Proceedings of the International Conference on
Information Technologies (InfoTech-2018)
20-21 September 2018, Bulgaria

SOLVING THE GLOBAL TRAJECTORY OPTIMIZATION PROBLEM WITH ARCHIVE-BASED DIFFERENTIAL EVOLUTION

Vladimir Stanovov, Shakhnaz Akhmedova, Eugene Semenkin

Reshetnev Siberian State University of Science and Technology
e-mails: vladimirstanovov@yandex.ru, eugenesemenkin@yandex.ru, shahnaz@inbox.ru
Russian Federation

Abstract: The paper describes application of differential evolution with modified mutation strategy to the global trajectory optimization problems. The problems are provided by the European Space Agency and represent trajectories of several well-known spacecraft, namely, Cassini, Rosetta and Messenger. Using archive based differential evolution, global best solutions were found for these problems, and the best known solution was found for the Cassini mission.

Key words: Differential evolution, global trajectory optimization, spacecraft trajectory optimization.

1. INTRODUCTION

In the last decades the evolutionary computation has shown itself to be successful in many areas and applications. First of all, evolutionary and swarm optimization techniques appeared to be very effective in solving complicated problems of search and design, where other methods are ineffective. These problems include numerical, integer, binary optimization, as well as permutation problems, such as travelling salesman problem. Moreover, evolutionary algorithms, namely genetic programming is capable of solving complex design problems for symbolic regression, control algorithm generation and many others [1].

One of the important numerical optimization problems is the spacecraft global trajectory optimization problem (GTOP) [2]. The main objective of such here is to find an optimal flight program, so that the spacecraft would achieve the desired orbit or speed and direction with respect to some object. These problems usually contain tens of variables and represent significant problems for classical optimization techniques, because they are often defined as complex programs, involving a lot of computation.

Evolutionary and swarm optimization techniques have been previously applied to such problems, and have shown themselves quite successful. However, the computation resources, required to achieve global optimum is usually unacceptable. In this paper we modify the differential evolution technique by introducing archive-based mutation strategy with selective pressure to boost exploration capabilities of the algorithm. To test the algorithm, we used four global trajectory optimization problems, provided by the European Space Agency (ESA) [3].

The rest of the paper is organized as follows: in section 2 we briefly describe the differential evolution technique used, as well as modifications made, section 3 contains GTOP definition and description. The experimental results and discussion is provided in section 4. Finally, section 5 provides the conclusions.

2. DIFFERENTIAL EVOLUTION AND ARCHIVE-BASED MUTATION

Differential evolution (DE) is a popular evolutionary optimization method proposed by Storn and Price in [4]. Due to its simplicity in implementation, high efficiency and only a few tunable parameters, DE has found various applications in many real-world problems [5].

2.1. Basic algorithm description

DE is a population based method, i.e. when considering the optimization problem

$$F(\boldsymbol{x}) \to \min_{\boldsymbol{x}}$$

where F(**x**) is the optimized function, **x** = $x_1 \ldots x_D$ is the vector of D optimized variables. The DE starts with a population of N vectors, represented as $x_{i,j}$, where i = 1…N, j = 1…D. At the initialization phase, all points are set to random values in the according search range

$$x_{i,j} = x_{lb,j} + rand(0,1) \cdot (x_{ub,j} - x_{lb,j})$$

where $x_{ub,j}$ and $x_{lb,j}$ are upper and lower boundaries for variable j.

The main DE cycle contains application of mutation, crossover and selection steps to the population of N solutions for G generations. The mutation operator is the main DE search operator; the classical DE mutation is called DE/rand/1:

$$v_{i,j} = x_{r1,j} + F \cdot (x_{r2,j} - x_{r3,j})$$

where r1, r2 and r3 are mutually exclusive random indexes from [1, N], F is the scaling factor, and $v_{i,j}$ is the mutant vector. The mutation operator makes use of the distance between random population vectors to generate new data points. Note that r1, r2 and r3 are generated only once for every mutant vector generation, not for every vector component j.

The next step is crossover, which combines the genetic information of two vectors to generate the trial vector. The crossover operation is performed with probability Cr. In addition, the jrand index is introduced, which is set as rand(1,D) to make sure that at least one variable from the mutant vector is present in the trial vector:

$$t_{i,j} = \begin{cases} v_{i,j}, rand(0,1) < Cr \; or \; j = jrand \\ x_{i,j}, otherwise \end{cases}$$

The last step is the selection: if the trial vector's fitness value $F(t_{i,j})$ is lower than the one of the i-th individual in the population $F(x_{i,j})$, then this individual is replaced.

2.2. Control parameters adaptation and modifications

The two main parameters of DE are the scaling factor F and crossover rate Cr. They influence the convergence speed and thus tuning them accordingly allows balancing the exploration and exploitation capabilities.

There are many various DE parameter adaptation techniques existing in the literature, including jDE [6], JADE [7], SaDE [8], SHADE [9] and many others, which differ in their basic ideas and effect. In this paper we will adapt the tuning method used for SHADE algorithm, namely the Success History based Adaptation with 5 memory cells containing promising F and Cr couples, which are updated by weighted Lehmer mean procedure. The successful F and Cr values for updating are recorded after every improvement of the fitness function. The initial values for F and Cr are set to 0.5, and new F values are generated by Cachy distribution with scale parameter equal to 0.1, while new Cr values are generated by Normal distribution with standard deviation of 0.1, as proposed in [9].

The mutation strategy for solving GTOP problems was modified to enhance the exploration capabilities of the algorithm by adding the archive set. The archive set was used in the SHADE algorithm in the current-to-pbest/1 mutation strategy, however, here we modify the rand/1 strategy so that

$$v_{i,j} = x_{r1,j} + F \cdot (x_{ra2,j} - x_{ra3,j})$$

where r1 is the random index in [1, N], while ra2 and ra2 are random indexes drawn from [1,N+NA], where NA is the archive set. The archive set is filled by potentially good solutions: every time a vector is updated by selection procedure, the individual, that is replaces by the trial vector, is copied to the archive of size NA. Initially, the archive is empty, and it keeps growing as the algorithm proceeds. When the archive is full, the new inserted vector replaces random old vector in the archive.

In addition to this, the indexes ra2 and ra3 are selected in accordance to the rank-based selection procedure. More precisely, the ranks are assigned to the total set of N individuals and NA archive vectors based on their fitness values, where better fitness gets larger rank

$$Rank_i = (N + NA - i) + 1,\ i = 1 \ldots N + NA$$

$$pr_i = \frac{Rank_i}{\sum_{i=1}^{N+NA} Rank_i}$$

The ra2 and ra3 are then generated using discrete distribution, where the probability pr_i of a vector to be chosen is proportional to its rank.

3. SPACECRAFT GLOBAL TRAJECTORY OPTIMIZATION

The trajectory optimization problems are very difficult to solve due to the fact that function calculation involves the computation of a complex dynamical system. Usually, such problems are formulated so that the objective function value to be optimized is the amount of propellant, needed to achieve the desired goal. While the optimized values are usually planet visiting times, however, the trajectories, obtained by this method are usually not very good. A more complicated method is to allow deep space maneuvers (DSM), which gives the opportunity to get better trajectories, but complicates the problem: the number of local minima increases, as well as the search space sensitivity. Various methods were proposed, and DE appeared to be quite promising in solving this class of problems [10].

Here we will consider four GTO problems provided by ESA, namely Cassini-1, Cassini-2, Rosetta and Messenger. Each of these problems are available [11] to optimize the flight trajectories of well-known missions.

First of all, Cassini-1 and Cassini-2 are two variants of the Cassini spacecraft mission to reach Saturn, the difference is that Cassini-2 allows DSM, and the goal is to fly-by Saturn, while Cassini-1 goal is to reach its orbit. The Cassini-1 6 parameters are presented in Table 1.

Table 1.

Variable	***Lower bound***	***Upper bound***	***Units***
t0	-1000	0	MJD2000
T1	30	400	days
T2	100	470	days
T3	30	400	days
T4	400	2000	days
T5	1000	6000	days

The Rosetta mission’s goal is to reach 67P/Churyumov–Gerasimenko comet and explore it, sending the Philae module. Both Cassini-2 and Rosetta have 22 parameters due to DSM allowed.

The Messenger mission’s goal is to reach Mercury orbit and stay on it. The problem definition has 18 parameters. Same as for other problems, the optimization goal is to minimize the amount of propellant required.

4. EXPERIMENTAL RESULTS

The program system used for testing was written using the C++ language and ran on Ubuntu 16.04 LTS. For every problem tested, 25 runs of the algorithm have been performed. For all tests, the population size was set to 5000, and the total computational resource was set to $D \cdot 10^6$. For Messenger GTOP, the computational resource was set to $5 \cdot D \cdot 10^6$.

The best found solution for Cassini-1 problems, as well as known alternative solution is presented in Table 2.

Table 2.

Variable	***Best found solution (this study)***	***Manfred Stickel, Max-Planck-Institut fuer Astronomie (PSO)* [12]**	***Difference***
t0	-789.77	-789.81	0,04
T1	158.317	158.302	0,015
T2	449.386	449.386	0
T3	54.7115	54.7489	0,0374
T4	1024.74	1024.36	0,38
T5	4552.89	4552.30	0,59
DeltaV	4.93071	4.9307	0.00001

As can be seen from the table, the solution found by the modified DE is almost the same as the known solution for this problem.

For Cassini-2 problem, the best goal function value was 8.382940, while other solutions are 8.383091 (G. Danoy, B. Dorronsoro, P. Bouvry, University of Luxemburg), and 8.383184 (M. Schlueter, J. Fiala, M. Gerdts, University of Glasgow, University of Birmingham) [12]. So, the modified DE was capable of finding the best known solution to this problem up to date, although the difference is not large. The goal function values close to 8.383 were obtained in 4 runs out of 25, i.e. 12%.

For Rosetta problem, the best goal function value was 1.343, while other solutions are 1.343 (M. Vasile, E. Minisci, University of Glasgow), and 1.344 (B. Addis, A. Cassioli, M. Locatelli, F. Schoen, Global Optimization Laboratory, University of Florence) [12]. Moreover, this solution was found in 40% of the cases, i.e. this problem was relatively simple to solve for modified DE.

The Messenger GTOP appeared to be most complicated, and for this problem only suboptimal solution was found, with goal function value of 8.652. Other studies reported 8.630 (F. Biscani, M. Rucinski and D.Izzo, European Space Agency) and 8.631 (B. Addis, A. Cassioli, M. Locatelli, F. Schoen , Global Optimization Laboratory, University of Florence and University of Turin), while the third result is only 8.703 (T., Vinko, D., Izzo, European Space Agency) [12].

Considering the results, obtained by modified differential evolution algorithm, we may say that it can be successfully used to solve this class of global trajectory optimization problems, and achieve the goal in a limited time. The time required to perform one run of the algorithm is: for Cassini-1: 5 minutes, Cassini-2: 10 minutes, Rosetta: 10 minutes, Messenger: 60 minutes. For comparison, to get the same result on Cassini-2, the Mixed Integer Distributed Ant Colony Optimization (MIDACO) solver required 50 days of search, and Hybrid Hierarchical cellular Genetic Algorithm (HH-cGA) required 42 hours [13].

5. CONCLUSION

In this paper we have proposed a modification of the differential evolution algorithm for solving the global trajectory optimization problems for several spacecraft. For three out of four problems, optimal solutions were found, while for the Messenger problem, only suboptimal solution was discovered. This is probably due to the limited computational resource or algorithm settings. However, for Cassini-2 problem, the best known up-to-date solution was found in a relatively small time.

Thus, we may conclude that the differential evolution with modified archive-based mutation strategy with selective pressure is an efficient optimization method, applicable to solving various real-world optimization problems.

REFERENCES

[1] Das, S., Mullick, S.S., Suganthan, P.N. (**2016**) Recent Advances in Differential Evolution – an Updated Survey, Swarm and Evolutionary Computation, Vol. 27, pp. 1–30.

[2] Addis, B., Cassioli, A., Locatelli, M., and Schoen F (**2011**). Global Optimization for the Design of Space Trajectories. Computational Optimization and Applications, 48(3), pp. 635-652.

[3] Izzo, D. (**2010**). Global Optimization and Space Pruning for Spacecraft Trajectory Design, Spacecraft Trajectory Optimization, Conway, B. (Eds.), Cambridge University Press, pp.178-199.

[4] Storn, R., Price, K., (**1997**). Differential Evolution – a Simple and Efficient Heuristic for Global Optimization over Continuous Spaces, Journal of Global Optimization. Vol. 11(4).

[5] Neri, F., Tirronen, V. (**2009**). Recent advances in differential evolution: a survey and experimental analysis. Artif Intell Rev (2010) 33:61–106.

[6] Brest, J., Greiner, S., Boskovic, B., Mernik, M., Zumer, V. (**2006**). Selfadapting control parameters in differential evolution: A comparative study on numerical benchmark problems, IEEE Trans. Evol. Comput., vol. 10, no. 6, pp. 646–657.

[7] Zhang, J. Sanderson, A. C. (**2009**). JADE: Adaptive differential evolution with optional external archive," IEEE Trans. Evol. Comput., vol. 13, no. 5, pp. 945–958.

[8] Qin, A. K., Huang, V. L., Suganthan, P. N. (**2009**). Differential evolution algorithm with strategy adaptation for global numerical optimization, IEEE Trans. Evol. Comput., vol. 13, no. 2, pp. 398–417.

[9] Tanabe, R., Fukunaga, A. (**2013**). Evaluating the performance of SHADE on CEC 2013 benchmark problems, in: Proceedings of the 2013 IEEE Congress on Evolutionary Computation (CEC), IEEE, pp. 1952–1959.

[10] Islam, Sk. M. Das, S. Ghosh, S. Roy, S. Suganthan, P. N. (**2012**). An adaptive differential evolution algorithm with novel mutation and crossover strategies for global numerical optimization. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 42.2, 482-500.

[11] Musegaas, P., (**2012**). Optimization of Space Trajectories Including Multiple Gravity Assists and Deep Space Maneuvers. Thesis Report, Delft University of Technology, 256 p.

[12] Izzo, D. Vinkó, T. (**2005**). Global trajectory optimization database, (available at https://www.esa.int/gsp/ACT/inf/projects/gtop/gtop.html)

[13] Danoy, G., Dorronsoro, B., Bouvry, P. (**2012**), New State-of-the-art Results for Cassini2 Global Trajectory Optimization Problem, Acta Futura vol. 5, pp. 65-72.