APPROXIMATE ESTIMATES OF ORBIT TRANSFER COST FOR EFFICIENT MISSION ANALYSIS AND DESIGN

Giulio Avanzini,[*] Danilo Zona,[†] Francesco Marchetti,[‡] and Edmondo Minisci,[§]

Symbolic Regression is investigated as a tool for identifying analytical expressions which provide an estimate of orbit transfer cost, evaluated in terms of required velocity increment, as a function of initial and target orbit geometry. Different approaches are considered to identify the best approach to sample the problem parameter space and the algorithm which performs better, in the framework of Genetic Programming. Each resulting method is tested for five different orbit transfer geometries between coplanar circular and elliptical orbits. Results demonstrate the viability of the approach, although when the number of problem parameter increases, computational cost becomes sizeable. Also, local minima may be filtered by the regression.

INTRODUCTION

The estimate of the cost of orbit transfer maneuvers from an initial orbit to a desired target orbit is the cornerstone of any space mission analysis and design method. Unless very simple mission scenarios are dealt with, the mathematical difficulties of the problem and the complexity of its numerical solution result into a considerable computational burden. More in particular, interplanetary missions which involve planetary flybys[1] or missions aimed at visiting a sequence of orbiting targets[2] require the solution of a Mixed Integer Nonlinear Programming (MINLP) problem, which may feature several local optima over a possibly very wide search space.

The resulting optimization problem is often tackled by means of a nested approach,[3] where the inner optimization problem determines the optimal transfer from one target to the following one, for a given sequence of targets, whereas the outer problem pursues the identification of the optimal sequence. Different (and possibly competing) merit functions may be defined, such as overall transfer time and total ΔV, together with other mission constraints. When a relatively high number of targets needs to be reached during a single mission (as it may be required by an active debris removal mission), the outer optimization problem can become hardly treatable. An accurate solution of the inner problem may cause the overall computation cost to become prohibitive, even if high performance parallel computing techniques are adopted. Approximate heuristic approaches have been proposed in the past[4,5] for providing estimates of transfer cost, while recasting the preliminary design phase in the form of a (still computational demanding) combinatorial optimization problem. Once an optimal sequence of encounters with orbiting objects is identified, the optimization can be refined on the basis of a more accurate simulation of the sequence of maneuvers.

[*]Professor, Department of Engineering for Innovation, Università del Salento, Campus Ecotekne (Building "O"), Strada per Monteroni, 73100, Lecce (Italy).

[†]Ph.D. Student in Engineering of Complex Systems, Department of Engineering for Innovation, Università del Salento, Campus Ecotekne (Building "O"), Strada per Monteroni, 73100, Lecce (Italy).

[‡]Ph.D. Student, Department of Mechanical and Aerospace Engineering, University of Strathclyde, 75 Montrose Street, G1 1XJ, Glasgow (UK).

[§]Senior Lecturer, Department of Mechanical and Aerospace Engineering, University of Strathclyde, 75 Montrose Street, G1 1XJ, Glasgow (UK).

The availability of an accurate estimate of the transfer cost associated to each transfer leg of the sequence of maneuvers clearly represents a significant advantage in the framework of the preliminary mission design phase. This paper presents possible methods to derive accurate heuristics to be used during this phase, for increasingly more complex mission scenarios. Modern machine learning methods are exploited in order to define a regressor which provides an analytical formulation for estimating the cost of a transfer arc between an initial and a target obit, as a function of the geometric parameters which identify the problem.

The objective of the paper is twofold. On one side, the use of a Symbolic Regression (SR) approach is proposed as an alternative to other more common machine learning methods, such as artificial neural networks and deep learning. SR searches the space of mathematical expressions to find the one(s) that best fit a given dataset, and in doing so, it is able to provide interpretable models, as opposed to black-box ones. For this work a Genetic Programming (GP) based SR will be considered.[6,7] Differently from other Evolutionary Algorithms (EAs), in the GPs the individuals are mathematical expressions structured as trees (Fig. 1), where numerical coefficients and input variables are called terminal nodes, while the other nodes in the GP tree are predefined primitive functions. In this work, both single gene and multi gene GPs[7,8] will be considered and compared. As a further contribution, performance of a regressor found on the basis of a regular sampling of the parameter space of the problem is compared with randomized techniques, where problem parameters are sampled stochastically.

The study starts from a relatively simple test case, represented by transfer between circular coplanar orbits. Such a configuration minimizes the number of relevant geometric parameters (namely, the ratio between the radii of initial and final orbits, r_2/r_1 and transfer angle θ between the departure and arrival points). The cost of the transfer, represented by the sum of the ΔV's required for the two impulsive maneuvers at departure and arrival, depends on three parameters only, r_2/r_1, θ, and transfer time, Δt, and it is estimated from the solution of a two point orbital boundary value problem, for the considered geometry and transfer time. This elementary scenario is considered only with the objective of tailoring the SR algorithms, before challenging it on more complex geometries, such as transfer between elliptical coplanar orbits. In this case, four more parameters are required to define the problem geometry, namely, eccentricity e_1 and e_2 of the initial and final orbit, angular separation ω between the periaxes, and initial position along the starting orbit, ν_0, so that the number of total independent parameters becomes equal to 7.

For non-coplanar orbit transfers, the angle between the orbit planes needs to be considered. Moreover, the angular position of the periaxes of both orbits, counted with respect to the line at the intersection of the two orbit planes, ω_1 and $\omega_@$, need to be accounted for independently. The total number of independent parameters is thus equal to 9. This latter case will be dealt with in a future research. In all the cases, the regressor is expected to provide an analytical formulation for the total ΔV required for the considered orbit transfer, as a function of the parameters which describe problem geometry and transfer time.

In what follows, the next section provides some details on the technique adopted for performing the symbolic regression. The third section describes in more detail the geometries of the problems and the solution algorithm adopted for the solution of Lambert problem, for the various configurations. Results are then presented and discussed. The final section highlights some concluding remarks and highlights future steps of the research.

A NOTE ON GENETIC PROGRAMMING

Genetic Programming (GP)[6] is an Evolutionary Algorithm (EA) capable of producing regression or classification models in explicit mathematical form. When applied to regression, the approach is termed symbolic regression because the produced regression model is in symbolic form. The output of a GP model is depicted in Figure 1 and can be read as $x - 4 + y + 5$.
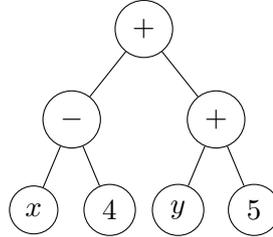
Figure 1: Representation of a GP tree.

In the GP algorithm, an initial population composed of randomly generated individuals is created. Then, during the evolutionary process, the individuals are combined using the crossover and selection operators resulting in an offspring population. Subsequently, from the offspring and the parent population individuals are selected according to their fitness to form the parent population for the next generation of the evolutionary process. The actions above are repeated, until a termination criteria is met, which can be represented by either a maximum number of generations or finding the best individual in the current population with a fitness level below (or above) a prescribed threshold. The fitness is defined by the user and its goal is to guide the evolutionary process towards its minimum (or maximum).

Several GP variants were formulated in the past, each of which focuses on a particular aspect of the evolutionary process or on how the GP trees are shaped. In this work, two approaches to GP are selected, the Multi-Gene Genetic Programming (MGGP) and the classic GP. The MGGP[9] is a GP variant where multiple GP trees are considered simultaneously to build a linear combination model. If the output of an individual as the one in Figure 1 is represented by I, a MGGP individual will be composed as $p_0 + p_1 I_1 + p_2 I_2 + ... + p_n I_n$ where n is the maximum number of genes considered. The parameters p of this linear combination are then optimized using a least square optimization algorithm. This optimization approach is faster than other local or global approaches, therefore it can be performed for each individual at each generation without resulting in a prohibitive computational cost. As a result, better performance can be achieved by the MGGP in comparison to other GP formulations.

For both of the GP and MGGP approaches considered in this work, a variant developed in a previous work is used, the Full Inclusive Genetic Programming (FIGP),[10] resulting in the FIGP and Full Inclusive Multi-Gene Genetic Programming (FIMGGP) algorithms. The FIGP was developed to promote and maintain the population's diversity throughout the evolutionary process. To do so, both the genotypic and phenotypic diversity of the individuals is considered, and, regarding the phenotypic diversity, both the training and validation fitnesses are considered to avoid overfitting.
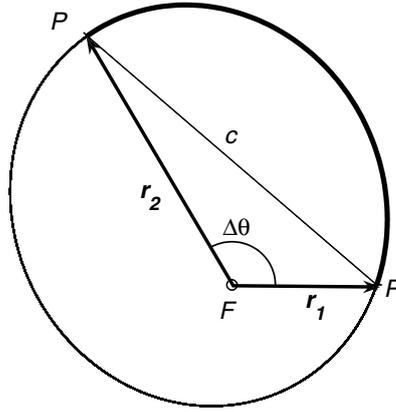
Figure 2: Geometry of Lambert problem (from Ref. 11).

ESTIMATE OF ORBIT TRANSFER COST

The Classic Lambert Problem

The orbit two-point boundary value problem (also known as Lambert problem) is represented by the determination of the Keplerian orbit arc of amplitude θ, which takes an orbiting body from an initial point P_1 at distance r_1 from the primary body, to P_2, at a distance r_2, in a prescribed time Δt. Its definition stems from Lambert's theorem, which states that the transfer time between any two points along a Keplerian orbit arc depends on three parameters only, namely, the chord length c between the points, the sum of initial and final distances, $r_1 + r_2$, and the semimajor axis of the orbit passing through P_1 and P_2 (Fig. 2). Hence, when the geometry of the problem is fixed, the transfer time between P_1 and P_2 depends on a single parameter.

The orbits passing trough P_1 and P_2 can thus be parameterized as a function of a single free variable, such as the component of the eccentricity vector normal to the chord, as it is done in Ref. 11. The solution of Lambert problem can thus be recast into the identification of the free parameter which results into the prescribed transfer time. Once the orbit arc is identified, initial and final velocity are easily determined. The total cost of the transfer is defined by the sum of the ΔV's required by impulsive maneuvers to inject the spacecraft from the initial orbit onto the transfer arc and from the latter onto the target orbit, $\Delta V_{tot} = \Delta V_1 + \Delta V_2$.

Planar Case

The simplest possible geometry of a single orbit transfer leg is represented by the transfer between coplanar circular orbits. In this case (Fig. 3.a), if distances are scaled with respect to the initial orbit radius, r_1, and times are scaled with respect to the period of the initial orbit, T_1, the problem geometry is defined by only two parameters, namely $\rho = r_2/r_1$, and the transfer angle θ. The prescribed duration of the transfer, $\tau = t/T_1$, in nondimensional terms, provides the third (and last) independent variable of the problem, so that the regressor is required to provide an analytical approximation of the total cost of the transfer

$$\Delta V_{tot} = f_{CC}(\rho; \theta; \tau)$$

where the subscript CC stands for circular coplanar orbits.
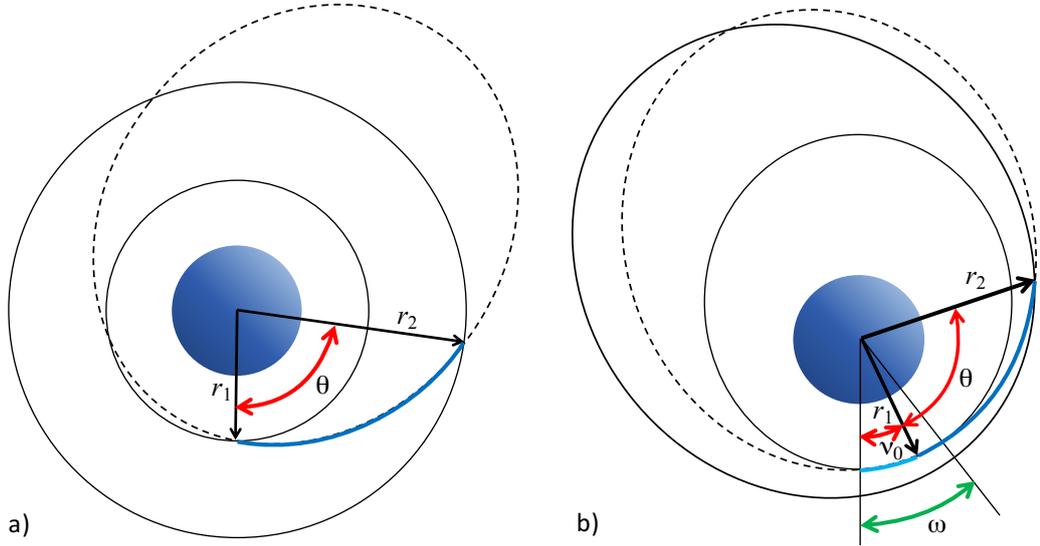
Figure 3: Geometry of coplanar transfer problems for circular (a) and elliptical (b) orbits.

The mission scenario becomes more complex when coplanar elliptical orbits are dealt with, as represented in Fig. 3.a. In this case the geometry of the problem must account for the eccentricities, e_1 and e_2, of initial and target orbits, and the angle ω between their periaxes. The ratio $\rho = a_2/a_1$ is now taken between the semimajor axes of the two orbits. A coasting arc on the initial orbit also needs to be included among the independent variables, provided that the transfer cost between the two orbits depends on the radius $r_1(\nu_0)$ at departure. The overall cost of the transfer thus becomes

$$\Delta V = f_{EC}(\rho, e_1, e_2, \omega; \nu_0, \theta; \tau)$$

where the subscript EC stands for elliptic coplanar orbits.

Transfer Between Non-Coplanar Orbits

Note that the transfer angle, θ, is determined by initial and final positions on orbits lying on different planes, and its value can be replaced by the true anomaly at the arrival point on the target orbit, ν_F. Anyway, this latter case is introduced here for the sake of completeness, but only planar transfers will be considered in the section of Results.

Practical Scenarios

Bounds on the values of i, e_1, and e_2 can be identified, depending on the particular mission scenario considered. As an example, almost circular orbits can result into small values for both e_1 and e_2. Similarly, if the targets of the missions fly along neighboring orbits, also the value of i can be small. At the same time the value of transfer time τ can be bounded by mission requirements. As an example of such a scenario, collecting debris fragments in Low Earth Orbits (LEO) typically deals with low eccentricity orbits. If mission time is not an issue, it is also possible to exploit orbit perturbations (in particular, precession of the line of the nodes under the J_2 effect) to minimize the required orbit plane change. Conversely, the values of ω_1, ω_2, ν_0, and θ can vary in general from $-\pi$ to π rad. These variables need to be sampled over their entire range of admissible variation during the process of identification of the regressor.

5
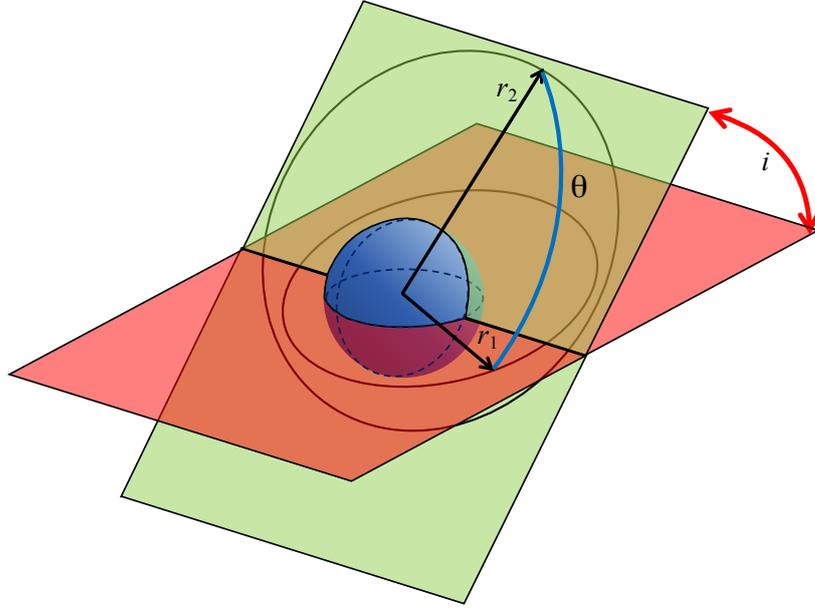
Figure 4: Geometry of non-coplanar transfer problems for elliptical orbits..

RESULTS AND DISCUSSIONS

In this section the results obtained for the performed tests are presented. Two scenarios are considered, namely transfers between coplanar circular orbits and transfers between coplanar elliptical orbits. These scenarios will be referred to as circular and elliptical transfers respectively. As described in the previous Section, in the circular transfer reference model, ΔV_{tot} depends on $\rho = r_2/r_1$, θ and τ, whereas in the elliptical transfer reference model ΔV_{tot} depends on $\rho = a_2/a_1$, e_1, e_2, ω, ν_0, θ and τ. Therefore, the goal of the numerical experiment is to obtain a regression model of the circular and elliptical transfer models using the FIGP and FIMGGP algorithms described above. Moreover, two approaches to build the dataset are considered: a regular grid and a Latin Hypercube Sampling (LHS). Therefore a total of eight experiments are performed: 1) Circular transfer-FIGP-grid dataset, 2) Circular transfer-FIGP-LHS dataset, 3) Circular transfer-FIMGGP-grid dataset, 4) Circular transfer-FIMGGP-LHS dataset, 5) Elliptical transfer-FIGP-grid dataset, 6) Elliptical transfer-FIGP-LHS dataset, 7) Elliptical transfer-FIMGGP-grid dataset, 8) Elliptical transfer-FIMGGP-LHS dataset.

The aim of these experiments is: 1) to observe differences of performances between FIGP and FIMGGP algorithms; 2) evaluate the influence of dataset sampling on performance; 3) assess the applicability of the FIGP and FIMGGP to a simple regression problem and to a more complex one. To obtain a preliminary statistics, Five simulations are performed for each of the eight experiments.

Datasets preprocessing

For the circular case, the grid dataset is composed by a $50(\rho)\times50(\theta)\times40(\tau)$ grid, for a total of 100000 points. The LHS dataset is also composed of 100000 points. For the elliptical case, all independent variables are sampled in 6 points. The resulting grid dataset is thus composed by $6 \times 6 \times 6 \times 6 \times 6 \times 6 \times 6 = 279\,936$ points. The LHS dataset is composed of $300\,000$ points. For both algorithms the considered datasets are divided into Train, Validation and Test sets. First

a split between TrainValidation (80%) and Test (20%) is performed, and then the TrainValidation set is further split into Train (80% of TrainValidation) and Validation (20% of TrainValidation) sets. Therefore the final subdivision will be Train=64%, Validation=16% and Test=20% of the complete dataset. The Train and validation sets are used during the evolutionary process while the test set is used at the end to assess the overfitting degree of the obtained results on data never seen before. For that reason, the results presented hereafter are obtained from the individuals that showed the least overfitting. To do so, for each generation the individual with the best training fitness is saved. From these individuals, the one with the best validation fitness is picked as it is the one with least overfitting. This individual is then applied on the test data to obtain the results presented. In order to properly apply the FIGP and FIMGGP, all the dataset were scaled with the min max approach. i.e., each input feature was scaled from their original range to the [0,1] range using their min and max values, and the output was scaled from the range [0, 40] to [0, 1] too.

Table 1 provides all relevant parameters for the FIGP and FIMGGP algorithms. The meaning of all the parameters is detailed in the paper.[10] Here, it can be pointed out that, given the different dimensionality (and then difficulty) of the two cases, both algorithms were set to use a bigger population for the elliptical transfer case, and the FIMGGP algorithm was also set to use a higher number of genes.

Results

The plots shown hereafter depict the results obtained in terms of the evolution of fitness ($RMSE$) during the training and validation phases, and R^2 score of the models derived by means of the procedure outlined above on the train, validation and test datasets. In the fitness evolution plots, the inset depicts the last 200 generations to show whether or not the algorithm reached convergence. The statistics are evaluated considering the five simulations performed for each experiment. Starting from the coplanar circular transfer case, Figures 5b and 5a depicts the results of the FIGP with the grid dataset while Figures 6b and 6a concerns the FIGP with the LHS dataset. Comparing these results no significant difference in performance emerges when using the grid or LHS datasets.
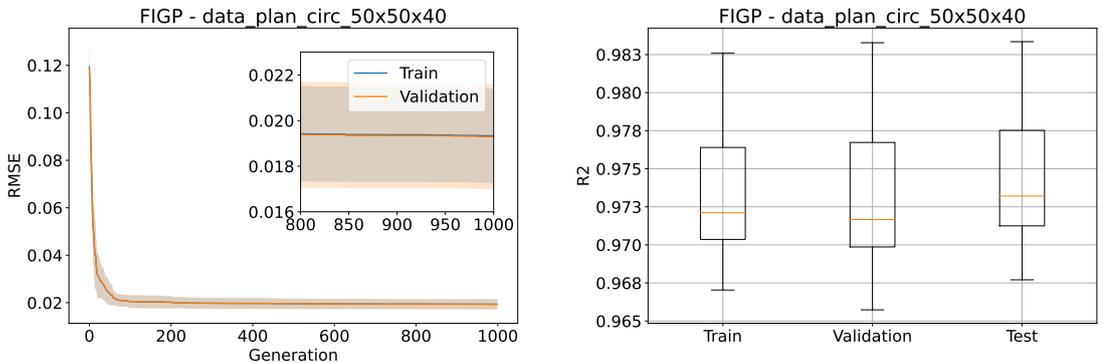
(a) Training and validation fitness evolution (solid lines: mean value; shaded area: standard deviation).

(b) R^2 scores on the train, validation and test sets

Figure 5: Results of the FIGP models on the circular transfer case with the grid dataset.

Figures 7b and 7a show the results of the FIMGGP on the circular transfer case with the grid dataset while Figures 8b and 8a refer to the results obtained with the LHS dataset. Again, no

Table 1: Settings for the FIGP and FIMGGP algorithms.

	FIGP	FIMGGP
Population Size	500 individuals (750 for the elliptical case)	
Maximum Generations	1000	
Stopping criteria	Reaching maximum number of generations	
Maximum number of genes	/	10 (12 elliptical case)
Crossover probability	$0.75 \rightarrow 0.25$	
Low level Crossover probability	/	0.7
Mutation probability	$0.25 \rightarrow 0.75$	
Evolutionary strategy	$\mu + \lambda$	
μ	Population size	
λ	Population Size $\times$ 1.2	
Number of Ephemeral constants	1	
Limit Height	40	15
Limit Size	40	20
Selection Mechanism	Inclusive Tournament	
Double Tournament fitness size	2	
Double Tournament parsimony size	1.6	
Number Niches Length	6	
Number Niches Fitness Training	6	
Number Niches Fitness Validation	6	
Tree creation mechanism	Ramped half and half (min size=1, max size=4)	
Mutation mechanisms	Uniform (55%), Shrink (5%), Insertion (25%), Mutate Ephemeral (15%)	
Crossover mechanism	One point crossover	
Primitives Set	$+, -, *, add3, mul3, tanh, .^2, log, exp,$ $sin, cos, \sqrt{.}, tan, arcsin, arccos, arctan$	
Fitness measure	RMSE	
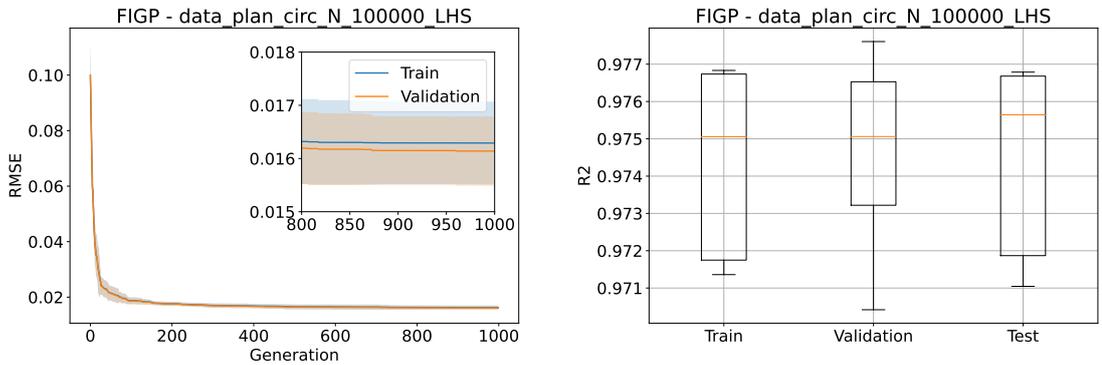
(a) Training and validation fitness evolution (solid lines: mean value; shaded area: standard deviation).

(b) R^2 scores on the train, validation and test sets

Figure 6: Results of the FIGP models on the circular transfer case with the LHS dataset.

significant difference is observable between the results obtained for datasets generated following different procedures, but, as a remarkable result, performance of the FIMGGP appear significantly better, when compared to those of the FIGP. The FIGP achieves a median R^2 score of 0.974 on the test set, while the FIMGGP achieves a median R^2 score of 0.997. Moreover it is interesting to notice that the FMIGGP did not reached convergence as shown in Figure 7a. A further relevant aspect can be highlighted by observing the R^2 scores on the train, validation and test sets for all the experiments of the circular transfer case. Both the FIGP and FIMGGP showed no overfitting since the R^2 scores on the train, validation and test dataset are comparable in terms of median and statistical distribution. This is a desirable behaviour in regression applications, since the models derived must perform well also on unseen data.

(a) Training and validation fitness evolution (solid lines: mean value; shaded area: standard deviation).

(b) R^2 scores on the train, validation and test sets
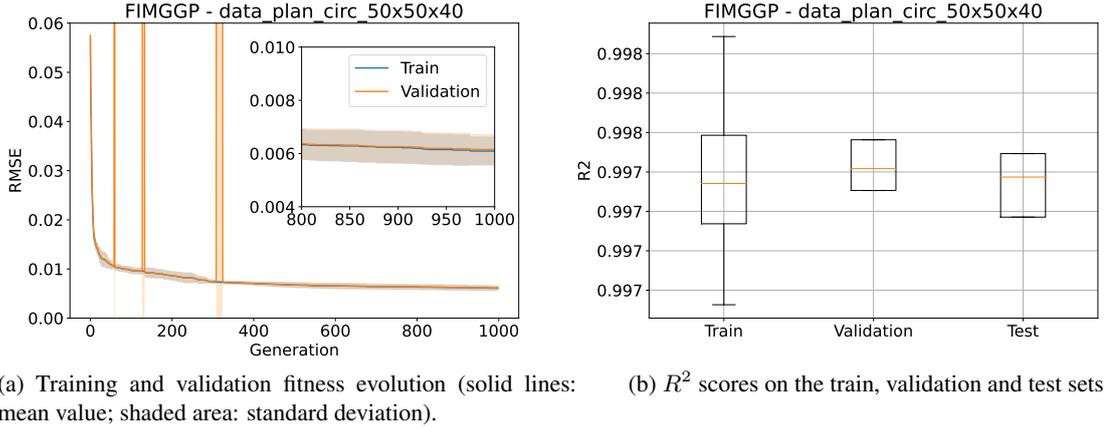
Figure 7: Results of the FIMGGP models on the circular transfer case with the grid dataset.

(a) Training and validation fitness evolution (solid lines: mean value; shaded area: standard deviation).

(b) R^2 scores on the train, validation and test sets
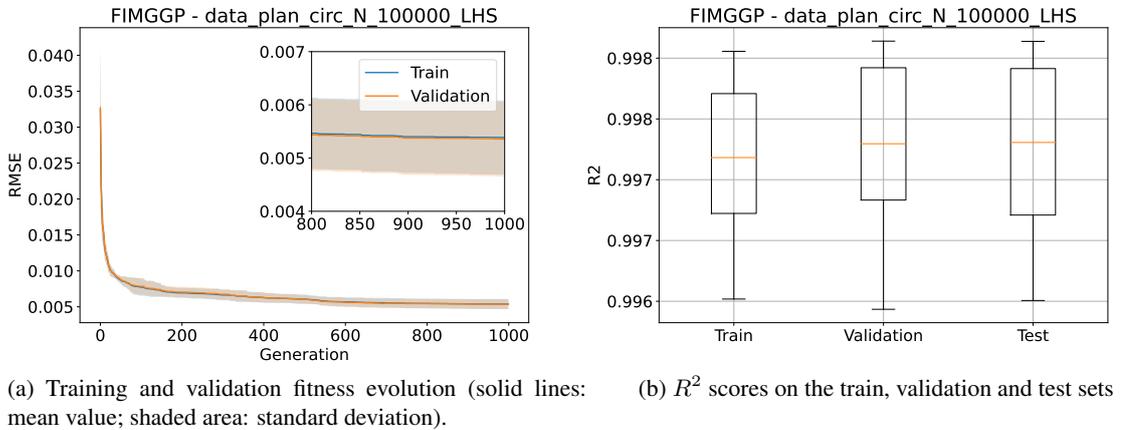
Figure 8: Results of the FIMGGP models on the circular transfer case with the LHS dataset.

When the elliptical transfer case is dealt with, Figures 9b and 9a present the results of the FIGP models obtained with the grid dataset, whereas Figures 10b and 10a show the performance of models obtained with the LHS dataset. By looking at the R^2 score it can be observed how in this case results obtained with the grid dataset are slightly better. The same conclusion can be drawn by looking at Figures 11b and 12b, which depict the R^2 scores of the models obtained with the FIMGGP on

(a) Training and validation fitness evolution (solid lines: mean value; shaded area: standard deviation).

(b) R^2 scores on the train, validation and test sets
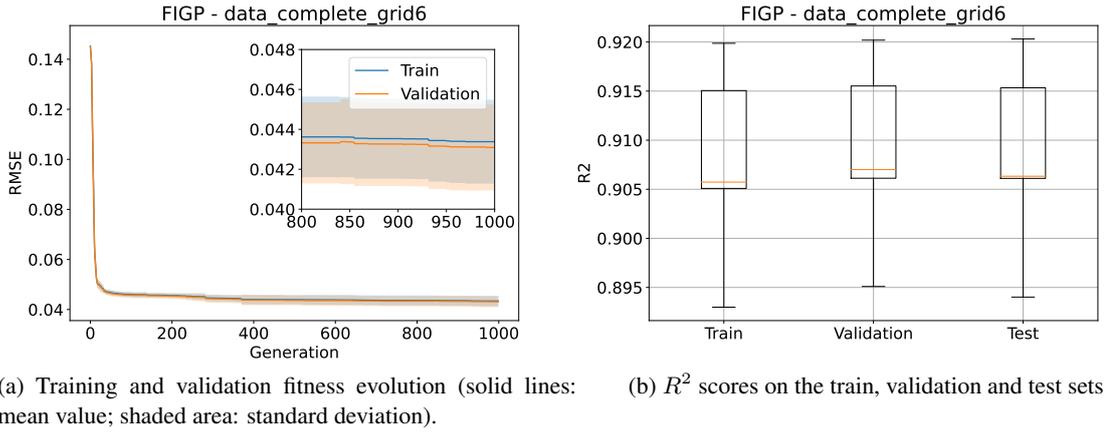
Figure 9: Results of the FIGP models on the elliptical transfer case with the grid dataset.

the grid and LHS datasets, respectively. Nevertheless, these improvements are marginal, provided that the R^2 score improves of 0.1 only. Apparently, a more regular structure of the grid can cover the range of the input features better, compared to the LHS one, especially considering that, in order to keep the total number of points in the dataset within reasonable limits for an acceptable overall computation cost of the procedure, the number of sampled points within the interval of each parameter needs to be small.

As for the circular transfer case, the FIMGGP performs better than the FIGP and produces models with higher R^2 scores, with an average 0.9 for the FIGP versus 0.93 for the FIMGGP. The results obtained for the elliptical transfer case are worse than those obtained for the circular transfer case. This is due to the greater complexity of the problem and the computational issues experienced in order to tackle it. In fact, to achieve the same level of performance bigger datasets should be used and the hyperparameters of both algorithms should be set to increase their performances, e.g. by increasing the number of individuals in the population or the number of genes in the FIMGGP. Even if this was considered and attempted, due to limitations in terms of computational capacities, the dataset dimensions, as well as the population and the number of genes had to be limited and far smaller than a more proper setting. This limitation is also the reason for a relevant discrepancy between the train, validation and test R^2 scores in Figures 11b and 12b, if compared with those obtained in the other circular transfer cases. A slight overfitting is observed in the models produced by the FIMGGP on the elliptical case with grid dataset, while the opposite is observed in the models produced by the FIMGGP on the elliptical case with LHS dataset. More simulations should be performed to improve the statistical relevance of the results.

As far as convergence of the algorithms is concerned, in the half of the experiments the algorithms reached convergence, as highlighted by Figures 5a, 6a, 11a and 12a, that is, for the FIGP applied to the circular transfer with both dataset and the FIMGGP applied to the elliptical transfer with both dataset. In the remaining experiments, shown in Figures 7a, 8a, 9a and 10a, a slight decrease of fitness is still observable at the end of the evolution, stopped by the prescribed maximum number of generations. Therefore, an improvement in the results could be expected for the FIMGGP applied to the circular transfer and the FIGP applied to the elliptical transfer. Nonetheless, more simulations should be performed to have a better understanding of convergence trends.

10

(a) Training and validation fitness evolution (solid lines: mean value; shaded area: standard deviation).

(b) R^2 scores on the train, validation and test sets
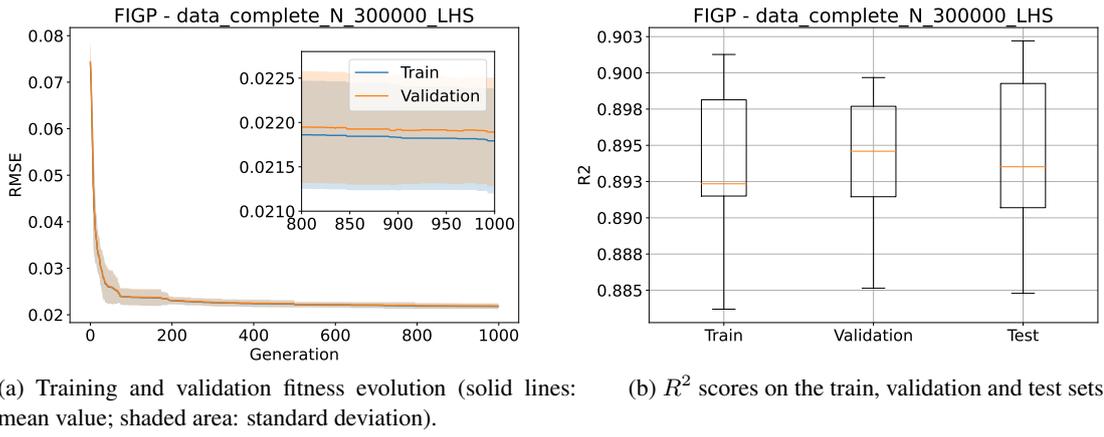
Figure 10: Results of the FIGP models on the elliptical transfer case with the LHS dataset.

(a) Training and validation fitness evolution (solid lines: mean value; shaded area: standard deviation).

(b) R^2 scores on the train, validation and test sets
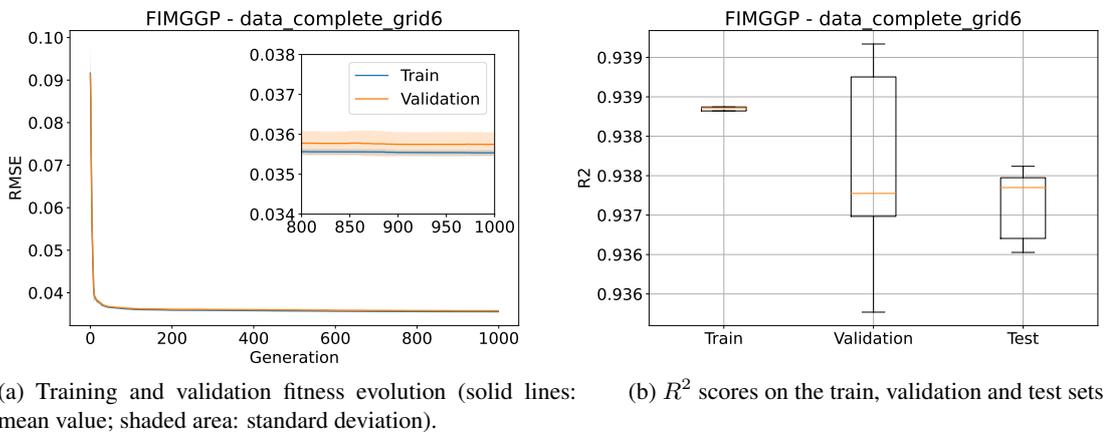
Figure 11: Results of the FIMGGP models on the elliptical transfer case with the grid dataset.

(a) Training and validation fitness evolution (solid lines: mean value; shaded area: standard deviation).

(b) R^2 scores on the train, validation and test sets
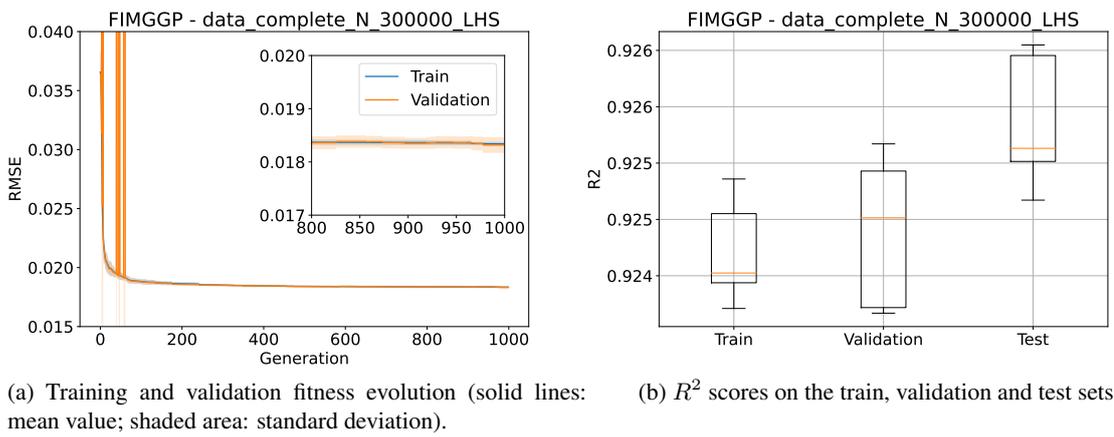
Figure 12: Results of the FIMGGP models on the elliptical transfer case with the grid dataset.

An example of an analytical expression is reported below, for the result obtained by the FIMGGP for the circular case with the LHS dataset:

$$f = a_1 * x_2 + a_2 * tanh\left(2.0 * x_3 + sin(cos(x_2^{(1/2)})) + sin(x_3) + a_3 * x_2^2\right)$$

$$-a_4 * tanh\left(x_2 + 3.0 * x_3 + x_3^{(1/2)}\right) + a_5 * atan\left(a_{20} * x_1 * exp\left(a_6 * x_2 * cos(x_3^{(1/2)})\right)\right) +$$

$$a_7 * atan(x_2) + a_8 * exp(x_2) +$$

$$a_9 * x_1 * \left(x_3 + cos(x_3) + sin\left(cos\left(\frac{(a10 * x_3 * exp(x_1))}{(a_{11} * exp(2.0 * x_1) * x_3^2 + 1.0)^{(1/2)}}\right)\right)\right) +$$

$$a_{12} * x_1 * atan\left(sin(cos(a_{19} * x_2^{(3/2)}))\right) + a_{13} * x_1 * x_3^{(1/2)} + a_{14} * x_3 * atan\left(a_{15} * x_2^{(3/2)}\right) +$$

$$a_{16} * x_1 * \left(x_3 + sin(x_3) + \frac{1}{(a_{17} * exp(2.0 * atan(exp(x_2))) * sin(x_3)^2 + 1.0)^{(1/2)}}\right) + a_{18}$$

with
$a_1 = 0.087360387809065978559353027321777$
$a_2 = 0.50730323464088744600530844763853$
$a_3 = 5.8207$
$a_4 = 0.10236149627635197922881360454994$
$a_5 = -0.049151188058235671662909993528956$
$a_6 = 5.8207$
$a_7 = -0.28280097835513151594000191835221$
$a_8 = 0.087360387809065978559353027321777$
$a_9 = 0.37386358151020654405627396954515$
$a_{10} = 37.81091457$
$a_{11} = 1429.6652606198382849$
$a_{12} = -0.023731194133736366325138078536838$
$a_{13} = -1.4027793666815768780509188218275$
$a_{14} = 0.006536365068816417552011177605209$
$a_{15} = 56.79198783$
$a_{16} = 0.35836947399350699283715471210598$
$a_{17} = 95.19709761$
$a_{18} = -0.39835856783426271121584250067826$
$a_{19} = 5.8207$
$a_{20} = 5.8207$

Critical cases

When the implementation of accurate meta-models is the aim of the work, metrics such as the $RMSE$ and R^2 can only give a partial indication of the real suitability of the model, because particular structures may actually be missing in the data-sets. To better check the suitability of the GP meta-models, a series of random internal cuts, from a random x_{d0} point to another random x_{d1} point in the problem parameter space, have been performed and the GP models compared to the actual model for the circular case (edge points for the cuts are reported in Table 2). This preliminary analysis showed that:

- in most cases the meta-models produced by the multi-gene GP match well the actual model (Fig. 13a shows an example for ΔV_{Tot} vs ρ along $CutA$), missing narrow local minima, such as those seen on the left side of Fig. 14a and Fig. 15a, related to cuts $CutB$ and $CutC$, respectively;

- in most cases, the usually simpler meta-models produced by the single-gene GP match sufficiently well the actual model (Fig. 13b shows an example for ΔV_{Tot} vs ρ along $CutA$), but they also miss narrow local minima, such as those on the left side of Fig. 14b and Fig. 15b, related to cuts $CutB$ and $CutC$, respectively.

Table 2: Initial and final points for three representative cuts for domain exploration

$CutA$	$x_{d0} = [13.3612946231533, 5.48312713605869, 16.881832765281]$
	$x_{d1} = [12.4102673117917, 5.56708052182314, 2.95221227958872]$
$CutB$	$x_{d0} = [16.6046594692667, 3.91721535154401, 12.1465971896452]$
	$x_{d1} = [0.620812707156099, 6.0168096200283, 9.62617572884094]$
$CutC$	$x_{d0} = [8.4550486096238, 0.674865437728657, 8.69763560472183]$
	$x_{d1} = [0.386246441681431, 6.27656301355617, 17.9354245735156]$
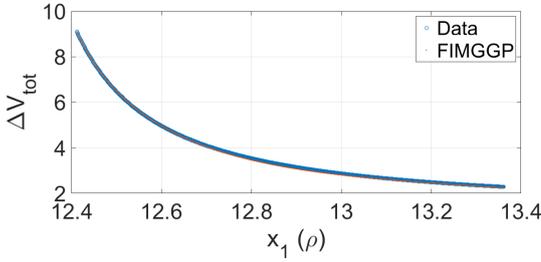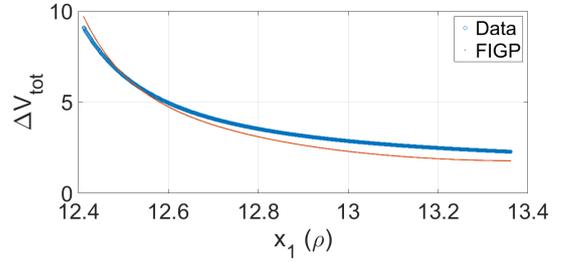
(a) FIMGGP model trained with LHS data   (b) FIGP model trained with LHS data

Figure 13: Behaviour of the GP models for the internal cut Case2 - $x_1(\rho)$ view
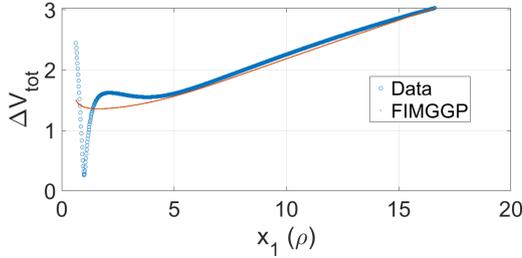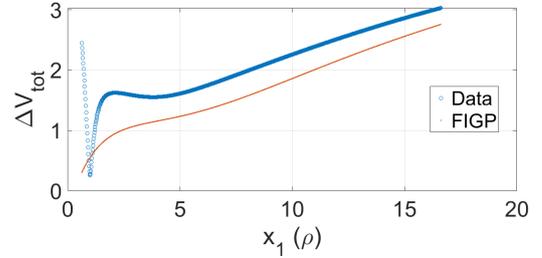
(a) FIMGGP model trained with LHS data   (b) FIGP model trained with LHS data

Figure 14: Behaviour of the GP models for the internal cut Case1 - $x_1(\rho)$ view

CONCLUSIONS AND FUTURE WORK

Use of Genetic Programming (GP) for training a Symbolic Regressor is investigated as a tool for the identification of analytical metamodels, which estimate velocity increment required by orbit maneuvers between coplanar circular and elliptical orbits as a function of geometric parameters which describe the initial and target orbits. Several approaches were tested for both circular and

(a) FIMGGP model trained with LHS data   (b) FIGP model trained with LHS data
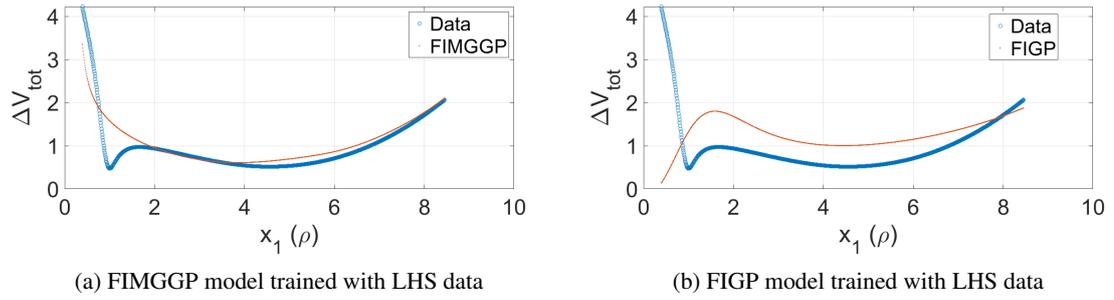
Figure 15: Behaviour of the GP models for the internal cut Case3 - $x_1(\rho)$ view

elliptical orbit cases, with different GP algorithms (single gene and multi-gene GPs) and dataset grid techniques (regular grid vs Latin Hypercube Sampling), for a total of 8 experiments. The effects of the sampling techniques appears as marginal, although a slight improvement in performance is observed for regular sampling on the more complex elliptic transfer case. Conversely, multi-gene GPs outperforms the single gene GPs in all the considered tests, although in all cases, fine structures, such as local narrow minima, are missed. Future work will address the extension of the approach to more complex operational scenarios, including plane change maneuvers and the use of low-thrust propulsion system. As far as performance of the training algorithm is concerned, further study is required for improving numerical aspects of the algorithm in order to deal with larger datasets, for better sampling the parameter space of the problem.

REFERENCES

[1] G. Colasurdo, A. Zavoli, A. Longo, L. Casalino, and F. Simeoni, "Tour of Jupiter Galilean moons: Winning solution of GTOC6," *Acta Astronautica*, Vol. 102, 2014, pp. 190–199.

[2] L. Federici, A. Zavoli, and G. Colasurdo, "Evolutionary Optimization of Multirendezvous Impulsive Trajectories," *International Journal of Aerospace Engineering*, Vol. 2021, 2021, p. 19.

[3] D. Zona, L. Federici, A. Zavoli, and G. Avanzini, "Evolutionary Optimization for Active Debris Removal Mission Planning," *IEEE Access*, Vol. 11, 2023, pp. 41019–41033.

[4] M. Cerf, "Multiple Space Debris Collecting Mission: Optimal Mission Planning," *Journal of Optimization Theory and Applications*, Vol. 167, 2015, pp. 195–218.

[5] H.-X. Shen and L. Casalino, "Simple ΔV Approximation for Optimization of Debris-to-Debris Transfers," *Journal of Spacecraft and Rockets*, Vol. 58, No. 2, 2021, pp. 575–580.

[6] J. R. Koza, *Genetic programming: on the programming of computers by means of natural selection*. Cambridge: MIT press, 1992.

[7] F. Marchetti and E. Minisci, "Inclusive Genetic Programming," *Genetic Programming* (T. Hu, N. Lourenço, and E. Medvet, eds.), Cham, Springer International Publishing, 2021, pp. 51–65.

[8] D. P. Searson, "GPTIPS 2: an open-source software platform for symbolic data mining," 2015.

[9] D. Searson, M. Willis, and G. Montague, "Co-evolution of non-linear PLS model components," *Journal of Chemometrics*, Vol. 21, No. 12, 2007, pp. 592–603.

[10] F. Marchetti, M. Castelli, I. Bakurov, and L. Vanneschi, "Full Inclusive Genetic Programming," *[SUBMITTED TO] Evo* 2024 Conference*, 2024.

[11] G. Avanzini, "A Simple Lambert Algorithm," *Journal of Guidance, Control, and Dynamics*, Vol. 31, 2008, pp. 1587–1694.