Massively Parallelized Co-evaluation for Many-Objective Space Trajectory Optimization

Martin Schlueter
Information Initiative Center,
Hokkaido University, Sapporo, Japan.
schlueter@midaco-solver.com

Masaharu Munetomo
Information Initiative Center,
Hokkaido University, Sapporo, Japan.
munetomo@iic.hokudai.ac.jp

ABSTRACT

This contribution presents numerical results for optimizing a many-objective space mission trajectory benchmark under consideration of massively parallelized co-evaluation of solution candidates. The considered benchmark is the well-known *Cassini1* instance published by the *European Space Agency* (ESA) extended to four objectives. The MIDACO optimization software represents an evolutionary algorithm based on Ant Colony Optimization (ACO) and is applied to solve this benchmark with a varying fine-grained parallelization factor (**P**) ranging from one to 1024. It can be shown that the required number of sequential steps to solve this benchmark can be significantly reduced by applying massive parallelization, while still maintaining a sufficient high number of well distributed non-dominated solutions in the objective space.

CCS CONCEPTS

• **Theory of computation → Evolutionary algorithms**;

KEYWORDS

Many-Objective Optimization, Parallelization, Cassini1, ACO

ACM Reference Format:
Martin Schlueter and Masaharu Munetomo. 2018. Massively Parallelized Co-evaluation for Many-Objective Space Trajectory Optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference 2018 (GECCO '18 Companion).* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3205651.3205674

1 INTRODUCTION

Space mission trajectory design is a challenging and active area for applying optimization algorithms. Since 2005 the Advanced Concept Team (ACT) of the European Space Agency (ESA) publishes a database of Global Trajectory Optimization (GTOP) benchmarks [3] formulated as single-objective optimization problems.

The easiest and most widely used instance of the GTOP set is the *Cassini1* problem, which consist of six variables and four constraints (see [3] for details). This contribution considers a many-objective extension of this benchmark, which was introduced in Schlueter et al. [5] and which consists of four objectives. Table 1 list those four objectives together with their description and units.

GECCO '18 Companion, July 15–19, 2018, Kyoto, Japan

ACM ISBN 978-1-4503-5764-7/18/07. . . $15.00.
https://doi.org/10.1145/3205651.3205674

Table 1: Four Objectives for Cassini1 Benchmark

Objective	Name	Unit
F1	Total ΔV (including ΔV_∞)	Km/Sec
F2	Time of Flight	Days
F3	Launch Date	MJD2000
F4	Launch ΔV_∞	Km/Sec

Ant Colony Optimization (ACO) in general and the MIDACO software in particular has been shown to be efficient for trajectory design optimization, see for example [1], [4] or [6]. This contribution investigates the impact of massively parallelized co-evaluation of solution candidates on the overall number of sequential steps (called *"Blocks"*, see Section 2.1 in [7] for details) required to solve the *Cassini1* benchmark to its best-known solution.

Besides the expected reduction of sequential steps by applying massive parallelization, this contribution is also concerned with the impact of parallelization on the amount and distribution of non-dominated solutions among the four-dimensional objective space. It can be shown, that even for massive parallelization the presented approach delivers a sufficient large and well distributed set of non-dominated solutions.

2 NUMERICAL RESULTS AND CONCLUSIONS

This section presents numerical results of applying a beta version of MIDACO 6.0 on the four-objective *Cassini1* benchmark. The parallelization factor **P**, which defines the amount of parallel processed solution candidates within one sequential algorithmic step, is varied from one to 1024, with unit steps on a *log_2* scale (see first column of Table 2). For each value of **P**, 30 individual test runs are conducted, each using a different random-seed and using the original lower bounds as starting point. An individual test run is considered successful and stopped, if the best known value (4.9307) in the first objective (F1) is reached within a precision of 0.1%. Note that 0.1% is the official required precision upon which ACT/ESA considers the benchmark to be solved. In regard to the many-objective nature of this problem, the first objective is set as overall target function to be minimized while the remaining three objectives are only filtered for pareto-dominance. This is done because the first objective is the most complex and critical one and it provides a stopping criteria in regard to a global best known solution.

Table 2 lists the numerical results for the best out of 30 test runs and the average results in regard to the required number of sequential steps (called *"Blocks"*) and overall required function evaluation (called *"Eval"*).

Table 2: Best and Average Results for different P

	Best run		Average	
P	*Blocks*	*Eval*	*Blocks*	*Eval*
1	255,017	255,017	**1,114,440**	1,114,440
2	175,815	351,630	**947,207**	1,894,414
4	120,180	480,720	**785,786**	3,143,145
8	45,035	360,280	**176,586**	1,412,695
16	68,058	1,088,928	**156,488**	2,503,819
32	106,382	3,404,224	**142,003**	4,544,125
64	4,409	282,176	**116,850**	7,478,434
128	1,868	239,104	**93,457**	11,962,525
256	3,922	1,004,032	**90,172**	23,084,202
512	3,300	1,689,600	**51,711**	26,476,424
1024	2,322	2,377,728	**26,654**	27,293,866

From Table 2 it can be seen that the average number of *Blocks* can be significantly reduced from **1,114,440** in the serial case (**P**=1) to **26,654** in the massively parallelized case of **P**=1024. In other words: While the MIDACO algorithm required about 1.1 million sequential function evaluation in the serial case to reach the best-known solution in high precision, the same solution could be reached within only about 26 thousand blocks of sequential function evaluation, whereas each such block contained 1024 individual function evaluation. The *"Eval"* column in Table 2 shows the number of total function evaluation corresponding to each **P**. In the serial case, the best run required 255,017 function evaluation in total to reach the best known solution.

From Table 2 it can be further seen that the number of *Blocks* for the best out of 30 runs shows a non-monotonic behavior and significant variance (e.g. 106.382 *Blocks* for **P**=32 versus only 4,409 *Blocks* for **P**=64). This great variance is explained by the highly non-linear nature of the objective landscape, which implies a strong dependence on the random-seed used for each individual test run.

The overall best test run was reached for a parallelization factor of **P**=128 and it required only 1,868 sequential steps. Figure 1 and Figure 2 display the final set of non-dominated solutions respectively for the best run of the **P**=1 and **P**=1024 case. Note that while Figure 2 shows visibly less non-dominated solutions than Figure 2, it still captures the most relevant trade-off part of the front between the total ΔV (F1) and flight time (F2). The large difference in the algorithmic behavior between the serial and massively parallelized case is also well observable by the quite different scattering of the set of the last 30,000 evaluation. In Figure 1 and Figure 2 the position of the individual MIDACO solution is highlighted as semi-transparent green hexagon. Note that the symbol size of the non-dominated solution is varied in the plots to illustrate the corresponding launch date.

Future research may further investigate the impact of parallelization by applying rigorous multi-objective optimization performance measures, like the hypervolume indicator (HV) and the inverse generalized distance (IGD) measure. Also, a larger number of test runs is desired in order to counter-measure the high variance observed by the results of the individual best runs in the second column of Table 2.

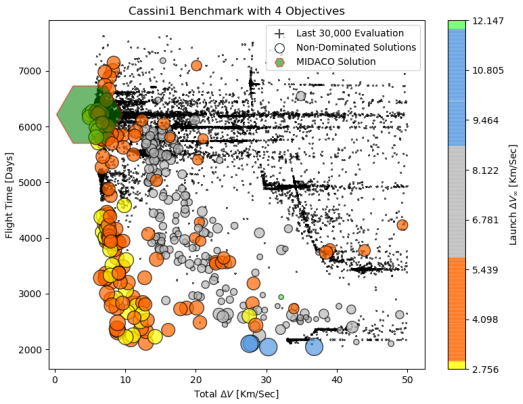

Figure 1: Solution illustration of best run for P=1

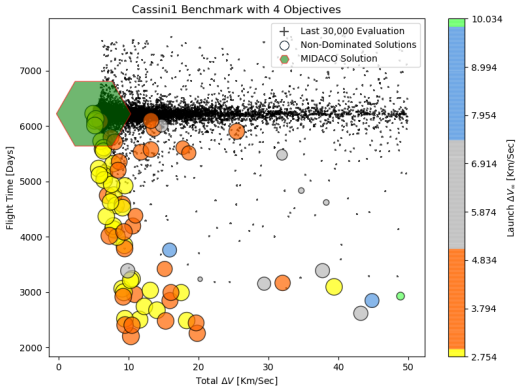

Figure 2: Solution illustration of best run for P=1024

REFERENCES

[1] Ceriotti, M.: Global Optimisation of Multiple Gravity Assist Trajectories. Ph.D. Thesis, Univ. of Glasgow, Glasgow, Scotland, U.K., 2010.
[2] Izzo, D.: Global Optimization and Space Pruning for Spacecraft Trajectory Design. *Spacecraft Trajectory Optimization* Conway, B. (Eds.), Cambridge University Press, pp.178-199, 2010.
[3] European Space Agency (ESA) and Advanced Concepts Team (ACT). Gtop database - global optimisation trajectory problems and solutions, Software available at http://www.esa.int/gsp/ACT/inf/projects/gtop/gtop.html, 2013.
[4] Schlueter M., Munetomo M.: Parallelization Strategies for Evolutionary Algorithms for MINLP. Proceedings of the IEEE Congress on Evolutionary Computation (IEEE-CEC), 635 – 641, 2013.
[5] Schlueter M., Yam C.H., Watanabe T., Oyama A.: Parallelization Impact on Many-Objective Optimization for Space Trajectory Design. Int. J. Mach. Learn. Comput., 6(1), pp. 9 - 14, 2016.
[6] Schlueter M., Erb S., Gerdts M., Kemble S., Rueckmann J.J.: MIDACO on MINLP Space Applications. Advances in Space Research, 51(7), pp. 1116-1131, 2013.
[7] Schlueter M., Munetomo M.: MIDACO parallelization scalability on 200 MINLP benchmarks. Journal of Artificial Intelligence and Soft Computing (De Gruyter), Vol. 7, Issue 3, pp. 171-181, 2017.