DESIGN SPACE PRUNING HEURISTICS AND GLOBAL OPTIMIZATION METHOD FOR CONCEPTUAL DESIGN OF LOW-THRUST ASTEROID TOUR MISSIONS

A Dissertation
Presented to
The Academic Faculty

by

Kristina Alemany

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Aerospace Engineering

Georgia Institute of Technology
December 2009

DESIGN SPACE PRUNING HEURISTICS AND GLOBAL OPTIMIZATION METHOD FOR CONCEPTUAL DESIGN OF LOW-THRUST ASTEROID TOUR MISSIONS

Approved by:

Dr. Robert Braun, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Ryan Russell
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Panagiotis Tsiotras
School of Aerospace Engineering
Georgia Institute of Technology

Dr. John-Paul Clarke
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Jon Sims
NASA Jet Propulsion Laboratory

Date Approved: October 27, 2009

“The mediocre teacher tells. The good teacher explains. The superior teacher demonstrates. The great teacher inspires.”

-- William Arthur Ward

To all teachers who inspire their students to reach for the stars…

ACKNOWLEDGEMENTS

At times our own light goes out and is rekindled by a spark from another person. Each of us has cause to think with deep gratitude of those who have lighted the flame within us.

-- Albert Schweitzer

First and foremost, I would like to thank God for all the blessings He has bestowed upon me, the greatest of which has been the love and support of family. I would like to express my never-ending gratitude to my parents, who were the first to instill in me a love of learning and a strong work ethic, and have always believed in me and supported me in each and every endeavor I have pursued in my life. You were the first to light the flame within me, and constantly inspire me to pursue my goals and dreams. And to my brother Andrew, whose good-natured teasing about becoming a "doctor" has always made me laugh. Often times, laughter has truly been the best medicine.

I am grateful for the opportunity to have worked with my advisor, Dr. Robert Braun, during my time at Georgia Tech. I am especially thankful for the flexibility he allowed me in choosing not only my dissertation topic, but also in allowing me to finish my dissertation remotely from California. I would also like to express my gratitude to the other members of my thesis committee, who all enthusiastically provided me with helpful advice and insight towards the completion of my dissertation. I would also like to thank Dr. John Olds, who first gave me the opportunity to work in the Space Systems Design Lab at Georgia Tech. It seems like each and every day I apply something I learned from Dr. Olds. He was an inspirational teacher and mentor.

I am thankful for all of my friends that have supported me throughout the dissertation process, particularly my labmates and friends at Georgia Tech. They made sure that no day was dull or boring, even the many late nights and weekends spent in lab. They were always there to laugh with, cry with, and share my achievements with. Thank you for your indispensable help in classes and on my research, and simply for being my friends. I would like to especially thank Ian Clark and Chris Tanner, who devoted many hours to helping me set up my work on the computer cluster, and Richard Otero, without whose Fortran help I would not have been able to complete my dissertation. To Bailey and Griffey, thank you for being my constant companions while working on my dissertation from home in California, and for never failing to put a smile on my face.

Last, and certainly not least, I do not have the words to properly thank my fiancé, Devin Kipp, who has been my unwavering source of support and encouragement, even through the most trying times. You have been endlessly patient, and it is your faith in me that has enabled me to complete the PhD program, beginning with qualifying exams and ending with my PhD dissertation and defense. Having you in my life to share this accomplishment with makes it all the more special.

And to everyone else who has inspired me along the way, thank you. I would not be the person I am today without the influence of my friends, family, teachers, and mentors.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

ANN	Artificial neural network
CSI	Constant specific impulse
DDP	Differential dynamic programming
ENC	Evolutionary neurocontroller
GA	Genetic algorithm
GTOC	Global Trajectory Optimization Competition
GTSP	Generalized traveling salesman problem
LTTT	Low-thrust Trajectory Tools Team
SEP	Solar-electric propulsion
SOI	Sphere-of-influence
TOF	Time of flight
TSP	Traveling salesman problem
VSI	Variable specific impulse
a	Semi-major axis
C	Constraint equation
e	Eccentricity
$\mathrm{E}(XY)$	Expected value of the product of random variables X and Y
g_0	Acceleration of gravity at Earth's surface
$\vec{\mathrm{h}}$	Angular momentum vector
H	Hamiltonian
i	Inclination
I_{sp}	Specific Impulse

J	Objective function
k_i	Percentage of asteroid pairs eliminated in Leg i (for pruning phase)
L	Cost function (integral over entire time domain)
m	Mass
M	Mean anomaly
M_f	Final mass
$\hat{\mathrm{n}}$	Direction vector of line of nodes
N_{GA}	Number of genetic algorithm runs for a single asteroid combination in the branch-and-bound algorithm
P	Power
$\vec{\mathrm{r}}$	Radius vector
T	Thrust magnitude
t	Time
$\vec{\mathrm{u}}$	Control vector
$\vec{\mathrm{v}}$	Velocity vector
V_∞	Hyperbolic excess velocity
$\vec{\mathrm{x}}$	State vector
$\dot{\vec{\mathrm{x}}}$	Derivative of the state vector
$\Delta \mathrm{V}$	Delta-V (change in velocity)
$\Delta \mathrm{V}_{\mathrm{opt}}$	Optimal, two-impulse, phase-free ΔV (for pruning phase)
θ_{wedge}	Angle between the angular momentum vectors of two orbits
λ	Lagrange multiplier
μ	Gravitational constant
μ_{X}	Sample mean of X
$\rho_{\mathrm{X,Y}}$	Correlation coefficient between two random variables, X and Y

σ_X	Sample standard deviation of X
υ	True anomaly
φ	Cost function in terms of final state and time
ω	Argument of periapsis
Ω	Longitude of the ascending node

SUMMARY

Electric propulsion has recently become a viable technology for spacecraft, enabling shorter flight times, fewer required planetary gravity assists, larger payloads, and/or smaller launch vehicles. With the maturation of this technology, however, comes a new set of challenges in the area of trajectory design. Because low-thrust trajectory optimization has historically required long run-times and significant user-manipulation, mission design has relied on expert-based knowledge for selecting departure and arrival dates, times of flight, and/or target bodies and gravitational swing-bys. These choices are generally based on known configurations that have worked well in previous analyses or simply on trial and error. At the conceptual design level, however, the ability to explore the full extent of the design space is imperative to locating the best solutions in terms of mass and/or flight times.

Beginning in 2005, the Global Trajectory Optimization Competition posed a series of difficult mission design problems, all requiring low-thrust propulsion and visiting one or more asteroids. These problems all had large ranges on the continuous variables – launch date, time of flight, and asteroid stay times (when applicable) – as well as being characterized by millions or even billions of possible asteroid sequences. Even with recent advances in low-thrust trajectory optimization, full enumeration of these problems was not possible within the stringent time limits of the competition.

This investigation develops a systematic methodology for determining a broad suite of good solutions to the combinatorial, low-thrust, asteroid tour problem. The target application is for conceptual design, where broad exploration of the design space is critical, with the goal being to rapidly identify a reasonable number of promising solutions for future analysis. The proposed methodology has two steps. The first step applies a three-level heuristic sequence developed from the physics of the problem, which allows for efficient pruning of the design space. The second phase applies a global optimization scheme to locate a broad suite of good solutions to the reduced problem. The global optimization scheme developed combines a novel branch-and-bound algorithm with a genetic algorithm and an industry-standard low-thrust trajectory optimization program to solve for the following design variables: asteroid sequence, launch date, times of flight, and asteroid stay times.

The methodology is developed based on a small sample problem, which is enumerated and solved so that all possible discretized solutions are known. The methodology is then validated by applying it to a larger intermediate sample problem, which also has a known solution. Next, the methodology is applied to several larger combinatorial asteroid rendezvous problems, using previously identified good solutions as validation benchmarks. These problems include the 2nd and 3rd Global Trajectory Optimization Competition problems. The methodology is shown to be capable of achieving a reduction in the number of asteroid sequences of 6-7 orders of magnitude, in terms of the number of sequences that require low-thrust optimization as compared to the number of sequences in the original problem. More than 70% of the previously known good solutions are identified, along with several new solutions that were not previously reported by any of the competitors. Overall, the methodology developed in this investigation provides an organized search technique for the low-thrust mission design of asteroid rendezvous problems.

CHAPTER I

INTRODUCTION

With the recent launches of Deep Space 1, SMART-1, Hayabusa, and Dawn, electric propulsion has become a viable option for solar system exploration.[1,2,3,4] Electric propulsion has the potential to result in shorter flight times, fewer required planetary gravity assists, and/or smaller launch vehicles.[5] One major challenge of low-thrust missions is in the area of trajectory design and optimization. At present, mission design often relies on local optimization of the low-thrust trajectories using expert-based starting points for departure and arrival dates and selection of gravitational swing-bys. These choices are generally based on known configurations that have worked well in previous analyses or simply on trial and error. At the conceptual-design level, however, exploring the full extent of the design space – over a large range of potential launch dates, flight times, and target bodies – is important in order to select the best possible set of solutions for additional higher fidelity analysis. Global optimization is difficult because this design space is often multi-modal and discontinuous. In choosing an analysis technique, there exists an important tradeoff between the accuracy of the results and computing time required. Over the past several years, numerous improvements have been made in the areas of both low-thrust trajectory optimization and the application of global optimization methods to the low-thrust problem.

Missions to asteroids have become a high priority over the past several years. Asteroids are of significant scientific interest because of the possibility of an Earth impact and their connection to the formation of the solar system and potentially to life on Earth. The NEAR mission, for example, which orbited the asteroid 433 Eros, was interested in answering questions related to the nature and origin of near Earth objects, for several reasons.[6] First, asteroids are the primary source of large body collisions with Earth, thereby influencing evolution of the atmosphere and life. Second, asteroids provide clues to the nature of the early solar system processes and conditions, as these are often preserved on small bodies such as asteroids, comets, or meteorites. The near-Earth asteroids are of particular interest because they are believed to contain clues to the nature of the building blocks from which the inner planets were formed. Finally, the NEAR mission was interested in measuring the properties of 433 Eros, in order to establish a connection between meteorites and the history of asteroids, to better quantify the nature of their impact hazard to Earth. NEAR was able to achieve these science goals with the use of a high-thrust propulsion system. The goal of the Dawn mission, which intends to orbit the asteroids Vesta and Ceres, is to better understand the conditions and processes present in the early solar system. Dawn uses low-thrust propulsion, in the form of an ion propulsion system adapted from the Deep Space 1 mission. The propulsion system uses Xenon propellant, and can achieve a maximum thrust level of 92 mN and a maximum specific impulse (I_{sp}) of 3200 s. Dawn launched in September of 2007 and conducted a Mars gravity assist in February of 2009. The spacecraft will arrive at the first asteroid, Vesta, in August, 2011. After a nine month stay at Vesta, the spacecraft will depart for Ceres, and arrive in February, 2015.

The Global Trajectory Optimization Competition (GTOC) was created in 2005 as an example of the types of challenges mission designers face when designing low-thrust trajectories to multiple bodies in the solar system. Since its inception, there have been four editions of competition, all dealing with designing low-thrust trajectories to asteroids. In each competition, entrants were given four weeks to solve the problem. GTOC1 required participants to maximize the change in semi-major axis of the asteroid 2001 TW229 by impacting it with an electric-propelled spacecraft.[7] The spacecraft could employ both thrusting and planetary gravity assists en route to the asteroid, while trying to maximize the following quantity: $J = m_f \left| \vec{U}_{rel} \cdot \vec{v}_{ast} \right|$. In the objective function equation, m_f is the final mass of the spacecraft, $\vec{U}_{rel}$ is the velocity of the spacecraft relative to the asteroid at arrival and $\bar{v}_{ast}$ is the heliocentric velocity of the asteroid. GTOC2 required participants to design a low-thrust trajectory that rendezvous with one asteroid in each of four predefined groups, while maximizing the ratio of final mass to total time of flight.[8,9] For this problem, no gravity assists were allowed. GTOC3 also involved a multiple-asteroid rendezvous mission, but in this case the goal was to design a low-thrust trajectory that would rendezvous with three asteroids out of a single group of 140 and then return to Earth.[10] Gravity assists of Earth were allowed and the objective function was to maximize a weighted combination of mass ratio and the minimum stay time at the three asteroids. Most recently, the GTOC4 problem asked participants to maximize the number of asteroids visited (via a flyby) en route to a rendezvous with a final asteroid, without the use of any gravity assists.[11] There were 1436 candidate asteroids for participants to choose from.

In light of the recent developments in electric propulsion and emerging scientific interest in asteroids, this work will focus on the development of a methodology for solving a multiple-asteroid rendezvous low-thrust mission design problem at the conceptual design level. Two specific types of asteroid rendezvous problems are considered. First is the case of rendezvousing with one asteroid from each of a given number of predetermined groups, as presented in the GTOC2 problem. This type of mission would be relevant if the goal were to visit asteroids with different scientific properties. The second type of problem is to rendezvous with several asteroids out of a single group, such as the Near Earth Asteroids (NEAs). In either case, a spacecraft could return to Earth at the end of the mission duration, which would be representative of a sample return mission. Because the target application is conceptual design, the goal will be to identify a large set of good solutions to a given multiple-asteroid rendezvous mission. Unlike the GTOC competitions, which required only a single best solution to be submitted, the result of the methodology will be a suite of solutions that could then be carried forward into the more detailed design phases, where higher fidelity analysis with additional constraints and objectives could be applied to the problem. In this work, the methodology developed is applied to several multiple asteroid rendezvous problems over a wide range of problem sizes, in order to demonstrate its efficiency at located a family of good conceptual design solutions.

1.1 Low-Thrust Trajectory Optimization Methods and Tools

As aforementioned, one of the challenges in employing electric propulsion comes in the area of mission design. Optimal control theory provides the basis for the low-thrust trajectory optimization used in mission design. The basic optimal control problem, presented in Equations 1 through 3, involves determining the control history (*u*) that minimizes some performance index (*J*). Equation 1 represents the dynamics of the system, written as a set of differential equations, each of which is a function of the state, *x*, the control, *u*, and the time, *t*. Equation 2 represents the cost function, *J*. Here, it is presented in Bolza form, which contains two terms – the first is a function of the final state and time and the second is an integral over the entire time domain. Finally, Equation 3 represents the constraint equation, which can be comprised of control constraints and/or state constraints.

$$\dot{x} = f(x,u,t) \tag{1}$$

$$J = \varphi\left(x\left(t_f\right), t_f\right) + \int_{t_0}^{t_f} L\left(x(t), u(t), t\right) dt \tag{2}$$

$$C\left(x(t), u(t), t\right) = 0 \quad \forall\ t \in \left[t_0, t_f\right] \tag{3}$$

For the low-thrust trajectory optimization problem, the thrust magnitude and direction along the trajectory make up the control history, and the cost function is to maximize the mass at the final state and time (equivalent to minimizing propellant consumption over the entire trajectory), assuming a fixed initial spacecraft mass. The dynamics for this problem are specified in Equation 4, assuming two-body motion. The control is given by Equation 5, and consists of the thrust-direction unit vector, the thrust magnitude, and the power. For a variable specific impulse trajectory, *c*, the exhaust velocity, is a function of the jet power and thrust, as presented in Equation 6. For a constant specific impulse trajectory, such as those used in the GTOC problems, the power is not required as a control variable, and *c* is constant.

$$\dot{X} = \begin{bmatrix} \dot{\bar{r}} \\ \dot{\bar{v}} \\ \dot{m} \end{bmatrix} = \begin{bmatrix} \dot{\bar{v}} \\ -\left(\left(\mu / r^3\right)\bar{r} + \left(T / m\right)\bar{u}\right) \\ -\left(T / c\right) \end{bmatrix} \tag{4}$$

$$\bar{\mathrm{u}}_{\mathrm{c}} = \begin{bmatrix} \bar{\mathrm{u}} \\ \mathrm{T} \\ \mathrm{P} \end{bmatrix} \tag{5}$$

$$\mathrm{P} = \frac{\mathrm{Tc}}{2} \tag{6}$$

For each leg of the trajectory, the spacecraft's initial conditions are determined by the position and velocity of the departure body at a specified time. At rendezvous, the spacecraft must also match the position and velocity of the target body. The final time, t_f, may be fixed or free, depending on the problem formulation. These terminal state constraints are given in Equation 7.

$$C = \begin{bmatrix} \bar{r}_{s/c}\left(t_f\right) - \bar{r}_t\left(t_f\right) \\ \bar{v}_{s/c}\left(t_f\right) - \bar{v}_t\left(t_f\right) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{7}$$

Finally, there are additional constraints on the maximum thrust and power, as specified by the chosen spacecraft and engine parameters.

In general, there are two types of methods for solving the local trajectory optimization problem – direct and indirect.[12,13,14,15] Indirect methods are based on

Pontryagin's Minimum Principle, which minimizes the cost function by minimizing the Hamiltonian, which is given in Equation 8. Furthermore, the costate equations, presented in Equation 9, must be satisfied. This can be also formulated as a maximization problem, depending on the particular problem being solved.

$$H(x,\lambda,u,t) = L(x,u,t) + \lambda^T f(x,u,t) \tag{8}$$

$$\lambda(t) = -\frac{\partial H}{\partial x} \tag{9}$$

Finding a solution to this problem, however, is often difficult because the convergence domain for such problems tends to be small, and is sensitive to the initial guesses of the costate variables (λ), which are not physically intuitive. In order to solve these problems, a homotopy chain is often used, where the solution to a similar problem is known, and that problem is changed slightly and solved with the initial guesses of the known problem in order to step closer to the problem of interest.[16] Therefore, typical indirect methods are difficult to implement within an automated, global optimization program due to the long execution times, small region of convergence, and required user oversight. Additionally, the level of accuracy achieved by indirect methods is generally not required during the conceptual mission design phase.

Direct methods, on the other hand, parameterize the optimal control problem and use nonlinear programming (NLP) techniques to directly optimize the performance index. A variety of direct trajectory optimization methods exist, including numerical integration, collocation, and differential inclusion.[17,18,19,20,21] The number of design variables for direct methods can become very large, and therefore these problems are sometimes limited by available NLP techniques. Additionally, because direct methods require the discretization of a continuous problem, the solution is mathematically sub-optimal, although the accuracy is generally sufficient for use in conceptual design. The main advantages of direct method techniques are their increased computational efficiency and more robust convergence. The solution is generally less sensitive to the initial guesses and those initial guesses are more physically intuitive, which make direct methods preferable for implementing within an automated global optimization scheme.

Differential dynamic programming (DDP) also parameterizes the control variables, providing a large convergence domain and decreasing the sensitivity to poor initial guesses. As compared to direct methods, DDP is less sensitive to the high dimensionality of the low-thrust trajectory optimization problem as it transforms the large problem into a succession of low dimensional sub-problems. Quadratic programming is then used on each resulting quadratic sub-problem to solve for controls that improve the trajectory locally. The states and objective function are then calculated forward in time using the updated controls, and the process is repeated until the problem has converged. One disadvantage of DDP is that it is most effective for smooth unconstrained problems. Low-thrust problems, however, tend to include numerous constraints and can be highly non-smooth. In recent work, Lantoine and Russell have modified the traditional DDP algorithm to create a hybrid differential dynamic programming algorithm that addresses some of the weaknesses of DDP. The hybrid approach uses first- and second-order state transition matrices to calculate the partial derivates required for optimization, and combines DDP with NLP techniques to increase its robustness and efficiency.[222324]

Finally, there also exist hybrid methods which numerically integrate the Euler-Lagrange equations and control the spacecraft based on the primer vector.14[,25] As in the direct method, hybrid methods solve a nonlinear programming problem, but with the Lagrange multipliers making up part of the parameter vector while maximizing or minimizing some cost function. Hybrid methods search numerically for the set of parameters that extremize the cost function, while explicitly satisfying kinematic boundary constraints. According the work by Gao and Kleuver, the advantages of hybrid trajectory optimization methods include a significant reduction in the design space and improved accuracy (as compared to direct methods) with a larger convergence domain and faster problem convergence (as compared to indirect methods).[25]

1.1.1 Improvements to Indirect Methods

Two of the main difficulties with utilizing indirect methods have been the requirement of non-intuitive initial guesses of the costate variables, along with the small region of convergence. For low-thrust trajectory optimization, an adjoint control transformation can be employed to give physical meaning to the initial guesses of the costate variables. A recent example of how this can be applied to a mission design problem is presented by Ranieri at the University of Texas at Austin.[14,26] He replaces the velocity costates with angles that describe the direction of the thrust. These new unknowns have physical significance; therefore, intelligent estimates of their initial guesses can be made. Ranieri applies this technique to solving roundtrip, time-constrained trajectories with I_{sp} constraints and mass discontinuities, for both Mars and Jupiter applications with variable and constant specific impulse engines. Two cases for roundtrip trajectories to Mars are presented in Figure 1, one with constant specific impulse (CSI) and one with variable specific impulse (VSI). For the CSI case, a coast-thrust-coast sequence is assumed for each leg of the trajectory. As can be seen, the CSI trajectory closely approximates the VSI solution.

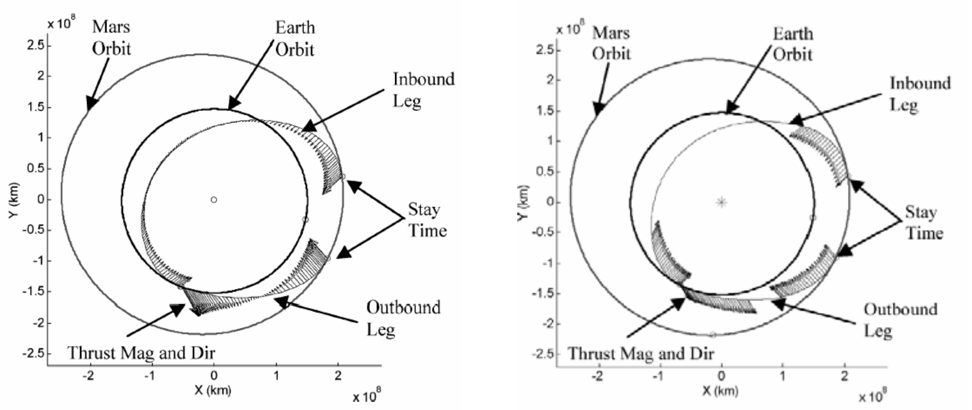

Figure 1: Mars roundtrip trajectory results from Ranieri: variable I_{sp} (left), constant I_{sp} (right).[14]

Another example of applying a costate transformation is presented by Russell.[27] In his work, the unknown initial position and velocity co-states are replaced with more physically meaningful quantities: α and β (the in- and out-of-plane orientation angles, which represent the direction of the initial thrust), *S* (the switching function), and their time derivatives. This transformation is applied to the initial guesses for the targeting routine and then directly iterates on the co-states. This transformation is part of a larger effort, which applies primer vector theory to a global low-thrust trade study algorithm. This methodology is applied to two multiple-revolution problems in the restricted three-body problem: a phase-free transfer between two distant retrograde orbits at Europa and a phase-free transfer from a distant near circular orbit at Earth to a distant retrograde orbit at the Moon.

1.1.2 Improvements to Direct Methods

Sims and Flanagan developed a new direct method, which is implemented in MALTO, a tool intended for the preliminary design of low-thrust trajectories including those with gravity assists.[12,13] As shown in Figure 2, the trajectory is divided into legs that begin and end at control nodes. Typically, these control nodes represent planets or other bodies, but could also represent free points in space. On each leg is a match point, and the trajectory is propagated forward from the previous control node and backward from the subsequent control node to the match point. Each leg is also subdivided into numerous segments containing an impulsive ΔV at the middle of each segment. In the limit, as the number of segments is increased, this approximates the continuous thrust problem. The magnitude of the ΔV is limited by the total amount of ΔV that could be accumulated over the entire segment for the continuous thrust case. Propagation of the trajectory assumes two-body motion, and gravity assists are assumed to cause an instantaneous change in the direction of the V_∞ vector.

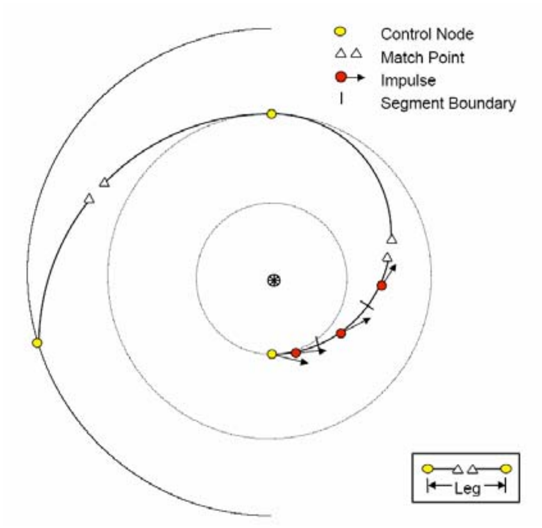

Figure 2: Trajectory structure of the Sims and Flanagan direct method.[12]

This trajectory structure leads to a large, sparse, constrained, nonlinear optimization problem, which is solved using the program SNOPT.[28] At the beginning and ending control nodes, the independent variables include the velocity of the spacecraft relative to the body, the mass of the spacecraft, and the corresponding epoch. At an intermediate body, there are two sets of variables – one at arrival and one at departure – to account for potential changes in velocity for a flyby, changes in mass, or changes in time for a rendezvous. The majority of the independent variables are comprised of the components of the thrust vector on each segment. Additional independent variables can include the reference power of the spacecraft and the specific impulse. Each of these independent variables has associated upper and lower bounds. The primary optimization constraints are that the position, velocity, and mass of the spacecraft must be continuous at the match points. Additionally, the magnitude of the thrust on each segment is constrained by the power available for thrusting. Other constraints can include the mass at the initial control node, the V_∞ vector at departure, at an intermediate body, or at arrival, the time of flight and propellant mass between any two control nodes, and the minimum allowable distance from the Sun.

In the initial paper by Sims and Flanagan, the authors applied their direct method to several different trajectories, verifying their results by comparison to SEPTOP. SEPTOP is a heritage, low-thrust trajectory optimization code that implements an indirect method. It will be described in further detail in Section 1.2.4. Trajectory verification was performed for a flyby of the asteroid Vesta with a Mars gravity assist, a rendezvous with the comet Tempel-1, and a flyby of Pluto with two Venus gravity assists and one Jupiter gravity assist. With their direct method, even simple initial guesses for thrust direction and magnitude worked well in arriving at the solution. For the initial guess, they assume that the thrust varies linearly between nodes, with the direction at the nodes being perpendicular to the radius vector at that point. The solutions for the three reference missions compared well to those obtained using SEPTOP. For the Vesta and Tempel 1 trajectories, SEPTOP had difficulty converging for some of the cases, while the Sims and Flanagan method converged readily. Furthermore, SEPTOP could handle at most two intermediate flybys, so the Earth-Venus-Venus-Jupiter-Pluto trajectory had to be broken into two trajectories in SEPTOP. Using the Sims and Flanagan method, any number of intermediate bodies can be analyzed. They do note that for more complicated trajectories, however, the optimization does not always converge with the initial starting conditions, so a fair amount of user manipulation is still required to arrive at a converged solution.

In addition to the reference missions analyzed by Sims and Flanagan, several papers by investigators at Purdue University include results using GALLOP, a trajectory optimization tool developed at Purdue based on the Sims and Flanagan direct method.[29,30] These additional trajectories include a rendezvous with Ceres via Mars, an Earth-Venus-Earth-Mars-Jupiter trajectory, an Earth-Venus-Jupiter trajectory, an Earth-Mars-Jupiter trajectory, and an Earth-Earth-Mars-Jupiter trajectory. These solutions helped to further validate the method as well as demonstrate its ability to handle a number of different flyby problems with numerous intermediate bodies. In the Purdue studies, the initial guesses were generated using a shape-based analytic method (described in Section 1.2.3).

Building on the Sims and Flanagan method, Yam at Purdue University explored different approaches to parameterize the ΔV in an effort to decrease run time[31,32]. The optimization variables for the N-vector formulation, which was used in the initial Sims and Flanagan model, consist of the ΔV components on each segment. The node formulation, suggested by Yam, replaces the ΔV magnitudes with a set of on/off nodes that define the switching point from null-thrust to maximum-thrust and vice versa. In the Chebyshev formulation, the ΔV angles are modeled as a Chebyshev series, with the optimization variables consisting of the coefficients of the Chebyshev series. A Chebyshev series of degree k can be defined as follows, where $T_k(u)$ is the Chebyshev polynomial of degree k, u is the independent variable of the Chebyshev polynomial, and c_i are the coefficients of the series.

$$c_0 T_0(u) + c_1 T_1(u) + \cdots + c_k T_k(u) \tag{10}$$

Finally, the Node + Chebyshev formulation uses on/off nodes to parameterize the ΔV magnitude and a Chebyshev series to model the ΔV angles.

In Ref. 32, four different case studies are examined to determine the performance of each of the four ΔV parameterizations. These case studies include a simple Earth-Jupiter rendezvous mission, a flyby of the asteroid Vesta with a Mars gravity assist, an Earth-Mars cycler mission, and an Earth-Mercury rendezvous. Based on the performance of each ΔV parameterization method, it is clear that the best formulation is problem dependent, although time savings can be realized over the original N-Vector formulation developed by Sims and Flanagan. The Node + Chebyshev formulation tended to have the fastest run times for the largest range of problems; however, it did have problems with convergence in some instances. For large problems, the tolerances had to be relaxed in order for the Node + Chebyshev formulation to arrive at a solution. The Node + Chebyshev method appears to be the most beneficial for searching broad areas of the design space. On the other hand, the N-Vector formulation is the most stable, although it was not always the fastest approach, and in some cases, it was significantly slower. The N-Vector formulation is therefore a good standard method when only a small number of cases need to be performed.

1.1.3 Analytic, Shape-Based Methods

Indirect and direct methods tend to be computationally intensive because the trajectory must be numerically integrated or propagated. An analytic method, on the other hand, has the potential to significantly reduce run times by eliminating the need for numerical integration and instead solving for an analytic solution to the equations of motion.

Petropoulos, at Purdue University, developed a shape-based method intended for quickly searching a broad design space and generating initial guesses to then be used in a more accurate trajectory optimization program.[33,34,35] This method assumes that the spacecraft trajectory follows a predetermined shape, from which the thrust profile can be determined. With the correct choice of shape, there exists an analytic solution to the equations of motion. The motion of the spacecraft between planets can either be purely conic (coasting) or involve thrusting. Each leg can be characterized as thrust, thrust-coast, or coast-thrust. For the thrusting segments, the in-plane motion of the spacecraft is assumed to follow an exponential sinusoid shape, given by Equation 11, where k_0, k_1, k_2, and ϕ are all constants that define the shape of the trajectory:

$$r = k_0 e^{k_1 \sin(k_2\theta+\phi)} \tag{11}$$

Gravity assists are modeled as instantaneous changes in the heliocentric spacecraft velocity (with no change in position). Out-of-plane motion is based on an analysis of the orbital angular momentum vector, where the out-of-plane angle and speed are approximated by the in-plane angular momentum and velocity components.

This method has been applied to a number of different trajectories, one of which was a rendezvous with the asteroid Ceres with an intermediate flyby of Mars, assuming thrust-only legs.[29,30,33,35] A search was done for departure dates ranging from 1990 to 2049 with launch V_∞ between 0.75 km/s and 2 km/s. Figure 3 (left) plots the resulting arrival V_∞ for each of the cases analyzed. This broad search allows mission designers to choose the best points to examine further with higher-fidelity trajectory optimization methods. In this study, the best point from the shape-based analysis was then used as an initial guess for GALLOP. The result had good agreement with an optimal solution presented by Sauer in an earlier study. Another trajectory analyzed by Petropoulos was Earth-Venus-Earth-Mars-Jupiter (EVEMJ). A sweep of departure dates from 1975 to 2049 was analyzed, with an increment of 10 days. Additionally, values of launch V_∞ between 0.5 km/s and 2 km/s were considered. For this case, the in-plane propellant mass fraction was the parameter of interest, which is plotted in Figure 3 (right). As before, the best trajectory from this broad design space exploration was used as an initial guess in GALLOP in order to optimize the solution.

A shape-based method was also applied by the winning team at the 2005 1st Global Trajectory Optimisation Competition.[36] The objective of the optimization problem was to maximize the change in the semi-major axis of asteroid 2001 TW229 after impacting it with a spacecraft employing low-thrust propulsion. The initial mass of the spacecraft was given, along with the thruster's I_{sp} and maximum thrust level. Additionally, a launch date window of 20 years was given, with a maximum time of flight of 30 years.[37] In approaching this problem, the winning JPL team took a two-step approach.[38] First, they searched over a large range of the solution space using a shape-based method, then honed in on the most promising portion with a more accurate local optimization method. The JPL team considered 15 different gravity assist combinations, and then conducted a grid search for each combination over launch date and launch V_∞ values using the shape-based method. The best solutions from the grid search (high values of arrival V_∞) were then passed on to MALTO to examine in more detail.

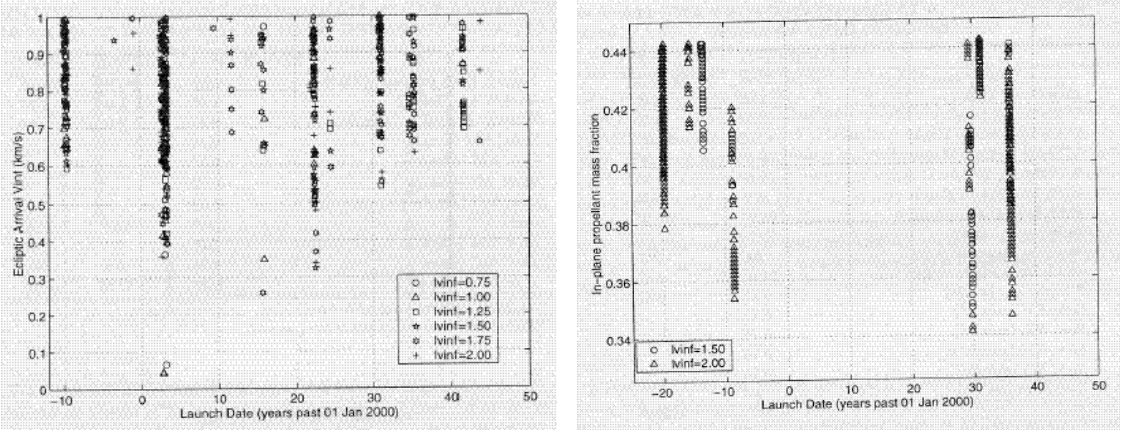

Figure 3: Results of shape-based method design space exploration for Earth-Mars-Ceres (left) and EVEMJ (right).[33]

1.1.4 Low-Thrust Trajectory Optimization Tools

There are a wide variety of available tools for low-thrust trajectory optimization, many based on the methods described above. In 2002, NASA established the Low-Thrust Trajectory Tools Team (LTTT) to improve the agency's low-thrust trajectory analysis capability and to create a common set of low-thrust trajectory tools.[39,40] Under the effort, five new tools were developed, and 32 reference missions were identified that would be relevant to future NASA missions and would test the capabilities of these new tools. The reference missions include missions with multiple gravity assists as well as flybys of and rendezvous with comets and asteroids. In general, the new tools are of higher fidelity, easier to learn and use, and can analyze a broader range of missions than the previously existing set of tools.

Prior to the LTTT effort, the primary low-thrust trajectory analysis tools for most of NASA's preliminary design studies were CHEBYTOP, VARITOP, SEPTOP, and SAIL.[34] CHEBYTOP uses Chebychev polynomials to represent state variables, which are then differentiated and integrated in closed form to solve a variable-thrust trajectory. This solution can then be used to approximate a constant thrust trajectory. While it is considered a low-fidelity program, it is highly valued for its ability to rapidly assess large trade spaces. It cannot, however, analyze multi-leg missions and is limited to the heliocentric sphere of influence. VARITOP, SEPTOP, and SAIL all use calculus of variations in the formulation of the state and co-state equations, which are integrated numerically to solve the two-point boundary value problem. The programs differ in their solar electric propulsion, nuclear electric propulsion, and solar sail models. These tools can also only handle heliocentric trajectories, and are considered to be medium-fidelity.

The tools developed under the LTTT effort are all considered to be medium- to high-fidelity trajectory tools.[34] MALTO was developed at JPL based on the method by Sims and Flanagan described in Section 1.2.2. It is considered to be medium fidelity. This tool has been used for numerous trajectory design studies, including the trajectories for the Jupiter Icy Moons Orbiter. The remaining LTTT tools are all considered to be high-fidelity. COPERNICUS, developed at the University of Texas at Austin, is an n-body tool with a high degree of flexibility. The user can model a number of different missions, with varying gravitational bodies, objective functions, optimization variables, constraint options, and levels of fidelity. Additionally, it can model multiple spacecraft, as well as optimize for both constant and variable specific impulse trajectories. COPERNICUS employs multiple shooting and direct integration for targeting and state propagation.[41] Mystic was developed by Greg Whiffen at JPL, and implements Static/Dynamic Optimal Control (SDC), which was developed by Whiffen. SDC is a nonlinear optimal control method designed to optimize both static variables and dynamic variables (functions of time) simultaneously.[42] The program is robust enough to take advantage of gravity assists if a flyby body is near the reference trajectory. Mystic was used to design the Dawn trajectory, and after being flight qualified, is expected to be used to validate the other tools. OTIS 4.0 is an upgraded version of the program originally developed by NASA Glenn Research Center and Boeing for launch vehicle trajectory analysis.[43,44] This tool employs a direct method for low-thrust trajectory optimization, using nonlinear programming techniques to solve the implicit integration problem. SNAP, developed at NASA Glenn Research Center, is the final tool developed under the LTTT effort. SNAP's distinguishing feature is its ability to propagate planet-centered trajectories, including aspects such as atmospheric drag, shadowing, and higher-order gravity models. It does not, however, contain an optimizer.

With the exception of SNAP, the various tools described above were compared for a number of different low-thrust mission scenarios. Ref. 39 provides an overview of five of the 32 reference missions examined, and compares in detail the results of the various tools. In general, it was found that the low, medium, and high fidelity tools arrived at very similar answers when their input assumptions were consistent. The high fidelity tools do not necessarily provide significant improvements in accuracy, but are able to model more complex missions. Low fidelity tools, on the other hand, have the advantage of faster execution times, rapid trade study analysis, and are often much easier to learn and implement.

In addition to the LTTT tools, several recent university-developed tools have been created for low-thrust trajectory optimization. Petropoulos at Purdue University incorporated a low-thrust gravity assist capability to STOUR (Satellite Tour Design Program) to create STOUR-LTGA (Satellite Tour Design Program – Low Thrust, Gravity Assist), which automatically searches for gravity-assist trajectories. In this program, the user specifies a sequence of gravity-assist bodies, a range of launch dates, a range on launch V_∞, and constraints on various parameters, such as time of flight and propellant consumption. STOUR-LTGA employs a shape-based method to approximate the shape of the trajectory and analytically solve the equations of motion, as described in Section 1.2.3. Also developed at Purdue University, GALLOP implements the direct method formulated by Sims and Flanagan, originally found in MALTO.[31,32] Note that while COPERNICUS was developed under the LTTT program, it was also developed by university researchers.

1.2 Global Optimization Methods and Applications

Many of the tools described above not only implement a trajectory optimization method for finding the optimal control history of the spacecraft (thrust magnitude and direction), but also include some ability to optimize for other parameters such as launch date or arrival date. Because the design space is multi-modal with respect to launch date, a gradient-based optimizer cannot guarantee convergence to the global optimum. The optimizer typically converges to the local minimum closest to the given initial guesses. If a broad search space is desired, such as in the case of the STOUR-LTGA examples, a domain-spanning, global optimization method is required.

1.2.1 Evolutionary Algorithms

One of the most well known types of global optimization methods are evolutionary algorithms, which are domain spanning, probabilistic optimization algorithms based on the Darwinian theory of evolution.[45] One of the more well known of these evolutionary algorithms is the genetic algorithm (GA).[46,47,48] Although there are numerous variations, the simple genetic algorithm begins with a random initial population, which is made up of a set of individuals. Each individual in the population represents a single value for each of the design variables. This generally results in a random scatter of points over the design space. Each set of design variables is referred to as a chromosome and is typically encoded as a binary string, which must be mapped to the real values of the variables. The design variables are discretized between their lower and upper bounds. In each generation, the population is subjected to certain genetic operators such that the population will “evolve” and improve its fitness (objective function). The typical genetic operators are reproduction, crossover, and mutation. The purpose of reproduction is to weed out the members of the population with low fitness values, and to keep those with high fitness values. Crossover combines two “parents” by switching parts of their chromosome strings with each other to create two “children”. Mutation is responsible for switching individual bits in a chromosome string. Because there is no necessary condition for optimality, the convergence criteria is generally chosen either as a maximum number of generations (iterations) or a certain number of generations with no change in the objective function and/or design variables. As the generations progress, there should be a steady improvement in both the average fitness of the population as well as the fitness of the best member. In general, at the termination of the GA, the population will be clustered around the global optimum.

One of the main advantages of genetic algorithms is their ability to find a global optimum in a discrete, multi-modal design space. They can also handle a large number of variables, and require no initial guesses for the design variables. Genetic algorithms, however, do have some disadvantages. Because of the probabilistic nature of the algorithm, there is no guarantee that the optimal solution will be found. Therefore, the GA must generally be run more than once to ensure optimality. Genetic algorithms also require a large number of iterations, and therefore function calls, in comparison to a gradient-based method. Finally, if the original design space is comprised of continuous design variables, the discretized solution will generally not correspond to the precise global optimum. A common practice is to use the solution obtained by the GA as an initial guess to a gradient-based optimizer, in order to improve the accuracy of the solution.

Several variations on the standard genetic algorithm have also been developed. In order to locate multiple local optima, a sharing function can be added to the GA, which draws upon the theory of niche and speciation in Darwinian evolution.[49] The purpose of the sharing function is to degrade an individual's fitness function based on its proximity to neighboring individuals. As a result, the largest number of individuals will converge to the local optimum with the best fitness value, with fewer individuals converging to optima with lesser fitness. The number of optima found by a genetic algorithm with sharing is a function of the size of the population. Another genetic algorithm variant addresses multi-objective problems.[50] One way of dealing with multiple objectives is to solve for the Pareto-optimal set, which encompasses the set of non-dominated solutions. When comparing two solutions, $\mathbf{x_1}$ and $\mathbf{x_2}$, $\mathbf{x_1}$ dominates $\mathbf{x_2}$ if (1) $\mathbf{x_1}$ is no worse than $\mathbf{x_2}$ in all objectives and (2) $\mathbf{x_1}$ is strictly better than $\mathbf{x_2}$ in at least one objective. Therefore, the Pareto-optimal set contains all the solutions that are not dominated by any other solutions. This concept has been implemented in NSGA-II (non-dominated sorting genetic algorithm), developed by the Kanpur Genetic Algorithm Laboratory.[51]

Genetic algorithms have been applied to a number of different trajectory optimization problems, beginning with their application to ballistic (high thrust) transfers and gravity assist problems.[52,53,54,55] For the high-thrust case, solving for a single trajectory is much less time-consuming and is generally done using a Lambert Solver. Therefore, a genetic algorithm, even with the large number of required function calls, is appropriate for global optimization.

Gage and Braun applied a genetic algorithm with a sharing function to impulsive Earth-Mars trajectories in order to optimize for launch date and time of flight. Figure 4 plots ΔV as a function of departure date and transfer time. As can be seen, this is a multi-modal space, and for conceptual design, it is desirable to locate each of the local minima, which was successfully accomplished using a sharing function.

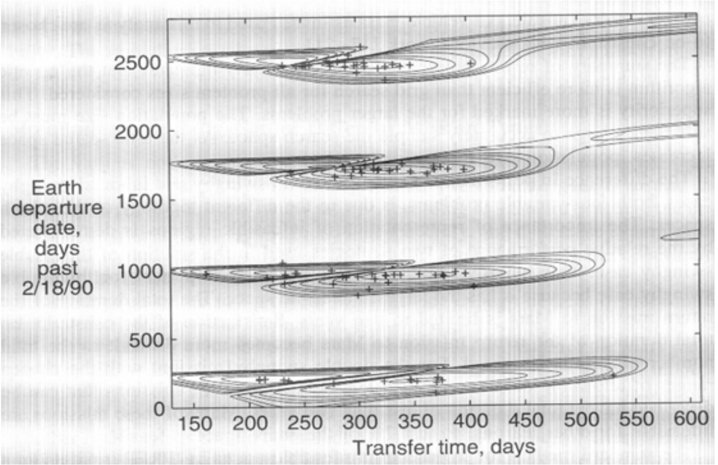

Figure 4: Results of a genetic algorithm with a sharing function applied to an Earth-Mars impulsive transfer.[54]

Several studies have also attempted to apply genetic algorithms to solve for the optimal control parameters of the low-thrust problem.[56,57] This approach, however, has not shown any benefit over direct or indirect methods for trajectory optimization, again because of the large number of function calls required by the GA.

More recently, several authors have attempted to apply the genetic algorithm to selecting the global parameters of the optimization problem, combined with a direct or indirect method for solving for the optimal control history of the spacecraft.[58,59,60] De Pascale proposes a method for combining a genetic algorithm with an analytic shape-based method to optimize low-thrust gravity-assist trajectories. The trajectory is divided into sub-arcs, which are chosen to be either coast arcs or low-thrust arcs. The two-point boundary value problem for the coast arcs is solved with a Lambert solver, while the low-thrust arcs are solved using a shape-based method based on the work by Petropoulos. De Pascale uses an exponential trigonometric shape to analytically solve the equations of motion. Gravity assists are modeled as instantaneous changes in the heliocentric velocity. The genetic algorithm is used in conjunction with a static penalty function, in order to handle constraints. The full set of design variables includes the departure V_∞, the right ascension and declination at launch, the velocities at each of the encounter bodies, the sequence of encounter bodies, the pericenter-radius for each flyby, and the number of revolutions around the Sun for each phase.

De Pascale applies this method to several different trajectories. First, a simple low-thrust transfer to Mars is examined. The solutions obtained match very closely to existing optimal solutions for this problem. Ballistic (two-impulse) missions to Jupiter were then examined, using the full set of design variables, so that the gravity assist sequence was not predetermined. Several promising trajectory paths resulted: EVEEJ, EMMJ, and EVVEJ. When low-thrust trajectories to Jupiter were considered, however, the author did not use the full set of design variables, but instead optimized the trajectory for predetermined sequences of gravity assists (EVJ, EVVJ, and EMMJ). It was not clear if the method had failed for the full set of design variables in the low-thrust case or if it had not been attempted.

Woo, Coverstone, and Cupples proposed a method combining a genetic algorithm with SEPTOP, which uses an indirect method for solving the optimal control problem. One of the key features of this work is the procedure for reducing the size of the parameter space before applying the GA/SEPTOP hybrid method. Trajectories previously generated by SEPTOP are used to limit the size of the design space through a number of different methods: R-ratio analysis, delivered mass estimation, thruster modeling, ballistic approximation, and phase calculation. The genetic algorithm is then used to search the reduced parameter space, which generates inputs to run SEPTOP. SEPTOP returns the convergence error to the genetic algorithm as a measure of the fitness of the initial input. Results are generated for a series of outer-planet missions with a single Venus gravity assist. In previous work, this hybrid procedure was also successfully applied to the design of a trajectory for a sample return to the comet Tempel 1. For this problem, however, the reduction of the parameter space could not be applied because there were no previously generated trajectories.

Vavrina and Howell at Purdue University combine a genetic algorithm with GALLOP[61], the low-thrust trajectory optimization code described in Section 1.2.4, which is based on the direct method by Sims and Flanagan. The design variables controlled by the genetic algorithm are as follows: time at departure, $\vec{V}_\infty$ at departure, initial mass, time of arrival and departure at any intermediate bodies, $\vec{V}_\infty$ at arrival and departure of any intermediate bodies, time at arrival of the final body, $\vec{V}_\infty$ at arrival of the final body, and the thrust vector on each segment. The objective function is to maximize the final mass at the arrival body. Because of the large number of variables required to represent the thrust vectors, the thrust is represented in spherical coordinates, with the two thrust angles modeled using a Chebyshev series and the thrust magnitude modeled using an on/off formulation. GALLOP also has control to locally optimize the design variables it has been passed. Vavrina and Howell use a combination of two inheritance schemes – Lamarckian and Baldwinian. In Baldwinian inheritance, the resulting locally optimized fitness function is assigned to the original design variables passed to GALLOP by the GA. In Lamarckian inheritance, on the other hand, the new values of the locally optimized design variables replace the original design variables in the GA population.

Vavrina and Howell apply their hybrid technique to a number of low-thrust trajectories: Earth – Mars rendezvous, Earth – Venus – Earth – Jupiter – Pluto rendezvous, Earth – Mars – Earth – Jupiter – Pluto rendezvous, and Earth – Venus – Earth – Jupiter – Neptune rendezvous. For each of the multiple gravity-assist trajectories, the genetic algorithm required 171 design variables, and a population size of 200 was chosen. The GA was run for 150 generations, requiring 30,000 GALLOP evaluations. After the 150 generations, the GA had yet to converge on a single solution for any of the trajectories, but clear bands of feasible solutions had emerged. While the hybrid method did require a large number of GALLOP evaluations, the best final mass found for each trajectory matched or exceeded the best solutions presented in the literature.

1.2.2 Evolutionary Neurocontrollers

Evolutionary algorithms can also been combined with artificial neural networks (ANNs), to create an Evolutionary Neurocontroller (ENC). These have recently been applied to trajectory optimization problems by Dachwald at the Institute of Space Simulation in Germany.[62,63,64] Artificial neural networks are inspired by information processing in animal nervous systems, in that they will learn from experience, generalize previous examples to new ones, and extract essential information from noisy input data. ANNs are composed of processing elements called neurons that are organized into neuron layers. Figure 5 illustrates an example of a feedforward ANN, with a layered topology and three layers.

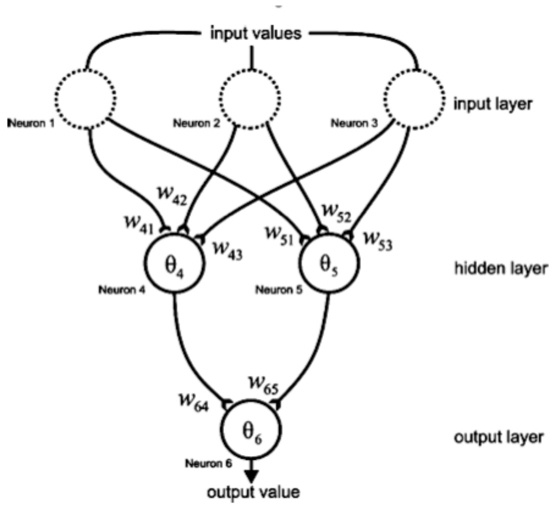

Figure 5: Example of a layered, feed-forward neural network.[62]

Depending on the function used for the neurons, a neural network can be regarded as a continuous parameterized function, called a network function, which simply maps a set of inputs to a set of outputs. If a training set exists – the correct output for a set of given inputs – then the network error can be measured and used to learn the optimal network function. If a training set does not exist, then it becomes a reinforcement learning problem, where the optimal behavior must be learned through interaction with the environment. For an evolutionary neurocontroller, an evolutionary algorithm is used to find the neurocontroller's optimal network function.

Dachwald's work applies an evolutionary neurocontroller to solar sail trajectories, which have very low thrust magnitudes, thereby exhibiting solutions with many revolutions around the Sun. Furthermore, the objective function is generally to minimize the time of flight since there is no propulsion required for a solar sail. More recently, however, Dachwald applied his method to solar electric propulsion (SEP) spacecraft as well. In his formulation, a trajectory is the result of a spacecraft steering strategy that controls the spacecraft's thrust vector according to the current state of the spacecraft relative to the target. An artificial neural network is then used to implement the spacecraft steering strategy, with the evolutionary algorithm used to optimize the neurocontroller parameters. Figure 6 illustrates how such a formulation works for the SEP trajectory. The neural network pictured below illustrates how the inputs for a SEP trajectory are mapped to outputs, as per Dachwald's formulation. Here, the inputs represent the difference in the spacecraft's state and its target at any point along the trajectory. The output then corresponds to the control parameters that will result in the spacecraft meeting its target constraints at the specified final time.

In his example case, Dachwald utilized the evolutionary neurocontroller to optimize the launch date in addition to the spacecraft steering strategy. He did not, however, consider problems with multiple legs or encounter bodies. The evolutionary neurocontroller was applied to a Mercury rendezvous and a near-Earth asteroid rendezvous, and compared to similar problems in the literature. Dachwald's method was able to locate solutions better than those presented in the literature, due to its ability to search a large portion of the design space.

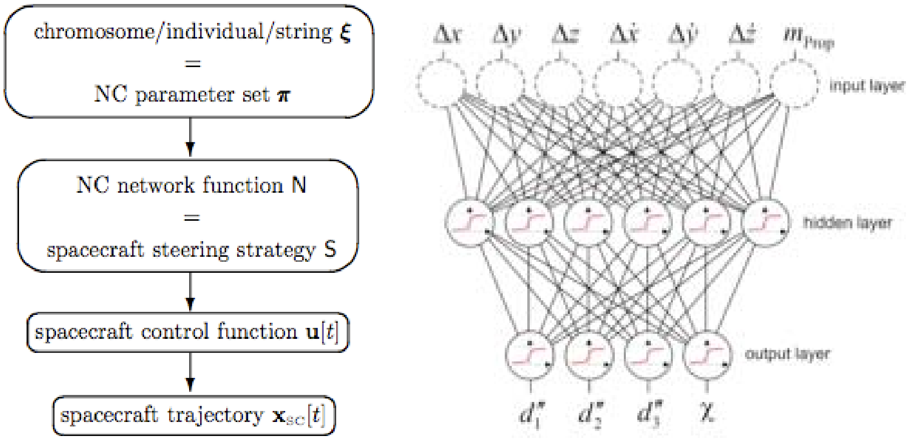
chromosome/individual/string ξ
=
NC parameter set $\boldsymbol{\pi}$
NC network function N
=
spacecraft steering strategy S
spacecraft control function $\mathbf{u}[t]$
spacecraft trajectory $\mathbf{x}_{SC}[t]$
Δx
Δy
Δz
$\Delta \dot{x}$
$\Delta \dot{y}$
$\Delta \dot{z}$
m_{Prop}
input layer
hidden layer
output layer
d_1^*
d_2^*
d_3^*
χ

Figure 6: Converting an evolutionary algorithm chromosome into a spacecraft trajectory (left); example neurocontroller that implements a spacecraft trajectory (right).[63]

Carnelli later extended Dachwald's method to include low-thrust trajectories with gravity assists.. An evolutionary neurocontroller is combined with a steepest descent method used to optimize the gravity assist maneuvers. As before, the ENC searches for the optimal parameter set (steering strategy) that forces the spacecraft's state from its initial state to the target body's final state, along a trajectory that obeys the dynamic constraints and terminal constraints, while maximizing some cost function and potentially crossing the sphere of influence (SOI) of an assisting body. Instead of choosing some sequence of gravitational assists a priori, the ENC is freely allowed to choose the spacecraft controls, and a gravity assist is performed only if that steering strategy takes the trajectory through the SOI of some intermediate planet. Because the relative size of each planet's SOI is very small in comparison to the scale of the overall trajectory, their size had to be inflated. Otherwise, the ENC would be very unlikely to ever find a gravity assist trajectory. When the chosen steering strategy does take the spacecraft within a planet's SOI, a steepest-descent algorithm is used to determine the optimal pointing distance for the gravity assist maneuver. Making these modifications allowed Carnelli to successfully apply this method to a Pluto flyby trajectory via Jupiter and a Mercury rendezvous via Venus.

1.2.3 Combinatorial Optimization

Choosing the optimal asteroid sequence in an asteroid tour mission design problem is by nature a combinatorial and integer optimization problem. The distinguishing feature of such problems is that the variables belong to a discrete set where there is not a continuum of alternatives. One can write the linear integer programming problem, which has no continuous variables, as follows[65]:

$$\max\{cx : Ax \leq b, x \in Z_{+}^{n}\} \tag{12}$$

In Equation 12, Z_{+}^{n} is the set of nonnegative integral n-dimensional vectors, and $x = (x_1,\ldots,x_n)$ are the variables or unknowns. An instance of the problem is specified by the data (c, A, b), where c is an n-dimensional vector, A is an m x n matrix, and b is an m-dimensional vector. While this represents the general problem, it can be altered in a number of ways to represent myriad types of discrete optimization problems. One example of a formulation of Equation 12 is the 0-1 knapsack problem, which is one of the most basic and common problems in combinatorial optimization. This problem deals with choosing a subset of projects to maximize the sum of their values while not exceeding some budget constraint. From Equation 12, the projects are represented by the variables $x_1,\ldots x_n$, where a value of 1 indicates that project j is in the subset and a value of 0 indicates that it is not. The jth project has a cost of a_j and a value of c_j, and b represents the overall budget constraint.

Another common type of combinatorial optimization problem, which is more directly applicable to the asteroid tour problem, is the traveling salesman problem. The classic version of this problem deals with solving the following scenario: given a set of cities and the distance between each pair of cities, determine the shortest route to visit each city exactly once while returning to the city of origin.[66] Solution methods generally fall into two categories: algorithms for finding exact solutions and heuristic algorithms, which find good solutions but can not be proven to be optimal. Exact solutions can typically be implemented successfully only for small problems, while heuristic methods are used for larger problems where computation time of an exact method would become prohibitive.

A commonly used exact algorithm is branch-and-bound, which branches the original problem into successively smaller sub-problems. Each subset contains a relaxed version of the original problem, which is easier to solve. The procedure continues until each branch has resulted in either a feasible solution or is shown to contain no solution better than one already obtained. Branch-and-bound methods result in locating the global optimum. Similarly, dynamic programming also takes advantage of problem decomposition, where the optimal solution to a given problem is expressed in terms of optimal solutions of smaller sub-problems.[67]

One of the most commonly known heuristic algorithms is the nearest neighbor algorithm, also referred to as the greedy algorithm. In this method, the local optimum is chosen at each step. For the classic TSP, for example, this would equate to choosing the closest city at each step, until all of the cities have been visited. Another common heuristic method is to use minimum spanning trees. A spanning tree is a collection of *(n-1)* edges which join all *n* cities into a tree-structure. This can then be extrapolated to create a tour, where each city is only visited once. While the heuristic methods do not solve for the optimum solution, they can at least provide lower and upper bounds on the optimum. However, one of the biggest challenges of heuristic methods is establishing performance guarantees – i.e., bounds on how far the solution will be from the optimum in the worst case.

The classic TSP has many analogous features to the asteroid rendezvous problem, where the “distance” between each asteroid is instead a combination of propellant consumption and time of flight. Some major differences do exist, however, between the classic TSP and the asteroid rendezvous problem. First, the asteroid problem does not require visiting every asteroid; only one asteroid must be visited in each defined group. Furthermore, the spacecraft does not have to return to Earth, the point of origin. More importantly, the cost function ("distance") between each pair of asteroids is not known a priori; instead, it changes with time and is not easily computable. For each instance in time, for example, calculating the cost to go from asteroid *i* to asteroid *i+1* requires solving a low-thrust trajectory optimization problem to determine the optimal thrust profile that minimizes propellant expenditure for the given time of flight.

Several variations on the classic traveling salesman problem have been studied, which address some of the asteroid tour problem complexities. In the time-dependent (or moving-target) TSP, the cost of traveling from city *i* to city *j* changes as a function of time. Work has been done on developing reliable heuristic methods with provable performance bounds for restricted versions of the time-dependent problem, such as where each target moves with a constant speed and direction and the pursuer has a maximum speed greater than the speed of each of the targets.[68] In the Generalized Traveling Salesman Problem (GTSP), all of the targets are partitioned into clusters, and the problem is transformed into finding the shortest route while visiting at least one target in each cluster. Another version of the GTSP requires that exactly one target must be visited in each cluster.[69,70] Finally, the wandering salesman problem, also known as the messenger problem, does not require returning to the point of origin, but instead deals with finding the least cost route from *u* to *v*.[70]

Several authors have formulated trajectory optimization problems as traveling salesman problems. Stodgell and Spencer posed the problem of autonomous satellite servicing as a multi-objective wandering salesman problem with dynamically moving vertices.[71] Their specific problem can be defined as follows: given a set of target satellites, find tours that visit each target exactly once, while rendezvousing with the target for some minimum stay time and minimizing total flight time and ΔV. In order to solve this problem, NSGA-II was used in order to deal with the multiple objective functions. In this problem, only impulsive maneuvers were considered, and a multi-revolution Lambert algorithm was used to compute the required ΔV to transfer between two orbits for a specified time of flight. The genetic algorithm was then used to determine the following global design variables: tour order, initial departure time, flight time between each target satellite, and stay time at each target. This formulation was applied to several different target satellite configurations, with up to six target satellites. Figure 7 plots the resulting Pareto frontier for six target satellites in six different orbit planes separated by one degree each. For this test case, the optimal tour order and corresponding time variables were successfully found by the genetic algorithm for all three random initial populations considered. For some of the other test cases, however, the genetic algorithm would prematurely converge to a sub-optimal tour order depending on the initial population.

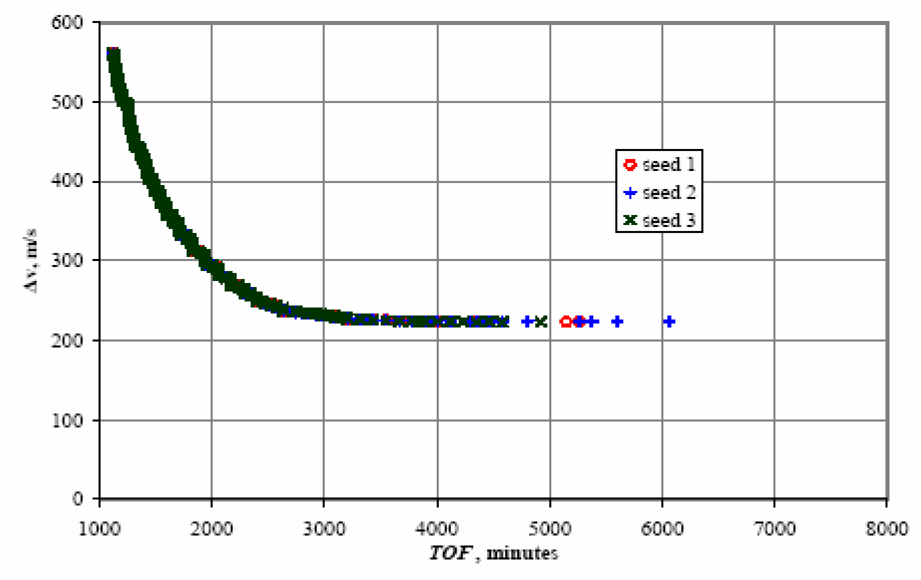

Figure 7: Pareto frontier for satellite rendezvous problem with six targets.[71]

Wall and Conway posed the optimal control problem as a motorized traveling salesman problem, where the salesman drives a car with two bounded controls: steering angle velocity and acceleration.[72] As in the original TSP, the salesman must still visit each city once and then return to the origin. The goal of the motorized TSP is to determine the control history of the steering angle and acceleration to minimize the total travel time. The control history was discretized, and a genetic algorithm was used to determine the optimal solution of the resulting control parameters. This method was then extended to apply to several different low-thrust orbit transfer problems, including an Earth-Mars transfer, a super-synchronous to geosynchronous orbit transfer, and a circle-to-circle transfer. The control parameters were parameterized using a variety of methods, and a genetic algorithm was again used to solve for the control parameters. Finally, this methodology was applied to an asteroid interception problem, where the spacecraft departs from Earth and must visit three asteroids out of a population of eight, this time using impulsive maneuvers. The objective function was to minimize the ΔV. A branch-and-bound method was used as the outer-loop solver to determine the optimal asteroid sequence, while a genetic algorithm was used to solve for the transfer times.

The variants on the classic traveling-salesman problem described above – time-dependent TSP, generalized TSP, and wandering TSP – all model the various aspects of the asteroid problem. In developing solution techniques for these variants, however, each type of TSP has been addressed separately. No exact algorithm has been developed for a solving a problem that includes all three of these variants. Furthermore, the cost function between each pair of targets requires only a simple distance calculation. Even the exact algorithms for solving the time-dependent TSP are good only for very restricted cases, where the cost function is still easily calculated. The papers by Stodgell and Spencer and by Wall and Conway solve a combinatorial rendezvous problem that has been formulated as a traveling-salesman problem. In both papers, genetic algorithms are successfully used to solve for the optimal target sequence as well as the other system-level variables such as departure dates and flight times. Both of these studies solve problems on a much smaller scale than the asteroid tour problem being solved in this thesis.

As aforementioned, branch-and-bound is one type of algorithm that can be applied to the traveling salesman problem. It can also be applied to the more general integer programming problem,[65,73] which was described in Equation 12. The idea behind this algorithm is to solve versions of the relaxed problem, which removes the integer restriction from the variables, in order to set bounds on the integer problem. These bounds are then used to eliminate branches of the branch-and-bound tree until all nodes are either pruned or solved. This yields the optimal integer solution of the original integer programming problem. Figure 8 illustrates a segment of a branch-and-bound search tree.

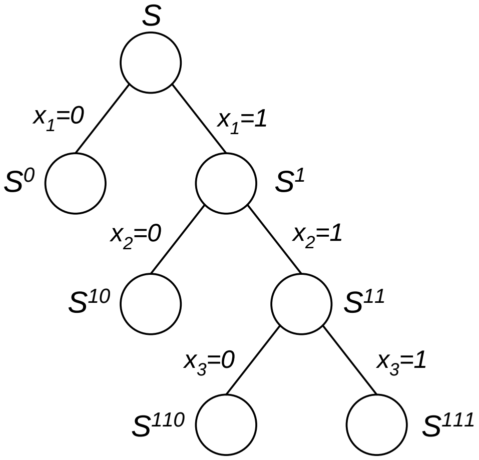

Figure 8: Portion of a branch-and-bound search tree.

In describing the general branch-and-bound algorithm, L is a collection of integer programs, each of which takes the form $z^i_{IP} = \max\{cx : x \in S^i\}; \quad S^i \subseteq S$. Associated with each problem in L is an upper bound $\overline{z}^i \geq z^i_{IP}$. First, the optimal relaxed solution is determined for the problem as a whole. If the variables that yield the optimal solution all take integer values, then the algorithm is terminated and the optimal solution is found. Generally, this is not the case, so branches of the tree must begin to be enumerated. There are numerous techniques for deciding the order in which these should be evaluated, but for this example, let us assume that the next relaxed solution will be for S^0. Therefore, x_1 is fixed to a value of 0, and a relaxed solution is computed, where all remaining variables are allowed to take any value. This continues down the tree until an integer solution is computed. This integer solution becomes the lower bound on the problem. The process continues down the various branches of the tree, computing relaxed solutions at each node. If the relaxed solution is less than the best known integer solution, the remainder of the branch can be pruned out of the search tree. If a better integer solution is found, that becomes the lower bound on the integer programming problem. When all branches have either been pruned out or yield integer solutions, the optimal solution to the integer programming problem is known.

While it is seen how the asteroid rendezvous problem could be formulated as a branch-and-bound search tree, the relaxed problem does not exist for the asteroid problem. Because the discrete variables are either the asteroids themselves or a binary value indicating whether or not each asteroid is visited, the variables can only take discrete values. Therefore, there is no solution for fractional values of the variables. The idea of an enumerative and logical search process, however, could be applied to choosing the optimal sequence of asteroids.

1.3 Research Objectives and Contributions

In recent years, missions to asteroids and comets have gained significant scientific interest, while electric propulsion has become a viable option for spacecraft exploring these bodies. Conceptual design of these missions relies on the ability to quickly generate solutions for a wide variety of launch dates, flight times, and arrival bodies, in order to find trajectories that minimize propellant mass, flight time, or other parameters of interest. Until recently, however, a thorough exploration of the design space was challenging because indirect methods for low-thrust trajectory optimization were time consuming, user-intensive, and often dependent on already known solutions to similar problems.

Recent developments in the area of low-thrust trajectory optimization, namely advances in direct and shape-based analytic methods, have enabled approximate solutions, suitable for conceptual design, to be generated more rapidly. Because the solution space with respect to many of the mission design parameters (launch date, flight times, etc) is multi-modal, genetic algorithms have been used in many instances in an attempt to locate the global optimum. Several authors have also applied combinatorial optimization methods to the spacecraft rendezvous problems, and some of the methods and techniques used in these studies can be applied to solving the larger asteroid tour problem.

Conceptual design of spacecraft missions requires the ability to explore large portions of the design space in order to locate the best set of solutions. At the conceptual design level, mass and flight time are generally the most important engineering parameters of interest. Over a large design space, the difference between the best trajectories will generally differ only slightly, and that difference may be smaller than the error generated by approximations made at the conceptual design level. Furthermore, when proceeding beyond the conceptual design phase, there are a number of other factors that must be considered when choosing a final trajectory (e.g., science objectives, reconnaissance view angles, and telecom considerations). As a result, at the conceptual design level, it is perhaps more important to identify a broad suite of good solutions across the design space, than a single optimum solution.

This study presents a systematic methodology for efficiently determining a broad suite of good solutions to combinatorial low-thrust asteroid rendezvous problems. The target application is for conceptual design, where broad exploration of the design space is critical. The proposed methodology has two steps, the first that quickly eliminates poor solutions from the design space, and the second that then locates the best solutions from the reduced design space. The goal of the pruning step is to quickly reduce the size of the problem by several orders of magnitude. This is accomplished using heuristics specific to the physics of the underlying problem, in order to identify areas of the design space that will not likely yield favorable solutions in terms of the objective function. Heuristic methods, however, cannot guarantee that only poor solutions will be eliminated from the design space. The goal of this first phase is to ensure that a large percentage of the best solutions remain for the second phase. In this second phase, a global optimization algorithm is applied to the reduced design space to locate the best set of solutions. The global optimizer is responsible for solving for the following design variables: asteroid combination, launch date, times of flight, and stay times. This system-level optimization is coupled with a local low-thrust trajectory optimization scheme that determines the optimal control history of the spacecraft in order to minimize propellant for a given set of global optimization variables.

The methodology developed is used to predict the solution to a range of test problems with a known optimal soltution, and once verified, is applied to several larger asteroid tour problems, including versions of the GTOC2 and GTOC3 problems. The GTOC competition problems were chosen because a set of solutions is available from the competition results which can be used as a benchmark in evaluating the performance of the methodology.

The following are the key contributions of this work:

(1) A three-level heuristic sequence is developed based on the physics of the problem that allows for efficient pruning of the design space. In reducing the size of the design space, a majority of the better solutions are maintained. This pruning methodology is verified through solution of an intermediate-size sample problem whose solution is obtained through complete enumeration of the design space. The pruning methodology is shown to apply well across a range of low-thrust

asteroid tour mission design problems and relies on user-defined parameters to effectively tailor the degree of design space pruning based on the available computational resources.

(2) A global optimizer is combined with a low-thrust trajectory optimization method to locate a broad suite of good solutions for the reduced problem. This approach combines an innovative branch-and-bound algorithm (to solve for the optimal asteroid sequence) with a genetic algorithm (which solves for the optimal departure date, times of flight, and stay times for a given asteroid sequence), and finally with a low-thrust trajectory optimization program (which determines the optimal thrust profile that maximizes final mass). The global optimization scheme is able to consistently locate the best known solution, along with a suite of good solutions across the design space. A strategy was developed to set the initial lower bound in the branch-and-bound algorithm (a user-defined parameter) as a means of controlling the number of required low-thrust optimizations required and the number of good solutions found.

(3) When the global optimization scheme is combined with the heuristic screening process, a systematic methodology for identification of a broad suite of good solutions to low-thrust, multiple asteroid rendezvous, conceptual mission design problems is achieved. In addition to a wide range on each of the continuous variables, the problems to which the methodology is applied in this investigation are characterized by as many as 41 billion discrete asteroid sequences. The key contribution of this methodology is the ability to locate a suite of good solutions, as opposed to just a single optimum solution. In locating these good solutions, the overall methodology is able to reduce the number of asteroid sequences that

require low-thrust optimization by 6-7 orders of magnitude, as compared to the number of asteroid sequences in the original problem.

The remainder of this dissertation is broken down into five chapters. Chapter 2 outlines the approach taken in developing the methodology, and presents results for each of the techniques examined as applied to a small test problem with a known solution. Chapter 3 presents the final methodology that was developed. Chapter 4 then validates the methodology, by applying it to an intermediate-sized sample problem, also with a known solution. Chapter 5 applies the methodology to two large problems – a modified version of the GTOC3 problem and the GTOC2 problem. Finally, Chapter 6 presents a summary and conclusions, along with recommendations for further work.

CHAPTER II

DEVELOPMENT OF METHODOLOGY

2.1 Approach

The methodology developed in this dissertation is applicable to large, combinatorial asteroid rendezvous problems. The approach taken in developing this methodology, however, can be applied to developing similar methodologies for solving other trajectory design problems with large design spaces. The first piece of the methodology involves pruning the design space by several orders of magnitude, by eliminating solutions that do not produce favorable values of the objective function. In order to do this, heuristics must be chosen that are specific to the physics of the underlying problem. A number of metrics were considered for the pruning process, and were evaluated based on both their ability to approximate low-thrust final mass as well as their speed of execution. In order to test their effectiveness, a small test problem was created and solved. The candidate metrics could then be compared to the corresponding low-thrust final mass for each set of design variables. This chapter will describe each candidate pruning metric in detail, and then present the results of each as applied to the sample problem.

Next, an appropriate global optimization scheme must be selected that can search the full design space to locate a suite of good solutions. The global optimizer is responsible for solving for the system-level variables: asteroid sequence, Earth launch date, times of flight, and stay times. Each function call of the global optimizer calls a low thrust trajectory optimization routine, which is responsible for solving for the optimum $\vec{V}_\infty$ at Earth departure and the thrust profile along the trajectory that maximizes final mass. Each candidate method is evaluated against the sample problem both for how many of the best known solutions it is able to reliably locate as well as speed of execution.

2.2 Candidate Pruning Methods

A number of pruning techniques were considered that were believed to approximate the physics of the low-thrust trajectory problem. Each metric was evaluated to determine which ones could be used to reliably eliminate areas of the design space that do not yield high values of the objective function. These pruning metrics generally fell into three categories: (1) ephemeris-based metrics, which use parameters such as semi-major axis, inclination, and longitude of the ascending node, (2) approximations to the low-thrust trajectories, including two-impulse Lambert solutions with either single or multiple revolutions, and (3) metrics that attempt to take phasing into consideration.

2.2.1 Ephemeris-Based Pruning Techniques

A number of ephemeris-based metrics were examined for potential use during the pruning phase of the methodology. Many of these are based on the basic orbital elements of the asteroid orbits, as illustrated in Figure 9. In this diagram, the orbit is referenced to the heliocentric-ecliptic frame, which is centered at the Sun, with the fundamental plane (*I*-*J*) defined by the plane of the Earth's revolution around the Sun. Additionally, $\vec{\mathrm{h}}$ represents the angular momentum vector of the orbit, $\bar{\mathrm{r}}$ is the radius vector from the center of the orbit to the current position of the body, $\bar{\mathrm{e}}$ is the eccentricity vector, which points in the direction of periapsis, v is the true anomaly, ω is the argument of periapsis, $\bar{\mathrm{n}}$ is the line of nodes, and Ω is the longitude of the ascending node.

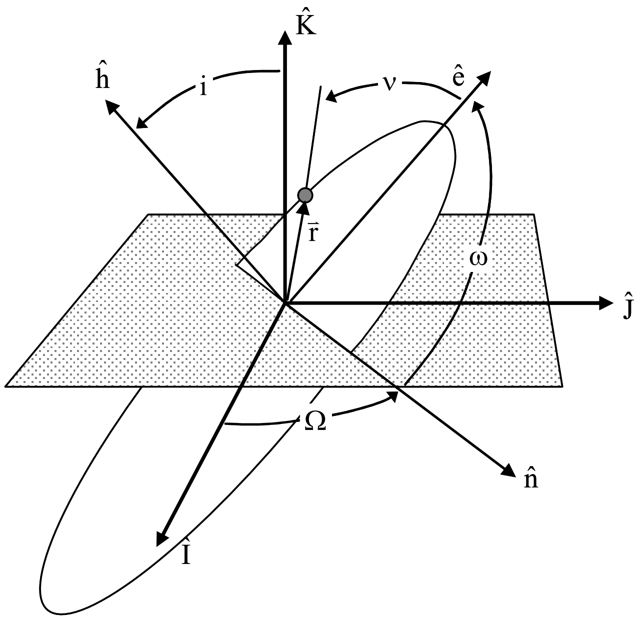

Figure 9: Classical orbital elements.

The first ephemeris-based pruning metric considered is the semi-major axis of each asteroid's orbit, which is defined as half of the major axis of the orbit ellipse. Here, semi-major axis is used as a surrogate for distance from the Sun. Visiting the asteroids in either increasing or decreasing order makes intuitive sense, in order to minimize fuel consumption. Because time of flight appears in the objective function, however, visiting the asteroids in order of increasing semi-major axis would be necessary to minimize the overall flight time of the mission.

The next ephemeris-based pruning metric considered is the change in inclination between the orbits of two asteroids, where inclination is defined as the angle between the angular momentum vector of the orbit and the vector normal to the ecliptic plane. The angular momentum vector is calculated as normal to the orbital plane. Using inclination change between two orbits as a pruning metric is based on the conjecture that large inclination changes require significant amounts of propellant, as is the case for impulsive orbit transfers.

Similarly, the change in the longitude of the ascending node between two orbits is evaluated. The longitude of the ascending node is defined as the angle between the $\hat{I}$ unit vector, generally pointing in the direction of the vernal equinox, and the point where the body crosses through the fundamental plane in a northerly direction, measured counterclockwise. While inclination change can have a significant effect on propellant consumption, the orientation of the two orbits is also important. Two methods for combining the change in inclination and the change in the longitude of the ascending node between two orbits are considered. First, each metric is normalized and weighted as follows:

$$J = W_i \cdot abs\left(\frac{\Delta i}{(\Delta i)_{\max}}\right) + W_\Omega \cdot abs\left(\frac{\Delta\Omega}{(\Delta\Omega)_{\max}}\right) \tag{13}$$

Inclination change varies from 0 to 180 degrees, while ascending node change varies from -180 to 180 degrees. Second, the angle between the angular momentum vectors, θ_{wedge}, of the two orbits was calculated as follows:

$$\cos(\theta_{wedge}) = \frac{\vec{h}_i \cdot \vec{h}_{i+1}}{\left\|\vec{h}_i\right\| \cdot \left\|\vec{h}_{i+1}\right\|} \tag{14}$$

Several other ephemeris-based methods are also considered. First is to choose asteroids in Group 1 (furthest from the Sun) with low energies – therefore, asteroids with the smallest values of semi-major axis. In theory, these should be the most accessible out of the entire set. Another ephemeris-based method would screen out asteroid combinations based on distance between their orbits. This can be done by calculating the distance between the first asteroid's apoapsis (the furthest distance from the Sun in the orbit) and the second asteroid's periapsis (the closest distance from the Sun in the orbit).

Finally, both the eccentricity of a single orbit and the change in eccentricity between two orbits is evaluated.

2.2.2 Phase-Free, Ballistic Approximations

In addition to the ephemeris-based pruning methods, two-impulse Lambert solutions are evaluated for potential use as a pruning metric. The optimal, phase-free, two-impulse solution calculates the minimum delta-V transfer between two orbits, disregarding the actual location of the chosen asteroids. Of course, there is no guarantee that the optimal asteroid configuration for a given asteroid pairing will occur during the date range given in the problem, but the idea behind this technique is to identify the most "reachable" asteroids. In general, ballistic solutions will best approximate the low-thrust trajectories when the ratio of thrust time to flight time is low for the low-thrust trajectories.

Given two points in space, there are two elliptic orbits for a given semi-major axis that connect those two points.[74] The two orbits constitute the "short-way" and "long-way" transfers, or changes in true anomaly of less than and greater than 180 degrees, respectively. Therefore, for a given semi-major axis, there are two associated values of delta-V, one for the short-way transfer and one for the long-way transfer. Furthermore, for each transfer orbit, any number of revolutions can be made, each resulting in a different time of flight. In the case of circular orbits, Shen and Tsiotras show that for a given value of semi-major axis, the short-way transfer always has a lower delta-V value than the long-way transfer. As an example, Figure 10a plots delta-V as a function of the semi-major axis of the short-way transfer trajectory from asteroid 2006 QQ56 to asteroid Chicago, both with starting true anomalies of zero. As can be seen, the function is unimodal. Therefore, the minimum delta-V solution can easily be found using any gradient-based optimizer.

In order to determine the optimum, two-impulse, phase-free delta-V between any two orbits, the true anomalies at departure and arrival are discretized between 0 and 360 degrees. Each possible combination of departure true anomaly and arrival true anomaly defines $\boldsymbol{r_1}$, $\boldsymbol{r_2}$, and the transfer angle, from which the minimum ΔV solution can be obtained. Figure 10b plots contours of minimum delta-V for each value of departure and arrival true anomaly, also for the transfer from "2006 QQ56" to Chicago. Because the solution space is multi-modal, a grid search is used to determine the approximate optimal solution.

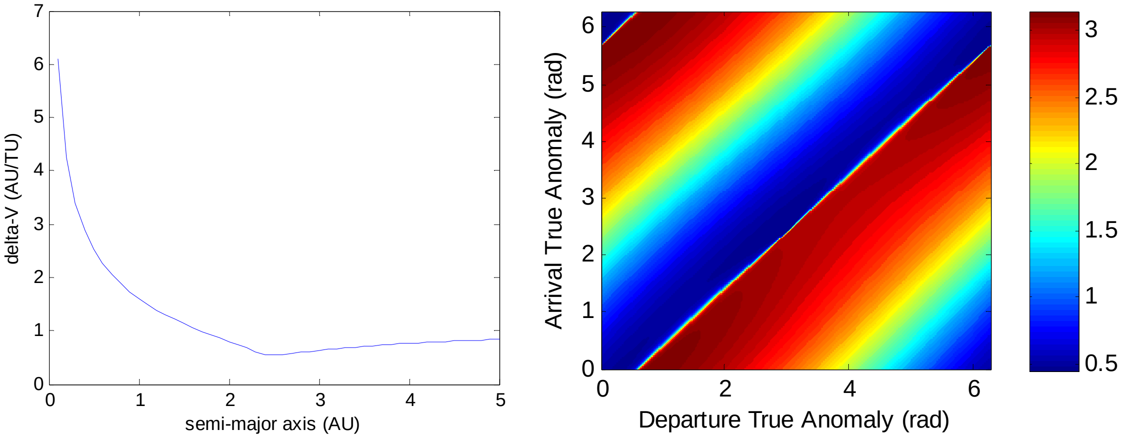

Figure 10: (a) Delta-V as a function of the transfer orbit semi-major axis for a two-impulse transfer (left); (b) contour plot of the minimum two-impulse delta-V transfers over all departure and arrival true anomalies (right).

2.2.3 Pruning Techniques Based on Phasing

The final set of pruning metrics considered takes phasing into consideration. Once again, two-impulse Lambert solutions are calculated, now using the actual asteroid ephemeris data for given departure dates and flight times. In this case, because time of flight is a consideration, the Lambert problem is reformulated in order to solve for the minimum delta-V given $\boldsymbol{r_1}$, $\boldsymbol{r_2}$, and the time of flight. For a given $\boldsymbol{r_1}$, $\boldsymbol{r_2}$, and time of flight, there are $2N_{max} + 1$ solutions to the multi-revolution Lambert problem, where N_{max} is the maximum possible number of revolutions for a given time of flight. All $2N_{max} + 1$ solutions must be calculated in order to determine the minimum delta-V solution.

Another possible approach to address phasing is to determine when the Group 1 asteroids are at their perihelion, based on the assumption that it is most efficient to rendezvous with the last asteroid near its perihelion passage, where the orbital energy is the least. The previous asteroids and departure dates are then chosen such that the spacecraft will in fact arrive at the final asteroid in the vicinity of perihelion. This, in fact, was the pruning approach taken by the winning team in GTOC2.

2.3 Candidate Global Optimization Methods

The goal of the global optimization component of the methodology is to search the reduced design space and locate a suite of good solutions, where the design variables consist of the asteroid sequence, Earth departure date, flight times, and stay times. For a given value of each of the global variables, the global optimization method must call a low-thrust trajectory optimization routine in order to determine the thrust profile that maximizes final mass. The asteroid sequence is a discrete, combinatorial problem. The departure date, flight times, and stay times, however, are continuous variables, but the objective function (final mass) is multimodal with respect to these variables. Two different schemes for solving for the global variables are evaluated, as illustrated in Figure 11 and Figure 12. The design variables listed in the figures represent those required for a rendezvous with four asteroids and no return to Earth. Furthermore, the objective function illustrated is the ratio of final mass to time of flight, with a total flight time constraint of twenty years. First, a single method is used to solve for all of the global variables. Alternatively, the variables are divided, and a two-level optimization scheme is employed. An outer loop is responsible for finding the optimal asteroid sequence, while an inner loop calculates the optimal departure date, flight times, and stay times for a given asteroid sequence. Two optimization methods are considered for the

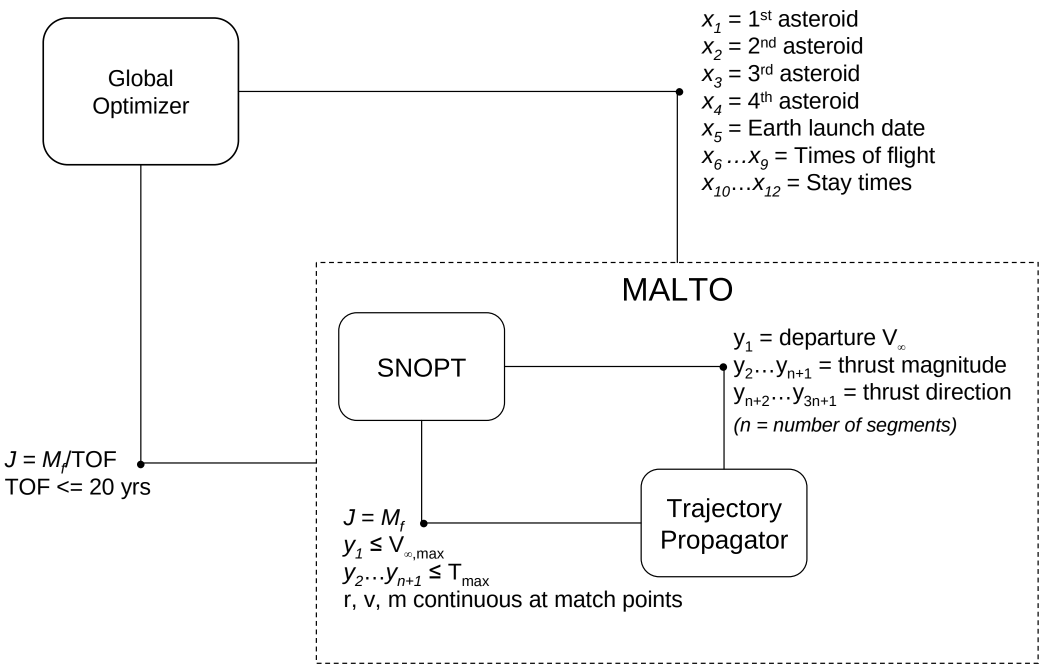
Global Optimizer
x_1 = 1st asteroid
x_2 = 2nd asteroid
x_3 = 3rd asteroid
x_4 = 4th asteroid
x_5 = Earth launch date
$x_6 \ldots x_9$ = Times of flight
$x_{10} \ldots x_{12}$ = Stay times
MALTO
SNOPT
y_1 = departure V_∞
$y_2 \ldots y_{n+1}$ = thrust magnitude
$y_{n+2} \ldots y_{3n+1}$ = thrust direction
(n = number of segments)
$J = M_f$/TOF
TOF <= 20 yrs
Trajectory Propagator
$J = M_f$
$y_1 \leq V_{\infty,max}$
$y_2 \ldots y_{n+1} \leq T_{max}$
r, v, m continuous at match points

Figure 11: Single-level global optimization scheme.

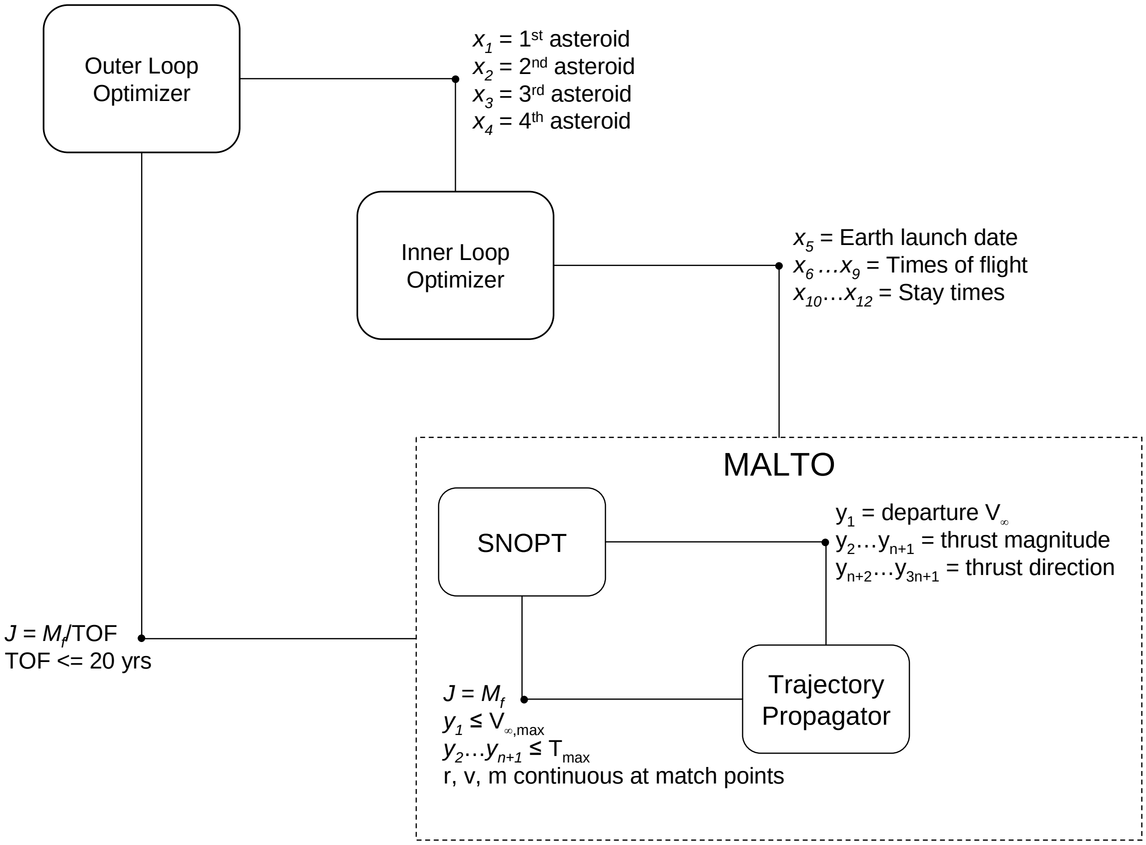
Outer Loop Optimizer
x_1 = 1st asteroid
x_2 = 2nd asteroid
x_3 = 3rd asteroid
x_4 = 4th asteroid
Inner Loop Optimizer
x_5 = Earth launch date
$x_6 \ldots x_9$ = Times of flight
$x_{10} \ldots x_{12}$ = Stay times
MALTO
SNOPT
y_1 = departure V_∞
$y_2 \ldots y_{n+1}$ = thrust magnitude
$y_{n+2} \ldots y_{3n+1}$ = thrust direction
$J = M_f$/TOF
TOF <= 20 yrs
Trajectory Propagator
$J = M_f$
$y_1 \leq V_{\infty,max}$
$y_2 \ldots y_{n+1} \leq T_{max}$
r, v, m continuous at match points

Figure 12: Multi-level global optimization scheme.

outer loop: a genetic algorithm and a branch-and-bound-methodology. A genetic algorithm is evaluated for the inner loop optimizer. The inner loop then calls MALTO, which is responsible for the low-thrust trajectory optimization. How MALTO is incorporated into the methodology is explained in greater detail in Section 2.4.

2.3.1 Genetic Algorithm

As explained earlier, genetic algorithms are a class of evolutionary algorithms which are domain-spanning, probabilistic optimization algorithms based on the Darwinian theory of evolution. The version of genetic algorithm in consideration begins with an initial random population, using a binary representation of each design variable. The number of bits chosen for each variable determines the resolution of that variable. These binary values are then mapped to their corresponding decimal values based on the chosen resolution and the range assigned to each variable. Additionally, the population size, which remains constant throughout the optimization routine, is a user-defined parameter that must be chosen. The genetic algorithm relies on three operations within each generation to improve the overall "fitness" of the population – reproduction, crossover, and mutation.

The purpose of the reproduction operation is to choose the best candidate designs from the population and allow them to pass to the next operation. The method used in this work is Tournament Selection, which is stochastic in nature. In this method, a certain number of "combatants" are randomly chosen from the population. The candidate design with the best objective function wins the tournament and is placed in the post-reproduction pool. This process is repeated until the post-reproduction pool is completed, maintaining the same population size as the initial population. Because the reproduction operation requires knowledge of the fitness of each member of the population, the fitness evaluations (i.e., function calls) are made prior this operation.

The next operation is crossover, which mimics two parents having two children and passing on their characteristics to them. This process assumes better designs can be created by splicing together parts of two known good designs. In crossover, two candidate designs ("parents") are chosen out of the post-reproduction table. For each set of parents, there is some probability that crossover will occur. If there is no crossover, the two parents are passed unchanged into the post-crossover population table. In this genetic algorithm, two-point crossover has been implemented. An entry and exit bit are randomly selected, and for the bits between the entry bit and the exit bit, the two parents switch bits. The children are then passed into the post-crossover pool. This process is continued with every set of two candidate designs in the post-reproduction population.

Mutation is the final operation in the genetic algorithm process. Because the entire process is stochastic in nature, it is possible to contain a column of data in the population table that is all zeros or all ones. Neither reproduction nor crossover would allow a bit in such a column to change. Mutation, therefore, provides an opportunity for this to occur. In this process, each candidate design has some probability of undergoing mutation. If mutation does not occur, the candidate design passes unchanged to the post-mutation table. If mutation does occur, string-wise mutation is implemented, where one bit in the chromosome string is randomly selected. This bit is flipped and the chromosome string is then passed to the post-mutation pool. The post-mutation population is then passed back to the reproduction operation, and the three processes are repeated until the algorithm converges on a final solution. The genetic algorithm is considered to be converged when there is no change in the best overall solution after a certain number of iterations ("generations").

In order to limit the number of required function calls to the low-thrust trajectory optimization routine, an archiving scheme is used, whereby each candidate design evaluated is saved in a table. This also enables a number of good solutions to be found, along with the global optimum.

2.3.2 Branch-and-Bound

The branch-and-bound methodology considered here is a variation on the general branch-and-bound method that uses linear programming relaxations, as presented in Section 1.2.3. In an integer programming problem the relaxation step involves removing the integer constraints and solving for the solution to the continuous problem. Relaxing the integer constraints is not possible, however, when the integer variables represent discrete asteroids choices. Therefore, in place of the linear programming relaxation, a two-impulse Lambert solution is used to provide an upper bound on the low-thrust solution for the branch-and-bound algorithm. The proposed branch-and-bound algorithm is based on the conjecture that the two-impulse solutions provide a reliable upper-bound to the low-thrust problem. This assumption will be examined on the sample problem.

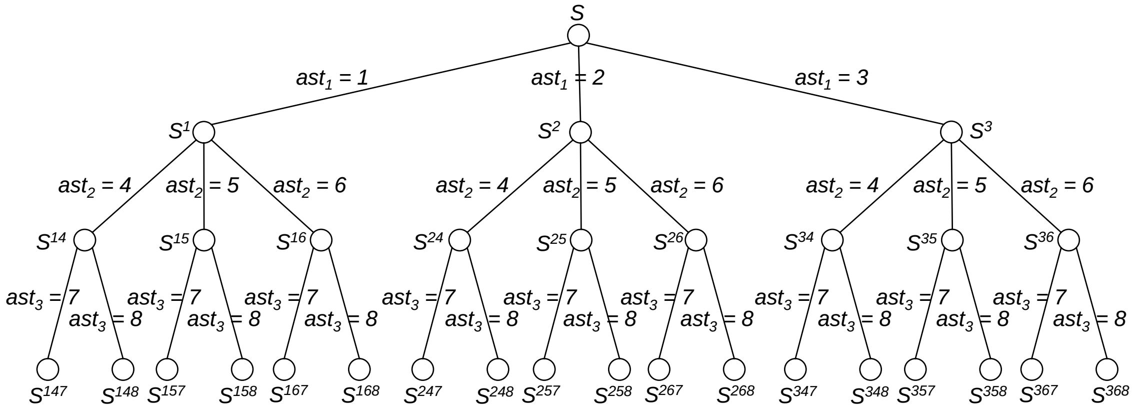

Figure 13: Example branch-and-bound tree.

The search tree enumerates all possible asteroids sequences, a small example of which is illustrated in Figure 13. The first branch represents the choice of the first asteroid to visit from Earth. The next branch represents the second asteroid to visit in the sequence and so on. The branch-and-bound tree is used only to solve for the optimal asteroid sequence – the optimal departure date, flight times, and stay times must be obtained using another method. In order to begin the algorithm, a known low-thrust optimal solution must first be obtained. Therefore, a single sequence must be chosen and the optimal value of the objective function for that sequence is calculated, based on a low-thrust trajectory. This becomes the lower bound on the objective function. Subsequently, branches of the tree are evaluated, using the optimal two-impulse Lambert solutions as a surrogate for the LP-relaxation. If the relaxed solution is less than the lower bound, that branch of the tree can be pruned. Otherwise, that branch must be maintained and the optimal low-thrust solution must be calculated.

Using Figure 13 as an example, the low-thrust optimal solution for branch S^{147} is calculated first. This branch corresponds to *asteroid 1* = 1, *asteroid 2* = 4, and *asteroid 3* = 7. These values are simply indices that refer to particular asteroids. As aforementioned, another method must be used to determine the optimal departure date, flight times, and stay times for the asteroid sequence corresponding to that particular branch. The optimal low-thrust solution for this asteroid sequence sets a lower bound on the objective function. Once this lower bound has been set, the search process begins at the top of the branch-and-bound tree. To illustrate how branches of the tree are pruned out, a relaxed solution is calculated for S^2. This relaxed solution is the two-impulse optimal solution for Earth to *asteroid 1* = 2, over all departure dates, flight times, and stay times set by the problem. If the relaxed solution is less than the lower bound, the entire branch of the tree below S^2 can be pruned. If not, further depth is required on that branch. The relaxed solution is then calculated for S^{24}. Again, if that solution is less than the lower bound, that branch is pruned. If not, then the relaxed solution to S^{247} is calculated. If the relaxed solution is still greater than the lower bound, then the low-thrust optimal solution for that entire asteroid sequence (*asteroid 1* = 2, *asteroid 2* = 4, and *asteroid 3* = 7) must be calculated. If the resulting low-thrust solution is greater than the lower bound, then a new lower bound is set. This process continues until all of the branches of the tree are either pruned or their low-thrust optimal solutions are calculated.

The order in which the branches are explored will have a strong impact on the number of optimal low-thrust solutions which must be calculated. Several methods will be examined to determine which order results in the most efficient algorithm.

2.4 Low-Thrust Trajectory Optimization

MALTO is a low-thrust trajectory optimization algorithm based on the direct method by Sims and Flanagan, which was described in Section 1.1.2. For a given function call to MALTO, the global optimizer passes the following variables: asteroid sequence, Earth departure date, times of flight between each asteroid, and the stay time at each asteroid. Initial mass is also passed to MALTO, although in many instances, such as in the GTOC problems, this is a fixed value. Additionally, MALTO requires initial guesses for the following variables: departure and arrival V_∞ vectors and the thrust magnitude and direction on each segment of the trajectory. The number of segments is a user-defined variable in MALTO, which remains constant throughout the MALTO optimization process. A constraint may also be placed on the magnitude of the V_∞ vector at Earth departure. While any initial guess can be chosen for the remaining V_∞ vectors, they must equal zero in the final optimized trajectory. The thrust can be modeled in two ways: (1) using the Cartesian coordinates of the thrust vector on each segment (T_x, T_y, T_z) or (2) using the thrust magnitude and two thrust angles on each segment (T_{mag}, T_{lat}, T_{lon}). Within MALTO, SNOPT is used to solve for the Earth departure V_∞ vector and the thrust profile to maximize the final spacecraft mass, while meeting all of the internal MALTO constraints.

When the genetic algorithm sends a set of global variables to MALTO, MALTO is used to optimize the trajectory leg-by-leg. First, MALTO calculates the optimum final mass for the trajectory from Earth to the 1st asteroid, with the fixed time of flight passed from the GA. It then uses the final mass as the initial mass for the next leg of the trajectory – 1st asteroid to 2nd asteroid – and optimizes this leg. This process continues until all legs have been optimized. This leg-by-leg approach produces the same results as a single end-to-end optimization, but requires significantly less computation time.

Several approaches were evaluated for choosing the initial guesses into MALTO, such that MALTO is able to converge on the optimum solution for a given trajectory:

1. The optimal two-impulse solution is calculated for a given trajectory (using the same launch date and flight times). These values are then used as the initial guesses for the V_∞ vectors, with the initial guess of the thrust set to zero all along the trajectory.

2. Both the V_∞ vectors and the thrust are set to zero.

3. The V_∞ vectors are set to zero, and the thrust magnitude is set to its maximum value with the direction of the thrust pointed in the direction of the spacecraft velocity vector ($T_{mag} = T_{max}$, $T_{lat} = 0$, $T_{lon} = 0$).

Which initial guess approach worked best depends on the particular asteroid pair, departure date, and time of flight. As an example, Figure 14 plots the optimum final mass values for a transfer from Earth – 2006 QQ56, with a 600-day time of flight, 30 segments, over a range of departure dates, using each of the three initial guess approaches tested. As can be seen, the third approach consistently results in MALTO locating the optimum solution, while the second approach finds the optimum solution for all but a handful of points. The first approach (ballistic initial guess for V_∞ and zero thrust), however, results in MALTO converging to a sub-optimal solution across all departure dates considered. In another example, Figure 15 plots the optimum final mass values for a transfer from Medusa – Kostinsky, with a 1200-day time of flight and 30 segments, again using the three initial guess approaches. For this transfer, the first initial guess approach performs best, followed by the third approach. The second approach (zero V_∞, zero thrust), however, converges to a sub-optimal solutions for a large number of departure dates. Therefore, for each leg optimized in MALTO, all three approaches are used, and the best solution is kept.

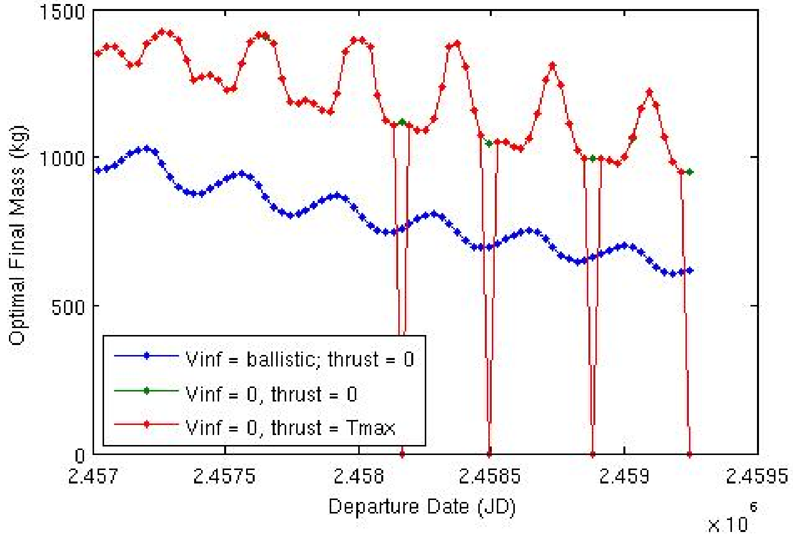

Figure 14: Optimum final mass for Earth – 2006 QQ56 with a 600-day time of flight, using three different approaches for the initial guess of the MALTO variables.

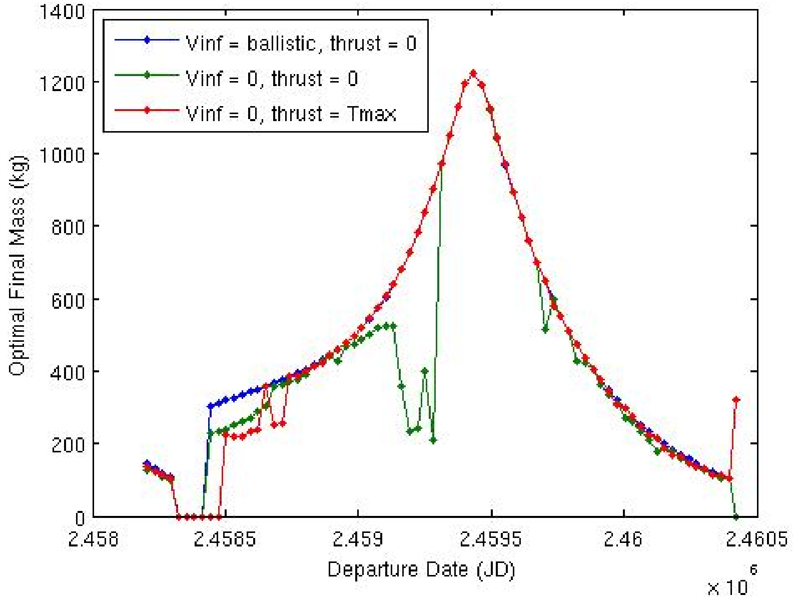

Figure 15: Optimum final mass for Medusa – Kostinsky with a 1200-day time of flight, using three different approaches for the initial guess of the MALTO variables.

Next, the effect of the chosen number of segments is examined. The greater number of segments used, the closer the discretized trajectory approximates the true low-thrust trajectory. More segments, however, also results in greater computation time because of a larger number of variables required to represent the thrust profile. Additionally, more segments leads to more variables, which creates a more difficult problem for SNOPT to solve. In general, increased computation time is required as the number of segments on the trajectory increases. As an example, the optimum low-thrust final mass was calculated over a range of departure dates, again for a transfer from Earth – 2006 QQ56, with a 600-day time of flight, using several different values for the number of segments. Figure 16 plots the percent difference in final mass for each of the number of segments considered, as compared to the final mass computed using 90 segments. As can be seen, the optimum final mass varies by less than half a percent as a function of the number of segments into which the trajectory is discretized. Figure 17 then plots the MALTO run time per trajectory across the range of number of segments considered in Figure 16. Based on these results and the application to conceptual design, it is appropriate to keep the number of segments small, in order to reduce computation time.

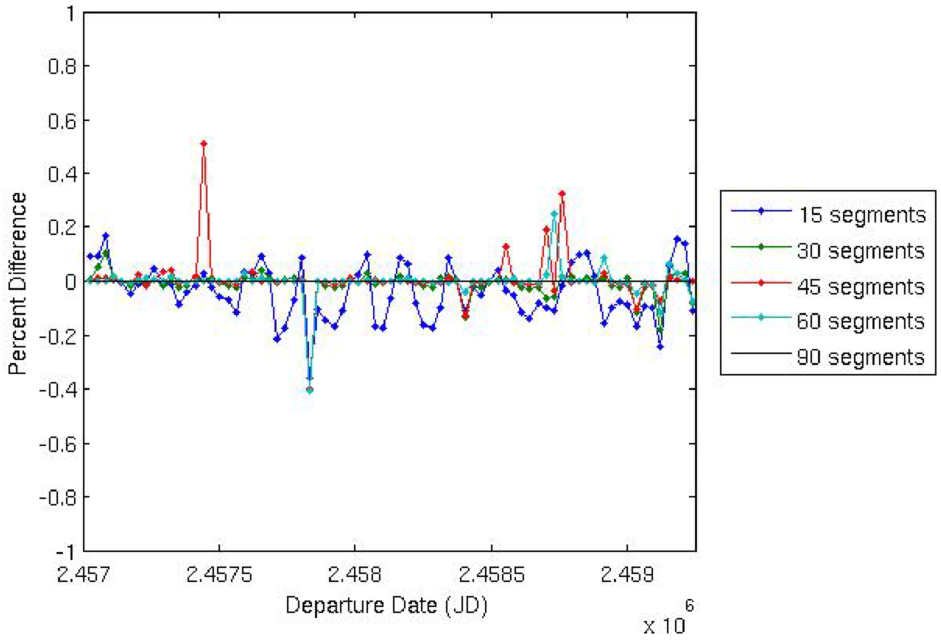

Figure 16: Effect of number of segments on the optimum final mass for Earth – 2006 QQ56, with a 600-day time of flight.

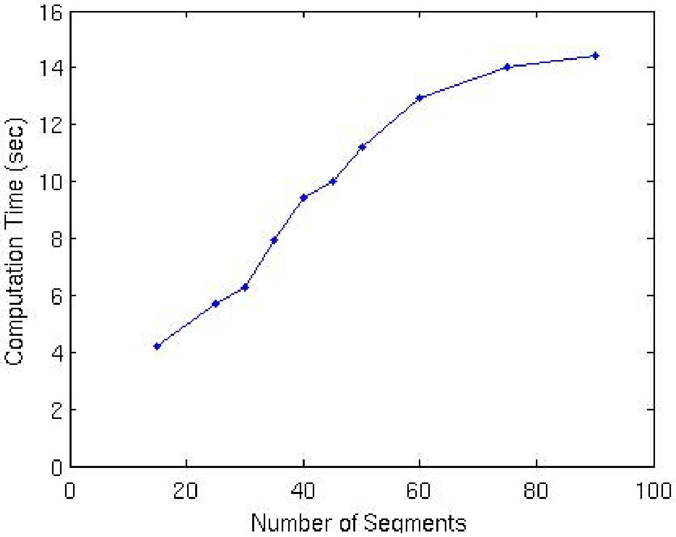

Figure 17: Effect of number of segments on computation time for Earth – 2006 QQ56, with a 600-day time of flight.

2.5 Small Sample Problem

In order to test each of the candidate pruning and global optimization methods, a small sample problem with a known solution was developed. The sample problem contains twenty four total asteroids, split evenly between three groups, leading to 3072 discrete asteroid combinations. These asteroids were randomly chosen from the set of asteroids provided in the GTOC2 competition, and can be found in Appendix A. Figure 18 plots these asteroids, as a function of their semi-major axis, eccentricity, and inclination.

The objective function is to maximize the final mass of the spacecraft, and the following constraints are placed on the flight times: Earth to Group 4 ≤ 600 days, Group 4 to Group 2/3 ≤ 1800 days, and Group 2/3 to Group 1 ≤ 1200 days. These constraints assume that the asteroids will be visited in the following order: Earth – Group 4 – Group 2/3 – Group 1. Applying this constraint reduces the number of discrete asteroid combinations to 512. The validity of this assumption will be addressed in Section 2.5.1. Lastly, the launch window must fall between 2015 and 2025, inclusive, and the stay time at each asteroid is fixed at 90 days. While flight time does not directly appear in the objective function, it is dealt with implicitly in the chosen constraints. Launch from Earth is constrained by a hyperbolic excess velocity (V_∞) of up to 3.5 km/s with no constraint on direction. The spacecraft has a fixed initial mass of 1500 kg, which does not change with launch V_∞, and a minimum final mass of 500 kg. The propulsion is modeled to have a constant specific impulse of 4000 s and a maximum thrust level of 0.1 N, and can be turned on and off as needed.

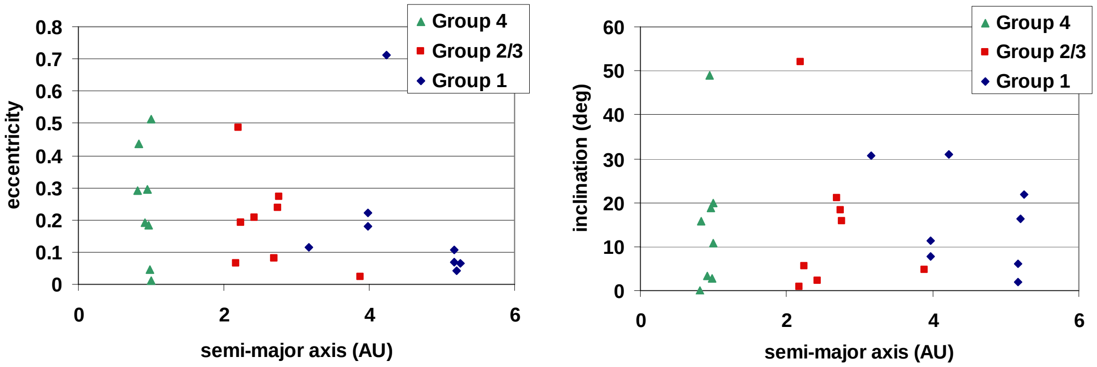

Figure 18: Set of asteroids for sample problem.

Within the sample problem, MALTO was used to perform the local trajectory optimization. The design space was discretized in terms of launch date and times of flight, and each leg of the trajectory was analyzed separately. At first, only asteroid combinations following the assumed group order were considered (all other possible combinations were examined later). The launch date from Earth was discretized in 30-day steps, and the time of flight to the first asteroid (Group 4) was discretized in 100 day steps up to the 600-day constraint. MALTO was used for each case to determine the departure V_∞ and thrust profile that maximizes the final mass at the arrival asteroid, based on a 1500 kg initial spacecraft mass. The time of flight for the second leg was also discretized in 100 day increments, up to 1800 days. For each feasible Leg 1 trajectory (final mass greater than 500 kg), the corresponding Leg 2 trajectory was calculated to each of the Group 2/3 asteroids, for each of the discretized times of flight. Finally, the set of Leg 3 trajectories was calculated in a similar fashion, starting from all of the feasible Leg 2 trajectories (final mass greater than 500 kg). This approach locates the entire set of (discretized) feasible solutions, and ranks them by final mass.

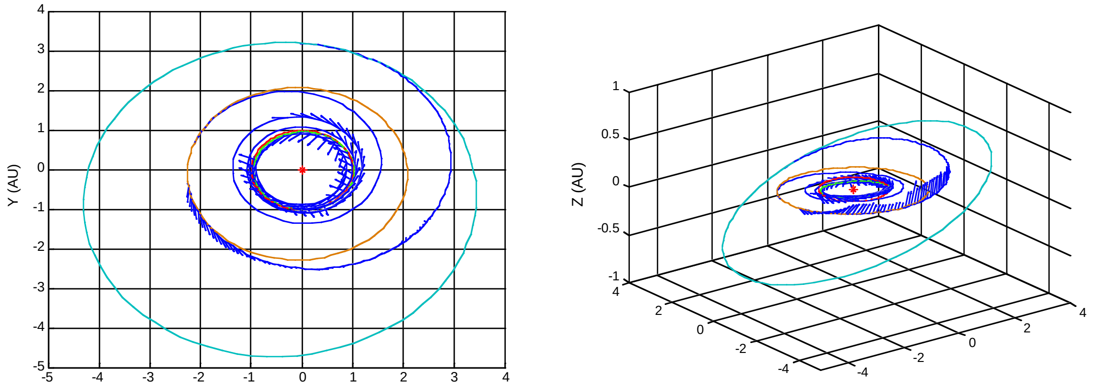

Figure 19: Optimal solution for the small sample problem.

The resulting set of feasible solutions contains 115 of the possible 512 asteroid combinations initially examined. This set of solutions contains 4 Group 1 asteroids, 6 Group 2/3 asteroids, and all 8 Group 4 asteroids (although not every permutation of these 18 asteroids). The best solution, plotted in Figure 19, visits the following asteroids: 2006 QQ56 – Medusa – Kostinsky. The spacecraft departs Earth on March 1, 2015 with a launch V_∞ of 2.59 km/s. The time of flight for each leg is 600 days, 1600 days, and 1200 days, respectively. Interestingly, even though time of flight does not appear explicitly in the objective function, the flight time for the second leg is not equal to its upper bound. While an 1800-day time of flight would result in a larger final mass for that particular leg, the shorter flight time results in better phasing for the third leg, thereby maximizing the overall final mass of the trajectory. The total flight time from Earth departure to the final asteroid rendezvous is 3580 days, which includes the two 90-day stay times at each intermediate asteroid, and the arrival mass is 903 kg. Table 1 lists the 10 best asteroid combinations, ordered in terms of final mass. Table 2 lists the Keplerian orbital elements of each of the asteroids that appear in Table 1, in the J2000 heliocentric ecliptic frame.

Table 1: Ten best asteroid combinations for sample problem ranked by final mass.

Earth Dep. Date	Ast. 1	Ast. 2	Ast. 3	Leg 1 TOF (days)	Leg 2 TOF (days)	Leg 3 TOF (days)	Mf (kg)
03/01/2015	"2006 QQ56"	Medusa	Kostinsky	600	1600	1200	904
08/22/2016	"2006 QQ56"	Hertha	Telamon	600	1800	1200	856
03/29/2021	Apophis	Hertha	Pandarus	300	1800	1200	843
01/01/2015	"2002 AA29"	Medusa	Kostinsky	600	1700	1200	831
09/11/2018	"2006 QQ56"	Geisha	Kostinsky	600	1700	1200	826
08/28/2015	"2006 QQ56"	Geisha	Caltech	600	1700	1200	812
03/01/2015	"2004 FH"	Medusa	Kostinsky	500	1800	1200	807
09/06/2019	"2006 QQ56"	Medusa	Potomac	600	1800	1200	804
07/18/2017	"2006 QQ56"	Geisha	Potomac	600	1800	1200	798
12/5/2019	Apophis	Medusa	Potomac	500	1800	1200	787

Table 2: Orbital elements of asteroids in the J2000 heliocentric ecliptic frame.

Asteroid Name	Group #	semi-major axis (AU)	eccentricity	inclination (deg)	longitude of the asc. node (deg)	argument of periapsis (deg)
"2006 QQ56"	4	0.987	0.047	2.83	163.33	332.96
"2002 AA29"	4	0.994	0.013	10.74	106.47	100.61
"2004 FH"	4	0.818	0.289	0.021	296.18	31.32
Apophis	4	0.922	0.191	3.33	204.46	126.40
Geisha	2/3	2.24	0.193	5.66	78.34	299.88
Medusa	2/3	2.17	0.065	0.937	159.65	251.13
Hertha	2/3	2.43	0.207	2.31	343.90	340.04
Kostinsky	1	3.99	0.220	7.64	257.11	163.00
Telamon	1	5.17	0.108	6.09	341.01	111.19
Pandarus	1	5.17	0.068	1.85	179.86	37.74
Caltech	1	3.16	0.114	30.69	84.61	294.92
Potomac	1	3.98	0.181	11.40	137.51	332.82

2.5.1 Evaluation of Pruning Techniques on the Sample Problem

Each of the pruning metrics described in Section 2.2 are applied to the sample problem to determine their efficacy at reducing the size of the design space while maintaining a majority of the best solutions. For a majority of the pruning metrics, the metric value for each asteroid pairing is compared to the maximum low-thrust final mass for that pairing, for each leg of the trajectory. The correlation coefficient is then calculated between the metric in question and the low-thrust mass. Correlation coefficient varies between -1 and +1, where a value of -1 corresponds to perfect negative correlation and a value of +1 corresponds to a perfect positive correlation. The correlation coefficient between two random variables, *X* and *Y*, can be computed as follows, where *E(XY)* is the expected value of the product of *X* and *Y*, μ is the sample mean, and σ is the sample standard deviation:

$$\rho_{X,Y} = \frac{E(XY) - \mu_X \cdot \mu_Y}{\sigma_X \cdot \sigma_Y} \tag{15}$$

The first class of pruning metrics examined is ephemeris-based, beginning with visiting the asteroids in order of increasing semi-major axis. Applying this restriction to the sample problem reduces the number of asteroid sequences from 3072 to 512. This eliminates only two asteroid sequences with feasible solutions, both for the following order: Earth – Group 4 – Group 1 – Group 2/3. The maximum final mass for each of these two sequences, however, is only 608 kg and 524 kg, which ranks these solutions 59th and 105th out of the now 117 feasible asteroid sequences.

The next metric considered is the change in inclination between the orbits of two asteroids. Figure 20 plots the maximum final mass for each leg-by-leg asteroid pair, over the date range considered, as a function of the absolute value of the inclination change between the starting and ending body (no differences were found in the results if a distinction was made between positive and negative inclination changes). As will be true for all similar plots presented, only asteroid pairs that were actually analyzed are plotted. For example, four of the eight Group 4 asteroids yielded no feasible Leg 1 trajectories, and were therefore not considered in analyzing subsequent Leg 2 and Leg 3 trajectories. For Leg 1, because there were only eight possible pairs, additional Group 4 asteroids were randomly selected and analyzed in order to add more data points. Furthermore, any asteroid sequences that resulted in a maximum final mass of less than 500 kg were deemed infeasible and appear as 0 kg in the plots.

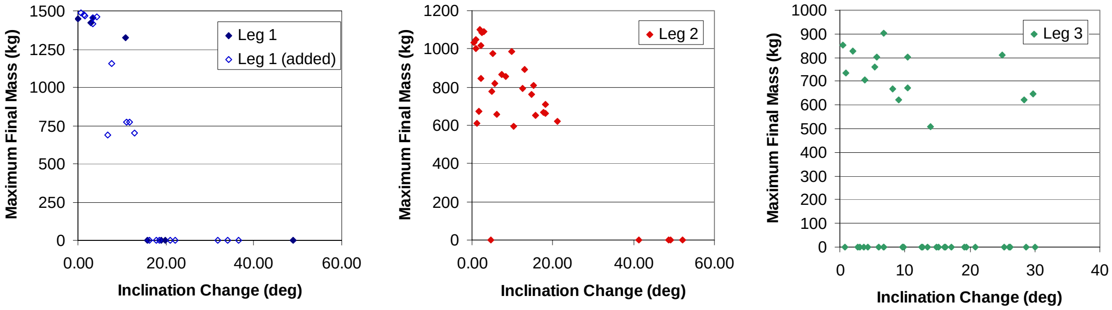

Figure 20: Maximum final mass for each asteroid sequence as a function of inclination change.

For Leg 1 and Leg 2, there is a perceptible correlation between final mass and inclination change, where smaller values of inclination change result in larger values of final mass. The correlation coefficient between inclination change and mass for these two trajectory legs is -0.823 and -0.803, respectively. Leg 3, however, yields a correlation coefficient of only -0.203, indicating a weak correlation between final mass and inclination change. The final masses for Leg 3 are dependent on both the goodness of that particular asteroid pair but also on the final mass and arrival date of the corresponding Leg 2 trajectory, which also depends on the final mass and arrival date from the corresponding Leg 1 trajectory. If all of the Leg 3 asteroid pairs are analyzed over a range of departure dates for a starting mass of 1500 kg, the correlation coefficient between final mass and inclination change is improved to -0.514. Therefore, a pruning metric such as inclination is much more effective for early trajectory legs, where the final mass is not nearly as dependent on upstream results.

Next, the longitude of the ascending node is included with inclination as a candidate pruning metric, using the two methods described in Section 2.2.1. The first method, which involves normalizing each metric and then combining them with weights, is less correlated to mass than using inclination change alone, even for a variety of weightings. The second method, which uses the angle between the angular momentum vectors (θ_{wedge}), results in correlation coefficients of -0.823, -0.790, and -0.423 for Legs 1, 2, and 3, respectively. If all of the Leg 3 asteroid pairs are again considered with a starting mass of 1500 kg, the correlation coefficient for the third leg is improved to -0.833. This increase in the correlation for the Leg 3 asteroid pairs indicates that θ_{wedge} is a good predictor of low-thrust mass, as long as the initial mass for all pairs is equal. When upstream information affects the initial mass and departure date, however, θ_{wedge} no longer approximates low-thrust mass as well. Figure 21 plots the maximum final mass for each of the asteroid pairs as a function of θ_{wedge}

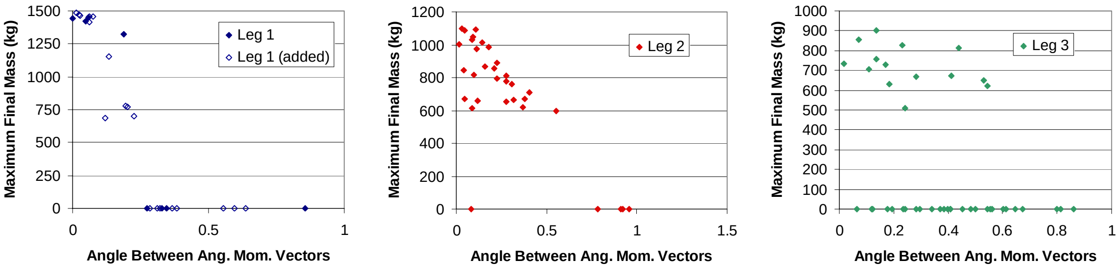

Figure 21: Maximum final mass for each asteroid combination as a function of wedge angle, the angle between the two angular momentum vectors.

None of the remaining ephemeris-based pruning metrics considered prove to be reliable for pruning the sample problem. The first method is to choose Group 1 asteroids with low energies – therefore, asteroids with the smallest values of semi-major axis. When the maximum final mass for each Group 1 asteroid is compared to its semi-major axis, the correlation coefficient is only -0.402. Furthermore, the Group 1 asteroid with the second largest semi-major axis (Pandarus, a = 5.17 AU) appears numerous times in the final set of feasible trajectories, including in the 3rd best overall trajectory. These results suggest that the semi-major axis of the final asteroid is not a reliable pruning metric. Finally, neither eccentricity nor the distance between the two orbits yield any meaningful correlation with low-thrust final mass.

The next pruning metric evaluated is the optimal, two-impulse, phase-free ΔV. Figure 22 plots the maximum final mass for each asteroid pair as a function of ΔV. The correlation coefficients for each leg are -0.785, -0.866, and -0.186, respectively. Therefore, the optimal two-impulse, phase-free ΔV appears to be a good predictor of low-thrust final mass for the first two legs. Once again, the correlation for Leg 3 is small. When the correlation coefficient is computed for the Leg 3, 1500 kg initial mass data, however, the resulting value is -0.871.

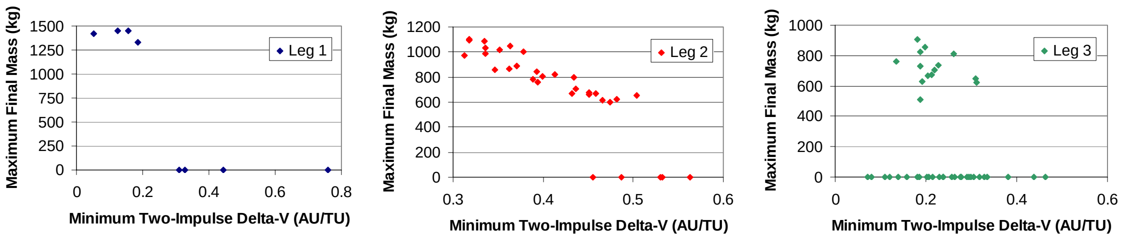

Figure 22: Maximum final mass for each asteroid pair as a function of the minimum, phase-free, two-impulse ΔV.

The final type of pruning metric considered takes phasing into account, in order to eliminate areas of the departure date, flight time, and/or stay time domain for particular asteroid combinations. First, two-impulse, multi-revolution Lambert solutions are calculated in an attempt to locate departure dates and flight times that yield low ΔV values for each asteroid combination. The resulting high-thrust ΔV values are then used to calculate equivalent mass values, using Equation 16, to better compare them with the low-thrust results.

$$M_f = M_i \exp\left[\frac{-\Delta V}{I_{sp} g_0}\right] \tag{16}$$

The first approach is to compare the two-impulse and low-thrust mass-optimal solutions for a given asteroid pair as a function of departure date, with a fixed time of flight. The two-impulse solution presented is the lowest ΔV Lambert solution over all possible number of revolutions.

Figure 23 compares the two-impulse and low-thrust mass-optimal solutions from Earth to 2006 QQ56 for a 600-day time of flight. While the peaks in the two solutions generally occur for the same departure dates, the two-impulse solution does not consistently represent the low-thrust solution over the entire range of dates. Figure 24 then plots the two-impulse and low-thrust mass-optimal solutions (both with initial masses of 1500 kg) for the Leg 3 transfer from Chicago – Kostinsky, with a 1200-day time of flight. In this case, the mass-optimal departure dates for the two solutions have very little correlation. These results are typical of the results generated for all of the asteroid pairs, when comparing the two-impulse and low thrust solutions for a fixed time of flight. Therefore, it can be concluded that the two-impulse mass-optimal solutions can not be used for pruning departure dates with fixed flight times.

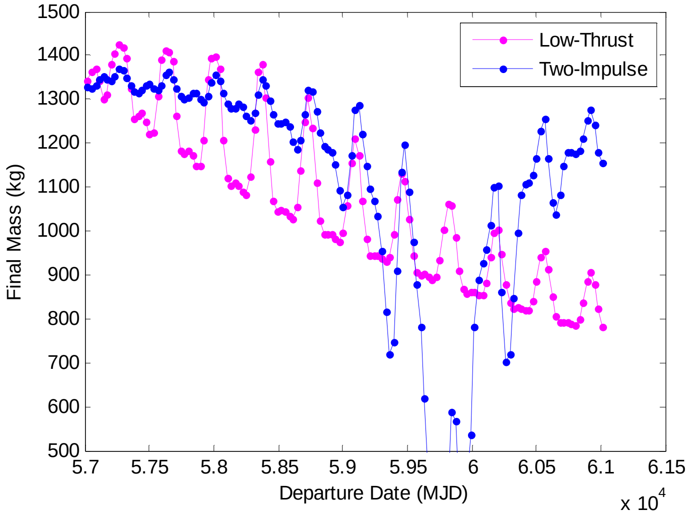

Figure 23: Comparison of two-impulse and low-thrust mass-optimal solutions for Earth – 2006 QQ56 with a 600-day time of flight.

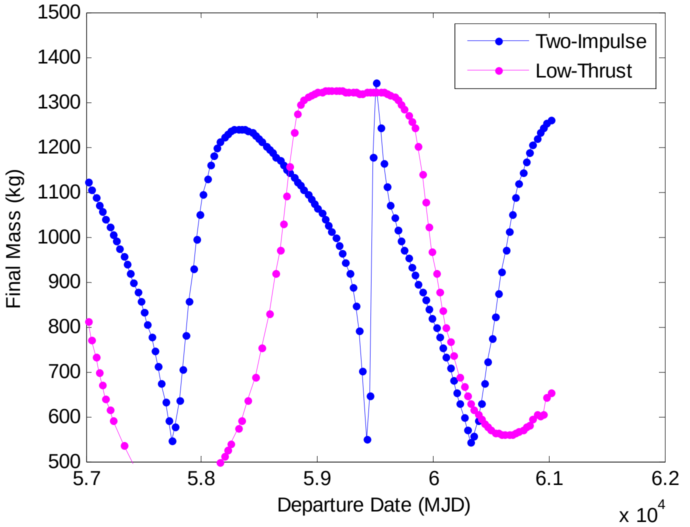

Figure 24: Comparison of two-impulse and low-thrust mass-optimal solutions for Chicago – Kostinsky with a 1200-day time of flight.

The next approach is to compare the two-impulse and low-thrust mass-optimal solutions for a given asteroid pair as a function of departure date only, without fixing the time of flight. Therefore, for each departure date, the mass-optimal solutions are calculated over the specified range of times of flight. Figure 25 plots the two-impulse and low-thrust mass-optimal solutions for the transfer from Earth to Apophis, with a free time of flight up to 600 days. Even without the fixed time of flight, the two solutions do not show good correlation. Again, these results are typical of the comparison between the two-impulse and low-thrust solutions for each of the asteroid pairings.

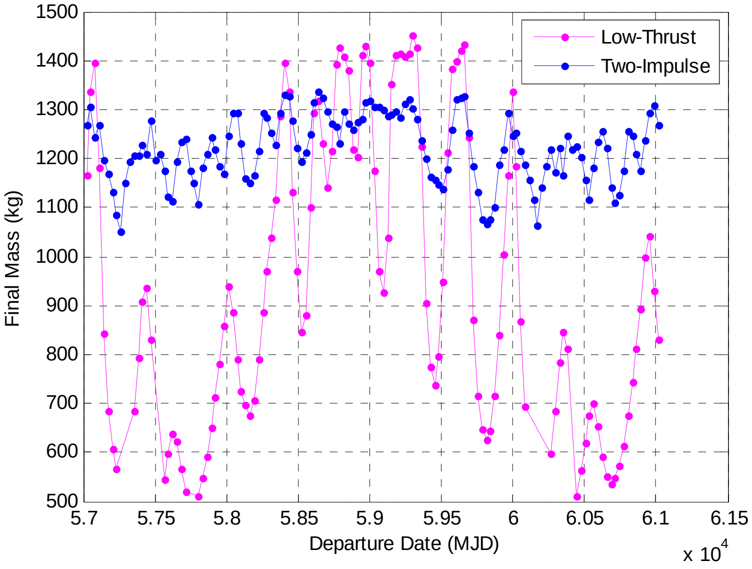

Figure 25: Comparison of two-impulse and low-thrust mass-optimal solutions for Earth – Apophis with a time of flight up to 600 days.

Therefore, it appears that using a two-impulse approximation to eliminate areas of the time domain is not feasible. In some cases, a good low-thrust solution could be missed when using a high-thrust screening strategy to choose good departure dates and flight times. The opposite, however, is also true, in that good high-thrust solutions can translate into poor low-thrust solutions.

The next phasing metric examined is to intercept the Group 1 asteroids near their perihelion. The previous asteroids and departure dates could be then chosen such that the spacecraft will arrive at the final asteroid at the specified time. Figure 26 plots the final mass of all the feasible trajectories from the sample problem as a function of the true anomaly of the last asteroid at arrival, where a true anomaly of zero degrees corresponds to perihelion. While there does appear to be a cluster of high final mass solutions in the vicinity of 30 degrees, most of these solutions are for a single asteroid sequence. Figure 27 then plots the arrival true anomaly for the maximum mass solution for the top twenty asteroid sequences. From this figure, it is apparent that there is no strong correlation between arrival true anomaly at the final asteroid and the resulting final mass.

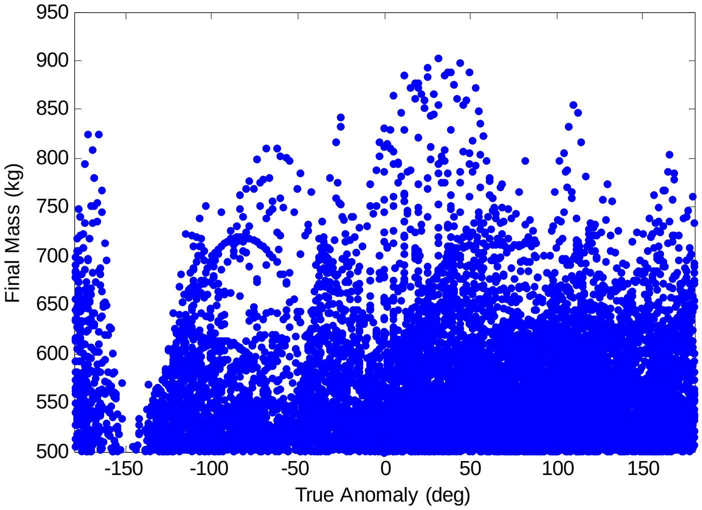

Figure 26: Final mass of all feasible trajectories as a function of arrival true anomaly at final asteroid.

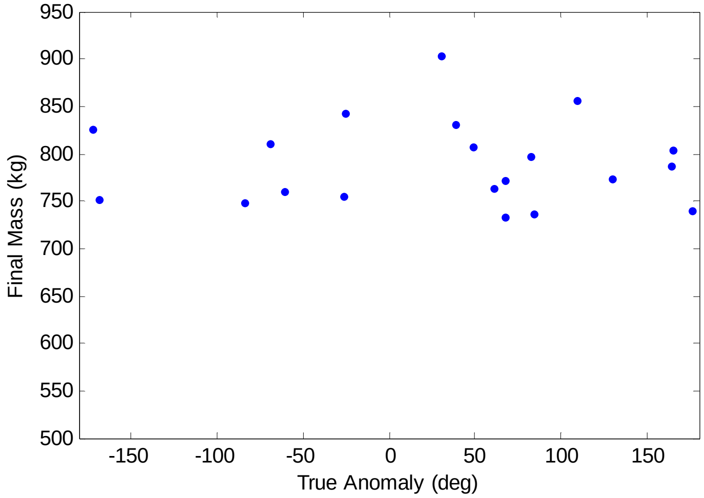

Figure 27: Maximum final mass for top twenty asteroid sequences as a function of arrival true anomaly at final asteroid.

Based on the evaluation of all the candidate pruning metrics on the sample problem, three metrics were chosen to be incorporated into the final methodology, based on their correlation between final mass and each metric's value: sorting by increasing semi-major axis, the angle between the angular momentum vectors (θ_{wedge}) and the phase-free, optimal, two-impulse ΔV. Each of these is used to eliminate asteroid sequences or pairs, but not to eliminate departure dates or flight times. Furthermore, for wedge angle and optimal two-impulse ΔV, the ability of each metric to act as a predictor of low-thrust mass decreases for each subsequent trajectory leg, due to the effect of the previous legs on the initial mass and departure date.

Increasing semi-major axis is applied first to all asteroid sequences, since it is a binary metric. Either an asteroid sequence meets this criteria or it does not. Those sequences that do not meet the criteria are eliminated from the design space. For the sample problem, this step reduces the number of asteroid sequences from 3072 to 512 (a factor of 6). Three different techniques were then evaluated for applying the remaining two pruning metrics. First, all of the metrics are combined for each full asteroid sequence i, as illustrated in Equation 17, where a small value of W_i is better:

$$W_i = 0.5 \cdot \frac{\sum_i^N \theta_{wedge,Legi}}{\max\left(\sum_i^N \theta_{wedge,Legi}\right)} + 0.5 \cdot \frac{\sum_i^N \Delta V_{opt,Legi}}{\max\left(\sum_i^N \Delta V_{opt,Legi}\right)} \tag{17}$$

Each pruning metric is summed over all of the legs in the sequence, where each leg represents each asteroid pair in the sequence, and then normalized to fall between zero and one. The two summed metrics are then combined with an equal weighting. By ranking all of the asteroid sequences using this single value, a user-defined percentage of sequences can be eliminated.

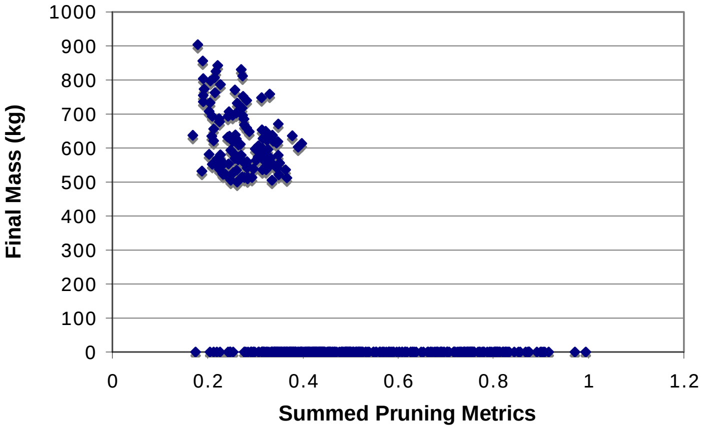

Figure 28: Final mass as a function of the summed pruning metric (Equation 17) for each asteroid sequence remaining in the small sample problem.

Figure 28 plots W_i from Equation 17 against the corresponding optimal low-thrust final mass for each asteroid sequence in the small sample problem. In order to keep all feasible solutions (all solutions with a final mass greater than 500 kg) in the design space, up to 55% of the asteroid sequences can be eliminated. In order to keep all of the top ten asteroid sequences in the design space, up to 85% of the sequences can be eliminated. Finally, to keep just the optimum solution in the design space, up to 99% of the sequences can be eliminated. The main drawback to this method, however, is that each pruning metric must be calculated for each possible asteroid pair in the design space. While this is not a problem for the small sample problem, it will become computationally intensive for significantly larger problems.

Second, each pruning metric is still summed over all the legs for each sequence, but each metric is applied individually, in sequence. Therefore, all of the asteroid sequences are first ranked as a function of θ_{wedge}, summed over each of the legs. A user-defined percentage of sequences is then eliminated from the design space. Next, the remaining sequences are ranked as a function of the optimal phase-free, two-impulse ΔV, again summed over all the legs. A user-defined percentage of sequences is then eliminated from the design space. This approach requires less computation time, since the optimal ΔV needs to be computed for a smaller number of sequences. Figure 29 and Figure 30 plot the optimal low-thrust final mass as a function of θ_{wedge} and optimal phase-free, two-impulse ΔV, summed over all the legs for each asteroid sequence in the sample problem.

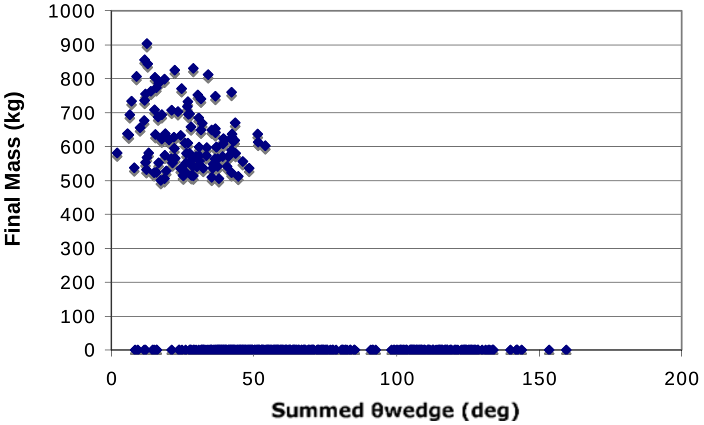

Figure 29: Final mass as a function of θ_{wedge}, summed over all legs for each asteroid sequence remaining in the small sample problem.

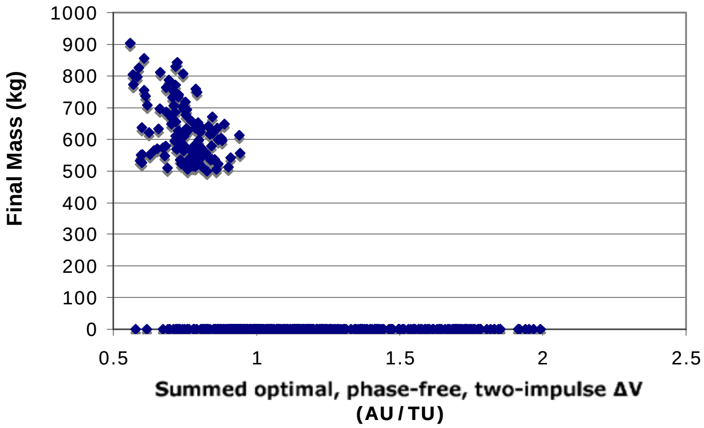

Figure 30: Final mass as a function of optimal, phase-free, two-impulse ΔV summed over all legs for each asteroid sequence remaining in the small sample problem.

Assuming the same percentage is eliminated for each pruning metric, the largest percentage that can be eliminated for each metric to keep all feasible solutions in the design space is 36%. This percentage results in reducing the overall number of asteroid sequences from 512 to 201. In order to keep all of the top ten asteroid sequences in the design space, up to 64% can be eliminated for each metric. This results in 66 remaining asteroid sequences. Finally, to keep the optimum solution in the design space, up to 93% can be eliminated for each metric. This results in just one asteroid sequence remaining in the design space, which is the optimal solution.

The last pruning approach considered applies the metrics sequentially leg-by-leg. First, θ_{wedge} is used to eliminate a user-defined percentage of Leg 1 (Earth – 1st asteroid) asteroid pairs. Full asteroid sequences that include this pair are eliminated from the design space. θ_{wedge} is then used to eliminate a user-defined percentage of Leg 2 (1st asteroid – 2nd asteroid) asteroid pairs, and is finally used to eliminate a user-defined percentage of Leg 3 (2nd asteroid – 3rd asteroid) asteroid pairs. Next, the optimal, two-impulse, phase-free ΔV is applied to the reduced design space using the same leg-by-leg approach. This approach requires the least computation time, since the design space is reduced one metric at a time, and subsequent metrics need only be calculated for the reduced-size problem. Furthermore, the percentages eliminated on each leg can be decreased for legs further from Earth. As was shown previously, the pruning metrics are good approximations of low-thrust final mass for the first two legs; the correlations drop significantly for the third leg, however. Therefore, the percentage eliminated for Leg 3 can be chosen to be smaller than for the first two legs. Table 3 summarizes this third pruning approach as applied to the sample problem for a range of percentages. Using this approach, the percentages chosen for each leg can be tailored to account for the reduced ability of each pruning metric to approximate low-thrust mass for later legs. As shown in Table 3, if the percentage of asteroids eliminated for Leg 1, Leg 2, and Leg 3, respectively, is chosen to be 10%, 10%, and 10%, no feasible solutions are eliminated. Percentages of 30%, 25%, and 15% result in feasible solutions being eliminated, but none of the asteroid sequences in the top ten being eliminated. For percentages higher than these values, however, asteroid sequences in the top ten begin to be eliminated.

Table 3: Performance of pruning method for a range of elimination percentiles.

Percent Eliminated Leg 1	Percent Eliminated Leg 2	Percent Eliminated Leg 3	# Remaining Asteroid Sequences	# Feasible Sequences Eliminated	Max. Mass of Eliminated Sequnces (kg)
10%	10%	0%	424	0	0
10%	10%	10%	368	0	0
20%	15%	10%	240	4	653
25%	20%	10%	179	7	653
30%	25%	15%	131	19	653
30%	25%	20%	118	26	812

In comparing the three approaches, the goal is to reduce the size of the design space as much as possible while keeping a majority of the best asteroid sequences in the design space. For the sample problem, each approach is compared based on its ability to keep the top ten asteroid sequences in the design space. Furthermore, the number of asteroid pairs for which the pruning metrics must be computed is important to consider, particularly when scaling the approach to larger problems. In the first approach, up to

85% of the asteroid sequences could be eliminated without eliminating any of the ten best sequences. This results in 73 remaining asteroid combinations. This approach, however, requires calculating each metric for all 136 distinct asteroid pairs. Applying the second approach to the sample problem, up to 87% of the asteroid sequences can be eliminated, resulting in 66 remaining asteroid sequences. Using this approach, however, θ_{wedge} must be calculated for all 136 asteroid pairs, but the optimal two-impulse, phase-free ΔV must be calculated for only 62 asteroid pairs. Finally, the third approach can eliminate up to 74% of asteroid sequences, leaving 131 remaining sequences. θ_{wedge} must be calculated for 104 asteroid pairs, while the optimal, two-impulse, phase-free ΔV must be calculated for 77 asteroid pairs. Therefore, all three approaches are comparable in that some require calculating the pruning metrics for more asteroid pairs but enable a greater reduction in the design space, and vice versa. The third approach is selected because for larger problems, a greater reduction in the design space will realize an even greater savings in the number of pruning metrics that must be calculated. This is an important consideration, when the number of asteroid sequences is in the millions or billions. When applying the pruning approach to a problem, the percent of asteroid pairs eliminated in each leg for each pruning metric is chosen based on two factors. First is the desired reduction in the design space – the smaller the resulting number of asteroid sequences, the larger the percent eliminated must be. The more asteroid combinations that are eliminated, however, the greater chance there is of eliminating some of the best sequences. Therefore, a user would choose a desired reduction in the number of asteroid sequences and then find suitable values for the percentages based on that desired final value.

Table 3 illustrates how the percentages can be incrementally increased until the desired number of asteroid sequences remains. For a general problem, however, the number of good sequences eliminated is of course unknown. Table 4 summarizes the results when the following percentage values are applied to the sample problem: 30%, 25%, and 15% for Leg 1, Leg 2, and Leg 3, respectively. The number of asteroid sequences is reduced from 3072 to 131, and only 19 feasible sequences were eliminated. The best sequence eliminated has a final mass of 653 kg, which ranks 37th. Therefore, none of the top solutions are eliminated and the design space is reduced by a factor of 23.

Table 4: Pruning methodology applied to sample problem.

Pruning Metric	Trajectory Leg	% Sequences Eliminated	# Sequences Eliminated
$a_i < a_{i+1}$	All	N/A	2560
θ_{wedge}	Leg 1	30%	128
θ_{wedge}	Leg 2	25%	96
θ_{wedge}	Leg 3	15%	42
ΔV_{opt}	Leg 1	30%	41
ΔV_{opt}	Leg 2	25%	46
ΔV_{opt}	Leg 3	15%	28

2.5.2 Evaluation of Global Optimization Methods on the Sample Problem

The global optimization methods under consideration are all tested on the full sample problem (instead of the pruned design space), in order to evaluate them on the largest possible design space. For the genetic algorithm, two approaches are considered, as was presented in Figure 11 and Figure 12: (1) a single-level approach, where all global design variables are solved for simultaneously, and (2) a two-level approach, where an outer loop optimizer solves for the asteroid sequence and an inner loop optimizer solves for the time variables. The branch-and-bound is used solely in the two-level configuration, with the branch-and-bound method solving for the asteroid sequence and a genetic algorithm solving for the time variables.

The basic genetic algorithm is applied to the sample problem in a variety of ways. In each case, the following settings must be chosen:

- Population size
- Number of bits for each variable – controls the resolution of the discretization
- Stall generations – number of generations with no change in the best ever objective function after which the genetic algorithm is considered converged
- Maximum number of generations
- Tournament size – number of individuals that participate in each round of the tournament selection
- Probability of crossover
- Probability of mutation

Different values of these settings are tested in each case. For each set of values, ten runs of the genetic algorithm are carried out, and the number of runs that yield the optimum solution are recorded. This value will be referred to as the solution success percentage, and is used as an indicator of the performance of the genetic algorithm. In evaluating the genetic algorithm on the sample problem, the GA functions by using a table look-up of the sample problem data, as opposed to directly calling MALTO. Therefore, for each asteroid sequence, departure date, and set of flight times, the corresponding optimum low-thrust final mass is looked up based on the solution set from the sample problem. In order to achieve this, the discretization for the GA variables is set to equal the discretization used in solving the sample problem. The table look-up significantly decreases the run time of the GA, allowing more variations to be evaluated.

The first version of the genetic algorithm attempts to solve for all of the global design variables at once: asteroid sequence, Earth departure date, and times of flight. The asteroid sequence is handled in two different ways. First, three variables are used to represent the three asteroids in the sequence: asteroid 1, asteroid 2, and asteroid 3. Second, a single variable is used to represent the asteroid sequence number, where all of the possible asteroid sequences are ordered and assigned an index number. Including

Earth departure date and three times of flight, the first method requires seven total design variables, while the second case requires five design variables. Neither case, however, is successful at reliably finding the global optimum solution. In the first case, the highest solution success percentage was only 10%. Therefore, only 1 of 10 runs of the GA yielded the global optimum solution. Four of the ten cases, however, found the correct asteroid sequence, but not the optimal departure dates and flight times. In the second configuration of the asteroid design variables, the highest solution success percentage was again only 10%. The first case required on average 65 generations and 889 function calls, while the second case required on average 62 generations and 821 function calls. Table 5 lists the values of the GA settings that were used to obtain the stated results.

Table 5: Settings for the single-level genetic algorithm.

GA Setting	Value, Case 1	Value, Case 2
Design Variables	Ast. 1, Ast. 2, Ast. 3, Earth dep. date, TOF 1, TOF 2, TOF 3	Asteroid sequence, Earth dep. date, TOF 1, TOF 2, TOF 3
Population Size	100	100
Max. Generations	200	200
Stall Generations	50	50
Tournament Size	4	4
Crossover Proability	0.8	0.8
Mutation Probablity	0.1	0.1

The second version of the genetic algorithm uses a two-level approach, where the outer loop GA solves for the asteroid sequence and the inner loop GA solves for the departure date and flight times for each particular sequence. This approach has a much higher success rate at finding the global optimum. First, the two loops were examined separately, beginning with the outer loop. The design variables for the outer loop are indices representing asteroid 1, asteroid 2, and asteroid 3. Again, a table look-up of the maximum final mass for each asteroid sequence was used to decrease the run time. The solution success percentage in this case was 60%, using the settings listed in Table 6. An average of 27 generations was required with 111 total function calls. The number of generations being so close to the stall generations indicates that the optimum solution is being found either in the random initial population or in the first couple of generations. Therefore, the number of stall generations required for convergence could be decreased in order to decrease the number of required function calls.

Table 6: Settings for inner and outer loop genetic algorithm.

GA Setting	Value, Outer Loop	Value, Inner Loop
Design Variables	Ast. 1, Ast. 2, Ast. 3	Earth dep. date, TOF 1, TOF 2, TOF 3
Population Size	50	50
Stall Generations	25	25
Tournament Size	4	4
Crossover Proability	0.8	0.8
Mutation Probablity	0.1	0.1

Next, the inner loop portion of the GA was examined on the top ten asteroid sequences (from Table 1), using the settings listed in Table 6. Again, the GA was run ten times for each asteroid sequence. Overall, the solution success rate was 33%, requiring an average of 33 generations and 220 function calls. In order to try and increase the success rate of the genetic algorithm, the population size was increased to 100 and the GA was again applied to the same ten asteroid sequences. Because a larger population size increases the number of function calls, the number of stall generations required for convergence was decreased to 15. As a result, the solution success rate improved to 56%, and the average number of generations required decreased to 21. The average number of function calls, however, increased to 395 per run of the genetic algorithm. These two cases illustrate the important tradeoff between the performance of the genetic algorithm and the required number of function calls. Finally, the outer and inner loop genetic algorithms were combined and applied to the overall sample problem, using the settings in Table 7. As before, for each case, the outer loop genetic algorithm was run ten times. For each function call, the outer loop genetic algorithm calls the inner loop genetic algorithm, which determines the optimal departure date and flight times for the particular asteroid sequence chosen by the outer loop optimizer. As seen above, the success rate of the inner loop GA is less than 100%, which means that there is no guarantee that a single run of the GA will yield the optimum solution. Therefore, each time the inner loop GA is called, it is actually run several times and the best solution is then sent back to the outer loop GA. The first case is based on the baseline values of the GA settings from Table 6. For each function call of the outer loop GA, the inner loop GA is run 3 times. The resulting solution success rate (the percentage of time the outer loop locates the optimal asteroid sequence, along with the optimal departure date and flight times found by the inner loop) is 60%. The outer loop ran for an average of 30 generations, and required on average 181 function calls to the inner loop optimizer. Although a table look-up of the sample problem data was still used, the equivalent number of MALTO function calls would be nearly 78,000.

Table 7: Settings and performance of the multi-level genetic algorithm.

	GA Setting	Case 1	Case 2	Case 3	Case 4
Outer Loop	Population Size	50	50	50	100
	Stall Generations	25	10	10	10
	Tournament Size	4	4	4	4
	Crossover Probability	0.8	0.8	0.8	0.8
	Mutation Probability	0.1	0.1	0.1	0.1
Number of runs of inner loop per function call:		3	3	5	3
Inner Loop	Population Size	50	50	50	100
	Stall Generations	25	10	10	10
	Tournament Size	4	4	4	4
	Crossover Probability	0.8	0.8	0.8	0.8
	Mutation Probability	0.1	0.1	0.1	0.1
Solution Success Rate		60%	40%	60%	80%
Avg. Number of Generations		30	13	12	11
Avg. Number of Func. Calls		181	144	146	270
Avg. Number of MALTO Calls		77775	43341	68504	140466

In an attempt to reduce the number of required function calls, the number of stall generations required for convergence was lowered in Case 2. While the number of outer loop function calls was reduced to 144 and the number of MALTO function calls was reduced to just over 43,000, the solution success rate also decreased to 40%. Therefore, more runs of the genetic algorithm would be required to find the optimal solution with the same confidence as for Case 1. Next, the number of runs of the inner loop GA per function call was increased to five, keeping the remaining settings the same as from Case 2. This resulted in raising the success rate of the GA back to 60%, while reducing the number of required MALTO calls from Case 1. Finally, the population size was increased to 100 for both the inner and outer loops to try to raise the success rate above 60%. While the success rate was increased to 80%, the number of calls to MALTO also increased to over 140,000. The results presented in Table 7 indicate that the multi-level genetic algorithm is successful at locating the optimal solution more than half the time (depending on the settings chosen). The number of MALTO runs required, however, makes this method prohibitive, particularly as the problem size increases. Each end-to-end MALTO run takes on the order of 10 seconds, which would require anywhere from 5 days (for Case 2) to 16 days (for Case 4) for a single run of the genetic algorithm on a single processor.

Next, the branch-and-bound method presented in Section 2.3.2 is evaluated as the outer-loop optimizer. As aforementioned, it relies on the ability of the two-impulse approximation to act as an upper bound for the optimal low-thrust solution. For each trajectory leg (Earth – Asteroid 1, Earth – Asteroid 1 – Asteroid 2, and Earth – Asteroid 1 – Asteroid 2 – Asteroid 3), the mass-optimal two-impulse solution is compared to the mass-optimal low-thrust solution for each possible asteroid sequence. The two-impulse optimal solutions represent the minimum ΔV solutions over all possible number of revolutions, using the same departure date range and times of flight as the sample problem. The optimal solutions are found using a grid search. The corresponding mass is calculated using the same specific impulse and initial mass as for the low-thrust problem. The results are plotted in Figure 31, Figure 32, and Figure 33, sorted by the two-impulse final mass.

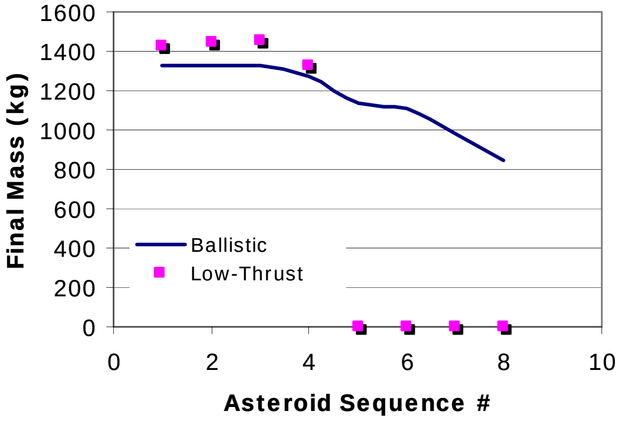

Figure 31: Comparison of mass-optimal low-thrust and two-impulse solutions for all Earth – Asteroid 1 sequences.

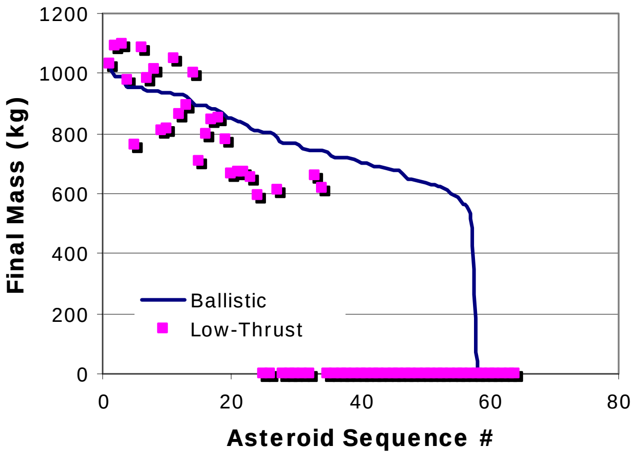

Figure 32: Comparison of mass-optimal low-thrust and two impulse solutions for all Earth – Asteroid 1 – Asteroid 2 sequences.

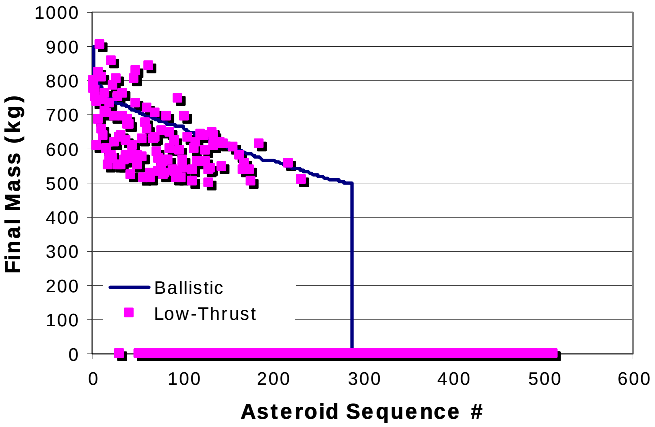

Figure 33: Comparison of mass-optimal low-thrust and two-impulse solutions for all Earth – Asteroid 1 – Asteroid 2 – Asteroid 3 sequences.

The figures above indicate that as calculated, the two-impulse solutions do not provide a reliable upper bound. However, if the two-impulse solutions are shifted slightly, then they could provide an upper bound for the low-thrust solutions. For the sample problem, this would require a multiplication factor of 1.09, 1.14, and 1.21 for Leg 1, Leg 1 + Leg 2, and Leg 1 + Leg 2 + Leg 3, respectively. Of course, on a larger problem, this multiplication factor is not known a priori. Choosing a value of this parameter that is too small will result in a number of good sequences being pruned out during the branch-and-bound procedure. Choosing a value that is too large, however, will result in an unnecessarily large number of low-thrust trajectory optimizations to be carried out because very few asteroid sequences will be pruned. The solution is to iterate on the best value of this multiplication factor during the branch-and-bound procedure, as will be illustrated on the sample problem.

Another important aspect of the branch-and-bound method is the order in which the branches are evaluated. Choosing this order in an intelligent fashion can significantly reduce the number of asteroid sequences that require low-thrust trajectory optimization. If an asteroid sequence with a high final mass is evaluated early in the process, then more branches will be pruned out than if all low mass sequences are evaluated initially. One way to choose the order in which to evaluate the various branches is based on the pruning metrics which were calculated in the previous phase of the methodology. For each full asteroid sequence, the pruning metrics can be combined as follows, where the two pruning metrics are weighted equally:

$$W_i = 0.5 \cdot \left(\frac{\sum_i \theta_{\text{wedge,Legi}}}{\max\left(\theta_{\text{wedge}}\right)} \right) + 0.5 \cdot \left(\frac{\sum_i \Delta V_{\text{Legi}}}{\max\left(\Delta V\right)} \right) \tag{18}$$

For each asteroid sequence *i*, W_i will fall between 0 and 1, with smaller numbers being better. The first sequence evaluated, which is used to set the initial lower bound, is chosen based on the results of Equation 18. The branches are then evaluated in sequential order based on the weighted combination of the pruning metrics.

The first example applies the branch-and-bound method without any multiplier on the two-impulse solutions. For the sample problem, the first asteroid sequence evaluated is Earth – 2006 QQ56 – Medusa – Pandarus. The resulting optimal low-thrust solution is 638 kg. This becomes the current lower bound. The next asteroid sequence, based on the pruning metric rank, is Earth – 2006 QQ56 – Chicago – Pandarus. First, the optimal two-impulse solution is calculated for the first leg: Earth – 2006 QQ56, yielding an optimal final mass of 1371 kg. Because this is greater than the current lower bound, the next level down must be calculated. The resulting two-impulse optimal solution for Earth – 2006 QQ56 – Chicago is 824 kg. Again, this branch can not be pruned. Finally, the two-impulse optimal for the entire sequence, Earth – 2006 QQ56 – Chicago – Pandarus, is 736 kg. Because this is still greater than the current lower bound, the optimal low-thrust solution for this asteroid sequence must be calculated. This sequence does not yield a feasible solution (M_f < 500 kg), and therefore, the previous lower bound remains the best known solution thus far. The third ranking asteroid sequence is Earth – 2006 QQ56 – Medusa – Kostinsky. The same process is carried out, and again the optimal two-impulse solution for the entire asteroid sequence is not sufficient to prune out that branch. The resulting low-thrust optimal solution is 904 kg, which happens to be the optimal solution for the sample problem. Of course, for a general problem, the optimal solution is not known, so all of the branches have to be evaluated.

When the entire branch-and-bound method is completed, the low-thrust optimum of only 4 asteroid sequences has to be computed. Of the 8 Earth – Asteroid 1 two-impulse solutions computed, 1 resulted in that branch being pruned. Of the remaining 56 Earth – Asteroid 1 – Asteroid 2 trajectories, 41 resulted in that branch being pruned. That left 120 two-impulse optimizations of the full sequence (Earth – Asteroid 1 – Asteroid 2 – Asteroid 3), of which only 4 required low-thrust optimizations.

The results of this first iteration of the branch-and-bound method can then be used to revise the multiplier on the two-impulse solutions. The low-thrust optimum solutions that were calculated can be compared to the corresponding two-impulse solutions. Of these asteroid sequences, only one yields a low-thrust final mass that is greater than the impulsive solution. The required increase in the impulsive solution sets the multiplication factor at 1.15. The branch-and-bound procedure can then be repeated using this new multiplication factor on all of the impulsive solutions. Now, none of the Earth – Asteroid 1 branches can be pruned out, requiring the optimal impulsive solution to be found on 8 additional Earth – Asteroid 1 – Asteroid 2 trajectories. Furthermore, a total of 224 two-impulse optimizations are now required for the full sequence. Finally, 10 additional low-thrust trajectory optimizations must be carried out, for a total of 14 over the two iterations of the branch-and-bound procedure. All of the new low-thrust solutions are less than their corresponding impulsive solutions; therefore, no more iteration on the impulsive multiplier is required. After two iterations, not only was the global optimum solution found, but 5 of the top 10 asteroid sequences were also located. Figure 34 plots all of the feasible asteroid sequences in the sample problem in blue, sorted by final mass. All of the sequences identified by the branch-and-bound algorithm are identified in pink.

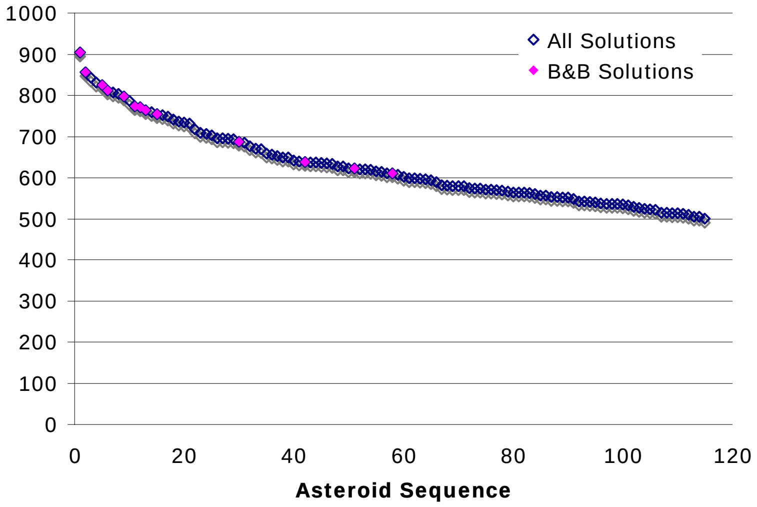

Figure 34: Asteroid sequences identified by applying branch-and-bound algorithm to small sample problem.

If the optimum solution is found early in the branch-and-bound algorithm, as is the case for the sample problem, a number of good solutions will be pruned out due to the high value of the lower bound. Therefore, once the branch-and-bound algorithm has completed and found the optimal solution, additional low-thrust optimizations can be carried out to find additional good solutions. A smaller value of the lower bound can be selected, and all asteroid sequences with two-impulse optimal solutions greater than that bound (using the final impulsive multiplier value of 1.15) can be passed to the genetic algorithm to calculate their low-thrust optimum. For example, if additional solutions greater than 800 kg were desired, there are 65 additional asteroid sequences whose two-impulse optimum final mass (multiplied by 1.15) is greater than 800 kg. If the low-thrust optimum is calculated for each of these sequences, then all of the top ten asteroid sequences are found.

As was seen above, a multiplier of 1.21 was required to fully bound the low-thrust solutions with the impulsive solutions, while the iterative branch-and-bound method resulted in a final multiplier of only 1.15. Of course, there is no guarantee that the correct multiplier has been obtained without evaluating the low-thrust optimal for all branches of the search tree, but this would defeat the purpose of using the impulsive approximation as a bound. For the sample problem, only 3 full asteroid sequences were pruned out incorrectly using the multiplier value of 1.15. In general, to increase the confidence that a good multiplier has been obtained, and therefore the optimal solution has been found, several additional low-thrust optimal solutions can be computed after the branch-and-bound procedure has been completed. This not only adds more data points from which to iterate on the impulsive multiplier, if necessary, it also adds more good asteroid sequences to the final solution. Therefore, after the branch-and-bound was completed on the sample problem, the low-thrust optimal was found for the ten best impulsive solutions. Although this did not result in the multiplier value being changed, it did find an additional asteroid sequence that falls in the top ten for the sample problem.

The same procedure can then be repeated, without using the pruning metrics to rank the asteroid sequences, in order to compare the number of low-thrust optimizations that must be carried out. In this case, the order in which they are evaluated is based solely on proceeding in numerical order using the asteroid's identification numbers. In just the first iteration (impulsive multiplier equal to 1), 268 low-thrust trajectory optimizations must be carried out, as compared to just fourteen required low-thrust optimizations over two iterations using the previous technique. From this example, it is clear that evaluating the branches in a sequence based on the pruning metric ranking saves significant computation time. Evaluating the branch-and-bound branches in this manner reduces the number of asteroid sequences that require low-thrust optimization because a large value of the lower bound (best known low-thrust solution) is set early in the algorithm, thereby pruning out branches of the search tree with lower values of two-impulse optima. Figure 35 plots the optimal low-thrust solutions for the 512 asteroid sequences in the branch-and-bound tree, as a function of the normalized sum of the pruning metrics. This represents the order in which these sequences would be evaluated during the branch-and-bound algorithm. As expected, most of the better solutions occur for higher ranked sequences, resulting in fewer required low-thrust optimizations than if a random order were used in the branch-and-bound algorithm.

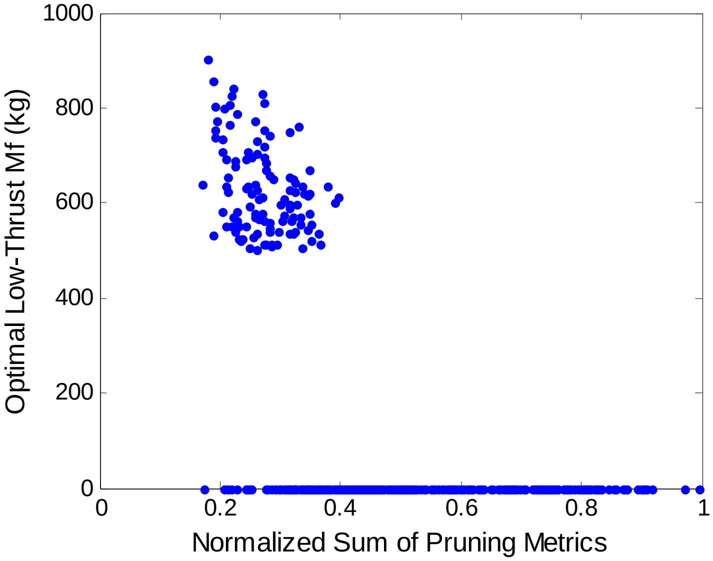

Figure 35: Low-thrust optima as a function of the normalized sum of the pruning metrics (branch-and-bound ranking), for the small sample problem.

The branch-and-bound lower bound can also be selected intelligently. Up to this point, the branch-and-bound algorithm was initiated without a known low-thrust solution and therefore a lower bound of zero. This lower bound is then incrementally increased as better low-thrust solutions are found. For the small sample problem, the optimum sequence is the third ranked sequence and also the third sequence evaluated in low-thrust. While it is expected that better solutions will be found early for larger problems, there may still be a relatively large number of poor sequences that must be evaluated in low-thrust before the better solutions are found and the lower bound is correspondingly increased. If the branch-and-bound algorithm were started with a non-zero lower bound, however, a lot of these low-thrust optimizations of poor sequences could be eliminated.

The initial lower bound is a user-defined value, based on estimating the optimal value of the objective function based on the physics of the problem being solved. An iterative approach could be utilized in setting this initial lower bound. If, for example, an initial lower bound is chosen and the best low-thrust solution found at the completion of the branch-and-bound algorithm is less than this lower bound, the value chosen was potentially set too high. The initial lower bound could then be decreased to match the best low-thrust solution found in order to capture additional good solutions (the sequences that already required low-thrust optimization would not have to be re-evaluated).

Another important observation can be made from Figure 35, which could also serve to reduce the number of low-thrust optimizations required. As was previously highlighted, the best solutions are generally evaluated early in the branch-and-bound algorithm, as a result of using the pruning metrics to prioritize the order of the asteroid sequences. For the sample problem, the top ten known solutions all rank in the first 73 asteroid sequences (out of 512) – this represents the top 14% in terms of the branch-and-bound ranking. Furthermore, seven of the top ten solutions rank within the top 5% of all ranked sequences. Therefore, in solving a large problem where time and computational resources are limited, it is also possible to consider only analyzing some user-defined percentage of the branch-and-bound tree (based on computational time constraints) and still locate a majority of the good solutions.

While these additional modifications to the branch-and-bound algorithm are not applicable on the small sample problem, they will be evaluated on the larger problems considered in Chapter 5 to determine if these same trends hold and how large a reduction in the number of required low-thrust optimizations can be achieved as a function of the number of good solutions found.

CHAPTER III

OVERVIEW OF METHODOLOGY

The final methodology combines a three-level heuristic pruning step, which quickly reduces the size of the design space, with a multi-level global optimization step, which locates a suite of good solutions. Each of the steps of the methodology is presented here, followed by a discussion of the assumptions and scope.

3.1 Overview of Methodology

The pruning phase of the methodology applies three pruning metrics, in order of required computation time:

(1) $a_i \leq a_{i+1}$ – keep only asteroid sequences where the semi-major axis of the asteroids increases from one asteroid to the next

(2) θ_{wedge} – angle between the angular momentum vectors of asteroid pairs

- *a.* θ_{wedge} is computed for all Leg 1 (Earth – 1st asteroid) asteroid pairs; the worst k_1 percent is eliminated from the design space.
- *b.* θ_{wedge} is computed for all remaining Leg 2 (1st asteroid – 2nd asteroid) asteroid pairs; the worst k_2 percent is eliminated from the design space.
- *c.* θ_{wedge} is computed for all remaining Leg 3 (2nd asteroid – 3rd asteroid) asteroid pairs; the worst k_3 percent is eliminated from the design space.
- *d.* θ_{wedge} is computed for all remaining Leg 4 (3rd asteroid – 4th asteroid) asteroid pairs; the worst k_3 percent is eliminated from the design space.

(3) ΔV_{opt} – optimal two-impulse, phase-free delta-V between asteroid pairs

- *a.* ΔV_{opt} is computed for all remaining Leg 1 (Earth – 1st asteroid) asteroid pairs; the worst k_1 percent is eliminated from the design space.

b. ΔV_{opt} is computed for all remaining Leg 2 (1st asteroid – 2nd asteroid) asteroid pairs; the worst k_2 percent is eliminated from the design space.

c. ΔV_{opt} is computed for all remaining Leg 3 (2nd asteroid – 3rd asteroid) asteroid pairs; the worst k_3 percent is eliminated from the design space.

d. ΔV_{opt} is computed for all remaining Leg 4 (3rd asteroid – 4th asteroid) asteroid pairs; the worst k_3 percent is eliminated from the design space.

The percentages eliminated on each leg (k_1, k_2, k_3, k_4,) are user-defined constants, which are chosen based on the desired reduction in the design space. As a general rule of thumb, two considerations should be made when choosing the percentages. First, the percentages should decrease for each subsequent leg, particularly for legs beyond Leg 1 (k_1) and Leg 2 (k_2). A 5% to 10% decrease from leg-to-leg is appropriate. Second, the maximum pruning percentage (k_1, Leg 1 percentage) should be kept as small as possible, while still achieving the desired overall reduction in the design space and decreasing the percentages for subsequent legs. The more asteroid sequences that are eliminated, the greater the chances are of eliminating a large number of good solutions from the design space. Therefore, these values must be chosen to balance the required computation time of solving a larger problem with the risk of eliminating good solutions from the design space.

Once the design space has been pruned, and the desired number of remaining asteroid sequences has been achieved, the multi-level global optimization scheme is applied to the reduced problem. This approach combines a branch-and-bound algorithm, which solves for the optimal asteroid sequence, with a genetic algorithm, which solves for the optimal departure date, times of flight, and stay times for a given sequence. Additionally, the genetic algorithm is linked with MALTO, which maximizes final mass for a given set of global variables. The global optimization scheme is outlined as follows:

(1) All asteroid sequences are ranked as a function of the normalized sum of the pruning metrics. In order to set the initial value of the lower bound, one of the following three techniques can be employed:

  a. For the top ranked sequence, the genetic algorithm is run N_{GA} times to solve for the optimum low-thrust solution. The optimal final mass of the asteroid sequence is set as the current lower bound on the design space.

  b. If a solution is already known for a given sequence (from previous work, for example), this can be used as the current lower bound.

  c. An approximation can be made for the initial value of the lower bound, based the physics of the problem being solved.

(2) Multiplier on the optimal, impulsive solutions is set to 1.

(3) Beginning with the next highest-ranked sequence, the branches begin to be evaluated.

  a. The optimal, multi-rev, two-impulse solution (over the same date range and time of flight range specified in the low-thrust problem) is calculated for the first segment of the branch, i.e., Earth – 1st asteroid.

    i. If the optimal, multi-rev, two-impulse solution is less than the current lower bound, the rest of that branch is pruned out. Continue to Step 4.

    ii. If the optimal, multi-rev, two-impulse solution is greater than the current lower bound, continue to next segment.

  b. The optimal, multi-rev, two-impulse solution is calculated for the first and second segments of the branch, i.e., Earth – 1st asteroid – 2nd asteroid.

    i. If the optimal, multi-rev, two-impulse solution is less than the current lower bound, the rest of that branch is pruned out. Continue to Step 4.

        ii. If the optimal, multi-rev, two-impulse solution is greater than the current lower bound, continue to next segment.

    c. Above process is continued until the end of the branch has been reached, i.e., Earth – 1st asteroid – 2nd asteroid - … - nth asteroid.

        i. If the optimal, multi-rev, two-impulse solution is greater than the current lower bound, compute the optimal low-thrust solution for the entire sequence (by running the genetic algorithm N_{GA} times).

        ii. If the optimal low-thrust solution is greater than the current lower bound, update the current lower bound.

    d. Continue above steps until all branches have been solved or pruned.

(4) Once all branches have been evaluated, compare the known low-thrust optimal solutions with their corresponding optimal, multi-rev, two-impulse solutions. Update the impulsive multiplier such that all known low-thrust solutions are bounded by the two-impulse solutions.

    a. If the impulsive multiplier needs to be updated, repeat Step 3 with the new value of impulsive multiplier.

    b. Otherwise, terminate branch-and-bound.

The algorithm specifies calculating the optimal two-impulse solutions for a number of asteroid sequences. In this work, this is calculated using a grid search, but a genetic algorithm, a gradient-based method with multiple starting points, or another method could also be employed.

If after the first iteration of the branch-and-bound, no low-thrust optimum solutions are calculated (other than the first optimum calculated to set the lower bound, if applicable), it is possible that the impulsive multiplier need not be updated. Instead of terminating the branch-and-bound with only the lower bound computed, a small set of additional low-thrust optima can be calculated based on the sequences that have the best two-impulse optimum solutions. Therefore, a larger sample size is available to more accurately determine the next required value for the impulsive multiplier. This procedure can also be carried out at the end of the branch-and-bound algorithm (after multiple iterations) to ensure that the correct value of the impulsive multiplier has been converged on. This process also serves to potentially locate additional good solutions that were not calculated during the branch-and-bound algorithm. This is especially important if the optimal solution is located early in the branch-and-bound algorithm, since the high value of the lower bound will eliminate a large number of potentially good solutions. While locating the optimum solution early in the algorithm will minimize the number of low-thrust optimizations that must be carried out, it also results in a larger number of good solutions being pruned out. The number of good solutions desired (in addition to the optimum) will determine the number of additional low-thrust optimizations to be carried out after the branch-and-bound algorithm as terminated.

Furthermore, if the branch-and-bound algorithm is started with an estimate made for the initial value of the lower bound (step 1c), and a satisfactory set of low-thrust solutions is not found, the lower bound can be incrementally decreased and the branch-and-bound algorithm re-run with the new initial lower bound. Both in this scenario and in the iterative approach to setting the impulsive multiplier, it is not necessary to re-run the two-impulse or low-thrust optimizations that have already been calculated. In order to minimize computation time, both the two-impulse and low-thrust optima should be saved for each asteroid sequence for which they are evaluated. A simple table look-up can then be used for these sequences as opposed to rerunning the optimizations for each iteration through the branch-and-bound algorithm. In this manner, only the new asteroid sequences requiring optimization need to be evaluated.

A similar iterative approach can also be taken with regards to the pruning phase. If time permits upon completion of the methodology, the pruning percentages could then also be relaxed, increasing the number of asteroid sequences passed to the global optimization phase. The branch-and-bound algorithm can then be reapplied in an attempt to locate additional good solutions.

3.2 Assumptions and Scope

The methodology presented here is for the conceptual design of low-thrust trajectories that rendezvous with multiple asteroids or other small bodies and if desired, return to Earth. The methodology is applicable to both small and large domain problems, and is scalable depending upon the computational resources available to the user.

While the methodology does model flybys (of gravitational bodies or the asteroids themselves), it can be used as an initial screening technique to identify the best asteroid sequences independent of flybys. As will be seen in Section 5.1, in many cases, the sequences that yield the best solutions in terms of mass and time of flight without flybys also yield the best solutions when flybys are considered. In these cases, the flybys serve to improve upon already good asteroid sequences. Because the addition of flybys to a trajectory is not a trivial task, the methodology proves useful in greatly reducing the number of asteroid sequences that need to be considered for the inclusion of these flybys. Modeling of flybys within this methodology is a potential area of future work.

At several steps in the methodology, two-impulse solutions are used to approximate the low-thrust trajectories – these are used in the pruning phase as well as in the branch-and-bound algorithm as a surrogate for relaxed solutions. For the asteroid pairings and sequences examined in this work, this proved to be a good approximation. In general, this will be the case when the ratio of thrust time to trajectory time is low for the low-thrust trajectory. While this is true for many low-thrust trajectories, there are cases where continuous or near-continuous thrusting over a majority of the trajectory is optimal. In these cases, a two-impulse approximation would not be appropriate. If the trend information provided by the two-impulse solutions is not a good surrogate for the associated low thrust problem, the methodology would have to be modified by identifying another metric that could be used in the place of the two-impulse approximation.

The branch-and-bound algorithm requires that the objective function increase or decrease monotonically as it proceeds down each segment of a branch. For example, final mass can only decrease as additional legs of the trajectory are examined, i.e., going from Earth – Body 1 to Earth – Body 1 – Body 2 to Earth – Body 1 - ··· - Body N. As long as this is true, branches of the tree can be pruned out before reaching the bottom branch, decreasing the computation time required. If this is not the case, however, the branch-and-bound algorithm could be modified such that entire branches must be evaluated before deciding whether or not they had to be pruned. For example, the two-impulse optimal solution for the entire Earth – Body 1 - ··· - Body N trajectory would have to be solved for to determine if that branch is pruned or needs to be evaluated in low-thrust. In this case, the step-by-step algorithm for the global optimization phase would instead proceed as follows:

(1) All asteroid sequences are ranked as a function of the normalized sum of the pruning metrics. In order to set the initial value of the lower bound, one of the three techniques outlined above would be employed.

(2) Multiplier on the optimal, impulsive solutions is set to 1.

(3) Beginning with the next highest-ranked sequence, the branches begin to be evaluated.

    a. The optimal, multi-rev, two-impulse solution (over the same date range and time of flight range specified in the low-thrust problem) is calculated for the full branch, i.e., Earth – 1st asteroid – 2nd asteroid - … - nth asteroid.

        i. If the optimal, multi-rev, two-impulse solution is greater than the current lower bound, compute the optimal low-thrust solution for the entire sequence (by running the genetic algorithm N_{GA} times).

        ii. If the optimal low-thrust solution is greater than the current lower bound, update the current lower bound.

    b. Continue above steps until all branches have been solved or pruned.

(4) Once all branches have been evaluated, compare the known low-thrust optimal solutions with their corresponding optimal, multi-rev, two-impulse solutions. Update the impulsive multiplier such that all known low-thrust solutions are bounded by the two-impulse solutions.

    a. If the impulsive multiplier needs to be updated, repeat Step 3 with the new value of impulsive multiplier.

    b. Otherwise, terminate branch-and-bound.

CHAPTER IV

VALIDATION OF METHODOLOGY

In order to validate the methodology, an intermediate-sized sample problem is created and solved. The proposed methodology is applied to this intermediate problem, in order to validate its ability to locate a suite of good solutions.

4.1 Intermediate Sample Problem

The intermediate sample problem was created with the same parameters and constraints as the small sample problem, which was outlined in Section 2.5. This problem, however, is an order or magnitude larger, consisting of three groups of twelve asteroids, for a total of 10,368 discrete asteroid sequences. Figure 36 plots the asteroids in the intermediate problem, as a function of their semi-major axis, inclination, and eccentricity. The orbital elements for these asteroids can also be found in Appendix A. Once again, the objective function is to maximize final mass, with constraints placed on the time of flight of the individual trajectory legs.

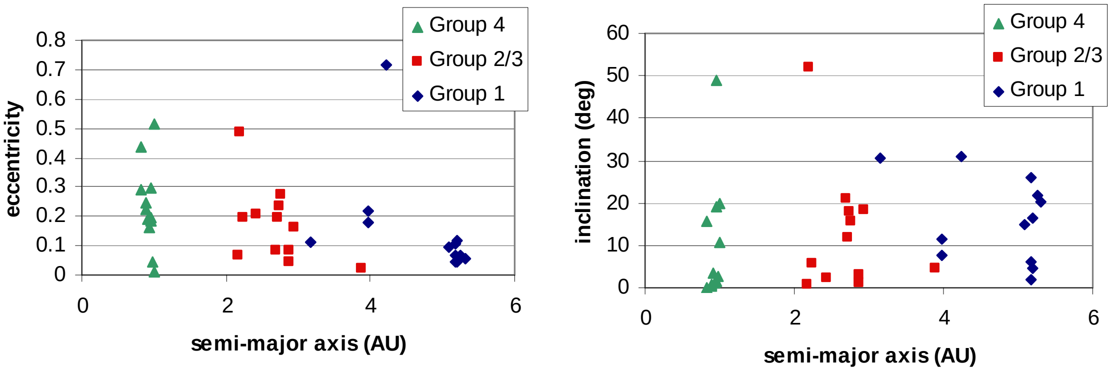

Figure 36: Set of asteroids for intermediate sample problem.

The intermediate problem was solved in the same manner as the small sample problem, by discretizing the departure date and flight times and using MALTO to solve for each trajectory leg. Once again, all possible combinations were analyzed to have a basis for evaluation of the performances of the pruning and global optimization phases. The best solution visits the following asteroids: 2006 QQ56 – Medusa – Kostinsky. The spacecraft departs Earth on March 1, 2015 with a launch V_∞ of 2.59 km/s. The time of flight for each leg is 600 days, 1600 days, and 1200 days, respectively. The total mission time is 3580 days, which includes the two 90-day stay times at each asteroid, and the arrival mass is 903 kg. This is actually the same optimal solution as for the small sample problem, and is plotted in Figure 19.

Table 8: Ten best asteroid combinations for intermediate problem ranked by final mass.

Earth Dep. Date	Ast. 1	Ast. 2	Ast. 3	Leg 1 TOF (days)	Leg 2 TOF (days)	Leg 3 TOF (days)	Mf (kg)
03/01/2015	2006 QQ56	Medusa	Kostinsky	600	1600	1200	904
01/01/2015	2004 VJ1	Medusa	Kostinsky	500	1800	1200	870
08/22/2016	2006 QQ56	Hertha	Telamon	600	1800	1200	856
03/05/2020	2006 FH36	Medusa	Potomac	500	1800	1200	843
03/29/2021	Apophis	Hertha	Pandarus	300	1800	1200	843
10/12/2018	2006 FH36	Geisha	Kostinsky	500	1800	1200	834
01/01/2015	2002 AA29	Medusa	Kostinsky	600	1700	1200	831
09/11/2018	2006 QQ56	Geisha	Kostinsky	600	1700	1200	826
06/12/2024	2004 VJ1	Medusa	Potomac	600	1800	1200	820
08/28/2015	2006 QQ56	Geisha	Caltech	600	1700	1200	812

Table 9: Orbital elements of asteroids in Table 9, in the J2000 heliocentric ecliptic frame.

Asteroid Name	Group #	semi-major axis (AU)	eccentricity	inclination (deg)	longitude of the asc. node (deg)	Argument of periapsis (deg)
2006 QQ56	4	0.987	0.047	2.83	163.33	332.96
2002 AA29	4	0.994	0.013	10.74	106.47	100.61
Apophis	4	0.922	0.191	3.33	204.46	126.40
2004 VJ1	4	0.944	0.164	1.29	233.54	332.36
2006 FH36	4	0.954	0.199	1.59	280.92	154.81
Geisha	2/3	2.24	0.193	5.66	78.34	299.88
Medusa	2/3	2.17	0.065	0.937	159.65	251.13
Hertha	2/3	2.43	0.207	2.31	343.90	340.04
Kostinsky	1	3.99	0.220	7.64	257.11	163.00
Telamon	1	5.17	0.108	6.09	341.01	111.19
Pandarus	1	5.17	0.068	1.85	179.86	37.74
Caltech	1	3.16	0.114	30.69	84.61	294.92
Potomac	1	3.98	0.181	11.40	137.51	332.82

Table 8 lists the 10 best asteroid sequences, ordered in terms of final mass. Table 9 lists the Keplerian orbital elements of each of the asteroids that appear in Table 8, in the J2000 heliocentric ecliptic frame.

4.2 Application of Methodology to Intermediate Problem

4.2.1 Pruning Phase

First, the pruning phase of the methodology is applied to the intermediate problem. The first step of the pruning phase requires keeping only asteroid sequences where the semi-major axis of each asteroid increases from one asteroid to the next. This first step reduces the number of asteroid sequences in the design space from 10,368 to 1,728 (a factor of 6 reduction). Only two sequences were eliminated that yield feasible solutions (final mass greater than 500 kg), and their optimum final masses are 608 kg and 524 kg. These solutions rank 191st and 467th, respectively. Therefore, this first pruning step is effective in reducing the size of the design space without eliminating the best asteroid sequences.

The second and third steps in the pruning phase use two metrics to eliminate a user-chosen percentage of asteroid pairs from each leg of the trajectory. Since this problem is not significantly larger than the small sample problem, the same percentage reductions are applied to each leg: $k_1 = 0.3$, $k_2 = 0.25$, and $k_3 = 0.15$. The first of these two metrics – the angle between two asteroids' angular momentum vectors – reduces the number of asteroid sequences from 1,728 to 824 (factor of 2). The second metric – the optimal, phase-free, impulsive ΔV – further reduces the number of asteroid sequences from 824 to 416 (factor of 2).

Overall, the pruning procedure reduces the number of asteroid sequences by a factor of 25, from 10,368 to 416. While 199 feasible sequences are eliminated, only one sequence in the top ten is eliminated (M_f = 831, ranked 7th overall). The next best sequence eliminated has a final mass of 760 kg, which ranks 25th. Figure 37 plots in blue the 1728 remaining asteroid sequences after the first pruning metric is applied, ordered by optimal low-thrust final mass. In pink are the asteroid sequences remaining after the second and third pruning metrics are applied. As can be seen, a majority of the good asteroid sequences remain in the design space after the pruning procedure is applied. Table 10 summarizes the pruning procedure as applied to the intermediate problem.

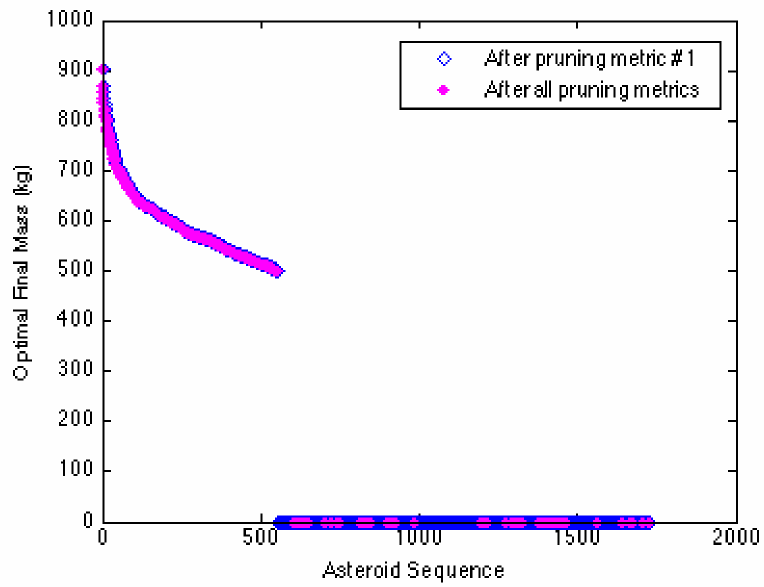

Figure 37: Asteroid sequences remaining in design space after 1st pruning metric (blue) and 2nd & 3rd pruning metrics (pink).

Table 10: Pruning methodology applied to intermediate problem.

Pruning Metric	Trajectory Leg	% Asteroid Pairs Eliminated	# Sequences Eliminated
$a_i < a_{i+1}$	All	N/A	8640
wedge angle	Leg 1	30%	432
wedge angle	Leg 2	25%	324
wedge angle	Leg 3	15%	130
impulsive ΔV	Leg 1	30%	174
impulsive ΔV	Leg 2	25%	152
impulsive ΔV	Leg 3	15%	100

In order to further validate the pruning phase of the methodology, the correlations between each pruning metric and optimal low-thrust final mass were calculated, as was done for the small sample problem while developing the methodology. Figure 38 plots the maximum low-thrust final mass for each asteroid pair, for each trajectory leg, as a function of the angle between the angular momentum vectors. As expected, there appears to be a strong correlation between this pruning metric and final mass, particularly for the Leg 1 and Leg 2 asteroid pairs. Similarly, Figure 39 plots the maximum low-thrust final mass for each asteroid pair, for each trajectory leg, as a function of the minimum phase-free, two-impulse ΔV of each asteroid pair. The same results are observed, where there is a strong correlation between ΔV and final mass for the first two trajectory legs, but a much smaller correlation for the third leg.

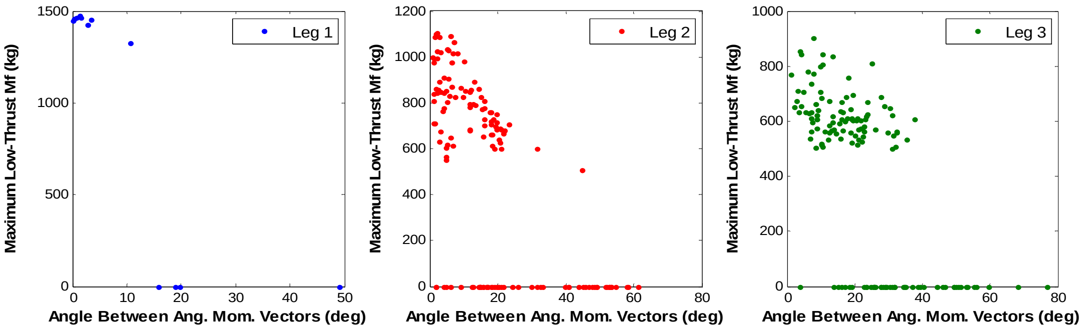

Figure 38: Maximum final mass for each asteroid pairing as a function of the angle between the two angular momentum vectors.

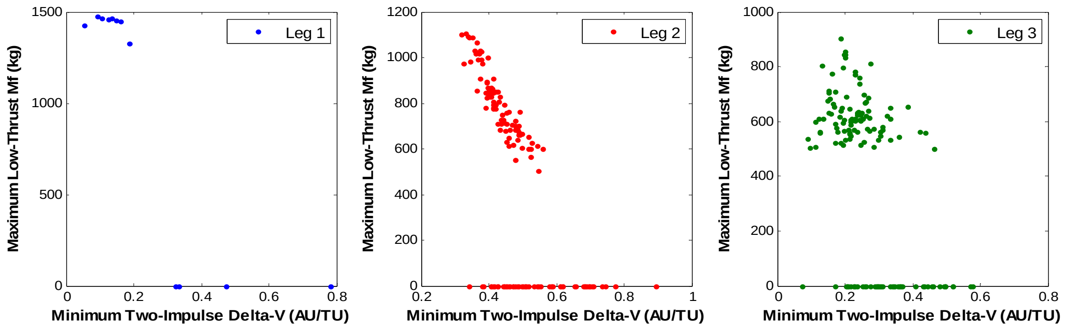

Figure 39: Maximum final mass for each asteroid combination as a function of the minimum, phase-free, two-impulse ΔV.

Table 11 presents the correlation coefficients for each of the cases plotted above. While the correlation coefficients are not as close to -1 (indicating perfect negative correlation) as for the small sample problem, there is still a good correlation between the two pruning metrics and low-thrust final mass. In particular, the Leg 2 correlations are lower than was observed in the small sample problem – the Leg 3 correlation for wedge angle is actually higher than the Leg 2 correlation. If the plots in Figure 39 are examined closely, however, there are many cases of low final mass values for good (low) values of the pruning metrics, which is contributing to the lower correlation coefficient. What is important, however, is high final mass values do not exist for poor (high) values of the pruning metrics, which would cause those asteroid pairings to be pruned from the design space. Therefore, these two pruning metrics can be reliably used to prune the design space, without eliminating the best asteroid sequences.

Table 11: Correlation (coefficients) between pruning metrics and low-thrust final mass for the intermediate sample problem.

	Wedge Angle	Delta-V
Leg 1	-0.81	-0.83
Leg 2	-0.64	-0.66
Leg 3	-0.66	-0.54

4.2.2 Global Optimization Phase

The global optimization scheme is then applied to the reduced problem. First, the remaining asteroid sequences are ranked by the normalized sum of the pruning metrics, in order to determine the order in which they will be evaluated in the branch-and-bound process. Then, the optimum low-thrust final mass for the highest ranked asteroid sequence is obtained using the genetic algorithm combined with MALTO, in order to determine the initial lower bound on low-thrust final mass. In this case, the result is an initial lower bound of 638 kg. The first iteration of the branch-and-bound algorithm is carried out without any impulsive multiplier on the "relaxed" solutions (optimal two-impulse ΔV). The first iteration eliminates zero of seven Earth – 1st asteroid branches, 11 of 48 Earth – 1st asteroid – 2nd asteroid branches, and 306 of the remaining 319 Earth – 1st asteroid – 2nd asteroid – 3rd asteroid branches. This leaves 12 asteroid sequences for which the low-thrust optimum must be calculated (one was already calculated to set the initial lower bound).

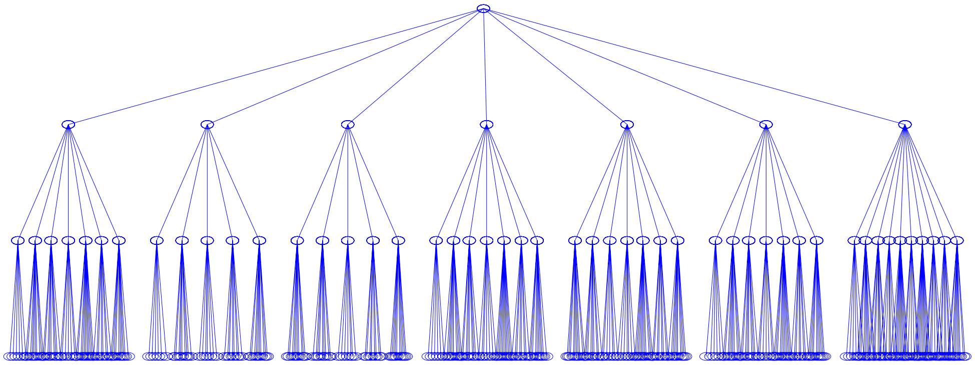

Figure 40: Branch-and-bound tree enumerating all asteroid sequences remaining in the intermediate sample problem after the pruning phase.

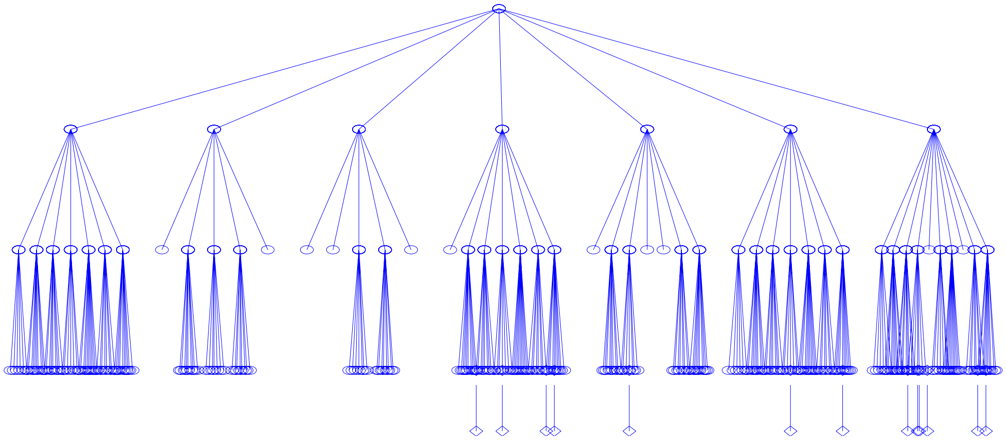

Figure 41: Branch-and-bound tree illustrating asteroid sequences pruned out by the first iteration of the branch-and-bound algorithm on the sample problem.

Figure 40 and Figure 41 are graphic representations of the first iteration of the branch-and-bound algorithm. The branch-and-bound tree in Figure 40 enumerates all of the asteroid sequences remaining after the pruning phase. The tree in Figure 41 illustrates the branches of the tree that were pruned out during the first iteration of the branch-and-bound. The sequences that required low-thrust optimization are represented by the dashed lines and diamonds at the bottom of the figure.

The genetic algorithm was run three times to calculate the low-thrust optimum for each asteroid sequence that was not pruned out by the branch-and-bound algorithm. At the end of the first iteration, the best asteroid sequence found has a low-thrust final mass of 904 kg, which from previous enumeration is the best known solution. When the low-thrust optimum solutions are compared to their corresponding two-impulse optimal solutions, however, the impulsive multiplier must be increased to 1.145. Figure 42 plots the results of this first branch-and-bound iteration. The blue line plots the optimal two-impulse final mass, sorted from largest to smallest, for every asteroid sequence. The green dots plot the corresponding optimum low-thrust final mass for the 13 sequences that were evaluated. Finally, the red line plots the new optimal two-impulse final mass, based on the new value of the impulsive multiplier.

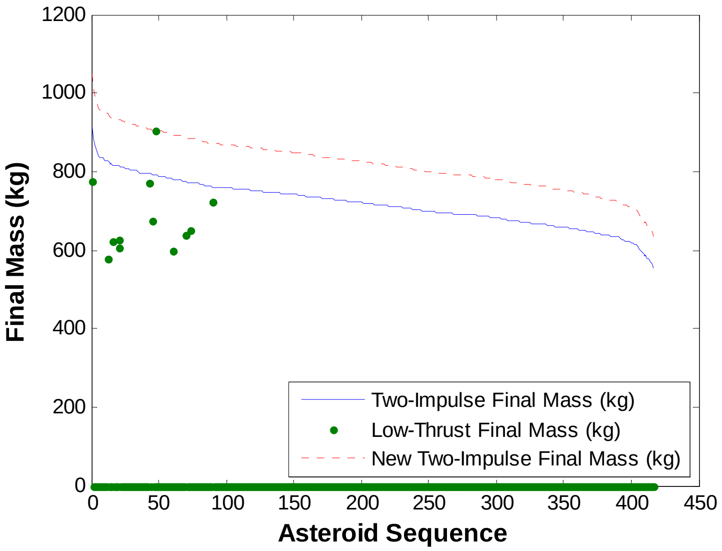

Figure 42: Results of the 1st iteration of the branch-and-bound algorithm applied to the intermediate sample problem.

Because the impulsive multiplier had to be updated, a second iteration of the branch-and-bound algorithm is required. In the second iteration, none of the Earth – 1st asteroid – 2nd asteroid branches can be pruned. Therefore, the optimal two-impulse solution for all asteroid sequences must be calculated. Furthermore, the low-thrust optimum must be computed for an additional 42 asteroid sequences. After the second iteration, however, the impulsive multiplier does not need to be updated again, and the algorithm is considered to be converged. Figure 43 plots the results of this 2nd iteration.

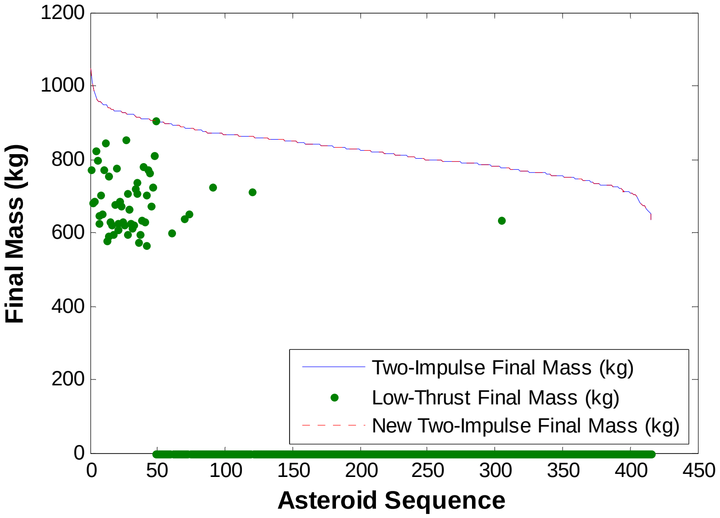

Figure 43: Results of the 2nd iteration of the branch-and-bound algorithm applied to the intermediate sample problem.

Figure 44 plots all of the asteroid sequences remaining in the design space at the start of the global optimization step, ordered by low-thrust final mass (blue dots). Plotted in pink are the 55 sequences found during the global optimization. Not only is the optimum solution found, but also five of the top ten asteroid sequences in the known solution space for the intermediate problem are identified. The overall branch-and-bound process required the two-impulse optimal ΔV to be calculated for all 416 asteroid sequences, and the low-thrust optimal was calculated for 55 asteroid sequences.

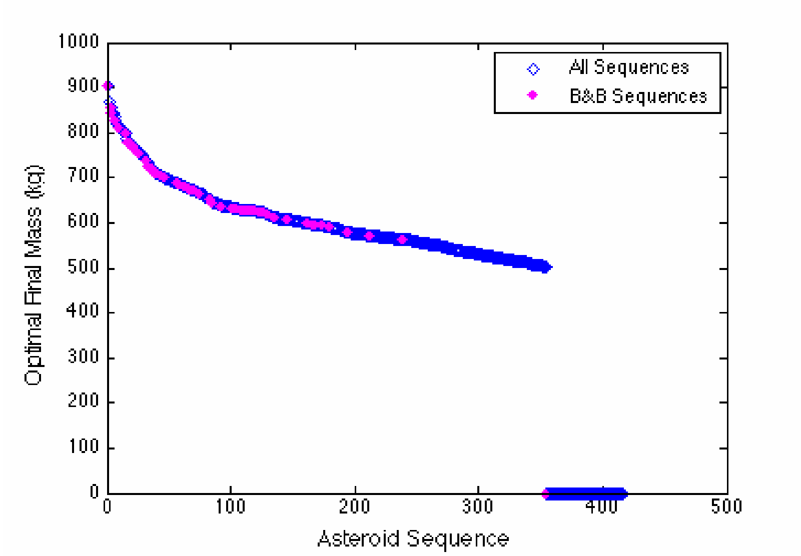

Figure 44: Asteroid sequences identified by applying branch-and-bound algorithm to intermediate problem.

Table 12 lists the top ten asteroid sequences for the intermediate sample problem (the same as are listed in Table 8). For each sequence, it is indicated whether that particular sequence remained in the design space after the pruning phase and whether or not that sequence was found by the branch-and-bound algorithm. The goal of the methodology is two-fold. First, it should maintain a majority of the best solutions in the design space after the pruning phase. This has been achieved on the intermediate sample problem, since nine of the top ten sequences remain. Second, the branch-and-bound algorithm should then locate a suite of good solutions. This aspect of the methodology has also been achieved, since the branch-and-bound successfully located five of the top ten best solutions, along with the optimum solution (found during the 1st iteration).

Table 12: Effectiveness of the methodology at locating the top ten solutions to the intermediate sample problem.

Asteroid Sequence	Mf (kg)	Remaining After Pruning Phase?	Found By Branch-and-Bound?
2006 QQ56 – Medusa – Kostinsky	904	√	√
2004 VJ1 – Medusa – Kostinsky	870	√	
2006 QQ56 – Hertha – Telamon	856	√	√
2006 FH36 – Medusa – Potomac	843	√	√
Apophis – Hertha – Pandarus	843	√	
2006 FH36 – Geisha – Kostinsky	834	√	
2002 AA29 – Medusa – Kostinsky	831		
2006 QQ56 – Geisha – Kostinsky	826	√	√
2004 VJ1 – Medusa – Potomac	820	√	
2006 QQ56 – Geisha – Caltech	812	√	√

CHAPTER V

APPLICATION OF METHODOLOGY TO LARGER PROBLEMS

In this chapter, the methodology is applied in full to two larger problems where the global optimum solution is unknown. The first problem is derived as a modified version of the 3rd Global Trajectory Optimization Competition, while the second problem is modified version of the 2nd Global Trajectory Optimization Competition. For each problem, a number of known good solutions exists from the competition results, which will serve as benchmarks to evaluate the effectiveness of the methodology. For both of these problems, the goal is to find a suite of good solutions for subsequent analysis with higher fidelity methods. Additionally, the methodology is applied to the full version of the GTOC2 problem, subject to the time limitations of the competition, in order to determine where the best solution found in that timeframe would have placed in the competition.

For the larger problems, which will require greater computational resources, a computer cluster is utilized to carry out the low-thrust trajectory optimizations. The genetic algorithm is run on a computer cluster comprised of fifteen nodes. A Matlab code runs on the master node of the cluster, which executes the genetic algorithm and distributes the MALTO runs to each of the nodes. The master node contains two AMD Opteron processors at 2.2 GHz each. MALTO then runs on the cluster nodes. A Fortran script creates the input files required for each of the MALTO runs, based on a batch script sent to each node by Matlab. Seven of these nodes contain two AMD Opteron processors with 2.2 GHz each and 5 GB of RAM. The remaining eight nodes contain two dual core AMD Opteron processors with 2.4 GHz each and 12 GB of RAM. The operating system on the computer cluster is Ubuntu 8.04 LTS.

Because of limitations in the Matlab Distributed Computing Server, each generation of the genetic algorithm was manually distributed to the nodes – e.g., if sixty function calls to MALTO were required during a given generation, four function calls were sent to each node. Section 6.2 outlines recommendations for future work in order to decrease the run time of the genetic algorithm.

5.1 Modified GTOC3 Problem

In 2007, the 3rd Global Trajectory Optimization Competition (GTOC3) posed another asteroid rendezvous problem.[10] For this problem, participants had to find the best possible trajectory, again using electric propulsion, that would rendezvous with three near-Earth asteroids out of a single group of 140 candidates, and then return to Earth. Figure 45 plots the candidate asteroids, as a function of semi-major axis, eccentricity and inclination.

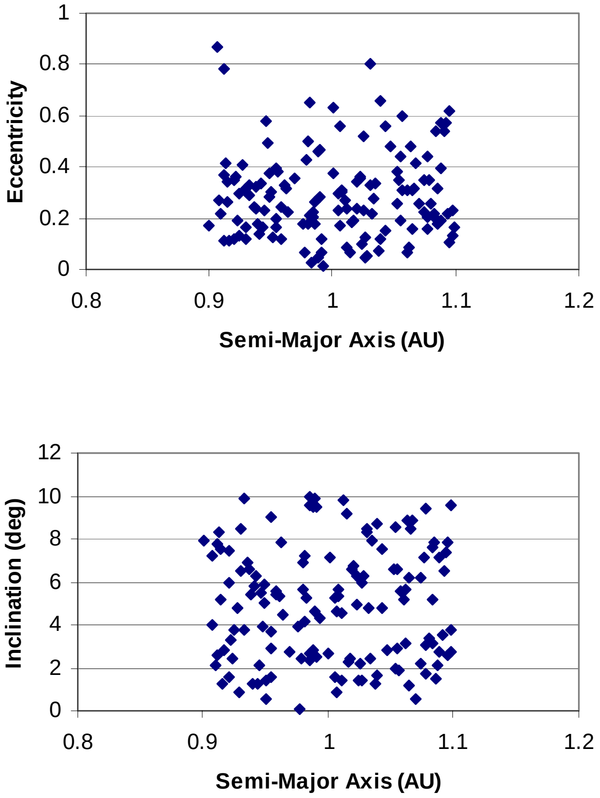

Figure 45: GTOC3 set of asteroids.

Because these are all near-Earth asteroids, their semi-major axes all fall between 0.9 AU and 1.1 AU. All have low inclination orbits – less than 10 degrees – while their eccentricities range from near circular to a maximum value of 0.87. Note that this is a significantly different design space than in the GTOC2 problem.

In addition, the objective function for GTOC3 is also slightly different, as it involves maximizing a combination of final mass and stay time at the asteroids:

$$J = \frac{m_f}{m_i} + K \frac{\min\limits_{j=1,3}(\tau_j)}{\tau_{max}} \tag{19}$$

In Equation 19, m_i and m_f are the spacecraft initial and final mass, τ_j is the stay time at the j^{th} asteroid, τ_{max} is the maximum allowable flight time, and K = 0.2. Unlike GTOC2, gravity assists using Earth were permitted. Additional problem constraints are listed in Table 13.

Table 13: Constraints on GTOC3 problem.

Constraint	Value	Notes
Earth Launch V_∞	≤ 0.5 km/s	Unconstrained direction
Earth Launch Date	2016 – 2025	Inclusive
Asteroid Stay Time	≥ 60 days	
Total Flight Time	≤ 10 years	
Spacecraft Initial Mass	2000 kg	
Thruster Isp	3000 s	
Maximum Thrust	0.15 N	Can be turned on/off at will No constraint on direction

For the purpose of validating the proposed methodology, the problem is altered slightly. The objective function is changed to maximizing the final mass at Earth return. This does not change the problem significantly, since the maximum value of the stay time portion of the GTOC3 objective function contributes only marginally to the overall objective function. The first portion of the original objective function will be on the order of 0.8 (assuming a final mass of 1600 kg). The second portion however, can only range from 0.003 (assuming a minimum stay time of 60 days) to 0.067 (assuming a maximum stay time of 3.33 years, which is based on times of flight of zero). More realistically, the second portion of the objective function will range from 0.003 to approximately 0.04. (This same increase in the objective function can be obtained by increasing the final mass by 80 kg.) Because the stay time is being removed from the objective function, the stay time at each asteroid is also being fixed at 60 days. The other alteration of the original GTOC3 problem is to disallow Earth gravity assists.

The best asteroid sequence found during the competition was Earth – Earth – Earth – 2000 SG344 – Earth – 2004 QA22 – 2006 BZ147 – Earth – Earth. With the Earth gravity-assists removed, the mass-optimal solution for this sequence is 1590 kg. This was found by doing a grid search, with increments of 30 days on departure date and 100 days on flight time. A genetic algorithm with a higher resolution than the grid search was also run several times and was unable to find a better solution. The optimal trajectory found by the grid search launches on December 13, 2024 with a launch V_∞ of 0.5 km/s, and has a total flight time of 9.8 years. For reference, Table 14 summarizes this trajectory, which is the current best known solution to the modified GTOC3 problem.

Table 14: Summary of best known trajectory for modified GTOC3 problem.

Departure Body	Arrival Body	Departure Date (MJD)	Time of Flight (days)	Stay Time (days)	Departure Mass (kg)	Arrival Mass (kg)
Earth	2000 SG344	60658	1000	60	2000	1946.4
2000 SG344	2004 QA22	61718	1100	60	1946.4	1795.5
2004 QA22	2006 BZ147	62878	600	60	1795.5	1715.0
2006 BZ147	Earth	63538	700	60	1715.0	1589.8

The initial problem has a total of nearly 2.7 million possible asteroid sequences. For this problem, the goal of the pruning phase of the methodology will be to reduce the number of asteroid sequences by two orders of magnitude. This will leave on the order of 10,000 asteroid sequences in the design space, which is small enough to be handled by the global optimization phase, but large enough to be confident that many of the best asteroid sequences have not been pruned out. Because all of the asteroids are near-Earth asteroids, and therefore have similar semi-major axes, the first pruning metric will not be employed on this problem – the requirement that the asteroid sequence must increase sequentially in semi-major axis. Furthermore, because only total time of flight is constrained, and not part of the objective function, this pruning metric becomes even less relevant. Therefore, only the last two pruning metrics will be used: θ_{wedge} and optimal phase-free, two-impulse ΔV. In order to achieve the desired size of the design space, the following percentage reductions are applied to the problem, for both pruning metrics: 70% for Leg 1, 60% for Leg 2, 50% for Leg 3, and 25% for Leg 4. This reduces the number of asteroid sequences to 10,311. As a check, the best known sequence, presented above, is still in the design space.

Next, the global optimization step, combining the branch-and-bound method with the genetic algorithm, is applied to the reduced problem. Because there is already a best known asteroid sequence, it is used as the initial lower bound on low-thrust final mass Note that this sequence also happens to be the highest ranked sequence based on the normalized sum of the pruning metrics and so would be the first low-thrust trajectory evaluated by the branch-and-bound algorithm. For each asteroid sequence where the low-thrust optimum must be computed, the genetic algorithm is run three times, using the settings listed in Table 15 and Table 16.

Table 15: Settings for genetic algorithm within branch-and-bound, as applied to the modified GTOC3 problem.

GA Setting	Value
Population Size	90
Stall Generations	10
Tournament Size	4
Crossover Probability	0.8
Mutation Probability	0.1

Table 16: Design variables for genetic algorithm within branch-and-bound, as applied to the modified GTOC3 problem.

Design Variable	Units	# Bits	Lower Bound	Upper Bound
Earth Departure Date	JD	8	2457388	2461041
TOF, Leg 1	days	6	200	2000
TOF, Leg 2	days	6	200	2000
TOF, Leg 3	days	6	200	2000
TOF, Leg 4	days	6	200	2000

The settings presented in Table 15 were chosen based partially on the results of applying the genetic algorithm to the small and intermediate sample problems and on general rules of thumb for genetic algorithms. In general, the population size should be roughly two to three times the length of the chromosome string. Based on the number of variables and the bits chosen for each variable, the chromosome string is 32 bits long, which leads to a population size of between 64 and 96. The results of the sample problems led to reducing the number of stall generations, since it was observed that after ten generations without a change in the objective function, the genetic algorithm rarely found a better solution, so increasing the number of stall generations needlessly increases the required number of function evaluations. Finally, the tournament size, crossover probability, and mutation probability were based on the values of each setting that appeared to work best on the sample problems and also on general genetic algorithm rules of thumb.

As the number of bits chosen for each variables increases, the resolution of the solution increases, but so does the required population size and therefore the number of function calls. The number of bits chosen, shown in Table 16, attempts to balance these factors. The resulting discretization for Earth departure date is approximately 15 days and the discretization for the times of flight is approximately 28 days. The bounds chosen for the times of flight represent the minimum realistic time of flight for any of the trajectory legs and the maximum time of flight per leg that would likely yield an overall time of flight below the constraint.

First, the branch-and-bound algorithm is applied without any multiplier on the optimal two-impulse mass solutions. Figure 46 plots the results of this first iteration. Each asteroid sequence is plotted as a function of its optimal impulsive ΔV, with the red line indicating the best known low-thrust solution at the end of the iteration. Any points that fall below this red line are pruned from the branch-and-bound tree. During the first iteration, all of the asteroid sequences are pruned, since none of the optimal impulsive solutions are greater than 1590 kg. The optimal impulsive final mass for this asteroid sequence is 1477 kg, leading to a required multiplier of 1.077 for the next branch-and-bound iteration.

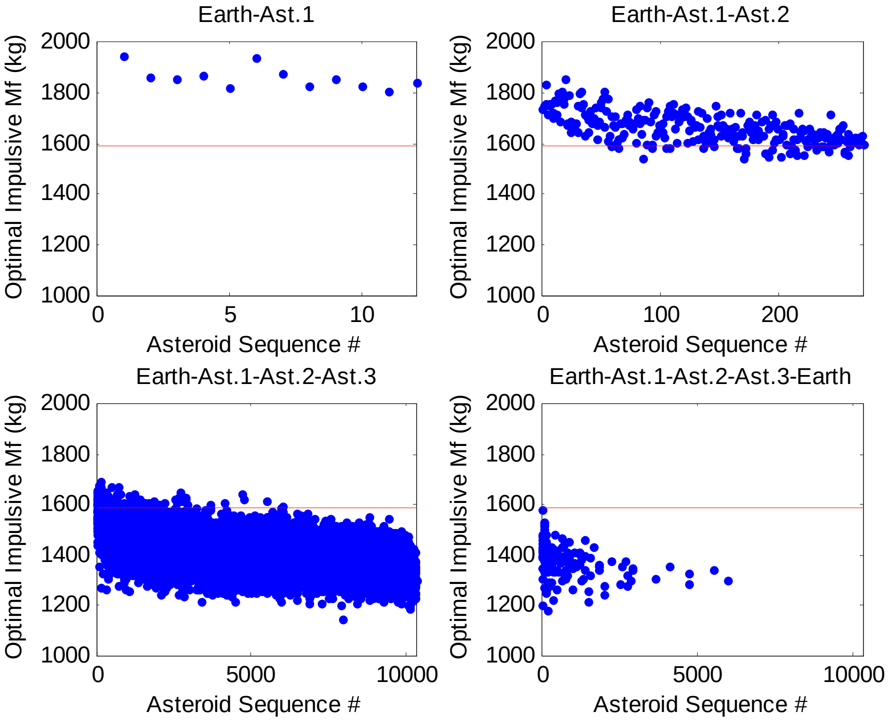

Figure 46: Branch-and-bound results, iteration #1 (impulsive multiplier = 1).

Figure 47 plots the results of the second iteration. The low-thrust optimum solution is calculated for an additional seven asteroid sequences. In the process, an asteroid sequence with a higher low-thrust final mass is located, raising the current best known solution to 1621 kg. Based on comparing the eight known low-thrust final masses to their corresponding impulsive optimal solutions, the multiplication factor must be increased again, to a value of 1.096.

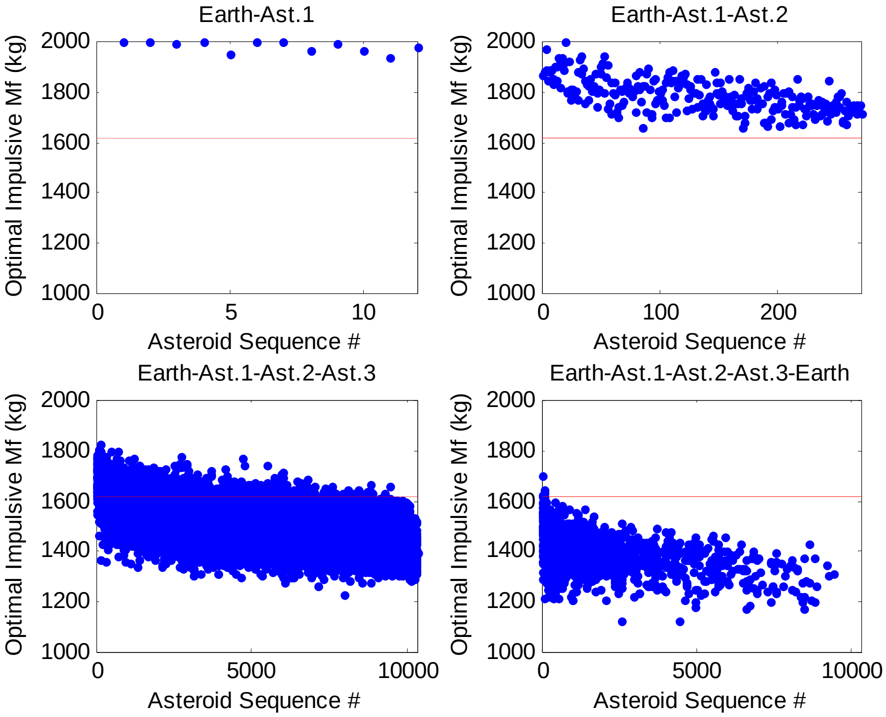

Figure 47: Branch-and-bound results, iteration #2 (impulsive multiplier = 1.077).

Figure 48 plots the results of the third iteration. Nine additional low-thrust optima are computed during this iteration. The best known solution, however, is not improved upon. Furthermore, the impulsive multiplier does not have to be updated, as it is sufficient to bound all of the known low-thrust solutions. Therefore, the branch-and-bound algorithm is considered converged after three iterations, with a final impulsive multiplier of 1.096 and a best low-thrust final mass of 1621 kg.

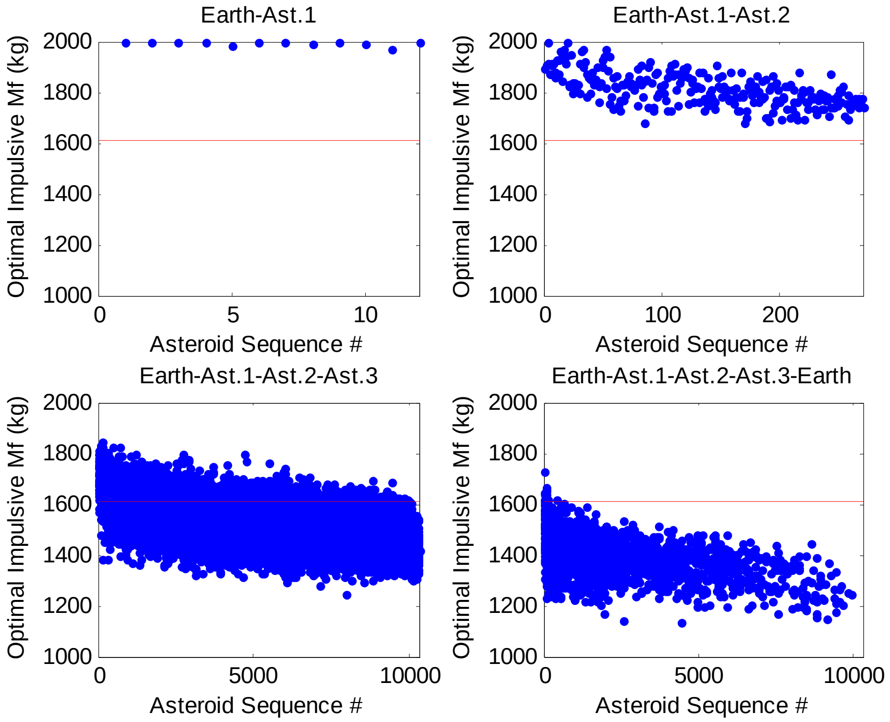

Figure 48: Branch-and-bound results, iteration #3 (impulsive multiplier = 1.096).

Overall, low-thrust optima for 17 asteroid sequences were computed. Impulsive optimum solutions for all of the Earth – Asteroid 1 (12), Earth – Asteroid 1 – Asteroid 2 (270), and Earth – Asteroid 1 – Asteroid 2 – Asteroid 3 (10,311) sequences had to be computed. Impulsive optima for only 1795 full sequences (Earth – Asteroid 1 – Asteroid 2 – Asteroid 3 – Earth), however, were computed. All of the impulsive optima were calculated using a simple grid search. Table 17 summarizes the best asteroid sequences found in this investigation during the branch-and-bound process. The asteroids are indicated by an index number – the corresponding asteroid names can be found in Appendix B. The original best known solution, presented in Table 14, ranks third.

Another important observation is that the best low-thrust solutions found by the methodology are among the highest ranked sequences according to the normalized sum of the pruning metrics. Of the 10,311 asteroid sequences remaining prior to the branch-and-bound algorithm, the top five sequences identified in Table 18 are ranked #2, #19,

#1, #63, and #16, respectively. Therefore, these will be evaluated early in the branch-and-bound process, which is important for two reasons. First, the lower bound on the objective function will be set to a large value early in the algorithm, minimizing the number of low-thrust optimization that must be carried out. Second, if time and computing resources were not available to run the branch-and-bound algorithm to completion, many of these solutions could still be identified. This is a consistent trend of the methodology, which will be seen again on the GTOC2 problem.

Table 17: Top 5 solutions identified to the modified GTOC3 problem.

Asteroid Sequence	Earth Departure	TOF 1 (days)	TOF 2 (days)	TOF 3 (days)	TOF 4 (days)	Low-Thrust M_f (kg)
E - 76 - 88 - 49 - E	06/13/2017	1086	657	457	1000	1621
E - 88 - 76 - 49 - E	12/09/2016	514	1029	514	1314	1597
E - 49 - 37 - 85 - E	12/13/2024	1000	1100	600	700	1590
E - 96 - 88 - 49 - E	04/17/2017	971	743	429	1143	1589
E - 88 - 19 - 49 - E	03/05/2019	1286	771	971	400	1587

Finally, the low-thrust optimum for an additional thirty asteroid sequences – based on the summed pruning metric ranking – is computed (again by running the genetic algorithm three times) to test the effectiveness of the branch-and-bound method in finding the best solutions. Figure 49 plots all of the asteroid sequences where the low-thrust optimum is known, along with the corresponding optimal impulsive solution, multiplied by 1.096 (the final value of the impulsive multiplier). All of the new low-thrust solutions are bounded by the impulsive final mass. Additionally, no better solutions are found than those listed in Table 17. Therefore, it appears that the branch-and-bound method was successful when applied to the modified GTOC3 problem.

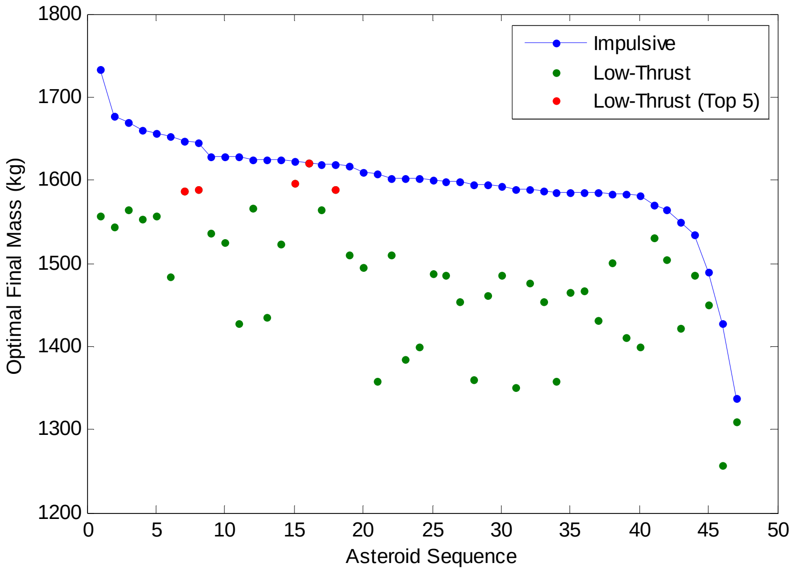

Figure 49: Comparison of optimum impulsive final mass (multiplied by 1.096) and optimum low-thrust final mass for modified GTOC3 problem.

Even though the original GTOC3 problem was altered, the proposed methodology could still be applied to the original competition, and doing so could have yielded the winning asteroid sequence. The only changes made to the competition problem were to alter the objective function and to disallow gravity assists. Promising asteroid sequences could be identified by applying the methodology to the simplified problem, and a handful of sequences could then be analyzed in greater detail. At this point, gravity assists could be added and asteroid stay time considerations could be accounted for.

As a benchmark, Table 18 lists the top ten previously known solutions from the competition, where each intermediate "E" indicates an Earth flyby (the first and last "E" indicate Earth departure and arrival). Additionally, the last column in the table indicates if that particular asteroid sequence (excluding the Earth flybys) was found by applying the methodology to the modified problem, and what its rank was in the modified problem. Note that the top four competition solutions all contain the same asteroid sequence, simply with different Earth flybys added. This sequence corresponds to the third best sequence found when applying the methodology to the modified GTOC3 problem (Table 17). Additionally, the sixth and ninth best solutions from the competition correspond to the fourth best solution to the modified problem; the eighth best solution from the competition corresponds to the seventh best solution to the modified problem; the tenth best solution from the competition corresponds to the fifth best solution to the modified problem.

The only asteroid sequence from Table 18 that was not identified by the present methodology is 88-96-49, which generated the fifth and seventh best solutions during the competition with the inclusion of Earth flybys. This asteroid sequence remained in the design space after the pruning phase of the methodology, but was pruned out during the branch-and-bound algorithm. In order to check if it was pruned out correctly, its optimal low-thrust final mass was computed, and was found to be 1565 kg. Its corresponding impulsive optimal mass is 1478 kg, which with the final impulsive multiplier of 1.096, bounds the low-thrust solution. As evidenced by these results, applying the methodology to the modified problem could be used as an initial screening test to identify the most promising asteroid sequences for more detailed analysis.

Table 18: Top ten solutions from GTOC3 competition.

Asteroid Sequence	Final Mass (kg)	Min. Stay Time (days)	J	Found by methodology?
E-E-E-49-E-37-85-E-E	1733	60	0.870	√ (3rd)
E-E-49-E-37-85-E-E	1730	60	0.868	√ (3rd)
E-49-E-37-85-E-E	1721	60	0.864	√ (3rd)
E-49-E-E-37-85-E-E	1717	60	0.862	√ (3rd)
E-88-E-96-49-E	1647	245	0.837	
E-96-E-88-49-E	1647	211	0.835	√ (4th)
E-88-E-96-E-49-E	1658	60	0.832	
E-E-96-76-E-49-E	1649	60	0.828	√ (7th)
E-96-E-88-49-E	1633	165	0.826	√ (4th)
E-88-19-49-E	1606	62	0.806	√ (5th)

5.2 GTOC2 Problem

In 2006, the 2nd Global Trajectory Optimization Competition (GTOC2)[8,9] posed a trajectory optimization problem of a "Grand Asteroid Tour." Over the span of four weeks, 26 organizations attempted to design the best possible trajectory, using electric propulsion, that would rendezvous with one asteroid from each of four defined groups. Only 15 of the 26 teams were able to submit solutions by the deadline, and only 11 of those solutions satisfied all of the problem constraints. The given objective function rewarded trajectories with low propellant consumption and low total flight time. Earth launch date, Earth launch V_∞, times of flight, and stay times at each asteroid were free design variables. Figure 50 plots the set of asteroids for the GTOC2 problem, as a function of inclination, eccentricity, and semi-major axis.[75]

Group 4, which is comprised of asteroids closest to Earth, contains 338 asteroids. Group 3 has 300 asteroids, Group 2 has 176 asteroids, and Group 1, whose asteroids are the furthest from Earth, has 96 asteroids. As such, this problem permits 41 billion possible discrete asteroid combinations, which is four orders of magnitude larger than the GTOC3 problem, which was examined in Section 5.1. The size of this problem is increased further when launch date, arrival dates, stay times, and thrust profile are included as free design variables. The objective was to maximize the ratio of final mass to total time of flight, as is presented in Equation 20.

$$J = \frac{M_f}{TOF} \tag{20}$$

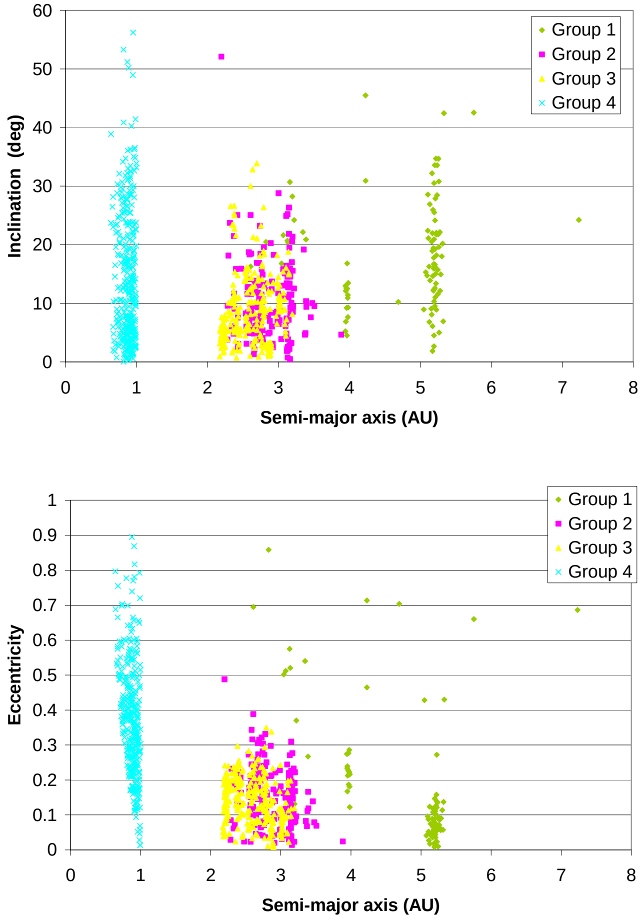

Figure 50: GTOC2 set of asteroids.

Table 19 presents the constraints on the problem. No gravity assists were permitted in the competition.

Table 19: Constraints on GTOC2 problem.

Constraint	Value	Notes
Earth Launch V_∞	≤ 3.5 km/s	Unconstrained direction
Earth Launch Date	2015 – 2035	Inclusive
Asteroid Stay Time	≥ 90 days	
Total Flight Time	≤ 20 years	
Spacecraft Initial Mass	1500 kg	1000 kg of available propellant
Thruster I_{sp}	4000 s	
Maximum Thrust	0.1 N	Can be turned on/off at will No constraint on direction

For reference, Table 20 lists the eleven solutions submitted that met all of the problem constraints. It is interesting to note that only five asteroids appear more than once in the submitted solutions, illustrating how large the design space is and how many different asteroid sequences can yield good solutions.

Table 20: Feasible solutions submitted to GTOC2.

Asteroid Sequence	V_∞ (km/s)	Final Mass (kg)	Time of Flight (yrs)	J (kg/yr)
2004 UT1 – Echo – Concordia – Scholl	3.50	898.2	9.106	98.64
2004 QA22 – Medusa – Misa – Guinevere	3.50	913.9	10.394	87.93
2003 YN107 – Reginhald – Dido – 1992 SU21	2.58	829.0	9.523	87.05
2003 YN107 – Pilcher – Vanadis – Cunningham	2.45	835.2	9.777	85.43
1999 AO10 – Photographica – Veritas – Potomac	2.18	861.0	10.096	85.28
2004 QA22 – Euterpe – Lydia – Tuckia	3.23	859.1	10.170	84.48
2005 QP11 – Chantal – Aglaja – 1998 QB32	3.50	890.5	10.796	82.48
2006 FH36 – Russia – Ceraskia – Cunningham	3.50	826.1	10.816	76.67
2003 YN107 – Ariadne – Galatea – Guinevere	2.46	864.1	11.509	75.08
2004 QA22 – Medusa – Oceana – Hohmann	3.50	735.9	12.941	56.87
2006 SP19 – Zelia – Eurydike – Nestor	3.50	536.3	19.195	27.94

The methodology was next applied to the GTOC2 problem, but because of the large size of the problem, this solution was tackled in two steps. First, a smaller version was solved, which has a reduced set of asteroids and approximately 400 million discrete asteroids sequences. Note that this problem is more than two orders of magnitude larger than the modified GTOC3 problem solved in Section 5.1. The modified version of the GTOC2 problem still has all of the same problem parameters and constraints as the original competition problem. Next, the methodology is applied to the full GTOC2 problem, given the time constraints of the GTOC2 competition.

5.2.1 Modified GTOC2 Problem

In creating the modified GTOC2 problem, the asteroids that yielded the top seven solutions from the competition – all with J > 80 kg/yr – were kept in the design space, in order to have a benchmark with which to evaluate the performance of the methodology. The remaining asteroids were chosen at random, keeping roughly the same proportion of asteroids from each group as in the original problem. Figure 51 plots the asteroids for the modified GTOC2 problem, as a function of inclination, eccentricity, and semi-major axis. There are 107 Group 4 asteroids, 95 Group 3 asteroids, 56 Group 2 asteroids, and 30 Group 1 asteroids. Based on the figure, the distribution of asteroids relative to inclination, eccentricity, and semi-major axis appears similar to the original problem (Figure 50).

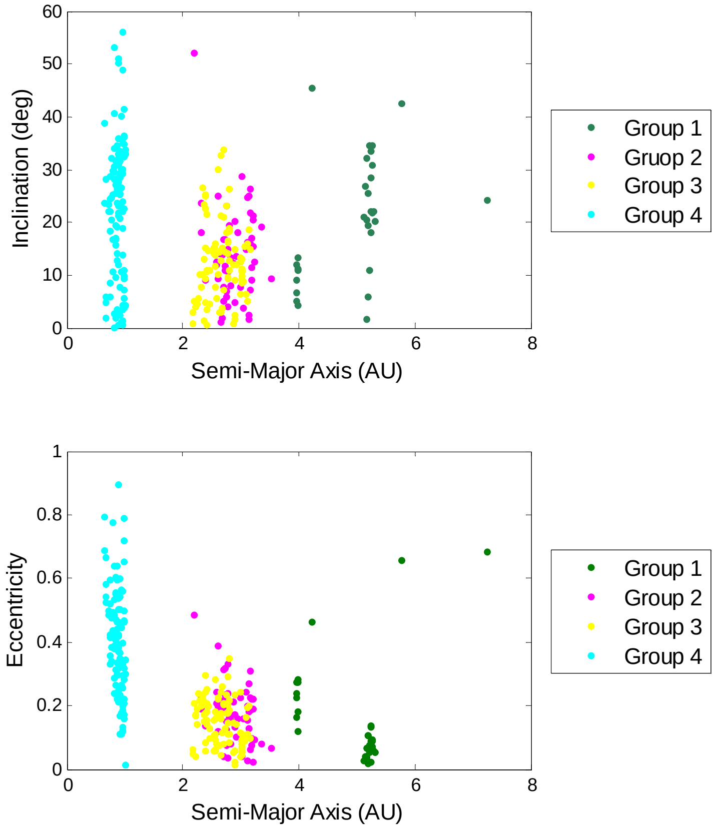

Figure 51: Set of asteroids for modified GTOC2 problem.

The best previously known solutions to the modified problem are listed in Table 20. The results of applying the methodology to the modified problem can then be compared to these results to determine how well this methodology performs at finding a set of good solutions.

The modified GTOC2 problem has 409,852,800 discrete asteroid sequences. The goal of the pruning phase will be to reduce this number by four orders of magnitude, to approximately 40,000 sequences. The first pruning metric removes all asteroid sequences that do not increase in semi-major axis from one asteroid to the next. From Figure 51, it is clear that all remaining sequences will visit a Group 4 asteroid first and a Group 1 asteroid last. Only the order of the Group 2 and Group 3 asteroids will need to be considered, greatly decreasing the amount of time required to apply this pruning metric. The increasing semi-major axis constraint reduces the number of asteroid sequences from just over 400 million to approximately 17 million (factor of 12). Next, the pruning percentages must be chosen in order to further reduce the number of asteroid sequences to around 40,000. The following percentages were chosen, keeping in mind that the percentage eliminated should decrease for each leg: 65% for Leg 1, 60% for Leg 2, 55% for Leg 3, and 40% for Leg 4. This reduces the number of asteroid sequences to 38,121, which achieves the desired order of magnitude reduction. Table 21 lists the number of asteroid sequences eliminated for each pruning metric as applied to each trajectory leg.

Table 21: Pruning methodology applied to modified GTOC2 problem.

Pruning Metric	Trajectory Leg	% Asteroid Pairs Eliminated	# Sequences Eliminated
$a_i < a_{i+1}$	All	N/A	39,277,560
wedge angle	Leg 1	65%	11,172,000
wedge angle	Leg 2	60%	3,558,120
wedge angle	Leg 3	55%	1,137,090
wedge angle	Leg 4	40%	443,608
impulsive ΔV	Leg 1	65%	462,524
impulsive ΔV	Leg 2	60%	174,646
impulsive ΔV	Leg 3	55%	67,941
impulsive ΔV	Leg 4	40%	23,150

Of the remaining sequences, the 1st, 3rd, 4th, and 7th best known solutions from Table 20 all remain in the design space.

Next, the global optimization phase is applied to the reduced design space, beginning with ranking the remaining asteroid sequences by the normalized sum of the pruning metrics. Due to the GTOC2 objective function, which includes the overall time of flight, a slight modification was made in the calculation of the optimal two-impulse solutions, which are used to determine the upper bounds during the branch-and-bound process. When the optimal two-impulse solutions were initially being calculated, it was discovered that many of the optimal solutions were falling in the very low time of flight range. As the low-thrust optimal solutions were calculated, it became apparent that most of the asteroid sequences that yield very good impulsive solutions at low times of flight do not translate to good low-thrust solutions. While low times of flight are possible for impulsive trajectories, low-thrust trajectories generally require longer times of flight to be feasible. Therefore, a restriction was placed on the minimum time of flight when calculating the impulsive solutions. This change was initially made in order to better model the low-thrust trajectories. However, it also served to reduce the number of low-thrust optimizations that would be required, since the optimal impulsive objective function for many of the asteroid sequences was greatly reduced when applying the time of flight restriction. For this problem, this minimum allowable time of flight for the impulsive solutions was chosen at 7 years, which was estimated as the minimum realistic time of flight for the low-thrust trajectories. Figure 52 plots the optimal impulsive solutions for the asteroid sequences remaining after the pruning phase, with and without a time of flight restriction.

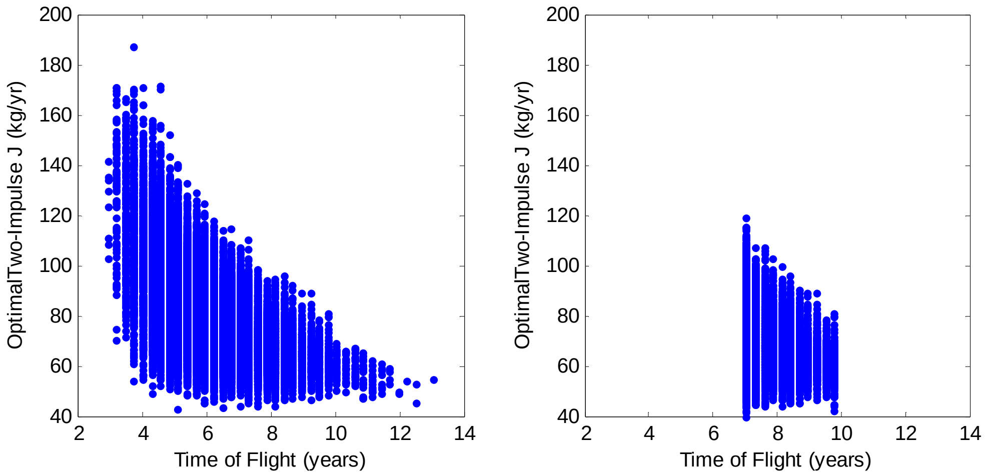

Figure 52: Optimal impulsive solutions, with and without the time of flight restriction, for the modified GTOC2 problem.

Table 22 lists the settings used for the genetic algorithm as applied to this problem, and Table 23 lists the design variables, bounds, and discretization. The number of bits for each design variable was chosen so that the discretization on all of the time variables was approximately equal. For the bits chosen, the variables are discretized in approximately seven day increments. Because of the greater number of low-thrust optimizations required in this problem, the genetic algorithm is run only once for each asteroid sequence. At the end of the branch-and-bound algorithm, the genetic algorithm is then run three times on the most promising solutions, in an attempt to improve upon their objective functions. Using these settings, on average, a single run of the genetic algorithm requires 1746 function calls (end-to-end optimizations by MALTO) and 49 generations to converge. With the available computing resources, each genetic algorithm run takes 70 minutes on average.

Table 22: Settings for the genetic algorithm as applied to the modified GTOC2 problem.

GA Setting	Value
Population Size	200
Stall Generations	20
Tournament Size	4
Crossover Proability	0.8
Mutation Probablity	0.1

Table 23: Design variables for the genetic algorithm as applied to the modified GTOC2 problem.

Design Variable	Units	# Bits	Lower Bound	Upper Bound
Earth Departure Date	JD	10	2457023	2464328
TOF, Leg 1	days	8	200	2000
TOF, Leg 2	days	8	200	2000
TOF, Leg 3	days	8	200	2000
TOF, Leg 4	days	8	200	2000
Stay Time, Ast. 1	days	5	90	360
Stay Time, Ast. 2	days	5	90	360
Stay Time, Ast. 3	days	5	90	360

Although a list of known solutions is available, the branch-and-bound algorithm is started without setting the best solution as the lower bound, since in the competition there were no a priori known solutions. Therefore, the first iteration of the branch-and-bound requires the low-thrust optimum to be computed for the highest ranked asteroid sequence (in terms of the normalized sum of the pruning metrics). The initial lower bound, J_{min}, is 40.29 kg/yr. This iteration is carried out without any multiplier on the impulsive solutions. Overall, the first iteration requires the low-thrust optimization of 809 asteroid sequences, which took approximately 39 days to complete. Additionally, it requires the impulsive optimization of all the asteroid sequences – none of the branches were pruned out until the final trajectory leg. The best solution found corresponds to the best solution found during the GTOC2 competition. Its objective function is 98.64 kg/yr, which ranks 674th in terms of the normalized sum of the pruning metrics. During the execution of the branch-and-bound algorithm, it is the 378th asteroid sequence for which the low-thrust optimum had to be computed. Because all of the low-thrust solutions are bounded by the impulsive solutions, no additional iterations need to be carried out.

Figure 53 plots the evolution of the branch-and-bound algorithm. Along the x-axis is the asteroid sequence number, ranked by the normalized sum of the pruning metrics. The green dots correspond to the low-thrust solutions, while the red dots correspond to the impulsive optima for each asteroid sequence. The blue line is the current lower bound on the objective function – it increases as better low-thrust solutions are found. The plot is shown for all 38,121 sequences and also for just the first 1000. Figure 54 plots all of the low-thrust and impulsive solutions, but sorted by the optimal impulsive objective function for the asteroid sequences. From here, it can be seen that all of the low-thrust solutions are indeed bounded, and the impulsive multiplier does not need to be updated.

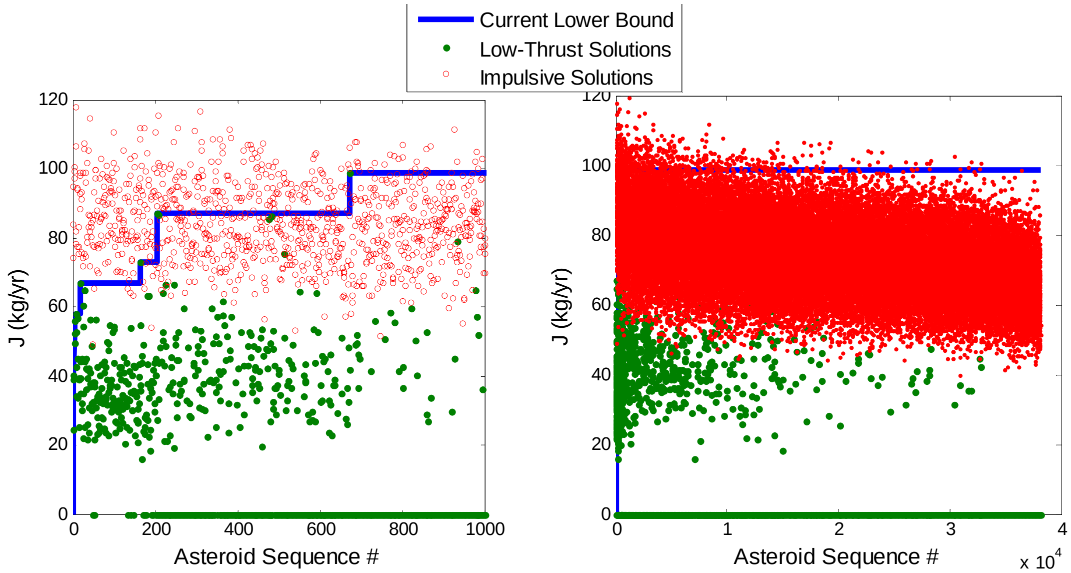

Figure 53: Results of branch-and-bound algorithm applied to modified GTOC2 problem.

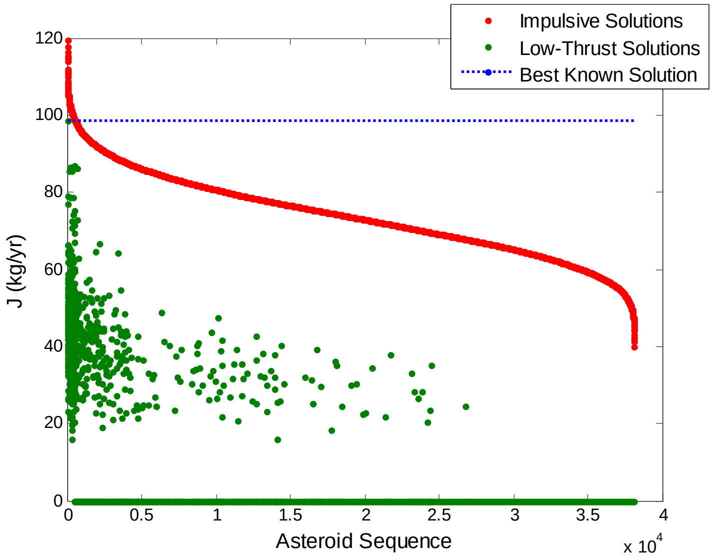

Figure 54: Comparison of impulsive and low-thrust optimal solutions for modified GTOC2 problem.

Table 24 illustrates another important observation, which is that six of the top seven solutions found are ranked in the top 2% of asteroid sequences in the reduced design space, based on the normalized sum of the pruning metrics. While the branch-and-bound algorithm was run to completion, the best solutions were found early on in the process. As has been shown in the previous problems, this can greatly reduce the number of low-thrust optimizations required, due to the lower bound being set to a high value early in the process. Additionally, however, if limited time were available to solve this asteroid rendezvous problem, such as during the competition, the branch-and-bound process would not need to be run to completion to find a handful of good solutions (and in this case, the best known solution). This approach will be explored further in the subsequent section, which addresses solving the full GTOC2 problem.

Table 24 summarizes the top seven solutions found by the methodology, all of which have J > 80 kg/yr. The "B&B Ranking" indicates the rank based on the normalized sum of the pruning metrics, as calculated for the branch-and-bound algorithm. The 1st, 3rd, 4th, and 7th best asteroid sequences from the GTOC2 competition were all found. Furthermore, a better solution was found for the asteroid sequence identified as the 7th best in the competition. The remaining competition solutions were eliminated during the pruning phase (none were pruned out by the branch-and-bound algorithm). Of note, three good sequences with J > 80 kg/yr (not identified by GTOC2 competitors) were also found through application of this methodology. With additional computing resources or additional time, less stringent pruning metrics could have been selected, highlighting the trade between pruning strength and available computational resources.

Table 24: Top seven solutions found by the methodology for the modified GTOC2 problem.

Asteroid Sequence	Final Mass (kg)	Time of Flight (yr)	J (kg/yr)	B&B Ranking	Notes
2004 UT1 – Echo – Concordia – Scholl	898.2	9.11	98.64	#674	#1 GTOC2 competition
2003 YN107 – Reginhald – Dido – 1992 SU21	829.0	9.52	87.05	#205	#3 GTOC2 competition
2003 YN107 – Echo – Concordia – Scholl	761.9	8.81	86.54	#210	New sequence identified in this investigation
2003 YN107 – Chantal – Aglaja – 1998 QB32	834.8	9.67	86.29	#482	New sequence identified in this investigation
2003 YN107 – Pilcher – Vanadis – Cunningham	835.2	9.78	85.43	#4467	#4 GTOC2 competition
2005 QP11 – Chantal – Aglaja – 1998 QB32	812.9	9.52	85.39	#476	Better solution to #7 GTOC2 sequence
2003 YN107 – Euterpe – Lydia – Tuckia	722.9	8.73	82.83	#785	New sequence identified in this investigation

5.2.2 Full GTOC2 Problem

The methodology is then applied to the full GTOC2 problem, using the same constraints as were given during the competition. The competition problem was released on November 6, 2006 and the deadline to submit results was December 4, 2006, a period of exactly four weeks. As a benchmark, Table 25 lists the top ten previously known solutions to the full GTOC2 problem. Note that seven of these solutions are from the competition, while the remaining three were identified in this investigation while solving the modified GTOC2 problem.

Table 25: Top ten known solutions to full GTOC2 problem.

Asteroid Sequence	Final Mass (kg)	Time of Flight (yrs)	J (kg/yr)
2004 UT1 – Echo – Concordia – Scholl	898.2	9.106	98.64
2004 QA22 – Medusa – Misa – Guinevere	913.9	10.394	87.93
2003 YN107 – Reginhald – Dido – 1992 SU21	829.0	9.523	87.05
2003 YN107 – Echo – Concordia – Scholl	761.9	8.81	86.54
2003 YN107 – Chantal – Aglaja – 1998 QB32	834.8	9.67	86.29
2003 YN107 – Pilcher – Vanadis – Cunningham	835.2	9.777	85.43
2005 QP11 – Chantal – Aglaja – 1998 QB32	812.9	9.52	85.39
1999 AO10 – Photographica – Veritas – Potomac	861.0	10.096	85.28
2004 QA22 – Euterpe – Lydia – Tuckia	859.1	10.170	84.48
2003 YN107 – Euterpe – Lydia – Tuckia	722.9	8.73	82.83

The pruning phase is applied to the full set of asteroid sequences, with a goal of reducing the size of the design space by four orders of magnitude. First, only asteroid sequences with increasing semi-major axes are kept in the design space. This reduces the number of asteroid sequences by a factor of 24, from 41 billion to 1.7 billion. Next, the following pruning percentages are applied to the problem, for θ_{wedge} and ΔV_{opt}: 65% for Leg 1, 60% for Leg 2, 50% for Leg 3, and 35% for Leg 4. This further reduces the number of asteroid sequences to 3.9 million. After the pruning phase, seven of the ten solutions from Table 25 remain in the design space.

Next, the global optimization phase is allowed to run for two weeks. Although the competition allowed for four weeks to complete the problem, two weeks were reserved for initial setup time and for refining the final solution and formatting it according to the competition guidelines. In order to limit the number of low-thrust optimizations required, an aggressive estimate for the optimum solution is chosen, based on a time of flight of 10 years and a final mass of 1000 kg. This corresponds to an objective function of 100 kg/yr, which serves as the initial lower bound in the branch-and-bound algorithm. The genetic algorithm is run only once for each asteroid sequence that requires low-thrust optimization, using the settings listed in Table 22 and Table 23, from the modified GTOC2 problem. During the two weeks, 308 asteroid sequences were optimized in low-thrust, which encompassed the top 3,000 ranked asteroid sequences in the branch-and-bound tree, based on the normalized sum of the pruning metrics. Figure 55 plots the evolution of the branch-and-bound algorithm over the two weeks, with the lower bound plotted in blue (100 kg/yr), the optimal two-impulse solutions plotted in red, and the optimal low-thrust solutions found by the genetic algorithm plotted in green. All of the low-thrust optima calculated are bounded by their corresponding two-impulse solutions, as illustrated in Figure 56.

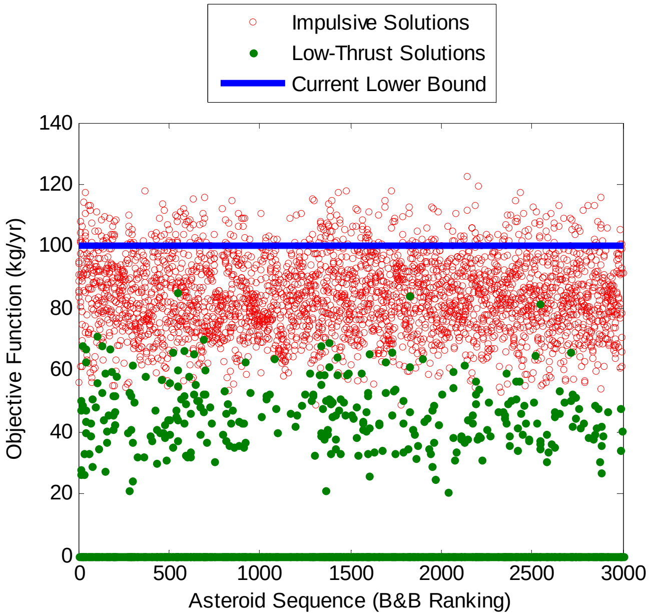

Figure 55: Results of branch-and-bound algorithm applied for two weeks to full GTOC2 problem.

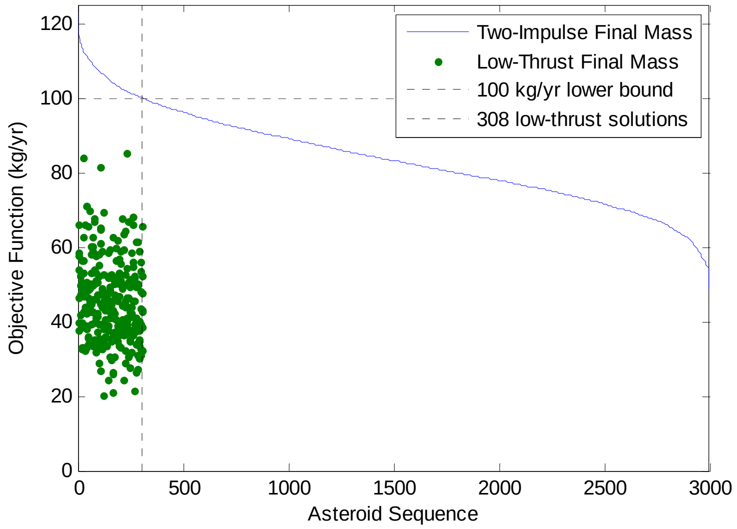

Figure 56: Comparison of impulsive and low-thrust optimal solutions for full GTOC2 problem.

While none of the previously known solutions in Table 25 are found, three new asteroid sequences with J > 80 kg/yr are found. The best of these has a low-thrust optimum of 87.31 kg/yr, which would have ranked 3rd in the GTOC2 competition. The asteroid sequence that yields this solution is: 2003 YN107 – Phyllis – Budrosa – Kostinsky. It departs Earth on December 1, 2023, and has times of flight for each leg of 539 days, 1605 days, 511 days, and 645 days, respectively, for a total mission time of 9.77 years. The final mass is 853 kg. The second new sequence found with an objective function greater than 80 kg/yr is: 2003 YN107 – Phyllis – Hygiea – Guinevere. Its optimum solution yields an objective function of 83.88 kg/yr, with a total time of flight of 9.35 years and a final mass of 784 kg. The third new good sequence found is: 2001 FR85 – Erigone – Rosselia – Scholl, which has an objective function of 81.47 kg/yr, with a total time of flight of 10.09 years and a final mass of 822 kg.

If more time or computing resources were available, additional good solutions could be found for the full GTOC2 problem. Still assuming the same four orders of magnitude reduction during the pruning phase, Table 26 lists all of the known solutions greater than 80 kg/yr, along with their rank in the branch-and-bound algorithm, according to the normalized sum of the pruning metrics ("Rank in Pruned Problem"). The 2nd, 9th, and 10th best known solutions were eliminated during the pruning phase, and therefore have no results listed in the table. Because many of the branches of the tree are pruned out based on their relaxed two-impulse solutions, only a subset of these sequences must be optimized in low-thrust.

Table 26: Known solution to GTOC2 problem with J > 80 kg/yr.

Asteroid Sequence	J (kg/yr)	Rank in Pruned Problem	# Low-Thrust Opt. Required
#1, GTOC2	98.64	74,505	4,514
#2, GTOC2	87.93	---	---
New	87.31	547	74
#3, GTOC2	87.05	24,003	1,867
New	86.54	24,524	1,890
New	86.29	76,812	4,620
#4, GTOC2	85.43	801,999	---
#7, GTOC2 (improved)	85.39	56,030	3,663
#5, GTOC2	85.28	---	---
#6, GTOC2	84.48	---	---
New	83.88	1821	203
New	82.83	124,226	---
New	81.47	2,541	272

Table 26 also lists the number of low-thrust optimizations that are required to find each sequence in the table ("# Low-Thrust Opt. Required"), assuming an initial lower bound of 100 kg/yr and also assuming that no solutions greater than 100 kg/yr are found. If solutions greater than 100 kg/yr are found, fewer low-thrust optimizations would be required – more branches of the tree would be pruned out based on the corresponding increase in the lower bound – although some of the solutions in Table 26 may be pruned out as well. The two-impulse optima were only calculated for the first 78,000 asteroid sequences, which encompass the top 2% of the sequences remaining after the pruning phase, based on the results of the modified GTOC2 and modified GTOC3 problems. For these two problems, the best known solutions were all located within the top 2% of the ranked sequences in the branch-and-bound algorithm. Since the 7th and 12th best known solutions (listed in Table 26) fall outside this range, the number of low-thrust optimizations required to locate these two solutions is unknown.

Figure 57 plots the number of asteroid sequences that require low-thrust optimization as a function of the number of asteroid sequences evaluated in the branch-and-bound, for the first 78,000 ranked sequences (plotted in red are the data points from Table 26). In other words, Figure 57 illustrates the efficiency of the branch-and-bound algorithm at pruning out branches of the tree using the relaxed two-impulse solutions. In the range considered, approximately seven percent of the sequences in the branch-and-bound tree require low-thrust optimization – the rest are pruned out due to their optimal two-impulse solution being less than the lower bound of 100 kg/yr. As the number of sequences evaluated in the branch-and-bound tree increases, the percent that require low-thrust optimization decreases, due to the fact that the optimal two-impulse solutions tend to decrease as a function of their ranking in the branch-and-bound algorithm. This trend was observed previously in Figure 53, which plots the evolution of the branch-and-bound algorithm for the modified GTOC2 problem. The number of low-thrust optimizations required as a function of the number of branch-and-bound sequences evaluated can also be plotted for the modified GTOC2 problem, in order to illustrate this trend for a problem where the entire branch-and-bound tree was evaluated. As expected, Figure 58 illustrates how the percent of sequences that require low-thrust optimization decreases towards the end of the branch-and-bound algorithm.

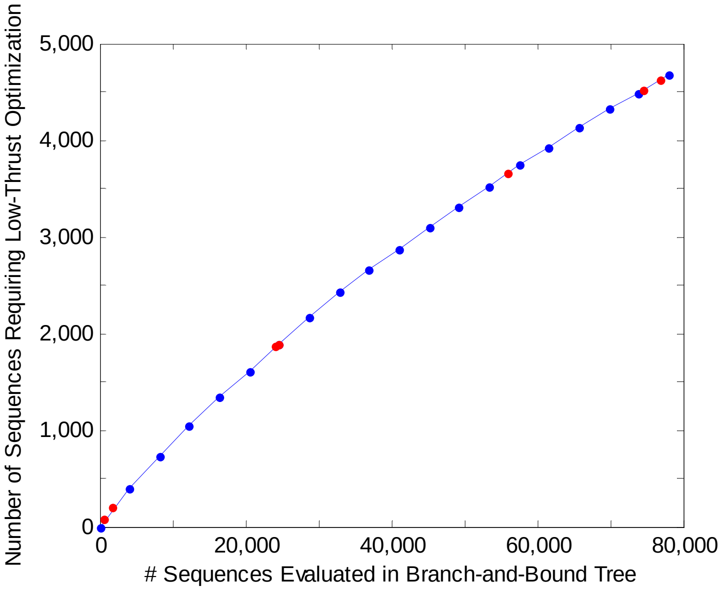

Figure 57: Number of sequences requiring low-thrust optimization as a function of the number of sequences evaluated in the branch-and-bound tree, for the full GTOC2 problem, assuming an initial lower bound of 100 kg/yr.

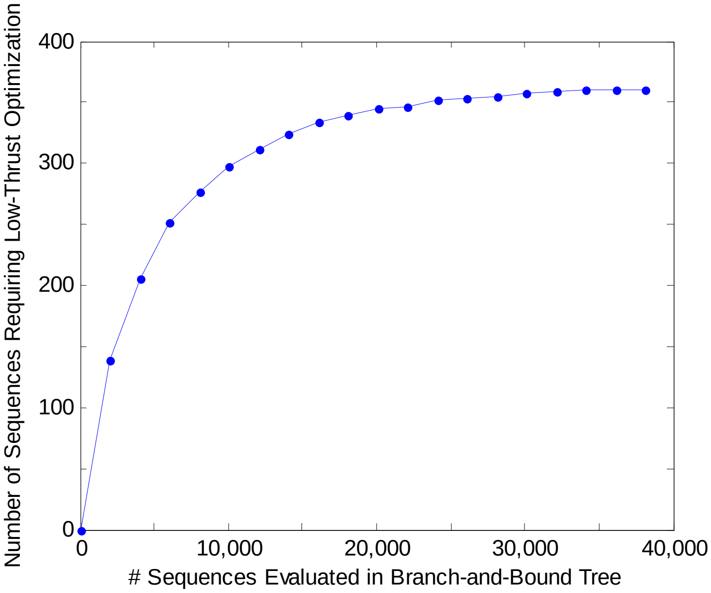

Figure 58: Number of sequences requiring low-thrust optimization as a function of the number of sequences evaluated in the branch-and-bound tree, for the modified GTOC2 problem, assuming an initial lower bound of 100 kg/yr.

Based on the results in Table 26 and Figure 57, in order to locate eight of the top thirteen solutions to the full GTOC2 problem, the low-thrust optimization of 4,620 asteroid sequences would be required, from a total of over 41 billion sequences in the original problem (a reduction of seven orders of magnitude). In order to locate the winning solution to GTOC2, the low-thrust optimization of 4,514 asteroid sequences would be required. With improvements in the parallel distribution of the low-thrust optimizations, along with efficiency improvements in the various codes used (e.g., switching from Matlab to Fortran), it is likely possible to complete this number of genetic algorithm executions in the month timeframe. Furthermore, it is likely that a number of additional solutions with J > 80 kg/yr would be located in those 4,620 sequences, based on the fact that three new good solutions were found in just the first 308 sequences in the branch-and-bound tree.

5.3 Sensitivity of Methodology to User-Defined Parameters

In the previous two sections, it was shown how the methodology was able to successfully identify the best known solution as well as a family of good solutions for the modified GTOC2 and modified GTOC3 problems. For the modified GTOC2 problem, the design space was pruned by four orders of magnitude and the low-thrust optimization of 809 asteroid sequences was required. For the modified GTOC3 problem, the design space was pruned by two orders of magnitude and the low-thrust optimization of 17 asteroid sequences was required. For the full GTOC2 problem, it was shown that in two weeks, three previously unknown solutions were identified with J > 80 kg/yr, the best of which would have placed third in the competition. Furthermore, for the full GTOC2 problem, it would be possible to identify eight of the top thirteen known solutions with J > 80 kg/yr after the low-thrust optimization of 4,620 asteroid sequences.

These results are based on particular values of the pruning metrics, along with additional assumptions such as the initial value of the lower bound for the branch-and-bound algorithm. The number of required low-thrust optimizations is primarily dependent on the reduction in the design space that can be achieved by the pruning phase of the methodology and the initial lower bound set by the highest ranked sequence following the pruning step. This section will examine the sensitivity to the various user-defined parameters on the performance of the methodology, demonstrating how this methodology can be tailored to the amount of computation time available.

5.3.1 Pruning Phase Sensitivity to Selection of Leg Pruning Percentages

For the pruning phase of the methodology, the largest reduction has been by a factor of approximately four orders of magnitude on the modified and full versions of the GTOC2 problem. In achieving that reduction, however, several of the best known solutions were eliminated. For the modified GTOC2 problem, Table 27 illustrates how many known solutions are eliminated during the pruning phase, depending on the user-defined pruning percentages. The table includes only solutions known prior to the evaluation of the modified GTOC2 problem – it does not include the new solutions found by the methodology. The percentages presented in each column of the table represent the percent of asteroid pairs eliminated for each leg of the trajectory for both pruning metrics. Once the pruning percentage applied to Leg 1 (k_1) reaches approximately 70%, the best known solution is eliminated. Furthermore, once the overall reduction in the number of asteroid sequences exceeds approximately four orders of magnitude, all of the best known solutions are eliminated from the design space. Conversely, in order to keep all of the best known solutions in the design space, less than two orders of magnitude can be eliminated during the pruning phase.

Table 27: Best known solutions remaining in design space for varying orders of magnitude reduction during the pruning phase, for the modified GTOC2 problem.

Asteroid Sequence	J (kg/yr)	5 orders of magnitude	4 orders of magnitude	3 orders of magnitude	2 orders of magnitude	<2 orders of magnitude
		4,781 sequences remaining	38,121 sequences remaining	470,791 sequences remaining	4,248,084 sequences remaining	4,765,633 sequences remaining
		75%-70%-65%-50%	65%-60%-55%-40%	50%-45%-40%-25%	25%-20%-15%-10%	20%-20%-15%-10%
#1, GTOC2	98.64		√	√	√	√
#2, GTOC2	87.93				√	√
#3, GTOC2	87.05		√	√	√	√
#4, GTOC2	85.43		√	√	√	√
#7, GTOC2	85.39[*]		√	√	√	√
#5, GTOC2	85.28					√
#6, GTOC2	84.48				√	√

The same analysis was conducted on the full GTOC2 problem, again using just the solutions known before the methodology was applied, which now include the additional three good solutions found on the modified GTOC2 problem. The results are presented in Table 28. Seven of the top ten known solutions remain in the design space of the full GTOC2 problem if the pruning phase reduces the number of sequences by four orders of magnitude. If that reduction is increased to five orders of magnitude, however, only one of the best known solutions remains. Note that to reach approximately the same number of remaining sequences as in the modified GTOC2 problem, a six order of magnitude pruning reduction is required.

[*] The objective function of 85.36 kg/yr represents the improved solution found for the 7th place GTOC2 asteroid sequence.

Table 28: Best known solutions remaining in the design space for varying orders of magnitude reduction during the pruning phase, for the full GTOC2 problem.

Asteroid Sequence	J (kg/ yr)	6 orders of magnitude	5 orders of magnitude	4 orders of magnitude	3 orders of magnitude	2 orders of magnitude
		46,661 sequences remaining	490,897 sequences remaining	3,917,173 sequences remaining	41,153,546 sequences remaining	425,208,487 sequences remaining
		80%-75%-70%-50%	75%-70%-60%-40%	65%-60%-50%-35%	45%-45%-40%-25%	25%-20%-15%-10%
#1, GTOC2	98.64			√	√	√
#2, GTOC2	87.93					√
#3, GTOC2	87.05			√	√	√
New Seq.	86.54	√	√	√	√	√
New Seq.	86.29			√	√	√
#4, GTOC2	85.43			√	√	√
#7, GTOC2	85.39			√	√	√
#5, GTOC2	85.28					
#6, GTOC2	84.48					√
New Seq.	82.83			√	√	√

Finally, this pruning analysis is conducted for the modified GTOC3 problem, the results of which are presented in Table 29. When the methodology was applied in Section 5.1, the pruning phase achieved a reduction in the design space of approximately two orders of magnitude, from 2.7 million sequences to 10,311. For this problem, however, a reduction of up to four orders of magnitude could have been achieved without eliminating any of the ten best known solutions. Beyond four orders of magnitude, however, a majority of these solutions are eliminated.

Table 29: Best known solutions remaining in design space for varying orders of magnitude reduction during the pruning phase, for the modified GTOC3 problem.

Asteroid Sequence	M_f (kg)	4+ orders of magnitude	4 orders of magnitude	3 orders of magnitude	2 orders of magnitude
		65 sequences remaining	299 sequences remaining	2,306 sequences remaining	10,311 sequences remaining
		85%-80%-75%-60%	80%-75%-70%-60%	75%-70%-60%-40%	70%-60%-50%-25%
E - 76 - 88 - 49 - E	1621		√	√	√
E - 88 - 76 - 49 - E	1597	√	√	√	√
E - 49 - 37 - 85 - E	1590	√	√	√	√
E - 96 - 88 - 49 - E	1589		√	√	√
E - 88 - 19 - 49 - E	1587	√	√	√	√
E - 88 - 49 - 19 - E	1567	√	√	√	√
E - 96 - 76 - 49 - E	1565		√	√	√
E - 88 - 11 - 49 - E	1558	√	√	√	√
E - 88 - 129 - 49 - E	1557		√	√	√
E - 88 - 76 - 96 - E	1554	√	√	√	√

5.3.2 Global Optimization Phase Sensitivity to Selection of Initial Lower Bound

For the global optimization portion of the methodology, the algorithm can be tuned based on the initial value of the lower bound chosen for the branch-and-bound algorithm. Shown previously, Figure 53 illustrates the evolution of the branch-and-bound algorithm for the modified GTOC2 problem. In this figure, all of the asteroid sequences with impulsive optima (plotted in red) greater than the current lower bound (plotted in blue) must be optimized in low-thrust. These results were based on running the branch-and-bound algorithm without an initial value for the lower bound – therefore, the lower bound is set by applying the genetic algorithm to the first asteroid sequence to determine its low-thrust optimum. If an estimate is made for the initial value of the lower bound based on the underlying physics of the problem, this would eliminate some of the up-front low-thrust optimizations, few of which generally yield good solutions.

For the modified GTOC2 problem, without an initial estimate of the lower bound of the objective function, the low-thrust optimization of 809 asteroid sequences was required. Figure 59 plots the number of optimizations that would have been required for different values of the initial lower bound. Of course, if the initial lower bound is set too high, it is possible that all solutions will be pruned out by the branch-and-bound algorithm. For this particular problem, the initial bound can be set as high as 105.9 kg/yr, and still find all of the known solutions greater than 80 kg/yr (105.9 represents the minimum value of the impulsive optima for the known asteroid sequences with low-thrust optima greater than 80 kg/yr). In practice, however, if the initial lower bound is set too high and no good solutions are found, it could be incrementally decreased until a satisfactory set of good solutions were found.

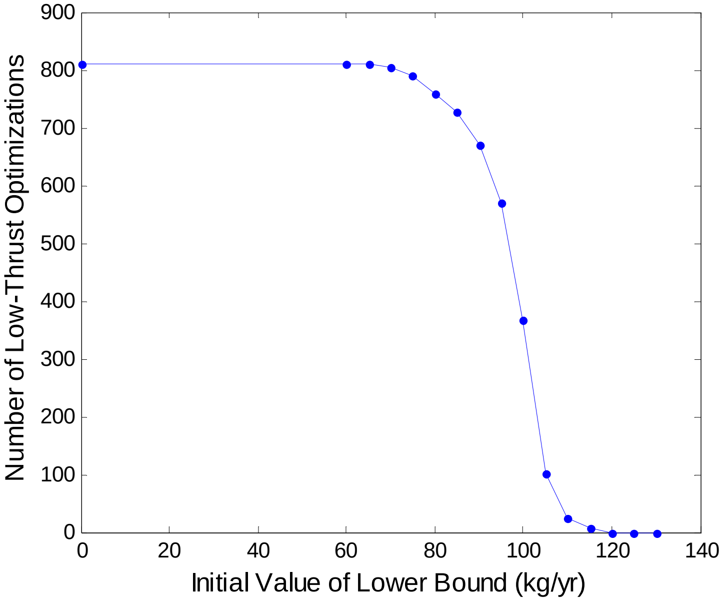

Figure 59: Number of low-thrust optimizations required as a function of the initial value of the lower bound during the branch-and-bound algorithm for the modified GTOC2 problem.

The same plot is constructed for the full GTOC2 problem in Figure 60. The number of low-thrust optimizations required in this case, however, only encompasses the first 78,000 sequences, which represents the top 2% of sequences based on the branch-and-bound ranking (normalized sum of the pruning metrics). Additionally, the plotted results are based only on the known low-thrust solutions (see Table 26). If additional good solutions exist in the design space, however, it is possible that the number of low thrust optimizations would decrease. It is also possible that if low-thrust solutions exist in which the corresponding two-impulse optima are better than the best known solutions, that some of these sequences would be pruned out during the branch-and-bound algorithm. Therefore, the results in Figure 60 are approximate, but illustrate the benefit of choosing an aggressive estimate for the initial lower bound.

As with the modified GTOC2 problem, for the full GTOC2 problem, 105.9 kg/yr is the maximum value of the lower bound that still locates all of the known solutions in the top 78,000 with J > 80 kg/yr. Setting the initial lower bound to this value would reduce the number of low-thrust optimizations from nearly 30,000 to 1,394. As seen previously, if an initial estimate of 100 kg/yr was made, 4,620 asteroid sequences would require low-thrust optimization.

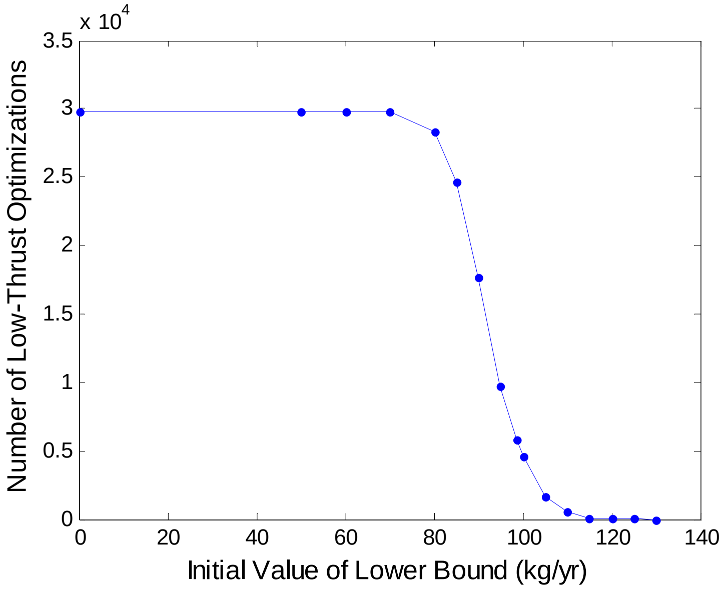

Figure 60: Number of low-thrust optimizations required as a function of the initial value of the lower bound during the branch-and-bound algorithm (top 78,000 sequences) for the full GTOC2 problem.

CHAPTER VI

CONCLUSIONS AND RECOMMENDATIONS

This work presents an organized search technique for the low-thrust mission design of multiple asteroid rendezvous mission, at the conceptual design level. Specifically, a methodology is developed for quickly determining a broad set of good solutions to the large, combinatorial asteroid rendezvous problem. The proposed methodology combines two-steps, the first which quickly eliminates poor solutions from the design space through a three-level heuristic pruning sequence, and the second which locates a set of good solutions from the reduced design space using a global optimization algorithm. This approach combines an innovative branch-and-bound algorithm (to solve for the optimal asteroid sequence) with a genetic algorithm (which, in tandem with a low-thrust trajectory optimization program, solves for the optimal departure dates, times of flight, and stay times for a given asteroid sequence). This methodology was applied to several problems, ranging in size from several thousand possible asteroid sequences to over 41 billion. The methodology is able to consistently locate the best known solution, along with a suite of good solutions across the design space. The performance of the proposed methodology at efficiently pruning the design space and then locating the best set of solutions is summarized in this chapter.

6.1 Performance of Methodology

6.1.1 Pruning Phase

The goal of the pruning phase of the methodology is to quickly reduce the size of the design space by several orders of magnitude, while maintaining a majority of the best solutions. In order to do so, a sequence of heuristics was developed, specific to the physics of the underlying asteroid tour problem, which is able to identify areas of the design space that will not likely yield favorable solutions in terms of final mass. The heuristics chosen were based on their effectiveness at pruning a small sample problem, and were then verified on an intermediate sample problem with a known solution, before being applied to the GTOC3 and GTOC2 competition problems. The three metrics chosen for the pruning phase are as follows:

1) ***Increasing semi-major axis***. Only asteroid sequences where the semi-major axis increases from one asteroid to the next are kept in the design space. The rationale behind this metric is that visiting asteroids in sequential order in terms of distance from the Earth minimizes propellant expenditure. In order to eliminate phasing, semi-major axis is used as a surrogate for distance. Additionally, the restriction is limited to increasing semi-major axis in order to minimize the time of flight. For problems where flight time is not a consideration, and the Earth departure V_∞ is large, it may make sense to visit the furthest asteroid first and then visit the remaining asteroids in order of decreasing distance. Furthermore, if return to Earth is required at the end of the trajectory, decreasing order could also be considered. For certain problem, such as the GTOC3 problem, where all of the asteroids have similar values of semi-major axis, this pruning metric may not be applicable.

2) ***Angle between the angular momentum vectors.*** This metric is applied to asteroid pairings. The angle between the angular momentum vectors takes two factors into account: the difference in inclination and the relative orientation between two orbits. In general, as the change in inclination increases between an initial and final orbit, so does the ΔV required and therefore the propellant required. However, inclination change is not sufficient, in that the orientation between two orbits must be considered. In Section 2.5.1, it was shown that the angle between the angular momentum vectors of the two orbits has a strong

correlation with the propellant mass required. As with the first metric, phasing is not taken into consideration here.

3) ***Optimal, phase-free, two-impulse delta-V.*** This third metric is also applied to asteroid pairings, and uses the optimal, phase-free, two-impulse ΔV as a surrogate for low-thrust propellant mass. While phasing was initially considered, there was not a strong correlation between the impulsive and low-thrust propellant masses for specific departure dates and/or flight times. Therefore, phasing is not taken into consideration here. The optimal, phase-free, two-impulse delta-V for a given asteroid pair is determined using a Lambert solver, by solving for the minimum ΔV over discretized values of departure and arrival true anomaly. This heuristic is only applicable for low-thrust trajectories that can be well approximated by impulsive solutions, namely those with low ratios of thrust time to time of flight.

In the asteroid rendezvous problem, the pruning metrics are applied sequentially in the order presented. The order was chosen to minimize the time required for this phase, since each subsequent metric requires additional computation time. Furthermore, the second and third metrics are applied sequentially to each trajectory leg, and a user-defined percentage is eliminated from each. These percentages are chosen in order to balance the desired reduction in the number of asteroid sequences remaining in the design space with the risk of eliminating good solutions.

Table 30 presents a summary of the results of the pruning phase of the methodology applied to each of the problems examined. The results presented are for the baseline pruning percentages used in this work; Section 5.3 presented sensitivities to the pruning metrics for the large problems. For the small and intermediate sample problems, the optimal (discretized) solution is known for all asteroid sequences. Therefore, the performance of the pruning phase is known absolutely. For the modified GTOC3 problem, the best set of solutions is unknown, so the performance of the pruning phase is compared against the best known set of asteroid sequences identified in the competition, excluding the Earth gravity assists. In this case, the asteroid sequences that generated the top ten competition solutions all remain in the design space.

Table 30: Summary of pruning phase applied to each problem.

Problem Description	Initial # Sequences	Final # Sequences	% Remaining	Remaining Sequences
Small Sample Problem	3.07×10^3	1.31×10^2	4.3%	– Top 36 sequences all remain
Intermediate Sample Problem	1.04×10^4	4.16×10^2	4.0%	– 23 of top 24 sequences remain – 7th best eliminated
Modified GTOC3 Problem	2.69×10^6	1.03×10^4	0.38%	– All of top 10 competition solutions remain
Modified GTOC2 Problem	4.10×10^8	3.31×10^4	0.0093%	– Of the top 7 competition solutions > 80 kg/yr, #1, #3, #4, and #7 remain
Full GTOC2 Problem	4.11×10^{10}	3.92×10^6	0.0095%	– Of the top 10 known solutions > 80 kg/yr (includes additional solutions found in modified problem), all but #2, #8, and #9 remain.

Similarly, for the modified and full GTOC2 problems, the best set of solutions is unknown, so the pruning phase is again compared against the best known solutions identified during the competition. In this case, that is considered to be all of the competition solutions with J > 80 kg/yr. For the modified GTOC2 problem, four of the top seven best sequences remain in the design space after the pruning phase is applied. Three additional solutions not reported by any of the GTOC2 competitors, all greater than 80 kg/yr, are also found during the global optimization phase, increasing the number of known solutions greater than 80 kg/yr to ten. For the full GTOC2 problem, seven of these top ten best known solutions remain after the pruning phase. Therefore, for each problem examined, it can be concluded that the pruning phase succeeded in keeping a majority of the best solutions in the design space.

The performance of the pruning phase is clearly dependent on the user-defined percent reduction in the size of the design space. If a smaller number of sequences are eliminated, a larger percentage of the best solutions remain. As was seen in Section 5.3, the design space can generally be reduced by up to four orders of magnitude, while still maintaining a majority of the best solutions.

6.1.2 Global Optimization Phase

The goal of the global optimization phase is to identify the best set of solutions from the reduced design space, where the system-level optimizer is responsible for determining the following design variables: asteroid combination, launch date, times of flight, and stay times. The system-level optimizer is coupled with a local low-thrust trajectory optimization program (MALTO) that determines the optimal control history of the spacecraft in order to minimize propellant for a given set of system-level variables. A three-level global optimization scheme was developed, which if run to completion, identifies both the global optimum and a set of best solutions, which can then be carried forward into the more detailed design phases. This strategy was described in Figure 12 in Section 2.3. The outer loop optimizer consists of a branch-and-bound algorithm, which is responsible for identifying the optimal asteroid sequence. This novel approach to branch-and-bound uses optimal two-impulse solutions as a surrogate for constraint relaxations to set bounds on the problem. As in the pruning phase, this assumption only holds for low-thrust trajectories where the ratio of the thrust time to time of flight is small, which is the case for the problems examined in this work. The inner loop optimizer, which is called for each asteroid sequence that requires optimization, consists of a genetic algorithm. The inner loop variables are the departure date, times of flight, and asteroid stay times.

Finally, for each set of inner loop design variables, MALTO is used as the low-thrust trajectory optimization algorithm. MALTO is responsible for determining the maximum final mass for a given set of outer and inner loop design variables.

Table 31 presents the results of the global optimization phase as applied to each of the asteroid tour problems. In all cases, the best known solution was found. By nature, the branch-and-bound algorithm should always locate the optimal solution, contingent on two conditions. First, the optimal impulsive solution must not incorrectly prune out the optimal sequence – the purpose of including the impulsive multiplier is to reduce the probability of this occurring. Second, the genetic algorithm must find the optimal solution to that particular asteroid sequence. Ideally, the genetic algorithm is run several times – the baseline value is three – in order to improve the chances of the optimal solution being found. Because of the large size of the modified and full GTOC2 problems, the genetic algorithm was nominally run only once for each asteroid sequence. The most promising cases were then run twice more in an attempt to improve those solutions.

The most important performance consideration of the global optimization phase is how many of the best solutions are found by the optimizer, which is presented in Table 31 for each of the problems examined. In order to evaluate this performance, the top ten previously known solutions for each problem were considered (top seven known solutions for the modified GTOC2 problem, since only the top seven competition solutions had values of $J > 80$ kg/yr). Out of those top ten solutions that remained after the pruning phase, the number found by the multi-level optimization scheme was determined. For the modified GTOC3 problem, eight of these top ten solutions were found, and for the modified GTOC2 problem, all (four) of these previously identified solutions were found. Additionally, three new solutions with $J > 80$ kg/yr not reported by any of the GTOC2 competitors were identified in this investigation for the modified GTOC2 problem.

Table 31: Summary of global optimization phase applied to each problem.

Problem Description	# Branch-and-Bound Iterations	# Sequences Requiring Low-Thrust Optimization	Best Set of Sequences Found
Small Sample Problem	2	14	– Optimal solution found – 5 of the top 10 remaining solutions found
Intermediate Sample Problem	2	55	– Optimal solution found – 5 of top 9 remaining solutions found
Modified GTOC3 Problem	3	17	– Best known solution found – 8 of the top 10 remaining solutions found
Modified GTOC2 Problem	1	809	– Best known solution found – 7 solutions found with J>80 kg/yr: – 4 of 4 competition solutions remaining after pruning phase – 3 new solutions not reported in competition
Full GTOC2 Problem	N/A	308 (in 2 weeks)	– Three new solutions with J>80 kg/yr found that were not reported in competition – Better solution would have ranked 3rd in GTOC2

For the full GTOC2 problem, the branch-and-bound algorithm was allowed to run for two weeks to simulate the time constraints from the competition. In those two weeks, 308 asteroid sequences were optimized in low-thrust. Of these, three previously unknown asteroid sequences were identified with J > 80 kg/yr, the best of which would have placed third in the competition. Furthermore, Section 5.2.2 analyzed how many low-thrust optimizations would be required to locate the additional known good solutions. It was determined that using an initial guess for the lower bound of 100 kg/yr (same assumption as was made in the two weeks of analysis), 4,620 asteroid sequences would have to be optimized in low-thrust in order to locate eight of the thirteen known solutions with J > 80 kg/yr. The thirteen known solutions include the seven reported solutions from the GTOC2 competition along with the six additional solutions found in this work.

6.1.3 Overall Performance

Table 32 summarizes the overall performance of the proposed methodology on each of the problems presented in this study (the full GTOC2 problem is not included in these results as the methodology was not run to completion for this problem). Two metrics are of importance here. First is the ratio of the number of asteroid sequences that require low-thrust optimization to the number of sequences in the initial design space. This is a measure of the efficiency that can be achieved by the overall methodology. For the modified GTOC3 and modified GTOC2 problems, this results in an overall reduction of between 6 and 7 orders of magnitude. Second is the number of good solutions found by the overall methodology. Because the best solutions are not known for the GTOC2 and GTOC3 problem, the solutions previously identified during the competitions are used as a benchmark.

Tailoring of the methodology to the amount of computing time available has also been demonstrated. For the large problems examined, it is possible to reduce the number of asteroid combinations by up to four orders of magnitude and still keep a majority of the best known solutions in the design space. Clearly, based on user-assigned values, the design space could be pruned less aggressively at the expense of more required low-thrust optimizations during the global optimization phase. Another user-defined setting, the initial lower bound in the branch-and-bound algorithm, was also shown to directly affect the number of required low-thrust optimizations required and the number of good solutions found. An aggressive estimate can serve to greatly reduce the number of low-thrust optimization required; however, it can also prune out some of the good solutions if set too high. A strategy was developed in which this value could be set high based on the underlying physics of the problem and incrementally lowered until a satisfactory set of good solutions was identified.

Table 32: Summary of overall performance of methodology as applied to each problem.

Problem Description	Initial # Sequences	# Low-Thrust Optimizations	$\frac{\#\,\mathrm{LT}_{\mathrm{OPT}}}{\#\,\mathrm{Sequences}}$	# Top 10 Known Solutions Found
Small Sample Problem	3.07×10^{3}	14	0.45%	5
Intermediate Sample Problem	1.04×10^{4}	55	0.53%	5
Modified GTOC3 Problem	2.69×10^{6}	17	0.00063%	8
Modified GTOC2 Problem	4.10×10^{8}	809	0.00020%	7[*]

6.2 Conclusions

Based on the results presented above, the methodology developed in this investigation can be concluded to be effective at locating both the best known solution and a set of good solutions for low-thrust, combinatorial, asteroid rendezvous problems. Combining the pruning and global optimization steps, the methodology is able to significantly reduce the size of the design space: for large problems, a 6-7 order of magnitude reduction is achievable in terms of the number of asteroid sequences that require low-thrust optimization. Furthermore, with the available computing resources, the methodology was run to completion for problems with up to 400 million asteroid sequences. For this largest problem – the modified GTOC2 problem – 809 asteroid

[*] Includes three new solutions not previously identified during the competition

sequences were optimized in low-thrust and the entire branch-and-bound algorithm required just over one month to complete. For the large problems examined in full – the modified GTOC2 and modified GTOC2 problems – the methodology was able to locate at least 70% of the best known solutions.

For the GTOC3 problem, the methodology was applied to a modified version that did not include gravity assists and did not take asteroid stay time into account in the objective function. However, the asteroid sequence that yielded the winning solution in the GTOC3 competition was identified, along with the asteroid sequences that yielded eight of the top ten competition solutions. Only 17 asteroid sequences had to be optimized in low-thrust. Applying this algorithm to the GTOC3 competition would have clearly left ample time to add gravity assists to the best sequences found, and to add the stay time consideration back into the objective function. This illustrates the ability of the methodology to be used as an initial screening technique for problems that require more complicated trajectories, such as gravity assists, and/or additional objective function terms.

For the modified GTOC2 problem, the methodology was able to locate four of the seven known solutions with objective function values greater than 80 kg/yr, including the best known solution. Additionally, three additional asteroid sequences greater than 80 kg/yr, which were not reported by the GTOC2 competitors were also found.

Due to resource limitations, the methodology was not run to completion for the full GTOC2 problem. However, in just two weeks of run time, three new solutions greater than 80 kg/yr were identified, the best of which would have placed 3rd in the original GTOC2 competition. If more time and/or computing resources were available, an incremental approach could be taken, as presented in Section 5.3. In order to locate the best known solution and eight of the top thirteen known solutions (all with J > 80 kg/yr), approximately 4,620 asteroid sequences would have to be optimized in low-thrust (assuming an initial lower bound of 100 kg/yr and that no solutions greater than 100 kg/yr are identified). The approach utilized in solving the GTOC2 problem illustrates the flexibility of the methodology to be tuned to available time and computing resources. Additionally, because the best solutions tend to be found early in the branch-and-bound algorithm, a set of good solutions can be identified in a short amount time, even for problems containing billions of possible asteroid sequences.

The methodology presented in this work is applicable to the conceptual design of low-thrust, combinatorial asteroid rendezvous missions, subject to the assumptions outlined in Section 3.2. First, the methodology does not take flyby trajectories into account. As was seen on the GTOC3 problem, however, the methodology can still be used as an initial screening technique to identify good sequences independent of flybys, with the plan to add these flybys subsequently. Second, in both the pruning phase and in the branch-and-bound algorithm, the methodology assumes that two-impulse optima are good surrogates for low-thrust optima. This generally occurs when the ratio of thrust time to time of flight for the low-thrust trajectories is small, as was the case of the problems examined in this work. If this were not true, the general framework presented could still be applied; however, an alternate metric would have to be identified that better approximates the low-thrust solutions. Finally, the branch-and-bound algorithm was shown to be feasible under the assumption that the objective function monotonically increases or decreases for each subsequent section in each branch of the tree. If this were not the case, the objective function could either be simplified, as was done for the GTOC3 problem, or the relaxed solution for each entire branch would be solved for in order to determine if that branch is pruned or optimized in low-thrust. Therefore, the framework presented provides some flexibility for problem modification on a case-by-case basis, if the problem of interest varies from a low-thrust, multiple asteroid rendezvous mission design problem.

The methodology and overall framework developed provides an organized search technique for the low-thrust mission design of asteroid tour missions. The intended application is for the conceptual design phase, where the ability to quickly explore the full extent of the design space is imperative to locating a broad suite of good solutions.

6.3 Recommendations for Future Work

Opportunities exist for future work to apply, modify, and improve upon the methodology presented. The following is a list of potential areas for future work:

1) ***Explore methods for incorporating phasing into the pruning phase of the methodology.*** The three-level heuristic sequence developed for this problem removes asteroid sequences from the design space but does not eliminate any of the time domain for particular asteroids, asteroid pairs, or asteroid sequences. None of the phasing metrics examined proved to be reliable predictors of low-thrust final mass. Additional work could delve deeper into this problem, in an attempt to incorporate phasing into the pruning aspect of the methodology.

2) ***Improve upon the evolutionary algorithm (inner loop optimizer).*** The evolutionary algorithm chosen for the inner loop optimizer is a simple genetic algorithm, using basic methods for reproduction, crossover, and mutation. Additionally, it archives all of the solutions found, to reduce the number of calls to MALTO and to keep track of the best solution found, in case it does not continue to subsequent generations. However, there are countless opportunities for improving the evolutionary algorithm used in the inner loop optimizer. One example would be to include the inheritance scheme implemented by Vavrina and Howell in Ref. 61, where MALTO would be free to locally optimize the global variables passed to it by the genetic algorithm. The goal of improving the evolutionary algorithm would be to reduce the number of function calls required

(calls to MALTO) and to increase the solution success rate of a single run of the genetic algorithm.

3) ***Improve the distributed computing framework to increase the speed of genetic algorithm runs.*** Genetic algorithms are excellent candidates for distributed computing, since each generation requires a number of parallel function calls – in this case, to MALTO or the low-thrust optimization algorithm of choice. For this work, due to software limitations, the function calls to MALTO were manually distributed to each node of the computer cluster (e.g., if fifty function calls are required and there are ten nodes, five function calls are sent to each node). A large improvement in computing time could be realized if the distributed computing were automated.

4) ***Method for determining impulsive multiplier in branch-and-bound.*** Currently, the branch-and-bound method requires an impulsive multiplier to be placed on the optimal impulsive solutions that serve as surrogates for relaxed solutions. This value is determined in an iterative fashion, beginning with no impulsive multiplier, and increasing its value each iteration as required to bound all of the known low-thrust solutions. A possible area for future work would be to explore methods for determining the value of this multiplier in a more rigorous manner, in order to avoid the iterative process and to help ensure that its final value is large enough to bound all of the low-thrust solutions.

5) ***Explore the possibility of applying traveling salesman solution techniques to asteroid tour problem.*** Different versions of the traveling salesman model include many of the aspects of the asteroid tour problem, as was presented in Section 1.2.3. While solutions techniques have been developed for each of these

types of the TSP, they have not been incorporated into a single problem which represents so many different variants. For example, the dynamic aspect of the trajectory problem is represented by the time-dependent TSP, the fact that not all asteroids have to be visited is represented by the generalized TSP, and if a return to the point of origin is not required, that is represented by the wandering TSP. Each of these variants has its own solutions technique, but future work could attempt to incorporate these different approaches into a single solution technique that could be applied to the asteroid tour problem.

6) ***Better initial guesses for MALTO.*** One of the most difficult challenges in this work was determining appropriate initial guesses for MALTO, which could be automated within the framework of the global optimization framework, and that would reliably produce the optimum solution for a range of asteroid sequences and values of the time variables. While a number of techniques were examined, three different initial guesses were chosen, which results in MALTO being run three times for each trajectory that must be optimized. Future work could attempt to determine a more rigorous method for determining an appropriate initial guess, while reducing the amount of computation time required.

7) ***Apply methodology/framework to additional problems.*** The problems examined in this work were chosen because they all had known solutions against which the results generated by the methodology could be compared. There are additional problems of interest, however, to which the methodology could be applied.

   a. In 2003, the National Research Council completed a decadal survey on solar system exploration.[76,77] Its task was to develop a science strategy for solar system exploration for the upcoming decade, by determining the most important scientific questions currently facing planetary science. Furthermore, the study was tasked with creating a prioritized list of mission options that could best seek to answer those questions, one of which is an asteroid rover/sample return mission. In its Cosmic Vision for space science, the European Space Agency (ESA) also identified a near-Earth object sample return mission as one of its priorities in the 2015-2025 timeframe.[78] Although in both instances, the sample return mission called for would only visit a single asteroid, it is pointed out in the Cosmic Vision that a full understanding of the populations, histories, and relationships of asteroids and meteors would eventually require sample return missions to asteroids in each of the spectral classes. Therefore, a multiple asteroid sample return mission would eventually be of interest to both NASA and ESA.

The methodology could be applied to the conceptual design of such a mission, in order to determine its feasibility in the near-term from a mission design standpoint. The goal of the mission design would be to rendezvous with two near-Earth asteroids and then return to Earth, while maximizing final mass at Earth return, maximizing the stay time at each asteroid, minimizing overall time of flight, and minimizing the arrival velocity at Earth (the objective function could consider one or more of these objectives and implement the remaining objectives as constraints). The mission would implement low-thrust propulsion, the specifics of which would be based on currently available technologies in terms of thrust and specific impulse. The asteroids could consist of a single group of all known near-Earth asteroids (currently totaling 6,496)[79] or multiple groups of asteroids based on their scientific interest and/or value. The methodology would be applied in the same way as it was applied to the GTOC3 (single group of asteroids) or GTOC2 (multiple groups of asteroids) problem. Because only two asteroids are visited and Earth return is required, the first pruning metric would not be employed (increasing semi-major axis).

b. In 2009, the Augustine Commission completed its report on the future of NASA human spaceflight.[80] Several options for initial exploration beyond low-Earth orbit were described, one of which consists of visiting a series of locations and objects in the inner solar system. In this Flexible Path architecture, the time duration and complexity of the missions slowly builds, beginning with human missions to the Lagrange points, followed by missions to near-Earth objects, and finally human missions to Mars. It calls for visiting several near-Earth objects in order to return samples, practice operations near a small body, and potentially practice in-situ resource utilization. While the plans call for only a single asteroid rendezvous, an interesting offshoot would be to look at a multiple-asteroid rendezvous human mission, in order to increase the mission duration before the first human mission to Mars.

   Two possible related mission design problems could be of interest in this case. First would be the trajectory design of the human mission to the asteroids. This would likely employ chemical (high-thrust) propulsion, but the methodology could still be applied to determine a set of good solutions. In this case, the objective would be to minimize ΔV and time of flight (or simply constraint time of flight to some upper bound) and to minimize Earth arrival velocity. The pruning phase of the methodology could be applied as-is, but the branch-and-bound would be altered in that there would be no low-thrust optimization at the end. The “relaxed” solutions would instead be the actual impulsive solutions. Alternatively, the relaxed solutions could still consist of two-impulse solutions and instead of low-thrust, multiple-impulse trajectories could be considered.

Second, if in-situ resource utilization is to be tested at the asteroids, it may be useful to pre-deploy those assets prior to the human missions. Because time of flight is not nearly as constrained for cargo delivery missions, low-thrust propulsion could be employed in this case. For this problem, the goal would be to visit two asteroids, while maximizing final mass (no Earth return required). The results of the low-thrust cargo delivery missions and the high-thrust human missions could be combined in order to locate pairs of asteroids that would be easily accessible to both mission types.

APPENDIX A

SET OF GTOC2 ASTEROIDS

Table 33 lists all of the asteroids in the full GTOC2 problem, including their SPK-ID number, common name, orbital elements (semi-major axis, eccentricity, inclination, longitude of the ascending node, argument of periapsis, mean anomaly, and epoch), and group number. Additionally, for each asteroid, the table indicates if that asteroid was part of each of the problems examined: the small sample problem (s), the intermediate sample problem (i), and the modified GTOC2 problem (m). The asteroids are sorted by their group and then by their SPK-ID number.

Table 33: GTOC2 Asteroids.

SPK-ID	Name	a (AU)	e	i (deg)	LAN (deg)	arg. periapsis (deg)	M (deg)	Epoch (MJD)	Group	Prob
2002062	Aten	0.9667013	0.18271178	18.932519	108.62768	147.94205	334.92171	54000	4	s,i,m
2002100	Ra-Shalom	0.83206393	0.4364584	15.757434	170.87687	355.99521	158.8253	54000	4	s,i,m
2002340	Hathor	0.84389537	0.44995917	5.8539782	211.51958	39.938047	303.69494	54000	4	
2003362	Khufu	0.98946745	0.46856161	9.9183703	152.50975	54.982849	230.97821	54000	4	m
2003554	Amun	0.97371428	0.28048062	23.361458	358.67621	359.38617	297.0219	54000	4	
2003753	Cruithne	0.99774049	0.51478951	19.8093	126.29699	43.743147	261.34727	54000	4	s,i
2005381	Sekhmet	0.94744835	0.29610804	48.973045	58.562459	37.413535	345.61516	54000	4	s,i,m
2005590	"1990 VA"	0.98568232	0.27958167	14.186004	216.34471	34.416179	218.05496	54000	4	
2005604	"1992 FE"	0.92728217	0.40535123	4.796818	312.00418	82.422924	192.01877	54000	4	
2033342	"1998 WT24"	0.71851637	0.41798842	7.3416842	82.004826	167.28181	149.53901	54000	4	
2065679	"1989 UQ"	0.91486343	0.26464041	1.2914011	178.35188	14.927526	324.04271	54000	4	
2066063	"1998 RO1"	0.99093072	0.72009385	22.666222	351.92132	151.06633	312.1291	54000	4	m
2066146	"1998 TU3"	0.78727129	0.48380682	5.4099722	102.31935	84.562367	277.80779	54000	4	
2066391	"1999 KW4"	0.64231117	0.68842863	38.891521	244.9322	192.59839	260.28645	54000	4	m
2066400	"1999 LT7"	0.85520974	0.57256097	9.0653163	79.918316	341.27615	303.78632	54000	4	
2068347	"2001 KB67"	0.96282557	0.37978055	17.139554	245.97862	243.8006	304.03906	54000	4	
2085770	"1998 UP1"	0.99873332	0.3449384	33.178463	18.403672	234.37669	74.743373	54000	4	m
2085953	"1999 FK21"	0.73876145	0.70312537	12.598508	180.54824	172.32944	193.41695	54000	4	
2085989	"1999 JD6"	0.88271351	0.63293609	17.047146	130.29026	309.14389	57.092802	54000	4	
2086450	"2000 CK33"	0.9680575	0.41479377	18.106551	124.91312	215.55604	86.689225	54000	4	
2086667	"2000 FO10"	0.85926966	0.59473781	14.284786	208.40354	172.40027	208.35792	54000	4	m
2087309	"2000 QP"	0.84745188	0.46306937	34.745734	294.31189	188.12379	171.16429	54000	4	m
2087684	"2000 SY2"	0.85873344	0.64269364	19.234294	162.11324	47.709401	287.41396	54000	4	m
2088213	"2001 AF2"	0.9539439	0.59522617	17.814259	114.31678	194.94151	172.78666	54000	4	
2096590	"1998 XB"	0.90795755	0.35118792	13.597697	75.804099	202.65036	198.06344	54000	4	
2099907	"1989 VA"	0.72853347	0.59476	28.792699	225.62898	2.8083713	218.4728	54000	4	m

2099942	Apophis	0.92226308	0.1910585	3.3313256	204.46	126.39552	84.786507	54000	4	s,i
2105140	"2000 NL10"	0.9142539	0.81702607	32.514695	237.49863	281.51518	154.92805	54000	4	
3005821	"1992 BF"	0.90800988	0.27176599	7.2538905	315.473	336.42244	186.28381	54000	4	
3005964	"1994 TF2"	0.99299794	0.28400259	23.752266	175.27593	349.68503	252.92589	54000	4	
3005969	"1994 WR12"	0.75686007	0.39730327	6.8638137	63.069077	205.68112	168.83864	49709	4	
3005970	"1994 XL1"	0.67082068	0.5263641	28.163343	252.69674	356.52092	0.48117591	54000	4	m
3005972	"1991 VE"	0.89088169	0.66458077	7.2190179	62.010882	193.51103	336.36256	54000	4	
3005973	"1995 CR"	0.90672544	0.86843062	4.0352333	342.77703	322.39388	99.988219	54000	4	
3007848	"1997 UH9"	0.83004916	0.47478051	25.492374	42.445929	180.85609	86.169308	54000	4	m
3009717	"1997 AC11"	0.9132051	0.36807412	31.677089	116.97656	141.55437	290.00574	54000	4	m
3010201	"1997 MW1"	0.93754452	0.34638355	12.773843	260.05979	203.7136	229.81762	54000	4	
3010207	"1997 NC1"	0.8655388	0.20828137	16.718068	96.571071	16.625509	314.20271	54000	4	
3011815	"1998 DG16"	0.89669302	0.3581831	16.208789	344.42506	356.81644	222.69406	54000	4	
3012397	"1993 VD"	0.87624175	0.5514582	2.0627017	2.7408046	253.64179	352.82705	54000	4	
3013030	"1998 HE3"	0.87849063	0.44056444	3.3998126	53.890616	309.00862	327.02451	54000	4	
3013071	"1998 HD14"	0.96311766	0.31262107	7.8072693	183.97257	260.72001	53.391542	54000	4	
3014113	"1998 ST27"	0.81932815	0.53002281	21.049912	197.61161	322.42211	146.69697	54000	4	
3014114	"1998 SZ27"	0.9032239	0.50379849	23.425461	166.83655	47.499929	92.326803	51081	4	m
3014184	"1998 SD9"	0.70287493	0.5042051	2.9029512	167.15519	6.2466136	188.83079	51077	4	
3015691	"1998 VF32"	0.85113036	0.44357995	23.750628	236.336	320.85172	257.14348	54000	4	m
3016523	"1998 XX2"	0.74123992	0.36744228	6.9691489	74.57572	152.86115	301.0636	54000	4	
3017039	"1998 VR"	0.87579594	0.3181469	21.802114	46.44937	170.67926	55.367825	54000	4	m
3017060	"1999 AQ10"	0.93722796	0.23454007	6.5601162	327.40619	299.48051	33.408243	54000	4	
3017309	"1999 AO10"	0.91140605	0.11077323	2.6226597	313.34135	7.6536829	82.979128	54000	4	m
3019650	"1999 MN"	0.67402486	0.66520869	2.0164845	80.792574	9.8818082	236.01441	54000	4	m
3020946	"1999 HF1"	0.81902244	0.46252854	25.657332	155.93408	253.33986	114.23065	54000	4	
3021790	"1998 SD15"	0.93250655	0.34499429	26.793225	183.98793	35.805081	63.591671	54000	4	m
3024030	"1999 VX25"	0.89999766	0.13956765	1.6633228	55.303545	151.72637	221.34283	54000	4	
3025763	"2000 AC6"	0.85357291	0.28634449	4.6958454	101.79881	187.91717	3.8151398	54000	4	
3025764	"2000 AF6"	0.87831922	0.41135064	2.6926995	110.86572	200.06223	197.96191	54000	4	
3025765	"2000 AZ93"	0.74678009	0.36001309	8.6005618	277.58834	7.9287065	310.47211	54000	4	m
3027730	"1998 XE12"	0.87828555	0.73912509	13.432252	280.10884	353.0528	273.61263	54000	4	
3028808	"2000 CH59"	0.86324958	0.42308796	3.2721008	214.29152	109.00818	253.08418	54000	4	
3029428	"1999 YK5"	0.82941678	0.55831156	16.741231	349.66292	292.73797	117.89073	54000	4	m
3031020	"2000 BD19"	0.876503	0.89503754	25.676583	333.80753	324.24248	205.0929	54000	4	m
3031176	"2000 EB14"	0.89556289	0.49535956	11.560793	162.90292	139.56794	175.79773	54000	4	
3031177	"2000 ED14"	0.83508053	0.56664333	13.775537	3.9767428	310.03182	116.79868	54000	4	
3031178	"2000 EE14"	0.66184819	0.53291455	26.470152	155.80371	197.79925	234.65129	54000	4	
3031183	"2000 ET70"	0.94696812	0.12351254	22.321499	331.20642	46.371289	169.54532	54000	4	m
3031186	"2000 EW70"	0.93766812	0.32108762	5.4192095	178.31731	125.30451	330.94723	54000	4	
3031703	"2000 BM19"	0.74122654	0.35849219	6.8900577	70.682966	247.33611	283.11028	54000	4	
3035165	"2000 EZ106"	0.92871575	0.44688014	40.260068	358.52887	313.90075	275.64076	51624	4	m
3035166	"2000 EA107"	0.92969018	0.45572011	28.578789	52.953803	277.98821	281.16831	54000	4	
3036363	"2000 HB24"	0.81552756	0.43044014	2.6620264	55.249024	17.867756	355.02601	54000	4	m
3039898	"2000 GD2"	0.75784422	0.47657125	32.143436	358.1575	16.943112	78.616265	54000	4	m
3042555	"2000 LG6"	0.91610235	0.11212565	2.8299469	72.769203	7.7459151	237.46661	54000	4	
3046648	"2000 OK8"	0.98475099	0.22113264	9.9847342	304.64098	166.10892	312.02601	54000	4	
3050515	"2000 SP43"	0.81138241	0.46688594	10.355937	350.69072	224.30388	170.08819	54000	4	
3053717	"2000 PJ5"	0.87266158	0.37357638	51.180869	124.4379	7.6049849	322.8322	54000	4	m
3054338	"2000 SZ162"	0.92940488	0.16742376	0.89619274	14.741749	131.29453	116.32414	54000	4	

3054373	"2000 UK11"	0.88325596	0.24874524	0.7818067	237.8882	293.04609	279.96198	54000	4	i,m
3057545	"2000 WO107"	0.91136333	0.78064631	7.783026	69.374003	213.58275	302.66343	54000	4	
3061547	"2000 RN77"	0.95118128	0.31832272	16.093766	312.84998	211.71057	12.786194	54000	4	
3062815	"2000 UH11"	0.87028331	0.4223829	32.228534	29.823765	187.37091	271.42008	54000	4	m
3063058	"2000 WC1"	0.87949364	0.26271471	17.412879	50.862594	229.98231	141.9183	54000	4	
3063789	"2000 UR16"	0.90366249	0.43874889	11.744223	33.888091	228.71238	38.109895	54000	4	
3063823	"2000 WP19"	0.85448153	0.28864592	7.6757121	55.931647	221.95088	252.14788	54000	4	
3064315	"1998 XN17"	0.98185794	0.2096551	7.2453491	85.995719	226.33812	105.51781	54000	4	m
3067492	"2000 YS134"	0.85678956	0.22462344	3.4909518	97.371774	189.32344	247.32303	54000	4	m
3067616	"2001 BE10"	0.8233799	0.3689377	17.513089	297.87042	30.585412	332.21202	54000	4	
3068066	"2001 BA16"	0.9402947	0.13738105	5.7684736	115.62224	242.83832	185.9995	54000	4	
3069758	"2001 CP36"	0.71449449	0.4073083	10.551364	331.00521	353.49934	284.26514	54000	4	
3070801	"2001 CQ36"	0.93967934	0.17625793	1.2921429	31.947763	342.489	166.51755	54000	4	m
3071939	"2001 BB16"	0.8542365	0.17238766	2.0261914	122.57094	195.57522	217.07665	54000	4	
3072196	"2001 ED18"	0.98914235	0.057313022	11.62947	357.82492	306.54524	94.254912	54000	4	
3072273	"2001 FR85"	0.98269502	0.02797089	5.2439975	183.09711	233.5423	352.59284	54000	4	
3072291	"2001 FO127"	0.88860879	0.15914197	7.2897211	189.24157	200.81708	356.38095	54000	4	
3072413	"2001 CK32"	0.72538055	0.38255603	8.1364752	109.55428	234.13343	173.96749	54000	4	
3074756	"2001 HY7"	0.91403391	0.41209793	5.209049	205.39319	210.98323	203.67515	54000	4	
3076722	"2001 FZ57"	0.94421086	0.60426198	20.663998	22.148342	339.94382	205.64386	54000	4	
3076775	"2001 HC"	0.87462622	0.49934979	23.74621	32.651464	28.155709	337.96255	54000	4	
3079950	"2001 OT"	0.93386971	0.32328704	12.090374	295.94754	142.88185	155.09714	54000	4	
3081066	"1998 SV4"	0.81647942	0.64201038	53.296773	177.26469	359.48387	110.45555	54000	4	m
3089251	"2001 QP153"	0.89155077	0.21371778	50.207445	317.70623	244.31983	122.67802	54000	4	m
3092114	"1993 DA"	0.93557259	0.09336139	12.377796	329.18587	354.30512	193.60748	54000	4	
3092124	"1994 GL"	0.68441915	0.50211708	3.6341904	197.22215	179.0533	179.7846	49450	4	
3092144	"1996 BG1"	0.89755766	0.28065043	3.8139909	139.95032	150.27456	206.58537	50107	4	
3092156	"1996 XZ12"	0.97997322	0.49926135	5.6585162	251.74134	55.816361	72.906521	50427	4	m
3092192	"1998 SO"	0.73134122	0.69855898	30.350464	176.17726	359.82079	118.70808	54000	4	
3092226	"1999 LK1"	0.90718857	0.33269924	11.907748	240.07869	223.55372	286.28021	54000	4	
3092245	"1999 VW25"	0.92884396	0.1120346	10.776168	232.31222	354.5409	62.090196	54000	4	m
3092253	"2000 EM26"	0.81607747	0.46980368	3.8734528	345.26231	23.964009	77.21737	54000	4	
3092260	"2000 HO40"	0.74399955	0.52412868	5.9877829	30.318292	6.0866893	173.17855	51662	4	m
3092272	"2000 RH60"	0.8258742	0.55130757	19.643821	177.92578	354.37058	189.62089	54000	4	m
3092324	"2001 RU17"	0.95860272	0.240769	13.877065	170.77281	330.28055	356.41634	54000	4	
3092325	"2001 RV17"	0.91409916	0.34251754	7.5207799	154.15697	4.3166007	117.71373	54000	4	
3092339	"2001 SQ263"	0.94807271	0.49150795	3.951064	327.3059	262.35483	225.1088	54000	4	
3092347	"2001 TD"	0.95414371	0.16607503	9.011133	13.218041	241.34418	217.30782	54000	4	
3092357	"2001 RY47"	0.90650678	0.39299432	17.604042	11.3038	213.96837	38.677025	54000	4	
3092370	"2001 TW1"	0.91129913	0.5259771	31.336012	27.399044	208.76826	338.40154	54000	4	
3092377	"2001 TD2"	0.9619541	0.48147436	19.038812	12.90751	199.00974	201.23957	54000	4	
3092380	"2001 TX44"	0.87475176	0.54598912	15.203483	57.856362	135.96125	231.29666	54000	4	
3092386	"2001 TD45"	0.79674105	0.77741935	25.419665	30.332708	212.38759	52.895742	54000	4	m
3092390	"2001 UP"	0.88493377	0.28666987	7.7052214	25.592661	133.00062	225.31406	54000	4	
3102665	"2001 WF49"	0.75103189	0.3734296	18.179766	239.71613	358.3795	188.25934	52242	4	
3102680	"2001 XU1"	0.79739141	0.54626241	27.159807	69.743317	208.48745	19.485949	54000	4	
3102687	"2001 XY10"	0.87176578	0.38714096	30.995333	92.982658	219.67403	59.371016	54000	4	m
3102718	"2001 YE4"	0.67636878	0.54206925	4.8003637	306.08255	318.32119	221.27778	52271	4	m
3102727	"2002 AX1"	0.8799063	0.54157069	33.006701	294.73391	25.324129	351.02314	54000	4	m
3102728	"2002 AY1"	0.77873106	0.43775061	29.88594	287.90059	323.84803	192.59727	54000	4	m

3102731	"2002 AB2"	0.84083923	0.38595466	13.181771	103.14846	149.5432	282.4749	54000	4	
3102744	"2002 AU4"	0.85560809	0.3736	17.179764	99.516019	205.13541	153.88991	54000	4	
3102756	"2002 AO11"	0.91555629	0.16340646	13.072181	295.18464	306.60939	16.12534	54000	4	
3102762	"2002 AA29"	0.99426257	0.013065778	10.742631	106.46935	100.60964	164.92244	54000	4	s,i
3102779	"2002 BN"	0.87507581	0.54652018	27.746329	115.8071	147.21096	156.49332	54000	4	m
3102787	"2002 CD"	0.97987696	0.1766903	6.879208	8.7510894	331.56631	51.187861	54000	4	
3114017	"2002 CQ11"	0.97888269	0.42840825	2.4597657	81.442256	272.77239	357.32802	54000	4	
3114023	"2002 CW11"	0.86557884	0.22561468	3.1333566	137.63141	210.35146	42.307944	54000	4	
3114026	"2002 CC14"	0.8198196	0.40303203	12.607732	137.70231	217.99794	186.56678	54000	4	
3114104	"2002 DB4"	0.8577063	0.36944345	16.599775	234.34539	94.099107	12.146551	54000	4	
3117427	"2002 EM7"	0.92124305	0.36305932	1.5475656	347.22658	57.667418	131.23247	54000	4	
3117446	"2002 FW1"	0.82348932	0.341939	6.5978745	164.19778	223.15648	134.38613	54000	4	
3117447	"2002 FB3"	0.76147828	0.60181997	20.270204	203.66158	148.24457	40.981494	54000	4	
3117460	"2002 FT5"	0.96703756	0.30044155	28.064829	7.9945393	31.452424	26.207003	54000	4	
3117468	"2002 FT6"	0.98825788	0.46273267	9.4897329	188.61692	226.69016	281.87596	54000	4	m
3120861	"2002 GB"	0.99226321	0.52902156	22.554197	40.865148	8.3075997	259.155	54000	4	
3120863	"2002 GQ"	0.76830954	0.37557174	10.628605	189.48132	206.28661	4.8467488	54000	4	
3120884	"2002 EZ16"	0.92160627	0.56640928	30.14657	262.94414	25.307036	177.07735	54000	4	m
3124996	"2002 JX8"	0.7701479	0.30533751	4.3155863	68.645093	338.18528	337.37454	54000	4	
3125004	"2002 JC"	0.81884734	0.39093655	40.85275	69.427745	306.91181	146.81126	54000	4	m
3125009	"2002 JW15"	0.89868318	0.26626348	11.765305	218.53494	175.17419	259.49608	54000	4	
3126183	"2002 JR100"	0.92470305	0.29777782	3.7633964	203.56944	253.41942	71.988181	54000	4	
3127391	"2002 LY1"	0.95500888	0.37938048	2.9089622	248.23483	133.95352	127.29616	54000	4	
3127401	"2002 LT24"	0.71992412	0.49553466	0.76027896	166.75667	282.02629	166.8299	54000	4	
3127406	"2002 LT38"	0.84475671	0.31396358	6.1999594	259.60686	162.73931	25.047051	54000	4	
3130459	"2002 MQ3"	0.9135579	0.27422333	36.28443	109.10504	346.86651	103.49846	54000	4	
3131055	"2002 NN4"	0.87651922	0.43432919	5.4175723	259.65972	222.15883	198.16622	54000	4	
3132092	"2002 OA22"	0.93591413	0.24289976	6.9056071	174.41506	318.27888	347.97534	54000	4	
3133156	"2002 QY6"	0.81697506	0.69905755	12.748378	164.33324	355.26495	358.67083	54000	4	
3134264	"2002 RR25"	0.96703422	0.30996246	13.538525	349.95955	156.04858	305.86572	54000	4	
3134268	"2002 RW25"	0.82550846	0.28643033	1.3250455	92.022439	71.690414	355.44719	54000	4	
3136734	"2002 SP"	0.90470877	0.60068666	20.869143	350.95397	169.36365	127.32394	54000	4	m
3137844	"2002 TZ66"	0.93000656	0.12057551	8.4784677	13.140614	223.07557	275.96035	54000	4	
3141527	"2002 UA31"	0.79876331	0.48723132	30.700644	209.35617	358.62917	331.99547	54000	4	m
3141535	"2002 VV17"	0.83739563	0.43662198	9.6948341	222.32902	348.73063	247.83159	54000	4	m
3141538	"2002 VE68"	0.7236569	0.41051242	8.9801315	231.66157	355.51291	280.154	54000	4	
3143084	"2002 VX91"	0.98456037	0.20136283	2.331352	216.87263	78.254887	72.980025	54000	4	
3143121	"2002 XB"	0.90584107	0.23739246	25.53355	245.93098	351.2243	346.24144	54000	4	m
3144153	"2002 XP37"	0.95516322	0.35943369	21.528325	265.95557	317.58094	263.35691	54000	4	
3144155	"2002 XY38"	0.90911752	0.21719253	2.0854213	160.69647	118.5063	274.21168	54000	4	
3144531	"2002 XS90"	0.80953528	0.24208909	34.160061	81.700317	178.61437	246.44399	54000	4	m
3145517	"2003 AK18"	0.87611211	0.38410512	7.3889096	301.85306	23.557751	298.6393	54000	4	
3146499	"2003 AF23"	0.87482965	0.42612924	23.236807	286.83005	43.944275	283.29971	54000	4	m
3147579	"2003 CA4"	0.9203212	0.11970839	7.4793473	139.95971	172.95921	223.42272	54000	4	
3150768	"2003 EM1"	0.95752824	0.051791757	15.269684	346.05269	23.978949	75.424192	54000	4	
3150774	"2003 EO16"	0.93470009	0.24963774	13.220376	178.04024	167.88093	155.74913	54000	4	
3151641	"2003 FK1"	0.70741718	0.48594178	23.375604	177.59264	196.51278	158.54796	52724	4	m
3151644	"2003 FU3"	0.8585099	0.39390777	13.050528	21.684072	339.2499	342.06502	54000	4	
3151655	"2003 FY6"	0.73081812	0.58168944	6.6462784	359.37936	29.591842	322.20328	54000	4	
3152309	"2003 GS"	0.89306537	0.21856995	12.031238	196.34869	181.81855	202.52777	54000	4	m

3152317	"2003 GQ22"	0.87232869	0.18199623	17.024551	199.60692	168.68648	275.52077	54000	4	
3153508	"2003 HB"	0.84990274	0.380551	18.108305	70.456253	306.68814	322.20206	54000	4	
3153509	"2003 HM"	0.81317508	0.27040152	26.275789	30.512696	19.241924	24.021864	54000	4	
3153530	"2003 HT42"	0.81508903	0.26213285	4.8810454	39.08647	351.8594	56.376705	54000	4	
3154503	"2003 KO2"	0.72744426	0.5109053	23.505997	215.53035	203.9569	308.37372	54000	4	
3154513	"2003 KZ18"	0.94882172	0.33060063	23.901897	250.048	154.67071	73.518182	54000	4	m
3154520	"2003 LH"	0.96054013	0.14975606	10.795804	247.33008	238.12675	294.20537	54000	4	
3156302	"2003 LN6"	0.85723098	0.21046138	0.63241404	215.81727	210.40318	248.98378	54000	4	
3157339	"2003 NZ6"	0.79330244	0.49250313	18.233456	124.69036	311.55026	87.736143	54000	4	
3160723	"2003 RU11"	0.88859382	0.18325773	4.6515135	178.77306	316.20899	88.522675	54000	4	
3160748	"2003 SW130"	0.88518207	0.30354972	3.6597329	176.57871	47.968282	321.24813	54000	4	
3160799	"2003 SD220"	0.82842354	0.2097393	8.4615437	274.20954	326.31267	130.70576	54000	4	
3160853	"2003 TG2"	0.90779594	0.31592396	25.432456	200.72375	355.12527	314.04643	54000	4	
3163736	"2003 TL4"	0.77653201	0.38182999	12.146048	220.16201	321.8564	329.76908	54000	4	
3164401	"2003 UY12"	0.70092845	0.59594346	16.508885	22.958041	200.59164	116.86678	54000	4	
3164404	"2003 UC20"	0.78130319	0.3368181	3.794778	188.87281	59.273754	206.13498	54000	4	
3164431	"2003 UT55"	0.97915383	0.14704531	17.058952	212.81145	287.43237	269.39447	52939	4	
3167348	"2003 WU21"	0.90863991	0.5445115	28.539008	57.597699	140.64874	280.45945	52966	4	m
3167353	"2003 WP25"	0.99058994	0.12109868	2.5230027	42.306512	225.08689	97.336963	54000	4	
3167367	"2003 WT153"	0.88991985	0.18053258	0.3560634	65.466287	138.0166	5.1901772	54000	4	
3170202	"2003 YJ"	0.93049583	0.19888123	19.460508	89.811876	165.60388	236.09578	54000	4	
3170203	"2003 YX1"	0.87876494	0.26655962	5.7547748	89.91715	222.81766	232.34244	54000	4	
3170204	"2003 YS17"	0.93033425	0.31303949	6.5240763	99.18043	134.48003	274.13873	54000	4	
3170208	"2003 YG136"	0.96902587	0.35503538	2.735198	86.544614	127.98154	235.18049	54000	4	
3170221	"2003 YN107"	0.99414961	0.013466245	4.3015637	265.1633	83.895555	14.977607	54000	4	m
3170242	"2003 YR1"	0.89864059	0.45058732	29.268846	86.451474	138.05774	357.61805	54000	4	m
3172322	"2004 BY1"	0.88403687	0.22188425	3.6171905	299.09124	28.23779	217.61998	54000	4	
3174187	"2004 BT58"	0.96090346	0.38494142	17.667309	300.91031	45.800414	38.593947	54000	4	
3175337	"2004 DH2"	0.94402813	0.40024487	23.023004	157.35712	216.07589	41.130168	54000	4	
3176187	"2004 DA53"	0.88455581	0.33025347	5.141462	336.708	50.031926	128.14063	54000	4	
3177176	"2004 EW"	0.98950948	0.27977013	4.6639542	343.45054	55.822201	302.36837	54000	4	
3177188	"2004 EU9"	0.88044837	0.50520426	28.587726	161.13348	202.3902	144.08901	54000	4	m
3177193	"2004 EL20"	0.81459703	0.26847244	7.5899015	356.20573	337.56064	8.6811993	54000	4	
3177197	"2004 ER21"	0.90031102	0.17108277	7.9562364	357.43335	343.24023	181.87279	54000	4	m
3177202	"2004 FH"	0.81796821	0.28900036	0.021373399	296.18068	31.319869	14.154055	54000	4	s,i,m
3177226	"2004 FM17"	0.88559948	0.24938109	6.7662088	170.07059	196.2384	197.01913	54000	4	
3177232	"2004 FG29"	0.87856871	0.49234168	3.5104099	183.39904	142.00897	280.59227	54000	4	
3177234	"2004 FJ29"	0.91350452	0.34920303	33.467907	195.47359	210.26718	55.132809	54000	4	m
3179349	"2004 GP"	0.69653246	0.48849431	14.547286	115.75337	278.56009	230.04016	54000	4	
3179363	"2004 HC"	0.78916702	0.59877773	28.975304	203.02793	159.32865	47.977419	54000	4	
3180192	"2004 HT59"	0.97997642	0.22337695	11.135007	214.71419	112.10254	87.29048	54000	4	m
3182186	"2004 JW20"	0.95274176	0.56156984	14.734804	235.26374	207.42161	290.45178	54000	4	
3182187	"2004 JX20"	0.90121009	0.26577903	10.52491	101.99185	348.94624	27.260244	54000	4	
3182823	"2004 KG1"	0.83006288	0.40744127	1.9081961	243.48958	213.17629	141.06958	54000	4	m
3182829	"2004 KH15"	0.96114304	0.17028425	35.070972	78.132599	350.0398	355.76643	54000	4	m
3182833	"2004 KH17"	0.71192116	0.49856275	22.055369	79.22656	340.67122	148.13048	54000	4	m
3183847	"2004 LO2"	0.91488208	0.3513503	25.453919	82.670132	309.52078	121.70682	54000	4	m
3184475	"2004 MD6"	0.95062062	0.56280186	29.334285	263.9296	231.36764	230.38682	54000	4	m
3249978	"2004 FU162"	0.82681962	0.39218262	4.1644361	191.24861	139.79199	262.65627	53100	4	
3249980	"2004 QB3"	0.95021121	0.4167979	14.339061	327.20744	232.1766	88.266623	53241	4	

3250193	"2004 QG13"	0.95199695	0.17441825	56.22508	151.86061	9.1122288	165.27237	53244	4	m
3250195	"2004 QD14"	0.94312385	0.33803687	6.2477966	75.46293	109.19721	210.80311	54000	4	
3250293	"2004 QA22"	0.95092007	0.12170564	0.57411596	175.15212	28.533305	202.67682	54000	4	m
3251510	"2004 RU10"	0.90403261	0.65680238	15.91456	119.11494	65.154363	317.11312	54000	4	
3251512	"2004 RX10"	0.92044085	0.35110701	5.9585478	173.8943	333.85506	323.2308	54000	4	
3252104	"2004 RO111"	0.96120257	0.32880847	5.3345585	199.45033	280.81436	315.18661	54000	4	
3253645	"2004 ST2"	0.95427638	0.1863123	22.053165	356.79811	226.8361	173.13211	54000	4	m
3254500	"2004 SD20"	0.87511021	0.46481928	21.334924	46.654234	94.378224	53.416337	54000	4	m
3255174	"2004 SW26"	0.73778019	0.41619723	18.418049	180.36366	359.49238	183.96342	53272	4	m
3255464	"2004 SB56"	0.86591757	0.23763089	18.701611	302.13149	233.48043	3.3642175	54000	4	
3255465	"2004 SC56"	0.7670961	0.42892764	4.7613942	202.37822	322.48787	207.26302	54000	4	
3255879	"2004 TA1"	0.90794321	0.24989018	13.512538	14.576143	200.48313	239.92139	54000	4	
3256321	"2004 TD10"	0.75098496	0.44284208	2.6217937	48.487884	136.67192	208.67795	54000	4	
3256580	"2004 TR12"	0.89500976	0.2095971	19.353928	155.44787	77.702026	230.72107	54000	4	
3256583	"2004 TP13"	0.97640702	0.16192739	36.517136	12.606993	138.64939	238.04986	53289	4	m
3257077	"2004 TN20"	0.94624489	0.25852582	14.059494	203.10884	56.491094	130.79876	54000	4	m
3258062	"2004 UH1"	0.95420397	0.39678012	3.7129817	29.862271	120.62117	303.80802	54000	4	m
3258076	"2004 UT1"	0.9644237	0.22112866	4.5080307	211.98291	294.21659	275.85009	54000	4	m
3261401	"2004 VZ"	0.94051139	0.2439765	16.218207	225.83194	297.07694	286.7039	54000	4	
3261681	"2004 VJ1"	0.94377277	0.16445192	1.2936976	233.53862	332.35792	220.29651	54000	4	i
3262331	"2004 VG64"	0.96831267	0.6554068	36.276971	208.92346	43.876742	42.361252	54000	4	m
3262569	"2004 WC1"	0.85560093	0.16979483	10.348597	54.94469	179.78693	299.55207	54000	4	
3263232	"2004 XG"	0.83752123	0.2980902	1.2031678	285.35506	0.81535776	244.86104	54000	4	
3263233	"2004 XJ"	0.88711985	0.1708655	12.187234	253.39186	15.930031	209.35138	54000	4	
3263448	"2004 XK14"	0.74868835	0.43382753	3.0995667	307.15697	302.8351	107.2211	54000	4	
3263449	"2004 XL14"	0.76001344	0.40978133	21.488824	85.680513	157.51318	99.054976	54000	4	
3263451	"2004 XN14"	0.93149302	0.26650978	10.738706	120.95563	115.66393	228.79036	54000	4	m
3263793	"2004 XY60"	0.64024467	0.7968061	23.746543	122.70088	130.7697	23.609494	54000	4	m
3264188	"2004 YC"	0.86840745	0.31327764	6.0667175	263.47717	47.285452	167.70475	54000	4	
3264189	"2004 YD"	0.84262456	0.24011097	12.086974	265.61112	7.8051078	264.56936	54000	4	
3264547	"2004 YA5"	0.80883312	0.5361478	28.149668	269.94609	31.209738	243.68826	54000	4	m
3265905	"2005 AY28"	0.872222	0.56873666	5.8691155	117.66371	155.76728	265.87081	54000	4	
3265909	"2005 BE"	0.88383727	0.42113866	31.187501	116.00428	168.67007	191.20433	54000	4	m
3266031	"2005 BU"	0.84659761	0.30073285	12.969629	296.89813	38.72732	170.98457	54000	4	m
3266035	"2005 BO1"	0.94878176	0.35599452	10.674391	113.41088	174.10611	131.31082	54000	4	
3267564	"2005 CN61"	0.99121273	0.068706638	9.5265808	147.02475	248.61695	326.2324	54000	4	
3273458	"2005 EP1"	0.89266375	0.77031035	16.320488	344.39249	328.04398	297.39556	53433	4	
3273782	"2005 ES70"	0.76389989	0.38338784	20.484115	353.00599	351.28138	192.40885	53441	4	m
3273788	"2005 EK70"	0.95945314	0.13537803	30.002363	329.84879	347.06403	8.5373459	54000	4	m
3274691	"2005 FC"	0.91854667	0.27307317	12.946712	0.23650582	310.74842	150.4935	54000	4	
3274905	"2005 FN"	0.93308734	0.33019919	3.7480425	177.42193	120.84034	160.1944	54000	4	
3275869	"2005 GO21"	0.7532051	0.34020433	24.917136	272.74031	156.55353	293.86874	54000	4	m
3275978	"2005 GR33"	0.77884535	0.38380306	28.00374	22.727001	334.54877	268.0763	54000	4	
3276398	"2005 GE60"	0.95886166	0.24585539	5.5684355	229.95645	112.70238	75.266559	54000	4	
3276601	"2005 GB120"	0.79131389	0.39482129	9.1506832	161.0855	243.61206	140.13948	54000	4	
3276686	"2005 GZ128"	0.95138211	0.13570386	18.653347	203.11495	230.49457	316.55945	54000	4	
3277400	"2005 HN3"	0.85478599	0.33557573	7.8992289	59.628165	6.0811678	42.263504	54000	4	
3278402	"2005 KA"	0.84051334	0.21493604	2.9048383	226.50305	181.14322	102.53286	54000	4	
3279867	"2005 MB"	0.98527456	0.79257712	41.420794	88.69818	42.781343	197.64349	54000	4	m
3283218	"2005 MF5"	0.80376703	0.38177862	29.514513	98.090608	349.71695	85.132792	54000	4	m

3283227	"2005 MR5"	0.85281863	0.29558723	27.788514	263.65459	190.52382	9.5232994	54000	4	m
3283249	"2005 MO13"	0.86343014	0.41102769	6.3146007	176.759	250.09525	64.949008	54000	4	
3283679	"2005 NE21"	0.7892811	0.49636097	10.639743	289.8331	194.62252	51.811201	54000	4	m
3283835	"2005 NW44"	0.77941341	0.48307519	6.0356351	114.63866	0.58825253	89.042008	54000	4	
3283950	"2005 NJ63"	0.86926394	0.42248981	26.58707	120.89719	1.6808852	337.42373	54000	4	
3285073	"2005 OU1"	0.97601434	0.31997874	12.52351	309.81948	204.75039	202.77323	54000	4	
3288855	"2005 QC5"	0.89356017	0.36457579	9.4605918	48.218676	108.6557	278.29264	54000	4	
3288933	"2005 QP11"	0.97558635	0.17580176	3.9570556	334.9335	119.6269	297.85286	54000	4	m
3289173	"2005 QQ87"	0.99925501	0.30304443	33.871509	155.09527	54.330821	115.24323	54000	4	m
3289739	"2005 RB3"	0.87749296	0.39370241	36.068592	165.88104	320.37879	349.70934	54000	4	m
3290865	"2005 SG"	0.98106367	0.28308734	34.840215	23.566708	205.51961	110.17821	54000	4	m
3291224	"2005 SP9"	0.8655333	0.62371237	27.756769	354.90012	228.05435	158.12306	54000	4	
3292261	"2005 TM"	0.84131167	0.41651256	5.2041768	8.67502	151.88261	240.44876	53647	4	
3293790	"2005 TQ45"	0.82682197	0.23366556	25.622782	14.945903	185.79892	267.99709	54000	4	
3293831	"2005 TE49"	0.94924874	0.37694757	5.0055771	195.25531	304.934	276.71907	53652	4	
3293922	"2005 TG50"	0.92386804	0.13387525	2.4267858	346.0492	199.58911	220.00037	54000	4	
3293923	"2005 TH50"	0.83735026	0.22569769	0.73313176	196.83563	18.129628	155.00452	53655	4	
3297182	"2005 UE1"	0.89288955	0.17018689	5.6532049	32.652164	139.63883	260.57564	54000	4	
3297356	"2005 UL5"	0.93729026	0.56966238	14.293632	58.989362	127.58469	263.15198	54000	4	
3297379	"2005 UV64"	0.95814923	0.11600732	5.4166472	216.10028	313.89196	220.99061	54000	4	
3297628	"2005 VK1"	0.74145111	0.42653472	24.545328	223.74752	358.53892	309.99123	54000	4	m
3297629	"2005 VL1"	0.89095046	0.22521846	0.2501009	37.454728	228.7358	132.91306	54000	4	i
3299721	"2005 VN5"	0.94479231	0.23311402	2.0866116	49.351328	115.03663	247.96876	54000	4	
3304566	"2005 WS3"	0.6716719	0.57517628	23.030433	69.435747	176.04117	9.3177124	54000	4	
3305028	"2005 WJ56"	0.95853738	0.15192594	21.623399	288.1221	297.85814	119.57933	54000	4	
3306579	"2005 XZ7"	0.96807507	0.32044152	32.651067	75.70229	238.56512	87.195098	53713	4	m
3307228	"2005 XT77"	0.84064591	0.26639019	17.248864	84.890649	149.93067	220.28289	54000	4	m
3307229	"2005 XV77"	0.78419138	0.41420507	16.853707	282.3189	9.2001035	165.71215	54000	4	m
3309039	"2005 YS"	0.7109288	0.55050252	19.581348	288.69066	327.84503	281.67673	54000	4	
3309091	"2005 YO3"	0.76032651	0.371895	12.795812	274.41391	20.046772	181.31049	54000	4	
3309092	"2005 YR3"	0.81856961	0.27264748	3.6070987	71.203243	222.54974	148.62055	54000	4	
3309828	"2005 YQ96"	0.74387078	0.33313323	22.196585	282.7454	339.97538	249.41827	54000	4	m
3309832	"2005 YO128"	0.8212713	0.2963478	15.586801	281.79624	328.32981	221.01954	54000	4	
3309857	"2005 YU128"	0.77172747	0.32160915	7.7314462	100.74026	189.88574	184.97114	54000	4	
3309858	"2005 YV128"	0.9229554	0.51225824	14.137932	127.80299	191.7352	14.612568	54000	4	
3311964	"2006 AM4"	0.98261634	0.64901708	4.1646477	123.37688	139.52651	173.2487	54000	4	
3313739	"2006 BA9"	0.91259519	0.36592669	8.3160351	305.98482	25.497647	43.203772	54000	4	
3314789	"2006 BQ147"	0.81990323	0.4222185	24.380862	146.84225	153.17094	103.18551	54000	4	
3315649	"2006 CJ"	0.67622812	0.75501232	10.296226	303.38991	29.482862	217.22446	54000	4	
3324253	"2006 DS14"	0.86368636	0.33670256	26.531002	162.3011	187.48359	57.179409	54000	4	m
3324656	"2006 DM63"	0.69549487	0.49762915	1.7812336	336.43224	17.429135	140.24267	53794	4	
3328632	"2006 FK"	0.92250426	0.34340855	14.622177	15.228858	3.1455032	356.30048	54000	4	
3329255	"2006 FH36"	0.95446956	0.19860002	1.5906721	280.91861	154.81012	276.65717	54000	4	i,m
3329278	"2006 GB"	0.95908392	0.17937767	10.060842	183.9189	242.88627	281.19884	54000	4	
3330155	"2006 HV5"	0.84194719	0.31558662	31.806967	35.992828	317.68035	246.80424	53846	4	m
3330538	"2006 HR29"	0.98525825	0.26341245	9.5375754	232.82844	212.56695	246.14269	54000	4	
3330688	"2006 HV50"	0.84884937	0.26080106	16.418388	34.794089	22.616138	331.26257	54000	4	
3333079	"2006 JF42"	0.67190303	0.58163186	5.9724883	41.039218	17.670959	14.966162	54000	4	m
3337162	"2006 MD12"	0.83860414	0.60517513	27.26727	291.85793	174.50168	318.38756	54000	4	m
3337325	"2006 NL"	0.84773943	0.57581932	20.07954	115.27511	29.325849	199.74045	54000	4	

3338368	"2006 QQ23"	0.80375651	0.28451361	3.439796	4.9652624	124.76779	235.02335	54000	4	
3339082	"2006 QQ56"	0.98674925	0.046543451	2.8266463	163.3305	332.95836	225.00633	54000	4	s,i
3341199	"2006 RJ1"	0.95075741	0.30076367	1.4144836	93.52479	110.27498	136.8701	54000	4	
3342322	"2006 SE6"	0.80476877	0.34642503	4.8343905	347.5677	182.91696	190.23549	54000	4	
3342323	"2006 SF6"	0.94928888	0.28045436	5.865093	228.17418	305.53284	190.56084	54000	4	
3342642	"2006 RO36"	0.90638531	0.23131827	23.88335	270.9998	261.22068	144.12389	54000	4	
3343104	"2006 SP19"	0.88152606	0.29156747	4.5585453	358.58023	165.97431	205.79041	54000	4	
3344169	"2006 SF77"	0.92191222	0.32902583	32.483792	1.2876999	224.38265	103.38686	54000	4	m
3347493	"2006 SU217"	0.98577785	0.17475527	2.6419584	194.39292	38.266669	110.3162	54000	4	
3348144	"2006 TL"	0.94018863	0.39616796	11.569969	195.46581	315.13044	265.13542	54013	4	
3350632	"2006 TS7"	0.94745779	0.58305705	5.5041004	225.42431	299.84819	272.59296	54021	4	
3350633	"2006 TU7"	0.85107872	0.46865622	2.9133876	92.154235	68.436379	235.83518	54000	4	
2000003	Juno	2.6676188	0.25819419	12.971682	170.12215	247.82331	75.986335	54000	3	
2000005	Astraea	2.5736689	0.19269048	5.3685634	141.6853	357.50991	266.07961	54000	3	
2000006	Hebe	2.4251131	0.20172572	14.752027	138.74385	239.55732	326.18977	54000	3	m
2000007	Iris	2.3854903	0.23142212	5.527283	259.72283	145.41092	349.83031	54000	3	
2000008	Flora	2.2015458	0.15620731	5.8884175	110.96399	285.39792	246.60931	54000	3	
2000009	Metis	2.3871957	0.1214414	5.5765165	68.973442	5.6901086	127.8487	54000	3	
2000011	Parthenope	2.4522002	0.10011807	4.6247051	125.62803	195.29555	178.97445	54000	3	m
2000012	Victoria	2.334669	0.22056537	8.3638054	235.53818	69.58648	301.90748	54000	3	
2000014	Irene	2.5849886	0.16811947	9.1072838	86.461325	96.32871	156.36084	54000	3	
2000015	Eunomia	2.643284	0.18718147	11.738272	293.27326	97.914615	309.03614	54000	3	
2000017	Thetis	2.4700724	0.1344116	5.58717	125.60807	136.00231	216.015	54000	3	
2000018	Melpomene	2.2955326	0.21870798	10.125237	150.53455	228.00031	43.626263	54000	3	m
2000020	Massalia	2.4091511	0.14287954	0.70691821	206.50811	255.50551	346.24964	54000	3	m
2000023	Thalia	2.6273733	0.2329636	10.145257	67.227831	59.312781	351.82574	54000	3	
2000025	Phocaea	2.3999647	0.25544399	21.584123	214.268	90.161994	33.436221	54000	3	m
2000026	Proserpina	2.656335	0.086901823	3.5621827	45.885083	193.16118	138.34082	54000	3	
2000027	Euterpe	2.3476948	0.17191778	1.5837422	94.806111	356.77913	153.80173	54000	3	m
2000028	Bellona	2.7780323	0.14824802	9.4013833	144.50295	342.55305	15.27945	54000	3	m
2000029	Amphitrite	2.5540421	0.072585419	6.096443	356.49859	63.459167	253.78959	54000	3	
2000030	Urania	2.3665367	0.12638171	2.0974832	307.77461	86.72257	223.47454	54000	3	
2000032	Pomona	2.5879007	0.08299776	5.5306247	220.57549	339.79731	22.315567	54000	3	
2000033	Polyhymnia	2.865151	0.3376296	1.87042	8.59314	338.24659	128.61147	54000	3	
2000037	Fides	2.6414184	0.17665525	3.0732741	7.4126651	62.695182	6.3427201	54000	3	m
2000039	Laetitia	2.7686996	0.11419824	10.382913	157.17103	209.57309	36.855297	54000	3	
2000040	Harmonia	2.2678373	0.046566607	4.2556875	94.292217	268.90779	220.356	54000	3	
2000042	Isis	2.4419555	0.22279671	8.5295276	84.397918	236.6316	96.043894	54000	3	
2000043	Ariadne	2.2032648	0.16794702	3.4679469	264.93471	15.950712	191.974	54000	3	
2000057	Mnemosyne	3.1493067	0.1182737	15.20008	199.33961	212.88978	50.32014	54000	3	
2000060	Echo	2.393591	0.18203332	3.6020835	191.80383	270.41631	64.514348	54000	3	m
2000061	Danae	2.9819926	0.1678048	18.21925	333.7717	13.81601	110.25816	54000	3	
2000063	Ausonia	2.395578	0.12605849	5.7855725	337.91498	295.63623	356.80652	54000	3	
2000067	Asia	2.4213289	0.18484185	6.0269995	202.72444	106.29921	156.01781	54000	3	
2000068	Leto	2.782613	0.18536679	7.9716224	44.182403	305.38918	3.378668	54000	3	
2000071	Niobe	2.7549414	0.176456	23.255563	316.10551	267.4553	352.31621	54000	3	m
2000073	Klytia	2.6660162	0.041362	2.37313	7.23877	54.66187	191.9127	54000	3	
2000079	Eurynome	2.4444158	0.19213223	4.6226653	206.80097	200.35481	123.7389	54000	3	
2000080	Sappho	2.296534	0.2003401	8.66477	218.82041	139.11469	234.60882	54000	3	
2000082	Alkmene	2.7598915	0.2244582	2.8333141	25.636344	110.38231	269.9886	54000	3	

2000089	Julia	2.5500653	0.18377079	16.140829	311.64793	45.00997	104.93539	54000	3	m
2000100	Hekate	3.0933126	0.1650934	6.43021	127.33508	185.87533	130.37461	54000	3	m
2000101	Helena	2.5829817	0.14151875	10.198982	343.4749	347.82922	101.84486	54000	3	
2000103	Hera	2.7026333	0.0795678	5.421	136.27878	190.13768	52.67649	54000	3	
2000113	Amalthea	2.3755464	0.087703611	5.0372639	123.59522	79.053555	337.73369	54000	3	m
2000115	Thyra	2.3806207	0.1915	11.59731	308.99545	96.7467	221.98288	54000	3	
2000116	Sirona	2.7695069	0.13755329	3.5690216	64.03691	93.101434	318.91078	54000	3	
2000118	Peitho	2.4371741	0.1633635	7.74344	47.7451	33.63437	198.28777	54000	3	
2000119	Althaea	2.5812942	0.0810231	5.77832	203.73791	171.29982	338.06631	54000	3	m
2000123	Brunhild	2.6942954	0.1218106	6.42781	307.95504	124.95905	16.10407	54000	3	
2000124	Alkeste	2.6301198	0.076540038	2.9507408	188.18585	63.155748	230.10546	54000	3	m
2000126	Velleda	2.4389509	0.105986	2.9245	23.47891	327.99036	91.12117	54000	3	
2000138	Tolosa	2.4486111	0.1624635	3.20806	54.95009	260.0207	143.2047	54000	3	
2000149	Medusa	2.1744038	0.0653143	0.93695	159.64764	251.12748	253.78577	54000	3	s,i,m
2000151	Abundantia	2.5917414	0.0331371	6.4444	39.04645	134.55059	7.94691	54000	3	
2000158	Koronis	2.8685446	0.0565242	1.00337	278.55081	143.9388	3.12319	54000	3	m
2000167	Urda	2.8527952	0.0336352	2.21049	166.44873	125.83834	28.77731	54000	3	
2000169	Zelia	2.3578594	0.13075	5.50226	354.82138	334.69519	350.02999	54000	3	
2000170	Maria	2.5532105	0.0646822	14.4023	301.46438	157.69423	300.03171	54000	3	m
2000172	Baucis	2.3803732	0.1142941	10.03136	332.0841	359.24594	288.80348	54000	3	
2000174	Phaedra	2.859052	0.1458847	12.12754	327.80693	289.76608	316.3647	54000	3	m
2000178	Belisana	2.4599218	0.0438536	1.89948	51.19998	211.55291	281.47495	54000	3	
2000179	Klytaemnestra	2.970494	0.1155056	7.81766	252.12535	105.37094	104.12798	54000	3	
2000180	Garumna	2.7200002	0.1690067	0.87057	312.70655	175.49316	231.86403	54000	3	
2000181	Eucharis	3.1402839	0.1982904	18.7985	143.59272	317.42132	113.3454	54000	3	m
2000182	Elsa	2.4179605	0.18506679	2.0028901	107.27931	309.90481	156.27731	54000	3	
2000183	Istria	2.7931846	0.349925	26.37466	142.01678	264.16968	253.8036	54000	3	m
2000186	Celuta	2.3617979	0.1499119	13.17267	14.87064	315.24035	18.13794	54000	3	m
2000188	Menippe	2.7629704	0.1775287	11.73414	241.2132	68.31891	295.83763	54000	3	
2000189	Phthia	2.4499896	0.0370351	5.17924	203.61856	166.82805	133.14401	54000	3	
2000192	Nausikaa	2.404096	0.24621664	6.8170145	343.41423	29.840171	221.89748	54000	3	
2000196	Philomela	3.1149136	0.02269419	7.2609296	72.55494	199.92548	346.38828	54000	3	
2000197	Arete	2.7408956	0.1602645	8.79318	81.68189	246.00807	318.73523	54000	3	
2000198	Ampella	2.4593202	0.2278889	9.30913	268.52803	88.55903	290.62198	54000	3	
2000202	Chryseis	3.0764769	0.0962809	8.82984	137.05498	0.94421	72.40597	54000	3	
2000204	Kallisto	2.67313	0.1719995	8.27148	205.21481	55.82695	137.45271	54000	3	
2000208	Lacrimosa	2.891708	0.0154312	1.74941	4.5539	125.39114	191.44426	54000	3	m
2000215	Oenone	2.7669402	0.0344195	1.69006	25.05574	321.44768	34.44622	54000	3	
2000218	Bianca	2.6657744	0.11716641	15.226448	170.87994	60.735184	77.80577	54000	3	m
2000219	Thusnelda	2.3539776	0.2230137	10.84215	200.9526	142.27416	336.21897	54000	3	
2000221	Eos	3.0115033	0.1032757	10.8869	141.94429	195.91626	104.86267	54000	3	
2000228	Agathe	2.2013308	0.241223	2.53831	313.42898	18.77463	353.21101	54000	3	
2000230	Athamantis	2.3832188	0.060890097	9.4382745	239.96029	139.39921	230.91656	54000	3	m
2000234	Barbara	2.3858256	0.2442783	15.35266	144.63948	192.16925	134.46103	54000	3	m
2000235	Carolina	2.8829304	0.0598554	9.02719	66.23928	211.97564	170.0847	54000	3	m
2000236	Honoria	2.8031571	0.1873378	7.68294	186.13681	174.03894	258.17166	54000	3	
2000237	Coelestina	2.7622571	0.0733517	9.75594	84.44051	201.72651	197.94377	54000	3	
2000243	Ida	2.8611801	0.045985314	1.1375042	324.18546	108.3985	28.35885	54000	3	
2000245	Vera	3.1015974	0.19769977	5.1773835	61.525116	327.52007	193.52253	54000	3	m
2000254	Augusta	2.195187	0.1215113	4.51431	28.54322	233.09692	329.78941	54000	3	

2000258	Tyche	2.6152135	0.2051338	14.29304	207.70149	154.94955	38.8324	54000	3	
2000262	Valda	2.5523258	0.2141044	7.70896	38.71081	24.60432	119.04846	54000	3	
2000264	Libussa	2.802004	0.1338519	10.43526	49.77904	340.02132	217.85622	54000	3	
2000270	Anahita	2.1980431	0.15082857	2.3654059	254.5632	80.326577	210.76351	54000	3	
2000277	Elvira	2.8871937	0.0873359	1.16195	231.61197	135.57413	264.50517	54000	3	
2000287	Nephthys	2.3527778	0.023729805	10.023053	142.48241	120.55732	223.14593	54000	3	
2000288	Glauke	2.7554606	0.2101927	4.32933	120.56933	83.12444	123.7304	54000	3	
2000295	Theresia	2.7959558	0.1703209	2.70624	276.12026	148.56188	226.83366	54000	3	
2000296	Phaetusa	2.2287841	0.1597919	1.74678	121.59151	252.57079	286.30861	54000	3	
2000305	Gordonia	3.1048717	0.1859286	4.44565	207.85299	260.20059	112.25397	54000	3	
2000306	Unitas	2.3578262	0.15049	7.26805	142.03554	167.61862	189.61238	54000	3	
2000311	Claudia	2.8978498	0.0078441	3.22488	81.1549	40.37368	271.97283	54000	3	
2000312	Pierretta	2.7808834	0.1616625	9.03534	6.74435	260.2729	0.17345	54000	3	
2000321	Florentina	2.8862686	0.0430939	2.59384	40.46183	30.91652	123.78883	54000	3	
2000328	Gudrun	3.1079701	0.1126465	16.07973	352.61251	101.00197	41.63909	54000	3	m
2000339	Dorothea	3.0128991	0.09462	9.92931	173.79528	160.47059	317.36304	54000	3	m
2000340	Eduarda	2.7459333	0.1173656	4.67867	27.11612	42.5169	323.31427	54000	3	
2000341	California	2.1992684	0.1937417	5.66779	29.1945	293.41228	8.27634	54000	3	
2000346	Hermentaria	2.7950112	0.1025065	8.76066	92.16618	289.94435	152.33019	54000	3	
2000351	Yrsa	2.7639952	0.156579	9.19393	99.44622	31.66235	228.34098	54000	3	
2000352	Gisela	2.1939318	0.1501057	3.38211	247.42532	144.23938	62.80526	54000	3	
2000354	Eleonora	2.8004843	0.11333869	18.379466	140.45317	7.156191	84.076712	54000	3	m
2000364	Isara	2.2209348	0.1490966	6.00474	105.61761	312.93786	210.62896	54000	3	
2000374	Burgundia	2.7796261	0.0798103	8.98647	219.23568	27.75383	246.94554	54000	3	
2000376	Geometria	2.2888879	0.1714106	5.43035	302.25603	316.30919	318.79015	54000	3	
2000378	Holmia	2.7767367	0.1295631	7.01027	232.7592	156.09891	167.15252	54000	3	
2000384	Burdigala	2.6511976	0.1483651	5.60405	48.10942	33.56521	73.08764	54000	3	
2000385	Ilmatar	2.8472955	0.12656961	13.565001	345.24165	188.09878	148.57539	54000	3	
2000387	Aquitania	2.7391293	0.23705267	18.134397	128.31422	157.68246	180.64512	54000	3	s,i,m
2000389	Industria	2.6089236	0.065160176	8.1342143	282.55924	263.55197	227.94975	54000	3	
2000394	Arduina	2.7601348	0.229052	6.22412	67.37124	269.66619	179.60716	54000	3	
2000397	Vienna	2.6347117	0.2465269	12.835543	228.26765	139.38269	93.681871	54000	3	m
2000402	Chloe	2.5584357	0.1127408	11.82138	129.5359	18.17371	115.8071	54000	3	m
2000403	Cyane	2.8107151	0.0966235	9.15501	244.84291	251.86352	320.86082	54000	3	
2000416	Vaticana	2.7913658	0.21853534	12.862244	58.208003	198.84647	225.9835	54000	3	m
2000421	Zahringia	2.5407221	0.2827883	7.77214	187.53759	209.22374	44.4314	54000	3	m
2000432	Pythia	2.3691579	0.1462883	12.13154	88.87357	174.15765	110.98779	54000	3	
2000443	Photographica	2.2155777	0.040032951	4.2304545	175.55548	348.80375	213.66194	54000	3	m
2000453	Tea	2.1829344	0.1089484	5.5578	11.82328	220.16628	299.0741	54000	3	
2000458	Hercynia	2.9945128	0.2423835	12.62356	134.95413	274.82091	141.13805	54000	3	m
2000459	Signe	2.6205643	0.2094949	10.29674	29.57945	19.25982	334.1296	54000	3	
2000462	Eriphyla	2.8740387	0.0830898	3.1911	105.32977	250.36867	293.74567	54000	3	i
2000470	Kilia	2.404738	0.0933121	7.22745	173.30542	46.2614	88.91894	54000	3	
2000471	Papagena	2.8861595	0.23353842	14.98526	84.095024	314.48203	42.91019	54000	3	m
2000472	Roma	2.5439924	0.0938108	15.79966	127.26845	296.35407	218.67478	54000	3	
2000477	Italia	2.4151108	0.1882294	5.28847	10.73814	322.28708	359.09369	54000	3	
2000478	Tergeste	3.0152043	0.0882385	13.17546	234.02285	241.42639	277.32205	54000	3	m
2000480	Hansa	2.6444378	0.0466305	21.29332	237.39549	212.40356	342.11821	54000	3	m
2000482	Petrina	2.9986012	0.1030376	14.46735	179.55439	88.07899	322.25031	54000	3	
2000487	Venetia	2.6712337	0.0864081	10.23447	114.89867	281.04144	229.32471	54000	3	

2000496	Gryphia	2.1987751	0.079568	3.78914	207.7666	258.15505	259.49828	54000	3	
2000502	Sigune	2.3826518	0.1791063	25.00994	133.09316	19.42627	26.13608	54000	3	m
2000509	Iolanda	3.0645243	0.0899784	15.41164	217.81003	157.19742	309.23183	54000	3	m
2000513	Centesima	3.0200353	0.0784204	9.71732	184.69574	222.61276	193.18467	54000	3	
2000519	Sylvania	2.78994	0.185968	11.01574	44.81205	303.08502	61.36499	54000	3	m
2000529	Preziosa	3.0167164	0.0952662	11.02327	65.28007	336.21389	338.78737	54000	3	m
2000532	Herculina	2.7705832	0.17861371	16.313485	107.60159	76.778716	81.370926	54000	3	m
2000533	Sara	2.9792699	0.0425945	6.55229	180.57574	40.39599	315.61114	54000	3	m
2000534	Nassovia	2.883731	0.0570623	3.27661	94.2555	333.48245	104.92852	54000	3	
2000540	Rosamunde	2.2187693	0.0900753	5.57618	202.26254	337.04541	92.34394	54000	3	
2000542	Susanna	2.9068967	0.140981	12.06846	153.25681	214.26388	203.84649	54000	3	m
2000548	Kressida	2.2830652	0.1843779	3.87117	108.5126	320.26734	178.898	54000	3	
2000549	Jessonda	2.6819981	0.2607464	3.96628	291.64694	156.97231	44.20735	54000	3	m
2000550	Senta	2.5885648	0.2210401	10.11404	270.82874	44.6313	246.80346	54000	3	m
2000556	Phyllis	2.465813	0.1016516	5.23196	286.23223	177.69124	103.79414	54000	3	
2000562	Salome	3.0200435	0.0947892	11.12606	70.78883	261.54824	0.27796	54000	3	
2000563	Suleika	2.7115251	0.2362785	10.24831	85.46154	336.61987	31.05904	54000	3	
2000565	Marbachia	2.4441963	0.1283881	10.99229	226.06462	290.83268	264.72051	54000	3	m
2000571	Dulcinea	2.4096288	0.2426047	5.22678	3.25658	27.65282	336.72112	54000	3	
2000574	Reginhild	2.2520296	0.2397801	5.68496	336.85928	76.87759	280.01942	54000	3	m
2000579	Sidonia	3.0098865	0.0827351	11.02146	82.83799	231.62261	199.22196	54000	3	
2000582	Olympia	2.6093875	0.2250105	30.01247	155.81846	309.97325	326.65629	54000	3	m
2000584	Semiramis	2.3743584	0.2329791	10.7257	282.30101	84.784938	276.38474	54000	3	
2000599	Luisa	2.7703969	0.2938211	16.67178	44.68837	292.98145	185.01868	54000	3	m
2000611	Valeria	2.9811262	0.1184348	13.44867	189.87939	253.66966	105.58655	54000	3	m
2000616	Elly	2.5543067	0.0578729	14.96176	356.29941	108.68871	107.63951	54000	3	m
2000619	Triberga	2.5202746	0.0752591	13.78251	187.56918	178.39417	21.96518	54000	3	
2000622	Esther	2.416276	0.24156635	8.6414972	142.12698	256.47956	228.26596	54000	3	
2000631	Philippina	2.7905149	0.0854026	18.93284	224.78721	279.07028	175.92013	54000	3	m
2000633	Zelima	3.0197599	0.0877737	10.90825	147.54125	185.51247	261.16568	54000	3	
2000639	Latona	3.0180846	0.1023474	8.57595	280.09687	67.46954	300.6397	54000	3	
2000642	Clara	3.1960193	0.118776	8.14064	6.78999	112.02326	51.50708	54000	3	
2000644	Cosima	2.6016046	0.1544563	1.04065	109.95042	268.66057	230.99733	54000	3	
2000651	Antikleia	3.0236314	0.0963724	10.76903	38.21433	355.83247	296.88522	54000	3	
2000653	Berenike	3.0142155	0.044892	11.28504	133.23306	49.90575	204.14329	54000	3	
2000658	Asteria	2.8545654	0.0617484	1.50589	351.20658	61.9558	222.70176	54000	3	
2000660	Crescentia	2.5333023	0.1064984	15.21514	157.15003	104.86147	26.85243	54000	3	m
2000661	Cloelia	3.0162704	0.0366429	9.25742	336.01605	169.75515	302.94086	54000	3	
2000669	Kypria	3.0126289	0.0824955	10.78161	170.90917	114.17126	319.51092	54000	3	m
2000673	Edda	2.8147227	0.0107688	2.87946	226.90118	256.96914	146.86493	54000	3	
2000674	Rachele	2.9256113	0.1924971	13.51228	58.23889	42.34252	156.94179	54000	3	
2000675	Ludmilla	2.7677939	0.2042498	9.80224	263.42953	151.95144	49.7925	54000	3	m
2000686	Gersuind	2.588401	0.269099	15.68124	243.42817	88.35887	104.57222	54000	3	
2000695	Bella	2.5396531	0.1599762	13.85554	275.79136	79.48882	27.07343	54000	3	m
2000708	Raphaela	2.6712774	0.0832228	3.48812	355.32832	197.68247	275.39522	54000	3	
2000714	Ulula	2.5353112	0.0572934	14.27153	234.0746	230.23879	331.17093	54000	3	m
2000716	Berkeley	2.8129656	0.0860072	8.4957	146.29921	53.18106	175.64261	54000	3	
2000720	Bohlinia	2.8869458	0.0134885	2.3584	35.93197	104.59844	1.17256	54000	3	
2000736	Harvard	2.2019306	0.1649511	4.37435	135.97225	200.35955	320.7405	54000	3	
2000737	Arequipa	2.5912677	0.2429092	12.36024	184.98656	133.69582	255.05591	54000	3	

2000742	Edisona	3.0109608	0.119093	11.21532	64.367	286.12007	103.97612	54000	3	
2000749	Malzovia	2.2431611	0.1736152	5.38885	109.88041	128.43565	263.27234	54000	3	
2000753	Tiflis	2.3290415	0.2213364	10.08948	61.48843	203.04489	68.30372	54000	3	m
2000770	Bali	2.2209539	0.151461	4.3891	44.8096	17.75077	8.33105	54000	3	
2000775	Lumiere	3.010628	0.0748552	9.28083	298.00083	169.97933	273.2585	54000	3	m
2000782	Montefiore	2.1799237	0.0385567	5.26248	80.53587	81.47485	288.82101	54000	3	
2000797	Montana	2.5346964	0.0602577	4.50134	238.47756	352.71322	72.6863	54000	3	
2000800	Kressmannia	2.1930117	0.2020272	4.2661	325.26844	347.23211	303.5713	54000	3	
2000807	Ceraskia	3.0163759	0.0668862	11.30563	132.34846	341.59535	226.7289	54000	3	m
2000811	Nauheima	2.8951889	0.0757905	3.13615	130.95759	180.32103	194.10173	54000	3	
2000824	Anastasia	2.7946416	0.1328303	8.11508	141.74643	140.23895	47.45443	54000	3	
2000839	Valborg	2.6138575	0.1538848	12.60576	338.27524	339.22259	144.00512	54000	3	
2000847	Agnia	2.7838528	0.0939305	2.48025	271.19928	128.06198	166.21445	54000	3	
2000851	Zeissia	2.2282453	0.0907124	2.39137	141.24049	7.12275	115.0055	54000	3	
2000858	El Djezair	2.8092647	0.1035334	8.88289	67.30295	175.63874	73.08798	54000	3	
2000864	Aase	2.2082083	0.1899383	5.44451	163.21265	193.85566	326.6475	54000	3	
2000876	Scott	3.011357	0.1077304	11.3311	151.14979	210.64758	313.11061	54000	3	
2000883	Matterania	2.2379184	0.1993118	4.71591	285.70745	42.16645	232.0517	54000	3	
2000888	Parysatis	2.7086863	0.1942403	13.85879	124.23326	297.67398	349.84095	54000	3	
2000897	Lysistrata	2.5415771	0.0947401	14.32899	258.05353	22.97731	293.86656	54000	3	m
2000901	Brunsia	2.2237482	0.2215654	3.44429	265.30654	68.06748	201.12121	54000	3	
2000925	Alphonsina	2.6997365	0.081273206	21.068739	299.73888	201.94338	171.13251	54000	3	s,i,m
2000937	Bethgea	2.231828	0.2178144	3.69552	243.80823	72.00629	311.97017	54000	3	
2000939	Isberga	2.2465816	0.1774328	2.5884	327.28426	5.94831	215.58057	54000	3	
2000945	Barcelona	2.6371401	0.162079	32.84988	318.39121	161.12384	7.3489	54000	3	m
2000951	Gaspra	2.2093123	0.17412152	4.1024509	253.21825	129.49617	113.11692	54000	3	m
2000962	Aslog	2.9048251	0.1017686	2.60193	145.66692	225.13159	69.99682	54000	3	
2000963	Iduberga	2.2475028	0.1378807	7.98932	62.56217	4.90994	44.92148	54000	3	
2000966	Muschi	2.7208903	0.1277792	14.39219	72.62607	178.16174	144.49032	54000	3	m
2000968	Petunia	2.8690829	0.1349555	11.59647	209.00875	298.94054	78.47586	54000	3	
2000974	Lioba	2.5327706	0.111811	5.46304	86.76519	301.95786	106.25134	54000	3	
2000975	Perseverantia	2.8338758	0.0354774	2.55965	38.85387	52.91719	345.36437	54000	3	
2001029	La Plata	2.8902401	0.022333	2.42891	30.1445	140.88708	327.05951	54000	3	m
2001043	Beate	3.0918753	0.0468296	8.92769	159.56785	157.83728	246.86415	54000	3	
2001047	Geisha	2.2407153	0.1930689	5.66428	78.33893	299.87517	162.4419	54000	3	s,i
2001052	Belgica	2.2358075	0.1440351	4.69483	99.67349	297.39036	80.72568	54000	3	m
2001055	Tynka	2.1983055	0.2076132	5.2722	147.21064	176.30191	0.29895	54000	3	m
2001058	Grubba	2.1965147	0.187668	3.68964	221.93836	93.99614	318.1367	54000	3	
2001078	Mentha	2.2700948	0.1382134	7.367	93.94718	43.88779	68.71652	54000	3	
2001079	Mimosa	2.8768912	0.0437337	1.17674	329.6275	106.23144	163.95851	54000	3	i
2001087	Arabis	3.0136918	0.0953399	10.0704	30.51174	28.59489	336.51307	54000	3	
2001088	Mitaka	2.2013686	0.1962884	7.65469	54.58213	319.43627	74.07687	54000	3	
2001112	Polonia	3.0214941	0.1014478	8.99516	302.99073	86.60968	259.12752	54000	3	m
2001129	Neujmina	3.0275955	0.0795809	8.60007	269.61883	134.63107	180.36891	54000	3	
2001133	Lugduna	2.1861828	0.1868778	5.37678	58.33863	306.58117	287.54725	54000	3	
2001140	Crimea	2.7718627	0.1109592	14.13293	72.1973	311.2347	288.89636	54000	3	m
2001148	Rarahu	3.0137128	0.1155158	10.84361	145.68035	174.52707	245.78469	54000	3	
2001185	Nikko	2.2379296	0.1053667	5.70048	71.99598	2.23257	182.64599	54000	3	
2001186	Turnera	3.0185508	0.1082265	10.75844	43.21724	294.1858	222.2892	54000	3	
2001215	Boyer	2.5776575	0.1332289	15.91705	123.81564	265.72624	93.16039	54000	3	

2001216	Askania	2.2322234	0.1793551	7.60351	121.67187	144.58578	32.44269	54000	3	
2001223	Neckar	2.868483	0.0605204	2.55052	41.07034	10.09496	109.17591	54000	3	
2001224	Fantasia	2.3050331	0.1986313	7.87488	258.27234	128.83203	192.44571	54000	3	m
2001245	Calvinia	2.8931802	0.077307	2.88642	151.89411	207.55894	296.80457	54000	3	
2001249	Rutherfordia	2.2241593	0.0759426	4.87109	259.11704	223.39854	41.71452	54000	3	
2001252	Celestia	2.6956455	0.204297	33.8886	141.08271	62.91168	193.19854	54000	3	m
2001274	Delportia	2.2289372	0.1135176	4.39773	327.29125	244.48473	288.76466	54000	3	
2001286	Banachiewicza	3.0223851	0.0884699	9.73977	200.93254	102.70785	4.68589	54000	3	
2001289	Kutaissi	2.8612633	0.0584571	1.61241	193.25201	117.98548	59.06832	54000	3	
2001306	Scythia	3.1478297	0.096918	14.91028	274.51946	134.41113	160.8342	54000	3	m
2001307	Cimmeria	2.2510661	0.0962503	3.94621	233.97124	207.16335	128.16734	54000	3	
2001314	Paula	2.2950475	0.1750754	5.24428	264.77378	144.00208	318.51489	54000	3	
2001329	Eliane	2.6180742	0.1709445	14.4656	132.21704	165.03163	26.91872	54000	3	m
2001336	Zeelandia	2.8506578	0.0598015	3.19404	97.51433	218.75154	19.58499	54000	3	
2001339	Desagneauxa	3.019179	0.0560568	8.68943	291.08859	166.96765	224.50772	54000	3	m
2001350	Rosselia	2.857687	0.0873096	2.93719	139.67262	239.65381	327.23494	54000	3	
2001391	Carelia	2.545694	0.1678673	7.58659	103.57028	85.14437	96.86838	54000	3	
2001401	Lavonne	2.2263167	0.1800958	7.28695	277.67193	70.88823	155.25928	54000	3	
2001415	Malautra	2.2234386	0.08697	3.42654	329.4137	240.54671	304.88173	54000	3	
2001416	Renauxa	3.0236638	0.102367	10.04353	353.02629	62.48575	166.59297	54000	3	
2001418	Fayeta	2.2418907	0.2038894	7.19825	355.20138	324.07708	277.18144	54000	3	
2001422	Stromgrenia	2.2478713	0.1669438	2.67554	201.72553	170.75431	259.86018	54000	3	
2001434	Margot	3.0188203	0.0609659	10.81365	152.79993	142.70098	60.04093	54000	3	m
2001442	Corvina	2.8732162	0.0811964	1.25469	221.18371	126.91285	149.86304	54000	3	
2001449	Virtanen	2.2225604	0.1423314	6.63863	110.83068	131.9687	179.36625	54000	3	
2001500	Jyvaskyla	2.2420833	0.190578	7.44355	20.04662	16.77771	79.33813	54000	3	
2001504	Lappeenranta	2.3994278	0.1585168	11.04154	94.96816	51.38235	86.57975	54000	3	m
2001532	Inari	3.0037347	0.0553451	8.7858	330.92626	127.1195	284.23106	54000	3	
2001533	Saimaa	3.0128199	0.0341753	10.69007	156.87489	359.37335	324.18222	54000	3	
2001584	Fuji	2.3765044	0.1942035	26.6425	305.46984	187.97207	263.98458	54000	3	
2001601	Patry	2.2338815	0.12969	4.94351	74.77234	196.50973	69.6855	54000	3	
2001602	Indiana	2.2446852	0.1036564	4.16426	75.1916	73.33204	325.00527	54000	3	
2001619	Ueta	2.2410031	0.1755953	6.21376	61.59932	328.19457	270.73248	54000	3	
2001621	Druzhba	2.2300159	0.1197028	3.16935	182.00627	238.19504	329.30148	54000	3	
2001636	Porter	2.234451	0.1280727	4.4334	168.50633	238.94113	63.78235	54000	3	
2001644	Rafita	2.5505794	0.153738	7.01216	270.95555	197.49709	124.28199	54000	3	
2001648	Shajna	2.2356559	0.2070075	4.56623	130.50431	134.30702	151.96224	54000	3	
2001657	Roemera	2.3493066	0.2345214	23.40366	105.44236	53.94026	238.70086	54000	3	
2001665	Gaby	2.4134306	0.2075508	10.8324	91.66581	6.12477	194.44845	54000	3	m
2001681	Steinmetz	2.6953244	0.207426	7.22492	94.62869	0.72186	0.45966	54000	3	m
2001707	Chantal	2.2187809	0.1709882	4.03832	6.28529	42.38917	107.0253	54000	3	m
2001711	Sandrine	3.0143118	0.1116368	11.08177	134.93638	252.71894	330.46795	54000	3	m
2001717	Arlon	2.1959125	0.1286555	6.19124	340.64383	115.78748	92.40988	54000	3	
2001723	Klemola	3.0115989	0.0458738	10.92085	150.00584	6.18978	172.39803	54000	3	m
2001755	Lorbach	3.0902268	0.0498287	10.69528	157.33044	327.98813	229.63128	54000	3	
2001830	Pogson	2.1881582	0.0558756	3.95267	147.57026	335.13972	22.26099	54000	3	
2001842	Hynek	2.2662171	0.1801126	5.35433	153.5978	125.43054	264.22915	54000	3	
2001990	Pilcher	2.1741611	0.0512001	3.13114	193.75332	12.25806	241.19664	54000	3	m
2002000	Herschel	2.3822032	0.2970119	22.74792	292.18933	129.81927	140.16102	54000	3	m
2002050	Francis	2.3256838	0.2370149	26.57894	72.61229	170.76705	36.08949	54000	3	m

2002052	Tamriko	3.0071574	0.0836701	9.50768	213.98224	207.86004	301.1236	54000	3	m
2002089	Cetacea	2.5336949	0.156226	15.39516	102.80752	287.08913	59.54672	54000	3	m
2002090	Mizuho	3.0715557	0.1351973	11.79893	340.05302	337.43316	329.75184	54000	3	
2002111	Tselina	3.0177442	0.0909077	10.48847	167.35643	233.75672	284.67168	54000	3	
2002156	Kate	2.2423419	0.2018135	5.35454	17.28794	4.19018	170.84981	54000	3	
2002345	Fucik	3.0156037	0.0784936	9.14384	304.07412	139.06296	287.37178	54000	3	
2002411	Zellner	2.2254446	0.0866982	1.61434	131.05565	129.58459	199.82879	54000	3	
2002422	Perovskaya	2.3282179	0.1986802	6.40677	160.1336	52.03561	295.16215	54000	3	
2002430	Bruce Helin	2.3627815	0.2139148	23.44678	46.00305	309.62649	16.44569	54000	3	m
2002510	Shandong	2.2529475	0.1963001	5.26923	102.98427	209.36488	28.3665	54000	3	
2002830	Greenwich	2.3780438	0.2064344	25.32361	49.12056	140.44195	88.0689	54000	3	m
2012746	"1992 WC1"	2.2382462	0.1908783	4.71175	265.31898	81.63331	100.23076	54000	3	m
2000010	Hygiea	3.1366114	0.11799269	3.8423915	283.45768	313.00295	56.250366	54000	2	
2000016	Psyche	2.9197742	0.13948318	3.0956171	150.34418	227.86709	101.78424	54000	2	
2000021	Lutetia	2.4350514	0.16374276	3.0644634	80.91394	250.10776	231.18992	54000	2	
2000022	Kalliope	2.9091742	0.10280545	13.710708	66.236811	356.08934	3.206616	54000	2	m
2000024	Themis	3.1307722	0.13212508	0.7597936	36.007074	107.94133	257.16605	54000	2	
2000031	Euphrosyne	3.1498399	0.22591347	26.31623	31.239635	62.002727	356.86228	54000	2	m
2000034	Circe	2.6851564	0.1087005	5.50356	184.534	330.0982	178.03603	54000	2	
2000035	Leukothea	2.9897564	0.228466	7.93736	353.8186	213.97512	58.38133	54000	2	m
2000036	Atalante	2.7470479	0.30349954	18.431224	358.47404	47.141652	25.346092	54000	2	
2000038	Leda	2.7424397	0.15148566	6.9542317	295.91795	168.63723	86.025664	54000	2	m
2000041	Daphne	2.7654349	0.27184993	15.764855	178.16309	46.221564	226.08638	54000	2	s,i
2000047	Aglaja	2.8776367	0.13528824	4.984635	3.2446812	314.60266	204.79556	54000	2	m
2000054	Alexandra	2.7122777	0.1964305	11.80389	313.45001	345.58093	81.759176	54000	2	i,m
2000055	Pandora	2.7585286	0.14476162	7.1847686	10.539693	4.2608489	93.446378	54000	2	
2000058	Concordia	2.6998953	0.043183319	5.0578201	161.29511	34.44369	352.92475	54000	2	m
2000066	Maja	2.6451	0.1733656	3.04729	7.66789	43.73999	46.22178	54000	2	
2000069	Hesperia	2.980066	0.16704444	8.5813425	185.1207	289.98554	137.4484	54000	2	
2000070	Panopaea	2.6162373	0.1812037	11.58479	47.80504	255.87084	78.00096	54000	2	
2000074	Galatea	2.7784206	0.23987342	4.0751328	197.31394	174.52383	15.554851	54000	2	m
2000075	Eurydike	2.673238	0.3047786	5.00253	359.48157	339.5614	3.77453	54000	2	
2000078	Diana	2.6201204	0.20736367	8.6876426	333.58458	151.41908	330.57112	54000	2	
2000081	Terpsichore	2.8537425	0.2108515	7.81207	1.50533	50.1751	129.18942	54000	2	
2000086	Semele	3.1156178	0.20751135	4.8208178	86.451459	307.76761	247.06828	54000	2	
2000090	Antiope	3.1571193	0.15623753	2.2197774	70.234113	242.52717	58.626905	54000	2	
2000095	Arethusa	3.067408	0.1489283	12.99847	243.14905	155.04622	308.59388	54000	2	
2000097	Klotho	2.668392	0.25700227	11.78307	159.77806	268.6762	351.69754	54000	2	
2000099	Dike	2.6637091	0.19688701	13.858287	41.678878	196.02815	281.54555	54000	2	m
2000104	Klymene	3.1549271	0.15268145	2.7914565	41.872068	30.942654	188.30844	54000	2	
2000105	Artemis	2.3739092	0.17644901	21.460972	188.35744	56.502696	7.3945065	54000	2	
2000107	Camilla	3.4773045	0.078537037	10.047151	173.13537	309.88106	346.53934	54000	2	
2000110	Lydia	2.7340757	0.0781141	5.97379	56.99462	281.76827	284.77871	54000	2	m
2000111	Ate	2.594318	0.10084142	4.923668	305.89396	165.92105	61.815975	54000	2	
2000120	Lachesis	3.1157251	0.059447364	6.9548919	341.51217	231.93546	132.32475	54000	2	
2000121	Hermione	3.4573207	0.13841548	7.599757	73.217451	295.98151	233.08272	54000	2	
2000125	Liberatrix	2.7420783	0.08125	4.65594	169.16097	110.22377	246.10837	54000	2	
2000128	Nemesis	2.7489736	0.12733828	6.2542741	76.459095	302.41385	159.53128	54000	2	
2000129	Antigone	2.8677223	0.21285798	12.218021	136.44012	108.17944	90.343092	54000	2	m
2000132	Aethra	2.6093823	0.3881	25.05101	258.91641	254.37088	276.89429	54000	2	m

2000134	Sophrosyne	2.5632069	0.1166245	11.588804	346.2139	83.692584	86.432954	54000	2	
2000135	Hertha	2.4280968	0.20653464	2.3056068	343.89794	340.03522	220.0661	54000	2	s,i
2000136	Austria	2.2865589	0.0847066	9.56953	186.53627	132.61377	346.10322	54000	2	
2000137	Meliboea	3.1184487	0.21988728	13.42254	202.45635	106.78731	297.02006	54000	2	
2000143	Adria	2.7624553	0.0703414	11.46947	333.23845	250.94068	174.63933	54000	2	
2000144	Vibilia	2.6544514	0.23556544	4.8084344	76.486474	293.65938	40.157765	54000	2	
2000145	Adeona	2.6731693	0.14450786	12.636769	77.453037	44.944313	140.69001	54000	2	
2000146	Lucina	2.7189094	0.064905901	13.074324	84.178218	143.44602	130.23606	54000	2	
2000147	Protogeneia	3.1342865	0.0340697	1.93499	248.73039	106.84911	167.74868	54000	2	
2000156	Xanthippe	2.7331251	0.22220581	9.7483653	242.18207	337.90684	196.55934	54000	2	
2000159	Aemilia	3.1010391	0.11097779	6.1274027	134.33009	335.49307	282.20833	54000	2	
2000161	Athor	2.3795391	0.1374982	9.05318	18.78424	294.39304	102.63249	54000	2	m
2000163	Erigone	2.367393	0.1903735	4.8058965	160.36447	297.49222	26.693449	54000	2	
2000168	Sibylla	3.3757402	0.0673035	4.63413	206.46806	168.09446	325.53015	54000	2	
2000171	Ophelia	3.1334027	0.12927424	2.5460896	100.54761	58.061999	89.493077	54000	2	m
2000173	Ino	2.7416607	0.20834644	14.207656	148.35779	228.00507	117.48297	54000	2	
2000175	Andromache	3.1847599	0.2324518	3.21884	21.39951	321.25122	130.21293	54000	2	
2000185	Eunike	2.7394441	0.1271362	23.22034	153.94657	224.0974	264.43276	54000	2	m
2000187	Lamberta	2.7321898	0.23670797	10.597906	21.904126	195.14152	153.41892	54000	2	
2000194	Prokne	2.6181429	0.23614207	18.485624	159.51944	162.83374	284.62328	54000	2	
2000195	Eurykleia	2.8803618	0.04057	6.96866	7.20269	123.4987	102.20799	54000	2	
2000200	Dynamene	2.7369494	0.13381634	6.9014922	324.69997	86.014728	305.98131	54000	2	
2000201	Penelope	2.6787336	0.1792631	5.75756	157.11692	181.24502	349.42127	54000	2	
2000205	Martha	2.7767208	0.0361837	10.69477	212.0672	171.02616	158.14767	54000	2	m
2000206	Hersilia	2.7403923	0.040846283	3.7803055	145.27928	302.04137	284.70264	54000	2	
2000207	Hedda	2.2836787	0.0286959	3.80295	29.28743	192.12819	87.89963	54000	2	
2000209	Dido	3.1445726	0.062655847	7.1714093	0.78538285	250.07104	34.32137	54000	2	m
2000211	Isolda	3.0409546	0.16227833	3.8821037	263.75641	174.76297	309.57543	54000	2	m
2000216	Kleopatra	2.797122	0.25036608	13.133755	215.66472	179.3494	181.60763	54000	2	
2000224	Oceana	2.6444588	0.0460098	5.83838	353.0192	284.05593	257.00632	54000	2	
2000232	Russia	2.5502931	0.1778829	6.0708	152.52032	51.12076	64.32308	54000	2	
2000238	Hypatia	2.9086285	0.0879187	12.4025	184.19336	207.19158	178.19462	54000	2	
2000240	Vanadis	2.6643756	0.2068432	2.10451	115.22263	300.50223	324.5337	54000	2	m
2000250	Bettina	3.153134	0.1270277	12.82149	24.05332	75.74685	142.31316	54000	2	
2000266	Aline	2.8043578	0.15736045	13.390532	236.00128	151.25527	16.632347	54000	2	m
2000304	Olga	2.4030494	0.2214414	15.83975	159.22264	172.3242	191.50577	54000	2	
2000313	Chaldaea	2.3756945	0.1794471	11.6459	176.82084	315.81027	13.19227	54000	2	
2000325	Heidelberga	3.202845	0.1680242	8.54318	345.28451	67.86966	111.00173	54000	2	
2000326	Tamara	2.3176268	0.19054712	23.724144	32.334512	238.50135	100.32007	54000	2	m
2000329	Svea	2.4764225	0.0238695	15.88431	178.54468	52.19986	96.05795	54000	2	
2000334	Chicago	3.885343	0.024215343	4.642853	130.2253	151.07984	5.201958	54000	2	s,i
2000338	Budrosa	2.9120186	0.0201877	6.03807	287.63451	123.54059	295.61868	54000	2	
2000342	Endymion	2.5681538	0.128678	7.34603	232.74759	225.4028	193.24988	54000	2	
2000344	Desiderata	2.5948773	0.31596263	18.356075	48.25056	237.38162	172.27458	54000	2	
2000345	Tercidina	2.3253732	0.061343146	9.751305	212.79371	229.68958	8.5651683	54000	2	
2000347	Pariana	2.6131906	0.1641544	11.69434	85.84551	84.86164	248.97015	54000	2	
2000350	Ornamenta	3.1112374	0.1561224	24.90012	90.19672	338.9865	249.8531	54000	2	m
2000356	Liguria	2.7559652	0.2397877	8.23048	354.86413	78.80956	333.70592	54000	2	
2000360	Carlova	2.9986233	0.1817627	11.71228	132.65874	288.55851	27.11377	54000	2	
2000369	Aeria	2.6491767	0.0976895	12.7077	94.38445	269.55336	10.20211	54000	2	

2000373	Melusina	3.1131734	0.1465925	15.44736	4.0578	347.77644	202.41227	54000	2	
2000375	Ursula	3.1231061	0.107686	15.93372	336.67192	344.59878	192.61115	54000	2	
2000380	Fiducia	2.6788222	0.1132572	6.15571	95.21741	240.54282	21.26849	54000	2	
2000381	Myrrha	3.2204713	0.0955587	12.52422	125.34741	137.52796	102.99105	54000	2	m
2000382	Dodona	3.1156156	0.1770362	7.40193	313.60394	270.78992	81.18543	54000	2	
2000386	Siegena	2.8945485	0.17292904	20.254208	166.94275	220.14756	65.82279	54000	2	m
2000388	Charybdis	3.0067013	0.05887506	6.4580588	354.62692	333.05707	49.966331	54000	2	
2000393	Lampetia	2.7790703	0.33144266	14.870607	212.51713	91.047917	126.41939	54000	2	m
2000395	Delia	2.7858859	0.0838058	3.35143	259.62654	12.95644	167.79539	54000	2	
2000404	Arsinoe	2.5943355	0.1983143	14.115702	92.674305	121.11608	271.32457	54000	2	m
2000405	Thia	2.5837501	0.24463276	11.950979	255.29974	309.27005	353.02963	54000	2	m
2000407	Arachne	2.6248052	0.0707861	7.53482	294.83119	82.01003	40.65893	54000	2	
2000410	Chloris	2.7285227	0.2365507	10.92226	97.21109	172.00263	59.08877	54000	2	m
2000413	Edburga	2.584411	0.3431661	18.71569	103.92107	252.68584	308.8369	54000	2	
2000414	Liriope	3.5083344	0.0689192	9.54215	110.74346	326.64868	352.52959	54000	2	m
2000418	Alemannia	2.5920701	0.119635	6.82571	249.0928	125.96347	103.5344	54000	2	
2000423	Diotima	3.0676241	0.041046488	11.240443	69.554082	206.55983	290.60245	54000	2	
2000429	Lotis	2.6070974	0.12333371	9.5273188	220.04862	168.7474	325.59031	54000	2	m
2000441	Bathilde	2.8060136	0.0827663	8.1419	253.86288	201.35201	313.76537	54000	2	m
2000442	Eichsfeldia	2.3449721	0.0714341	6.06243	135.03435	85.04871	278.05265	54000	2	
2000444	Gyptis	2.771173	0.17286275	10.278705	195.8366	155.21028	322.30187	54000	2	
2000445	Edna	3.1997043	0.1909852	21.37172	292.4142	79.8239	291.93822	54000	2	m
2000448	Natalie	3.1381577	0.1843979	12.71486	37.37604	295.24661	107.79207	54000	2	
2000449	Hamburga	2.5526781	0.1708912	3.08986	86.03696	46.37252	342.694	54000	2	
2000466	Tisiphone	3.3583767	0.082421992	19.16314	291.19859	245.93322	345.95823	54000	2	m
2000481	Emita	2.7389796	0.1582352	9.8583	67.0244	348.66307	80.4701	54000	2	
2000488	Kreusa	3.1579038	0.1692412	11.500393	84.980185	68.917033	289.65081	54000	2	
2000489	Comacina	3.1535619	0.038868229	12.976896	167.18408	12.619788	7.9310261	54000	2	
2000490	Veritas	3.1685051	0.0989462	9.26516	178.5047	196.80207	120.23447	54000	2	m
2000494	Virtus	2.9888645	0.0568189	7.07552	38.39191	216.78947	177.45085	54000	2	
2000497	Iva	2.8579034	0.2973859	4.82186	6.77463	2.80643	228.05837	54000	2	
2000498	Tokio	2.6503781	0.2251495	9.50401	97.49661	241.2688	75.92796	54000	2	
2000508	Princetonia	3.1607974	0.014003443	13.36355	44.510226	179.85303	129.14648	54000	2	
2000511	Davida	3.1659016	0.18564446	15.938414	107.6717	338.52068	247.71727	54000	2	m
2000516	Amherstia	2.6796943	0.2735302	12.95664	328.88736	258.53574	296.93286	54000	2	
2000521	Brixia	2.7417331	0.2813483	10.59138	89.69902	316.09633	246.32803	54000	2	
2000535	Montague	2.5690206	0.0229292	6.78274	84.87326	68.24631	20.05767	54000	2	
2000558	Carmen	2.9064565	0.0431148	8.3662	143.87801	317.36518	215.4325	54000	2	
2000559	Nanon	2.7122622	0.064323766	9.3105401	112.21021	130.74	212.48385	54000	2	
2000566	Stereoskopia	3.3823202	0.1108082	4.89863	80.26508	294.61109	331.04968	54000	2	
2000569	Misa	2.6561647	0.1828515	1.29584	301.97392	141.80532	34.01162	54000	2	m
2000583	Klotilde	3.1705289	0.1619588	8.25067	257.98239	253.69818	179.04336	54000	2	
2000585	Bilkis	2.4303409	0.1296983	7.5572	180.37753	327.97351	205.455	54000	2	
2000593	Titania	2.6975729	0.2179818	16.89206	76.18743	30.92169	286.61962	54000	2	m
2000602	Marianna	3.0914405	0.2440861	15.07537	331.64478	45.83763	302.7661	54000	2	m
2000618	Elfriede	3.1900659	0.078422036	17.012019	111.19981	227.86543	239.39252	54000	2	m
2000635	Vundtia	3.1443395	0.0786644	11.03908	183.46648	219.42795	168.40125	54000	2	
2000654	Zelinda	2.2968024	0.23192318	18.124633	278.57048	214.07158	112.35919	54000	2	m
2000701	Oriola	3.0157244	0.0315655	7.1136	244.14352	313.54037	228.69453	54000	2	
2000702	Alauda	3.1943626	0.022868874	20.60234	289.97403	352.37054	337.72603	54000	2	m

2000712	Boliviana	2.5738464	0.1881205	12.781487	231.04829	181.1623	80.890211	54000	2	
2000713	Luscinia	3.391033	0.16608003	10.359739	217.79282	136.08384	286.71598	54000	2	
2000735	Marghanna	2.7292584	0.3214867	16.87956	43.03	310.136	332.75553	54000	2	m
2000751	Faina	2.5502414	0.153174	15.61479	78.93492	302.31969	163.86706	54000	2	
2000755	Quintilla	3.1724209	0.1466565	3.23894	177.2636	42.40449	145.37775	54000	2	
2000764	Gedania	3.1850605	0.1070606	10.07278	259.37313	154.7591	100.1314	54000	2	
2000772	Tanete	3.0035141	0.092042448	28.783444	64.045057	142.09532	205.57057	54000	2	m
2000776	Berbericia	2.9350442	0.16086041	18.245702	79.867084	306.61868	249.80628	54000	2	i,m
2000785	Zwetana	2.5692316	0.2105935	12.73156	72.18331	129.96803	128.71164	54000	2	m
2000786	Bredichina	3.1687258	0.1671064	14.55243	89.92703	133.60068	115.60687	54000	2	
2000791	Ani	3.115249	0.1996704	16.38494	130.11717	202.04482	217.95885	54000	2	m
2000798	Ruth	3.0142313	0.0413488	9.22952	214.52891	41.83994	8.942	54000	2	m
2000814	Tauris	3.1498691	0.3093096	21.83526	88.86011	297.08583	85.74032	54000	2	m
2000821	Fanny	2.7776856	0.207103	5.37832	209.91865	32.90585	163.61022	54000	2	
2000849	Ara	3.1547846	0.1953547	19.48619	228.51247	63.9943	160.73835	54000	2	
2000860	Ursina	2.7945762	0.1091712	13.31523	309.55072	19.97543	210.65456	54000	2	
2000872	Holda	2.7313074	0.0784015	7.36689	194.94024	20.02667	299.6105	54000	2	
2000907	Rhoda	2.7977888	0.1637967	19.57368	43.17347	88.29047	211.84409	54000	2	m
2000931	Whittemora	3.185335	0.2233883	11.44311	111.48085	313.89267	191.3093	54000	2	m
2000977	Philippa	3.1153896	0.0293926	15.19978	75.90784	83.30353	169.14985	54000	2	m
2001015	Christa	3.208478	0.081469182	9.4580072	120.49107	282.54502	235.74511	54000	2	
2001028	Lydina	3.3947356	0.1185204	9.39108	63.48771	25.06466	47.074	54000	2	
2001061	Paeonia	3.1383114	0.2082506	2.49726	91.29504	303.85151	225.10094	54000	2	
2001082	Pirola	3.1216608	0.1808789	1.85016	148.02332	188.61985	134.3123	54000	2	m
2001093	Freda	3.1304429	0.2707514	25.20975	55.69696	251.94642	183.03763	54000	2	m
2001102	Pepita	3.0687473	0.1173315	15.81039	216.84653	114.79816	256.27357	54000	2	
2001277	Dolores	2.6992491	0.239092	6.9664	247.25447	47.368	145.46956	54000	2	
2001445	Konkolya	3.1234889	0.1772867	2.28497	89.30058	270.99423	282.75712	54000	2	
2001461	Jean-Jacques	3.1272276	0.0423866	15.32426	104.84379	334.56144	191.21196	54000	2	
2001580	Betulia	2.1968759	0.48791122	52.097105	62.32879	159.50668	121.45456	54000	2	s,i,m
2001606	Jekhovsky	2.6913513	0.3150995	7.70009	190.77887	142.30677	251.45853	54000	2	m
2001625	The NORC	3.2016194	0.2218822	15.55532	322.144	282.75245	181.27996	54000	2	m
2001639	Bower	2.5719099	0.1512801	8.42685	324.37971	105.46543	72.49232	54000	2	
2001794	Finsen	3.1288813	0.1552413	14.50739	221.50437	335.55787	231.28079	54000	2	
2001931	Capek	2.5404735	0.27278	8.2461	182.53508	163.92287	45.11593	54000	2	
2001963	Bezovec	2.4215853	0.21073058	25.051232	106.98322	355.82804	179.40627	54000	2	
2002379	Heiskanen	3.1646802	0.2762638	0.46741	151.27404	177.62201	154.27748	54000	2	
2002407	Haug	2.9224974	0.2211028	2.47635	342.34897	10.80364	43.83693	54000	2	
2000225	Henrietta	3.3888671	0.2668163	20.887897	197.19179	104.2599	310.40788	54000	1	
2000361	Bononia	3.9554449	0.21241778	12.631655	18.960258	68.162409	239.42006	54000	1	
2000588	Achilles	5.1947906	0.1465778	10.32134	316.59634	132.48447	216.43354	54000	1	
2000617	Patroclus	5.2266795	0.1381751	22.03587	44.35812	307.5839	183.22681	54000	1	m
2000624	Hektor	5.2272807	0.023528317	18.193628	342.80533	185.13006	151.0004	54000	1	m
2000659	Nestor	5.1922184	0.1166258	4.51942	350.88872	341.45996	318.17164	54000	1	i
2000911	Agamemnon	5.25414	0.065868016	21.788152	338.02074	80.347917	253.5207	54000	1	s,i
2000944	Hidalgo	5.7544224	0.66019798	42.532955	21.557042	56.72622	43.435582	54000	1	m
2001038	Tuckia	3.9619932	0.227483	9.22988	58.20488	307.19412	201.38868	54000	1	m
2001172	Aneas	5.1915116	0.10319405	16.683194	247.39536	49.477769	250.4737	54000	1	
2001173	Anchises	5.3227896	0.1371044	6.90993	283.91796	40.01722	205.92255	54000	1	
2001208	Troilus	5.2374808	0.0909268	33.56575	48.55332	295.69051	204.17234	54000	1	m

2001345	Potomac	3.9798138	0.1808098	11.40178	137.5123	332.82006	181.8755	54000	1	s,i,m
2001362	Griqua	3.2218943	0.3698017	24.2039	121.39505	262.11257	18.88301	54000	1	
2001404	Ajax	5.3026206	0.1135323	18.00703	332.97289	59.33793	267.83412	54000	1	
2001437	Diomedes	5.1600747	0.0433457	20.52178	315.83789	129.75256	244.24908	54000	1	m
2001583	Antilochus	5.1071568	0.0522215	28.55192	221.3685	186.33737	260.04144	54000	1	
2001749	Telamon	5.1722902	0.1078736	6.08844	341.00747	111.19397	197.09829	54000	1	s,i,m
2001754	Cunningham	3.9500648	0.1672029	12.11822	163.24455	111.93204	295.64241	54000	1	m
2001867	Deiphobus	5.1330811	0.0437056	26.9088	283.69667	358.70429	250.58354	54000	1	m
2001873	Agenor	5.2491321	0.0922724	21.85938	197.91284	356.47324	335.83439	54000	1	m
2001902	Shaposhnikov	3.971589	0.2236554	12.49318	59.4253	267.89541	47.99795	54000	1	
2002207	Antenor	5.1267457	0.0172127	6.81023	159.1796	300.66192	81.0411	54000	1	
2002223	Sarpedon	5.1985031	0.0140337	15.99057	220.98833	51.37102	279.17581	54000	1	
2002241	Alcathous	5.2059683	0.0670944	16.60687	267.98621	291.17519	324.5333	54000	1	
2002260	Neoptolemus	5.1896664	0.0439744	17.78242	86.57157	319.93149	252.91054	54000	1	
2002357	Phereclos	5.1921105	0.0438295	2.66967	179.31425	72.48815	292.0237	54000	1	
2002363	Cebriones	5.1655208	0.0356465	32.19648	211.83521	51.98333	290.96291	54000	1	m
2002456	Palamedes	5.1372368	0.074439	13.90266	327.41517	93.63287	230.61779	54000	1	
2002483	Guinevere	3.9660655	0.2763629	4.49881	252.1627	182.86999	226.77818	54000	1	m
2002674	Pandarus	5.1717545	0.0678804	1.85443	179.86277	37.74182	321.3662	54000	1	s,i,m
2002759	Idomeneus	5.1700791	0.0660091	21.96815	171.23528	8.41746	126.14053	54000	1	
2002760	Kacha	3.9834018	0.1223004	13.46141	352.816	155.94779	327.81918	54000	1	m
2002797	Teucer	5.1057194	0.0878189	22.39234	69.9464	47.69826	175.56344	54000	1	
2002893	Peiroos	5.17689	0.0764874	14.64719	108.76807	171.07065	241.30879	54000	1	
2002906	Caltech	3.1616197	0.1137911	30.6904	84.60812	294.92209	165.70717	54000	1	s,i
2002920	Automedon	5.1130804	0.0269571	21.11915	230.96163	196.35511	223.31577	54000	1	m
2002959	Scholl	3.9440841	0.2742882	5.23244	121.33837	284.68371	297.55373	54000	1	m
2003063	Makhaon	5.1854315	0.0584969	12.17397	287.88023	204.89888	178.85982	54000	1	
2003134	Kostinsky	3.9793318	0.2201713	7.63747	257.10507	163.00334	272.12822	54000	1	s,i
2003317	Paris	5.2132675	0.1259699	27.88043	135.9195	149.13296	258.45877	54000	1	
2003548	Eurybates	5.1604558	0.0888397	8.07572	43.54071	26.8289	246.50665	54000	1	
2003552	Don Quixote	4.2298497	0.71337656	30.907935	350.30192	316.99246	237.325	54000	1	s,i
2003708	"1974 FV1"	5.220393	0.1577286	13.36738	291.17902	56.85664	189.19123	54000	1	
2003709	Polypoites	5.2609102	0.0614257	19.60369	187.17563	245.59639	219.05819	54000	1	
2003793	Leonteus	5.1948389	0.0891835	20.92981	200.52401	262.18506	206.49881	54000	1	
2004035	"1986 WD"	5.2795635	0.0562528	12.134	233.73474	197.12702	232.16471	54000	1	
2004063	Euforbo	5.1695887	0.1181985	18.94705	113.52578	317.49315	233.45724	54000	1	
2004086	Podalirius	5.2256622	0.1209423	21.7387	54.95767	356.03329	270.21371	54000	1	
2004489	"1988 AK"	5.2761594	0.0602848	22.16378	86.70204	5.3875	190.45873	54000	1	m
2004543	Phoinix	5.0957247	0.0959612	14.73139	325.4165	84.13779	262.49894	54000	1	i
2004709	Ennomos	5.1962516	0.0204462	25.51675	253.23497	89.40599	213.85689	54000	1	m
2004754	Panthoos	5.1909721	0.0080923	12.34371	155.23002	213.07335	186.6018	54000	1	
2004791	Iphidamas	5.1792451	0.0461581	25.94351	261.4468	165.16477	99.89186	54000	1	i
2004833	Meges	5.2562716	0.0928909	34.68256	101.76168	278.74281	271.01003	54000	1	m
2004834	Thoas	5.2357128	0.1355501	28.45078	76.07509	350.67143	223.76284	54000	1	m
2004836	Medon	5.1823797	0.1077418	19.41902	82.04271	34.03318	195.75845	54000	1	m
2004837	Bickerton	3.1977504	0.1314229	28.22335	327.27954	42.92711	296.90452	54000	1	
2005012	Eurymedon	5.2652415	0.0857824	4.99793	34.82759	333.0567	293.9424	54000	1	
2005025	"1986 TS6"	5.2111349	0.0753205	11.01688	347.86522	72.09646	233.10794	54000	1	m
2005130	Ilioneus	5.2456965	0.0099851	15.70813	242.53246	103.5806	169.16343	54000	1	
2005144	Achates	5.2263273	0.2720738	8.89879	322.85738	330.63494	227.70228	54000	1	

2005209	"1989 CW1"	5.1533221	0.0495159	9.06609	322.75689	104.50996	257.03537	54000	1	
2005259	Epeigeus	5.1886873	0.0731641	15.93135	67.46291	199.37861	37.88812	54000	1	
2005264	Telephus	5.2043321	0.1107648	33.57519	121.90801	358.9272	178.74752	54000	1	
2006090	"1989 DJ"	5.3136952	0.0571189	20.18055	328.52802	72.38497	259.09504	54000	1	i,m
2006984	Lewiscarroll	3.9692255	0.1871492	16.80098	206.29149	247.90573	216.503	54000	1	
2007119	Hiera	5.2067455	0.1031099	19.26732	285.66811	119.08349	235.10541	54000	1	
2007641	"1986 TT6"	5.2186575	0.0535495	34.68862	242.05333	227.73421	184.77791	54000	1	m
2009661	Hohmann	3.9462035	0.2336894	12.9873	56.85707	288.56144	8.57451	54000	1	
2009799	"1996 RJ"	5.1942001	0.0479145	30.51326	259.54994	113.20009	280.79516	54000	1	
2011351	"1997 TS25"	5.2491115	0.0638223	11.5758	251.09162	159.70997	276.73337	54000	1	
2011395	"1998 XN77"	5.2073823	0.0672427	24.14544	213.23624	117.37947	328.72511	54000	1	
2011396	"1998 XZ77"	5.2054308	0.0640564	12.58719	195.68829	175.96375	289.49289	54000	1	
2011542	"1992 SU21"	3.9501468	0.2391642	6.87574	16.88982	48.9603	229.49648	54000	1	m
2012444	Prothoon	5.2602165	0.0713189	30.80052	213.19945	64.17095	256.62017	54000	1	m
2012714	Alkimos	5.2062335	0.0356216	9.51657	298.91469	163.16174	210.69903	54000	1	
2014268	"2000 AK156"	5.2873272	0.0906027	14.94484	284.67115	123.62387	237.9447	54000	1	
2014569	"1998 QB32"	3.977245	0.2859047	10.90696	346.1705	37.5375	338.80357	54000	1	m
2015278	Paquet	3.9846457	0.2157149	9.29657	344.88781	64.13255	258.62785	54000	1	
2015436	"1998 VU30"	5.2056216	0.0439938	16.26675	253.42915	178.28135	227.83254	54000	1	s,i
2016070	"1999 RB101"	5.1235811	0.1236231	16.25994	300.91274	352.61268	243.85689	54000	1	
2016560	"1991 VZ5"	5.0765482	0.0408516	15.29915	100.75566	157.40487	261.64642	54000	1	
2016974	"1998 WR21"	5.2149294	0.0702616	15.01434	241.62526	134.19018	272.57798	54000	1	
2020898	Fountainhills	4.2287314	0.4647309	45.4942	293.29012	234.87212	212.42781	54000	1	m
2032511	"2001 NX17"	5.0492697	0.4282629	8.93717	285.89591	345.49142	172.48229	54000	1	
2100004	"1983 VA"	2.6090888	0.69471344	16.289474	77.431206	11.737654	142.26727	54000	1	
3035962	"2000 EJ37"	4.6900143	0.7036003	10.21541	183.5988	111.51786	202.03203	54000	1	
3046844	"2000 PG3"	2.8266851	0.85854077	20.464243	326.68865	138.58443	74.835735	54000	1	
3061681	"2000 SB1"	3.346388	0.5398696	22.15021	277.03743	145.21595	341.7947	54000	1	
3079876	"2001 KX67"	3.1268516	0.5748146	20.68275	245.06986	115.06575	323.76911	54000	1	
3081550	"2001 OB74"	3.0440863	0.5018147	16.79417	260.38958	38.40177	356.90507	54000	1	
3089425	"2001 QQ199"	5.3324536	0.4300086	42.44729	213.05295	193.03549	136.63782	54000	1	
3091801	"2001 QF6"	7.2342917	0.6863551	24.22652	144.27642	240.32669	85.9533	54000	1	m
3117599	"2002 CX174"	3.0704328	0.5119076	21.62245	355.71658	225.09482	290.06416	54000	1	
3169278	"2003 WV157"	3.1343661	0.5204205	20.4243	252.16872	250.66259	169.20267	54000	1	

APPENDIX B

SET OF GTOC3 ASTEROIDS

Table 34 lists all of the asteroids in the full GTOC3 problem, including their SPK-ID number, common name, and orbital elements (semi-major axis, eccentricity, inclination, longitude of the ascending node, argument of periapsis, mean anomaly, and epoch). The asteroids are sorted by their SPK-ID number.

Table 34: GTOC3 Asteroids.

spkid	name		a (AU)	e	i (deg)	LAN (deg)	arg. periapsis (deg)	M (deg)	epoch (MJD)
1	2004	ER21	0.900329	0.17105522	7.95592	357.43143	343.23658	52.625406	54200
2	1995	CR	0.906736	0.86845445	4.03569	342.77208	322.39782	328.3017003	54200
3	1992	BF	0.9080026	0.27176608	7.25394	315.47182	336.42045	54.111859	54200
4	2002	XY38	0.9096341	0.21765254	2.09641	159.63529	119.41817	141.6151818	54200
5	1999	AO10	0.911399	0.11078484	2.62269	313.33854	7.63927	309.5515364	54200
6	2000	WO107	0.9114029	0.78065556	7.7829	69.37362	213.58377	169.2181562	54200
7	2006	BA9	0.9125604	0.3659925	8.31532	305.97027	25.50724	269.3288907	54200
8	2001	HY7	0.9139376	0.41209549	5.20961	205.38304	211.00626	69.2586068	54200
9	2001	RV17	0.9140581	0.34250016	7.52263	154.12079	4.35647	343.2637171	54200
10	1989	UQ	0.9152487	0.26483239	1.29152	178.29646	15.0241	189.1719934	54200
11	2000	LG6	0.9163425	0.11221324	2.83037	72.72983	7.8651	102.1739718	54200
12	2003	CA4	0.9203666	0.1197246	7.47925	139.95566	172.94835	86.6971188	54200
13	2004	RX10	0.9204306	0.35114382	5.9586	173.89189	333.85636	186.4545742	54200
14	2002	EM7	0.9212257	0.36304986	1.54747	347.21646	57.6856	354.1537351	54200
15	Apophis		0.9222614	0.19105942	3.33131	204.45915	126.38557	307.3630785	54200
16	2005	TG50	0.9238455	0.13388842	2.42676	346.04551	199.60593	81.9756648	54200
17	2002	JR100	0.924675	0.29782641	3.76334	203.5673	253.42378	293.6715722	54200
18	1992	FE	0.9272536	0.40535098	4.79702	311.9926	82.42602	52.7808967	54200
19	2000	SZ162	0.9294449	0.16754816	0.89598	14.74941	131.30472	336.3249638	54200
20	2002	TZ66	0.9300326	0.12059229	8.47831	13.13956	223.06212	135.7560272	54200
21	2003	YS17	0.9303548	0.31304457	6.52409	99.17972	134.4818	133.8075539	54200
22	2006	VX2	0.9330736	0.29005873	9.86637	47.38582	126.40932	69.0741375	54200
23	2005	FN	0.9330778	0.33020963	3.74806	177.42092	120.83813	18.9011538	54200
24	2002	OA22	0.9359164	0.24295478	6.90562	174.41213	318.28487	205.6768657	54200
25	1999	AQ10	0.937275	0.23448424	6.55836	327.40416	299.49467	250.5925532	54200
26	2000	EW70	0.9376817	0.32110421	5.41918	178.31485	125.311	188.0399576	54200
27	2001	CQ36	0.9396849	0.17623802	1.29202	31.94641	342.49359	22.9155357	54200
28	2001	BA16	0.9403125	0.13737419	5.76859	115.61889	242.8389	42.1907675	54200
29	2004	QD14	0.9426399	0.33810282	6.25178	75.45204	109.28368	66.0525408	54200
30	2004	VJ1	0.9437535	0.16446907	1.29373	233.53809	332.36301	75.2928449	54200
31	2005	VN5	0.9447613	0.23312124	2.08656	49.35063	115.04505	102.617135	54200
32	2006	TS7	0.9466118	0.57983607	5.46456	225.45909	299.72601	103.8271836	54200
33	2001	SQ263	0.9480374	0.49151246	3.95115	327.30349	262.3598	78.649304	54200
34	2005	TE49	0.9492487	0.37694757	5.00558	195.25531	304.934	276.7190717	53652
35	2006	SF6	0.9493911	0.28039283	5.86507	228.16645	305.51522	43.6895559	54200

36	2006	RJ1	0.9508113	0.30070707	1.4145	93.51404	110.28027	349.5007972	54200
37	2004	QA22	0.9508977	0.12172568	0.57414	175.15217	28.54873	55.2451141	54200
38	2001	TD	0.9541288	0.16608077	9.01107	13.21665	241.33953	68.8170052	54200
39	2004	UH1	0.9541748	0.39680796	3.71292	29.85969	120.62228	155.2936477	54200
40	2006	FH36	0.9544687	0.1986	1.59072	280.91015	154.77967	128.0764061	54200
41	2002	LY1	0.9550094	0.37931534	2.9091	248.2326	133.96031	338.4972621	54200
42	2005	UV64	0.9580862	0.11600696	5.41663	216.09819	313.93026	71.146594	54200
43	2005	GE60	0.9588639	0.24584744	5.56845	229.9559	112.69363	285.2220831	54200
44	2004	RO111	0.9611941	0.32883148	5.33444	199.44692	280.82215	164.3568742	54200
45	1998	HD14	0.9630822	0.31264453	7.80717	183.96655	260.73453	261.9364311	54200
46	2004	UT1	0.9644067	0.22113792	4.50799	211.98171	294.22407	123.9756016	54200
47	2003	YG136	0.968992	0.35504091	2.7352	86.5446	127.9831	81.832236	54200
48	2005	QP11	0.9755418	0.1757465	3.95693	334.92822	119.64718	142.418984	54200
49	2000	SG344	0.9774002	0.06697124	0.11024	192.31139	274.9223	180.3781477	54200
50	2002	CQ11	0.9788526	0.42841032	2.45977	81.43849	272.76681	200.8820974	54200
51	2002	CD	0.9798418	0.17672247	6.87919	8.74939	331.55652	254.4324731	54200
52	1996	XZ12	0.9799732	0.49926135	5.65852	251.74134	55.81636	72.9065206	50427
53	1998	XN17	0.9818487	0.20968172	7.24543	85.99494	226.33101	308.1410001	54200
54	2006	AM4	0.9818736	0.649191	4.12836	123.29217	139.66996	15.6480498	54200
55	2001	FR85	0.9826891	0.02793117	5.24394	183.09543	233.53197	194.9601844	54200
56	2002	VX91	0.9846133	0.20141992	2.33173	216.8382	78.23357	274.8314614	54200
57	2000	OK8	0.9847498	0.22112979	9.98502	304.63783	166.12112	153.7352685	54200
58	2006	HR29	0.9852749	0.26346811	9.54116	232.80061	212.57562	87.7117748	54200
59	2006	SU217	0.9858344	0.17456034	2.64217	194.38475	38.23562	311.7733113	54200
60	2002	FT6	0.9882602	0.46273082	9.48968	188.6153	226.6889	122.5236582	54200
61	2006	QQ56	0.9883457	0.04505596	2.7989	161.62964	330.03731	70.0709228	54200
62	2004	EW	0.9894133	0.27979222	4.66406	343.43945	55.79267	142.6799405	54200
63	Khufu		0.9894148	0.46856366	9.91863	152.50644	54.98951	71.2602183	54200
64	2003	WP25	0.9907792	0.12120994	2.52245	42.1994	224.96577	297.5344418	54200
65	2005	CN61	0.9907967	0.06870481	9.52323	146.97133	248.33039	166.3675529	54200
66	2003	YN107	0.9927025	0.01393412	4.30184	264.84372	80.48505	217.7923676	54200
67	2005	UH6	1.0006444	0.63236792	2.64874	19.21603	200.24778	235.9241374	54200
68	2006	FV35	1.0010847	0.37754605	7.09998	179.61929	170.85813	226.9041918	54200
69	2000	EE104	1.0047041	0.29345413	5.24196	25.96806	280.93086	266.5400007	54200
70	2000	PH5	1.0051359	0.2301559	1.60139	278.41759	278.53077	327.3858309	54200
71	2001	GO2	1.0064333	0.16803052	4.61512	193.60694	265.29267	63.1828205	54200
72	2001	XX4	1.0065836	0.55673375	0.84697	127.00806	186.85712	159.7164678	54200
73	2005	TC51	1.0075124	0.3055468	5.66968	199.48658	288.09442	97.8673657	54200
74	1999	JV6	1.0075874	0.31111687	5.31393	124.62191	235.56835	198.4820189	54200
75	2000	QX69	1.0104975	0.27149424	4.58184	150.52076	73.62628	265.8465193	54200
76	2006	JY26	1.011128	0.08427146	1.42373	45.04477	277.02812	241.3233221	54200
77	2005	WK4	1.0116893	0.23719672	9.8329	138.33302	74.39146	357.2955237	54200
78	2002	PN	1.014505	0.0689143	9.14377	309.5259	107.45923	113.1536664	54200
79	2005	CN	1.0160246	0.18501785	2.31341	308.83895	321.30873	287.5129436	54200
80	2000	AG6	1.0176695	0.1899321	2.43499	283.11721	276.29949	318.9700596	54200
81	2002	TY59	1.0191848	0.23375184	6.6107	9.84962	259.11368	214.0602972	54200
82	2001	FC58	1.0201136	0.34316691	6.76713	174.76709	261.06877	26.774746	54200
83	Asclepius		1.0223202	0.35703111	4.9126	180.39166	255.1881	230.9053982	54200
84	1998	MW5	1.0228657	0.36267778	6.28735	80.47794	26.63877	320.6303075	54200
85	2006	BZ147	1.023656	0.09861161	1.40819	140.15053	95.17529	318.8393785	54200
86	2006	EK53	1.0251033	0.51704999	2.2214	5.21615	41.17404	67.9974415	53808
87	2005	BG28	1.0257488	0.22716044	6.13219	313.53121	80.85145	108.3498278	54200
88	1991	VG	1.0268385	0.04918621	1.44562	73.9738	24.50924	245.9824779	54200
89	2006	BJ55	1.0270842	0.1282408	5.94186	307.70442	288.23104	323.2412364	54200
90	2006	DQ14	1.0276275	0.05302857	6.29663	155.36066	292.38941	89.799583	54200

91	2006	FW33	1.0302366	0.80327705	8.33808	13.49756	349.3455	183.7475318	53821
92	2002	RS129	1.0311837	0.32895355	8.44276	339.00692	246.98177	220.7092649	54200
93	2001	CC21	1.0325665	0.21931701	4.80877	75.60574	179.29542	231.6344488	54200
94	2000	AF205	1.0339259	0.27673625	2.40834	220.16324	127.26933	54.1327783	54200
95	2001	CB21	1.034767	0.33365146	7.9034	353.8601	271.66177	240.5059958	54200
96	2001	GP2	1.0378059	0.07394752	1.27901	196.88525	111.25934	142.9922893	54200
97	1991	JW	1.0384084	0.11835385	8.72093	54.03708	301.86269	255.3126581	54200
98	2001	AD2	1.0393562	0.65972685	1.6545	211.34163	111.06615	43.7195481	54200
99	2006	SG7	1.0427851	0.56104837	4.76	2.33453	133.25344	111.879932	54200
100	2006	SY5	1.0432951	0.15244708	7.56343	336.0463	175.54676	40.3103043	54200
101	2001	TX1	1.047051	0.48246959	2.79874	159.32271	354.0293	321.7376165	54200
102	2001	RB12	1.051899	0.3813088	6.61497	333.31433	141.59767	343.3547373	54200
103	2003	MM	1.0531696	0.25621042	8.54111	127.72944	19.74445	279.2228606	54200
104	1996	FG3	1.0542884	0.34982316	1.9903	299.88579	23.91208	324.6499271	54200
105	2005	UG5	1.0554579	0.18966714	2.86666	35.78771	112.03922	24.9772368	54200
106	2002	JD9	1.0556339	0.44049688	6.62426	208.47676	138.83217	292.4444528	52404
107	2005	XY4	1.0563008	0.59843265	1.90691	163.16224	143.29953	139.3874036	54200
108	2005	LU3	1.0570624	0.30829536	5.58047	80.77844	71.77661	317.1981038	54200
109	1999	CG9	1.0606143	0.06251501	5.15795	138.8473	315.50051	211.2047726	54200
110	2005	YK	1.0611429	0.30763539	5.62187	269.6859	80.38776	131.7596963	54200
111	2001	QJ142	1.0622514	0.08631514	3.10602	184.48645	63.83549	124.4464309	54200
112	Castalia		1.0631546	0.48332785	8.88727	325.66257	121.30415	330.3741366	54200
113	2004	GD	1.064401	0.30749889	6.22219	26.72785	280.99671	191.2173454	54200
114	2006	HE2	1.0646155	0.15656928	1.17974	200.48697	90.08023	255.2455945	54200
115	2003	JC13	1.0659747	0.31508073	8.50584	205.9377	171.89361	82.4081951	54200
116	2002	JE9	1.0677456	0.41671603	8.82762	200.16103	255.35021	242.9946642	54200
117	2002	AW	1.0697566	0.2561265	0.56907	162.77042	118.2922	107.2813785	54200
118	2002	BF25	1.0741513	0.2221592	6.23395	306.10315	77.43594	319.7553187	54200
119	2002	XQ40	1.0744728	0.35058905	2.17913	270.64565	72.74243	17.1024107	54200
120	1997	XR2	1.0768851	0.20119085	7.1731	250.83926	84.55575	204.5497089	54200
121	2005	WC	1.0776112	0.4432993	1.74605	31.86884	133.55751	30.3343547	54200
122	2001	QE71	1.0776821	0.15846392	3.03659	148.60211	96.3364	78.0191173	54200
123	Bacchus		1.0780803	0.34953607	9.43429	33.1723	55.20031	7.2648785	54200
124	2006	GU2	1.080348	0.25631038	3.38025	197.18848	266.16714	26.9925859	54200
125	2006	CK	1.0826863	0.21429682	5.17308	310.98309	264.54637	317.653771	54200
126	2005	EY95	1.0834122	0.53844388	3.1694	73.09689	341.91012	4.8238963	54200
127	2001	TE2	1.0836375	0.1969092	7.60965	171.3078	35.69412	127.4288695	54200
128	2006	LH	1.0847903	0.31604449	7.81951	95.33561	264.62857	290.2705427	53890
129	2004	JN1	1.0853701	0.17556003	1.49675	144.04411	1.92689	270.3576509	54200
130	1998	SH36	1.0878859	0.57093192	2.12941	218.1013	278.56258	120.649136	54200
131	2005	CD69	1.0884135	0.18756634	2.78191	336.61216	264.4367	246.4800692	54200
132	2001	WT1	1.0886769	0.39699185	7.1582	74.03236	180.45066	53.3923001	54200
133	1998	FH12	1.0913606	0.53979829	3.55846	108.69104	284.3919	63.1123787	54200
134	2001	FD58	1.0920485	0.57527056	6.5017	341.3105	45.86707	222.15772	54200
135	2006	AN	1.0934731	0.21999476	7.4039	277.72573	273.42998	333.2696155	54200
136	1997	YM9	1.0953326	0.10361643	7.84208	94.82634	51.60205	359.0655644	54200
137	2003	UX34	1.0953726	0.61584396	2.56686	4.68824	218.14782	137.4209641	54200
138	2006	CT	1.0975637	0.23090015	2.73548	285.70653	82.61744	121.0512446	54200
139	1999	SH10	1.0976168	0.12987539	9.55702	178.64789	118.75966	252.1744405	54200
140	2004	FM32	1.0984144	0.16200779	3.7612	184.51438	298.27291	277.7685811	54200

REFERENCES

[1] Rayman, M.D., Williams, S. N., “Design of the First Interplanetary Solar Electric Propulsion Mission,” Journal of Spacecraft and Rockets, Vol. 39, No. 4, July-August 2002, pp. 589-595.

[2] Schoenmaekers, J., Horas, D., Pulido, J.A., “SMART-1: With Solar Electric Propulsion to the Moon,” 16th International Symposium on Space Flight Dynamics, Pasadena, CA, 2001.

[3] Uesugi, K.T., “Space Engineering Spacecraft (MUSES) Program in ISAS Featuring Its Latest Mission ‘HAYABUSA’,” Proceedings of International Conference on Recent Advances in Space Technologies, 20-22 November 2003, pp. 464-471.

[4] Rayman, M.D., Fraschetti, T.C., Raymond, C.A., Russell, C.T., “Dawn: A mission in development and exploration of main belt asteroids Ceres and Vesta,” Acta Astronautica, Vol. 58, 2006, pp. 605-616.

[5] Williams, S.N., Coverstone-Carroll, V., “Benefits of Solar Electric Propulsion for the Next Generation of Planetary Exploration Missions,” The Journal of the Astronautical Sciences, Vol. 45, No. 2, April-June 1997, pp. 143-159.

[6] Cheng, A.F., “Near Earth Asteroid Rendezvous: Mission Overview,” Space Sciences Reviews, Vol. 82, No. 1-2, October 1997, pp. 3-29.

[7] Izzo, D., “1st ACT global trajectory optimisation competition: Problem description and summary of the results,” Acta Astronautica, Vol. 61, Issue 9, November 2007, pp. 731-734.

[8] “JPL Announces 2nd Global Trajectory Optimisation Competition (GTOC2),” Advanced Concepts Team, European Space Agency, [http://www.esa.int/gsp/ACT/newsroom/NewsArchive/New14_Oct07_GTOC2.htm. Accessed 25 September 2009].

[9] “Problem Description for the 2nd Global Trajectory Optimisation Competition,” [http://www.polito.it/eventi/gtoc3/gtoc2_problem.pdf Accessed 25 September 2009].

[10] "3rd Global Trajectory Optimisation Competition," Politecnico di Torino, [http://www2.polito.it/eventi/gtoc3/. Accessed 6 November 2009].

[11] "4th Global Trajectory Optimisation Competition," Centre National D'Etudes Spatiales, [http://cct.cnes.fr/cct02/gtoc4/index.htm. Accessed 6 November 2009].

[12] Sims, J.A., Flanagan, S.N., "Preliminary Design of Low-Thrust Interplanetary Mission," AAS 99-338, AAS/AIAA Astrodynamics Specialist Conference, Girdwood, Alaska, 16-19 August 1999.

[13] Sims, J.A., Finlayson, P.A., Rinderle, E.A., Vavrina, M.A., Kowalkowski, T.D., "Implementation of a Low-Thrust Trajectory Optimization Algorithm for Preliminary Design", AIAA 2006-6746, AIAA/AAS Astrodynamics Specialist Conference, 21-24 August 2006, Keystone, CO.

[14] Ranieri, C.L., Ocampo, C.A., "Optimization of Roundtrip, Time-Constrained Finite Burn Trajectories via an Indirect Method," Journal of Guidance, Control, and Dynamics, Vol. 28, No. 2, March-April 2005, pp. 306-314.

[15] Kluever, C.A., "Optimal Low-Thrust Interplanetary Trajectories by Direct Method Techniques," Journal of the Astronautical Sciences, Vol. 45, No. 3, July-Sept. 1997, pp. 247-262.

[16] Dewell, L., Menon, P., "Low-Thrust Orbit Transfer Optimization Using Genetic Search," AIAA Guidance, Navigation, and Control Conference and Exhibit, AIAA, Reston, VA, 1999, pp. 1109-1111.

[17] Betts, J.T., "Survey of Numerical Methods for Trajectory Optimization", Journal of Guidance, Control, and Dynamics, Vol. 21, No. 2, March-April 1998 pp. 193-207.

[18] Betts, J.T., "Optimal Interplanetary Orbit Transfers by Direct Transcription," Journal of the Astronautical Sciences, Vol. 42, No. 3, July-Sept. 1994, pp. 247-268.

[19] Coverstone-Carroll, V., Williams, S.N., "Optimal Low-Thrust Trajectories using Differential Inclusion Concept," Journal of the Astronautical Sciences, Vol. 42, No. 4, Oct.-Dec. 1994, pp. 379-393.

[20] Hargraves, C. R., and Paris, S.W., "Direct Trajectory Optimization Using Nonlinear Programming and Collocation," Journal of Guidance, Control, and Dynamics, Vol. 10, No. 4, 1987, pp. 338–342.

[21] Tang, S., and Conway, B. A., "Optimization of Low-Thrust Interplanetary Trajectories Using Collocation and Nonlinear Programming," Journal of Guidance, Control, and Dynamics, Vol. 18, No. 3, 1995, pp. 599–604.

[22] Lantoine, G., Russell, R.P., "A Hybrid Differential Dynamic Programming Algorithm for Robust Low-Thrust Optimization," AIAA 2008-6615, AIAA/AAS Astrodynamics Specialist Conference and Exhibit, Honolulu, Hawaii, 18-21 August 2008.

[23] Lantoine, G., Russell, R.P., "A Fast Second-Order Algorithm For Preliminary Design of Low-Thrust Trajectories," IAC-08-C1.2.5, 59th International Astronautical Congress, Glasgow, Scotland, 29 September – 3 October 2008.

[24] Lantoine, G., Russell, R.P., "The Stark Model: An Exact, Closed-Form Approach to Low-Thrust Trajectory Optimization," 21st International Symposium on Space Flight Dynamics, Toulouse, France, 28 September – 2 October 2009.

[25] Gao, Y., Kluever, C.A., "Low-Thrust Interplanetary Orbit Transfers Using Hybrid Trajectory Optimization Method with Multiple Shooting", AIAA 2004-5088, AIAA/AAS Astrodynamics Specialist Conference and Exhibit, August 2004, Providence, RI.

[26] Ranieri, C.L., Ocampo, C.A., "Optimizing Finite-Burn, Round-Trip Trajectories with Isp Constraints and Mass Discontinuities" Journal of Guidance, Control, and Dynamics, Vol. 28, No. 4, July-August 2005, pp. 775-781.

[27] Russell, R., "Primer Vector Theory Applied to Global Low-Thrust Trade Studies," AAS 06-156, 16th AAS/AIAA Space Flight Mechanics Conference, Tampa, Florida, 22-26 January 2006.

[28] Gill, P.E., Murray, W., Saunders, W.A., "User's Guide for SNOPT Version 7: A FORTRAN Package for Large-Scale Nonlinear Programming," University of California, San Diego, 16 June 2008.

[29] McConaghy, T.T., Debban, T.J., Petropoulos, E., Longuski, M., "Design and Optimization of Low-Thrust Trajectories with Gravity Assists", Journal of Spacecraft and Rockets, Vol. 40, No. 3, May-June 2003.

[30] Okutsu, M., Yam, C.H., Longuski, J.M., "Low-Thrust Trajectories to Jupiter via Gravity Assists from Venus, Earth, and Mars", AIAA 2006-6745, AIAA/AAS Astrodynamics Specialist Conference, 21-24 August 2006, Keystone, CO.

[31] Yam, C. H. and Longuski, J., M., "Optimization of Low-Thrust Gravity-Assist Trajectories with a Reduced Parameterization of the Thrust Vector," AAS/AIAA Astrodynamics Specialist Conference and Exhibit, AAS Paper 05-375, 1995, Lake Tahoe, CA.

[32] Yam, C.H., Longuski, J.M., "Reduced Parameterization for Optimization of Low-Thrust Gravity-Assist Trajectories: Case Studies", AIAA 2006-6744, AIAA/AAS Astrodynamics Specialist Conference, 21-24 August 2006, Keystone, CO.

[33] Petropoulos, A.E., Longuski, J.M., "A Shape-Based Algorithm for the Automated Design of Low-Thrust, Gravity-Assist Trajectories," AAS 01-467, AAS/AIAA Astrodynamics Specialist Conference, Quebec City, Quebec, Canada, 30 July – 2 August 2001.

[34] Petropoulos, A.E., Longuski, J.M., Vinh, N.X., "Shape-Based Analytic Representations of Low-Thrust Trajectories for Gravity-Assist Applications," AAS/AIAA Astrodynamics Specialist Conference, AAS Paper 99-337, Girdwood, AL, August 1999.

[35] Petropoulos, A.E., Longuski, J.M, "Automated Design of Low-Thrust Gravity-Assist Trajectories," AIAA/AAS Astrodynamics Specialist Conference, AIAA 2000-4033, Denver, CO, 14-17 August 2000.

[36] "1st ACT Global Trajectory Optimisation Competition," European Space Agency. [http://www.esa.int/gsp/ACT/mission_analysis/globaloptimisationcompetition.htm. Accessed 27 January 2007].

[37] Izzo, D., "Problem Description – 1st ACT Competition on Global Trajectory Optimisation," European Space Agency, November 2005. [http://www.esa.int/gsp/ACT/doc/ACT-4100-DI-The%20Problem_V4.pdf. Accessed 27 January 2007].

[38] Petropoulos, A., Kowalkowski, T., Parcher, D., Finlayson, P., Rinderle, E., Vavrina, M., Sims, J., Russell, R., Lam, T., Williams, P., Whiffen, G., Strange, N., Johannesen, J., Yen, C.W., Sauer, C., Lee, S., Williams, S., "Response to the First ACT Competition on Global Trajectory Optimisation," NASA Jet Propulsion Lab, Pasadena, CA, 5 December 2005. [http://trs-new.jpl.nasa.gov/dspace/handle/2014/38539?mode=simple. Accessed 27 January 2007].

[39] Polsgrove, T., Kos, L, Hopkins, R., Crane, T., "Comparison of Performance Predictions for New Low-Thrust Trajectory Tools", AIAA 2006-6742, AIAA/AAS Astrodynamics Specialist Conference, 21-24 August 2006, Keystone, CO.

[40] Kos, L, Polsgrove, T., Hopkins, R., Thomas, D., "Overview of the Development for a Suite of Low-Thrust Trajectory Analysis Tools", AIAA 2006-6743, AIAA/AAS Astrodynamics Specialist Conference, 21-24 August 2006, Keystone, CO.

[41] Ocampo, C., "COPERNICUS: A Trajectory Design and Optimization System," University of Texas, Austin, TX. [www.ieec.fcr.es/libpoint/abstracts/ocampo.pdf. Accessed 25 January 2007].

[42] Whiffen, G.J., "Mystic: Implementation of the Static Dynamic Control Algorithm for High-Fidelity, Low-Thrust Trajectory Design," AIAA/AAS Astrodynamics Specialist Conference, 21-24 August 2006, Keystone, CO.

[43] Paris, S.W., Riehl, J.P., Sjauw, W.K., "Enhanced Procedures for Direct Trajectory Optimization Using Nonlinear Programming and Implicit Integration," AIAA/AAS Astrodynamics Specialist Conference and Exhibit, 21-24 August 2006, Keysone, CO.

[44] Riehl, J.P., Paris, S.W., Sjauw, W.K., "Comparison of Implicit Integration Methods for Solving Aerospace Trajectory Optimization Problems," AIAA/AAS Astrodynamics Specialist Conference and Exhibit, 21-24 August 2006, Keysone, CO.

[45] Back, T., Evolutionary Algorithms in Theory and Practice, Oxford University Press, New York, NY, 1996.

[46] Gen, M., Cheng, R., Genetic Algorithms and Engineering Design, John Wiley & Sons, Inc., New York, NY, 1997.

[47] Man, K.F., Tang, K.S., Kwong, S., Genetic Algorithms, Springer, London, England, 1999.

[48] Schaefer, Robert, Foundation of Global Genetic Optimization, Springer, Berlin, Germany, 2007.

[49] Goldberg, D.E., Richardson, J., "Genetic Algorithms with Sharing for Multimodal Function Optimization," Proceedings of the 2nd International Conference on Genetic Algorithms, Lawrence Erlbaum Associates: Hillsdale, NJ, 1987.

[50] Deb, K., Multi-Objective Optimization using Evolutionary Algorithms, John Wiley & Sons: Chichester, Great Britain, 2001.

[51] Deb., K., Pratap, A., Agarwal, S., Meyarivan, T., "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II," Transactions on Evolutionary Computation, Vol. 6, No. 2, April 2002, pp. 182-197.

[52] Bessette, C.R., Spencer, D.B., "Identifying Optimal Interplanetary Trajectories through a Genetic Approach", AIAA 2006-6306, AIAA/AAS Astrodynamics Specialist Conference, 21-24 August 2006, Keystone, CO.

[53] Crain, T. P., Bishop, R.H., Fowler, W., "Interplanetary Flyby Mission Optimization Using a Hybrid Global-Local Search Method," Journal of Spacecraft and Rockets, Vol. 37, No. 4, July-Aug. 2000, pp. 468-474.

[54] Gage, P.J., Braun, R.D., Kroo, I.M., "Interplanetary Trajectory Optimization Using a Genetic Algorithm", The Journal of the Astronautical Sciences, Vol. 43, No. 1, January-March 1995, pp. 59-75.

[55] Vasile, M., Pascale, P. De, "Preliminary Design of Multiple Gravity-Assist Trajectories", Journal of Spacecraft and Rockets, Vol. 43, No. 4, July-August 2006, pp. 794-805.

[56] Dewell, L., Menon, P., "Low-Thrust Orbit Transfer Optimization Using Genetic Search," AIAA Guidance, Navigation, and Control Conference and Exhibit, AIAA, Reston, VA, 1999, pp. 1109-1111.

[57] Wuerl, A., Crain, T., Braden, E., "Genetic Algorithm and Calculus of Variations-Based Trajectory Optimization Technique," Journal of Spacecraft and Rockets, Vol. 40, No. 6, Nov.-Dec. 2003, pp. 882-888.

[58] De Pascale, P., Vasile, M., Finzi, A.E., "A Tool for Preliminary Design of Low Thrust Gravity Assist Trajectories", Advances in the Astronautical Sciences, Vol. 119, Part III, 2004, pp. 2315-2334.

[59] Woo, B., Coverstone, V.L., "Low-Thrust Trajectory Optimization Procedure for Gravity-Assist, Outer-Planet Missions", Journal of Spacecraft and Rockets, Vol. 43, No. 1, January-February 2006, pp. 121-129.

[60] Morimoto, M., Yamakawa, H., Yoshikawa, M., Abe, M., Yano, H., "Trajectory design of multiple asteroid sample return missions," Advances in Space Research, Vol. 34, Issue 11, 2004, pp. 2281-2285.

[61] Vavrina, Matthew A., Howell, Kathleen C., "Global Low Thrust Trajectory Optimization through Hybridization of a Genetic Algorithm and a Direct Method," AIAA/AAS Astrodynamics Specialist Conference and Exhibit, Honolulu, HI, 18-21 August 2008.

[62] Dachwald, B., "Optimization of Interplanetary Solar Sailcraft Trajectories Using Evolutionary Neurocontrol," Journal of Guidance, Control, and Dynamics, Vol. 27, No. 1, January-February 2004, pp. 66-72.

[63] Dachwald, B., "Evolutionary Neurocontrol: A Smart Method for Global Optimization of Low-Thrust Trajectories," AIAA/AAS Astrodynamics Specialist Conference and Exhibit, Providence, RI, 16-19 August 2004.

[64] Carnelli, I., Dachwald, B., Vasile, M., Seboldt, W., Finzi, A.E., "Low-Thrust Gravity Assist Trajectory Optimization Using Evolutionary Neurocontrollers," Advances in the Astronautical Sciences, Vol. 123, Part III, 2005, pp. 1911-1928.

[65] Nemhauser, G.L., Wolsey, L.A, Integer and Combinatorial Optimization, John Wiley & Sons, 1999.

[66] Lawler, E.L., Lenstra, J.K., Ronnooy Kan, A.H.G., Shmoys, D.B., The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization, John Wiley & Sons: Chichester [West Sussex]; New York, 1985.

[67] Kreher, D.L., Stinson, D.R., Combinatorial Algorithms: Generation, Enumerations, and Search, CRC Press: Boca Raton, FL, 1999.

[68] Helvig, C.S., Robins, G., Zelikovsky, A., "The Moving-Target Traveling Salesman Problem," Journal of Algorithms, Volume 49, 2003, pp. 153-174.

[69] Laporte, G., Asef-Vaziri, A., Sriskandarajah, C., "Some Applications of the Generalized Travelling Salesman Problem," The Journal of the Operational Research Society, Vol. 47, No. 12, December 1996, pp. 1461-1467.

[70] Gutin, G., Punnen, A.P., The Traveling Salesman Problem and Its Variations, Springer, 2002.

[71] Stodgell, T. R., Spencer, D.B., "Satellite Rendezvous Tours Using Multiobjective Evolutionary Optimization," AAS 07-382, AAS/AIAA Astrodynamics Specialist Conference, Mackinac Island, MI, 19-23 August 2006.

[72] Wall, B.J., Conway, B.A., "Developing a Systematic Approach to the Use of Genetic Algorithms for the Solution of Optimal Spacecraft Trajectory Problems," AAS 07-161, AAS/AIAA Space Flight Mechanics Meeting, Sedona, AZ, 28 January – 1 February 2007.

[73] Vanderplaats, G.N., Numerical Optimization Techniques for Engineering Design, Vanderplaats Research & Development, Inc: Colorado Springs, CO, 2001.

[74] Shen, Tsiotras, "Optimal Two-Impulse Rendezvous Using Multiple Revolution Lambert Solutions," Journal of Guidance, Control, and Dynamics, Vol. 26, No. 1, January-February 2003, pp. 50-61.

[75] "Final Rankings and Brief Descriptions of the Returned Solutions and Methods Used for the 2nd Global Trajectory Optimisation Competition," [http://www2.polito.it/gtoc3/gtoc2_rankings.pdf. Accessed 25 September 2009].

[76] New Frontiers in the Solar System: An Integrated Exploration Strategy, National Research Council, Then National Academies Press, Washington D.C., 2003.

[77] Opening New Frontiers in Space: Choices for the Next New Frontiers Announcement of Opportunity, National Research Council, Then National Academies Press, Washington D.C., 2008.

[78] Cosmic Vision: Space Science for Europe 2015-2025, European Space Agency, ESA Publications Division, Noordwijk, The Netherlands, 2005.

[79] "Near Earth Object Program," National Aeronautics and Space Administration, [http://neo.jpl.nasa.gov/cgi-bin/neo_elem. Accessed 6 November 2009].

[80] "Seeking a Human Spaceflight Program Worth of a Great Nation: Review of U.S. Human Spaceflight Plans Committee," [http://images.spaceref.com/news/2009/396093main_HSF_Cmte_FinalReport.pdf. Accessed 22 October 2009].

VITA

Kristina Alemany was born in New York, NY, and spent her childhood in Briarcliff Manor, NY. She graduated from Briarcliff High School in 1999 as the valedictorian of her class. Kristina then went on to Princeton University to pursue a Bachelor of Science in Engineering in Mechanical and Aerospace Engineering and a certificate in Latin American Studies. She graduated magna cum laude in 2003. While at Princeton, Kristina captained the Princeton Women's Club Soccer team, which earned a berth in the NIRSA national championships during her senior season. Kristina also taught wilderness first aid classes to students training to become backpacking leaders in the university's Outdoor Action program.

The following fall, Kristina continued her studies at the Georgia Institute of Technology under the guidance of Dr. John Olds in the Space Systems Design Lab. She received a Master's Degree in Aerospace Engineering from Georgia Tech in 2005. In the fall of 2005, Kristina began working towards her PhD under the guidance of Dr. Robert Braun, also in the Space Systems Design Lab at Georgia Tech. During her time at Georgia Tech, Kristina worked on a diverse range of projects, including conceptual spacecraft and architecture design and optimization, launch vehicle trajectory optimization, conceptual design of EDL (entry, descent, and landing) systems, and low-thrust trajectory optimization. She also served as the teaching assistant for two graduate classes – Advanced Design Methods II and Orbital Mechanics. In addition to her studies, Kristina was the president of the Georgia Tech Women's Club Soccer Team, which placed 2nd at the 2007 Region II Tournament. Kristina spent her summers working at NASA Langley Research Center, Lockheed Martin Space Systems, and NASA Jet Propulsion Laboratory.

Kristina began work at the Aerospace Corporation in El Segundo, California in July 2009. She works in the Space Architecture Department under Inki Min. In her spare time, she enjoys traveling as well as pursuing a number of outdoor activities, including soccer, softball, hiking and backpacking, scuba diving, and skiing. She also enjoys spending time with her two dogs and her fiancé, Devin Kipp, who also received a Master's Degree in Aerospace Engineering from Georgia Tech in 2005 and is currently working at NASA Jet Propulsion Laboratory.