Performance Comparison of Genetic Algorithm, Differential Evolution and Particle Swarm Optimization Towards Benchmark Functions

Seng Poh Lim
Department of Computer Science,
Faculty of Computing, Universiti Teknologi Malaysia,
81310 Skudai, Johor, Malaysia.
lawrencess87@yahoo.com

Habibollah Haron
Department of Computer Science,
Faculty of Computing, Universiti Teknologi Malaysia,
81310 Skudai, Johor, Malaysia.
habib@utm.my

Abstract— **Genetic algorithm (GA), Differential Evolution (DE) and Particle Swarm Optimization (PSO) are always implemented to solve different kinds of complex optimization problems. Each method contains its own advantages and the performance varies based on different case studies. There are many Soft Computing (SC) methods which can generate different result for the same optimization problems. However, no exact result is produced because random function is usually applied in SC methods. The performance maybe is affected by the parameter setting or operations inside each method. Therefore, the motivation of this paper is to compare the performance of GA, DE and PSO by using the same parameters setting and optimization problems. The experiments can prove that although same parameters setting are applied, but different fitness and time can be obtained. Based on the result, GA was proven to perform better compared to DE and PSO in obtaining highest number of best minimum fitness and faster than both methods.**

Keywords-Genetic Algorithm; Differential Evolution; Particle Swarm Optimization; Optimization; Benchmark Functions; Performance

I. INTRODUCTION

Genetic algorithm (GA), Differential Evolution (DE) and Particle Swarm Optimization (PSO) are popular SC techniques that always implemented to solve different kinds of complex optimization problems. Each SC method contains its own advantages while using it and the performance varies based on different case studies. Sometimes the advantages contain drawbacks due to each method performs differently in each optimization problem. Some SC methods contain complicated operations, but still able to produce excellent results. While some methods are simple, but due to the parameters setting, hence poor results are produced.

Same optimization problems were tested by using different methods because the researchers want to test the performance of each method, such as to obtain the most optimum result or to test the speed of the method. If the result obtained is nearer to actual optimum or the speed is faster, hence that method is considered to contain better performance compared to the others. There are many methods in SC which can generate different result for the same optimization problems. Hence, the performance of SC methods is a good research field to be further explored. Therefore, the motivation of this paper is to compare the performance of GA, DE and PSO by using the same parameters setting and optimization problems (benchmark functions). The results can prove that although same parameters setting are applied, but different results (min, max and average) and speed (time) can be obtained, which can clearly shown the performance of each method.

This paper is organized as follows. In Section 2, theories and previous works on GA, DE and PSO are reviewed. Section 3 describes the flows of experiment for each method in testing the benchmark functions. Analysis and discussion on the experimental result are presented in Section 4. Conclusion, future work and direction are demonstrated in the last section.

II. LITERATURE REVIEW

This section demonstrates on theories and previous works that related to Genetic Algorithm, Differential Evolution and Particle Swarm Optimization.

A. Genetic Algorithm

Genetic Algorithm (GA) was introduced by John Holland in 1975, which is based on the principle of genetics and evolution [1,2]. It is suitable to be used as a tool in solving the optimization and searching problems. GA contains some operations with several techniques in performing the algorithm. Different result will be produced when different techniques are implemented because the way of the techniques perform is different [3]. The performance of GA will be raised [4] if proper operation techniques are implemented in GA, such as Rank selection technique is implemented along with Single point crossover technique.

A set of population is randomly generated in order to execute the GA operations. It contains several chromosomes which will be sub divided into several genes [2]. The chromosomes will be evaluated by using objective or fitness function. Fitness function is used to identify the strength of the chromosome towards the problems [5,6] and is formed based on objective function [7]. It will be continued with the remaining GA operations after the fitness evaluation process was finished. Generally, fitness values will be used by most of the selection operation techniques in selecting the parent. This

This work is financed by UTM Zamalah.

is due to the chromosome that holds excellent genetic materials should be carried forward to the next generation. Indirectly, the termination speed and result will be affected by selection operation.

New generation children are produced by merging the genetic materials of 2 parents in crossover operation while new genetic materials are added into the chromosome during mutation operation. In addition, both operations are highly based on the probability generated by the algorithm [2]. Maximum generation is usually used as the termination criteria to terminate the operations in producing the children. All the new offsprings (children) will be produced and evaluated by using the same fitness function [2] after all operations were finished. If the children are better compared to the parents based on fitness values, hence the old generation chromosomes will be replaced with the new generation chromosomes [2] in the replacement operation.

The shape and dielectric constant of the object are reconstructed by using GA in [8] work. GA is implemented to minimize the cost function in the inversion procedure. GA is used in [9] work to manage the data problems because it contains the characteristics such as random, iteration and evolution. While for [10] work, the parameters in the next iterations is modified with some rules by using GA to solve the optimizing process for inverse problems. The minimization problems between real and predicted measurement in [11] work is handled by using GA whereby the parameters are coded into GA and recovered through the global optimization.

B. Differential Evolution

Differential Evolution (DE) was introduced by Storn and Price in 1997 as a stochastic direct search method [12]. It is a population-based search strategy and method for mathematical optimization of multidimensional functions [5,13]. DE is better and more simple to be applied compared with other evolutionary algorithms [14] due to its simple implementation and fast convergence characteristics [15]. Generally, random function is used to generate the genes for each chromosome and fitness function is used to evaluate the fitness of chromosome. Trial and target vectors were involved in the operations in producing the next generation children.

Three indices will be randomly chosen to form the mutant vector during mutation operation. These indices must be different from the base index i. Weighted difference, F is used in DE mutation operation in controlling the amplification of the differential variation [12]. Trial vector will be formed after crossover operation was performed. Furthermore, one index will be randomly chosen during this operation is to ensure that at least one new mutant vector can be inserted into the new generation trial vector. Trial vector will be evaluated using the same fitness function and compared with target vector during selection operation. Better fitness trial vector will be selected to replace the old generation. Hence, new generation is produced. The operations will be continued until the termination criteria are reached.

DE is applied in [16] work by crossing the parameter from one parent with another parent by mutating the parameter. DE is implemented in [17] work in minimizing the discrepancy for measured and estimated data with respect to parameter for spline expansion. Besides that, the author also compared the performance of DE and PSO. The numerical result showed that DE is better than PSO in terms or speed and accuracy. DE is applied in [18] work to deal with the image reconstruction of Electrical Impedance Tomography (EIT). DE is also implemented in [15] work for the reconstruction of the front shape of different 2D conducting targets hidden behind walls and wall parameters estimation. In addition, DE is used in [19] work in optimizing the Bezier curve fitting.

C. Particle Swarm Optimization

Particle Swarm Optimization (PSO) was developed by Kennedy and Eberhart in 1995. It is a population-based stochastic optimization technique and global optimization approach [5] which is based on bird flocking and fish schooling [20]. Same as GA and DE, a set of population which contains particles are initialized based on the case studies. Each "particle" mentioned in PSO actually is the potential solution [21] which is analogous to a bird [22]. Each particle contains a position and velocity vector [23]. The particles are evolved by cooperation and competition among the individuals [21] in searching for the optimum result.

Fitness value of each particle in the population will be evaluated using objective function [22]. The particle that contains the best fitness will be selected as the global best particle, G_{BEST}. Each particle will remember its best position, P_{BEST} and velocity of each particle, V will be determined after the position of the particles has been updated. V_{max} is the maximum velocity which helps to keep the swarm under control [23] and will be fixed as half of the range [24]. In addition, c1 and c2 are the acceleration constants which normally assigned with the value of 2 [23,25]. After particles were updated, they will be evaluated using the same fitness function. P_{BEST} will be updated and G_{BEST} will be replaced by other particles that contain better fitness compared to it after updating process was performed. The updating process will be terminated after termination criteria were fulfilled.

PSO should be upgraded with some modern technologies [26]. Authors of [21] introduced inertia weight, w to improve the performance of PSO. It plays the role of balancing among the global and local search [21] and manages the affect of previous velocity on new velocity [27]. Social agent is presented by [28] as a new parameter in controlling the velocity [26]. The new parameter has improved the speed of convergence, which is better compared to the PSO with inertia weight [28]. PSO has been reported better than GA in finding the solution [22]. PSO is implemented in [29] work to obtain proper parameterization data points for Bézier surface. PSO is proposed by authors of [30] to explore the optimum fitting points and inspect the ability in reconstructing the objects.

III. FLOW OF EXPERIMENT

This paper makes a performance comparison between GA, DE and PSO by using benchmark functions as the case study. Hence, GA1 method from [3] is used as the GA method, while DE method from [12] and PSO method from [21] are used in this paper to perform the comparison. Chromosome with value encoding scheme is used for GA and DE while particle with value encoding is used for PSO in performing the experiment. The length of chromosome and particle is defined based on the dimension size of benchmark function, *d*. Each bit of chromosome and particle represents the parameter value, x_i in the benchmark function to be optimized.

A. Genetic Algorithm Method

The GA method is performed as algorithm below:

1) n set of chromosomes are randomly generated in the range.
2) Fitness of chromosomes is evaluated using the fitness function, f(x).
3) Selection: 2 set of chromosomes are randomly selected to be the parent in producing the children.
4) Crossover: 1 random number is generated. If this random number is less than the GA crossover probability, hence crossover operation is performed. Else, continue to next step.
5) Mutation: 1 random number is generated. If this random number is less than the GA mutation probability, hence mutation operation is performed. Else, continue to next step.
6) Fitness of children is evaluated by using f(x).
7) Replacement: Old generation parents are replaced by new generation children.
8) If termination criteria are not fulfilled, go to 3. Else, the algorithm is terminated.

B. Differential Evolution Method

The DE method is performed as algorithm below:

1) n set of chromosomes are randomly generated in the range.
2) Fitness of chromosomes is evaluated using the fitness function, f(x).
3) Mutation: 3 indices (r1, r2, r3) are randomly chosen from the population to form the mutant vector in producing the children.
4) Crossover: 1 random number is generated. If this random number is less than the DE crossover probability or equal to the randomly chosen index, hence crossover operation is performed and trial vector is formed. Else, continue to next step.
5) Fitness of trial vector is evaluated by using f(x).
6) Selection: Rank selection is used to choose and compare trial vector and target vector. Smallest fitness vector will be selected.
7) If termination criteria are not fulfilled, go to 3. Else, the algorithm is terminated.

C. Particle Swarm Optimization Method

The PSO method is performed as algorithm below:

1) n set of particles are randomly generated in the range.
2) Fitness of particles is evaluated using the fitness function, f(x).
3) P_{BEST} for each particle is determined, highest fitness particle is selected as G_{BEST}.
4) Velocity and position of particles are calculated and updated based on G_{BEST}.
5) Fitness of particles is evaluated by using f(x).
6) If termination criteria are not fulfilled, go to 3. Else, the algorithm is terminated.

D. Parameters Setting and Benchmark Functions

The parameters setting are assigned based on the suggestions from previous works. Standard parameters setting are applied because different methods are compared in this paper. Hence, the results can show and prove the performance of each method accurately in obtaining the minimum value. Table I shows the parameters setting involved in each method. GA parameters (Parameter 1 to 5) are referred from [3]. There are no standard parameters setting for DE. Due to mutation and crossover for DE is considered as one process, hence Parameter 6 is applied with the range of standard GA crossover probability. Parameter 7 is assigned with a smaller value because it will affect a lot on the mutant vector. Parameter 8 to 12 for PSO is referred from [23-26]. Although 0.8 is the standard value for Parameter 8, due to it will affect the previous velocity value, hence smaller value is assigned. Maximum generation is used as the termination criteria.

Table II shows the benchmark mathematical functions with the range of dataset which are used as the fitness function to test the performance of each method. Each benchmark function contains different dataset which is randomly generated and there are no standard dataset in testing benchmark functions. The benchmark functions are referred from our previous work in [3] and also [31-34]. Please refer to [3] and [31-34] for the details of each benchmark function.

Each benchmark function was tested 10 times and the minimum and maximum optimum values were recorded. This is to show that various optimum values can be produced using the same method. Average of 10 times testing for all functions are calculated and used as another performance measurement. In addition, average time in producing the result has been taken as another comparison in order to prove the speed of each method. Therefore, the method that able to produce the most minimum and average values with fastest speed is considered as the best method with excellent performance. All the experiments are conducted by using Dev C++.

IV. Experimental Results

There is no mathematical proof in testing the convergence of GA [7] and no exact result will be obtained from GA [3]. This statement is also valid for other SC methods due to unsure when the algorithm will be terminated and different result will be produced each time the experiment is performed. Hence, this experiment is only to test the performance of each method by comparing the minimum value and average time taken in producing result.

Based on the literature review and result in Table III, analysis and comparison of each method were tabulated in Table IV. The analysis shows that GA contains 5/10 best minimum fitness, 3/10 best average fitness and 10/10 faster than DE and PSO. It proved that GA is the best method which fulfills the aim of this paper by obtaining highest number of best minimum fitness and fastest method compared to DE and PSO. Although GA can perform well in the optimization problems, but it highly depends on the techniques implemented in the operations [3]. From the view of speed, DE contains 10/10 faster than PSO. However, based on the aspect of best minimum and average fitness, PSO contains

3/10 best minimum fitness and 6/10 best average fitness, which is better than DE. The result in Table III shows that PSO is only a bit slower than DE, which concludes that PSO contains better performance than DE.

TABLE I. PARAMETERS SETTING INVOLVED IN EACH METHOD

No	Parameter	Value
1.	Number of generation	2000
2.	Population size, *n*	40
3.	Number of dimension, *d*	30
4.	GA Crossover probability	0.7
5.	GA Mutation probability	0.01
6.	DE Crossover probability	0.8
7.	DE Differential weight	0.3
8.	PSO Inertia weight	0.7
9.	PSO Acceleration constant	2
10.	PSO Random number	[0,1]
11.	PSO Maximum velocity	half of range
12.	PSO Maximum position	range of dataset
13.	Number of testing	10

TABLE II. BENCHMARK FUNCTIONS WITH THE RANGE

No	Benchmark Function	Range [min,max]	Act. Opt.
1.	Sphere, $f_1(x)$	[-5.12,5.12]	0
2.	Ackley, $f_2(x)$	[-30,30]	0
3.	Rastrigin, $f_3(x)$	[-5.12,5.12]	0
4.	Zakharov, $f_4(x)$	[-5,10]	0
5.	Axis parallel hyper-ellipsoid, $f_5(x)$	[-5.12,5.12]	0
6.	Griewank, $f_6(x)$	[-600,600]	0
7.	Sum of Different Power, $f_7(x)$	[-1,1]	0
8.	Schwefel Double Sum, $f_8(x)$	[-65.536,65.536]	0
9.	Quartic with Noise, $f_9(x)$	[-1.28,1.28]	0
10.	Michalewitz, $f_{10}(x)$	[0,π]	-0.966d

For $x_i = 0$, $i = 1,\ldots,d$, where d is the dimension size of benchmark function

GA performs faster than other methods because steady state replacement is implemented. Worst chromosome is replaced in each generation with better chromosome. DE sometimes performs better in obtaining minimum value and generational update for DE is good in terms of generalized previous generation chromosome. However, DE is slower compared to GA in terms of speed. Each DE chromosome is compared with the trial vector in producing new children. PSO must also update all particles location after G_{BEST} was updated. This can be shown by the result whereby same maximum generation was used, but DE and PSO required longer time compared to GA. Hence, GA can perform even better if standard time usage is used as the termination criteria. However, it still depends on the operation techniques used in GA, which can affect the overall result. In addition, GA contains more operations to be performed, but it still able to perform faster and better than DE and PSO.

New genetic values will be introduced by DE during breeding operation while new particle location will be updated by PSO using the updating process. The search space can be improved and better result can be obtained. However, the drawback is longer time will be used in producing the result. Two GA methods, namely GA1 and GA2 with different operation techniques were applied in [3] to test the performance of GA. The authors showed that GA1 did not introduce any new genetic values to the structure because interchanging was performed, but new chromosome structure was formed. When GA2 introduces new genetic values during the mutation operations, it takes even longer time for the method in searching for optimum value.

Actually, DE mutation operation is dependent on the crossover probability whereby the result will only be applied during crossover operation. Hence, mutation can be considered only take place when crossover was performed. As stated by author of [35], DE does not have separate mutation operator. PSO is too much dependent on the G_{BEST} and parameters usage. Other particles will trap in local optimum because influence by G_{BEST}. Besides that, a lot of parameters were used in PSO. All parameters are used to decrease the result of particles as the iteration is increasing.

Although the parameters are linearly decreasing when the iteration is increasing, but it still can influence a lot on the result produced. P_{BEST} and previous velocity are used in updating the current particle location will let the particle moving away from optimum result. Differential weight for DE is used to reduce the result during mutation operation while inertia weight, acceleration constant and random number are used by PSO in decreasing the velocity value during updating process. But for GA, it did not contain any parameters that will directly influence the result during performing the operations. This statement shows that GA contains the pure result and did not use any parameters to reduce its result.

TABLE III. EXPERIMENTAL RESULTS

Func.	Act. Opt.	Me-thod	Min	Max	Average	Ave. Time (s)
$f_1(x)$	0	GA	0.0008	0.1978	0.0535	0.0641
		DE	0.0040	0.3654	0.0932	0.4951
		PSO	0.0099	0.0719	0.0237	0.5206
$f_2(x)$	0	GA	0.0169	3.6098	0.6513	0.0770
		DE	0.5545	4.5842	2.2593	0.7646
		PSO	0.8398	2.3545	1.7527	0.7648
$f_3(x)$	0	GA	0.0563	19.1326	4.6809	0.0752
		DE	5.0154	12.3525	9.7992	0.7385
		PSO	46.1047	116.1930	79.3988	0.7605
$f_4(x)$	0	GA	0.2972	1.6587	0.7395	0.0662
		DE	0.0649	1.9861	0.6328	0.5206
		PSO	0.0613	0.6920	0.2835	0.5595
$f_5(x)$	0	GA	0.0019	1.5947	0.4844	0.0629
		DE	0.0032	0.7700	0.2245	0.5124
		PSO	0.1685	1.1863	0.4755	0.5301
$f_6(x)$	0	GA	2414.3400	2420.5900	2416.5120	0.0837
		DE	91.0157	206.6490	164.3646	0.9084
		PSO	361.0020	1081.0000	621.9848	1.1379
$f_7(x)$	0	GA	1.0000E-06	5.4100E-04	1.0300E-04	0.0660
		DE	8.9361E-11	1.9694E-06	3.1227E-07	0.4943
		PSO	1.6500E-16	2.4500E-14	5.2400E-15	0.5278
$f_8(x)$	0	GA	1.0000E-06	5.4100E-04	1.0300E-04	0.0660
		DE	8.9361E-11	1.9694E-06	3.1227E-07	0.4943
		PSO	1.6500E-16	2.4500E-14	5.2400E-15	0.5278
$f_9(x)$	0	GA	0.0827	0.4930	0.2535	0.0649
		DE	0.0054	0.1037	0.0242	0.4986
		PSO	0.0342	0.0982	0.0578	0.5231
$f_{10}(x)$	-28.98	GA	-24.9448	-22.3162	-23.9686	0.1050
		DE	-23.7439	-17.0515	-20.0908	1.2357
		PSO	-22.9425	-14.9033	-19.0958	1.3344

Suggestions in handling the weaknesses were listed in Table IV. Both Generational and Steady state update in GA still contain limitations. Although whole population chromosomes in Generational update were replaced with new chromosomes, it still required longer time in searching for

TABLE IV. COMPARISON OF EACH METHOD

Method	Performance			Advantage	Weakness	Suggestion
	Min. Fitness	Ave. Fitness	Ave. Time			
GA	Contain 5/10 best minimum fitness.	Contain 3/10 best average fitness.	Contain 10/10 faster than DE and PSO.	1. If suitable operation techniques were implemented, GA can be terminated faster and the performance can be increased [3]. 2. The diversity of chromosome was increased by using breeding operations.	1. Generational update i. Whole population will be replaced with next generation chromosomes. ii. The best solution can only be found from the global optimum. Hence, did not have a stable result. 2. Steady state update i. At most 2 new chromosomes can be introduced in each generation, but it still depending on the fitness of the children. 3. No any new genetic values can be introduced to the population, although new chromosome structure can be formed during crossover and mutation.	1. Some new genetic values should be introduced into the population without increasing too much time for GA in searching optimum result.
DE	Contain 2/10 best minimum fitness.	Contain 3/10 best average fitness.	Contain 10/10 faster than PSO.	1. Perform better compared to GA [36,37]. 2. Using combination of the same population chromosome in forming new generation.	1. Probably none of previous generation chromosomes are carried forward to the next generation. However, better result can be produced. 2. Crossover and mutation operation were performed as one process [35].	1. If crossover and mutation operation were not performed, let the chromosome learned towards global optimum.
PSO	Contain 3/10 best minimum fitness.	Contain 6/10 best average fitness.	Contain 0/10 faster than GA and DE.	1. Velocity and position value can be controlled by using velocity clamping parameter [38]. 2. Easier to be implemented compared to GA [25]. 3. Less number of parameters to be tuned compared to GA [25].	1. Too much dependent on global best position i. Other particles will trap in local optimum because influence by global best position. 2. Previous velocity and best position were referred will make the particle position value increasing and moving away from the global best solution and optimum result. 3. Too much and dependent on the parameters which were used to decrease the result.	1. Decrease number of parameters usage.

local optimum. Hence, steady state update is preferred to be applied in GA. The weaknesses of steady stated update can be handled by introducing more new genetic values, besides normal breeding operations. This can also improve the searching space of GA by avoid trapping in local optimum.

If the generated number is less than DE crossover probability, hence the mutant vector values are not applied into the new trial vector. Therefore, crossover and mutation is considered as one process. The limitation can be handled by introducing new operation that let the chromosome to learn towards global optimum. If normal breeding operations are not performed, new genetic values still can be introduced from the new operation. PSO is too much dependent on the parameters usage. The parameters will guide the particles to move away from global optimum. In addition, too many parameters were referred in updating the particles location have increased the time taken in producing the result. Parameters usage should be decreased to avoid the result influenced by using parameters.

As a conclusion, so far there are no research works introduce GA and DE to learn and approximate towards global optimum, which is the good characteristic of PSO. Besides updating the local chromosome, they can approximate towards global optimum, which is nearer to optimal result. Therefore, the suggestions can be applied into the new hybrid methods by testing its performance using the same benchmark functions.

V. CONCLUSION, FUTURE WORK AND DIRECTION

GA was proven to perform better compared to DE and PSO. It can obtain highest number of best minimum fitness and faster than both methods. However, the GA operation techniques still play an important role, which can affect the performance of GA. Although DE is faster than PSO, but based on minimum and average fitness, PSO contains better performance than DE. The operation in updating DE chromosomes and PSO particles is one of the problems which affect the performance of both methods. Although new values were introduced, but longer time was needed in producing the result. In addition, two operations of DE based on one parameters and too many parameters usage for PSO are the drawbacks for both methods. Although DE and PSO are weaker than GA, but the limitations can still be handled based on the suggestions given in the previous section.

For future work, as suggested by authors of [3] in improving the GA performance, other GA operation techniques can be tested and maybe better result can be obtained. Furthermore, same dataset can be used to test with different methods and the most optimum value method can be obtained. This can also prove the performance of the methods and techniques. The parameters setting can be adjusted so that most optimum value can be obtained for each method and maybe termination criteria can be fulfilled faster.

Future direction will be focused in modification on the operations inside each method. Different operation techniques can be substituted in order to improve overall result produced. Besides that, new operation or enhancement can be applied on the methods and it might improve the result. In addition, hybrid between GA, DE or PSO with other SC methods, such as Ant or Bee Colony Optimization can be implemented for each method. Probably this will improve the result obtained and more approximate towards the actual optimum value. DE mutation and crossover maybe can be separated and the performance can be improved. Maybe current particle location concept for PSO can be removed to avoid the particle to fly away from actual optimum, although velocity clamping concept is implemented. This idea can reduce the time in searching optimum result.

REFERENCES

[1] S. N. Sivanandam and S. N Deepa, Introduction to Genetic Algorithm, Springer, 2008.

[2] S. P. Lim and H. Haron, "Surface reconstruction techniques: a review," Artif Intell Rev, DOI: 10.1007/s10462-012-9329-z, Springer, pp. 1-20, 2012.

[3] S. P. Lim and H. Haron, "Performance of different techniques applied in genetic algorithm towards benchmark functions," ACIIDS 2013, LNAI 7802, Part I, 2013, pp. 255–264.

[4] F. Vavak and T. C. Fogarty, "A comparative study of steady state and generational genetic algorithms for use in nonstationary environments," LNCS 1143, 1996, pp. 297-304.

[5] A. P. Engelbrecht, Computational Intelligence, John Wiley & Sons, Ltd, 2002.

[6] N. Mansour, M. Awad, and K. El-Fakih, "Incremental genetic algorithm," The International Arab Journal of Information Technology, vol. 3, no. 1, pp. 42–47, 2006.

[7] S. Rajasekaran and G. A. V. Pai, Neural Networks, Fuzzy Logic, and Genetic Algorithms Systhesis and Applications, Prentice-Hall of India Private Limited, 2007.

[8] C. T. Lin, W. C. Cheng, and S. F. Liang, "Neural–network–based adaptive hybrid–reflectance model for 3–d surface reconstruction," IEEE Transactions On Neural Networks, vol. 16, no. 6, pp. 1601–1615, Nov. 2005.

[9] X. Cheng, J. Wang, and Q. Wang, "Leak–mending and recruitment of incomplete points data in 3d reconstruction based on genetic algorithm," Third International Conference on Natural Computation, 2007, pp. 259–263.

[10] A. Saeedfar and K. Barkeshli, "Shape reconstruction of three–dimensional conducting curved plates using physical optics, NURBS modeling, and genetic algorithm," IEEE Transactions On Antennas And Propagation, vol. 54, no. 9, pp. 2497–2507, 2006.

[11] S. Wang and A. P. Dhawan, "Shape–based reconstruction of skin lesion for multispectral nevoscope using genetic algorithm optimization," 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 2007, pp. 488–491.

[12] R. Storn and K. Price, "Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces," Journal of Global Optimization 11, pp. 341–359, 1997.

[13] T. Weise, (2009). Global Optimization Algorithms – Theory and Application 2nd Ed. [Online]. Available at: http://www.it–weise.de/

[14] S. Das and P. N. Suganthan, "Differential evolution a survey of the state-of-the-art," IEEE Transactions on Evolutionary Computation, vol. 15, no. 1, pp. 4 – 31, 2011.

[15] M. Dehmollaian, "Through–wall shape reconstruction and wall parameters estimation using differential evolution," IEEE Geoscience and Remote Sensing Letters, vol. 8, no. 2, pp. 201–205, 2011.

[16] C. Studholme, (2001). Deriving Camera and Point Location From a Series of Photos Using Numerical Optimization. [Online]. Available at http://www.cs.toronto.edu/~cvs/geometry/GeometryProject.pdf.

[17] I. T. Rekanos, "Shape reconstruction of a perfectly conducting scatterer using differential evolution and particle swarm optimization," IEEE Transactions on Geoscience And Remote Sensing, vol. 46, no. 7, pp. 1967–1974, 2008.

[18] Y. Li, L. Rao, R. He, G. Xu, Q. Wu, M. Ge, and W. Yan, "Image reconstruction of EIT using differential evolution algorithm," Proceedings of the 25th Annual International Conference of the IEEE EMBS. Cancun, Mexico, 2003, pp. 1011–1014.

[19] P. Pandunata and S. M. Shamsuddin, "Differential evolution optimization for bezier curve fitting," 2010 Seventh International Conference on Computer Graphics, Imaging and Visualization, 2010, pp. 68-72.

[20] J. Kennedy and R. Eberhart, "Particle swarm optimization," IEEE International Conference on Neural Networks, Perth, Australia, 1995, pp. 1942–1948.

[21] Y. H. Shi and R. Eberhart, "A Modified Particle Swarm Optimizer," IEEE International Conference on Evolutionary Computation Proceedings, 1998, pp. 69-73.

[22] M. Chandrasekaran, M. Muralidhar, C. Murali Krishna, and U. S. Dixit, "Application of soft computing techniques in machining performance prediction and optimization: a literature review," Int J Adv Manuf Techol 46, pp. 445–464, 2009.

[23] J. Robinson, S. Sinton, and Y. Rahmat-Samii, "Particle swarm, genetic algorithm, and their hybrids optimization of a profiled corrugated horn antenna," IEEE Antennas and Propagation Society International Symposium, vol. 1, 2002, pp. 314-317.

[24] A. A. A. Esmin, G. Lambert-Torres, and G. B. Alvarenga, "Hybrid evolutionary algorithm based on pso and ga mutation," Proceedings of the Sixth International Conference on Hybrid Intelligent Systems, 2006, pp. 57.

[25] M. A. Panduro, C. A. Brizuela, L. I. Balderas, and L. I. Acosta, "A comparison of genetic algorithms, particle swarm optimization and the differential evolution method for the design of scannable circular antenna arrays," Progress In Electromagnetics Research B, vol. 13, 2009, pp. 171-186.

[26] Q. Bai, "Analysis of particle swarm optimization algorithm," Computer and Information Science, vol. 3, no. 1, pp. 180-184, 2010.

[27] K. Premalatha and A. M. Natarajan, "Hybrid PSO and GA for global optimization," Int. J. Open Problems Compt. Math., vol. 2, no. 4, pp. 598-608, 2009.

[28] D. Srinivasan, and T. H. Seow, "Particle swarm inspired evolutionary algorithm (PS-EA) for multiobjective optimization problems," Congress on Evolutionary Computation, 2003, pp. 2292-2297.

[29] A. Gálvez, A. Cobo, J. Puig–Pey, and A. Iglesias, "Particle swarm optimization for bézier surface reconstruction," LNCS 5102, 2008, pp. 116–125.

[30] F. Forkan, and S. M. Shamsuddin, "Kohonen–swarm algorithm for unstructured data in surface reconstruction," Fifth International Conference on Computer Graphics, Imaging and Visualization, 2008, pp. 5–11.

[31] M. Molga and C. Smutnicki, (2005). Test function for optimization needs. [Online]. Available at: http://www.zsd.ict.pwr.wroc.pl/files/docs/functions.pdf. pp. 1–43.

[32] D. Ortiz-Boyer, C. Hervás-Martínez, and N. García-Pedrajas, "A Crossover Operator for Evolutionary Algorithms Based on Population Features," Journal of Artificial Intelligence Research 24, pp. 1-48, 2005.

[33] A. R. Hedar, Test Functions for Unconstrained Global Optimization. [Online]. Available at: http://www-optima.amp.i.kyotou.ac.jp/member /student/hedar/Hedar_files/TestGO_files/Page364.htm.

[34] C. Robertson. (2002). Quartic Function with Noise, f_7. [Online]. Available:http://www.dca.fee.unicamp.br/WSC7/presentations/presentation-55/node18.html.

[35] T. Mantere, (2007). A Min-Max DE/GA Hybrid for Constrained Problems. [Online]. Available at: www.uva.fi/~timan/Min_max_GA_ 2007.pdf.

[36] X. L. Dong, S. Q. Liu, T. Tao, S. P. Li, and K. L. Xin, "A comparative study of differential evolution and genetic algorithms for optimizing the design of water distribution systems," *Journal of Zhejiang University-SCIENCE A (Applied Physics & Engineering)*, pp. 674-686, 2012.

[37] T. Tušar and B. Filipič, "Differential Evolution Versus Genetic Algorithms in Multiobjective Optimization," *LNCS*, vol. 4403, 2007, pp. 257–271.

[38] J. Kennedy, R. C. Eberhart, and Y. Shi. (2001). *Swarm intelligence*. Morgan Kaufmann Publishers, San Francisco.