Oral Presentation

The 7th TSME International Conference on Mechanical Engineering
13-16 December 2016

TSME ICOME 2016 Mechanical Engineering Innovation for Green Society

CST0013

Four-bar linkage path generation using meta-heuristics – algorithm comparison

W. Phukaokaew[1], S. Sleesongsom[2], S. Bureerat[1]*

[1] Sustainable and Infrastructure Research and Development Center, Department of Mechanical Engineering, Faculty of Engineering, KhonKaen University, KhonKaen City, Thailand
[2] Department of Mechanical Engineering, Faculty of Engineering, Chiangrai College, Chiangrai City, Thailand

sujbur@kku.ac.th

Abstract

In this research, the use of meta-heuristics for path generation of a four-bar linkage is demonstrated. Three problems of path generation were posed as a constrained optimization problem. A simple penalty function technique was used to deal with design constraints while seven meta-heuristics including artificial bee colony optimization (ABC), real code ant colony optimization (ACOR), population-based incremental learning (PBIL), differential evolution(DE), self-adaptive differential evolution (JADE), teaching learning based optimization (TLBO) and a grey wolf optimizer (GWO) were employed to solve the problems. Comparative results and the effect of the constraint handling technique are illustrated and discussed.

Keywords **:** mechanism synthesis, path generation, four-bar linkage, optimization, meta-heuristics.

1. Introduction

A four-bar linkage is the simplest mechanism used in many machines, which composes of four simple binary links, four revolute joints where one link is fixed as the frame. Some applications for this mechanism is a crank-rocker, a window wiper, a door closing mechanism, and rock crushers etc. From such popularity, many researchers therefore study on four bar mechanism synthesis. The well-known type of the four bar mechanism synthesis is path generation [1-5]. The path generation or path synthesis is a mechanism synthesis to make a point on a coupler link move along the path defined by a designer [3]. Much interesting work has been presented [3-4]. They proposed to study the path generation of the four bar mechanism by using meta-heuristic optimizers and studied the comparative performance. Furthermore, the work by Ebrahim and Payvandy proposed the efficient constraint handling technique, which is believed to be able to enhance the performance of the path generation [4].

It has been found that the advantages of using meta-heuristics are robustness, simplicity to use and independence of function derivatives, but they lack of a convergence speed and consistency [6]. However, in the last decade, many algorithms in the group of meta-heuristics are developed, which are expected to enhance their searching performance in both convergence speed and consistency. These algorithms include the artificial bee colony optimization (ABC), real code ant colony optimization (ACOR), ant colony optimization technique(ACO), simulated annealing (SA), differential evolution(DE), shuffled frog leaping (SFL), memetic algorithms (MA), population -based incremental learning(PBIL), genetic algorithm (GA), particle swarm optimization (PSO), self-adaptive differential evolution (JADE), and modified krill herd (MKH) etc [7]. All these algorithms have their own advantages and disadvantages depending on a problem being solved.

The objective of this research is to investigate on search performance of 7 algorithms for four-bar linkage path generation. The algorithms include the artificial bee colony optimization (ABC), the teaching learning based optimization (TLBO), the real code ant colony optimization (ACOR), the differential evolution (DE), the population-based incremental learning(PBIL), the self-adaptive differential evolution (JADE) and the grey wolf optimizer (GWO).

The rest of this paper is organized as follows. Section 2 details the position analysis of a four-bar linkage, the objective function and the constraint handling technique. All of MHs are presented in Section 3. A design problem and its conditions as well as a numerical experiment are given in Section 4, while the design results are in Section 5. The conclusions and discussion of the study are finally drawn in Section 6.

2. Position analysis of a four-bar mechanism, an objective function and a constraint handling technique

2.1 Position analysis

From Fig. 1, the vector loop equation of the mechanism can be written as

$$\mathbf{r}_1 + \mathbf{r}_2 + \mathbf{r}_3 + \mathbf{r}_4 = 0 \qquad (1)$$

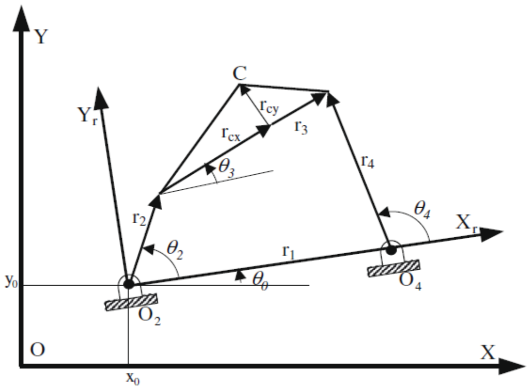

Fig. 1 four-bar linkage in global coordinate system[3].

The vector loop equation can be rewritten as components in x-y coordinates, which enable the solutions for the angle θ_3 and θ_4 by using the eliminating unknown method (see more detail in [3, 4]), leading to

$$\theta_{3_{1,2}} = 2\tan^{-1}\left(\frac{-E\pm\sqrt{E^2-4DF}}{2D}\right) \tag{2}$$

$$\theta_{4_{1,2}} = 2\tan^{-1}\left(\frac{-B\pm\sqrt{B^2-4AC}}{2A}\right) \tag{3}$$

where

$$A = \cos\theta_2 - K_1 - K_2\cos\theta_2 + K_3 \tag{4}$$

$$B = -2\sin\theta_2 \tag{5}$$

$$C = K_1 - (K_2+1)\cos\theta_2 + K_3 \tag{6}$$

$$D = \cos\theta_2 - K_1 - K_4\cos\theta_2 + K_5 \tag{7}$$

$$E = -2\sin\theta_2 \tag{8}$$

$$F = K_1 - (K_4-1)\cos\theta_2 + K_5 \tag{9}$$

$$K_1 = \frac{r_1}{r_2} \tag{10}$$

$$K_2 = \frac{r_1}{r_4} \tag{11}$$

$$K_3 = \frac{r_2^2 - r_3^2 + r_4^2 + r_1^2}{2r_2r_4} \tag{12}$$

$$K_4 = \frac{r_1}{r_3} \tag{13}$$

$$K_5 = \frac{r_4^2 - r_1^2 - r_2^2 - r_3^2}{2r_2r_3} \tag{14}$$

The position of the coupler C is defined as:

$$\begin{bmatrix} C_X \\ C_Y \end{bmatrix} = \begin{bmatrix} \cos\theta_0 & -\sin\theta_0 \\ \sin\theta_0 & \cos\theta_0 \end{bmatrix} \begin{bmatrix} C_{Xr} \\ C_{Yr} \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \tag{15}$$

where

$$C_{Xr} = r_2\cos\theta_2 + r_{cx}\cos\theta_3 - r_{cy}\sin\theta_3 \tag{16}$$

$$C_{Yr} = r_2\sin\theta_2 + r_{cx}\sin\theta_3 + r_{cy}\cos\theta_3 \tag{17}$$

2.2 Objective function and Constraint handling

The objective function includes two parts as the main part and the constraint penalty part [3]. The main part of objective function is the position error between the obtained points from mechanism (C) and the target points on the path (C_d) initiated by a designer, while the constraints part is inserted in objective function as the penalty function value. The first constraint (h_1) is set in such a wat that the input crank can rotate with a complete revolution in either direction (clockwise or counter clockwise). The second constraint (h_2) is assigned to control link lengths, which is in accordance with the Grashof's criterion. The design variables (***X***) include r_1, r_2, r_3, r_4, r_{cx}, and r_{cy}, the global position of the O_2 (x_O, y_O), the angle of frame 1 (θ_0) and the input angle (θ_2).
The optimization problem can then be written as:

$$\min\ f_{obj} = \sum_{i=1}^{N}\left[(C_{Xd}(X) - C_X(X))^2 + (C_{Yd}(X) - C_Y(X))^2\right] + M_1h_1(X) + M_2h_2(X) \tag{18}$$

subject to:

$\min(r_1, r_2, r_3, r_4) = \text{crank}$

$2\min(r_1, r_2, r_3, r_4) + 2\max(r_1, r_2, r_3, r_4) < (r_1 + r_2 + r_3 + r_4)$

$\theta_2^1 < \theta_2^2\ \ldots < \theta_2^N$

$\mathbf{x}_l \leq \mathbf{x} \leq \mathbf{x}_u.$

where $\mathbf{x} = \{\ r_1, r_2, r_3, r_4\ , r_{cx}, r_{cy}, \theta_0, x_0, y_0, \theta_2^1, \theta_2^2, \ldots, \theta_2^N\ \}^T$, N is the number of points on the prescribed or desired curve, and M_1 and M_2 are constants of a very high value that penalizes the objective function when the constraints fail. In this work, the values of M_1 and M_2 are set as 1000. The constraint handling is shown as follows.

$$\left.\begin{matrix} h_1(X) = 0 \\ h_2(X) = 0 \end{matrix}\right\}\quad \text{true}$$

$$\left.\begin{matrix} h_1(X) = 1 \\ h_2(X) = 1 \end{matrix}\right\}\quad \text{false}$$

2.3 Flowchart

The flow diagram of the entire algorithm for the synthesis of mechanism is shown in Fig. 2.

3. Meta-heuristics

In a comparative study, the meta-heuristic algorithms that are used in this study are briefly detailed as:

3.1 Artificial Bee Colony (ABC)

ABC is an optimizer mimicking bee behavior in finding food sources. Food positions are analogous to design solutions where employed and onlooker bees are assigned to find the best food position. An indicator for best food is a solution fitness value or an objective function value of an optimization problem [7].

3.2 Real code ant colony optimization (ACOR)

Ant colony optimization was first proposed for solving combinatorial optimization such as a travelling salesman problem. It was then upgraded for general real parameter optimization. The method is based on using a set of design solutions which is traditionally called a population as with other evolutionary algorithms.

3.3 Population based incremental learning (PBIL)

The original PBIL [8] algorithm was governed by binary searching similar to genetic algorithms (GA) without crossover operation. The method uses the so-called probability vector to represent a binary population.

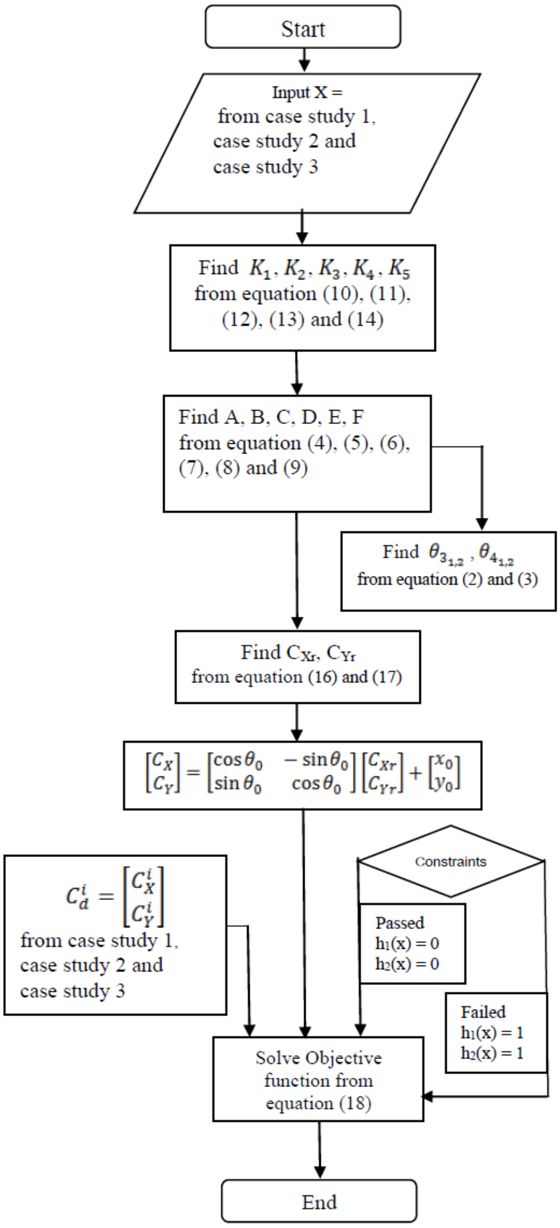

Fig. 2 Flow diagram for four-bar linkage mechanism design.

3.4 Differential evolution (DE)

DE is one of the most popular and powerful EAs which is a population-based stochastic optimization method. It starts with an initial population, which is randomly generated when no preliminary knowledge about the solution space is available. The DE operators include mutation, crossover, and selection. These operators are used to maintain population diversity, as well as to avoid a premature convergence. The DE scheme used in this study can be classified as the standard DE/best/2/bin algorithm [9].

3.5 Grey wolf optimizer (GWO)

The GWO algorithm imitates a hunting mechanism of grey wolves in which the wolves classify themselves as alpha, beta, gamma, and omega groups. The design solution vector is thought of as the position of a wolf. The leader alpha found based on an objective function value and other groups are used in the recombination process to improve solutions in a population generation by generation until the optimum is reached [10].

3.6 Self-adaptive differential evolution (JADE)

JADE is an optimizer with self-adaptive parameter settings of DE. It is regarded as one of the most powerful DEs. For example, the scaling factor (F) used in DE mutation and the crossover probability (CR) are crucial parameters for its performance and is also problem-dependent. As a consequence, the development of self-adaptive DE began, which to some extent can improve DE search performance. Adaptive schemes of JADE can update the control parameters based on their historical record of success [11].

3.7 Teaching learning based optimization (TLBO)

TLBO [12] exploits the concept of teaching and learning behavior of a teacher and students in a classroom. Surely, all students will follow their teacher and they often learn from each other where the clever one will teach another. With such an idea, TLBO is formulated in such a way that its reproduction process has two main operators namely teaching and learning phases. The algorithm is a population-based optimizer where a population of design solutions is improved iteratively until reaching the termination criterion.

4. Case studies

The design problems used to study the performance of the meta-heuristics have 3 cases. All optimization runs are performed in Matlab software. To study the performance of the EAs, the different parameters defining all algorithms are tabulated in Table 1.

Case-1 : Path generation without prescribed timing

Design variables are X

$$X = [r_1, r_2, r_3, r_4, r_{cx}, r_{cy}, \theta_0, x_0, y_0, \theta_2^1, \theta_2^2, \theta_2^3, \theta_2^4, \theta_2^5, \theta_2^6]$$

Target points are $C_d^i = \begin{bmatrix} C_X^i \\ C_Y^i \end{bmatrix}$

C_d^i = [(20,20),(20,25),(20,30),(20,35),(20,40),(20,45)]

Limits of the variables:

$$5 \le r_1, r_2, r_3, r_4 \le 60$$
$$-60 \le r_{cx}, r_{cy}, , x_0, y_0 \le 60$$
$$0 \le \theta_0\ \theta_2^1, \dots, \theta_2^6 \le 2\pi$$

Case-2 : Path generation with prescribed timing

Design variables are X

$$X = [r_1, r_2, r_3, r_4, r_{cx}, r_{cy}, \theta_0, x_0, y_0]$$
$$\theta_2^i = \left[\frac{\pi}{6}, \frac{\pi}{3}, \frac{\pi}{2}, \frac{2\pi}{3}, \frac{5\pi}{6}, \pi\right]$$

Target points are

C_d^i= [(0,0), (1.9098,5.8779), (6.9098, 9.5106), (13.09, 9.5106), (18.09,5.8779), (20,0)]

Limits of the variables:

$$5 \le r_1, r_2, r_3, r_4 \le 50$$
$$-50 \le r_{cx}, r_{cy}, , x_0, y_0 \le 50$$
$$0 \le \theta_0 \le 2\pi$$

Case-3 : Path generation without prescribed timing

Design variables are X

$$X = [r_1, r_2, r_3, r_4, r_{cx}, r_{cy}, \theta_0, x_0, y_0, \theta_2^1, \theta_2^2, \theta_2^3, \theta_2^4, \theta_2^5, \theta_2^6, \theta_2^7, \theta_2^8, \theta_2^9, \theta_2^{10}]$$

Target points are

C_d^i = [(20,10),(17.66,15.142),(11.736,17.878), (5,16.928), (0.60307,12.736), (0.60307,7.2638), (5,3.0718), (11.736, 2.1215), (17.66,4.8577), (20,10)]

Limits of the variables:

$$5 \le r_1, r_2, r_3, r_4 \le 80$$
$$-80 \le r_{cx}, r_{cy}, , x_0, y_0 \le 80$$
$$0 \le \theta_0\ \theta_2^1, \dots, \theta_2^{10} \le 2\pi$$

Table 1 Parameters of the meta-heuristics.

Parameters	ABC	ACOR	PBIL	DE	GWO	JADE	TLBO
Number of initial population	50	50	50	50	50	50	50
Crossover probability	-	-	-	0.7	-	0.5	-
Mutation Probability	-	-	0.05	-	-	-	-
Mutation shift	-	-	0.2	-	-	-	-
Learning rate	-	-	0.5	-	-	-	-
Scaling factor	-	-	-	0.7	-	0.5	-
Crossover rate	-	-	-	0.8	-	0.5	-
Number of optimization run	30	30	30	30	30	30	30
Generation number	500	500	500	500	500	500	500

5. Results and Discussion

The optimization of the mechanism synthesis is then tackled to obtain the optimized link lengths which minimize the objective function. Three case studies without and with prescribed timing are considered for performance testing of the optimizers.

The results of case-1 obtained from the seven optimizers are shown in Table 2. This table shows the number of successful runs (an optimizer can find a feasible solution) and the best linkage that gives the minimum error obtained from using each algorithm. Fig. 3 shows the path traced by the coupler point at the final solution. In this case the target points are 6 points. It is found that the DE gives the best result (error = 0.0166), the second is JADE (error = 0.1162) and the worst in this case is PBIL (error = 6.4088) while the method that gives the best mean and the number of successful runs is the ABC, the second best is JADE and the worst is ACOR. All of algorithms give the number of successful runs 100.00 % except GWO (83.34 %) and PBIL (80.00%). It is found that DE gives the better result than the previous work by [3] (error = 0.1227).

For the Case-2, the number of target points of 6, the results obtained from using the various MHs are shown in Table 3. The best linkages and coupler curves obtained from DE and JADE are shown in Fig. 4 and 5, respectively. In this case, DE and JADE are the winner (error = 1.6063) and the worst is ACOR (error = 6.2798). DE gives the best mean value of the objective function while the second best is JADE and the worst is PBIL. GWO gives 96.67 % of the number of successful runs while another algorithm gives 100% results. It is found that both DE and JADE give the better results than the previous work by [3] (error = 2.3496).

In case-3, the prescribed curve with 10 points, the results obtained by those algorithms are shown in Table 4 and Fig. 6. From the results, it is found that DE gives the best result (error = 0.1641), but it is the worst when considering the number of successful runs. The second best is JADE (error = 0.1809) which gives the best result for the number of successful runs and the worst error is from ACOR (error = 22.7428). It is found that both DE and JADE gives the better result than the previous work by [3] (error = 1.9523).

CST0013

Table 2 Comparative results for Case-1.

Case-1 : Path generation without prescribed timing				
Parameter	ABC	ACOR	PBIL	DE
r_1	33.2095	5.0000	47.5810	47.8395
r_2	19.3605	58.3005	5.0000	16.8195
r_3	30.7455	60.0000	24.5140	55.4460
r_4	12.0015	54.6265	47.5810	50.1990
r_{cx}	-15.5640	20.2800	40.6440	-51.9000
r_{cy}	60.0000	-41.544	-44.5200	19.3200
x_0	-55.6680	9.1560	-25.1640	58.5480
y_0	47.3160	31.8000	-5.8080	46.524
θ_0	4.4102	5.6159	0.2029	5.7944
θ_2^1	1.2874	5.3910	3.0404	6.1921
θ_2^2	2.0797	5.9741	3.8510	0.0251
θ_2^3	0.0000	0.0000	0.0000	0.0000
θ_2^4	2.3185	0.6616	5.0674	0.2476
θ_2^5	2.4567	0.9921	6.2832	0.3607
θ_2^6	2.5918	1.3276	6.2833	0.4788
error	1.3924	6.2442	6.4088	0.0166
mean	3.6359	14.8816	13.6510	6.9528
std	1.1381	3.8724	6.3992	7.4026
max	6.3398	22.1334	26.8656	20.9165
min	1.3924	6.2442	6.4088	0.0166
Success	30	30	24	30

* Success = no. of successful runs
* error = the objective function (minimum error)

Table 2 Comparative result of Case-1 (Continue).

Case-1 : Path generation without prescribed timing			
Parameter	GWO	JADE	TLBO
r_1	36.3830	32.9950	35.0850
r_2	6.6555	9.8125	6.9470
r_3	36.5920	29.0130	38.6160
r_4	32.1480	22.9520	59.9890
r_{cx}	-10.7160	34.7040	41.0520
r_{cy}	-48.5760	29.5560	-5.7960
x_0	59.9400	-13.4520	-12.9960
y_0	59.9880	59.7960	55.2720
θ_0	4.5936	4.1450	3.9226
θ_2^1	5.6907	2.0527	1.7134
θ_2^2	0.1766	2.5240	2.6276
θ_2^3	0.0000	0.0000	0.0000
θ_2^4	1.0926	3.3848	3.6945
θ_2^5	1.5293	3.8353	4.2782
θ_2^6	2.2921	4.4077	5.2370
error	1.4035	0.1162	0.1731
mean	7.3330	4.1089	9.9519
std	4.2549	2.6255	5.8397
max	18.5829	9.0904	20.9165
min	1.4035	0.1162	0.1731
Success	25	30	30

* Success = no. of successful runs
* error = the objective function (minimum error)

Table 3 Comparative result of Case-2.

Case-2 : Path generation with prescribed timing				
Parameter	ABC	ACOR	PBIL	DE
r_1	50.0000	38.3225	35.4830	50.0000
r_2	9.9365	8.30750	9.3560	5.0000
r_3	17.4740	20.0840	19.5170	7.0295
r_4	49.5050	50.0000	34.0340	48.1325
r_{cx}	35.8900	7.6200	50.0000	16.9800
r_{cy}	-0.6400	-23.0000	1.6100	12.9500
x_0	30.5700	31.0200	43.5500	12.2000
y_0	-28.1000	-9.2900	-33.8700	-16.0000
θ_0	0.9029	1.8058	1.2158	0.0427
error	3.8352	6.2798	5.1998	1.6063
mean	8.3035	9.5809	17.9088	3.5618
std	2.0502	2.0056	6.9858	2.4568
max	12.2279	13.4239	45.3574	8.8974
min	3.8352	6.2798	5.1998	1.6063
Success	30	30	30	30

* Success = no. of successful runs
* error = the objective function (minimum error)

Table 3 Comparative result of Case-2 (Continue).

Case-2 : Path generation with prescribed timing			
Parameter	GWO	JADE	TLBO
r_1	49.5770	50.0000	42.1115
r_2	5.9270	5.0000	5.0000
r_3	16.3310	7.0295	7.2815
r_4	45.8690	48.1325	39.9290
r_{cx}	50.0900	16.9800	16.0000
r_{cy}	3.7700	12.9500	15.0100
x_0	30.2600	12.2000	11.9200
y_0	-42.6200	-16.0000	-16.7300
θ_0	0.8200	0.0427	6.2826
error	4.3641	1.6063	1.7784
mean	7.5665	4.0046	5.4913
std	2.8148	2.4396	2.6898
max	13.6796	9.5103	12.3104
min	4.3641	1.6063	1.7784
Success	29	30	30

* Success = no. of successful runs
* error = the objective function (minimum error)

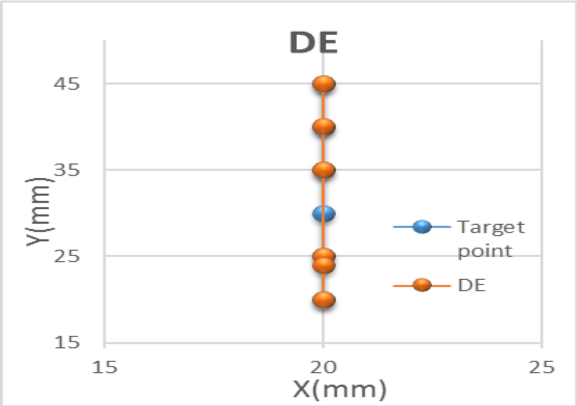

Fig. 3 coupler curves obtained in Case-1.

CST0013

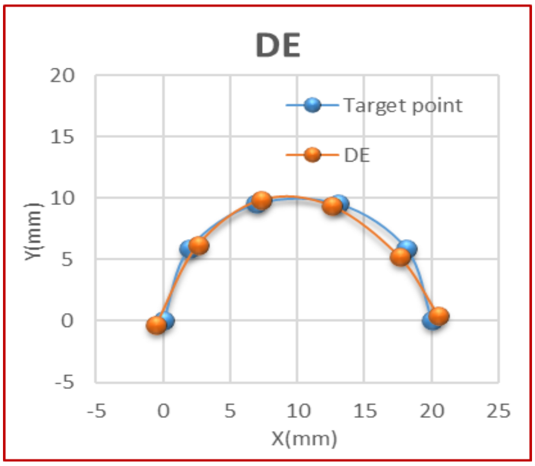

Fig. 4 coupler curves obtained in Case-2.

Table 4 Comparative result of Case-3.

Case-3 : Path generation without prescribed timing				
Parameter	ABC	ACOR	PBIL	DE
r_1	5.4575	80.0000	63.0650	65.4875
r_2	49.6025	5.0000	7.4225	8.0075
r_3	80.0000	77.8925	67.9025	24.8750
r_4	78.4700	80.0000	19.5125	51.9725
r_{cx}	10.8160	-6.2880	-23.2320	-5.0240
r_{cy}	-51.3440	-7.5840	7.7440	-2.4640
x_0	11.3440	10.0480	2.5760	9.9840
y_0	14.4800	24.4480	33.5500	4.4800
θ_0	4.5578	0.0000	2.0270	3.4608
θ_2^1	5.8069	0.0000	4.0539	2.8626
θ_2^2	6.2021	0.0000	5.0674	3.5412
θ_2^3	0.7584	2.8387	0.0000	4.2091
θ_2^4	2.5736	3.0228	0.4053	4.9040
θ_2^5	3.3081	3.5494	0.8105	5.5839
θ_2^6	3.7567	3.8038	1.4187	6.2819
θ_2^7	4.2380	4.1255	2.0270	0.7201
θ_2^8	4.7846	4.8676	2.6352	1.4231
θ_2^9	5.2754	5.7692	3.0404	2.1394
θ_2^{10}	5.7139	6.2832	3.8510	2.8614
error	2.2048	22.7428	7.5858	0.1641
mean	6.5769	151.7558	11.1130	1.9322
std	2.3471	104.6985	3.1175	2.5694
max	10.3337	371.1291	17.6738	4.8795
min	2.2048	22.7428	7.5858	0.1641
Success	29	121	10	3

* Success = no. of successful runs
* error = the objective function (minimum error)

Table 4. Comparative result of Case-3 (Continue).

Case-3 : Path generation without prescribed timing			
Parameter	GWO	JADE	TLBO
r_1	38.0375	71.3150	16.4300
r_2	5.0525	8.0075	5.0300
r_3	32.7725	53.2775	60.5075
r_4	40.4750	79.955	60.7175
r_{cx}	-20.3200	-9.5040	2.8320
r_{cy}	21.7280	6.6720	28.2560
x_0	11.9360	9.4880	33.4080
y_0	39.8880	-1.5200	-6.2400
θ_0	1.1668	3.9081	5.8779
θ_2^1	4.9091	2.3989	0.3531
θ_2^2	6.2009	3.0857	0.4009
θ_2^3	0.7766	3.7925	2.5981
θ_2^4	1.2566	4.4975	2.9851
θ_2^5	1.7266	5.1736	3.4067
θ_2^6	2.4894	5.8685	3.9207
θ_2^7	2.9355	0.2884	4.1513
θ_2^8	3.7228	0.9726	5.9678
θ_2^9	4.3279	1.6826	0.0496
θ_2^{10}	4.8481	2.3970	0.3437
error	7.7794	0.1809	10.1405
mean	16.0530	5.1536	23.5171
std	8.4502	2.0642	10.4886
max	27.0156	11.5682	40.3854
min	7.7794	0.1809	10.1405
Success	4	27	7

* Success = no. of successful runs
* error = the objective function (minimum error)

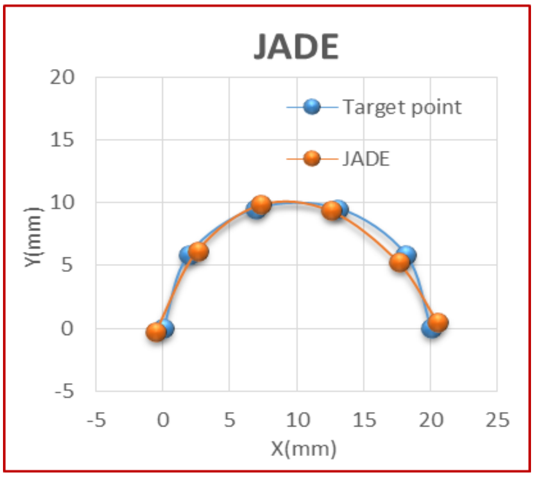

Fig. 5 coupler curves obtained in Case-2.

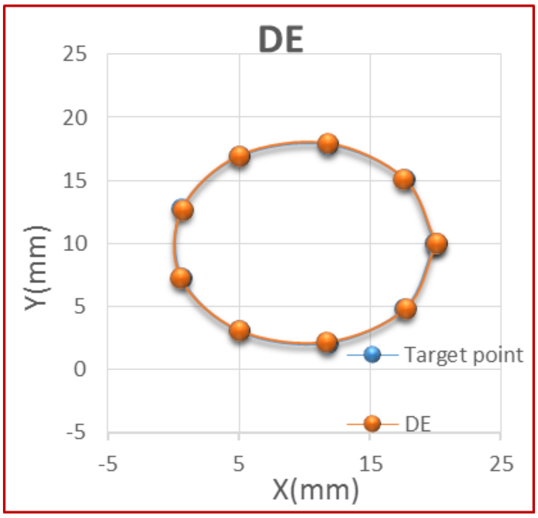

Fig. 6 coupler curves obtained in Case-3.

6. Conclusions

The paper presents the technique to find out the parameters of a four-bar linkage for a given path using the meta-heuristics including ABC, ACOR, PBIL, DE, GWO, JADE and TLBO. The comparative performance of the meta-heuristics show that the DE and self-adaptive JADE are superior to the others. Furthermore, the results show the constraint handling technique that is used in many previous studies, which is a penalty function results in many unsuccessful runs. That means there is no guarantee that using this technique will give good results. This implies that future work on path generation should focus on the constraint handling technique and using self-adaptive meta-heuristics rather than those who require initial parameter settings.

Acknowledgement

The authors are grateful for the financial support provided by the Thailand Research Fund (TRF, BRG5880014), and Khon Kaen University.

References

[1] Sleesongsom, S. and Bureerat, S. (2015).Optimal synthesis of four-bar linkage path generation through evolutionary computer. *Research and Applications in Mechanical Engineering*, Vol. 3(2) , pp. 46-53.

[2] Tong, Y., Myszka, D.H., Murray, A.P. (2013). Four-bar linkage synthesis for a combiation of motion and path-point generation. paper presented in *ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Oregon, USA.

[3] Acharyya, S.K. and Mandal, M. (2009). Performance of EAs for four-bar linkage synthesis. *Mechanism and Machine Theory*, Vol.85 , pp. 1784-1794.

[4] Ebrahimi, S. and Payvandy, P. (2015). Efficient constrained synthesis of path generating four-bar mechanisms based on the heuristic optimization algorithms. *Mechanism and Machine Theory*, Vol.85, pp. 189-204.

[5] Yu, H.Y., Zhao, Y.W., and Wang, Z.X. (2013). Study on numerical comparison method of four-bar straight-line guidance mechanism. *Journal of Harbin Institute of Technology* , Vol. 20 , pp. 63-72.

[6] Sleesongsom, S. and Bureerat, S.(2015). Morphing wing structural optimization using opposite-based population-based incremental learning and multigrid ground elements. *Mathematical Problems in Engineering*, Vol. 2015, Article ID 730626.

[7] Shaheen, A.M., Spea, S.R., Farrag, S.M., and Abido, M.A. (2015). A review of meta-heuristic algorithms for reactive power planning problem. *Ain Shams Engineering Journal* (Inpress).

[8] Baluja, S. Population-based incremental learning: a method for integrating genetic search based function optimization and competitive learning, Technical ReportCMU_CS_95_163, 1994, Carnegie Mellon University.

[9] Storn, R.M. and Price, K.V. Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces, *Journal of Global Optimization*, Vol. 11, 1997, pp. 341-359.

[10] Mirjalili, S., Mirjalili, S. M., Lewis, A. (2014). Grey Wolf Optimizer, *Advances in Engineering Software* , vol. 69, pp. 46-61.

[11] Zhang, J. and Sanderson, A.C. JADE: adaptive differential evolution with optional external archive, *IEEE Transaction on Evolution Computation*, Vol. 13(5), 2009, pp. 945-958.

[12] Rao, R.V. and Patel, V. (2012). An improved teaching-learning-based optimization algorithm for solving unconstrained optimization problems. *Scientia Iranica D*, Vol. 20, pp. 710-720.