01 Sep 2008

Enhanced Particle Swarm Optimizer for Power System Applications

Yamille del Valle

M. Digman

A. Gray

J. Perkel

et. al. For a complete list of authors, see https://scholarsmine.mst.edu/ele_comeng_facwork/1594

Recommended Citation

Y. del Valle et al., "Enhanced Particle Swarm Optimizer for Power System Applications," *Proceedings of the 2008 IEEE Swarm Intelligence Symposium (2008, St. Louis, MO)*, Institute of Electrical and Electronics Engineers (IEEE), Sep 2008.
The definitive version is available at https://doi.org/10.1109/SIS.2008.4668333

Enhanced Particle Swarm Optimizer for Power System Applications

Y. del Valle, *Student Member, IEEE*, M. Digman, A. Gray, J. Perkel, *Student Member, IEEE*,
G. K. Venayagamoorthy, *Senior Member, IEEE*, and R. G. Harley, *Fellow, IEEE*

Abstract-- **Power system networks are complex systems that are highly nonlinear and non-stationary, and therefore, their performance is difficult to optimize using traditional optimization techniques. This paper presents an enhanced particle swarm optimizer for solving constrained optimization problems for power system applications, in particular, the optimal allocation of multiple STATCOM units. The study focuses on the capability of the algorithm to find feasible solutions in a highly restricted hyperspace. The performance of the enhanced particle swarm optimizer is compared with the classical particle swarm optimization (PSO) algorithm, genetic algorithm (GA) and bacterial foraging algorithm (BFA). Results show that the enhanced PSO is able to find feasible solutions faster and converge to feasible regions more often as compared with other algorithms. Additionally, the enhanced PSO is capable of finding the global optimum without getting trapped in local minima.**

Index Terms—**Flexible AC Transmission Systems (FACTS), Static VAR compensators, Particle Swarm Optimization, Genetic Algorithm, Bacterial Foraging Algorithm.**

I. INTRODUCTION

The optimization of power system performance can be accomplished by improving the voltage profile, increasing the power transmission capability, controlling power flow and others. Flexible AC Transmission System (FACTS) devices, such as the STATCOM, SVC and SSSC, can be used for these purposes due to their capability to achieve numerous control functions fast and accurately [1].

In order to obtain the maximum benefit for each FACTS device, the optimal location and size of each unit have to be carefully determined. This problem is particularly challenging since power system networks are complex systems that are highly nonlinear and non-stationary. Additionally, some of the decision variables can only take integer values and the constraints given by the desired system performance make it difficult to find feasible solutions.

Simple heuristic approaches are traditionally applied for determining the location and size of FACTS devices in small power systems. However, more scientific methods are required for larger power networks.

Traditional optimization techniques such as mixed integer linear and non-linear programming have been investigated to solve this problem. Unfortunately, difficulties arise due to multiple local minima and the overwhelming computational effort [2], [3].

Evolutionary Computation Techniques (ECT) have been recently employed to solve the optimal allocation of FACTS devices. Different algorithms such as Genetic Algorithms (GA) [2], [4], [5], Evolutionary Programming (EP) [6] and Particle Swarm Optimization (PSO) [7], [8], have been tested with promising results. Nevertheless, the canonical versions of these algorithms sometimes do not provide an efficient search mechanism to find feasible solutions fast and easily.

The purpose of this paper is to show the application of an enhanced particle swarm optimizer for power system applications, in particular, for the optimal location and sizing of multiple STATCOM units in a power system. The criterion used in finding the best solution is to improve, at minimum cost, the voltage profile of the system such that the voltage deviations at each bus do not exceed a predefined set value.

A statistical analysis is performed to show that the enhancement applied to the PSO algorithm allows the search process to be more efficient in finding feasible solutions and global minimum as compared with the canonical PSO version and other evolutionary computation techniques.

Section II briefly presents the fundamentals of each optimization method, while section III describes the characteristics of the optimization problem to be solved. Section IV presents simulation results, statistical analysis and comparison between methods. Finally section V states all concluding remarks.

II. OPTIMIZATION ALGORITHMS

A. Particle Swarm Optimization (PSO)

PSO uses the concept of swarm intelligence to obtain an optimal solution. It does so by searching the feasible space with several particles that each represents a possible solution to the problem. Their positions are updated using both the individual particle's knowledge as well as the combined knowledge of the entire swarm [9], [10].

In a real-number space, the PSO algorithm considers that each particle's position is defined by a vector $\vec{x}_i \in \mathcal{R}^n$. At

Y. del Valle, M. Digman, A. Gray, J. Perkel and R.G. Harley are with Department of Electrical and Computer engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (yamille.delvalle, mdigman, agray, jperkel and rharley @gatech.edu)

G. K. Venayagamoorthy is with Real-Time Power and Intelligent Systems Laboratory in the Department of Electrical and Computer Engineering at the Missouri University of Science and Technology, Rolla, MO 65409, USA (gkumar@ieee.org).

$$v_i(t) = \begin{bmatrix} r_1 \cdot v_{max}(1) & r_2 \cdot v_{max}(2) & r_3 \cdot v_{max}(3) & r_4 \cdot v_{max}(4) \end{bmatrix} \quad (4)$$

iteration t, the particle's position vector $\vec{x}_i(t)$, is determined by adding the previous position vector $\vec{x}_i(t-1)$ and the particle's velocity $\vec{v}_i(t)$, as shown in (1).

$$\vec{x}_i(t) = \vec{x}_i(t-1) + \vec{v}_i(t) \quad (1)$$

The velocity of the particle is determined by both the individual and group experiences:

$$v_i(t) = w_i \cdot v_i(t-1) + c_1 \cdot r_1 \cdot (p_i - x_i(t-1)) + ... \\ c_2 \cdot r_2 \cdot (p_g - x_i(t-1)) \quad (2)$$

where,

w_i is a positive number between 0 and 1,

c_1 *and* c_2 are two positive numbers called the cognitive and social acceleration constants, respectively,

r_1 *and* r_2 are random numbers with uniform distribution in the range of [0, 1],

p_i is the individual best position found by the particle,

p_g is the global best position found by the entire swarm.

In order to avoid the divergence of the swarm, the maximum allowable velocity for the particles is controlled by the parameter v_{max}. A different value of v_{max} can be defined for each dimension of the problem hyperspace.

In the case where integer variables are included in the optimization problem, the PSO algorithm can be reformulated by rounding off the particle's position to the nearest integer.

B. Enhanced Particle Swarm Optimization

For this particular application, the canonical PSO algorithm described in the previous section is enhanced by adding a basic logic to the particles to facilitate the search through the problem hyperspace.

The additional logic in each individual is defined by the following rules:

- If the corresponding particle's *pbest* and the *gbest* positions are both feasible solutions (i.e. solutions that satisfy all the constraints of the problem), then the velocity update is performed according to (2).

- If the particle has not found a feasible solution yet, then the velocity update equation is replaced by:

$$v_i(t) = w_i \cdot v_i(t-1) + c \cdot rand \cdot (p_g - x_i(t-1)) \quad (3)$$

where,

c is a single acceleration constant: $c = c_1 + c_2$,

rand is a random number with uniform distribution in the range of [0, 1].

In the case when the particle has not found a feasible solution by itself then it is better to rely on the social rather than the self knowledge, thus the particle follows the best particle in the swarm.

- If none of the particles have found a feasible solution (the *gbest* value and the *pbest* value are both infeasible) then

the velocity of the particles are updated using a random value of the maximum velocity as shown in (4).

where,

r_h is a random number with uniform distribution in the range of [0, 1].

$v_{max}(h)$ is the maximum velocity in the hth dimension of the problem hyperspace.

In this last case, when a feasible solution has not been found by any member of the swarm, the particles may get confused by following the directions represented by the *gbest* and *pbest* positions. As a consequence the particles move erratically in the problem hyperspace. Therefore, it is advantageous to assign random values to the velocity component so that only the limits represented by the maximum velocity are considered.

C. Genetic Algorithm (GA)

GA is an evolutionary computation technique that patterns itself after Charles Darwin's "survival of the fittest" concept. Each individual, in this case a chromosome, represents a possible solution to the problem. Through selection of parents, crossover between them, and mutation of the offspring, the population evolves and, after a number of generations, it approaches an optimal solution [11], [12].

After the population data is initialized randomly, the fitness of each chromosome is evaluated through the use of a fitness function. Higher ranking individuals have fitness values that are closer to the optimal fitness value and vice versa. After the fitness of each chromosome has been assessed, a subgroup of chromosomes is selected to become the parents for the next generation.

There are several ways to determine which members of the population will produce offspring. For this application, elitism and "roulette wheel" are used. Elitism copies a percentage of the highest ranking members of the current population into the new population, the rest of the chromosomes are then selected using the "roulette wheel", a method where higher-ranking individuals receive higher probabilities of being selected as parents.

Once the two parents are chosen, crossover between them will produce two offspring. After the crossover, there is a chance that any number of the offspring's genes may be mutated or altered. Each gene of the new chromosome is given the possibility of mutation, in other words, the genes are treated independently and this results in anywhere from zero to all genes being mutated.

The previous generation is replaced by the new generation and the entire process is repeated until a terminating condition is reached.

D. Bacterial Foraging Algorithm (BFA)

BFA is based on the movement patterns of *E. coli* in the intestines. Each individual, in this case a bacterium, represents a possible solution to the problem [13]-[15].

The algorithm considers four successive processes: Chemotaxis, Swarming, Reproduction and Elimination.

a) <u>Chemotaxis</u>: the bacteria move towards better nutrient concentrations avoiding noxious substances, and search for ways out of neutral media. The bacterium takes a tumble followed by a tumble or a tumble followed by a run. For N_c number of chemotactic steps the direction of movement after a tumble is given by (5).

$$\theta^i(j+1,k,l) = \theta(j,k,l) + C(i) \cdot \varphi(j) \qquad (5)$$

where,
$C(i)$ is the step size taken in direction of the tumble,
j is the index for the chemotactic step,
k is the index for the number of reproduction step,
l is the index for the number of elimination-dispersal event,
$\varphi(j)$ is the unit length random direction taken at each step.

If the fitness function value at $\theta^i(j+1, k, l)$ is better than the one corresponding to $\theta^i(j, k, l)$ then the bacterium takes another step of size $C(i)$ in that direction. This process continues until the number of repetitions per chemotactic cycle reaches a maximum of N_s.

b) <u>Swarming</u>: the bacteria in times of stress release attractants to signal bacteria to swarm together. Each bacterium also releases a repellant to signal others to be at a minimum distance from it. Thus all the bacteria will have a cell-to-cell attraction via attractant and cell-to-cell repulsion via repellant. The equation involved in the process is:

$$J_{cc}(\theta, P(j,k,l)) = \sum_{i=1}^{S} J_{cc}^i(\theta, \theta^i(j,k,l))$$

$$= \sum_{i=1}^{S}\left[-d_{attract} \cdot \exp\left(-w_{attract} \cdot \sum_{m=1}^{p}(\theta_m - \theta_m^i)^2\right)\right] + \qquad (6)$$

$$\sum_{i=1}^{S}\left[h_{repellant} \cdot \exp\left(-w_{repellant} \cdot \sum_{m=1}^{p}(\theta_m - \theta_m^i)^2\right)\right]$$

where,
$d_{attract}$ is the depth of the attractant,
$w_{attract}$ is a measure of the width of the attractant,
$h_{repellant}$ is the height of the repellant effect,
$w_{repellant}$ is a measure of the width of the repellant,
p is the number of parameters to be optimized,
S is the number of bacteria.

The bacteria moving towards better nutrient concentrations can be represented by:

$$J(i,j,k,l) + J_{cc}(\theta, P) \qquad (7)$$

where,
$J(i,j,k,l)$ is the fitness function.

c) <u>Reproduction</u>: after N_c chemotactic steps, the population of bacteria is allowed to reproduce. S_r $(S_r=S/2)$ bacteria having the worst fitness function value die and the remaining S_r are allowed to split into two thus keeping the population size constant.

d) <u>Elimination-Dispersal</u>: at an elimination-dispersal event,

each bacterium is eliminated with a probability of p_{ed}. This probability p_{ed} should not be large or it can lead to an exhaustive search.

III. PROBLEM DESCRIPTION

The problem to be addressed consists of finding the optimal placement (bus number) and power rating (MVA) of multiple STATCOM units in a power system, based on its steady state performance. Such a problem can be stated as a constrained optimization problem where the main objective is to find the best positions of the STATCOM units to minimize the bus voltage deviations throughout the power system, using a minimum (cost efficient) size for each STATCOM. In addition, other operating constraints are imposed such as keeping all voltage deviations within ±5% of the corresponding nominal values.

A 45 bus system, part of the Brazilian power network, is used for this study [8]. This Brazilian system (Fig.1) has two distinct load centers, suggesting that the voltage support should be done through two STATCOM units.

A. Objective Function

There are two goals that have to be accomplished: (i) to minimize the voltage deviations in the system and (ii) to have the minimum possible STATCOM sizes (minimum cost per each device). Thus, two metrics J_1 and J_2 are defined as in (8) and (9).

$$J_1 = \sqrt{\sum_{1}^{N}(V_k - 1)^2} \qquad (8)$$

where,
J_1 is a voltage deviation metric,
V_k is the p.u. value of the voltage at bus k,
N is the total number of buses,

$$J_2 = 100,000 \cdot \sum_{1}^{M} \eta_p \qquad (9)$$

where,
J_2 is a STATCOM size metric,
M is the number of STATCOM units to be allocated,
η_p is the size in MVA of STATCOM unit p.

The STATCOM size metric in (9) considers the cost of a typical STATCOM to be roughly 100,000 $/MVA [16].

The multi-objective optimization problem can now be defined using the weighted sum of both metrics J_1 and J_2 to create the overall objective function J shown in (10).

$$J = \omega_1 \cdot J_1 + \omega_2 \cdot J_2 \qquad (10)$$

The weight for each metric is adjusted to reflect the relative importance of that goal with respect to the other. In this case, it is decided to give equal importance to both metrics, giving values of $\omega_1 = 1$ and $\omega_2 = 2 \cdot 10^{-8}$, so that the two terms in the objective function are comparable in magnitude.
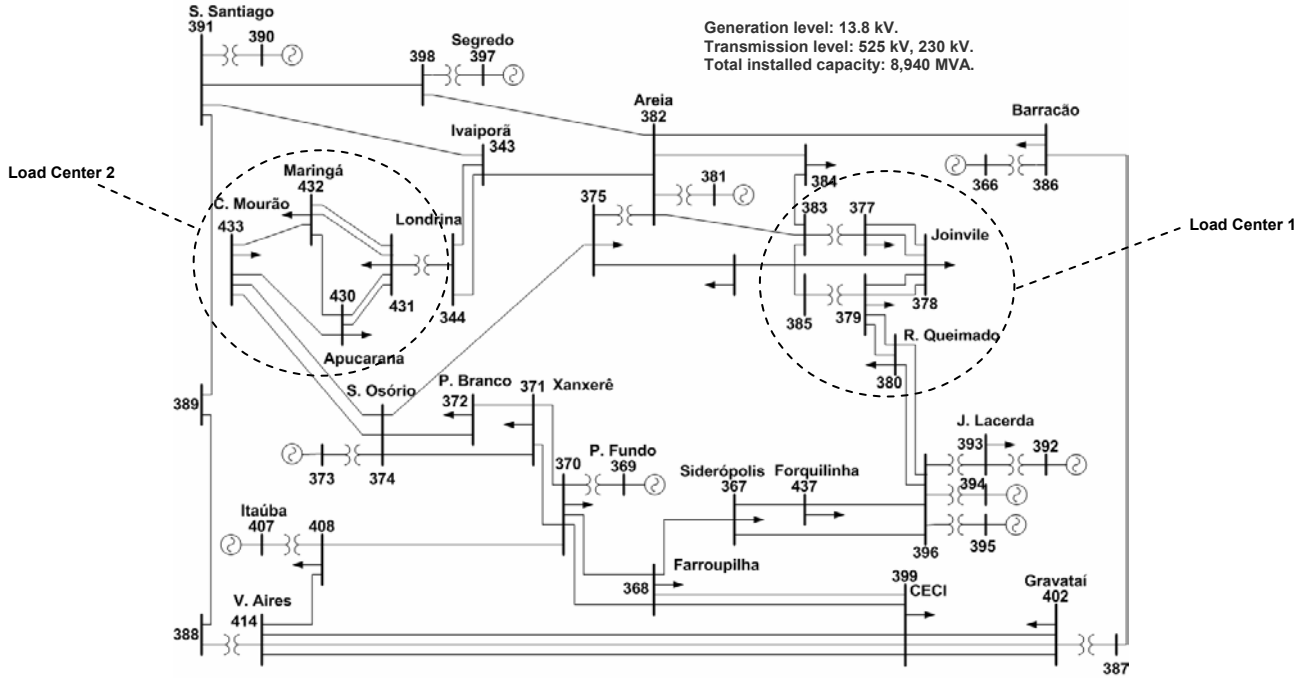
Fig. 1. One line diagram of the 45 bus 10 machine section of the Brazilian power system.

B. Decision Variables

The decision variables are the location of the STATCOM units and their sizes.These variables can be arranged in a vector as:

$$x_i = [\lambda_1 \quad \eta_1 \quad ... \quad \lambda_M \quad \eta_M], \qquad x_i \in Z^{2M} \qquad (11)$$

where,
λ_p is the location (bus number) of STATCOM unit p,
η_p is the size (MVA) of STATCOM unit p.

All components of the decision vector are integer numbers, thus $x_i \in Z^{2M}$.

C. Constraints

There are several constraints in this problem regarding the characteristics of the power system and the desired voltage profile. Each constraint represents a limit in the search space, which in this particular case corresponds to:

- Generator buses are omitted from the search process since they have voltage regulators to regulate the voltage.
- Bus numbers are limited to the range from 1 to N.
- Only one STATCOM unit can be connected at each bus.
- The size of each unit is between 0 and 250 MVA.
- The desired voltage profile requires N additional restrictions defined as:

$$0.95 \le V_k \le 1.05 , \quad k : 1 \to N \qquad (12)$$

Each solution that does not satisfy the above constraints is considered infeasible.

D. Problem Complexity

From the optimization point of view, the optimal allocation of STATCOM units in the power system is a very complex problem since it involves non-linear optimization and integer variables. Additionally, the voltage profile constraints limit the feasible regions to a small subset of the total problem hyperspace.

In order to determine the feasible regions and global optimal solution, an exhaustive search is performed. The exhaustive search is based on searching along the extreme points of the integral polytope defined by the search space constraints.

One power flow is required to evaluate each possible combination of values in the decision vector shown in (11), thus the exhaustive search requires 37,187,500 power flows to explore the total problem hyperspace.

Among the complete hyperspace, just 15 pairs of locations (λ_1, λ_2) can provide feasibility to allocate STATCOM units 1 and 2. This value corresponds to 2.52% of the 595 total possible combinations. Considering both locations and sizes, there are 414,750 combinations that meet all constraints, representing just 1.12% of the total hyperspace. Fig. 2 shows graphically the proportion of feasible regions in the total problem hyperspace.

Additionally, the result of the exhaustive search indicates that the global optimal solution is to place one STATCOM unit of 75 MVA at bus 378 and the second unit of 92 MVA at bus 433. After the devices are optimally placed, all voltage deviations are in the range of ±5% and the voltage deviation metric J_1 sees an improvement of 26.5 % from its original value of 0.2482.
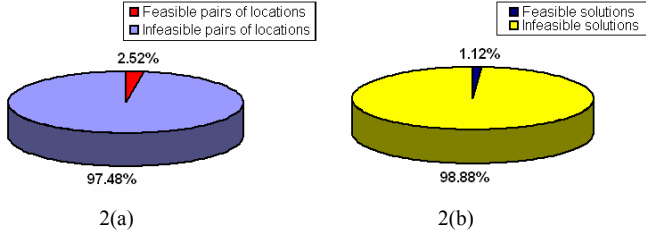
2(a)   2(b)

Fig. 2(a): Percentage of feasible locations over total possible combinations.
Fig. 2(b): Percentage of feasible solutions over total problem hyperspace.

IV. SIMULATION RESULTS

A. Convergence into feasible regions

In order to evaluate the performance of the optimization algorithms, 150 trials are carried out for each one of them. At each trial, the number of power flow evaluations is recorded until the first feasible solution is found. If no feasible solution is found, then the algorithms stops automatically when the number of power flow evaluations reaches a maximum number of 2000 power flows.

A performance indicator called Success Rate is calculated to determine the percentage of time that the algorithm is able to converge into feasible regions.

In order to use statistical parameters to evaluate the performance of each optimization technique, the Anderson-Darling normality test is first performed to measure of how likely the data (in this case the number of power flows) come from a normal distribution [17].

Performing the normality test is necessary in order to determine whether or not the means and standard deviations of these data sets are valid metrics to assess the differences between the techniques. In all cases, the Anderson-Darling p-values are less than 0.005 indicating that, with better than 99.5% certainty, the data are not normally distributed; therefore other statistical distribution has to be used.

Fig. 3 shows the histogram for each technique. Based on observation, Weibull distribution is considered appropriate to analyze the data. This distribution is used extensively to study extreme valued data, in this particular case, the number of power flows to the *first* feasible solution [18].
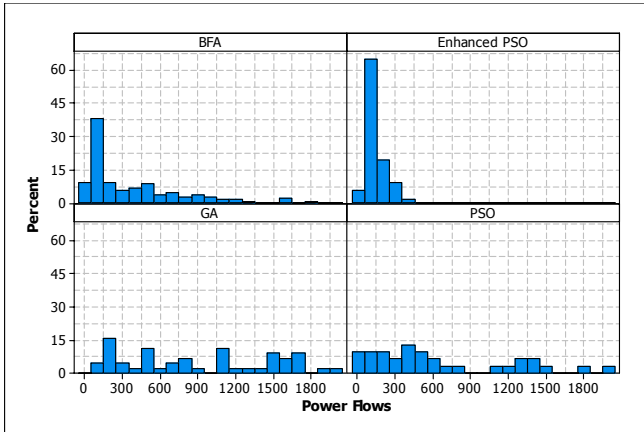

Fig. 3: Histogram for each algorithm

For each of the four datasets, a two-parameter Weibull distribution is fitted to each dataset using a standard statistical software package. In each case, the correlation is greater than 0.95, indicating that the choice of Weibull is suitable.

Fig. 4 shows the resulting probability plots for each technique and Table I shows the corresponding statistical parameters.
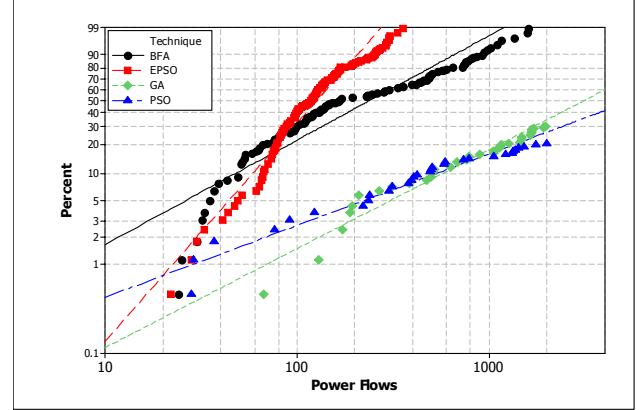

Fig. 4: Weibull plots all algorithms

TABLE I: STATISTICAL VALUES TWO-PARAMETER WEIBULL DISTRIBUTION

	Enhanced PSO	PSO	GA	BFA
Minimum PF	22	28	67	24
Maximum PF	379	1992	1972	1834
Success Rate	100	20.7	30	100
Scale (α)	147	8650	4329	326
Shape (β)	2.5	0.8	1.1	1.2

Table I indicates that, based on the ranges for the number of power flow evaluations, the proposed enhanced PSO is faster in finding feasible solutions as compared with all other algorithms. Moreover its Success Rates is 100% versus 20.7% for canonical PSO and 30% in the case of GA.

Additionally, the Weibull parameters, α and β, carry important physical meanings. The scale parameter, α, corresponds to the characteristic time (or number of power flows) to find the first feasible solution. This is defined as the number of power flows needed to obtain a feasible solution in 63.2% of the trials.

The shape parameter, β represents the slope produced by data when plotted on a Weibull plot (Fig. 4). More interestingly, the shape parameters provide insight into how the algorithms are able to seek out feasible solutions. Shape parameters greater than one imply increasing ability to locate feasible solutions. Enhanced PSO is the only algorithm that falls into this category. GA and BFA both have shape parameters that are slightly greater than one while PSO is slightly less than one. Clearly, the enhanced PSO offers the most efficient means of locating the feasible regions.

Fig. 4 allows one to read off the probability of obtaining a feasible solution in any number of power flows (or less) for each of the techniques. Equally, the probability may be specified and then the maximum number of power flows required may be determined.

Fig. 4 shows that the enhanced PSO distribution has the steepest slope followed by BFA. On inspection, it appears that

GA and PSO have very similar slopes indicating that their performances in locating their first feasible solutions are similar.

In addition, the resulting characteristic time to find a feasible solution was 147 and 326 power flows for enhanced PSO and BFA, respectively. The canonical PSO and GA were only able to find feasible solutions in at most 30% of the trials while the rest of the values are censored. This leads to characteristic times of 4329 and 8650 for GA and PSO, respectively.

B. Global Optimality

For further comparison of the performance of enhanced PSO and BFA algorithms, their capabilities for finding the global optimal solution are investigated.

Statistical values for the optimal solutions found are calculated over a set of 50 trials each. In this case, the Anderson-Darling normality test gives p-values grater 0.05, indicating that the data have Normal distribution for both cases.

Table II provides the additional indicators to evaluate the accuracy in finding the optimal solutions.

TABLE II: STATISTICAL ANALYSIS FOR OPTIMAL SOLUTIONS

Parameter	Enhanced PSO	BFA
Minimum objective function value (J)	0.51745	0.52441
Maximum objective function value (J)	0.68390	0.96422
Average objective function value (J)	0.58791	0.74765
Standard deviation objective function value (J)	0.04167	0.09654

The accuracy in finding the optimal solution is considerably better in the case of the enhanced PSO algorithm with a standard deviation of 0.0417 as compared to 0.0965 of BFA that is more than two times larger. In terms of the maximum and average values of the objective function value indicate a clear advantage of the enhanced PSO over BFA. Furthermore, the enhanced PSO algorithm finds the global optimum for this problem.

V. DISCUSSION

The characteristic times to first feasible solution obtained from the Weibull analysis indicate that the enhanced PSO offers substantial performance gains as compared to the canonical PSO. Furthermore, its performance is also superior to BFA and GA.

In addition, the examination of the shape parameters indicates that only enhanced PSO has increasing ability to locate feasible solutions.

In terms of Success Rate, both BFA and enhanced PSO were 100% successful in obtaining a feasible solution within the 2000 power flow horizon. Both canonical PSO and GA require substantially more power flows to locate feasible regions.

Considering global optimality, the enhanced PSO is able to find the global optimal solution while BFA finds a near optimal solution. Since both algorithms have stochastic components, the results over 50 trials are calculated indicating

that the enhanced PSO have smaller range for the objective function value, better average value and much higher accuracy as compared with BFA.

Overall, given the statistical analysis presented in this paper, the proposed enhanced PSO algorithm clearly outperforms the canonical PSO, GA and BFA in converging into feasible regions and finding the global optimum.

VI. APPENDIX

TABLE III: PSO PARAMETERS

Parameter	Optimal value
Number of particles	20
Inertia constant (w_i)	Linear decrease (0.9 to 0.1)
Individual acceleration constant (c_1)	2.5
Social acceleration constant (c_2)	1.5
V_{max} for bus location	9
V_{max} for STATCOM size	50
Maximum number of iterations	100

TABLE IV: GA PARAMETERS

Parameter	Optimal value
Number of chromosomes	20
Percentage of elite members	10%
Crossover probability	85%
Mutation probability	5%
Maximum number of generations	100

TABLE V: BFA PARAMETERS

Parameter	Optimal value
Number of bacteria	20
Number of chemotactic cycles (N_c)	30
Number of swim steps (N_s)	3
Number of reproductions (N_{re})	3
Number of elimination-dispersal loops (N_{ed})	2
Probability of elimination (P_{ed})	0.5
Maximum distance ($C(i)$)	4
Attraction coefficients $d_{attract}$ and $w_{attract}$	0.1
Repellent coefficient d_{repel} and w_{repel}	0.05

VII. REFERENCES

[1] N.G. Hingorani, and L. Gyugyi, *Understanding FACTS; Concepts and Technology of Flexible AC Transmission Systems*, IEEE Press, New York, 2000.

[2] H. Mori, and Y. Goto, "A parallel tabu search based method for determining optimal allocation of FACTS in power systems," Proc. of the International Conference on Power System Technology (PowerCon 2000), vol. 2, 2000, pp. 1077-1082.

[3] N. Yorino, E.E. El-Araby, H. Sasaki, and S. Harada, "A new formulation for FACTS allocation for security enhancement against voltage collapse," *IEEE Trans. on Power Systems*, vol. 18, no. 1, pp. 3-10, Feb. 2003.

[4] L.J. Cai, I. Erlich, and G. Stamtsis, "Optimal choice and allocation of FACTS devices in deregulated electricity market using genetic algorithms," Proc. of the IEEE PES Power Systems Conference and Exposition, vol. 1, 2004, pp.201-207.

[5] S. Gerbex, R. Cherkaoui, and A.J. Germond, "Optimal location of multi-type FACTS devices in a power system by means of genetic algorithms," *IEEE Trans. on Power Systems,* vol. 16, no. 3, pp. 537-544, Aug. 2001.

[6] W. Ongsakul, and P. Jirapong, "Optimal allocation of FACTS devices to enhance total transfer capability using evolutionary programming," Proc. of the IEEE International Symposium on Circuits and Systems (ISCAS 2005), vol. 5, 2005, pp. 4175-4178.

[7] M. Saravanan, S. Slochanal, P. Venkatesh, P. Abraham, "Application of PSO technique for optimal location of FACTS devices considering system loadability and cost of installation," in Proc. of the IEEE 7th

International Power Engineering Conference (IPEC) 2005. Vol. 2, Dec. 2005, pp. 716-721.

[8] Y. del Valle, J.C. Hernandez, G.K. Venayagamoorthy and R.G. Harley, "Multiple STATCOM Allocation and Sizing Using Particle Swarm Optimization", Proceedings of the IEEE Power Systems Conference and Exposition (PSCE) 2006, Atlanta, Georgia, USA, October 29 – November 30, 2006, pp. 1884 - 1891.

[9] R. Eberhart, Y.Shi and J. Kennedy, *Swarm intelligence*. San Francisco, CA: Morgan Kaufmann, 2001

[10] Y. del Valle, G. K. Venayagamoorthy, S. Mohagheghi,, J. C. Hernandez, and R. G. Harley, "Particle Swarm Optimization: Basic Concepts, Variants and Applications in Power System," *IEEE Transactions on Evolutionary Computation*, Vol. 12, No. 2, April 2008, pp: 171-195.

[11] L. Schmitt, "Theory of genetic algorithms," Theoretical Computer Science (259), 2001, pp. 1-61.

[12] M. Vose, *The simple genetic algorithm: foundations and theory*, MIT Press, Cambridge, MA, 1999.

[13] T.K. Das, G.K. Venayagamoorthy, "Bio-inspired Algorithms for the Design of Multiple Optimal Power System Stabilizers: SPPSO and BFA", Conference Record of the 2006 IEEE Industry Applications Conference, 41st IAS Annual Meeting. Vol. 2, 8-12 Oct. 2006, pp:635 – 641.

[14] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," Control System Magazine, IEEE, vol. 22, Issue 3, pp. 52-67, June 2002.

[15] W.J. Tang, Q.H. Wu, J.R. Saunders, "Bacterial Foraging Algorithm For Dynamic Environments", IEEE Congress on Evolutionary Computation, 2006. CEC 2006. 16-21 July 2006, pp: 1324 – 1330.

[16] S. Mohagheghi, "Adaptive critic designs based neurocontrollers for local and wide area control of a multimachine power system with a static compensator", PhD Thesis, Georgia Institute of Technology, 2006.

[17] G. E. P.Box, W. G. Hunter, J. S. Hunter, *Statistics for Experimenters*, New York, John Wiley & Sons, Inc, 1978.

[18] R. B. Abernethy, *The New Weibull Handbook*, North Palm Beach, FL, Robert B. Abernethy, 1993.