

Distributed energy resource short-term scheduling using Signaled Particle Swarm Optimization

J. Soares, M. Silva, T. Sousa, Z. Vale, H. Morais

abstract

Distributed Energy Resources (DER) scheduling in smart grids presents a new challenge to system operators. The increase of new resources, such as storage systems and demand response programs, results in additional computational efforts for optimization problems. On the other hand, since natural resources, such as wind and sun, can only be precisely forecasted with small anticipation, short-term scheduling is especially relevant requiring a very good performance on large dimension problems. Traditional techniques such as Mixed-Integer Non-Linear Programming (MINLP) do not cope well with large scale problems. This type of problems can be appropriately addressed by metaheuristics approaches. This paper proposes a new methodology called Signaled Particle Swarm Optimization (SiPSO) to address the energy resources management problem in the scope of smart grids, with intensive use of DER. The proposed methodology's performance is illustrated by a case study with 99 distributed generators, 208 loads, and 27 storage units. The results are compared with those obtained in other methodologies, namely MINLP, Genetic Algorithm, original Particle Swarm Optimization (PSO), Evolutionary PSO, and New PSO. SiPSO performance is superior to the other tested PSO variants, demonstrating its adequacy to solve large dimension problems which require a decision in a short period of time.

Keywords:

Distributed energy resource scheduling, Mixed integer non-linear programming, Particle swarm optimization, Short-term scheduling

1. Introduction

Presently, Power Systems (PS) use a diversity of energy resources, including Distributed Generation (DG), especially based on Renewable Energy Sources (RES), storage units and Demand Response (DR). The integration of these resources and the establishment of liberalized and competitive markets require specific technical conditions which should be satisfied by the Smart Grid (SG) concept [1e3]. According to the Energy Independence and Security Act of 2007, several key features related to smart grid are reported in references [4e6].

The main difficulties faced with RES are the continuity and reliability problems associated with the unpredictable nature of the primary natural energy sources. The output of some renewable generation, such as wind generators and photovoltaic systems, is determined by the climate and weather conditions and operating patterns will therefore be constrained by these natural conditions.

The main problem with storage is the expensive investment that it requires, therefore limiting its use. In the SG context, these resources require new methodologies for control and operation.

The paper focuses on the short-term scheduling of energy resources in SG, considering intensive penetration of DG, on the storage and load curtailment opportunities enabled by demand response programs. Short-term economic dispatch [7e11] is a very relevant function in modern energy systems. It consists of programming the electric generation correctly in order to reduce the operational cost. Recently the use of wind power generation and photovoltaic units has significantly increased [12]. Additionally, demand response is presently recognized as a very relevant energy resource that should be considered jointly with generation and storage resources for cost optimization [13e15].

The use of deterministic optimization techniques to solve the problem of distributed energy resources scheduling requires significant computer resources and, for real power systems, often requires long execution times, which do not cope with operation requirements.

Distributed Energy Resources (DER) significantly increased the number of variables that must be considered in the economic dispatch problem. It is very easy to have thousands of variables on

a relatively small network. Therefore it is necessary to develop new methodologies to improve the efficiency of economic dispatch methods able to cope with the new paradigms of power systems, namely aiming to obtaining quick response for optimization problems with many variables.

Artificial intelligence (AI) techniques, namely metaheuristics inspired by biological processes, have advantages in terms of computational requirements compared with the traditional optimization techniques. Particle Swarm Optimization (PSO) is inspired by the social behavior of bird flocking and has been successfully used in many power systems problems [14,16e21].

The authors propose a new implementation methodology called Signaled Particle Swarm Optimization (SiPSO) to solve the DER short-term scheduling. The proposed SiPSO presents faster convergence, better robustness and reduces execution time for the same solution quality, when compared with classic PSO and some of its most successful variants. This allows addressing large and complex optimization problems with less computational resources. For that reason it is very advantageous for network operators, aggregators, and for individual players acting in the SG context, since it allows them to respond to price variations in real time.

This paper compares the use of five alternative methods for scheduling DER, namely Mixed Integer Non-Linear Programming (MINLP) implemented in General Algebraic Modeling System (GAMS™) [22], Genetic Algorithms (GA), Particle Swarm Optimization (PSO), Evolutionary Particle Swarm Optimization (EPSO) [23], New Particle Swarm Optimization (NPSO) [24] and the SiPSO, proposed by the authors, implemented in MATrix LABoratory (MATLAB®) [25].

After this introduction, Section 2 presents the proposed SiPSO methodology. Section 3 describes the problem formulation and the methodologies used to implement the short-term energy resources scheduling problem. Section 4 presents a case study with 99 distributed generators, 208 loads, and 27 storage units. Finally, Section 5 presents the most important conclusions of the present work.

2. Signaled Particle Swarm Optimization

The difficulties of traditional deterministic approaches, to address the scheduling of distributed energy resources in a realistic environment, motivate the use of AI methods. The high computational execution time to find the solution and memory intensive requirements are the most important difficulties that should be overcome, especially for medium and large networks and for players of medium and small size with limited computing resources. GAs and particle swarm intelligence have been used to solve some optimization problems in power systems with similar characteristics [2,14,16,18,26].

The traditional PSO relies on externally fixed particles' velocity limits, inertia, memory and cooperation weights without changing these values throughout the swarm search process (PSO iterations) [26,27]. In very complex problems this can compromise the diversity of the solutions because swarm movements are limited to the velocities and weights initially fixed.

To overcome this limitation several enhanced versions of the classic PSO have been proposed [18,23,24,28]. There are many other variants of PSO, some of them related to more specific problems (such as multi-objective optimization functions) [29e31]. As it is impractical to compare the proposed method with every other technique, the most referred and recent variants evidencing good results in problems with similar characteristics have been selected for comparison. In [23] the authors introduced mutation of the strategic parameters (inertia, memory, cooperation) and selection by stochastic tournament. The method is called Evolutionary Particle Swarm Optimization (EPSO) and proved to be proficient in

several optimization problems [23]. The authors also propose replicating the particles in order to increase the probability of finding more solutions that enhance the diversity of the search space.

In [24] the authors propose a modification of the velocity equation to include particle's bad experience component besides the global best memory introduced earlier. The bad experience component helps to remember its previously visited worst position. The method is called New Particle Swarm Optimization (NPSO). The authors claim superiority over conventional PSO in terms of convergence and robustness properties. The execution time is slightly higher when compared with classic PSO due to additional computation requirements to process bad experience component. There is no mutation process as in EPSO.

Although with EPSO it is possible to change weights through the search process adding more diversity to the search space, particles' velocity limits remain unchanged during the iterative process. In some cases it can be better to change the velocity limits based on an intelligent mechanism since mutation implemented in EPSO is still a stochastic process. This idea is discussed in the present paper and has originated a new method to implement this metaheuristic.

In the proposed method, mutation of the strategic parameters already seen in EPSO is used due to its benefits. The originality of the proposed methodology is in the variables that can be marked up to allow changing the maximum and minimum velocity limits throughout the search process. These changes happen according to the results of an intelligent mechanism. The proposed algorithm is called Signaled Particle Swarm Optimization (SiPSO).

In this paper a particle is a set of one or more variables that correspond to the problem's variables. The main innovative characteristic of SiPSO consists in the communication between the particles' evaluation stage and the SiPSO process. When evaluating a given solution, it is possible to conclude that changing certain variables in a specific direction (velocity) would improve the solution fitness or even help in constraint handling. Therefore, a mechanism called signaling has been adopted. This mechanism allows an intelligent adjustment of the velocity limits that are initially set. In the traditional version of PSO the velocity limits are prefixed and cannot be changed during PSO iterations, nor in EPSO or NPSO. In other words, SiPSO makes possible to boost the velocity magnitude during the evolving process in an intelligent way. Typically it is intended to boost the speed of a given particle's variable with the objective of charging its value significantly.

The methodology uses three strategic parameters (w_i) already seen in EPSO, namely: inertia, memory, and cooperation. At the beginning of the process the values of these weights are randomly generated between 0 and 1. After that, the particle's weights are changed in each iteration using a Gaussian mutation distribution according to (1):

$$*w_i = w_i + \delta N(0,1) \quad (1)$$

where:

* w_i New mutated weights of particle i

w_i Weights of particle i

δ learning parameter with a range between 0 and 1

A high value of δ adds more importance to mutation. $N(0,1)$ is a random number following a normal distribution with mean equal to 0 and variance equal to 1. Once again, the strategic parameters are limited to values between 0 and 1 in this stage.

The bad experience component from NPSO is not used in SiPSO, as we concluded with experiments that adding this component worsened the SiPSO performance. However, an implementation of NPSO, as proposed in [24], was carried out by the authors and is used in the case study for comparison purposes.

Eq. (2) allows the calculation of the new particle's velocity that depends on particle's present velocity, best past experience (memory) and group's experience (cooperation).

$$v_i = w_{i(inertia)} v_i + w_{i(memory)} (b_i - x_i) + w_{i(coop)} (bG - x_i) \quad (2)$$

where:

b_i Best past experience of particle i
 bG Best global experience of all the particles
 v_i Velocity of particle i
 v_i New calculated velocity of particle i
 x_i Position of particle i
 $w_{i(inertia)}$ Inertia weight component
 $w_{i(memory)}$ Memory weight component
 $w_{i(coop)}$ Cooperation weight component
 The new positions x_i for each particle are then calculated according to (3).

$$x_i = x_i + v_i \quad (3)$$

where:

x_i New calculated position of particle i
 After applying the movement equation to each particle, SiPSO evaluates the fitness of the new positions and the best bG solution is stored across iterations. During the evaluation, the variables that could improve the fitness function or eliminate constraints violations are marked. The identification of the variables that should be signaled depends on the optimization problem that is being addressed. For instance, in the minimization of an economic problem with X, Y, and Z variables where Z is the less expensive, the program can identify that Z is the cheaper one and that it should have a greater value. With this information SiPSO can try to increase its value across iterations without violating the constraints. The engineer or programmer should identify which variables are best suitable to be signaled during the evaluation stage and design an algorithm to identify which variables should be signaled across iterations to improve solution fitness or handle constraints in the best way. The criteria to define which variables are chosen to be signaled should include the following, (although they are not restricted to them):

- variables that can easily relive constraint violations if changed in a certain direction;
- variables that cannot be changed by direct repair method;
- variables that are not easily corrected by direct repair method and if changed in a certain direction could improve the fitness function.

A signaling vector for each particle is maintained across the process to enable the communication between evaluation and

SiPSO core process. These array elements can assume one of the following three values: 0, 1 or -1. The size of this array (number of columns) corresponds to the number of variables in the problem. The set of signaling vectors constitutes a signaling matrix for the swarm, with as many lines as the number of particles set in SiPSO. The value 0 means that a given variable has not been signaled. The value 1 means that the variable has been signaled to gain more speed in the positive direction and -1 means that the variable has been signaled to gain speed in the opposite direction.

The resulting new maximum and minimum velocity limits of a given particle's variable are evaluated according to (4) and (5), respectively:

$$MaxVel = MaxVel + Boost\ Speed * Signaling\ Positives \quad (4)$$

$$MinVel = MinVel + Boost\ Speed * Signaling\ Negatives \quad (5)$$

where:

$MaxVel$ Initial max. velocity of particles
 $Boost\ Speed$ Vector with the variables boost speed
 $Signaling\ Positives$ Vector with the signaled variables (positive velocity)
 $MinVel$ Initial min. velocity of particles
 $Signaling\ Negatives$ Vector with the signaled variables (negative velocity)

$Signaling\ Positives$ is obtained from the signaling vector, built with its positive values (equal to 1) and with zeros in the other positions. $Signaling\ Negatives$ is also obtained from the Signaling vector, being built with its negative values (equal to -1) and with zeros in the other positions.

Fig. 1 presents the signaling process of SiPSO. In the evaluation stage the variables are identified and in the movement stage the velocity limits of the marked variables are updated. In each SiPSO's iteration the velocity values are randomly generated between the lower and upper velocity limits.

In early versions of PSO, velocities are generated randomly only once, in the beginning of the process, according to the fixed maximum and minimum velocities of the variables' particle.

To better understand how the signaling mechanism works, let us follow a simple example for one particle considering that n variables are used in the search process.

Table 1 presents the data for a given particle with n variables ($V1$ to Vn column). This table presents the state of SiPSO variables before and after the signaling. The boost speed vector is initially fixed. The elements of the boost speed vector represent the variation on the speed to be used for each variable when it is signaled. Let us consider that the vectors were initialized as shown in Table 1. The signaling vector is always initialized with zeros at start.

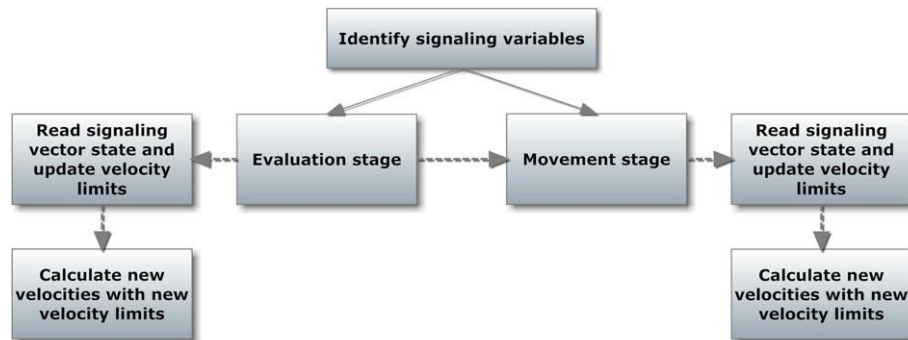


Fig. 1. SiPSO signaling process.

Table 1
SiPSO example.

			Variables					
			V ₁	V ₂	V ₃	V ₄	-	V _n
Signaling	Before	Signaling Vector	0	0	0	0	-	0
		Max. Velocity	10	10	20	10	-	10
		Min. Velocity	0	0	-10	0	-	-5
	Initially fixed	Boost Speed Vector	100	50	150	200	-	100
After		Signaling Vector	0	1	0	-1	-	0
		Max. Velocity	10	60	20	10	-	10
		Min. Velocity	0	0	-10	-200	-	10

The values for *max. velocity* and *min. velocity* in Table 1 represent the initial velocity limits for two different states, namely before the signaling process and after the signaling process. After the signaling process, considering that the *signaling vector* took the values presented in Table 1, the resulting values for *max. velocity* and *min. velocity* are shown. Analyzing these values, V₂ and V₄ were identified to change their velocity limits in the next movements. For V₂, the maximum velocity, after signaling, changes from 10 to 60; for V₄, the minimum velocity changes from 0 to -200, according to the *boost speed vector* and to the *signaling vector*, respectively. For instance, V₂ was boosted by 50 (*boost speed vector*) from its initial velocity of 10 (*max. velocity* before signaling) resulting in a new velocity of 60 (*max. velocity* after signaling).

Section 3C explains in more detail the SiPSO signaling process for the DER scheduling problem addressed in this paper.

3. Energy resource scheduling

In this paper it is proposed the use of the Signaled Particle Swarm Optimization (SiPSO) method, presented in Section 2, to efficiently obtain the solution for a smart grid short-term energy resource scheduling. For comparison purposes, the same problem is solved using Mixed Integer Non-Linear Programming (MINLP) in GAMS [22], Genetic Algorithms (GA), traditional Particle Swarm Optimization (PSO), New Particle Swarm Optimization (NPSO) and Evolutionary Particle Swarm Optimization (EPSO). The obtained results are compared so that the validity of the proposed approach is checked and the advantages and drawbacks of each approach can be discussed.

Energy resource management aims to minimize the operation costs and considers the available resources: generation, storage, and demand response. The problem formulation considers the equipment technical characteristics, their operation costs, and the envisaged demand response actions.

$$\sum_{W=1}^{N_W} P_{Wind}(W,t) + \sum_{Pv=1}^{N_{Pv}} P_{Photovoltaic}(Pv,t) + \sum_{Fc=1}^{N_{Fc}} P_{FuelCell}(Fc,t) + \sum_{S=1}^{N_S} P_{StorageDischarge}(S,t) + \sum_{LC=1}^{N_{LC}} P_{LoadCurtailment}(LC,t) = \sum_{L=1}^{N_L} P_{Load}(L,t) + \sum_{S=1}^{N_S} P_{StorageCharge}(S,t); \forall t \in \{1, \dots, T\} \quad (7)$$

3.1. Mathematical formulation

This sub-section presents the mathematical formulation used for the Mixed-Integer Non-Linear Programming (MINLP) approach considering 1 h periods, corresponding to the hourly operation planning modeling.

The objective function (6) of the mixed-integer non-linear model is formulated with the aim of finding the minimum cost for each period ($t \in$ usually 1 h [32]). Eqns. (7)-(17) refer to the constraints that are considered.

$$\text{Minimize } f = \left[\begin{aligned} & \sum_{W=1}^{N_W} P_{Wind}(W,t) \times C_{Wind}(W,t) \\ & + \sum_{Pv=1}^{N_{Pv}} P_{Photovoltaic}(Pv,t) \times C_{Photovoltaic}(Pv,t) \\ & + \sum_{Fc=1}^{N_{Fc}} P_{FuelCell}(Fc,t) \times C_{FuelCell}(Fc,t) \\ & - \sum_{S=1}^{N_S} P_{StorageCharge}(S,t) \times C_{StorageCharge}(S,t) \\ & + \sum_{S=1}^{N_S} P_{StorageDischarge}(S,t) \times C_{StorageDischarge}(S,t) \\ & + \sum_{LC=1}^{N_{LC}} P_{LoadCurtailment}(LC,t) \times C_{LoadCurtailment}(LC,t) \end{aligned} \right]; \forall t \in \{1, \dots, T\} \quad (6)$$

where:

- $C_{FuelCell}(Fc,t)$ Generation cost of fuel cell unit Fc in period t
 - $C_{LoadCurtailment}(LC,t)$ Energy cost of load curtailment LC in period t
 - $C_{Photovoltaic}(Pv,t)$ Generation cost of photovoltaic unit Pv in period t
 - $C_{StorageCharge}(S,t)$ Charging cost of storage unit S in period t
 - $C_{StorageDischarge}(S,t)$ Discharging cost of storage unit S in period t
 - $C_{Wind}(W,t)$ Generation cost of wind unit W in period t
 - N_{Fc} Number of fuel cells
 - N_{LC} Number of curtailable loads
 - N_{Pv} Number of photovoltaic panels
 - N_S Number of storage units
 - N_W Number of wind turbines
 - $P_{FuelCell}(Fc,t)$ Active power generation of fuel cell unit Fc in period t
 - $P_{LoadCurtailment}(LC,t)$ Load curtailment of load LC in period t
 - $P_{Photovoltaic}(Pv,t)$ Active power generation of photovoltaic unit Pv in period t
 - $P_{StorageCharge}(S,t)$ Active power charge of storage unit S in period t
 - $P_{StorageDischarge}(S,t)$ Active power discharge of storage unit S in period t
 - $P_{Wind}(W,t)$ Active power generation of wind unit W in period t
 - t Period
 - T Simulation time horizon
- Subject to the following constraints:

- Power Balance in each period t

where:

- $P_{Load}(L,t)$ Active power demand of load L in period t
- N_L Number of loads

- Wind generation limits in each period t

$$P_{Wind}(W,t) \leq P_{WindLimit}(W,t); t \in \{1, \dots, T\}; W \in \{1, \dots, N_W\} \quad (8)$$

where:

$P_{WindLimit}(W,t)$ Maximum active power generation of wind unit W in period t

- Photovoltaic generation limits in each period t

$$P_{Photovoltaic}(Pv,t) \leq P_{PhotovoltaicLimit}(Pv,t); t \in \{1, \dots, T\}; Pv \in \{1, \dots, N_{PV}\} \quad (9)$$

where:

$P_{PhotovoltaicLimit}(Pv,t)$ Maximum active power generation of photovoltaic unit Pv in period t

- Fuel cell limits in each period t

$$P_{FuelCell}(Fc,t) \leq P_{FuelCellLimit}(Fc,t); t \in \{1, \dots, T\}; Fc \in \{1, \dots, N_{FC}\} \quad (10)$$

where:

$P_{FuelCellLimit}(Fc,t)$ Maximum active power generation of fuel cell unit Fc in period t

- Storage units limits in each period t

$$0 \leq E_{Storage}(S,t) \leq E_{StorageLimit}(S,t); t \in \{1, \dots, T\}; S \in \{1, \dots, N_S\} \quad (11)$$

where:

$E_{Storage}(S,t)$ Active energy stored in unit S in period t

$X_{(S,t)}$ Maximum active energy stored in unit S in period t

- Storage units discharge limits in each period t

$$P_{StorageDischarge}(S,t) \leq P_{StorageDischargeLimit}(S,t) \times X_{(S,t)}; t \in \{1, \dots, T\}; S \in \{1, \dots, N_S\}; X \in \{0, 1\} \quad (12)$$

where:

$P_{StorageDischargeLimit}(S,t)$ Maximum power discharge of stored in unit S in period t

$X_{(S,t)}$ Binary variable for storage discharge

- Storage units maximal charge limits in each period t

$$P_{StorageCharge}(S,t) \leq P_{StorageChargeLimit}(S,t) \times Y_{(S,t)}; t \in \{1, \dots, T\}; S \in \{1, \dots, N_S\}; Y \in \{0, 1\} \quad (13)$$

where:

$P_{StorageChargeLimit}(S,t)$ Maximum power charge of stored in unit S in period t

$Y_{(S,t)}$ Binary variable for storage charge

- A storage unit cannot charge and discharge at the same time in each period t

$$X_{(S,t)} + Y_{(S,t)} \leq 1; t \in \{1, \dots, T\}; S \in \{1, \dots, N_S\}; X \text{ and } Y \in \{0, 1\} \quad (14)$$

- Storage units maximal discharge limits in each period t considering the storage state

$$P_{StorageDischarge}(S,t) \times \Delta t - E_{Storage}(S,t-1) \leq 0; t \in \{1, \dots, T\}; S \in \{1, \dots, N_S\}; \Delta t = 1 \quad (15)$$

- Storage unit maximal charge limits in each period t considering the storage state

$$P_{StorageCharge}(S,t) \times \Delta t + E_{Storage}(S,t-1) \leq E_{StorageLimit}(S,t); t \in \{1, \dots, T\}; S \in \{1, \dots, N_S\}; \Delta t = 1 \quad (16)$$

- Storage units balance

$$E_{Storage}(S,t) = E_{StorageBattery}(S,t-1) - P_{StorageDischarge}(S,t) \times \Delta t + P_{StorageCharge}(S,t) \times \Delta t; t \in \{1, \dots, T\}; S \in \{1, \dots, N_S\}; \Delta t = 1 \quad (17)$$

The initial state of the considered storage units ($E_{Storage}(S,t=0)$) is known as a result of the previous hour operation data.

The mixed-integer non-linear programming approach to the envisaged problem has been implemented in GAMS [22] – a professional optimization tool.

3.2. Genetic algorithm approach

Genetic Algorithms (GA) are inspired on genetic biological processes, with the goal of finding the best solution of combinatorial problems. In fact, this type of algorithm can only guarantee a local optimal solution but has advantages as it requires less computational resources than traditional approaches [33].

In order to address the problem considered in this paper, the genes of each GA individual can be of five different types, corresponding to: wind generation, photovoltaic generation, fuel cell generation, storage units' charge/discharge, and load curtailment. Each individual has 334 genes, corresponding to the 99 DG units, 27 storage units and 208 controlled loads.

The initial population in GA is randomly generated except for wind turbines that are initialized with the maximum output they can supply for each period.

After setting the initial population, the simulation is performed to reach the final configuration. GA will automatically select the best chromosome at every generation. Thus, at the end of generation the chromosome with the lowest cost is obtained.

After a problem sensitivity analysis, the GA parameters used to solve the envisaged short-term scheduling problem are the following:

- Size of population: 35
- Number of generations: 100
- Fitness scaling: Proportional
- Probability of crossover: 0.93
- Crossover function: Two point
- Mutation function: Uniform
- Elitism: 2 chromosomes.

3.3. SiPSO approach

The methodology proposed in Section 2 is used to solve the resource scheduling problem.

The initial particles population in SiPSO is equal to GA, they are randomly generated except for wind turbines that are initialized with the maximum output they can supply for each period. Wind generation has the lower generation costs of the available resources

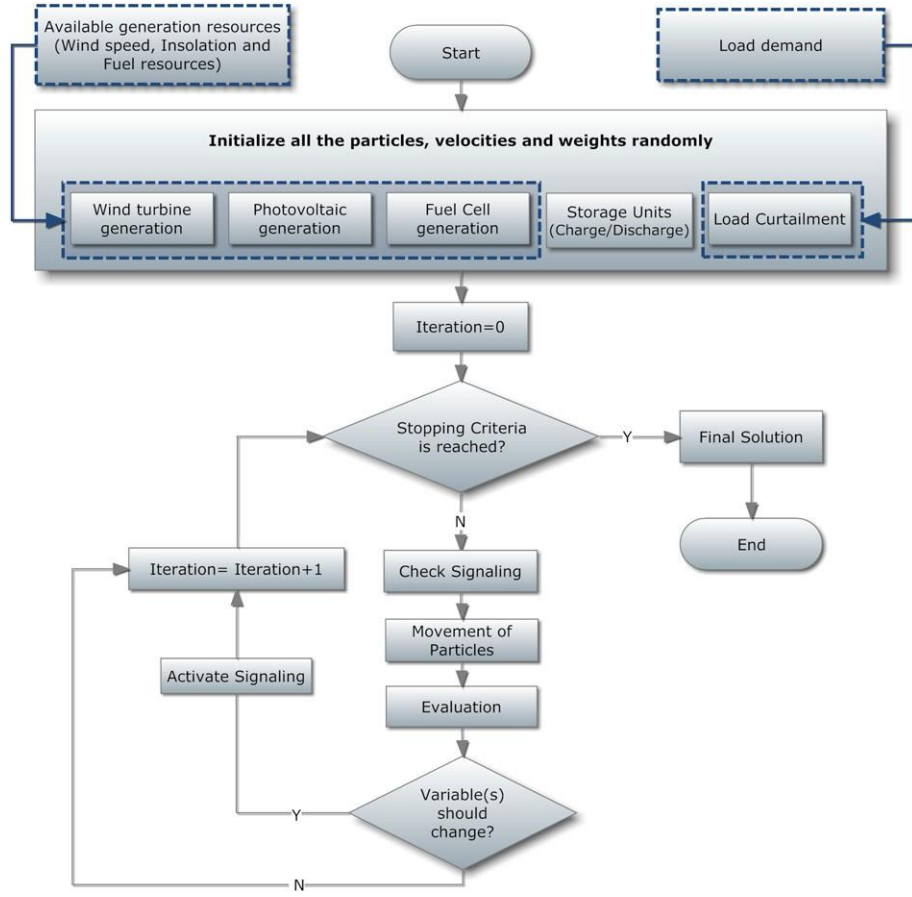


Fig. 2. SiPSO flowchart.

in the present problem and is not dispatchable, so wind based generation (when available) should have priority. This method revealed the achievement of better solutions than a completely random initialization. However, this does not mean that SiPSO does not reduce wind generation when necessary. The control variables of the problem are the generator production, storage charge and discharge and load curtailment values. The dispatch optimization is done for the next hour as previously described.

As SiPSO is a stochastic method, it is possible that the random values assumed for the generation variables exceed the load for a given period. This can be solved by direct or indirect repair using penalizations in the fitness function. The authors opted for the direct repair, lowering generation or refusing load curtails in order to match system restrictions (in this case, the system energy balance). This is done according to a priority cost list for which the higher production costs are put in first place. A deficit in generation can also happen. In that case SiPSO will increase the unit's generation or put into service other generators. This efficient process enables a very good performance as there are only a few checks to make in order to change the generation values.

The present problem variables picked for SiPSO signaling process are the storage units' decision variables. Other variables like generation and load curtailment decision variables could also be signaled. The storage units' usage needs to satisfy storage units' constraints, storage balance, discharging and charging limits. Even if discharging or charging storage units is more profitable than using the other generation resources available, it is not recommended to change these values in the evaluation phase. Instead, the signaling mechanism is used for storage units charging and discharging variables. This is justified because it would be necessary to

check storage units' constraints more than once, every time a storage unit variable is changed, resulting in a worse computational performance.

The proposed SiPSO signaling process allows solving this problem faster. For instance, if a given storage unit discharging is activated, in the next movement iteration SiPSO will change the limits of the velocities for that variable, according to (4) or (5) in Section 2.

The parameters used to address the envisaged scheduling problem are the following:

- Number of Particles: 10
- Number of max iterations: 50
- Initial Max velocity: 400
- Initial Min velocity: -400
- Boost Speed Vector: 1000 for every particle
- Mutation of inertia weight: Gaussian Mutation
- S mutation factor: 0.8.

Fig. 2 presents the SiPSO flowchart.

Table 2
Case study energy resource data.

Energy resources		Number of units	Maximum/minimum price	
			Scenario 1	Scenario 2
Fuel cell		34	0.80e1.00	0.80e1.00
Photovoltaic		31	0.30e0.50	0.45e0.75
Wind		34	0.30e0.50	0.45e0.75
Storage	Charging	27	0.30e0.50	0.54e0.90
	Discharging		0.50e0.70	0.50e0.70
Load		208	1.00e2.00	1.00e2.00

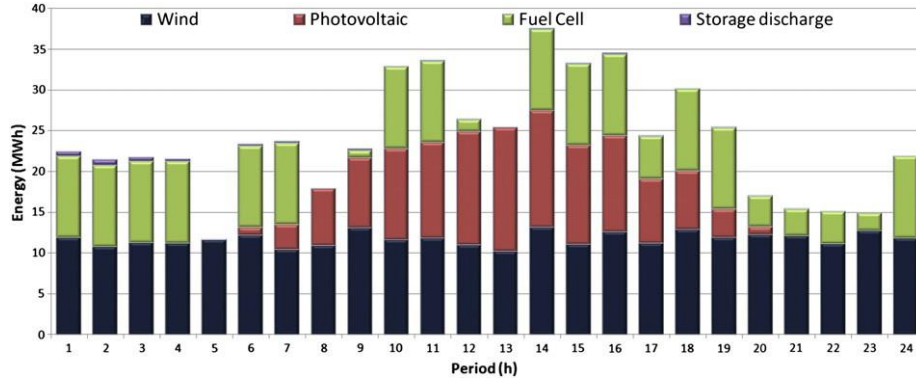


Fig. 3. Resource scheduling results for SiPSO e Scenario 1.

34. PSO, EPSO and NPSO approaches

The classic PSO approach used for comparison purposes is based on [27,34]. Also EPSO and NPSO were implemented according to the respective authors' references [23,24]. The algorithms were developed on the same platform and with the same basic PSO code structure. All variants of PSO use the order of merit direct repair method to achieve power balance. The parameters used in the case study for the above methodologies are the same that are used for SiPSO, except for mutation and boost speed parameters which are not used in the classic PSO approach or in NPSO approach. NPSO acceleration coefficients were 4 for best position and 1 for worst position. The cooperation coefficient was set to 2. The parameters in metaheuristics on this paper were set using empirical studies.

4. Case study

The case study presented in this section considers 99 distributed generators, 208 loads, and 27 storage units. The optimal energy resources scheduling is determined for each hour in a 24 h period for 2 different scenarios. The capacity of the storage units for the next hour depends on the current state (result of the optimization for the previous hour), that will influence the results in the following hour. The data used for the 208 loads (consumers) are real data of Portuguese medium voltage consumers in the north region of the country, which have been metered by *EDP Distribuição*. The data used for the distributed generation resources are based on distributed generation penetration forecasts and on real equipment and primary source data. The complete resource database used in this case study can be obtained from the following

link: http://www.gecad.isep.ipp.pt/papers/Database/DB_ERM_2011.htm.

Scenario 2 differs from Scenario 1 in the photovoltaic, wind and storage resources in terms of price and power. In Scenario 2, the wind and photovoltaic price and power were increased by 50%. The storage charging price and power were increased by 80% and 300% respectively. Table 2 summarizes the energy resources costs considered for each scenario.

The presented case study considers the hourly Distributed Energy Resources (DER) scheduling and results are shown for 24 consecutive periods of 1 h, i.e. for a complete day. It is important to note that all 24 optimizations are independent from each other. DER scheduling for period t is undertaken in period t , considering the operation state resulting from the schedule already used for the previous periods.

The SiPSO results are compared with the results obtained with the deterministic approach, i.e. Mixed-Integer Non-Linear Programming (MINLP), and with the used metaheuristic approaches: Genetic Algorithms (GA), traditional Particle Swarm Optimization (PSO), New Particle Swarm Optimization (NPSO) and Evolutionary Particle Swarm Optimization (EPSO).

All methodologies used to solve the scenarios of this case study have been tested on a PC compatible with one Intel Xeon W3520 2.66 GHz processor, with 4 Cores, 3 GB of random-access-memory (RAM) and Windows Seven Operating System.

4.1. Results for Scenario 1

Fig. 3 shows the results of energy resource scheduling obtained with the SiPSO methodology. SiPSO allocates storage discharge

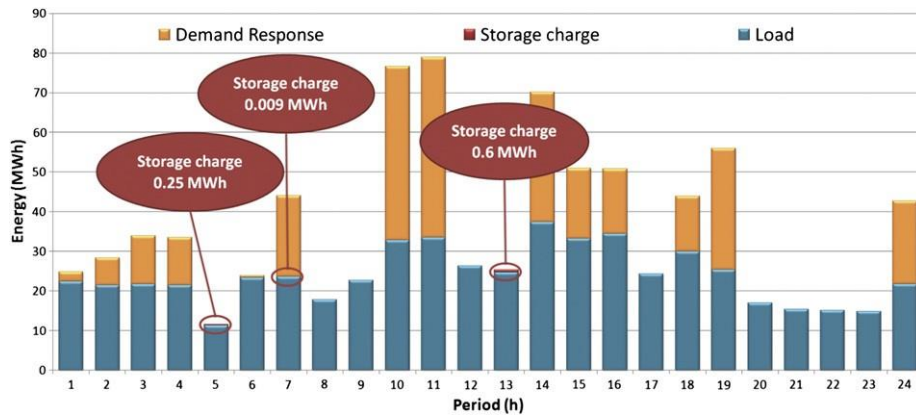


Fig. 4. Load and storage charge for SiPSO e Scenario 1.

Table 3
Robustness test e cost and time comparison over 1000 trials.

Methodologies	Best		Worst		Mean		Mean E. time	
	(m.u.)		(m.u.)		(m.u.)		(s)	
	Scenario 1	Scenario 2	Scenario 1	Scenario 2	Scenario 1	Scenario 2	Scenario 1	Scenario 2
MINLP	660,183	601,952	e	e	e	e	120.34	134.90
GA	661,948	623,111	665,836	630,537	663,615	626,339	7.17	6.74
PSO	667,070	626,040	684,540	656,327	674,910	639,447	3.07	3.18
EPSO	660,834	613,342	667,460	627,840	663,480	618,190	6.70	6.85
NPSO	661,591	617,654	663,870	632,102	661,990	622,916	3.10	3.22
SiPSO	659,802	604,810	667,430	618,234	660,810	607,938	3.43	3.45

Table 4
PSO approaches error analysis e over 1000 trials.

Methodologies	Absolute error				Standard deviation	
	Scenario 1		Scenario 2		Scenario 1	Scenario 2
	(m.u.)	(%)	(m.u.)	(%)	(m.u.)	(m.u.)
PSO	6887e24,357	1.04e3.69	24,088e54,375	4.00e9.03	5117	8733
EPSO	651e7277	0.09e1.10	11,390e25,888	1.89e4.30	1924	4342
NPSO	1408e3687	0.21e0.56	15,702e30,150	2.61e5.01	644	4259
SiPSO	0e7247	0.00e1.10	2858e16,282	0.48e2.71	2214	3928

between hour 1 and hour 4. In these periods SiPSO uses storage units' discharge because the discharge price is more advantageous than the other resources prices. The photovoltaic generation is more used in the periods with more solar radiation, with the photovoltaic generation achieving its maximum value at hours 12 and 13. The wind generation is dispatched with more intensity than the other resources, because it presents a lower price, being more competitive to the system operator than the other resources.

Fig. 4 presents the load diagram, total storage charge and load curtailment in each hour. SiPSO reduces the load in the peak hours, where the load diagram achieves the maximum consumption and the combination of all energy resources is not sufficient to supply the load. The storage charge only happened in 3 h and it had not a significant impact on the total load diagram (combination of the load supplied and the storage charge).

Table 3 presents the best, worst and mean objective function values and the computational execution time for Scenario 1 and 2 for the six used methodologies.

The total operation cost for the considered day is the sum of the 24 objective functions, determined for each period. The SiPSO approach achieved the lowest total cost and the MINLP achieved the second lowest total cost. GA, PSO, NPSO and EPSO approaches achieved worse solutions than MINLP and SiPSO.

As expected, the metaheuristic approaches present lower computational execution times than MINLP approach, confirming

their advantage to address large dimension problems that require a decision in a short period of time.

The results show that the SiPSO method presents the third fastest computational execution time (after PSO and NPSO) and results in a solution with the lowest cost. The SiPSO reached the best solution in a competitive computational execution time showing the effectiveness of the proposed methodology. SiPSO presents a slightly higher time execution than PSO due to the undertaken mutation calculations and signaling process. In order to obtain better PSO solutions the used parameters could be adjusted, for instance by increasing the number of iterations/particles. However, in that case PSO would take more time than SiPSO.

The key to successfully reach good results with SiPSO was to include a heuristic method (changing certain variables' velocity limits in a specific direction) that makes this method more suitable for the short-term energy scheduling problem. The main reason to achieve good results with SiPSO was the use of the proposed signaling process strategically applied to the relevant set of variables. In fact the best SiPSO's solution in 1000 trials is better than the reference technique MINLP.

The MINLP execution time is 120 s, i.e about 35 times slower than SiPSO.

The smart grid model can aggregate several consumers with different DG units. This means that even on a small network a high number of DG units are expectable and as a consequence many

Table 5
Energy resource resultsfor Scenario 1.

		MINLP	GA	PSO	EPSO	NPSO	SiPSO
Wind	Energy (kWh)	278,795	279,453	279,130	278,543	279,091	279,382
	Cost (m.u.)	112,179	112,502	112,344	112,056	112,324	112,470
PV	Energy (kWh)	129,356	128,109	127,056	129,356	128,810	129,356
	Cost (m.u.)	51,856	51,243	50,766	51,856	51,588	51,856
Fuel cell	Energy (kWh)	161,178	160,369	160,912	160,790	160,678	160,794
	Cost (m.u.)	142,257	141,604	142,237	141,937	141,832	141,935
Storage charging	Energy (kWh)	250	0	0	0	0	859
	Cost (m.u.)	125	0	0	0	0	426
Storage discharging	Energy (kWh)	2750	2500	544	2388	750	3359
	Cost (m.u.)	1690	1532	329	1456	459	2067
Load curtailment	Energy (kWh)	274,582	277,733	280,488	277,055	278,803	276,100
	Cost (m.u.)	352,326	355,067	361,395	353,529	355,388	351,900
Total	Cost (m.u.)	660,183	661,948	667,071	660,834	661,591	659,802

Table 6
Convergence test over 1000 trials.

Methodologies	Mean
	Iteration/generation
GA	42.36
PSO	43.21
EPSO	37.20
NPSO	39.57
SiPSO	34.33

decision variables on the optimization problem. Deterministic approaches do not work well when the problem variables increase and execution time is exponential, whereas metaheuristics approaches do not suffer much from this problem.

Looking at the obtained execution time values, the SiPSO advantage may seem irrelevant for the considered hourly scheduling. However, in practice, it is determinant to obtain an efficient resource scheduling because the adopted scheduling should be based on studies considering a set of scenarios and not a single scenario. In fact, it can be desirable to fuzzy some data (e.g. the load or wind production) and consider some incidents and/or repairing actions, what requires running several optimizations for the same period.

Table 4 presents the results of the error analysis resulting from the robustness test for each PSO approach and for both scenarios as well.

It can be seen that the proposed SiPSO technique performs better among the other PSO approaches in terms of absolute error. This value represents the difference between the objective function when compared to the reference MINLP technique. SiPSO error range is between 0 and 7247 over the 1000 trials in Scenario 1. The minimum range in this scenario is considered by the authors with a value of 0, and thus 0% error, because the objective function of SiPSO approach is better than MINLP technique. Therefore, when the objective function value is better than the MINLP reference technique the error is considered to be zero. The standard deviation in this table gives the indication of the variability of the objective function cost over the 1000 trials for each approach and for both scenarios. SiPSO standard variation in Scenario 1 is the 3rd best among the 4 approaches and presents the best standard deviation in Scenario 2. It should be noted that although a high standard deviation indicates a high variability over random runs, the proposed SiPSO technique clearly presents the best mean objective function values over the 1000 trials for both scenarios (see Table 3).

Table 5 presents the scheduled energy and the operation cost for each resource for Scenario 1.

Comparing the MINLP and SiPSO costs, in the first hour MINLP achieved lower cost due to a better use of storage charge and discharge, which influenced the initial state for the following hour. This happens because MINLP discharges all the storage resources in hour 1 whereas SiPSO keeps some energy stored in the storage units, which can be used later. As the schedule adopted for a specific hour determines the state of the following hours, although MINLP obtained a better solution than SiPSO in hour 1 the overall 24 period cost achieved by SiPSO is lower.

SiPSO's best management of storage units avoided the need to make a more intensive use of highly expensive resources (namely load curtailment and fuel cells), allowing to achieve lower total costs.

Table 6 depicts the results of the convergence tests for this scenario. The mean value corresponds to the last iteration for which the fitness solution improves. The mean value corresponds to 1000 trials with 24 optimizations each. SiPSO obtained the smaller value of all the considered PSO variants, confirming its faster convergence speed.

4.2 Results for Scenario 2

This section presents the results for Scenario 2 where capacity storage was increased and more generation is available. The prices are also different from Scenario 1 as can be confirmed in Table 2.

Fig. 5 presents the results of the energy resource scheduling obtained with the proposed methodology. In this scenario, SiPSO allocates the storage discharge for every hour with the exception of period 5. Similarly to scenario 1, the photovoltaic generation achieves its maximum value at hours 12 and 13 and the wind generation is dispatched with more intensity than the other resources due to its lower price.

Fig. 6 shows the load diagram, total storage charge and load curtailment in each period. Storage charge is more intensively used than in Scenario 1, being scheduled for eleven periods. This has a significant impact in the total load diagram because storage discharge in the upcoming periods is managed according to the available storage units' charge which is determined by the charging scheduled in previous periods.

The total operation cost for the considered day is the sum of the 24 objective functions, determined for each period. The MINLP achieved the lowest total cost and SiPSO achieved the second lowest total cost. SiPSO remains behind with less than 1% difference in the total cost whereas all the other PSO variants result in solutions with higher costs, as already happened in Scenario 1.

As in Scenario 1, the metaheuristic approaches present lower computational execution time than the MINLP approach,

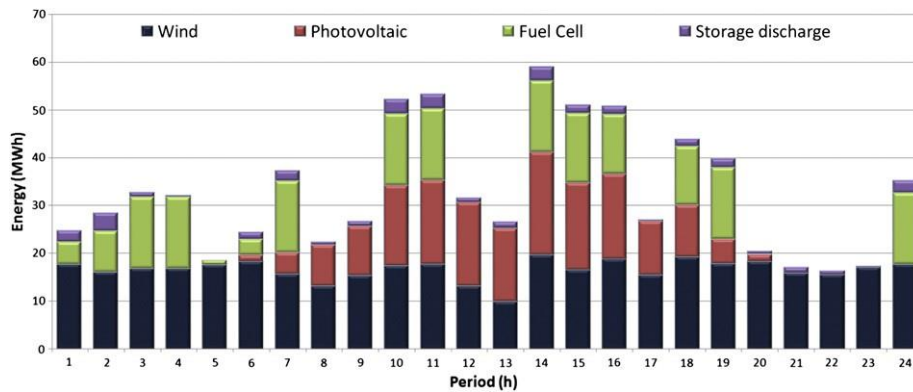


Fig. 5. Resource scheduling results for SiPSO e Scenario 2.

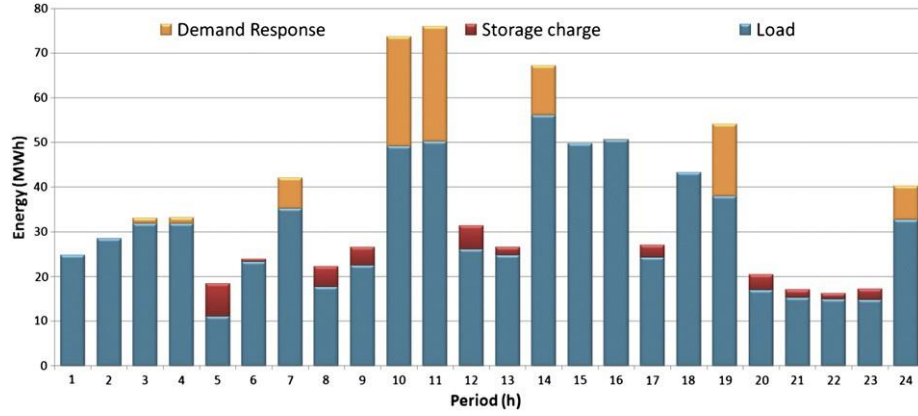


Fig. 6. Load and storage charge for SiPSO e Scenario 2.

Table 7
Energy resource results for the Scenario 2.

		MINLP	GA	PSO	EPSO	NPSO	SiPSO
Wind	Energy (kWh)	398,921	378,342	395,285	397,498	390,554	396,686
	Cost (m.u.)	238,643	223,928	235,684	237,481	232,432	236,862
PV	Energy (kWh)	179,359	169,578	165,505	171,557	170,590	178,575
	Cost (m.u.)	106,301	99,475	97,801	101,178	100,328	105,696
Fuel cell	Energy (kWh)	173,441	178,733	166,440	168,732	176,682	177,159
	Cost (m.u.)	152,795	158,128	147,136	149,274	156,794	156,328
Storage charging	Energy (kWh)	43,040	0	14,150	25,077	17,504	35,490
	Cost (m.u.)	35,664	0	2834	21,142	14,993	29,089
Storage discharging	Energy (kWh)	43,790	7500	17,536	26,670	20,923	36,794
	Cost (m.u.)	27,059	4597	10,774	16,625	13,051	22,706
Load curtailment	Energy (kWh)	95,660	114,057	117,516	108,751	106,887	94,407
	Cost (m.u.)	112,818	136,984	146,479	129,926	130,043	112,306
Total	Cost (m.u.)	601,952	623,111	626,040	613,342	617,654	604,810

confirming their advantage in this aspect. In what concerns execution time, the situation is similar to the one already commented for Scenario 1. Also, in Scenario 2 SiPSO is about 35 times faster than MINLP.

Table 7 presents the scheduled energy and the operation cost for each resource for Scenario 2.

Comparing the MINLP and SiPSO costs, one can see that the SiPSO approach uses less generation resources except for the use of fuel cells. As in Scenario 1, SiPSO uses less load curtailment.

Comparing SiPSO and others metaheuristics schedules, the proposed methodology uses a better management of storage units avoiding the need to make a more intensive use of highly expensive resources (namely load curtailment and fuel cells) thus achieving lower total costs.

5. Conclusion

The present paper proposes a modified Particle Swarm Optimization (PSO) approach e Signaled Particle Swarm Optimization (SiPSO) e which is based on the change of velocity limits during the search process.

The advantages, the adequacy and practical interest of the proposed method is illustrated by using it to solve the short-term energy resource scheduling problem in the smart grid context. SiPSO's performance is compared with other five alternative methods, namely Mixed Integer Non-Linear Programming (MINLP), Genetic Algorithms (GA), PSO, Evolutionary Particle Swarm Optimization (EPSO), and New Particle Swarm Optimization (NPSO).

SiPSO presents very competitive execution time (about 35 times faster than the deterministic approach for the presented case

studies) and only a slightly higher execution time than PSO and NPSO. Moreover, SiPSO is able to achieve very good low cost solutions, which compete directly with the ones obtained by the deterministic approach. This makes SiPSO suitable to address complex high dimension problems, even when relatively low performance computational resources are used.

Power systems are increasingly making intensive use of Distributed Energy Resources (DER), partially driven by renewable based generation technologies, which has significantly increased the dimension of short-term resource scheduling problem. This makes the use of the proposed methodology very relevant to achieve a satisfactory operation cost in a competitive time. The proposed method enables a significant number of scenarios to be considered within the available time frame for short-term resource scheduling. In this way, a more efficient use of the available renewable based generation and other distributed energy resources is provided to smart grid players.

The good results obtained with the proposed method proved that it is adequate to handle realistic problems involving a large number of variables and requiring fast execution time. Its use to address short-term DER scheduling has been proved. Furthermore, its use to solve other complex problems with similar characteristics in other fields, such as production scheduling, transportation scheduling and rescheduling, seems very promising.

Acknowledgments

The authors would like to acknowledge EDP Distribuição, SA (the Portuguese Electricity Distribution Company), FEDER Funds through COMPETE program and by National Funds through FCT

under the projects FCOMP-01-0124-FEDER: PEst-OE/EEI/UI0760/2011, PTDC/EEA-EEL/099832/2008, and PTDC/SEN-ENR/099844/2008.

References

- [1] Garrity TF. Getting smart. *IEEE Power Energy M* 2008;6(2):38e45.
- [2] Venayagamoorthy GK. Potentials and promises of computational intelligence for smart grids. *IEEE Pow Ener Soc Ge*;2009;2142e7.
- [3] Blumsack S, Fernandez A. Ready or not, here comes the smart grid! *Energy*. (0).
- [4] Public Law 110-140. Energy Independence and Security Act of 2007; 2007.
- [5] (NETL) USDoEDatNETL. Systems view of the modern grid; 2007.
- [6] Committee UDoE EA. Smart grid system report; July 2009.
- [7] Zhu J. Optimization of power system operation. Wiley-IEEE Press; 2009.
- [8] Huneault M, Galiana FD. A survey of the optimal power flow literature. *IEEE Transactions on Power Systems* 1991;6(2):762e70.
- [9] Vlachogiannis JG, Lee KY. Economic load dispatch - a comparative study on heuristic optimization techniques with an improved coordinated aggregation-based PSO. *IEEE Transactions on Power Systems* 2009;24(2):991e1001.
- [10] Zhang ZS. Quantum-behaved particle swarm optimization algorithm for economic load dispatch of power system. *Expert Systems with Applications* 2010;37(2):1800e3.
- [11] Vahidinasab V, Jadid S. Joint economic and emission dispatch in energy markets: a multiobjective mathematical programming approach. *Energy* 2010;35(3):1497e504.
- [12] Zhang YN, Kang LY, Cao BG, Huang CN, Wu GH. Renewable energy distributed power system with wind power and Biogas generator. In: T& D Asia: 2009 Transmission & distribution conference & exposition: Asia and Pacific; 2009. p. 74e9.
- [13] Palma-Behnke R, Cerda JL, Vargas LS, Jofre A. A distribution company energy acquisition market model with integration of distributed generation and load curtailment options. *IEEE Transactions on Power Systems* 2005;20(4):1718e27.
- [14] Faria P, Vale Z, Soares J, Ferreira J. Demand response management in power systems using a particle swarm optimization approach. *IEEE Intelligent Systems*, 2011, doi:10.1109/MIS.2011.35.
- [15] Faria P, Vale Z. Demand response in electrical energy supply: an optimal real time pricing approach. *Energy* 2011;36(8):5374e84.
- [16] AlRashidi MR, El-Hawary ME. A survey of particle swarm optimization applications in electric power systems. *IEEE Trans Evolut Comput* 2009;13(4):913e8.
- [17] Bonabeau E, Dorigo M, Theraulaz G. Swarm intelligence: from natural to artificial systems. Oxford University Press; 1999.
- [18] del Valle Y, Venayagamoorthy GK, Mohagheghi S, Hernandez JC, Harley RG. Particle swarm optimization: basic concepts, variants and applications in power systems. *IEEE Trans Evolut Comput* 2008;12(2):171e95.
- [19] Yuan X, Su A, Yuan Y, Nie H, Wang L. An improved PSO for dynamic load dispatch of generators with valve-point effects. *Energy* 2009;34(1):67e74.
- [20] Moghaddam AA, Seifi A, Niknam T, Alizadeh Pahlavani MR. Multi-objective operation management of a renewable MG (micro-grid) with back-up micro-turbine/fuel cell/battery hybrid power source. *Energy*. (0).
- [21] Kennedy J, Eberhart R. Particle swarm optimization. *Conference particle swarm optimization*, vol. 4. p. 1942e1948 vol. 4.
- [22] GAMS. GAMS - the solver manuals; 2001. Washington: DC 2007.
- [23] Miranda V, Fonseca N. EPSO - Evolutionary Particle Swarm Optimization, a new algorithm with applications in power systems. *Conference Proceedings*. In: IEEE/PES Transmission and distribution Conference and Exhibition 2002: asia Pacific, vols. 1e3; 2002. p. 745e50.
- [24] Selvakumar AI, Thanushkodi K. A new particle swarm optimization solution to nonconvex economic dispatch problems. *IEEE T Power Syst* 2007;22(1):42e51.
- [25] Chapman SJ. MATLAB programming for engineers. In: Toronto, Ont.: Thomson, editor. 4th international student ed. London: Thomson Learning [distributor]; 2008.
- [26] Lee KY, El-Sharkawi MA. Modern heuristic optimization techniques: theory and applications to power systems. Piscataway, N.J: IEEE Press; Hoboken, N.J.:Wiley-Interscience; 2008.
- [27] Kennedy J, Eberhart R. Particle swarm optimization. In: 1995 IEEE international conference on neural networks proceedings, vol. 1-6; 1995. p. 1942e8.
- [28] Lee TY. Optimal spinning reserve for a wind-thermal power system using EIPSO. *IEEE T Power Syst* 2007;22(4):1612e21.
- [29] Agrawal S, Panigrahi BK, Tiwari MK. Multiobjective particle swarm algorithm with fuzzy clustering for electrical power dispatch. *IEEE T Evolut Comput* 2008;12(5):529e41.
- [30] de Oca MAM, Stutzle T, Birattari M, Dorigo M, Frankenstein's PSO. A composite particle swarm optimization algorithm. *IEEE T Evolut Comput* 2009;13(5):1120e32.
- [31] Chen WN, Zhang J, Chung HSH, Zhong WL, Wu WG, Shi YH. A novel set-based particle swarm optimization method for discrete optimization problems. *IEEE T Evolut Comput* 2010;14(2):278e300.
- [32] Morais H, Kádár P, Faria P, Vale ZA, Khodr HM. Optimal scheduling of a renewable micro-grid in an isolated load area using mixed-integer linear programming. *Renewable Energy* 2010;35(1):151e6.
- [33] Sivanandam SN, Deepa SN. Introduction to genetic algorithms. Berlin: Springer; 2008.
- [34] Shi YH, Eberhart R. A modified particle swarm optimizer. 1998 IEEE international conference on evolutionary computation - proceedings; 1998. pp. 69e73.