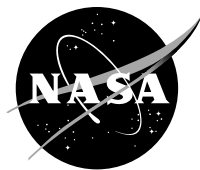# Direct Method Transcription For a Human-Class Translunar Injection Trajectory Optimization

Kevin E. Witzberger
Glenn Research Center, Cleveland, Ohio

Tom Zeiler
The University of Alabama, Tuscaloosa, Alabama

NASA STI Program . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NASA Aeronautics and Space Database and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include creating custom thesauri, building customized databases, organizing and publishing research results.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at *http://www.sti.nasa.gov*

- E-mail your question via the Internet to *help@ sti.nasa.gov*

- Fax your question to the NASA STI Help Desk at 443–757–5803

- Telephone the NASA STI Help Desk at 443–757–5802

- Write to:
  NASA Center for AeroSpace Information (CASI)
  7115 Standard Drive
  Hanover, MD 21076–1320

Direct Method Transcription For a Human-Class Translunar Injection Trajectory Optimization

Kevin E. Witzberger
Glenn Research Center, Cleveland, Ohio

Tom Zeiler
The University of Alabama, Tuscaloosa, Alabama

Acknowledgments

The authors would like to acknowledge Dr. Rajnish Sharma for his review and comments.

Direct Method Transcription For a Human-Class Translunar Injection Trajectory Optimization

Kevin E. Witzberger
National Aeronautics and Space Administration
Glenn Research Center
Cleveland, Ohio 44135

Tom Zeiler
The University of Alabama
Tuscaloosa, Alabama 35487

Abstract

This paper presents a new trajectory optimization software package developed in the framework of a low-to-high fidelity 3 degree-of-freedom (DOF)/6-DOF vehicle simulation program named Mission Analysis Simulation Tool in Fortran (MASTIF) and its application to a translunar trajectory optimization problem. The functionality of the developed optimization package is implemented as a new "mode" in generalized settings to make it applicable for a general trajectory optimization problem. In doing so, a direct optimization method using collocation is employed for solving the problem. Trajectory optimization problems in MASTIF are transcribed to a constrained nonlinear programming (NLP) problem and solved with SNOPT, a commercially available NLP solver. A detailed description of the optimization software developed is provided as well as the transcription specifics for the translunar injection (TLI) problem. The analysis includes a 3-DOF trajectory TLI optimization and a 3-DOF vehicle TLI simulation using closed-loop guidance.

Introduction

Ever since the National Aeronautics and Space Administration's (NASA's) Apollo program ended in the early 1970's, humans have not traveled beyond Earth orbit (BEO). Various issues have hampered BEO travel for humans including the fact that human space flight (HSF) is costly and requires continued commitment from each new administration. HSF is also risky, as evidenced in 2003 with the Space Shuttle's Columbia disaster. In response to the Columbia Accident Investigation Report (CAIB), in January 2004, President George W. Bush announced a new United States space policy—Vision For Space Exploration (VSE)(Refs. 1 and 2). The VSE established the goal of returning humans to the Moon by 2020 as preparation for a future human Mars mission. As a result of the VSE, NASA established the Constellation Program[1]. In order to return humans to the Moon, a good deal of research and trade studies was conducted in support of returning humans to the Moon. The focus of this paper is on the transfer trajectory from Earth to the Moon, i.e., the translunar injection (TLI) trajectory. The TLI problem is a trajectory optimization problem; deliver the most mass to the Moon while meeting mission constraints. This is not a new trajectory problem—it was solved during the Apollo program using indirect optimization methods.

The research presented in this paper leads to a solution of the TLI problem; however, a direct optimization method is employed to solve the problem, transcribing it to a nonlinear programming (NLP) problem that is solved with a commercial solver, SNOPT (Ref. 3). The robustness and flexibility offered by the direct method serve as the primary motivations of this research work as it is amenable to different types of trajectory optimizations and mission constraint requirements. Furthermore, the direct method

---

[1] The Constellation Program effectively ended during the time of this research.

applied to the problem is implemented using collocation techniques, similar to those described in Reference 4, in generalized settings in a 3 degree-of-freedom (DOF)/6-DOF vehicle simulation program named Mission Analysis Simulation Tool in Fortran (MASTIF) (Refs. 4 and 5). MASTIF development began in 2006 and was successfully used to independently verify certain Ares I trajectories (Ref. 6).

Typically, vehicle simulation and trajectory optimization (hereafter referred to as just optimization) software programs are separate programs. Combining optimization and 3-DOF/6-DOF vehicle simulation software into the same tool makes the software tool rare. The authors' are aware of just one other software program with similar capabilities, Program to Optimize Simulated Trajectories (POST) (Ref. 7). This combination addresses present issues that arise from having separate programs and offers benefits as well. Frequently, the outputs of the optimization program are inputs into the vehicle simulation program. Also, the required modeling inputs (initial conditions, propulsion, aerodynamics, etc.) can differ between different programs. In usual practice, the best an analyst can do is to ensure equivalent modeling between different tools, which is a potential source of errors. Trajectories are sometimes optimized in a piece-wise fashion because establishing a first guess for a complete trajectory can be difficult. This issue can sometimes be remedied when the same tool has dual capabilities such as vehicle simulation and optimization (especially when closed-loop guidance is available for use in the software for vehicle simulations). For example, the entire trajectory can be simulated to achieve an initial trajectory guess (as input) for the (complete) trajectory optimization and not just a certain portion of the trajectory.

Generally, trajectory optimization tools are limited to 3-DOF modeling of the vehicle. Extending the vehicle model to 6-DOF is not readily attainable because of an additional burden of significant software development or redevelopment. However, when implemented in the same tool as a 6-DOF vehicle simulation, extending the trajectory optimization software to 6-DOF is expected to be readily attainable because the built-in 6-DOF infrastructure (e.g., equations of motion) already exists. Another major benefit of combining the vehicle simulation and trajectory optimization software in a single software package is that code maintenance is required only on one software package instead of two. The list below summarizes the motivations for combining vehicle simulation and trajectory optimization software.

- Reduce modeling errors
- Save time
- Easier/better initial guesses
- Readily extendable to 6-DOF optimization
- Fewer programs to maintain

As stated above, the research presented in this paper has two main parts. The first part is the description of a novel application of a direct method within the framework of 3/6-DOF vehicle simulation software tool and its application to the TLI trajectory optimization problem. The vehicle configuration includes an Earth Transfer Stage (ETS), an orbiter, lander, and a bipropellant chemical engine. The modeled trajectory initiates from a circular low Earth orbit (LEO) with main engine ignition; the ETS and propellant tanks are jettisoned an hour after ignition. The trajectory ends with the spacecraft flying over the north pole of the Moon. The second part of this paper highlights the dual capability of the software tool by utilizing the results of the TLI optimization (the open-loop solution) in the form certain orbital states at main engine cutoff (MECO) as input data into a vehicle simulation (the other "mode" in MASTIF) that assesses a potential candidate for the closed-loop guidance algorithm. The closed-loop guidance algorithm assessed is a variant of the Space Shuttle's guidance named Powered Explicit Guidance (PEG); the fact that it is flight-proven makes it a good candidate (Ref. 8). Finally, the results from the open-loop and closed-loop solution are compared to show approximate agreement and efficacy of MASTIF's capabilities.

Nomenclature

I_{sp}	specific impulse
T_{vac}	vacuum thrust
MR	mixture ratio
m_0	initial mass
m_{ETS}	ETS mass
m_{LOX}	LOX oxidizer tank mass
m_{LH2}	LH2 fuel tank mass
m_{FPR}	flight performance reserve propellant tank mass
m_{RCS}	RCS propellant tank mass
$g_{Earth,SL}$	magnitude of Earth's gravity vector at sea level (9.80665 m/s)
GM	Moon's gravitational constant (4902.801076 km^3/sec^2)

Translunar Injection Optimization Problem

The Earth to Moon transfer problem to be solved consists of the following sequence of events and modeling assumptions. Starting in circular 29° LEO, at the epoch of January 1, 2018, (midnight), perform a main engine burn to place the spacecraft on the correct 3-day transfer ellipse while maximizing the mass at MECO. The start-up and shut-down transients of the engine are ignored, but can be included in the future. The atmospheric drag is also ignored for now. Approximately 1 hr after the start of the burn, the spacecraft jettisons the ETS. Flight performance reserve (FPR) propellant left over from the ascent to orbit and reaction control system (RCS) propellant (carried in two separate propellant tanks) are also jettisoned with the ETS. The propellant in these two tanks is not available for use in the 3-DOF optimization; it reserved for 6-DOF simulations. Propellant remaining in the tanks from the main engine burn is carried along to the Moon, but could easily be jettisoned, as modeling is refined for other future needs.

The spacecraft is to arrive at the Moon flying over the North Pole at a 100 km pericynthion altitude. The relatively small ΔV required to capture the spacecraft into a lunar orbit is not currently modeled. Table 1 lists the LOX/LH2 bipropellant engine's specific impulse, I_{sp}, vacuum thrust, T_{vac}, and mixture ratio, MR. The spacecraft's initial mass, m_0, includes the mass of the ETS, m_{ETS}, and the mass of the filled fuel and oxidizer tanks, m_{LH2} and m_{LOX}, respectively. Two additional tanks store propellant set aside for the reaction control system (RCS), m_{RCS}, and flight performance reserve (FPR), m_{FPR}; no propellant is drawn from these tanks for the TLI trajectory optimization. Partitioning the propellant this way makes it convenient for future 6-DOF Monte-Carlo analyses, if required.

TABLE 1.—PROPULSION AND MASS PROPERTIES[a]

I_{sp} (sec)	T_{vac} (kN)	MR	m_0 (kg)	m_{ETS} (kg)	m_{LOX} (kg)	m_{LH2} (kg)	m_{FPR} (kg)	m_{RCS} (kg)
448	1,070.6	5.5	187,901	25,463	81,887	14,889	1,449	342

[a]See Nomenclature

Gravitational effects and first order harmonics (J2) from the Earth, Sun, and Moon are modeled. The values for the physical constants for these three bodies came from a Jet Propulsion Laboratory (JPL) lunar modeling document (Ref. 9). Ephemeris data (Earth, Moon, and Sun time-dependent positions) came from the JPL's SPICE software.[2]

---

[2] http://naif.jpl.nasa.gov/naif/

Discretization of Translunar Injection Problem

Two vehicles and three (total) phases are utilized to model the problem. The first vehicle includes the spacecraft and the ETS and is modeled with two phases. The first phase is the TLI burn and the second phase is a nearly 1 hr coast phase. The second vehicle does not include the ETS and the FPR and RCS propellant tanks. Its primary role is to easily facilitate the jettison of the ETS and two propellant tanks within the software design of MASTIF. The second vehicle has one long coast phase. The first two phases are collocation phases (described next) whereas the last phase is an explicitly integrated phase using a high order Runge-Kutta integrator (Ref. 10).

The equations of motion in each collocation phase are discretized by segments that contain nodes. Dimensional time is transformed by a Legendre-Gauss-Lobatto (LGL) distribution method that results in placing the nodes on a non-dimensional time grid spanning from –1 on the left segment boundary to +1 on the right segment boundary (Ref. 11). Each segment contains at least three nodes to enable interpolated state and control values to be obtained at every other node. There is no maximum limit on the number of nodes; however, it must be an odd number due to the mathematical formulation adopted. The number of nodes (per segment) corresponds directly to the degree of the interpolating polynomial. The expressions for the polynomials are provided in the Appendix; an efficient numerical implementation is the ACM algorithm 211 (Refs. 12 and 13).

Figure 1 shows a schematic of the vehicle and phase layout (with minimal number of nodes and a single segment per phase for clarity) and Figure 2 depicts the double nodes that are utilized at all phase, k, boundaries to enable discontinuities in the vehicle states, $\bar{s}$, and/or controls, $\bar{u}$, and will be discussed in further detail in the next section.
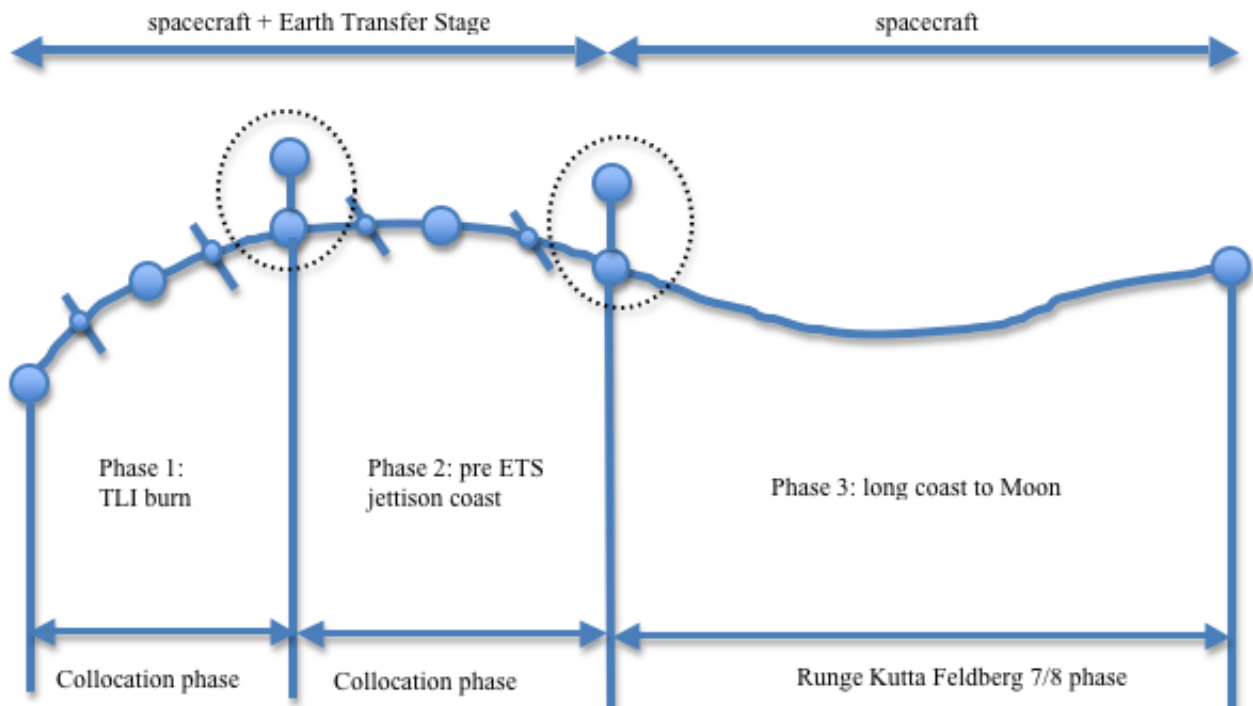

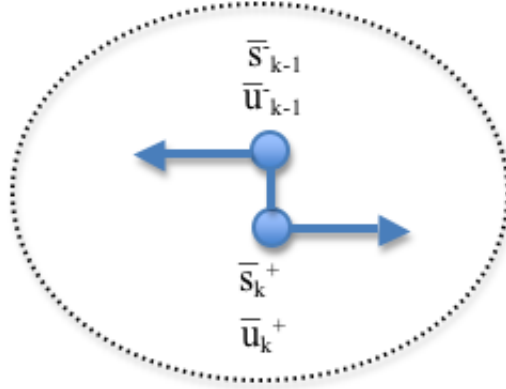Figure 1.—Vehicle and phase schematic.

Figure 2.—Double nodes at phase boundaries.

The discretization utilizes a global numbering scheme for the nodes, i, that results in odd-numbered and even-numbered nodes starting with the first node ($i = 1$) contained in the first segment within the first phase. The global numbering scheme is an implementation design; all interpolations are local (segment level). Potential solutions to the states and controls are provided by SNOPT at each segment's odd-numbered nodes. These states are used to calculate the true state derivatives at these odd-numbered nodes. At each segment's even-numbered nodes, Hermite polynomial interpolations (using values of states and derivatives at the segment's odd-numbered nodes) are used to estimate the states, which can then be used to estimate the state derivatives (Ref. 13). The difference between the estimated state derivatives and the true state derivates at each segment's even-numbered nodes are known as defects, Δ. These defects are nonlinear constraints.

In general, there exist different discretization schemes, expecting the same results within the range of accuracy that might vary from user to user. Discretization is advantageous because it enables the user to control solution accuracy, which is especially important during the initial attempts at a solution when less accuracy is required. Permitting user control over the number of nodes per segment and the number of segments per phase is analogous to controlling the order of integration method and integration step-size, respectively, in an explicit integration. This numerical control essentially allows accuracy to vary from phase to phase according to the modeled dynamics of the problem. In this paper, the discretization process is iterative and to accurately represent the trajectory, there must be enough grid points (placement is also important) considered for the applied algorithm.

Problem Formulation

Now, a general optimal control problem with prescribed dynamical system states, $\bar{x}(t)$, and a control input, $\bar{u}(t)$, minimizes a cost function

$$J = J(\bar{x}(t),\bar{u}(t),t), \quad \bar{x} \in \Re^{n \times 1}, \bar{u} \in \Re^{m \times 1} \tag{1}$$

subject to the system's equations of motion,

$$\dot{\bar{x}} = f(\bar{x}(t),\bar{u}(t),t) \tag{2}$$

and any initial constraints

$$\bar{\chi}(\bar{x}(t_0),t_0) = \bar{0}, \quad \bar{\chi} \in \Re^{p \times 1} \tag{3}$$

terminal constraints

$$\overline{\Psi}(\overline{x}(t_f), t_f) = \overline{0}, \quad \overline{\Psi} \in \Re^{q \times 1} \tag{4}$$

and path constraints

$$\overline{c}(\overline{x}(t), \overline{u}(t), t) \leq \overline{0}, \quad \overline{c} \in \Re^{r \times 1} \tag{5}$$

where $t_0 \leq t \leq t_f$ and t_0 and t_f are initial and final times, respectively.

In MASTIF, the optimal control problem is transcribed to a constrained NLP optimization problem of the form:

$$\min J(\overline{x}) \quad \text{s.t.} \quad \overline{L} \leq \begin{pmatrix} \overline{x} \\ \overline{F}(\overline{x}) \\ \overline{G}(\overline{x}) \end{pmatrix} \leq \overline{U} \tag{6}$$

and solved by using SNOPT, where $\overline{L}$ and $\overline{U}$ are the lower and upper bounds, respectively. Note that in this formulation $\overline{x}$ is an array containing all the independent variables *including the controls,* $\overline{u}$. $\overline{F}(\overline{x})$ is an input array into SNOPT containing linear and nonlinear constraints and the objective function, $J(\overline{x})$. Lastly, $\overline{G}(\overline{x})$ is an (optional) input array into SNOPT containing the first order partial derivatives of $\overline{F}(\overline{x})$. Currently, derivatives are not calculated by MASTIF; SNOPT uses finite differencing as an estimate of the partial derivatives. The arrays given in Equation (6) are defined below beginning with the equations of motion.

Equations of Motion

The equations of motion are modeled in EME2000 Cartesian coordinates (denoted with a 'j2k' subscript). The velocity vector is

$$\dot{\overline{r}}_{j2k} = \overline{v}_{j2k}. \tag{7}$$

With thrust being the only (modeled) non-gravitational force exerted on the vehicle, the acceleration term becomes

$$\dot{\overline{v}}_{j2k} = \frac{T_{vac}}{m} \hat{m}_{j2k} + \overline{g}_{j2k} \tag{8}$$

where T_{vac} is the vacuum thrust (atmospheric effects are neglected) given in Table 1, $\hat{\mu}_{j2k}$ is the direction of thrust (unit magnitude) expressed in EME2000 coordinates. The mass, m, is given by Equation (9) below and is a function of a table lookup of total remaining propellant in all the propellant tanks

$$m = \begin{cases} f\big(\text{table}[m_{\text{LOX}} + m_{\text{LH2}} + m_{FPR} + m_{RCS}]\big) + m_{ETS} & \text{if phase} \leq 2 \\ f\big(\text{table}[m_{\text{LOX}} + m_{\text{LH2}}]\big) & \text{if phase} = 3. \end{cases} \tag{9}$$

The last term in Equation (8), $\overline{g}_{j2k}$, is the gravitational acceleration due to the Earth, Sun, and Moon:

$$\overline{g}_{j2k} = \overline{g}_{j2k,\text{Earth}} + \overline{g}_{j2k,\text{Sun}} + \overline{g}_{j2k,\text{Moon}}. \tag{10}$$

To complete the equations of motion, the mass flow rates of each propellant tank are

$$\bar{\dot{m}}_{tnk} = \begin{cases} \begin{bmatrix} \dot{m}_{LOX} \\ \dot{m}_{LH2} \\ \dot{m}_{FPR} \\ \dot{m}_{RCS} \end{bmatrix} & \text{if phase} \leq 2 \\ \begin{bmatrix} \dot{m}_{LOX} \\ \dot{m}_{LH2} \end{bmatrix} & \text{if phase} = 3 \end{cases} \tag{11}$$

where the LH2 and LOX tank mass flow rates are calculated as

$$\dot{m}_{LH2} = \frac{-\dot{m}_{eng}}{MR+1} \tag{12}$$

and

$$\dot{m}_{LOX} = \frac{-\dot{m}_{eng}MR}{MR+1} \tag{13}$$

where

$$\dot{m}_{eng} = \frac{T_{vac}}{I_{sp}g_{Earth,SL}}. \tag{14}$$

No propellant is drawn from the RCS and FPR propellant tanks; therefore, the mass flow rates for these tanks are zero.

Controls, Objective Function, and Constraints

To complete the transcription process, the controls, $\bar{u}$, the objective function, J, and the constraints are defined in this section. The first phase is the only phase with active controls and a Local-Vertical Local-Horizontal (LVLH) Euler set is selected for steering during the burn arc

$$\bar{u} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \tag{15}$$

and is loosely referred to as roll, pitch, and yaw set, respectively.

To meet the mission objective of maximizing mass, the objective function is the minimum engine burn time, $t_{b_{eng}}$, which is equivalent to maximizing cutoff mass. It is also the same as minimizing the duration of phase 1 (Δt_1):

$$J = t_{f_1} - t_{0_1} = t_{b_{eng}} \tag{16}$$

where t_{0_1} and t_{f_1} are the initial and final time for phase 1, respectively.

The initial LEO is constrained in a way that ensures the orbit is nearly circular, has the correct size and is correctly tilted. The orientation of the orbit plane is prescribed by the ascending node angle, which is unconstrained. Optimal timing is achieved by allowing the argument of perigee and true anomaly to be unconstrained. The three initial constraints are:

$$\overline{\chi} = \begin{bmatrix} a(t_{o_1}) - a \\ inc(t_{0_1}) - inc \\ e(t_{0_1}) - e \end{bmatrix} \quad (17)$$

where the values for the semi-major axis, inclination, and eccentricity are given in Table 2.

TABLE 2.—LEO INITIAL SEMI-MAJOR AXIS, INCLINATION AND ECCENTRICITY

a (km)	inc (degree)	e
6,563	29	1.0×10^{-8}

To ensure that the spacecraft arrives on the desired trajectory, three final constraints, relative to the Moon, are used in the last phase (phase 3)

$$\overline{\Psi} = \begin{bmatrix} alt_p(t_{f_3}) - alt_p \\ inc(t_{f_3}) - inc \\ \dot{r}(t_{f_3}) - \dot{r} \end{bmatrix} \quad (18)$$

where the radial velocity component is given by

$$\dot{r} = \frac{e \sinh(H)\sqrt{-aGM}}{a(1 - e\cosh(H))}. \quad (19)$$

The radial velocity component constraint ensures that the pericynthion altitude, alt_p, is attained and is the altitude of closed approach. It is equivalent to arriving at an anomaly of zero. The semi-major axis, a, is negative because the trajectory is hyperbolic relative to the Moon. H and GM are the hyperbolic anomaly and the Moon's gravitational constant, respectively.

Table 3 lists the values for the lunar arrival constraints.

TABLE 3.—DESIRED LUNAR ARRIVAL CONDITIONS

alt_p (km)	inc (degree)	$\dot{r}$ (km/s)
100	90	1.0×10^{-8}

Now, just for organizational convenience, the vehicle states are grouped together in a new state array as

$$\overline{s} = \begin{bmatrix} \overline{r}_{j2k} \\ \overline{v}_{j2k} \\ \overline{m}_{tnk} \end{bmatrix} \quad (20)$$

where it is understood that $\overline{m}_{tnk}$ is the integral of Equation (11) and refers to all four propellant tanks in the first and second phase and only the fuel and oxidizer tanks in the third phase. The discretization process results in the NLP states becoming discretized as

$$\bar{x} = \begin{bmatrix} \bar{s}_i \\ \bar{u}_i \\ \bar{t}_{0_k} \\ \overline{\Delta t_k} \end{bmatrix}, \quad i = 1,3,5,\dots, \ k = 1,2,3^3 \tag{21}$$

The remaining constraints are linear. These constraints ensure that the start time of one phase is also the end time of the previous phase. These constraints are given as

$$\bar{\eta} = \begin{bmatrix} t_{0_1} + \Delta t_1 - t_{0_2} \\ t_{0_2} + \Delta t_2 - t_{0_3} \end{bmatrix} \tag{22}$$

where t_{0_1} and t_{0_2} indicate the start times for phase 1 and 2, respectively.

Additional linear constraints given below are required to ensure continuity across phases with the vehicle states

$$\bar{\zeta}_s = \begin{bmatrix} \bar{s}_2^+ - \bar{s}_1^- \\ \bar{s}_3^+ - \bar{s}_2^- \end{bmatrix} \tag{23}$$

and the controls

$$\bar{\zeta}_u = \begin{bmatrix} \bar{u}_2^+ - \bar{u}_1^- \\ \bar{u}_3^+ - \bar{u}_2^- \end{bmatrix} \tag{24}$$

where $\bar{s}_2^+$ and $\bar{s}_3^+$ indicate the vehicle state values at the first node of phase 2 and 3, respectively, and $\bar{s}_1^-$ and $\bar{s}_2^-$ indicate the values at the last node of phase 1 and 2, respectively (likewise for the controls). Refer to Figures 1 and 2 in the previous section for a schematic representation. Because of the mass jettisons that occur at the end of phase 2, mass is allowed to be discontinuous between phase 2 and 3.

Again for convenience, the linear constraints, given by Equations (22) to (24), and nonlinear constraints, given by Equations (17), (18) and the defects, $\bar{\Delta}$, are grouped into a linear constraint array

$$\bar{\xi} = \begin{bmatrix} \bar{\eta} \\ \bar{\zeta}_s \\ \bar{\zeta}_u \end{bmatrix} \tag{25}$$

and a nonlinear constraint array

$$\bar{\delta} = \begin{bmatrix} \bar{\chi} \\ \bar{\Psi} \\ \overline{\Delta}_{i+1} \end{bmatrix} \quad i = 1,3,5,\dots^4 \tag{26}$$

where the defects (evaluated at all of the even-numbered nodes) are the differences between the estimated state derivatives, $\tilde{\dot{s}}$, and the true state derivatives, $\dot{s}$, as described previously,

---

[3] i is global node number and k is phase number
[4] i is global node number

$$\overline{\Delta} = \tilde{\bar{s}} - \bar{\bar{s}}. \tag{27}$$

This permits the dependant variable array, $\overline{F}(\bar{x})$, in Equation (6) to be represented as

$$\overline{F}(\bar{x}) = \begin{bmatrix} J \\ \overline{\delta} \\ \overline{\xi} \end{bmatrix} . \tag{28}$$

The TLI problem is now completely transcribed to a constrained NLP problem using Equations (21) and (28).

Closed-Loop Guidance With Powered Explicit Guidance

PEG is the closed loop guidance algorithm used on the Space Shuttle to handle all phases of its exo-atmospheric powered flight. It uses closed-form equations for the propulsive acceleration term (called thrust integrals) to accommodate constant thrust or constant acceleration phases of flight. The fuel-optimal time histories of the unit thrust vector and its rate of change are derived using Calculus of Variations (CoV). PEG can support up to 7 different end conditions. PEG's simplicity and heritage make it attractive. Heritage is an important factor in the design selection process because proven designs can significantly reduce risk and costs associated with testing. Slightly different versions of PEG appear in the literature and have minor differences in the thrust integral terms (Refs. 14 and 15). The version in MASTIF is based on Reference 14.

Numerical Implementation and Results

The optimization procedure in MASTIF is iterative, primarily for solution accuracy. A procedure known as grid or mesh refinement ensures that the quantity and/or placement of nodes represent the solution accurately (Ref. 16). An automated grid refinement routine is a future development effort for MASTIF. For now, solution accuracy can still be assessed and even improved by simply adding more nodes and/or segments.

Solution accuracy is assessed by taking the optimal control time histories, $\bar{u}^*(t)$, (captured at all the odd nodes), and the optimal initial vehicle states, $\bar{s}^*(t_0)$, (captured at the first node of the first phase), from the optimization mode and then explicitly integrating the equations of motion using $\bar{s}^*(t_0)$ and $\bar{u}^*(t)$ from the initial start time, t_0, to the final time, t_f. The resulting trajectory is hereafter referred to as the explicit solution whereas the solution obtained from the optimization is hereafter referred to as the implicit solution because of the two collocation phases that do not require integration of the equations of motion. Any differences between the two solutions are errors with the explicitly integrated trajectory regarded as the truth model. Because the last phase is a Runge-Kutta phase, the errors are due to the implicit solutions in the first two phases. This process of accessing the solution accuracy revealed that the lunar arrival constraints had the largest errors.

TABLE 4.—IMPLICIT VERSUS EXPLICIT ABSOLUTE LUNAR ARRIVAL ERRORS

nNodes$_{odd}$	alt_p (km)	inc (degree)	$\dot{r}$ (km/s)
21	571.179	7.085	1.157
35	20.707	1.65	0.183
61	1.521	0.00016	0.00055

Table 4 shows how these errors were reduced with additional nodes ($n\text{Nodes}_{odd}$). There will be a tradeoff between accuracy and execution time. For the purpose of this research, the errors from the solution using 61 odd nodes were deemed acceptable and will be used for the remainder of the paper.

Open-Loop Solution

The solution to the constrained NLP problem, Equation (6), results in the optimal control, $\bar{u}^*$, for the powered portion of flight. These controls ensure that mass is maximized, while meeting all initial and terminal constraints.

The θ time history is shown as a cubic spline curve fit through the data points at the odd nodes for the powered portion of flight in Figure 3. Because nearly all the motion is within the Earth-to-Moon transfer plane, out-of-plane motion is minimal ($< 0.5°$) ϕ and ψ angles are not shown.

The mass time history is shown in Figure 4. Mass decreases linearly during the main engine burn until MECO. It remains constant during the second phase of flight. One hour after the start of the main engine burn, two propellant tanks and the ETS are jettisoned which result in the mass discontinuity. The double nodes at phase boundaries as discussed previously enable this discontinuity. The rest of the transfer is unpowered; therefore, mass remains constant.

The spacecraft's north to south lunar approach trajectory is shown in Figure 5 as well as the Moon's orbital path.
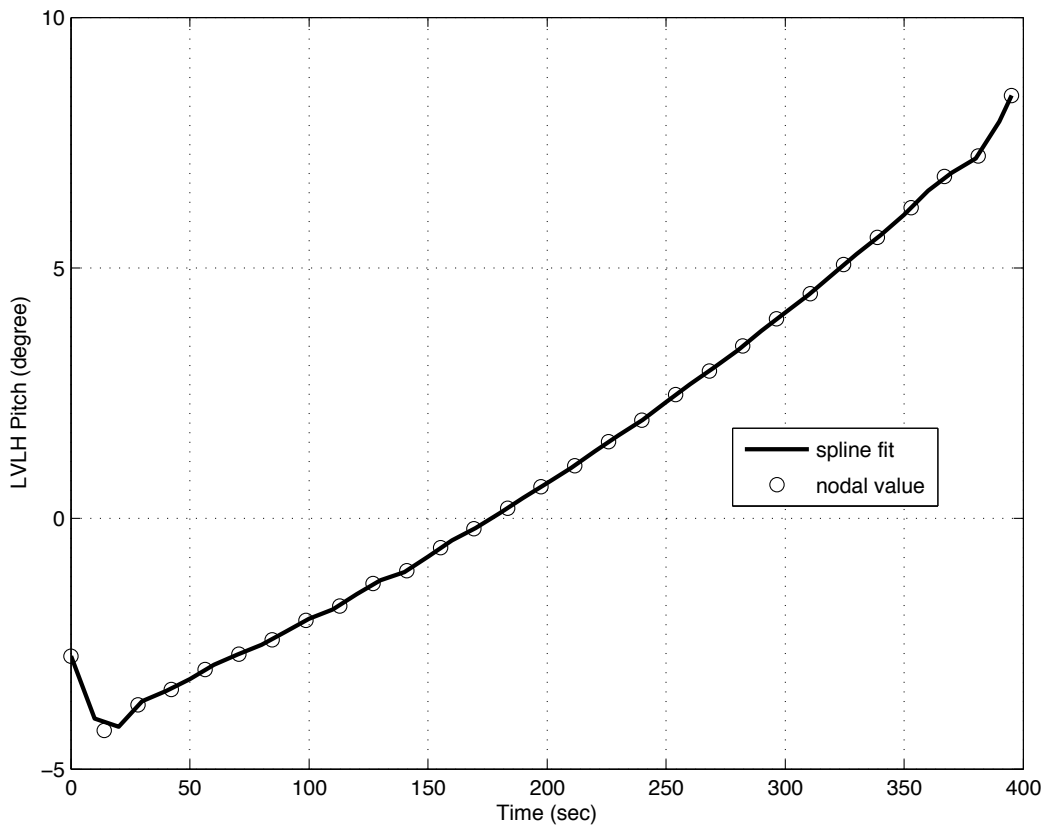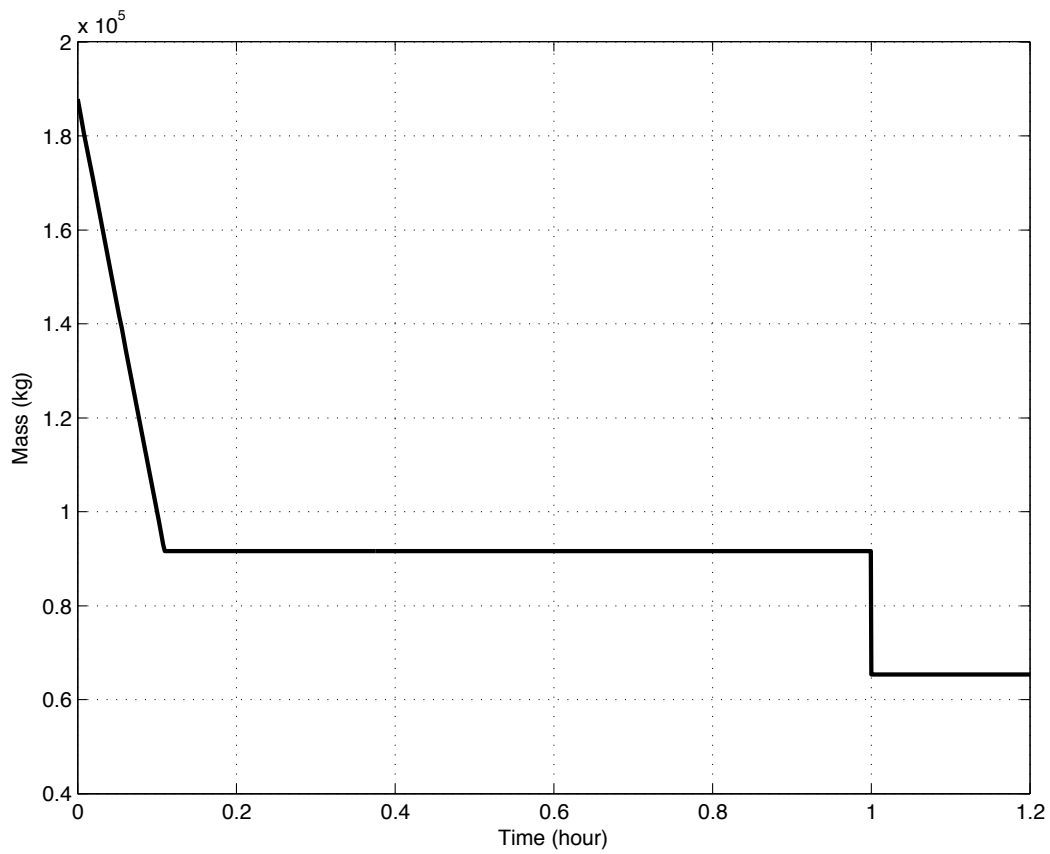

Figure 3.—Optimal θ time history.

Figure 4.—Vehicle mass time history.

direction of Moon's motion

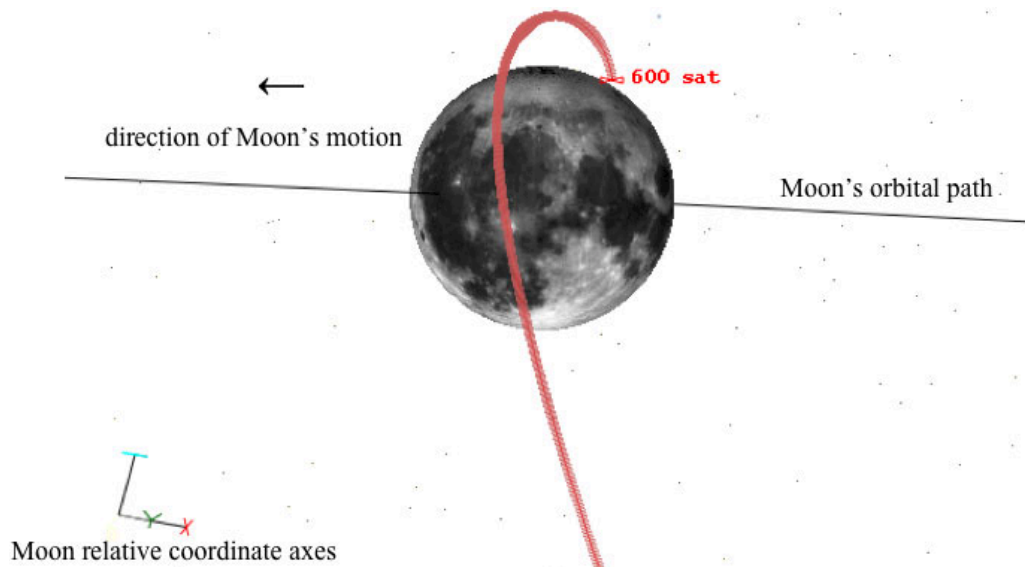Moon's orbital path

600 sat

Moon relative coordinate axes

Figure 5.—Lunar approach trajectory (600 sat is the name of the spacecraft in visualization tool).

All constraints were satisfied within acceptable tolerances. The optimized values for the three free orbital angles (right ascension of ascending node, Ω, argument of perigee, ω, and true anomaly, ν, are shown in Table 5. The burn starts nearly 18° (~ 1 min) before perigee passage ($\nu = 360°$).

TABLE 5.—OPTIMAL LEO DEPARTURE ANGLES

Ω (degree)	ω (degree)	ν (degree)
100.1550	226.9605	341.8067

Table 6 shows that the burn lasts for nearly 400 sec and lists the propellant remaining at MECO.

TABLE 6.—BURN TIME AND PROPELLANT REMAINING AT MECO

$t_{b_{eng}}$ (sec)	m_{LOX} (kg)	m_{LH2} (kg)
395.260595	387	70

The actual ΔV required, ΔV_{actual}, to perform the burn is compared to the ideal, ΔV_{ideal}, in Table 7 (Ref. 17). The difference is due to the gravity loss associated with having a non-zero flight path angle for the finite burn and a zero flight path angle for the instantaneously modeled burn (the ideal velocity change).

TABLE 7.—VELOCITY CHANGE FROM THE TLI BURN

ΔV_{ideal} (m/s)	ΔV_{actual} (m/s)	ΔV_{gLoss} (m/s)
3,142.843	3,157.415	14.572

Closed-Loop Solution

The targeted end conditions for PEG are the MECO values of flight path angle, γ, radius magnitude, R, and velocity magnitude, V, inclination, inc, and right ascension, Ω, which are obtained from MASTIF's optimization mode (the open-loop solution) and are listed in Table 8. Perfect navigation state knowledge is assumed. Guidance is called at a frequency of 1 Hz. Numerical integration of the equations of motion is performed with a second order Runga-Kutta with a fixed step-size of 0.01 sec.

TABLE 8.—PEG TARGET MECO PARAMETERS

R (km)	V (km/s)	γ (degree)	inc (degree)	Ω (degree)
6,730.5465	10.7965	8.0187	28.9387	100.0423

The closed-loop LVLH θ time history during the TLI burn is compared to the open-loop solution in Figure 6. After five iterations, PEG converges to the same initial θ angle as the optimized open-loop solution. At each guidance cycle, the converged solution from the previous cycle is used to start the iteration. The simulation proceeds in this manner the until the guidance commanded engine shutdown, which indicates that the vehicle has arrived at the MECO target. Differences of 1.5° are seen at the end of the burn.
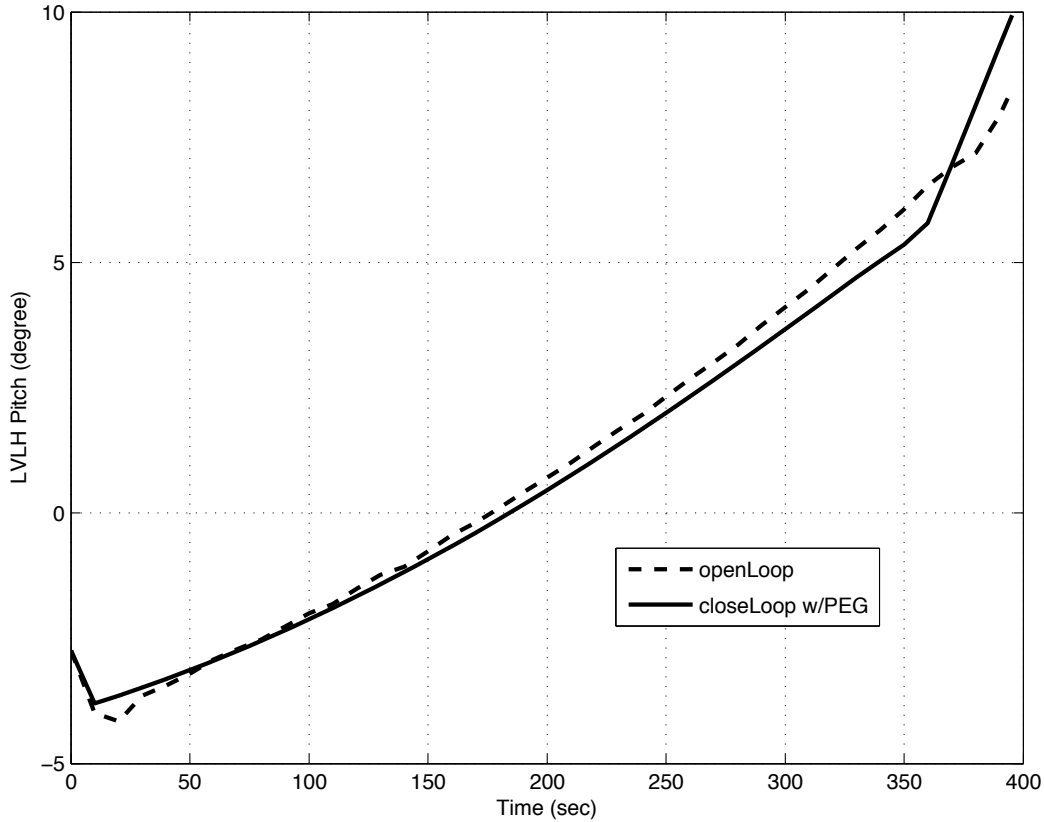
Figure 6.—θ angle comparison.

An open-loop and closed-loop performance comparison, as characterized by the burn time required (and the resulting remaining propellant), is shown in Table 9. The open-loop burn time is rounded to two decimal places because the finest resolution achievable for the closed-loop burn time is 100 Hz due to the selected integration frequency. The performance of PEG compares favorably.

TABLE 9.—OPEN-LOOP AND CLOSED-LOOP MECO TIME AND LOX/LH2 REMAINING

Method	$t_{b_{eng}}$ (sec)	m_{LOX} (kg)	m_{LH2} (kg)
Open-loop	395.26	387	70
Closed-loop	395.29	381	69

In order for the closed-loop solution to have the correct lunar arrival conditions, the errors at MECO must be zero, which they are not, as shown in Table 10. Errors are present because closed-loop systems use approximate formulations to derive closed-form solutions. Different PEG input parameters can be adjusted to minimize the errors at MECO, but result in slightly different θ angle profiles. No adjustments could be made to null all the MECO errors.

TABLE 10.—CLOSED-LOOP MECO ERRORS

R (m)	V (m/s)	γ (degree)	inc (degree)	Ω (degree)
1,318	1.32	0.044	0.00015	10.00044

Table 11 shows that MECO errors translate to lunar arrival errors (for the closed-loop system). The error in pericynthion altitude is 225 km and the inclination error is 6°. To ensure that the spacecraft arrives at the desired lunar location, mid course corrections (MCC) would be needed during the transfer and/or MECO target parameters could be updated (instead of fixed) during the burn. Both a targeting scheme and MCC modeling are potential future tasks.

TABLE 11.—LUNAR ARRIVAL CONDITIONS

Parameter	Open-loop	Closed-loop
alt_p (km)	100	325
inc (degree)	90	96
$\dot{r}$ (km/s)	0	–0.4

Conclusion

Trajectory optimization software utilizing a direct optimization method using collocation was developed and implemented as a new mode in a low-to-high fidelity 3-DOF/6-DOF vehicle simulation tool named MASTIF. This newly developed optimization mode is used to solve a human-class TLI trajectory problem by transcribing the trajectory optimization problem to a constrained NLP problem. So far as known from the available literature, this paper presents new research work using a direct method to solve this human-class translunar trajectory problem.

MASTIF's software capability is rare because it can be utilized as a trajectory optimization program and a vehicle simulation tool to design and test GN&C algorithms. MASTIF's 6-DOF architecture can readily be leveraged to extend the optimization mode to 6-DOF. Currently, the authors' are aware of only one program (POST) used within NASA with that functionality. To demonstrate MASTIF's versatility, a closed-loop guidance scheme with heritage, PEG, is assessed using the results, in the form of fixed MECO target parameters, from the optimization (open-loop) solution. Results show the efficacy of the discretization scheme and, additionally, that using fixed MECO target parameters would be acceptable for the first design cycle analysis, but that one or more MCCs would be required to ensure the spacecraft's desired lunar arrival conditions are satisfied.

The synergy from combining trajectory optimization and 3/6-DOF vehicle simulation software into a single software tool is expected to be a more efficient means to designing, testing, and evaluating vehicle GN&C software because model fidelity remains consistent for the trajectory optimization and vehicle simulation.

Appendix: Hermite Interpolating Polynomials

The expressions below are nearly exactly as those given in Reference 12. The interpolated state value is given by

$$\tilde{s}(t) = \sum_{i=1}^{n} \alpha_i(t)s(t_i) + \sum_{i=1}^{n} \beta_i(t)\dot{s}(t_i) \tag{A.1}$$

where n is a function of the number of nodes per segment, $n\text{Nodes}_{seg}$,

$$n = \frac{n\text{Nodes}_{seg} + 1}{2}. \tag{A.2}$$

The order of the polynomial corresponds to $n\text{Nodes}_{seg}$, which is a user input.

The generic functions, $\alpha_j(t)$ and $\beta_j(t)$ are

$$\beta_j(t) = \left[\prod_{\substack{i=1 \\ i \neq j}}^{n} \left(\frac{t - t_i}{t_j - t_i} \right)^2 \right] (t - t_j) \tag{A.3}$$

$$\alpha_j(t) = \left[\prod_{\substack{i=1 \\ i \neq j}}^{n} \left(\frac{t - t_i}{t_j - t_i} \right)^2 \right] \left(1 + 2(t_j - t) \right) \sum_{\substack{i=1 \\ i \neq j}}^{n} \frac{1}{t_j - t_i} \tag{A.4}$$

and it is understood that the subscript j is a dummy index. Likewise, the interpolated state derivative value is

$$\dot{\tilde{s}}(t) = \sum_{i=1}^{n} \dot{\alpha}_i(t)s(t_i) + \sum_{i=1}^{n} \dot{\beta}_i(t)\dot{s}(t_i) \tag{A.5}$$

where $\dot{\alpha}(t)$ and $\dot{\beta}(t)$ are obtained by the product rule of differentiation.

References

1. Columbia Accident Investigation Board, "The Columbia Accident Investigation Board Report," NASA, 2003.
2. NASA, "The Vision for Space Exploration," PDF file, 2004.
3. P.E. Gill, W. Murray and M.A. Saunders, "SNOPT: An SQP Algorithm For Large Scale Constrained Optimization," *Society for Industrial and Applied Mathematics*, Vol. 47, No. 1, 2005, pp. 99–131.
4. C.R. Hargraves and S.W. Paris, "Direct Trajectory Optimization Using Nonlinear Programming and Collocation," *Journal of Guidance, Control, and Dynamics*, Vol. 10, No. 4, 1987, pp. 338–342.
5. K.E. Witzberger, D.A. Smith, M.C. Martini and T.W. Wright, "MASTIF User's Manual," to be released, NASA Glenn Research Center.
6. S.A. Cook and T. Vanhooser, "The Next Giant Leap: NASA's Ares Launch Vehicles Overview," *Aerospace Conference*, 2008 IEEE, Big Sky, Montana, March 1–8, 2008.
7. S.A. Striepe, and et al., "Program To Optimize Simulated Trajectories (POST II)," NASA Langley Research Center and Lockheed Martin Corporation, Version 1.1.6.G, January 2004.
8. R.L. McHenry and et al., "Space Shuttle Ascent Guidance, Navigation, and Control," *Journal of the Astronautically Sciences,* Vol. XXVII, No. 1, 1979, pp. 1–38.
9. R.B. Roncoli, "Lunar Constants and Models Document," *JPL D-32296,* Jet Propulsion Laboratory, Pasadena, CA. 2005.
10. E. Fehlberg, "Classical fifth-, sixth-, seventh-, and eight-order runge-kutta formulas with step-size control," NASA TR R-287, NASA, 1968.
11. C. Canuto, A.Q. Hussaini and T.A. Tang, *Spectral Methods in Fluid Dynamics,* Springer-Verlag, 1987.
12. S.W. Paris, J.P. Riehl and W.K. Sjauw, "Enhanced Procedures for Direct Trajectory Optimization Using Nonlinear Programming and Implicit Integration," *AIAA/AAS Astrodynamics Specialist Conference and Exhibit,* Keystone, Colorado, August 21-24, 2006.
13. G.R. Schubert, "Algorithm 211, Hermite Interpolation," *Communications of the ACM,* Vol. 6, No. 10, 1963, p. 617.
14. R.F. Jaggers, "An Explicit Solution to the Exoatmospheric Powered Flight Guidance and Trajectory Optimization Problem for Rocket Propelled Vehicles," *Guidance and Control Conference*, Hollywood, Florida, August 8-10, 1977.
15. R.F. Jaggers, "Asymmetrical Booster Ascent Guidance and Control System Design Study, Volume V, Space Shuttle Powered Explicit Guidance," NASA CR 140191, NASA, 1976.
16. J.T. Betts, "Practical Methods for Optimal Control Using Nonlinear Programming," *Society for Industrial and Applied Mathematics*, 2001.
17. Plain text file. "Exlx_Flt3.0_Option_1_North_to_South_RKF78_MODIFIED.txt." From NASA Johnson Space Center, 2010.

REPORT DOCUMENTATION PAGE

1. REPORT DATE *(DD-MM-YYYY)* 01-01-2012	2. REPORT TYPE Technical Memorandum	3. DATES COVERED *(From - To)*

4. TITLE AND SUBTITLE Direct Method Transcription For a Human-Class Translunar Injection Trajectory Optimization	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) Witzberger, Kevin, E.; Zeiler, Tom	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER WBS 789822.01.02.03

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration John H. Glenn Research Center at Lewis Field Cleveland, Ohio 44135-3191	8. PERFORMING ORGANIZATION REPORT NUMBER E-17896

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, DC 20546-0001	10. SPONSORING/MONITOR'S ACRONYM(S) NASA
	11. SPONSORING/MONITORING REPORT NUMBER NASA/TM-2012-217215

12. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified-Unlimited
Subject Category: 13
Available electronically at http://www.sti.nasa.gov
This publication is available from the NASA Center for AeroSpace Information, 443-757-5802

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This paper presents a new trajectory optimization software package developed in the framework of a low-to-high fidelity 3 degrees-of-freedom (DOF)/6-DOF vehicle simulation program named Mission Analysis Simulation Tool in Fortran (MASTIF) and its application to a translunar trajectory optimization problem. The functionality of the developed optimization package is implemented as a new "mode" in generalized settings to make it applicable for a general trajectory optimization problem. In doing so, a direct optimization method using collocation is employed for solving the problem. Trajectory optimization problems in MASTIF are transcribed to a constrained nonlinear programming (NLP) problem and solved with SNOPT, a commercially available NLP solver. A detailed description of the optimization software developed is provided as well as the transcription specifics for the translunar injection (TLI) problem. The analysis includes a 3-DOF trajectory TLI optimization and a 3-DOF vehicle TLI simulation using closed-loop guidance.

15. SUBJECT TERMS
Trajectory optimization; Trajectory analysis

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON STI Help Desk (email:help@sti.nasa.gov)
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U	UU	26	19b. TELEPHONE NUMBER *(include area code)* 443-757-5802