JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 48, No. 1, January 2025

Solving Multi-Objective Optimal Control Problems Using a Multiresolution Approach

Ben Parsonage* and Christie Maddock†
Department of Mechanical and Aerospace Engineering, University of Strathclyde, Glasgow, United Kingdom

https://doi.org/10.2514/1.G008223

This paper presents an adaptive multiresolution strategy for multi-objective optimal control problems. The optimal control problem is solved using a direct approach, with individualistic grid adaptation facilitated by a local error analysis at element boundaries. Multiple objectives are considered using a dominance-based approach applying both local and global search methods to a collaborative population of unique solutions. These aspects are simultaneously incorporated via a novel application of evolutionary algorithms for adaptive optimal control problems. Together, this avoids the need for a priori specification of the quantity and temporal location of element boundaries and the set of scalarization weights defining the multi-objective descent directions. Solution fidelity can thus increase concurrently with the exploration of the design space, which leads to increased numerical efficiency while propagating and maintaining population diversity. The benefits of the proposed approach over traditional uniform-grid implementations are demonstrated. Results show that the multiresolution approach is capable of striking an effective balance between solution fidelity, population diversity, and computational cost unachievable using uniform grids.

I. Introduction

PARTICULARLY within the field of aerospace engineering, many multidisciplinary design optimization (MDO) problems include trajectory optimization and control as a primary discipline. Such problems can be formulated as optimal control (OC) problems, describing systems where overall performance is comparably dependent on the time-dependent (dynamic) control law by which it must operate as it is on time-independent (static) system design parameters.

Solving OC problems can become prohibitively expensive as complexity increases. Most practical examples include some number of state constraints, control constraints, and equality/inequality boundary and/or path constraints. As a consequence, solutions may exhibit high nonlinearity or discontinuities in state/control profiles and/or their respective higher-order derivatives [1]. Furthermore, when control variables are represented as linear functions within the system differential equations, bang-bang solutions can often be expected [2]. Considering direct transcription methods, solution accuracy can therefore be highly dependent on the grid resolution of the discretization nodes at which the system state and control variables are evaluated. If the constraint functions are to be satisfied to within a strict tolerance (as is often the case in MDO), a high-resolution (dense) grid is typically required to capture this difficult behavior. However, the use of dense grids may necessitate a high computational budget given the resulting size of the nonlinear programming (NLP) problem and the associated augmentation of search space dimensionality. Such a strategy may therefore remain impractical for most applications, especially if the NLP problem is not sparse [2,3]. Furthermore, as the number of grid points increases, the chosen solver may become ill-conditioned and exhibit poorer convergence characteristics, ultimately failing to acceptably converge at all [1].

Reducing the cost of OC problems can be approached via a progressive nonuniform refinement of the grid resolution [2–7]. This entails the adaptive addition, subtraction, or redistribution of discretization nodes to concentrate the grid around nonsmooth regions (or, particular areas of the solution that dominate the overall accuracy), while smooth regions remain relatively sparse.[‡] In this manner, a multiresolution grid may capture highly nonlinear or discontinuous behavior within a significantly reduced computational budget.

Zhao and associates [1,8,9] present a multiresolution approach for solving OC problems using direct collocation. Enabled by the work of Jain and Tsiotras [3], the algorithm may start from any uniform initial mesh with an odd number of grid points and refine the mesh at locations where the interpolation error remains outside a specific threshold. The resulting method is extended to generalized dyadic grids. In a related work, Zhao and Li [10] adapt and apply this method to optimize the descent trajectory of a Mars reentry vehicle with respect to various boundary and path constraints. A similar approach is proposed by Li et al. [11], where a mesh refinement procedure and trajectory optimization scheme are implemented in a nested, bilevel structure. In the inner level, a convex optimization algorithm locally solves the trajectory problem, with the solution subsequently passed to the outer level for grid refinement. This approach is applied to solve a trajectory planning problem for spacecraft relative motion.

Research in this area predominantly considers only single-objective problems, with methods for multi-objective problems far more scarce. This is likely due to one significant complicating factor: the orthogonal objective solutions may exhibit nonlinear/nonsmooth behavior within entirely different regions of the time domain. With reference to the prescribed grid resolution problem, this implies that to properly represent a multi-objective tradeoff in a numerically efficient manner, individual solutions of a Pareto-optimal set must be separately discretized upon their own unique, nonuniform grids. Some authors have attempted to navigate this complication through the application of scalarization methods [12], reducing the multi-objective problem into a (or series of) single-objective subproblem(s) [13–15]. Scalarization requires the specification of an overarching aggregate objective function (or dimensionless utility function) that contains the weighted contributions of each sub-objective (representing the relative importance attributed to each objective [16]). Accordingly, a single-weight vector produces a single solution, representing a certain level of compromise between sub-objectives. Solutions treated in isolation thus allow the simple incorporation of iterative mesh refinement schemes. To represent a tradeoff surface, however, a series of weights must be provided (as seen in [17]), and thus a series of separate refinement procedures must follow. This approach is seen in [13], where the authors use the Runge–Kutta discretization method to transcribe the problem into an NLP problem with

Received 22 January 2024; accepted for publication 1 August 2024; published online Open Access 18 October 2024.

*Ph.D. Researcher; ben.parsonage@strath.ac.uk (Corresponding Author).
†Senior Lecturer.

[‡]This can be considered a form of *h*-refinement, as opposed to adapting the order of the approximating polynomial (*p*-refinement).

solution states and controls defined upon refinable dyadic grids. The scalarized aggregate function is then solved using the local optimization method.

However, it may be difficult to assign appropriate values of weights a priori in order to give a well-spread representation of the Pareto front. Put simply, a uniformly spread set of weights may not result in a similarly uniform spread of solutions in objective space. Furthermore, a particular set of weights may entirely miss many interesting or relevant solutions or indeed give no feasible solutions at all. Ding et al. [18] attempt to circumvent this particular issue by replacing the weights-based approach with a physical programming method [19]. In this case, the overall cost function is determined by combining each of the four included objectives using a series of preference functions. Applying this method in conjunction with a *hp*-adaptive pseudospectral method allows the optimizer to produce a number of solutions, each defined on a unique nonuniform grid, equal to the number of defined preference functions. This approach then is still inherently limited in terms of the detail to which the Pareto front may be represented.

This paper continues the philosophy of this research, presenting a method of reducing the computational burden associated with multi-objective optimal control problems (MOCPs) in engineering design. The method focuses on adaptively increasing the resolution at which a set of discretized control solutions, each representing a particular balance between the problem objectives, may be evaluated. This is achieved by concentrating computational resources (in the form of discretization nodes/grid points) around areas of high complexity independently for each solution.

A novel aspect of the proposed approach is the simultaneous handling of various grid resolutions within an evolutionary framework, allowing solutions to independently evolve concurrent with their propagation through objective space. This is achieved through a unique social operator designed to facilitate collaborative action between solutions of variable vector lengths. A complete MOCP may thus be treated as an open-ended evolutionary design problem. This ultimately facilitates a more complete, higher-fidelity representation of the trade-off surface in a more numerically efficient manner. The advantages of this approach are summarized as follows:

1) Nonsmooth/discontinuous regions within each solution are automatically detected, and thus the a priori specification of both the number and distribution of discretization elements (as in [5,6]) is not required.

2) In reference to the previous point, this implies that each agent may be initialized with only one single element, representing a relatively small NLP and thus computationally inexpensive to evaluate, allowing greater initial global exploration capability. From this initialization, the progressive refinement of solutions thus implies greater numerical efficiency relative to employing uniformly discretized and/or refined solutions.

3) The criterion for deciding the region(s) in which to introduce new grid points is based on the relative local error evaluated at element boundaries. This avoids the need for any secondary optimization, as in [2,20,21].

4) Accordingly, this approach does not require any costly integrations, such as the highly accurate Romberg quadratures employed in [20], which can be particularly expensive for nonlinear dynamics [3].

5) Due to the multilevel nature of dyadic grid structures, existing nodes remain fixed across successive iterations, which benefits the overall convergence characteristics [3].

6) User specification of multi-objective scalarization weights or preference functions is not required, as a population of multiresolution solutions can be concurrently advanced toward the Pareto front via cooperative social actions designed to promote population diversity [22].

7) Solution-specific nonlinear/discontinuous behavior may be efficiently captured during an evolutionary process. This ultimately allows a more accurate representation of the multi-objective tradeoff for reduced computational effort relative to traditional uniform-grid approaches.

The remainder of this paper is organized as follows: Section II outlines the general formulation of the proposed methods, including the nonuniform grid refinement and a collaborative evolutionary operator capable of handling solutions of variable grid resolution. Section III presents an overview of the complete optimization process, incorporating each of the newly proposed methods within an established algorithm structure. Section IV presents the application of the proposed methods to two demonstrative MOCPs. Performance is analyzed with respect to computational cost and solution accuracy compared with a traditional uniform-grid approach. Section V discusses the presented results, and Sec. VI offers conclusions and recommendations.

II. Methodology

The following section details the general formulation of the proposed approach. This includes the nonuniform refinement procedure for discretized finite-element grids and a variable-length social operator that allows their incorporation into a global evolutionary algorithm.

A. Discretization

In this study, continuous OC problems are transcribed into finite-dimensional NLP problems using direct finite elements in time (DFET) [23] for the nonlinear system state and control functions. Bernstein polynomials of a fixed order n are employed as basis functions of the general form:

$$B_{v,n}(t) = \binom{n}{v} t^v (1-t)^{n-v} \quad 0 \le v \le n, 0 \le t \le 1 \tag{1}$$

where t is time. The reader is referred to [24] for a detailed description of this approach, including the practical advantages of Bernstein polynomials that support their use in this application.

B. Mesh Refinement

The collection of discretization nodes τ that represent the interface between each finite element D of a solution vector $\boldsymbol{x}$ over a normalized time interval may be considered a uniform dyadic mesh $\mathcal{V}_{j,N}$ conforming to the general form:

$$\mathcal{V}_{j,N} = \left\{ \tau_{j,k} \in [0,1] : \tau_{j,k} = \frac{k}{2^j N}, 0 \le k \le 2^j N, 0 \le j \le J_{\max} \right\} \tag{2}$$

where $\tau_{j,k}$ are the time coordinates of the interface nodes with corresponding spatial index k. The positive integers j and $J_{\max}$ represent the current and maximum allowable resolution level and N is the number of elements present in the initial uniform mesh discretization. By definition, a dyadic grid is obtained by successively subdividing a uniform initial grid such that $\mathcal{W}_{j,N}$ denotes the set of grid points belonging to $\mathcal{V}_{j+1,N}$ but not $\mathcal{V}_{j,N}$; i.e.,

$$\mathcal{W}_{j,N} = \left\{ \hat{\tau}_{j,k} \in [0,1] : \hat{\tau}_{j,k} = \frac{2k+1}{2^{j+1}N}, \; 0 \le k \le 2^j N - 1, 0 \le j \le (J_{\max} - 1) \right\} \tag{3}$$

Hence, $\tau_{j+1,k} \in \mathcal{V}_{j+1,N}$ if

$$\tau_{j+1,k} = \begin{cases} \tau_{j,k/2}, & k \text{ is even} \\ \hat{\tau}_{j,(k-1)/2} & \text{otherwise} \end{cases} \tag{4}$$

N can be any positive integer for the generalized dyadic grid, which is more convenient for optimization.[§] An example of a uniform dyadic grid with $N = 1$ and $J_{\max} = 7$ is shown in Fig. 1, where only the nodes corresponding to $\mathcal{V}_{0,N}$ and $\mathcal{W}_{j,N}, 0 \le j \le (J_{\max} - 1)$ have been included for clarity.

[§]Throughout this work, however, $N = 1$.

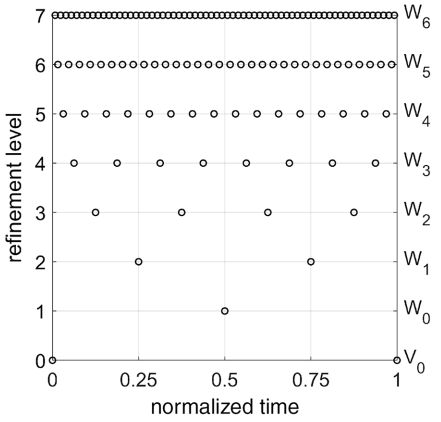

Fig. 1 A uniform dyadic grid conforming to Eqs. (2) and (3), where $N = 1$ and $J_{\max} = 7$.

The subspaces $\mathcal{V}_{j,N}$ are nested:

$$\mathcal{V}_{0,N} \subset \mathcal{V}_{1,N} \subset \cdots \subset \mathcal{V}_{J_{\max},N} \tag{5}$$

with

$$\lim_{J_{\max}\to\infty} \mathcal{V}_{J_{\max},N} = [0,1] \tag{6}$$

The sequence of subspaces $\mathcal{W}_{j,N}$ satisfies the property $\mathcal{W}_{j,N} \cap \mathcal{W}_{l,N} = \varnothing$ for all $j \neq l$. If element subdivision is performed in a nonuniform manner, Eq. (2) can be modified to represent the resulting nonuniform grid:

$$\begin{aligned} \mathcal{G} &= \{\tau_{j_i,k_i} \in [0,1] : 0 \le k_i \le 2^{j_i}\} \\ & J_{\min} \le j_i \le J_{\max} \quad \text{for} \quad i = 1,\ldots,R \\ & \tau_{j_i,k_i} < \tau_{j_{i+1},k_{i+1}} \quad \text{for } i = 1,\ldots,R-1 \end{aligned} \tag{7}$$

where $\mathcal{G} \subset \mathcal{V}_{J_{\max},N}$ and R represents the number of available resolution levels. Of note, working with dyadic grids is essentially equivalent to using interpolating wavelets for the analysis of the underlying function [3] and retains one of the major benefits of wavelet-based analyses: multiresolution functional representations [25,26].

Grid adaptation is accomplished by first assessing the mesh quality. Temporal finite elements exhibit spectral convergence of the boundary values as the number of nodes increases and a slower convergence of the discontinuity at the boundaries [23]. Hence, controlling the discontinuities at element boundaries guarantees control of the global error. By defining the discontinuity at each interface node τ as

$$\boldsymbol{\delta}_\tau = |\boldsymbol{x}^l - \boldsymbol{x}^r| \tag{8}$$

where superscripts l and r denote the values to the left and right of the interface node τ, respectively,[¶] one can thus obtain the relative local error as an indicator of grid quality:

$$\boldsymbol{\epsilon}_\tau = \frac{\boldsymbol{\delta}_\tau}{\max[\boldsymbol{x}_{UB} - \boldsymbol{x}_{LB}, |\boldsymbol{x}_{UB}|, |\boldsymbol{x}_{LB}|]} \tag{9}$$

where $\boldsymbol{x}_{UB}$ and $\boldsymbol{x}_{LB}$ are, respectively, the user-defined upper and lower bounds of the state vector $\boldsymbol{x}$.

Using this approach, one does not require any additional procedure or integration, as sufficient information may be derived from the discontinuities at the boundaries of each finite element. A given grid may therefore be modified by inserting additional interface nodes at the midpoints of all finite elements not satisfying $\epsilon_\tau \le \epsilon_{\text{tol}}$, where ϵ_{tol} is a user-specified tolerance. This then preserves the nonuniform dyadic structure according to Eq. (7).

This refinement procedure requires the associated basis functions spanning each temporal element to be split in the same manner. A useful property of Bernstein polynomials in this regard is that any Bézier curve (sequential collection of Bernstein polynomials) can be exactly represented by two smaller subcurves of the same order, as depicted graphically in Fig. 2. Alternative basis functions, such as Chebyshev polynomials, would have to be refitted after each grid adaptation step. In this study, a vectorized, nondestructive, recursive subdivision scheme [27] is employed that uses linear algebra to define the coordinate points of such subcurves.

As an example, Fig. 3 depicts two extremal solutions to a classic MOCP sourced from the literature [2,3,28], solved using the multiresolution refinement approach of this paper. In each case, it can be seen that grid nodes have been successfully concentrated to an appropriate degree within different areas of the time domain.

C. Multiresolution MACS

For the purposes of this study, the proposed grid refinement approach is used to augment the multi-agent collaborative search (MACS) algorithm, a dominance-based memetic solver for multiobjective optimization (MOO) problems under development at the University of Strathclyde [29].

MACS initializes a population of virtual agents at random locations in the search space, where an "agent" comprises an OC solution vector and corresponding objective function and constraint function values. Every agent in turn explores its local neighborhood through a series of individualistic actions. The population as a whole then performs a set of collaborative, social actions to concurrently advance toward a complete representation of the Pareto front. At every user-specified number of iterations (and again at the final iteration), a gradient-based local optimization utilizing Pascoletti–Serafini scalarization [30] is performed on each agent.

The MACS algorithm has been extensively described in the following works [22,29,31]. However, the current formulation of MACS assumes that all solution vectors conform to a predetermined structure of fixed length, and thus it is incompatible with adaptive problems. In order for the proposed grid refinement procedure to be integrated into MACS, a new social operation, one that allows the handling of variable-length discretization grids and therefore enables open-ended evolution, is required.

The individualistic actions within MACS operate in isolation for each member of the current population and thus need only minimal modification to incorporate continuously adapting solutions. However, considering the collaborative actions (operations that combine information from multiple members of a population, e.g., memetic crossover), accounting for open-ended evolution implies the need to identify common points or sequences between two individuals without assuming one-to-one correspondence between vector components. This type of identification is utilized within variable-length-chromosome genetic algorithms (VLC-GAs) [32,33], a subclass within the wider field of genetic algorithms [34]. Munetomo et al. [35] present a crossover mechanism for VLC-GAs called same-point crossover, involving two operators: one-point (1P) and two-point (2P) crossover. Partial

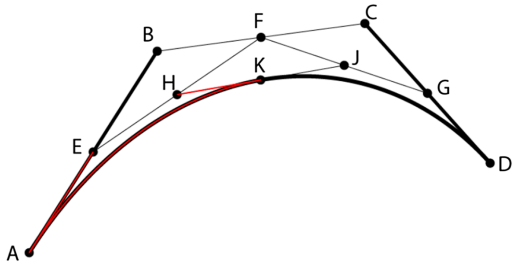

Fig. 2 The Bézier curve defined by nodes ABCD may be perfectly bisected into the curves AEHK and KJGD.

[¶]In other words, $\boldsymbol{x}^l$ represents the final value of the state basis function corresponding to the element immediately preceding τ, and $\boldsymbol{x}^r$ represents the initial value of the state basis function corresponding to the element following τ.

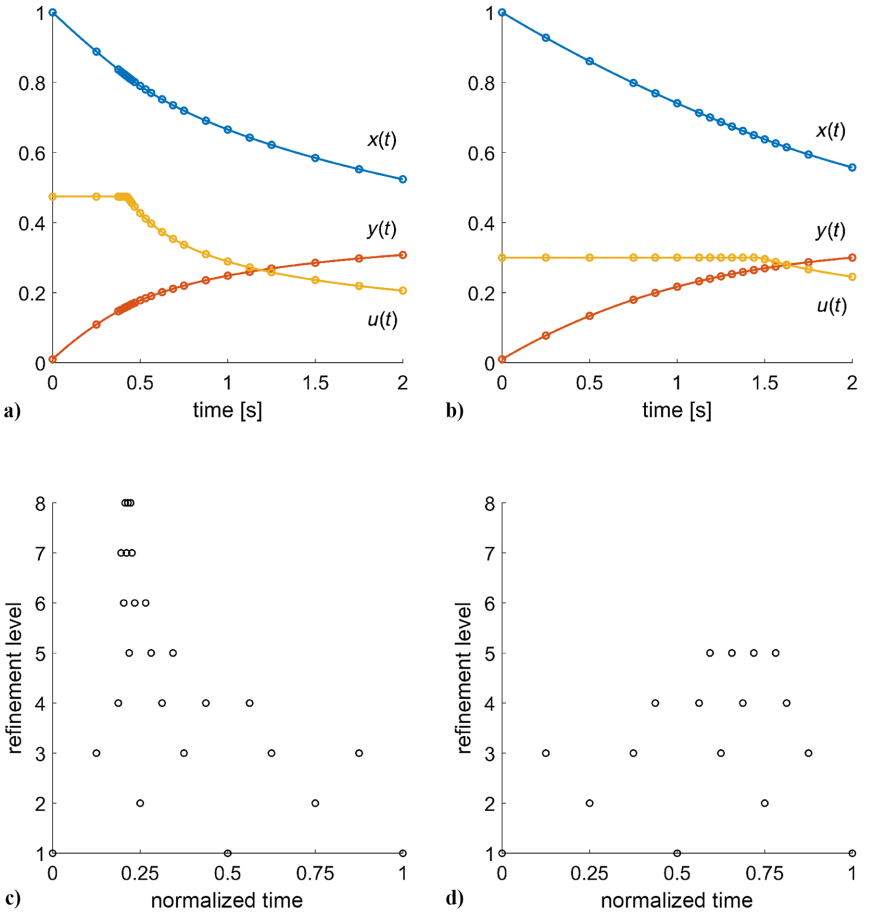

Fig. 3 Extremal multi-objective solutions (including corresponding grid distributions) to the problem presented in [2,3,28].

chromosomes** are thus exchanged at the loci of identical genes between both parent chromosomes. This mechanism has since been inherited by most (if not all) subsequent VLC-GAs. Among other well-known methods for variable-length crossover are Goldberg's messy GA [36], species adaptation genetic algorithms [37], synapsing variable-length crossover [38], the virtual virus method [39], and the more recently proposed same-adjacency crossover [40].

Representing a novel application of VLC methodology, this work proposes to leverage the imposed dyadic grid structures described in Sec. II.B to quickly identify common vector components between multiresolution OC solutions, i.e., using the nodal intersection between two or more unique nonuniform grids to effectively facilitate direct comparison and/or recombination. The precise manner in which this may be performed is now discussed.

1. One-Point Crossover

Consider two individual solution vectors of respective lengths L and K, $\boldsymbol{x} = [x_0, x_1, \ldots, x_i, \ldots, x_L]$ and $\boldsymbol{x}^* = [x_0^*, x_1^*, \ldots, x_j^*, \ldots, x_K^*]$, with corresponding multiresolution grids $\mathcal{G}^{(\boldsymbol{x})}$ and $\mathcal{G}^{(\boldsymbol{x}^*)}$. Given that each agent within the population is initialized upon an identical uniform grid discretization, i.e., $N^{(\boldsymbol{x})} = N^{(\boldsymbol{x}^*)}$, and that individual grid elements can only be perfectly bisected, the intersection between $\boldsymbol{x}$ and $\boldsymbol{x}^*$ will include at least one intermediate node (excluding the trivial case of common initial and final nodes) such that

$$\mathcal{G}^{(\boldsymbol{x})} \cap \mathcal{G}^{(\boldsymbol{x}^*)} \neq \varnothing, \quad 2 \leq R^{(\boldsymbol{x},\boldsymbol{x}^*)} \leq 2^{J_{\max}} \tag{10}$$

The simplest variable-length social action that can thus be considered is that of classical one-point (1P) crossover. In this case, a single intersection node is selected at random, $x_i = x_j^*$, with all components following x_i and x_j^* exchanged between the parent solutions to create two new individuals (see Fig. 4a). This is functionally equivalent to the cut and splice operations introduced in Goldberg's messy GA [36].

2. Two-Point Crossover

One-point crossover has the feature that solution vector components nearer the extremes of each individual are much more likely to be separated than components nearer the middle [37]. Intuitively, then, two-point (2P) crossover may avoid this inherent bias by instead randomly selecting two intermediate nodes from the set of intersections between $\boldsymbol{x}$ and $\boldsymbol{x}^*$, exchanging the enclosed subsequence (see Fig. 4b). In this case, grid point intersections conform to

$$\mathcal{G}^{(\boldsymbol{x})} \cap \mathcal{G}^{(\boldsymbol{x}^*)} \geq 2, \quad 2 \leq R^{(\boldsymbol{x},\boldsymbol{x}^*)} \leq 2^{J_{\max}} \tag{11}$$

3. Dyadic Crossover

A known flaw within both preceding methods is the potential for lack of offspring diversity should the crossover point(s) be chosen too close to the initial and/or final nodes [40]. In such cases, this is equivalent to saying that the length of the exchanged segment is comparable to the lengths of the parent vectors themselves. An interesting consideration within the context of this study is that, as the population is progressively evolved, the number of intersections between similar solutions is likely to increase. This therefore allows a progressively finer level of crossover to occur between similar solutions, thus reducing the overall difference between agents comprising successive generations, which itself has implications for the convergence characteristics of the algorithm as a whole.

However, while the potential for smoother social cooperation is introduced, crossover points are still only selected at random; thus, there is no guarantee that crossover at a higher level of resolution will

**In this context, "chromosome" is taken as equivalent to the problem solution vector.

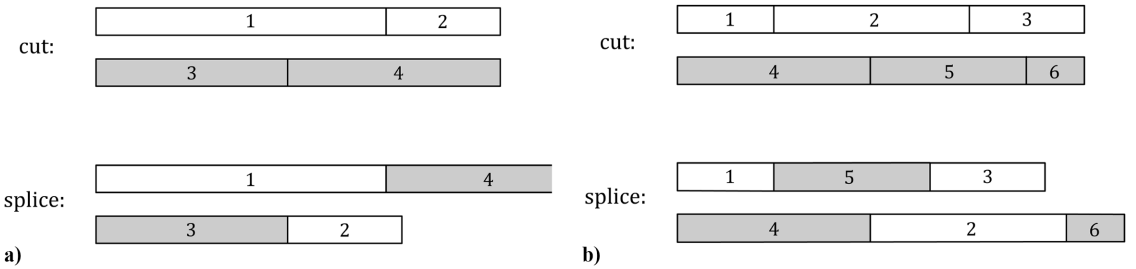

Fig. 4 Crossover performed using a) one and b) two point-pairs between parent vectors.

actually occur. Indeed, Harvey [37] provides a modification to classical two-point crossover in order to choose the second point such that the similarity between the exchanged segments is maximized. "Similarity" in this sense is measured according to the longest common subsequence, defined as the "longest uninterrupted matching substring of gene values (alleles) found between two strings of arbitrary length" [37].

In this study, a bias is introduced weighted toward the exchange of higher-resolution segments. In this manner, as the number of intersection points between solutions increases, information exchange will occur at as high a resolution level as is possible. This aims to minimize the potential disruptive effects caused by blind exchange of larger subsequences (a not uncommon feature of unbiased 1P/2P crossover) and thus promotes a more gradual convergence in objective space. Furthermore, if the selection of crossover points is allowed to include either the initial or final nodes, the biased crossover operation (referred to now as dyadic crossover) can effectively switch freely between 1P and 2P classifications, guided by the resolution of the parent individuals.

4. *VL Differential Evolution*

A final consideration must be given to the use of differential evolution (DE) [41], one of the default social operations within the original MACS algorithm. When a particular agent $\boldsymbol{x}_p$ is selected for DE, a displacement vector $\boldsymbol{dx}_p$ is defined through combination with three randomly selected individuals, $\boldsymbol{x}_{p_1}$, $\boldsymbol{x}_{p_2}$, and $\boldsymbol{x}_{p_3}$, from the population:

$$\boldsymbol{dx}_p = \alpha \boldsymbol{e}((\boldsymbol{x}_p - \boldsymbol{x}_{p_1}) + \boldsymbol{F}(\boldsymbol{x}_{p_2} - \boldsymbol{x}_{p_3})) \tag{12}$$

where α is a random number between 0 and 1, F is a user-specified constant, and $\boldsymbol{e}$ is a mask vector of elements of either 0 or 1 according to

$$e_j = \begin{cases} 1, & \text{if } \alpha_2 < CR \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

where α_2 is another random number between 0 and 1, and the crossover ratio CR is another user-specified constant. The resulting trial position thus reads

$$\boldsymbol{x}_{\text{trial}} = \boldsymbol{x}_p + \boldsymbol{dx}_p \tag{14}$$

To ensure that $\boldsymbol{x}_{\text{trial}}$ lies within the search domain, some components of $\boldsymbol{dx}_p$ are suppressed while α is reduced. If $\boldsymbol{x}_{\text{trial}}$ is determined to represent an improvement with respect to the problem objectives, it is included into the population.

When considering that the agents $\boldsymbol{x}_p$, $\boldsymbol{x}_{p_1}$, $\boldsymbol{x}_{p_2}$, and $\boldsymbol{x}_{p_3}$ may have inconsistent vector lengths, it is clear that Eq. (12) cannot be directly evaluated. Instead, the grid intersection between all individuals is evaluated and used to construct directly comparable segments of constant vector length, to which Eq. (12) may then be successfully applied. In principle then, Eq. (12) is applied only to masked subsequences of the selected individuals representing mutually shared segments, identified via the computation of dyadic grid point intersections in normalized time. Similarly as for the crossover operators, this modification implies that the effects of the new DE operation will naturally strengthen proportionally to the level of nodal intersection observed between solutions.

III. Algorithm

The components introduced in this work, namely, the dyadic grid refinement and the variable-length crossover operations, are applied as an extension to the existing MACS algorithm. The methods themselves are relatively general, and thus MACS is here used primarily as a convenient test bed with which to demonstrate their usability.

Figure 5 illustrates the complete process flow, with the proposed modifications to the existing MACS algorithm structure indicated with red-dotted boxes. The complete algorithm operates as follows: A population of agents is initialized, where each represents a solution to an MOCP discretized upon a uniform initial grid using DFET [23,24]. Each agent is first modified in isolation using the individualistic actions of the original algorithm. Next, each agent is modified further using the proposed variable-length crossover action, using grid-point intersections as a basis for selecting suitable crossover points. The bilevel optimization structure [31] of the original algorithm is retained, where solutions produced by both the individualistic and social operations (i.e., the outer level) are initially considered only candidate solutions and passed to an inner-level local optimization step to enforce feasibility. If the resulting solution then represents an improvement, it is included in the current population, otherwise discarded. At every user-specified number of iterations, the gradient-based step of the original algorithm is performed and immediately supplemented by the proposed dyadic grid refinement step. By doing so, local optimality may be guaranteed before the mesh refinement step. Given the useful properties of Bernstein basis functions described in Sec. II.B, no further treatment of the population is necessary following grid refinement. This whole process

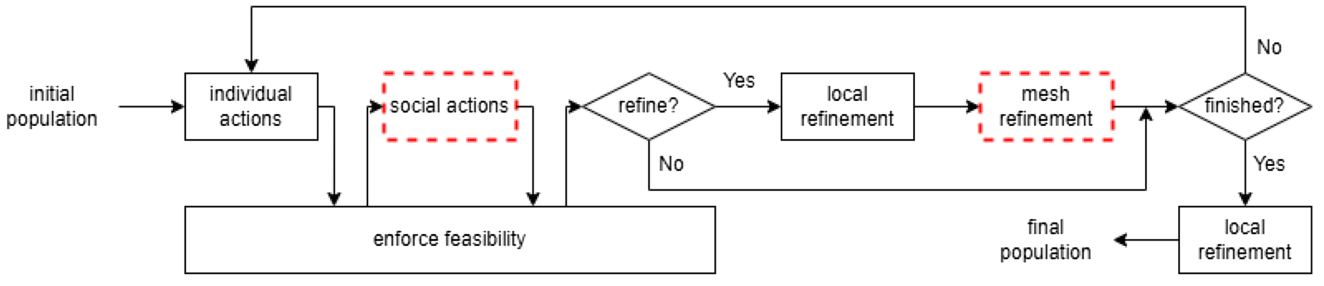

Fig. 5 Simplified process flowchart of the modified multiresolution MACS algorithm.

repeats until either a maximum budget of function evaluations or total iterations is met.

By integrating these new steps into the original algorithm, the population is allowed to evolve into a set of multiresolution solutions defined upon independent nonuniform discretization grids. This then allows complex, nonlinear/discontinuous behavior otherwise unique to any particular solution to be efficiently identified and captured as part of an evolutionary process.

IV. Test Cases

This section presents two test cases intended to demonstrate the usability of the proposed multiresolution refinement procedure for MOCPs. For each examined case, the proposed approach is compared with a traditional uniform grid approach at successive levels of resolution. Algorithm performance is measured by tracking the mean and standard deviation of the maximum relative local error, overall execution time, number of function evaluations, and minimum achieved objective function values at each iteration. Additionally, the quality and spread of the final Pareto-front approximation in each case is assessed against a reference Pareto-optimal set of solutions via the inverted generational distance (IGD) and averaged Hausdorff distance (AHD) metrics. As remarked by Schütze et al. [42], the IGD metric is sensitive to the number of elements in the reference Pareto front and in the computed one, hence the inclusion of the AHD in the presented comparisons.

All computations were performed using the ARCHIE-WeSt High Performance Computer. Individual runs utilized a single Intel Xeon Gold 6138 20-core 2.0 GHz CPU, where 40 nodes comprise a standard compute node (on Lenovo SD30 servers) of 192 GB RAM per node. All results for each case are averaged over 10 independent analyses.

A. Moon Lander

The following MOCP problem represents the controlled descent of a lunar landing vehicle [43], based on the formulation presented by Darby et al. [7] and Feng et al. [44]. The problem is to determine the control law $\boldsymbol{u}$ that minimizes the total impulse acting on the vehicle while also minimizing the maximum allowable control force:

$$\min_{t_f,\boldsymbol{u}}[J_1, J_2]^T = \left[\int_{t_0}^{t_f} u \, \mathrm{d}t, u_{\max}\right]^T \tag{15}$$

The system state dynamics are given by

$$\dot{h} = v \tag{16}$$

$$\dot{v} = -g + u \tag{17}$$

where h is altitude, v is velocity, g is the gravity force, and u is the upward control force. Boundary conditions are taken from [7,44] as $h(0) = h_0 = 100$ ft, $v(0) = v_0 = -10$ ft/s, $h(t_f) = h_f = 0$ ft, and $v(t_f) = v_f = 0$ ft/s. A path constraint on the control force is similarly defined as

$$0 \le u \le u_{\max}$$

The gravity of the moon is taken as $g = 5.315$ ft/s^2, the final time t_f is free, and the maximum control force is an optimizable static parameter $u_{\max} \in [0, 12]$. The values of all boundary conditions, gravity force, and maximum control force were chosen to coincide with the works [7,44] such that solutions could be directly validated and compared.

The optimal solution(s) may be represented analytically using

$$h^*(t) = \begin{cases} -\dfrac{3}{4}t^2 + v_0 t + h_0, & t < s \\ \dfrac{3}{4}t^2 + (-3\ s + v_0)t + \dfrac{3}{2}s^2 + h_0, & t \ge s \end{cases} \tag{18}$$

$$v^*(t) = \begin{cases} -\dfrac{3}{2}t + v_0, & t < s \\ \dfrac{3}{2}t + (-3\ s + v_0), & t \ge s \end{cases} \tag{19}$$

$$u^*(t) = \begin{cases} 0, & t < s \\ u_{\max}, & t \ge s \end{cases} \tag{20}$$

where the temporal location of the control discontinuity, s, is given by

$$s = \frac{t_f}{2} + \frac{v_0}{3} \tag{21}$$

with

$$t_f = \frac{2}{3}v_0 + \frac{4}{3}\sqrt{\frac{1}{2}v_0^2 + \frac{3}{2}h_0} \tag{22}$$

System state and control profiles are represented by element-specific second-order Bernstein polynomials. A population of 10 individual agents was initially discretized upon uniform grids of resolution level $J_{\min} = 0$. The population was evolved according to the proposed multiresolution modification to the MACS algorithm (Sec. III). A relative local error tolerance of $\epsilon_{\text{tol}} \le 10^{-6}$ was imposed at element boundaries, with a mesh refinement frequency of 10 iterations. Inner-level optimization was completed using the MATLAB fmincon solver with an Sequential Quadratic Programming (SQP) algorithm where function, constraint violation, and step tolerances are set to $1\text{e}-6$, $1\text{e}-12$, and $1\text{e}-9$, respectively. The complete process was terminated upon reaching a maximum number of iterations, $n_{\text{iter,max}} = 200$.

Figure 6 shows the time histories of the state and control profiles from an example set of solutions, including the locations of all corresponding grid points (represented by circle markers) for each. The color of each individual solution in Fig. 6a corresponds to all profiles of the same color across Figs. 6b–6d. Additionally, Fig. 7 shows the dyadic distribution of grid points for both orthogonal objective solutions with respect to resolution level J across the normalized time domain. The proposed approach is seen to concentrate additional collocation points around the single control discontinuity within each solution. The characteristic bang-bang behavior of the control profile and the resultant effects on the system state are thus more accurately captured. Extremal objective function values correspond to those reported in [7,44], as well as the analytical solutions obtained via Eqs. (18–20). Of note is the lack of constraint on the final time t_f. This means that the dyadic crossover procedure, operating fundamentally using nodal intersections within a normalized time domain, is primarily exchanging the form of the solution rather than specific numeric values.

Table 1 presents the performance comparisons between the proposed multiresolution approach and a globally dense uniform grid approach, where the latter is separately repeated at increasing levels of resolution. Reported are the 10-run average values, with corresponding standard deviations included in parentheses. Intuitively, an inverse correlation between mesh resolution and relative local error is observed across the uniform grid cases. Furthermore, higher mesh densities incur an exponential increase in computational costs given the associated evolution of the NLP problem size. The multiresolution approach, on the other hand, is seen to temper this inherent tradeoff, achieving competitive mesh error values in significantly reduced execution times. As an example, to reach a maximum relative local error lower than that achieved via the multiresolution method, a dense grid of resolution level $J = 7$ is required as a minimum. By comparison, the required CPU time of the multiresolution approach is only 4.14% that of the $J = 7$ grid, respectively. Furthermore, there seems to be no benefit to the dense grid approach even when considering the orthogonal objective function solutions, with in fact a 2.67 and 0.19% deterioration in J_1 and J_2, respectively.

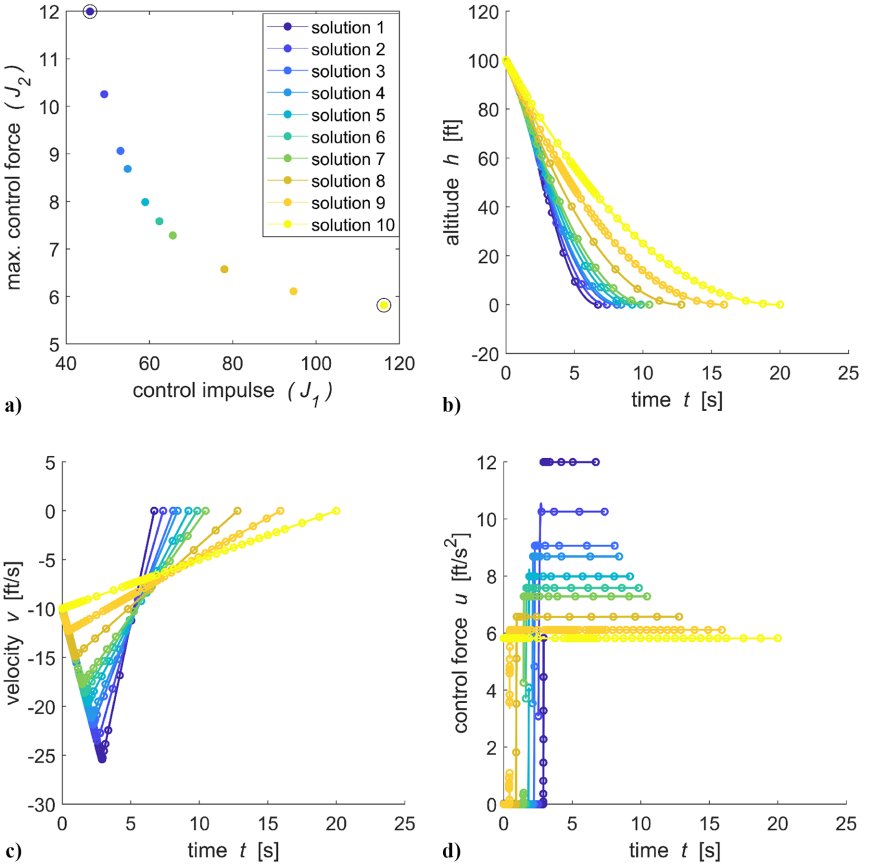

Fig. 6 Multi-objective solutions to Problem IV.A (Moon Lander), including a) Pareto front and corresponding b, c) state and d) control profiles.

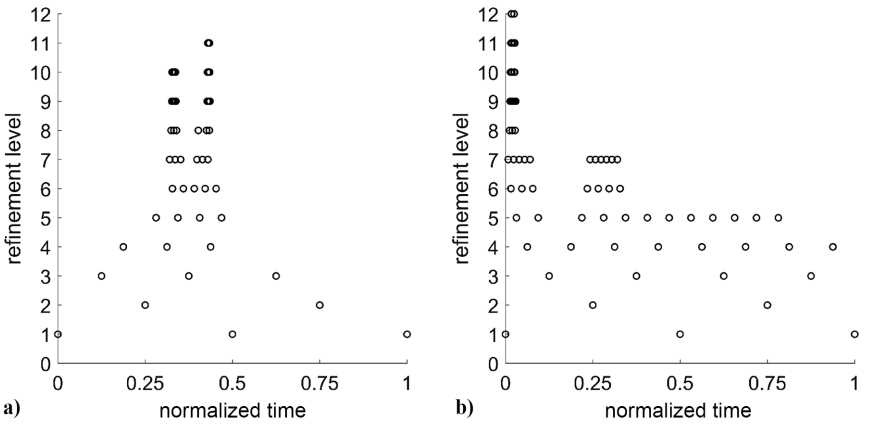

Fig. 7 Dyadic distribution of collocation points for the orthogonal solutions to a) objective 1 and b) objective 2 of Problem IV.A (Moon Lander).

Table 1 Mean statistics for Problem IV.A (Moon Lander), recorded at $n_{\text{iter}} = 200$ (standard deviations included in parentheses). Bold text indicates the best achieved value in each field

Resolution level $J_{\min}$	$J_{\max}$	Mesh error, $\epsilon_{\max}$	No. function evaluations, n_{feval}	Run time, s	Objective functions, $\boldsymbol{J}_{\text{best}}$
3	3	9.27e−3 (2.35e−3)	**9,845 (3.59e + 2)**	**689** (8.72e + 1)	[46.433 (1.22e + 0), 5.845 (7.30e−2)]
4	4	4.66e−3 (1.77e−3)	11,663 (5.74e + 2)	1,446 **(3.68e + 1)**	[46.814 (2.99e + 0), 5.858 (1.08e−1)]
5	5	2.48e−3 (1.13e−3)	13,874 (1.36e + 3)	3,298 (7.21e + 1)	[45.980 (6.68e−1), 5.860 (1.35e−1)]
6	6	1.06e−3 (2.67e−4)	18,909 (2.39e + 3)	13,922 (7.00e + 2)	[46.907 (2.72e + 0), **5.818 (1.20e−2)**]
7	7	**5.06e−4 (1.95e−4)**	29,892 (5.38e + 3)	94,988 (5.40e + 3)	[47.200 (2.03e + 0), 5.846 (7.32e−2)]
0	15	6.29e−4 (6.15e−4)	42,873 (4.90e + 3)	3,932 (8.61e + 2)	[**45.973 (3.12e−1)**, 5.835 (3.50e−2)]

Considering the final approximation of the Pareto front in each case, the mean and corresponding standard deviations of the IGD and AHD metrics for each method are reported in Table 2. For this case, the reference 10-point Pareto front is generated by solving Eqs. (18–20) with $u_{\max}$ increased in regular intervals between the minimum analytical value (where $s = 0$) and the prescribed upper bound.

Table 2 Mean values (standard deviations included in parentheses) of IGD and AHD for all uniform and multiresolution solutions to Problem IV.A (Moon Lander). Bold text indicates the best achieved value in each field

Resolution level		Inverted generational	Averaged Hausdorff
J_{min}	J_{max}	distance (IGD)	distance (AHD)
3	3	1.3086 (0.6455)	5.7792 (**1.4925**)
4	4	1.3854 (0.7940)	5.6274 (2.0465)
5	5	1.7463 (0.9165)	6.3041 (2.1602)
6	6	1.3214 (0.6192)	**5.0238** (2.0216)
7	7	1.5167 (0.4960)	5.4606 (1.2809)
0	15	**0.8557 (0.3078)**	5.0446 (2.2234)

Reasonable agreement in IGD across each method is observed. This may be due in part to the simplicity of the parent problem, meaning that the final objective values are less dependent on the accuracy with which the optimal state and control profiles can be captured. Regardless, the IGD metric is seen to roughly increase with mesh resolution, with the multiresolution approach achieving the lowest value and associated variance. The AHD metric shows a somewhat similar relationship, though still retaining reasonable parity across all resolutions. As stated by Schütze et al. [42], AHD penalizes outliers to a greater degree than IGD, thus acting more as a worst-case measure. Given the parity in number of agents included in each method and the reference Pareto front (thus avoiding any inherent IGD bias), it can be suggested that the multiresolution approach suffers more from outliers despite slight improvement in the overall approximation of the Pareto front.

The usability of the proposed approach is fundamentally defined by the tradeoff between achievable solution accuracy and computational expense. Figure 8 shows the values of maximum relative local error and AHD recorded at each iteration for each multi-/uniform-resolution process. Firstly, a clear hierarchy can be seen in the error values achievable via each level of mesh density. Corresponding to the progressive refinement process, the multiresolution method is seen to move down through each level rather than stagnate at any particular limit. Secondly, the value of AHD associated with the multiresolution approach is seen to reach relatively low values markedly faster than the higher resolution uniform approaches, suggesting that the algorithm is successfully achieving and maintaining a well-spread population from earlier stages in the optimization process. This complimentary behavior, in accordance with the results presented in Tables 1 and 2, can be more explicitly depicted via the combined metric shown in Fig. 8c.

Figure 9 shows the accumulation of total CPU time and function evaluations for each method relative to the number of completed iterations. While the multiresolution approach demonstrates efficiency savings in CPU time relative to the hierarchy of uniform-grid examples, a significant accumulation of objective function evaluations is observed during later iterations. This rise can be seen to occur primarily during the local search step, suggesting an increasing difficulty for the gradient-based optimizer to converge on a locally optimal solution following successive grid refinement. This may be attributed to excessive growth of the associated NLP problem (an acknowledged risk with variable-length representations in evolutionary algorithms [45]). In this work, NLP bloat is treated only indirectly through the user specification of mesh error tolerance. That is, an appropriate tolerance value prevents the algorithm from actively adding excessive new nodes. However, it is acknowledged that this does not prevent heavily refined segments from being relocated by the crossover mechanism to areas within the solution vector that do not require such levels of detail. The explicit identification and removal of unnecessary nodes is a complex problem and thus originally considered with the scope of this study, though it remains an important subject of future work.

The open-ended evolution of the NLP problem size is shown in Fig. 10. Of interest here is that even though the multiresolution population contains elements at double the resolution level of the

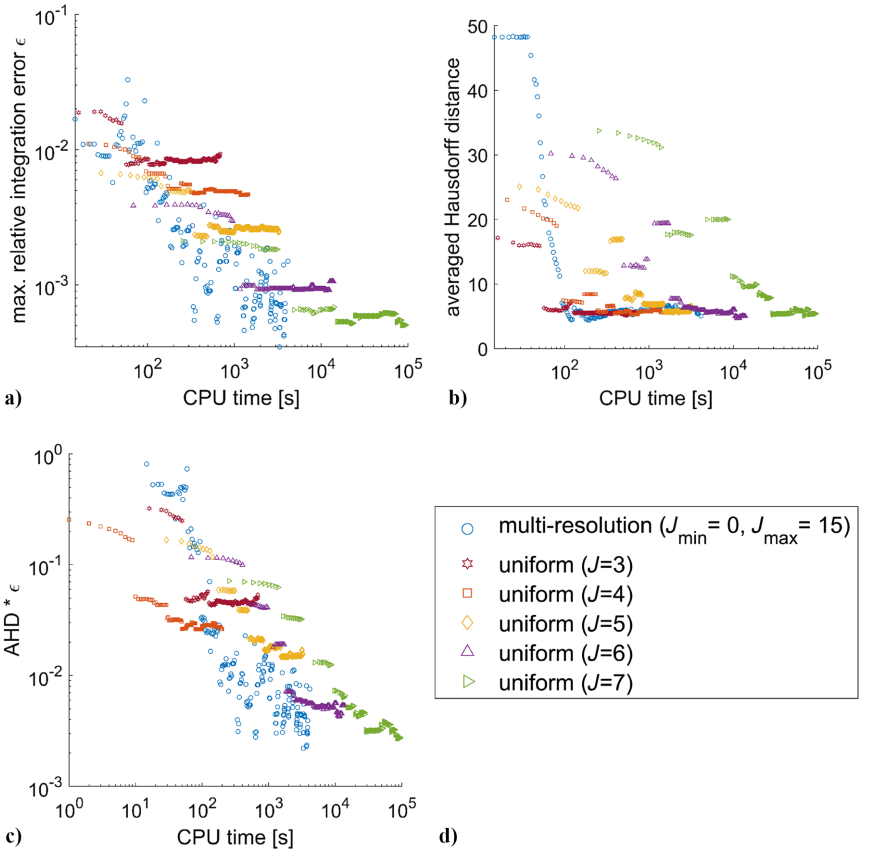

Fig. 8 Relative mesh error and AHD metrics relative to the total execution time for Problem IV.A (Moon Lander).

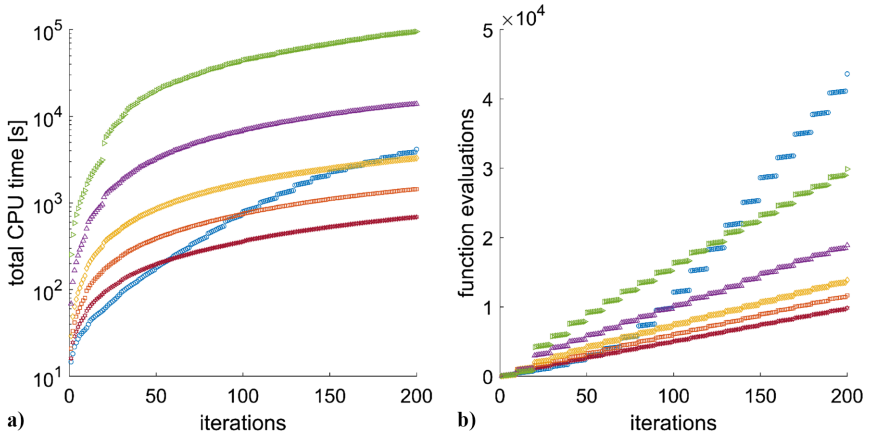

Fig. 9 a) Total execution time and b) number of function evaluations of Problem IV.A (Moon Lander) relative to the number of completed iterations.

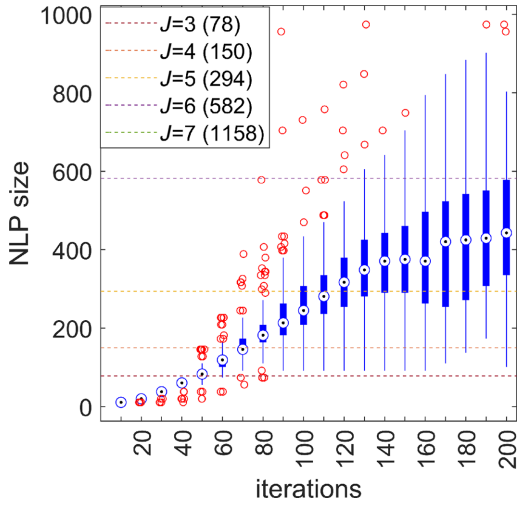

Fig. 10 NLP problem size across the included population relative to the number of completed iterations of Problem IV.A (Moon Lander).

densest uniform grid considered, the average solution vector size at the final iteration is still relatively small in comparison. While most likely due to the functionally simple nature of the state/control profiles surrounding the discontinuity, this nevertheless demonstrates the efficiency savings offered by the proposed approach.

B. Shuttle Reentry

This second problem, adapted from the classic test case by Betts [21], considers maximizing the cross-range distance $\phi(t_f)$ of a space-shuttle-like vehicle during atmospheric reentry while minimizing the peak convective heat flux q_U experienced during flight. The MOCP may be expressed as

$$\min_{t_f,\boldsymbol{u}}[J_1, J_2]^T = [-\phi(t_f), q_U]^T \tag{23}$$

subject to the following dynamics on the position vector $\boldsymbol{r} = [r, \theta, \phi]$ and the velocity vector $\boldsymbol{v} = [v, \gamma, \psi]$:

$$\dot{r} = \dot{h} = v\sin\gamma \tag{24}$$

$$\dot{\theta} = \frac{v\cos\gamma\sin\psi}{r\cos\phi} \tag{25}$$

$$\dot{\phi} = \frac{v\cos\gamma\cos\psi}{r} \tag{26}$$

$$\dot{v} = -\frac{D}{m} - g\sin\gamma \tag{27}$$

$$\dot{\gamma} = \frac{L\cos\beta}{mv} - \left(\frac{g}{v} - \frac{v}{r}\right)\cos\gamma \tag{28}$$

$$\dot{\psi} = \frac{L\sin\beta}{mv\cos\gamma} + \frac{v\cos\gamma\sin\psi\tan\phi}{r} \tag{29}$$

with altitude h, longitude θ, latitude ϕ, velocity v, flight path angle γ, and heading angle ψ. Consistent with the problem definition in [2], the boundary conditions at initial t_0 and final t_f times are specified in Table 3.

Path constraints are given as

$$\begin{aligned} -90\ \text{deg} \le \alpha \le 90\ \text{deg} \\ -89\ \text{deg} \le \beta \le 1\ \text{deg} \\ |\,\dot{\gamma}(t)\,| \le 2\ \text{deg/s} \\ |\,\dot{\psi}(t)\,| \le 2\ \text{deg/s} \\ q \le q_U \end{aligned} \tag{30}$$

where the control variables α and β are the angle of attack and bank angle, respectively. The force of gravity is taken as

$$g = \frac{\mu}{r^2} \tag{31}$$

where $r = R_e + h$ and the Earth radius $R_e = 20{,}902{,}900$ ft. The aerodynamic forces acting on the vehicle are given by

$$L = \frac{1}{2}C_L S\rho v^2 \tag{32}$$

$$D = \frac{1}{2}C_D S\rho v^2 \tag{33}$$

where the coefficients of lift and drag (C_L and C_D, respectively) are defined by the following fitted relationships:

Table 3 Initial and final conditions for Problem IV.B (Shuttle Reentry)

	Initial ($t = 0$)		Final ($t = t_f$)	
Variable	Symbol	Value	Symbol	Value
Altitude (h), ft	$h(0) = h_0$	260,000	$h(t_f) = h_f$	80,000
Longitude (θ), deg	$\theta(0) = \theta_0$	0	$\theta(t_f) = \theta_f$	Free
Latitude (ϕ), deg	$\phi(0) = \phi_0$	0	$\phi(t_f) = \phi_f$	Free
Velocity (v), ft/s	$v(0) = v_0$	25,600	$v(t_f) = v_f$	2500
Flight path angle (γ), deg	$\gamma(0) = \gamma_0$	−1	$\gamma(t_f) = \gamma_f$	−5
Heading angle ψ, deg	$\psi(0) = \psi_0$	90	$\psi(t_f) = \psi_f$	Free

$$C_L = a_0 + a_1\hat{\alpha} \tag{34}$$

$$C_D = b_0 + b_1\hat{\alpha} + b_1\hat{\alpha}^2 \tag{35}$$

where $\hat{\alpha}$ is the angle of attack in radians and the constant values $a_0 = -0.20704$, $a_1 = 0.029244$, $b_0 = 0.07854$, $b_1 = -0.61592 \times 10^{-2}$, and $b_2 = 0.621408 \times 10^{-3}$. Atmospheric density ρ is defined by

$$\rho = \rho_0 \exp[-h/h_r] \tag{36}$$

where the sea-level density is $\rho_0 = 0.002378$ lb/ft^2 and reference altitude $h_r = 23{,}800$ ft. The convective heat flux at the vehicle nose, q, is calculated using

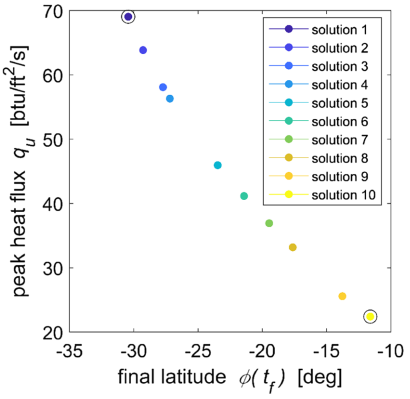

Fig. 11 Multi-objective Pareto front for Problem IV.B (Shuttle Reentry).

$$q = q_a q_r \tag{37}$$

where

$$q_a = c_0 + c_1\hat{\alpha} + c_2\hat{\alpha}^2 + c_3\hat{\alpha}^3 \tag{38}$$

$$q_r = 17700\sqrt{\rho}(0.0001v)^{3.07} \tag{39}$$

and is constrained by a maximum peak value $q_U \in [0, 70]$, itself a static parameter. The constant values are $c_0 = 0.0672181$, $c_1 = -0.19213774 \times 10^{-1}$, $c_2 = 0.21286289 \times 10^{-3}$, and $c_3 = -0.10117249 \times 10^{-5}$. The reentry trajectory begins at an altitude where the aerodynamic forces are small, with a vehicle weight of $m = 203{,}000$ lb.

This problem is solved in the same manner as for the previous problem, with largely the same numerical settings. In this case, however, seventh-order Bernstein polynomials are used to represent the state and control profiles, and the function, constraint violation, and step tolerances of the inner-level optimization step were set to $1e-6$, $1e-12$, and $1e-9$, respectively. Lastly, the maximum number of allowable iterations for this case was set to $n_{\text{iter,max}} = 120$.

Figure 11 shows the spacing of an example set of solutions in objective space, with Figs. 12 and 13 displaying the corresponding color-coded time histories of selected states and controls. Additionally, Fig. 14 shows the dyadic distribution of grid points for both orthogonal objective solutions. This example features complex state and control profiles with no apparent discontinuity (though including distinct areas of rapid change). However, the similarity in form between solutions across the population can be clearly seen. Importantly, this demonstrates that important areas are being discovered, preserved, and propagated throughout the optimization process. Objective function values correspond to those reported in [2,46].

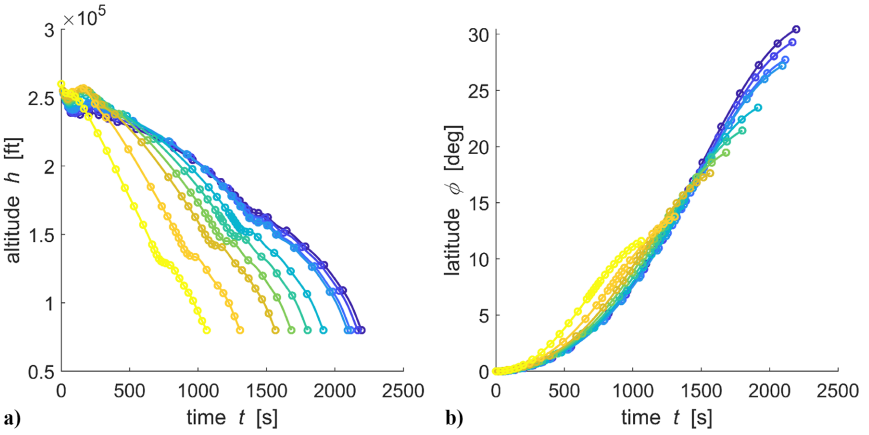

Fig. 12 Multi-objective state profiles for Problem IV.B (Shuttle Reentry).

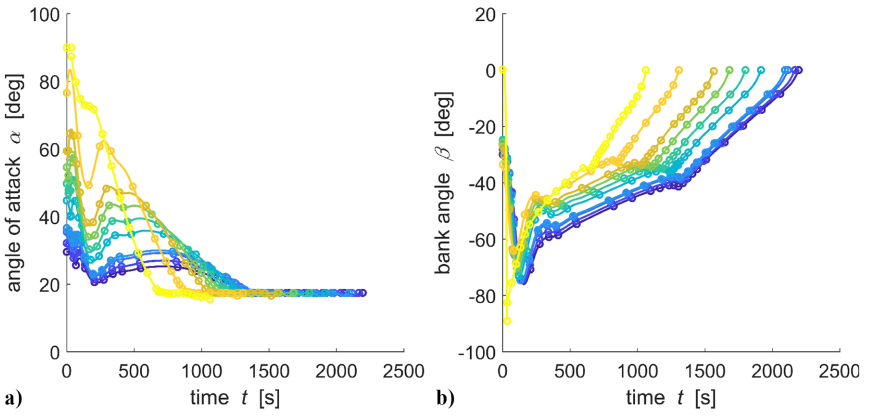

Fig. 13 Multi-objective control profiles for Problem IV.B (Shuttle Reentry).

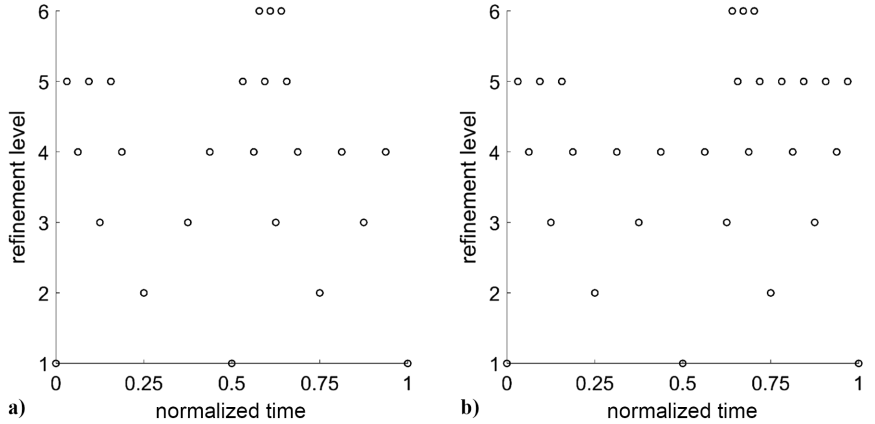

Fig. 14 Dyadic distribution of collocation points for the orthogonal solutions to a) objective 1 and b) objective 2 for Problem IV.B (Shuttle Reentry).

Table 4 presents the performance comparisons between the multiresolution approach and the series of globally dense uniform grids. In a similar manner to the previous cases, the multiresolution approach achieves a favorable balance between solution accuracy and computational cost relative to the uniform grid examples. For example, the highest level of dense grid considered here, $J = 5$, was unable to reach the same level of maximum relative mesh integration error as the multiresolution approach within the set number of iterations, despite the latter terminating with 12.7% of the computational cost. Furthermore, as in the previous example, there seems to be no significant benefit to the dense grid approach when considering the orthogonal objective function solutions, with a 1.02% improvement and a 17.7% deterioration in J_1 and J_2, respectively, relative to the multiresolution approach.

The mean values and corresponding standard deviations of IGD and AHD for each method are reported in Table 5, where the reference Pareto front is taken directly from [2,46]. In this case, the multiresolution approach achieves superior values for mean and variance according to both metrics. This may be due to a number of reasons. At lower mesh resolutions, the optimizer may not be able to capture the finer features of the state/control profiles, and thus, while easily achieving a well-spread population, the approximation accuracy of the Pareto front and mesh errors remains relatively poor. On the other hand, higher mesh densities may more capably capture detailed behavior, but the resulting size of the NLP problem (e.g., 2057 total variables for the $J = 5$ grid) and the inherent existence of overrefined areas hinder the spreading of the population in objective space, as indicated by the lower IGD values relative to AHD. The ability of the multiresolution approach to efficiently manage the gradual refinement of the population so as to maintain global exploratory capability can also be seen in Fig. 15.

Figure 16 shows the accumulation of total CPU time and objective function evaluations, respectively, relative to the number of completed iterations. Of interest in this case is the more rapid accumulation of CPU time at earlier stages of the multiresolution process. This suggests a difficulty for the optimizer to produce feasible solutions at lower levels of refinement. This is further supported by Fig. 17, showing that the evolution of NLP problem size progresses in almost a uniform manner until around 50 iterations (or above resolution level $J = 3$). As in the previous example, excessive numbers of objective function evaluations are prevalent during later (>70) iterations, though Fig. 17 at least indicates that true NLP bloat is being tempered given the apparent convergence of average NLP size. This could possibly be attributed to the indirect nature of solution growth mentioned earlier, in that refinement will not occur indefinitely without the mesh error continuously failing to meet the required tolerance.

Table 5 Mean values (standard deviations included in parentheses) for IGD and AHD for all uniform and multiresolution solutions to Problem IV.B (Shuttle Reentry). Bold text indicates the best achieved value in each field

Resolution level $J_{\min}$	$J_{\max}$	Inverted generational distance (IGD)	Averaged Hausdorff distance (AHD)
2	2	1.44 (1.27)	4.76 (3.87)
3	3	0.93 (0.29)	3.27 (0.81)
4	4	1.66 (1.00)	5.26 (3.17)
5	5	1.02 (0.54)	3.42 (1.57)
0	8	**0.59** (**0.05**)	**2.51** (**0.27**)

V. Discussion

The examples presented in Sec. IV illustrate the efficiency of the proposed multiresolution approach compared to a more traditional uniform-grid approach for MOCPs. Test case IV.A (Moon Lander) incorporates simple state dynamics with a single control discontinuity. The solutions display a sharp, bang-bang-type control profile that is effectively captured by the proposed approach throughout the population of MO solutions. Of interest is the demonstration that, given a free final time, refined areas may be propagated across the population via the exchange of normalized time segments. Test case IV.B (Shuttle Reentry) includes multiple control variables and complex state/control profiles with areas of high nonlinearity, though with no clear discontinuities. In this case, the multiresolution approach is seen to preserve the form of the optimal trajectory throughout the population, with individualistic refinement applied where appropriate. This case again

Table 4 Mean statistics for Problem IV.B (Shuttle Reentry), recorded at $n_{\text{iter}} = 120$ (standard deviations included in parentheses). Bold text indicates the best achieved value in each field

Resolution level $J_{\min}$	$J_{\max}$	Mesh error, $\epsilon_{\max}$	No. function evaluations, n_{feval}	Run time, s	Objective functions, $\boldsymbol{J}_{\text{best}}$
2	2	5.15e−3 (1.71e−3)	**27,578 (3.75e + 3)**	**10,330 (4.28e + 3)**	[−0.506 (5.65e−2), 22.16 (1.32e + 0)]
3	3	3.66e−4 (1.78e−5)	39,967 (6.70e + 3)	43,108 (1.98e + 4)	[−0.530 (9.12e−3), **21.95** (4.16e−2)]
4	4	1.43e−4 (9.85e−6)	50,818 (1.14e + 4)	116,624 (1.91e + 4)	[−0.529 (8.00e−3), 27.95 (1.09e + 1)]
5	5	5.26e−5 (**5.33e−6**)	63,374 (1.02e + 4)	640,130 (8.63e + 4)	[**−0.534 (5.10e−4)**, 26.39 (7.50e + 0)]
0	8	**4.11e−5** (4.50e−5)	84,834 (3.05e + 4)	81,155 (1.19e + 4)	[−0.529 (8.56e−3), 22.42 (**2.36e−2**)]

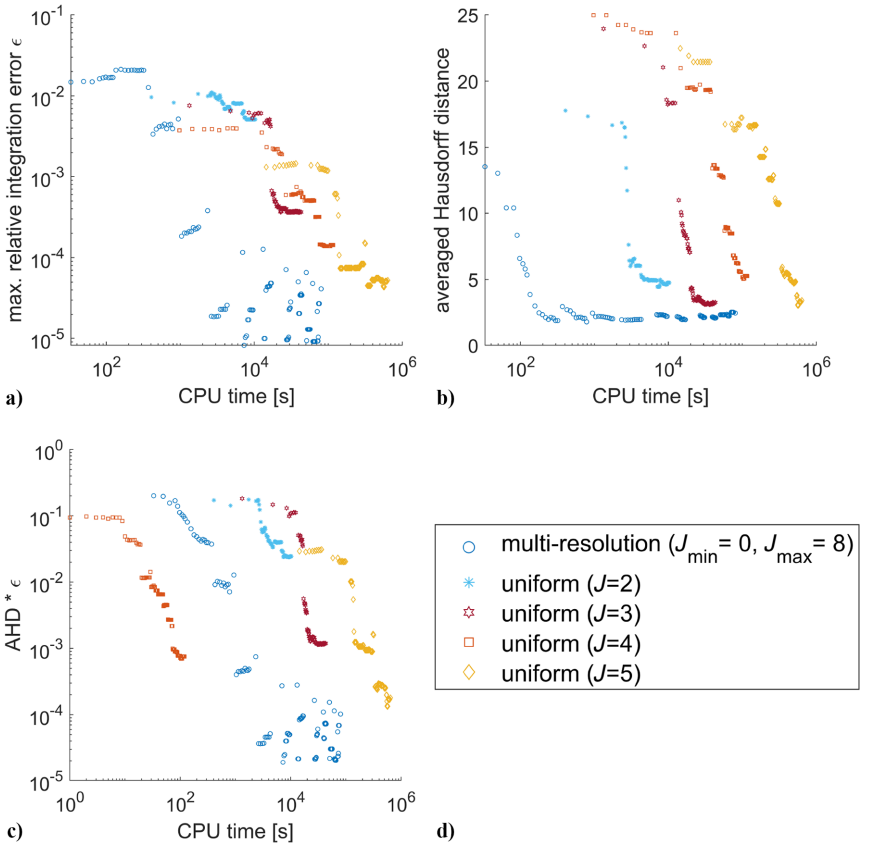

Fig. 15 Problem IV.B (Shuttle Reentry) mesh error and AHD relative to the total CPU time.

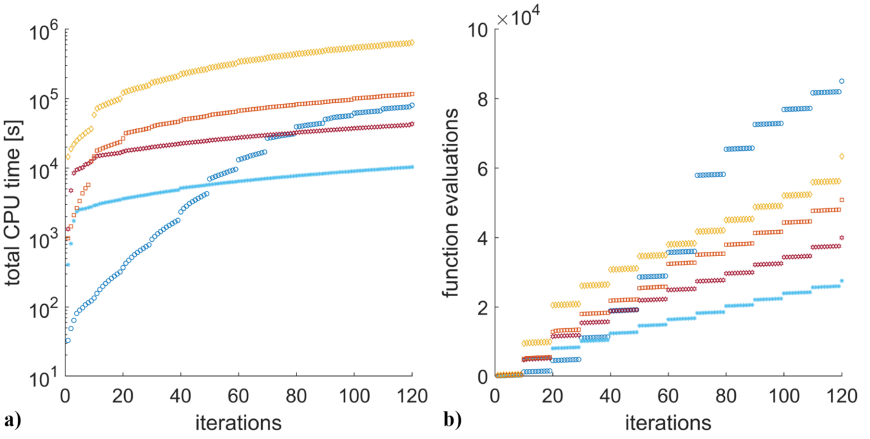

Fig. 16 a) Total execution time and b) number of function evaluations of Problem IV.B (Shuttle Reentry) relative to the number of completed iterations.

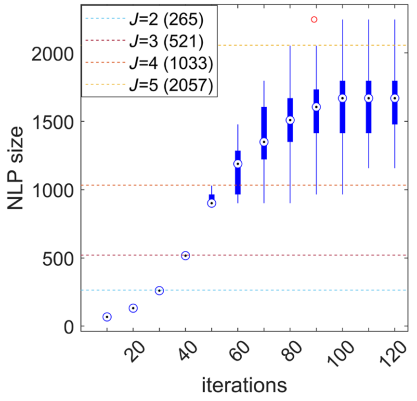

Fig. 17 NLP problem size across the included population relative to the number of completed iterations of Problem IV.B (Shuttle Reentry).

demonstrates cost savings over a uniform-grid approach for the same level of accuracy.

Common to both cases then is the reduction in computational cost required to reach specified levels of relative local error compared to selected uniform grid resolutions (themselves chosen in order to remain within practical computational bounds). However, a greater reliance on objective function evaluations is observed, particularly in later iterations when the multiresolution populations begin to converge. While potentially merely a symptom of the periodic local refinement procedure requiring more intensive computation given the increased NLP problem size, this may also correspond to the inherent probability of retaining/propagating overrefined segments, thus introducing unnecessary computational burden through excessive NLP bloat. Such behavior may form as a result of particular areas of a solution undergoing refinement during earlier stages of the process, later becoming unnecessary as the population develops. An additional factor could be the propagation of overrefined segments purely by

association with a favorable subsequence (i.e., one that improves the objective function values of a solution when inserted via crossover, despite containing segments of unnecessarily high resolution). This issue is in large introduced by considering that the removal of overrefined segments is only at this stage performed in an indirect manner. More specifically, an overrefined segment can only be removed if it fails to propagate across successive generations, either via substitution during social action or by the parent(s) being removed from the current archive. An issue here is that such occurrences are only possible if there exists a solution with superior fitness despite the absence of that particular segment. If an overrefined solution exists but is sufficiently dominant within the population, it shall remain undisturbed. Clearly then an additional step introduced to validate each individual, identifying and removing overrefined segments, would be a valid subject of future work. Of note is that this behavior seems less pronounced for the Shuttle Reentry test case, possibly suggesting an overly strict value of relative local error tolerance ϵ_{tol} for the Moon Lander example.

Adapting the degree of the polynomial basis functions, thus enabling full *hp*-adaptivity, was initially determined with the scope of this work, though it remains an interesting subject for future study. For example, the work [47], in reference to the problem reproduced in Sec. IV.B, suggests a positive effect in solution accuracy and cost by increasing the polynomial degree (within user-defined applicability bounds) as opposed to the number of mesh elements.

VI. Conclusions

This paper presents an adaptive mesh refinement approach for solving direct MOCPs within an evolutionary architecture. Each individual solution is discretized upon an independent nonuniform dyadic grid using DFET. New grid points may be added at each iteration according to a criteria based on simple local errors evaluated at element boundaries. The transcribed problem is solved using a new version of the MACS algorithm, where the original formulation has been modified to incorporate adaptive, multiresolution solution vectors. This includes both the proposed dyadic mesh refinement procedure and new social operators to enable collaborative action between solutions of inconsistent grid resolution. Concerning the latter, common segments between unique individuals are identified using the nodal intersections of their associated discretization grids. Next, a biased variable-length crossover technique that prioritizes the exchange of higher resolution segments is applied to create entirely new candidate solutions.

The proposed approaches are applied to two demonstrative test cases, where in each the associated MOCP is treated as an open-ended evolutionary design problem. As the primary motivation of this work was a desire to increase the achievable fidelity of MOCP solutions as well as the numerical efficiency, the computational cost, achievable solution accuracy, and population spread/diversity were measured for each case and compared against a traditional uniform-grid approach. An order of magnitude reduction in computational cost required to reach a desired level of mesh integration error was observed for both cases. Furthermore, the progressive nature of population refinement was shown to facilitate a more complete and diverse representation of the Pareto front.

Acknowledgments

This work was partially funded by an UKRI Engineering and Physical Sciences Research Council (EPSRC) Doctoral Training Partnership. Computational results were obtained using ARCHIE-WeSt High Performance Computer (www.archie-west.ac.uk).

References

[1] Zhao, J., and Li, S., "Modified Multiresolution Technique for Mesh Refinement in Numerical Optimal Control," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 12, 2017, pp. 3324–3334. https://doi.org/10.2514/1.G002796

[2] Betts, J. T., and Huffman, W. P., "Mesh Refinement in Direct Transcription Methods for Optimal Control," *Optimal Control Applications and Methods*, Vol. 19, No. 1, 1998, pp. 1–21. https://doi.org/10.1002/(ISSN)1099-1514

[3] Jain, S., and Tsiotras, P., "Trajectory Optimization Using Multiresolution Techniques," *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 5, 2008, pp. 1424–1436. https://doi.org/10.2514/1.32220

[4] Betts, J. T., "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 193–207. https://doi.org/10.2514/2.4231

[5] Ross, I. M., and Fahroo, F., "Pseudospectral Knotting Methods for Solving Optimal Control Problems," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 3, 2004, pp. 397–405. https://doi.org/10.2514/1.3426

[6] Gong, Q., and Ross, I. M., "Autonomous Pseudospectral Knotting Methods for Space Mission Optimization," *16th AAS/AIAA Space Flight Mechanics Meeting*, American Astronautical Soc., San Diego, CA, Paper 06–151, 2006.

[7] Darby, C. L., Hager, W. W., and Rao, A. V., "An *hp*-Adaptive Pseudospectral Method for Solving Optimal Control Problems," *Optimal Control Applications and Methods*, Vol. 32, No. 4, 2011, pp. 476–502. https://doi.org/10.1002/oca.957

[8] Zhao, J., and Shang, T., "Dynamic Optimization Using Local Collocation Methods and Improved Multiresolution Technique," *Applied Sciences (Switzerland)*, Vol. 8, No. 9, 2018, p. 1680. https://doi.org/10.3390/app8091680

[9] Zhao, J., and Li, S., "Adaptive Mesh Refinement Method for Solving Optimal Control Problems Using Interpolation Error Analysis and Improved Data Compression," *Journal of the Franklin Institute*, Vol. 357, No. 3, 2020, pp. 1603–1627. https://doi.org/10.1016/j.jfranklin.2019.11.024

[10] Zhao, J., and Li, S., "Mars Atmospheric Entry Trajectory Optimization with Maximum Parachute Deployment Altitude Using Adaptive Mesh Refinement," *Acta Astronautica*, Vol. 160, July 2019, pp. 401–413. https://doi.org/10.1016/j.actaastro.2019.03.027

[11] Li, B., Zhang, H., Zheng, W., and Wang, L., "Spacecraft Close-Range Trajectory Planning via Convex Optimization and Multi-Resolution Technique," *Acta Astronautica*, Vol. 175, Oct. 2020, pp. 421–437. https://doi.org/10.1016/j.actaastro.2020.05.051

[12] Andersson, J., "A Survey of Multiobjective Optimization in Engineering Design," Ph.D. Thesis, Linktjping Univ., Linköping, Sweden, 2000.

[13] Yang, P., and Qi, R., "Reentry Trajectory Optimization for Hypersonic Vehicle Based on Improved Mesh Refinement Techniques," *Chinese Control Conference, CCC*, Vol. 2016, Inst. of Electrical and Electronics Engineers, New York, Aug. 2016, pp. 5473–5478. https://doi.org/10.1109/ChiCC.2016.7554207

[14] Tsiantis, N., and Banga, J. R., "Using Optimal Control to Understand Complex Metabolic Pathways," *BMC Bioinformatics*, Vol. 21, No. 1, 2020, pp. 1–33. https://doi.org/10.1186/s12859-020-03808-8

[15] Zou, W., Zhang, Q., Gao, Q., and Feng, Z., "Wavelet Multi-Resolution Approximation for Multiobjective Optimal Control," *PLoS ONE*, Vol. 13, No. 8, 2018, pp. 1–13. https://doi.org/10.1371/journal.pone.0201514

[16] de Weck, O. L., "Multiobjective Optimization: History and Promise," *Invited Keynote Paper, GL2-2, The Third China-Japan-Korea Joint Symposium on Optimization of Structural and Mechanical Systems*, Vol. 2, Kanazawa, Japan, Oct. 2004, p. 34.

[17] Santos, L. S., de Souza, K. M. F., Bandeira, M. R., Ruiz Ahón, V. R., Peixoto, F. C., and Prata, D. M., "Dynamic Optimization of a Continuous Gas Lift Process Using a Mesh Refining Sequential Method," *Journal of Petroleum Science and Engineering*, Vol. 165, June 2018, pp. 161–170. https://doi.org/10.1016/j.petrol.2018.02.019

[18] Ding, D., Guo, H., Huang, L., and Wang, Y., "Multi-Objective Optimization of Reentry Trajectory for Hypersonic Gliding Vehicle," *CGNCC. 2016 - 2016 IEEE Chinese Guidance, Navigation and Control Conference*, Inst. of Electrical and Electronics Engineers, New York, 2017, pp. 1436–1443. https://doi.org/10.1109/CGNCC.2016.7829000

[19] Messac, A., "Physical Programming: Effective Optimization for Computational Design," *AIAA Journal*, Vol. 34, No. 1, 1996, pp. 149–158. https://doi.org/10.2514/3.13035

[20] Betts, J. T., Biehn, N., Campbell, S. L., and Huffman, W. P., "Compensating for Order Variation in Mesh Refinement for Direct Transcription Methods II: Computational Experience," *Journal of Computational and Applied Mathematics*, Vol. 143, No. 2, 2002, pp. 237–261. https://doi.org/10.1016/S0377-0427(01)00509-X

[21] Betts, J. T., *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, Soc. for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2010, pp. 247–256, Chap. 6. https://doi.org/10.1137/1.9780898718577

[22] Ricciardi, L. A., and Vasile, M., "Improved Archiving and Search Strategies for Multi Agent Collaborative Search," *Computational Methods in Applied Sciences*, Vol. 48, Jan. 2019, pp. 435–455. https://doi.org/10.1007/978-3-319-89988-6_26

[23] Vasile, M., "Finite Elements in Time: A Direct Transcription Method for Optimal Control Problems," *AIAA/AAS Astrodynamics Specialist Conference*, AIAA Paper 2010-8275, 2010. https://doi.org/10.2514/6.2010-8275

[24] Ricciardi, L. A., and Vasile, M., "Direct Transcription of Optimal Control Problems with Finite Elements on Bernstein Basis," *Journal of Guidance, Control, and Dynamics*, Vol. 42, No. 2, 2019, pp. 229–243. https://doi.org/10.2514/1.G003753

[25] Harten, A., "Multiresolution Representation of Data: A General Framework," *SIAM Journal on Numerical Analysis*, Vol. 33, No. 3, 1996, pp. 1205–1256, https://www.jstor.org/stable/2158503.

[26] Mallat, S. G., "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation," *Fundamental Papers in Wavelet Theory*, Vol. 11, No. 7, 1989, pp. 674–693. https://doi.org/10.1109/34.192463

[27] "A Primer on Bezier Curves," Pomax, 2023, https://pomax.github.io/bezierinfo/.

[28] Citron, S. J., *Elements of Optimal Control*, Holt, Rinehart and Winston, New York, 1969.

[29] Vasile, M., and Zuiani, F., "Multi-Agent Collaborative Search: An Agent-Based Memetic Multi-Objective Optimization Algorithm Applied to Space Trajectory Design," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, Vol. 225, No. 11, 2011, pp. 1211–1227. https://doi.org/10.1177/0954410011410274

[30] Pascoletti, A., and Serafini, P., "Scalarizing Vector Optimization Problems," *Journal of Optimization Theory and Applications*, Vol. 42, No. 4, 1984, pp. 499–524. https://doi.org/10.1007/BF00934564

[31] Ricciardi, L. A., "Multi-Objective Hybrid Optimal Control with Application to Space Systems," Doctoral Dissertation, Univ. of Strathclyde, Glasgow, Scotland, 2019. https://doi.org/10.48730/3bws-4294

[32] Smith, S. F., "A Learning System Based on Genetic Adaptive Algorithms," Ph.D. Thesis, Univ. of Pittsburgh, Pittsburgh, PA, 1980. https://doi.org/10.5555/909835

[33] Koza, J. R., "The Genetic Programming Paradigm: Genetically Breeding Populations of Computer Programs to Solve Problems," *Dynamic, Genetic, and Chaotic Programming*, Stanford Univ., Dept. of Computer Science, Stanford, CA, June 1992, pp. 203–321. https://doi.org/10.1109/TAI.1990.130444

[34] Holland, J. H., "Genetic Algorithms," *Scientific American*, Vol. 267, No. 1, 1992, pp. 66–73, http://www.jstor.org/stable/24939139.

[35] Munetomo, M., Takai, Y., and Sato, Y., "Migration Scheme for the Genetic Adaptive Routing Algorithm," *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, Vol. 3, Inst. of Electrical and Electronics Engineers, New York, 1998, pp. 2774–2779. https://doi.org/10.1109/ICSMC.1998.725081

[36] Goldberg, D., "Messy Genetic Algorithms : Motivation, Analysis, and First Results," *Complex Systems*, Vol. 3, No. 5, 1989, pp. 493–530, http://www.complex-systems.com/abstracts/v03_i05_a05.html.

[37] Harvey, I., "*The SAGA Cross: The Mechanics of Recombination for Species with Variable-Length Genotypes*," *Parallel Problem Solving from Nature*, Vol. 2, Elsevier, Amsterdam, North-Holland, 1992, pp. 269–278.

[38] Hutt, B., and Warwick, K., "Synapsing Variable-Length Crossover: Meaningful Crossover for Variable-Length Genomes," *IEEE Transactions on Evolutionary Computation*, Vol. 11, No. 1, 2007, pp. 118–131. https://doi.org/10.1109/TEVC.2006.878096

[39] Burke, D. S., De Jong, K. A., Grefenstette, J. J., Ramsey, C. L., and Wu, A. S., "Putting More Genetics into Genetic Algorithms," *Evolutionary Computation*, Vol. 6, No. 4, 1998, pp. 387–410. https://doi.org/10.1162/evco.1998.6.4.387

[40] Qiongbing, Z., and Lixin, D., "A New Crossover Mechanism for Genetic Algorithms with Variable-Length Chromosomes for Path Optimization Problems," *Expert Systems with Applications*, Vol. 60, Oct. 2016, pp. 183–189. https://doi.org/10.1016/j.eswa.2016.04.005

[41] Madavan, N. K., "Multiobjective Optimization Using a Pareto Differential Evolution Approach," *Proceedings of the 2002 Congress on Evolutionary Computation*, Vol. 2, Inst. of Electrical and Electronics Engineers, New York, 2002, pp. 1145–1150. https://doi.org/10.1109/CEC.2002.1004404

[42] Schütze, O., Esquivel, X., Lara, A., and Coello, C. A., "Using the Averaged Hausdorff Distance as a Performance Measure in Evolutionary Multiobjective Optimization," *IEEE Transactions on Evolutionary Computation*, Vol. 16, No. 4, 2012, pp. 504–522. https://doi.org/10.1109/TEVC.2011.2161872

[43] Meditch, J., "On the Problem of Optimal Thrust Programming for a Soft Lunar Landing," *IEEE Transactions on Automatic Control*, Vol. 9, No. 4, 1964, pp. 477–484. https://doi.org/10.1109/TAC.1964.1105758

[44] Feng, Z., Zhang, Q., Ge, J., Peng, W., Yang, T., and Jie, J., "Mesh Adaptation Method for Optimal Control with Non-Smooth Control Using Second-Generation Wavelet," *IEEE Access*, Vol. 7, Sept. 2019, pp. 135,076–135,086. https://doi.org/10.1109/ACCESS.2019.2941539

[45] Merlevede, A., Åhl, H., and Troein, C., "Homology and Linkage in Crossover for Linear Genomes of Variable Length," *PLoS ONE*, Vol. 14, No. 1, 2019, pp. 1–16. https://doi.org/10.1371/journal.pone.0209712

[46] Ricciardi, L. A., and Vasile, M., "MODHOC – Multi Objective Direct Hybrid Optimal Control," 2018, https://strathprints.strath.ac.uk/67085/.

[47] Patterson, M. A., Hager, W. W., and Rao, A. V., "A ph Mesh Refinement Method for Optimal Control," *Optimal Control Applications and Methods*, Vol. 36, No. 4, 2015, pp. 398–421. https://doi.org/10.1002/oca.2114

J. Rogers
Associate Editor