A T-cell algorithm for solving dynamic economic power dispatch problems

Victoria S. Aragón[1], Carlos A. Coello Coello[2], and Mario G. Leguizamón[1]

[1] *Laboratorio de Investigación y Desarrollo en Inteligencia Computacional, Universidad Nacional de San Luis - Ej. de Los Andes 950, San Luis (5700), ARGENTINA*
{vsaragon, legui}@unsl.edu.ar

[2] *CINVESTAV-IPN (Evolutionary Computation Group), Departamento de Computación, Av. IPN No. 2508, Col. San Pedro Zacatenco, México D.F. 07300, MÉXICO*
ccoello@cs.cinvestav.mx

Abstract

This paper presents the artificial immune system IA_DED (*Immune Algorithm Dynamic Economic Dispatch*) to solve the Dynamic Economic Dispatch (DED) problem and the Dynamic Economic Emission Dispatch (DEED) problem. Our approach considers these as dynamic problems whose constraints change over time. IA_DED is inspired on the activation process that T cells suffer in order to find partial solutions. The proposed approach is validated using several DED problems taken from specialized literature and one DEED problem. The latter is addressed by transforming a multi-objective problem into a single-objective problem by using a linear aggregating function that combines the (weighted) values of the objectives into a single scalar value. Our results are compared with respect to those obtained by other approaches taken from the specialized literature. We also provide some statistical analysis in order to determine the sensitivity of the performance of our proposed approach to its parameters. Part of this work was presented at the XXV Argentine Congress of Computer Science (CACIC), 2019.

Keywords: Artificial immune systems, dynamic economic dispatch problem, dynamic economic emission dispatch problem, metaheuristics

Resumen

Este artículo presenta el sistema inmune artificial IA_DED (*Immune Algorithm Dynamic Economic Dispatch*) para resolver el problema de despacho de energía económico dinámico (DED) y el problema de despacho de energía económico dinámico que tiene en cuenta la emisión de gases (DEED). Nuestro enfoque considera estos problemas como problemas dinámicos cuyas restricciones cambian con el tiempo. IA_DED está inspirado en el proceso de activación que sufren las células T del sistema inmune para encontrar soluciones parciales. El enfoque propuesto se valida utilizando varios problemas de DED tomados de literatura especializada y un problema DEED. El último se aborda transformando un problema múlti-objetivo en un problema de un solo objetivo mediante el uso de una función agregativa lineal que combina los valores ponderados de dos objetivos en un solo valor escalar. Nuestros resultados se comparan con respecto a los obtenidos por otros enfoques tomados de la literatura especializada. También proporcionamos un análisis estadístico para determinar la sensibilidad del desempeÃśo de nuestro enfoque a sus parámetros. Parte de este trabajo fue presentado en el XXV Congreso Argentino de Informática (CACIC), 2019.

Palabras claves: Sistemas inmunes artificiales, problema de despacho de energía económico dinámico, problema de despacho de energía económico dinámico con emisión de gases, metaheurísticas

1 Introduction

Electrical energy is generated by transforming some other type of energy (chemical combustion, nuclear fission, kinetic energy of flowing water and wind, solar photo-voltaic and geothermal power, among others) into electrical energy through a procces called electricity generation. This transformation happens at a power station by electromechanical generators. It is the first step of an electrical supply system. Then, electrical energy is transmitted and distributed to consumers by means of specialized systems.

The energy requirements from a city, region or country vary throughout the day. This variation depends on many factors, such as: types of existing industries in the area and shifts performed on their production, weather (extremes of heat or cold), type of appliances that are most frequently used, type of water heaters installed at homes, the season of the year and the time of day at which the energy demands are considered, among others. The generation of electrical energy should respond to the demand curve; that is, if energy demand is increased, power supply must also increase and vice versa.

Since the 1920s, researchers have paid attention to the Static Economic Dispatch (SED) problem [1], i.e., the problem of determining how much energy has to produce each generator from a power system, in order to minimize the production cost and to satisfy some constraints such as: load demand, maximum and minimum limits and prohibited operating zones. However, the SED problem satisfies only one load demand regardless of future load demands or the generators' lifetime.

Also, if the gradients for temperature and pressure inside the boiler and turbine are kept inside safe limits [1] the generators' lifetime can last longer. Consequently, the SED problem was extended with a ramp rate and a constraint which preserves the life of the units or generators [2], limiting the rate of increase or decrease of the power output.

This extension originated the Dynamic Economic Dispatch (DED) problem. In this optimization problem, a sequence of load demands has to be met by minimizing the production cost while some constraints are met. Also, if environmental protection to reduce pollutant and atmospheric emissions caused by thermal power station are considered, the problem becomes a Dynamic Economic Emission Dispatch (DEED) problem where emission and fuel cost need to be minimized.

On the other hand, any time-dependent problem can be considered as a dynamic problem. Such problems can change the objective function, the constraints or both. A change over a constraint exists when the problem conditions change (for instance, how much energy has to produce the system at one point). So, in this paper, the DED and DEED problems are considered as dynamic problems whose load demands constraint change over time in a random fashion.

In a previous work, presented at the *XXV Argentine Congress of Computer Science* (CACIC)[3], an algorithm based on an artificial immune system (AIS) to solve the DED problem was developed and validated. The present study extends this previous work by incorporating to IA_DED the ability to solve DEED problems maintaining the ability to minimize the production cost as well as the time invested to find it. Considering T load demands by day, the problem is regarded as a sequence of T problems. But, each problem (at time i): 1) depends on the solution produced for the previous problem (at time $i-1$) and 2) conditions its successor (at time $i+1$).

The remainder of this paper is organized as follows. Section 2 defines the DED and DEED problems. Section 3 provides a short review of some of the approaches which have been used to solve the DED and DEED problems. Section 4 presents considerations about the DED and DEED problems. In Section 5, we describe our proposed algorithm. In Section 6, we present the test problems used to validate our proposed approach as well as its parameters settings. In Section 7, we present our results and we discuss and compare them with respect to other approaches. Finally, in Section 9, we present our conclusions and some possible paths for future research.

2 Problem Formulation

In the DED problem the main aim is to minimize the total production cost (TC) associated with N dispatch units for a time period:

$$cost(P^t) = \sum_{i=1}^{N} F_i(P_i^t) \tag{1}$$

$$TC = \sum_{t=1}^{T} cost(P^t) \tag{2}$$

where TC is the fuel cost over the whole dispatch period, $cost(P^t)$ is the fuel cost for the t^{th} interval, $P^t = (P_1^t, P_2^t, \ldots, P_N^t)$ is the power output of each unit at time t, T is the number of intervals in the period, N is the number of generators or units in the system, P_i^t is the power of the i^{th} unit at time t (in MW) and F_i is the fuel cost for the i^{th} unit (in \$/h).

The simplest fuel cost function (i.e., smooth) can be expressed as a single quadratic function:

$$F_i(P_i^t) = a_i(P_i^t)^2 + b_i P_i^t + c_i \tag{3}$$

where a_i , b_i and c_i are the fuel consumption cost coefficients of the i^{th} unit. But, if the valve-point effects are taken into account, the fuel cost function becomes non-smooth and the i^{th} unit is expressed as the sum of a quadratic and a sinusoidal function in the form:

$$F_i(P_i^t) = a_i(P_i^t)^2 + b_i P_i^t + c_i + \mid e_i sin(f_i(P_{min_i} - P_i^t)) \mid \tag{4}$$

where e_i and f_i are the fuel cost coefficients of the i^{th} unit with valve-point effects.

Furthermore, in the DEED problem, another objective function is considered in order to reduce the amount of atmospheric pollutants released into the air [4] (i.e., the *Emission*). Emission is defined by:

$$Emission = \sum_{t=1}^{T} emission_t(P^t) \quad (5)$$

$$emission_t(P^t) = \sum_{i=1}^{N} \alpha_i (P_i^t)^2 + \beta_i P_i^t + \gamma_i + \eta_i exp(\delta_i P_i^t) \quad (6)$$

where α_i, β_i, γ_i, η_i and δ are emission coefficients of unit i. According to [5, 4], $Emission$ and TC are combined in order to get just one objective function, so, the DEED problem consists in minimizing:

$$CostEmission = wEmission + (1-w)TC \quad (7)$$

where $w \in [0,\ 1]$ is used to generate a trade-off between the fuel cost and the emission according to the user's preferences.

Regardless of the fuel cost function (Eqs. (3) or (4)), the minimization of TC is subject to:

1. Power Balance Constraint: the power generated has to be equal to the power demand required. It is defined as:

$$\sum_{i=1}^{N} P_i^t - P_D^t - P_L^t = 0 \quad (8)$$

where $t = 1, 2, \ldots, T$. P_D^t is the power demand at time t, and P_L^t is the transmission power loss at time t (in MW). This value considers the transmission loss due to the geographical distribution of the power stations. Although this value can be determined by means of a power flow solution, in this paper, we use Kron's formula which represents the losses as a function of the output level of the system generators and it uses some B-matrix loss coefficients. This is the most popular approach to find an approximate value of the losses. The general form of the loss formula using B-coefficients is:

$$P_L^t = \sum_{i=1}^{N}\sum_{j=1}^{N} P_i^t B_{ij} P_j^t + \sum_{i=1}^{N} B0_i P_i^t + B00 \quad (9)$$

If transmission power loss is not considered, $P_L^t = 0$.

2. Operating Limit Constraints: units have physical limits regarding the minimum and maximum power they can generate:

$$P_{min_i} \leq P_i^t \leq P_{max_i} \quad (10)$$

where P_{min_i} and P_{max_i} are the minimum and maximum power output of the i^{th} unit in MW, respectively.

3. Ramp Rate Limits: they restrict the operating range of all on-line units. Such limits indicate how quickly the unit's output can be changed:

$$\begin{cases} P_j^t - P_j^{(t-1)} \leq UR_j & if \quad P_j^t > P_j^{(t-1)} \\ P_j^{(t-1)} - P_j^t \leq DR_j & if \quad P_j^t < P_j^{(t-1)} \end{cases} \quad (11)$$

where $P_j^{(t-1)}$ is the output power of the j^{th} unit at a previous hour and UR_j and DR_j are the ramp-up and ramp-down limits of the j^{th} unit in MW, respectively. Due to ramp-rate constraints, Eq. (10) is replaced by:

$$max(P_{min_j}^t, P_j^{(t-1)} - DR_j) \leq P_j^t \quad (12)$$

and

$$P_j^t \leq min(P_{max_j}^t, P_j^{(t-1)} + UR_j) \quad (13)$$

such that

$$\begin{cases} P_{min_j}^t = max(P_{min_j}, P_j^{(t-1)} - DR_j) \\ P_{max_j}^t = min(P_{max_j}, P_j^{(t-1)} + UR_j) \end{cases} \quad (14)$$

4. Prohibited Operating Zones: the operation of the units is restricted due to steam valve operation conditions or to vibrations in the shaft bearing. Thus, a unit with prohibited operating zones has a discontinuous input-output power generation characteristic which gives rise to additional constraints on the unit operating range.

$$\begin{cases} P_{min_i} \leq P_i^t \leq PZ_{i,1}^L & or \\ PZ_{i,k-1}^U \leq P_i^t \leq PZ_{i,k}^L, & or \\ PZ_{i,n_1}^U \leq P_i^t \leq P_{max_i} & k = 2, 3, ..., n_i \end{cases} \quad (15)$$

where n_i is the number of prohibited zones of the i^{th} unit, and k is the index of the prohibited operating zones of the i^{th} unit. $PZ_{i,k}^L$ and $PZ_{i,k}^U$ are the lower and upper bounds of the k^{th} prohibited operating zones of unit i.

3 Literature Review

Artificial Intelligence (AI) techniques are appropiate to solve the DED problem because this is a real-world problem with several particular features that make it difficult to solve, since its nonlinear search space is nonsmooth, discontinuous and non-differentiable. In fact, if valve-point effects or prohibited zones are considered, then we are dealing with a nonconvex problem [6].

This section aims to highlight how the DED problem has been tackled using different AI techniques, rather than providing a comprehensive description of each of them. These methods include: neural networks [7, 8, 9], simulated annealing [10], evolutionary algorithms [11, 12, 13, 14, 6, 15], differential evolution [16, 17, 18], particle swarm optimization [19, 20, 21, 14], Harmony Search [5, 22], and Artificial Immune Systems [23]. Additional techniques have been reported in [14, 5, 24, 25, 26, 27, 28, 29, 30, 31]. Some researchers have reported the use of hybrid approaches, such as [32, 33, 34, 35, 5, 36, 22, 37, 38]. Other iterative methods are reported in [39, 40, 41] which minimize T subproblems instead of an NT problem.

4 Our considerations about DED and DEED Problems

For a DED problem with N units and T time intervals, a feasible solution for the whole dispatch period is a power output sequence where Eq. (8) to Eq. (15) must be met. This solution $\in \Re^{N\times T}$ and it has the following form:

$$(P_1^1, P_2^1, \ldots, P_N^1, P_1^2, P_2^2, \ldots, P_N^2, \ldots, P_1^T, P_2^T, \ldots, P_N^T)$$

Let $(P_1^i, P_2^i, \ldots, P_N^i)$ be a partial feasible solution for the ith interval (subproblem i). In this paper, we consider DED and DEED problems as a sequence of T economic dispatch problems, where a relationship is kept between the solutions from consecutive intervals (i^{th} and $(i+1)^{th}$). This relationship is based on the following: 1) after power demand for the i^{th} interval is determined, the units remain with a specified power output and 2) the ramp rate limits. These two facts will determine the operational limits for the $(i+1)^{th}$ interval.

A traditional population approach to solve the DED problem will require to find from the C solutions ($\in \Re^{N\times T}$, with population size C) a feasible one. Let's assume that we have the next population:

$$\begin{cases} (P_{1,1}^1, .., P_{N,1}^1, P_{1,1}^2, .., P_{N,1}^2, .., P_{1,1}^T, .., P_{N,1}^T) \\ \ldots.. \\ (P_{1,i}^1, .., P_{N,i}^1, P_{1,i}^2, .., P_{N,i}^2, .., P_{1,i}^T, .., P_{N,i}^T) \\ \ldots \\ (P_{1,C}^1, .., P_{N,C}^1, P_{1,C}^2, .., P_{N,C}^2, .., P_{1,C}^T, .., P_{N,C}^T) \end{cases}$$

where $P_{h,s}^t$ indicates the power output of the h^{th} unit at time t for the s^{th} population's individual, with $h = 1, \ldots, N$, $t = 1, \ldots, T$ and $s = 1, \ldots, C$. This population has C starting points $(P_{1,s}^1, P_{2,s}^1, \ldots, P_{N,s}^1$ with $s = 1, \ldots, C)$. Each of them will constrain the operational limits for the next intervals, within each individual. Thus, a traditional population approach will carry out C search processes in parallel. Hence, time and computational effort is invested in C different search processes.

In this work, we adopt a divide-and-conquer approach dividing a problem defined in an NT space into T subproblems in an N space. In general, the divide and conquer approach works by breaking down a problem into subproblems; then, each of these subproblems is properly solved. In a further step, the solutions to these subproblems are combined to obtain the solution to the original problem. So, our approach searches partial solutions (i.e., for each interval) by taking a previous partial solution as a starting point. That is, for the $(i+1)^{th}$ interval, the best partial solution from the i^{th} interval determines the operating limits for all solutions of the $(i+1)^{th}$ interval. Moreover, the best partial solution of the $(i+1)^{th}$ interval will be the starting point for the next interval, and so on. Thereby, when a new power demand arrives, all solutions have the same operational limits because they all adopt the same starting point, i.e., the best solution from the previous interval.

Our approach considers the DED problem as a dynamic problem with constraints that change over time. These are the power balance constraint (Eq. (8)) and the rate ramp limits (Eq. (12 and 13)). We search a partial feasible solution for interval 1, then for interval 2, and so on until T intervals had been reached.

Based on the work reported in [5, 4], we transformed the multi-objective problem (DEED) into a single objective problem through the use of a linear aggregating function.

5 Our Proposed Algorithm

Here, an artificial immune system originally designed to solve DED [3] is extended to solve DEED problems. It is based on the activation process that T cells suffer. This process is divided in two parts: proliferation and differentiation [42].

The proposed approach is called IA_DED (Immune Algorithm for Dynamic Economic Dispatch problem). It works on a cells population. Each cell is activated in order to find partial feasible solutions. Special receptors present on the cells surface, called T cell receptors (TCR), are used to represent the decision variables of the problem. In this case, each variable represents a thermal unit, so a TCR has N variables.

5.1 Activation Process

The proliferation process clones N times each cell and the differentiation process changes these clones so that they acquire specialized functional properties. The differentiation process to be applied depends on the feasibility cell.

- **Differentiation for feasible cells:** Based on a probability P_a, each unit exchanges part of its output power with another unit from the same cell. The idea is to take a value (called d) from a unit (say i) and add it to another unit (say j). i^{th} and j^{th} units are modified according to: $cell.TCR_i = cell.TCR_i - d$ and $cell.TCR_j = cell.TCR_j + d$, where $d = U(0, P_c * min(cell.TCR_i - P^t_{min_i}, P^t_{max_j} - cell.TCR_j))$, $U(w_1,w_2)$ refers to a random number with a uniform distribution in the range (w_1,w_2) and P_c is a change factor ($P_c \in [0,1]$). The best from among the clones and the original cell passes to the next iteration.

- **Differentiation for infeasible cells:** the number of decision variables to be changed is determined by a random number $U(1,N)$. Each variable to be changed is chosen in a random way and it is modified according to: $cell.TCR_i^{'} = cell.TCR_i \pm m$, where $cell.TCR_i$ and $cell.TCR_i^{'}$ are the original and the mutated decision variables, respectively. $m = U(0,1) * (cell.ECV + cell.ICS)$. In a random way, it is decided if m will be added or subtracted to $cell.TCR_i$. If the procedure cannot find a TCR_i' in the allowable range, then a random number with a uniform distribution is assigned to it ($cell.TCR_i^{'} = U(cell.TCR_i, P^t_{max_i})$ if m should be added or $cell.TCR_i^{'} = U(P^t_{min_i}, cell.TCR_i)$, otherwise). If the clone is feasible, then the differentiation process stops. Otherwise, the process is applied to the clone instead of the infeasible original cell. This methodology is repeated until N differentiations have been applied or a feasible clone had been reached.

5.2 Handling Constraints

Different violation rates are calculated for equality and inequality constraints. They are called ECV and ICS, respectively, and are detailed next.

- At time t, for each cell j, its ECV_j is calculated as $ECV_j = \mid \sum_{i=1}^{N} TCR_i^t - P_D^t - TCR_L^t \mid)$, where TCR_i^t, P_D^t and TCR_L^t are the output power for unit i, the load demand and the loss transmission, respectively. This rate indicates how far is the generated power from the demanded power. Thus, if $ECV_j > 0$ then the generated power by cell j is larger than the demanded power and if $ECV < 0$, the power generated by cell j is lower than the required power.

- ICS_j represents the inequality constraints sum for cell j, at time t. For each cell, the rate is calculated as $ICS_j = \sum_{i=1}^{N} \sum_{j=1}^{n_i} poz(TCR_i, i, j)$
  $poz(p,i,j) =$
  $\begin{cases} \min(p - PZ_{i,j}^L, PZ_{ij}^U - p) & if p \in [PZ_{i,j}^L, PZ_{ij}^U] \\ 0 & otherwise \end{cases}$
  where n_i indicates the number of prohibited operating zones and $[PZ_{i,j}^L, PZ_{ij}^U]$ is the j^{th} prohibited range for the i^{th} unit. So, if some TCR_i falls in a prohibited zone, the closer distance, between TCR_i and the prohibited zone limit is added to the rate.

A cell is considered feasible only if it produces at least the load demand but it has to be less than a predetermined ϵ ($0 \leq ECV < \epsilon$) for problems with transmission loss or the exact load demand ($ECV = 0$), otherwise. And any TCR must fall in a prohibited zone ($ICS = 0$).

The algorithm works in the following way (see Algorithm 1). First, the TCRs are randomly initialized within the limits of the units (Step 1) (interval 1). Then, ECV and ICS are calculated for each cell (Step 2). Only if a cell is feasible, its objective function value is calculated (Step 3). Next, the following steps are repeated T times (Step 5 to 25): while a predetermined number of objective function evaluations had not been reached and 5×10^7 iterations had not been performed, the cells are proliferated and differentiated according to their feasibility (Step 7). After the activation process, the best solution at time t is recorded. The time (interval) is increased (Step 11) and new operational limits are updated according to Eq. (14) (Steps 12-15). Those units whose power outputs fall outside the new operational limits are replaced by random values from the new valid limits (Steps 16-22). Since the power outputs could change, the TPs are updated and the cells are re-evaluated according to the new power demand (Step 24) and the corresponding objective function value as well, if applicable (Step 25). Finally, (Step 27) the final solution is the union of the solutions found at times 1, 2 to T ($BEST$).

Algorithm 1 IA_DED Algorithm

1: $C \leftarrow$ Initialize_Population();
2: Evaluate_Constraints(C);
3: Evaluate_Objective_Function(C);
4: **for** $t \leq$ T **do**
5:     $top \leftarrow 0$;
6:     **while** A number of evaluations has not been reached $\wedge$ $top < 5 * 10^7$ **do**
7:         Activation_Process(C);
8:         $top{+}{+}$;
9:     **end while**
10:     $best_t \leftarrow$ Search_best_at_Population(C);
11:     $t{+}{+}$;
12:     **for** $j \leq$ N **do**
13:         $P^t_{min_j} = max(P_{min_j}, best_{t-1} - DR_j)$
14:         $P^t_{max_j} = min(P_{max_j}, best_{t-1} + UR_j)$
15:     **end for**
16:     **for** $i \leq \mid C \mid$ **do**
17:         **for** $j \leq$ N **do**
18:             **if** $cell_i.TCR_j \notin [P^t_{min_j}, P^t_{max_j}]$ **then**
19:                 $cell_i.TCR_j \leftarrow U(P^t_{min_j}, P^t_{max_j})$
20:             **end if**
21:         **end for**
22:     **end for**
23:     Update_output_power(C);
24:     Evaluate_Constraints(C);
25:     Evaluate_Objective_Function(C);
26: **end for**
27: $BEST \leftarrow (best_1, best_2, \ldots, best_T)$;

6 Numerical Experiments

The proposed algorithm was tested on six 24-h dynamic power systems (T=24). The first one is a 5-unit system, in which all the units have valve-point effects and transmission losses. The B coefficients were taken from [26]. The total load demand of the system is 14577 MW. The system data and power load demands were taken from [10]. The second example is a 6-unit system with 26 buses, and 46 transmission lines. In this case, valve-point effects are not considered but transmission loss is considered. All units have two prohibited zones and the total load demand is 25954 MW. The data and daily load demands for this problem were taken from [36]. The third system has 10 thermal units, all of which consider valve-point effects but not transmission losses. The total load demand is 40108 MW. The data and daily load demands for this problem were taken from [26]. An extension from this is the 30-unit system where the units are tripled to get a 30 units system. It has the same cost characteristics with valve point load effects. The load pattern is taken as three times the value which is considered in the 10 unit system for a 24 h time period. The fifth power system has 15 generating units (15-unit system), it doesn't consider valve-point effects but it takes into account transmission losses. Four units (2, 5, 6 and 12) have prohibited operating zones. The total load demand is 60981 MW. The data and daily load demands for this problem were taken from [4]. The last test case is a 54-unit system, which comprises 54 thermal units (33 coal-fired units, 11 gas-fired units and 10 oil-fired units) as well as 8 hydro plants [26]. The detailed data of this system were taken from [43]. Thermal units 5, 10, 11, 28, 36, 43, 44 and 45 have valve load effects cost and thermal units 7, 10, 30, 34, 35 and 47 have POZs limitations [44]. Thermal units 8, 9, 10, 11, 15, 16, 17, 18, 21, 22, 23, 24, 28, 29, 30, 31, 32, 33, 34, 35 and 36 can generate emission but this issue is not tracked in this case. The total load demand of the system is 111600 MW. The data pertaining to the demand were taken from [4]. The 5-unit-DEED system has the same data used in the 5-unit system but emissions are considered. In this case, wet set w = 0.5 [4].

Table 1 provides the most relevant features of the problems previously described as well as the maximum number of function evaluations performed by IA_DED. The algorithm was implemented in Java (v. 1.6.0_24) under Linux (Ubuntu 12.04) on a Pentium IV personal Computer while the experiments were performed on an Intel Q9550 Quad Core processor running at 2.83GHz and with 4GB DDR3 1333Mz in RAM. For each problem, 100 independent runs were performed.

7 Comparison of Results and Discussion

Several methods are selected to be compared with our proposed algorithm, according to the chosen test cases. The comparison of results is presented in Tables 2 and 3. These tables show the following: the best, mean, worst, standard deviation as well as the running times obtained by each of the approaches, when available. Due to space restrictions, the integer costs are shown but they are not rounded up. For all the test problems, IA_DED found feasible solutions in all the runs performed, considering the parameters setting given in Table 1, except for the 10-unit system where feasible solutions were found in 86% of the runs. The

Table 1: Test Problems Features

Problem	Objective	P_L	POZ	MaxEv	C	P_c	P_a
5-unit system	non-smooth	Yes	No	19000	10	0.1	0.01
6-unit system	smooth	Yes	Yes	2000	20	0.5	0.1
10-unit system	non-smooth	No	No	5000	10	0.9	0.1
15-unit system	smooth	Yes	No	30000	20	0.9	0.1
30-unit system	non-smooth	No	No	50000	5	0.9	0.1
54-unit system	non-smooth	No	Yes	40000	5	0.9	0.01
5-DEED-unit System	non-smooth	Yes	No	2000	5	0.9	0.1

running times are compared in an indirect manner, to give a rough idea of the computational costs of the different algorithms considered in our comparative study.

Analyzing Table 2, the best total fuel cost obtained by IA_DED is $43699, for the 5-unit system. This cost was outperformed by ICA [26] and DE-SQP [37], but the computational cost is not reported for any of these two approaches. The other approaches, for which the computational cost is reported, required minutes to obtain feasible solutions. In contrast, IA_DED could find very quickly (in seconds instead of minutes), an acceptable solution.

A similar situation occurs when the 6-unit system and the 10-unit system are considered. For the 6-unit system, IA_DED exceeds by $104 the cost found by SAMF [40], but our approach obtained this best total fuel cost just in 0.924 seconds while SAMF [40] required 1.965 seconds. For the 10-unit system, IA_DED exceeds by $1397 the cost found by EBSO [29], but this approach reports a running time of 0.205 minutes, i.e., 12.3 seconds. The other approaches took times measured in minutes to find feasible solutions, whereas our proposed approach took only 2.552 seconds.

Considering Table 3 (15-unit system) IA_DED outperformed all the considered approaches. It finds a solution whose total fuel cost is $759302 in 2.660 seconds. Thus, our proposed approach found the best solution requiring the lowest running time. However, the Brent-Method [39] found an acceptable solution in only 0.53 seconds.

For the 30-unit system, IA_DED obtained a best total fuel cost of $3056592, outperforming all the approaches with respect to which it was compared, except for EBSO [29]. EBSO produced a solution which is only 0.08% cheaper than the one produced by IA_DED, but it required 634% more time than IA_DED.

For the 54-unit system, IA_DED outperformed all the other approaches with respect to which it was compared, in terms of the total fuel cost. IA_DED just required 13.169 seconds to find this solution, whose cost is $1717901. In this case, OCD [41] found a feasible solution which is 3% more expensive than the one produced by IA_DED but it produced it in only 0.132 seconds.

For the 5-unit-DEED system, IA_DED obtained, in 1.216 seconds, a solution whose total fuel cost is $45169 and whose emissions are 18774 lb/day. This solution is less expensive than the solutions produced by the approaches considered for comparison purposes but it releases into the air 144 lb/day more than the NPAHS solution [4].

The best scheduling solution obtained by our proposed IA_DED can be downloaded from `http://www.lidic.unsl.edu.ar/node/461`.The source code of our proposed approach can be obtained from the first author of this paper, upon request.

It is worth noting that the methods considered in this paper, which sub-divide the whole dispatch into T periods such as the Brent Method [39], SAMF [40, 41], and IA_DED, are able to find high-quality solutions in seconds rather than minutes.

8 Statistical Analysis

The parameters required by IA_DED are: population size (C), maximum number of objective function evaluations, change factor (P_c), differentiation probability (P_a) and tolerance factor (ϵ). This last parameter was set to 0.9 for all the test problems that consider transmission losses. To analyze the effect of C, P_c and P_a on IA_DED's behavior, we tested it with different parameters settings. As part of this process, some preliminary experiments were performed to discard some parameter values. Hence, the selected parameter levels were the following: a) Three levels for the population size C (5, 10 and 20 cells), b) Three levels for the probability P_c (0.1, 0.5 and 0.9) and c) Two levels for the probability P_a (0.01 and 0.1).

Thus, we have 18 parameters settings for six problems. They are identified as C <size>-P_c <Prob>-P_a <Prob>, where C, P_c and P_a indicate the population size and the probabilities, respectively. The box plot method was selected to visualize the distribution of the objective function values for each power system. This allowed

Table 2: Comparison of results. The best values are shown in **boldface**. The last column indicates the running time (s ≡ seconds and m ≡ minutes). - denotes that the value was not available.

Problem/ Algorithm	Best($)	Mean($)	Worst($)	Std.	Time
5-unit system					
ICA [26]	**43117**	**43144**	**43209**	19.821	-
DE-SQP [37]	43161	-	-	-	-
ABC [14]	44045	-	-	-	-
PSO [14]	44253	-	-	-	-
HS [5]	44376	-	-	-	2.8m
AIS [23]	44385	44758	45553	-	4m
GA [14]	44862	-	-	-	-
IA_DED	43699	45081	46383	593.68	8.925s
6-unit system					
SAMF [40]	**313363**	-	-	-	1.965s
Brent Method[39]	313405	-	-	-	0.078s
BPSO-DE [36]	314025	314144	314351	-	21.89s
IA_DED	313467	**313497**	**313534**	14.58	0.924s
10-unit system					
EBSO [29]	**1017147**	**1017526**	**1017891**	147.01	0.205m
ICA [26]	1018467	1019291	1021795	-	-
CSADHS [22]	1018681	1018718	1018760	-	2.72m
CDHS [22]	1018683	1018743	1018793	-	2.95m
CSAPSO [21]	1018767	1019874	-	-	0.467m
ICPSO [20]	1019072	1020027	-	-	0.467m
HHS [5]	1019091	-	-	-	12.233m
CDE method3 [18]	1019123	1020870	1023115	-	0.32m
DE [16]	1019786	-	-	-	11.15m
DHS [22]	1019952	1020025	1020107	-	3.34m
AHDE [35]	1020082	1022474	-	-	1.10m
AIS [23]	1021980	1023156	1024973	-	19.01m
ECE [24]	1022271	1023334	-	-	0.5271m
BCO-SQP [38]	1032200	-	-	-	3.24m
IA_DED	1018544	1020193	1022064	764.04	2.552s

us to determine the robustness of our proposed algorithm with respect to its parameters. Figures 1 to 6 show in the x-axis the parameter combinations and the y-axis indicates the objective function values for each problem expressed in \$. Furthermore, we also performed an analysis of variance (ANOVA). The hypotheses considered were the following:

- Null Hypothesis: there is no significant difference among the averages of the objective values. If there are differences, they are due to random effects.

- Alternative Hypothesis: there is a combination of level values for which the average of the objective values are significantly different and such differences are not due to random effects.

As the results do not follow a normal distribution, we applied the Kruskal-Wallis test, to perform ANOVA and then the Tukey method in order to determine the experimental conditions for which significant differences exist. The results obtained by ANOVA proved the Null Hypothesis for several combinations of parameters. However, the Alternative Hypothesis was proved, too.

After the statistical analysis of the results obtained by our proposed approach, for the six test problems, we can infer the following general conclusions. For both the 5-unit system and the 15-unit system, there are no significant differences when C is fixed and the probabilities vary. However, the median values improve with a small change factor. For the 6-unit system, when C is increased, better results are obtained and they have significant differences. Increasing the change factor from 0.1 to 0.5 and 0.9 improves the results with significant differences. For the 10-unit system, increasing the change factor from 0.1 to 0.5 and 0.9 improves the results with significant differences. When $C = 5$ or $C = 10$, increasing P_c from 0.5 to 0.9, also improves the results. In general, best median values are obtained with the highest probability set for the application of the differentiation operator. For the 30-unit system, increasing the change factor improves the results with significant differences. Contrary to the previous case, the best median values are obtained with the lowest probability established for the application of the differentiation operator. Considering the 54-unit system, for $C = 5$, increasing the change factor from 0.1 to 0.5 and 0.9 produces better results and they present significant differences. For $C = 10$ or $C = 20$, increasing the probabilities produces better results.

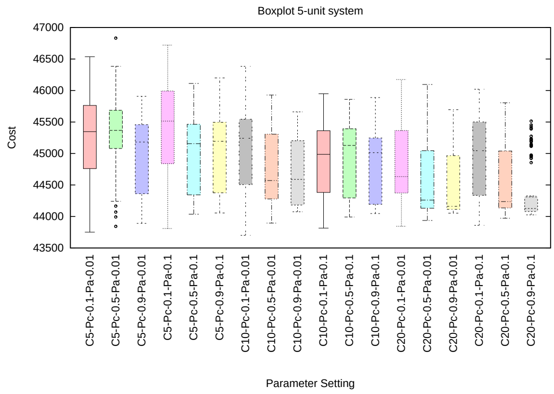

Figure 1: Box plots for 5-unit system.

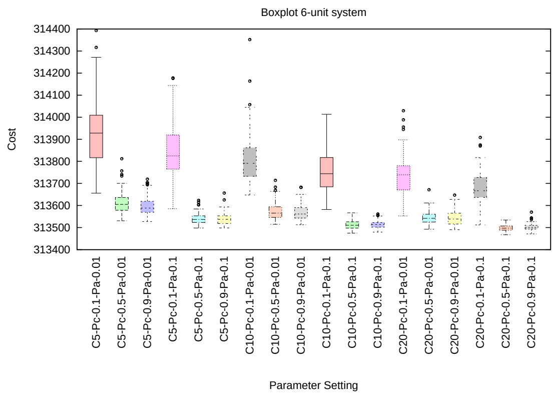

Figure 2: Box plots for 6-unit system.

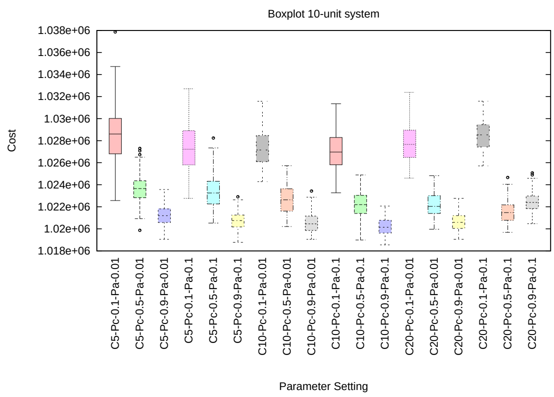

Figure 3: Box plots for 10-unit system.

Table 3: Comparison of results. The best values are shown in **boldface**. The last column indicates the running time (s ≡ seconds and m ≡ minutes). - denotes that the value was not available.

Problem/ Algorithm	Best($)	Mean($)	Worst($)	Std.	Time
15-unit system					
SAMF [40]	759406	-	-	-	2.951s
NPAHS [4]	759603	759779	759988	-	250.0s
CSADHS [22]	759689	759766	**759845**	-	3.36m
SGHS[4]	759897	760118	760343	-	303.3s
HS[4]	765560	765959	766370	-	678.3s
IHS[4]	765600	765942	766403	-	681.5s
GHS[4]	769074	769627	770428	-	1935.1s
Brent Method[39]	760287	-	-	-	0.53s
IA_DED	**759302**	**759542**	760125	149.59	2.660s
30-unit system					
HHS [5]	3057313	-	-	-	27.65m
ICPSO [20]	3064497	3071588	-	-	1.03m
CDE method3 [18]	3083930	3090542	-	-	0.67m
ECE [24]	3084649	3087847	-	-	2.1375m
EBSO [29]	3054001	3054697	3055944	-	0.95m
IA_DED	3056592	3060513	3064397	1545.83	7.756s
54-unit system					
OCD [41]	1772724	-	-	-	0.132s
ICA [26]	1807081	1809664	1811388	-	-
IA_DED	**1717901**	**1718127**	**1718411**	108.08	13.169s
5-unit-DEED system					
HS [4]	33249	-	-	-	1192.00s
		TC: 47375		Emission: 19123	
IHS [4]	33388	-	-	-	1243.75s
		TC: 47946		Emission:18830	
GHS[4]	34216	-	-	-	2317.70s
		TC:49503		Emission: 18929	
SGHS[4]	32417	-	-	-	860.s
		TC:45870		Emission: 18964	
DE[4]	34079	-	-	-	1326.22s
		TC:48882		Emission: 19276	
PSO-CF[4]	34198	-	-	-	1068.90s
		TC:49211		Emission: 19185	
NPAHS [4]	**31913**	-	-	-	235.93s
		TC: 45196		Emission: **18630**	
IA_DED	31972	**32353**	**32748**	128.18	1.216s
		TC: **45169**		Emission: 18774	

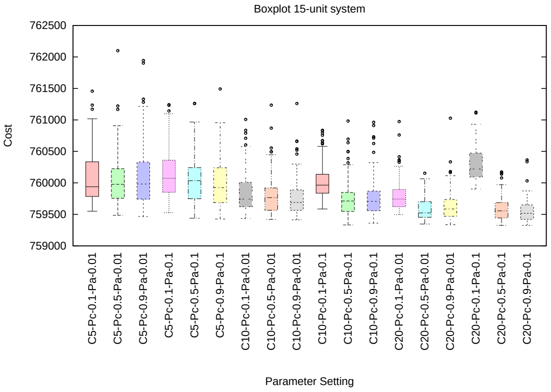

Figure 4: Box plots for 15-unit system.

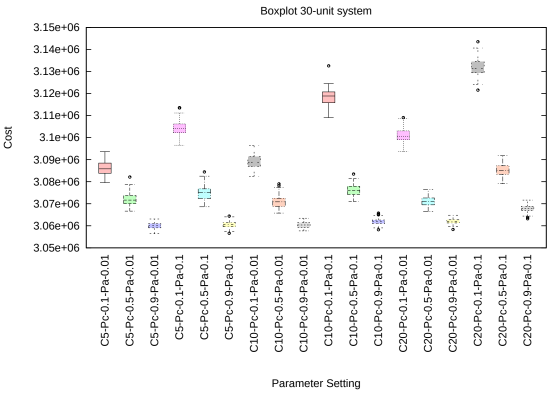

Figure 5: Box plots for 30-unit system.

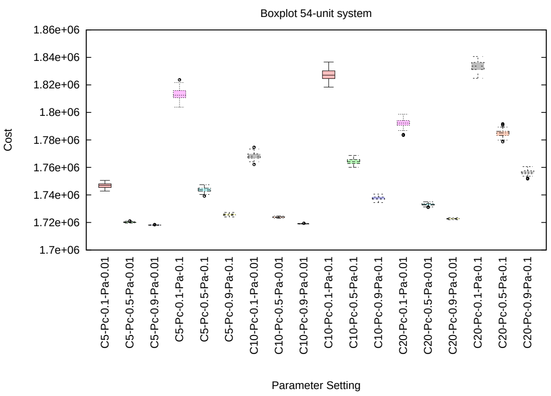

Figure 6: Box plots for 54-unit system.

9 Conclusions and Future Work

This paper presented an algorithm inspired on the T-Cells of the immune system, called IA_DED, which was used to solve dynamic economic dispatch problems. IA_DED is able to handle the different types of constraints that are involved in this type of problem: power balance constraints with and without transmission loss, operating limit constraints, ramp rate limit constraints and prohibited operating zones. Additionally, it can handle both smooth and non-smooth functions as well as atmospheric emissions.

At the beginning, the search performed by IA_DED is based on a simple differentiation operator which takes an infeasible solution and modifies some of its decision variables by taking into account their constraint violation. Once the algorithm finds a feasible solution, a different differentiation operator is applied. This operator modifies two decision variables at a time, it decreases the power in one unit, and it selects another unit to generate the power that has been taken.

Our proposed approach was validated with six test problems having different features. Comparisons were provided with respect to several approaches that have been reported in the specialized literature. Our proposed approach produced competitive results in all cases, being able to outperform some of the other approaches when running times are considered. Also, it showed an acceptable behavior in a DEED problem. The best performance of our proposed algorithm is observed in the largest systems with which it was tested. Furthermore, the best results were obtained when the highest change factor probability was used. As part of our future work, we are interested in testing the algorithm with even larger systems and we intend to incorporate renewable energy resources.

Funding

The first two authors acknowledge support from Universidad Nacional de San Luis, project no. P03-1018 / 22F435. The second author acknowledges support from CONACyT project no. 2016-01-1920 (*Investigación en Fronteras de la Ciencia 2016*) and from a project from the 2018 SEP-Cinvestav Fund (application no. 4).

Competing interests

The authors declare that they have no conflict of interest.

References

[1] X. Xia and A. Elaiw, "Optimal dynamic economic dispatch of generation: A review,"

Electric Power Systems Research, vol. 80, no. 8, pp. 975–986, 2010.

[2] X. Han and H. Gooi, "Effective economic dispatch model and algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 29, no. 2, pp. 113–120, 2007.

[3] V. S. Aragón, C. A. Coello Coello, and M. G. Leguizamón, "An Artificial Immune System for Solving Dynamic Economic Power Dispatch Problems," in *XXV Congreso Argentino de Ciencias de la Computación*, Universidad Nacional de Rio Cuarto, 2019.

[4] Q. Niu, H. Zhang, K. Li, and G. W. Irwin, "An efficient harmony search with new pitch adjustment for dynamic economic dispatch," *Energy*, vol. 65, pp. 25–43, 2014.

[5] V. R. Pandi and B. K. Panigrahi, "Dynamic economic load dispatch using hybrid swarm intelligence based harmony search algorithm," *Expert Systems with Applications*, vol. 38, pp. 8509–8514, 2011.

[6] F. Zaman, S. M. Elsayed, T. Ray, and R. A. Sarker, "Configuring two-algorithm-based evolutionary approach for solving dynamic economic dispatch problems," *Engineering Applications of Artificial Intelligence*, vol. 53, pp. 105–125, 2016.

[7] Y. Fukuyama and Y. Ueki, "An application of neural network to dynamic dispatch using multi processors," *IEEE Transactions on Power Systems*, vol. 9, pp. 1759–1765, November 1994.

[8] R.-H. Liang, "A neural-based redispatch approach to dynamic generation allocation," *IEEE Transactions on Power Systems*, vol. 14, pp. 1388–1393, November 1999.

[9] F. Benhamida and R. Belhachem, "Dynamic constrained economic/emission dispatch scheduling using neural network," *Power Engineering and Electrical Engineering*, vol. 11, pp. 1–9, March 2013.

[10] C. Panigrahi, P. Chattopadhyay, R. Chakrabarti, and M. Basu, "Simulated annealing technique for dynamic economic dispatch," *Electric Power Components and Systems*, vol. 34, no. 5, pp. 577–586, 2006.

[11] W. R. Barcelo and P. Rastgoufard, "Dynamic economic dispatch using the extended security constrained economic dispatch algorithm," *IEEE Transactions on Power Systems*, vol. 12, pp. 961–967, May 1997.

[12] F. Li, R. Morgan, and D. Williams, "Towards more cost saving under stricter ramping rate constraints of dynamic economic dispatch problems-a genetic based approach," in *Second International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications (GALESIA'97)*, pp. 221–225, September 1997.

[13] W. Ongsakul and J. Tippayachai, "Parallel micro genetic algorithm based on merit order loading solutions for constrained dynamic economic dispatch," *Electric Power Systems Research*, vol. 61, no. 2, pp. 77–88, 2002.

[14] S. Hemamalini and S. Simon, "Dynamic economic dispatch using artificial bee colony algorithm for units with valve-point effect," *European Transactions on Electrical Power*, vol. 21, pp. 70–81, 2011.

[15] M. F. Zaman, S. M. Elsayed, T. Ray, and R. A. Sarker, "Evolutionary algorithms for dynamic economic dispatch problems," *IEEE Transactions on Power Systems*, vol. 31, pp. 1486–1495, March 2016.

[16] R. Balamurugan and S. Subramanian, "Differential Evolution-based Dynamic Economic Dispatch of Generating Units with Valve-point Effects," *Electric Power Components and Systems*, vol. 36, no. 8, pp. 828–843, 2008.

[17] X. Yuan, L. Wang, Y. Yuan, Y. Zhang, B. Cao, and B. Yang, "A modified differential evolution approach for dynamic economic dispatch with valve-point effects," *Energy Conversion and Management*, vol. 49, no. 12, pp. 3447–3453, 2008.

[18] Y. Lu, J. Zhou, H. Qin, Y. Wang, and Y. Zhang, "Chaotic differential evolution methods for dynamic economic dispatch with valve-point effects," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 2, pp. 378–387, 2011.

[19] Z. L. Gaing and T. C. Ou, "Dynamic economic dispatch solution using fast evolutionary programming with swarm direction," in *2009 4th IEEE Conference on Industrial Electronics and Applications*, pp. 1538–1544, May 2009.

[20] Y. Wang, J. Zhou, H. Qin, and Y. Lu, "Improved chaotic particle swarm optimization algorithm for dynamic economic dispatch problem with valve-point effects," *Energy Conversion Management*, vol. 51, pp. 2893–2900, December 2010.

[21] Y. Wang, J. Zhou, Y. Lu, H. Qin, and Y. Wang, "Chaotic self-adaptive particle swarm optimization algorithm for dynamic economic dispatch problem with valve-point effects," *Expert Systems with Applications*, vol. 38, pp. 14231–14237, October 2011.

[22] R. Arul, G. Ravi, and S. Velusami, "Chaotic self-adaptive differential harmony search algorithm based dynamic economic dispatch," *International Journal of Electrical Power & Energy Systems*, vol. 50, pp. 85–96, September 2013.

[23] S. Hemamalini and S. Simon, "Dynamic economic dispatch using artificial immune system for units with valve-point effect," *International Journal of Electrical Power & Energy Systems*, vol. 33, pp. 868–874, 2011.

[24] A. I. Selvakumar, "Enhanced cross-entropy method for dynamic economic dispatch with valve-point effects," *International Journal of Electrical Power & Energy Systems*, vol. 33, no. 3, pp. 783–790, 2011.

[25] T. Niknam, M. R. Narimani, J. Aghaei, S. Tabatabaei, and M. Nayeripour, "Modified Honey Bee Mating Optimisation to solve dynamic optimal power flow considering generator constraints," *IET Generation, Transmission & Distribution*, vol. 5, pp. 989–1002, October 2011.

[26] B. Mohammadi-ivatloo, A. Rabiee, A. Soroudi, and M. Ehsan, "Imperialist competitive algorithm for solving non-convex dynamic economic power dispatch," *Energy*, vol. 44, pp. 228–240, August 2012.

[27] T. Niknam, R. Azizipanah-Abarghooee, and A. Roosta, "Reserve Constrained Dynamic Economic Dispatch: A New Fast Self-Adaptive Modified Firefly Algorithm," *IEEE Systems Journal*, vol. 6, pp. 635–646, December 2012.

[28] T. Niknam, R. Azizipanah-Abarghooee, and J. Aghaei, "A new modified teaching-learning algorithm for reserve constrained dynamic economic dispatch," *IEEE Transactions on Power Systems*, vol. 28, pp. 749–763, May 2013.

[29] T. Niknam and F. Golestaneh, "Enhanced Bee Swarm Optimization Algorithm for Dynamic Economic Dispatch," *IEEE Systems Journal*, vol. 7, pp. 754–762, December 2013.

[30] K. Meher, R. Swain, and C. Chanda, "Solving dynamic economic dispatch problems facilitating meta-heuristic technique," *International Journal of Engineering Research & Technology*, vol. 4, September 2015.

[31] Hardiansyah, "A Modified ABC Algorithm for Solving Non-Convex Dynamic Economic Dispatch Problems," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 6, pp. 2621–2628, December 2016.

[32] P. Attaviriyanupap, H. Kita, E. Tanaka, and J. Hasegawa, "A Hybrid EP and SQP for Dynamic Economic Dispatch with Nonsmooth Fuel Cost Function," *IEEE Power Engineering Review*, vol. 22, pp. 411–416, May 2002.

[33] A. Abdelaziz, M. Kamh, S. Mekhamer, and M. Badr, "A hybrid HNN-QP approach for dynamic economic dispatch problem," *Electric Power Systems Research*, vol. 78, no. 10, pp. 1784–1788, 2008.

[34] X. Yuan, A. Su, Y. Yuan, H. Nie, and L. Wang, "An improved PSO for dynamic load dispatch of generators with valve-point effects," *Energy*, vol. 34, no. 1, pp. 67–74, 2009.

[35] Y. Lu, J. Zhou, H. Qin, Y. Li, and Y. Zhang, "An adaptive hybrid differential evolution algorithm for dynamic economic dispatch with valve-point effects," *Expert Systems with Applications*, vol. 37, no. 7, pp. 4842–4849, 2010.

[36] K. Vaisakh, P. Praveena, S. R. M. Rao, and K. Meah, "Solving dynamic economic dispatch problem with security constraints using bacterial foraging PSO-DE algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 39, no. 1, pp. 56–67, 2012.

[37] A. M. Elaiw, X. Xia, and A. M. Shehata, "Hybrid DE-SQP and hybrid PSO-SQP methods for solving dynamic economic emission dispatch problem with valve-point effects," *Electric Power Systems Research*, vol. 103, pp. 192–200, October 2013.

[38] M. Basu, "Hybridization of bee colony optimization and sequential quadratic programming for dynamic economic dispatch," *International Journal of Electrical Power & Energy Systems*, vol. 44, no. 1, pp. 591–596, 2013.

[39] K. Chandram, N. Subrahmany, and M. Sydulu, "Brent method for dynamic economic dispatch with transmission losses," *Iranian Journal Electrical and Computer Engineering*, vol. 8, no. 1, pp. 16–22, 2009.

[40] S. Ganesan and S. Subramani, "Dynamic Economic Dispatch Based on Simple Algorithm," *International Journal of Computer and Electrical Engineering*, vol. 3, no. 2, pp. 226–232, 2011.

[41] A. Rabiee, B. Mohammadi-Ivatloo, and M. Moradi-Dalvand, "Fast Dynamic Economic Power Dispatch Problems Solution Via Optimality Condition Decomposition," *IEEE Transactions on Power Systems*, vol. 29, no. 2, pp. 982–983, 2014.

[42] L. Nunes de Castro and J. Timmis, *Artificial Immune Systems: A New Computational Intelligence Approach*. New York: Springer-Verlag, 2002.

[43] M. Shahidehpour. Available from: motor.ece.iit.edu/data/SCUC_118test.xls [accessed in 2015].

[44] M. Shahidehpour. Available from: motor.ece.iit.edu/data/118_nonsmooth.xls [accessed in 2015].