_{m_M}^{t_f} \end{bmatrix} \tag{4}$$

where each E_m^t is a real value representing the quantity of energy taken from an PEV's battery. Each row represents the behaviour of a single PEV over the full period; each column represents the behaviour of all PEVs at a single time-slot. An individual in the EA is just a matrix E_M, unrolled to give a vector of real-valued functions, that is:

$$E_1^{t_i}, \cdots, E_1^{t_f}, E_2^{t_i}, \cdots, E_2^{t_f}, \cdots, E_M^{t_i}, \cdots, E_M^{t_f} \tag{5}$$

Based on these definitions, the total energy taken across all M PEVs at each $t \in [t_i, t_f]$ can be calculated as:

$$E_t \triangleq \sum_{m \in M} E_m^t \tag{6}$$

To automatically find good energy consumption scheduling solutions, defined in Equation 4, we need to define a fitness (cost) function that indicates the quality of our evolved solution. First, we focus our attention in designing a cost function that tries to create valid solutions in terms of using the maximum allowed energy from each PEV (i.e., guaranteeing that a minimum state of charge (SoC) is left at the time of departure t_f).

From Equation 3, we know the amount of energy available from $m \in M$ at any given period of time t denoted by E_m^t. Because each PEV can be charged at work and the distance from home to work remains constant, it is fair to assume the knowledge of a minimum SoC expressed in kW, denoted as m_{SoC}, at the time of departure t_f for each $m \in M$, so that it can reach work and be recharged at a lower rate. From this, we let the DE to assess a potential solution, denoted in Equation 4, measuring the amount of energy taken from the PEVs. This is defined as:

$$f_l(E_M) \triangleq \textbf{ maximise } \frac{1}{\#\{m \in M\}} \sum_{m \in M} \frac{E_m + (E_m + 1)(m_{SoC} - E_m^{t_i})}{m_{SoC}\left(E_m^{t_i} - m_{SoC}\right)} \tag{7}$$

Equation 7 guides evolutionary search towards a local optimum solution since it only encourages the finding of solutions that maximise the use of allowable energy taken from PEVs' batteries. Thus, there is a necessity to further enrich this equation, so that a higher quantity of energy is taken from the PEVs' batteries whenever deemed necessary (e.g., higher consumption during high peak periods). We achieve this by using Equations 2 and 6 that indicate the load across all household units L_t at time t and the total energy taken across all PEVs E_t at time t, respectively; and we define a degree of importance for each time slot as t_r. Putting everything together we have:

$$f_g(E_M) \triangleq f_l(E_M) + \textbf{ maximise } \frac{1}{\#\{m \in M\}} t_r \sum_{t=t_i}^{t_f} \frac{E_t}{L_t t_r} \forall t_r < T_r - \frac{1}{\#\{m \in M\}} t_r \sum_{t=t_i}^{t_f} \frac{E_t}{L_t t_r} \forall r \geq T_r \tag{8}$$

where T_r is a threshold that denotes the number of time slots that are considered critical (i.e., high peak period). In this work, as defined in this section and we discuss further afterwards, a number of time slots is defined by t_i and t_f, where a third is considered critical ($T_r = 20$).

3 EXPERIMENTAL SETUP

3.1 Household Units

To test the scalability of our proposed approach, we simulated the consumption of 40 and 80 household units, where each of them uses between 10 and 20 appliances. As indicated throughout the paper, the goal is to use PEVs' batteries in an intelligent way to partially satisfy energy demand from the end-use consumers (recall that we work under the assumption that the PEVs can be charged at work).

To this end, we simulated that around 20% of household units account for an PEV. To make this problem dynamic, we allowed the patterns of arrival (t_i), departure (t_f) and initial State of Charge (SoC) for each of the PEVs to vary for each of the 30 simulated working days. More specifically, the arrival and departure time for each of the PEVs have a 90-minute time frame starting at t_i =17:00 and t_f =6:30, respectively (i.e., arrival

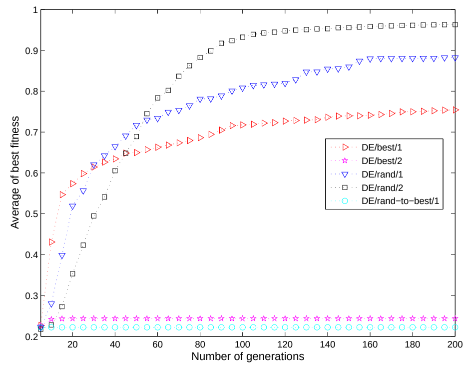


Fig. 2. Average of best fitness values of 30 independent runs for each of the five types of mutation operators tested in this work (see Section 2), using 500 individuals and 200 generations, to maximise energy consumption from electric vehicles' batteries (Equation 7). Higher values are preferred.

time could be between 17:00 and 18:30, whereas departure time could be between 6:30 and 8:00). The initial SoC_{t_i} for each of the PEVs for each of the simulated days is set between 48% and 60% and the final SoC_{t_f} is set between 30% and 35% to allow each PEV to reach work. Table 1 summarises the parameters used to simulate our scenario. We ran our simulations for a period of 30 days of simulated time.

3.2 Scenarios

As indicated in Section 2, we defined a bottom-up approach, where we defined, first, a fitness function that tries to maximise the energy that can be taken from the PEVs' batteries while ensuring that each of them reaches work, described in Equation 7. We then enriched the fitness function by trying to also reduce the highest load demands at the substation transformer, described in Equation 8 (i.e., use the most amount of energy from the batteries at high-peak time while at the same time ensuring the PAR remains low). We tested both fitness functions for 40 and 80 household units, resulting in four different scenarios.

3.3 Differential Evolution

As mentioned in Section 2, differential evolution's performance, as any other evolution-based algorithm, depends, among other things, on the values associated to the parameters that need to be specified for the algorithm (e.g., population size, number of generations), in general, and in the type of operator used, in particular.

No *a priori* knowledge is available to presume which mutation operator will perform better in the previously defined problem. To this end, we executed 30 independent runs of our proposed approach for each of the mutation variants, e.g., DE/rand/1, DE/best/1 (150[9] independent runs in total to find only the best mutation strategy) using

[9] 30 independent runs * 5 variants of the mutation operator.

Parameter	*Value*
Number of household units	40, 80
Number of appliances	Uniform in [10,20]
Number of PEVs	≈ 20% of houses have one PEV
Arrival and departure time	t_i =[17:00,18:30] t_f =[6:30,8:00]
Frequency of making a decision	15 minutes
Number of times slots T	60
State of Charge at t_i	Uniform in $[48, 60]$
State of Charge at t_f	Uniform in $[30, 35]$

Table 1. Summary of parameters used for our smart grid system.

Parameter	*Value*
Population size	500
Length of the individual	T (see Table 1)
Height of the individual	Number of PEVs (see Table 1)
Generations	200
Crossover rate	0.5
Mutation strategy	DE/rand/2
Elitism	1 individual
Termination criterion	Maximum number of generations
Independent runs	30

Table 2. Summary of parameters used for our evolutionary algorithm.

the first proposed fitness function (Equation 7) which maximises the energy taken from 11 PEVs' batteries to complement the energy consumption of 40 household units averaged over 30 days. Figure 2 shows the performance by measuring the average of best fitness per generation for each of the five mutation variants, using a population size of 500 individuals and 200 generations.

Clearly, the mutation strategy DE/rand/2 achieved the best performance and we used it to run our experiments to automatically find a (nearly) optimal solution. To obtain meaningful results, we performed 30 independent runs for each of the scenarios explained in the previous paragraphs (we executed 30 * 4 runs in total[10]). Runs were stopped when the maximum number of generations was reached.

As mentioned in Section 2, every element of the DE vector represents how much energy can be taken from the batteries of the PEVs. We make a decision every 15 minutes. Thus, the length of the individual that represent the solution is the number of time slots defined between 17:00 and 8:00am, whereas the height is defined by the number of electric vehicles used, as defined in Equation 4. The parameters used in our experiments are summarised in Table 2.

4 RESULTS

In the following paragraphs, we will analyse: (a) how the PEVs' batteries were used to partially satisfy the demand of a set of household units, (b) when the highest consumption from PEVs' batteries occurred, and finally, (c) the implications of the new consumption model via the analysis of the peak-to-average-ratio.

[10] 30 independent runs, 4 different scenarios (i.e., 40 and 80 household units, trying to maximise: (a) energy consumption from PEVs, and (b) energy consumption from PEVs while also considering reducing highest load peaks; for each of the set of household units used in this work).

4.1 Maximising Energy Consumption from PEVs' batteries

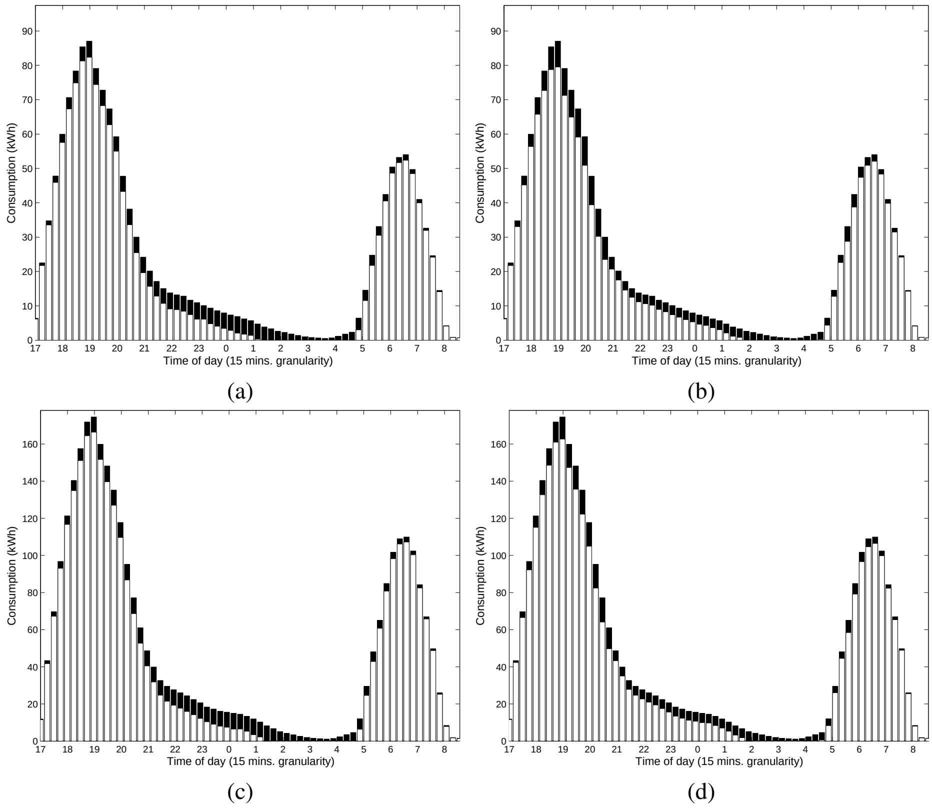


Fig. 3. Average of 30-day energy consumption for 40 (top) and 80 (bottom) household units, each using between 10-20 appliances. The consumption of energy from the transformer alone is shown by the white-filled bars whereas the black-filled bars represent the consumption taken from electric vehicles' batteries. Maximising energy consumption from electric vehicles only and maximising energy consumption from electric vehicles while considering also reducing highest load peaks are shown in the left-hand side and right-hand side of the figure, respectively.

Let us start analysing our approach on how the batteries of the PEVs helped to partially satisfy the consumption demand from a set of household units. The averaged consumption over a period of 30 days of these household can be seen in Figure 3 (a, b) and (c, d) for 40 and 80 houses, respectively.

In the left-hand side of this figure, we show the distribution of consumption of both transformer and PEVs' batteries proposed by the differential evolution algorithm, when trying to maximise the consumption of energy from the PEVs' batteries via Equation 7. More specifically, it aims at using all the possible energy available from the batteries while guaranteeing that each PEV has a minimal SoC at the time of departure (see Table 1) that guarantees that each PEV will reach work. The white-filled bars represent the electricity taken from the substation transformer whereas the remaining consumption to fulfil the load demand is taken from the PEVs' batteries. The latter is shown by the black-filled bars.

Because we are interested in using the PEVs' batteries as mobile energy storage units, we are particularly interested in seeing how the energy consumption from these is managed by the differential evolution algorithm. In the first instance of our algorithm (i.e., maximising the energy consumption from the batteries of PEVs with associated constraints as formally described in Equation 7, as mentioned previously), it is expected that the energy taken from the batteries would not follow a particular pattern (e.g., there is no correlation between the amount of energy consumption from PEVs and the energy needed by a number of household units). Indeed, this is the case as seen in the left-hand side of Figure 3. For example, notice how the consumption from PEVs' is proportionally similar during both high-peak (e.g., 18:30 - 19:30) and low-peak periods (e.g., 22:00 - 23:00).

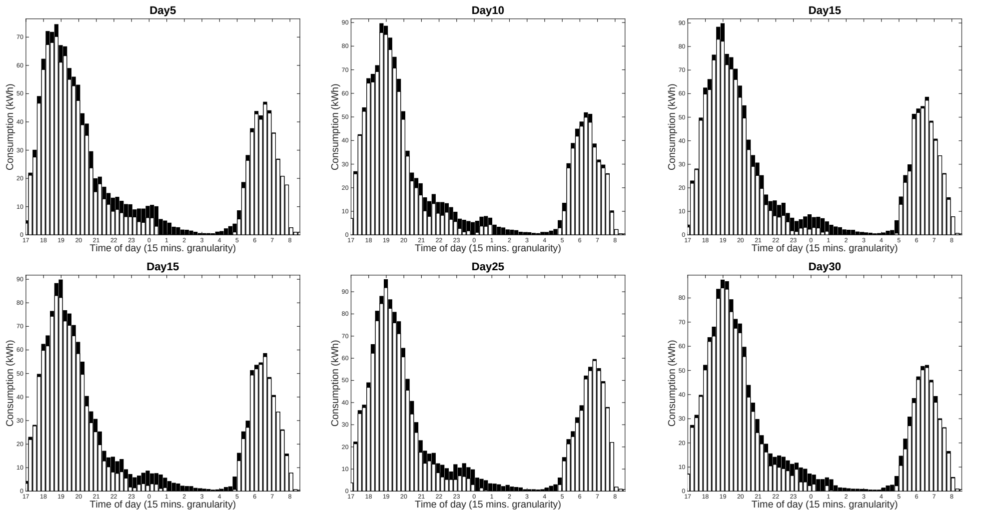
The situation is more encouraging when we consider the second instance of our algorithm (i.e., maximising energy consumption from PEVs' batteries while considering high-peak periods as formally described in Equation 8), shown in the right-hand side of Figure 3. As it can be observed, the proposed enriched fitness function is able to automatically produce results that can reduce the load peaks from the substation transformer by using more electricity from the PEVs' batteries. For example, notice how the consumption of energy from batteries is higher during high-peak periods (e.g., 18:30 - 19:30) and lower during low-peak periods (e.g., 22:00 - 23:00). Details on the consumption, per day for six days, can be seen in Figures 4 and 5 when using 40 and 80 household units, respectively. The first two rows and the last two rows of these figures show the behaviour observed when using Equations 7 and 8, respectively.

4.2 Consumption from PEV's batteries

In the previous paragraphs, we discussed and showed the results obtained by our approach using two fitness (cost) functions, formally described in Equations 7 and 8. It is clear that the latter function is able to use a higher quantity of energy from the PEVs' batteries during high-peak periods compared to the effects observed when using the former function, as shown in the right-hand and left-hand side of Figure 3, respectively, using 40 and 80 household units. This averaged result over a period of 30 simulated working days, however, does not inform us in detail when the highest consumption from batteries occurred (e.g., when and how much consumption from the batteries for every of the simulated days occurred). Some insight can be gained when analysing some days (see Figures 4 and 5) but this still is limited since, due to page-limit constraints, not all days can be shown.

To this end, we kept track of the consumption from the PEVs' batteries during the simulated period of time (i.e., 17:00 - 8:00) for every day of the simulated days. The patterns of such consumption are shown in Figure 6 (a, b) and (c, d) for 40 and 80 household units, respectively.

Let us start our analysis when maximising the energy that can be taken from the batteries while ensuring that each PEV has the minimum SoC at the time of departure, 40 houses trying to maximise the use of PEV's batteries



40 houses trying to maximise the use of PEVs' batteries while attempting to reduce high peak loads

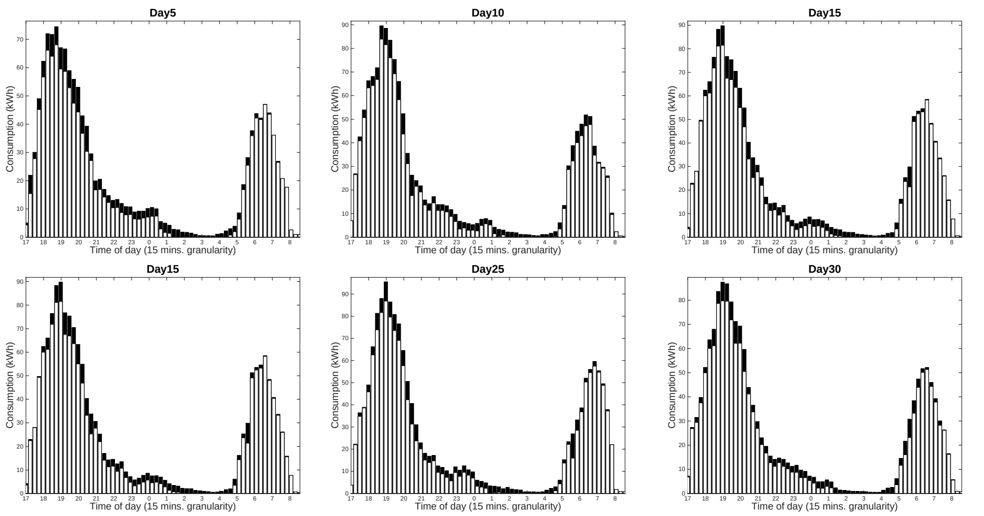
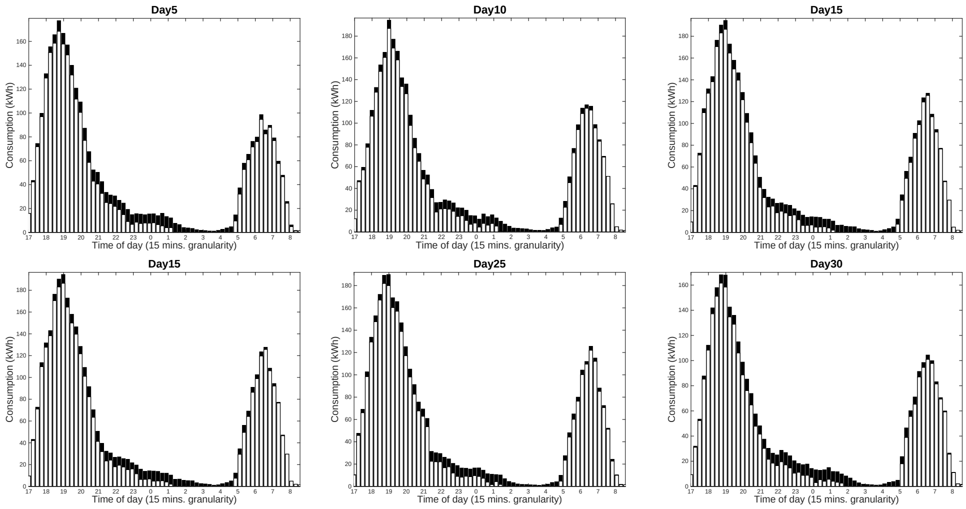


Fig. 4. Consumption per day (only 6 days chosen randomly) for 40 household units using each between 10 and 20 appliances. The consumption of energy from the transformer alone is shown by the white-filled bars whereas the black-filled bars represent the consumption taken from electric vehicles batteries. Maximising energy consumption from electric vehicles only is shown in the first two rows and maximising energy consumption from electric vehicles while considering also reducing highest load peaks is shown in the last two rows.

80 houses trying to maximise the use of PEV's batteries



80 houses trying to maximise the use of PEVs' batteries while attempting to reduce high peak loads

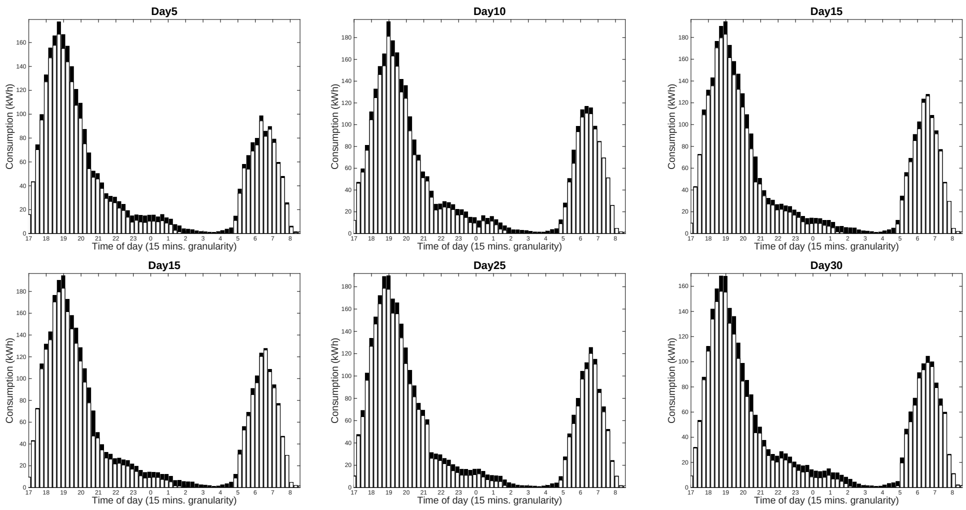


Fig. 5. Consumption per day (only 6 days chosen randomly) for 80 household units using each between 10 and 20 appliances. The consumption of energy from the transformer alone is shown by the white-filled bars whereas the black-filled bars represent the consumption taken from electric vehicles batteries. Maximising energy consumption from electric vehicles only is shown in the first two rows and maximising energy consumption from electric vehicles while considering also reducing highest load peaks is shown in the last two rows.

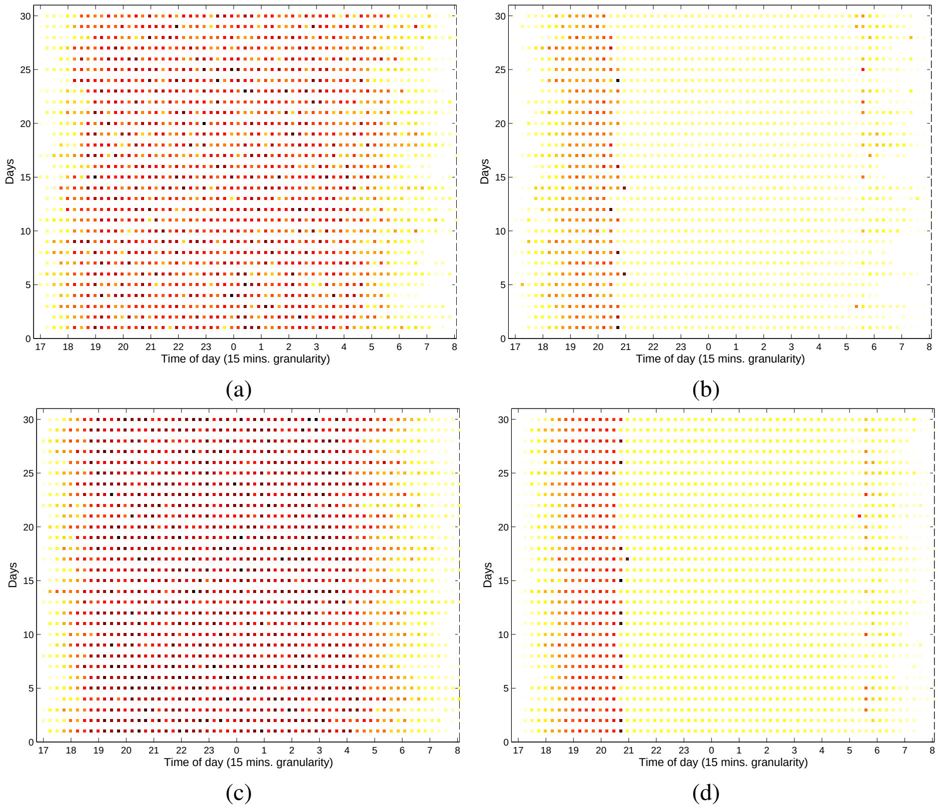


Fig. 6. Energy quantity taken from 11 (a, b) and 21 (c, d) electric vehicles over the range of time period studied in this work, from 17:00 until 8:00 (shown in the x-axis), for 30 days (shown in the y-axis) to help with the energy consumption of 40 (a, b) and 80 (c, d) household units. Darker-filled circles represent higher energy quantity taken from the PEVs' batteries. The enriched cost function, described in Equation 8, follows a well-defined desired pattern (b, d), whereas the cost function that tends to find local optimum solutions, described in Equation 7, tends to have a rather undesirable random pattern (a, c).

defined in Equation 7. The consumption pattern of this is shown in Figure 6 (a) and (c) for 40 and 80 household units, respectively. It should be noted that the higher the consumption from batteries is, the darker the dot. We can see that a random pattern is achieved by the cost function shown in Equation 7. That is, for every recorded day, shown in the y-axis, the amount of energy taken from the batteries is rather random regardless of the period time, shown in the x-axis, except from 17:00-18:30 and 6:30 – 8:00, where the consumption from batteries is low. This can be explained due to the availability of PEVs during these periods. That is, as indicated in Section 3, each PEV has its own time of arrival and departure which varies during these periods of time.

We continue our analysis on the proposed enriched maximisation cost function, see Equation 8, that aims at using the most amount of energy from the batteries of the PEVs while ensuring that each has a minimum SoC at the time of departure, and that tries to reduce the highest peak loads. The consumption pattern from the batteries is shown in Figure 6 (b) and (d) for 40 and 80 household units, respectively. This is a mirror image of what we discussed in the previous paragraph. That is, there is a well-defined pattern for each of the simulated days, shown in the *y*-axis, during the period of study, shown in the *x*-axis of the figure. We can observe that this cost function indeed achieves at using the most amount of energy when it is needed the most (high-peaks) as shown by the darker-filled squares while ensuring that the constraints are not violated (e.g., minimum SoC at the time of departure).

4.3 Peak-To-Average Ratio

As indicated previously, the peak-to-average ratio (PAR) is calculated by the maximum load demand for a period of time over the average load demand for the same period. It has been shown that a lower PAR is preferred [16].

We calculated the PAR considering the consumption from the substation transformer. Figure 7 shows the PAR for 40 (left-hand side) and 80 (right-hand side) household units for each of the 30 working simulated days using our proposed approach. It is easy to observe that a higher PAR is achieved by the fitness (cost) function formally defined in Equation 7, which goal is to use the most amount of energy from PEVs' batteries while at the same time aims at guaranteeing that each PEV has a minimum SoC at the time of departure compared to that PAR achieved by the enriched fitness function formally described in Equation 8 that is built on the top of Equation 7, which also tries to reduce the highest peak loads.

This, in fact, is to be expected given that the fitness function described in Equation 8 does consider an associated ranking system (recall that a third of time slots are considered critical, i.e., high peak period) that is able to reflect smoothly the consumption from the substation transformer as shown by the low PAR achieved by this enriched fitness function for each day of the 30 simulated days, denoted by the white-filled bars in Figure 7.

5 CONCLUSIONS

Demand-Side Management (DSM) refers to programs that aim to control the energy consumption at the customer side of the meter. Different techniques have been proposed to achieve this. The most popular techniques are those based on smart pricing (e.g., critical-peak pricing, real-time pricing). One major limitation of smart pricing is the fact that the electricity price is proportional to the electricity demand. This is particularly true for the time-of-use smart pricing adopted in some countries, where there is a financial incentive to use the electricity at night given its lower cost compared to its cost during day time. To alleviate this problem, we proposed the development of a demand-side *autonomous intelligent* management system that exploit plug-in electric vehicles' (PEV) batteries. More precisely, our system uses the PEV's batteries to partially and temporarily fulfil the demand of end-use consumers instead of using only the electricity available from a substation transformer.

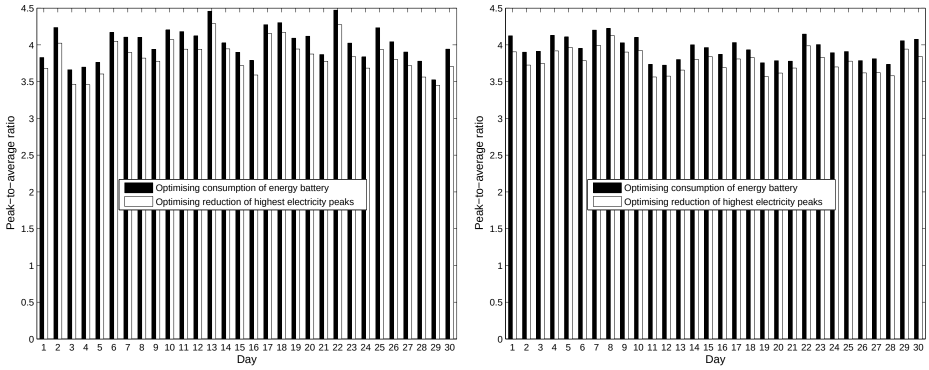


Fig. 7. Peak-to-average ratio (PAR) load demand achieved by our proposed approach when trying to maximise energy consumption from PEVs' batteries (black-filled bars) vs. when trying to maximise energy consumption from PEVs' batteries while aiming at reducing highest load peaks (white-filled bars), for 40 and 80 household units shown at the left-hand side and right-hand side of the figure, respectively. A lower PAR is preferred.

To this end, we used an stochastic bio-inspired method, differential evolution, given its natural representation (encoding of a solution) that allows to make fine-grained decision in terms of the exact energy that can be taken from PEVs' batteries to partially and temporarily fulfil energy requirements from a set of household units. To effectively do so, we proposed two fitness (cost) functions that achieve: (a) to use the maximum allowed energy from PEVs while still guaranteeing they can complete a journey, and (b) to use the maximum energy consumption from PEVs batteries while considering reducing high-peak periods.

From experimental results, it is clear that the enriched fitness function is able to use the most amount of energy from PEVs, it is also able to reduce peak loads and it is also able to achieve a lower PAR compared to the other 'simple' fitness function proposed in this work.

Acknowledgements

Edgar Galván López's research is funded by an ELEVATE Fellowship, the Irish Research Council's Career Development Fellowship co-funded by Marie Curie Actions. The first author would also like to thank the TAO group at INRIA Saclay & LRI - Univ. Paris-Sud and CNRS, Orsay, France for hosting him during the outgoing phase of the ELEVATE Fellowship. The authors would like to thank all the reviewers for their useful comments that helped us to significantly improve our work.

References

1. T. Bäck, D. B. Fogel, Z. Michalewicz (Eds.), Evolutionary Computation 1: Basic Algorithms and Operators, IOP Publishing Ltd., Bristol, UK, 1999.

2. A. E. Eiben, J. E. Smith, Introduction to Evolutionary Computing, Springer Verlag, 2003.
3. E. Galván-López, J. McDermott, M. O'Neill, A. Brabazon, Defining locality in genetic programming to predict performance, in: IEEE Congress on Evolutionary Computation, IEEE, 2010, pp. 1–8.
4. D. Fagan, M. ONeill, E. Galván-López, A. Brabazon, S. McGarraghy, An analysis of genotype-phenotype maps in grammatical evolution, in: A. Esparcia-Alczar, A. Ekrt, S. Silva, S. Dignum, A. Uyar (Eds.), Genetic Programming, Vol. 6021 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2010, pp. 62–73.
5. E. Galván-López, S. Dignum, R. Poli, The effects of constant neutrality on performance and problem hardness in gp, in: Proceedings of the 11th European conference on Genetic programming, EuroGP'08, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 312–324.
URL http://dl.acm.org/citation.cfm?id=1792694.1792723
6. J. McDermott, E. Galván-López, M. ONeill, A fine-grained view of gp locality with binary decision diagrams as ant phenotypes, in: R. Schaefer, C. Cotta, J. KoÅodziej, G. Rudolph (Eds.), Parallel Problem Solving from Nature, PPSN XI, Vol. 6238 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2010, pp. 164–173.
URL http://dx.doi.org/10.1007/978-3-642-15844-5_17
7. J. Lohn, G. Hornby, D. Linden, An evolved antenna for deployment on nasas space technology 5 mission, in: U.-M. OReilly, T. Yu, R. Riolo, B. Worzel (Eds.), Genetic Programming Theory and Practice II, Vol. 8 of Genetic Programming, Springer US, 2005, pp. 301–315.
URL http://dx.doi.org/10.1007/0-387-23254-0_18
8. E. Galván-López, J. M. Swafford, M. O'Neill, A. Brabazon, Evolving a ms. pacman controller using grammatical evolution, in: C. D. Chio, S. Cagnoni, C. Cotta, M. Ebner, A. Ekárt, A. Esparcia-Alcázar, C. K. Goh, J. J. M. Guervós, F. Neri, M. Preuss, J. Togelius, G. N. Yannakakis (Eds.), EvoApplications (1), Vol. 6024 of Lecture Notes in Computer Science, Springer, 2010, pp. 161–170.
9. B. Cody-Kenny, E. Galván-López, S. Barrett, locogp: Improving performance by genetic programming java source code, in: Proceedings of the Companion Publication of the 2015 on Genetic and Evolutionary Computation Conference, GECCO Companion '15, ACM, New York, NY, USA, 2015, pp. 811–818. doi:10.1145/2739482.2768419.
URL http://doi.acm.org/10.1145/2739482.2768419
10. E. Galván-López, R. Poli, C. Coello, Reusing code in genetic programming, in: M. Keijzer, U.-M. OReilly, S. Lucas, E. Costa, T. Soule (Eds.), Genetic Programming, Vol. 3003 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2004, pp. 359–368.
URL http://dx.doi.org/10.1007/978-3-540-24650-3_34
11. E. Galván-López, Efficient graph-based genetic programming representation with multiple outputs, International Journal of Automation and Computing 5 (1) (2008) 81–89. doi:10.1007/s11633-008-0081-4.
URL http://dx.doi.org/10.1007/s11633-008-0081-4
12. G. M. Masters, Renewable and Efficient Electric Power Systems, Wiley-Interscience, 2004.
13. E. Galván-López, C. Harris, L. Trujillo, K. R. Vázquez, S. Clarke, V. Cahill, Autonomous demand-side management system based on monte carlo tree search, in: IEEE International Energy Conference (EnergyCon), IEEE Press, Dubrovnik, Croatia, 2014, pp. 1325 – 1332.
14. Pacific Northwest GridWise Testbed Demonstration Projects, Part I. Olympic Peninsula Project (October 2007).
15. E. Galvan, C. Harris, I. Dusparic, S. Clarke, V. Cahill, Reducing electricity costs in a dynamic pricing environment, in: Proc. Third IEEE International Conference on Smart Grid Communications (SmartGridComm), IEEE Press, Tainan, Taiwan, 2012, pp. 169 – 174.
16. A. Mohsenian-Rad, V. Wong, J. Jatskevich, R. Schober, A. Leon-Garcia, Autonomous demand-side management based on game-theoretic energy consumption scheduling for

the future smart grid, Smart Grid, IEEE Transactions on 1 (3) (2010) 320 –331. doi:10.1109/TSG.2010.2089069.

17. E. Galván-López, T. Curran, J. McDermott, P. Carroll, Design of an autonomous intelligent demand-side management system using stochastic optimisation evolutionary algorithms, Neurocomputing 170 (2015) 270–285. doi:http://dx.doi.org/10.1016/j.neucom.2015.03.093. URL http://www.sciencedirect.com/science/article/pii/S0925231215009303
18. W. Kempton, S. E. Letendre, Electric vehicles as a new power source for electric utilities, Transportation Research Part D: Transport and Environment 2 (3) (1997) 157 – 175. doi:http://dx.doi.org/10.1016/S1361-9209(97)00001-1. URL http://www.sciencedirect.com/science/article/pii/S1361920997000011
19. W. Kempton, J. Tomic, Vehicle-to-grid power fundamentals: Calculating capacity and net revenue, Journal of Power Sources 144 (1) (2005) 268–279. doi:10.1016/j.jpowsour.2004.12.025. URL http://dx.doi.org/10.1016/j.jpowsour.2004.12.025
20. A. Brooks, E. Lu, D. Reicher, C. Spirakis, B. Weihl, Demand dispatch: Using real-time control of demand to help balance generation and load, IEEE Power & Energy Magazine, 8 (2010) 20 – 29.
21. R. Storn, K. Price, , Journal of Global Optimization 11 (4) (1997) 341–359. doi:10.1023/A:1008202821328. URL http://dx.doi.org/10.1023/A%3A1008202821328
22. R. Poli, Analysis of the publications on the applications of particle swarm optimisation, J. Artif. Evol. App. 2008 (2008) 4:1–4:10. doi:10.1155/2008/685175. URL http://dx.doi.org/10.1155/2008/685175
23. K. Y. C. C. et al., Large-scale energy storage systems ise2. london, u.k.: Imperial college london.
24. Z. Wang, C. Gu, F. Li, P. Bale, H. Sun, Active demand response using shared energy storage for household energy management, Smart Grid, IEEE Transactions on 4 (4) (2013) 1888–1897. doi:10.1109/TSG.2013.2258046.
25. J. M. Eyer, G. P. Corey, Energy Storage for the Electricity Grid: Benefits and Market Potential Assessment Guide. A study for the DOE Energy Storage Systems Program., Prepared by Sandia National Laboratories.
26. J. M. Eyer, J. J. Iannucci, G. P. Corey, Energy storage benefits and market analysis handbook: a study for the DOE Energy Storage Systems Program., Prepared by Sandia National Laboratories.
27. A. Mohd, E. Ortjohann, A. Schmelter, N. Hamsic, D. Morton, Challenges in integrating distributed energy storage systems into future smart grid, in: Industrial Electronics, 2008. ISIE 2008. IEEE International Symposium on, 2008, pp. 1627–1632. doi:10.1109/ISIE.2008.4676896.
28. D. E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, 1st Edition, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1989.
29. J. R. Koza, Genetic Programming: On the Programming of Computers by Means of Natural Selection, MIT Press, Cambridge, MA, USA, 1992.
30. A. K. Qin, V. L. Huang, P. N. Suganthan, Differential evolution algorithm with strategy adaptation for global numerical optimization, Trans. Evol. Comp 13 (2) (2009) 398–417. doi:10.1109/TEVC.2008.927706. URL http://dx.doi.org/10.1109/TEVC.2008.927706