Sensitivity-Driven Experimental Design to Facilitate Control of Dynamical Systems

Joseph Hart · Bart van Bloemen Waanders · Lisa Hood · Julie Parish

Received: 24 December 2021 / Accepted: 27 January 2023 / Published online: 13 March 2023
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract
Control of nonlinear dynamical systems is a complex and multifaceted process. Essential elements of many engineering systems include high-fidelity physics-based modeling, offline trajectory planning, feedback control design, and data acquisition strategies to reduce uncertainties. This article proposes an optimization-centric perspective which couples these elements in a cohesive framework. We introduce a novel use of hyper-differential sensitivity analysis to understand the sensitivity of feedback controllers to parametric uncertainty in physics-based models used for trajectory planning. These sensitivities provide a foundation to define an optimal experimental design which seeks to acquire data most relevant in reducing demand on the feedback controller. Our proposed framework is illustrated on the Zermelo navigation problem and a hypersonic trajectory control problem using data from NASA’s X-43 hypersonic flight tests.

Keywords Hyper-differential sensitivity analysis · Optimal experimental design · Optimal control · Trajectory planning

Communicated by Jason L. Speyer.

Bart van Bloemen Waanders
bartv@sandia.gov

Joseph Hart
joshart@sandia.gov

Lisa Hood
lghood@sandia.gov

Julie Parish
jparish@sandia.gov

1 Scientific Machine Learning, Albuquerque, USA
2 Navigation, Guidance & Ctrl II, Albuquerque, USA
3 Autonomy for Hypersonics, Albuquerque, USA
4 Sandia National Laboratories, P.O. Box 5800, Albuquerque, NM 87123, USA
1 Introduction

Optimal control has been successfully applied to a range of engineering applications from managing chemical plants to trajectory planning of aerospace vehicles. Even though solution techniques have reached impressive levels of maturity, achieving stable and robust performance while mitigating uncertainties remains a key challenge. For instance, trajectory planning for hypersonic vehicles utilizes primal–dual interior point solvers applied to dynamics that are discretized with time-adaptive [7] pseudospectral methods [8]. Yet significant sources of uncertainties, including environmental conditions and aerodynamic properties, result in deviations from the open-loop solution which are too great for feedback controllers to overcome. In hypersonic trajectory planning, approximating aerodynamics using wind tunnel experiments, flight data, or high-fidelity numerical models is expensive and challenging. Consequently, the goal of selecting the most informative data acquisition strategies limited to a sampling budget is an ongoing research topic.

This paper introduces post-optimality sensitivities as a metric to help determine the most effective sampling strategy. The goal is to improve the information content of the underlying models so that a feedback controller can better track the open-loop solution. The question of how, where, and when to sample is central and because sampling real data is expensive, a judicious approach must be implemented. We introduce a workflow that captures high-fidelity simulation through an approximation to serve as a constraint in an optimization formulation. Through the implicit function theorem, post-optimality sensitivities can identify the most influential sources of uncertainties relevant to optimal solutions. Hyper-differential sensitivity analysis (HDSA) utilizes advanced numerical linear algebra for efficient computation of these sensitivities. The algorithmic generality of HDSA has been demonstrated on a range of applications problems [12, 23, 24]. In the context of trajectory planning for hypersonic vehicles, a higher-order numerical model is sampled for aerodynamic coefficients in the open-loop formulation to determine solutions that avoid saturating feedback controllers.

Design of experiments (DOE) in the statistics literature [18] provides a mature theory for how experiments (physical or computer simulations) should be performed in support of specific statistical analysis goals. For instance, DOE has provided many state-of-the-art sampling algorithms to support analysis of computer models. We appeal to optimal experimental design (OED) theory [1, 4, 6, 10, 14, 20, 21, 25] which solves an optimization problem to produce a design minimizing uncertainty in the solution of an inverse problem. There are many ways to define optimality, but they are typically based on measuring posterior uncertainty in Bayesian formulations of the inverse problem. Examples of such uncertainty measurements are the trace (or determinant) of the posterior covariance and the Kullback–Leibler divergence between the prior and posterior distribution. The different measures of uncertainty are known as the alphabetic optimality criteria [4]. The focus of this paper will be on finding sampling strategies in the service of optimal control solutions rather than inverse problems. We take inspiration from the OED literature, but define a new optimality criterion based on HDSA. Our new control-oriented OED formulation is applicable for both physical and computer simulation experiments, though we focus on designing sampling strategies for computer simulations in this article.
Fig. 1 Analysis workflow consisting of three sequential phases: (1) trajectory ODEs supported by surrogates that feeds into an objective function, (2) open-loop optimal control formulation that feeds into a closed-loop optimal control problem which is instrumented for HDSA calculations, and (3) OED to determine the sampling strategy from a high-fidelity surrogate.

Our algorithmic approach is organized in three phases, each consisting of supporting elements, depicted in Fig. 1. The first phase concerns modeling. This feeds into the second phase where optimization is used to control the system. Given a model and control strategy, the third phase uses OED to direct data acquisition which seeks to reduce the modeling uncertainties most relevant to the control solution. The first phase encapsulates the dynamics through a set of ordinary differential equations (ODEs). These ODEs often contain parameters which are estimated via a table lookup or polynomial-based surrogates based on experiments or expensive high-fidelity computations. The users specification of an objective and constraints completes the necessary modeling to pose the optimization problems in phase 2. Within this phase, there are two distinct optimization problems, open-loop optimal control to generate a reference trajectory and closed-loop optimal control to equip the system with feedback to mitigate the effects of uncertainty. However, closed-loop control is challenging in the face of significant uncertainty, so we introduce HDSA on the closed-loop control problem to prioritize which uncertainties place the greatest strain on the feedback controller. This sensitivity information flows into phase 3 where we pose an OED problem which is solved using mixed-integer programming. The resulting design informs data acquisition which is incorporated into the phase 1 modeling to complete the iterative workflow.

Our contributions consist of (1) introducing hyper-differential sensitivities to identify sources of uncertainty that place the greatest demand on a feedback controller; (2) defining an optimal experimental design criterion which leverages the hyper-differential sensitivities to direct data acquisition most relevant to reducing demand on the feedback controller; and (3) demonstrating our proposed framework to find an optimal trajectory path using dynamics that are representative of a real hypersonic vehicle. The remainder of the article is organized as follows. Section 2 introduces a general problem formulation for nonlinear control motivated by our hypersonic flight.
exemplar. We present an interpretation of hyper-differential sensitivity analysis in Sect. 3 to facilitate a connection between trajectory planning and feedback control. These sensitivities are used to develop a novel optimal experimental design in Sect. 4. We illustrate these ideas in Sect. 5 on the Zermelo navigation problem, an insightful prototype, and an application to the NASA X-43 hypersonic vehicle [11]. A concluding perspective is given in Sect. 6.

2 Problem Formulation

We consider control of a nonlinear dynamical system governed by ordinary differential equations. Letting \( t \) denote time, \( T > 0 \) denote a final time, \( x : [0, T] \to \mathbb{R}^n \) be time-dependent state variables (\( \dot{x} \) denotes their time derivative), and \( u : [0, T] \to \mathbb{R}^m \) denote controllers, we consider

\[
\begin{aligned}
\min_{x, u, T} & \int_0^T C_{\text{run}}(x(t), u(t), t)dt + C_{\text{final}}(x(T), u(T), T) \\
\text{s.t.} & \quad \dot{x}(t) = f(t, x, u, g(x, u)) \quad t \in (0, T) \\
& \quad x(0) = x_0 \\
& \quad \eta(t, x, u, g(x, u)) \leq 0 \quad t \in (0, T) \\
& \quad \Upsilon(x(T)) = 0
\end{aligned}
\]

where \( C_{\text{run}} \) and \( C_{\text{final}} \) denote running and final time objective functions, \( \eta \in \mathbb{R}^r \) denotes a set of \( r \) inequality constraints and \( \Upsilon \in \mathbb{R}^s \) denotes a set of \( s \) terminal equality constraints. The constraints \( \eta \) and \( \Upsilon \) are frequently a critical part of the problem formulation where \( \eta \) ensures that the system operates in a safe regime (bounding temperature, pressure, etc.) while \( \Upsilon \) enforces satisfaction of a desired final state. The solution of (\( \text{OL}(g) \)) frequently called the open-loop control strategy.

We assume that the state dynamics

\[
f(t, x, u, g(x, u)) \in \mathbb{R}^n
\]

are of high fidelity, but the overall model fidelity is limited by errors in

\[
g(x, u) \in \mathbb{R}^\ell
\]

which models external forces, system properties, and/or the environment. In our motivating hypersonics application, \( f \) corresponds to the equation of motion for a vehicle and \( g \) models aerodynamic forces acting on it.

Throughout the article, we will assume that there exists a “true” model \( g^* \) which the system experiences in practice. Rather than knowing \( g^* \), we have access to an approximation \( \bar{g} \approx g^* \). Solving (\( \text{OL}(g) \)) with \( g = \bar{g} \) generates a reference solution which we denote as \((\bar{x}(t), \bar{u}(t), \bar{T})\). Running the system with controller \( u = \bar{u} \) is called open-loop control and will yield a state trajectory \( x_o \) which satisfies the dynamics (1)
Table 1
Summary of notation for the state, control, and model. In each row, the left column denotes the solution of the ODE system (1) with final time $T = \bar{T}$, control $u$ given by the center column, and model $g$ given by the right column.

| State | Control | Model |
|-------|---------|-------|
| $x$   | $\bar{u}$ | $\bar{g}$ |
| $x_o$ | $\bar{u}$ | $g^*$ |
| $x_c$ | $\bar{u} + \Delta u(x)$ | $g^*$ |

with $u = \bar{u}$ and $g = g^*$. For many systems, $x_o$ is far from $\bar{x}$ due to the error $g^* - \bar{g}$ and hence open-loop control is insufficient. To combat this challenge, closed-loop control seeks to use feedback of the system state in order to adjust the control strategy. In other words, it defines a feedback law $\Delta u(x)$, mapping the state to a control response, to augment $\bar{u}$. This yields a closed-loop trajectory $x_c$ which satisfies the dynamics (1) with $u = \bar{u} + \Delta u(x)$ and $g = g^*$. Table 1 summarizes this notation for the readers convenience.

Determining a feedback strategy $\Delta u(x)$ such that $x_c \approx \bar{x}$ is challenging for highly nonlinear systems. Prevalent challenges include physical constraints of the system which limit the computational power available for feedback, fast time scales which mandate rapid computation, and strong nonlinearities which require time-intensive computation to resolve. To combat this challenge, we consider evaluating $g^*(x, u)$ at a small number of state-control inputs which must be selected judiciously in order to produce a reference solution which may be more easily tracked by a feedback controller. We propose to use hyper-differential sensitivity analysis coupled with a novel optimal experimental design formulation to direct data acquisition. This guides offline computation to enable better trajectory generation and feedback control.

### 3 Hyper-Differential Sensitivity Analysis

To determine which sources of error in $g$ place the greatest strain on the feedback controller $\Delta u(x)$, let $e = x - \bar{x}$ denote the deviation of the state $x$ from the reference trajectory $\bar{x}$ and consider the reference tracking optimization problem

$$
\min_{e, v} \frac{1}{2} \int_0^T ||e(t)||^2 dt + \lambda \int_0^T ||v(t)||^2 dt \quad \text{(RT}(g))
$$

subject to

$$
\begin{cases}
\dot{e}(t) = f(t, \bar{x} + e, \bar{u} + v, g(\bar{x} + e, \bar{u} + v)) - f(t, \bar{x}, \bar{u}, g(\bar{x}, \bar{u})) \\
\dot{e}(0) = 0,
\end{cases}
$$

where $v : [0, T] \rightarrow \mathbb{R}^m$ augments the controller $\bar{u}$ and $\lambda \geq 0$. The ODE system (2) follows from differentiating $e = x - \bar{x}$ with respect to $t$ and substituting the dynamics (1). The user specification of $\lambda$ determines how much control effort is permitted. The choice is problem-specific as it relates to the control cost. It may be tuned if the solution appears to be overly conservative or aggressive. Also note that the inequal-

\[\text{Springer}\]
ity constraints \( \eta \) and terminal constraints \( \Upsilon \) are not included in \(( RT(\mathbf{g}))\). They are implicit since \( \mathbf{x} \) will satisfy them and \(( RT(\mathbf{g}))\) seeks to track \( \mathbf{x} \); however, omitting them simplifies our subsequent analysis.

For a given \( \mathbf{g} \), the solution of \(( RT(\mathbf{g}))\) gives the control \( \mathbf{v} \), augmenting \( \mathbf{u} \), such that the state optimally tracks \( \mathbf{x} \). We interpret \( \mathbf{v} \) as the optimal closed-loop controller, which cannot be computed in practice since \( \mathbf{g}^* \) is unknown, but is useful for analysis. Contrary to feedback controllers \( \Delta \mathbf{u} : \mathbb{R}^n \to \mathbb{R}^m \) which map the state \( \mathbf{x}(t) \) to a control strategy \( \Delta \mathbf{u}(\mathbf{x}(t)) \), \( \mathbf{v}(t) \) is defined on the time interval \([0, T]\). This allows us to consider the optimal closed-loop control without requiring a specific functional form for \( \Delta \mathbf{u}(\mathbf{x}) \), thus making our framework applicable for a wide range of feedback control strategies. To facilitate our analysis, we assume that \( \mathbf{v} \) and \( \mathbf{g} \) belong to Hilbert spaces (for instance, square integrable functions) and that \( \mathbf{f} \) and \( \mathbf{g} \) are twice continuously differentiable.

Hyper-differential sensitivity analysis (HDSA) [12] considers the sensitivity of the solution of an optimization problem with respect to perturbations of parameters appearing in the model. In the context of reference tracking, HDSA provides insights about how perturbations of \( \mathbf{g} \) change the optimal closed-loop controller \( \mathbf{v} \). To simplify our exposition, let \( \mathbf{s}(t, \mathbf{v}, \mathbf{g}) \in \mathbb{R}^n \) denote the solution of the reference tracking ODE system (2), which we assume to be unique for any given \(( \mathbf{v}, \mathbf{g}) \). Then, the reduced space problem

\[
\min_{\mathbf{v}} J(\mathbf{v}; \mathbf{g}) := \frac{1}{2} \int_{0}^{T} ||\mathbf{s}(t, \mathbf{v}(t), \mathbf{g})||^2 dt + \frac{\lambda}{2} \int_{0}^{T} ||\mathbf{v}(t)||^2 dt
\]  

(3)

admits the same control solution \( \mathbf{v} \) as \(( RT(\mathbf{g}))\). Observe that \( \mathbf{v} \equiv 0 \), the function mapping \([0, T]\) to 0, is the global minimizer for (3) when \( \mathbf{g} = \mathbf{\overline{g}} \) (since \( \mathbf{x} \) was generated with \( \mathbf{g} = \mathbf{\overline{g}} \)).

Noting that, for any fixed \( \mathbf{g} \), the derivative of \( J \) with respect to \( \mathbf{v} \), \( J_{\mathbf{v}} \), equals zero at the minima of \( J \), we apply the implicit function theorem to the equation

\[
J_{\mathbf{v}}(\mathbf{v}; \mathbf{g}) = 0.
\]

Since \( J_{\mathbf{v}}(\mathbf{v}, \mathbf{\overline{g}}) = 0 \), the implicit function theorem yields the existence of an operator

\[
\mathcal{F} : \mathcal{N}(\mathbf{\overline{g}}) \to \mathcal{N}(\mathbf{\overline{v}}),
\]

defined on neighborhoods of \( \mathbf{\overline{g}} \) and \( \mathbf{\overline{v}} \), such that

\[
J_{\mathbf{v}}(\mathcal{F}(\mathbf{g}); \mathbf{g}) = 0,
\]

i.e., \( \mathcal{F} \) maps models \( \mathbf{g} \) to optimal closed-loop controllers \( \mathbf{v} \). Further, \( \mathcal{F} \) is differentiable and its derivative is given by

\[
\mathcal{F}_{\mathbf{g}}(\mathbf{\overline{g}}) = -\mathcal{H}^{-1} \mathcal{B},
\]
where $H = J_{v,v}$ is the hessian of $J$ with respect to $v$, and $B = J_{v,g}$ is the mixed second derivative of $J$ with respect to $v$ and $g$, both evaluated at $(\bar{v}, \bar{g})$.

Since $F(g^*)$ defines the optimal closed-loop controller to track $x$, the control effort (norm of the closed-loop controller) required to track the reference trajectory may be approximated as

$$||F(g^*)|| = ||F(g^*) - F(\bar{g})|| \approx ||F_g(\bar{g})(g^* - \bar{g})||$$

since $F(\bar{g}) = 0$ and $F(g^*) - F(\bar{g}) = F_g(\bar{g})(g^* - \bar{g}) + O(||g^* - \bar{g}||^2)$ by Taylor’s theorem. This leads us to define the hyper-differential sensitivity for a perturbation in the direction $\delta g$ as

$$S(\delta g) = \frac{||F_g(\bar{g})\delta g||}{||\delta g||}.$$  \hspace{1cm} (4)

We interpret $S(\delta g)$ as the change in the optimal closed-loop controller $v$, augmenting $\bar{u}$, when $\bar{g}$ is perturbed in the direction $\delta g$.

Directions $\delta g$ which maximize $S$ correspond to errors in $g$ that will require the greatest feedback effort to overcome. Our proposed approach finds these high-sensitivity directions to inform an optimal experimental design which seeks to improve $\bar{g}$ in these directions, thus reducing demand on the feedback controller. In the context of our hypersonic application, $S(\delta g)$ identifies which flight configurations (a combination of altitude, velocity, and control surface geometry) will place the greatest stress on a feedback controller seeking to track the reference flight trajectory.

**Discretization**

Since $g$ is infinite dimensional (a function), we discretize perturbations of $\bar{g}$ and represent them with a finite-dimensional vector of parameters $\theta$. For simplicity of the exposition, we focus on a single component of $\bar{g}(x, u)$, without loss of generality denote it by $\bar{g}_k(x, u)$. In practice, the procedure described below may be done component-wise for a vector-valued $\bar{g}(x, u)$. Since $\bar{g}_k$ is a model which depends on $(x, u)$, we parameterize a perturbation of $\bar{g}_k$ in the form

$$p_k(\theta^k; x, u) = 1 + \sum_{i=1}^{L_k} \theta_i^k \phi_i^k(x, u),$$

where $\{\phi_i^k\}_{i=1}^{L_k}$ is a suitable basis (chosen by the user) and $\theta^k = (\theta_1^k, \theta_2^k, \ldots, \theta_{L_k}^k)^T \in \mathbb{R}^{L_k}$ is a vector parameterizing the perturbation. In practice, $\phi_i^k$ may only depend on a subset of $(x, u)$. We retain all input arguments for generality but $\phi_i^k$ may be constant with respect to some of the input arguments. As an example, if $\bar{g}_k$ only depends on the first state and first control variables, $x_1$ and $u_1$, then we may define a two-dimensional mesh over a range of $x_1$ and $u_1$ values and define $\{\phi_i^k\}_{i=1}^{L_k}$ as local basis functions on this mesh. Then, each component of $\theta$ corresponds to perturbing the model $g_k$ in a specific neighborhood of $(x_1, u_1)$ values. Because of the differentiability assumptions
in HDSA, we require that $\phi^k_i$ is twice continuously differentiable and suggest splines or radial basis functions as a useful discretization to accommodate smoothness and locality.

In general, $g(x, u) \in \mathbb{R}^\ell$ so we concatenate the vectors $\theta^k$ for each component to define

$$\theta = \begin{pmatrix} \theta^1 \\ \theta^2 \\ \vdots \\ \theta^\ell \end{pmatrix} \in \mathbb{R}^q$$

where $q = L_1 + L_2 + \cdots + L_\ell$. Letting $P(\theta) \in \mathbb{R}^{\ell \times \ell}$ be a diagonal matrix whose $(k, k)$ entry is $p_k(\theta^k; x, u)$, we observe that

$$P(\theta)\overline{g}(x, u)$$

parameterizes perturbations of $\overline{g}$ such that $\theta = 0$ corresponds to the nominal model $\overline{g}$.

Upon discretization of the control problem, which we summarize in Appendix A, the sensitivity operator $F_g(\overline{g})$ is approximated by the matrix $D \in \mathbb{R}^{K_m \times q}$, where $K_m$ is the dimension of the discretized control variables. If $K_m$ or $q$ is a moderate size (for instance $\mathcal{O}(10)$), we may construct and store $D$ explicitly. For large-scale applications where this is not possible, $D$ may be approximated using a singular value decomposition. This approach is efficient for large problems which admit low rank structure, as is commonly observed in engineering applications. We refer the reader to [12, 23] for more details.

4 Using HDSA to Inform Optimal Experimental Design

Our goal is to use the sensitivity information in $D$ to determine points $\{(x_i, u_i)\}$ where we will evaluate $g^*(x_i, u_i)$. By choosing points to reduce the error in the directions of greatest sensitivity, we generate a reference trajectory which the feedback controller may track more effectively. However, acquiring such data comes at a significant computational or experimental cost which is prohibitive. In the context of our motivating hypersonics example, the components of $g$ correspond to aerodynamic forces and are estimated by constructing surrogate models based on wind tunnel experiments or computational fluid dynamics simulations. As the aerodynamics depend on the vehicle geometry, controller positions, and Mach number, we have limited evaluations of $g$ relative to its complex dependencies on many variables and parameters. As the components of $\theta$ correspond to our estimation of $g$ at a particular configuration (geometry and Mach number), and the systems vary over many configurations in flight, it is imperative that we collect data in the most important configurations. A poor experimental design results in unproductive compute time and experimental efforts, and ultimately a delay or failure to yield a successful controller.
The proposed experimental design does not treat the uncertainties statistically, as it common in optimal experimental design for inverse problems [2, 3]. Rather, we work in a deterministic framework focusing on the error \( g^* - \overline{g} \) and consider which components of this error create the greatest demand on the closed-loop controller. For this reason, we will use the terminology error instead of uncertainty.

Our optimal experimental design criterion depends on three quantities, the hyper-differential sensitivity, the expected error in the model \( g \), and the informativeness of the experiments. We will combine these three quantities, represented by matrices \( D \), \( A \), and \( B \), to define an optimality criterion. The sensitivity information in \( D \) couples the dynamics, nominal model, and reference tracking objective, while the matrices \( A \) and \( B \), introduced below to represent error and information gain, permit the user to incorporate domain expertise when available. Decomposing the control problem, model error estimation, and information gain from experiments enables systematic integration of application-specific insights with mathematical rigor in the optimality criterion.

Let \( A \in \mathbb{R}^{q \times q} \) be a diagonal matrix whose \((i, i)\) entry is an estimation of the error in the \( i^{th} \) component of \( \theta \). In many cases, we have some a priori insights into the relative error between components of \( \theta \) and this information may be embedded in \( A \) so that the experimental design may exploit it. For instance, there may be greater uncertainty in aerodynamics at higher Mach numbers. If nothing is known about the error \( g^* - \overline{g} \) a priori, we may take \( A \) as the identity matrix. The resulting optimality criterion will be unaffected by scaling \( A \) by a constant, so capturing the magnitude of the error is irrelevant. Rather, the information we define in \( A \) is the magnitude of the error in the components of \( \theta \) relative to one another.

Performing an experiment (computational or physical) will give an improvement in our estimate of some components of \( \theta \). We represent this by defining a vector \( w = (w_1, w_2, \ldots, w_d)^T \in \{0, 1\}^d \) where \( d \) is the number of possible experiments and the entries of \( w \) are 1 if we conduct the experiment and 0 if we do not. We define

\[
 r_{i, j} \in [0, 1] \quad i = 1, 2, \ldots, q, \quad j = 1, 2, \ldots, d
\]

as the relative reduction in error in the \( i^{th} \) component of \( \theta \) when the \( j^{th} \) experiment is conducted. Assuming that components of \( \theta \) correspond to state-control pairs \((x_i, u_i)\), \( i = 1, 2, \ldots, d \), a useful error reduction model is

\[
 r_{i, j} = \exp \left( -\gamma_j^2 \| (x_i, u_i) - (x_j, u_j) \|^2 \right) - \epsilon_j,
\]

where \( \gamma_j > 0 \) is a correlation length parameter and \( \epsilon_j \geq 0 \) is the observation error which may be zero if high-fidelity experiments are performed, or may be adapted to model experiments of varying fidelities or noise levels. When no domain expertise is available, \( \gamma_j \) may be specified based on the smoothness of \( \overline{g} \); however, when appropriate the user may impose additional domain knowledge by the specification of it.
To model the error reduction provided by a set of experiments, define $B(w) \in \mathbb{R}^{q \times q}$ as the diagonal matrix whose $(i, i)$ entry is given by

$$(B(w))_{i,i} = b_i(w) = \left(1 - \sum_{j=1}^{d} w_j r_{i,j}\right)^+,$$

where $(x)^+ = \max\{x, 0\}$ is the plus function. This models our ability to inform the $i^{th}$ component of $\theta$ through a combination of experiments such that $b_i(w) \in [0, 1]$ is the relative reduction in error after conducting the experiments defined by $w$.

To the end that collecting new data reduces the demand on the feedback controller, we would like to minimize the hyper-differential sensitivities. Ideally, for any given design we should recompute the reference trajectory and sensitivities to measure the reduction in sensitivity by the design. This is computationally intractable, and rather we use the matrix $D$ which is computed once using the nominal $\bar{g}$. To model the decrease in sensitivity, we consider the updated sensitivity matrix $U(w) = DAB(w) \in \mathbb{R}^{Km \times q}$, which imbeds our estimation of error $A$ and its reduction $B(w)$ given the experiment $w$. We interpret $U(w)$ as an approximation of the hyper-differential sensitivities given that the experiments defined by $w$ are conducted.

Let $\kappa_j > 0$, $j = 1, 2, \ldots, d$ denote the cost of experiment $j$ and $\kappa_B > 0$ denote the experimental budget. We pose an optimization problem to minimize the sensitivity matrix $U(w)$ given the budget constraint $\kappa_B$. With an objective of minimizing the sensitivity in an average sense, and a desire to achieve computational efficiency, we adopt the Frobenius norm (squared for mathematical convenience). This yields the optimization problem

$$\min_{w \in \{0,1\}^d} \text{Tr}(U(w)^T U(w))$$

s.t.

$$\sum_{j=1}^{d} \kappa_j w_j \leq \kappa_B,$$

where $\text{Tr}$ denotes the matrix trace. A solution of (5) is called an optimal experimental design (including the possibility of multiple feasible designs yielding the same objective function value).

We observe that (5) is an integer program with a linear constraint. Inspection of the objective function reveals that we may reformulate (5) into a standard mixed-integer quadratic program. In particular, recalling that matrix multiplication commutes in the trace, $B(w)$ and $A$ are diagonal matrices, and

$$U(w)^T U(w) = B(w)^T A^T D^T DAB(w),$$
we have
\[
\operatorname{Tr}(U(w)^T U(w)) = \sum_{i=1}^{q} d_i^2 a_i^2 b_i(w)^2, \tag{6}
\]
where \(d_i^2\) and \(a_i\) are the diagonal entries of \(D^T D\) and \(A\), respectively. The scalar \(d_i\) corresponds to the sensitivity function (4) evaluated at the \(i^{th}\) basis function. Physically, we may interpret (6) as scaling the sensitivity \(d_i\) by the uncertainty level \(a_i\) and the expected information gain \(b_i(w)\). Letting \(r_i = (r_{i,1}, r_{i,2}, \ldots, r_{i,d})^T\) and \(c_i = d_i a_i\), \(i = 1, 2, \ldots, q\), (5) is equivalent to
\[
\min_{w \in \{0, 1\}^d} \sum_{i=1}^{q} c_i^2 \left(1 - w^T r_i\right)^2_s \tag{7}
\]
s.t.
\[
\sum_{j=1}^{d} \kappa_j w_j \leq \kappa_B.
\]
Since the entries of \(r_i\), \(i = 1, 2, \ldots, q\), are nonnegative, we may rewrite (7) as
\[
\min_{w \in \{0, 1\}^d, s \in \mathbb{R}^q} \sum_{i=1}^{q} c_i^2 s_i^2 \tag{8}
\]
s.t.
\[
\sum_{j=1}^{d} \kappa_j w_j \leq \kappa_B, \quad s_i \geq 0, \quad i = 1, 2, \ldots, q, \quad s_i \geq 1 - w^T r_i, \quad i = 1, 2, \ldots, q,
\]
where \(s = (s_1, s_2, \ldots, s_q)^T \in \mathbb{R}^q\) denotes slack variables which allow us to move the plus function into inequality constraints. This poses (8) in a standard form for a mixed-integer quadratic linear program.

Notice that the coefficients \(c_i\), \(i = 1, 2, \ldots, q\), may be computed a priori using \(D\) (or a low rank approximation of it) and \(A\) (which is user-defined). Hence, (8) may be fully specified to an integer programming code without requiring access to the original control problem or additional differential equation solves. Thanks to the maturity of quadratic linear programming we are able to generate optimal experimental designs efficiently. The results in this article are generated using Gurobi [9] to solve (8).

Our derivations are based on user-specified parameterizations of the model error and error reduction is achieved by collecting new data. We have not designed our
approach around uncertainty estimates build into statistical models, for instance, if \( g \) is a Gaussian process emulator. However, in such cases the uncertainty estimates may be used to define the entries of \( A \) and \( B \) instead of our proposed parameterizations. We presented the algorithm above to ensure generality without assuming error estimation information from \( g \), but our approach can incorporate error estimates when they are available.

5 Numerical Results

In this section, we provide two numerical results to illustrate important properties of the proposed method and demonstrate its effectiveness. Subsection 5.1 provides an illustrative example using the Zermelo problem. We demonstrate the method on a hypersonic trajectory control application using aerodynamics from the NASA X-43 hypersonic vehicle in Sect. 5.2.

In both examples, we use a pseudospectral discretization as described in Appendix A and we call IPOPT [26] to solve the resulting nonlinear program. The first and second derivatives are computed using algorithmic differentiation node-wise in the discretization and assembled to compute exact gradients, Jacobians, and Hessians for the discretized problem.

Algorithm 1 provides a summary of the proposed approach. In it, we let \( \overline{g} \) denote our nominal model, \((\overline{x}, \overline{u}, \overline{T})\) denote the reference solution generated using \( g = \overline{g} \), \( \tilde{g} \) denote our improved model after acquiring new data to improve the model, and \((\tilde{x}, \tilde{u}, \tilde{T})\) denote the updated reference solution generated using \( g = \tilde{g} \). Our goal is that the reference trajectory \( \tilde{x} \) is easier to track than \( x \) since the additional data we collect reduce error in the directions which place the greatest strain on the feedback controller.

Algorithm 1 Computation and use of HDSA for experimental design

1: Solve the open-loop problem (OL(\( g \))) with \( g = \overline{g} \) to generate a reference solution \((x, u, T)\)
2: Compute the hyper-differential sensitivities for the reference tracking problem (RT(\( g \)))
3: Determine the optimal experimental design by solving (8)
4: Evaluate \( g^* \) at the design points and use this data to fit an improved model \( \tilde{g} \)
5: Solve the open-loop problem (RT(\( g \))) with \( g = \tilde{g} \) to generate a reference solution \((\tilde{x}, \tilde{u}, \tilde{T})\)
6: Design a feedback controller to track \( \tilde{x} \)

5.1 Zermelo Problem

The Zermelo problem [27], a classical navigation problem in the optimal control literature, considers control of a boat being driven by a current. Mathematically, it is modeled by an ODE system with states \( x = (x_1, x_2) \) corresponding to the position of a boat along a river (with the river running parallel to the \( x_1 \)-axis). The left panel of Fig. 2 depicts the problem setup. The optimal control problem under consideration is
\[
\min_{x, u} -x_1(1) \quad \text{(9)}
\]

\[
\begin{aligned}
\dot{x}_1(t) &= \cos(u(t)) + g(x_1(t))x_2(t) \quad t \in (0, 1) \\
\dot{x}_2(t) &= \sin(u(t)) \quad t \in (0, 1) \\
x_1(0) &= 0 \\
x_2(0) &= 0 \\
-\pi &\leq u(t) \leq \pi \quad t \in (0, 1) \\
x_2(1) &= 0,
\end{aligned}
\]

where \( u : [0, 1] \to \mathbb{R} \) is the heading of the boat which is driven by the current \( g(x_1(t))x_2(t) \). This corresponds to the boat going out into the river \((x_2 > 0)\) and being driven downstream (in the positive \( x_1 \) direction) and then navigating back to the river bank \( x_2 = 0 \) at time \( t = 1 \). The optimization objective is to maximize the range downstream.

We consider uncertainty in the current \( g(x_1) \) which we seek to reduce in order to develop a robust control strategy. The right panel of Fig. 2 shows the “true” \( g \), denoted as \( g^* \), and our initial estimate of it, denoted as \( \overline{g} \). We assume that \( g^* \) may be evaluated at a limited number of \( x_1 \) values to improve \( \overline{g} \). Our goal is to determine where to evaluate \( g^* \) so that we may ensure control of the system. This emulates the common application scenario where model parameters are surrogates constructed from limited evaluations of high-fidelity models or controlled experiments.

**Trajectory and Controller**

We solve the trajectory optimization problem (9) with \( \overline{g} \) to generate a reference trajectory and controller \((\overline{x}, \overline{u})\). A LQR (linear quadratic regulator) feedback controller is used to track the nominal trajectory \( \overline{x} \). In particular, we linearize the dynamics along the nominal trajectory and subtract \( \overline{x} \), then solve a Riccati equation to deter-
mine optimal feedback gains for the linearized system. Though this fails to account for nonlinear interactions in the dynamics, it provides a computationally efficient method for online feedback control to reduce deviations due to parametric uncertainty. The left panel of Fig. 3 displays the reference trajectory $\bar{x}$, the open-loop trajectory $x_o$, and the closed-loop trajectory $x_c$. The right panel of Fig. 3 shows the reference (open loop) and feedback (closed loop) controllers, $\bar{u}$ and $\bar{u} + \Delta u$.

The open-loop trajectory undershoots the reference trajectory as a result of $g$ overestimating $\bar{g}$ for larger values of $x_1$. The feedback reduces this overshoot, but is unable to successfully track the reference trajectory.

**Hyper-Differential Sensitivities**

We adopt radial basis functions $\phi_i(x_1) = \exp(-w(x_1 - c_i)^2), i = 1, 2, \ldots, 30$, which are depicted in the left panel of Fig. 4, to discretize $g$. The resulting sensitivities, as a function of $x_1$, are shown in the right panel of Fig. 4.

To gain greater physical insight from these sensitivities, Fig. 5 plots $\bar{x}$ in the $x_1x_2$ phase space (left) and $\dot{x}_2(t)$ in the time domain (right) with the sensitivities overlaid with colored dots (the color denoting the sensitivity magnitude). We observe that the
highest sensitivity occurs when \( \dot{x}_2 \) is nearly its minimum. This portion of the trajectory, around \( t = 0.85 \), is when the boat is aggressively maneuvering back to the shore as the large magnitude negative \( x_2 \) velocity is rapidly pushing the boat back toward the shore. The region of second greatest sensitivity is around \( t = 0.3 \) when the boat is reducing its \( x_2 \) velocity so that it does not float too far out into the river (in the \( x_2 \) direction). We observe that these high-sensitivity regions correspond to areas where \( \dot{x}_2(t) \) exhibits its greatest nonlinearity as a function of \( t \). This highlights the state coupling as the sensitivities correspond to a discretization of \( g(x_1) \), but the high-sensitivity regions correspond to nonlinearities in \( \dot{x}_2 \).

**Experimental Design**

Following the formulation of Sect. 4, we seek an experimental design to determine \( \kappa_B = 3 \) design points for which we will evaluate \( g^\star \). We take a uniform mesh on \( x_1 \) with 30 nodes, \( A \) as the identity matrix (reflecting no prior knowledge of the error \( g^\star - \overline{g} \)), and an error reduction model

\[
 r_{i,j} = \exp(-\gamma_i^2(x_1(w_i) - x_1(w_j))^2),
\]

where \( x_1(w_k) \) denotes the \( k^{th} \) node in the \( x_1 \) mesh and \( \gamma_i \) is a user-defined correlation length parameter determined by scaling a nominal value by the integral of \( \phi_i | \frac{\partial \overline{g}}{\partial x_1} | \) so that it is informed by the smoothness of \( \overline{g} \). With an experiment cost \( \kappa_j = 1 \) for all \( j \) and budget \( \kappa_B = 3 \), we solve (8) to determine an experimental design.

Using evaluations of \( g^\star \) at these three points, we define the improved model \( \tilde{g} = \overline{g} + \sum_{j=1}^{3} \beta_j \psi_j \), where the \( \psi_j \)'s are radial basis functions centered at the experimental design points and the \( \beta_j \)'s are chosen so that \( \tilde{g} \) interpolates \( g^\star \) at the design points. The improved surrogate \( \tilde{g} \) is displayed in Fig. 6 alongside \( g^\star \) and \( \overline{g} \). The sensitivities from Fig. 4 are overlaid in Fig. 6 to highlight the position of the experimental design points (indicated by black dots) corresponding to the high-sensitivity regions. In particular, two points are near the local maxima of the sensitivity curve, while the other point is placed between them. These design points are seeking to balance the high sensitivity with uniformity and placement of points in regions of greater nonlinearity.
Fig. 6 True, initially estimated, and improved models $g^*$, $\tilde{g}$, and $\hat{g}$ (left axis), with experimental design points, denoted by black dots, and hyper-differential sensitivities (right axis).

Fig. 7 Trajectories in the $x_1, x_2$ phase space (left) and controllers in the time domain (right) with the reference solution $(\tilde{x}, \tilde{u})$ generated using the nominal current model $\tilde{g}$. The reference trajectory $\tilde{x}$ is the solution of (9) with $g = \tilde{g}$, while the open- and closed-loop trajectories are generated using the controllers $\tilde{u}$ and $\tilde{u} + \Delta u$ with the current model $g^*$.

**Improved Trajectory and Controller**

We resolve the open-loop and LQR feedback control problems using the improved model $g = \hat{g}$. The updated reference, open-loop, and closed-loop trajectories are shown in Fig. 7. We observe that the open-loop solution is closer to the reference since the uncertainty in $g$ was reduced. The closed-loop solution improves the open solution with a small feedback effort to track the reference trajectory (compare the open- and closed-loop controllers in the right panel of Fig. 7). In particular, there is a notable decrease in the feedback control effort from Fig. 3 to Fig. 7. This was the goal of the optimal experimental design.
Comparison of Experimental Designs

Our previous analysis does not demonstrate how beneficial the proposed experimental design is relative to other options (for instance, an analyst manually choosing where to evaluate $g^*$ without aid from computation). To address this question, we perform a computational test in which we sample different experimental designs. In particular, we generate 100 different random 3 point designs, and for each design measure the reference tracking error and closed-loop control effort. To ensure a fair comparison, for each random design we (1) compute $\tilde{g}$ by augmenting $g$ with the new samples, (2) resolve the open-loop problem to generate a reference trajectory corresponding to $\tilde{g}$, (3) and determine an LQR feedback with $g = \tilde{g}$. We repeat this experiment for 100 different high-fidelity models (generated by perturbing $\bar{g}$ with a random linear combination of polynomial basis functions) and average the resulting closed-loop control effort and reference tracking error.

Figure 8 displays the average feedback control effort and reference tracking error using the 100 random designs as well as the optimal design from Fig. 8.

We observe that the experimental design generated by solving (8) is not optimal in the sense of attaining a minimum feedback effort to track the reference trajectory; however, we should not expect it to be. The design generated by solving (8) is optimal for the error model specification, information gain assumptions ($r_{ij}$), and the linear approximation HDSA makes to the parameter to optimal controller mapping. The error and information gain quantities will never be fully known in practice; nonetheless, our framework provides a pragmatic way to incorporate best estimates of them with computational insights in a systemic framework which outperforms the majority of random designs. The design may be easily improved when domain expertise is available, but is not dependent on it.
5.2 X-43 Trajectory Control

Control, guidance, and navigation of hypersonic vehicles is a challenging problem and active research area due to complex nonlinear physics and fast time scales. We demonstrate the utility of our proposed method to control a three-degree-of-freedom point-mass model of a hypersonic glide vehicle using an aerodynamic surrogate model trained on data from the NASA X-43 flight tests [13, 15–17]. Our algorithms are developed to aid future designs and trajectory plans, but the X-43 serves as a useful prototype thanks to the availability of its aerodynamic data.

Let \((x, y, z)\) be the position of the vehicle modeled in the north, east, down coordinates, \(v\) be the velocity, \(\gamma\) be the flight path angle, and \(\psi\) be the azimuth angle. The vehicle is controlled by the angle-of-attack \(\alpha\), side slip angle \(\beta\), and roll angle \(\phi\). This gives state and control vectors \(x = (x, y, z, v, \gamma, \psi)^T \in \mathbb{R}^6\) and \(u = (\alpha, \beta, \phi)^T \in \mathbb{R}^3\), respectively. Given an initial state \(x_0 \in \mathbb{R}^6\) and target position \((x_{\text{target}}, y_{\text{target}}, z_{\text{target}})\), we consider the free final time optimal control problem

\[
\min_{x,u,T} (x - x_{\text{target}})^2 + (y - y_{\text{target}})^2 + (z - z_{\text{target}})^2
\]

s.t.

\[
\begin{align*}
\dot{x} &= v \cos(\gamma) \cos(\psi) & t \in (0, T) \\
\dot{y} &= v \cos(\gamma) \sin(\psi) & t \in (0, T) \\
\dot{z} &= -v \sin(\gamma) & t \in (0, T) \\
\dot{v} &= -D_m - g \sin(\gamma) & t \in (0, T) \\
\dot{\gamma} &= \frac{L \cos(\phi) - S \sin(\phi)}{m v} - \frac{g \cos(\gamma)}{v} & t \in (0, T) \\
\dot{\psi} &= \frac{L \sin(\phi) + S \cos(\phi)}{m v \cos(\gamma)} & t \in (0, T) \\
x(0) &= x_0
\end{align*}
\]

\[
\begin{align*}
&k Q \sqrt{\rho v^3} \leq Q_{\text{max}} & t \in (0, T) \\
&\frac{1}{2} \rho v^2 \leq q_{\text{max}} & t \in (0, T) \\
&\sqrt{L^2 + D^2} \leq n_{\text{max}} & t \in (0, T) \\
&(x - c_x)^2 + (y - c_y)^2 \leq r_c^2 & t \in (0, T) \\
&\alpha_l \leq \alpha \leq \alpha_u & t \in (0, T) \\
&\beta_l \leq \beta \leq \beta_u & t \in (0, T) \\
&\phi_l \leq \phi \leq \phi_u & t \in (0, T),
\end{align*}
\]

where (12) are path constraints corresponding to the maximum heating rate \(Q_{\text{max}}\), dynamic pressure \(q_{\text{max}}\), load factor \(n_{\text{max}}\), and a cylindrical no flight zone centered at \((x_c, y_c)\) with radius \(r_c\), and controller bounds \(\alpha_l, \alpha_u, \beta_l, \beta_u, \phi_l, \phi_u\).

The ODE system (11) and path constraints (12) depend on the aerodynamics model for drag, lift, and side forces

\[
D = (F_N \sin(\alpha) + F_A \cos(\alpha)) \cos(\beta) - F_Y \sin(\beta)
\]

\[
L = F_N \cos(\alpha) - F_A \sin(\alpha)
\]
\[ S = (F_N \sin(\alpha) + F_A \cos(\alpha)) \sin(\beta) + F_Y \cos(\beta), \]

where

\[
F_N = \frac{1}{2} \rho v^2 A_{ref} C_N \\
F_A = \frac{1}{2} \rho v^2 A_{ref} C_A \\
F_Y = \frac{1}{2} \rho v^2 A_{ref} C_Y
\]

are the aerodynamic forces in the normal, axial, and yaw directions and \( C_N(M, \alpha) \), \( C_A(M, \alpha) \), and \( C_Y(M, \alpha, \beta) \) are the non-dimensional aerodynamic coefficients which depend on the Mach number (a function of velocity and altitude) \( M \), angle-of-attack \( \alpha \), and in the case of \( C_Y \), the side slip angle \( \beta \). An exponential model \( \rho = \rho_0 \exp(z/H) \) is used to emulate the atmospheric density. The constants \( m \), \( A_{ref} \), and \( k_Q \) are the mass, reference area, and heat flux coefficient of the vehicle, respectively; \( g \) is the acceleration due to gravity.

All results will be reported in normalized units. In particular, each quantity is normalized via the linear transformation \( L(x) = (x - a)/(b - a) \), where \( a \) and \( b \) are the minimum and maximum values, respective. For the controllers, these minimum and maximum values are the bounds over which the aerodynamic model is valid. For the other quantities such as time, position, velocity, etc., the lower and upper bounds are the minimum and maximum values attained in the trajectory.

We consider uncertainty in the aerodynamic coefficients

\[
g(M, \alpha, \beta) = (C_N(M, \alpha), C_A(M, \alpha), C_Y(M, \alpha, \beta))^T \in \mathbb{R}^3,
\]

which are determined by fitting a surrogate model to aerodynamic data. Notice that \( g \) depends on \( M \) which is a function of state variables rather than a state variable itself, and that the third component of \( g \) depends on an additional variable, \( \beta \), which the first two components do not.

**Trajectory and Controller**

A nominal estimate for the aerodynamic coefficients, \( \overline{g} \), is given by a polynomial model constructed from the NASA X-43 flight data. To illustrate the concepts of this article, we define the true aerodynamics \( g^* \), which are not available in practice, as a perturbation of \( \overline{g} \) which we defined so that we can analyze the controller performance (perform experiments with \( g^* \)). We solve the trajectory optimization problem (10) with \( \overline{g} \) to generate a reference trajectory and controller \((\overline{x}, \overline{u})\) and adopt an LQR feedback controller as in the Zermelo problem.

Figure 9 displays the reference trajectory \( \overline{x} \), the open-loop trajectory \( x_o \), and the closed-loop trajectory \( x_c \), in the spatial coordinates \((x, y, z)\). In particular, the top panel shows the trajectories in 3D, while the bottom row shows each 2D cross section. The no fly zone is shown by the cylinder and time stamps (in normalized time units) mark the location of the vehicle at times \( t = 0, 0.2, 0.4, 0.6, 0.8, 1.0 \). We observe that
Fig. 9 Trajectories with a reference solution generated using the nominal aerodynamic model $\mathbf{g}$. The reference trajectory $\mathbf{x}$ is the solution of (10) with $\mathbf{g} = \mathbf{g}$, while the open- and closed-loop trajectories are generated using the aerodynamic model $\mathbf{g}^\star$. The top panel shows the trajectory in three-dimensional space, while the bottom row shows two-dimensional views of the trajectory for easier visualization.

The open-loop trajectory $\mathbf{x}_o$ diverges far from the reference trajectory $\mathbf{x}$ due to errors in the aerodynamic coefficients. The addition of the LQR feedback yields a closed-loop trajectory $\mathbf{x}_c$ which is improved, but is still insufficient to track the reference. Figure 10 shows the controllers corresponding to the trajectories in Fig. 9. The open-loop controller is $\mathbf{u}$, computed by solving (11), and the closed-loop controller is $\mathbf{u} + \Delta \mathbf{u}$, generated using the LQR feedback controller. The horizontal lines correspond to the upper and lower bounds on the controllers (12). Considerable feedback effort is needed after $t = 0.6$, and it violates the bounds on the controller. This large feedback effort, which improves the trajectory but still fails to track the reference, is what we seek to reduce via better characterization of the aerodynamics in the most important flight configurations.

Hyper-Differential Sensitivities and Experimental Design

We compute hyper-differential sensitivities to determine which flight configurations $(M, \alpha, \beta)$ result in aerodynamic uncertainty placing the greatest demand on the feedback controller. Figure 11 displays the hyper-differential sensitivities, plotted in the $(M, \alpha, \beta)$ phase space, with the reference trajectory $\mathbf{x}$ overlaid. Note that the aerodynamic coefficients $C_N$ and $C_A$ only depend on $M$ and $\alpha$ so we plot them in a two-dimensional phase space. Since $C_Y$ also depends on $\beta$, we provide two plots to
Fig. 10 Controllers used to generate the trajectories in Fig. 9. The open-loop controller $\mathbf{u}$ is generated by solving (10) with $\mathbf{g} = \mathbf{g}$, while the closed-loop controller $\mathbf{u} + \Delta \mathbf{u}$ uses LQR feedback around the nominal trajectory. The horizontal lines denote the lower and upper bounds on the controllers.

show the sensitivities. The top panel of Fig. 11 is the three-dimensional view, while the bottom right panel projects $C_Y$ onto the $(M, \alpha)$ phase space by averaging in the $\beta$ coordinate and renormalizing so that the maximum sensitivity in the $(M, \alpha)$ phase space equals the maximum in the $(M, \alpha, \beta)$ phase space. This additional normalization is necessary because the sensitivities are along a two-dimensional curve in a three-dimensional phase space and hence typical averaging is not appropriate.

We observe that the greatest sensitivity corresponds to axial force ($C_A$) during the initial glide phase $t \in (0, .5)$ and the normal force ($C_N$) during the diving phase...
Fig. 12 Trajectories with a reference solution generated using the nominal aerodynamic model $\tilde{g}$. The reference trajectory $\tilde{x}$ is the solution of (10) with $g = \tilde{g}$, while the open- and closed-loop trajectories are generated using the aerodynamic model $g^*$. The top panel shows the trajectory in three-dimensional space, while the bottom row shows two-dimensional views of the trajectory for easier visualization.

Fig. 13 Controllers used to generate the trajectories in Fig. 12. The open-loop controller $\tilde{u}$ is generated by solving (10) with $g = \tilde{g}$, while the closed-loop controller $\tilde{u} + \Delta u$ uses LQR feedback around the nominal trajectory. The horizontal lines denote the lower and upper bounds on the controllers.

$t \in (0.65, 0.75)$. This is because the axial force during the glide phase dictates the range of the vehicle over this longer time horizon, while the normal force during the dive phase determines how quickly the vehicle descends. Notice that the high axial force sensitivity during the glide phase, $t \in (0, .5)$, results in needing feedback during the dive phase. This is because the vehicle is less controllable at higher altitude due to the smaller atmospheric density, so the range error induced by poor estimation of aerodynamics in this portion of the trajectory must be compensated for later in the flight. The high normal force sensitivity during the diving phase, $t \in (0.65, 0.75)$,
also demands feedback during the same time interval. The cumulative effect of these uncertainties explains why considerable feedback effort is needed for $\tau > 0.6$.

An optimal experimental design is computed to determine $\kappa_B = 8$ points in the $(M, \alpha, \beta)$ phase space where we evaluate the true aerodynamics $g^*$. These points are indicated by the cyan dots in Fig. 11 which are concentrated in the high-sensitivity region and spread in such a way as to optimize the information gain by interpolating between them. In this example, we define the error reduction model $r_{i,j}$ using the derivatives of $\tilde{g}$ so that it reflects smoothness in the aerodynamics.

**Improved Trajectory and Controller**

Using the optimal experimental design, we evaluate the high-fidelity model $g^*$ at $\kappa_B = 8$ points and use this data to improve the aerodynamic model. As in the Zermelo problem, $\tilde{g}$ is defined by adding radial basis functions to $\tilde{g}$ so that $\tilde{g}$ interpolates $g^*$ at the design points. The trajectory planning problem (10) is resolved with the improved aerodynamic model $\tilde{g}$. Figure 12 shows the reference trajectory $\tilde{x}$, open-loop trajectory, and closed-loop trajectory (generated using an LQR feedback) with this improved aerodynamic model. We observe that the closed-loop trajectory does not perfectly track the reference $\tilde{x}$; however, it is a significant improvement relative to the previous trajectory in Fig. 9. More notably, Fig. 13 shows the open- and closed-loop controllers corresponding to these trajectories. We observe that the feedback effort needed in the closed-loop setting is considerably less than what was previously required in Fig. 10. This illustrates the utility of the proposed approach in that we were able to avoid violating the control bounds and significantly reduce the demand on the feedback controller.

**6 Conclusion**

Optimal control in the face of uncertainty and nonlinearity is challenging in many respects. A variety of approaches exist for both generating nominal trajectories which are robust to uncertainty and designing feedback controllers which aid to mitigate uncertainties. The proposed hyper-differential sensitivity analysis (HDSA) and optimal experimental design approach introduced in this article complement these approaches by seeking to prioritize uncertainties and direct data acquisition which may reduce them. For challenging problems such as autonomous hypersonic flight, wind tunnel testing, computational fluid dynamics, trajectory planning, and feedback control must be organically coupled to overcome the vast uncertainty, strong nonlinearity, and fast time scales faced in practice. As a tool which bridges these areas, hyper-differential sensitivity analysis is poised to contribute invaluable insight and guidance for vehicle development, flight planning, and onboard control.

The contributions of this article focused on how HDSA directs data acquisition in the service of improving trajectory planning. Our approach was demonstrated on a prototype using aerodynamics from the NASA X-43 vehicle, which serves as a useful prototype for future hypersonic vehicle developments. Because of its interpretation as the feedback control effort required to overcome uncertainty, there is opportunity to leverage HDSA for other aspects of both trajectory planning and feedback controller
design. By analyzing the sensitivity of the reference tracking optimization problem, rather than simply the model dynamics, HDSA provides unique insight which support decision-making.

The interplay between uncertainty calibration, trajectory planning, and feedback control pervades many applications and hence the proposed approach, which was demonstrated for control of a hypersonic vehicle, has broad applicability in the engineering sciences. The sensitivities may be computed rapidly, making them applicable for online processes. As a whole, our optimal experimental design approach may also be useful in online settings if the high-fidelity data source may be queried in a real-time fashion, for instance, through experimental measurements during operation. Such ideas may be explored in the context of engineering systems such as manufacturing or chemical processing.

Our sensitivity-driven optimal experimental design presents a new perspective on data acquisition for optimal control problems. Traditionally, experimental design has focused on statistical estimation and seeks to achieve desirable statistical properties. Our proposed approach is deterministic and grounded in optimal control, yet mimics many important characteristics of its statistical counterpart.

Acknowledgements This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the US Department of Energy or the US Government. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International, Inc., for the US Department of Energy’s National Nuclear Security Administration under contract DE-NA-0003525. The dynamics for the Zermelo navigation problem can be recreated from the mathematical description in the paper. The aerodynamic coefficients for the X-43 vehicle are proprietary and are not available. Other datasets generated during and/or analyzed during the current study are not publicly available due to Sandia National Laboratories policies, but special arrangements could be considered upon reasonable request.

Appendix A

This appendix provides details on the discretization of optimal control problems using adaptive pseudospectral methods. The fundamental ideas are common in fluid dynamics [5] and became popular in trajectory planning thanks to the pioneering work of Fahroo and Ross [8, 22]. Adaptive pseudospectral methods have gained significant attention in recent years thanks to the work of Rao and collaborators [7, 19].

For simplicity of the exposition, we consider problems of the form

$$\min_{x, u} \int_0^T C_{\text{run}}(x(t), u(t), t) dt + C_{\text{final}}(x(T), u(T))$$  

s.t.

$$\dot{x}(t) = f(t, x, u) \quad t \in (0, T)$$
$$x(0) = x_0.$$  

(OC)

(13)
and note that the subsequent developments may be easily extended for the more general case with inequality constraints, final time constraints, and a free final time. We have suppressed dependence on $g$ for notational simplicity.

Pseudospectral methods discretize the dynamics (13) by representing the state and control in a finite-dimensional basis and collocating the ODE system at a finite number of nodes in time. In particular, we consider a partition of the time interval $[0, T]$ into $p$ subintervals

$$
(0, T) = \bigcup_{i=1}^{p} (t_{i-1}, t_i),
$$

where $0 = t_0 < t_1 < t_2 < \cdots < t_p = T$. We approximate the state variables using $N$ local (in the subintervals) Lagrange polynomials at Gauss–Lobatto nodes. Let $\{\mathcal{Y}_i\}_{i=1}^{N}$ denote this set of basis functions and $y = (y_1^1, y_2^1, \ldots, y_N^1, \ldots, y_1^2, \ldots, y_N^2, \ldots, y_n^m)^T \in \mathbb{R}^{Kn}$, $Kn = nN$, denote the coordinate representation of the approximation, i.e.,

$$
x_k(t) \approx \sum_{i=1}^{N} y_i^k \mathcal{Y}_i(t).
$$

The controller is also discretized via an expansion in basis functions $\{\mathcal{Z}_j\}_{j=1}^{M}$; however, they may and in many cases are different than $\{\mathcal{Y}_i\}_{i=1}^{N}$. In particular, the state basis functions are adapted to resolve the dynamics, whereas the control basis is designed to enforce the users desired smoothness in the control solution, although it may also be adapted to focus nodes in regions with fast time scales if the user desires. Let $z = (z_1^1, z_2^1, \ldots, z_M^1, \ldots, z_1^2, \ldots, z_M^2, \ldots, z_m^M)^T \in \mathbb{R}^{Km}$, $Km = mM$, be the coordinates for the control, i.e.,

$$
u_k(t) \approx \sum_{j=1}^{M} z_j^k \mathcal{Z}_j(t).
$$

We collocate the dynamics (13) by differentiating (15) and evaluating the derivative at the time nodes. Enforcing that $\dot{x}(t) = f(t, x, u)$ at the time nodes yields the system of equations

$$
r_{coll}(y, z) = Y_t y - \xi(y, z) = 0,
$$

where $\xi(y, z)$ is a vector corresponding to stacking together evaluations of $f(t, x, u)$ at the time nodes and $Y_t$ is a matrix populated with time derivatives of the basis functions $\{\mathcal{Y}_i\}$. The vector $r_{coll}(y, z)$ contains $(N - p - 1)n$ nonlinear equations as we do not collocate at $\{t_i\}_{i=0}^{p}$, the interface, initial, and terminal time nodes. Rather, the initial conditions are enforced directly yielding $n$ equations which we denote as $r_{init}(y, z) \in \mathbb{R}^n$, and the terminal nodes on each subinterval $\{t_i\}_{i=1}^{p}$ are enforced by requiring that $x(t_i) - x(t_{i-1}) = \int_{t_{i-1}}^{t_i} f(t, x, u)$. This yields $pn$ additional equations.
which we denote as \( r_{\text{inter}}(y, z) \in \mathbb{R}^{pn} \). Combining \( r_{\text{coll}}, r_{\text{init}}, \) and \( r_{\text{inter}} \) yields a system of \( nN \) nonlinear equations

\[
\mathbf{r}(y, z) = \begin{pmatrix} r_{\text{coll}}(y, z) \\ r_{\text{init}}(y, z) \\ r_{\text{inter}}(y, z) \end{pmatrix} = 0
\]

whose solution is the coordinates for an approximate solution of the dynamics (13). To ensure a quality approximation, the time interval partition (14) is adapted to allocate time nodes in regions of faster dynamics.

The objective function is discretized by approximating the integral term

\[
\int_0^T C_{\text{run}}(x(t), u(t), t)dt \approx \sum_{i=1}^N w_i c_{\text{run}}^i(y, z),
\]

where \( \{w_i\}_{i=1}^N \) are the integration weights corresponding to applying Gauss–Lobatto integration on the subintervals and \( c_{\text{run}}^i(y, z) \) is the evaluation of \( C_{\text{run}} \) at the \( i^{th} \) time node using the discretized state and control. We represent the final time objective \( C_{\text{final}}(x(T), u(T)) \) with \( c_{\text{final}}(y, z) \) by evaluating \( C_{\text{final}} \) using the coordinates for the final time node.

The discretized optimal control problem is

\[
\min_{y, z} \sum_{i=1}^N w_i c_{\text{run}}^i(y, z) + c_{\text{final}}(y, z) \\
\text{s.t.} \\
r(y, z) = 0
\]

and may be solved using standard nonlinear programming methods.

References

1. Alexanderian, A., Gloor, P.J., Ghattas, O.: On Bayesian A- and D-Optimal Experimental Designs in Infinite Dimensions. Bayesian Anal. 11(3), 671–695 (2016)
2. Alexanderian, A., Petra, N., Stadler, G., Ghattas, O.: A-optimal design of experiments for infinite-dimensional Bayesian linear inverse problems with regularized \( \ell_0 \)-sparsification. SIAM J. Sci. Comput. 36(5), A2122–A2148 (2014)
3. Alexanderian, A., Petra, N., Stadler, G., Ghattas, O.: A fast and scalable method for A-optimal design of experiments for infinite-dimensional Bayesian nonlinear inverse problems. SIAM J. Sci. Comput. 38(1), A243–A272 (2016)
4. Anthony, C.: Atkinson and Alexander N. Donev. Optimum Experimental Designs, Oxford (1992)
5. Claudio Canuto, M.: Yousuff Hussaini, Alfio Quarteroni, and Thomas A. Zang Jr. Spectral Methods in Fluid Dynamics. Springer-Verlag, New York (1988)
6. Chaloner, K., Verdinelli, I.: Bayesian experimental design: A review. Stat. Sci. 10(3), 273–304 (1995)
7. Darby, C.L., Hager, W.W., Rao, A.V.: An hp-adaptive pseudospectral method for solving optimal control problems. Optimal Control Applications and Methods 32, 476–502 (2011)
8. Fahroo, F., Michael, Ross I.: Direct trajectory optimization by a Chebyshev pseudospectral method. AIAA J. Guid. Control Dyn. 25(1), 160–166 (2002)
9. Gurobi Optimization. Gurobi optimization reference manual. http://www.gurobi.com (2021)
10. Haber, E., Horesh, L., Tenorio, L.: Numerical methods for experimental design of large-scale linear ill-posed inverse problems. Inverse Prob. 24(055012), 125–137 (2008)
11. Harsha, P.T., Keel, L.C., Castrogiovanni, A., Sherrill, R.T.: X-43A vehicle design and manufacture. AIAA (2011)
12. Hart, J., van Bloemen Waanders, B., Hertzog, R.: Hyper-differential sensitivity analysis of uncertain parameters in PDE-constrained optimization. Int. J. Uncertain. Quant. 10(3), 225–248 (2020)
13. Hood, L.G., Bennett, G. Parish, J.J.: Model fidelity studies for rapid trajectory optimization. In: AIAA SciTech Forum (2019)
14. Horesh, L., Haber, E., Tenorio, L.: Optimal Experimental Design for the Large-Scale Nonlinear Ill-Posed Problem of Impedance Imaging, pp. 273–290. Wiley, Hoboken (2010)
15. Karlgaard, C.D., Tartabini, P.V., Blanchard, R.C., Kirsch, M., Toniolo, M.D.: Hyper-X post-flight-trajectory reconstruction. J. Spacecr. Rocket. 43(1), 105–115 (2006)
16. McClinton, C.R.: X-43-Scramjet power breaks the hypersonic barrier: Dryden Lectureship in Research for 2006. In: AIAA Aeroscpace Sciences Meeting and Exhibit (2006)
17. McClinton, C.R., Rausch, D.R.: Hyper-X program status. In: 10th AIAA/NAL-LASDA-ISAS International Space Plans and Hypersonic Systems and Technologies Conference (2001)
18. Morris, M.: Design of Experiments: An Introduction Based on Linear Models. CRC Press, Boca Raton (2010)
19. Patterson, M.A., Rao, A.V.: GPOPS-II a MATLAB software for solving multiple-phase optimal control problems using hp-adaptive gaussian quadrature collocation methods and sparse nonlinear programming. ACM Trans. Math. Softwa., 41(1) 2014
20. Pázman, A.: Foundations of Optimum Experimental Designs. D. Reidel Publishing Co., Dordrecht (1986)
21. Pukelsheim, F.: Optimal Design of Experiments. John Wiley & Sons, New-York (1993)
22. Ross, M., Karpenko, M.: From theory to flight: I. A review of pseudospectral optimal control. Ann. Rev. Control 36, 182–197 (2012)
23. Saibaba, A.K., Hart, J., van Bloemen Waanders, B.: Randomized algorithms for generalized singular value decomposition with application to sensitivity analysis. In: Numerical Linear Algebra with Applications, p. e2364 (2021)
24. Sunseri, I., Hart, J. van Bloemen Waanders, B., Alexanderian, A.: Hyper-differential sensitivity analysis for inverse problems constrained by partial differential equations. Inverse Problems, 36(12) (2020)
25. Uciński, D.: Optimal Measurement Methods for Distributed Parameter System Identification. CRC Press, Boca Raton (2005)
26. Wachter, A., Biegler, L.T.: On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Math. Program. 106, 25–57 (2006)
27. Zlobec, S.: Stable Parametric Programming, chapter Zermelo’s Navigation Problems. Springer (2001)

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.