MISSION FEASIBILITY FROM TRAJECTORY OPTIMIZATION AND THE STATE OF SPACE SYSTEMS RESEARCH AT THE UNIVERSITY OF AUCKLAND

Darcey R. Graham, Nicholas J. Rattenbury, John E. Cater

New Zealand has very recently become a space-faring nation, and so it is at an exciting time deciding where its interests lie. The current state of space systems research at the University of Auckland, where focus is on inexpensive small satellites, is presented with methods to assess the feasibility of future missions based on trajectory optimization. The low-thrust and low-\(\Delta v\) capabilities of both old and novel electric propulsion systems place significant limitations on future missions, so limiting \(\Delta v\) by minimizing fuel requirements will be the objective of trajectory optimization. Different methods of trajectory optimization are compared.

INTRODUCTION

Thanks to Rocket Lab USA, New Zealand became the 11\textsuperscript{th} nation to successfully launch objects to space. However, space systems research in New Zealand is still developing the means to fully utilize this. This paper provides an overview of ongoing research undertaken at the Te Pūnaha Ātea – Auckland Space Institute at the University of Auckland, with a focus on research in mission feasibility.

Research in space systems at the University of Auckland focuses on the development of systems for small, inexpensive satellites such as CubeSats. The propulsion systems utilized are correspondingly small, with research being undertaken regarding electric propulsion and solar sails. These all have very low thrust capabilities. Owing to the small size and mass allowances of the satellites involved, they also have significant fuel usage limitations. As such, in trajectory design, the impulsive models such as Hohmann transfers used to model high-thrust systems such as chemical thrusters are inappropriate. Trajectories must instead be designed using low thrust models, and the dynamics of the system can be exploited to find transfers with energy low enough to be accomplished by small spacecraft.

Complex trajectories can be computed in the context of the circular restricted three-body problem (CR3BP). The CR3BP includes nonlinear dynamics which may be exploited to provide low energy transfers. Methods for achieving this in the context of mission design and feasibility analysis will be described in this paper.

Once an initial trajectory is designed, the fuel use can be further minimized using optimization techniques. There are a large number of optimization techniques available, several often appropriate for any given task. Hybridization of optimization techniques may be used to select the advantages
of several different techniques at once, and to overcome specific disadvantages. Trajectory optimization is used to minimize or maximize a condition within a model, in this case maximizing final fuel mass in order to minimize the change in velocity $\Delta v$ required to make the transfers.

This paper is structured as follows: an overview of space systems research at the University of Auckland; the model and methods used to assess the feasibility of the use of different low thrust propulsion systems are outlined; and finally the optimization techniques possible are assessed qualitatively.

**SPACE SYSTEMS RESEARCH AT THE UNIVERSITY OF AUCKLAND**

Research currently underway at the Institute includes projects in deployables, synthetic aperture radar (SAR), electric propulsion, and astrodynamics. Deployable and inflatable devices are being developed with work on a foldable panel reflectarray antenna for SAR, and work inspired by origami is being undertaken to develop new folding patterns for use in deploying solar sails. Undergraduate students are involved to guide more students into the industry; The Auckland Programme for Space Systems (APSS) has proven extremely useful in this regard. APSS is an undergraduate competition open to students from any discipline. The winning team build and launch their own CubeSat. The first iteration of this competition, the satellite APSS-1, is due to be launched by Rocket Lab later this year and will be the first spacecraft designed, built and launched in New Zealand.

A very active area of space systems research at the Institute is in in-space propulsion, specifically electric propulsion. Work is being done in collaboration with the Australian National University (ANU) on the “pocket rocket”, a plasma source for plasma thrusters. This form of propulsion is small enough to fit on a CubeSat, but provides very little thrust. Even methods of trajectory design used on existing electric propulsion missions are not always applicable here because the mass constraints on small satellites are so significant, and because many of these designs rely on bus vehicles using chemical thrusters to transport the spacecraft some of the way.

Thus new trajectories must be designed for these systems. The present research will determine the capabilities of spacecraft built by the University of Auckland, assessing the feasibility of potential space missions. This is centred around the development of trajectory design methods to find trajectories of extremely low energies, exploiting the chaotic dynamics of the dynamical systems involved.

**TRAJECTORY DESIGN**

In order to test the feasibility of mission concepts, a preliminary trajectory design can be created and used to test the reach of propulsion systems. Of interest to the Institute are high-Earth orbits and interplanetary trajectories to Venus. An example of the latter, which is the main focus of this paper, is shown in figure 1.

Gaining low energy access to interplanetary trajectories is possible via the unstable (first, second, and third) Lagrange points. The Lagrange points and the family of periodic orbits around them are associated with dynamical structures called manifolds, tube-like collections of all trajectories which naturally evolve towards (stable) or away from (unstable) the Lagrange point. These reach into the interior (between the primary and the Lagrange point) and exterior (outside the Lagrange point) regions of the system as shown in figure 2.

After reaching a halo orbit from the stable manifold, the exterior unstable manifold can be followed out of the primary’s sphere of influence and patched to an interplanetary trajectory. By in-
vestigating the ability of the propulsion systems to achieve this, we can assess how promising they each are for interplanetary trajectories. The method considered here is one of patched manifolds.\textsuperscript{6–8} Other methods are possible, such as a patched periodic method.\textsuperscript{9}
The full $n$-body problem is too computationally expensive for preliminary design, so a simplified version, the circular restricted three-body problem (CR3BP),\textsuperscript{10} is often used. This dynamical system considers two large masses $M_1$ and $M_2$, the primaries. A third body, the secondary, has a negligible mass. The CR3BP models the spacecraft’s movement through space as it is affected by its gravitational attraction to the primaries. This model splits the system into three stages: the escape phase, the interplanetary phase, and the capture phase. In both escape and capture phases, $M_1$ is the Sun and the secondary is the spacecraft. In the escape phase, $M_2$ is the Earth, and in the capture phase, it is Venus. Figure 3 demonstrates the system.

The equations of motion are:

\[
\begin{align*}
\ddot{x} &= 2\dot{y} + \frac{\partial U}{\partial x}, \\
\dot{y} &= -2\dot{x} + \frac{\partial U}{\partial y}, \\
\ddot{z} &= \frac{\partial U}{\partial z},
\end{align*}
\]  

where $x, y, z$ are the coordinates on the x, y, and z axes, and $U$ is the pseudo-potential of the system. Propulsion can be modelled by including a thrusting term in the equations of motion.\textsuperscript{6} The pseudo-potential is defined:

\[
U = \frac{x^2 + y^2}{2} + \frac{(1 - \mu)}{r_1} + \frac{\mu}{r_2},
\]

where $\mu$ is the mass fraction

\[
\mu = \frac{M_1}{M_1 + M_2},
\]
and $r_1$ and $r_2$ are the distances from the spacecraft to the Sun and Earth respectively;

$$r_1 = \sqrt{(x + \mu)^2 + y^2 + z^2},$$  \hspace{1cm} (6) $$r_2 = \sqrt{(x + \mu - 1)^2 + y^2 + z^2}. \hspace{1cm} (7)$$

Further terms can be included in the expression for $U$ to account for perturbations due to the gravitational pull of external bodies, radiation pressure, etc., but for this simplified preliminary study these additional factors have been neglected.

The dynamical system includes equilibrium (Lagrange) points where the pseudo-potential reaches zero, the locations of which can be computed using the method described in Koon et al. (2011). We are examining the L₁ point as this is between the Earth and Venus. The Lagrange points in the system are shown in figure 4.

Another important feature of the dynamical system is the Hill’s region. This is a forbidden region into which the spacecraft cannot travel and is bounded by the zero-velocity curve. The Hill’s region is determined by the Jacobi integral, the integral of motion of the differential equations of motion from equations 1 - 3:

$$J(x, y, z, \dot{x}, \dot{y}, \dot{z}) = 2U(x, y, z) - (\dot{x}^2 + \dot{y}^2 + \dot{z}^2), \hspace{1cm} (8)$$

which, when the velocity is zero, i.e. along the zero-velocity curve marking the boundaries of the Hill’s region, reduces to

$$J(x, y, z, \dot{x}, \dot{y}, \dot{z}) = 2U(x, y, z) = C. \hspace{1cm} (9)$$
Figure 5. Hill’s regions for different values of spacecraft Jacobi energy $C$: a) $C < C_1$, b) $C_1 < C < C_2$, c) $C_2 < C < C_3$, d) $C_3 < C < C_4 = C_5$, e) $C_4 = C_5 < C$. In each case the shaded region is the Hill’s region, the zero velocity curve is the black line, the green crosses are the Lagrange points, the red point is the Sun, and the blue point is the Earth. Unshaded (white) regions are the regions in which the spacecraft can travel.

$C$ is the Jacobi energy. As seen in figure 5, the zero-velocity curve depends on the Jacobi energy of the spacecraft. The Lagrange points each have their own Jacobi energy $C_n$ where $n$ is the number of the Lagrange point. When the Jacobi energy of the spacecraft $C$ has a value $C_1 < C < C_2$, such that the Jacobi energy of the spacecraft is slightly greater than $C_1$, a neck in the Hill’s region opens up between the two primaries. This neck is located at the $L_1$ point, allowing the spacecraft to pass by the $L_1$ point and transfer between the two primaries. Were an outer planet the target, we would require the case in figure 4c), where a neck appears at both $L_1$ and $L_2$ points.

Construction of Manifolds

As a starting point for the construction of manifolds around the $L_1$ point, we analytically construct halo orbits around the $L_1$ point. Halo orbits can be selected from the family of periodic orbits by their out-of-plane amplitude, an example of which is shown in figure 6.

The equations of motion 1-3 are used to propagate a spacecraft along the halo orbit from an arbitrary starting point on the orbit through a full period. From the initial position and velocity values $x, y, z, \dot{x}, \dot{y}, \dot{z}$, we construct a column vector $X$ which evolves in time around the halo orbit. From this we compute the state transition matrix, which is made up of the partial derivatives of the state:

$$\Phi(t,t_0) = \frac{\partial X(t)}{\partial X(t_0)}$$ (10)

where $x_0, y_0, z_0, \dot{x}_0, \dot{y}_0, \dot{z}_0$ are the initial positions and velocities, $t_0$ is the initial time, and $t$ is a selected later time. Initial conditions $\Phi(t_0, t_0) = I$ are applied. To propagate the state transition matrix around the whole orbit, it is propagated forwards in time by
Figure 6. Halo orbit around the 1st Lagrange point computed using the method in Richardson (1980), in three dimensions.

\[ \frac{d\Phi(t, t_0)}{dt} = A(t)\Phi(t, t_0) \] (11)

where the variational matrix \( A(t) \) is a very large matrix which can be broken down into four 3x3 matrices:

\[ A(t) = \frac{\partial \dot{X}(t)}{\partial X(t)} = \begin{pmatrix} O & I \\ \gamma & 2\Omega \end{pmatrix}, \] (12)

where

\[ O = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \Omega = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \gamma = \begin{pmatrix} \frac{\partial U_{xx}}{\partial x} & \frac{\partial U_{xy}}{\partial y} & \frac{\partial U_{xz}}{\partial z} \\ \frac{\partial U_{yx}}{\partial x} & \frac{\partial U_{yy}}{\partial y} & \frac{\partial U_{yz}}{\partial z} \\ \frac{\partial U_{zx}}{\partial x} & \frac{\partial U_{zy}}{\partial y} & \frac{\partial U_{zz}}{\partial z} \end{pmatrix}, \] (13)

and \( U_{xy} \) is the partial derivative, with respect to \( y \), of the partial derivative of the pseudo-potential \( U \) with respect to \( x \).
The state transition matrix propagated in this way around an entire orbit is called the monodromy matrix. As it is propagated around the whole orbit over one period, the monodromy matrix contains information about the state of the spacecraft at each point on the orbit.

The eigenvectors of the monodromy matrix give information about the stability of the halo orbit. The largest real eigenvalue of the monodromy matrix corresponds to the unstable manifold, the smallest to the stable manifold. By extracting their corresponding eigenvectors, the eigenvectors can be used to find the starting points $X_M$ for the manifolds:

$$X_M = X + \epsilon I v_i,$$

where $X$ is the state of the halo orbit at that point, $\epsilon$ is a small perturbation in the stable or unstable direction with a user-defined value, $I$ is a 6x6 identity matrix, and $v_i$ is the eigenvector of the manifold.

The equations of motion for the CR3BP can be solved to determine the path an object would follow starting at this small perturbation from the halo orbit. These paths make up the invariant manifolds. By starting with a perturbation in the unstable or stable direction, then integrating the equations of motion 1-3 forwards in time for the unstable manifold, backwards for the stable manifold, the manifolds as seen in figure 2 can be constructed.

**Constructing Trajectories**

The planned trajectory starts from parking orbits around Earth achievable by Rocket Lab’s Electron rocket. During thrusting phases, an additional thrusting term $T/m \hat{a}$ can be added to the equations of motion 1-3 to model the thrusting spacecraft, where $T$ is instantaneous thrust, $m$ is instantaneous spacecraft mass, and $\hat{a}$ is the acceleration vector in the x, y, or z direction depending on the direction evaluated.

After an initial thrusting phase the spacecraft propagates out in a spiral until its Jacobi energy is equal to that of the L1 point. A Poincaré section is constructed to determine where the trajectory crosses the stable manifold. Further thrusting brings it to the stable manifold, which it will travel along to the L1 point. The spacecraft will depart towards Venus along the exterior unstable manifold, as the stable manifold arrives at the halo orbit when integrated forwards in time and the unstable manifold when integrated backwards in time.

**OPTIMIZATION**

Once the model is built, the trajectory computed must be optimized. The trajectory is provided as an initial guess to an optimization algorithm, which produces the best possible trajectory given the mission constraints; in this case, limited mass and thrust. There are many different methods of doing this which produce valid results, but some will produce more optimal solutions than others. By assessing different techniques, we can choose the most appropriate one for our system. However, it should be noted that it is difficult to say which will produce the most optimal trajectory without building and testing each algorithm.

The objective of optimization is to minimize or maximize a cost function. In this case, required fuel mass be minimized in an optimal trajectory. The cost function is defined:
where $J_{\text{cost}}$ is the cost function, $t_0$ and $t_f$ are the initial and final times respectively, and $m_f$ is the final fuel mass at the end of a transfer.

Optimization is challenging for the system considered due to the exploitation of chaos in the construction of the invariant manifolds. The involvement of high instabilities and chaos gives the computed trajectories a high sensitivity to their initial guess. This sensitivity must be borne in mind when deciding on the optimization technique, as the technique must be robust to poor initial guesses to overcome this. Another challenge is that large numbers of manifolds must be computed in order to find feasible intersections with desired trajectories, making them computationally expensive. This means the chosen optimization technique must allow for non-intuitive systems and must be fast.

Multiple shooting\textsuperscript{19} reduces the sensitivity to the initial guess by splitting the interval over which the trajectory is integrated, reducing the propagation of errors from the initial guess. The trajectory is split into sections, and each section is optimized. The sections are then patched together. Because the patching is never perfect, optimization is then performed again over all the patched trajectory.

Trajectories can either be solved using an analytical or numerical approach. Analytical approaches typically use optimal control theory for optimization. This determines the time history of the trajectory while simultaneously satisfying constraints and minimizing the cost function. The systems considered contain non-linear problems with no analytical solutions, so numerical methods are used instead. Impulsive thrust models do not face this problem, but as continuous thrust models must use numerical integration to propagate them forwards, numerical methods must be used. This creates a two-point boundary value problem which are typically very difficult to solve.

There are two main types of numerical approach:\textsuperscript{20} indirect and direct. Indirect methods consider both state and input vectors based on Pontryagin’s Principle. These use calculus to give exact solutions to the optimization problem with few design variables. Indirect methods are not appropriate for the systems considered here as they depend strongly on the accuracy of the initial guess. As the co-states must be considered in analytical form are discretized, the problem becomes much larger and more computationally expensive.

Direct methods minimize the cost function by considering the state and input vectors. Unlike indirect methods, analytical expressions are not required. The method is flexible to new models and easy to use. Although less accurate than indirect methods, direct methods have the advantage of being more robust. They are less sensitive to the initial guess. However, by discretizing a continuous problem, they generate a large number of errors. Direct methods are a better choice than indirect for this model due to their robustness.

Differential Dynamic Programming (DDP)\textsuperscript{21, 22} divides the optimization problem into stages with their own associated subproblems, where a recurrence relation links the subproblems together. DDP can be applied to discrete problems, but is not an appropriate choice for continuous problems such as ours due to the curse of dimensionality: the state space increases exponentially when a large number of variables are considered such as in a continuous problem.
Nonlinear Programming

Nonlinear programming (NLP)\textsuperscript{23,24} is a popular gradient based computational technique used to solve optimization problems. Thanks to its popularity, many software packages are available, such as SNOPT\textsuperscript{25} and Matlab’s function \texttt{fmincon()}. With so much attention, NLP has become an accessible and reliable technique, giving fast and accurate results with much user support available.

As a gradient based method, NLP does require an initial guess, making it more sensitive to initial conditions. Poor guesses may lead to failure to converge on a solution. However, it is not impossible to use NLP for continuous models. If the constraints on the initial and final states are specified, a suboptimal solution can be fed into the NLP solver to generate an optimal solution. The suboptimal solution may be generated by numerically solving a trajectory problem in stages, typically done by splitting the trajectory into escape, interplanetary, and capture phases, and then patching them together.

Genetic Algorithms

Genetic algorithms (GA)\textsuperscript{26} are the first choice solver for many optimization problems in spacecraft trajectories as, like NLP, there are many easy-to-use options available. They are often used in gravity assist missions. GAs are probabilistic so can give different results when re-run, and have no convergence criteria. To counter this, a penalty parameter can be introduced.

An initial population is randomly generated. Following survival of the fittest seen in the natural world, with each iteration of the algorithm the poorest solutions are eliminated until an optimal solution is determined. GAs are better for impulsive trajectories than continuous ones, as they are better for difficult problems such as where the problem space is multimodal or discontinuous. It is computationally expensive because, as a population based search algorithm, it computes many functions at each step. GAs are sometimes not very precise as the design variables have poor resolution. Despite all this, GAs are useful for systems too complex for other methods where gradient information is unavailable. As they are population based, they do not give false positives when reaching local optima, so can be used as a global optimization tool.

Particle Swarm Optimization

Developed by Eberhart and Kennedy in 1995,\textsuperscript{27} particle swarm optimization (PSO) is inspired by birds flocking and fish schooling. Like GA, it is a form of evolutionary algorithm. Potential solutions dubbed “particles” are initiated in the problem space with a random position and velocity. They move through the problem space taking note of the optimization condition - in this case, final fuel mass of the trajectory - at each point. Each particle records its own personal best, and the best solution found by all particles in the swarm is also updated at each point. The velocities of all particles are then altered so that they accelerate towards both the best solution found by all particles and the best solution found by that individual particle. Scaling factors determine the weighting of the acceleration in the direction of each solutions. As they don’t simply travel towards the best solution found by particles in their random initial position, the particles continue to move around and explore better solutions. Any better solutions found update the points towards which all particles accelerate. Eventually, the particles will converge on the optimal solution.

PSO and GA have some key differences.\textsuperscript{17} PSO uses fewer operators than GA, eliminating the need to decide which operator is best at each stage. In GA, the numerical parameters which must be selected in advanced are the population size, crossover and mutation rates, but for PSO it is
population size, and the weightings of the accelerations towards optimal solutions which must be
selected in advance. These are easily tweaked. Manipulating the latter of these can control the rate
of convergence. Overall, PSOs can be a good choice of optimization algorithm as they are easily
implemented and require few parameters.

**Hybrid Techniques**

Global methods such as GAs or PSOs can be used as initial guesses for local methods such as
NLP.\(^\text{17,28}\) Hybridization allows the selection of the advantages of different types of algorithms,
particularly useful in systems sensitive to initial conditions. The global method can search the
entire problem space without sensitivity to an initial guess. The optimized solutions produced by
the global method can then be fed as a good initial guess into a local method to be refined further.

Multiple shooting can be used to feed initial guesses into direct methods. By increasing the level
of complexity in the optimization problem in stages, the robustness of the solution to initial condi-
tions can be improved. This includes the patched manifold methods where escape, interplanetary,
and capture trajectories are independently optimized before being patched together and optimized
again as a complete system.

Evaluating different optimization techniques in this qualitative manner, we consider the best tech-
nique for our system. Considering the above, a hybrid method appears to be the best choice. Given
this is a continuous system rather than an impulsive one, a PSO appears to be the best robust choice
as a global optimizer, producing an optimal solution which can then be fed into an NLP program.
The NLP program is the best choice of local optimizer as they are easy to use with many programs
and support available, are fast, and are accurate. Their only issue is a sensitivity to the initial guess,
which is greatly reduced by providing the NLP with a good initial guess optimized already by a
PSO.

**CONCLUSION**

The space industry in New Zealand is just taking shape, with the help of research at the University
of Auckland. Current progress is being made in the field of small, inexpensive satellites, but at this
erly stage there are possibilities for projects in other areas too. A plan has been made for assessing
the feasibility of the different technologies available. A low energy trajectory will be designed and
optimized using a hybrid PSO/NLP algorithm exploiting the non-linear dynamics around the 1\(^{\text{st}}\)
Lagrange point, namely its stable and unstable manifolds, to assess how far and to which orbits
different propulsion systems can access. A full quantitative comparison of optimization algorithms’
performances on trajectories modelled may be useful in selecting algorithms in future.

**ACKNOWLEDGMENT**

The authors thank Scott Dahlke, Philip Sharp, Félicien Filleul, Nico Reichenbach, and Jan Krecke
for their comments on this work.

**REFERENCES**


