

Optimal Bidding in Local Energy Markets using Evolutionary Computation

Fernando Lezama and Joao Soares
 GECAD Research Group, Polytechnic of Porto (ISEP/IPP),
 Porto, Portugal
 Email: {flz, jan}@isep.ipp.pt

Zita Vale
 Polytechnic of Porto (ISEP/IPP),
 Porto, Portugal
 Email: zav@isep.ipp.pt

Abstract—Increased adoption of distributed resources and renewables in distribution networks has led to a significant interest in local energy transactions at lower levels of the energy supply chain. Local energy markets (LM) are expected to play a crucial part in guaranteeing the balance between generation and consumption and contribute to the reduction of carbon emissions. Besides, LMs aim at increasing the participation of small end-users in energy transactions, setting the stage for transactive energy systems. In this work, we explore the use of evolutionary algorithms (EAs) to solve a bi-level optimization problem that arises when trading energy in an LM. We compare the performance of different EAs under a realistic case study with nine agents trading energy in the day-ahead LM. Results suggest that EAs can provide solutions in which all agents can improve their profits. It is shown the advantages in terms of profits that an LM can bring to market participants, thereby increasing the tolerable penetration of renewable resources and facilitating the energy transition.

Index Terms—Bi-level optimization; Differential evolution; Evolutionary computation; Local energy market; Renewable energy.

I. INTRODUCTION

The dawn of Smart Grids (SG) is raising new opportunities and challenges that must be sought to attain efficiency and environmental goals [1]. The rapid introduction of dispersed renewable energy sources has led to significant interest in local energy transactions at the lower level [2]. In this context, local energy markets (LM) provide an opportunity to trade renewable energy at the local level and contribute to the reduction of carbon emissions. This will empower the end-users to take a large part in the energy community while promoting a sustainable electricity transition [3]. Trading in LMs can be complex and time-intensive. Therefore authors in [4] proposed an automated negotiation based on intelligent agents. A project in Germany, named LAMP, has been testing LMs using innovative blockchain technology [5]. With LAMP, it is possible to evaluate the projected market prices and level of self-consumption.

This research has received funding from FEDER funds through the Operational Programme for Competitiveness and Internationalization (COMPETE 2020), under Project POCI-01-0145-FEDER-028983; by National Funds under FCT Portuguese Foundation for Science and Technology, under Projects PTDC/EEI-EEE/28983/2017(CENERGETIC) and UID/EEA/00760/2019; Joao Soares is supported by FCT under CEECIND/02814/2017 grant.

Evolutionary Computation (EC) is particularly an interesting area for solving complex optimization problems in several domains [1]. In this paper, we adopt EC as the primary set of tools to address the optimal bidding in LMs. We believe that EC can address the bi-level optimization problem in reasonable computation time without suffering the typical computational burden seen in deterministic methods. Results show that agents participating in LMs using a bidding strategy based on social welfare can obtain higher profits compared to the case of no LM access. The research outcomes are further discussed in the remaining sections of the paper.

II. PROBLEM FORMULATION

We consider a day-ahead LM bidding optimization problem, in which agents submit bids/offers to maximize their profits (or equivalently minimize their costs). We assume that agents of the type consumers only, small producers, and prosumers (i.e., consumers with generation capabilities) coexist in this LM. Also, agents have access to the main grid, which works as a back-up system. Therefore, as in [4], agents can trade energy in the LM with prices between the feed-in tariff (c_F) and the grid electricity tariff (c_G). In fact, it is assumed that $c_F < c_G$ and therefore buy/sell energy from the grid is less beneficial to agents than transacting energy in the LM. Figure 1 illustrates the local market scenario.

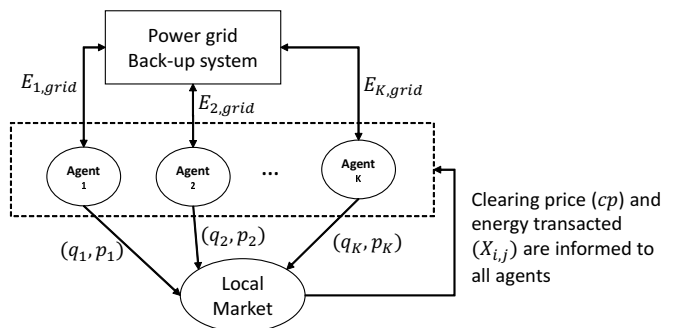


Fig. 1: Considered LM and the grid as back-up system.

A. Bi-level optimization problem

The LM bidding optimization problem can be modeled as a bi-level optimization problem. The upper-level corresponds to the maximization of agents' profits, and the lower-level

problem corresponds to the maximization of energy transacted in the LM. Therefore, the solution of the lower-level (after determining the clearing price) affects the upper-level by modifying the profits of all agents.

Consider a set of consumer agents $I = 1, 2, \dots, N_c$, and producer agents $J = 1, 2, \dots, N_p$, where each agent i wants to minimize its costs while agents j want to maximize their profits. The upper problem, therefore, is a multi-objective problem in which each agent wants to maximize/minimize their profits/costs.

The optimization problem for each consumer agents (minimization of costs) can be formulated as:

$$\text{minimize } C_i = \sum_j cp \cdot x_{j,i} + c^G \cdot Ebuy_{grid,i} \quad (1)$$

where cp is the LM clearing price, $x_{j,i}$ contains the energy sold by agent j to agent i in the LM, c^G is the grid price, and $Ebuy_{grid,i}$ is the energy bought by agent i from the grid.

On the other hand, producer agents try to maximize their profits regarding their production marginal cost as follows:

$$\text{maximize } P_j = \sum_i cp \cdot x_{j,i} + c^F \cdot Esell_{j,grid} - c^m \cdot G_j \quad (2)$$

where cp is the LM clearing price (equal for buyers and sellers), $x_{j,i}$ contains the energy sold by agent j to agent i , c^F is the feed-in tariff, $Esell_{j,grid}$ is the energy sold by agent j to the grid, and $c^m \cdot G_j$ represents the marginal cost associated to j . We assume that $c^m = 0$ for PV generation, and $c^m = c_{CHP}^m(G_j)$ (i.e., the marginal cost associated to a combined heat and power (CHP) generator) is defined as a monotone decreasing function [2]:

$$c_{CHP}^m(G_j) = \frac{b_{CHP} \cdot \sqrt{G_j}}{G_j} \quad (3)$$

where b_{CHP} is a cost factor of the CHP generation unit and G_j is the energy produced by the CHP.

The agents' profits/costs are influenced by the LM clearing price cp , which is determined in the lower-level problem and depends on the bidding process. The lower-level problem is modelled here as an asymmetric pool market in which bids and offers are allocated using a merit order procedure to determine the supply and demand curves for energy [6]. The clearing price is determined as the price in which demand equals supply (i.e., the *equilibrium price* (i.e., the clearing price)). Each agent's bid influences the market price, and therefore a strong inter-dependency of profits/costs exists. Since we are interested in increasing the overall mean profits of the system and at the same time provide solutions that distribute the earns among all agents, we model the objective function as:

$$\text{max } \sum_j (P_j)/N_p - \sum_i (C_i)/N_c \quad (4)$$

where P_j and C_i are objectives in conflict since agents want to achieve the best result for their own. In this paper, we

avoid a multi-objective formulation of the problem by using the summation of all individual costs. This can be done by considering that the solution of Eq. 4 is Pareto optimal for the multi-objective optimization problem [7]. A multi-objective formulation can be left as future work.

III. EVOLUTIONARY COMPUTATION TO SOLVE THE LM BI-LEVEL OPTIMIZATION PROBLEM

Evolutionary computation (EC) span algorithms for global optimization mostly inspired by evolutionary processes [1]. Typically, evolutionary algorithms (EA) are population-based solvers that act over an initial set of candidate solutions (i.e., a population or swarm) that is iteratively updated. The performance of solutions is measured by a given fitness function. Every iteration, solutions generated using specific operators are introduced into the population, replacing solutions with low performance. Thus, the population will gradually evolve towards a promising area of the search space following the principles of natural/artificial selection [1].

Many approaches have been developed in the field of SI and EC (e.g., particle swarm optimization (PSO), differential evolution (DE), and all their variants, just to mention a few). In fact, EAs can target different problems by defining an encoding of solutions (i.e., a way to represent solutions to the problem typically as vectors) and a fitness function to evaluate such solutions. Therefore, how to encode individuals for this problem, the fitness function, and the EC approaches used to solve the bi-level problem, are introduced in this section.

A. Encoding of solutions and fitness function

The optimization problem, seen as a whole, searches for the optimal bidding of agents in the LM to maximize their profits. Therefore, assuming we have $K = \{1, 2, \dots, N_k\}^1$, we seek to determine the best tuple $(q_k, p_k) \forall k \in K$ representing the optimal price and quantity to bid in the LM for each agent. The bidding also should be done for all $t \in T$ periods of the optimization process (i.e., $T = 24$ periods in the day-ahead market). Therefore, we define a vector $\vec{x} = \{[q_{k,t}] \cup [p_{k,t}]\}$ including the bids for quantity and price the k th agent will send to the LM. To avoid separating the agents by consumer and producer types, we use a sign convention in which a positive quantity represents a bid (i.e., buying in the market), while a negative quantity represents an offer (i.e., selling in the LM). Therefore, we can control the agent action by defining variable bounds in which a consumer agent can send bids in the market within the bounds $[0, L_{max}]$ (i.e., between 0 and their maximum consumption), while producer agents can send offers within the bounds $[-P_{max}, 0]$ (i.e., between 0 and their maximum production capacity). The bounds for prices are the same for all agents and within the range $[c^F, c^G]$. Figure 2 illustrates the structure of solutions to understand how the individual is encoded. Such solutions should be evaluated in an objective function (OF) that returns the mean average profit of all agents plus the standard deviation:

¹Notice that, different from Section II, in which producers and consumers are separated in two groups, this K set includes all agent

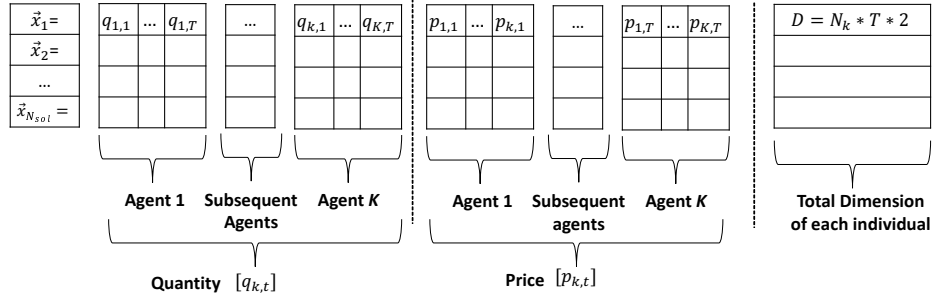


Fig. 2: Encoding of each individual in the vector space.

$$Fitness(\vec{x}) = -mean(Profits) + std(Profits) \quad (5)$$

where $mean(Profits)$ and $std(Profits)$ are functions that compute the average and standard deviation of the profits that all agents obtained considering the bids/offers encoded in the individual. The negative sign in the first term is used to transform the profits maximization problem into a minimization one. The less the value in Eq. (5), the better the mean profits achieved by all agents. $Profits \forall k \in K$ can be computed using Eqs. (1) and (2). The fitness function should perform a series of steps to be evaluated. In a first step, bids and offers are decoded from the vector and send to the market-clearing price function that uses a merit order mechanism to determine the clearing price (cp) and energy transacted between agents. After that, the individual profit of agents can be calculated using Eqs. (1) and (2). A simple deterministic mechanism is used internally to determine the amount of energy traded to the grid (i.e., variables $Ebuy_{i,grid}$ and $Esell_{j,grid}$ in Eqs. (1) and (2)). For instance, in the case of consumer i , $Ebuy_{i,grid} = L_{max,i} - \sum_j x_{j,i}$ which is equal to the remaining energy not obtained in the LM to satisfy its demand. On the other hand, producer j determines the energy sold to the grid according to a simple rule regarding its marginal cost c^m :

$$Esell_{j,grid} = \begin{cases} P_{max,j} - \sum_i x_{j,i} & \text{if } P_{marginal,j} > P_j \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

where $P_{marginal,j}$ are the profits that agent j could make considering the marginal cost of producing its maximum capacity and selling the non-traded energy in the LM to the grid; while P_j are the regular profits that the agent would make by selling only into the LM. With this simple heuristic, we guarantee that producers only sell to the grid if they can actually get some profits with their remaining capacity. Notice that, in the case of PV generation where the marginal cost is assumed to be 0, the agents will always sell its remaining energy to the grid according to this rule.

With all the available information, we can compute the profits/costs of all agents and store them in a vector:

$$Profits = [P_1, \dots, P_k] \forall t \in T \quad (7)$$

This vector includes all the profits agents achieved considering the bidding in \vec{x} , and therefore can be easily evaluated in Eq. (5).

IV. EVOLUTIONARY ALGORITHMS

Now that we defined the encoding of individuals and the fitness function, we can basically apply any EA to solve the problem. In this paper, we have chosen differential evolution (DE) and two of its variants HyDE and HyDE-DF (due to its success in many applications and easy implementation [8]), an improved version of the well-known Particle swarm optimization (PSO) [9], and the vortex search (VS) [10].

A. Differential evolution

DE uses a population (Pop) of individuals $\vec{x}_{j,i,G} = [x_{1,i,G}, \dots, x_{D,i,G}]$, where G is the generation number, and $i = [1, \dots, NP]$ is the number of individuals in the population, to optimize a function with D variables (i.e., the dimension of the problem). In the initialization stage, NP solutions are generated randomly within the range lower and upper ranges $[x_{lb,j}, x_{ub,j}]$ (i.e., $[0, L_{max}]$ for consumers bid quantities, $[-P_{max}, 0]$ for producers offers quantities, and $[c^F, c^G]$ for prices). DE follows the general iterative from of EAs by creating new solutions applying a mutation and recombination operator, and performing elitist selection (solutions with better profit survive for the next generation) in each iteration.

The mutation and recombination operator of the DE/rand/1 algorithm is defined as:

$$\vec{t}_{j,i,G} = \begin{cases} \vec{m}_{j,i,G} & \text{if } (rand_{i,j}[0,1] < Cr) \vee (j = Rnd) \\ \vec{x}_{j,i,G} & \text{otherwise} \end{cases} \quad (8)$$

where $\vec{m}_{i,G} = \vec{x}_{r1,G} + F(\vec{x}_{r2,G} - \vec{x}_{r3,G})$, and $\vec{x}_{r1,G}, \vec{x}_{r2,G}, \vec{x}_{r3,G} \in \text{Pop}$ are three random individuals from the Pop, mutually different from each other. F and Cr are the mutation and recombination parameters of DE, usually set in the range $[0, 1]$. The fitness function, (i.e., Eq. (5)), is used to evaluated the performance of new individuals. A detailed explanation of DE can be found in [11], [12].

B. HyDE and HyDE-DF

Hybrid-adaptive DE (HyDE) is a new self-adaptive version of DE proposed in [12]. HyDE uses an operator known as

DE/target-to-perturbed_{best}/1 that modifies the well-known DE/target-to-best/1 strategy [8] with a perturbation of the best individual (inspired by the evolutionary PSO [13]), and the self-adaptive mechanism of jDE [14]. The main operator of HyDE is defined as follows:

$$\vec{m}_{i,G} = \vec{x}_{i,G} + F_i^1(\epsilon \cdot \vec{x}_{best} - \vec{x}_{i,G}) + F_i^2(\vec{x}_{r1,G} - \vec{x}_{r2,G}) \quad (9)$$

where F_i^1 and F_i^2 , are scale factors in the range $[0, 1]$ independent for each individual i , and $\epsilon = \mathcal{N}(F_i^3, 1)$ is a random perturbation factor taken from a normal distribution with mean F_i^3 and standard deviation 1. F_i^1 , F_i^2 and F_i^3 are updated each iteration following the same rule of jDE algorithm (see Sect. III.B of [12]).

HyDE with decay function (HyDE-DF) is an improved version of HyDE used for function optimization [15]. It incorporates a decay function to perform a transition in the iteration process from the main operator of HyDE (Eq. 9) to the basic operator of DE (Eq. 7). This transition allows an enhance phase of exploration in the early stage of evolution and stress the exploitation in later stages of the optimization. HyDE-DF achieved third place (out of 36 algorithms) in the 100-digit challenge in CEC/GECCO 2019.

C. PSO and VS

We include in this study two more metaheuristic with a similar framework as the DE algorithm. Strictly speaking, PSO [9] belongs to the class of swarm intelligence (SI), in which particles (solutions to the problem) coordinate their actions by modifying their position towards the optimum value. Particles are evaluated in the fitness function and improve their position in each iteration (similar to the iterative process in EAs).

Vortex search (VS), on the contrary, is classified as a single-solution based metaheuristic, although it has an analogous framework to the EAs as well. In each iteration, N given number of neighbor solutions are generated using a multivariate Gaussian distribution around the initial solution. Those N solutions are evaluated in the fitness function, and the single-solution is updated with the best solution found. The iterative process is repeated until a stop criterion is met [10].

We do not included detailed explanations of the selected EAs for space limitations, but the reader can consult the cited references for further details.

V. RESULTS AND DISCUSSION

In this section, the case study and the application of a set of EAs to solve the optimal bidding problem in LM are presented.

A. Case Studies

We adopt a case study with nine agents, in which 3 of them are consumers, 3 are prosumers (i.e., consumers with PV generation capabilities), and 3 are CHP generators. To generate case study data, sample power profiles of residential houses and PV systems are built using the open datasets available in PES ISS website². We build three standard house power

profiles and a PV power profile (see Fig. 3). With these profiles, we generate agent data using a randomized function with uniform distribution, 20% around the standard profiles. Fig. 3 also provides the power ranges of the base profiles. We also consider generator agents corresponding to CHPs with a maximum generation capacity of 2kW and a marginal cost calculated with Eq. (3) with a factor $b_{CHP} = 0.18$ EUR/kWh [2]. Finally, feed-in and grid tariffs are set to $c^F = 0.12$ and $c^G = 0.28$ EUR/kWh as in [2].

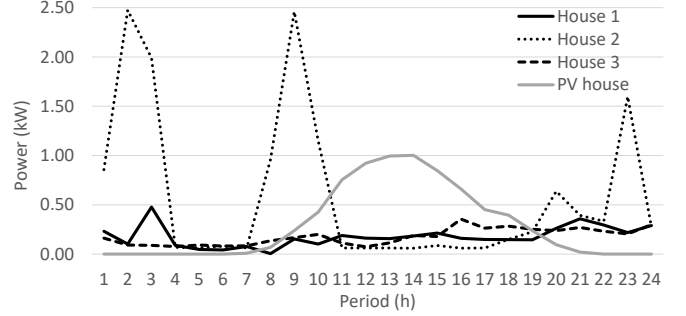


Fig. 3: Profiles used in the case study. Ranges of power (in kW): house 1 [0.18-0.48], house 2 [0.06-2.50], house 3 [0.07-0.36], PV (house) [0-1].

B. Algorithm settings

We have compared the performance of five EAs to solve the bi-level optimization problem proposed in Sect. II, namely the well-known DE [16], the more recently proposed HyDE and HyDE-DF [12], [15], an improved PSO [9], and the VS [10]. DE, HyDE, HyDE-DF, and PSO are population-based approaches with similar mechanisms, performing a number of function evaluations (FE) equal to the size of the population in each iteration. VS is not strictly speaking a population-based approach but evaluates in each iteration a given number of neighbor solutions, which results in the same number of FE per iteration as the other selected algorithms. The size of population/neighbor solutions has been set to 5 in a first experiment, and 20 in a second experiment, with a budget of 2000 iterations for all the algorithms.

The parameters for each algorithm were chosen according to other studies. For DE, the mutation factor and recombination constant (F and Cr) were set to the recommended values 0.5 and 0.9 respectively [16]. HyDE and HyDE-DF³ [12] are a self-adaptive parameter versions but initial values for F^i and Cr where set to 0.5. VS algorithm does not have any parameter to configure [10].

C. Performance comparison and results

The algorithms were run for 30 trials each, and the reported results correspond to the mean value of those 30 tests. Table I presents the total costs of the system, the costs by group of agents (i.e., consumers, prosumers, and producers), and

²Open data online at <http://sites.ieee.org/pes-iss/data-sets/>

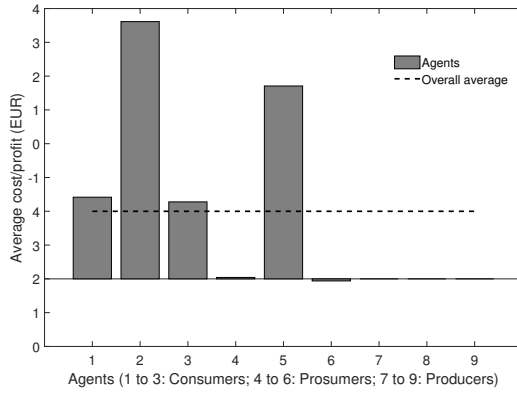
³HyDE and HyDE-DF algorithm implementation is available at: <https://fernandolezama.github.io/publication>

TABLE I: Overall costs/profits, fitness, and optimization time achieved by the tested EAs.

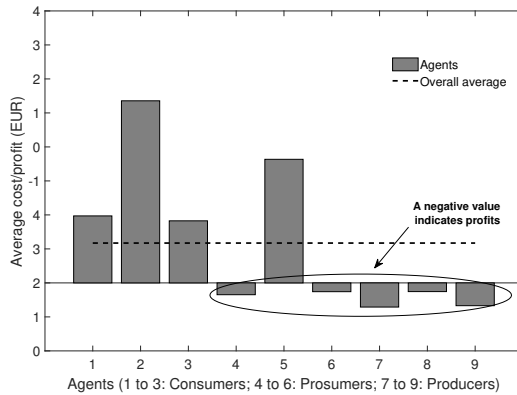
Population Size	EA	Overall Costs (EUR)	Costs/profits by group of agents (EUR)			Fitness	Std	Time (mins)
			Costs Consumers	Costs Prosumers	Profits Producers			
5	PSO	6.61	5.22	2.04	0.65	3.97	0.07	4.55
	DE	6.30	5.03	1.92	0.65	3.82	0.08	5.32
	HyDE	6.65	5.25	2.03	0.62	3.98	0.05	4.93
	HyDE-DF	6.48	5.18	2.02	0.72	3.92	0.06	4.99
	VS	5.95	4.90	1.83	0.78	3.68	0.08	5.40
20	PSO	6.42	5.13	1.97	0.67	3.88	0.05	19.06
	DE	5.78	4.89	1.75	0.87	3.61	0.05	20.02
	HyDE	6.44	5.19	1.98	0.73	3.90	0.05	18.25
	HyDE-DF	6.25	5.10	1.93	0.78	3.83	0.06	19.20
	VS	5.58	4.74	1.65	0.82	3.49	0.08	18.85
No LM		9.00	6.15	2.84	0.00	4.97	0.00	0.00

the average fitness, standard deviation, and time required by each algorithm to perform the 2000 iterations. The table also includes the overall costs/profits that agents can attain in the absence of an LM (used as a baseline). It can be seen that VS achieves the best overall costs for the system (corresponding also to the lower fitness). Compared to the baseline (the row with No LM), all groups of agents improve their costs/profits (e.g., obtaining profits in the case of producer agents). In fact, in the baseline, producer agents present 0 profits due to their

unwillingness to sell their energy capacity to the grid at a feed-in tariff. This is explained because CHP generators have a marginal cost associated, which results in a higher price than the feed-in tariff. Therefore, generators would not sell the energy to the grid if they cannot recover at least their marginal cost. The same conclusion can be achieved by observing the rows corresponding to the experiment with 20 individuals, only that in this case, the costs/profits improved for all the applied EAs.

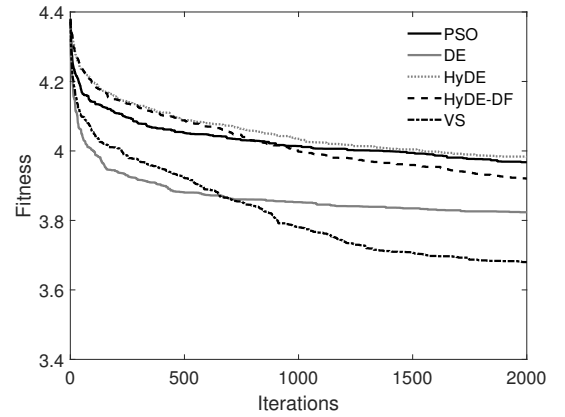


[a]

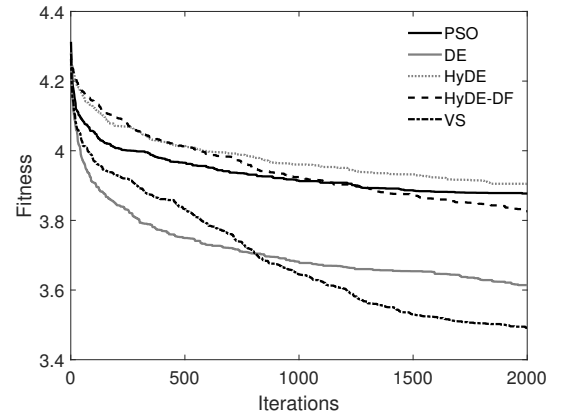


[b]

Fig. 4: Overall profits achieved by individual agents. [a] Baseline solution. [b] Best solution found with VS algorithm.



[a]



[b]

Fig. 5: Average convergence of tested EA. [a] Using $NP = 5$. [b] Using $NP = 20$.

To analyze the profits obtained by independent agents, we show a plot with the individual agents' profits. Figure 4a shows the individual profits achieved by the agents in the baseline case. This confirms that in the baseline when no LM is available, the best option for the generator agents is not selling anything to the grid. This turns in zero profits/costs for them, which is bad for the system as a whole. On the contrary, notice that the best solution found with VS (Fig. 4b achieves, on average, a higher mean profit value, and all agents improve their situation, i.e., consumer agents reduce their costs while generator agents (agents 7 to 9) and even prosumers (agents 4 to 6) have obtained some profits.

Finally, we analyze the convergence behavior of the algorithms in terms of their fitness value evolution. Fig. 5 shows the mean convergence curves of the EA using 5 and 20 solutions. It can be seen that the curves have practically the same shape for both cases, only that in the second case ($NP = 20$), the fitness achieved is slightly better for all the algorithms. Also, it can be observed that VS has a slower convergence on the first stage of evolution than DE, but in the long term, achieves the best performance. It is also interesting to appreciate that, despite the good performance showed by PSO, HyDE, and HyDE-DF in solving different optimization problems, they are outperformed by the simplest version of DE for this application. This might be another proof of the free-lunch theorem [17], and the analysis of such performance characteristics is an interesting venue for future research.

VI. CONCLUSIONS

In this paper, a set of EA has been applied to solve a bi-level optimization problem that arises in the context of LM. The problem of competitive agents is difficult to solve since all agents want to maximize their own profits, modifying the price in the lower level problem (i.e., the clearing price in the LM), resulting in a strong interdependence of their decisions. This problem can also be modeled as a multi-objective optimization problem, in which each agent wants to maximize their profits. We have modeled a fitness function aiming at the maximization of profits of all agents and a fair distribution of the benefits that the LM provides. However, several assumptions have been made to simplify the problem, and therefore, different venues for future research are opened. For instance, transactions in an LM must consider the effect in the distribution network and network constraints before implementation, so the model and simulation could be adapted accordingly. Also, the impact that bidding in the LM might have in other electricity markets (e.g., the wholesale market) is worth to be explored. Finally, the standard version of DE and VS algorithms showed the best performance in this application, so a margin for improvement must exist with specific heuristic rules for this problem.

- [1] J. Soares, T. Pinto, F. Lezama, and H. Morais, "Survey on complex optimization and simulation for the new power systems paradigm," *Complexity*, vol. 2018, 2018.
- [2] E. M. Mengelkamp, "Engineering local electricity markets for residential communities," Ph.D. dissertation, Karlsruher Institut für Technologie (KIT), 2019, 37.06.01; LK 01.
- [3] F. Lezama, J. Soares, P. Hernandez-Leal, M. Kaisers, T. Pinto, and Z. Vale, "Local energy markets: Paving the path towards fully transactive energy systems," *IEEE Transactions on Power Systems*, 2018.
- [4] E. Mengelkamp, J. Gärtner, and C. Weinhardt, "Intelligent agent strategies for residential customers in local electricity markets," in *Proceedings of the Ninth International Conference on Future Energy Systems*. ACM, 2018, pp. 97–107.
- [5] —, "Decentralizing energy systems through local energy markets: the lamp-project," in *Multikonferenz Wirtschaftsinformatik*, 2018.
- [6] S. Parsons, M. Marcinkiewicz, J. Niu, and S. Phelps, *Everything you wanted to know about double auctions, but were afraid to (bid or) ask*. Department of Computer & Information Science, University of New York, 2006.
- [7] M. R. Sandgani and S. Siropour, "Coordinated optimal dispatch of energy storage in a network of grid-connected microgrids," *IEEE Transactions on Sustainable Energy*, vol. 8, no. 3, pp. 1166–1176, July 2017.
- [8] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE transactions on evolutionary computation*, vol. 15, no. 1, pp. 4–31, 2011.
- [9] J. Soares, M. A. F. Ghazvini, M. Silva, and Z. Vale, "Multi-dimensional signaling method for population-based metaheuristics: Solving the large-scale scheduling problem in smart grids," *Swarm and Evolutionary Computation*, vol. 29, pp. 13–32, 2016.
- [10] B. Dogan and T. Olmez, "A new metaheuristic for numerical function optimization: Vortex search algorithm," *Information Sciences*, vol. 293, pp. 125 – 145, 2015.
- [11] S. Das, A. Abraham, U. K. Chakraborty, and A. Konar, "Differential evolution using a neighborhood-based mutation operator," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 3, pp. 526–553, 2009.
- [12] F. Lezama, J. a. Soares, R. Faia, T. Pinto, and Z. Vale, "A new hybrid-adaptive differential evolution for a smart grid application under uncertainty," in *IEEE Congress on Evolutionary Computation (CEC)*, July 2018, pp. 1–8.
- [13] V. Miranda and N. Fonseca, "EPSO-evolutionary particle swarm optimization, a new algorithm with applications in power systems," in *IEEE/PES Transmission and Distribution Conference and Exhibition*, vol. 2, Oct 2002, pp. 745–750.
- [14] J. Brest, S. Greiner, B. Boskovic, M. Mernik, and V. Zumer, "Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 6, pp. 646–657, Dec 2006.
- [15] F. Lezama, J. a. Soares, R. Faia, and Z. Vale, "Hybrid-adaptive differential evolution with decay function (HyDE-DF) applied to the 100-digit challenge competition on single objective numerical optimization," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, ser. GECCO '19, 2019, pp. 7–8.
- [16] K. Price, R. M. Storn, and J. A. Lampinen, *Differential evolution: a practical approach to global optimization*. Springer Science & Business Media, 2006.
- [17] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, April 1997.