M.Eng. Mechanical Engineering
MM4MPR Individual Research Paper 2019/20

Minimisation of Delta-V for Interplanetary Space Missions Through Creation of Automated Trajectory Design Tools

R. A. Watson, D. B. Hann
Faculty of Engineering,
University of Nottingham, UK

ABSTRACT

The minimisation of fuel requirements for an interplanetary space mission can have significant impact on mission feasibility. This project aims to provide academia and early phase mission designers a set of software tools that can be used in assessing the feasibility of potential trajectory concepts. Implementation will be through MATLAB, with the main parameter of focus being ΔV – the maximum change in velocity of a rocket. Through programming, a kinematic model, taking advantage of the patched conic approximation to simply trajectory calculation, is developed. ΔV required for interplanetary missions is computed as a function of time and an enumerative approach is taken to determine optimal times of departure and arrival at respective bodies. The required ΔV to patch together incoming and outgoing trajectories, during a gravity-assist flyby, is also programmed as a function of time. These software tools are then verified against test cases involving real mission data, such as NASA's upcoming mission to Mars and Voyager 2's flyby of Jupiter. Results showed that the software tools created provide sufficient accuracy to determine trajectory concepts and assess their feasibility as per the project objectives. These trajectories can then be passed on to more sophisticated, dynamic, multibody simulations with higher accuracy.

NOMENCLATURE

$\Delta V - Maximum\ Change\ in\ Velocity\ of\ the\ Rocket$
$V_{\odot} - Heliocentric\ Velocity$
$F - Gravitational\ Force\ Between\ Two\ Objects$
$G - Gravitational\ Constant$
$m_1 - Mass\ of\ Body\ 1$
$m_2 - Mass\ of\ Body\ 2$
$r - Distance\ Between\ Two\ Body's\ Centres\ of\ Mass$
$V_{hyp} - Hyperbolic\ Excess\ Velocity$
$V_p - Heliocentric\ Planet\ Velocity$
$\Delta - Aiming\ Radius$
$V_{SC} - Heliocentric\ Velocity\ of\ Spacecraft$
$t_1 - Time\ of\ Departure$
$t_2 - Time\ of\ Arrival$

1. INTRODUCTION

Innovation in the commercial space industry has led to the cost per kilogram of space mission launches to once again decrease, after a long plateau [1]. Despite the decreasing cost, spaceflight, for many, remains an expensive endeavour. As such any reduction launch mass is welcomed. The maximum change in velocity of a rocket through burning it's fuel is known as ΔV. Tsiolkovsky's ideal rocket equation [2] describes the exponential relationship between fuel fraction and ΔV, fuel fraction being the mass of the fuel relative to the gross mass of the rocket. Therefore, a reduction in ΔV requirement results in a reduced fuel requirement. With fuel accounting for most of a spacecraft's launch mass - upwards of 90% in some cases [3] - reducing fuel requirement is key to reducing launch mass and therefore mission cost.

This project aims to provide academia, and early phase mission designers, automated design tools for trajectory determination, achieved through optimisation of mission characteristics to minimise ΔV. These tools can be used and adapted for a range of scenarios, allowing mission designers to quickly assess the feasibility of preliminary trajectory concepts. Implementation will be through MATLAB.

1.1 Gravity Assist Flybys

Exploration of the outer solar system was made possible by the work of Michael Minovitch [4]. His discovery of gravity-assist trajectories allowed spacecraft to travel faster and further than previously thought possible. Such trajectories utilise the relative motion and orbital energy of solar system bodies to 'slingshot' spacecraft, altering their direction and speed relative to the sun, known as heliocentric velocity, $V_{\odot}$. The energy gained by the spacecraft during these manoeuvres is lost by the planet in accordance with the conservation of angular momentum. However, due to the massiveness of the planet relative to the spacecraft, this loss in orbital energy is negligible.

Gravity-assists were famously used as a means of interplanetary propulsion for the Voyager missions. Voyager 2, launched in 1967, used gravity-assist flybys of Jupiter, Saturn and Uranus, using each flyby of a planet to propel itself on to the next, arriving at Neptune 12 years later in 1989. The mission took advantage or a rare planetary alignment, occurring only once every 176 years and would have been impossible with chemically fuelled propulsion alone, attesting to the brilliance of Minovitch's discovery and the potential gravity-assist flybys have for reducing ΔV requirements.

Depending on the side of approach, and direction of motion of the planet, a spacecraft can either increase or decrease it's heliocentric velocity, as depicted in Figure 1. Both cases offer a reduction in ΔV requirements, with the latter being useful for scrubbing off speed in order to enter the capture orbit of a planet, or even land on it's surface.

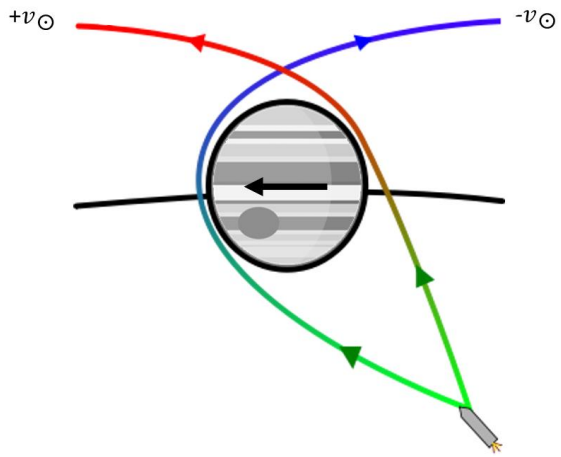

Figure 1 [5]: Variation in heliocentric velocity, $V_\odot$, during a gravity assist flyby, depending on the side of approach and direction of motion of the planet

1.2 Patched Conic Approximation

The software created makes use of the patched conic approximation to reduce problem complexity and computation time. This approximation is a method of simplifying trajectory calculations for a spacecraft in a multi-body environment, such as space, where the gravity of several planets or bodies, as well as the sun, simultaneously act on the spacecraft. The patched conic approximation assumes that only the gravity of the Sun influences the motion of the space craft, except during a gravity assist flyby, whereby only the gravity of the planet being flown by influences the motion of the spacecraft. This allows a multi-planet mission to be considered as a series of two-body problems that can be solved using classical mechanics. The sections of trajectories are then patched together forming a full mission trajectory. This is far easier and less computationally expensive than simulating the notoriously complex multibody problem, which cannot be solved analytically.

Whilst under influence of the sun, the reference frame is said to be heliocentric. Likewise, when under influence of the planet being flown by, the reference frame is said to be planetocentric. Newton's law of universal gravitation [6] describes the relationship between the gravitational force, F, acting between two objects of mass m_1 and m_2, and the distance between their respective centre of masses, r, where G is the gravitational constant.

$$F = G\frac{m_1 m_2}{r^2} \quad (1)$$

This force, being inversely proportional to the square of the distance, as seen in Eq.(1), means that at a certain point when travelling away from a planet, the gravitational force of the Sun acting on the spacecraft overwhelms that of the planet's. At this point, the reference frame used switches from planetocentric to heliocentric. An imaginary sphere surrounding the planet, known as a sphere of influence, can be used to define the border between these reference frames. The radius of this imaginary sphere is a function of the planet's mass relative to the Sun, as well as it's distance from the Sun [7]. Whilst the radii of spheres of influence are far larger than the radii of their respective bodies, the vast distance between planets in our solar system means the time spent in planetocentric reference frames is negligible. As such the patched conic approximation assumes flybys to be instantaneous. This reduces multi-planet missions to a series of heliocentric trajectories connected by gravity-assist flybys that serve to patch together heliocentric legs of the mission.

1.3 Interplanetary Trajectories

Due to motion of planets as they orbit the Sun, interplanetary missions can be compared to throwing a dart at a moving dartboard. That is to say; you need to aim at where the target will be when the projectile reaches it, rather than where the target currently is. The problem of computing a trajectory that connects two points in space, in a specified time, is known as Lambert's problem. If we consider a simple hypothetical Earth-Mars mission, where the date of departure from Earth is known, as is the desired date of arrival at Mars, then the time of flight is the difference between them. As the position vectors of solar system bodies are functions of time, the

planet's positions are also known, providing us with all the information needed to solve Lambert's problem and compute a heliocentric trajectory between the planets. Further analysis is then required to decide if such a trajectory is desirable or even feasible. Nevertheless solving Lambert's problem is fundamental to interplanetary flight.

Various methods of solving Lambert's problem exist. The software created uses a solver developed by Oldenhius [8]. The solver uses the Lancaster Blanchard method [9] with modifications from Goodings [10], as well as a method provided by Izzo et al [11]. The output of the solver is a pair of velocity vectors that represent the required heliocentric velocity at departure and arrival. Paired with the position vectors of the planets, these velocity vectors form a state vector at departure and arrival for the spacecraft. From these state vectors, all the orbital elements necessary to define a trajectory can be computed using a function from Curtis [12].

If the heliocentric velocities of the planets at departure and arrival are known, then the total ΔV for the heliocentric trajectory between the planets can be computed as a function of departure and arrival time [13]. Through calculating the required ΔV for each trajectory in a specified range of departure and arrival times, it is possible to determine the optimal combination that minimises ΔV.

1.4 Planetary Flyby Trajectories

As previously mentioned, if not looking to enter a capture orbit, it is the job of a gravity -assist flyby to patch together incoming and outgoing heliocentric trajectories. If the heliocentric trajectories before and after a flyby have been defined, then the heliocentric velocities upon entering and exiting a sphere of influence are known from the solution to Lambert's problem. Whilst it is possible to have an entirely unpowered, ballistic, gravity-assist flyby, often an undesirable alignment of solar system bodies causes the difference in incoming and outgoing heliocentric velocities to be too great for a ballistic gravity-assist flyby alone to patch. In this case, a small fuel burn of magnitude ΔV is carried out at the point of closest approach to the planet. This maneuver, known as an Oberth maneuver, maximizes the efficiency of the fuel burn as described by the Oberth effect [14]. Essentially, at higher speeds an engine creates greater mechanical energy than at lower speeds. Kepler's third law [15] dictates that a body orbiting another is travelling fastest at perigee - the point of closest approach. Therefore, to maximize efficiency and minimise ΔV, the fuel burn is carried out at this point.

To consider the geometry of a gravity-assist flyby, we first need to convert the heliocentric velocities entering and exiting the sphere of influence ($V_{\odot}^{IN}$ & $V_{\odot}^{OUT}$), as defined by the solution to Lambert's problem, to the planetocentric frame of reference. Known as hyperbolic excess velocity, V_{hyp}, the spacecraft's planetocentric velocity entering and exiting the sphere of influence can be calculated using Eq.(2) and Eq.(3), where V_p denotes the heliocentric velocity vector of the planet.

$$V_{hyp}^{IN} = V_{\odot}^{IN} - V_p \tag{2}$$

$$V_{hyp}^{OUT} = V_{\odot}^{OUT} - V_p \tag{3}$$

The sphere of influence bounds a closed system. Therefore, if the flyby is purely ballistic, the magnitude of V_{hyp}^{IN} is equal to that of V_{hyp}^{OUT}. The change in heliocentric velocity is a result of the turning of V_{hyp}^{IN}, known as the turning angle. The greater the turn angle the greater the heliocentric velocity change. Turn angle is dependent on the distance of closest approach, which itself is constrained by the radius of the planet and the altitude of it's atmosphere. It is common to use a safety factor of 1.1 times the planet's radius [16], however, in some cases, the existence of planetary rings further extends the minimum distance of closest approach. Whilst impossible to achieve in practice, the theoretical maximum potential ΔV a planet can provide a spacecraft during a flyby is derived in Lubunksy et al [17] and shown in Table 1.

Table 1: Maximum possible values of spacecraft velocity increase due to ballistic planetary flybys [17]

Body	ΔV_{max} **[km/s]**
Mercury	3.01
Venus	7.33
Earth	7.91
Mars	3.55
Jupiter	42.73
Saturn	25.62
Uranus	15.18
Neptune	16.75
Pluto	1.10

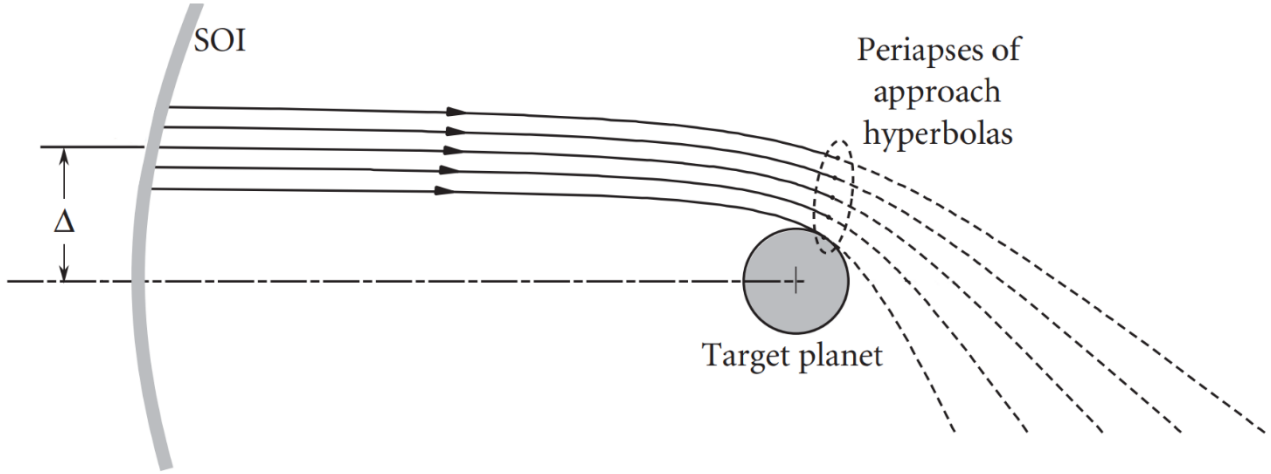

Figure 2: Variation in turn angle and distance of closet approach for varying aiming radii [18]

In a flyby, a spacecraft aims for a specific distance of closest approach in order to turn V_{hyp}^{IN} the desired angle to match V_{hyp}^{OUT}. To aim for a specific distance of closest approach, the guidance and navigation system of the spacecraft enters itself into the sphere of influence at a distance above the center line known as the aiming radius, Δ. Figure 2 from Curtis [18] neatly depicts how a variation in aiming radius results in a variation of distance of closest approach and therefore turn angle.

If an Oberth maneuver is necessary to patch the incoming and outgoing heliocentric trajectories, the magnitude of the fuel burn, ΔV, can be calculated and added to the ΔV of the heliocentric trajectories to provide a total ΔV for the mission.

2. METHODOLOGY

Creating useful software tools for early phase mission designers involved the programming of a series of functions that simulate a spacecraft's trajectory, relating ΔV to the time of departure and arrival at specified bodies. The enumerative approach taken allows the software created to visually display this data, aiding the mission designer in their trade off analysis of potential trajectory concepts. As each new function is dependent on the output of the previous, verification of the current function's outputs further verifies the outputs of previous functions.

2.1 Ephemeris Model

Fundamental to all calculations in the simulation are the positions of solar system bodies. An ephemeris model is exactly this, a 3-dimensional model of solar system bodies that contains information about their position and velocity over time. Whilst accurate position and velocity data of solar system bodies are known, creation of my own model allows highly specific position data to be returned quickly when a time is specified. Using orbital element data, and a method sourced from NASA's Jet Propulsion Laboratory (JPL) [19], I was able to program a function to return the heliocentric position vector of any of the 9 major solar system bodies, given a time as an input.

The approximate errors in right ascension, declination and distance for this method of modelling were provided by JPL [20] and are shown in Table 2 below.

Table 2: Approximate errors in right ascension, RA, declination, Dec, and distance, r, in the formulation of the ephemeris model [20]

Body	RA [arcsec]	Dec [arcsec]	r [1000 km]
Mercury	15	1	1
Venus	20	1	4
Earth-Moon Barycenter	20	8	6
Mars	40	2	25
Jupiter	400	10	600
Saturn	600	25	1500
Uranus	50	2	1000
Neptune	10	1	200
Pluto	5	2	300

Note that, due to the perturbing effects of the moon on the Earth's orbit, the Earth-Moon barycentre is used in the ephemeris model and throughout the project. For reference the barycentre is located at approximately 75% of the Earth's radius on average.

2.2 Trajectory Determination

With position vectors programmed as a function of time, Lambert's problem can now be solved. The solver used makes use of two methods to reduce computation time. This becomes especially useful when computing a solution to Lambert's problem for a large series of input parameters. The first method, whilst faster, is more likely to fail in finding a solution. If the first method fails to find a solution then the same input parameters are passed into the second method, which whilst slower, is far more robust and renowned.

As previously discussed, the output of the Lambert solver used, are a pair of heliocentric velocity vectors. These vectors represent the required velocity at departure and arrival to get from one position vector to the other in a specified time. To verify the solution provided by the lambert solver, data for a circular orbit test case was input, and the velocities output checked against hand calculations. Further verification entailed inputting data for planetary orbits and comparing the velocity outputs to that of JPL's HORIZONS system [21] which provides high accuracy ephemerides.

2.3 Interplanetary Optimisation

Once a heliocentric trajectory has been determined, the ΔV required to travel along such a trajectory can be calculated using Eq.(4) [22].

$$\Delta V = \left|V_{p1}(t_1) - V_{SC}(t_1)\right| + \left|V_{p2}(t_2) - V_{SC}(t_2)\right| \qquad (4)$$

Where V_{SC} represents the heliocentric velocity of the spacecraft, the values of which come from the solution to Lambert's problem. V_{p1} and V_{p2} are the velocities of the departure and arrival planet, respectively. As all parameters are a function of departure time, t_1, or arrival time, t_2, the ΔV for any trajectory between any two specified planets can be computed given departure and arrival times as inputs.

Through user specification of a range of arrival and departure times, the software creates a mesh of grid points in the x-y plane for all possible departure-arrival combinations. At each grid point, ΔV is computed using Eq.(5) with the respective departure and arrival times as inputs. The value is then stored on the z-axis. The program then returns the minimum value of ΔV along with the respective combination of departure and arrival times. The time step of the departure and arrival ranges, defined by the user, determines the refinement of the mesh. Decreasing the time step increases the mesh refinement and therefore probability of finding a more optimal departure-arrival combination and decreased ΔV value. However, this comes at the cost of increased computation time, as an increased number of grid points means an increase in the number of computations. Such data is often represented in a graph called a 'Porkchop Plot'. In such plots, the z-axis data, values of ΔV, are represented as a colour in a heatmap style fashion. This allows the data to represented in 2 dimensions, improving the readability of the data.

Verifying the determined optimal departure and arrival times was achieved through using the function to predict the optimal times for a hypothetical mission and comparing the results to that of an actual mission. The example presented in this paper is the prediction of launch and arrival times for the upcoming NASA mission to Mars.

2.4 Gravity-Assist Flyby Optimisation

With interplanetary heliocentric trajectories determined and potentially optimised, we can begin to consider optimising the conditions of a gravity assist flyby to minimise ΔV. It is assumed that the heliocentric velocity of the spacecraft entering the sphere of influence is equal to the output of the Lambert solver. This approximates the velocity to be that of arrival at the center of the respective body, not at the boundary of sphere of influence. It is also assumed that the object being flown by is spherically symmetric producing a uniform field of gravity.

The function created first computes the incoming and outgoing hyperbolic excess velocities. If their magnitudes are equal, the flyby is ballistic. Providing, that is, the required turn angle is achievable with a distance of closest approach above that of the minimum.
If the magnitudes of the hyperbolic excess velocities are not equal, an Oberth maneuver is necessary to patch the desired trajectories. As previously discussed, the patched conic approximation used throughout this model assumes flybys to be an instantaneous occurrence. As such, the Oberth maneuver's fuel burn is approximated as an instantaneous impulse, where the velocity of the spacecraft changes but not it's position. This results in the geometry of the flyby being two hyperbolas connected at the point of closest approach, where the fuel burn takes place. The magnitude of the fuel burn, ΔV, is dependent on the distance of closest approach. A method for determining this distance for a powered flyby is outlined in Wagner and Wie [23]. The function created uses this method to first

compute the semi-major axis of the two hyperbola, before using the Newton Rapshon numerical method to determine the eccentricity values of the hyperbolas. With this information the distance of closest approach, aiming radius and required ΔV to patch the trajectories can be computed.

Due to the infinite number of possible hyperbola geometries, determination of a hyperbola's eccentricity values using a numerical method is not always successful. If the initial guess is too far from the actual solution, then the process may diverge giving no solution. If the number of permitted iterations allowed by the program is too low and the tolerance of the solution too tight, the numerical method may be deemed unsuccessful by the software, despite potentially beginning to converge on a solution. Increasing the number of permitted iterations improves the probability of finding a converging solution that meets the tolerance requirements set by the user. The trade off being increased computation time once again.

As all parameters of the flyby were programmed as a function of time, the result of varying either the time of departure, time of flyby, or time of arrival at the destination planet can be monitored and optimised to minimise ΔV. To verify the gravity-assist flyby model, the parameters of a known ballistic flyby were input and the required ΔV to patch the trajectories, output by the function, was observed and compared. The example shown in this paper used Voyager 2's ballistic flyby of Jupiter as a test case.

3. RESULTS AND DISCUSSION

The objective of the project was to produce software tools accurate enough for early phase mission designers to assess the feasibility of potential trajectory concepts. Through comparison of real-world mission data and the results of the software tools, the success of the project objective will be assessed.

3.1 Interplanetary Optimisation

Table 3 shows the defined range of departure and arrival times passed into the interplanetary optimisation function.

Table 3: Time Ranges for departure and arrival used to produce Figure 3

Body	Minimum	Maximum
Earth (departure)	01/04/20	07/12/20
Mars (arrival)	05/08/20	05/02/22

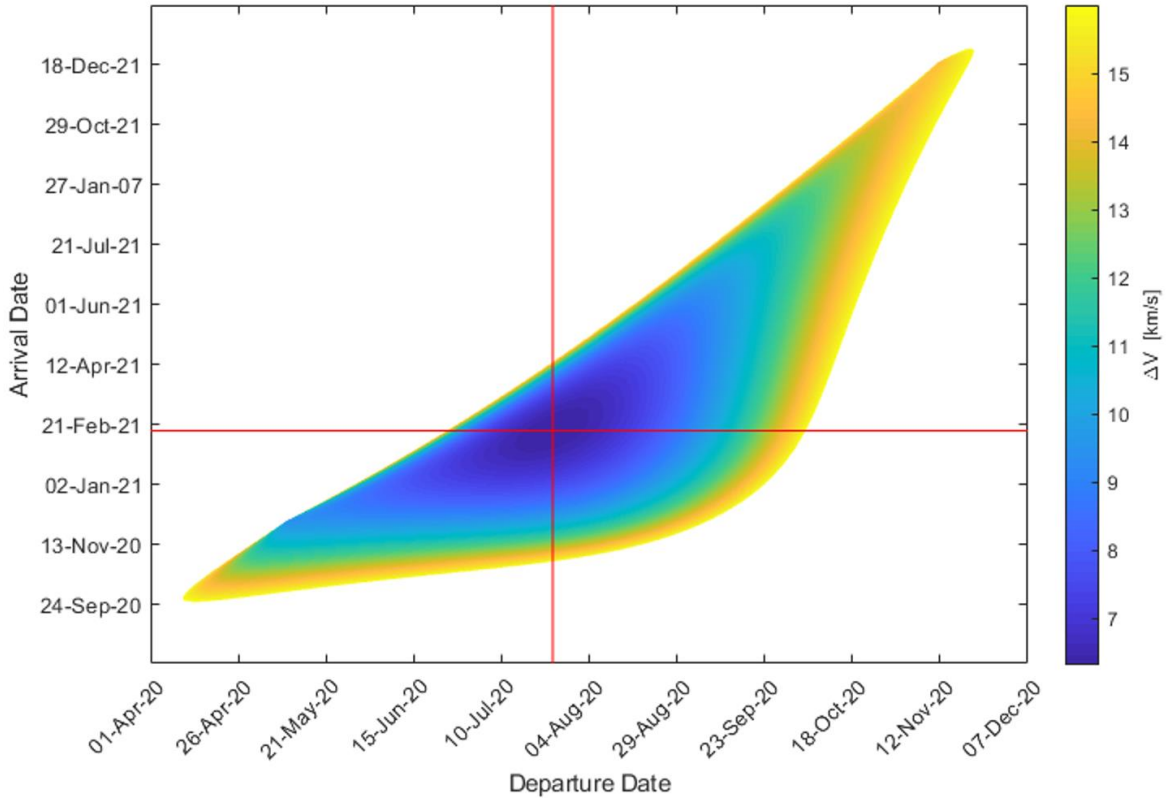

Figure 3: Porkchop plot showing the ΔV cost for varying departure and arrival dates for ranges defined in Table 3. Red crosshair indicates optimal combination of departure and arrival times to minimise ΔV, as determined by the software

1000 time steps were used between the minimum and maximum departure and arrival dates respectively, leading to grid of 1 million points. The ΔV cut off threshold was set to $16\ km/s$ as found in other literature, though the value is somewhat arbitrary and only really serves to better show the contrast in ΔV values.

The function then returned a porkchop plot, as seen in Figure 3, as well as the minimum possible ΔV value and corresponding departure-arrival times. Ticks on the x-axis, representing departure date, are in increments of 25 days, with the y-axis, representing arrival date at Mars, being in increments of 50 days.

The optimal departure date from Earth, as identified by the software, is 25/07/20. This falls within NASA's proposed launch window of 18/07/20 – 05/08/20, for their planned Mars Perseverance Rover mission [24]. If we assume NASA's ideal launch date is at the centre of their proposed launch window, then the optimal solution, as determined by the software tool created, is 2 days from this centre point of 27/07/20. The optimal arrival date at Mars, as identified by the software, is 15/02/21. 3 days from NASA's planned landing date of 18/02/21.

Taking a cross section of figure 3, and instead plotting ΔV values on the z-axis rather than colourising them, would result in figure 4, which shows the variation in ΔV for NASA's proposed launch window and fixed arrival date. The minima of this curve corresponds to a minimum ΔV value of $6.3176\ km/s$. This is a 0.07% increase in minimum ΔV compared to the software's determined minimum and gives some indication as to the accuracy of the model.

As the position vectors fed into Lambert solver are those of body centres, any trajectory determined starts and ends at the centre of a body, or in the case of Earth, it's barycentre. This approximation doesn't account for NASA's actual mission destination of Mars' surface, nor does it account for the deceleration necessary to land on Mars. The optimal combination of departure and arrival, as decided by NASA, may also be influenced by other factors such as the velocity on arrival at Mars. Each of the assumptions made causes a degree of error that can potentially account for the slight discrepancies observed in the determined optimal departure-arrival date combination. Despite the observed discrepancy, the software performed well, arguably meeting it's objective, identifying a proposed flight time of

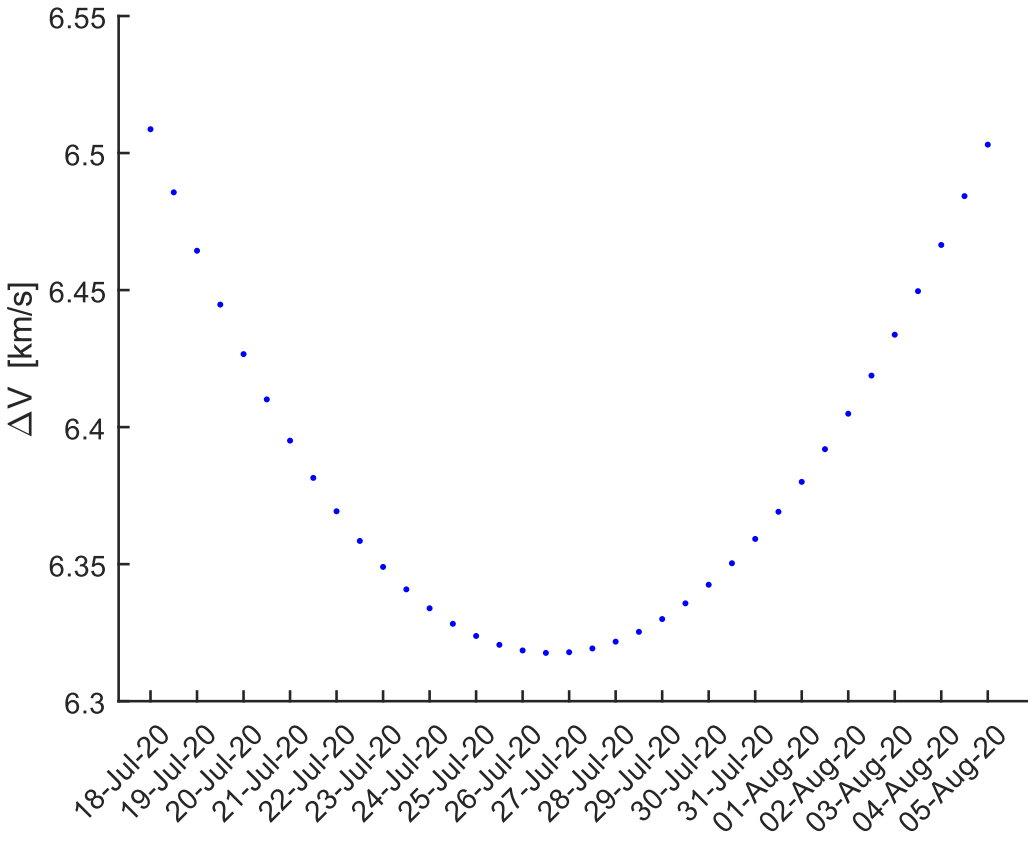

Figure 4: Graph showing variation in ΔV over NASA's proposed launch window of 18/07/20 – 05/08/20 assuming a fixed date of arrival at Mars of 18/02/21

207 days compared to NASA's proposed range of 194 – 212 days.

It should be noted that the software tool created only considers type-1 interplanetary trajectories. These are trajectories that fly less than 180 degrees around the Sun. Type-2 trajectories, categorised as flying greater than 180 degrees around the Sun, are often used to achieve a lower heliocentric velocity upon arrival at the target body. As the project is only concerned with using gravity-assist flybys as a means of interplanetary propulsion, a decrease in heliocentric arrival velocity is undesirable. As such type-2 trajectories were omitted from the model. This had the added benefit of simplifying the model and reducing computation time.

3.2 Gravity-Assist Flyby Optimisation

The test case used to verify the results of the gravity-assist flyby function was Voyager 2's ballistic flyby of Jupiter. Epoch data from NASA [25], used as the inputs for the function, are shown in Table 4. It should be noted that as Voyager 2 also performed a flyby of Saturn, the date approximated as being the spacecrafts arrival date at the planet, is in fact the date of it's closest approach during the flyby.

Table 4: Voyager 2's dates of departure, closest approach and arrival at respective planets compared to that proposed by the software created for a purely ballistic flyby

Body	Actual [25]	Proposed
Earth (departure)	20/08/77	20/08/77
Jupiter (flyby)	09/07/79	13/07/79
Saturn (arrival)	25/08/81	24/08/81

Using NASA's epoch data, the function created determined a fuel burn maneuver of magnitude $\Delta V = 0.1487\ km/s$ neccessery. Increasing the time of flight for the Earth-Jupiter leg of the mission by 0.58% and decreasing the time of flight for the Jupiter-Saturn leg by 0.14%, resulted in the software determining the flyby to be ballistic. The proposed dates as determined by the software can be found in Table 4.

Discrepancies in epochs can be accounted for due to assumptions and approximations made in the model, as previously stated. Despite the lack of consideration for course corrections or external forces, the model performs at a sufficient level to meet its objective.

4. CONCLUSION

Due to the exponential relationship between ΔV and fuel fraction, exploration of the outer solar system was thought to be impossible until the discovery of gravity-assist trajectories. Minimising the ΔV requirement for a mission has significant impact on it's feasibility, as a reduction in ΔV leads to a reduced fuel requirement and therefore launch cost. The aim of this project was to supply academia and early stage mission designers with software tools able to determine trajectory concepts, minimising ΔV through optimisation of mission characteristics. These software tools serve as an initial feasibility study for potential missions and the data from them can be passed on to more advanced, higher accuracy simulation systems.

An ephemeris model was created to provide positional data as a function of time. Interplanetary and gravity-assist flyby functions, drawing upon this positional data, were then created, with ΔV being programmed as a function of time. The software tools created were then verified against real life mission test cases. The interplanetary optimisation tool was used to predict the optimal departure and arrival date combination for the upcoming NASA Earth-Mars mission. The gravity-assist flyby tool was tested against data from Voyager 2's ballistic flyby of Jupiter. Results showed that, despite the approximations and assumptions used in simplifying the model, the software tools are sufficiently accurate to meet the requirements of the objective.

REFERENCES

[1] **Jones, H.W.** *The Recent Large Reduction in Space Launch Cost (2018).* 48th International Conference on Enviromental Systems, p. 2.

[2] **Tsiolkovsky, K. E.** *Exploratoin of Spcae Using Reaction Devices (1903).*

[3] **Pettit, D.** *The Tyranny of the Rocket Equation (2012).* [Online, Accessed: 08/05/20] Available at: https://www.nasa.gov/mission_pages/station/expeditions/expedition30/tryanny.html

[4] **Minovitch, M.** *An Alternative Method for the Detemination of Elliptic and Hyprbolic Trajectories (1961).* Jet Proulsion Laboratory, Technical Memo #312-118.

[5] **Kerbal Space Program.** *Gravity Assist Diagram [edited].* Figure 1. [Online, Accessed: 08/05/20] Available at: https://wiki.kerbalspaceprogram.com/wiki/File:Gravity_Assist.svg

[6] **Newton, I.** *Principia (1728).* Book 3, General Scholium, p.392 in Volume 2 of Andrew Motte's English Translation (1729).

[7] **Minovitch, M.** *An Alternative Method for the Detemination of Elliptic and Hyprbolic Trajectories (1961).* Jet Proulsion Laboratory, Technical Memo #312-130, p. 2.

[8] **Olenhuis, R.** *Robust Solver for Lambert's Orbital-Boundary Value Problem (2018).* [Online, Accessed: 08/05/20] Available at: https://www.github.com/rodyo/FEX-Lambert

[9] **Lancaster, E.R., Blanchard, R.C.** *A unified form of Lambert's theorem (1969).* NASA technical note TN D-5368,1969.

[10] **Gooding, R.H.** *A procedure for the solution of Lambert's orbital boundary-value problem (1990).* Celesticl Mech Dyn Astr 48, pp. 145-165.

[11] **Vinkó, D., Tamás, B., Izzo, C.** *Benchmarking Different Global Optimisation Techniques for Preliminary Space Trajectory Design (2007).* 58th International Astronautical Congress.

[12] **Curtis, H.** *Orbital Mechanics for Engineering Students (2005).* Elsevier Butterworth Heinemann, pp. 606-610.

[13] **Woolley, R, C., Whetsel, C. W.** *On the Nature of Earth-Mars Porkchop Plots (2013).* Jet Propulsion Laboratory, California Institute of Technology, p. 3.

[14] **Oberth, H.** *Wege zur Raumschiffahrt (1929).* Oldenbourg. 3rd edition.

[15] **Kepler, J.** *New Astronomy Translated by William, H. (1992).* Cambridge University Press.

[16] **Ceriotti, M.** *Global Optimisation of Multiple Gravity Assist Trajectories (2010).* PhD Thesis, University of Glasgow, p. 39.

[17] **Labunksy, A. V., Papkov, O. V., Sukhanov, K. G.** *Multiple Gravity Assist Interplanetary Trajectories (1998).* Gordon and Breach, p. 15.

[18] **Curtis, H.** Aiming Radius Diagram (as seen in Figure 2) [edited]. *Orbital Mechanics for Engineering Students (2005).* Elsevier Butterworth Heinemann, p. 371.

[19] **NASA Jet Propulsion Laboratory, California Institute of Technology.** *Keplerian Element Data for Approximate Positions of the Major Planets.* [Online, Accessed: 08/05/20] Available at: https://ssd.jpl.nasa.gov/txt/aprx_pos_planets.pdf

[20] **NASA Jet Propulsion Laboratory, California Institute of Technology.** *Keplerian Element Data for Approximate Positions of the Major Planets – Approximate Errors.* [Online, Accessed: 08/05/20] Available at: https://ssd.jpl.nasa.gov/?planet_pos

[21] **NASA Jet Propulsion Laboratory, California Institute of Technology.** *HORIZONS Web-Interface.* [Online, Accessed: 08/05/20] Available at: https://ssd.jpl.nasa.gov/horizons.cgi#top

[22] **Woolley, R, C., Whetsel, C. W.** *On the Nature of Earth-Mars Porkchop Plots (2013).* Jet Propulsion Laboratory, California Institute of Technology, p. 3.

[23] **Wagner, S., Wie, B.** *Hybrid Algorithm for Multiple Gravity-Assist and Impulse Delta-V Maneuvers (2015).* Journal of Guidance, Control and Dynamics, Vol. 38, No. 11, pp. 2097-2098.

[24] **NASA Science.** *Mars 2020 Mission Perseverance Rover (2020).* [Online, Accessed: 08/05/20] Available at: https://mars.nasa.gov/mars2020/

[25] **NASA Jet Propulsion Laboratory, California Institute of Technology.** *Voyager Missions Timeline.* [Online, Accessed: 08/05/20] Available at: https://voyager.jpl.nasa.gov/mission/timeline/#event-voyager-2-launches

