Artificial Neural Networks for Multiple NEA Rendezvous Missions with Continuous Thrust

Giulia Viavattene* and Matteo Ceriotti†
University of Glasgow, Glasgow G12 8QQ, United Kingdom

The interest for near-Earth asteroids for scientific studies and, in particular, for potentially hazardous asteroids requires the space community to perform multiple-asteroid missions with close-up observations. To this end, multiple near-Earth asteroid rendezvous missions can help reduce the cost of the mission. Given the enormous number of asteroids, this work proposes a method based on artificial neural networks (ANNs) to quickly estimate the transfer time and cost between asteroids using low-thrust propulsion. The neural network output is used in a sequence search algorithm based on a tree-search method to identify feasible sequences of asteroids to rendezvous. The rendezvous sequences are optimized by solving an optimal control problem for each leg to verify the feasibility of the transfer. The effectiveness of the presented methodology is assessed through sequences of asteroids of interest optimized using two low-thrust propulsion systems, namely solar electric propulsion and solar sailing. The results show that ANNs are able to estimate the duration and cost of low-thrust transfers with high accuracy in a modest computational time.

Nomenclature

a	=	semi-major axis, m
a_c	=	sail characteristic acceleration, m/s^2
a_{max}	=	low-thrust maximum acceleration, m/s^2
b	=	ANN biases
e	=	eccentricity
f, g	=	in-plane modified equinoctial elements
$\mathcal{F}^l$	=	ANN activation function
i	=	inclination, deg
I_{sp}	=	specific impulse, s
j, k	=	out-of-plane modified equinoctial elements

*Ph.D. Candidate, James Watt School of Engineering, James Watt (South) Building; g.viavattene.1@research.gla.ac.uk.
†Lecturer, James Watt School of Engineering, James Watt (South) Building; Matteo.Ceriotti@glasgow.ac.uk. Member AIAA.

J = performance index

L = true longitude, deg

m = spacecraft mass, kg

M = mean anomaly, deg

$\mathbf{N}$ = thrust direction vector

n_{rev} = number of revolutions

p = semilatus rectum, m

$\mathbf{r}$ = Sun-spacecraft position vector, m

t = time, s

T_{max} = maximum thrust, N

$\mathbf{u}$ = control vector

$\mathbf{x}$ = state vector

w = ANN weights

$\mathbf{y}$ = ANN output vector

$\mathbf{y}_t$ = ANN target output vector

α = cone angle, deg

ΔV = velocity increment, km/s

$\lambda_{1,2,3}$ = shaping parameters

μ = gravitational parameter, m^3/s^2

ϕ = phasing parameter, deg

Ω = longitude of the ascending node, deg

ω = argument of periapsis, deg

I. Introduction

Since the 1960s, scientists and engineers have dedicated a great effort to the study of near-Earth asteroids (NEAs), given the significant role they played in the geological and biological evolution of Earth, the possible exploitation of asteroids' resources, and Earth protection from future collisions. Indeed, potentially hazardous asteroids (PHAs) can impact Earth along their orbits, representing a potential threat to the life on Earth [1].

Asteroids are characterized by very irregular sizes, shapes, densities, compositions and magnetic fields. That is the reason why close-up observations of these objects are necessary to improve the knowledge of the diversity of these objects and to support any future mitigation action. To this end, NEA rendezvous is one of the most important objectives in space exploration. To reduce the cost for each observation and increase the possibility of visiting multiple asteroids of

interest in a single mission, multiple NEA rendezvous (MNR) missions are preferred [2–5].

MNRs are highly demanding in terms of ΔV, thus the more suitable propulsion system is a low-thrust propulsion, since it can deliver the same ΔV using less propellant than high-thrust systems [6]. Low-thrust propulsion considered in this paper is solar electric propulsion (SEP) [7, 8] and solar sailing [9–11].

The preliminary design of MNR missions is a complex global optimization problem that can be decomposed into two parts: a large discrete combinatorial part and a continuous part. The first one aims at identifying the most advantageous sequence of asteroids, while the second one requires the numerical solution of an optimal trajectory control problem to find the optimal control law for the trajectory with minimum propellant mass expenditure and/or minimum time of flight.

According to the NASA's database[*], almost 22,000 NEAs have been discovered until now. Among those objects, almost 2,000 are characterized by an Earth Minimum Orbit Intersection Distance (MOID) lower than 0.05 AU and an estimated diameter greater than 150 m. These are classified as PHA. To identify the best sequences of NEAs, all the permutations of asteroids should be considered and analyzed up to a certain number of asteroids in the sequence, since the scientific community gives no clear priorities in the selection of NEAs to design the mission. It follows that trillions of permutations between these objects need to be investigated. In general, low-thrust transfers have no analytical closed-form solution, thus a numerical optimization strategy must be used to find a solution to the trajectory design problems, which are generally computationally demanding and time consuming.

Given the large number of asteroids discovered until now and the enormous number of permutations of objects in a sequence, designing such a mission represents a significant challenge. Six out of eight Global Trajectory Optimization Competitions (GTOC)[†] deal with asteroid-related problems.

Different methodologies have been presented to solve this problem. Most of them propose to identify the sequence of asteroids to visit by means of a simplified model and, successively, convert the solutions found into feasible low-thrust trajectories. A solution, for instance, to the fourth GTOC competition[‡] to search for the largest number of NEAs to visit within a given mission duration using low-thrust, involved the definition of the sequences by means of impulsive thrusts before performing the final low-thrust optimization of the sequences[§]. Differently, Peloni et al. [2] used a shape-based approach to approximate the low-thrust trajectories and determined the sequences through a search-and-prune algorithm. This approach allowed them to design a mission with five NEAs as encounters within 10 years using the near-term solar-sail technology [2].

In previous studies, machine learning was used successfully to resolve very complex problems in the aerospace sector. A method based on an evolutionary algorithm and Artificial Neural Networks (ANNs) was employed to determine

[*]Data available through the link https://cneos.jpl.nasa.gov/orbits/elements.html (accessed on 2020-01-10)

[†]Data available through the link https://sophia.estec.esa.int/gtoc_portal/ (accessed on 2020-01-10)

[‡]Bertrand, R., Epenoy, R., and Meyssignac, B., "Final Results of the 4th Global Trajectory Optimisation Competition," 2009, available online at https://sophia.estec.esa.int/gtoc_portal/wp-content/uploads/2012/11/gtoc4_problem_description.pdf (accessed on 2020-01-10)

[§]Bertrand, R., Epenoy, R., and Meyssignac, B., "Final Results of the 4th Global Trajectory Optimisation Competition," 2009, available online at http://mech.math.msu.su/~iliagri/gtoc4/gtoc4_final_results.pdf (accessed on 2020-01-10)

trajectories to a single NEA using solar sailing [12]. This method was demonstrated to be more efficient in finding solutions than the traditional optimal control methods. Hennes et al. [13] used machine learning methods to identify the low-thrust trajectory with minimum fuel mass required to perform transfers between the main belt asteroids. Mereta et al. [14] compared the accuracy of various machine learning techniques, including an ANN, to determine the initial guess of optimal low-thrust transfers between NEAs. Other applications of neural networks also include the solution of optimal control problems [15], where the network is trained to identify the optimal control history in different cases of pinpoint landing, and orbit prediction [16], where historical orbit determination and prediction data are used to train the network to improve the orbit prediction accuracy. An ANN was also employed in the determination of multi-target missions using solar sails with a characteristic acceleration $a_c = 0.75$ mm/s^2 [3]. However, the latter appears to be considerably higher than what is considered "near-term" ($0.2 \leq a_c \leq 0.35$ mm/s^2), according to ENEAS+ (Exploration of NEAs with solar Sailcraft) mission studies [17], the Gossamer Roadmap technology study [18], and the research by Peloni at al. [2].

Other works on interplanetary low-thrust trajectory optimization have been conducted. The mission design can be formulated as an hybrid optimal control problem [19], with an outer-loop solving the discrete optimization problem and an inner loop solving the continuous optimization problem. The outer loop utilizes the "null-gene" transcription [20] and a discrete genetic algorithm (GA). The inner loop utilizes the Sims-Flanagan transcription [21] combined with the monotonic basin hopping global search algorithm [22]. The Sims-Flanagan transcription has also been used in combination with a global optimization heuristic to generate an initial guess for a medium-high fidelity optimization [23]. This allows a trajectory designer to rapidly evaluate a wide range of solutions to the complex problem of identifying low-thrust, multiple-flyby trajectories. Also, the global, low-thrust, interplanetary trajectory optimization problem has been solved using an hybridization of a GA and a gradient-based direct method [24]. Since it removes the difficulty and biases due to the initial guess generation, this method can provide near globally optimal solutions. Similarly, an automatic trajectory planning approach is developed by [25, 26], where a modified Ant Colony Optimization algorithm is used to find the optimal sequences of multiple gravity assists.

The purpose of this paper is to investigate how artificial intelligence and, in particular, ANN can be used to quickly estimate the transfer cost and duration between asteroids and, together with a method to select the sequences of targets, design a MNR using *near-term* low-thrust propulsion systems.

Once trained, a neural network can potentially *quickly* explore any possible transfer between pairs of asteroids, eliminating the need of considering pruned or restricted sets of asteroids. Moreover, a well-trained network can potentially predict the cost and duration of interplanetary transfers, reducing considerably the computational time and effort with respect to the optimization techniques. This is particularly important given the large number of NEAs and the extensive amount of permutations between them to define a sequence of encounters.

There are multiple challenges in using an ANN for the design of MNRs. Firstly, the selection of network inputs

affects the complex trajectory optimization problem; thus, different parametrizations of the orbit need to be investigated. Also, the phasing of the asteroids, which is essential for rendezvous and fly-by transfers, should be accounted for in these inputs. Secondly, defining the topology and the hyper-parameters of the network for this particular application is not straightforward and requires the definition and solution of an optimization problem. Also, it is paramount to identify the most appropriate method to generate the training database, containing a sufficiently large number of samples, and the strategy for the training of the network. The paper will discuss how these challenges are addressed.

The paper is organized as follows. In Sec. II, a neural network is designed, trained and optimized to quickly estimate the cost and duration of interplanetary transfers. The training database is generated collecting optimized NEAs rendezvous transfers. The sequence search algorithm to determine the most suitable sequences of asteroids is described in Sec. III, while in Sec. IV the optimization method, used to verify the feasibility of the sequences, is illustrated. In Sec. V, the effectiveness of the method is demonstrated by optimizing the selected sequences of asteroids of interest. Finally, Sec. VI concludes the paper with a summary of the results.

II. Neural Networks for Transfer Cost Mapping

ANNs are very complex computing systems inspired by the biological neural networks which constitute animal brains [27]. In the effort to emulate the cognitive abilities of biological organisms, ANNs can learn and model complex non-linear relationships. In theory, multi-layered feedforward ANNs have the exceptional capability to approximate any non-linear mapping to any degree of accuracy [28]. Thus, ANNs can also be trained to calculate, in a fraction of time needed with optimal control solvers, the cost of a trajectory in terms of ΔV and time of flight (TOF), given the departure and arrival orbits. The estimates provided by the network can help to select candidates for the most convenient sequence of asteroids to visit, which will consequently be further analyzed through optimization methods. Thanks to their generalization property, ANNs can learn the function that relates inputs and outputs in the training database and generalize this function to predict the cost and duration of transfers between asteroids and departure dates, which were not included in the training [16]. It follows that any possible permutation of any set of NEAs can potentially be examined.

The aim of this paper is to explore how ANNs can be used to find the best sequence of multiple asteroids to visit during a MNR mission using a near-term low-thrust propulsion system. To this end, the ANN is trained using a database of low-thrust transfers, which are obtained using a shape-based method. The trained network is then employed within a sequence search algorithm which uses a tree-search method to search for the candidate optimal sequences of asteroids. The sequences which best satisfy the mission requirements are transcribed into the collocation polynomials using the Radau pseudospectral method [29][30] and converted into low-thrust trajectories using the non-linear programming (NLP) solver IPOPT [31]. Lastly, the feasibility of the sequences is assessed by means of local optimization techniques.

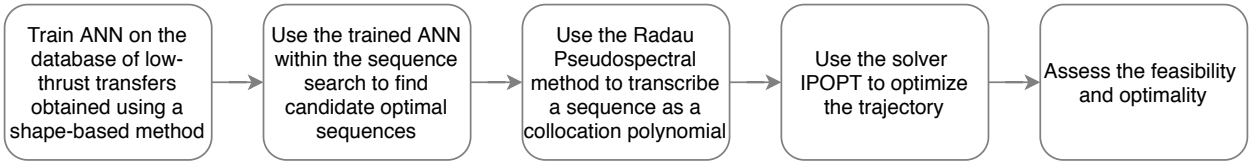


Fig. 1 Flow of the steps of the proposed optimization methodology.

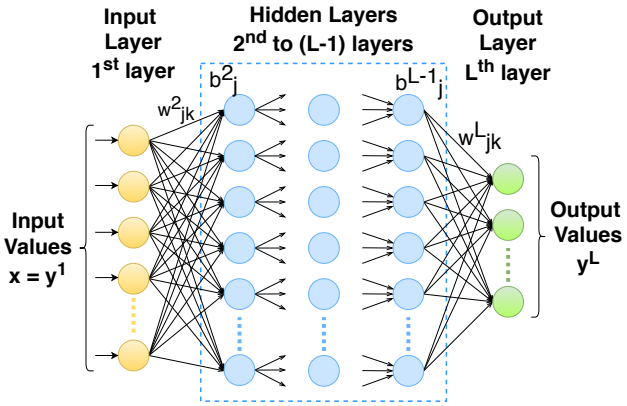


Fig. 2 Illustration of an ANN with L layers.

A schematic flow showing the steps of the proposed optimization method is given in Figure 1.

A wide variety of ANNs exists to be suitable for multiple applications, however, only feedforward ANNs are considered in this study. A network comprises of a number of neurons, which are organized in layers. A generic illustration of a neural network is shown in Figure 2. Each neuron is connected directly to the neurons of the successive layer, so that the information moves from the first layer, known as *input* layer, through the *hidden* layers to the last layer, known as *output* layer. In the following the working principle of ANNs is summarized. For a more detailed explanation, the interested reader is referred to Refs. [32, 33]

To learn and model the complex non-linear relationship between the departure and arrival orbits and the cost of the transfer between the two, the network needs to be trained by means of an appropriate database containing the corresponding inputs and outputs (or *targets*). The training of the network aims at identifying the *network function* by minimizing the *network error* between the outputs generated by the network and the targets. To determine this function, the weights w^l_{jk} associated to the connection between the k-th and j-th neuron and the biases b^l_j of the j-th neuron of the l-th layer need to be adjusted in order to minimize the network error. The j-th neuron of the l-th layer equals to:

$$y^l_j = \mathcal{F}^l \left(\sum_k w^l_{jk} y^{(l-1)}_k + b^l_j \right) \tag{1}$$

where subscripts and superscripts identify the neuron and hidden layer, respectively. $\mathcal{F}^l$ is the activation function of the

l-th layer, which maps from the neuron's weighted input values onto a single output value. The *sigmoid* function, which is commonly used for feedforward networks, can be defined as follows:

$$s_\gamma(x) = \frac{1}{1 - e^{-x/\gamma}} \tag{2}$$

where γ defines the slope of the function. In this way, each j-th neuron is computed by means of the Eq. 1, as follows:

$$y_j = \frac{1}{1 - e^{-\left(\sum_k w_{jk}^l y_k^{(l-1)} + b_j^l\right)/\gamma_j}} \tag{3}$$

with y_j being the output of the j-th neuron and $y_j \subseteq (0, 1)$. Differently, each i-th neuron of the first (input) layer provides one component of the input vector, $x_i \subseteq \mathbb{R}$, unchanged, thus $\mathbf{y^1} = \mathbf{x}$. In this case, the network function is a parametrized function that maps from an input set $\mathcal{X}$ to an output set $\mathcal{Y}$:

$$N = \mathcal{X} \subseteq \mathbb{R}^{n_i} \rightarrow \mathcal{Y} \subseteq (0, 1)^{n_0} \tag{4}$$

with n_i and n_0 being the number of input and output neurons, respectively.

The performance of the network can be affected by varying the number of layers and neurons for each hidden layer and also by other parameters of the network, such as the learning algorithm, activation function, learning rate, or even the type of parameters contained in the input vector. It is essential to identify the best combination of the network's architecture and parameters to minimize the network error for this application of transfer cost and time mapping. In particular, to design an optimal neural network, the training database generation is presented in Sec. II.A, and the network's architecture and hyper-parameters are tuned in Sec. II.B.

A. Training Database Generation

The training database includes the inputs to the network and the desired outputs, which are the orbital characteristics of the departure and arrival orbits and the position of the objects along their orbits (inputs) and the cost ΔV and TOF of the transfer between these two orbits (outputs). The NEA orbital characteristics are obtained from the NASA's near-Earth Object Program[¶]. NEAs of particular interest from the scientific point of view, for their composition and orbital dynamics are included in the database.

Among these, the database includes Potentially Hazardous Asteroids (PHAs) and near-Earth Object Human Space Flight Accessible Targets Study (NHATS) objects [34], selected by NASA as NEAs that might be accessible by future human space flight missions. These asteroids are selected depending on the characteristics of the asteroids themselves and of the mission required to reach them, departing from Earth. Different asteroids are chosen if different constrains

[¶]Data available through the link `https://ssd.jpl.nasa.gov/sbdb_query.cgi#x` (accessed on 2021-01-17)

Table 1 Criteria used in the selection of the NHATS database.

Parameter	Lower Bound	Upper Bound
Total ΔV, km/s	–	12
Mission duration, days	–	450
Stay time at the object, days	8	–
Launch date	2020	2045
Absolute magnitude parameter	–	26
Orbit condition code	–	7

are specified for the mission parameters, i.e, the total ΔV required, total mission duration, stay time at the object and launch date interval. The criteria used for this work to select the NHATS database[‖] are presented in Table 1. Note that the absolute magnitude parameter H indicates the visual magnitude an observer would record if the asteroid were placed 1 AU away, and the orbit condition code (OCC) takes into account the orbit determination accuracy.

To improve the converge rate of low-thrust optimal control problems, the highly inclined ($i \geq 20°$) and highly eccentric ($e \geq 0.4$) objects are excluded. These criteria result in 6286 asteroids, with about 300 PHAs and about 1450 NHATS. The reference time of 2019-04-27 (MJD 58600) is used to determine the position of the asteroids along their orbits. Although 6286 objects are considered for the simulations and the preliminary design of MNRs, the permutations between only 100 NEAs are considered for the generation of the network training database, for a total of 10,100 transfers. The purpose of this is to verify the *generalization* propriety of the neural network. Indeed, if the training of the network is successful, the ANN is able to generalize to provide the transfer costs between NEAs, which are not included in the training database and with different launch dates.

To determine the target outputs of the network, i.e., the cost and duration of a low-thrust transfer, a low-thrust optimal control problem needs to be solved. Solving an optimization problem is generally very computationally expensive and requires an accurate first guess to identify an appropriate solution [35]. To compute a trajectory solution with a reduced computational time and effort, a shape-based approach is used in this study to generate the training database. In this case, the trajectory is defined analytically and there is no optimization involved, so there is no guarantee that the solution found is optimal. However, the shape of the minimum-cost rendezvous trajectory can be approximated for the given range of launch dates with zero departure and arrival velocity, TOF, and the number of revolutions [36][37]. Thus, this choice is appropriate for the preliminary search to design MNRs.

The shape-based method defines a trajectory $\mathbf{r}(t)$ as a parametrized analytical curve connecting two points in a central force field [38, 39]. The necessary acceleration that the propulsive system needs to provide to fly the calculated trajectory is obtained by computing the control thrust required to satisfy the dynamics, as follows:

[‖] Data available through the link `https://cneos.jpl.nasa.gov/nhats/` (accessed on 2021-01-07)

Table 2 Bounds definition for the parameters used in the shape-based method.

Parameter		Lower Bound	Upper Bound
Launch date		2020-01-01	2030-12-30
TOF, days		400	1500
Acceleration, mm/s^2		–	0.1
Propellant ratio		–	0.6
N. revolution		0	4
Shaping parameters	λ_1	-0.5	0.5
	λ_2	-0.1	0.1
	λ_3	-0.01	0.01

$$\mathbf{a} = \ddot{\mathbf{r}} + \mu \frac{\mathbf{r}}{r^3} \tag{5}$$

where μ is the gravitational parameter of the central body, i.e., the Sun, and $\mathbf{r}$ is the position vector in the Cartesian reference frame.

The shape of the trajectory can be defined using the following linear-trigonometric relationship:

$$\mathbf{x} = \mathbf{x_0} + \mathbf{x_1}(L - L_0) + \boldsymbol{\lambda} \sin(L - L_0 + \phi) \tag{6}$$

which defines the solution of the full 3-D trajectory based on a set of modified equinoctial elements $\mathbf{x} = [p, f, g, h, k, L]^T$. The vector $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \lambda_3]$ contains the shaping parameters to be determined using a genetic algorithm, and $\mathbf{x}_0$ and $\mathbf{x}_1$ are defined from the boundary conditions, and the phase parameter ϕ is empirically set.

The bounds of the parameters used in the shape-based algorithm are specified in Table 2, considering a SEP system with a specific impulse I_{sp} of 3000 s and maximum thrust T_{max} of 0.1 N. For each individual transfer between two selected bodies, the shape-based method is run. The training database is built by storing, for each transfer, the parametrization of the orbits of the departure and arrival asteroids, the angular position of the relative asteroids at the departure date, and the cost and TOF of the minimum-time transfer.

For the training process the database is divided into three sets: training, validation and test. The training set is used to train the network and obtain the weights and biases that minimize the mean square error (MSE) between the output vector of the network $\mathbf{y}$ and the target vector $\mathbf{y}_t$. The MSE can be defined as follows [28]:

$$\mathcal{E}_{MSE} = \frac{1}{N} \sum_{i=1}^{N} ||y_i - y_{t,i}||^2 \tag{7}$$

with N being the number of nodes of the output layer of the network.

The validation and test sets aim at demonstrating the generalization performance of the network. These sets contain

Table 3 Network performance for different parametrizations of the orbit.

Orbit parametrization	Correlation	Validation-set error
COE	0.855	0.530
EE	0.856	0.487
MEE	0.925	0.236
Cartesian	0.551	0.761
Delaunay	0.694	0.862
eH	0.908	0.221

samples that are not included in the training, so they are totally new cases for the network. In particular, the validation set is used to eventually evidence the presence of overfitting during the training. Overfitting occurs when the MSE of the training database continues to decrease, but the MSE of the validation set starts increasing. When this condition happens for a consecutive number of iterations, the training is stopped. In this case, the weights and biases are set equal to those for which the minimum of the validation error is registered. The test set is used only after the training process is concluded to evaluate the performance of the network with totally new situations. The training algorithm used and the database division into training, validation and test sets are discussed in the next section.

B. Network Architecture and Parameter Tuning

The aspects that are considered in designing the network are mainly twofold: the type of inputs to be fed into the network, and the architecture and parameters of the network itself. There exists a combination of these aspects that provides the optimal performance of the network, however this is not known *a priori* and needs to be determined by trial and error.

1. Network Input Analysis

The input to the network is a parametrization of the departure and arrival orbits and the position of the objects along their orbits, between which the cost of the transfer is to be computed as the output. The literature offers different parametrizations of the orbit, which are convenient depending on the application. Among these, the classical orbital elements (COE), the modified equinoctial elements (MEE) that are used to build the training database, equinoctial elements (EE), Cartesian coordinates, Delaunay elements, and eccentricity and angular momentum vectors (eH) [40, 41]. This section studies how the different orbital parametrizations affect the network performance.

To this end, a network with two hidden layers and 80 neurons in each layer is used. The activation function chosen is the *sigmoid* function and the *gradient-descent* algorithm is adopted for the training. The learning rate is set to 0.01, which is the highest value that does not cause divergence in the training process. The batch size is 200, and the database is divided so that 70% constitute the training set, 15% the validation set, and 15% the test set [42]. The performance of the network, in terms of correlation and validation-set error, is presented in Table 3 for the considered parametrizations

Table 4 GA search space and optimal values for the network hyper-parameters.

ANN Parameter	Search space	Optimal value
Number of hidden layers	2,3,4	4
Number of neurons of each hidden layers	[70, 100]	80
Activation function of each hidden layers	sigmoid, tansig	sigmoid
Learning algorithm	Levemberg-Marquardt	Levemberg-Marquardt
Learning rate	[0.001, 0.1]	0.01
Batch size	[100, 1000]	200
Database division rates (training : validation : test)	70:15:15, 80:10:10	70:15:15

of the orbit.

The best correlation is obtained when MEE are used as inputs ($C_{MEE} = 0.925$), which also presents a low validation-set error ($e_{MEE} = 0.236$). The latter is slightly lower when the eH parametrization is used, but a poorer correlation is registered in this case. However, the priority is given in this case to the highest correlation, since this is indeed representing the performance of the network in all the three training, validation and test phases. Thus, for the remainder of this paper, MEE are used to describe the departure and arrival orbits as inputs to the network.

2. Network Architecture & Parameter Analysis

The methodology followed to design the network architecture and tune its hyper-parameters is as follows. First, the parameters which mostly affect the performance of the network are identified. This is done by studying the effects of changing one network parameter at a time and analyzing the response of the network. Secondly, these parameters are optimized so that the performance of the network improves. Thereafter, the solutions which show the highest correlation are further analyzed to define the values of the network parameters which generally ensure a better performance. In this way the solution space can be reduced for future analyses. Finally, the solution that is considered optimal is identified.

The preliminary analysis shows that the parameters which mostly influence the accuracy of the network results are the number of hidden layers, the number of neurons and the activation function for each hidden layer, learning algorithm, learning rate, batch size and database division rates. The momentum constant and the L2 regularization parameter do not play a significant role in defining the performance of the network. The optimization of all these parameters requires to search over a large solution space. For this reason, a global optimization procedure is preferred over a grid search being an exhaustive computationally expensive approach. The network parameters are set as solution vector of a GA with the intent of finding the near-optimal solution, which maximizes the correlation coefficient (*fitness function*). The search space of the GA is detailed in Table 4. After the optimization, some values of the parameters appear to be more likely to give a correlation coefficient higher than 0.9. These are shown in Figure 3.

The network architecture and parameters which offer the best performance are presented in Table 4. A network with

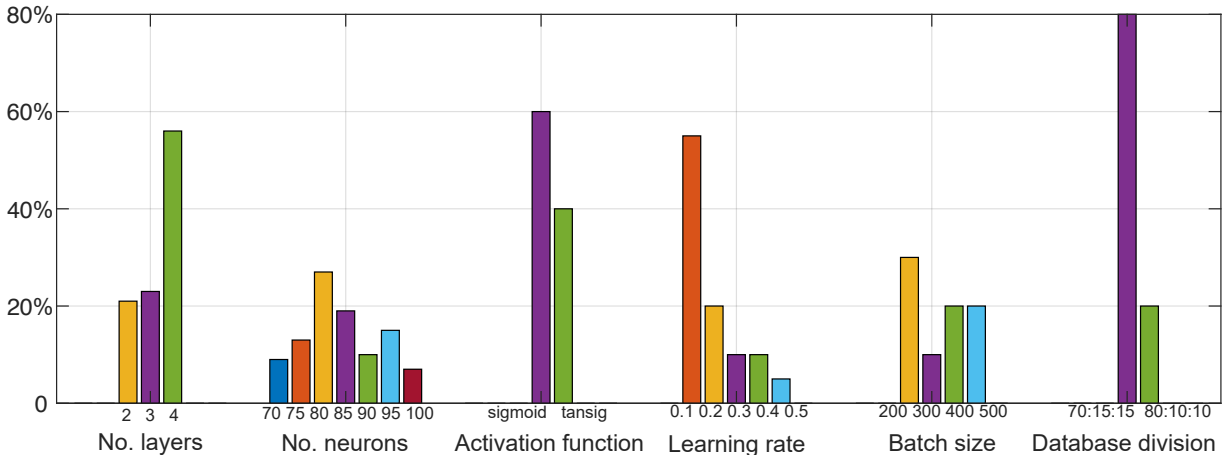
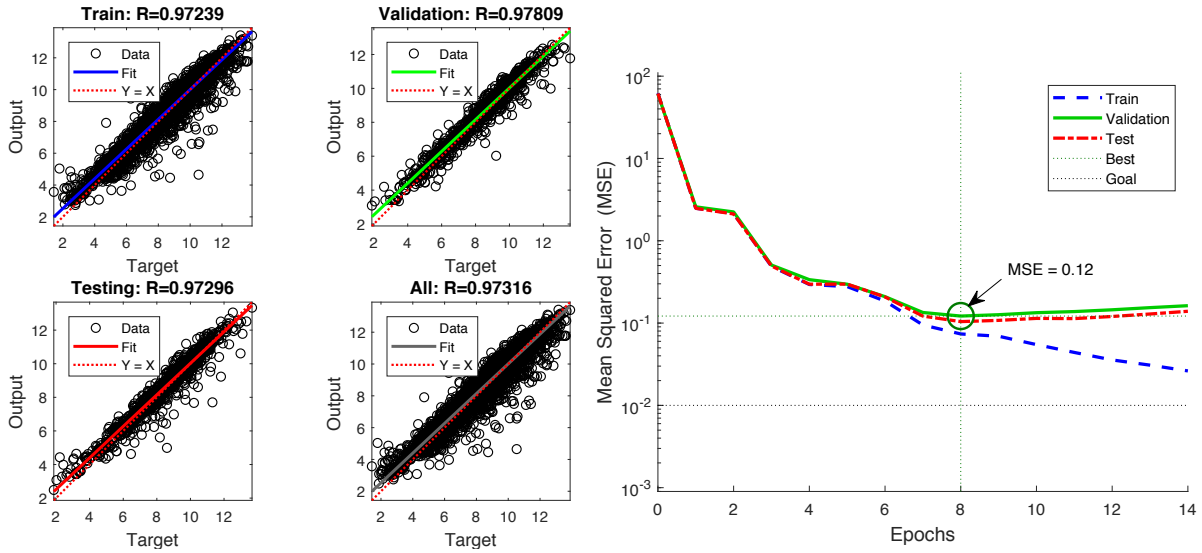


Fig. 3 Representation of the likelihood of each parameter value to give a correlation coefficient $R \geq 0.9$.



(a) Regression analysis

(b) MSE variation during the training

Fig. 4 Regression analysis (a) and performance (b) of the network with the best architecture and parameters identified by the GA.

four hidden layers and 80 neurons and the sigmoid activation function for each hidden layer is built. The database of 10,100 low-thrust transfers is divided into 70% of training set, 15% validation set, and 15% test set. Figure 4 illustrates the performance of the selected network. The regression plots in Fig. 4(a) present how well the network outputs (Y-axis) fit the targets (X-axis) with respect to the training, validation, test sets, and all of them combined. Note that the outputs and targets are ΔV and TOF, so their normalized values are presented in the plot. A perfect fit, thus a perfect performance of the network, is obtained when the data fall along the line with a unit slope and zero y-intercept. This means that the relationship between the outputs and the targets is $\mathbf{y} = \mathbf{x}$. The final correlation coefficient R obtained is

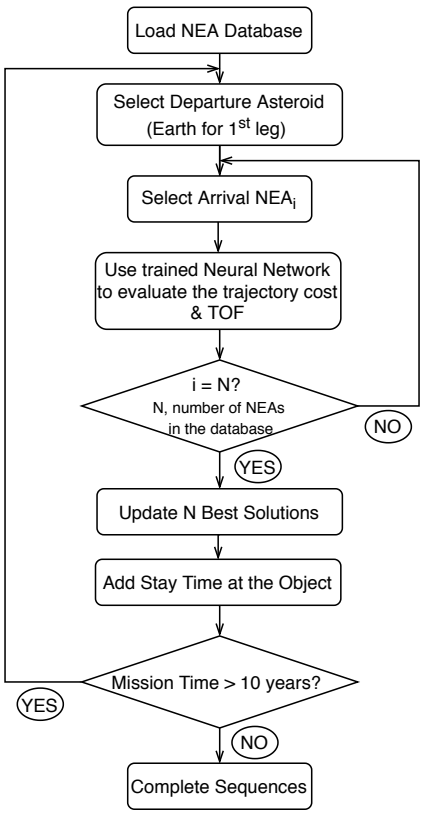


Fig. 5 Sequence search algorithm.

about 0.97, which indicates a very accurate fitting.

Figure 4(b) shows how the MSE of the training, validation and test datasets changes along the training epochs. The network weights and biases are set equal to those that provided the lowest MSE for the validation set, which occurs in this case at epoch 8 and is equal to 0.12. This indicates that the network function accurately describes the relationship between departure and final orbits and cost of the transfer between the two. Since the fitting accuracy is equal to 0.97 also for the validation and test sets, it can be deduced that the network can generalize well to new cases not experienced during the training. It can be concluded that, to a first approximation, a neural network can replace the complex optimization process required to compute a low-thrust transfer to identify its cost and duration.

III. Sequence Search Algorithm

To identify the most promising sequences of asteroids to visit in a MNR mission, the sequence search algorithm, schematically illustrated in Fig. 5, is implemented. The sequence search starts from Earth at departure date 2035-01-01. The departure date is chosen to be outside of the time frame used to compute the transfers included in the training of the network, so this serves also to test the generalization property of the network. Firstly, the full database of NEAs is loaded. The ephemerides of the asteroids are updated at each departure time t_i, with i indicating the i^{th} leg of the

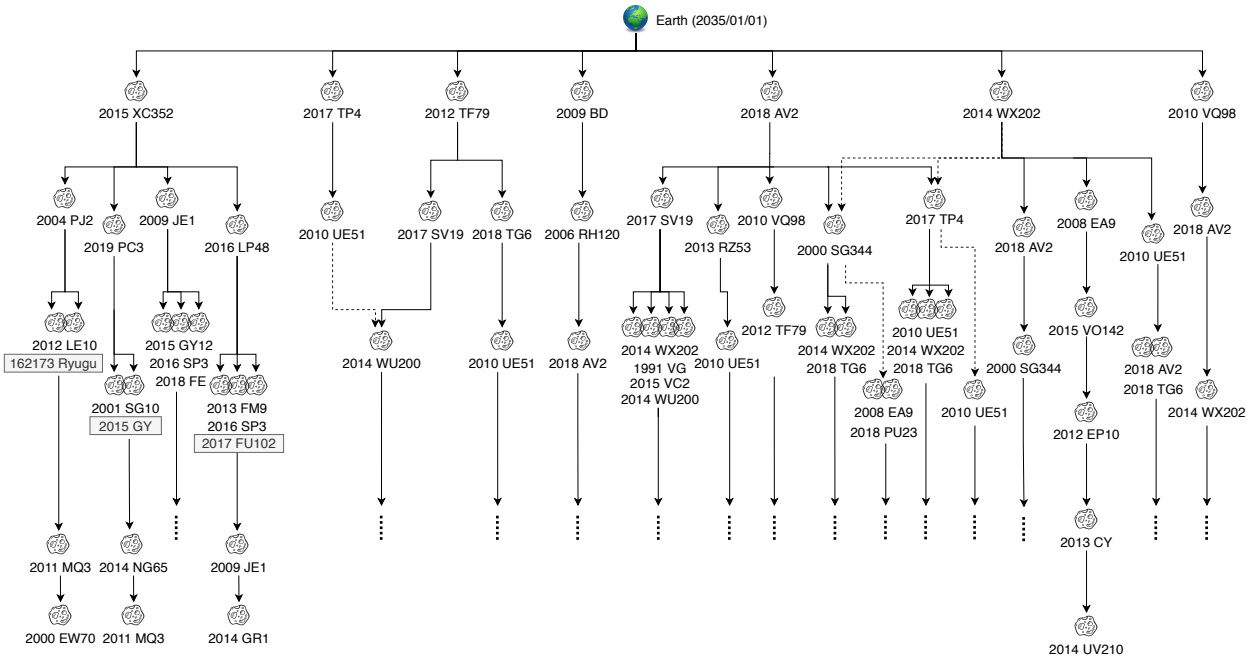


Fig. 6 Tree graph of the sequences which were found by the sequence search algorithm for the launch date 2035-01-01.

sequence. Secondly, the trained ANN is used to calculate the cost and the TOF of each transfer from Earth to all the NEAs available in the database.

Once all the transfers are evaluated, only N_{max} = 200 of the best transfers in terms of the lowest TOF are stored. This limit of N_{max} best transfers is set to reduce the otherwise enormous number of permutations given the large number of NEAs. A stay time at the object of 100 days is added to allow for close-up NEA observations. At this point, the arrival object becomes the departure object of the following leg, for which the same procedure is iterated following a *tree-search method*. Each node of the tree represents a trajectory and how one proceeds through its branches depends on the mission objective which, in this case, is the TOF minimization. The sequence is complete once the total mission duration exceeds 10 years.

In this work the launch date is fixed as the purpose of the algorithm is to prove the effectiveness of the use of ANN to design low-thrust, multiple asteroid missions. This can be complemented with a systematic search on several launch dates if a launch window is given. Figure 6 shows the obtained sequences visiting more than five asteroids which were found for departure date 2035-01-01. Here only the first three legs are fully displayed because the number of asteroids visited grows rapidly with each leg. For illustration, only the sequences which are analyzed and optimized in this paper are fully shown.

IV. Sequence Optimization

Once the NEA multiple rendezvous sequence is found, an optimal control problem (OCP) is formulated and solved to calculate the 3-D low-thrust trajectory. In this study, two types of low-thrust propulsion systems are considered: solar electric propulsion (SEP) and solar sailing (SS).

The state vector of the system, $\mathbf{x}$, is expressed in modified equinoctial elements, adjoined by the spacecraft mass for SEP:

$$\mathbf{x_{SEP}} = [p, f, g, h, k, L, m]^T \tag{8}$$

Since there is no significant variation in the SS case, the spacecraft mass m is not included in the state vector, which is then defined as follows:

$$\mathbf{x_{SS}} = [p, f, g, h, k, L]^T \tag{9}$$

The following set of ordinary differential equations of motion describes the equations of the dynamics:

$$\dot{\mathbf{x}}(t) = \mathbf{A}(\mathbf{x})\mathbf{a} + \mathbf{b}(\mathbf{x}) \tag{10}$$

with $\mathbf{a}$ being the acceleration generated by the propulsion system, which is considered to be available at any time and regardless of the Sun distance, and $\mathbf{A}(\mathbf{x})$ and $\mathbf{b}(\mathbf{x})$ being, respectively, the matrix and the vector of the dynamics. A full definition of $\mathbf{A}(\mathbf{x})$ and $\mathbf{b}(\mathbf{x})$ can be found in Ref. [43]. The propulsive acceleration is different depending on the propulsion system considered and it can be described as follows:

1) for SEP

$$\mathbf{a} = \frac{T_{max}}{m}\mathbf{N} \tag{11}$$

where T_{max} is the maximum thrust that can be generated from the considered propulsion system, and $\mathbf{N} = [N_r, N_\theta, N_h]^T$ indicates the acceleration direction and magnitude vector in radial, transverse, out-of-plane coordinates. The mass of the spacecraft m changes with time while it is thrusting as described by the following mass differential equation:

$$\dot{m} = -\frac{T_{max} \cdot |\mathbf{N}|}{I_{sp} g_0} \tag{12}$$

with $|\mathbf{N}|$ being the magnitude of $\mathbf{N}$, which accounts for the thrust throttling, and I_{sp} being the specific impulse of the relative propulsion system. In this case, $I_{sp} = 3000$ s.

2) for SS

$$\mathbf{a} = a_c \left(\frac{r_E}{r}\right)^2 \cos^2(\alpha)\hat{\mathbf{N}} \tag{13}$$

where the term a_c is the so-called characteristic acceleration of the sail and it expresses the acceleration provided by a solar sail facing the Sun at the average Sun-Earth distance, i.e., $r_E = 1$ AU. The term r indicates the magnitude of the position vector with respect to the Sun, while the cone angle α is the angle between the Sun-spacecraft direction and and the sail normal unit vector, which is $\hat{\mathbf{N}} = [N_r, N_\theta, N_h]^T$.

The OCP consists in finding the optimal control history $\mathbf{u}(t) \equiv \mathbf{N}(t)$ so that the time of flight is minimized. Thus, the performance index to minimize is:

$$J = \int_{t_0}^{t_f} dt \tag{14}$$

while satisfying the dynamics defined in Eq. (10) and the path constraints as follows:

$$\begin{aligned} 0 < ||\mathbf{N}|| < 1 \qquad & \text{for SEP} \\ ||\hat{\mathbf{N}}|| = 1 \qquad & \text{for SS} \end{aligned} \tag{15}$$

where $\mathbf{N}$ can vary for SEP to allow for thrust throttling. The control vector is bounded so that N_r, N_θ, $N_h \in [-1, 1]$ for SEP, while N_θ, $N_h \in [-1, 1]$ and $N_r \in [0, 1]$ for SS to take into account the inability of the solar sail to thrust towards the Sun. Moreover, since the intent of this work is to design MNRs, the state vector needs to be bounded to ensure that the rendezvous conditions are satisfied, which are:

$$\begin{cases} \mathbf{r}(t_d) - \mathbf{r}_d(t_d) = \mathbf{0} \\ \mathbf{v}(t_d) - \mathbf{v}_d(t_d) = \mathbf{0} \end{cases} \quad \text{and} \quad \begin{cases} \mathbf{r}(t_a) - \mathbf{r}_a(t_a) = \mathbf{0} \\ \mathbf{v}(t_a) - \mathbf{v}_a(t_a) = \mathbf{0} \end{cases} \tag{16}$$

where $\mathbf{r}$ and $\mathbf{v}$ are the position and velocity of the spacecraft, $\mathbf{r}_d$ and $\mathbf{v}_d$ are the position and velocity of the departure body at the departure time t_d, and $\mathbf{r}_a$ and $\mathbf{v}_a$ are the position and velocity of the arrival body at the arrival time t_a, respectively, as taken from the ephemerides, as specified in Sec. II.A.

Given the sequences found through the sequence search, the algorithm implemented to find the optimal low-thrust trajectories performs as follows:

1) The algorithm automatically computes the initial guess for each leg separately by solving a Lambert problem, given the departure orbit, arrival orbit and desired time of flight, which is provided by the ANN during the sequence search. When solar sailing is used, the solution obtained for the SEP case is adopted as the initial guess.
2) The algorithm optimizes the 3-D trajectory leg by leg. During the optimization phase, the solution of each leg

is constrained to start at least 20 days after the arrival of the previous leg. This is done to avoid overlapping between two consecutive legs and allow some time for close-up observations.

3) The optimized multiple NEA rendezvous trajectory is built if at least one feasible solution is found for each leg of the mission.

Each leg is optimized using the TOF predicted by the neural network during the sequence search as the initial guess. Also, a window of ±100 days is considered for the departure and arrival time t_0 and $t_f = t_0 + T$, with T being the TOF of the transfer, so that the best epochs for departure and arrival can be selected by the optimizer. The stay time at each asteroid is consequently adjusted considering the TOF of each leg and the departure date of the successive leg. The optimal control problem is solved by using a discrete non-linear programming (NLP) together with a variable-order adaptive Radau collocation method [30, 44, 45]. The NLP solver IPOPT [31] has been used within this study.

V. Multiple NEA Rendezvous Mission Design

Table 5 Orbital characteristics of the NEAs visited in the optimized sequences.

Sequence A	2015 XC352	2004 PJ2	162173 Ryugu	2011 MQ3	234145 (2000 EW70)
Classification	NHATS	PHA	NHATS	NHATS	PHA
a, AU	1.016	1.418	1.19	1.12	0.94
e, -	0.15	0.34	0.19	0.11	0.32
i, deg	4.49	2.58	5.88	5.68	5.43
Ω, deg	98.76	317.18	251.62	274.03	178.06
ω, deg	83.48	281.68	211.43	301.14	125.72
M, deg	18.90	296.80	3.98	259.48	283.31
Orbit Class	Apollo	Apollo	Apollo	Apollo	Aten
Sequence B	2015 XC352	2016 LP48	2017 FU102	2009 JE1	2014 GR1
Classification	NHATS	-	NHATS	NHATS	NHATS
a, AU	1.016	1.05	1.29	1.37	1.30
e, -	0.15	0.40	0.29	0.29	0.28
i, deg	4.49	10.40	3.01	8.03	5.81
Ω, deg	98.76	87.26	12.69	46.15	168.58
ω, deg	83.48	51.87	231.18	223.20	89.99
M, deg	18.90	74.28	120.00	335.01	322.75
Orbit Class	Apollo	Apollo	Apollo	Apollo	Apollo

Note: M is calculated on 2019-04-27

A. NEA Sequence Analysis

The methodology and the tools described above are used to calculate the sequences of asteroids to visit with minimum TOF and, subsequently, optimize the full trajectories. To compute the N_{max} = 200 sequences, as described in

Table 6 Mission parameters of the optimized NEA sequence A and comparison with the values estimated by the ANN (in brackets).

Leg	Propulsion	Departure	Arrival	TOF, days	ΔV, km/s	Stay Time, days
Earth - 2015 XC352	SEP	2035-01-01	2037-06-04	885 (882)	8.86 (8.96)	97
	SS	2035-02-08	2037-04-04	746	-	20
2015 XC352 - 2004 PJ2	SEP	2037-09-09	2040-01-20	863 (878)	10.67 (11.14)	54
	SS	2037-04-24	2039-12-15	943	-	400
2004 PJ2 - 162173 Ryugu	SEP	2040-03-14	2042-09-21	920 (775)	15.03 (12.93)	100
	SS	2041-01-18	2042-10-08	628	-	338
162173 Ryugu - 2011 MQ3	SEP	2042-12-29	2045-02-11	775 (637)	12.29 (9.76)	21
	SS	2043-09-20	2045-05-13	609	-	200
2011 MQ3- 2000 EW70	SEP	2045-05-09	2047-03-17	677 (702)	10.43 (8.94)	–
	SS	2045-11-29	2047-07-28	606	-	

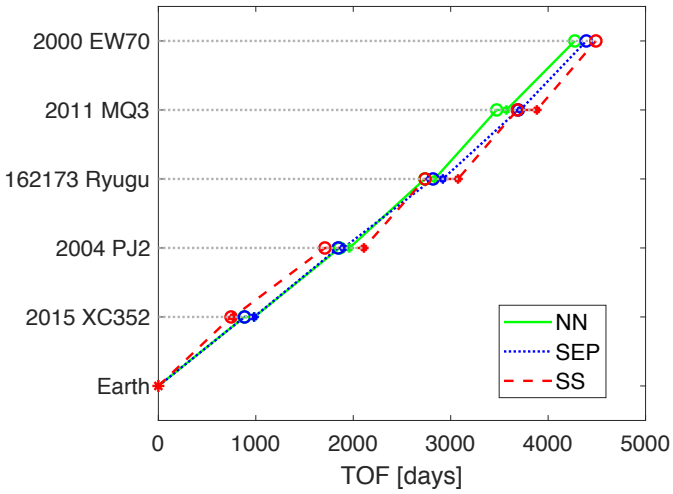


Fig. 7 NEA sequence A: TOF per leg and stay time at each asteroid.

Sec. III, the sequence search algorithm required about 13.3 hours on a machine with a Core i7 processor at 3.4 GHz.

Two sequences have been selected for the full optimization. These have been selected as they present the largest variation in semi-major axis a (Sequence A) and largest variation in both inclination i and eccentricity e (Sequence B). These variations are registered in a single leg, i.e., between one asteroid and the next asteroid in a sequence. This is preferred over considering the greatest variation of these values in total because the algorithm works on asteroid-to-asteroid transfers, which is what is desired to verify and validate. The encountered bodies of the selected sequences are characterized in Table 5. As shown, the majority of the objects are PHA and NHATS. Note that the semi-major axis, eccentricity or inclination between two consecutive objects are not changing monotonically, as shown

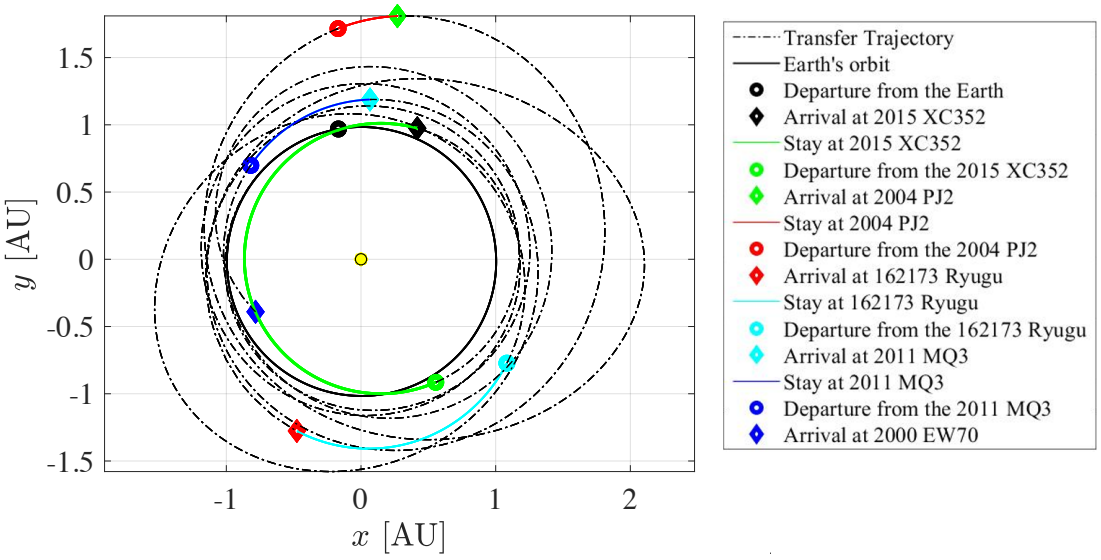


Fig. 8 NEA sequence A: heliocentric ecliptic-plane view .

Table 7 Mission parameters of the optimized NEA sequence B and comparison with the values estimated by the ANN (in brackets).

Leg	Propulsion	Departure	Arrival	TOF, days	ΔV, km/s	Stay Time, days
Earth - 2015 XC352	SEP	2035-01-01	2037-06-04	885 (882)	8.86 (8.96)	51
	SS	2035-02-08	2037-04-04	746	-	277
2015 XC352 - 2016 LP48	SEP	2037-07-25	2039-10-25	821 (673)	8.12 (9.78)	20
	SS	2037-11-28	2039-08-26	635	-	300
2016 LP48 - 2017 FU102	SEP	2039-11-14	2042-02-10	819 (859)	13.17 (12.25)	24
	SS	2040-06-21	2042-09-05	806	-	20
2017 FU102 - 2009 JE1	SEP	2042-03-06	2044-11-14	984 (804)	15.03 (13.22)	100
	SS	2042-09-26	2045-07-27	1035	-	31
2009 JE1 - 2014 GR1	SEP	2045-02-22	2046-11-26	642 (663)	9.79 (10.26)	–
	SS	2045-08-27	2047-09-26	760	-	

in the sequence of NEAs found in Ref. [2]. Indeed, the variation of these variables can be positive or negative while transferring to the next object. This is particularly interesting because it allows the system to be more flexible and visit an unrestricted range of NEAs of interest.

These trajectories are optimized according to the methodology presented in Sec. IV using two different propulsion systems: a solar-electric propulsion system with a maximum thrust $T_{max} = 0.1$ N or a solar sailing system with a characteristic acceleration $a_c = 0.2$ mm/s^2. The characteristics of the MNRs for each of the selected sequences are

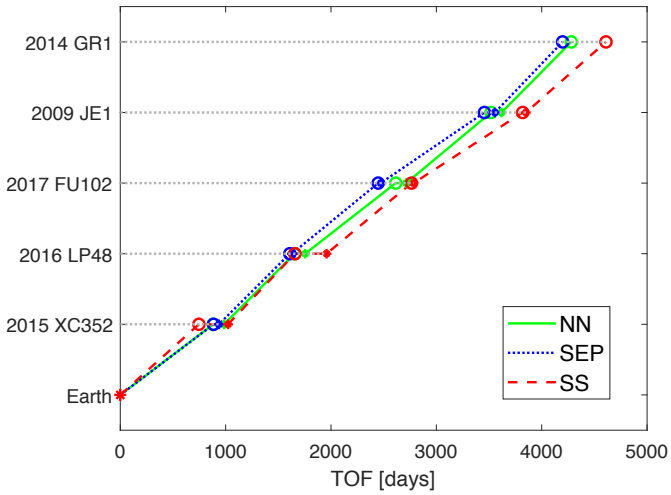


Fig. 9 NEA sequence B: TOF per leg and stay time at each asteroid.

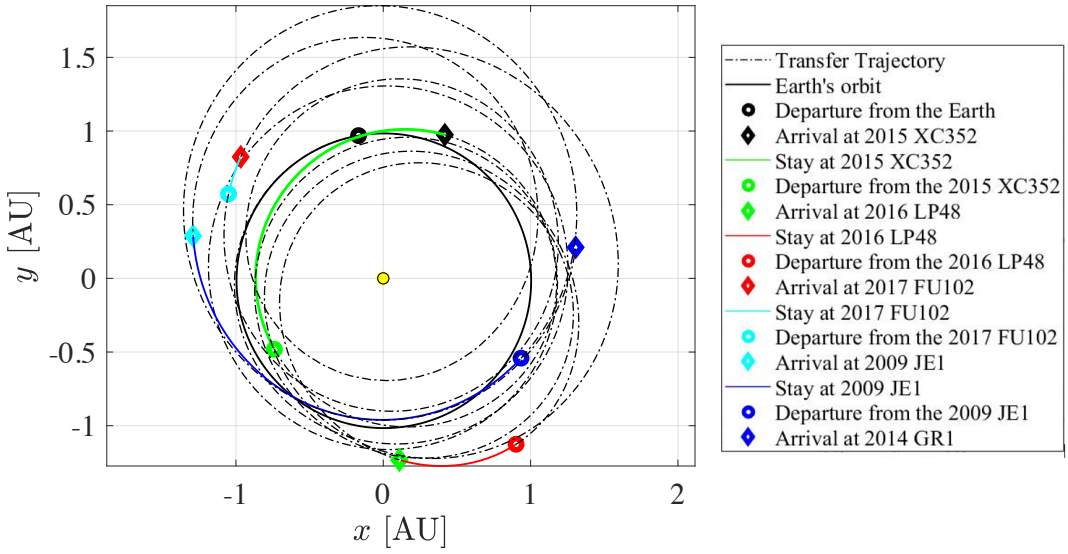


Fig. 10 NEA sequence B: heliocentric ecliptic-plane view.

reported in Tables 6 and 7, respectively. The tables report the departure and arrival date, TOF and ΔV of each leg of the sequences. Note that the values of TOF and ΔV in brackets are those found by the ANN in the sequence search algorithm, which were used as initial guess for the optimizer, and they are shown for comparison. In the sequence search, the stay time at each object was fixed to 100 days, so it is not reported in the tables.

As a visual representation may help, Fig. 7 and 9 show the NEAs visited as function of the time of flight for Sequence A and B, respectively, so that the difference in TOF between ANN, SEP and SS can be easily appreciated. For both sequences, the solar sailing case requires slightly more time to complete the mission. The sail a_c is chosen to be conservative according to Ref. [18], so generally longer stay times with respect to the SEP case are necessary to catch

Table 8 Orbital characteristics of the NEAs visited in the optimized sequences.

Sequence C	2014 WX202	2008 EA9	2015 VO142	2012 EP10	2013 CY	2014 UV210
Classification	NHATS	NHATS	NHATS	NHATS	NHATS	NHATS
a, AU	1.036	1.049	1.075	1.05	1.114	1.155
e, -	0.0589	0.0745	0.1259	0.1160	0.1342	0.1316
i, deg	0.41	0.44	0.28	1.03	0.78	0.60
Ω, deg	243.99	124.46	95.26	348.00	302.58	92.15
ω, deg	214.07	343.15	20.47	105.87	149.58	351.56
M, deg	51.76	134.18	336.18	293.59	114.38	134.91
Orbit Class	Apollo	Apollo	Apollo	Apollo	Apollo	Apollo

Note: M is calculated at 2019-04-27

up the shorter transfer times. Additionally, the network was trained for identifying the cost and duration of low-thrust SEP transfers specifically, so a larger error for the SS case can be expected.

The heliocentric ecliptic-plane view of all the feasible complete trajectories optimized are shown in Fig. 8 and 10 for Sequences A and B, respectively. Earth's orbit is plotted and the stay times at each asteroid is highlighted against the full trajectory, as specified in the legend.

The total ΔV required to complete the SEP missions are $\Delta V = 57.28$ km/s for sequence A (Table 6) and $\Delta V = 54.97$ km/s for sequence B (Table 7). Considering the first sequence flown by a system with dry mass $m_{dry} = 350$ kg, which is similar to the one of the *Deep Space 1* by NASA [46], where a SEP system is used with specific impulse $I_{sp} = 3000$ s, the initial mass m_0 of the spacecraft would be:

$$m_0 = m_{dry} e^{\left(\frac{\Delta V}{g_0 I_{sp}}\right)} \approx 2400 \text{ kg} \tag{17}$$

with g_0 being the standard acceleration due to gravity. This equates to a mass ratio $\frac{m_{dry}}{m_0} \approx 0.15$. Comparing this number with the specifications offered in one of the last GTOC problems[**], it is convenient to allow for a larger mass ratio (minimum of about $\frac{m_{dry}}{m_0} = 0.4$) for a system propelled by a low-thrust engine. This does not apply to the SS case, as it does not required the use of propellant to fly the Sequences A and B.

To take into account the requirements of multiple encountered missions when using a SEP system, the algorithm is run again including a constraint on $\Delta V \leq 5$ km/s per phase. One of the sequences obtained is selected and optimized (Sequence C). The orbital characteristics of the visited bodies are detailed in Table 8 and the mission parameters are presented in Table 9. Also, the heliocentric ecliptic-plane view of the complete trajectory is plotted in Fig. 11. In Sequence C, six asteroids are visited in less than 10 years, requiring a total ΔV of 17.95 km/s. From Eq. (17), a mass

[**]Casalino, L., and Colasurdo, G., "Problem Description for the 7th Global Trajectory Optimisation Competition," Global Trajectory Optimisation Portal, 2014. Available through the link https://sophia.estec.esa.int/gtoc_portal/wp-content/uploads/2014/09/gtoc7_problem_description.pdf (accessed on 2021-01-07)

Table 9 Mission parameters of the optimized NEA sequence C and comparison with the values estimated by the ANN (in brackets).

Leg	Propulsion	Departure	Arrival	TOF, days	ΔV, km/s	Stay Time, days
Earth - 2014 WX202	SEP	2035-01-21	2036-07-24	550 (570)	2.50 (3.13)	87
2014 WX202 - 2008 EA9	SEP	2036-09-29	2038-02-09	498 (456)	3.02 (2.80)	91
2008 EA9 - 2015 VO142	SEP	2038-05-11	2039-10-28	535 (555)	3.25 (3.04)	68
2015 VO142 - 2012 EP10	SEP	2040-01-04	2041-05-01	483 (402)	2.92 (2.97)	100
2012 EP10 - 2013 CY	SEP	2041-08-09	2042-12-15	493 (514)	2.99 (2.76)	56
2013 CY - 2014 UV210	SEP	2043-02-09	2044-07-30	537 (557)	3.27 (2.91)	–

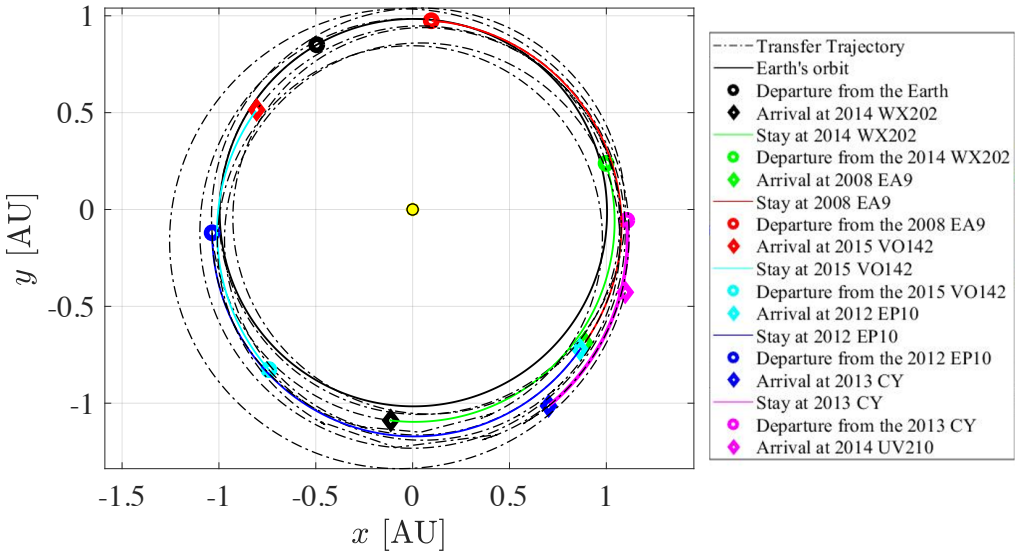


Fig. 11 NEA sequence C: heliocentric ecliptic-plane view.

fraction of 0.54 is obtained in this case. Limiting the ΔV available for each phase of the sequence allows to meet the requirement on the dry mass ratio of 0.4 and, consequently, on the maximum ΔV of the mission. In this way, the sequences identified by the ANN can be realistically flown with a SEP system.

The successful optimization of Sequences A, B, and C shows that the ANN is able to approximate well the cost and duration for low-thrust rendezvous transfers between NEAs, also when involving large changes of the orbital parameters.

B. ANN Prediction Error

From the sequences studied above, the differences in TOF and ΔV between the values obtained by the ANN and the results obtained from the trajectory optimization are generally bounded. To quantify this deviation, the average

Table 10 Average percentage error between ANN and optimized results.

Sequence	$\mathcal{E}_{TOF}$ for SEP	$\mathcal{E}_{\Delta V}$ for SEP	$\mathcal{E}_{TOF}$ for SS
A (Table 6)	7.8%	10.8%	13.7%
B (Table 7)	8.9%	9.1%	13.1%
C (Table 9)	6.7%	9.8%	–
Mean Error	**7.8%**	**9.9%**	**13.4%**

percentage error is computed considering only the TOF and ΔV for each SEP transfer, since these are the only two parameters predicted by the network. The average relative error for TOF and ΔV are calculated as follows:

$$\mathcal{E}_{TOF} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{|TOF_{i,opt} - TOF_{i,ANN}|}{TOF_{i,opt}}\right) \tag{18}$$

$$\mathcal{E}_{\Delta V} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{|\Delta V_{i,opt} - \Delta V_{i,ANN}|}{\Delta V_{i,opt}}\right) \tag{19}$$

with N being the number of legs in the trajectory. The average percentage errors are shown for each of the sequences studied in Table 10. When considering the error computed for sequence A, B, and C together, the mean error overall is 7.8% for TOF and 9.9% for ΔV. This suggests that an ANN can obtain a satisfactory accuracy for the preliminary orbit design.

This confirms the validity of the methodology and the capability of the neural network designed and trained as described in Sec. II to predict the cost and duration of a transfer given the orbits of the departure and target asteroids.

Larger differences in TOF and ΔV with respect to the average values can sometimes occur. This is due to the fact that in the sequence search algorithm the stay times of 100 days and, consequently, the departure times of each leg ($t_0 = T_0 + t_{stay}$) are imposed *a priori*, with T_0 being the mission duration until that point and t_{stay} the stay time. Differently, in the trajectory optimization process the stay time adapts to allow for the most convenient departure and arrival points depending on the relative position of the two bodies. This results in a difference in departure date, arrival date, and stay time which inevitably affects the computed TOF and ΔV of a transfer.

As an additional note, the deviation between ANN and SS optimization are also reported in Table 10, showing an overall error of 13.4%. The error is slightly higher compared to SEP, as expected, because the network was specifically trained with SEP transfers. However, this is still an acceptable error demonstrating that this methodology also allows for preliminary solar sailing mission design with near-term sail performance.

C. Computational Time & Accuracy Analysis

One of the advantages of using an ANN to determine the cost and TOF of the transfers between pairs of asteroids is the reduction in the computational time. To show an estimation of the computational time and performance of the

trained network, this is compared to other methods used in previous works. For instance, the sequence search algorithm used by Peloni et al. [2], where the tree of multiple-NEA trajectories is obtained by approximating the transfers using a shape-based method.

Peloni et al. performed the search on a set of launch dates spanning about 10 years with a step size of three months, starting from the launch date 28 November 2019, for a total of 41 launch dates. The input database of NEAs included only PHAs and NHATS asteroids, for a total of 1,801 objects. The propulsion system used by Peloni et al. is a solar sail with $a_c = 0.2$ mm/s^2.

The objective of the analysis is to compare the time to estimate successfully the cost of a low-thrust transfer when using a shape-based method or the ANN, and the average time required to complete a sequence-search run. The same database and the same launch dates, used in Ref. [2], are utilized here so that the same sequences can be obtained and a head-to-head comparison can be performed.

A database is generated, containing PHA and NHATS asteroids only and selecting the NHATS objects with the same criteria used in Ref. [2], resulting in a database containing a total of 2,768 objects; this is almost a thousand NEAs more than the one used in Ref. [2], due to the fact that the database used by Peloni at al. was generated in August 2015, and numerous NEAs have been discovered since. To partially take into account the decrease in the sail acceleration when it is not directed along the Sun-spacecraft direction, a SEP system with $a_{max} = 0.1$ mm/s^2, calculated from the T_{max} and the mass of the system at departure, is used in this analysis. The fact that a different propulsion system is chosen does not affect the computational time required by the sequence search algorithm, however it may have an effect on the accuracy of the results, so it will be taken into account during the accuracy analysis.

In Ref. [2] the runs were conducted on three computers: two with a 3.4 GHz Core i7 and one with a 2.3 GHz AMD Opteron 6376. Peloni et al. give an indication of the required computational time considering only the runs carried out in the latter, which is the slowest machine. During this analysis a machine with a 3.4 GHz Core i7 was used. Taking into account the performance test benchmark results of the two machines, CPU Mark[††] suggests that the machine with a 2.3 GHz AMD Opteron 6376 is roughly 20% slower in computational speed than the 3.4 GHz Core i7. Although the authors are aware that additional factors can play a role in affecting the computing speed of the machines, this approximation is sufficient to evaluate the order of magnitude of the difference in the computational power required for the two methodologies. For this reason the time required for the simulations in Ref. [2] will be improved by 20% for a fair comparison.

Firstly, the computational time required for the estimation of the cost and TOF of a low-thrust transfer is analyzed. Each successful run of the shape-based method in Peloni et al. takes on average about 60 s on 2.3 GHz AMD Opteron 6376, so we assume it takes roughly 48 s on average on the machine with a 3.4 GHz Core i7. Differently, it takes less

[††]Data available through the link https://www.cpubenchmark.net/compare/Intel-i7-3770-vs-AMD-Opteron-6376/896vs2000 (accessed on 2020-06-03)

Table 11 Speed comparison between the algorithm with the use ANN and without (i.e., the methodology proposed in Ref. [2]).

Simulation	Method in Ref. [2]	Method with ANN
Low-thrust transfer	48 s	0.49 s
Sequence-search run	33 days	7.66 hours

Table 12 Mission parameters of a low-thrust trajectory departing from Earth on 2025-04-30 and visiting five NEAs. Comparison between the values obtained by the method used in Ref. [2] and by the ANN.

Leg	Method	TOF, days	Optimal TOF, days	Error (vs. Optimal)
Earth - 2000 SG344	Ref. [2]	680	657	3.5%
	ANN	634		3.5%
2000 SG344 - 2015 JD3	Ref. [2]	500	436	14.7%
	ANN	414		5.1%
2015 JD3 - 2012 KB4	Ref. [2]	644	584	10.3%
	ANN	603		3.3%
2012 KB4 - 2008 EV5	Ref. [2]	647	576	12.3%
	ANN	639		10.9%
2008 EV5 - 2014 MP	Ref. [2]	625	560	11.6%
	ANN	606		8.2%
Average	Ref. [2]			10.5%
	ANN			6.2%

than a second (0.49 s on average) for the ANN to perform the same estimation. This already suggests the huge impact of ANN on the computational time of a sequence-search run where thousands of transfers are investigated.

In Ref. [2], for 41 launch dates the search found 4800 sequences with five encounters and at least one PHA. The authors specify that, for each launch date, a sequence-search run took on average 33 days (41.3 days on the 2.3 GHz AMD Opteron 6376) to complete. So, considering that 4800 sequences are found for 41 launch dates, we can assume that, on average, almost 120 sequences per launch date are found. For the launch date 2025-04-30 the sequence-search run with the ANN takes 7.66 hours, meaning that using an ANN within the sequence search makes the algorithm almost 100 times faster. Table 11 summarizes the comparison in the computational time between the method in Ref. [2] and the method with ANN for the simulation of a low-thrust transfer and a sequence-search run.

Acknowledging the difference in time for the computation of a single low-thrust transfer between the shape-based method used in Ref. [2] and the ANN, it is expected that as the number of sequences to analyze increases, the difference in computational time will be greater and the advantage of using an ANN will be more considerable.

Once the improvement in computational speed given by the use of an ANN is assessed, a comparison of the accuracy of the two methods is also conducted to ensure that the improvement in speed does not come at the expenses of the

solution accuracy. Peloni et al. [2] present and optimize the sequence found for the launch date 30 April 2025. The same sequence of asteroids is obtained during our sequence-search run.

The results are detailed in Table 12, in which the TOF obtained through the ANN is compared with the TOF in Peloni et al. using their shape-based method. To measure the accuracy of both methods, the estimated TOF values are compared with respect to the optimal ones. The ANN achieves a smaller error with respect to the shape-based method in Ref. [2], which can also be due to the difference in the propulsion systems used in the two cases. However, we can conclude that using a well-trained ANN maintains a good level of accuracy, while dramatically decreasing the computational time required.

VI. Conclusions

A methodology for the preliminary design of near-Earth asteroid (NEA) multiple rendezvous (MNR) missions using low-thrust propulsion systems was investigated. Artificial neural networks (ANNs) enable a quick estimation of the cost and time of flight of interplanetary low-thrust transfers, allowing for a fast analysis of thousands of transfer options. The shape-based method is used to generate the training database instead of using an optimization method, since the shape-based method is able to compute interplanetary trajectory solutions with good approximation, without the need for determining an appropriate initial guess. The parameters of the network are optimized and a genetic algorithm is used to identify the best combination of network parameters to ensure a high performance of the network. A neural network with four hidden layers and 80 neurons in each hidden layer would appear to be suitable for the transfer cost mapping starting from the orbital characteristics of the orbits of the departure and target asteroids. Indeed, the neural network so designed and trained can obtain an accurate fitting between targets and outputs with 0.97 in correlation coefficient. An ANN can be used within a sequence search algorithm to efficiently identify the sequence of asteroids to visit in MNRs. Finally, the selected sequences are optimized to verify the feasibility of the trajectory with the provided propulsion system.

To increase the possibility of visiting asteroids of interest for science, planetary defense, and technology demonstration, Potentially Hazardous Asteroids (PHAs) and near-Earth Object Human Space Flight Accessible Target Study (NHATS) asteroids are included in the database in this study. Among the sequences obtained, three of them are selected and fully optimized to present the results of this research and demonstrate the validity of this methodology. These are sequences with more than five encounters, some of which present large variations of the orbital parameters of the asteroids in the consecutive legs.

The performance of the presented methodology using an ANN for the preliminary design of MNR missions with continuous thrust is demonstrated to give good results with both solar-electric propulsion and solar sailing. A final average error of 7.8% in TOF and 9.9% in ΔV with respect to the final optimized SEP trajectories and 13.4% in TOF with respect to the final optimized SS trajectories.

Finally, the computational time and the accuracy of the proposed algorithm with the neural network is compared to other methodologies that do not involve an ANN. The analysis shows that, when using an ANN, the computational time is reduced by two orders of magnitude and that the same level of accuracy is maintained. This represents a major advantage in the preliminary design of MNRs. According to these results, using a neural network to predict the cost and duration of low-thrust transfers for the preliminary design of MNR missions is effective in significantly reducing the computational time, while still allowing for a sufficient accuracy of the solutions.

Acknowledgments

Giulia Viavattene gratefully acknowledges the support received for this research from the James Watt School of Engineering at the University of Glasgow for funding the research under the James Watt sponsorship program.

References

[1] Lissauer, J. J., and de Parter, I., *Fundamental Plantary Science*, Cambridge University Press, 2013.

[2] Peloni, A., Ceriotti, M., and Dachwald, B., "Solar-Sail Trajectory Design for a Multiple Near-Earth-Asteroid Rendezvous Mission," *Journal of Guidance, Control, and Dynamics*, Vol. 39, No. 12, 2016, pp. 2712–2724. https://doi.org/10.2514/1.G000470.

[3] Song, Y., and Gong, S., "Solar-Sail Trajectory Design of Multiple Near Earth Asteroids Exploration Based on Deep Neural Network," *Aerospace Science and Technology*, Vol. 91, 2019, pp. 28–40. https://doi.org/10.1016/j.ast.2019.04.056.

[4] Mereta, A., and Izzo, D., "Target selection for a small low-thrust mission to near-Earth asteroids," *Astrodynamics*, Vol. 2, No. 3, 2018, pp. 249–263.

[5] Izzo, D., Simoes, L. F., Yam, C. H., Biscani, F., Di Lorenzo, D., Bernardetta, A., and Cassioli, A., "GTOC5 : Results from the European Space Agency and University of Florence," *Acta Futura*, Vol. 8, No. November, 2014, pp. 45–56. https://doi.org/10.2420/AF08.2014.45.

[6] Wertz, J. R., *Orbit & Constellation Design & Management*, Springer, New York, 2009.

[7] Jahn, R. G., *Physics of Electric Propulsion*, McGraw-Hill Book Company, New York, 2006.

[8] Goebel, D. M., and Katz, I., *Fundamentals of Electric Propulsion: Ion and Hall Thrusters*, Jet Propulsion Laboratory - California Institute of Technology, California, 2008.

[9] McInnes, C. R., *Solar Sailing - Technology, Dynamics and Mission Applications*, 1999. https://doi.org/10.1007/978-1-4471-3992-8.

[10] Zeng, X., Gong, S., and Li, J., "Fast solar sail rendezvous mission to near Earth asteroids," *Acta Astronauta*, Vol. 105, 2014, pp. 40–56. https://doi.org/10.1016/j.actaastro.2014.08.023.

[11] Heiligers, J., Fernandez, J. M., Stohlman, O. R., and Wilkie, W. K., "Trajectory design for a solar-sail mission to asteroid 2016 HO3," *Astrodynamics*, Vol. 3, 2019, pp. 231–246. https://doi.org/10.1007/s42064-019-0061-1.

[12] Dachwald, B., "Optimization of Interplanetary Solar Sailcraft Trajectories," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 1, 2004.

[13] Hennes, D., Izzo, D., and Landau, D., "Fast Approximators for Optimal Low-Thrust Hops Between Main Belt Asteroids," *IEEE Symposium Series on Computational Intelligence (SSCI)*, 2016. https://doi.org/10.1109/SSCI.2016.7850107.

[14] Mereta, A., Izzo, D., and Wittig, A., "Machine Learning of Optimal Low-Thrust Transfers Between Near-Earth Objects," *Hybrid Artificial Intelligent Systems*, 2017, pp. 543–553. https://doi.org/10.1007/978-3-319-59650-1_46.

[15] Sánchez-Sánchez, C., and Izzo, D., "Real-time optimal control via Deep Neural Networks: study on landing problems," *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 3, 2018, pp. 1122–1135. https://doi.org/10.1023/B:CJOP.0000010527.13037.22, URL http://arxiv.org/abs/1610.08668.

[16] Peng, H., and Bai, X., "Artificial Neural Network–Based Machine Learning Approach to Improve Orbit Prediction Accuracy," *AIAA Journal of Spacecraft and Rockets*, Vol. 55, No. 5, 2018, pp. 1–13. https://doi.org/10.2514/1.A34171.

[17] Dachwald, B., and Seboldt, W., "Multiple Near-Earth Asteroid Rendezvous and Sample Return Using First Generation Solar Sailcraft," *Acta Astronautica*, Vol. 57, No. 11, 2005, pp. 864–875.

[18] Dachwald, B., Boehnhardt, H., Broj, U., Geppert, U., Grundmann, J., Seboldt, W., Seefeldt, P., Spietz, P., Johnson, L., Kührt, E., Mottola, S., Macdonald, M., McInnes, R., Vasile, M., and Reinhard, R., "Gossamer Roadmap Technology Reference Study for a Multiple NEO Rendezvous Mission," *Advances in Solar Sailing*, 2014, pp. 211–226.

[19] Englander, J., and Conway, B., "An Automated Solution of the LowThrust Interplanetary Trajectory Problem," *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 1, 2008, pp. 15–27. https://doi.org/10.2514/1.G002124.

[20] Englander, J., Conway, B., and Williams, T., "Automated Mission Planning via Evolutionary Algorithms," *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 6, 2012, pp. 1878—1887. https://doi.org/10.2514/1.54101.

[21] Sims, J., and Flanagan, S., "Preliminary Design of Low-Thrust Interplanetary Missions," *AAS/AIAA Astrodynamics Specialist Conference, AAS paper 99-0328*, 1999. URL https://ntrl.ntis.gov/NTRL/dashboard/searchResults/titleDetail/N20000057422.xhtml.

[22] Vasile, M., Minisci, E., and Locatelli, M., "Analysis of Some Global Optimization Algorithms for Space Trajectory Design," *AIAA Journal of Spacecraft and Rockets*, Vol. 47, No. 2, 2010, pp. 334–344. https://doi.org/10.2514/1.45742.

[23] Englander, J., Ellison, D., and Conway, B., "Global optimization of low-thrust, multiple-flyby trajectories at medium and medium-high fidelity," *Advances in the Astronautical Sciences*, , No. January, 2014, pp. 1539–1558.

[24] Vavrina, M., and Howell, K., "Global Low-Thrust Trajectory Optimization Through Hybridization of a Genetic Algorithm and a Direct Method," *AIAA/AAS Astrodynamics Specialists Conference*, , No. August, 2008. https://doi.org/10.2514/6.2008-6614.

[25] Ceriotti, M., and Vasile, M., "Automated multigravity assist trajectory planning with a modified ant colony algorithm," *Journal of Aerospace Computing, Information and Communication*, Vol. 7, No. 9, 2010, pp. 261–293. https://doi.org/10.2514/1.48448.

[26] Ceriotti, M., and Vasile, M., "MGA trajectory planning with an ACO-inspired algorithm," *Acta Astronautica*, Vol. 67, No. 9-10, 2010, pp. 1202–1217. https://doi.org/10.1016/j.actaastro.2010.07.001.

[27] Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L., and Zurada, J., *Artificial Intelligence and Soft Computing*, Springer, Zakopane, Poland, 2013. https://doi.org/10.1007/978-3-642-38658-9.

[28] Goodfellow, I., Bengio, Y., and Courville, A., *Deep Learning*, The MIT Press, Cambridge, Massachusetts, 2016.

[29] Patterson, M. A., Hager, W. W., and Rao, A. V., "A ph Mesh Refinement Method for Optimal Control," *Optimal Control Applications and Methods*, Vol. 36, No. 4, 2014, pp. 398–421. https://doi.org/10.1002/oca.2114.

[30] Patterson, M. A., and Rao, A. V., "GPOPS-II: A MATLAB Software for Solving Multiple-Phase Optimal Control Problems usign hp-Adaptive Gaussian Quadrature Collocation Methods and Sparse Nonlinear Programming," *ACM Transactions on Mathematical Software*, Vol. 41, No. 1, 2014. https://doi.org/10.1145/2558904.

[31] Wachter, A., and Biegler, L. T., "On the Implementation of an Interior-Point Filter Line-search Algorithm for Large-scale Non-linear Programming," *Mathematical Programming*, Vol. 106, No. 1, 2006, pp. 25–57. https://doi.org/10.1007/s10107-004-0559-y.

[32] Rojas, R., *Neural Networks: A Systemic Introduction*, Springer, New York, 1996. https://doi.org/10.1016/0893-6080(94)90051-5.

[33] Caudill, M., and Butler, C., *Naturally Intelligent Systems*, MIT Press, Cambridge, London, 1990.

[34] Barbee, B. W., Esposito, T., Pinon, E., Hur-Diaz, S., Mink, R. G., and Adamo, D. R., "A Comprehensive Ongoing Survey of the Near-Earth Asteroid Population for Human Mission Accessibility," *AIAA Guidance, Navigation and Control Conference, AIAA Paper 2010-8368*, 2010.

[35] Alemany, K., and Braun, R. D., "Survey of Global Optimization Methods for Low- Thrust, Multiple Asteroid Tour Missions," *17th AAS/AIAA Space Flight Mechanics Meeting, AAS 07-211*, 2007, pp. 1–20.

[36] Taheri, E., and Abdelkhalik, O., "Shape Based Approximation of Constrained Low-Thrust Space Trajectories using Fourier Series," *Journal of Spacecraft and Rockets*, Vol. 49, No. 3, 2012, pp. 535–546.

[37] Li, S., Zhu, Y., and Wang, Y., "Rapid Design and Optimization of Low-Thrust Rendezvous/Interception Trajectory for Asteroid Deflection Missions," *Advances in Space Research*, Vol. 53, No. 4, 2014, pp. 696–707.

[38] Petropoulos, A. E., and Longuski, J. M., "Automated Design of Low-Thrust Gravity-Assist Trajectories," *AIAA Paper 2000-4033*, 2000.

[39] De Pascale, P., and Vasile, M., "Preliminary Design of Low-Thrust Multiple Gravity-Assist Trajectories," *AIAA Journal of Spacecraft and Rockets*, Vol. 43, No. 5, 2006, pp. 1065–1076. https://doi.org/10.2514/1.19646.

[40] Wiesel, W., *Spaceflight Dynamics: Third Edition*, CreateSpace, Cambridge, Massachusetts, 2017.

[41] Wakker, K. F., *Fundamentals Of Astodynamics*, Faculty of Aerospace Engineering, Delft University of Technology, 2015.

[42] Bengio, Y., "Practical Recommendations for Gradient-Based Training of Deep Architectures," *Springer Lecture Notes in Computer Sciences*, Vol. 7700, No. Neural Networks: Tricks of the Trade, 2012, pp. 437–478. https://doi.org/10.1007/978-3-642-35289-8_26.

[43] Betts, J. T., *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, 2nd ed., SIAM Press, Philadelphia, 2010.

[44] Rao, A. V., Benson, D. A., Darby, C. L., Patterson, M. A., Francolin, C., Sanders, I., and Huntington, G. T., "Algorithm 902: GPOPS, a MATLAB Software for Solving Multiple-Phase Optimal Control Problems using the Gauss Pseudospectral Method," *ACM Transactions on Mathematical Software*, Vol. 37, No. 2, 2010. https://doi.org/10.1145/1731022.1731032.

[45] Garg, D., Patterson, M. A., Francolin, C., Darby, C. L., Huntington, G. T., Hager, W. W., and Rao, A. V., "Direct trajectory optimization and costate estimation of finite-horizon and infinite-horizon optimal control problems using a Radau pseudospectral method," *Computational Optimization and Applications*, Vol. 49, No. 2, 2011, pp. 335–358. https://doi.org/10.1007/s10589-009-9291-0.

[46] "Deep Space 1 Asteroid Flyby," *NASA, Press Kit*, 1999.