

ADVANCES IN GLOBAL OPTIMISATION FOR SPACE TRAJECTORY DESIGN

Dario Izzo

25th International Symposium on Space Technology and Science
(E-mail: dario.izzo@esa.int)

Abstract

We describe, in the framework of global optimisation, four trajectory optimisation problems, representative of three different important typologies often encountered in space mission design. After giving a brief description of the astrodynamical details necessary to implement the problems we solve them by applying Differential Evolution and Multiple Particle Swarm Optimisation keeping the number of objective function evaluations to a minimal amount to ensure low computational times. We show how these techniques may be usefully exploited to perform fast optimisations getting first estimates of relevant mission design parameters thus helping to take early decisions on the overall spacecraft design.

1 Introduction

Finding the best way to transfer a spacecraft between planets or orbits is a complex problem. The traditional methodologies used to design the spacecraft trajectories have evolved, in the last years, by taking advantage of a number of emerging new techniques that promise to radically simplify the task of finding optimal solutions [2]. The principal theoretical step that has to be taken in order to understand many of the new approaches is that of considering the trajectory design problem as a global optimisation problem. While this does not change the issue in its nature, it allows to account for the complex topology of the search spaces commonly encountered. Known issues limiting the efficiency of the various methods are the number of local min-

ima, their clustering around small portions of the search space and the small basin of attraction of the real global optimum typical of these kinds of problems [5]. Several studies have focused on the use of different numerical techniques to search efficiently the solution space and several algorithms have been tested and proved to be suitable in connection with trajectory design problems. The European Space Agency, and in particular its Advanced Concepts Team is carrying on a project aimed at coding and testing several different global optimisation strategies for a number of mission design problems that are of interest to the aerospace community. Genetic, evolutionary, ant search, simulated annealing, particle swarm, branch and bound algorithms have all been coded and tested in different problems, from the multiple gravity assist to the more complex problem of low-thrust trajectories, deep-space manoeuvres and weak stability boundary transfers. During this project, brought forward in collaboration with European Universities thanks to the Ariadna scheme [Ariadna is an ESA initiative created to stimulate research on enabling space research areas and on the development of new design methods. Subjects include theoretical physics, power systems, propulsion, trajectory design and optimisation, informatics and applied mathematics, biomimetics, and other subjects in which both space systems engineering competence and specific theoretical knowledge are required], several theoretical advances have been achieved. Most noteworthy, the Multiple Gravity Assist problem, with powered swing-by, has been shown to be characterised by a search space that may be opportunely pruned in polynomial time allowing for the branches to be searched by heuristic techniques with a drastic increase in reliability [3]. This allows to design an algorithm that solves the MGA prob-

*Copyright © 2006 by Dario Izzo. All rights reserved.
Published by the Japan Society for Aeronautical and Space Sciences and ISTS with permission.

lem, locating the global optimal solution in wide launch windows in a handful of seconds. The search space may thus be explored with an unprecedented accuracy and efficiency for this particular kind of problem. Unfortunately, whenever low-thrust arcs are considered, or deep space manouvres are included, the pruning technique developed is not applicable and we are still left with an NP-hard (non polynomial complexity) problem.

In this paper we apply two global optimisation techniques, resulted to be the best in the comparison performed during the Ariadna study [5], to four different trajectory design problems. representative of the following typologies: multiple gravity assist without deep space manouvres, multiple gravity assist with deep space manouvres, low-thrust arcs transfer via exponential sinusoids representation [6, 1]. The focus is here in trying to obtain fast preliminary results on complex problems that would otherwise require greater computational efforts to be solved.

Note that the problems here faced have been chosen for their interest in connection with the testing of global optimisation techniques rather than for their interest as trajectories. As a consequence they should be regarded as benchmark problems to test global optimisation techniques and have a mere academic interest. The problems have been coded in MATLAB and C++ and have been made available on-line in form of black box functions that can be downloaded from the ACT web pages [www.esa.int/gsp/ACT/mis-sion_analysis/BlackBoxProblems.htm].

2 The generic global optimisation problem

The generic global optimisation problem we consider is:

$$\begin{aligned} \text{find:} \quad & \mathbf{x} \in \mathcal{I} \\ \text{to maximise:} \quad & J(\mathbf{x}) \\ \text{subject to:} \quad & \mathbf{g}(x) < 0 \end{aligned} \quad (1)$$

where \mathcal{I} is an hyperrectangle in \mathbb{R}^N . Once we manage to write our trajectory optimisation in this rather generic form (this also constitutes the subject of a lively and interesting research field) we may solve the problem applying a global optimisation technique to it. In particular we here consider standard implementations of Differential Evolution

(DE) [7] and of Multiple Particle Swarm Optimisation (MPSO) [4]. These have already been proven to be quite efficient in connection to a number of trajectory optimisation problems [5]. To read a detailed description of the implementation of these algorithms used in this paper the reader may see [3]. The upper and lower limits on the state variables have been dealt with by replacing each component out of the bounds with a random number within the bounds. The inequality constraints have been implemented as linear penalty functions.

We will use throughout the paper a naming convention for transfers that use capital letters for the planets, and small letters for the manouvres (d=deep space manouvre, exp=exponential sinusoid arc). An EVVEJdS, for example, would be an Earth-Venus-Venus-Earth-Jupiter - deep space manouvre - Saturn transfer.

3 The EdVdM trajectory problem

Let us consider here an Earth-Venus-Mars transfer with deep space manouvres. To properly transcribe this problem in the form of Eq.(1) we must find a set of state variables \mathbf{x} that are unambiguously related to each possible trajectory that can be flown. Starting from the Earth, we let the starting date t_E , the outgoing planetocentric velocity velocity V_∞ and its direction to be free parameters. We need to choose two variables to describe the direction of the \mathbf{V}_∞ vector. We choose $u, v \in [0, 1]$, related to \mathbf{V}_∞ by the following equations:

$$\begin{aligned} \theta &= 2\pi u \\ \cos \varphi &= 2v - 1 \\ \mathbf{V}_\infty &= V_\infty [\cos \theta \sin \varphi \hat{\mathbf{i}} + \sin \theta \cos \varphi \hat{\mathbf{j}} + \cos \varphi \hat{\mathbf{k}}] \end{aligned}$$

This description is particularly useful in connection with the implementation of the global optimisers as it allows to select a uniformly distributed direction by randomly selecting u, v in their range. This will ensure that the populations initialized at random by the global optimisation algorithms will be uniformly distributed in the physical space. If one considered φ and θ as state parameters, an out of the ecliptic departure would be statistically unfavored when randomly selecting $\varphi \in [0, 2\pi]$ and $\theta \in [0, \pi]$. Once these four parameters are selected one may evaluate the trajectory leg that departs from the Earth. Next we must chose another parameter that

determines when the deep space manouvre will be applied. We chose $DSM1 = \Delta M/2\pi \in [0, N_1]$, the normalized mean anomaly difference between the Earth departure and the deep space manouvre along the first trajectory leg. By letting N_1 to be larger than one we may thus easily account for multiple revolutions. The next state variable is T_{2V} expressing the time (in days) from the deep space manouvre to the Venus encounter. Now we need to describe the geometry of the Venus planetocentric hyperbola. We need two more variables, namely $H_V = 1/r_p \in [0, 1/r_{pmin}]$ and $\zeta \in [0, 2\pi]$. These describe the rotation of the relative velocity via the relations:

$$\delta = \arcsin 1/e(H_V)$$

$$V_{out} = V_{in}[\cos(\delta)\hat{\mathbf{b}}_1 + \sin \zeta \sin \delta \hat{\mathbf{b}}_2 + \cos \zeta \sin \delta \hat{\mathbf{b}}_3]$$

where $e(H_V)$ is the functional relationship between the eccentricity of the planetocentric hyperbola and the variable H_V given by $e = 1 + V_{in}^2/H_V \mu_V$. The projection frame $\hat{\mathbf{b}}_i$ is defined by $\hat{\mathbf{b}}_1 = \mathbf{V}_{in}/V_{in}$, $\hat{\mathbf{b}}_2 = \hat{\mathbf{b}}_1 \wedge \mathbf{r}_V/r_V$ and $\hat{\mathbf{b}}_3 = \hat{\mathbf{b}}_2 \wedge \hat{\mathbf{b}}_1$ as to ensure that no singularities are possible. As we are now able to determine the heliocentric outbound trajectory from Venus, we only need two more variables to complete our description: $DSM2 \in [0, N_2]$ and T_{2M} . Our state vector will than be $\mathbf{x} = [t_E, V_\infty, u, v, DSM1, T_{2V}, H_V, \zeta, DSM2, T_{2M}]$.

As a numerical example let us consider the following:

$$\begin{aligned} \text{find:} \quad & \mathbf{x} \in \mathcal{I} \\ \text{to minimise:} \quad & V_\infty + \Delta V_1 + \Delta V_2 + \Delta V_{arr} \quad (2) \\ \text{subject to:} \quad & r_p > r_{pmin} \end{aligned}$$

where $\mathcal{I} = [0, 2000] \times [0, 7] \times [0, 1] \times [0, 1] \times [0, 2] \times [0, 400] \times [0, 1/8000] \times [0, 2\pi] \times [0, 2] \times [0, 500]$, $\Delta V_{1,2}$ represent the magnitude of the necessary deep space manouvres, V_∞ is the hyperbolic escape velocity, ΔV_{arr} is the arrival relative velocity. All in all we have a global optimisation problem of dimension $D = 10$ with $N_c = 1$ non linear constraints. DE and MPSO were applied for 50 times, allowing for a maximum of 30000 function evaluations in each run. The results, in terms of best and worse solution returned, together with the mean and the standard deviation σ of all the runs, are shown in Table 1. A visualization of the best tra-

Table 1: Results of 50 test on the EdVdM. 30000 function evaluations have been allowed, units are in km/s

Algorithm	best	worse	mean	σ
DE	8.154	10.13	9.000	0.431
MPSO	8.684	11.44	10.19	0.696

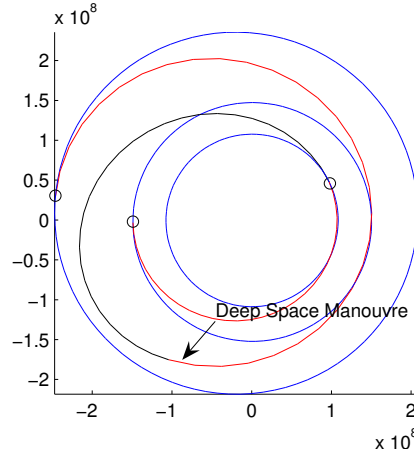


Figure 1: Best solution found for the EdVdM problem.

jectory found is given in Figure 1. For this optimal solution $V_{inf} = 3.11$ km/s, $\Delta V_1 = 0$ km/s, $\Delta V_2 = 2.92$ km/s and $\Delta V_{arr} = 2.12$ km/s.

4 The EdEdMdM trajectory problem

We here add some complexity to the previous problem and we consider an Earth-Earth-Mars-Mars transfer with deep space manouvres. In this case three deep space manouvres and two flybys have to be considered: using similar symbols as in the previous section, the state vector becomes $\mathbf{x} = [t_E, V_\infty, u, v, DSM1, T_{2V}, H_V, \zeta_E, DSM2, T_{2M}, H_M, \zeta_M, DSM3, T_{2arr}]$. The problem dimension is $D = 14$ with $N_c = 2$. In particular we consider the following:

$$\begin{aligned} \text{find:} \quad & \mathbf{x} \in \mathcal{I} \\ \text{to minimise:} \quad & \Delta V_1 + \Delta V_2 + \Delta V_3 + \Delta V_{arr} \quad (3) \\ \text{subject to:} \quad & \mathbf{r}_p > \mathbf{r}_{pmin} \end{aligned}$$

Table 2: Results of 50 test on the EdEdMdM. 20000 function evaluations have been allowed, units are in km/s

Algorithm	best	worse	mean	σ
DE	3.571	5.482	4.953	0.479
MPSO	4.94	8.042	6.169	0.713

where $\mathcal{I} = [0, 2000] \times [0, 1] \times [0, 1] \times [0, 1] \times [0, 2] \times [0, 500] \times [0, 1/7000] \times [0, 2\pi] \times [0, 2] \times [0, 600] \times [0, 1/6800] \times [0, 2\pi] \times [0, 2] \times [0, 600]$ and $\Delta V_{1,2,3}$ represent the magnitude of the necessary deep space manouvres. Note that the starting C3 is forced to be less than one. DE and MPSO are again applied for 50 times, allowing for a maximum of 20000 function evaluations for each run. The results, in terms of best and worse solution returned, together with the mean and the standard deviation σ of all the runs, are shown in Table 2. A visualization of the best trajectory found is given in Figure 2. For this particular solution the optimiser essentially returns an Earth-Mars-Mars transfer with $V_{inf} = 1$ km/s, $\Delta V_1 = 2.38$ km/s, $\Delta V_2 = 0.697$ km/s and $\Delta V_{arr} = 0.48$ km/s. It is particularly interesting to note how, from the numerical output of the global optimisation, we are able to determine that the first Earth-Earth transfer does not help in this case. It is as if the optimiser is able to realize that the fly-by sequence proposed is not optimal and eliminates the first redundant fly-by. This, unfortunately, is a rather special case and in general terms the problem of the selection of the best strategy remains still unsolved.

5 The EexpM trajectory problem

Let us consider an Earth-Mars transfer with low-thrust propulsion. It is possible to transcribe this problem in the form of Eq.(1), and to keep the problem dimension to $D = 3$, at the cost of using a simplified representation of the trajectory in terms of exponential sinusoids [1]: a trajectory shape first proposed by Petropoulos [6]. With reference to the quoted works we consider the state vector $\mathbf{x} = [t_E, T_{2M}, k_2]$ containing the departure date, the time of flight to Mars and one shape parameter.

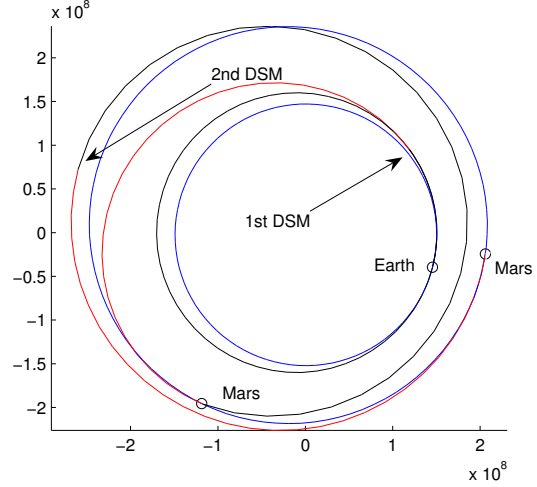


Figure 2: Best solution found for the EdEdMdM problem.

Table 3: Results of 50 test on the EexpM problem. 3000 function evaluations have been allowed, units are in km/s

Algorithm	best	worse	mean	σ
DE	7.315	7.369	7.332	0.0129
MPSO	7.336	7.723	7.452	0.114

The optimisation problem considered is:

$$\begin{aligned} \text{find: } & \mathbf{x} \in \mathcal{I} \\ \text{to minimise: } & \Delta V_{dep} + \Delta V_{lt} + \Delta V_{arr} \end{aligned} \quad (4)$$

where $\mathcal{I} = [2000, 5000] \times [200, 2000] \times [0.01, 1]$, ΔV_{dep} , ΔV_{arr} represent the boundary condition violations typical of the exponential sinusoids and ΔV_{lt} is the low-thrust contribution. Each algorithm was run for 50 times allowing for a maximum of 3000 function evaluations. The results, in terms of best and worse solution returned, together with the mean and the standard deviation σ of all the runs, are shown in Table 3. A visualization of the best trajectory found is given in Figure 3. For this optimal solution $\Delta V_1 = 1.936$ km/s, $\Delta V_2 = 0.944$ km/s, $\Delta V_{lt} = 4.438$ km/s. The maximum acceleration required was $a_{max} = 1.343e-4$ m/s² confirming a known difficulty of the exponential sinusoids to meet possible constraints on the thrust level.

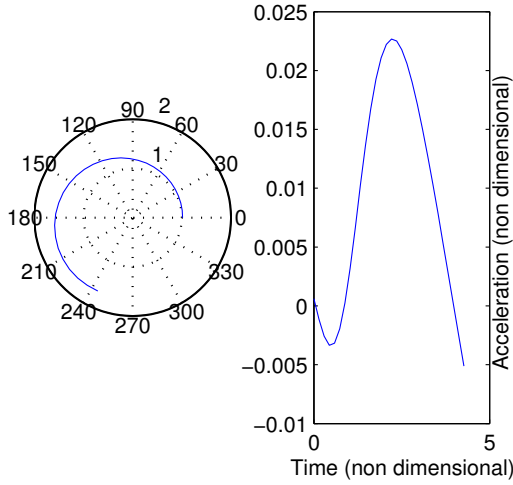


Figure 3: Best solution found for the EexpM problem.

6 The EVEVEJSA trajectory problem

This problem was inspired by the 1st ACT Global Trajectory Optimisation competition. This was a competition organized by ESA's Advanced Concepts Team to find the best trajectory that would impact the asteroid 2004 TW229 maximising the subsequent asteroid semi-major axis change [2]. We considered only ballistic trajectories with powered swing-bys. We took a 1500 kg spacecraft and an $I_{sp} = 2500$ s (while these data make no sense for a chemical propelled spacecraft they were used in the competition where a low-thrust transfer was considered). We used the same launch window and constraints given by the competition problem and we did not implement any constraint on the various ΔV given at the planets. We also selected a fixed fly-by sequence, Earth-Venus-Earth-Venus-Earth-Jupiter-Saturn-Asteroid, inverting the orbit angular momentum at Saturn.

Using standard astrodynamical routines (mainly a good Lambert's solver is required) it is possible to code this problem in the form of Eq.(1). This results in:

$$\begin{aligned} \text{find:} \quad & \mathbf{x} \in \mathcal{I} \\ \text{to maximise:} \quad & m_f \mathbf{U} \cdot \mathbf{v}_{ast} \\ \text{subject to:} \quad & \mathbf{r}_p > \mathbf{r}_{pmin} \end{aligned} \quad (5)$$

where \mathbf{x} contains the departure date from the Earth

Table 4: Results of 50 test on the EVEVEJSA. 60000 function evaluations have been allowed, units are in $kg \ km^2/s^2 \ 10^6$

Algorithm	best	worse	mean	σ
DE	1.656	0.526	1.140	0.220
MPSO	1.006	0.027	0.271	0.255

and the various time-of flights to the following encounter. The mass of the spacecraft at impact is m_f , the relative impact velocity \mathbf{U} and the asteroid heliocentric velocity \mathbf{v}_{ast} . The vector \mathbf{r}_p contains the peri-apses of the various planetocentric hyperbolas that have to be larger than the values contained in \mathbf{r}_{pmin} (the values used are those given in the problem description of the 1st ACT Global Trajectory Optimisation competition). We therefore have a global optimisation problem of dimension $D = 8$ with $N_c = 6$ constraints. As this is a multiple gravity assist problem, it is possible to use the pruning technique described in [3] to obtain a problem that can be solved in polynomial time. As we wanted to treat the problem as a blackbox, we here did not here exploit that result.

Each algorithm was run for 50 times allowing for a maximum of 60000 function evaluations. The results, in terms of best and worse solution returned, together with the mean and the standard deviation σ of all the runs, are shown in Table 4. A visualization of the best trajectory found is given in Figure 4.

7 Discussion

In general terms the advantage of the use of global optimisation algorithms in relation with trajectory problems stems entirely on the computational speed, on the simple implementation and on the possibility of getting a large number of different trajectories the designer can later choose from taking into account many different criteria. On the other hand whenever the problems gets very complicated, often the global optimum is missed.

As a matter of fact for the EVEVEJSA problem none of the two algorithms was able to locate, with the given settings, the best solution known so far to this particular problem that is $J = 1.844E6$ as reported in [www.esa.int/gsp/ACT/mission_analysis/EVEVEJSA.htm]. This could be

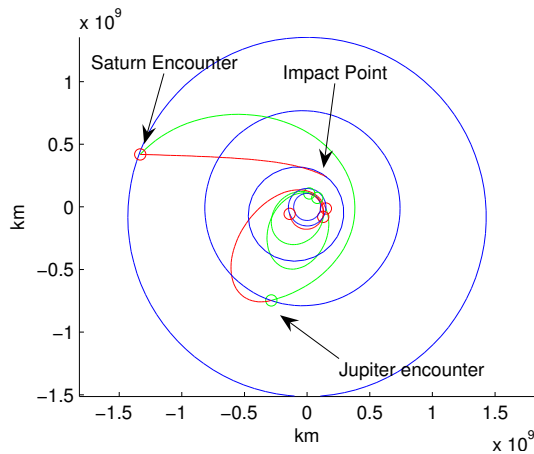


Figure 4: Best solution found with the DE implementation for the EVEVEJSA problem.

due to the limited number of function evaluations allowed together with the small population size used. While increasing these number would probably improve the efficiency of the solvers, it would also increase the computational time considerably. We must also note that while the parameters of the DE optimiser have been selected after a number of test campaigns, a minor effort was put in the selection of the parameters for the MPSO and that the numbers here reported should not lead to the definite conclusion that DE is outperforming MPSO. In a private communication, Manfred Stickel of the Max Planck institute told us he could solve this problem overnight with his implementation of Particle Swarm Optimisation. We end by observing that in a much longer run (four hours of computation) our distributed multistart version of Differential Evolution found the optimal result of $J = 1.844E6$. The same can be located with a much faster algorithm by using the pruning technique mentioned [3] that introduces a good deal of problem knowledge. For the EdVdM problem the algorithms both performed quite well, with DE being again consistently able to locate the global optimum. The problem becomes considerably more complex when we add more trajectory legs and already in the EdEdMdM case MPSO is not able to return a good solution, while DE worked better, but found only three trajectories with an objective function less than 4 km/s out of 50 runs. The

EexpM low-thrust problem is solved quite easily by both the techniques used, it has to be noted though, that the simplification introduced by the use of exponential sinusoids is quite significant. While the results may still be used as first guesses and then fed into an optimal control problem solver, it has to be proven that this starting guess would be within the basin of attraction of the real global optimum.

8 Conclusions

Global optimisation algorithms, and in particular the implementations considered of Differential Evolution and multiple Particle Swarm optimisation, may be used to obtain preliminary results when applied to a number of interesting trajectory optimisation problems. Their application without the use of ad hoc developed space pruning techniques is much less effective when the problem complexity increases and they start to suffer of the non polynomial complexity of the optimisation task. In these cases a much longer computational time is required and their use in place of other techniques is therefore questionable.

References

- [1] D. Izzo. Lambert's problem for exponential sinusoids. *to appear in Journal of Guidance Control and Dynamic*, 2006.
- [2] D. Izzo. 1st ACT Global Trajectory Optimisation Competition, [www.esa.int/gsp/ACT/] visited 23/04/2006.
- [3] D. Izzo, V. Becerra, D. Myatt, S. Nasuto, and M. Bishop. Search Space Pruning and Global Optimisation of Multiple Gravity Assist Spacecraft Trajectories. *submitted to Journal of Global Optimization*, 2006.
- [4] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of the Int. Conf. on Neural Networks*, pages 1942–1948, 1995.
- [5] D. Myatt, V. Becerra, S. Nasuto, and J. Bishop. Advanced global optimisation for mission analysis and design. *ESA Ariadna final report 03/4101*, available on-line at [www.esa.int/gsp/ACT/], 2004.

- [6] A. Petropoulos and J. Longuski. Shape-based algorithm for automated design of low-thrust, gravity-assist trajectories. *Journal of Spacecrafts and Rockets*, 41(5), 2004.
- [7] R. Storn and K. Price. A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11:341–359, 1997.