J. Indian Inst. Sci., Sept.–Oct. 2006, **86**, 439–455

Swarm intelligence approach to the solution of optimal power flow

K. S. SWARUP
Department of Electrical Engineering, Indian Institute of Technology Madras, Chennai 600 036, India.
email: swarup@ee.iitm.ac.in

Received on May 9, 2006; Revised on October 16, 2006.

Abstract

An efficient and reliable evolutionary-based meta-heuristic approach, termed as swarm intelligence, is presented for the solution of optimal power flow with both continuous and discrete variables. The continuous control variables are unit-active power outputs and generator-bus voltage magnitudes, while the discrete variables are transformer tap settings and switchable shunt devices. Particle swarm optimization, a new evolutionary computation technique based on swarm intelligence, is illustrated for two case studies of IEEE-30 bus system and 3-area IEEE RTS-96 system. Both normal and contingency states are considered for the optimal power flow solution. The feasibility of the proposed method is compared with a simple genetic algorithm. The algorithm is computationally faster, in terms of the number of load flows executed, and provides better results than other heuristic techniques.

Keywords: Power system optimization, optimal power flow, particle swarm optimization, genetic algorithm.

1. Introduction

Optimal power flow (OPF) problem is a static constrained nonlinear optimization problem, the solution of which determines the optimal setting for control variables in a power network respecting various constraints [1, 2]. OPF has been widely used in power system operation and planning [3]. Many techniques such as linear programming [4–6], nonlinear programming [7–10], and quadratic programming [11] have been applied to the solution of OPF problem. These methods rely on convexity to obtain the global optimum solution, and as such are forced to simplify the relationships to ensure convexity. However, the OPF problem is in general non-convex and, as a result, many local minima may exist. Classical optimization methods are highly sensitive to starting points and frequently converge to local optimum solution or diverge altogether.

Linear programming methods are fast and reliable but the main disadvantage is associated with piecewise linear cost approximation. Nonlinear programming methods have a problem of convergence and algorithmic complexity. Newton-based algorithm [2] has a problem in handling large number of inequality constraints. This method has a drawback—the convergence characteristics are sensitive to the initial conditions. Interior point (IP) methods [12–14] convert the inequality constraints to equality by the addition of slack variables. In IP, if the step size is not chosen properly, the sublinear problem may have a solution that is infeasible in the original nonlinear domain [13]. These methods are usually confined to specific cases of the OPF and do not offer great freedom in objective functions or the type of constraints that may be used. It is therefore important to develop new, more general and reliable algorithms.

Heuristic algorithms such as enhanced genetic algorithm [15], improved GA [16], refined GA [17], gradient projection method [18], and evolutionary programming [19] have been recently proposed for the OPF problem. Recent research has identified some deficiencies in GA performance. Recently, a new evolutionary computation technique, called particle swarm optimization (PSO), has been proposed and introduced by Angeline [20], and Kennedy [21]. This technique combines social psychology principles in socio-cognition human agents and evolutionary computations. PSO has been motivated by the behavior of organisms such as fish schooling and bird flocking. The particle swarm is an algorithm for finding optimal regions of complex search spaces through the interaction of individuals in a population of particles. Particle swarm adaptation has been shown to successfully optimize a wide range of continuous functions [22–25]. PSO algorithm has been used successfully for economic dispatch [26]. PSOs are found have better convergence properties than genetic algorithms for a particular domain of unit commitment [27].

The PSO algorithm is based on a metaphor of social interaction. It searches a space by adjusting the trajectories of individual vectors, called ‘particles’, as they are conceptualized as moving as points in multidimensional space. The individual particles are drawn stochastically towards the positions of their own previous best performances and the best previous performance of their neighbors. Since its inception, two notable improvements have been introduced on the initial PSO which attempt to strike a balance between two conditions. The first one introduced by Shi and Eberhart [23] uses an extra ‘inertia weight’ term which is used to scale down the velocity of each particle and this term is typically decreased linearly throughout a run. The second version introduced by Clerc and Kennedy [28] involves a ‘constriction factor’ in which the entire right side of the formula is weighted by a coefficient. Their generalized particle swarm model allows an infinite number of ways in which the balance between exploration and convergence can be controlled. The simplest of these is called Type-1 PSO.

This paper proposes an application of Type-1 PSO to OPF with both continuous and discrete control variables. The continuous controllable system quantities are generator MW, controlled voltage magnitude and switchable shunt device while the discrete ones are transformer tapping. The objective is to minimize the fuel cost by optimizing the control variables within their limits, so that no violation on other quantities (e.g. transmission-circuit loading, load bus voltage magnitude, generator MVAR) occurs in either the normal or outage case system operating conditions. The proposed approach has been tested on IEEE-30 bus system [7] for both normal and contingent case.

2. Optimal power flow problem formulation

The optimal power flow problem is a nonlinear optimization problem with nonlinear objective function and nonlinear constraints. The solution methods to optimal power flow using conventional methods are the widely used Newton method, gradient methods and interior point methods. Handling of a large number of different types of constraints is a limitation with these methods [1–14]. Successful application of evolutionary programming methods like genetic algorithms, evolutionary computation, PSO, etc. has reduced the limitations of the conventional methods to a great extent [15–19]

The OPF problem requires the solution of nonlinear equations, describing optimal and/or secure operation of a power system. The general OPF problem can be expressed as:

Minimize $$F(x, u), \tag{1}$$

subject to $$g(x, u) = 0, \tag{2}$$

$$h(x, u) \le 0, \tag{3}$$

where

$$x^T = [\delta\ V_L^T], \tag{4}$$

$$u^T = [P_G^T V_G^T t^T Q_{SH}^T]. \tag{5}$$

The load flow equations are:

$$0 = P_i - V_i \sum_{j=1}^{N_B} V_j (G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij}), \tag{6}$$

$$0 = Q_i - V_i \sum_{j=1}^{N_B} V_j (G_{ij} \sin\theta_{ij} - B_{ij} \cos\theta_{ij}). \tag{7}$$

where N_B is the number of buses.

The fuel cost function is given as:

$$F(x,u) \text{ is } \sum_{i=1}^{N_G} (a_i P_{Gi}{}^2 + b_i P_{Gi} + c), \tag{8}$$

where g(x, u) is a set of nonlinear equality constraints (power flow equations), and $h(x, u)$, a set of nonlinear inequality constraints of a vector argument x and u.

The generator fuel cost curve is usually considered as quadratic ($F = (a + bP + CP^2)$). In fact, it is a polynomial of higher order with sine (sin) or exponential (e) term, but is reduced to quadratic. The incremental fuel cost (IFC) is linear and this helps is determining the optimal solution (IFC = $dF/dP = \lambda$). In the Indian power scenario, there is an important need to reduce the power losses at the distribution level due to different voltage profiles. APDRP (accelerated power development reforms program) is one such effort made by the Indian power sector to reduce power losses by raising the distribution-level voltage from 440 volts to 11 kV. Real power loss minimization is different from real power optimization, but is strongly related to the context of reactive power optimization.

Vector x consists of dependent variables and vector u of control variables. The variables $h(x, u)$ constitute a set of a system operating constraints that include:

The state variable vector x consists of the following:

(a) Branch flow limits $|S_k| \leq S_k^{\max}$ $k = 1 \ldots nl$

(b) Voltage at load buses $V_{Lk}^{\min} \leq V_{Lk} \leq V_{Lk}^{\max}$ $k = 1 \ldots N_L$

(c) Generator MVAR $Q_{Gk}^{\min} \leq Q_{Gk} \leq Q_{Gk}^{\max}$ $k = 1 \ldots N_G$

(d) Slack bus MW $P_G^{\min} \leq P_G \leq P_G^{\max}$

The control variable vector u consists of the following:

(a) Generator MW except slack MW $P_{Gk}^{\min} \leq P_{Gk} \leq P_{Gk}^{\max}$

(b) Generator bus voltage $V_{Lk}^{\min} \leq V_{Lk} \leq V_{Lk}^{\max}$ $k = 1 \ldots N_G$

(c) Transformer tap setting $t_k^{\min} \leq t_k \leq t_k^{\max}$ $k = 1 \ldots ntran$

The transformer taps are discrete with a change step of 0.0125 pu.

(d) Bus shunt capacitor $b_{SCk}^{\min} \leq b_{SCk} \leq b_{SCk}^{\max}$ $k = 1 \ldots N_C$.

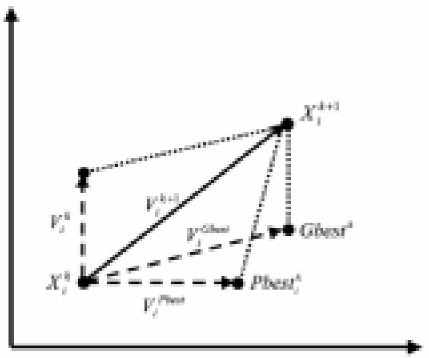
On load transformer taps (OLTC) are widely used in practice for (i) voltage control, and for (ii) reactive power control and dispatch. This is the usual procedure for real power loss minimization. The proposed method discusses the general procedure for loss minimization using control variables (generator voltage set-point, transformer tap, and reactive power control). The present problem does not address the situation in a deregulated environment, where tap changing may not be used from an ISO perspective for real-power OPF. This is not within the scope of the present work. The transformer tap is practically discrete, but is usually considered as continuous in the optimization problem for ease of solution. Handling discrete variables is difficult in the conventional method (NLP of optimization problem); hence the swarm intelligence method is effectively used to handle mixed variables (discrete and continuous) to obtain an optimal solution.

3. PSO

3.1. *Overview*

PSO is a population-based optimization method first proposed by Eberhart and colleagues [22, 23]. Some of the attractive features of PSO include the ease of implementation and the fact that no gradient information is required. It can be used to solve a wide array of different optimization problems. Like evolutionary algorithms, PSO technique conducts search using a population of particles, corresponding to individuals. Each particle represents a candidate solution to the problem at hand. In a PSO system, particles change their positions by flying around in a multidimensional search space until computational limitations are exceeded.

The PSO technique is an evolutionary computation technique, but it differs from other well-known evolutionary computation algorithms such as the genetic algorithms. Although a population is used for searching the search space, there are no operators inspired by the



X^k – current position,
X^{k+1} – modified position,
V^k – current velocity,
V^{k+1} – modified velocity
V^{pbest} – velocity based on pbest
V^{gbest} – velocity based on gbest

FIG. 1. Concept of modification of a searching point by PSO.

human DNA procedures applied on the population. Instead, in PSO, the population dynamics simulates a 'bird flock's' behaviour, where social sharing of information takes place and individuals can profit from the discoveries and previous experience of all the other companions during the search for food. Thus, each companion, *called particle*, in the population, which is called *swarm*, is assumed to 'fly' over the search space in order to find promising regions of the landscape. For example, in the minimization case, such regions possess lower function values than other, visited previously. In this context, each particle is treated as a point in a D-dimensional space, which adjusts its own 'flying' according to its flying experience as well as the flying experience of other particles (companions). In PSO, a particle is defined as a moving point in hyperspace. For each particle, at the current time step, a record is kept of the position, velocity, and the best position found in the search space so far.

3.2. *Algorithm*

The origins of PSO are best described as sociologically inspired, since the original algorithm was based on the sociological behavior associated with bird flocking [21]. The algorithm maintains a population of particles, where each particle represents a potential solution to an optimization problem. Let S be the size of the swarm, each particle i can be represented as an object with several characteristics. These characteristics are assigned the following symbols.

X_i: The current position of the particle;
V_i: The current velocity of the particle;
Y_i: The personal best position of the particle.

Figure 1 shows the concept of modification of a searching point by PSO. A population of particles is initialized with random positions and X_i velocities and V_i a function F_T* is evaluated, using the particle's positional coordinates as input values. Positions and velocities are adjusted and the function evaluated with the new coordinates at each time step. When a particle discovers a pattern that is better than any it has found previously, it stores the coordinates in a vector Y_i. The difference between the best point found by a particular agent and the individual's current positions is stochastically added to the current velocity, causing the trajectory to oscillate around that point. Further, each particle is defined within the context of a topological neighborhood comprising itself and some other particles in the population. The stochastically weighted difference between the neighborhood's best position G_j and the individual's current position is also added to its velocity, adjusting it for the next time step. These adjustments to the particle's movement through the space cause it to search around the two best positions.

Particle (X): It is a candidate solution represented by an *m*-dimensional vector, where *m* is the number of optimized parameters. At time *t*, the *i*th particle $X_i(t)$ can be described as $X_i(t) = [X_{i1}(t), X_{i2}(2), \ldots, X_{in}(t)]$, where *X*s are the optimized parameters and $X_{ik}(t)$ is the position of the *i*th particle with respect to the *k*th dimension; i.e. the value of the *k*th optimized parameter in the *i*th candidate solution.

Population, pop(t): It is a set of *n* particles at time *t*, i.e. $\text{pop}(t) = [X_1(t), X_2(t) \ldots X_n(t)]$.

Swarm: It is an apparently disorganized population of moving particles that tend to cluster together while each particle seems to be moving in a random direction [22].

Particle velocity: V(t): It is the velocity of the moving particles represented by an *m*-dimensional vector. At time *t*, the *i*th particle velocity $V_i(t)$ can be described as $V_i(t) = [V_{i1}(t), V_{i2}(2), \ldots, V_{in}(t)]$, where $V_{ik}(t)$ are the velocity components of the *i*th particle with respect to the *k*th dimension. The velocity update step is specified separately for each dimension $j \in 1 \ldots n$, so that V_{ij} denotes the *j*th dimension of the velocity vector associated with the *i*th particle. The velocity update equation is then

$$V_{ij}(t+1) = \chi \begin{pmatrix} V_{ij}(t) + \varphi_1 rand(0,\chi)(Y_{ij}(t) - X_{ij}(t)) + \\ \varphi_2 rand(0,\chi)(G_j(t) - X_{ij}(t)) \end{pmatrix}. \tag{9}$$

From the definition of the velocity of the equation it is clear that φ_2 regulates the maximum step service in the direction of the global best particle, and φ_1, the step size in the direction of the personal best position of the particle. The value of V_{ij} is clamped to the range $[-V_{i\max}, V_{\max}]$ to reduce the likelihood that the particle might leave the search space. The position of each particle is updated using the new velocity vector for that particle, so that

$$X_i(t+1) = X_i(t) + V_i(t+1). \tag{10}$$

Personal best (Pbest)*:* The Pbest position associated with the particle *i* is the best position that the particle has visited (a previous value of X_i) yielding the highest fitness value for that particle. For a minimization task, a position yielding the smaller function value is regarded as having fitness. The symbol F_T^* will be used to denote the objective function that is being minimized. The update equation for the Pbest position is presented in eqn (11).

$$Y_i(t+1) = \begin{cases} Y_i(t) & \text{if} \quad F_T^*(X_i(t+1)) \geq F_T^*(Y_i(t)) \\ X_i(t+1) & \text{if} \quad F_T^*(X_i(t+1)) < F_T^*(Y_i(t)) \end{cases} \tag{11}$$

Global best (Gbest): The Gbest offers a faster rate of convergence at the expense of robustness. It maintains only a single 'best solution', called the Gbest particle, across the entire particle in the swarm. This particle acts as an attractor, pulling all the particles towards it. Eventually all particles will converge to this position, so if it is not updated regularly, the swarm may converge prematurely.

Control parameters (φ_1 *and* φ_2): There are two important facts to consider when setting φ_1 and φ_2. The first fact is that the relation between the two values decides the point of attraction, which is given by:

$$\frac{\varphi_1 Y_i(t) + \varphi_2 G(t)}{\varphi_1 + \varphi_2}. \tag{12}$$

If $\varphi_1 \gg \varphi_2$, particle i will be much more attracted to the best found position found by itself, $Y_i(t)$, rather than the best position found by the neighborhood, $G(t)$, and vice versa if $\varphi_1 \ll \varphi_2$. In the other extreme case, where $\varphi_1 = 0$, the particle's own cognitive learning part is 0, and the whole swarm is attracted to one single point only, namely $G(t)$. Essentially, the swarm now turns into one big hill-climber. If $\varphi_1 = \varphi_2$, each particle will be attracted to the average $Y_i(t)$ and $G(t)$. Since φ_1 expresses how much the particle trusts its own past experience, it is called the *cognitive* parameter, and since φ_2 expresses how much it trusts the swarm, it is called the *social parameter.* In this implementation the control parameters are equal, i.e. $\varphi_1 = \varphi_2$. Thus, setting the control variables high enables the swarm to react rapidly to changes in the search, whereas if they are set low, the particles will react slowly and move in waves of huge magnitude and low frequencies.

Constriction factor (χ)*:* This factor may help in sure convergence. Low values of χ facilitate rapid convergence and little exploration and high values gives slow convergence and much exploration. Mathematician Maurice Clerc, who has proposed the constriction factor [26], has studied the particle swarm system by means of second-order differential equations. In doing so, it is possible to determine under which conditions the swarm will converge. In the constriction model we can set χ as a function of φ_1 and φ_2, so that convergence is ensured even without $V_{\max}$. An additional parameter k, which controls the convergence speed of the particles to the point of attraction, is added instead of χ. The supposed advantage of this shift from χ to k is that k can control swarm behavior more clearly and reliably: If k is close to 0, we get fast convergence (almost hill-climbing behavior), and if k is near 1 we get the slowest possible convergence with a high degree of exploration, which is desired for strongly multimodal problems. The constriction factor in the velocity update equation is represented by χ.

$$\chi = \frac{2\kappa}{|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}\,|}, \tag{13}$$

and

$$\varphi = \varphi_1 + \varphi_2 \quad \varphi > 4. \tag{14}$$

Let $\varphi_1 = \varphi_2 = 2.1$; substituting $\varphi = \varphi_1 + \varphi_2 = 4.2$.

Stopping criteria: These are the conditions under which the search process will terminate. In this study, the search will terminate if one of the following criteria is satisfied: (a) the number of iterations since the last change of the best solution is greater than a pre-specified number or (b) the number of iterations reaches the maximum allowable number.

Implementation

1. Choose the population size and the number of generations (number of iterations).
2. Select the control variables of a system, as state variables (X_i).
3. Set the time counter $t = 0$ and generate randomly n particles, $\{X_i(0), i = 1 \ldots n\}$ where $X_{ik}(0)$ is generated by random-selecting a value with uniform probability over the kth optimized parameter search space $[X_{\min}, X_{\max}]$. Similarly, generate randomly initial velocities of all particles, $\{V_i(0), i = 1 \ldots n\}$, where $V_i(0) = [V_{i1}(0), V_{i2}(0), \ldots, V_{in}(0)]$. $V_{ik}(0)$ is generated by randomly selecting a value with uniform probability over the kth dimension $[-V_{k\max}, V_{k\max}]$. Execute load flow considering the unit generations for each particle except for the slack bus, to evaluate the system transmission loss, slack bus generation and violation for the line flow limits and voltages at each bus. Maximum velocity of a particular dimension is given by eqn (14)

$$V_{k\max} = \frac{(X_{k\max} - X_{k\min})}{Na}. \tag{15}$$

where Na is number of intervals.

4. Evaluate the fitness for each particle according to the objective function (including penalty functions). The fitness function includes the total generation cost F_T and the penalty functions. The penalty function used in implementation is quadratic. It act as a soft constraint. The constraint includes the line flow limits, generator MVAR limit and the bus voltages at each bus.

   Augmented cost $((F_T^*))$

$$F_T^* = F_T + K_1 \sum_{i=1}^{nl} (I_i - I_i^{\max})^2 + K_2 (P_{G1} - P_{G1}^{\lim})^2 + K_3 \sum_{i=1}^{NL} (V_{Li} - L_{Li}^{\lim})^2 + K_4 \sum_{i=1}^{N_G} (Q_{Gi} - Q_{Gi}^{\lim})^2. \tag{16}$$

5. Set *Gbest _counter* = 1.
6. For each particle, as its best position, say it as *Pbest* set and assign *Gbest* that corresponds to the $X_i(0) = [X_{i1}(0), X_{i2}(0), \ldots, X_{in}(0)]$ particle shown by *Gbest_counter* from *Pbest*.
7. Update the time counter $t = t + 1$.
8. Using the global best and the individual best of each particle, the ith particle velocity in the kth dimension is updated as per eqn (8). It is worth mentioning that the second term represents the cognitive part of PSO where the particle changes its velocity based on its own thinking and memory. The third term represents the social part of PSO where the particle changes its velocity based on the social-psychological adaptation of knowledge. If a particle violates the velocity limits, set its velocity equal to the limit.
9. Based on the updated velocities, each particle changes its position according to eqn (9). If a particle violates the position limits in any dimension, set its position at the proper limit.
10. Personal best updating according to eqn (10).
11. After the first iteration, the *Gbest_counter* updates itself according to the minimum value of the fitness function from the *Pbest* set.
12. When any stopping criteria are satisfied stop program. Else go to step 7.

Table I
System description of case studies for IEEE-30 and RTS-96 bus systems

Variables	IEEE-30 Bus system	IEEE RTS-96 1-Area system (Area-A)	3-Area system (Area-B)	5-Area system (Area-C)
Buses	30	24	73	121
Branches	41	38	120	198
Generators	6	33	99	165
Generator buses	6	11	33	55
Shunts	9	1	3	5
Tap-changing transformers	4	5	16	26
Control variables	24	49	150	250
Equality constraints	2	2	2	2
Inequality constraints	96	134	410	680

4. Simulation results

The suitability of the proposed method has been tested for IEEE-30 bus, and the widely used IEEE RTS (reliability test system). They are chosen as they are benchmark systems, have more control variables and provide results for comparison of the proposed method. The approach can be generalized and easily extended to large-scale systems.

In this section, the PSO solution of the OPF is evaluated using two test systems: (i) IEEE-30 bus system [7], and 2) IEEE (reliability test system) RTS-96 [27]. Ten runs have been performed for each case examined. The results, which follow, are the best solution over these ten runs. The results are compared with GA. The system description of case studies of IEEE-30 bus and RTS-96 bus system are given in Table I.

Case study 1*: IEEE 30-bus system*

The EEE-30 bus system consists of six generators, four transformers, 41 lines, and nine shunt capacitors. In PSO and GA solution for OPF, the total control variables are 24: five unit active power outputs, six generator bus voltage magnitudes, four transformer tap settings, and nine bus shunt admittances. All generator active power, and generator bus voltages are continuous and transformer tap-setting and shunts are discrete variables. The limits of variables for the IEEE-30 bus system are given below.

Two different studies have been carried out on the IEEE-30 bus system.

No.	Description	Units	Variable type	Lower limits	Upper limits
1.	Voltage PQ-bus	pu	Continuous	0.95	1.05
2.	Voltage PV-bus	pu	Continuous	0.90	1.10
3.	Transformer taps	pu	Discrete Step size: 0.0125 (17 steps)	0.90	1.10
4.	Shunt capacitor	pu	Discrete Step Size: 0.010 (6 Steps)	0.0	0.05

Table II
GA and PSO parameters for best results of optimal power flow for IEEE-30 bus system

Sl no.	Genetic algorithm (GA)		Particle swarm optimization (PSO)	
	Parameters	Values	Parameters	Values
1.	Population	40	Population	20
2.	String size	155	φ_1	2.1
3.	Probability of crossover	0.800	φ_2	2.1
4.	Probability of mutation	0.005	κ	1
5.	Generations	150	Generations	150
6.	Number of load flows (Population × generations)	6000 (40 × 150)	Number of load flows (Population × generations)	3000 (20 × 150)

Case study 1*:* All control variables are continuous. Both normal network and contingency case with congestion were carried out.

Case study 2*:* All generator active power, and generator bus voltages are continuous and transformer tap-setting and shunts are discrete variables. Normal network and contingency case with congestion were studied. The congestion was created due to the outage of line 6–28 and the reduction in line limit from 0.32 to 0.12 pu of line (8–28).

The PSO and GA parameters used for the optimal power flow solution are given in Table II. The gene length for unit power output is 12 bits and for generator voltage magnitudes eight bits. They are both treated as continuous controls. The transformer tap value is encoded using eight bits. The bus shunt admittances can take six discrete values; each one is encoded using three bits.

Two sets of 10 runs were performed; the first GA with basic GA operators and the second is type 1 PSO (constriction factor approach). Table III shows the optimal setting of the generator bus voltages and generator active power for both PSO and GA. Figure 2 shows the variation of the total cost for normal (case 1) and with line 6–28 outage (case 2) for an IEEE-30 bus system. It can be observed that the total cost for the line outage is more than that for the normal case, which is as expected.

Table IV shows the optimal control variables obtained for the optimal power flow of the IEEE-30 bus system. The total fuel cost corresponding to the power generation is also pro-

Table III
Optimal active power generation and generator bus voltages for IEEE-30 bus system

Unit no.	Bus no.	Generator bus voltages and unit real power control			
		Particle swam optimization (PSO)		Genetic algorithm (GA)	
		Voltages [pu]	Unit real power [MW]	Voltages [pu]	Unit real power [MW]
1.	1	1.0850	175.44	1.0647	176.19
2.	2	1.0674	49.95	1.0447	48.99
3.	5	1.0322	20.91	1.0082	20.99
4.	8	1.0467	22.46	1.0147	21.95
5.	11	1.0837	11.68	1.0294	12.38
6.	13	1.0530	12.00	1.0759	12.20

Table IV
Control variables for cases 1 and 2 for the IEEE-30 bus system

Sl no.	I. Generator voltages			II. Power generation			III. Transformer taps			IV. Shunt capacitors		
	Gen voltage	Case 1 (pu)	Case 2 (pu)	PG	Case 1 (MW)	Case 2 (MW)	Transf Tap	Case 1 (pu)	Case 2 (pu)	Shunt Cap	Case 1 (pu)	Case 2 (pu)
1.	$\|VG_1\|$	1.0850	1.0518	PG_1	1.7515	1.5870	$Xtrf_{6-9}$	1.000	0.925	SC_{10}	0.0500	0.0500
2.	$\|VG_2\|$	1.0671	1.0188	PG_2	0.5020	0.5053	$Xtrf_{6-10}$	1.000	0.975	SC_{12}	0.0172	0.0138
3.	$\|VG_5\|$	1.0317	1.0015	PG_5	0.2151	0.2200	$Xtrf_{4-12}$	0.975	0.925	SC_{15}	0.0500	0.0000
4.	$\|VG_8\|$	1.0434	1.0033	PG_8	0.1950	0.1724	$Xtrf_{28-27}$	0.975	1025	SC_{17}	0.0322	0.0206
5.	$\|VG_{11}\|$	1.0816	1.0793	PG_{11}	0.1400	0.2125				SC_{20}	0.0349	0.0206
6.	$\|VG_{13}\|$	1.0471	1.0653	PG_{13}	0.1200	0.2277				SC_{21}	0.0450	0.0249
7.										SC_{23}	0.0450	0.0253
8.										SC_{24}	0.0104	0.0271
9.										SC_{29}	0.0071	0.0071
	Total fuel cost		$\sum_{i=1}^{6} F_i(P_i)(\$/\mathrm{h})$		800.7399 (Case 1)	812.3125 (Case 2)						

Case 1: Normal operation; Case 2: Contingent case with congestion.

vided. Table V shows the comparison of the cost of generation for the IEEE-30 bus system for the above cases with other available methods. Table VI shows the optimal control variables of shunt capacitors and transformer taps obtained from the optimal power flow for the IEEE-30 bus system. It can be observed from Table II that the number of load flows required for obtaining a best solution using PSO (3000) is half of that required by GA (6000).

Figure 3 shows the convergence of PSO and GA for the optimal power flow problem. The operating costs of the best solution in the normal operation achieved by the PSO and GA are, respectively, \$800.9194 and \$801.6738 per hour. It can be observed that the convergence of PSO is faster than that of GA while obtaining a better solution in lesser computational time.

Case study 2: IEEE RTS-96

The three-area IEEE RTS-96 [27] is a 73-bus, 120-branch system. It consists of three areas connected through five tie lines. The system description is given in Table I. Area-A unit

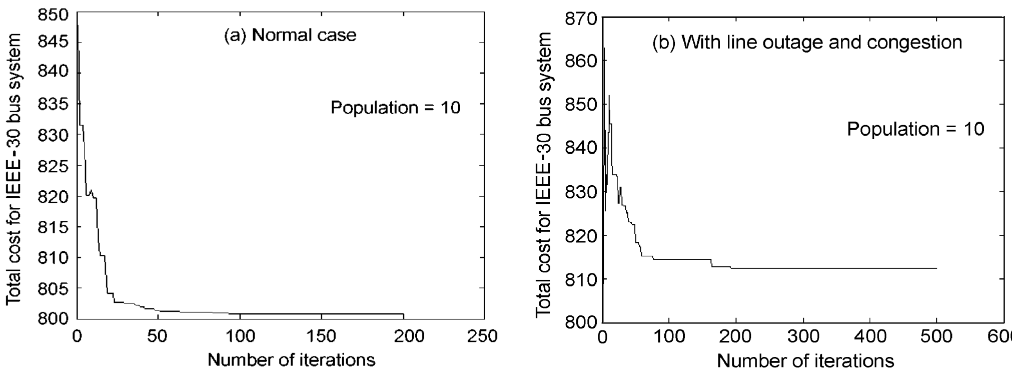


FIG. 2. Total cost of IEEE-30 bus system for population size of 10 for (a) normal and (b) line 6–28 outage and congestion.

Table V
Comparison of the cost of generation for IEEE-30 bus system for various cases

Sl no.	Techniques used	Cost ($/h) (Case 1)	(Case 2)
1.	Gradient projection method	804.583	–
2.	Simple genetic algorithm	802.06	–
3.	Improved genetic algorithm	800.81	812.33
4.	Enhanced genetic algorithm	802.40	–
5.	Particle swarm optimization	800.74	812.31

Case 1: Normal operation; Case 2: Contingent case with congestion.

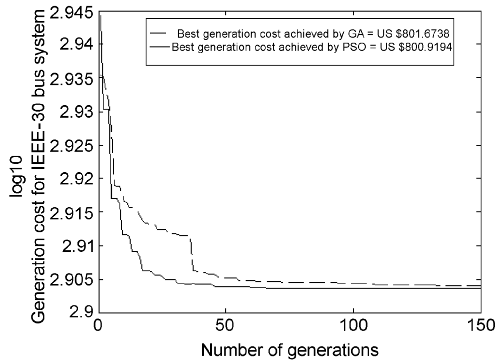


FIG. 3. Convergence curve for GA and PSO for IEEE-30 bus system.

Table VI
Control variables (shunt capacitor compensation and transformer tap) for IEEE-30 bus system

Sl no.	I. Transformer taps			II. Shunt capacitors		
	Transf tap	GA	PSO	Shunt cap	GA	PSO
1.	$Xtrf_{6-9}$	11	11	SC_{10}	6	1
2.	$Xtrf_{6-10}$	2	9	SC_{12}	3	1
3.	$Xtrf_{4-12}$	10	6	SC_{15}	5	5
4.	$Xtrf_{28-27}$	10	6	SC_{17}	5	6
5.				SC_{20}	5	1
6.				SC_{21}	6	5
7.				SC_{23}	6	1
8.				SC_{24}	6	5
9.				SC_{29}	5	1

Shunt capacitor compensation (0th Step = 0.0 MVAR and 4th Step = 5.0 MVAR). Transformer tap (0th Step = 0.9 pu and 17th Step = 1.1 pu).

Table VII
Fuel cost for IEEE-3-area RTS-96

Fuel/ unit type	Coal/ steam	Oil/ steam	Oil/ CT	Hydro	Nuclear
Price [$/Gcal]	5.01	10.49	34.45	0.00	2.65

cost data is derived from the heat rate data provided in [29] and the fuel cost data is listed in Table VII. The value of water is zero, assuming excessive inflows. Fuel costs of Areas-B and -C are selected three times of those of Area-A to impose exports from Area-A to Areas-B and -C. A contingency case with tie lines 107–203 and 123–217 out of service, under 90% peak load conditions, is studied. To impose congestion the ratings of tie lines 113–215 and 121–325 are reduced by 50% (to 250 MVA).

This system has a total of 150 control variables as follows: 98 unit active power outputs, 33 generator-bus voltage magnitudes, 16 transformer tap settings, and three bus shunt admittances. All generator active power, and generator bus voltages are continuous and transformer tap setting and shunts are discrete variables. The limits of variables for the IEEE RTS-96 system are given below.

Sl no.	Description	Units	Variable type	Lower limits	Upper limits
1.	Voltage PQ-bus	pu	Continuous	0.95	1.05
2.	Voltage PV-bus	pu	Continuous	0.95	1.10
3.	Transformer taps	pu	Discrete Step size: 0.0125 (17 steps)	0.90	1.10
4.	Shunt reactor capacitor	MVAR	Discrete Step size: 50 MVAR (4 steps)	–150.0	50.0

Table VIII
GA and PSO parameters for best results of optimal power flow for RTS-96 system

Sl no.	Genetic algorithm (GA)		Particle swarm optimization (PSO)	
	Parameters	Values	Parameters	Values
1.	Population	80	Population	20
2.	String size	1529	φ_1	2.1
3.	Probability of crossover	0.800	φ_2	2.1
4.	Probability of mutation	0.005	κ	1
5.	Generations	300	Generations	150
6.	Number of load flows	24000 (80 × 300)	Number of load flows	3000 (20 × 150)

The GA and PSO parameters used for the OPF solution for the best case are given in Table VIII. Two further studies have been carried out for the IEEE RTS-96 system.

1. Normal network.
2. Contingent case with congestion on tie lines.
   2.1. Contingent case with congestion on tie lines.
   2.2. Contingency: Outage of line (107–203) and (123–217).
3. Congestion: Reduction in tie line limit by 50% (250 MVA) of line (113–215) and (121–325).
4. Load change: 90% of peak load.

First, the unconstrained schedule is obtained by ignoring branch flow limits. Branch flow limits are ignored by selecting the corresponding penalty weight to zero in eqn (11). The unconstrained schedule results in a 60.2 MVA overloading of tie line 121–325. The corresponding operating cost is $221086.15 per hour. Next, the constrained schedule is calculated by activating the branch flow constraints. Tie line 121–325 flow is now reduced to 245.09 MVA (below the 250 MVA line rating). The operating cost is increased to $221,962.08 per hour due to congestion. The same of the best solution in the normal operation achieved by the PSO and GA are $221,962.08 and $222,430.39 per hour, respectively. Table VIII shows that the number of load flows required for obtaining a best solution using PSO is half of that of GA.

5. Computational requirements

In GA and PSO, if a population of size PS is allowed to evolve for a total number of NG generations, the product PS • NG determines the required number of load flows required and hence the computational requirements of both the methods. It is widely recognized among GA practitioners that the required fitness evaluation for a particular GA implementation depends on problem difficulty, which, in turn, depends on two factors: (i) the chromosome length, and (ii) the shape and characteristics of the fitness landscape. Problems with smooth fitness landscapes are easy to solve with GA. If the global optimum is located at the bottom of a steep gorge of the fitness landscape, GA may require a large

Table IX
Optimal generator bus voltages for IEEE RTS-96 system

Sl no.	Bus no.	GA (V pu)	PSO (V pu)	Sl no.	Bus no.	GA (V pu)	PSO (V pu)	Sl no.	Bus no.	GA (V pu)	PSO (V pu)
1.	101	1.0325	1.0146	12.	201	0.9687	1.0378	23.	301	1.0211	1.0267
2.	102	1.0351	1.0141	13.	202	0.9698	1.0371	24.	302	1.0162	1.0268
3.	107	0.9694	1.0496	14.	207	1.0295	1.0100	25.	307	1.0627	1.0556
4.	113	0.9883	1.0344	15.	213	1.0567	1.0666	26.	313	1.0083	1.0359
5.	114	1.0242	1.0520	16.	214	1.0298	1.0145	27.	314	0.9983	1.0404
6.	115	0.9824	1.0045	17.	215	1.0581	1.0092	28.	315	1.0122	1.0194
7.	116	1.0025	1.0114	18.	216	1.0590	1.0110	29.	316	0.9985	1.0182
8.	118	1.0610	1.0309	19.	218	1.0705	1.0065	30.	318	1.0432	1.0431
9.	121	1.0456	1.0274	20.	221	1.0715	1.0148	31.	321	1.0533	1.0304
10.	122	1.0589	1.0437	21.	222	1.0866	1.0593	32.	322	1.0559	1.0152
11.	123	0.9871	1.0007	22.	223	1.0571	1.0493	33.	323	0.9650	1.0425

number of fitness evaluations to locate it. In PSO, the particle contains the solution and time depends on the fitness evaluation, velocity and state updating equation.

Table IX shows the optimal generator bus voltages obtained by GA and PSO for IEEE RTS-96 system. Table X shows control variables (shunt capacitor compensation and transformer tap) obtained by GA and PSO for IEEE RTS-96 bus system. For the assessment of PSO- and GA-based OPF computational requirements, an experiment was designed. Four 3-area systems of increasing size are created, based on IEEE RTS-96 [27] (1-, 3-, and 5-area configurations). The optimal GA population size is varying according to system size. The population size and the number of generations for the system are given in Table XI, which summarizes the results of ten test runs in all test systems. The last three rows report the average (over 10 runs) computational requirements of PSO and GA. The number of generations (NG) to arrive at a good quality OFF solution is reported in the second row. A good-quality OFF solution is one with fitness value within 0.2% of the fitness obtained after allowing PSO and GA to evolve for respective generations. Table XI shows that the difference of the best and worst solutions increases slightly and the execution time increases considerably as the system size increases.

6. Conclusions

This paper presents a PSO solution to the optimal power flow problem and is applied to small- and medium-sized power systems. The main advantage of PSO over other modern

Table X
Control variables (shunt capacitor compensation and transformer tap) for IEEE RTS-96 bus system

Sl no.	Shunt cap	GA	PSO	Transf tap	GA	PSO	Transf Tap	GA	PSO	Transf Tap	GA	PSO
1.	SC_{106}	3	2	$Xtrf_{103-124}$	12	10	$Xtrf_{203-224}$	7	2	$Xtrf_{303-324}$	7	6
2.	SC_{206}	3	2	$Xtrf_{109-111}$	11	10	$Xtrf_{209-211}$	7	14	$Xtrf_{309-311}$	16	15
3.	SC_{306}	2	3	$Xtrf_{109-112}$	11	16	$Xtrf_{209-212}$	14	8	$Xtrf_{309-312}$	12	14
4.				$Xtrf_{110-111}$	9	11	$Xtrf_{210-211}$	9	5	$Xtrf_{310-311}$	6	8
5.				$Xtrf_{110-112}$	11	8	$Xtrf_{210-212}$	3	6	$Xtrf_{310-312}$	11	14
6.										$Xtrf_{323-325}$	8	3

Shunt capacitor compensation (0^{th} Step = 0.0 MVAR and 4^{th} Step= –150.0 MVAR)
Transformer tap (0^{th} Step = 0.9 pu and 17^{th} Step = 1.1 pu)

Table XI
Computational requirements of PSO and GA for IEEE RTS-96 bus system

Sl no.	IEEE RTS-96 bus system configuration		1-Area system		3-Area system		5-Area system	
			PSO	GA	PSO	GA	PSO	GA
1.	Population size		20	40	20	840	40	150
2.	No of generations (NG)		100	100	300	300	500	500
3.	Number of load flows		2000	4000	24000	4000	20000	75000
4.	Operating	Best	35557	35603	221961	222430	302163	302542
5.	Cost	Worst	35646	35698	222531	223063	302941	303511
6.	[Units/h]	% Difference	0.25%	0.26%	0.26%	0.28%	0.26%	0.2632%

heuristics is modeling flexibility, sure and fast convergence, less computational time than other heuristic methods. And it can be easily coded to work on parallel computers.

The main disadvantage of PSO and GA is that they are heuristic algorithms, and they do not provide the guarantee of optimal solution for the OPF problem. Type-1 PSO, which is based on constriction factor approach, is useful for obtaining high-quality solution in a very less time compared to other classical methods and GA.

The future work in this area consists of the applicability of PSO and GA solutions to large-scale OFF problems of systems with several thousands of nodes, utilizing the strength of parallel computers.

References

1. H. W. Dommel, and W. F. Tinney, Optimal power flow solutions, *IEEE Trans. Power Apparatus Systems*, **87**, 1866–1876 (1968).
2. D. I. Sun, B. Ashley, B. Brewer, A. Hughes, and W. F Tinney, Optimal power flow by Newton approach, *IEEE Trans. Power Apparatus Systems*, **103**, 2864–2880 (1984).
3. J. A. Momoh, R. J. Koessler, M. S. Bond, B. Stott, D. Sun, A. Papalexopoulos, and P. Ristanovic, Challenges to optimal power flow, *IEEE Trans. Power Systems*, **12**, 444–455 (1997).
4. B. Stott, and E. Hobson, Power system security control calculation using linear programming, Parts I and II, *IEEE Trans. Power Apparatus Systems*, **97**, 1713–1731 (1978).
5. B. Stott, and J. L. Marinho, Linear programming for power system network security applications, *IEEE Trans. Power Apparatus Systems*, **98**, 837–848 (1979).
6. R. Mota-Palomino, and V. H. Quintana, A penalty function-linear programming method for solving power system constrained economic operation problems, *IEEE Trans. Power Apparatus Systems*, **103**, 1414–1442 (1984).
7. O. Alsac, and B. Stott, Optimal load flow with steady state security, *IEEE Trans. Power Apparatus Systems*, **93**, 745–751 (1974).
8. R. R. Shoults, and D. T. Sun, Optimal power flow based on P-Q decomposition, *IEEE Trans. Power Apparatus Systems*, **101**, 397–405 (1982).
9. M. H. Bottero, F. D. Galiana, and A. R. Fahmideh-Vojdani, Economic dispatch using the reduced hessian, *IEEE Trans. Power Apparatus Systems*, **101**, 3679–3688 (1982).
10. J. A. Momoh, A generalized quadratic-based model for optimal power flow, *Proc. IEEE Int. Conf. Systems Man and Cybernetics*, Cambridge, MA, USA, Vol. 1, pp. 261–271, 1989 (1989).

11. R. C. Burchett, H. H. Happ, and K. A. Wirgau, Large-scale optimal power flow, *IEEE Trans. Power Apparatus Systems*, **101**, 3722–3732 (1982).

12. G. L. Torres, and V. H. Quintana, An interior-point method for nonlinear optimal power flow using voltage rectangular coordinates, *IEEE Trans. Power Systems*, **13**, 1211–1218 (1998).

13. J. A. Momoh, and J. Z. Zhu, Improved interior point method for OPF problems, *IEEE Trans. Power Systems*, **14**, 1114–1120 (1999).

14. H. Wei, H. Sasaki, J. Kubokawa, and R. Yokoyama, An interior point nonlinear programming for optimal power flow problems with a novel data structure, *IEEE Trans. Power Systems*, **13**, 870–877 (1998).

15. A. G. Bakirtzis, P. N. Biskas, C. E. Zoumas, and V. Petridis, Optimal power flow by enhanced genetic algorithm, *IEEE Trans. Power Systems*, **17**, 229–236 (2002).

16. L. L. Lai, J. T. Ma, R. Yokoyama, and M. Zhao, Improved genetic algorithms for optimal power flow under both normal and contingent operation states, *Int. J. Electl Power Energy Systems*, **19**, 287–292 (1997).

17. S. R. Paranjothi, and K. Anburaja, Optimal power flow using refined genetic algorithm, *Electric Power Components Systems*, **30**, 1055–1063 (2002).

18. K. Y. Lee, and Y. M. Park, A unified approach to optimal real and reactive dispatch, *IEEE Trans. Power Apparatus Systems*, **104**, 1147–1153 (1985).

19. J. Yuryevich, and J. Kit Po Wong, Evolutionary programming based optimal power flow algorithm, *IEEE Trans. Power Systems*, **14**, 1245–1250 (1999).

20. P. J. Angeline, Using selection to improve particle swarm optimization, *Proc. 1998 Int. Conf. on Evolutionary Computation–The IEEE World Congress on Computational Intelligence*, Anchorage, May 1998, pp. 84–89 (1998).

21. J. Kennedy, The particle swarm: Social adaptation of knowledge, *Proc. Int. Conf. on Evolutionary Computation (ICEC'97)*, pp. 303–308 (1997).

22. J. Kennedy, and R. Eberhart, Particle swarm optimization, *Proc. IEEE Int. Conf. on Neural Network*, Vol. 4, pp. 1942–1948 (1995).

23. Y. Shi, and R. Eberhart, A modified particle swarm optimizer, *Proc. 1998 Int. Conf. on Evolutionary Computation–The IEEE World Congress on Computational Intelligence*, Anchorage, May 1998, pp. 69–73 (1998).

24. He Zhenya, Wei Chengjian, Yang Luxi, Gao Xiqi, Yao Susu, R. C. Eberhart, and Yuhui Shi, Extracting rules from fuzzy neural network by particle swarm optimization, *Proc. 1998 Int. Conf. on Evolutionary Computation–The IEEE World Congress on Computational Intelligence*, Anchorage, May 1998, pp. 74–77 (1998).

25. J. Kennedy, and W. M. Spears, Matching algorithms to problems: An experimental test of the particle swarm optimization and some genetic algorithms on the multimodal problem generator, *Proc. IEEE Int. Conf. on Evolutionary Computation–The IEEE World Congress on Computational Intelligence*, pp. 78–83 (1998).

26. K. S. Swarup, and P. Rohit Kumar, A new evolutionary computation technique for economic dispatch with security constraints, *Int. J. Electl Power Energy Systems*, **28**, 273–283 (2006).

27. K. S. Swarup, and S. Yamashiro, A genetic algorithm approach to generator unit commitment, *Int. J. Electl Power Energy Systems*, **25**, 679–687 (2003).

28. M. Clerc, and J. Kennedy, The particle swarm-explosion, stability, and convergence in a multidimensional complex space, *IEEE Trans. Evolutionary Computation*, **6**, 58–73 (2002).

29. C. Grigg, P. Wong, P. Albrecht, R. Allan, M. Bhavaraju, R. Billinton, Q. Chen, C. Fong, S. Haddad, S. Kuruganty, W. Li, R. Mukerji, D. Patton, N. Rau, D. Reppen, A. Schneider, M. Shahidehpour, and C. Singh, The IEEE reliability test system-1996. A report prepared by the reliability test system task force of the application of probability methods subcommittee, *IEEE Trans. Power Systems*, **14**, 1010–1020 (1999).

List of principle symbols used

P_D	=	Total system demand
P_L	=	Total transmission loss
F_T	=	Total fuel cost
F_T^*	=	Augmented fuel cost
$P_{Gi}^{\max}$	=	Maximum operating limit of the ith generator
$P_{Gi}^{\min}$	=	Minimum operating limit of the ith generator
a_i, b_i, c_i	=	Cost coefficient of the ith generator
$J_J^{\max}$	=	Maximum magnitude of current in jth line
K_1	=	Line loading penalty factor
K_2	=	Penalty factor for the slack generation
K_3	=	Penalty factor for bus voltages
rand	=	Random number between 0 and $\varphi/2$
φ_i	=	Weighting factor
φ	=	Social coefficient
χ	=	Constriction coefficient
$Y_{i,j}$	=	Individual particles best fitness yet encountered
G_j	=	Best value so far in the group.
$V_{i,j}^k$	=	Velocity of agent i at iteration k
$X_{i,j}^k$	=	Current position of agent i at iteration k
Na	=	Number of intervals
Q_{GI}	=	Reactive power generated by i generator
V_{Li}	=	Voltage at ith load bus