A COMPREHENSIVE COMPARISON OF EVOLUTIONARY OPTIMIZATION LIMITED BY NUMBER OF EVALUATIONS AGAINST TIME CONSTRAINED

Jia Hui Ong and Jason Teo
Evolutionary Computing Lab, Faculty of Computing and Informatics, Universiti Malaysia Sabah, Sabah, Malaysia
E-Mail: ongkjh@gmail.com

ABSTRACT

In this study, the importance of optimization problems constrained by time is highlighted. Practically all evolutionary optimization studies have focused exclusively on the use of number of fitness evaluations as the constraining factor when comparing different evolutionary algorithms (EAs). This investigation represents the first study which empirically compares EAs based on time-based constraints against number of fitness evaluations. EAs which yield an optimum or near-optimum solutions is crucial for real-time optimization problems. Which EAs are able to provide near optimum solutions in time limited real-time optimization problems has never been answered before. To find out the answer for this question, four well-known and most commonly-used algorithms are tested. Particle swarm optimization (PSO), Differential Evolution (DE), Genetic Algorithms (GA), and Covariance Matrix Adaptation Evolution Strategy (CMA-ES) are tested in three different setups of experiments. A comprehensive and latest global optimization benchmark test suite is used in the form of the CEC 2015 Global Optimization Competition's 15 scalable test problems. The first experiment is to test the performance of these algorithms in expensive benchmark optimization problems that limit the number of fitness evaluations to 50N where N represents the number of optimization dimensions. The second experiment allows these algorithms to run up to the full 10000N evaluations. The last experiment will compare the performance of these algorithms limited by time to 300 milliseconds. The results obtained shows that DE can perform well in the 50N and 10000N evaluation. Critically, we have shown for the first time that in time-limited situations, DE is also the frontrunner by obtaining clearly better results compared to the other three well-known and widely used EAs.

Keywords: evolutionary optimization, time-limited optimization, CMA-ES, DE, GA, PSO, expensive optimization problems.

1. INTRODUCTION

Optimization problems are to find the best solutions for a given sets of problems. In continuous optimization problems, a range of values are allowed to be taken by the variables. GECCO and CEC were among the top conferences where researcher exhibit their works done on solving and finding the best solutions for the given test problems or on a particular optimization problems. The problems given in CEC only focus on finding best solutions and neglecting the time taken to achieve the desired results. In CEC 2014, a competition of real-parameter single objective expensive optimization was held but again the focus of the competition is to achieve the optimum solution although it was called an expensive optimization competition, their focus was on the solutions provided by the algorithms with more dimensions to be solved. The organizers also allows participant to implement surrogates-model to aid their algorithms. As such no clear answer of which algorithms performs better under a certain time frame can be found.

In GECCO 2010 Zhou and Tan [1] presented their work on PSO with triggered mutation, Chen [2] presented PSO with self-adjusting neighbours. Hildebrandt [3] presented the usage of GP in solving the complex shop floor scenarios. Similar to CEC conferences, the main focus of the papers presented is to solve optimization problems by providing the best solutions no matter how much time is taken.

Researchers try to address the problem face in expensive optimization by estimating or approximate the fitness. In fitness approximation, there are 3 popular method, instance-based learning method, machine learning method and statistical learning method. Instance-based method entails transforming the original functions to linear ones, and then using a linear programming technique, such as the Frank-Wolfe method [4] or Powell's quadratic approximation [5]. In machine learning, the techniques available are Clustering, Multilayer Perception Neural Networks and decision tree. Statistical Learning methods for fitness approximation (basically statistical learning models) as applied to EAs have gained much interest among researchers, and have been used in several successful GA packages. In these methods, single or multiple models are built during the optimization process to approximate the original fitness function. These models are also referred to as approximate models, surrogates or meta-models. Among these models, Polynomial Models, Kriging Models, and Support Vector Machines (SVM) are the most commonly used? Although fitness approximation were able to decrease the time of convergence, the question of which algorithms performs the best is a given critical time frame left unanswered. Likewise the focus of fitness approximation is to achieve best solution faster.

Researches that focus on stopping criteria [6], [7], [8] focus on how to stop the optimization process when the solutions reached optimum results. Conventional optimization process use number of evaluations as the termination criteria but it is not practical as the concern of these researches is to save cost and time in real world and expensive optimization problems. Some of the suggestion mentions in these researches are to compare other algorithms with the stopping criteria mention. But still the question of how and what is the performance of PSO, DE and SEA algorithms in a given time frame optimizations problems are not answer.

In expensive optimization problems, researcher address the problems of limited resources and time in running the large number of evaluations in order to obtain the best solutions. Chen [9] used PSO aided MIMO in transceiver design in order to obtain to the best solutions and at the same time lower the computational complexity and time complexity. Vasile and Croisard [10] tackle the space mission design in their work. The main focus of their work is to reduce the time take to compute the space mission design under uncertainty. Researcher work on engineering problems [11], network design [12], word analysis [13], digital circuits [14] all these real-world expensive optimization applications focus on reducing the complexity of the optimizations process. It can be observed that reducing time taken to obtain best solution were the focus of these researchers. This shows how important time is in real world applications. It is crucial to obtain solutions as fast as possible where expensive resources are involved. Yet if the questions of which algorithms that can produce ideal solutions in a given short time frame cannot be answer even though it is observe that time plays an important aspect in real-world optimization problems.

2. METHOD

A. Algorithms

To carry out this study, a review was made based on the existing papers on optimization problems shown in Table-1. Three algorithms were identified to be mostly use in optimization problems; they are particle swarm optimization (PSO), differential evolution (DE), Genetic Algorithms (GA). Another algorithms was chosen which is Covariance Matrix Adaptation Evolution Strategy (CMA-ES) based on papers that claims CMA-ES is among the best algorithms in solving optimization problems.

Table-1. Popular algorithms used in Engineering field.

Algorithms	IEEE	Science direct	Total
GA	2329	1345	3674
PSO	1292	750	2042
Differential Evolution	342	300	642
Ant Colony optimization	255	157	412
Bee optimization	142	95	237
Cuckoo algorithm	65	61	126
Firefly optimization	50	71	121
Bayesian Optimization Algorithms	64	0	64
Bacterial foraging optimization	30	22	52
Artificial immune system optimization	32	12	44

B. Benchmark problems

In CEC 2015, a competition on expensive optimization problems were organized. The benchmark problems used in the competition are used in this study. It comprises from $f1$ to $f15$ benchmark optimization problems as shown in Table-2.

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Table-2. Summary of CEC 2015 expensive optimization problems.

No.	Function	F_i^*
1	Rotated Bent Cigar Function	100
2	Rotated Discus Function	200
3	Shifted and Rotated Weierstrass Function	300
4	Shifted and Rotated Schwefel's Function	400
5	Shifted and Rotated Katsuura Function	500
6	Shifted and Rotated HappyCat Function	600
7	Shifted and Rotated HGBat Function	700
8	Shifted and Rotated Expanded Griewank's plus Rosenbrock's Function	800
9	Shifted and Rotated Expanded Scaffer's F6 Function	900
10	Hybrid Function 1 (N=3)	1000
11	Hybrid Function 2 (N=4)	1100
12	Hybrid Function 3 (N=5)	1200
13	Composition Function 1 (N=5)	1300
14	Composition Function 2 (N=3)	1400
15	Composition Function 3 (N=5)	1500

3. EXPERIMENT SETUP

There are three different setups for this study. The first setup will follow the rules of CEC 2015 expensive optimization competition where the numbers of evaluation are limited to 50N function evaluations. Second setup will follow the standard optimization that allowed maximum number of evaluations to 10000N. The last setup will be on time constrained evaluations. A time threshold is set to 300 milliseconds and once this threshold is reached the algorithm have to stop immediately and the best solution up to that moment are saved. The number of evaluations done in 300 milliseconds was recorded as well, in order to know how many evaluations can be done by these algorithms under 300 milliseconds.

Each algorithm will be tested on the fifteen benchmark optimization problems from f1 to f15 and each function will be run for 51 times. Difference in processor speed will affect the algorithms processing time measurements hence to avoid bias to any algorithms when running time evaluations, all experiments are run on the same PC.

The implementations of algorithms for this study are as follow:

- Covariance Matrix Adaptation (CMA) [13], λ = 4+ 3 log (N), $\mu = \lambda/2$.
- PSO with population size 100, initial velocity 1 and maximum velocity 3.
- Differential evolution, population size 100, Cr = .9, F = .2,
- Genetic Algorithms population size 100, Cr = .5, *Mutation Rate* = .1. Tournament selection, tournament size = 2

4. EXPERIMENT RESULTS

The results obtained are shown in the following tables. In Table-3 shows the average fitness for 50N evaluations. In the 50N evaluations, DE has the best average fitness in nine functions out of the fifteen functions. GA has the best average fitness in $f3$, $f4$, $f8$ *and* $f14$. In $f5$ and $f9$ PSO had shown the best results among the four algorithms. In Table-4 the full evaluations of 10000N are shown. DE again performs the best results in ten functions out of the fifteen functions. GA average fitness in $f4$ and $f14$ are the best among the algorithms while PSO has the best average fitness in $f5$ and $f9$. The average time needed to complete each functions are shown in Tables 5 and 6 for 50N and 10000N respectively. In 50N evaluations the average time taken for CMAES and DE to complete the evaluation is considerably quicker as to compare to the other algorithms but the performance of CMAES are not favourable as shown in Table-3. On the other hand DE processing time is quicker and yet manages to achieve better results from the other algorithms. In 10000N evaluations CMAES processing time are much quicker in twelve functions but the average fitness is not as good as DE.

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Table-3. Average fitness in 50N FITNESS evaluations.

f	CMAES	DE	GA	PSO
1	9.85E+08	3.48E+00	2.32E+06	1.23E+04
2	7.37E+04	7.45E-04	1.14E+04	6.96E+03
3	1.76E+01	7.85E+00	3.23E+00	6.34E+00
4	1.53E+03	1.25E+03	4.98E+00	8.49E+02
5	2.88E+00	1.09E+00	1.30E+00	0.00E+00
6	1.33E+00	1.83E-01	3.11E-01	3.10E-01
7	5.30E+00	2.15E-01	3.67E-01	4.35E-01
8	3.92E+03	2.38E+00	3.28E+00	2.65E+00
9	4.60E+00	3.61E+00	3.39E+00	3.14E+00
10	2.22E+06	2.46E+02	3.40E+04	1.13E+03
11	2.00E+01	2.14E+00	5.18E+00	5.95E+00
12	3.36E+02	3.32E+01	5.56E+01	4.18E+02
13	4.17E+02	3.16E+02	3.18E+02	3.18E+02
14	2.14E+02	1.99E+02	1.97E+02	1.99E+02
15	5.01E+02	1.48E+02	3.77E+02	2.10E+02

Table-4. Average fitness in 10000N fitness evaluations.

f	CMAES	DE	GA	PSO
1	1.71E+08	8.34E-09	6.65E+04	8.18E-09
2	9.79E+03	8.38E-09	3.35E+03	1.26E+01
3	1.67E+01	1.50E-01	1.07E+00	6.84E+00
4	1.99E+03	1.25E+01	3.94E-02	8.29E+02
5	1.06E+00	4.22E-01	6.25E-01	0.00E+00
6	1.20E-01	8.91E-02	1.09E-01	2.56E-01
7	4.44E-01	7.83E-02	1.34E-01	3.84E-01
8	1.13E+00	9.09E-01	1.28E+00	4.52E+00
9	4.61E+00	2.15E+00	2.72E+00	3.13E+00
10	1.25E+04	6.26E+00	1.84E+03	5.46E+02
11	3.61E+00	2.90E-04	1.93E+00	5.02E+00
12	2.77E+02	1.97E+01	2.49E+01	9.62E+01
13	3.66E+02	3.16E+02	3.15E+02	3.14E+02
14	2.05E+02	1.98E+02	1.86E+02	1.99E+02
15	3.45E+02	1.14E+02	2.46E+02	1.73E+02

Table-5. Average time (Milliseconds) used in 50N fitness evaluations.

f	CMAES	DE	GA	PSO
1	339.47	143.57	724.49	160.82
2	277.69	46.98	627.41	61.02
3	1662.45	13883.14	14737.90	14572.02
4	276.55	90.10	668.96	108.22
5	652.71	3669.90	4340.35	2628.27
6	271.16	45.00	627.16	60.75
7	271.12	35.98	618.37	51.53
8	290.88	64.49	647.94	154.71
9	276.90	70.88	651.69	142.67
10	278.78	104.92	683.82	137.73
11	568.49	2899.67	3468.08	2992.20
12	319.45	475.08	1062.22	512.61
13	318.43	490.94	1084.02	512.61
14	316.75	321.31	901.37	391.45
15	1723.75	14234.06	15071.90	15029.39

Table-6. Average time (Milliseconds) used in 10000N fitness evaluations.

f	CMAES	DE	GA	PSO
1	1717.63	277.27	142860.88	5090.04
2	1689.39	75.51	125565.82	5883.73
3	13182.55	334623.55	2872704.76	2877602.71
4	1292.08	15983.76	128483.98	20912.80
5	5216.53	724818.69	850334.10	1986.10
6	2447.31	8466.04	122500.90	10177.65
7	2358.84	6839.37	120671.24	9231.35
8	1341.35	11612.55	127165.55	15735.88
9	1064.63	12895.59	128060.90	22185.65
10	2013.18	18553.41	134361.80	25047.53
11	15901.00	83224.27	684875.33	554659.25
12	4891.67	91444.82	207991.29	94145.98
13	1290.47	96962.04	212442.33	96224.69
14	1049.88	61449.86	177165.51	72533.06
15	25997.02	970541.98	2946076.78	2768691.13

From Table-7, the average fitness for CMAES, DE, GA and PSO running under 300 milliseconds are shown. With just 300 milliseconds DE manage to achieve better results among the four algorithms especially in *f2*. In Table VIII the average number of evaluations is shown for the algorithms running under 300 milliseconds. In a short time span DE manage to have more evaluations done in nine functions out of the fifteen functions. For example in *f4* DE manage to do an average of 1778.41 evaluations are to compare to 449.63(CMAES), 220.71 (GA), and 875.63 (PSO). CMAES manage to more evaluations in six out of the fifteen functions, but the average fitness of CMAES is not as good as DE. The only exception is in *f13* where CMAES manage to get the best average fitness and done more evaluations as to compare to the other three algorithms.

ARPN Journal of Engineering and Applied Sciences

Table-7. Average fitness in 300 milliseconds.

f	CMAES	DE	GA	PSO
1	8.14E+09	8.68E-05	1.07E+07	1.17E+04
2	1.29E+08	0.00E+00	1.54E+04	7.01E+03
3	1.90E+01	1.10E+01	9.26E+00	9.66E+00
4	1.70E+03	9.32E+02	1.76E+01	8.14E+02
5	7.28E+00	1.83E+00	1.92E+00	1.90E+00
6	1.61E+00	9.65E-02	3.58E-01	3.03E-01
7	3.11E+01	8.01E-02	4.38E-01	5.00E-01
8	2.97E+05	1.56E+00	3.72E+00	2.48E+00
9	4.61E+00	3.01E+00	3.54E+00	3.14E+00
10	5.18E+08	1.58E+01	9.33E+04	9.97E+02
11	1.68E+03	5.95E+00	8.16E+00	8.90E+00
12	1.86E+03	4.88E+01	1.32E+02	1.29E+02
13	3.09E+02	3.16E+02	3.20E+02	3.17E+02
14	2.28E+02	1.98E+02	1.99E+02	1.98E+02
15	2.19E+03	4.87E+02	5.00E+02	3.61E+02

Table-8. Average number of fitness evaluations in 300 milliseconds.

f	CMAES	DE	GA	PSO
1	340.04	1135.98	201.29	963.12
2	350.16	3690.88	234.82	2620.65
3	54.57	10.43	9.78	10.00
4	449.63	1778.41	220.71	875.63
5	148.94	41.27	34.33	39.86
6	469.49	3770.78	241.14	3089.51
7	429.51	4713.35	244.49	3188.16
8	398.73	2536.02	234.06	1469.88
9	409.10	2299.31	231.61	831.76
10	365.39	1560.33	180.35	857.84
11	192.92	53.47	25.65	51.12
12	366.39	326.20	133.96	302.43
13	764.89	313.20	137.84	292.92
14	393.43	482.63	166.22	422.24
15	60.39	10.02	9.43	9.98

5. CONCLUSIONS AND FUTURE WORKS

From the results obtain, DE certainly shows promising results as to compare to the other three algorithms that were chosen. Especially in the time constrained evaluations under 300 milliseconds? This finding is critical to the real time optimizations problems as it can answer the questions of which algorithms can perform well under time constraineds evaluations. As in real time optimizations problems, it always requires solution to be given in a short time period, hence DE can be fine tune to try and achieve even better results. The actual real time optimizations problems test with different algorithms needs to be done. Searching for a set of parameter can will allow DE to perform even better in time constraineds evaluations will further increase the performance of DE under critical time evaluations.

ARPN Journal of Engineering and Applied Sciences

REFERENCES

[1] Y. Zhou and Y. Tan. 2010. Particle Swarm Optimization with triggered mutation and its implementation based on gpu. In Proceedings of Genetic and Evolutionary Computation Conference, GECCO. July 07-11, 2010, Portland, Oregon, USA.

[2] Z. Chen, Z. He, C. Zhang. 2010. Particle swarm optimizer with self-adjusting neighborhoods, In: Proceedings of Genetic and Evolutionary Computation Conference, GECCO July 07-11, 2010, Portland, Oregon, USA.

[3] T. Hildebrandt, J. Heger, B. Scholz-Reiter. 2010. Towards improved dispatching rules for complex shop floor scenarios: a genetic programming approach, Proceedings of the 12th annual conference on Genetic and evolutionary computation, July 07-11, 2010, Portland, Oregon, USA.

[4] Frank M.; Wolfe P. 1956. An algorithm for quadratic programming. Naval Research Logistics Quarterly. 3: 95.

[5] Powell M.J.D. 2002. UOBYQA: unconstrained optimization by quadratic approximation. Math. Program. 92, 555-582.

[6] M. Iwan, R. Akmeliawati, T. Faisal, H., M.A.A. Al-Assadi. 2012. Performance Comparison of Differential Evolution and Particle Swarm Optimization in Constrained Optimization. International Symposium on Robotics and Intelligent Sensors.

[7] K. Zielinski, R. Laur. 2007. Stopping Criteria for a Constrained Single-Objective Particle Swarm Optimization Algorithm Informatica 31(1):51-59.

[8] A. Chaudhuri, R. T. Haftka. 2013. A Stopping Criterion for Surrogate Based Optimization using EGO, 10th World Congress on Structural and Multidisciplinary Optimization, Orlando, FL, USA.

[9] S. Chen, W. Yao, H. R. Palally, L. Hanzo. 2010. Particle Swarm Optimisation Aided MIMO Transceiver Designs. Computational Intelligence in Expensive Optimization Problems Volume 2 of Adaptation Learning and Optimization. pp. 487-511.

[10] M. Vasile, N. Croisard. 2010. Robust Preliminary Space Mission Design Under Uncertainty. Computational Intelligence in Expensive Optimization Problems. Vol. 2. pp. 543-570 Adaptation Learning and Optimization; Vol. 2. Berlin.

[11] S. Kumar and K.F. Bauer. 2010. Exploring the use of Lean Thinking and Six Sigma in public housing authorities. The Quality Management Journal. 17(1): 29-46.

[12] J. Mattingley and S. Boyd. 2010. Real Time Convex Optimization in Signal Processing. IEEE Signal Processing Magazine.

[13] W. M. Yamada, T. W. Berger. 2010. Isolated Word Analysis Using Biologically-Based Neural Networks. Computational Intelligence in Expensive Optimization Problems Volume 2 of the Adaptation Learning and Optimization. pp. 637-670.

[14] C. Pilato, D. Loiacono, A. Tumeo, F. Ferrandi, P. L. Lanzi, D. Sciuto. 2010. Speeding-Up Expensive Evaluations in High-Level Synthesis Using Solution modelling and Fitness In heritance. Computational Intelligence in Expensive Optimization Problems.