



Ricciardi, Lorenzo A. and Vasile, Massimiliano and Toso, Federico and Maddock, Christie Alisa (2016) Multi-objective optimal control of ascent trajectories for launch vehicles. In: AIAA/AAS Astrodynamics Specialist Conference, 2016. AIAA Space Forum . American Institute of Aeronautics and Astronautics Inc, AIAA, Reston, VA. ISBN 9781624104459 , <http://dx.doi.org/10.2514/6.2016-5669>

This version is available at <https://strathprints.strath.ac.uk/60564/>

Strathprints is designed to allow users to access the research output of the University of Strathclyde. Unless otherwise explicitly stated on the manuscript, Copyright © and Moral Rights for the papers on this site are retained by the individual authors and/or other copyright owners. Please check the manuscript for details of any other licences that may have been applied. You may not engage in further distribution of the material for any profitmaking activities or any commercial gain. You may freely distribute both the url (<https://strathprints.strath.ac.uk/>) and the content of this paper for research or private study, educational, or not-for-profit purposes without prior permission or charge.

Any correspondence concerning this service should be sent to the Strathprints administrator: strathprints@strath.ac.uk

Multi-Objective Optimal Control of Ascent Trajectories for Launch Vehicles

Lorenzo A. Ricciardi*, Massimiliano Vasile†, Federico Toso‡ and Christie Alisa Maddock§

*Aerospace Centre of Excellence, Department of Mechanical & Aerospace Engineering
University of Strathclyde, Glasgow G1 1XJ, United Kingdom*

This paper presents a novel approach to the solution of multi-objective optimal control problems. The proposed solution strategy is based on the integration of the Direct Finite Elements Transcription method, to transcribe dynamics and objectives, with a memetic strategy called Multi Agent Collaborative Search (MACS). The original multi-objective optimal control problem is reformulated as a bi-level nonlinear programming problem. In the outer level, handled by MACS, trial control vectors are generated and passed to the inner level, which enforces the solution feasibility. Solutions are then returned to the outer level to evaluate the feasibility of the corresponding objective functions, adding a penalty value in the case of infeasibility. An optional single level refinement is added to improve the ability of the scheme to converge to the Pareto front. The capabilities of the proposed approach will be demonstrated on the multi-objective optimisation of ascent trajectories of launch vehicles.

Nomenclature

α	Angle of attack, rad
ψ	Boundary constraints
χ	Flight heading angle, rad
δ_T	Throttle
γ	Flight path angle, rad
λ	Latitude, rad N
\mathbf{g}	Algebraic constraints
\mathbf{u}	Control vector
\mathbf{x}	State vector
μ	Bank angle, rad
ω_E	Rotational velocity of Earth, rad/s
ρ	Atmospheric density, kg/m ³
θ	Longitude, rad E
A_e	Exit area of rocket nozzle, m ²
C_d	Coefficient of drag
C_L	Coefficient of lift
D	Drag force, N
g, g_0	Gravitational acceleration, m/s ²
h	Altitude above Earth surface, m
I_{sp}	Specific impulse, s
J, \mathbf{J}	Objective function(s)
L	Lift force, N

*Ph.D. Candidate, AIAA Student Member

†Professor, AIAA Member

‡Ph.D. Candidate, AIAA Student Member

§Lecturer, AIAA Member

m	Mass (of the vehicle), kg
m_p	Mass of on-board propellant, kg
R_E	Mean radius of Earth, km
S_{ref}	Aerodynamic reference area, m ²
T	Propulsive thrust, N
t	Time, s
v	Velocity, m/s

I. Introduction

MULTI-OBJECTIVE optimal control problems arise when one seeks a control program that is optimal with respect to multiple conflicting objectives. As such, any multi-objective approach should return many different solutions, each one consisting of a different trade-off between the different objectives. Only a limited number of methods exist to solve multi-objective optimal control problems. Coverstone-Carroll et al.¹ combined a genetic algorithm and an indirect approach to optimal control problems in a dual loop algorithm. In the outer loop, NSGAI generates vectors of co-states and other static optimisation parameters. For each set, the inner loop solves a single objective optimal control problem, minimising only one of the objectives. Ober-Blobaum et al.² used a direct transcription approach coupled with an approach that scalarised the multi-objective vector along directions pointing at predefined unreachable points in the criteria space. Each scalar problem was then solved with a standard NLP solver. Kaya & Maurer³ proposed a similar approach but used a smoothed version of Tchebycheff scalarisation to scalarise the multi-objective optimisation (MOO) problem. Englander & Vavrina⁴ proposed a dual loop algorithm in which the outer loop solved a multi-objective problem handling a set of categorical variables, and the inner loop solved a set of constrained optimal control problems using Monotonic Basin Hopping, solving for only one of the objectives.

This paper proposes a direct method based on the Direct Finite Elements in Time⁵ (DFET) transcription and the Multi Agent Collaborative Search⁶ (MACS) multi-objective optimisation algorithm. In DFET, the time domain is divided into a given number of segments, or elements, and the states and controls are approximated as arbitrary order polynomials on a spectral basis. DFET was successfully used to solve many difficult trajectory optimisation problems,⁷⁻⁹ and is a very robust transcription method with many interesting characteristics. As pointed out by Bottasso et al.,¹⁰ if used for forward time integration it is equivalent to some classes of implicit Runge-Kutta integration schemes. Moreover it can be extended to an arbitrary high order and supports full h-p adaptivity. MACS is a memetic evolutionary multi-objective optimisation algorithm in which a population of agents is initially seeded in the search space with a Latin Hypercube sampling. All agents can perform a set of individual actions to explore their neighborhood and improve their position. Non-dominated solutions are saved to an external archive. Some agents can, afterwards, perform social actions, exploiting the information coming from the archive or from other agents to simultaneously advance towards the Pareto front. Individual and social actions are repeated until a maximum number of function evaluations is reached. A special archiving strategy ensures a good distribution of points across the Pareto front.

In order to demonstrate its applicability, the proposed approach is tested on two test cases optimising the ascent trajectory of a launch vehicle. It is interesting to note that although much research has been dedicated to the optimisation of rocket ascent trajectories (with the seminal work of Goddard¹¹ dating back almost a century), most of that effort has been dedicated to include more sophisticated physics¹²⁻¹⁵ but little has been done on multi-objective optimal control. The only relevant work found was carried out by Pagano et al.¹⁶ In it rather detailed models were included, but the shape of the control law was defined a priori and only some of the parameters describing this shape were optimised. Moreover, only one cost function, the mass of the payload, was maximised. The other objectives were the violations of the constraints.

The paper is structured with the Section II explaining the theory and details of the specific approach, including the transcription through DFET and the multi-objective optimisation strategy based on MACS, while Section III the bi-level NLP problem formulation. Section IV details the models and configuration parameters for the two test cases on the ascent trajectory optimisation of rocket-based launch vehicles. Section V shows the results and discussion, followed by a short conclusion.

II. Direct Transcription of Multi-Objective Optimal Control Problems

II.A. Problem Formulation

Multi-objective optimal control problems can be formulated as:

$$\begin{aligned}
 \min_{\mathbf{u} \in U} \mathbf{J} &= [J_1, J_2, \dots, J_i, \dots, J_m]^T \\
 \text{s.t.} & \\
 \dot{\mathbf{x}} &= \mathbf{F}(\mathbf{x}, \mathbf{u}, t) \\
 \mathbf{g}(\mathbf{x}, \mathbf{u}, t) &\geq 0 \\
 \psi(\mathbf{x}_0, \mathbf{x}_f, t_0, t_f) &\geq 0 \\
 t &\in [t_0, t_f]
 \end{aligned} \tag{1}$$

where \mathbf{J} is a vector of objectives J_i that are functions of the state vector $\mathbf{x} : [t_0, t_f] \rightarrow \mathbb{R}^n$, control variable $\mathbf{u} \in L^\infty$ and time t . The functions \mathbf{x} belong to the Sobolev space $W^{1,\infty}$ while the objective functions are $J_i : \mathbb{R}^{n+2} \times \mathbb{R}^p \times [t_0, t_f] \rightarrow \mathbb{R}$. The objective vector is subject to a set of dynamic constraints with $\mathbf{F} : \mathbb{R}^n \times \mathbb{R}^p \times [t_0, t_f] \rightarrow \mathbb{R}^n$, algebraic constraints $\mathbf{g} : \mathbb{R}^n \times \mathbb{R}^p \times [t_0, t_f] \rightarrow \mathbb{R}^s$, and boundary conditions $\psi : \mathbb{R}^{2n+2} \rightarrow \mathbb{R}^q$. Note that the problem in Eq. (1) generally is non-smooth and can include a number of additional static parameters.

II.B. Problem Transcription

Problem (1) is transcribed into a multi-objective, non-linear programming problem via DFET.⁵ DFET was initially proposed by Vasile⁷ in 2000 and uses finite elements in time on spectral bases to transcribe the differential equations into a set of algebraic equations. Finite Elements in Time (FET) for the indirect solution of optimal control problems were initially proposed by Hodges et al.,¹⁷ and during the late 1990s evolved to the discontinuous version. As pointed out by Bottasso et al.,¹⁰ FET for the forward integration of ordinary differential equations are equivalent to some classes of implicit Runge-Kutta integration schemes, can be extended to arbitrary high-order, are very robust and allow full h-p adaptivity. In the past decade, direct transcription with FET on spectral bases has been successfully used to solve a range of difficult problems: from the design of low-thrust multi-gravity assist trajectories to Mercury⁸ and to the Sun,⁹ to the design of weak stability boundary transfers to the Moon, low-thrust transfers in the restricted three body problem and optimal landing trajectories to the Moon.⁷

For each individual cost function consider the following Bolza's problem:

$$\min_{\mathbf{u} \in U} J_i = \alpha_i \phi_i(\mathbf{x}_0, \mathbf{x}_f, t_0, t_f) + \beta_i \int_{t_0}^{t_f} L_i(\mathbf{x}, \mathbf{u}, t) dt \tag{2}$$

where α_i and β_i are positive weights. In multi-objective optimisation this formulation corresponds to a weighted sum scalarisation, which is known to be unable to represent points on non-convex regions of the Pareto front. To avoid this problem, this paper only considers cases with $(\alpha_i = 1, \beta_i = 0)$ or $(\alpha_i = 0, \beta_i = 1)$. The differential constraints can be recast in weak form and integrated by parts leading to,

$$\int_{t_0}^{t_f} \dot{\mathbf{w}}^T \mathbf{x} + \mathbf{w}^T \mathbf{F}(\mathbf{x}, \mathbf{u}, t) dt - \mathbf{w}_f^T \mathbf{x}_f^b + \mathbf{w}_0^T \mathbf{x}_0^b = 0 \tag{3}$$

where \mathbf{w} are the generalised weight functions and \mathbf{x}^b are the boundary values of the states, that may be either imposed or free. Let the time domain D be decomposed into N finite elements such that

$$D = \bigcup_{j=1}^N D_j(t_{j-1}, t_j) \tag{4}$$

and parametrise, over each D_j , the states, controls and weight functions as

$$\mathbf{x}(t) = \overset{N}{\underset{j=1}{\cup}} \mathbf{X}_j = \overset{N}{\underset{j=1}{\cup}} \sum_{s=0}^l f_{sj}(t) \mathbf{x}_{sj} \quad (5a)$$

$$\mathbf{u}(t) = \overset{N}{\underset{j=1}{\cup}} \mathbf{U}_j = \overset{N}{\underset{j=1}{\cup}} \sum_{s=0}^m g_{sj}(t) \mathbf{u}_{sj} \quad (5b)$$

$$\mathbf{w}(t) = \overset{N}{\underset{j=1}{\cup}} \mathbf{W}_j = \overset{N}{\underset{j=1}{\cup}} \sum_{s=0}^{l+1} h_{sj}(t) \mathbf{w}_{sj} \quad (5c)$$

where $\overset{N}{\underset{j=1}{\cup}}$ denotes the juxtaposition of the polynomials defined over each sub-interval, and the functions f_{sj} , g_{sj} and h_{sj} are chosen among the space of polynomials of degree l , m and $(l+1)$ respectively. It is practical to define each D_j over the normalised interval $[-1, 1]$ through the transformation,

$$\tau = 2 \frac{t - \frac{t_j - t_{j-1}}{2}}{t_j - t_{j-1}} \quad (6)$$

This way it is easy to express the polynomials f_{sj} , g_{sj} and h_{sj} as the Lagrange interpolation on the Gauss nodes in the normalised interval:

$$f_{sj} = \prod_{k=0, k \neq s}^l \frac{\tau - \tau_k}{\tau_s - \tau_k} \quad (7)$$

and similarly for g_{sj} and h_{sj} . Different Gauss nodes will lead to schemes with slightly different characteristics. In this work, Gauss-Lobatto nodes will be used for the generation of the polynomials for states and weight functions, while Gauss-Legendre nodes will be used for the controls. Substituting the definitions of the polynomials into the objective functions and integrating with Gauss quadrature formulas leads to,

$$\tilde{J}_i = \alpha_i \phi_i(\mathbf{X}_0^b, \mathbf{X}_f^b, t_0, t_f) + \beta_i \sum_{j=1}^N \sum_{k=1}^{l+1} \sigma_k L_i(\mathbf{X}_j(\tau_k), \mathbf{U}_j(\tau_k), \tau_k) \frac{\Delta t}{2} \quad (8)$$

and for the variational constraints leads to the system,

$$\sum_{k=1}^{l+1} \sigma_k \left[\dot{\mathbf{W}}_j(\tau_k)^T \mathbf{X}_j(\tau_k) + \mathbf{W}_j(\tau_k)^T \mathbf{F}_j(\tau_k) \frac{\Delta t}{2} \right] - \mathbf{W}_{p+1}^T \mathbf{X}_j^b + \mathbf{W}_1^T \mathbf{X}_j^b = 0 \quad (9)$$

where τ_k and σ_k are the Gauss nodes and weights, and $\mathbf{F}_j(\tau_k)$ is the shorthand notation for $\mathbf{F}(\mathbf{X}_j(\tau_k), \mathbf{U}_j(\tau_k), \tau_k)$. Gauss-Legendre weights and nodes are used for the numerical quadrature in this work, i.e., the polynomials generated through the Lagrange interpolation over the Gauss-Legendre or Gauss-Lobatto nodes are evaluated at the Gauss-Legendre nodes over each interval. With DFET, the optimal control problem in Eq. (2) was transcribed into the non-linear programming (NLP) problem (8), which in compact reads as:

$$\begin{aligned} & \min_{\mathbf{p}} \tilde{\mathbf{J}}(\mathbf{x}_s, \mathbf{p}) \\ & s.t. \\ & \mathbf{c}(\mathbf{x}_s, \mathbf{p}) \geq 0 \end{aligned} \quad (10)$$

where the vector \mathbf{x}_s contains all the nodal values for the states and $\mathbf{p} = [\mathbf{u}_s, \mathbf{x}_0, \mathbf{x}_f, t_0, t_f]^T$ collects all the static and dynamic control variables.

III. Bi-level Multi-Objective Formulation

Problem (10) is further translated into the following two-level optimisation problem:

$$\begin{aligned} & \min_{\mathbf{p}^*} \tilde{\mathbf{J}}(\mathbf{x}^*, \mathbf{p}^*) \\ & s.t. \\ & (\mathbf{x}^*, \mathbf{p}^*) = \operatorname{argmin} \{f(\mathbf{x}_s, \mathbf{p}) | \mathbf{c}(\mathbf{x}_s, \mathbf{p}) \geq 0\} \end{aligned} \quad (11)$$

Algorithm 1 Outer level - MACS2 with local refinement

- 1: Initialise control parameters \mathbf{p} , search directions λ , $iter = 0$ and population P_0
- 2: Initialise state vector \mathbf{x}_s
- 3: $iter \leftarrow iter + 1$
- 4: Perform individual actions: evaluate objective vector with Algorithm 2
- 5: Evaluate dominance and Tchebycheff scalarisation of the population and of the archive
- 6: Archive the non dominated solutions in A_g
- 7: $\mathbf{p} \leftarrow \mathbf{p}^*$, $\mathbf{x}_s \leftarrow \mathbf{x}_s^*$, $P^* \leftarrow P_{iter}$
- 8: Perform social actions: evaluate objective vector with Algorithm 2
- 9: Evaluate dominance and Tchebycheff scalarisation of the population and of the archive
- 10: Archive the non dominated solutions in A_g
- 11: $\mathbf{p} \leftarrow \mathbf{p}^*$, $\mathbf{x}_s \leftarrow \mathbf{x}_s^*$, $P_{iter+1} \leftarrow P^*$
- 12: **if** $fun_evals < max_fun_eval$ **then**
- 13: **if** $mod(iter, freq_refinement) == 0$ **then**
- 14: Perform Pascoletti-Serafini refinement.
- 15: **end if**
- 16: GoTo 3
- 17: **end if**

Algorithm 2 Inner level, Bi-level formulation

- 1: Initialise states and controls with \mathbf{x}_s and \mathbf{p}^c from outer level
- 2: Solve constraint equations and return $(\mathbf{x}^*, \mathbf{p}^*)$
- 3: **if** $(\mathbf{x}^*, \mathbf{p}^*)$ feasible **then**
- 4: Return $(\mathbf{x}^*, \mathbf{p}^*)$, $\tilde{\mathbf{J}}(\mathbf{x}^*, \mathbf{p}^*)$
- 5: **else**
- 6: Return $(\mathbf{x}^*, \mathbf{p}^*)$, $\tilde{\mathbf{J}} = \mathbf{L} + \mathbf{k} \cdot \max |\mathbf{c}|$
- 7: **end if**

and solved with a two level solution algorithm. Problem (11) defines two different optimisation problems at two different levels. The outer level handles the objective vector $\tilde{\mathbf{J}}$ and the control parameter \mathbf{p}^* . The inner level finds the state and control vectors \mathbf{x}^* and \mathbf{p}^* that satisfy the constraints \mathbf{c} and minimise an inner cost function f . In the implementation proposed in this paper the inner function is $f = 1$, therefore the inner level is only solving the constraints while the out level only handles the minimisation of the objective vector.

III.A. Solution Approach

The overall solution strategy to solve Problem (11) is summarised in Algorithms 1 and 2. MACS generates trial static and dynamic control parameters \mathbf{p}^c with a set of individual and social actions (Lines 4 and 8 in Algorithm 1). For each trial vector Algorithm 2 is invoked to attempt a solution of the constraint functions \mathbf{c} . If the candidate vector \mathbf{p}^c converges to a feasible solution $(\mathbf{x}^*, \mathbf{p}^*)$, the inner level returns the corresponding values $\tilde{\mathbf{J}}(\mathbf{x}^*, \mathbf{p}^*)$ for the objective functions, together with the feasible vector $(\mathbf{x}^*, \mathbf{p}^*)$. If the inner level does not lead to a convergence, a penalty vector is returned to the outer level together with the solution vector $(\mathbf{x}^*, \mathbf{p}^*)$ at the last iteration of the NLP. The penalty solution returned is a vector L large enough that any non feasible solution will surely be worse than the objective function of any feasible solution. This value is further penalised by a constant vector \mathbf{k} , multiplying the feasibility value at the last iteration returned by the local optimiser. After a candidate set of solutions is generated with the individual actions, the global archive A_g is updated with the non-dominated solutions. The same happens after the social actions have generated a further set of candidate solutions. For more details on the archiving strategy and the generation of individual and social actions, the interested reader can refer to Ricciardi and Vasile 2015.⁶

If all the solutions are infeasible the objective vector contains the constraint violation and a pure feasibility problem is solved until an initial set of feasible solutions is available. As an additional step to refine the solution and guarantee local optimality, every fixed number of iterations and at the last one, a set of a single objective problems arising from the Pascoletti-Serafini scalarisation of Problem (10) will be solved with a gradient based approach.

III.A.1. Pascoletti-Serafini refinement

The refinement of the solution requires the reformulation of problem (10) in a scalar form. Here we propose the use of a normalised version of the Pascoletti-Serafini scalarisation, also known as goal attainment method. Using the Pascoletti-Serafini scalarisation, for each agent, Eq. (10) is transcribed into:

$$\begin{aligned} \min_{\alpha > 0} \quad & \alpha \\ \text{s.t.} \quad & \\ & \lambda_{ij} \vartheta_{ij}(\bar{\mathbf{x}}, \bar{\mathbf{p}}) \leq \alpha \quad i = 1, \dots, m \\ & \mathbf{c}(\bar{\mathbf{x}}, \bar{\mathbf{p}}) \geq 0 \end{aligned} \quad (12)$$

where λ_j is the vector of Tchebycheff weights associated to agent j , ϑ_{ij} is the i^{th} component of the rescaled objective vector of the j^{th} agent and α is a slack variable. This reformulation of the problem is constraining the agent's the move, in criteria space, within the descent cone defined by the point $\alpha \mathbf{d}_j + \zeta_j$ along the direction $\mathbf{d}_j = (1/\lambda_{1j}, \dots, 1/\lambda_{ij}, \dots, 1/\lambda_{mj})$. The rescaled objective vector is

$$\vartheta_{ij}(\bar{\mathbf{x}}, \bar{\mathbf{p}}) = \frac{\tilde{J}_{ij}(\bar{\mathbf{x}}, \bar{\mathbf{p}}) - \tilde{z}_i}{z_{ij}^* - \tilde{z}_i} \quad i = 1, \dots, m \quad (13)$$

where \mathbf{z}_j^* is equal to $\tilde{\mathbf{J}}_j(\bar{\mathbf{x}}, \bar{\mathbf{p}}^c)$ and $(\bar{\mathbf{x}}, \bar{\mathbf{p}}^c)$ is the initial guess for the solution of (12). This way the components of $\vartheta_j(\bar{\mathbf{x}}, \bar{\mathbf{p}})$ have value 1 at the beginning of the local search and if the agent converges to the utopia point $\tilde{\mathbf{z}}$, the components of $\vartheta_j(\bar{\mathbf{x}}, \bar{\mathbf{p}})$ become all equal to 0. The choice of λ_j and $\tilde{\mathbf{z}}$ will be discussed in the following subsection. From the normalisation one can derive the components of the vector ζ_j :

$$\zeta_{ij} = \frac{z_i}{z_{ij}^* - \tilde{z}_i} \quad i = 1, \dots, m \quad (14)$$

The presence of the rescaling of the objectives, together with the choice of λ_j and $\tilde{\mathbf{z}}_j$, are the elements that distinguish the proposed approach from the one given in.³ Note that solving problem (12) already provides a non-dominated solution that can be potentially inserted in A_g and used to update P_k . Note that the size of the neighbourhood B_j is given by the parameter ρ_j . The point $(\bar{\mathbf{x}}, \bar{\mathbf{p}}^c)$ is taken at random in B_j if $(\bar{\mathbf{x}}, \bar{\mathbf{p}})_j$ did not change from the previous iteration, otherwise $(\bar{\mathbf{x}}, \bar{\mathbf{p}}^c) = (\bar{\mathbf{x}}, \bar{\mathbf{p}})_j$. If the local search returns an infeasible solution a penalty value \mathbf{L} plus k times the violation of the constraints is assigned to all cost functions.

III.A.2. Selection of λ and $\tilde{\mathbf{z}}$

In Kaya & Maurer,³ the MOOCP was tackled by first solving each of the two individual objectives and then choosing a set of evenly spaced weights, obtaining a set of directions \mathbf{d} . This approach has a two main limitations: first, since only a local strategy was employed, there is the possibility that the extreme values of the Pareto front generated are on a local Pareto front. Second, that the approach is not easy to generalise for more than two objectives: as already stated by its authors regarding the extension to the three objective case that "it is well known that the boundary of the Pareto front may very well lie outside the triangle formed by the three points".³ The proposed approach instead consists in assigning vector $\lambda_i = (0, 0, i, \dots, 0, 0)$ to agents solving subproblem i and vector $\lambda_j = \frac{(1, 1, 1, \dots, 1)}{\|(1, 1, 1, \dots, 1)\|}$ to all the other agents. The modified utopia point $\tilde{\mathbf{z}}$ is given by

$$\tilde{\mathbf{z}} = 2\mathbf{z} - \mathbf{z}_A^* \quad (15)$$

where \mathbf{z} and \mathbf{z}_A^* are respectively the utopia and nadir points of the current approximation to the Pareto front that is contained in the archive A_g .

IV. Case Studies

The multi-objective optimisation approach was applied to two case studies: a modified Goddard rocket problem and the upper, rocket-based ascent of a spaceplane to a low Earth orbit. The following section describes the system models used for the trajectory dynamics and control, including vehicle aerodynamics and propulsion models.

IV.A. Case study 1: Modified Goddard Rocket Problem

A multi-objective extension of the Goddard Rocket Problem was already presented in.¹⁸ In this work we extend the physical model to include aerodynamic drag, variation of air density with altitude and variation of mass due to fuel consumption. The thrusting magnitude is kept constant, but the rocket direction can be controlled by changing the thrusting angle u . The differential equations of motions are thus:

$$\dot{x} = v_x \quad (16a)$$

$$\dot{v}_x = \frac{T \cos(u) - \frac{1}{2} \rho_0 e^{-y} S_{ref} C_d (v_x^2 + v_y^2)^{0.5} v_x}{m} \quad (16b)$$

$$\dot{y} = v_y \quad (16c)$$

$$\dot{v}_y = \frac{T \sin(u) - \frac{1}{2} \rho_0 e^{-y} S_{ref} C_d (v_x^2 + v_y^2)^{0.5} v_y}{m} - g_0 \quad (16d)$$

$$\dot{m} = -\frac{T}{g_0 I_{sp}} \quad (16e)$$

where g_0 is the acceleration due to gravity (assumed to be constant), T is the thrust magnitude, ρ_0 is the air density at ground level, S_{ref} is the aerodynamic reference surface area, C_d is the drag coefficient related to that surface, x and y are the components of the position vector, v_x and v_y the components of the velocity vector, m is the current mass of the rocket, I_{sp} is the specific impulse of the rocket and u is the control, i.e., the thrusting angle. To keep consistency with previous literature models, the values of the constants are: $g_0 = 0.0016$ and $T = 0.004$, while for the other parameters the values $\rho_0 = 0.1$, $S = 1$, $C_d = 1$ and $I_{sp} = 937.5$ were employed.

At the initial time $t_0 = 0$, the launcher is at rest with a prescribed total mass \bar{m} :

$$\begin{aligned} x(t_0) &= 0 \\ v_x(t_0) &= 0 \\ y(t_0) &= 0 \\ v_y(t_0) &= 0 \\ m(t_0) &= \bar{m} = 1 \end{aligned} \quad (17)$$

At the final time t_f , the rocket has to be at a prescribed altitude \bar{h} with zero vertical velocity, irrespective of the horizontal velocity or position.

$$\begin{aligned} y(t_f) &= \bar{h} = 10 \\ v_y(t_f) &= 0 \end{aligned} \quad (18)$$

The objectives are the minimisation of mission duration and the maximisation of the final horizontal velocity. As can be seen from Eq. (16e), the fuel consumption rate depends only on thrust magnitude, gravity at sea level and specific impulse, all of which are constant. The minimisation of mission time for a given initial mass is therefore equivalent to the the minimisation of total fuel consumption or the maximisation of the payload at the final time.

$$\min_{t_f, u} [J_1, J_2]^T = [t_f, -v_x(t_f)]^T \quad (19)$$

IV.B. Case study 2: Spaceplane rocket-based ascent problem

The second case study looks at a more complex problem of the ascent trajectory of a single-stage-to-orbit spaceplane. The vehicle design employs a hybrid propulsion system that uses an air-breathing engine cycle for lower atmospheric flight, and switches to a rocket engine cycle at higher altitudes when the atmospheric density is reduced, meaning a decrease in the lift force, and in the efficiency of the air-breathing engine. For the test case here, only the rocket-based ascent flight segment is optimised assuming a planar trajectory contained in the equatorial plane.

The Earth is modelled as a perfect sphere of radius $R_E = 6378.137$ km with rotational velocity $\omega_E = 7.292115 \times 10^{-5}$ rad/s. The gravitational acceleration varies with the altitude h , $g = g_0 R_E^2 (h + R_E)^{-2}$. The atmospheric pressure p and density ρ are modelled using the International Standard Atmosphere model. The dynamic model uses a 3 DOF translational dynamics for a variable-mass point in the rotating Earth-centered

Earth-fixed (ECEF) reference frame. The forces modelled include gravity, aerodynamic lift and drag, and propulsive thrust. The equations of motion are given for the translational position $\mathbf{r} = [h, \lambda, \theta]^T$ and velocity $\mathbf{v} = [v, \gamma, \chi]^T$ vectors in spherical coordinates.¹⁹

$$\dot{h} = \dot{r} = v \sin \gamma \quad (20a)$$

$$\dot{\lambda} = \frac{v \cos \gamma \sin \chi}{r} \quad (20b)$$

$$\dot{\theta} = \frac{v \cos \gamma \cos \chi}{r \cos \lambda} \quad (20c)$$

$$\dot{v} = \frac{T \cos(\alpha) - D}{m} - g \sin \gamma + \omega_E^2 r \cos \lambda (\sin \gamma \cos \lambda - \cos \gamma \sin \chi \sin \lambda) \quad (20d)$$

$$\dot{\gamma} = \frac{T \sin(\alpha) + L}{mv} \cos \mu - \left(\frac{g}{v} - \frac{v}{r} \right) \cos \gamma + 2\omega_E \cos \chi \cos \lambda + \omega_E^2 \left(\frac{r}{v} \right) \cos \lambda (\sin \chi \sin \gamma \sin \lambda + \cos \gamma \cos \lambda) \quad (20e)$$

$$\dot{\chi} = \frac{L}{mv \cos \gamma} \sin \mu - \left(\frac{v}{r} \right) \cos \gamma \cos \chi \tan \lambda + 2\omega_E (\sin \chi \cos \lambda \tan \gamma - \sin \lambda) - \omega_E^2 \left(\frac{r}{v \cos \gamma} \right) \cos \lambda \sin \gamma \cos \chi \quad (20f)$$

The control vector is $\mathbf{u} = [\alpha, \mu, \delta_T]^T$ which controls the magnitude and direction of the thrust vector, due to the propulsion system, acting on the centre of mass of the vehicle. The throttle control $\delta_T \in [0, 1]$ dictates the fraction of maximum available thrust applied, which in turn determines the required propellant mass flow. The vehicle mass is $m = m_0 + m_{payload} + m_p$. As the only time-varying element is the propellant mass, the time derivative of the vehicle mass is $\dot{m} = -\dot{m}_p$.

The propulsion system is a standard rocket engine, with a calculated maximum available thrust at a given altitude $T(h)$ and a total mass flow rate for the propellants \dot{m}_p .

$$T = \delta_T (T_{vac} + p A_{exit}) \quad (21a)$$

$$\dot{m}_p = \frac{\delta_T T_{vac}}{g_0 I_{sp}} \quad (21b)$$

The configuration used in the test case assumes LOX/LH2 rocket engines with an I_{sp} of 450 s. The fuel mixture ratio is assumed constant and the two propellants are treated as a single mass flow. The maximum total thrust in a vacuum is set at $T_{vac} = 4$ MN. A penalty proportional to atmospheric pressure $p(h)$ and nozzle exit area A_{exit} is added to account for the change in thrust within atmosphere (due to nozzle expansion under pressure compared to in a vacuum).

The aerodynamic model of the vehicle predicts the total coefficients of lift C_L and drag C_d as a function of the angle of attack α and Mach number $M(h, v)$. Data sets were used to fit a set of polynomial-based surrogate models for C_L and C_d terms, as shown in Fig. 1.

The trajectory segment under test starts from a nominal transition point from air-breathing to rocket mode, and ascends until the vehicle reaches a 100 km circular, equatorial orbit. As the orbit is planar, contained within the orbital plane, the out-of-plane control $\mu = 0$. The initial and final boundary conditions on the problem are therefore,

$$\begin{aligned} h(t_0) &= 28.0 \text{ km} \\ \lambda(t_0) &= 0^\circ \text{N} \\ \theta(t_0) &= 0^\circ \text{E} \\ v(t_0) &= 1502 \text{ m/s} \\ \gamma(t_0) &= 10^\circ \\ \chi(t_0) &= 90^\circ \text{ (heading due east)} \\ m(t_0) &= 200000 \text{ kg} \end{aligned} \quad (22)$$

$$\begin{aligned} h(t_f) &= 100 \text{ km} \\ v(t_f) &= 7381 \text{ m/s} \\ \gamma(t_f) &= 0^\circ \end{aligned} \quad (23)$$

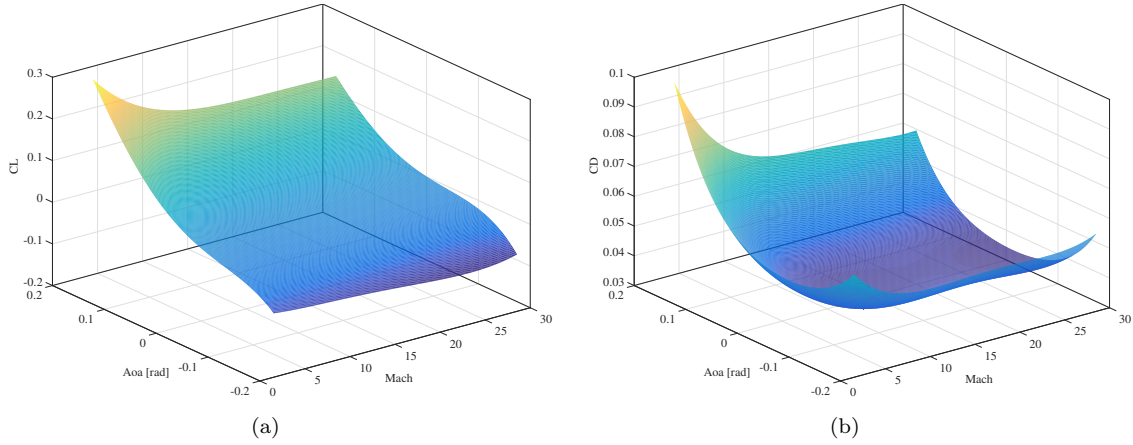


Figure 1. Aerodynamic coefficients of lift and drag over the super- and hypersonic regimes.

The objective functions are to maximise the payload mass into orbit by minimising the mass of the on-board fuel required, and minimising the integral over time of the fourth power of the axial accelerations experienced by the vehicle. The second objective has been formulated to heavily penalise high values of the loads while keeping a continuous and differentiable objective function.

$$\min_{t_f, u} [J_1, J_2]^T = \left[-m_f, \int_{t_0}^{t_f} \left(\frac{(T - D) \cos(\alpha) + L \sin(\alpha) - \frac{\mu}{r^2} \sin(\gamma + \alpha)}{m} \right)^4 dt \right] \quad (24)$$

V. Results

For Case Study 1, the problem in Eqs. (16)–(19) was solved using 4 elements of order 6 in the DFET transcription. The final time is an optimisation variable bounded between 100 s and 250 s. The control angle is bounded between $-\pi/2$ and $\pi/2$. In total there are 29 optimisation variables, only one of which, final time, is static. The problem was solved first with the bi-level approach only, followed by a bi-level plus an additional single level refinement. For the cheaper bi-level only approach, 30 independent runs were performed to evaluate the repeatability of the algorithm.

Results for the bi-level approach only are given in Fig. 2 showing the non dominated Pareto front of all 30 independent runs of MACS, and some sample trajectories and control histories. As it can be seen, the current approach can generate a well spread set of trade-off control laws. The solutions with intermediate time and final horizontal velocity appear to be rather smooth even if no gradient based technique was employed in the optimisation. The solution with minimum time appears to be close to an impulsive bang-bang solution, though a further investigation with h-p adaptivity would be required to understand exactly if that is the case or not. The solution with maximum time is not well resolved in the first element, indicating that more function evaluations or better search strategies are needed, and that the solution does not lie on the true Pareto front. However, the bi-level approach aims at the initial investigation of the trade-off between the various objectives and the control laws required to achieve those trade-off points. In Case Study 1, the overall trend of the control laws and how they impact the various objectives is clear: as expected, shorter times lead to lower final horizontal velocities. Overall, these solutions are very similar to the solutions reported in Ricciardi et al.,¹⁸ where a simpler version of this problem was solved.

Figure 3 shows analogous results for the same problem, but with the additional single level refinement, which was performed every 20 iterations of MACS. For this refinement, a maximum of 1000 evaluations of the objectives was allowed. As it is evident, the gradient based single level refinement is able to improve the quality of the solutions, removing all oscillations from the control laws except for the ones present in the steep region for point 1. As already pointed out by Vasile and Ricciardi,^{5,18} h-p refinement is required to accurately represent that steep variation.

For case study 2, the problem in Eqs. (20)–(24) was solved using again 4 elements of order 6 in the DFET transcription. The final time is an optimisation variable bounded between 30 s and 1200 s. The control angle was bounded between -10° and 10° , while throttle was bounded between 0 and 1. In total

there are 57 optimisation variables, one of which, final time, is static. The problem was solved directly with the bi-level plus additional single level refinement with a maximum of 1000 evaluations of the objectives for each call of *fmincon*. A comparison with the two single objective optimisations is shown in the results.

Figure 4 shows the Pareto front obtained with 50000 objectives evaluation. The extreme points of the front (red crosses) are used to perform the comparison with the single objective solutions obtained directly with *fmincon* (black stars). As evident from the graph, the multi-objective approach did not converge to the same value of the single objective for the first objective, indicating that more function evaluations were needed. On the other hand, the single objective solution of the second problem is dominated by the Pareto front computed by the multi-objective approach, indicating that the local optimiser stopped at a local minimum of the function which was not also a global minimum. To overcome this limitation, many attempts with different initial guesses would be needed, but the global approach here proposed performs these attempts automatically.

A comparison of some interesting quantities of the solutions of the extreme points of the front and of the single objective approach is shown in Figure 5. For the maximum payload case, the solution found by the multi-objective approach has a shorter duration and a slightly lower final mass, but much lower loads over time. For the minimum loads case, instead, the multi objective approach gives longer times and lower final mass, but again much lower loads over time. Interestingly, the solution found by the multi-objective optimiser has a dip in altitude at the beginning of the manoeuvre, which allows the spaceplane to gain velocity while saving propellant.

VI. Conclusions

This paper described a novel approach to the solution of multi-objective optimal control problems, transcribing the problem using a Direct Finite Elements in Time approach followed by a modified memetic Multi Agent Collaborative Search optimisation strategy. A bi-level NLP formulation was used, with and without single-level refinement. Results from two test cases for the ascent trajectory optimisation show the benefits of this approach: for the modified Goddard rocket problem it found a well defined Pareto front, and the control laws associated to each solution are also locally optimal. For the spaceplane case it found different and higher performance solutions with respect to a single objective optimisation, but did not converge to the same solution found by the pure gradient based single objective approach. Possible explanations for this could lie in the low number of evaluations of objectives in the single level refinement, or in the insufficient overall number of function evaluations. Nevertheless it shows a good trade-off between deliverable payload and maximum axial loads.

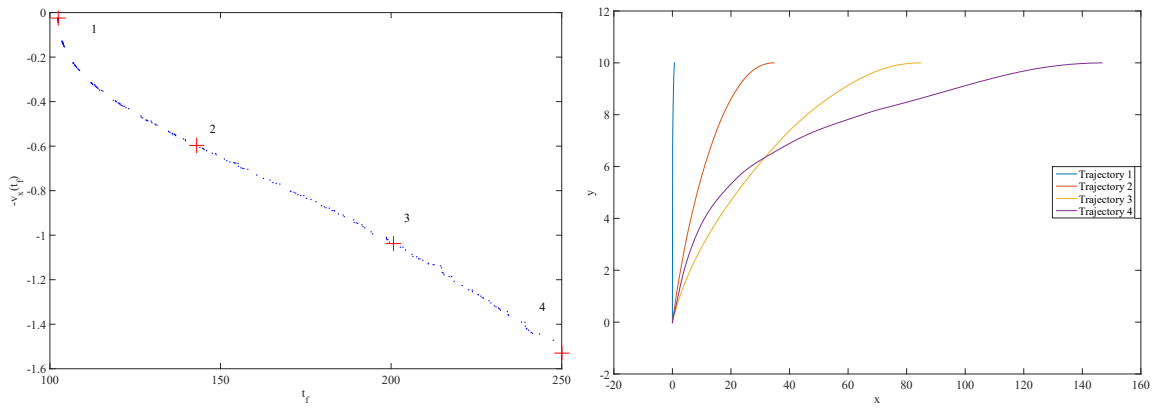
An interesting point to note from the first case study is that even though the purely evolutionary bi-level approach gives lower quality control laws with respect to the combined bi-level and single level refinement, the Pareto front obtained by the bi-level approach is practically indistinguishable from the one generated by the combined approach. Possible interesting future research directions could be taken to maximally exploit the cheaper evolutionary bi-level approach, and call the gradient based approach only when the improvement of the Pareto front becomes too slow. In addition, it could be interesting to find ways to make the single level approach computationally cheaper.

Acknowledgments

The research was partially funded by an ESA NPI grant (ref TEC-ECN-SoW-20140806) and Airbus Defence & Space.

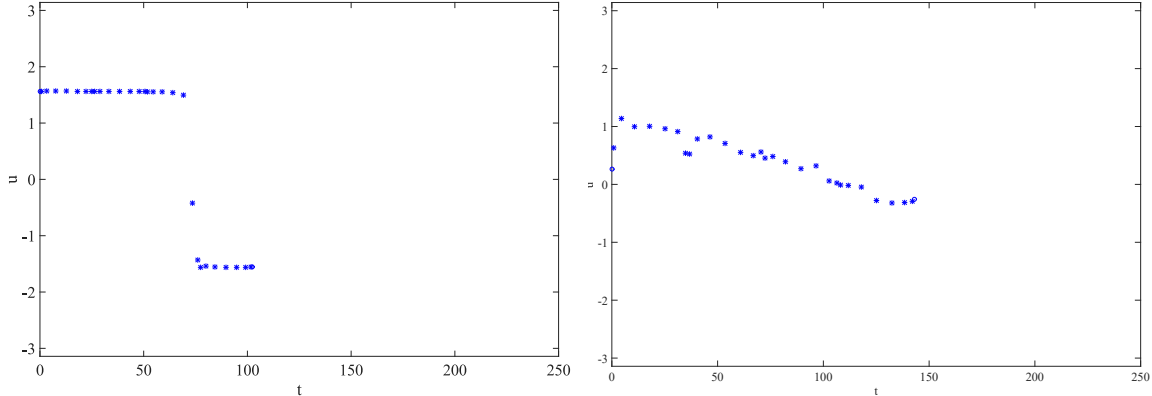
References

- ¹Coverstone-Carroll, V., Hartmann, J., and Mason, W., “Optimal multi-objective low-thrust spacecraft trajectories,” *Computer methods in applied mechanics and engineering*, Vol. 186, No. 2, 2000, pp. 387–402.
- ²Ober-Blobaum, S., Ringkamp, M., and zum Felde, G., “Solving multiobjective optimal control problems in space mission design using discrete mechanics and reference point techniques,” *IEEE Annual Conference on Decision and Control*, IEEE, 2012, pp. 5711–5716.
- ³Kaya, C. Y. and Maurer, H., “A numerical method for nonconvex multi-objective optimal control problems,” *Computational Optimization and Applications*, Vol. 57, No. 3, 2014, pp. 685–702.
- ⁴Englander, J. A. and Vavrina, M. A., “Multi-Objective Hybrid Optimal Control for Multiple-Flyby Interplanetary Mission



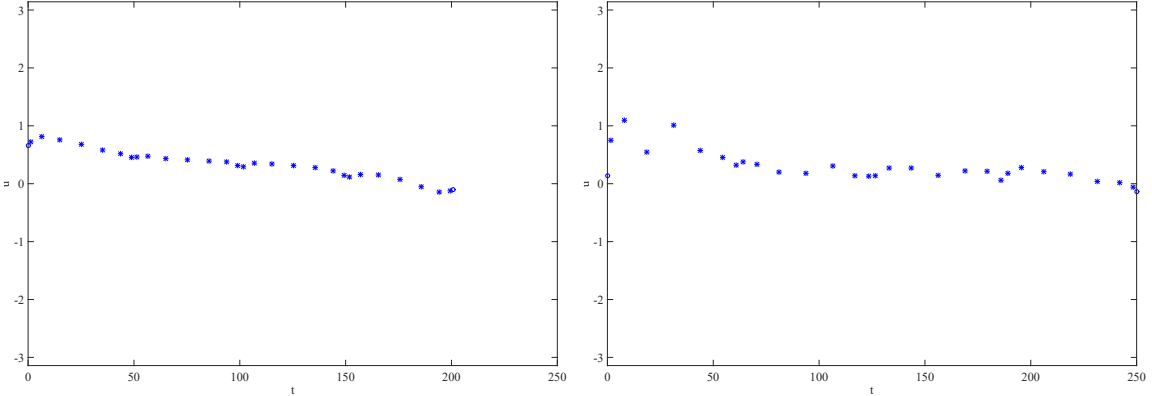
(a) Pareto front, with the representative points denoted by a red +

(b) Trajectory position of x vs y



(c) Control law for point 1 (top left + in Figure 1a)

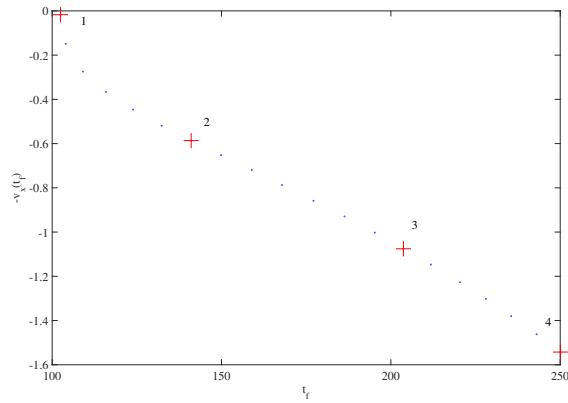
(d) Control law for point 2



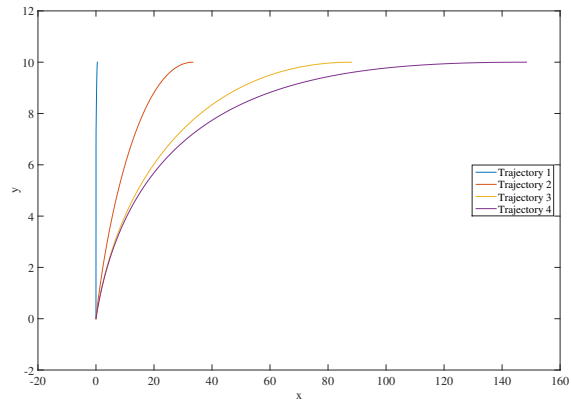
(e) Control law for point 3

(f) Control law for point 4 (bottom right + in Figure 1a)

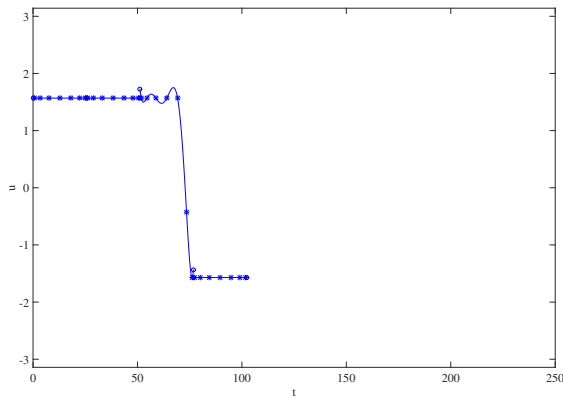
Figure 2. Pareto front and trajectories for 4 representative points and their respective control profiles for bi-level only approach. Dots represent the control nodes in the DFET transcription.



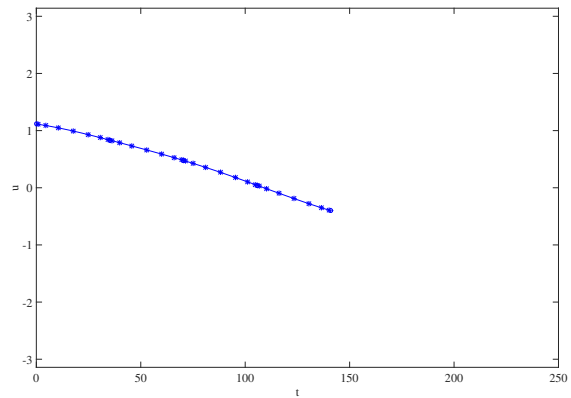
(a) Pareto front, with the representative points denoted by a red +



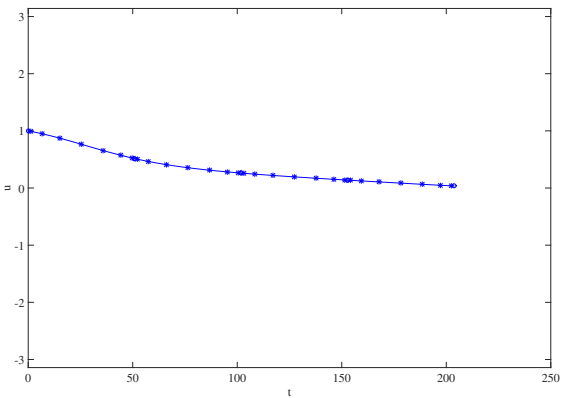
(b) Trajectory position of x vs y



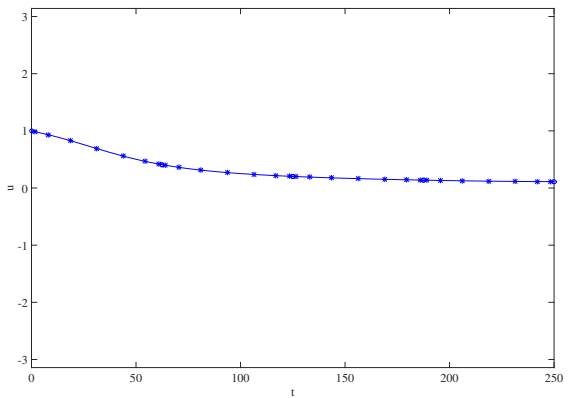
(c) Control law for point 1 (top left + in Figure 1a)



(d) Control law for point 2



(e) Control law for point 3



(f) Control law for point 4 (bottom right + in Figure 1a)

Figure 3. Pareto front and trajectories for 4 representative points and their respective control profiles for single level refinement approach

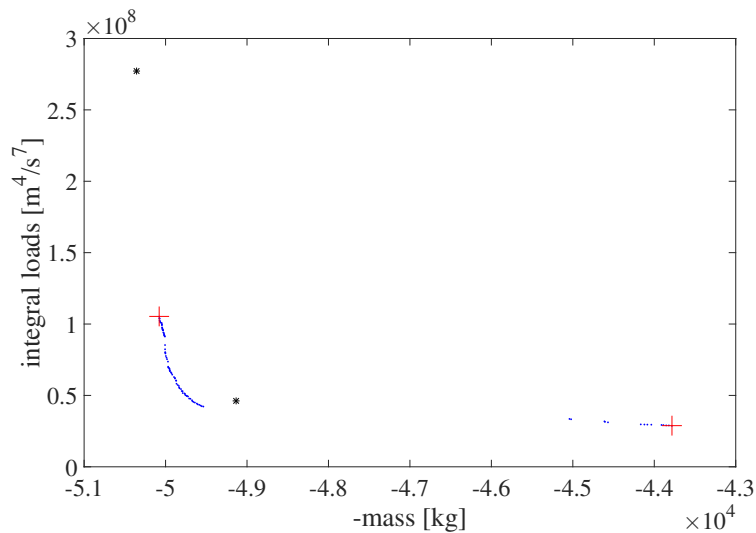


Figure 4. Pareto front for the spaceplane rocket problem. Non dominated solutions for multi-objective algorithm (blue dots), extreme values (red crosses), and comparison with single objective solutions (black stars)

Design Using Chemical Propulsion,” 2015.

⁵Vasile, M., “Finite Elements in Time: A Direct Transcription Method for Optimal Control Problems,” *AIAA/AAS Astrodynamics Specialist Conference*, 2010.

⁶Ricciardi, L. A. and Vasile, M., “Improved archiving and search strategies for Multi Agent Collaborative Search,” *International Conference on Evolutionary and Deterministic Methods for Design, Optimization and Control with Applications to Industrial and Societal Problems (EUROGEN)*, 2015.

⁷Vasile, M. and Finzi, A. E., “Direct lunar descent optimisation by finite elements in time approach,” *International Journal of Mechanics and Control*, Vol. 1, No. 1, 2000.

⁸Vasile, M. and Bernelli-Zazzera, F., “Optimizing low-thrust and gravity assist maneuvers to design interplanetary trajectories,” *Journal of the Astronautical Sciences*, Vol. 51, No. arXiv: 1105.1829, 2011, pp. 13–35.

⁹Vasile, M. and Bernelli-Zazzera, F., “Targeting a heliocentric orbit combining low-thrust propulsion and gravity assist manoeuvres,” *Operational Research in Space & Air*, Vol. 79, 2003.

¹⁰Bottasso, C. L. and Ragazzi, A., “Finite element and runge-kutta methods for boundary-value and optimal control problems,” *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 4, 2000, pp. 749–751.

¹¹Goddard, R. H., “A Method of Reaching Extreme Altitudes,” *Nature*, Vol. 105, 1920, pp. 809–811.

¹²Munick, H., “Goddard problem with bounded thrust,” *AIAA Journal*, Vol. 3, No. 7, 1965, pp. 1283–1285.

¹³Tsiotras, P. and Kelley, H. J., “Goddard problem with constrained time of flight,” *Journal of guidance, control, and dynamics*, Vol. 15, No. 2, 1992, pp. 289–296.

¹⁴Seywald, H. and Cliff, E. M., “Goddard problem in presence of a dynamic pressure limit,” *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 4, 1993, pp. 776–781.

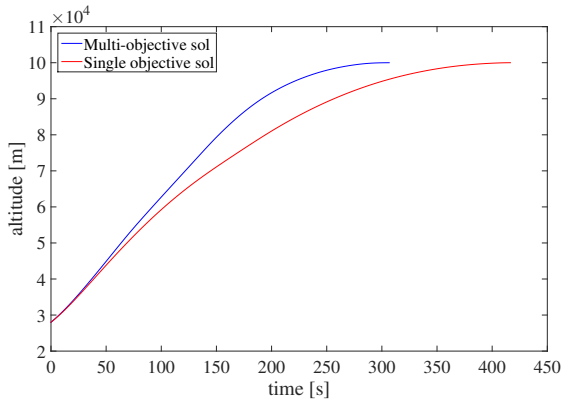
¹⁵Graichen, K. and Petit, N., “Solving the Goddard problem with thrust and dynamic pressure constraints using saturation functions,” *World Congress of The International Federation of Automatic Control, IFAC World Congress*, 2008.

¹⁶Pagano, A. and Mooij, E., *Global launcher trajectory optimization for lunar base settlement*, American Institute of Aeronautics and Astronautics (AIAA), 2010.

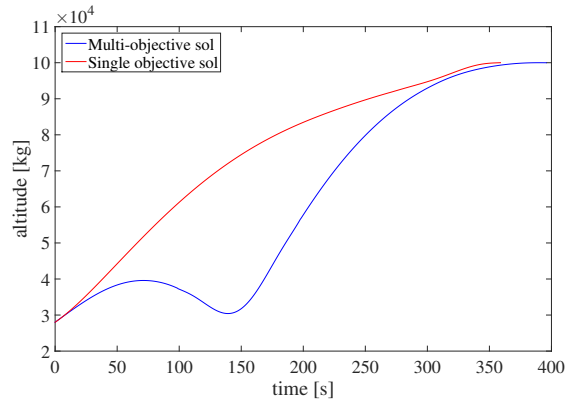
¹⁷Hodges, D. H. and Bless, R. R., “Weak Hamiltonian finite element method for optimal control problems,” *Journal of Guidance, Control, and Dynamics*, Vol. 14, No. 1, 1991, pp. 148–156.

¹⁸Ricciardi, L. A. and Vasile, M., “Global Solution of Multi-objective Optimal Control Problems with Multi Agent Collaborative Search and Direct Finite Elements Transcription,” *IEEE World Congress on Computational Intelligence, IEEE Congress on Evolutionary Computation*, 2016.

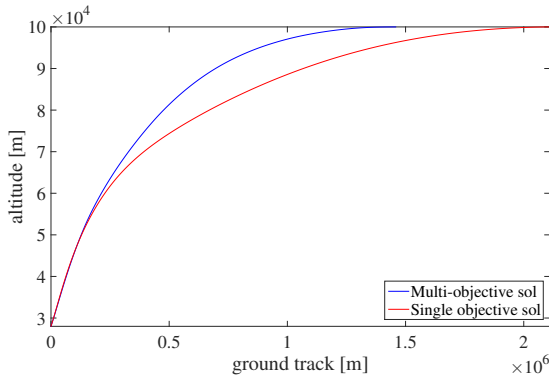
¹⁹Vinh, N., *Optimal Trajectories in Atmospheric Flight*, Elsevier, New York, 1981.



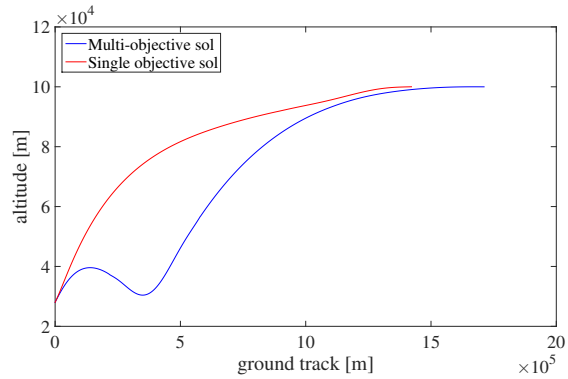
(a) Altitude vs time, maximum payload solutions



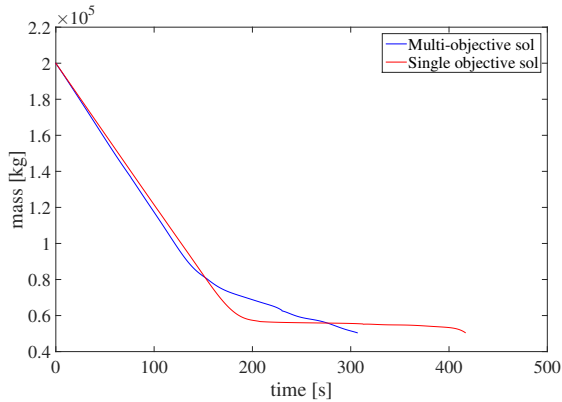
(b) Altitude vs time, minimum loads solutions



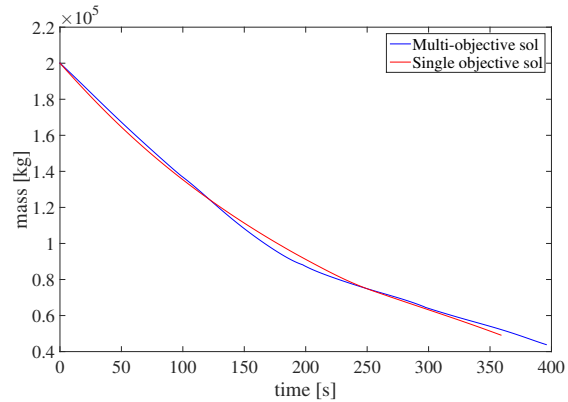
(c) Ground track vs altitude, maximum payload solutions



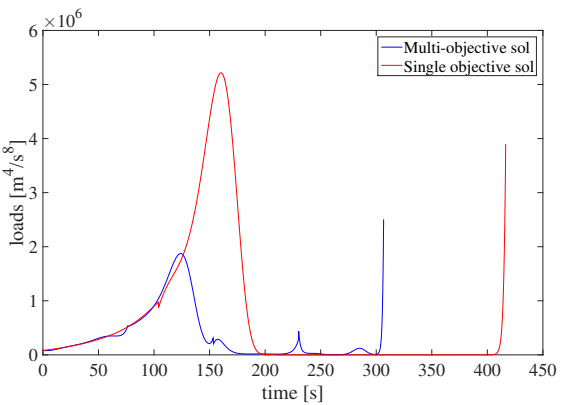
(d) Ground track vs altitude, maximum payload solutions



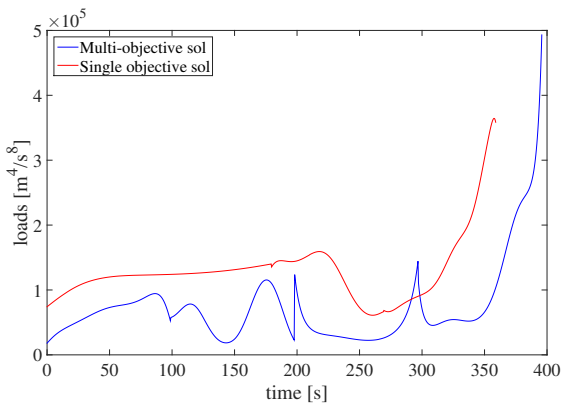
(e) Mass vs time, maximum payload solutions



(f) Mass vs time, minimum loads solutions



(g) Loads vs time, maximum payload solutions



(h) Loads vs time, minimum loads solutions

Figure 5. Comparison of altitude, ground track, mass variation and loads for the two extreme values between single objective and multi-objective solution