On Space Trajectory Optimization Using Hidden-Genes Genetic Algorithms

Shadi A. Darani[a] and Ossama Abdelkhalik[b]
Michigan Technological University, Houghton, Michigan 49931-1295.

I. Introduction

SPACE trajectory optimization is the process of searching for the optimal trajectory from one celestial body or orbit to another, such that the mission requirements are satisfied and a given objective is optimized. The objective can be minimizing the mission cost or fuel consumption, minimizing the mission duration, maximizing the number of visited asteroids, or a combination of these objectives. The earliest research on space trajectory optimization goes back to the work of Walter Hohmann on trajectory design of a spacecraft with impulsive thrusters between two coplanar orbits [1]. Cornelisse [2] showed that in the patched conics method, the cost of an interplanetary trajectory mission can be reduced by applying a deep-space maneuver (DSM). Several works have studied the effect of DSMs in different space missions [3–6]. Planetary flybys utilize the gravity of a planet to change the momentum vector of a spacecraft. Such trajectories that use DSMs and flybys are called Multi-Gravity-Assist-Deep-Space-Maneuver (MGADSM) trajectories.

To design a MGADSM interplanetary trajectory, many variables should be optimized depending on the mission type, such as launch and arrival dates and times, number of flybys, planets to flyby, number of DSMs, epoch of each DSM, direction and magnitude of each DSM, time of flight (TOF) between each two successive celestial bodies (leg), and flyby altitudes and rotation angles. These variables can be categorized into two groups of discrete design variables and continuous design variables, as shown in **Table 1**. Since some variables are related to others (e.g. flyby attitude

[a] PhD Candidate, Mechanical Engineering-Engineering Mechanics Department, 815 R.L. Smith Bldg, 1400 Townsend Dr, Houghton, MI 49931, USA; sahmadid@mtu.edu
[b] Associate Professor, Mechanical Engineering-Engineering Mechanics Department, 815 R.L. Smith Bldg, 1400 Townsend Dr, Houghton, MI 49931, USA; ooabdelk@mtu.edu; Member AIAA

depends on whether there is a flyby or not), the problem can be considered as a variable-size design space problem (VSDS), in which the number of optimization variables vary among different solutions. In other words, the number of flybys and DSMs are not known a priori and they determine the number of other variables needed to model the problem. These variables that determine the total number of variables in a solution are referred to as the architecture variables.

Table 1: Design variables in an interplanetary trajectory optimization problem

Discrete Variables	Continuous Variables
Number of flybys (m)	Departure date (t_d)
Flyby planets (P)	Arrival date (t_a)
Number of DSMs in each leg (n)	TOF
	Flyby pericenter altitude (h_p)
	Flyby rotation angles (η)
	DSMs epoch (ϵ)
	DSMs magnitudes and directions

Many global optimization methods have been investigated in different MGADSM problems, including heuristic algorithms [7–12], deterministic algorithms [13–15], or a combination of them [16, 17]. Deterministic methods use grid or tree search to explore the design space. Although these methods converge globally, they can be exhaustive, especially in more complex missions with high number of flybys/rendezvous and DSMs or large time windows. The obtained solutions also are usually sensitive to the grid size. Heuristic methods on the other hand do not need discretization of the search space and are more adaptive and hence are not usually exhaustive. Yet they rely on heuristics and parameters tuning. Genetic algorithms (GAs) [18–22], differential evolution [12, 23–25], and ant colony optimization [26] are some of the heuristic algorithms that have been proven to be efficient in MGADSM optimization problems.

The genetic algorithms are among the most popular heuristic methods in space trajectory design. The standard GAs assume the design variables of a solution as genes in a fixed-length chromosome.

By applying the evolutionary operations of selection, mutation, and crossover, the population of these chromosomes converges to the global optimal solution [27]. Since in general the flyby and DSM structures are not known a priori, it is not possible to use the standard GA for such problems without simplifications in the problem or modification to the algorithm [7, 28, 29]. One way of simplifying the problem is to prune the state space (assume fixed flyby sequence and number of DSMs) to limit the possible mission scenarios. A deterministic search is used in [29, 30] where the flyby sequence and DSMs are fixed and the search space is limited to a grid of points where the global optimization methods can be used. Another way of simplifying the problem is to use a nested loop solver to optimize the trajectory [8, 31]. The outer loop finds the optimal flyby sequence and the inner loop optimizes the trajectory for that scenario. Since not all the scenarios have the same number of flybys, this problem is a VSDS optimization. Early methods pruned the outer loop to solutions that the designer considered to include the optimal flyby sequence [32]. Later, automatic methods were proposed to find the flyby sequence in MGA trajectories. Some graphical methods use the energy contours against two variables that define the orbits for different planet flybys [33, 34]. This method can be used when all the flybys are considered non-powered and it is assumed that there is no DSMs. In [31] a maximum length for the flyby sequence is assumed and the outer loop is optimized using a binary genetic algorithm. By adding null variables that represents a "no flyby", variable-sized flyby sequences can be modeled in this method. For example, for a maximum flyby of two, the Earth-Venus-Mars (EVM) sequence is equivalent to a mission from Earth to Mars with a flyby around Venus and a null flyby that is not considered in the cost function. Genetic algorithm is also used for multiple phase maneuvers where there is both impulsive and continuous maneuvers [8]. Genetic Programming (GP) [35] is also among the earliest approaches that addressed the VSDS optimization problems. One of the earliest attempts in implementing gene expression in GA is to perform “cut and splice" on the chromosomes and applying a self adaptive recombination operator on them to yield individuals of variable lengths [36, 37]. In recent years, the role of histone in the regulation of DNA including gene expression and functionality of each cell was discovered [38], which resulted in the use of epigenetics through modification of histone in strongly-typed genetic programming [39].

Fig. 1: Solutions are represented as chromosomes (string of genes) in standard GA.

Inspired by the concept of gene expression in biology, the concept of Hidden Genes Genetic Algorithm (HGGA) was introduced to search for the optimal architecture and autonomously generate new design spaces [40, 41]. Reference [41] applied a simplified version of the HGGA for interplanetary trajectory optimization and demonstrated success in finding the best known solutions architectures for known benchmark problems. This original version of the HGGA implemented in [41] assumes a long chromosome for each solution where some of the genes are hidden; this approach is briefed below in Section I B. The HGGA implementation in [41], however, lacks a rigorous method for selecting the hidden genes in each generation. Recently, reference [42] presented mechanisms for selecting the hidden genes using tags that are appended to genes. Several evolution mechanisms of the tags were investigated in [42]. In this paper, four new mechanisms are introduced for tags evolution. The interplanetary trajectory optimization problem is solved using the proposed HGGA mechanisms and two different bench mark problems are presented. These problems include missions from Earth to Jupiter and Saturn. **Sections** I A and I B present necessary background on genetic algorithms and the concept of hidden genes. **Section** II presents HGGA tags evolution mechanisms introduced in this paper, **Section** III briefs the problem formulation for the interplanetary trajectory optimization. **Section** IV presents the numerical test cases on the two trajectory optimization benchmark problems.

A. Genetic Algorithm

In standard GAs, the variables of the optimization problem are coded in chromosomes. Each chromosome represents a solution and consists of the variables that are coded as genes. The objective of optimization determines the fitness of the solution. In **Figure** 1, a solution with N variables is shown as a chromosome with N genes $g_1, g_2, \ldots, g_N$. The genetic operations of selection, mutation and crossover are applied on a population of these chromosomes, and through generations (iterations), theses populations converge toward the optimal solution.

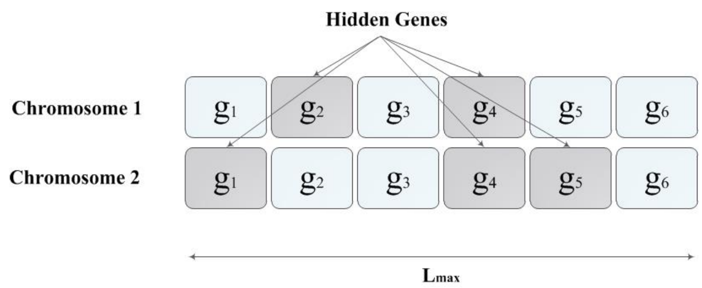


Fig. 2: Added hidden genes in two different chromosomes [40]

In the selection operation, two chromosomes are selected as parents from the generation pool. In general, the chromosomes that have better fitness values (objective function), have higher probability to be selected as parents. After the parents are selected, mutation and crossover operations are applied on them. As an example, in binary coding, genes are coded as 0s and 1s; in the mutation process gene 1 may change to 0 with a probability of p_m. In the crossover operation, parts of the chromosome strings are swapped in parents. For example, in single point crossover, a random point is selected in both parents and the genes of one side of that point are swapped in parents with a crossover probability of p_c to create new chromosomes. Some of the best chromosomes (elites) are transferred to the next generation with no change. By repeating the GA operations in each generation, the population converges to the optimal solution.

B. Hidden Genes Genetic Algorithm

To handle a VSDS (or architecture) optimization problem, the idea of turning genes on and off was adapted from biology in genetic algorithm. By setting the chromosome length equal to the length of the longest possible chromosome L_{max} (maximum number of design variables) and turning some genes off, different solutions (of different architectures) with lengths of 1 to L_{max} can be built while having the same length for all the chromosomes. Moreover, having similar lengths for all the chromosomes enables the implementation of the standard GA operations like crossover and mutation on them. The genes that are hidden are variables that do not affect the fitness of the solution; yet they carry information, go through GA operations, and may become active (not hidden) in future generations.

Assume that in a VSDS problem, two possible solutions are of lengths four and three, and the maximum number of variables allowed in this problem is six. To make it a Fixed-Size Design Space (FSDS) problem, two hidden genes are added to the first solution and three hidden genes are added to the second solution to make both chromosomes of length six (**Figure** 2). In this way, the added hidden genes do not affect the objective function and the chromosomes still represent the same solutions as those without the hidden genes; yet both the chromosomes now have similar lengths and standard GA operations can be applied to them. In other words, the GA starts by producing an initial population of chromosomes of length six, of which some of the genes may be hidden. Then, the selection, mutation, and crossover operations are applied on them to produce the next generation of chromosomes of length six, of which some of the genes may be hidden again. This algorithm is called Hidden Genes Genetic Algorithm. Based on the mechanisms that assign the hidden genes, an active gene may become hidden in the next generation and vice versa. The assignment mechanisms play an important role in how the HGGA evolves and affects the efficiency and convergence rate of the HGGA.

As an example, a single-point crossover operator in HGGA is shown in **Figure** 3. In this example, the genes go through the crossover operator regardless of the hidden genes positions, meaning that the genes are swapped at a random crossover point (in **Figure** 3 it is between the second and the third genes) with probability p_c just like a normal crossover operator. Then, some genes in the children chromosomes are assigned as hidden based on the hidden gene assignment mechanism. Depending on the mechanism, the hidden genes positions might not be similar to those of the hidden genes positions in the parents. The assignment mechanisms are explained in **Section** II.

II. Hidden Genes Evolution Mechanisms

As discussed in the previous section, the hidden genes assignment logic has an important role in the performance of the HGGA. The original work on HGGA [40] used a feasibility logic for assigning the hidden genes, in which all genes are assumed active unless the solution is infeasible. In the case of infeasible solutions, the algorithm hides genes, one by one, until the chromosome represents a feasible solution. In a more recent work, logical and stochastic operations were used to assign the hidden genes in a current generation based on their hidden/active status in the previous generation

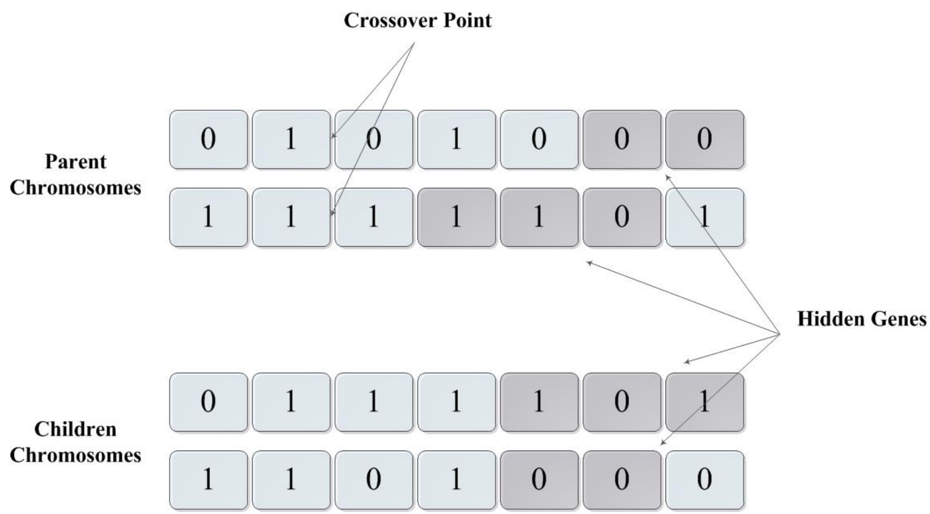


Fig. 3: An example for crossover in HGGA [40]

[42]. To implement this concept, a tag tag^i was assigned to each gene x_i as shown in **Figure** 4. The value of this binary tag determines whether the corresponding gene is hidden or active. This concept of tags is also found in biology where the protein (in the histone) of each gene determines if it is hidden or active. In this implementation, the tag plays the role of the histone. Reference [42] developed stochastic and logical operations for tags evolution. As an example, one of the effective logical operations is the Active OR logic, which assumes a tag is active in a current generation if it is active in any of its two parents, and it is hidden otherwise.

In this paper, two approaches for tags evolution are developed. The first approach exploits the tags concept, and develops new operations for evolving these tags over generations. The second approach presents the concept of using two tags (dominant and recessive Alleles) as agents that determine the status of the genes. These two approaches are detailed below.

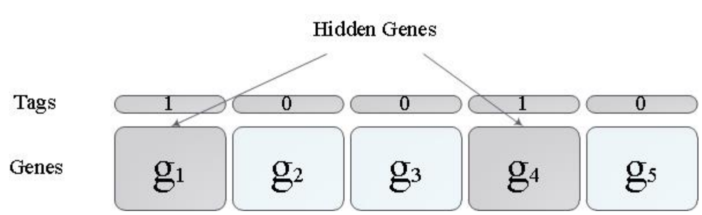


Fig. 4: The concept of tags in HGGA

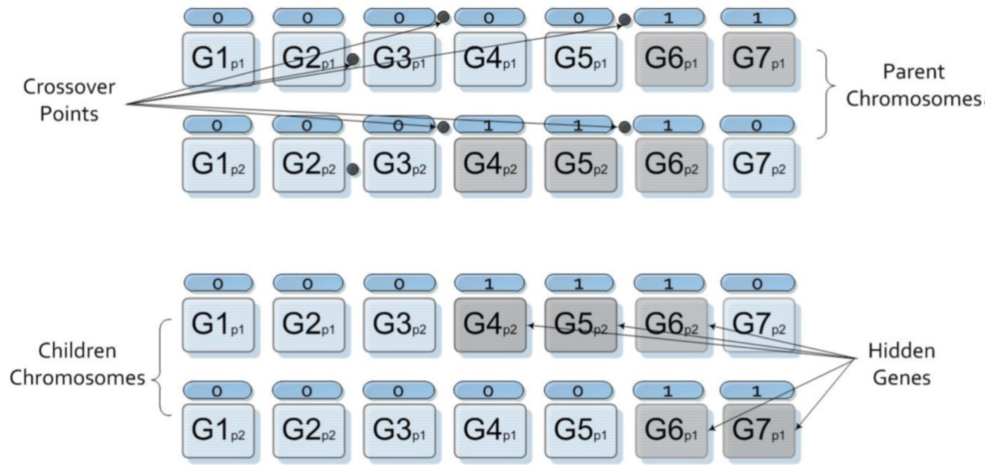


Fig. 5: Schematic of the Stochastic Mechanism

A. Single Tag

Using the concept of tags, four new evolution mechanisms are investigated in this study. These mechanisms utilize stochastic and/or logical methods to assign hidden genes in subsequent generations. These mechanisms are as follows:

1. Stochastic Mechanism

Similar to genes, the tags undergo mutation and crossover operations. The crossover points and mutated locations in tags, however, are different from those of the genes. **Figure 5** shows the schematic of the Stochastic Mechanism, where the tags undergo a two-point crossover while the genes undergo a single-point crossover. The crossover points for both genes and tags are chosen stochastically. Simulations were conducted to investigate the impact of the tags mutation probability on the optimization efficacy. In this investigation, the tags mutation probability was varied from 10^{-5} to 0.1. A mutation probability of 0.01 is found to yield the most fit solutions.

2. Logical Mechanism

This mechanism has two steps. After the selection of two parents, two temporary chromosomes are produced through a single-point crossover operation on genes, and an Active-OR logic on tags (as shown in **Figure 6**). These two intermediate chromosomes have the same tags. The fitness value of these two temporary chromosomes, J_1 and J_2, are calculated. The child (the offspring of the Logical Mechanism) is then computed as the weighted arithmetic crossover of the parents chromosomes and is closer to the parent whose intermediate chromosome has a better fitness J. The tags of the offspring child are calculated using the Active-OR logic on the parents tags. Let C be the gene string of the final offspring, λ be a random number between 0 and 1, P_{t_1} and P_{t_2} be the genes strings of the parents, then the gene string of the final offspring is:

$$C = \begin{cases} 0.5[(1+\lambda)P_{t_1} + 0.5(1-\lambda)P_{t_2}], \text{if } J_1 > J_2 \\ 0.5[(1-\lambda)P_{t_1} + 0.5(1+\lambda)P_{t_2}], \text{if } J_1 < J_2 \end{cases} \tag{1}$$

3. Short Mechanism

This mechanism exploits solutions that are more fit and shorter (better fitness value and more hidden genes). A fitness guided crossover is used for the genes with a modified objective function that is a function of the design variables in addition to the number of hidden genes in a solution. The tags are obtained by Hidden-OR operator on the parents tags. Consider a minimization problem, the modified objective function for the crossover operator would be:

$$f_{mod}(\vec{x}) = f(\vec{x}) - \sum_{i=1}^{M}(tag_i). \tag{2}$$

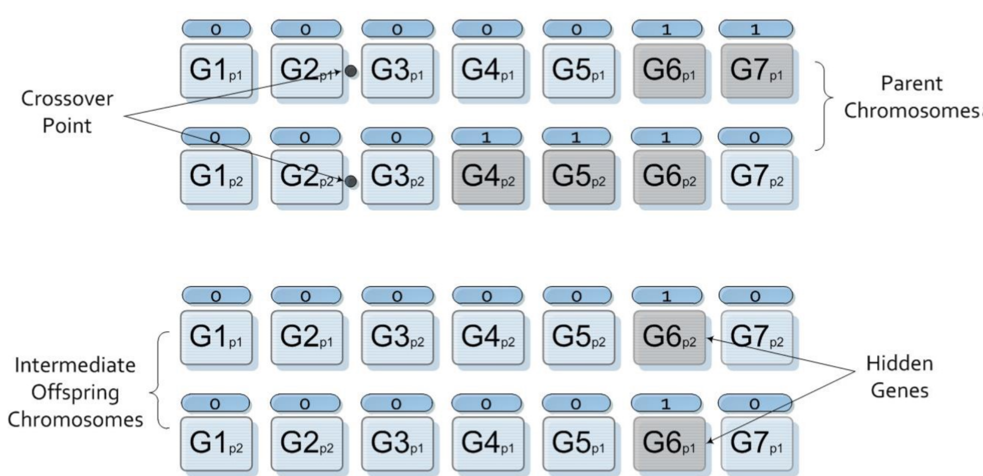


Fig. 6: Intermediate Chromosomes in The Logical Mechanism.

where M is the number of tags, $\vec{x}$ is the vector of variables (chromosome), and $f(\vec{x})$ is the fitness function. Since the tags can only have values of 1 and 0, solutions with better fitness and more hidden genes would be more fit. The offspring would be created based on **Equations** 1, with J_1 and J_2 evaluated as follows:

$$J_1 = f_{mod}(P_{t_1}) \tag{3}$$

$$J_2 = f_{mod}(P_{t_2}) \tag{4}$$

4. Long Mechanism

This mechanism exploits solutions that are more fit and longer (better fitness value and less hidden genes). In this case, the fitness guided crossover is applied on genes and the tags are obtained by Active-OR operator on the parents tags. The modified objective function for the genes crossover operator in a minimization problem would be:

$$f_{mod}(\vec{x}) = f(\vec{x}) + \sum_{i=1}^{M}(tag_i). \tag{5}$$

B. Two Tags (Alleles)

In this concept, the HGGA is developed by simulating Alleles and considering two tags for each gene, one recessive and one dominant. In biology, an allele is an alternative form of a gene. In human cells, there can be two Alleles (dominant or recessive) of a gene in each position on a chromosome. Dominant traits are expressed when the individual has one copy of the allele. On the other hand, the recessive traits are expressed only if the Alleles of a pair are homozygous (the individual has two copies of the allele). These principles and their traits were first discovered by Gregor Mendel [43, 44], and is named as Mendel's Law of Segregation.

This idea is adapted and modified to be applied on GA. Two sets of tags are considered for each chromosome, called Alleles. One allele is dominant and one is recessive and only the value of the dominant allele affects the status of the genes in the chromosome. Both dominant and recessive tags evolve through generations. Therefore, a recessive allele in the current generation may become a dominant allele in the next generation or vice versa. The Alleles have binary values of zero and one. If a dominant tag is one, the corresponding gene is hidden and if a dominant tag is zero, the corresponding gene is active. Choosing the dominant and recessive Alleles is based on the fitness value of the chromosome, i.e. the allele that results in better fitness value is chosen as the dominant one.

The selection operator is selected in this study to be one of the standard selection operators, e.g. the rank based operator that depends only on the fitness of the chromosome. The mutation operator is only applied on genes, while tags do not mutate. During the crossover operation, the single point crossover operator is applied on tags and genes separately and the tags crossover independently from the genes. The concept of this method is shown in **Figure** 7. The recessive tags in parent 1 crossover only with recessive tags of parent 2, and the dominant tags also crossover only with dominant tags. In this example, $g1_{p1}$ is the first gene in first parent; $g1_{p2}$ is the first gene in second parent, and so on. In the first child, the top allele set results in a more fit solution and therefore is dominant and the lower allele is recessive. In the second child, however, the lower allele results in a more fit solution and is considered the dominant allele.

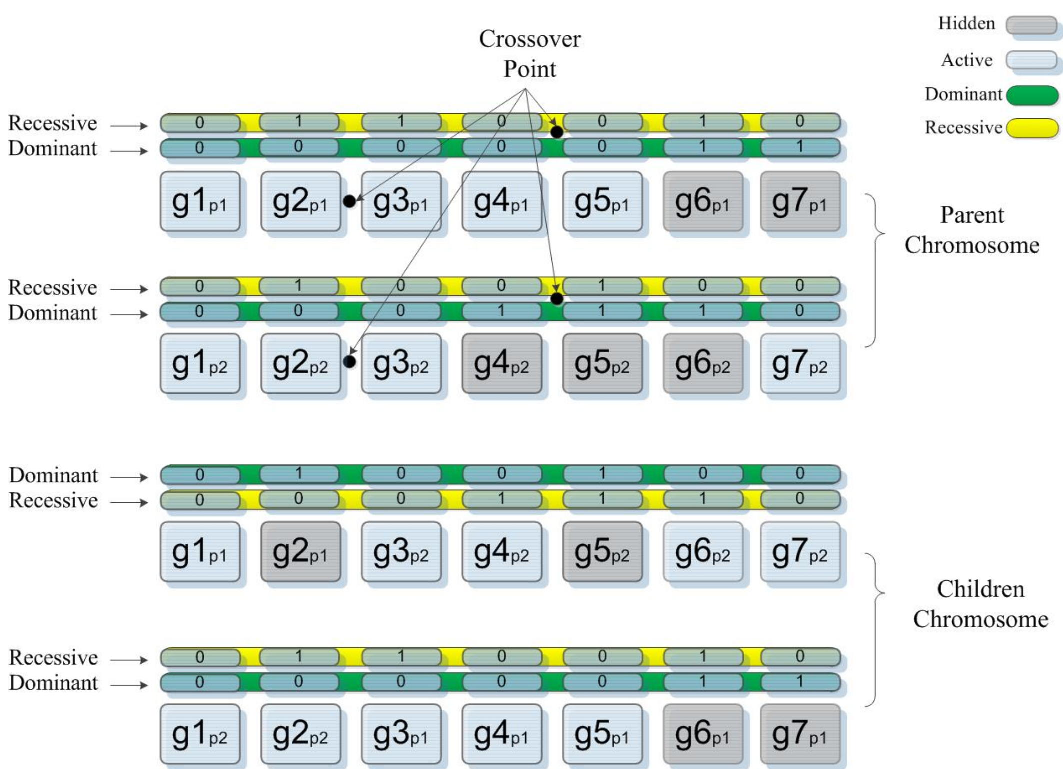


Fig. 7: The crossover operator in the Alleles concept of HGGA.

Comparison between the concept of Alleles and the four tag based mechanisms of **Section** II A is conducted through testing as detailed in **Section** III.

III. Interplanetary Trajectory Optimization Problems

Several examples of the complex trajectory optimization problems can be found in The Global Trajectory Optimization Competition online portal of the European Space Agency [45]. A typical problem statement can be written as follows: For a given range of the departure date from the home planet Earth, a given range of the arrival date to a target planet, and a given dry mass of the spacecraft, find the mission architecture (that is: how many flybys in the mission, and how many DSMs in each leg), as well as the dates and times of flybys, the flybys planets, the dates and times of DSMs, the amounts and directions of these DSMs, and the launch and arrival dates, such that the fuel mass needed for the whole mission is minimized. This is a VSDS optimization problem.

It is assumed in this study that the spacecraft operates with impulsive thrust and can have multiple DSMs in each leg. The objective function is to minimize the fuel consumption, which can be divided into departure (launch) impulse, arrival impulse, and DSMs maneuvers.

$$\Delta v_{tot} = ||\Delta V_d|| + ||\Delta V_a|| + \sum_{i=1}^{n} ||\Delta V_{DSM}|| \tag{6}$$

where $||\Delta V_d||$ is the launch impulse, $||\Delta V_a||$ is the arrival impulse, and $||\sum_{i=1}^{n} \Delta V_{DSM}||$ is the summation of costs of DSM maneuvers.

In the case of an n-impulse trajectory with no flybys (n DSMs in one mission leg), the independent design variables are assumed the departure and arrival time, the ΔV vector of first n impulses, and the epoch of the DSMs. Knowing the departure time, the planet heliocentric position vector can be determined (assumed equal to the heliocentric position vector of the spacecraft). Since the epoch of the first DSM and the initial velocity vector are known, the Kepler's equation can be used to propagate the position and velocity vector of the spacecraft at the DSM epoch. The velocity vector of the spacecraft after the DSM can be computed as the summation of the velocity vector of the spacecraft before the DSM and the DSM impulse vector. This procedure is repeated for all the transfer orbits of the trajectory except the last one, where the Lambert's problem is solved. Lambert's problem is a two-body boundary value problem that computes the trajectory using ini-

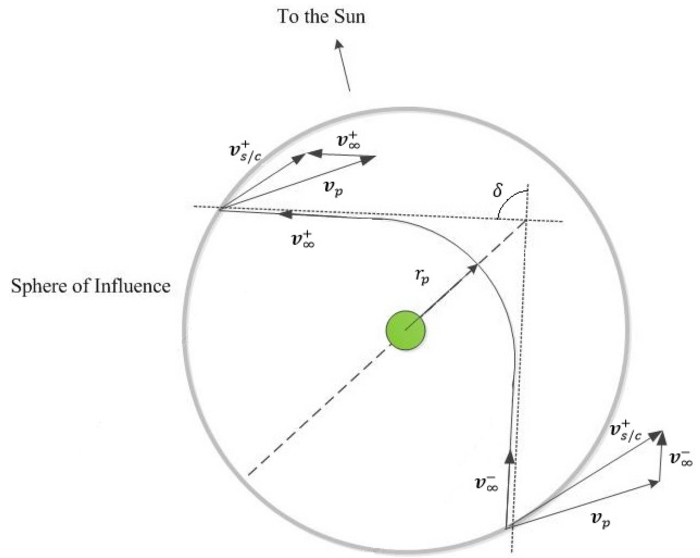


Fig. 8: Geometry of a non-powered flyby.

tial and final position vectors and TOF. For the last transfer orbit, the arrival time and hence the orbit's TOF are known. The planets position vector can be determined (equal to the spacecraft position vector at arrival) and therefore, the Lambert's problem can be used. This results in the arrival impulse for capture by the planet.

The spacecraft can have multiple powered or non-powered gravity-assist maneuvers (flybys). The momentum change in a flyby maneuver can impact the ΔV needed for the spacecraft during the mission. The spacecraft position vector during the flyby is assumed not to change and it is equal to the heliocentric position vector of the planet at the flyby instance.

$$\mathbf{r}^{-} = \mathbf{r}^{+} = \mathbf{r}_p \tag{7}$$

where $\mathbf{r}^{-}$ and $\mathbf{r}^{+}$ are the position vectors of the spacecraft before and after the flyby maneuver and $\mathbf{r}_p$ is the heliocentric position vector of the planet at the flyby instance. The velocity vector of the spacecraft after the flyby maneuver is determined by calculating the magnitude and direction of the velocity for powered and non-powered flybys as follows:

- Non-powered flyby: It is assumed that during the flyby, the linear momentum of the spacecraft changes only due to the gravity field of the planet. Hence, the magnitude of incoming and outgoing relative velocities are the same:

$$|\mathbf{v}_\infty^-| = |\mathbf{v}_\infty^+| = v_\infty \tag{8}$$

where $\mathbf{v}_\infty^-$ and $\mathbf{v}_\infty^+$ are the incoming and outgoing relative velocity vectors, respectively and are calculated as:

$$\mathbf{v}_\infty = \mathbf{v}_{S/C} - \mathbf{v}_p \tag{9}$$

$\mathbf{v}_{S/C}$ is the spacecraft velocity vector and $\mathbf{v}_p$ is the planet velocity vector (**Figure** 8). The direction of the outgoing velocity can be determined by the flyby plane rotation angle δ.

$$sin(\delta/2) = \frac{\mu_p}{\mu_p + r_{per} v_\infty^2} \tag{10}$$

where μ_p is the gravitational constant of the planet and r_{per} is the pericenter radius of the flyby which is a design variable. The maximum rotation angle is when the pericenter radius is minimum. If the required rotation angle is greater than the maximum achievable rotation angle, a powered flyby maneuver is needed. The total spacecraft velocity change in a non-powered flyby is then:

$$\Delta v_{npf} = 2 v_\infty sin(\delta/2) \tag{11}$$

- Powered flyby: Higher rotation angles can be gained by applying a small impulse during the flyby [46]. The spacecraft velocity on the periapsis trajectory is [47]:

$$v_m = \sqrt{v_\infty^2 + 2\mu_p / r_{per}} \tag{12}$$

Hence, the required change in velocity for powered flyby is:

$$\Delta v_{pf} = v_m^+ - v_m^- = \sqrt{{v_\infty^+}^2 + 2\mu_p/r_{per}} - \sqrt{{v_\infty^-}^2 + 2\mu_p/r_{per}} \tag{13}$$

The outgoing velocity of the spacecraft in heliocentric inertial frame can be calculated as follows [41]:

$$\mathbf{v}_\infty^+ = \mathbf{C}(\mathbf{v}_\infty^+)_L \tag{14}$$

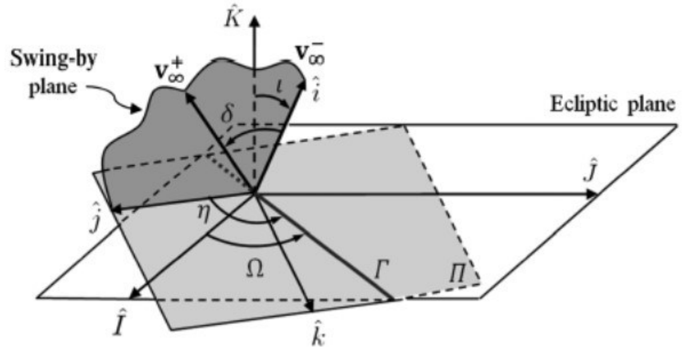


Fig. 9: The local and inertial frames [41].

where $(\mathbf{v}_{\infty}^{+})_L$ is the outgoing relative velocity vector expressed in the local frame $\hat{i}\hat{j}\hat{k}$ and $\mathbf{C} = [\hat{i} \quad \hat{j} \quad \hat{k}]$ is the transformation matrix between local frame and inertial frame. As shown in **Figure** 9, $(\mathbf{v}_{\infty}^{+})_L$ can be calculated as [41]:

$$(\mathbf{v}_{\infty}^{+})_L = v_{\infty}[cos(\delta) \quad sin(\delta) \quad 0]^T \tag{15}$$

The local frame is defined such that $\hat{i}$ is in the direction of the incoming relative velocity and $\hat{j}$ is perpendicular to $\hat{i}$ and is in the plane of the flyby maneuver. Line Γ in **Figure** 9 is the intersection of $\hat{j}\hat{k}$ plane (Π plane) and the inertial Ecliptic plane $\hat{I}\hat{J}$. The angle between $\hat{I}$ and Γ is Ω, and the angle between Γ and $\hat{j}$ is η. Also, ι is the inclination of plane Π to the Ecliptic plane. By this nomenclature, the unit directions can be derived as [41]:

$$\hat{i} = \frac{\mathbf{v}_{\infty}^{-}}{|\mathbf{v}_{\infty}^{-}|} \tag{16}$$

$$\hat{j} = \begin{bmatrix} cos(-\Omega) & sin(-\Omega) & 0 \\ -sin(-\Omega) & cos(-\Omega) & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(-\iota) & sin(-\iota) \\ 0 & -sin(-\iota) & cos(-\iota) \end{bmatrix} \times \begin{bmatrix} cos(-\eta) & sin(-\eta) & 0 \\ -sin(-\eta) & cos(-\eta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{17}$$

$$\hat{k} = \hat{i} \times \hat{j} \tag{18}$$

For the full MGADSM problem with m flybys and n_i DSMs in each leg ($i = 1 \ldots m$), the calculations for each leg is carried out as explained above. Departure and arrival dates and the TOF of each leg (except the last leg) are design variables. The TOF of the last leg can be calculated knowing the total TOF of the mission and the summation of the TOF of the other legs. Assume that there are n_1 DSMs in the first leg. Hence, there are $n_1 + 1$ transfer orbits in that leg. The calculations of the first n_l orbits are similar to the explanations on the n-impulse trajectory. For the last orbit, the velocity vector at the end point is the incoming heliocentric velocity of the flyby. The flyby is assumed non-powered if at least one DSM is in the following leg. Knowing the flyby pericenter altitude and rotation angle (design variables), the outgoing velocity (the spacecraft initial heliocentric velocity vector for the next leg) can be determined by carrying out the non-powered flyby calculations. This procedure is repeated for all the legs. In case of no DSMs in a leg, the initial flyby of that leg is assumed a powered flyby and the corresponding calculations can be used.

For all the problems, the J2 effect is ignored. Since Lambert's problem can have multiple solutions, the maximum number of revolutions is set to 5 and the best solution from Lambert's problem (lowest cost) is selected as the trajectory for the current leg.

To illustrate how this problem is a VSDS optimization problem, two sample solutions are shown as chromosomes in **Figure** 10. In this example, the hidden genes are shown with gray color. The top part of the figure shows the chromosomes in HGGA, with hidden genes and equal lengths, and the bottom part of the figure shows the equivalent chromosomes with no hidden genes and different lengths. As seen, depending on the number of flybys and DSMs, the length of the solutions can be variable. In the first solution, there is one flyby and one DSM, and in the second solution there are two flybys and two DSMs. Assume that it is required to send a spacecraft to planet Jupiter with the lowest cost (fuel consumption) within certain ranges for launch and arrival dates. The two solutions shown in **Figure** 10 can be interpreted as follows:

1. First Solution: A trajectory with one flyby around Venus (Earth-Venus-Jupiter) and one DSM in the second leg.

2. Second Solution: A trajectory with two flybys around Venus and Earth (Earth-Venus-Earth-Jupiter or EVEJ) and two DSMs in the first and the third legs.

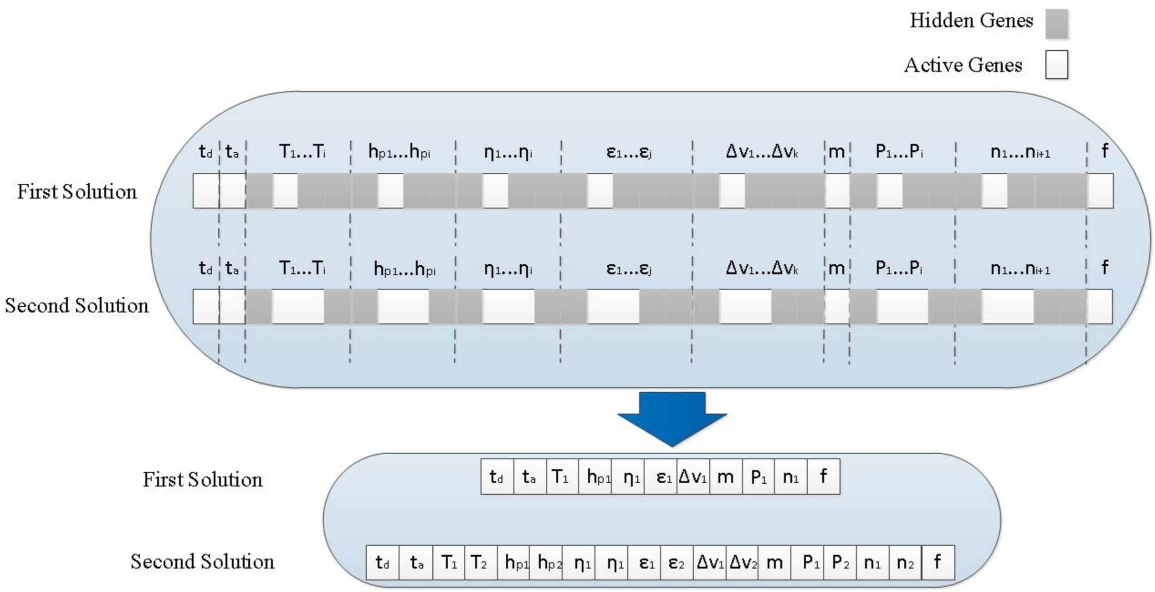


Fig. 10: An example of two different solutions for an interplanetary trajectory problem in HGGA (Earth to Jupiter), and the equivalent chromosomes with no hidden genes in GA.

This is a VSDS problem; the proposed tags/Alleles mechanisms can be used to search for the optimal solution and architecture.

IV. Numerical Results

In this paper, two benchmark problems are investigated: Earth to Jupiter and Earth to Saturn. The best known solutions for these problems can be found in the European Space Agency (ESA) website [48] and also in [41, 49]. Each problem is solved in two phases. In the first phase, it is assumed that there are no DSMs (zero-DSM phase) and a sub-optimal flyby sequence is obtained. The second phase is a multi-gravity-assist with DSMs phase (MGADSM phase) that uses a fixed flyby sequence (obtained in the first step) to optimize the rest of the design variables including the DSMs in the mission. This approach has shown to be computationally efficient [41] compared to a single model where all the variables including DSMs and flyby sequences are optimized together. For all the problems, the genes mutation probability is 0.01, the elite count is 10% of the population size, the crossover probability is 0.95, and the function tolerance (stopping condition of the algorithm) is 10^{-12}. For the sake of comparison, the lower and upper boundaries of the variables in all problems are compatible with the work done in [41, 48, 49].

A. Earth - Jupiter Mission Trajectory Optimization

The variable boundaries are listed in **Table 2**. The spacecraft can flyby around up to two planets in the solar system and there can be up to two DSMs in each leg. Hence, the chromosome has two genes for the flyby planets in the zero-DSM phase, and each flyby planet can be any one from one (Mercury) to eight (Neptune). Each flyby gene carries the planet identification number. One tag (two tags in the case of using the Alleles concept) is assigned to each flyby gene and if the tag of any of the flybys is one, the corresponding flyby is hidden. For example assume that the values of flybys are three (first flyby is around the third planet-Earth) and five (second flyby is around the fifth planet-Jupiter). If the tags are $[1, 0]$, the flyby around Earth is hidden and the solution has only one flyby around Jupiter. Similarly, for the MGADSM phase, there can be a maximum two DSMs in each leg. Since the maximum number of flybys is two, the maximum number of legs is three, and hence, the maximum number of DSMs is six. For each DSM, we need to compute the optimal time (T_{DSM}) at which this DSM occurs. A gene and a tag are added for each DSM time T_{DSM}, and hence, there are six gens and six tags for T_{DSMi} ($i = 1 \cdots 6$) in this mission. Note that if a flyby is hidden, then its leg disappears and all the DSMs in that leg automatically become hidden. Note also that even if a flyby exists, a DSM in its leg can be hidden depending on the value of its own tag. The range for each DSM is set between $[-5, -5, -5]$ km/s and $[5, 5, 5]$ km/s as shown in **Table 2**. So, the chromosome will have genes for $6 \times 3 = 18$ scalar components of the DSMs. Note that these 18 genes are classified in groups of three genes; hence if one DSM is hidden then its three genes get hidden together. The TOF for each leg is between 80 and 800 days except the last one. The duration of the last leg is determined by the launch and arrival dates and the TOF of the other legs. There is a gene for each TOF in the mission. Hence, we have three genes for the TOFs in this Jupiter mission. Note that there are no tags associated with the TOF genes since the state of each gene (hidden or active) is determined based on the flyby tags. If a flyby exists then there is an active gene for a TOF associated with it. Two genes for the two flyby altitudes h_p and two genes for the two flyby plane angles η are added. Similar to the TOF variables, no tags are needed for the h_p and η genes. There are also six genes for the departure impulse, flight direction, the arrival date and the departure date.

Table 2: Lower and upper bounds of Earth-Jupiter problem

Design Variable	Lower Bound	Upper Bound
Flyby 1 planet	$1\,(Mercury)$	$8\,(Neptune)$
Flyby 2 planet	1	8
DSM_i (km/s), $i = 1 \cdots 6$	$[-5, -5, -5]$	$[5, 5, 5]$
Flight Direction	Posigrade	Retrograde
Departure Date (t_0)	01 Sep.2016	30 Sep.2016
Arrival Date (t_f)	01 Sep.2021	31 Dec.2021
TOF (days)	$[80, 80]$	$[800, 800]$
Flyby normalized pericenter altitude (h_p)	$[0.1, 0.1, 0.1]$	$[10, 10, 10]$
Flyby plane rotation angle (η) (rad)	$[0, 0, 0]$	$[2\pi, 2\pi, 2\pi]$
Epoch of DSMs ($\epsilon_i, i = 1 \cdots 6$)	0.1	0.9

The population size is set to 500 and the number of generations is 500. Each simulation is repeated 100 times for the purpose of statistical analysis on the efficiency of the method. This problem is solved using different mechanisms and the cost values of each mechanism in the zero-DSM and the MGADSM models are reported in **Table** 3. All the mechanisms could find the optimal flyby sequence which is Earth-Venus-Earth-Jupiter (EVEJ) in their zero-DSM model. Based on the results of **Table** 3, the Stochastic Mechanism can find the lowest cost solution (10.1308 km/s) with one DSM in the first leg, while the Logical Mechanism results in the the highest cost. The detailed solution of the Stochastic Mechanism is presented in **Table** 4 and **Figure** 11.

Table 3: Cost values of Earth-Jupiter problem using different Mechanisms

Mechanism	Zero-DSM model (km/s)	MGADSM model (km/s)
Stochastic Mechanism	10.1612	10.1308
Logical Mechanism	11.0580	10.9822
Short Mechanism	11.2590	10.4483
Long Mechanism	13.2707	10.5075
Alleles	10.2374	10.1741

Table 4: Solution of Earth-Jupiter problem using Stochastic Mechanism

Mission parameter	Zero-DSM model	MGADSM model
Departure Date	$05-Sep-2016, 17:20:00$	$03-Sep-2016, 15:08:25$
Departure Impulse (km/s)	3.5233	3.4488
DSM date	–	$14-Jan-2017, 06:36:34$
DSM impulse (km/s)	–	$\|[0.0225, 0.0354, -0.0147]\| = 0.0444$
Venus flyby date	$05-Sep-2017, 14:13:06$	$06-Sep-2017, 17:54:14$
Post-flyby impulse (km/s)	1.5476×10^{-7}	2.3533×10^{-5}
Pericenter altitude (km)	1290.1954	876.0221
Earth flyby date	$29-Mar-2019, 22:43:33$	$29-Mar-2019, 04:20:01$
Post-flyby impulse (km/s)	0.4402	0.4478
Pericenter altitude (km)	637.8000	637.8000
Arrival date	$21-Sep-2021, 16:22:30$	$18-Sep-2021, 03:33:26$
Arrival impulse (km/s)	6.1961	6.1897
TOF (days)	$364.8702, 570.3545, 906.7354$	$368.1152, 568.4346, 903.9676$
Mission duration (days)	1840.5174	1844.499
Motion direction	posigrade	posigrade
Mission cost (km/s)	10.1612	10.1308

B. Earth-Saturn Mission Trajectory Optimization (Cassini 2)

A more complicated trajectory is the Cassini 2 mission that was designed by NASA, European Space Agency, and Italian Space Agency to discover the planet Saturn. The mission consists of a satellite that orbits Saturn and a lander for its moon Titan [50]. We consider the problem of designing the trajectory from Earth to rendez-vous with Saturn. The high number of potential flybys and the wide ranges for the design variable make this problem challenging. Here, a launch window of 30 days is selected for the mission for the sake of comparison with the reported results in the literature [41, 51]. The upper and lower boundaries of the design variables are shown in **Table** 5.

The goal is to optimize the trajectory to Saturn as a VSDS problem with unknown number of flybys and DSMs. The maximum number of flybys is set to four (with four corresponding tags

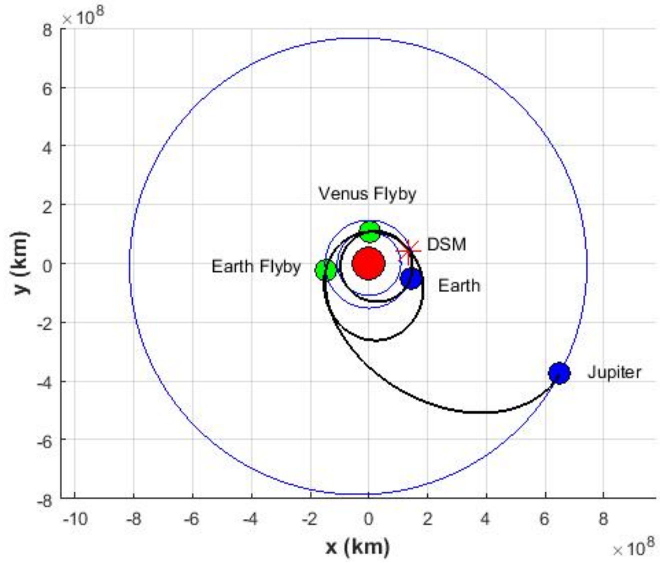
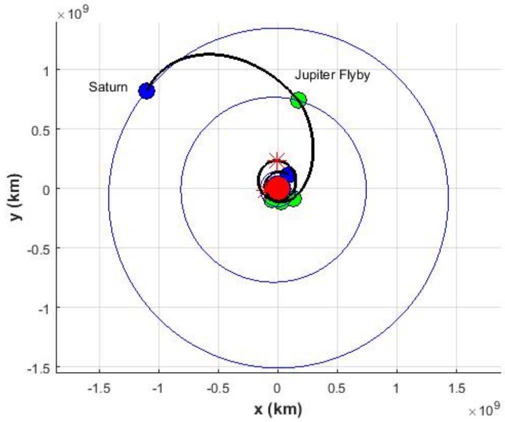


Fig. 11: EVEJ Trajectory for MGADSM Model using Stochastic Mechanism

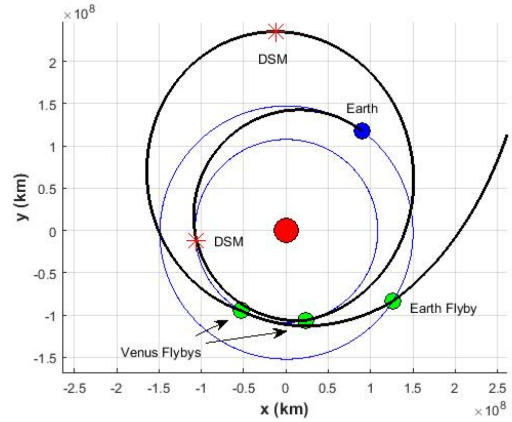
Table 5: Lower and upper bounds of Earth-Saturn problem

Design Variable	Lower Bound	Upper Bound
Flyby $\#i$ planet, $i = 1 \cdots 4$	$2\,(Venus)$	$5\,(Jupiter)$
$DSM_i(\text{km/s}), i = 1 \cdots 5$	$[-5, -5, -5]$	$[5, 5, 5]$
Flight Direction	Posigrade	Retrograde
Departure Date	01 Nov.1997	01 Dec.1997
Arrival Date	01 Jan.2007	30 Jun.2007
TOF (days)	$[100, 100, 30, 400]$	$[400, 500, 300, 1600]$
Flyby normalized pericenter altitude	$[0.05, 0.05, 0.15, 0.7]$	$[5, 5, 5.5, 290]$
Flyby plane rotation angle (rad)	$[-\pi, -\pi, -\pi, -\pi]$	$[\pi, \pi, \pi, \pi]$
Epoch of DSM ($\epsilon_i, i = 1 \cdots 5$)	0.01	0.9

in the zero-DSM phase) and the maximum number of DSMs in each of the five legs is one (with five corresponding tags in the MGADSM phase). For both phases, the population size is 500. The number of generations is selected to be 400 for the zero-DSM phase and 500 for the MGADSM phase. A niching method is used to help the optimization algorithm explore more of the design space [29, 41, 52]. In this niching method, every 20 generations, the current best solutions and other solutions with similar flyby sequences are given high cost. Moreover, every five generations



(a) EVVEJS trajectory



(b) Inner planets

Fig. 12: EVVEJS Trajectory for MGADSM Model using Stochastic Mechanism

a random solution is inserted in place of an elite solution. In doing so, the stochastic, logical, and Alleles mechanisms were able to find the optimal flyby sequence. Out of ten identical simulations, the stochastic mechanism can find the optimal sequence seven times (success rate of 70%), the logical mechanism two times (success rate of 20%), and the Alleles method five times (success rate of 50%). The short and long mechanisms were not able to find the optimal flyby sequence. For the MGADSM phase, only the mechanisms that were able to find the optimal sequence are investigated. The results are summarized in **Table** 6. The Stochastic Mechanism has the lowest cost of 8.4457 km/s with one DSM in the first and second leg (**Table** 7). The Logical Mechanism and the Alleles concept found solutions with higher cost values of 9.0539 km/s and 10.1364 km/s, respectively. The Stochastic Mechanism trajectory is shown in **Figure** 12.

Table 6: Results of Earth-Saturn problem using different Mechanisms

Mechanism	success rate of Zero-DSM model	cost of MGADSM model (km/s)
Stochastic Mechanism	70%	8.4457
Logical Mechanism	10%	9.0539
Short Mechanism	0%	–
Long Mechanism	0%	–
Alleles	50%	10.1364

Table 7: Solution of Earth-Saturn problem using stochastic mechanism

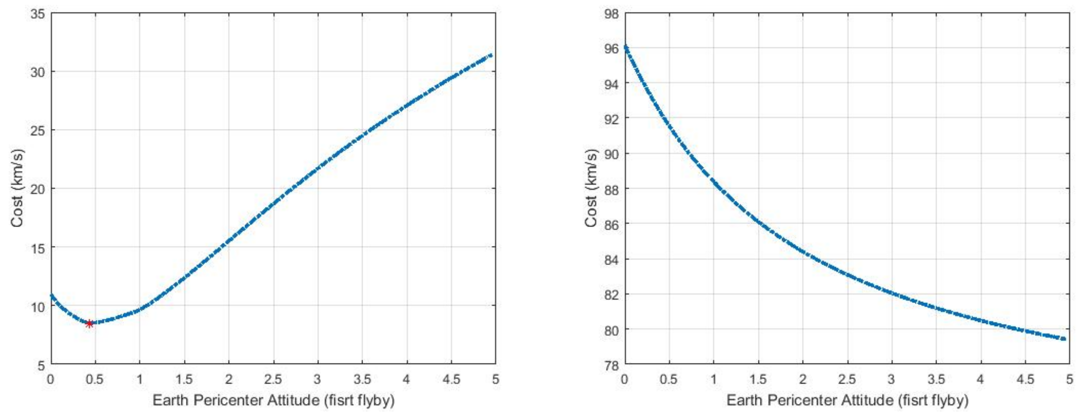
Mission parameter	Zero-DSM model	MGADSM model
Departure Date	$15 - Nov - 1997, 08:53:42$	$14 - Nov - 1997, 11:01:28$
Departure Impulse (km/s)	3.2676	3.2782
DSM date	–	$08 - Mar - 1998, 19:51:44$
DSM impulse (km/s)	–	$\|[0.16194 - 0.43175 - 0.21757]\| = 0.50987$
Venus flyby date	$02 - May - 1998, 08:45:56$	$29 - Apr - 1998, 02:02:05$
Post-flyby impulse (km/s)	1.8240	0
Pericenter altitude (km)	22685.3828	2066.8258
DSM date	–	$25 - Nov - 1998, 05:56:38$
DSM impulse (km/s)	–	$\|[0.39217 - 0.0014362 - 0.10418]\| = 0.40577$
Venus flyby date	$27 - Jun - 1999, 09:46:18$	$26 - Jun - 1999, 11:10:42$
Post-flyby impulse (km/s)	1.8873	1.19153×10^{-7}
Pericenter altitude (km)	12518.5172	605.2880
Earth flyby date	$19 - Aug - 1999, 15:52:32$	$19 - Aug - 1999, 16:16:48$
Post-flyby impulse (km/s)	$6.7970e - 07$	0.00031764
Pericenter altitude (km)	1966.2076	1855.5100
Jupiter flyby date	$31 - Mar - 2001, 08:45:42$	$29 - Mar - 2001, 09:31:40$
Post-flyby impulse (km/s)	$1.7135e - 05$	0.00022016
Pericenter altitude (km)	4920495.3477	4975803.05136
Arrival date	$13 - May - 2005, 08:21:04$	$22 - Mar - 2007, 08:14:11$
Arrival impulse (km/s)	4.2469	4.2513
TOF (days)	$167.99, 421.04, 53.25, 589.70, 2199.35$	$165.81, 423.41, 53.78, 588.35, 2183.54$
Mission duration (days)	3431.3442	3414.8886
Mission cost (km/s)	11.2259	8.4457

V. Discussion

The GTOP database consists of a wide variety of problems to asteroids and different planets, including Saturn and Mercury. The HGGA mechanisms in this paper are tested on Cassini 2 and Earth to Jupiter problems. The results presented in this paper show that the proposed mechanisms are capable of finding the optimal architecture of the mission (optimal flyby sequence as well as optimal number of DSMs).

The trajectory from Earth to Jupiter has been previously investigated using different evolutionary methods. For the sake of comparison, only the methods that assume an impulsive thrust are considered here. Olympio and Marmorat solved this problem using the primer vector theory [51]. By assuming a fixed flyby sequence as EVEJ and setting the variable ranges close to **Table** 2, a total cost of 10.267 km/s was found. The HGGA method with feasibility criteria (original HGGA) was also tested on this problem and found a solution of cost 10.178 km/s [41]. The Dynamic-size multiple population Genetic algorithm has also been tested on this problem and the cost of its solution is 10.125 km/s [49]. If the duration of the mission increases, the total cost would decrease. This has been shown in the works done by Musegaas [53] and Myatt et. al. [30]. Musegaas solved the Earth to Jupiter problem as a tuning step for a mission to Saturn (EVEJS). A fixed flyby sequence and large mission duration (almost 20 years and eight months) are assumed in solving the problem. The spacecraft can have powered flybys and is captured at Jupiter. No DSMs are assumed during the trajectory and by optimizing only the event times, the cost found is 7.0144 km/s. Myatt et al. solved the same problem assuming non-powered flybys and found a solution with a cost of 7.5483 km/s. The total time of the mission in here is not allowed to exceed five years and hence higher cost values are found. The cost found by Stochastic Mechanism is 10.1308 km/s which is slightly better than the solution found by the original version of the HGGA.

For the mission to Saturn, initial investigations show that the cost function is sensitive to the events dates (dates of performing DSMs and flybys). As an example, consider the variation of the cost function with the first flyby pericenter altitude h_p. **Figure** 13a shows the variation of the cost function with the pericenter altitude when all other variables are fixed at their optimal values; clearly the optimal solution corresponds to the red star in this figure. **Figure** 13b on the other hand shows the variation of the cost function with the pericenter altitude when the launch date is varied to a value different from its optimal value, while still keeping all other variables at their optimal values. Two observations can be noted from **Figure** 13. First, the impact of changing the launch date is significant on the cost; this can be depicted by comparing the cost values between the two



(a) All variables optimal

(b) All variables optimal except t_0

Fig. 13: Cost value vs. pericenter altitude of first flyby

figures (the vertical axis) with 50 days difference in their launch dates. Second, when the launch date is not optimal (**Figure** 13b) the line relating the cost to h_p is misleading to the optimizer. When the launch date is optimal, the cost decreases with decreasing h_p, while that is not the case when the launch date is not optimized. Hence, when optimizing the MGADSM phase, a small range is assumed around the zero-DSM variables.

The mission to Saturn has been investigated in many papers in different formats. EVEJS, Casini 1, Cassini 2 (easy and complete versions) are some of the variations on the mission that have been investigated. For the Cassini 2 (easy version), the minimum cost reported in the literature is 8.385 km/s when solving the VSDS problem [41] and it is 8.282 km/s when solving the problem assuming a known fixed flyby sequence [54, 55]. This problem is also solved using the PaGMO software using differential evolution and genetic algorithms [53]. PaGMO is an optimization software in which the user can define the problem and the optimization algorithm. The lowest cost found in this reference is 8.2379 km/s given a known fixed flyby sequence and one DSM. Other references have reported close cost values for this problem with a known fixed flyby sequence [56–58]. A list of these solutions can be found in the GTOP website [48]. In this study, only three mechanisms (the Stochastic, the logical, and the Alleles) were able to find the optimal flyby sequence. The Stochastic Mechanism found a cost of 8.4457 km/s for this mission, with one DSM in the first leg and one DSM in the second leg. Despite that this cost is slightly higher than the best known solution, the main advantage of the proposed method is its capability of searching for the optimal flyby and DSM architecture.

In all the tested problems, the Stochastic Mechanism found the lowest cost compared to the other mechanisms investigated in this paper. In the next section some statistical analysis is done on the mechanisms and their performance is compared.

VI. Conclusions

This paper demonstrated that the hidden genes genetic algorithm - with new evolution mechanisms for tags - has the capability of searching for the optimal architecture and solution in space trajectory optimization problems. The concepts of tags and Alleles in hidden genes are introduced in this paper, and different evolution mechanisms for the tags are investigated and compared based on their performance. These mechanisms found different solutions of different cost values and different success rates. In all the test cases, the stochastic mechanism could find the best flyby and DSM sequence, as well as the lowest cost value compared to other mechanisms.

Acknowledgments

This work was funded by National Science Foundation, award # 1446622.
Superior, a high performance computing cluster at Michigan Technological University, was used in obtaining the results presented in this publication.

[1] Hohmann, W., *Erreichbarkeit der himmelskorper*, R. Oldenbourg, Munich and Berlin, 1925.

[2] Cornelisse, J. W., "Trajectory Analysis for Interplanetary Missions," *ESA Journal*, Vol. 2, 1978, pp. 131–143.

[3] Navagh, J., "Optimizing Interplanetary Trajectories with Deep Space Maneuvers," Tech. rep., NASA Langley Research Center, No. CR4546, 1993.

[4] Molenaar, S., *Optimization of interplanetary trajectories with deepspace maneuvers - Model development and application to a Uranus orbiter mission*, Master's thesis, Delft University of Technology, the Netherlands, 2009.

[5] Bryan, J. M., *Global Optimization of MGA-DSM Problems Using the Interplanetary Gravity Assist Trajectory Optimizer (IGATO)*, Master's thesis, California Polytechnic State University, San Luis Obispo, 2011.

[6] Lizia, P. D. and Radice, G., "Advanced global optimisation for mission analysis and Design," 18139, European Space Agency, Ariadna Final Report, 2004.

[7] Vasile, M. and de Pascale, P., "Preliminary design of multiple gravity-assist trajectories," *Journal of Spacecraft and Rockets*, Vol. 43, No. 4, 2006, pp. 794–805.

[8] Chilan, C. M. and Conway, B. A., "Automated Design of Multiphase Space Missions Using Hybrid Optimal Control," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 5, 2013, pp. 1410–1424.

[9] Vasilea, M., Summerera, L., and Pascale, P. D., "Design of Earth - Mars transfer trajectories using evolutionary-branching technique," *Acta Astronautica*, Vol. 56, 2005, pp. 705 – 720.

[10] Dachwald, B., "Optimisation of solar sail interplanetary trajectories using evolutionary neurocontrol," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 1, 2004, pp. 66–72.

[11] Sentinella, M. R. and Casalino, L., "Cooperative evolutionary algorithm for space trajectory optimization," *Celestial Mechanics and Dynamical Astronomy, Springer*, Vol. 105, No. 1, 2009, pp. 105–211.

[12] Labroquere, J., Heritier, A., Riccardi, A., and Izzo, D., "Evolutionary Constrained Optimization for a Jupiter Capture," *Parallel ProblemSolving from Nature, PPSN XIII, Lecture Notes in Computer Science*, Vol. 8672, 2014.

[13] Gondelach, D. J. and Noomen, R., "Hodographic-Shaping Method for Low-Thrust Interplanetary Trajectory Design," *AIAA Journal of Spacecraft and Rockets*, Vol. 52, No. 3, 2015, pp. 728–738.

[14] Pascale, P. D. 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.

[15] Olympio, J. T. and Yam, C. H., "Deterministic Method for Space Trajectory Design with Mission Margin Constraints," *61st International Astronautical Congress*, Prague, CZ, 2010.

[16] Liu, J. and Dai, G., "The Application of Evolution-Branching Algorithm on Earth-Mars Transfer Trajectory," *Computational Intelligence and Intelligent Systems, ISICA*, Vol. 51, Springer, Berlin, Heidelberg, 2009.

[17] Simoes, L. F., Izzo, D., Haasdijk, E., and Eiben, A. E., "Multi-rendezvous Spacecraft Trajectory Optimization with Beam P-ACO," *EvoCOP 2017*, Amsterdam, Netherlands, 2017.

[18] Hughes, G. and McInnes, C. R., "Solar Sail Hybrid Trajectory Optimisation," *Advances in the Astronautical Sciences*, Vol. 109, 2001.

[19] Abdelkhalik, O. and Mortari, D., "Orbit design for ground surveillance using genetic algorithms," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 5, 2006, pp. 1231–1235.

[20] Rauwolf, G. A. and Coverstone-Carroll, V. L., "Near-optimal low-thrust orbit transfers generated by a genetic algorithm," *Journal of Spacecraft and Rockets*, Vol. 33, No. 6, 1996.

[21] Wagner, S., *Automated trajectory design for impulsive and low thrust interplanetary mission analysis*, Ph.D. thesis, Iowa State University, Ames, Iowa, 2014.

[22] Izzo, D., Hennes, D., and Riccardi, A., "Constraint Handling and Multi-Objective Methods for the Evolution of Interplanetary Trajectories," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 4, 2015, pp. 792–799.

[23] Izzo, D., "Advances in Global Optimisation for Space Trajectory Design," *Proceedings of the International Symposium on Space Technologyand Science*, Vol. 25, 2006.

[24] Shen, H., Zhou, J., Peng, Q., Li, H., and Li, J., "Multi-objective interplanetary trajectory optimization combining low-thrust propulsion and gravity-assist maneuvers," *Science China: Technological Sciences*, Vol. 55, No. 3, 2012, pp. 841–847.

[25] Izzo, D., Simoes, L. F., Martens, M., de Croon, G., Heritier, A., and Yam, C. H., "Search for a grand tour of the Jupiter Galilean moons," *Proceedings of the 15th annual conference on Genetic and evolutionary computation, ACM*, Vol. 15, 2013.

[26] Ceriotti, M., *Global optimisation of multiple gravity assist trajectories*, Ph.D. thesis, University of Glasgow, Glasgow, Scotland, United Kingdom, 2010.

[27] Goldberg, D. E., *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley Longman Publishing Co., 1989.

[28] Strange, N. J. and Longuski, J., "Graphical method for gravity-assist trajectory design," *Journal of Spacecraft and Rockets*, Vol. 39, No. 1, 2002, pp. 9–16.

[29] Izzo, D., Becerra, V. M., Myatt, D. R., Nasuto, S. J., and Bishop, J. M., "Search space pruning and global optimisation of multiple gravity assist spacecraft trajectories," *Journal of Global Optimization*, Vol. 38, 2007, pp. 283–296.

[30] Myatt, D. R., Becerra, V. M., Nasuto, S. J., and Bishop, J. M., "Advanced global optimisation for mission analysis and design," *European Space Agency, Ariadna Final Report*, , No. 03–4101a, 2004.

[31] Englander, J. A., Conway, B. A., and Williams, T., "Automated Mission Planning via Evolutionary Algorithms," *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 6, 2012, pp. 1878–1887.

[32] Bernelli-Zazzera, F., Lavagna, M., Armellin, R., Lizia, P. D., Toppu, F., and Berz, M., "Global Trajectory Optimisation: Can We Prune the Solution Space when Considering Deep Space Maneuvers?"

Tech. Rep. 064101, European Space Agency, Ariadna Final Report, 2007.

[33] Strange, N. and Longusk, J., "Graphical Method for Gravity-Assist Trajectory Design," *Journal of Spacecraft and Rockets*, Vol. 39, No. 1, 2002, pp. 9–16.

[34] Labunsky, A. V., Papkov, O. V., and Sukhanov, K. G., *Multiple gravity assist interplanetary trajectories*, Gordon and Breach Science Publishers, Newark, NJ, 1998.

[35] Forsyth, R., "BEAGLEâĂŤA Darwinian Approach to Pattern Recognition," *Kybernetes*, Vol. 10, No. 3, 1981, pp. 159–166.

[36] Smith, J. E. and Fogarty, T. C., "Adaptively parameterised evolutionary systems: Self adaptive recombination and mutation in a genetic algorithm," *International Conference on Evolutionary Computation – The 4th International Conference on Parallel Problem Solving from Nature*, Berlin, Germany, Sept. 1996.

[37] Smith, J. E., Bartley, M., and Fogarty, T. C., "Microprocessor design verification by two-phase evolution of variable length tests," *IEEE International Conference on Evolutionary Computation*, Indianapolis, Indiana, USA, 1997, pp. 184–189.

[38] Jenuwein, T. and Allis, C. D., "Translating the histone code," *Journal of Science*, Vol. 293, 2001.

[39] Tanev, I. and Yuta, K., "Epigenetic programming: Genetic programming incorporating epigenetic learning through modification of histones," *Information Sciences*, Vol. 178, 2008.

[40] Abdelkhalik, O., "Hidden Genes Genetic Optimization for Variable-Size Design Space Problems," *Springer, Journal of Optimization Theory and Applications*, Vol. 156, No. 2, 2012, pp. 450–468.

[41] Gad, A. and Abdelkhalik, O., "Hidden Genes Genetic Algorithm for Multi-Gravity-Assist Trajectories Optimization," *AIAA Journal of Spacecraft And Rockets*, Vol. 48, No. 4, 2011, pp. 629–641.

[42] Abdelkhalik, O. and Darani, S. A., "Hidden Genes Genetic Algorithms for Systems Architecture Optimization," *In Proceedings of the 2016 Genetic and Evolutionary Computation Conference*, Denver, Colorado, 2016, pp. 629–636.

[43] Hasan, H., *Mendel and the Laws of Genetics*, The Rosen Publishing Group, 2005.

[44] Bowler, P., *Evolution: the history of an idea*, Berkeley, University of California Press, 2003.

[45] European Space Agency, Mission Analysis and Design, Advanced Concepts Team, "Global Trajectory Optimisation Competition," https://sophia.estec.esa.int/gtoc_portal/.

[46] Olympio, J., "Designing Optimal Multi-Gravity-Assist Trajectories with Free Number of Impulses," *21st International Symposium on Space Flights Dynamics*, 2009.

[47] Takuto, I., Hoffman, J., and de Weck, O., "Method for Rapid Interplanetary Trajectory Analysis using V Maps with Flyby Options," *British Interplanetary Society*, Vol. 64, 2011, pp. 204–213.

[48] European Space Agency, A. C. T., "Cassini 2 Mission," http://www.esa.int/gsp/ACT/inf/projects/gtop/cassini2.html, Accessed: April 2017.

[49] Abdelkhalik, O. and Gad, A., "Dynamic-Size Multiple Populations Genetic Algorithm for Multigravity-Assist Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 3, 2012, pp. 520–529.

[50] Matson, D., Spilker, L., and Lebreton, J., "The Cassini/Huygens Mission to the Saturnian System," *Space Science Reviews*, Vol. 104, 2002, pp. 1–58.

[51] Olympio, J. and Marmorat, J. P., "Global Trajectory Optimization: Can We Prune the Solution Space When Considering Deep Space Maneuvers?" Rept. ao/1-5180/06/4101, European Space Agency, 2007.

[52] DiLizia, P. and Radice, G., "Advanced Global Optimisation Tools for Mission Analysis and Design," Rept. 18139/04/nl/mv, European Space Agency, 2004.

[53] Musegaas, P., *Optimization of Space Trajectories Including Multiple Gravity Assists and Deep Space Maneuvers*, Master's thesis, Delft University of Technology, the Netherlands, 2012.

[54] Schlueter, M., *Nonlinear mixed integer based Optimization Technique for Space Applications*, Master's thesis, The University of Birmingham, Birmingham, United Kingdom, 2012.

[55] Vasile, M., Minisci, E., and Locatelli, M., "An Inflationary Differential Evolution Algorithm for Space Trajectory Optimization," *IEEE Transactions on Evolutionary Computation*, Vol. 15, 2011, pp. 267–281.

[56] Addis, B., Cassioli, A., Locatelli, M., and Schoen, F., "A global optimization method for the design of space trajectories," *Computational Optimization and Applications*, Vol. 48, 2011, pp. 635–652.

[57] Danoy, G., Dorronsoro, B., and Bouvry, P., "New State-of-the-art Results for Cassini2 Global Trajectory Optimization Problem," *Acta Futura*, Vol. 5, 2012, pp. 5–72.

[58] Vinko, T., Izzo, D., and Bombardelli, C., "Benchmarking different global optimisation techniques for preliminary space trajectory design," *58th International Astronautical Congress*, Hyderabad, 2007.