Central Washington University

CS471 Optimization

Winter 2020

Genetic Algorithm, Differential Evolution, Particle Swarm Optimization, Sine Cosine Algorithm, Bat Algorithm, Directional Bat Algorithm, Ant Lion Optimizer, Quick Artificial Bee Colony

Student:
Chao Huang Lin
chao.huanglin@cwu.edu

Professor:
Dr. Donald Davendra
Donald.Davendra@cwu.edu

March 7, 2020

Contents

1 Introduction

For this project, many population-based optimization algorithms will be tested. These are Genetic Algorithm (GA) and Differential Evolution Algorithm (DE), Particle Swarm Optimization (PSO) and Sine Cosine Algorithm (SCA), Bat Algorithm(BA), Directional Bat Algorithm (DBA), the Ant Lion Optimizer (ALO), Quick Artificial Bee Colony (QABC).

1.1 Genetic Algorithm

The GA is a heuristic search and optimization technique that simulates the process of natural evolution. The main operations of GA are Selection, Crossover, Mutation, and Elitism.

1.2 Differential Evolution

The DE algorithm employs the difference of two randomly selected parameter vectors as the source fo random variations for a third parameter vector. Some advantages of DE are: few numbers of control parameters which make it easier to calibrate or tune, it is inherently parallel, and it has a faster convergence. There are many strategies of DE, in this project 10 different strategies will be tested.

1.3 Particle Swarm Optimization

The PSO is inspired by the flocking and schooling patterns of birds and fish, the key point of PSO is that it keeps tracking the personal best and global best variables, and make the particles move toward those best values.

1.4 Sine Cosine Algorithm

The SCA created in 2015 is very similar to PSO. However, it only keeps track of only the global optimal value, and in each iteration, it introduces sine and cosine formula which multiply with a decreasing range value.

1.5 Bat Algorithm

The BA proposed by Yang in 2010 is based on swarm intelligence algorithm, it is inspired by the echolocation behavior of microbats, as all the population-based algorithms the BA tries to find the optimal value by exploration and exploitation. At the beginning of the iterations, the algorithm focuses more on exploration by selecting in each iteration one of the best solutions (local minimums) randomly, then it adds a random walk step to it when this is better than the global best it will be accepted as a new solution. At the end part of the iteration the algorithm focus more in exploitation by moving the bats toward the best global best.

1.6 Directional Bat Algorithm

The DBA proposed by Chakri in 2016 is based on BA, it claims to be an improved version of BA adding 4 modifications that enhance the exploitation and exploration. The 1st modification consists of directional echolocation which makes the bat move in 2 combined directions (global best and random selected) instead of only one direction to the global best. The 2nd modification consists in decrease the search space through the iterations. The 3rd modification changed the formula about how the pulse rate and loudness increase and increase respectively. The 4th modification makes the algorithm to accept the new solution more often (increase exploration) and update the global best whenever the bat's random walk produces a better solution than the global optimal value.

1.7 Ant Lion Optimizer

The ALO algorithm simulates the population of antlions and the population of ants together, it mimics the interaction between antlions and ants in the trap. First, the antlion population is initialized randomly, then the elite antlion is located based on the fitness (cost), then each ant select an antlion using roulette wheel and random walk around it, also the same ant random walk around the elite ant, after that the new position of ant is calculated based on the two random walks, at the end, the antlions will update their positions to the best fitness ants position, and the elite antlion is also updated.

1.8 Quick Artificial Bee Colony

The QABC mimics the bee colony to find food, the algorithm is divided in 3 phases: the employed bee phase which update the initial food position based on mutation of a random dimension, the onlooker bee phase selects the food position based on fitness roulette wheel then it is updated with the best position in the selected food neighborhood, the scout bee phase converts the food position in to another random position when it cannot be improved anymore.

2 Method

The GA, DE, PSO, SCA, BA, DBA, ALO, QABC algorithms are coded using C++ object-oriented programming. The implemented classes are Population, Population-Benchmark, Functions, Runner, GeneticAlgorithm, DifferentialEvolution, ParticleSwarm, SineCosine, BatAlgorithm, DirectionalBatAlgorithm, AntLionOptimizer, QuickArtificialBeeColony class. Additionally, python script with jupyter notebook is implemented to read the configuration parameters and call the C++ executable, then it collects the result, displays it in table and figures, and prints the output in latex format.

The obtained results are from **50** runs of each algorithm with the population size equal to **200**, and the generations or iterations equal to **100**, the number of dimensions is set to 30.

The computer used to run the project has the following specification: Intel Core i7-9750H 2.6GHZ up to 4.5GHZ with 16 GB of RAM

2.1 Configuration Parameters

All evolution-based algorithms are very defendant of its parameters. To ensure that the optimization algorithm works correctly, the parameters had to be calibrated manually, the following tables show the result of the calibration:

The calibrated GA parameters are:

crossover rate	mutation rate	mutation value range	mutation precision	elitism rate
0.9	0.05	0.1	1	0.3

Table 1: Configuration parameters of GA obtained by manual tuning

The calibrated DE parameters are:

DE strategies	crossover rate	scaling factor F	scaling factor lambda
DE_best_1_exp	0.8	0.3	-
DE_rand_1_exp	0.9	0.1	-
DE_randbest_1_exp	0.9	0.4	0.4
DE_best_2_exp	0.9	0.25	-
DE_rand_2_exp	0.9	0.05	-
DE_best_1_bin	0.6	0.5	-
DE_rand_1_bin	0.8	0.2	-
DE_randbest_1_bin	0.7	0.5	0.5
DE_best_2_bin	0.8	0.4	-
DE_rand_2_bin	0.7	0.1	-

Table 2: Configuration parameters of DE obtained by manual tuning

The calibrated PSO parameters are:

c1	c2	w
1	1	0.7

Table 3: Configuration parameters of PSO obtained by manual tuning

SCA parameters from the research paper are:

a	r3
2	2

Table 4: Configuration parameters of SCA from the research paper

The calibrated BA parameters are:

r0	A0 min	A0 max (exploration)	alfa	gamma	f min	f max (exploitation)	ratio best
0.1	1	11	0.7	0.9	0	2	0.2

Table 5: Configuration parameters of BA obtained by manual tuning, the A0 Max was increased to enhance the exploration

The calibrated DBA parameters are:

r0	rinf	A0	Ainf	fmin	fmax
0.1	0.7	0.9	0.6	0	2

Table 6: Configuration parameters of DBA obtained by manual tuning

The ALO algorithm does not have any configuration parameter.
The calibrated QABC parameters are:

limit	neighborhood radius
50	0.5

Table 7: Configuration parameters of QABC obtained by manual tuning

3 Results

The results of applying different optimization algorithms with 18 benchmarking functions are displayed in the following pages:

3.1 Function 1: Schwefel

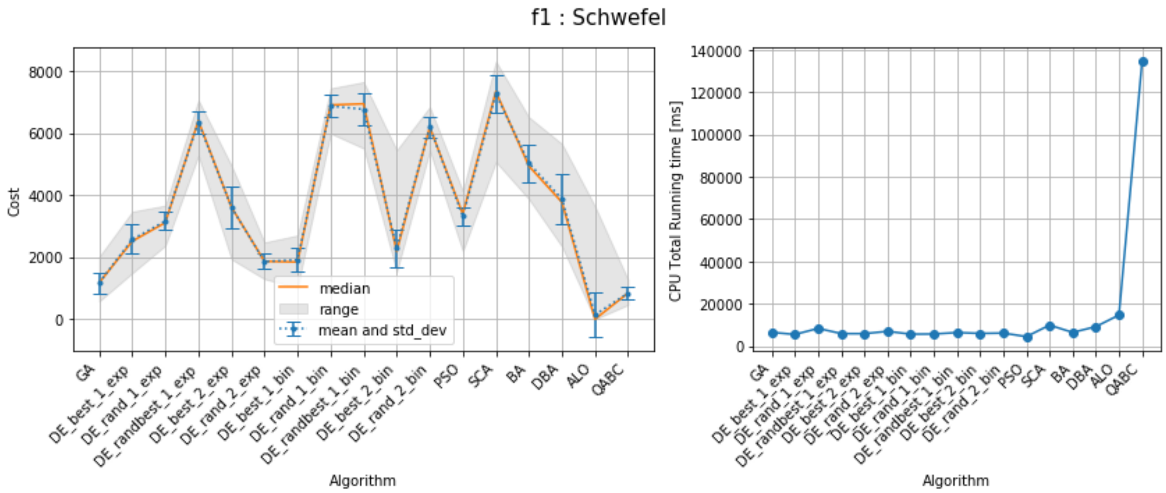

Figure 1: Cost and CPU total running time of Function 1: Schwefel

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	1177.430	336.443	1174.640	577.033	2016.630	6472
DE_best_1_exp	2575.090	470.369	2512.380	1486.380	3468.860	5493
DE_rand_1_exp	3161.440	294.324	3111.660	2373.020	3664.770	8361
DE_randbest_1_exp	6350.210	365.182	6397.040	5270.100	7058.650	5881
DE_best_2_exp	3607.240	683.564	3611.220	1932.470	4953.360	5866
DE_rand_2_exp	1856.560	247.222	1864.780	1293.870	2477.350	6958
DE_best_1_bin	1915.330	393.097	1843.630	1026.180	2703.440	5627
DE_rand_1_bin	6867.390	351.280	6902.280	5985.100	7440.960	5705
DE_randbest_1_bin	6763.960	513.070	6941.090	5505.640	7644.230	6376
DE_best_2_bin	2282.380	604.606	2179.800	1527.140	5450.490	5970
DE_rand_2_bin	6198.040	336.569	6196.650	5425.510	6839.930	6131
PSO	3316.760	285.908	3385.230	2187.680	4134.600	4461
SCA	7260.970	591.747	7335.860	5067.700	8307.730	9893
BA	5032.300	607.388	4934.480	3868.700	6510.220	6447
DBA	3869.680	805.661	3767.140	2338.970	5650.620	9172
ALO	143.976	705.096	0.012	0.000	3636.810	14647
QABC	843.371	194.354	843.779	465.484	1273.300	134612

Table 8: Function 1: Statistical Analysis of the Cost

Best Algorithm:
ALO, Cost (mean): 143.976000

3.2 Function 2: De Jong 1

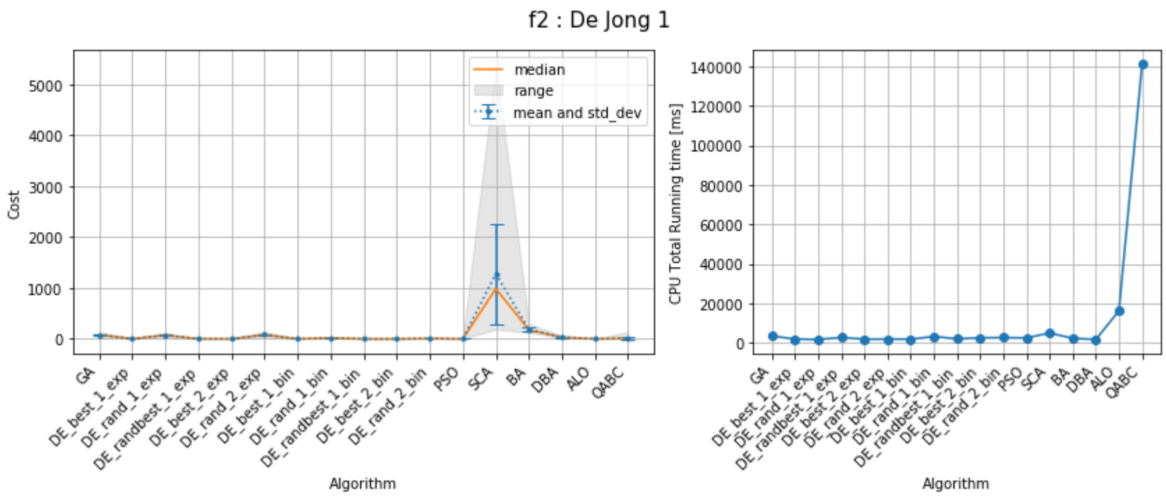

Figure 2: Cost and CPU total running time of Function 2: De Jong 1

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	78.974	25.345	77.429	36.792	138.514	3456
DE_best_1_exp	3.904	2.420	2.934	1.242	13.058	1930
DE_rand_1_exp	72.772	16.110	70.760	39.011	118.007	1731
DE_randbest_1_exp	2.641	1.121	2.210	1.328	6.652	2735
DE_best_2_exp	0.517	0.684	0.315	0.012	4.004	1854
DE_rand_2_exp	82.140	19.288	78.636	51.520	143.177	1914
DE_best_1_bin	0.003	0.002	0.002	0.001	0.010	1856
DE_rand_1_bin	19.226	4.716	18.215	10.036	32.483	3222
DE_randbest_1_bin	0.002	0.002	0.001	0.001	0.009	2038
DE_best_2_bin	0.006	0.002	0.005	0.001	0.011	2560
DE_rand_2_bin	7.767	1.655	7.618	3.633	11.410	2662
PSO	0.001	0.000	0.000	0.000	0.002	2479
SCA	1279.480	987.608	989.133	188.002	5406.840	5013
BA	186.497	40.068	174.257	126.007	298.708	2299
DBA	34.395	10.575	33.205	16.445	62.876	1726
ALO	0.010	0.024	0.002	0.000	0.141	16251
QABC	16.122	26.940	6.122	0.924	142.534	141338

Table 9: Function 2: Statistical Analysis of the Cost

Best Algorithm:
PSO, Cost (mean): 0.000508

3.3 Function 3: Rosenbrock's Saddle

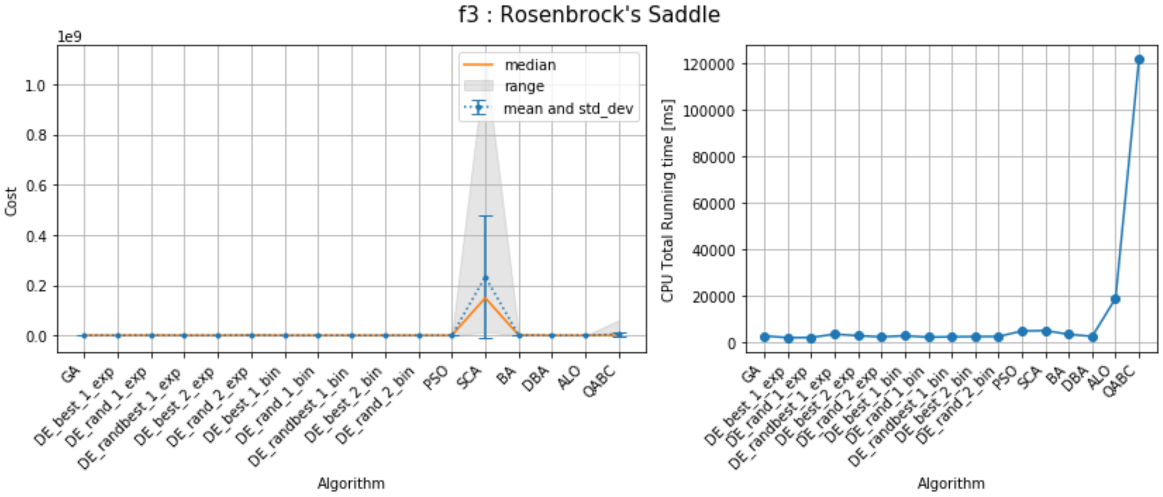

Figure 3: Cost and CPU total running time of Function 3: Rosenbrock's Saddle

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	338234.000	255343.000	247627.000	79114.900	1123070.000	2651
DE_best_1_exp	9718.930	7406.700	6387.750	729.830	31716.100	1886
DE_rand_1_exp	627829.000	235438.000	588275.000	190292.000	1233500.000	1964
DE_randbest_1_exp	1830.630	2491.330	1151.680	303.819	17076.200	3407
DE_best_2_exp	2690.410	3712.020	1288.490	99.414	17076.200	2738
DE_rand_2_exp	773060.000	289148.000	780377.000	237786.000	1354270.000	2254
DE_best_1_bin	197.640	339.038	132.118	27.725	2397.200	2699
DE_rand_1_bin	68334.800	29486.300	65296.200	16306.100	137469.000	2115
DE_randbest_1_bin	588.583	2810.820	116.636	28.548	20189.700	2288
DE_best_2_bin	583.328	1761.420	150.404	30.228	10217.200	2359
DE_rand_2_bin	37752.400	14346.500	34201.100	17248.000	80646.700	2394
PSO	60144.600	237466.000	81.905	16.871	1000080.000	4771
SCA	233206000.000	242472000.000	149098000.000	12934200.000	1100800000.000	4937
BA	1349370.000	956664.000	1197300.000	298282.000	5473630.000	3336
DBA	44443.000	40119.700	30786.900	6003.780	205884.000	2435
ALO	4.097	5.781	2.071	0.246	29.099	18833
QABC	4453380.000	9964280.000	856389.000	10012.700	61402400.000	121912

Table 10: Function 3: Statistical Analysis of the Cost

Best Algorithm:
ALO, Cost (mean): 4.096880
Two-Sample Z-Test Hypothesis Testing: confidence interval = 95%

Null hypothesis: The best algorithm and the tested one are equal
DE_randbest_1_bin, Cost (mean): 588.583000 , P value: 0.141500
PSO, Cost (mean): 60144.600000 , P value: 0.073300

3.4 Function 4: Rastrigin

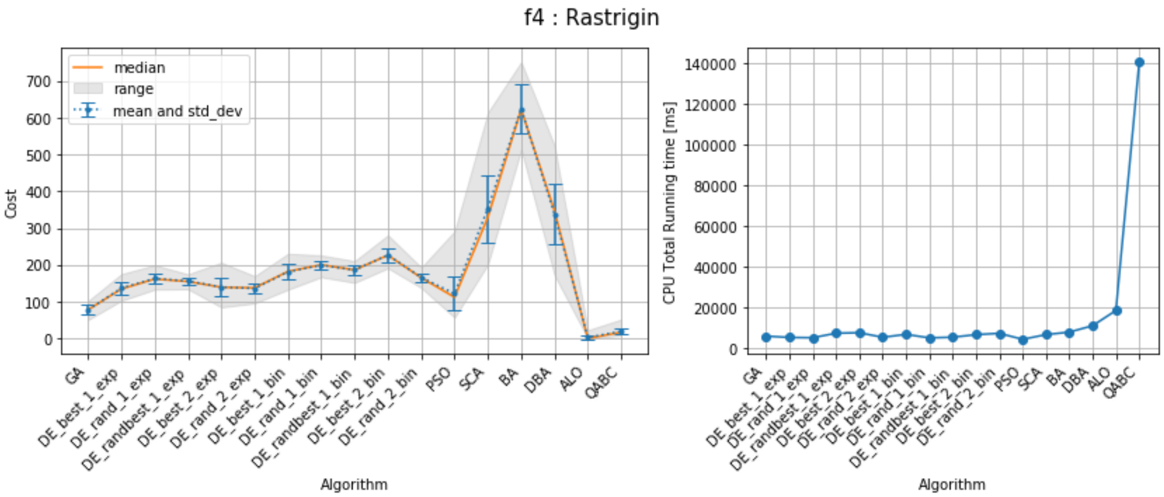

Figure 4: Cost and CPU total running time of Function 4: Rastrigin

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	77.903	12.640	77.920	50.668	102.212	5676
DE_best_1_exp	136.816	17.001	134.405	102.747	175.196	5042
DE_rand_1_exp	162.908	13.153	161.579	133.060	199.401	4906
DE_randbest_1_exp	155.888	10.306	154.740	134.001	174.658	7133
DE_best_2_exp	139.609	26.113	139.078	84.656	206.372	7364
DE_rand_2_exp	135.850	14.838	137.357	95.584	169.529	5114
DE_best_1_bin	181.146	21.153	182.012	132.811	230.488	6564
DE_rand_1_bin	199.177	12.428	199.979	166.736	226.073	4831
DE_randbest_1_bin	185.803	12.444	186.245	151.382	210.469	5157
DE_best_2_bin	226.498	19.282	226.041	190.959	280.766	6414
DE_rand_2_bin	165.820	11.904	165.069	137.397	193.276	7067
PSO	121.561	45.544	111.963	56.727	291.523	4114
SCA	352.991	91.334	329.088	200.891	614.182	6429
BA	623.680	66.934	622.340	512.705	751.182	7657
DBA	337.323	81.414	346.629	171.209	524.081	10809
ALO	2.744	5.065	0.052	0.000	22.851	18300
QABC	19.453	7.904	17.076	7.729	51.813	140594

Table 11: Function 4: Statistical Analysis of the Cost

Best Algorithm:
ALO, Cost (mean): 2.743820

3.5 Function 5: Griewangk

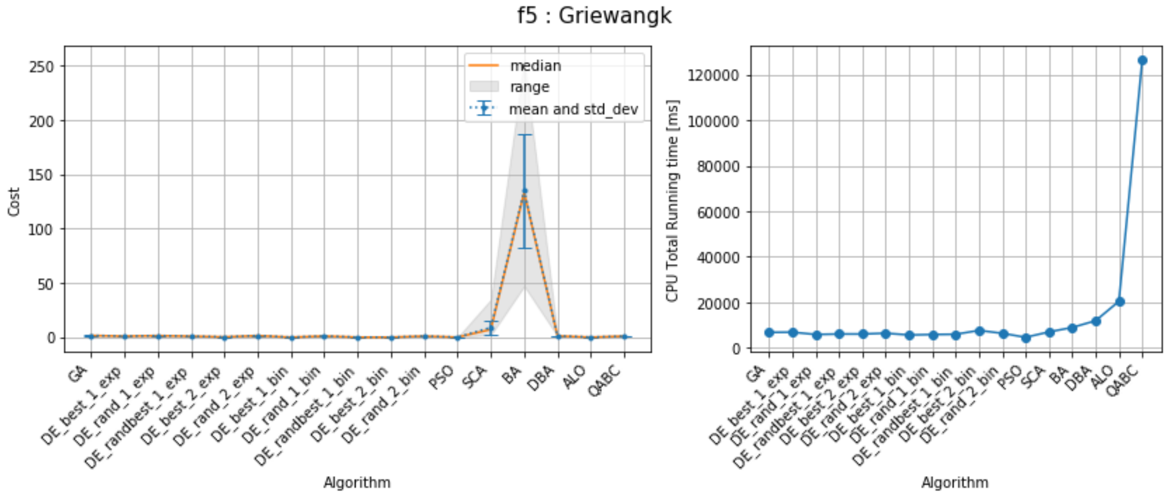

Figure 5: Cost and CPU total running time of Function 5: Griewangk

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	1.611	0.153	1.594	1.351	1.940	6783
DE_best_1_exp	0.989	0.053	0.998	0.778	1.082	6866
DE_rand_1_exp	1.456	0.098	1.447	1.263	1.770	5852
DE_randbest_1_exp	0.958	0.049	0.970	0.812	1.023	6068
DE_best_2_exp	0.296	0.199	0.238	0.039	1.036	6057
DE_rand_2_exp	1.513	0.121	1.491	1.322	1.895	6419
DE_best_1_bin	0.016	0.011	0.014	0.001	0.041	5637
DE_rand_1_bin	1.120	0.029	1.114	1.063	1.203	5831
DE_randbest_1_bin	0.019	0.043	0.011	0.002	0.312	5948
DE_best_2_bin	0.021	0.013	0.017	0.004	0.052	7632
DE_rand_2_bin	1.048	0.011	1.046	1.021	1.070	6377
PSO	0.015	0.014	0.010	0.000	0.051	4603
SCA	8.997	6.173	7.182	2.175	34.793	6977
BA	135.004	52.132	133.620	47.092	255.492	8935
DBA	1.221	0.080	1.199	1.084	1.461	11899
ALO	0.087	0.163	0.020	0.000	0.605	20427
QABC	0.923	0.332	0.951	0.184	2.133	126530

Table 12: Function 5: Statistical Analysis of the Cost

Best Algorithm:
PSO, Cost (mean): 0.015163
Two-Sample Z-Test Hypothesis Testing: confidence interval = 95%

Null hypothesis: The best algorithm and the tested one are equal
DE_best_1_bin, Cost (mean): 0.015512 , P value: 0.891800
DE_randbest_1_bin, Cost (mean): 0.019345 , P value: 0.517100

3.6 Function 6: Sine Envelope Sine Wave

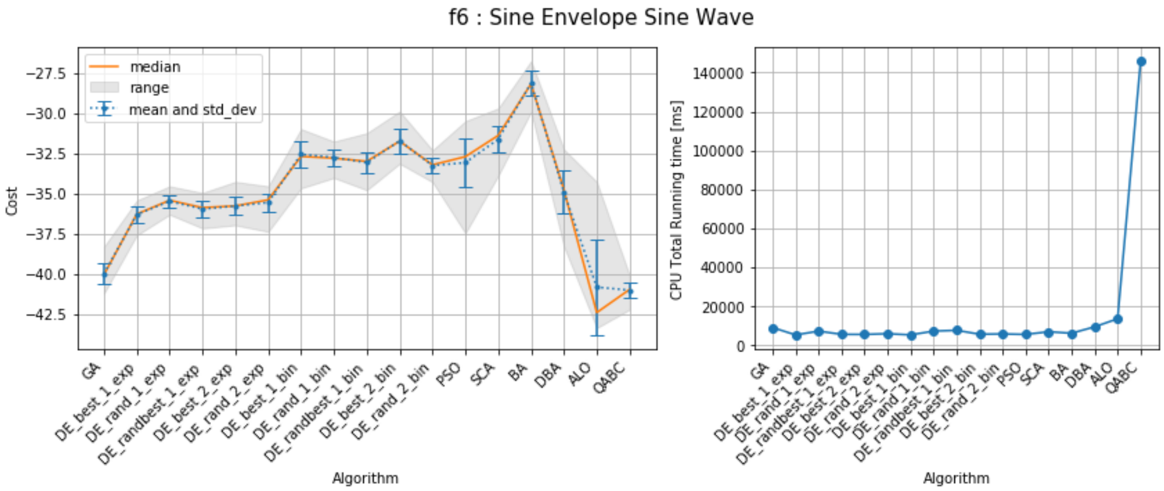

Figure 6: Cost and CPU total running time of Function 6: Sine Envelope Sine Wave

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	-40.002	0.631	-40.067	-41.221	-38.315	8813
DE_best_1_exp	-36.316	0.513	-36.290	-37.601	-35.437	5029
DE_rand_1_exp	-35.490	0.410	-35.431	-36.316	-34.551	6985
DE_randbest_1_exp	-35.974	0.504	-35.888	-37.168	-34.961	5327
DE_best_2_exp	-35.785	0.563	-35.781	-36.975	-34.275	5289
DE_rand_2_exp	-35.566	0.573	-35.403	-37.379	-34.553	5646
DE_best_1_bin	-32.570	0.818	-32.701	-34.677	-30.994	5053
DE_rand_1_bin	-32.765	0.530	-32.811	-34.031	-31.770	6995
DE_randbest_1_bin	-33.086	0.643	-33.008	-34.794	-31.248	7398
DE_best_2_bin	-31.734	0.781	-31.756	-33.147	-29.899	5386
DE_rand_2_bin	-33.268	0.463	-33.225	-34.295	-32.311	5527
PSO	-33.090	1.543	-32.717	-37.473	-30.503	5286
SCA	-31.634	0.819	-31.400	-33.899	-29.706	6667
BA	-28.149	0.781	-28.190	-29.925	-26.771	5898
DBA	-34.925	1.331	-34.773	-38.306	-32.263	9188
ALO	-40.830	2.972	-42.412	-43.367	-34.283	13334
QABC	-41.020	0.498	-40.974	-42.220	-40.105	145932

Table 13: Function 6: Statistical Analysis of the Cost

Best Algorithm:
QABC, Cost (mean): -41.019500
Two-Sample Z-Test Hypothesis Testing: confidence interval = 95%

Null hypothesis: The best algorithm and the tested one are equal
ALO, Cost (mean): -40.829700 , P value: 0.656000

3.7 Function 7: Stretch V Sine Wave

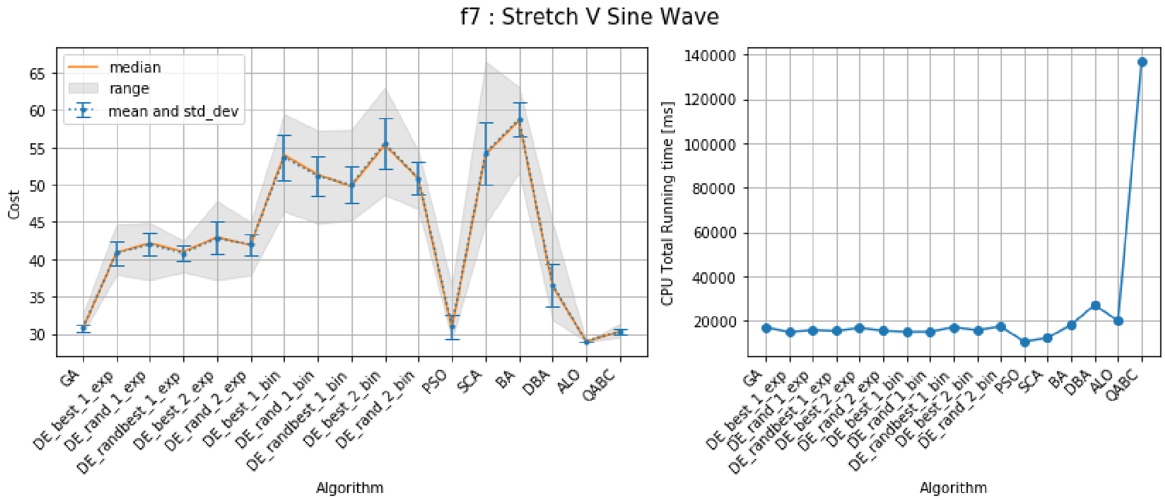

Figure 7: Cost and CPU total running time of Function 7: Stretch V Sine Wave

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	30.777	0.493	30.637	30.098	32.597	17025
DE_best_1_exp	40.850	1.588	40.876	37.916	44.676	15001
DE_rand_1_exp	41.978	1.532	42.163	37.218	44.892	15850
DE_randbest_1_exp	40.816	1.021	41.010	38.282	42.487	15443
DE_best_2_exp	42.859	2.180	42.957	37.208	47.807	16799
DE_rand_2_exp	41.992	1.434	41.902	37.848	45.007	15554
DE_best_1_bin	53.672	3.002	53.978	46.419	59.464	15068
DE_rand_1_bin	51.177	2.676	51.365	44.795	57.246	15111
DE_randbest_1_bin	49.976	2.505	49.775	45.226	57.358	17098
DE_best_2_bin	55.568	3.398	55.301	48.625	63.002	15780
DE_rand_2_bin	50.858	2.114	50.828	46.753	54.518	17416
PSO	31.012	1.611	30.717	29.223	36.431	10737
SCA	54.246	4.195	54.047	44.785	66.503	12351
BA	58.769	2.265	58.588	51.645	63.073	18124
DBA	36.617	2.867	36.403	31.872	44.937	27184
ALO	29.001	0.002	29.000	29.000	29.015	20013
QABC	30.344	0.416	30.303	29.522	31.351	136906

Table 14: Function 7: Statistical Analysis of the Cost

Best Algorithm:
ALO, Cost (mean): 29.000600

3.8 Function 8: Ackley One

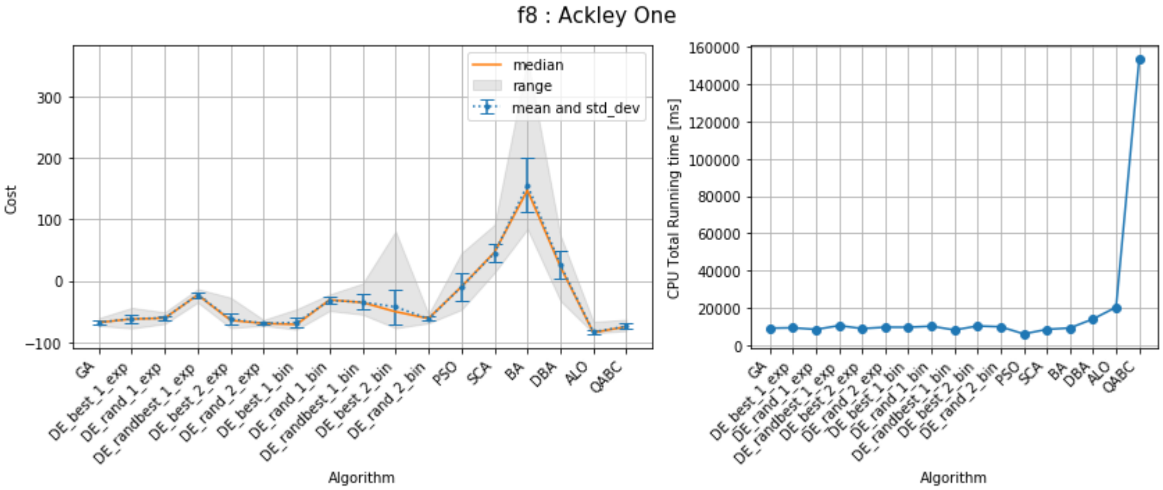

Figure 8: Cost and CPU total running time of Function 8: Ackley One

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	-67.417	3.237	-67.572	-72.879	-60.349	9100
DE_best_1_exp	-61.404	6.466	-61.799	-76.942	-43.532	9300
DE_rand_1_exp	-60.205	3.320	-60.538	-69.782	-50.805	8416
DE_randbest_1_exp	-22.692	4.501	-22.617	-34.842	-13.023	10487
DE_best_2_exp	-61.843	9.586	-63.954	-76.333	-27.033	8880
DE_rand_2_exp	-68.865	2.125	-69.084	-72.787	-63.195	9627
DE_best_1_bin	-67.551	8.129	-70.739	-78.936	-45.748	9595
DE_rand_1_bin	-31.669	5.516	-30.697	-48.526	-21.709	10158
DE_randbest_1_bin	-34.323	12.309	-35.075	-54.537	-4.112	8133
DE_best_2_bin	-42.254	27.942	-49.949	-76.552	80.813	10273
DE_rand_2_bin	-61.550	3.303	-60.960	-68.725	-53.954	9846
PSO	-9.326	23.182	-9.337	-46.594	45.752	6003
SCA	45.392	15.480	46.630	13.460	91.794	8446
BA	156.273	44.356	147.671	84.454	360.876	9197
DBA	26.276	22.895	23.395	-32.472	74.360	14025
ALO	-82.937	3.238	-84.022	-84.926	-66.280	20318
QABC	-73.668	4.353	-74.335	-81.199	-62.754	153446

Table 15: Function 8: Statistical Analysis of the Cost

Best Algorithm:
ALO, Cost (mean): -82.937300

3.9 Function 9: Ackley Two

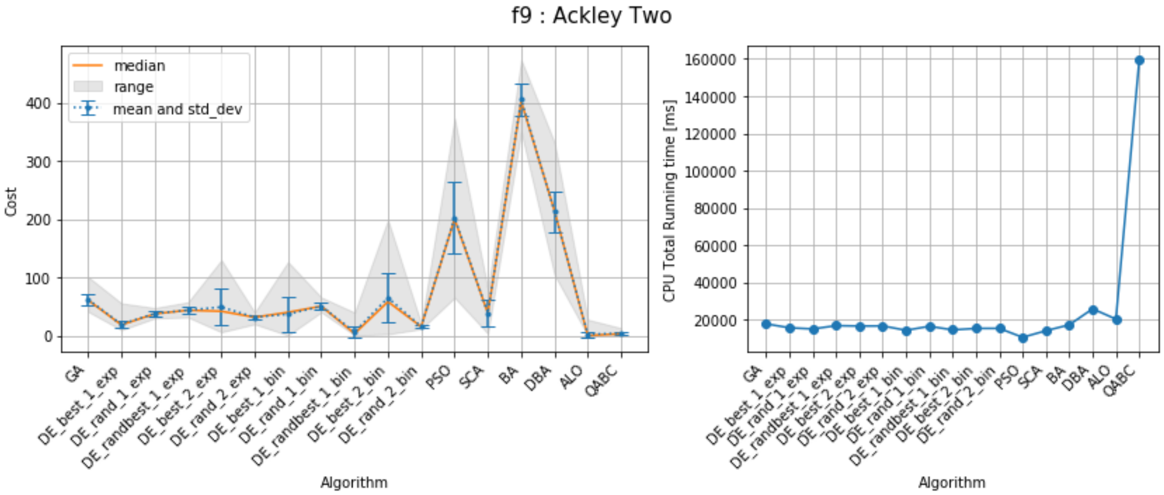

Figure 9: Cost and CPU total running time of Function 9: Ackley Two

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	62.092	9.922	60.943	41.098	100.911	17672
DE_best_1_exp	19.411	7.065	18.310	9.498	55.635	15560
DE_rand_1_exp	37.785	4.202	37.345	29.761	47.481	14931
DE_randbest_1_exp	43.745	5.974	43.491	31.380	57.456	16708
DE_best_2_exp	48.932	31.641	41.736	5.965	129.868	16502
DE_rand_2_exp	31.483	4.309	31.360	19.617	39.945	16549
DE_best_1_bin	36.661	29.576	39.904	0.453	126.648	14219
DE_rand_1_bin	50.290	5.767	50.434	40.195	65.683	16350
DE_randbest_1_bin	6.408	9.460	3.360	0.537	39.694	14443
DE_best_2_bin	64.942	41.876	58.198	2.416	198.798	15218
DE_rand_2_bin	15.539	2.268	15.219	11.294	20.311	15233
PSO	202.614	62.080	200.473	65.056	373.651	10430
SCA	38.528	22.067	38.226	3.317	78.865	13990
BA	405.805	28.293	402.429	350.212	473.726	17139
DBA	212.905	34.588	214.328	104.056	331.856	25744
ALO	1.984	4.520	0.082	-0.000	27.470	20220
QABC	4.167	2.589	3.045	1.101	12.912	159495

Table 16: Function 9: Statistical Analysis of the Cost

Best Algorithm:
ALO, Cost (mean): 1.984420

3.10 Function 10: Egg Holder

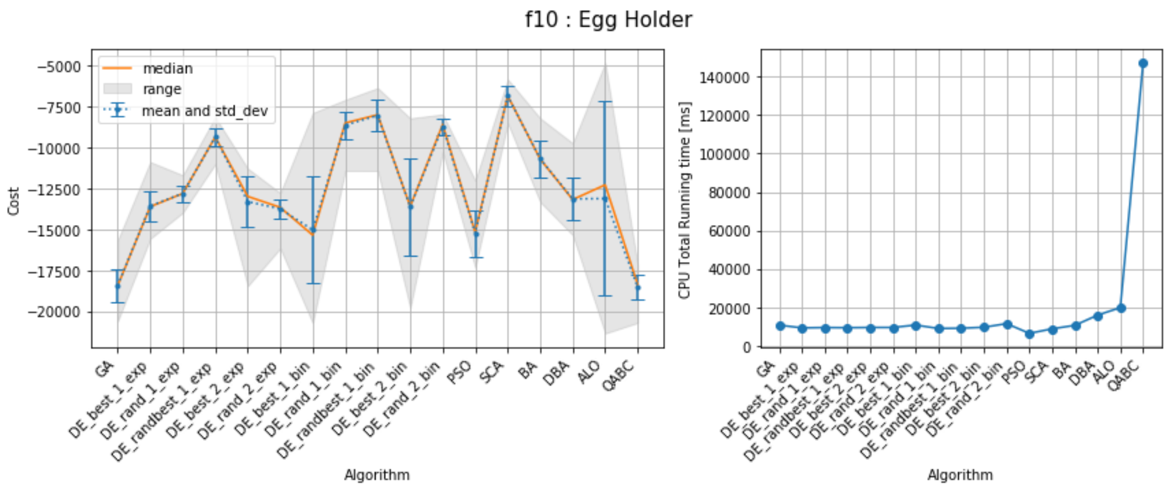

Figure 10: Cost and CPU total running time of Function 10: Egg Holder

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	-18418.900	977.959	-18430.300	-20603.200	-15644.600	10813
DE_best_1_exp	-13557.300	932.259	-13606.300	-15559.500	-10870.300	9376
DE_rand_1_exp	-12797.600	495.665	-12802.800	-13972.600	-11677.300	9545
DE_randbest_1_exp	-9350.060	547.252	-9350.280	-11006.200	-8130.270	9473
DE_best_2_exp	-13295.000	1554.550	-12978.600	-18453.100	-11267.900	9588
DE_rand_2_exp	-13738.500	572.322	-13646.900	-16154.200	-12763.400	9592
DE_best_1_bin	-15022.200	3252.180	-15345.800	-20717.200	-7873.760	10844
DE_rand_1_bin	-8622.710	846.749	-8491.390	-11412.700	-7072.010	9071
DE_randbest_1_bin	-8031.910	932.788	-7987.200	-11409.800	-6350.700	9200
DE_best_2_bin	-13618.200	2955.110	-13594.400	-19798.500	-8200.780	9699
DE_rand_2_bin	-8704.270	509.676	-8581.010	-10260.700	-7955.790	11585
PSO	-15220.500	1420.380	-15160.100	-17326.400	-12000.800	6534
SCA	-6848.860	657.900	-6830.020	-8502.900	-5820.280	8885
BA	-10682.700	1113.500	-10696.700	-13322.600	-8198.770	10692
DBA	-13130.600	1272.350	-13172.900	-15340.400	-9723.900	15998
ALO	-13111.800	5949.390	-12270.900	-21346.200	-4813.060	19904
QABC	-18530.000	762.266	-18362.400	-20697.200	-17170.400	147099

Table 17: Function 10: Statistical Analysis of the Cost

Best Algorithm:
QABC, Cost (mean): -18530.000000
Two-Sample Z-Test Hypothesis Testing: confidence interval = 95%

Null hypothesis: The best algorithm and the tested one are equal

GA, Cost (mean): -18418.900000 , P value: 0.526400

3.11 Function 11: Rana

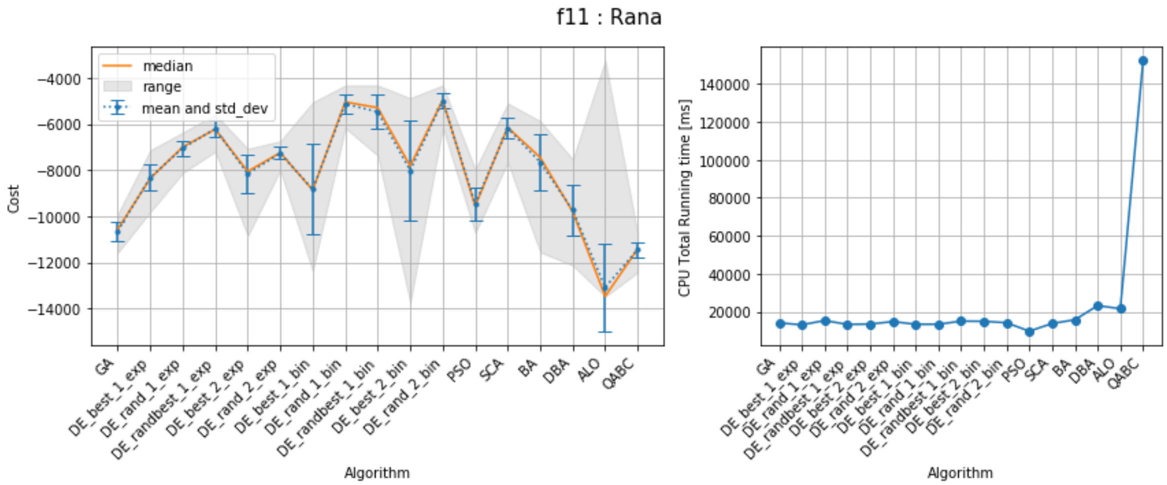

Figure 11: Cost and CPU total running time of Function 11: Rana

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	-10633.800	412.365	-10557.500	-11573.900	-9893.480	14036
DE_best_1_exp	-8311.270	549.006	-8352.470	-9807.410	-7124.900	13007
DE_rand_1_exp	-7039.710	324.882	-6987.830	-8115.000	-6356.910	15284
DE_randbest_1_exp	-6192.800	328.597	-6222.790	-7188.950	-5573.420	13246
DE_best_2_exp	-8152.820	816.937	-8049.310	-10845.000	-7062.040	13423
DE_rand_2_exp	-7262.460	267.475	-7235.960	-7999.080	-6731.420	14738
DE_best_1_bin	-8829.830	1958.180	-8853.740	-12374.700	-5048.090	13307
DE_rand_1_bin	-5121.090	391.404	-5038.050	-6165.390	-4314.580	13332
DE_randbest_1_bin	-5465.440	755.844	-5284.390	-7359.900	-4312.500	14967
DE_best_2_bin	-8011.050	2158.550	-7815.610	-13746.700	-4865.450	14887
DE_rand_2_bin	-4988.320	340.737	-4952.170	-6140.460	-4308.410	14109
PSO	-9466.610	714.160	-9553.660	-10720.100	-7949.880	9662
SCA	-6166.330	448.066	-6138.380	-7660.050	-5093.410	13729
BA	-7650.980	1194.080	-7442.000	-11550.000	-5860.490	15669
DBA	-9727.110	1084.830	-9795.040	-12124.800	-7519.440	23149
ALO	-13077.400	1900.260	-13463.900	-13475.000	-3230.800	21513
QABC	-11433.800	319.177	-11407.600	-12427.000	-10763.800	152489

Table 18: Function 11: Statistical Analysis of the Cost

Best Algorithm:
ALO, Cost (mean): -13077.400000

3.12 Function 12: Pathological

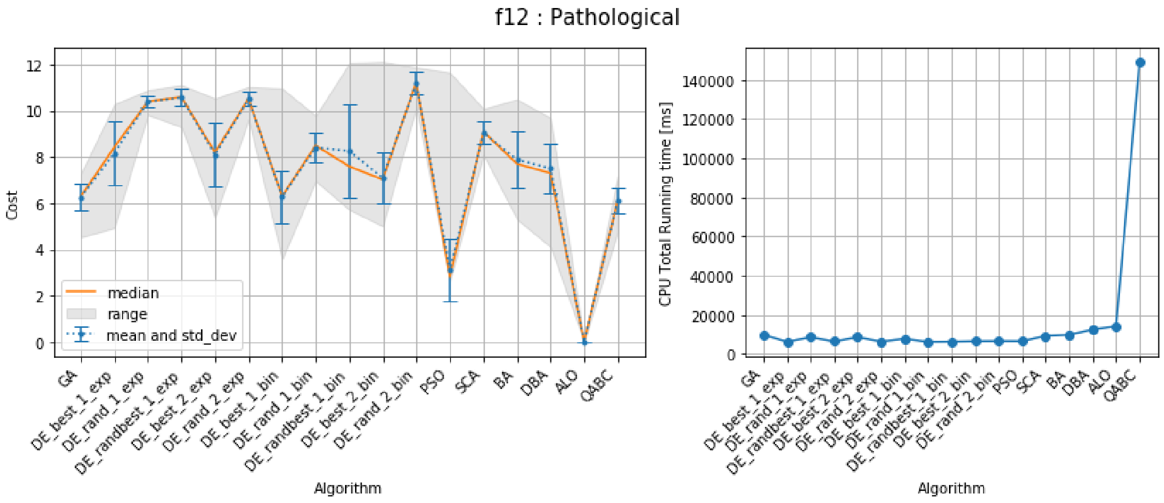

Figure 12: Cost and CPU total running time of Function 12: Pathological

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	6.259	0.570	6.286	4.526	7.279	9851
DE_best_1_exp	8.163	1.391	8.425	4.936	10.298	6374
DE_rand_1_exp	10.394	0.254	10.405	9.828	10.898	8685
DE_randbest_1_exp	10.577	0.353	10.596	9.297	11.115	6506
DE_best_2_exp	8.102	1.360	8.198	5.387	10.546	8719
DE_rand_2_exp	10.504	0.309	10.531	9.621	11.038	6438
DE_best_1_bin	6.280	1.135	6.297	3.581	10.973	7920
DE_rand_1_bin	8.415	0.633	8.502	6.965	9.797	6271
DE_randbest_1_bin	8.263	2.042	7.594	5.717	12.070	6381
DE_best_2_bin	7.099	1.131	7.029	5.009	12.116	6649
DE_rand_2_bin	11.204	0.465	11.243	10.140	11.888	6719
PSO	3.107	1.333	2.806	2.623	11.659	6667
SCA	9.048	0.486	9.099	8.103	10.097	9340
BA	7.888	1.240	7.703	5.305	10.502	9891
DBA	7.505	1.066	7.304	4.105	9.702	12532
ALO	0.000	0.000	0.000	0.000	0.000	14180
QABC	6.100	0.545	6.115	4.750	7.187	149141

Table 19: Function 12: Statistical Analysis of the Cost

Best Algorithm:
ALO, Cost (mean): 0.000001

3.13 Function 13: Michalewicz

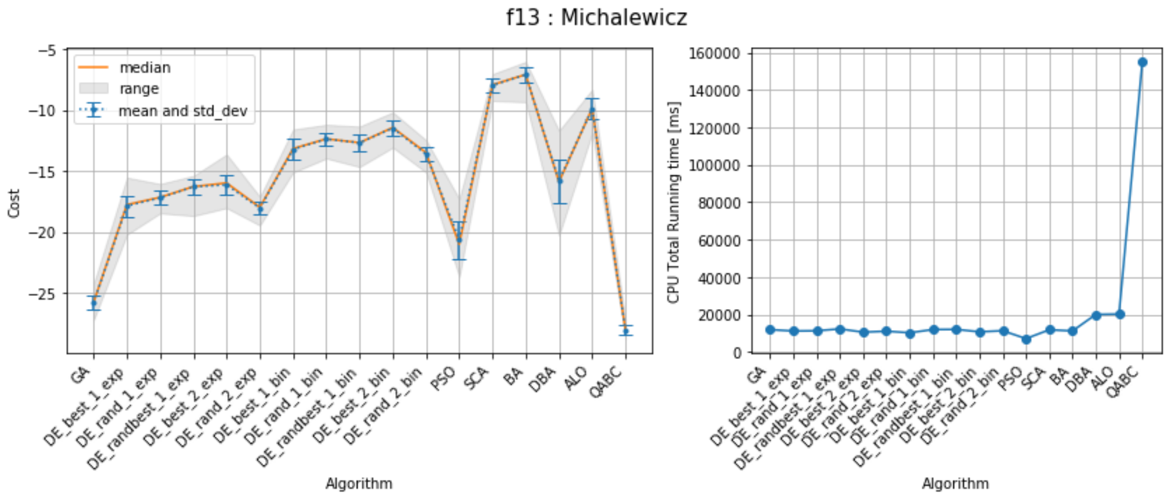

Figure 13: Cost and CPU total running time of Function 13: Michalewicz

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	-25.810	0.601	-25.786	-27.240	-24.254	11969
DE_best_1_exp	-17.869	0.869	-17.765	-20.223	-15.531	11256
DE_rand_1_exp	-17.198	0.552	-17.161	-18.466	-16.053	11372
DE_randbest_1_exp	-16.295	0.586	-16.281	-18.682	-15.389	12364
DE_best_2_exp	-16.132	0.824	-15.981	-18.035	-13.624	10631
DE_rand_2_exp	-18.048	0.537	-17.983	-19.467	-17.103	11158
DE_best_1_bin	-13.217	0.846	-13.156	-15.111	-11.576	10301
DE_rand_1_bin	-12.370	0.540	-12.353	-13.955	-11.186	12037
DE_randbest_1_bin	-12.677	0.645	-12.669	-14.656	-11.331	12112
DE_best_2_bin	-11.469	0.650	-11.424	-13.061	-10.161	10817
DE_rand_2_bin	-13.612	0.546	-13.490	-15.107	-12.452	11397
PSO	-20.638	1.551	-21.022	-23.698	-17.286	7168
SCA	-7.948	0.540	-7.908	-9.224	-7.022	11901
BA	-7.080	0.629	-7.065	-9.328	-6.014	11285
DBA	-15.808	1.778	-15.897	-20.288	-11.674	20091
ALO	-9.869	0.826	-9.844	-11.853	-8.287	20293
QABC	-28.056	0.360	-28.053	-28.779	-27.213	155071

Table 20: Function 13: Statistical Analysis of the Cost

Best Algorithm:
QABC, Cost (mean): -28.056200

3.14 Function 14: Masters’ Cosine Wave

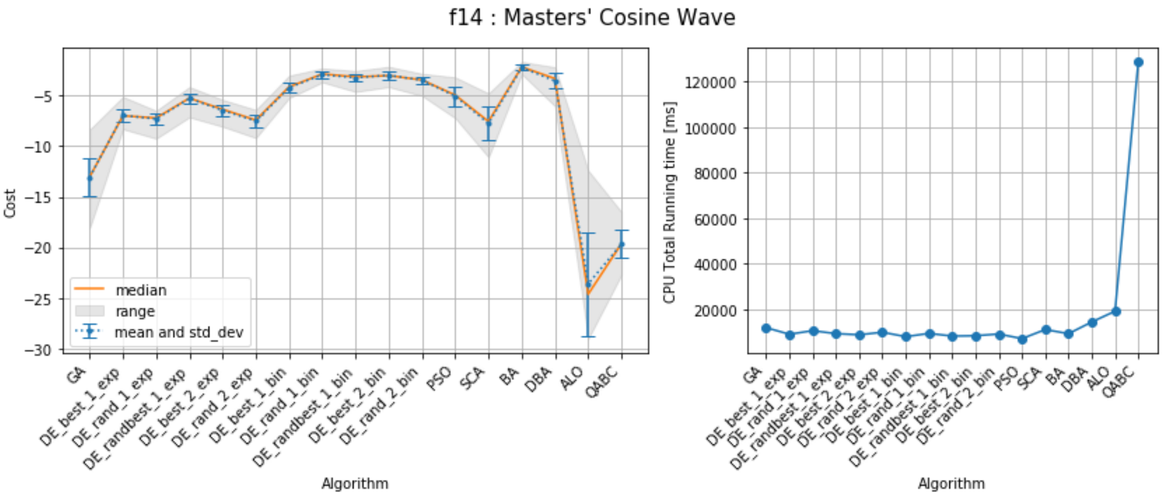

Figure 14: Cost and CPU total running time of Function 14: Masters’ Cosine Wave

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	-13.075	1.867	-13.053	-18.091	-8.366	12017
DE_best_1_exp	-6.978	0.629	-7.010	-8.324	-5.136	9120
DE_rand_1_exp	-7.323	0.559	-7.235	-9.259	-6.508	10677
DE_randbest_1_exp	-5.310	0.526	-5.276	-7.151	-4.169	9396
DE_best_2_exp	-6.480	0.555	-6.379	-8.078	-5.427	8875
DE_rand_2_exp	-7.531	0.615	-7.439	-9.196	-6.425	10009
DE_best_1_bin	-4.237	0.508	-4.167	-5.260	-3.049	8046
DE_rand_1_bin	-2.962	0.344	-2.890	-3.698	-2.330	9476
DE_randbest_1_bin	-3.257	0.396	-3.176	-4.638	-2.583	8318
DE_best_2_bin	-3.016	0.415	-3.051	-4.167	-2.170	8456
DE_rand_2_bin	-3.501	0.306	-3.479	-5.038	-2.860	9215
PSO	-5.079	0.989	-4.976	-7.243	-3.214	7182
SCA	-7.773	1.638	-7.575	-11.095	-4.797	11177
BA	-2.257	0.286	-2.194	-2.899	-1.683	9372
DBA	-3.576	0.762	-3.346	-6.064	-2.214	14502
ALO	-23.593	5.108	-24.618	-29.000	-12.413	19228
QABC	-19.663	1.403	-19.632	-22.697	-16.492	128718

Table 21: Function 14: Statistical Analysis of the Cost

Best Algorithm:
ALO, Cost (mean): -23.593500

3.15 Function 15: Quartic

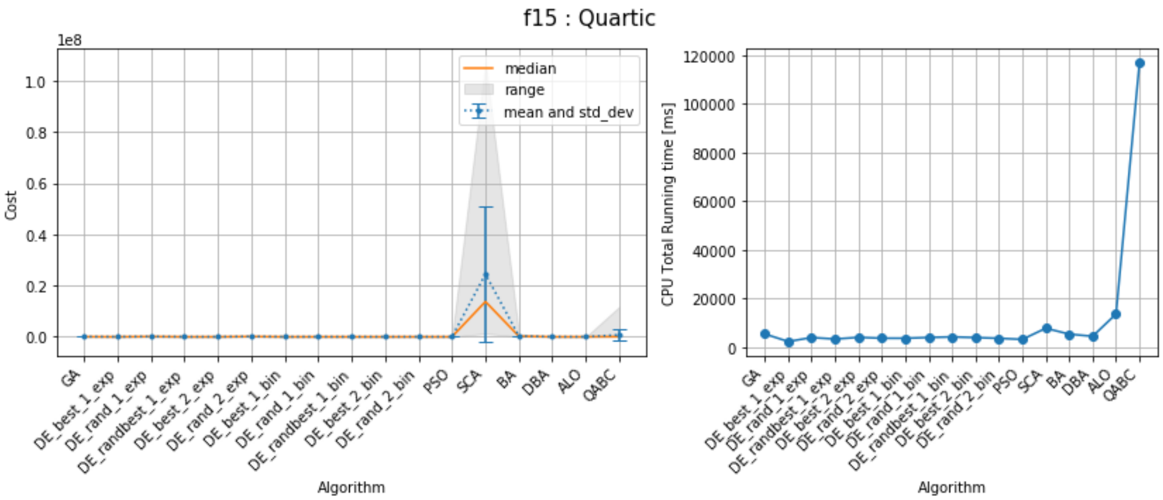

Figure 15: Cost and CPU total running time of Function 15: Quartic

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	46984.900	34978.300	37315.700	6928.200	169604.000	5392
DE_best_1_exp	1314.760	2152.120	542.914	58.099	13120.200	2289
DE_rand_1_exp	94210.600	36630.700	90025.900	36621.700	206680.000	3933
DE_randbest_1_exp	42.515	39.778	32.827	3.452	223.201	3305
DE_best_2_exp	482.483	1665.620	68.901	0.255	11286.400	4008
DE_rand_2_exp	108786.000	44324.900	104730.000	43637.400	233962.000	3648
DE_best_1_bin	0.058	0.191	0.008	0.000	1.218	3634
DE_rand_1_bin	9672.290	4586.410	8980.820	2442.980	21896.800	3953
DE_randbest_1_bin	1.162	4.008	0.004	0.000	25.487	4193
DE_best_2_bin	0.154	0.947	0.005	0.000	6.775	3919
DE_rand_2_bin	4337.030	1810.520	4223.770	1425.340	8816.050	3606
PSO	0.000	0.000	0.000	0.000	0.000	3175
SCA	24485300.000	26339600.000	13739900.000	1240020.000	107132000.000	7749
BA	284920.000	234015.000	205083.000	43519.000	1090270.000	5330
DBA	5711.110	4621.500	4375.370	522.062	24991.700	4446
ALO	0.000	0.001	0.000	0.000	0.011	13599
QABC	744271.000	2048010.000	61002.400	717.118	11782500.000	116950

Table 22: Function 15: Statistical Analysis of the Cost

Best Algorithm:
PSO, Cost (mean): 0.000001
Two-Sample Z-Test Hypothesis Testing: confidence interval = 95%

Null hypothesis: The best algorithm and the tested one are equal
DE_best_2_bin, Cost (mean): 0.154119 , P value: 0.249900
ALO, Cost (mean): 0.000291 , P value: 0.166100

3.16 Function 16: Levy

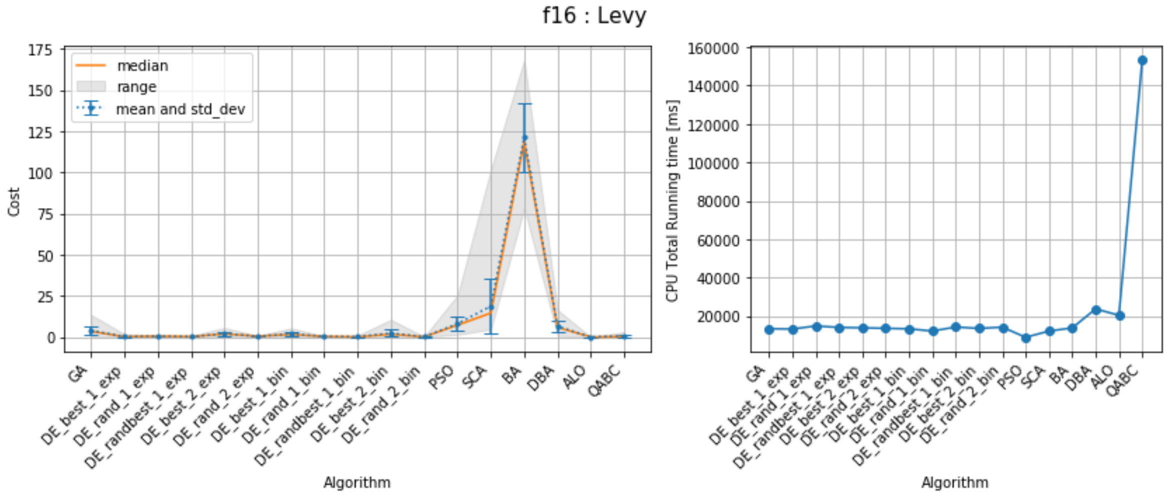

Figure 16: Cost and CPU total running time of Function 16: Levy

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	3.960	2.834	3.458	0.436	13.493	13389
DE_best_1_exp	0.237	0.293	0.163	0.029	1.801	13226
DE_rand_1_exp	0.545	0.093	0.529	0.356	0.866	14818
DE_randbest_1_exp	0.267	0.128	0.248	0.071	0.750	14064
DE_best_2_exp	1.960	1.371	1.863	0.180	5.636	13852
DE_rand_2_exp	0.479	0.110	0.506	0.232	0.791	13570
DE_best_1_bin	1.808	1.276	1.542	0.000	5.365	13335
DE_rand_1_bin	0.378	0.098	0.373	0.172	0.602	12157
DE_randbest_1_bin	0.169	0.220	0.090	0.000	0.998	14286
DE_best_2_bin	2.354	2.053	1.681	0.001	10.544	13525
DE_rand_2_bin	0.109	0.035	0.101	0.065	0.223	14213
PSO	8.061	4.483	7.266	1.453	25.543	8902
SCA	18.736	16.592	14.554	4.699	101.940	12164
BA	121.259	20.811	118.742	78.201	168.417	13816
DBA	6.496	3.249	5.884	0.713	16.514	23699
ALO	0.009	0.007	0.007	0.001	0.038	20387
QABC	0.625	0.823	0.262	0.001	3.122	153395

Table 23: Function 16: Statistical Analysis of the Cost

Best Algorithm:
ALO, Cost (mean): 0.008706

3.17 Function 17: Step

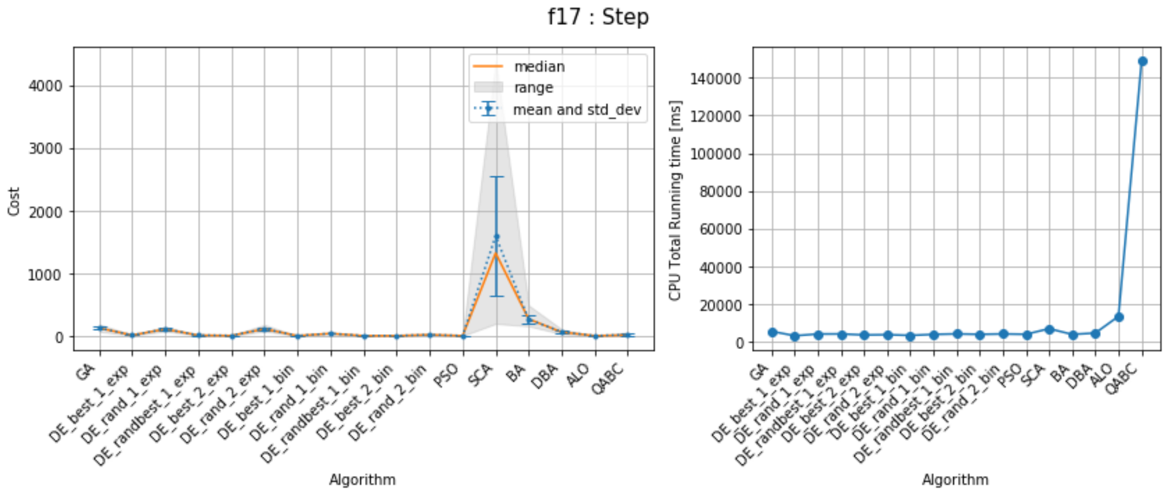

Figure 17: Cost and CPU total running time of Function 17: Step

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	130.398	25.691	129.853	73.077	191.044	5510
DE_best_1_exp	18.129	2.874	17.873	13.210	25.576	3133
DE_rand_1_exp	113.329	18.079	112.506	77.878	152.038	4069
DE_randbest_1_exp	16.601	2.580	15.769	13.297	26.210	4103
DE_best_2_exp	10.418	2.191	9.575	7.957	17.852	3562
DE_rand_2_exp	116.481	21.740	116.022	77.078	183.701	3703
DE_best_1_bin	7.726	0.125	7.688	7.586	8.382	3328
DE_rand_1_bin	43.973	4.971	44.096	34.179	54.382	3858
DE_randbest_1_bin	7.664	0.026	7.659	7.624	7.749	4140
DE_best_2_bin	7.849	0.125	7.813	7.680	8.173	3889
DE_rand_2_bin	24.468	2.811	24.025	18.362	32.973	4115
PSO	7.747	0.393	7.647	7.538	10.265	3918
SCA	1596.840	954.047	1325.510	200.306	4385.700	6904
BA	266.998	65.891	273.425	160.774	491.832	3865
DBA	68.378	14.470	65.440	37.291	111.663	4593
ALO	7.757	0.355	7.633	7.500	8.938	13435
QABC	24.875	14.389	18.739	9.270	60.462	148965

Table 24: Function 17: Statistical Analysis of the Cost

Best Algorithm:
DE_randbest_1_bin, Cost (mean): 7.663680
Two-Sample Z-Test Hypothesis Testing: confidence interval = 95%

Null hypothesis: The best algorithm and the tested one are equal
PSO, Cost (mean): 7.747080 , P value: 0.134400
ALO, Cost (mean): 7.756890 , P value: 0.063800

3.18 Function 18: Alphine

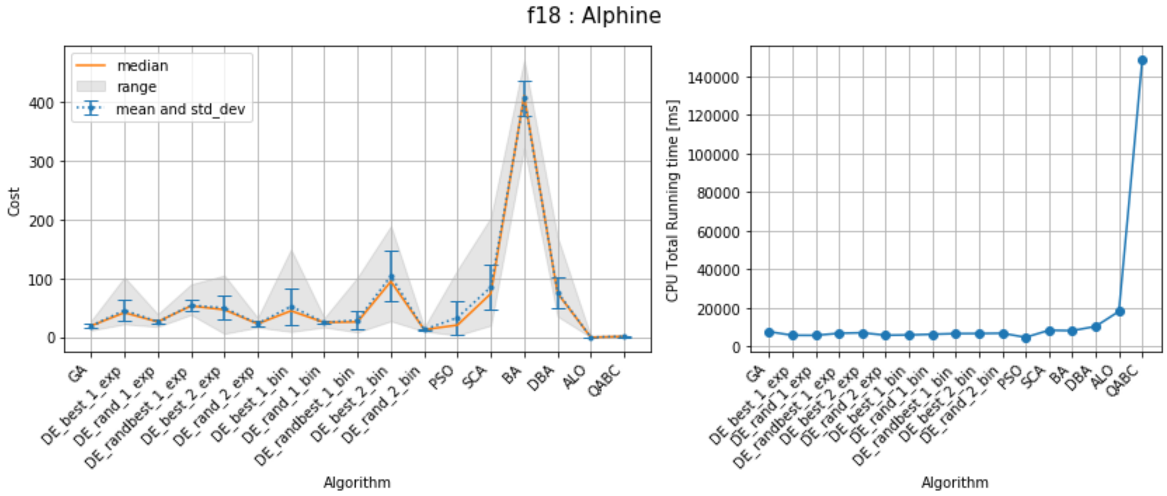

Figure 18: Cost and CPU total running time of Function 18: Alphine

algorithm	mean	std_dev	median	range_min	range_max	time_ms
GA	19.444	3.995	19.402	11.736	28.098	7603
DE_best_1_exp	45.735	16.844	41.709	22.263	101.806	5729
DE_rand_1_exp	26.307	3.822	26.238	17.811	39.302	5690
DE_randbest_1_exp	54.171	10.598	53.053	38.156	89.903	6774
DE_best_2_exp	49.735	20.520	46.743	6.120	104.916	7043
DE_rand_2_exp	23.120	3.796	22.701	16.791	33.174	5752
DE_best_1_bin	52.308	31.146	44.798	9.454	149.314	5909
DE_rand_1_bin	25.169	3.070	25.113	16.846	32.901	6251
DE_randbest_1_bin	29.170	15.816	26.118	8.616	101.965	6616
DE_best_2_bin	103.995	43.388	94.692	27.477	188.903	6656
DE_rand_2_bin	12.923	1.501	12.859	9.859	17.139	6792
PSO	33.375	28.432	20.681	3.310	114.762	4669
SCA	85.157	37.706	74.060	20.093	203.524	8336
BA	406.080	29.382	410.163	321.796	470.690	8132
DBA	76.595	26.062	74.892	35.441	170.203	10336
ALO	0.117	0.282	0.013	0.000	1.718	18135
QABC	1.431	0.864	1.188	0.245	4.504	148632

Table 25: Function 18: Statistical Analysis of the Cost

Best Algorithm:
ALO, Cost (mean): 0.116877

3.19 Summary

The table 27 shows a summary of the best mean cost obtained for each function.

algorithm	f1	f2	f3	f4	f5	f6	f7	f8	f9
GA	1177.430	78.974	338234.000	77.903	1.611	-40.002	30.777	-67.417	62.092
DE_best_1_exp	2575.090	3.904	9718.930	136.816	0.989	-36.316	40.850	-61.404	19.411
DE_rand_1_exp	3161.440	72.772	627829.000	162.908	1.456	-35.490	41.978	-60.205	37.785
DE_randbest_1_exp	6350.210	2.641	1830.630	155.888	0.958	-35.974	40.816	-22.692	43.745
DE_best_2_exp	3607.240	0.517	2690.410	139.609	0.296	-35.785	42.859	-61.843	48.932
DE_rand_2_exp	1856.560	82.140	773060.000	135.850	1.513	-35.566	41.992	-68.865	31.483
DE_best_1_bin	1915.330	0.003	197.640	181.146	0.016	-32.570	53.672	-67.551	36.661
DE_rand_1_bin	6867.390	19.226	68334.800	199.177	1.120	-32.765	51.177	-31.669	50.290
DE_randbest_1_bin	6763.960	0.002	588.583	185.803	0.019	-33.086	49.976	-34.323	6.408
DE_best_2_bin	2282.380	0.006	583.328	226.498	0.021	-31.734	55.568	-42.254	64.942
DE_rand_2_bin	6198.040	7.767	37752.400	165.820	1.048	-33.268	50.858	-61.550	15.539
PSO	3316.760	0.001	60144.600	121.561	0.015	-33.090	31.012	-9.326	202.614
SCA	7260.970	1279.480	233206000.000	352.991	8.997	-31.634	54.246	45.392	38.528
BA	5032.300	186.497	1349370.000	623.680	135.004	-28.149	58.769	156.273	405.805
DBA	3869.680	34.395	44443.000	337.323	1.221	-34.925	36.617	26.276	212.905
ALO	143.976	0.010	4.097	2.744	0.087	-40.830	29.001	-82.937	1.984
QABC	843.371	16.122	4453380.000	19.453	0.923	-41.020	30.344	-73.668	4.167

algorithm	f10	f11	f12	f13	f14	f15	f16	f17	f18
GA	-18418.900	-10633.800	6.259	-25.810	-13.075	46984.900	3.960	130.398	19.444
DE_best_1_exp	-13557.300	-8311.270	8.163	-17.869	-6.978	1314.760	0.237	18.129	45.735
DE_rand_1_exp	-12797.600	-7039.710	10.394	-17.198	-7.323	94210.600	0.545	113.329	26.307
DE_randbest_1_exp	-9350.060	-6192.800	10.577	-16.295	-5.310	42.515	0.267	16.601	54.171
DE_best_2_exp	-13295.000	-8152.820	8.102	-16.132	-6.480	482.483	1.960	10.418	49.735
DE_rand_2_exp	-13738.500	-7262.460	10.504	-18.048	-7.531	108786.000	0.479	116.481	23.120
DE_best_1_bin	-15022.200	-8829.830	6.280	-13.217	-4.237	0.058	1.808	7.726	52.308
DE_rand_1_bin	-8622.710	-5121.090	8.415	-12.370	-2.962	9672.290	0.378	43.973	25.169
DE_randbest_1_bin	-8031.910	-5465.440	8.263	-12.677	-3.257	1.162	0.169	7.664	29.170
DE_best_2_bin	-13618.200	-8011.050	7.099	-11.469	-3.016	0.154	2.354	7.849	103.995
DE_rand_2_bin	-8704.270	-4988.320	11.204	-13.612	-3.501	4337.030	0.109	24.468	12.923
PSO	-15220.500	-9466.610	3.107	-20.638	-5.079	0.000	8.061	7.747	33.375
SCA	-6848.860	-6166.330	9.048	-7.948	-7.773	24485300.000	18.736	1596.840	85.157
BA	-10682.700	-7650.980	7.888	-7.080	-2.257	284920.000	121.259	266.998	406.080
DBA	-13130.600	-9727.110	7.505	-15.808	-3.576	5711.110	6.496	68.378	76.595
ALO	-13111.800	-13077.400	0.000	-9.869	-23.593	0.000	0.009	7.757	0.117
QABC	-18530.000	-11433.800	6.100	-28.056	-19.663	744271.000	0.625	24.875	1.431

Table 26: mean cost of each optimization algorithm with different bench-mark functions. iterations:100 population:200

function_id	best_algorithm	best_cost (mean)	similar_result
1	ALO	143.976	
2	PSO	0.001	
3	ALO	4.097	DE_randbest_1_bin, PSO
4	ALO	2.744	
5	PSO	0.015	DE_best_1_bin, DE_randbest_1_bin
6	QABC	-41.020	ALO
7	ALO	29.001	
8	ALO	-82.937	
9	ALO	1.984	
10	QABC	-18530.000	GA
11	ALO	-13077.400	
12	ALO	0.000	
13	QABC	-28.056	
14	ALO	-23.593	
15	PSO	0.000	DE_best_2_bin, ALO
16	ALO	0.009	
17	DE_randbest_1_bin	7.664	PSO, ALO
18	ALO	0.117	

Table 27: best mean cost of each bench-mark functions

4 Conclusion

In this project many optimization algorithms have been implemented and tested, they are Genetic Algorithm, Differential Evolution Algorithm, Particle Swarm Optimization, Sine Cosine Algorithm, Bat Algorithm, Directional Bat Algorithm, Ant Lion Optimizer and Quick Artificial Bee Colony. In this report we only focus in QABC, the analysis and conclusions about another algorithms can be found in the previous reports. After running QABC for 50 times with the population size of 200 and number of iterations equal to 100 we found that QABC got the best result in 3 of the 18 functions. However, the running time of QABC is much slower than the previous algorithms since QABC need to calculate the euclidean distance in each neighborhood which it is very time consuming.