

<http://ansinet.com/itj>

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSI*net*

Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Dynamic Optimal Power Flow in FSWGs Integrated Power System

Gonggui Chen

College of Electrical and Electronic Engineering,
Huazhong University of Science and Technology, Wuhan 430074, Hubei Province, China

Abstract: Dynamic Optimal Power Flow (DOPF) is a typical complex multi-constrained non-convex non-linear programming problem when considering the valve-point effect of conventional generators. Moreover, it is mixed integer when considering the discreteness of reactive compensation devices for power system with Fixed Speed Wind Generators (FSWGs). DOPF model in FSWGs integrated power system is established in this study and then a hybrid Taguchi-Particle Swarm Optimization (TPSO) algorithm is proposed for solving the established DOPF model. This hybrid algorithm combines the well-known Particle Swarm Optimization (PSO) algorithm with the established Taguchi method which has been an important tool for robust design. This study clearly presents the improvements obtained despite the simplicity of the hybridization process. The Taguchi method is run only once in every iteration and therefore does not give significant impact in terms of computational cost. The method creates a more diversified population, which also contributes to the success of avoiding premature convergence. IEEE 39-bus system is used to illustrate the effectiveness of the proposed method compared with those obtained from PSO algorithm. The test results show that the proposed method is effective and has a certain practicality.

Key words: Wind power generation, dynamic optimal power flow, Taguchi, particle swarm optimization, reactive compensation

INTRODUCTION

Wind energy is the world's fastest growing renewable energy source. With the increasing levels of wind generator penetration in modern power systems, one of major challenges in the present and coming years is the optimization control, such as optimal power flow including wind farms (Li and Chen, 2008).

The well-known advantages of FSWGs (fixed speed wind generators) are it is robust, easy and relatively cheap for mass production, there are still widely use in power system with FSWGs. Since FSWGs always draw reactive power from the grid, in most cases, capacitors are connected in parallel to the generator to compensate for the reactive power consumption, the P-bus power flow model of FSWGs bus is proposed in this study.

In this study, the problems of DOPF (Dynamic Optimal Power Flow) including FSWGs are researched. The expectation model of wind generators' active power outputs is adopted. DOPF is a method to schedule the online generator outputs with the predicted load demands over a certain period of time so as to operate an electric power system most economically. Normally, it is solved by dividing the entire dispatch period into a number of small time intervals, then a static economic dispatch has been employed to solve the problem in each interval (Pothiya *et al.*, 2008; Urtubey and Costa, 2007). In this

study, DOPF model, which takes all conventional units cost minimum as the objective function and takes the whole time and the inherent relations of different intervals into account in FSWGs integrated system, is established.

DOPF is a typical complex multi-constrained non-convex non-linear programming problem when considering the valve-point effect of conventional generators. Moreover, it is mixed integer when considering the discreteness of FSWGs reactive compensation devices. However, both lambda-iterative and gradient technique methods in conventional approaches to the problems are calculus-based techniques and require a smooth and convex cost function and strict continuity of the search space.

On intelligent algorithms, easy to handle discrete variables, no special requirements for the objective function, no problems of dimension disaster, strong global search capability and the greater probability in finding out the global optimum solution are the prominent advantages of these algorithms. However, the computational accuracy and convergence rate are not high for most intelligent algorithm.

In current study, we see a myriad of improved algorithms. Many of these algorithms incorporate some improvement strategies in the algorithms itself while some improved algorithms are the result of hybridizing two algorithms (He *et al.*, 2008; Yuan *et al.*, 2009). The

Taguchi method is an established approach for robust design, applying the idea from statistical experiment design for evaluating and improvements in products, processes and equipment (Taguchi *et al.*, 2000, 2005; Ross, 1989). The key to this concept is to improve the quality of a product by minimizing the effect of the causes of variation without the elimination of the relevant causes. The two major tools used in the Taguchi method are: (1) Signal-to-Noise Ratio (SNR) which measures quality and (2) orthogonal arrays which are used to study many design parameters simultaneously (Taguchi *et al.*, 2000, 2005; Ross, 1989). This study adopts the concept of Taguchi method into PSO algorithm to improve its convergence which is so called Taguchi-PSO (TPSO). By applying the algorithm to dealing with IEEE 39-bus system, compared with PSO algorithm, the experimental results show that the algorithm is an effective way to solving DOPF problems in FSWGs integrated power system because of higher computational accuracy, better robustness and convergence performance.

IEEE 39-bus system is used to illustrate the effectiveness of the proposed method compared with those obtained from PSO algorithm. The test results show that the proposed method is effective and has a certain practicality.

DOPF MODEL IN FSWGS INTEGRATED POWER SYSTEM

Due to the random variation of the wind velocities and load demands, it is difficult to research the DOPF in the power system including wind farms. For simplifying this problem, the dividing-stage strategy is adopted in this study. According to the wind velocity forecasting curves and the load forecasting curves in the planning horizon, the expectations of wind generators' outputs and the load demands at dispatch interval can be calculated.

Constraints: Constraints include equality and inequality constraints. The equation constraint is the power flow formulation constraint while inequality constraints including generator power output, ramp rate and bus voltage are as in Eq. 1-4. For the FSWGs bus, absorbed reactive power from the power system is bounded as in Eq. 4. The constraints of real power generation limit and the ramp rate are taken into account as in Eq. 1.

$$\left\{ \begin{array}{l} \max(P_{i,\min}, P_i^{t-1} - D_{R_i}\Delta T) \leq P_i^t \leq \min(P_{i,\max}, P_i^{t-1} + U_{R_i}\Delta T) \\ i \in N_{gen} \end{array} \right. \quad (1)$$

$$Q_{G_i,\min} \leq Q_{G_i}^t \leq Q_{G_i,\max}, i \in N_{gen} \quad (2)$$

$$V_{i,\min} \leq V_i^t \leq V_{i,\max}, i \in N \quad (3)$$

$$Q_{ab}^t \leq Q_{ab}^{\min} \quad (4)$$

where, $P_{i,\min}$ and $P_{i,\max}$ are the maximum and minimum limits of the power generation of unit i , P_i^t is the real power output of unit i at the t th interval, P_i^{t-1} is the real power output of unit i at the $t-1$ th interval; U_{R_i} is the up-ramp limit of the i th generator (in units of MW/time-period) and D_{R_i} is the down-ramp limit of the i th generator (in units of MW/time-period); ΔT is time interval, N_{gen} is the number of conventional generating units and N is the number of system buses (excluding slack bus); V_i^t is the voltage magnitude output of bus i at the t th interval; $Q_{G_i}^t$ is the reactive power output of conventional generating unit i at the t th interval; Q_{ab}^t is the absorbed reactive power of wind farm bus at the t th interval; max is the maximum value of the variable, min is the minimum value of the variable.

After calculating the power flow, the state variables, power loss and real power output of the slack bus generator corresponding to the current control variables are available. The real power output of the slack bus generator will be set to the limit if it violates the limit. After handling overlimit of the real power output of the slack bus generator, the system power balance constraints as in Eq. 5 must meet, otherwise adding Eq. 5 as penalty terms to the objective function to form a generalized objective function. Details of the generalized objective function used in this study are given in Eq. 8.

$$\Delta P^t = \sum_{i=1}^{N_{gen}-1} P_i^t + P_{sl}^t + P_{w,sv}^t - P_{ls}^t - P_{ld}^t = 0 \quad (5)$$

where, ΔP^t is the unbalance of the real power at the t th interval, $N_{gen}-1$ represents the number of conventional generating units excluding the slack bus, P_{sl}^t is the real power output of the slack bus generator after handling its overlimit at the t th interval, P_{ls}^t is the total power loss at the t th interval, P_{ld}^t is the total load expectation at the t th interval, $P_{w,sv}^t$ is the expectation of wind generators' real power outputs at the t th interval.

Objective function: Due to the fact that wind generation does not consume the fuel, the utility must purchase all the energy produced by wind generating units. Consequently, the objective is to minimize the following total incremental fuel cost function F associated to N_{gen} dispatchable units for T intervals in the given time horizon, subject to the above-mentioned equality and inequality constraints.

$$\min F = \sum_{t=1}^T \sum_{i=1}^{N_{gen}} F(P_i^t) \quad (6)$$

The inclusion of valve-point loading effects makes the modeling of the fuel cost function of the unit more practical. This increases the non-linearity and local optima in the solution space. Also, the solution procedure can easily trap in the local optima in the vicinity of optimal value. The fuel cost function of the *i*th unit $F(P_i^t)$ with valve-point loadings are represented as follows:

$$F(P_i^t) = a_i + b_i P_i^t + c_i P_i^{t2} + |e_i \sin(f_i (P_{i,max} - P_i^t))| \quad (7)$$

where a_i , b_i and c_i are cost coefficients and e_i , f_i are constants from the valve-point effect of the *i*th generating unit.

Evaluation function: We must define the evaluation function for evaluating the fitness of each individual in the population. In the most of the nonlinear optimization problems, the constraints are considered by generalizing the objective function using penalty terms.

FSWGs always draw reactive power from the grid. According to the principle of the nearest reactive power compensation, the required reactive power of wind farm doesn't absorb from the system as much as possible, but mainly from the wind farm reactive compensation devices. Because reactive power compensation devices are group switched capacitors in wind farms, the maximum of absorbed reactive power of wind farm from the grid is set to one set of group switched capacitors which is considered by generalizing the objective function using penalty terms. Considering efficiency maximization of wind farm reactive power compensation devices, the injected reactive power of wind farm bus is not constrained.

To sum up, the above problems are generalized as follows:

$$\min \{ \sum_{t=1}^T \sum_{i=1}^{N_{gen}} F(P_i^t) + K_v \sum_{t=1}^T \sum_{i \in N_{PQ}} (V_i^t - V_i^{lim})^2 + K_Q \sum_{t=1}^T \sum_{i \in N_{gen}} (Q_{Gi}^t - Q_{Gi}^{lim})^2 + K_C \sum_{t=1}^T \sum_{i=1}^{N_{ab}} (Q_{ab}^t - Q_{ab}^{lim})^2 + K_D \sum_{t=1}^T (\Delta P^t)^2 \} \quad (8)$$

where, K_v , K_Q , K_C and K_D are variable overlimit penalty coefficients, V_i^t is the voltage magnitude of bus *i* at the *t*th interval (excluding the slack bus and PV bus); Q_{Gi}^t is the reactive power output of generator *i* at the *t*th interval; Q_{ab}^t is the absorbed reactive power of wind farm bus at the *t*th interval; V_i^{lim} and Q_{Gi}^{lim} denote the violated upper or lower limits; Q_{ab}^{lim} denotes the violated lower limit that is defined as:

$$Q_{ab}^{lim} = \begin{cases} Q_{ab}^{max}, Q_{ab}^t > Q_{ab}^{max} \\ Q_{ab}^t, Q_{ab}^t \leq Q_{ab}^{max} \end{cases} \quad (9)$$

where, Q_{ab}^{max} is the upper limit value of the absorbed reactive power of wind farm bus which is set to 1Mvar in the study.

In this study, K_v , K_Q and K_C are set to 1,1,1 respectively. Because the unbalance of the real power ΔP is hard to meet, an adaptive penalty function to handle penalty coefficient K_D is adopted, $K_D = k\sqrt{k} \gamma \beta^a$ where, k is the algorithm's current iteration number; β is a relative violated value of the constraints, γ is a multi-stage assignment value, a is the power of the penalty value.

Meanwhile, several experiments have been done in order to obtain the penalty parameters. In this study, if $\beta \leq 1$ then $\alpha = 1$, otherwise $\alpha = 2$. Furthermore, if $\beta \leq 0.001$, then $\gamma = 1$, else, if $\beta \leq 0.01$ then $\gamma = 10$, else, if $\beta \leq 0.1$ then $\gamma = 30$, else, if $\beta \leq 1$ then $\gamma = 100$, otherwise $\gamma = 300$.

THE CONCEPT OF TAGUCHI METHOD

Taguchi methods were developed by Genichi Taguchi as an aid to improving the implementation of total quality control in Japanese industry. He described his methods as 'quality engineering and, in fact, the methods take an engineering approach to the understanding of process information. The methods are based on the design of experiments to provide near optimal quality characteristics for a specific objective. Taguchi defines quality in a negative manner as the loss imparted to society from the time the product is shipped (Taguchi *et al.*, 2000).

The method is based on several statistical concepts which have proven to be valuable tool in the subject of quality improvement. Many Japanese manufacturers have applied this approach for improving product and process quality with unprecedented success. Taguchi essentially utilizes the conventional statistical tools, which has been simplified by identifying a set of stringent guidelines for experiment layout and the analysis of results. Recently Western industries have begun to recognize Taguchi's method as simple but highly effective approach in improving product and process quality. Taguchi method has a wide range of applications in other areas (Cornejo-Mazon *et al.*, 2008; Nava-Arenas *et al.*, 2008). Two major tools used in the Taguchi method are the orthogonal array and the SNR. This study incorporates the two-level orthogonal arrays (Table 1), whereby the two levels are denoted by number 1 and 2 in the Table 1. The basic concept of the structure and usage of two-level orthogonal arrays are briefly described here. This is adequate in the context of describing the development of the hybrid TPSO algorithm.

Table 1: $L_8(2^7)$ orthogonal array

Experiment No.	Factors							Fitness
	A	B	C	D	E	F	G	
	Column No.							
1	1	1	1	1	1	1	1	f_1
2	1	1	1	2	2	2	2	f_2
3	1	2	2	1	1	2	2	f_3
4	1	2	2	2	2	1	1	f_4
5	2	1	2	1	2	1	2	f_5
6	2	1	2	2	1	2	1	f_6
7	2	2	1	1	2	2	1	f_7
8	2	2	1	2	1	1	2	f_8

The general form of two-level standard orthogonal arrays can be represented by the following mathematical term:

$$L_n(2^{n-1}) \tag{10}$$

where, $n = 2^k$, n is the number of experimental runs, k is a positive integer which is greater than 1, 2 is the number of levels for each factor and $n-1$ is the number of columns in the orthogonal array. The size of the orthogonal array to be used depends on the problem instance. At a time only two particles (two factors) are executed at two different levels for n number of experiments. The two-level standard orthogonal arrays most often used in practice are $L_4(2^3)$, $L_8(2^7)$, $L_{16}(2^{15})$, $L_{32}(2^{31})$, $L_{64}(2^{63})$, $L_{128}(2^{127})$ and so on.

Orthogonal arrays are readily composed and are available from many texts (Ross, 1989). The way that they are constructed is to have (1) each level of every factor appear the same number of times in every column of the array and (2) each combination of factors between any two columns appears the same number of times. For any given values of (number of levels) and (number of factors), (number of tests) is determined as the smallest number that satisfies (1) and (2) above.

In this study, a two-level orthogonal array is used. There are m factors, where m is the number of design factors (variables) and each factor has two levels. If, $m < n-1$, only the first columns are used, while the other $n-1-m$ columns are ignored. For example, there are 6 factors with two levels for each factor. We only need 6 columns to allocate these factors and $L_8(2^7)$ is enough for this purpose because it has 7 columns. Table 1 shows an orthogonal array $L_8(2^7)$. The number on the left of each row is called the run number or experiment number and runs from 1 to 8. In fact, each row represents a particle with only first 6 columns. In our methodology, we randomly choose two particles in every iteration. For this example, perform 8 experiments based on Table 1 whereby

the number (either 1 or 2) of each column is the dimension from first and second particle, respectively.

Many designed experiments use matrices called orthogonal arrays for each experimental run and for analyzing the data. The array is called orthogonal because all columns can be evaluated independently of one another, as shown in Table 1. As one column represents a particle, therefore each column has its own fitness value (f_1, f_2, \dots, f_8). The primary goal in conducting this matrix experiment is to determine the best or the optimal level for each factor. In taguchi based PSO, two tools of taguchi method i.e. orthogonal array and SNR are used. The basic idea is to identify the one best particle in an orthogonal array experiment, which consists of a set of experiments conducted at two different levels.

Besides two-level orthogonal arrays, there exists three-level and four-level orthogonal arrays which can provide better precision to the specified process. The detailed description of how these orthogonal arrays are formed can be found in the books written by Ross (1989), Roy (1990) and Taguchi *et al.* (2000).

OVERVIEW OF PARTICLE SWARM OPTIMIZATION

Kennedy and Eberhart (1995) first introduced the Particle Swarm Optimization (PSO) method. It is one of the optimization method categorized in the family of evolutionary computation. The method has been found to be robust in solving real-world problems featuring non-differentiability, high dimension, multiple optima and non-linearity which is widely used in various fields (Heng *et al.*, 2006; Sutha and Kamaraj, 2008; Lai-Jun *et al.*, 2009; Yang *et al.*, 2009; Zakermoshfegh *et al.*, 2008). PSO algorithm is a model that mimics the movement of individuals (fishes, birds, or insects) within a group (school, flock and swarm). Similar to GA, a PSO consists of a population refining its knowledge of the given search space. PSO is inspired by particles moving around in the search space. The individuals in a PSO thus have their own positions and velocities. Each particle moves in the search space with velocity which is dynamically adjusted and balanced based on its own best movement (pBest) and the best movement of the group (gBest).

Instead of using evolutionary operators such as selection, mutation and crossover, each particle in the population moves in the search space with velocity which is dynamically adjusted and all particles are assumed to be of no volume. Each particle keeps track of its coordinates in the search space, which are associated with the best solution it has achieved so far. This value is known as pBest. Another best value that is tracked by the global

version of the particle swarm optimizer is the overall best value or the best solution in the population is called gBest.

The PSO concept consists of, at each time step, changing the velocity of each particle toward its pBest and gBest solutions. The movement is weighted by a random term, with separate random numbers being generated toward pBest and gBest values. For example, the *i*th particle consisting *d* dimensions is represented as $X_i = (X_{i,1}, X_{i,2}, X_{i,3}, \dots, X_{i,d})$. The same notation applied to the velocity, $V_i = (V_{i,1}, V_{i,2}, V_{i,3}, \dots, V_{i,d})$. The best previous position of the *i*th particle is recorded and represented as $pBest_i = (pBest_{i,1}, pBest_{i,2}, pBest_{i,3}, \dots, pBest_{i,d})$. In the case of minimization that we consider in this study, the value of $pBest_i$ with lowest fitness is known as gBest. The modification of velocity and position can be calculated using the current velocity and the distance from $pBest_i$ to gBest as shown in the following formulas

$$V_{i,j}^t = wV_{i,j}^{t-1} + \rho_1 r_1 (gBest_{i,j} - X_{i,j}^{t-1}) + \rho_2 r_2 (pBest_{i,j} - X_{i,j}^{t-1}) \quad (11)$$

$$X_{i,j}^t = X_{i,j}^{t-1} + V_{i,j}^t \quad (12)$$

where, $i \in 1 \dots N, j \in 1 \dots d, t \in 1 \dots T$ with *N* is the number of population size, *d* is the number of dimension and *T* is the number of maximum generation.

The position, *X* of each particle is updated for every dimension for all particles in each iteration. This is done by adding the velocity vector to the position vector, as described in Eq. 12 above. In Eq. 11, *w* is known as the inertia weight. This parameter was introduced by Shi and Eberhart (1998) to accelerate the convergence of PSO. Suitable selection of *w* provides a balance between global and local explorations, thus requiring less iteration on average to find sufficiently optimal solution. Low values result in particles moving in the region far from the optimum value before being tugged back. On the other hand, high values result in abrupt movement toward target regions.

The parameters ρ_1 and ρ_2 are set to constant values, which are normally given as 2.0 whereas r_1 and r_2 are two random values, uniformly distributed in [0,1]. The constants, ρ_1 and ρ_2 represent the weighting of the stochastic acceleration terms that pull each particle toward pBest and gBest positions.

TAGUCHI-PARTICLE SWARM OPTIMIZATION

Here, we illustrate with a simple example the concept of Taguchi-PSO. The Taguchi method is performed after the entire population is updated via PSO equations. Following this Taguchi method, further elaborations are given in a list of steps below. All the pseudocodes are

written in programming style for $L_8(2^7)$, other two-level standard orthogonal arrays such as $L_4(2^3)$, $L_{16}(2^{15})$, $L_{32}(2^{31})$, $L_{64}(2^{63})$ and $L_{128}(2^{127})$ etc., can be analogized.

- **Step 1:** Choose two particles randomly from the current population, noted as P_1 and P_2
- **Step 2:** Create 8 new particles; noted as $T_1, T_2 \dots T_8$ from P_1 and P_2 . (Each particle represents one experiment as shown in Table 1)
- **Step 3:** Assign all the relevant dimensions (columns A-G) to the 8 new particles $T_1, T_2 \dots T_8$ from P_1 and P_2 based on these columns in orthogonal array given in Table 1. Number 1 means the relevant dimension of new particle is copied from P_1 whereas number 2 means the relevant dimension is from P_2
- **Step 4:** Evaluate the fitness of $T_1, T_2 \dots T_8$ (refer to last column of Table 1)
- **Step 5:** Calculate the SNR of various factors (A, B, ..., G) defined as:

$$E(\text{factor}, \text{level}) = \sum_{\substack{i \in (\text{level}=1) \\ \text{or} (\text{level}=2)}}^8 f_i^2 \quad \text{factor} \in [A, B, \dots, G], \quad \text{level} \in [1, 2],$$

For instance, for the case of factor = B and level = 1, we have:

$$E(B, 1) = \sum_{i=1,2,5,6}^8 f_i^2 = f_1^2 + f_2^2 + f_5^2 + f_6^2$$

The same calculation is applied to the similar factor with level = 2:

$$E(B, 2) = \sum_{i=3,4,7,8}^8 f_i^2 = f_3^2 + f_4^2 + f_7^2 + f_8^2$$

- **Step 6:** Assign the relevant dimension to the optimal particle P^0 based on the following rule, taking factor =A in this case

```

If E(A,1) < E(A,2) Then
P0.x(A)=P1.x(A)
Else
P0.x(A)=P2.x(A)
Endif
    
```

- **Step 7:** Replace the worse particle among the two randomly chosen if the optimal particle is better than both. Also, update the gBest as follows

```

If P0.Fit < P1.Fit And P0.Fit < P2.Fit Then
If P1.Fit > P2.Fit Then           P1 = P0
If P1.Fit < gBest.Fit Then       gBest = P1
Elseif P2.Fit > P1.Fit Then     P2 = P0
If P2.Fit < gBest.Fit Then       gBest = P2
End If
End If
    
```

- **Step 8:** Continue with the next iteration until maximum generation is reached

Step 6 above is important as a better dimension value is assigned to optimal particle P^o from two randomly chosen particles. This technique provides an interesting philosophy of vertical optimization which is very useful for many real world optimization problems. For better performance in regards to convergence, the two particles in step 1 of Taguchi method are randomly chosen from pBest vector. As of such, the Taguchi method will therefore improve the dimensions value among pBest, an effective way leading to faster convergence and better accuracy. We employ the similar methodology in this work for better performance.

IMPLEMENTS

For DOPF problem including FSWGs, there are T dispatches by N_{gen}-1 conventional generating units and many wind farms (for simplicity, one wind farm is adopted, many wind farms may be analogized). A particle array of control variable vectors is:

$$P = \begin{pmatrix} P_1^1 & P_1^2 & \dots & P_1^t & \dots & P_1^T \\ P_2^1 & P_2^2 & \dots & P_2^t & \dots & P_2^T \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ P_n^1 & P_n^2 & \dots & P_n^t & \dots & P_n^T \\ Q_c^1 & Q_c^2 & \dots & Q_c^t & \dots & Q_c^T \end{pmatrix} \quad (13)$$

$$P = 1, 2, \dots, g \quad (14)$$

where, P is particle vector, g is the number of population particles, P_n^t is the real power output of nth generating unit at the tth interval, Q_c^t is the number of group switched capacitors in wind farm at the tth interval:

- **Step 1. Initialization:** For the complete g population particles, the candidate solution of each individual particle is randomly initialized within the feasible range in such a way that it should satisfy the constraint given by Eq. 1
- **Step 2. Power flow and fitness calculation:** Through the power flow calculation including wind farms, the state variables, power loss and real power output of the slack bus generator corresponding to the current control variables have been able to get. The real power output of the slack bus generator will be set to the limit if it violates the limit. After handling overlimit of the real power output of the slack bus generator, the system power balance constraints as in Eq. 5 must meet, otherwise adding Eq. 5 as penalty terms to

the objective function to form a generalized objective function. In this study, Eq. 8 is used as the fitness or evaluation function. This is a generalized fitness function used to evaluate the fitness of the candidate solution of each particle individual. Also, record the particle's position with the global best fitness as gBest, record the current position of each particle as its current pBest. Set the iteration count k = 0

- **Step 3. TPSO method:** Solve DOPF in FSWGs integrated power system using TPSO method
- **Step 4. Update velocity and position:** If k = k_{max} iteration stop; otherwise, update w^k, set k = k+1 and update velocity and position by Eq. 11-12
- **Step 5. Update pBest and gBest:** After power flow and fitness calculation, update pBest and gBest. Go to step 3

NUMERICAL RESULTS

To verify the effectiveness and efficiency of the adopted TPSO for DOPF problems including FSWGs, IEEE 39-bus power system is used as the test systems. The procedure has been implemented in MATLAB 7.0 programming language and numerical tests are carried on a Pentium 4 2.4 G computer. The wind farm including 60 wind generators with the same type, the rating power of which reaches 36 MW, is connected to the system at the bus 14. For the single FSWG, its rating voltage is 690 V, the excitation reactance x_m is 2.2059Ω, the sum of the stator reactance x₁ and rotor reactance x₂ are 0.1998Ω. For simplifying the analysis, the load size is considered invariable in the planning horizon. The planning horizon is divided into 12 intervals and every interval is 0.5 h. The wind generators' outputs are shown in Table 2. IEEE 39-bus system data are given by Zimmierman *et al.* (2009) and Victoire and Jeyakumar (2005). The parameters of the conventional generating units are shown in Table 3 and 4.

The maximum capacity of the reactive power compensation is 50 Mvar which constitutes of 50 set of group switched capacitors with per unit 1 Mvar. So, the upper limit value of the absorbed reactive power of wind farm bus is set to 1 Mvar in the study.

To demonstrate the superiority of the proposed approach for DOPF problems, simulation results have been compared with PSO method. Owing to the randomness in intelligent algorithms, two algorithms are executed 20 times when applied to the test system.

Table 2: The wind farm data in different periods

Stage	1	2	3	4	5	6	7	8	9	10	11	12
P _{wav} ^t (MW)	18	36	0	30	5	15	20	0	16	25	28	35

Table 3: The parameters of conventional generating units

Generator	a_i (\$/h)	b_i (\$/MWh)	c_i (\$/MW ² h)	D_i (MW/h)	U_i (MW/h)	P_i^0 (MW)	e_i (\$/h)	f_i (rad/MW)
G ₃₀	0.2	0.3	0.01	80	80	250	450	0.041
G ₃₁	0.2	0.3	0.01	80	80	572.9	600	0.036
G ₃₂	0.2	0.3	0.01	80	80	650	320	0.028
G ₃₃	0.2	0.3	0.01	50	50	632	260	0.025
G ₃₄	0.2	0.3	0.01	50	50	508	280	0.063
G ₃₅	0.2	0.3	0.01	50	50	650	310	0.048
G ₃₆	0.2	0.3	0.01	30	30	560	300	0.086
G ₃₇	0.2	0.3	0.01	30	30	540	340	0.082
G ₃₈	0.2	0.3	0.006	30	30	830	270	0.098
G ₃₉	0.2	0.3	0.006	30	30	1000	380	0.094

Table 4: The parameters and limits of conventional generating units

Generator	$Q_{i,max}$ -----(MVar)-----	$Q_{i,min}$	$V_{i,max}$ ----- (p.u.) -----	$V_{i,min}$	$P_{i,max}$ ----- (MW) -----	$P_{i,min}$
G ₃₀	9999	-9999	1.06	0.94	350	0
G ₃₁	9999	-9999	1.06	0.94	1145.55	0
G ₃₂	9999	-9999	1.06	0.94	750	0
G ₃₃	9999	-9999	1.06	0.94	732	0
G ₃₄	9999	-9999	1.06	0.94	608	0
G ₃₅	9999	-9999	1.06	0.94	750	0
G ₃₆	9999	-9999	1.06	0.94	660	0
G ₃₇	9999	-9999	1.06	0.94	640	0
G ₃₈	9999	-9999	1.06	0.94	930	0
G ₃₉	9999	-9999	1.06	0.94	1100	0

Table 5: Best solution obtained using TPSO method

Stage	1	2	3	4	5	6	7	8	9	10	11	12
P ₀₃₀ (MW)	254.6259	261.3006	271.1606	295.9102	306.0734	312.3618	296.9151	285.9975	309.4865	296.1915	315.6365	291.3349
P ₀₃₁ (MW)	559.2365	531.3081	568.5249	545.1256	519.7483	530.2933	515.4133	553.5996	524.6091	532.1246	525.8483	520.9931
P ₀₃₂ (MW)	668.2747	660.1902	669.429	666.1562	683.0371	674.0885	670.8214	669.3482	664.8019	679.2295	652.4174	650.9281
P ₀₃₃ (MW)	624.7418	622.3442	618.1788	622.6472	612.6111	598.5672	615.8161	628.3665	615.8157	598.8962	608.047	615.0311
P ₀₃₄ (MW)	505.9318	510.1707	494.5836	476.6036	498.0992	500.6739	504.9776	504.4464	499.6127	501.1885	517.2139	504.8214
P ₀₃₅ (MW)	644.5102	639.1326	641.2766	642.7508	645.9379	637.7323	651.984	638.8936	642.3732	647.3645	644.1022	646.5649
P ₀₃₆ (MW)	552.68	548.3306	551.9813	543.1459	545.1848	543.12	544.8427	539.1924	544.098	552.2539	549.8075	556.0777
P ₀₃₇ (MW)	539.4013	550.0231	549.7494	541.4442	547.0807	549.7038	551.9638	538.6662	551.0466	540.1841	536.2784	537.4133
P ₀₃₈ (MW)	829.0912	829.9197	831.1547	835.6637	837.3789	835.9434	840.3376	843.4381	830.3669	828.3651	831.6296	837.282
P ₀₃₉ (MW)	996.2026	1004.104	996.5893	992.6754	992.807	994.9221	980.6059	991.1308	994.358	991.3215	983.5499	997.3393
Q _c (Mvar)	26	31	40	37	28	27	18	37	44	27	30	26
V ₁₇ (p.u.)	1.015104	1.016188	1.01671	1.016895	1.015366	1.015533	1.014253	1.016396	1.017697	1.015472	1.016132	1.015694

Total production cost : 457039.8613 \$/h

Table 6: Best solution obtained using PSO method

Stage	1	2	3	4	5	6	7	8	9	10	11	12
P ₀₃₀ (MW)	268.7658	258.7243	281.3649	307.231	280.7399	307.6498	308.809	327.7243	311.9705	302.3885	298.0621	321.1744
P ₀₃₁ (MW)	543.5181	520.3182	501.6371	512.0913	526.223	514.3683	491.4921	492.4213	471.4391	471.5744	446.4973	453.5774
P ₀₃₂ (MW)	655.2717	655.4833	676.057	649.5119	679.2872	646.5458	642.8666	653.0834	662.003	686.4636	660.7034	656.3618
P ₀₃₃ (MW)	628.5717	631.9016	631.8742	616.6386	608.946	611.5581	610.0922	622.3505	621.5318	598.481	612.8068	597.3353
P ₀₃₄ (MW)	512.5234	517.1026	502.8713	495.2671	511.7135	504.3758	523.9461	522.5639	539.4942	516.5016	539.7707	546.5191
P ₀₃₅ (MW)	640.7321	631.1899	644.0308	643.3097	645.4117	655.2186	649.4062	634.1671	642.2616	648.7982	646.3999	624.3714
P ₀₃₆ (MW)	558.9129	569.1066	584.1066	574.3904	582.8516	575.9711	589.8552	577.5405	575.2699	585.2815	586.5097	581.1593
P ₀₃₇ (MW)	541.3456	542.7161	545.3616	538.9627	536.5182	536.8894	536.254	536.2534	541.7313	541.5959	544.8671	537.4367
P ₀₃₈ (MW)	833.1345	833.7514	831.73	832.8218	820.6794	835.6794	835.3575	833.5808	829.5042	818.7918	833.7918	829.3838
P ₀₃₉ (MW)	992.4712	997.2836	995.0622	992.6591	995.4322	990.0726	985.9602	994.0155	983.1012	998.1012	997.2805	1010.924
Q _c (Mvar)	31	28	24	26	24	26	25	19	38	24	33	29
V ₁₇ (p.u.)	1.015872	1.015701	1.014582	1.015632	1.014675	1.015392	1.015248	1.014384	1.016675	1.014988	1.016178	1.016173

Total production cost: 458852.7267 \$/h

For DOPF problem including FSWGs, Table 5 and 6 list the best control variables found by TPSO and PSO algorithm for IEEE 39-bus system, respectively. It is clearly shown that, by using TPSO, the total production cost savings of 1812.8654\$/h is obtained compared with PSO algorithm. Hence, it is justified that TPSO approach gives the exact minimum dispatch solution. From Table 7, the best, worst and average cost values are 457039.8613 \$/h, 458177.5291 \$/h, 457518.5863 \$/h and 458852.7267 \$/h, 460137.6551 \$/h, 459394.2806 \$/h, respectively with TPSO and PSO after

20 independent trials. From the results, the superiority of TPSO strategies over PSO can be noticed. The difference between the best and worst solutions are 1137.6678 \$/h with TPSO. At the same time, the difference between the best and worst solutions is 1284.9284\$/h with PSO. Moreover, the best and worst solutions obtained by TPSO are very close to the average value, which proves that TPSO is more robust and consistent. In conclusion, it is clearly shown that TPSO is the most accurate and gives the exact minimum dispatch solution.

Table 7: Comparison of best, worst and average cost values in the IEEE 39-bus system

Algorithms	Best	Worst	Average
TPSO	457039.8613	458177.5291	457518.5863
PSO	458852.7267	460137.6551	459394.2806

Table 8: Average execution time comparison in the IEEE 39-bus system

Methods	Average execution time (sec)	
	T_{fix}	T_{low}
TPSO	4235.83	—
PSO	4317.79	2140.45

The average execution time taken to complete the fixed number of iterations (T_{fix}) and the average execution time taken to converge into the lower solution range (T_{low}) for 20 trials are shown in Table 8 for IEEE 39-bus system.

For IEEE 39-bus system, PSO takes an average execution time of 4235.83 sec to complete 150 iterations. PSO converges faster than TPSO by reason of the small sub-memplex generation number of TPSO. In comparison to PSO, TPSO has additional components, i.e., the Taguchi method. This extra burdens increase the execution time of TPSO. TPSO takes 4317.79 sec more than PSO to complete 150 iterations. Nevertheless, TPSO takes only 2140.45 sec to converge into the lower solution range (457039-457107 \$/h), PSO are not able to converge into the lower solution range.

CONCLUSION

Considering the valve-point effect and ramp rate limits of conventional generators, as well as the discreteness of the reactive compensation devices including FSWGs, DOPF model, which takes the all conventional units cost minimum as the objective function and takes the whole time and the inherent relations of different stages into account in wind power integrated system, is established. The P-bus model of FSWG bus is adopted in power flow calculation in this study. A novel TPSO is proposed for solving the established DOPF model and the detailed methods of the algorithm are given. For DOPF problem including FSWGs, according to the principle of the nearest reactive power compensation, the required reactive power of wind farm doesn't absorb from the system as much as possible, but mainly from the wind farm reactive compensation devices. IEEE 39-bus system is used to illustrate the effectiveness of the proposed method compared with those obtained from PSO algorithm. The test results show that TPSO gives the exacter minimum dispatch solution and achieves better economic efficiency compared with PSO algorithm.

ACKNOWLEDGMENTS

The authors would like to thank anonymous reviewers. This work is supported by the Science Research Program of Education Bureau of Hubei Province under Grant No. D20092906.

REFERENCES

Cornejo-Mazon, M., M. Eugenia Jaramillo-Flores, T. Lourdes Villa and H.S. Humberto, 2008. Optimization of biomass production by *Ustilago maydis* in submerged culture using taguchi experimental design. *Biotechnology*, 7: 818-821.

He, D., F. Wang and Z.Z. Mao, 2008. A hybrid genetic algorithm approach based on differential evolution for economic dispatch with valve-point effect. *Int. J. Electrical Power Energy Syst.*, 30: 31-38.

Heng, X.C., Z. Qin, X.H. Wang and L.P. Shao, 2006. Research on learning bayesian networks by particle swarm optimization. *Inform. Technol. J.*, 5: 540-545.

Kennedy, J. and R. Eberhart, 1995. Particle swarm optimization. *Proceedings of the IEEE International Conference on Neural Networks*, Nov. 27-Dec. 01, Perth, WA, Australia, pp: 1942-1948.

Lai-Jun, W., S. Xiao-Ling and Z. Shi, 2009. New evolutionary algorithm applying to a type of facility location problem. *Inform. Technol. J.*, 8: 605-609.

Li, H. and Z. Chen, 2008. Overview of different wind generator systems and their comparisons. *IET Renew. Power Gener.*, 2: 123-138.

Nava-Arenas, D., A. Jimenez-Aparicio and H. Hernandez-Sanchez, 2008. Optimization of germination conditions of blue corn (*Zea mays* L.) by taguchi orthogonal array methodology. *Asian J. Plant Sci.*, 7: 682-686.

Pothiya, S., I. Ngamroo and W. Kongprawechnon, 2008. Application of multiple tabu search algorithm to solve dynamic economic dispatch considering generator constraints. *Energ Convers. Manage.*, 49: 506-516.

Ross, P.J., 1989. *Taguchi Techniques for Quality Engineering*. McGraw-Hill Publication, New York.

Roy, K.R., 1990. *A Primer on the Taguchi Method*. Van Nostrand Reinholds, New York.

Shi, Y. and R. Eberhart, 1998. A modified particle swarm optimizer. *Proceedings of the IEEE Congress on Evolutionary Computation*, May 4-9, Piscataway, New Jersey, pp: 69-73.

Sutha, S. and N. Kamaraj, 2008. Particle swarm optimization applications to static security enhancement using multi type facts devices. *J. Artif. Intell.*, 1: 34-43.

- Taguchi, G., S. Chowdhury and S. Taguchi, 2000. Robust Engineering. McGraw-Hill, New York.
- Taguchi, G., S. Chowdhury and Y. Wu, 2005. Taguchi's Quality Engineering Handbook. 1st Edn., John Wiley and Sons, Inc., Hoboken, New Jersey.
- Uturbey, W. and A.S. Costa, 2007. Dynamic optimal power flow approach to account for consumer response in short term hydrothermal coordination studies. IET Gener. Transm. Distrib., 1: 414-421.
- Victoire, T.A.A. and A.E. Jeyakumar, 2005. A modified hybrid EP-SQP approach for dynamic dispatch with valve-point effect. Int. J. Electr. Power Energy Syst., 27: 594-601.
- Yang, L.Y., J.Y. Zhang and W.J. Wang, 2009. Selecting and combining classifiers simultaneously with particle swarm optimization. Inform. Technol. J., 8: 241-245.
- Yuan, X., L. Wang, Y.C. Zhang and Y.B. Yuan, 2009. A hybrid differential evolution method for dynamic economic dispatch with valve-point effects. Expert Syst. Applied, 36: 4042-4048.
- Zakermoshfegh, M., S.A.A.S. Neyshabouri and C. Lucas, 2008. Automatic calibration of lumped conceptual rainfall-runoff model using particle swarm optimization. J. Applied Sci., 8: 3703-3708.
- Zimmerman, R.D., C.E.M. Sanchez and D. Gan, 2009. Matpower a MATLAB power system simulation package. <http://www.pserc.cornell.edu/matpower>.