Feedback Synthesis For Underactuated Systems Using Sequential Second-Order Needle Variations

§SAGE

Giorgos Mamakoukas[1], Malcolm A. MacIver[1,2,3] and Todd D. Murphey[1]

Abstract

This paper derives nonlinear feedback control synthesis for general control affine systems using second-order actions—the second-order needle variations of optimal control—as the basis for choosing each control response to the current state. A second result of the paper is that the method provably exploits the nonlinear controllability of a system by virtue of an explicit dependence of the second-order needle variation on the Lie bracket between vector fields. As a result, each control decision necessarily decreases the objective when the system is nonlinearly controllable using first-order Lie brackets. Simulation results using a differential drive cart, an underactuated kinematic vehicle in three dimensions, and an underactuated dynamic model of an underwater vehicle demonstrate that the method finds control solutions when the first-order analysis is singular. Moreover, the simulated examples demonstrate superior convergence when compared to synthesis based on first-order needle variations. Lastly, the underactuated dynamic underwater vehicle model demonstrates convergence even in the presence of a velocity field.

1 Introduction

With many important applications in aerial or underwater missions, systems are underactuated either by design—in order to reduce actuator weight, expenses, or energy consumption—or as a result of technical failures. In both cases, it is important to develop control policies that can exploit the nonlinearities of the dynamics, are general enough for this broad class of systems, and easily computable. Various approaches to nonlinear control range from steering methods using sinusoid controls (Murray and Sastry 1993), sequential actions of Lie bracket sequences (Murray et al. 1994) and backstepping (Kokotovic 1992; Seto and Baillieul 1994) to perturbation methods (Junkins and Thompson 1986), sliding mode control (SMC) (Perruquetti and Barbot 2002; Utkin 1992; Xu and Özgüner 2008), intelligent control (Brown and Passino 1997; Harris et al. 1993) or hybrid control (Fierro et al. 1999) and nonlinear model predictive control (NMPC) methods (Allgöwer et al. 2004). These schemes have been successful on well-studied examples including, but not limited to, the rolling disk, the kinematic car, wheeling mobile robots, the Snakeboard, surface vessels, quadrotors, and cranes (Bullo et al. 2000; Lin et al. 2014; Escareño et al. 2013; Reyhanoglu et al. 1996; Fang et al. 2003; Toussaint et al. 2000; Bouadi et al. 2007b,a; Chen et al. 2013; Nakazono et al. 2008; Shammas and de Oliveira 2012; Morbidi and Prattichizzo 2007; Roy and Asada 2007; Becker and Bretl 2010; Kolmanovsky and McClamroch 1995; Boskovic et al. 1999).

The aforementioned methods have limitations. In the case of perturbations, the applied controls assume a future of control decisions that do not take the disturbance history into account; backstepping is generally ineffective in the presence of control limits and NMPC methods are typically computationally expensive. SMC methods suffer from chattering, which results in high energy consumption and instability risks by virtue of exciting unmodeled high-frequency dynamics (Khalil 2002), intelligent control methods are subject to data uncertainties (El-Nagar et al. 2014), while other methods are often case-specific and will not hold for the level of generality encountered in robotics. We address these limitations by using needle variations to compute real-time feedback laws for general nonlinear systems affine in control, discussed next.

1.1 Needle Variations Advantages to Optimal Control

In this paper, we investigate using needle variation methods to find optimal controls for nonlinear controllable systems. Needle variations consider the sensitivity of the cost function to infinitesimal application of controls and synthesize actions that reduce the objective (Aseev and Veliov 2014; Shaikh and Caines 2007). Such control synthesis methods have the advantage of efficiency in terms of computational effort, making them appropriate for online feedback—similar to other model predictive control methods, such as iLQG

[1]Department of Mechanical Engineering, Northwestern University, Evanston, IL, USA
[2]Department of Biomedical Engineering, Northwestern University, Evanston, IL, USA
[3]Department of Neurobiology, Northwestern University, Evanston, IL, USA

Corresponding author:
Giorgos Mamakoukas, Department of Mechanical Engineering, Northwestern University, 2145 Sheridan Road, Evanston, IL 60208, USA.
Email: giorgosmamakoukas@u.northwestern.edu

(Todorov and Li 2005), but with the advantage, as shown here, of having provable formal properties over the entire state space. For time evolving objectives, as in the case of trajectory tracking tasks, controls calculated from other methods (such as sinusoids or Lie brackets for nonholonomic integrators) may be rendered ineffective as the target continuously moves to different states. In such cases, needle variation controls have the advantage of computing actions that directly reduce the cost, without depending on future control decisions. However, needle variation methods, to the best of our knowledge, have not yet considered higher than first-order sensitivities of the cost function.

We demonstrate analytically in Section III that, by considering second-order needle variations, we obtain variations that explicitly depend on the Lie brackets between vector fields and, as a consequence, the higher-order nonlinearities in the system. Later, in Section III, we show that, for classically studied systems, such as the differential drive cart, this amounts to being able to guarantee that the control approach is *globally* certain to provide descent at every state, despite the conditions of Brockett's theorem (Brockett 1983) on nonexistence of smooth feedback laws for such systems. We extend this result by proving that second-order needle variations controls necessarily decrease the objective for the entire class of systems that are controllable with first-order Lie brackets. As a consequence, provided that the objective is convex, second-order needle variation controls provably guarantee that the agent reaches the target in a collision-free manner in the presence of obstacles without relying on predefined trajectories.

1.2   Paper Contribution and Structure

This paper derives the second-order sensitivity of the cost function with respect to infinitesimal duration of inserted control, which we will refer to interchangeably as the second-order mode insertion gradient or mode insertion Hessian (MIH). We relate the MIH expression to controllability analysis by revealing its underlying Lie bracket structure and present a method of using second-order needle variation actions to expand the set of states for which individual actions that guarantee descent of an objective function can be computed. Finally, we compute an analytical solution of controls that uses the first two orders of needle variations.

This paper expands the work presented in Mamakoukas et al. (2017) by including the derivations of the MIH, the proofs that guarantee descent, and extensive simulation results that include comparisons to alternative feedback algorithms. Further, we extend the results to account for obstacles and prove the algorithm finds collision-free solutions for the controllable systems considered, including simulations of obstacle-avoidance for static and moving obstacles.

The content is structured as follows. In Section II, we provide relevant research background in the field of motion planning for controllable systems. In Section III, we prove that second-order needle variations guarantee control solutions for systems that are nonlinearly controllable using first-order Lie brackets. We use this result to provably generate collision-free trajectories that safely reach the target among obstacles provided convex objectives. In Section IV,

we present an analytical control synthesis method that uses second-order needle actions. In Section V, we implement the proposed synthesis method and present simulation results on a controllable, underactuated model of a 2D differential drive vehicle, a 3D controllable, underactuated kinematic rigid body and a 3D underactuated dynamic model of an underwater vehicle.

2   Existing methods for controllable systems

In this section, we present a review of some popular methods that are available for underactuated, controllable systems, followed by a discussion of techniques for collision-avoidance. An introduction to these methods, as well as additional algorithms for controllable systems, can be found in La Valle (2011).

2.1   Optimization Algorithms for Nonholonomic Controllable Systems

Nonholonomic systems are underactuated agents subject to nonintegrable differential constraints. Examples include wheeled agents that are not allowed to skid (e.g., unicycle, differential drive, tricycle). Nonholonomic systems are of interest to the control community because one needs to obtain solutions for motion planning tasks (Kolmanovsky and McClamroch 1995).

The concept of controllability is indispensable in the study of nonholonomic systems. Controllability analytically answers the existence of control solutions that move a certain agent between arbitrary states in finite time, and, in doing so, it reveals all possible effects of combined control inputs of underactuated systems that are subject to velocity, but not displacement, constraints.

A popular approach in controlling nonholonomic systems is piecewise constant motion planning (Sussmann 1991; Lafferriere and Sussmann 1993). Lafferriere and Sussmann (Lafferriere and Sussmann 1991, 1993) extend the original dynamics with *fictitious* action variables in the direction of the nested Lie brackets to determine a control for the extended system. They first compute the time the system must flow along each vector field, in a sequential manner, to accomplish a given motion of the extended system. Then, using the Campbell-Baker-Hausdorff-Dynkin (CBHD) formula (Strichartz 1987; Rossmann 2002; La Valle 2011), they recover the solution in terms of the original inputs of the system.

On the other hand, piecewise constant motion planning is model-specific, since the process of recovering the original inputs in terms of the fictitious actions changes for different number of inputs. In addition, solutions involve a sequence of individual actions that generate the Lie bracket motion. The actuation sequence grows increasingly larger for higher order brackets. Compensating for the third-order error in the CBHD formula involves two second-order Lie brackets and twenty successive individual inputs, each of infinitesimal duration (McMickell and Goodwine 2007). The sequence is described in detail by Lafferriere and Sussmann (1993). In practice, such actuation becomes challenging as the number of switches grows. The theoretically infinitesimal duration

of each input may be hard to reproduce in hardware, while, in the face of uncertainty and time-evolving trajectories, actuation consisting of a large sequence of controls (e.g., of twenty actions) is likely to change once feedback is received.

Another popular approach is steering using sinusoids (Brockett 1982; Murray and Sastry 1993; Murray et al. 1994; Sastry 2013; Teel et al. 1995; Laumond et al. 1998). This method applies sinusoidal control inputs of integrally related frequencies. States are sequentially brought into the desired configuration in stages, while the rest of the states remain invariant over a single cycle. This approach has been validated in generating motion of an underactuated robot fish (Morgansen et al. 2001).

Steering using sinusoids suffers from the complicated sequence of actions that grows as a function of the inputs involved. Moreover, besides also being model-specific, the method addresses each state separately, meaning each state gets controlled by its own periodic motion, requiring N periods for an N-dimensional system, leading to slow convergence. Further, solutions focus on the final states (at the end of each cycle) and not their time evolution, hence they may temporarily increase the running cost (consider the car example of Fig. 7 in Murray and Sastry (1993)). As with the method of piecewise constant motion planning, when tracking a moving target, these factors also compromise the performance of this approach.

Other trajectory generation techniques for controllable systems involve differential flatness (Lamiraux and Laumond 2000; Rathinam and Murray 1998; Ross and Fahroo 2004; Fliess et al. 1995; Rouchon et al. 1993) and kinematic reduction (Bullo and Lynch 2001; Lynch et al. 2000; Murphey and Burdick 2006). Control based on differential flatness uses outputs and their derivatives to determine control laws. However, as discussed in Choudhury and Lynch (2004), there is not an automatic procedure to discover whether flat outputs exist. Further, differential flatness does not apply to all controllable systems and motion planning is further complicated when control limits or obstacles are present (Bullo and Lynch 2001).

2.2　*Motion Planning for Controllable Systems in the Presence of Obstacles*

Controllability in its classical sense concerns itself with the existence of an action trajectory that can move the agent to a desired state, subject to the differential constraints posed by the dynamics, in the absence of obstacles. Controllability is an inherent property of the dynamics and reveals all allowable motion, disregarding the presence of physical constraints in the environment. This is true for the methods discussed in Section 2.1.

Feasible path planning amidst obstacles is often treated separately from the optimal control problem. Most commonly, feasible trajectories are generated with efficient path planners, such as rapidly-exploring random tree (RRT) and probabilistic road map (PRM) methods (LaValle and Kuffner Jr 2001; Hsu et al. 2002). The distinction between path planning and optimal control can be seen in work by Choudhury and Lynch (2004); Lynch et al. (2000) that uses such motion planners to generate trajectories among obstacles and then uses them as a reference to compute the optimal control. In this setting, nonholonomic motion consists of two stages, the path planning and the feedback synthesis that tracks the feasible trajectory.

Another solution to obstacle avoidance in motion planning is the use of barrier certificates (Prajna et al. 2007). Barrier certificates provably enforce collision-free behavior by minimally perturbing, in a least-squares sense, the control response in order to satisfy safety constraints. Feedback synthesis proceeds without accounting for obstacles and solutions are modified, only when necessary, via a quadratic program (QP) subject to constraints that ensure collision avoidance (Borrmann et al. 2015; Xu et al. 2015; Wang et al. 2017; Ames et al. 2014; Wu and Sreenath 2016).

Additional solutions to obstacle avoidance include compensating functions that eliminate local minima in the objective caused by the obstacles (Deng et al. 2008), as well as designing navigation functions using inverse Lyapunov expressions (Tanner et al. 2001). The former method computes the local minima in the objective and constructs a plane surface function to remove them and make the objective convex. This process can be cumbersome, as one would have to locate all local minima in the objective induced by the obstacles and then calculate the compensating function. On the other hand, navigation functions, described in Rimon and Koditschek (1992); Tanner et al. (2001), are globally convergent potential functions and are system-specific.

Several of these collision-avoidance algorithms are not system-specific and could be implemented with our controller, later outlined in Section IV. In simulation results, presented in Section V, we show collision-avoidance using only penalty functions in the objective, demonstrating that the proposed controller succeeds in tasks (collision avoidance) that traditionally require sophisticated treatment.

3　Needle Variation Controls based on Non-Linear Controllability

In this section, we relate the controllability of systems to first- and second-order needle variation actions. After presenting the MIH expression, we relate the MIH to the Lie bracket terms between vector fields. Using this connection, we tie the descent property of needle variation actions to the controllability of a system and prove that second-order needle variation controls can produce control solutions for a wider set of the configuration state space than first-order needle variation methods. As a result, we are able to constructively compute, via an analytic solution, control formulas that are guaranteed to provide descent, provided that the system is controllable with first-order Lie brackets. Generalization to higher-order Lie brackets appears to have the same structure, but that analysis is postponed to future work.

3.1　*Second-Order Mode Insertion Gradient*

Needle variation methods in optimal control have served as the basic tool in proving the Pontryagin's Maximum Principle (Pontryagin et al. 1962; Dmitruk and Osmolovskii 2014; Garavello and Piccoli 2005). Using piecewise dynamics, they introduce infinitesimal perturbations in control that change the default trajectory and objective (see
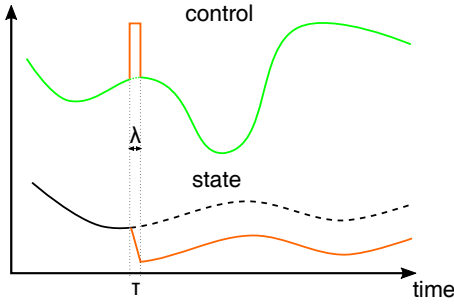
Figure 1. A fixed-value perturbation in the nominal control, introduced at time τ and with duration λ, and the associated variation in the state. In the limit $\lambda \to 0$, the control perturbation becomes a needle variation.

Fig. 1). Such dynamics are typically used in optimal control of hybrid systems to optimize the schedule of a-priori known modes (Egerstedt et al. 2006; Caldwell and Murphey 2016).

Here, instead, we consider dynamics of a single switch to obtain a new control mode u at every time step that will optimally perturb the trajectory(Ansari and Murphey 2016). The feedback algorithm presented in Ansari and Murphey (2016), however, only considers the first-order sensitivity of the cost function to a needle action and, as a result, often fails to provide solutions for controllable underactuated systems. By augmenting the algorithm with higher order information (via the MIH), we are able to provide solutions in cases when the first-order needle variation algorithm in Ansari and Murphey (2016) is singular.

Consider a system with state $x : \mathbb{R} \mapsto \mathbb{R}^N$ and control $u : \mathbb{R} \mapsto \mathbb{R}^{M \times 1}$ with control-affine dynamics of the form

$$f(t, x(t), u(t)) = g(t, x(t)) + h(t, x(t))u(t), \qquad (1)$$

where $g(t, x(t))$ is the drift vector field. Consider a time period $[t_o, t_f]$ and control modes described by

$$\dot{x}(t) = \begin{cases} f_1(x(t), v(t)), & t_o \le t < \tau - \frac{\lambda}{2} \\ f_2(x(t), u(\tau)), & \tau - \frac{\lambda}{2} \le t < \tau + \frac{\lambda}{2} \\ f_1(x(t), v(t)), & \tau + \frac{\lambda}{2} \le t \le t_f, \end{cases} \qquad (2)$$

where f_1 and f_2 are the dynamics associated with *default* and *inserted* control v and u, respectively. Parameters λ and τ are the duration of the inserted dynamics f_2 and the switching time between the two modes.

Note that the default control $v(t)$ is the input for the nominal trajectory—$v(t)$ could itself be the result of a different controller—which is then improved by the insertion of a new control vector $u(t)$ creating a switched mode f_2. In addition, while the default control $v(t)$ of the switched mode sequence in (2) may be time-dependent, the dynamics f_2 have control $u(\tau)$ that has a fixed value over $[\tau - \frac{\lambda}{2}, \tau + \frac{\lambda}{2}]$.

Given a cost function J of the form

$$J(x(t)) = \int_{t_o}^{t_f} l_1(x(t))\mathrm{d}t + m(x(t_f)), \qquad (3)$$

where $l_1(x(t))$ is the running cost and $m(x(t))$ the terminal cost, the mode insertion gradient (MIG), derived in Egerstedt et al. (2006), is

$$\frac{dJ}{d\lambda_+} = \rho^T (f_2 - f_1), \qquad (4)$$

where $\rho : \mathbb{R} \mapsto \mathbb{R}^{N \times 1}$ is the first-order adjoint state, which is calculated from the default trajectory and given by

$$\dot{\rho} = -D_x l_1{}^T - D_x f_1{}^T \rho \qquad (5)$$

$$\text{subject to: } \rho(t_f) = D_x m(x(t_f))^T.$$

We use the subscript λ_+ to indicate that a certain variable is considered after evaluating the limit $\lambda \to 0$. For brevity, the dependencies of variables are dropped. While the objective for needle variation controls has typically not included a control term, doing so is straightforward and yields similar performance. Work in Ansari and Murphey (2016) has considered objectives with control terms, and one can recompute the mode insertion gradient and mode insertion Hessian assuming the objective depends on u without impacting any of the rest of the approach.

The derivation of the mode insertion Hessian is similar to Caldwell and Murphey (2011) and is presented in the Appendix. For dynamics that do not depend on the control duration, the mode insertion Hessian (MIH)[1] is given by

$$\frac{d^2 J}{d\lambda_+^2} = (f_2 - f_1)^T \Omega (f_2 - f_1) + \rho^T (D_x f_2 \cdot f_2 + D_x f_1 \cdot f_1$$
$$- 2D_x f_1 \cdot f_2) - D_x l_1 \cdot (f_2 - f_1), \qquad (6)$$

where $\Omega : \mathbb{R} \mapsto \mathbb{R}^{N \times N}$ is the second-order adjoint state, which is calculated from the default trajectory and is given by

$$\dot{\Omega} = -D_x f_1{}^T \Omega - \Omega D_x f_1 - D_x^2 l_1 - \sum_{i=1}^{N} \rho_i D_x^2 f_1^i \qquad (7)$$

$$\text{subject to: } \Omega(t_f) = D_x^2 m(x(t_f))^T.$$

The superscript i in the dynamics f_1 refers to the ith element of the vector.

3.2 Dependence of Second Order Needle Variations on Lie Bracket Structure

The Lie bracket of two vectors $f(x)$, and $g(x)$ is

$$[f, g](x) = \frac{\partial g}{\partial x} f(x) - \frac{\partial f}{\partial x} g(x),$$

which generates a control vector that points in the direction of the net infinitesimal change in state x created by infinitesimal noncommutative flow $\phi_\epsilon^f \circ \phi_\epsilon^g \circ \phi_\epsilon^{-f} \circ \phi_\epsilon^{-g} \circ x_0$, where ϕ_ϵ^f is the flow along a vector field f for time ϵ (Murray et al. 1994; Jakubczyk 2001). Lie brackets are most commonly used for their connection to controllability (Rashevsky 1938; Chow 1940/1941), but here they will show up in the expression describing the second-order needle variation.

We relate second-order needle variation actions to Lie brackets in order to connect the existence of descent-providing controls to the nonlinear controllability of a system. Let $h_i : \mathbb{R} \mapsto \mathbb{R}^{N \times 1}$ denote the column control vectors that make up $h : \mathbb{R} \mapsto \mathbb{R}^{N \times M}$ in (1) and $u_i \in \mathbb{R}$ be the individual control inputs. Then, we can express dynamics as

$$f = g + \sum_i^M h_i u_i.$$

and, for default control $v = 0$, we can re-write the MIH as

$$\frac{d^2 J}{d\lambda_+^2} = \Big(\sum_{i=1}^{M} h_i u_i\Big)^T \Omega \sum_{j=1}^{M} h_j u_j + \rho^T \Big(\sum_{i=1}^{M} (D_x h_i u_i) \cdot g$$

$$- D_x g \cdot (h_i u_i) + \sum_{i=1}^{M} D_x h_i u_i \sum_{i=1}^{M} h_i u_i \Big) - D_x l_1 \sum_{i=1}^{M} h_i u_i.$$

Splitting the sum expression into diagonal ($i = j$) and off-diagonal ($i \neq j$) elements, and by adding and subtracting $2 \sum_i^M \sum_{j=1}^{i-1} (D_x h_i u_i)(h_j u_j)$, we can write

$$\sum_{i=1}^{M} D_x h_i u_i \sum_{i=1}^{M} h_i u_i = \sum_i^M \sum_j^{i-1} [h_i, h_j] u_i u_j$$

$$+ 2 \sum_i^M \sum_j^{i-1} (D_x h_i u_i)(h_j u_j)$$

$$+ \sum_{i=j=1}^{M} (D_x h_i u_i)(h_i u_i).$$

Then, we can express the MIH as

$$\frac{d^2 J}{d\lambda_+^2} = \sum_{i=1}^{M} \sum_{j=1}^{M} u_i u_j h_i^T \Omega h_j + \rho^T \Big(\sum_{i=2}^{M} \sum_{j=1}^{i-1} [h_i, h_j] u_i u_j$$

$$+ 2 \sum_{i=2}^{M} \sum_{j=1}^{i-1} (D_x h_i) h_j u_i u_j + \sum_{i=1}^{M} (D_x h_i) h_i u_i u_i$$

$$+ \sum_{i=1}^{M} [g, h_i] u_i \Big) - D_x l \Big(\sum_{i=1}^{M} h_i u_i \Big).$$

The expression contains Lie bracket terms of the control vectors that appear in the system dynamics, indicating that second-order needle variations incorporate higher-order nonlinearities. By associating the MIH to Lie brackets, we next prove that second-order needle variation actions can guarantee decrease of the objective for systems that are controllable with first-order Lie brackets.

3.3 Existence of Control Solutions with First- and Second-Order Mode Insertion Gradients

In this section, we prove that the first two orders of the mode insertion gradient can be used to guarantee controls that reduce objectives of the form (3) for systems that are controllable with first-order Lie brackets. The analysis is applicable to optimization problems that satisfy the following assumptions.

Assumption 1. *The vector elements of dynamics f_1 and f_2 are real, bounded, $\mathcal{C}^2$ in x, and $\mathcal{C}^0$ in u and t.*

Assumption 2. *The incremental cost $l_1(x)$ is real, bounded, and $\mathcal{C}^2$ in x. The terminal cost $m(x(t_f))$ is real and twice differentiable with respect to $x(t_f)$.*

Assumption 3. *Default and inserted controls v and u are real, bounded, and $\mathcal{C}^0$ in t.*

Under Assumptions 1-3, the MIG and MIH expressions exist and are unique. Then, as we show next, there are control actions that can improve any objective as long as there exists $t \in [t_o, t_f]$ for which $x(t) \neq x^*(t)$.

Definition 1. *A trajectory x^* described by a pair (x^*, u^*) is the global minimizer of the objective function $J(x^*(t))$ for which $J(x^*(t)) \leq J(x(t)) \, \forall \, x(t)$.*

Given Definition 1, a trajectory x^* described by a pair (x^*, u^*) is the global minimizer of the cost function in the unconstrained sense (not subject to the dynamics of the system) and satisfies $D_x J(x^*(t)) = 0$ throughout the time horizon considered.

Assumption 4. *The pair (x^*, u^*) describes the only trajectory x^* for which the unconstrained derivative of the objective is equal to zero (i.e., $D_x J(x^*(t)) = 0 \, \forall \, t \in [t_o, t_f]$).*

Assumption 4 is necessary to prove that the first-order adjoint is non-zero, which is a requirement for the controllability results shown in this work. It assumes that the objective function in the unconstrained sense does not have a maximizer or saddle point and has only one minimizer x^* described by (x^*, u^*) that indicates the target trajectory or location. It is an assumption that, among other choices, can be easily satisfied with a quadratic cost function that even includes penalty functions associated with physical obstacles.

Proposition 1. *Consider a pair (x, v) that describes the state and default control of (2). If $(x, v) \neq (x^*, v^*)$, then the first-order adjoint ρ is a non-zero vector.*

Proof. Using (3), and by Assumption 4,

$$x \neq x^* \Rightarrow D_x J(x(t)) \neq 0$$

$$\Rightarrow \int_{t_o}^{t_f} D_x l_1(x(t)) \mathrm{d}t + D_x m(x(t_f)) \neq 0$$

$$\Rightarrow \int_{t_o}^{t_f} D_x l_1(x(t)) \mathrm{d}t \neq 0 \text{ OR } D_x m(x(t_f)) \neq 0$$

$$\Rightarrow D_x l_1(x(t)) \neq 0 \text{ OR } D_x m(x(t_f)) \neq 0$$

$$\Rightarrow \dot{\rho} \neq 0 \text{ OR } \rho(t_f) \neq 0.$$

Therefore, if $x \neq x^*$, then $\exists \, t \in [t_o, t_f]$ such that $\rho \neq 0$.

Proposition 2. *Consider dynamics given by (2) and a trajectory described by state and control (x, v). Then, there are always control solutions $u \in \mathbb{R}^M$ such that $\frac{dJ}{d\lambda_+} \leq 0$ for some $t \in [t_o, t_f]$.*

Proof. Using dynamics of the form in (1), the expression of the mode insertion gradient can be written as

$$\frac{dJ}{d\lambda_+} = \rho^T(f_2 - f_1) = \rho^T\big(h(u - v)\big).$$

Given controls u and v that generate a positive mode insertion gradient, there always exist control u' such that the mode insertion gradient is negative, i.e. $u' - v = -(u - v)$. The mode insertion gradient is zero for all $u \in \mathbb{R}^M$ if the costate vector is orthogonal to each control vector h_i or if the costate vector is zero everywhere.[2]

Proposition 3. *Consider dynamics given by (2) and a pair of state and control $(x, v) \neq (x^*, v^*)$ for which $\frac{dJ}{d\lambda_+} = 0 \, \forall \, u \in \mathbb{R}^M$ and $\forall \, t \in [t_o, t_f]$. Then, the first-order adjoint ρ is orthogonal (under the Euclidean inner product) to all control vectors h_i.*

Proof. We rewrite (4) as

$$\frac{dJ}{d\lambda_+} = 0 \Rightarrow \rho^T \sum_i^M h_i(u_i - v_i) = 0$$

$$\Rightarrow \sum_i^M k_i w_i = 0 \ \forall \ w_i,$$

where $w_i = (u_i - v_i)$ and $k_i = \rho^T h_i \in \mathbb{R}$. The linear combination of the elements of k is zero for any w_i, which means k must be the zero vector. By Proposition 1, $\rho \neq 0$ for a trajectory described by a pair of state and control $(x, u) \neq (x^*, u^*)$ and, as a result, $\rho^T h_i = 0 \ \forall \ i \in [1, M]$.

Proposition 4. *Consider dynamics given by* (2) *and a pair of state and control* $(x, v) \neq (x^*, v^*)$ *for which* $\frac{dJ}{d\lambda_+} = 0 \ \forall \ u \in \mathbb{R}^M$ *and* $\forall \ t \in [t_o, t_f]$. *Further assume that the control vectors* h_i *and the Lie Bracket terms* $[h_i, h_j]$ *and* $[g, h_i]$—*where* $i, j \in [1, M]$—*span the state space* $\mathbb{R}^N$. *Then, there exist* i *and* j *such that either* $\rho^T[h_i, h_j] \neq 0$ *or* $\rho^T[g, h_i] \neq 0$.

Proof. Let $S = \{h_i, [h_i, h_j], [g, h_i]\} \ \forall \ i, j \in [1, M]$ be a set of vectors that span the state space $\mathbb{R}^N$ ($\text{span}\{S\} = \mathbb{R}^N$). Then, any vector in $\mathbb{R}^N$ can be written as a linear combination of the vectors in S. The first-order adjoint is an N-dimensional vector, which is non-zero for a trajectory described by a pair of state and control $(x, u) \neq (x^*, u^*)$ by Proposition 1. Therefore, it can be expressed as

$$\rho = c_1 h_1 + \cdots + c_M h_M + \sum_{i=2}^M \sum_{j=1}^{i-1} c'_{i,j}[h_i, h_j] + \sum_{i=1}^M c''_i[g, h_i], \tag{8}$$

where $c_i, c'_i, c''_i \in \mathbb{R}$. Left-multiplying (8) by ρ^T yields

$$\rho^T \rho = \sum_{i=1}^M c_i \rho^T h_i + \sum_{i=2}^M \sum_{j=1}^{i-1} c'_{i,j} \rho^T[h_i, h_j] + \sum_{i=1}^M c''_i \rho^T[g, h_i].$$

Given that $\frac{dJ}{d\lambda_+} = 0$, and by Proposition 3, ρ is orthogonal to all control vectors h_i (which also implies that the control vectors h_i do not span $\mathbb{R}^N$), the above equation simplifies to

$$\rho^T \rho = \sum_{i=2}^M \sum_{j=1}^{i-1} c'_{i,j} \rho^T[h_i, h_j] + \sum_{i=1}^M c''_i \rho^T[g, h_i].$$

Because $\rho^T \rho \neq 0$, there exists $i, j \in [1, M]$ and a Lie bracket term $[h_i, h_j]$, or $[g, h_i]$ that is not orthogonal to the costate ρ. That is,

$$\exists \ i, j \in [1, M] \text{ such that } \rho^T[h_i, h_j] \neq 0 \text{ OR } \rho^T[g, h_j].$$

First-order needle variation methods are singular when the mode insertion gradient is zero. When that is true, the next result—that is the main piece required for the main theoretical result of this section in Theorem 1—demonstrates that the second-order mode insertion gradient is guaranteed to be negative for systems that are controllable with first-order Lie Brackets, which in turn implies that a control solution can be found with second-order needle variation methods.

Proposition 5. *Consider dynamics given by* (2) *and a trajectory described by state and control* $(x, v) \neq (x^*, v^*)$ *for which* $\frac{dJ}{d\lambda_+} = 0$ *for all* $u \in \mathbb{R}^M$ *and* $t \in [t_o, t_f]$. *If the control vectors* h_i *and the Lie brackets* $[h_i, h_j]$ *and* $[g, h_i]$ *span the state space* ($\mathbb{R}^N$), *then there always exist control solutions* $u \in \mathbb{R}^M$ *such that* $\frac{d^2J}{d\lambda_+^2} < 0$.

Proof. See Appendix.

Theorem 1. *Consider dynamics given by* (2) *and a trajectory described by state and control* $(x, v) \neq (x^*, v^*)$. *If the control vectors* h_i *and the Lie brackets* $[h_i, h_j]$ *and* $[g, h_i]$ *span the state space* ($\mathbb{R}^N$), *then there always exists a control vector* $u \in \mathbb{R}^M$ *and a duration* λ *such that the cost function* (3) *can be reduced.*

Proof. The local change of the cost function (3) due to inserted control u of duration λ can be approximated with a Taylor series expansion

$$J(\lambda) - J(0) \approx \lambda \frac{dJ}{d\lambda_+} + \frac{\lambda^2}{2} \frac{d^2J}{d\lambda_+^2}.$$

By Propositions 2 and 5, either 1) $\frac{dJ}{d\lambda_+} < 0$ or 2) $\frac{dJ}{d\lambda_+} = 0$ and $\frac{d^2J}{d\lambda_+^2} < 0$. Therefore, there always exist controls that reduce the cost function (3) to first or second order.

4 Control Synthesis

In this section, we present an analytical solution of first- and second-order needle variation controls that reduce the cost function (3) to second order. We then describe the algorithmic steps of the feedback scheme used in the simulation results in Section V.

4.1 Analytical Solution for Second Order Actions

For underactuated systems, there are states at which ρ is orthogonal to the control vectors h_i (see Proposition 3). At these states, control calculations based only on first-order sensitivities fail, while controls based on second-order information still have the potential to decrease the objective provided that the control vectors and their Lie brackets span the state space (see Theorem 1). We use this property to compute an analytical synthesis method that expands the set of states for which individual actions that guarantee descent of an objective function can be computed.

Consider the Taylor series expansion of the cost around control duration λ. Given the expressions of the first- and second-order mode insertion gradients, we can write the cost function (3) as a Taylor series expansion around the infinitesimal duration λ of inserted control u:

$$J(\lambda) \approx J(0) + \lambda \frac{dJ}{d\lambda_+} + \frac{\lambda^2}{2} \frac{d^2J}{d\lambda_+^2}. \tag{9}$$

The first- and second-order mode insertion gradients used in the expression are functions of the inserted control $u(t)$ in (2). Equation (9) is quadratic in $u(t)$ and, for a fixed λ, has a unique solution which is used to update the control actions. Controls that minimize the Taylor expansion of the cost will

have the form

$$u^*(t) = \underset{u}{\operatorname{argmin}}\ J(0) + \lambda\frac{dJ}{d\lambda_+} + \frac{\lambda^2}{2}\frac{d^2J}{d\lambda_+^2} + \frac{1}{2}\|u\|_R^2, \quad (10)$$

where the MIH has both linear and quadratic terms in $u(t)$. We compute the minimizer of (10) to be

$$u^*(t) = [\frac{\lambda^2}{2}\Gamma + R]^{-1}[\frac{\lambda^2}{2}\Delta + \lambda(-h^T\rho)], \quad (11)$$

where $\Delta : \mathbb{R} \mapsto \mathbb{R}^{M\times 1}$ and $\Gamma : \mathbb{R} \mapsto \mathbb{R}^{M\times M}$ are respectively the first- and second-order derivatives of $d^2J/d\lambda_+^2$ with respect to the control u (see Appendix). These quantities are given by

$$\Delta \triangleq \Big[\big[h^T(\Omega^T+\Omega)h + 2h^T\cdot(\sum_{k=1}^{n}(D_xh_k)\rho_k)^T\big]v$$
$$+ (D_xg\cdot h)^T\rho - (\sum_{k=1}^{n}(D_xh_k)\rho_k)\cdot g + h^T D_x l^T\Big]$$

$$\Gamma \triangleq [h^T(\Omega^T+\Omega)h + h^T\cdot(\sum_{k=1}^{n}(D_xh_k)\rho_k)^T + \sum_{k=1}^{n}(D_xh_k)\rho_k\cdot h].$$

The parameter R, a positive definite matrix, denotes a metric on control effort.

The existence of control solutions in (11) depend on the inversion of the Hessian $H = \frac{\lambda^2}{2}\Gamma + R$. To practically ensure H is positive definite, we implement a spectral decomposition on the Hessian $H = VDV^{-1}$, where matrices V and D contain the eigenvectors and eigenvalues of H, respectively. We replace all elements of the diagonal matrix D that are smaller than ϵ with ϵ to obtain $\bar{D}$ and replace H with $\bar{H} = V\bar{D}V^{-1}$ in (11). We prefer the spectral decomposition approach to the Levenberg-Marquardt method ($\bar{H} = H+\kappa I \succ 0$), because the latter affects all eigenvalues of the Hessian and further distorts the second-order information. At saddle points, we set the control equal to the eigenvector of H that corresponds to the most negative eigenvalue in order to descend along the direction of most negative curvature (Murray 2010; Schnabel and Eskow 1990; Boyd and Vandenberghe 2004; Nocedal and Wright 2006).

Synthesis based on (11) provides controls at time t that guarantee to reduce the cost function (3) for systems that are controllable using first-order Lie brackets. Control solutions are computed by forward simulating the state over a time horizon T and backward simulating the first- and second-order costates ρ and Ω. As we see next, this leads to a very natural, and easily implementable, algorithm for applying cost-based feedback while avoiding iterative trajectory optimization.

4.2 Algorithmic Description of Control Synthesis Method

The second-order controls in (11) are implemented in a series of steps shown in Algorithm 1 and visualized in Fig. 2.

We compare first- and second-order needle variation actions by implementing different controls in Step 2 of Algorithm 1. For the first-order case, we implement controls

Algorithm 1

1. Simulate states and costates with default dynamics f_1 over a time horizon T

2. Compute optimal needle variation controls

3. Saturate controls

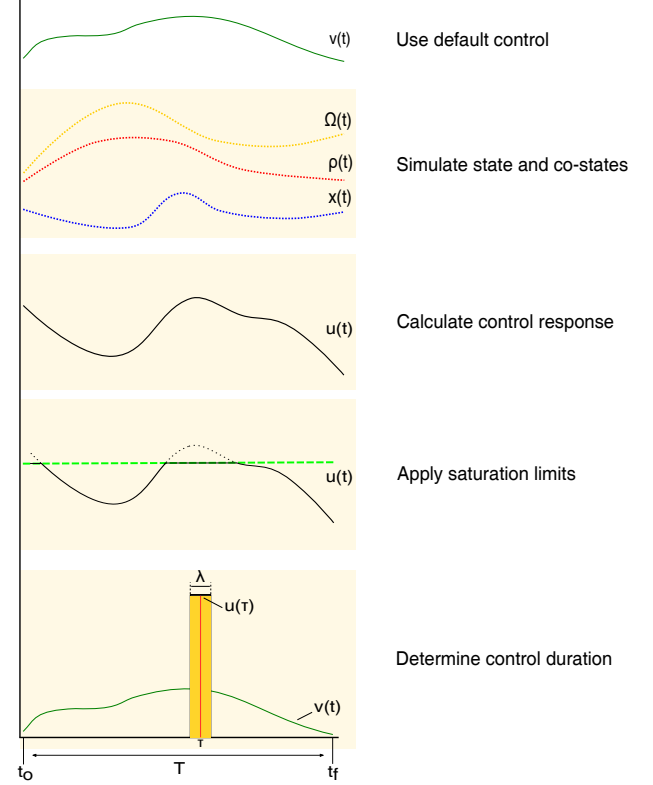4. Use a line search to find control duration that ensures reduction of the cost function (3)[3]

Figure 2. The steps of the controller outlined by Algorithm 1. Using the default control, the states and co-states are forward-simulated for the time horizon $[t_o, t_o+T]$. The optimal control response is computed from (11), and saturated appropriately. At the end, the algorithm determines the finite duration of the inserted single action, evaluated at an application time τ, with a line search.

that are the solution to a minimization problem of the first-order sensitivity of the cost function (3) and the control effort

$$u^*(t) = \min_{u}\ \frac{1}{2}\Big(\frac{dJ_1}{d\lambda_i^+} - \alpha_d\Big)^2 + \frac{1}{2}\|u\|_R^2$$
$$= (\Lambda + R^T)^{-1}(\Lambda v + h^T\rho\alpha_d), \quad (12)$$

where $\Lambda \triangleq h^T\rho\rho^T h$ and $\alpha_d \in \mathbb{R}^-$ expresses the desired value of the mode insertion gradient term (see, for example, Mamakoukas et al. (2016)). Typically, $\alpha_d = \gamma J_o$, where J_o is the cost function (3) computed using default dynamics f_1. For second-order needle variation actions, we compute controls using (11). As Fig. 2 indicates, the applied actuation is the saturated value of the control response of either (11) or (12), evaluated at the application time τ.

While we do not show it in this paper, it is shown in Ansari and Murphey (2016) that the first-order needle

variation control solutions (12) remain a descent direction after saturation. This result is extended in Mamakoukas et al. (2018) to show that the entire control signal, and not a needle action, over the time horizon remains a descent direction when saturated by an arbitrary amount. While we have not yet formally proved a similar property for the second-order needle variation controls (11), one can test and identify if the saturated controls would decrease the cost function before applying any actuation. Further, because the results of this work rely on the sign and not the magnitude of the control solutions, suggesting that the saturated second-order solutions in (11) also provide a descent direction.

Further, needle variation actuation as shown in Fig. 2 may be practically infeasible or at least problematic for motors due to the abrupt changes in the control. There are two remedies to this issue. First, introducing additional filter states associated with the control can constraint the changes in actuation (Fan and Murphey 2016). Second, one can show that the entire curve of the first-order needle variation solution is a descent direction (Mamakoukas et al. 2018). Assuming the same is true for the second-order solutions as well, one could either apply part of the continuous control solution around the time of application τ or filter the discontinuous actuation in hardware and still provide descent with more motor-friendly actuation.

4.3 Convergence in the presence of obstacles

We use Theorem 1 to show that the proposed needle-variation controller will always converge to the global minimizer for convex functions. This statement is true independent of the presence of obstacles.

Theorem 2. *Consider dynamics given by (2), a trajectory described by state and control $(x, v) \neq (x^*, v^*)$. Let $\tilde{x}^k$ describe the trajectory generated after k iterations of Algorithm 1. Further let $x \in \mathcal{X}_{free} \forall\ t \in [t_o, t_f]$, where $\mathcal{X}_{free} \subset \mathcal{X}$ denotes the collision-free part of the state-space. Consider an objective (3) that satisfies Assumption 2 and whose running cost term penalizes collisions, such that $J(\tilde{x}^k) > J(x)$ if $\exists\ t \in [t_o, t_f]$ where $\tilde{x} \notin \mathcal{X}_{free}$. If the objective $J(x)$ is convex with respect to the state x in the unconstrained sense and the control vectors h_i and the Lie brackets $[h_i, h_j]$ and $[g, h_i]$ span the state space $(\mathbb{R}^N)$, then the sequence of solutions $\{\tilde{x}^k\}$ generated by Algorithm 1 converges to x^*. Further, $\tilde{x}^k \in \mathcal{X}_{free} \forall\ k \in \mathbb{R}^+$.*

Proof. Algorithm 1 constructs control responses out of the first- and second-order mode insertion gradients. By extension of Theorem 1, it can guarantee to reduce the objective with each iteration (for some control u of duration λ). Therefore,

$$J(\tilde{x}^k) > J(\tilde{x}^{k+1}) \geq J_{min}, \qquad (13)$$

where $J_{min} = J(x^*)$ is the (only) minimizer of the convex objective. It follows that,

$$\lim_{k \to \infty} J(\tilde{x}^k) = J_{min}. \qquad (14)$$

Further, assume that there exists $t \in [t_o, t_f]$ and $k \in \mathbb{R}^+$ such that $\tilde{x}^k \notin \mathcal{X}_{free}$. Then $J(\tilde{x}^k) > J(x)$, which contradicts

(13). Using proof by contradiction, we conclude that

$$\tilde{x}^k \in \mathcal{X}_{free} \forall\ t \in [t_o, t_f], \ \forall\ k \in \mathbb{R}^+. \qquad (15)$$

Assuming that a collision-free path exists between the agent and the target, it is straightforward that the minimizer trajectory is the target's location. Therefore, from (14) and (15), Algorithm 1 generates a sequence $\{\tilde{x}^k\}$ that converges to the target safely.[4]

With regards to Theorem 2, we should alert the reader that Algorithm 1 may not guarantee collision-avoidance if the default trajectory is not collision-free, that is if there exists $t \in [t_o, t_f]$ such that $x \notin \mathcal{X}_{free}$. Further, the result of Theorem 2 can generalize to non-convex functions that have only one minimum.

4.4 Comparison to Alternative Optimization Approaches

Algorithm 1 differs from controllers that compute control sequences over the entire time horizon in order to locally minimize the cost function. Rather, the proposed scheme utilizes the *time-evolving sensitivity* of the objective to an infinitesimal switch from v to u and searches a one-dimensional space for a finite duration of a single action that will optimally improve the cost. It does so using a closed-form expression and, as a result, avoids the expensive iterative computational search in high-dimensional spaces, while it may still get closer to the optimizer with one iterate.

Specifically, in terms of computational effort, Algorithm 1 computes two $n \times 1$ (state (2) and first-order adjoint (5) variables) and one $n \times n$ (second-order adjoint (7)) differential equations and searches. All simulations presented in the paper are able to run in real time, including the final 13-dimensional system. However, real-time execution is not guaranteed for higher dimensional systems. Nevertheless, the presented algorithm runs faster than the iLQG method for the simulations considered here.

Further, compared to traditional optimization algorithms such as iLQG, needle variation solutions exist globally, demonstrate a larger region of attraction and have a less complicated representation on Lie Groups (Fan and Murphey 2016). These traits naturally transfer to second-order needle controls (11) that also contain the first-order information present in (12). In addition, as this paper demonstrates, the suggested second-order needle variation controller has formal guarantees of descent for systems that are controllable with first-order Lie brackets, which—to the best of our knowledge—is not provided by any alternative method.

Given these benefits, the authors propose second-order needle variation actions as a complement to existing approaches for time-sensitive robotic applications that may be subject to large initial error, Euler angle singularities, or fast-evolving (and uncertain) objectives. Next, we implement Algorithm 1 using first or second-order needle variation controls (shown in (12) and (11), respectively) to compare them in terms of convergence success on various underactuated systems.
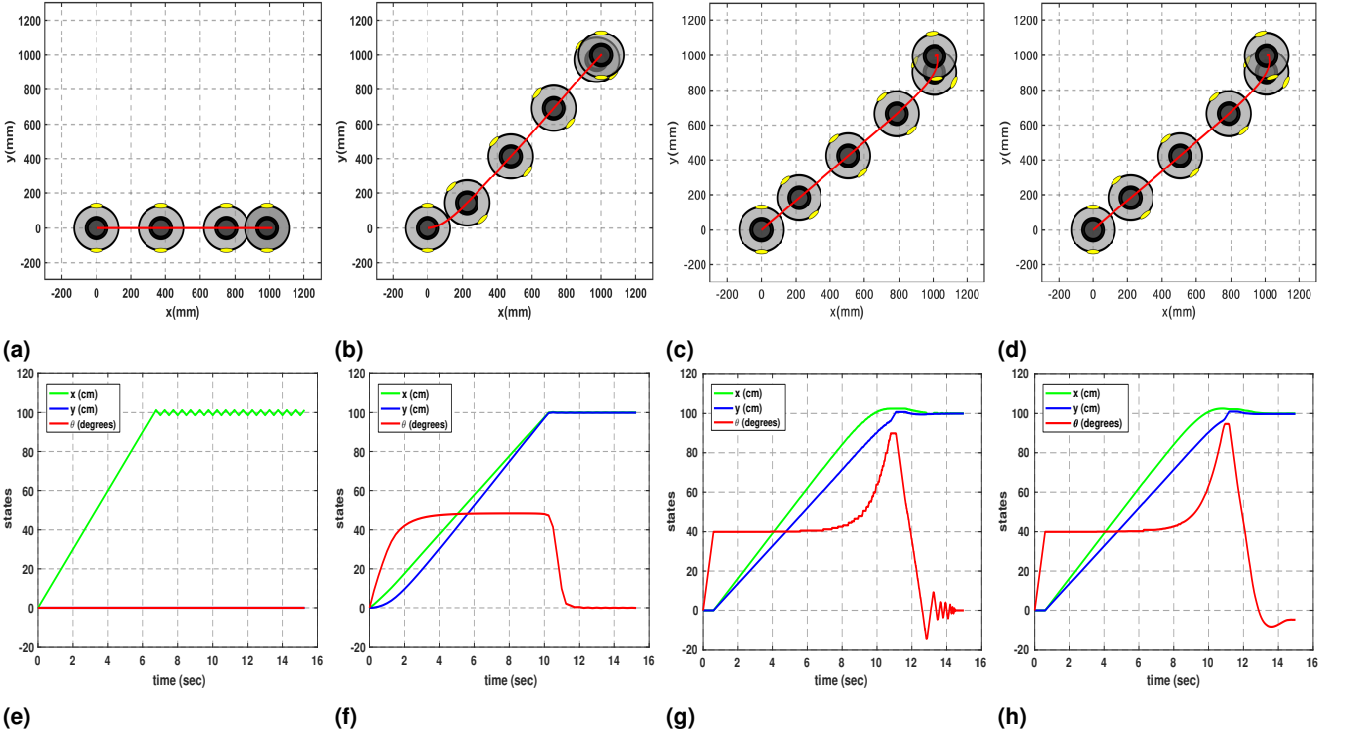
Figure 3. Differential drive using first-, second-order needle variation actions, iLQG, and DDP, from left to right. Snapshots of the system are shown at $t = 0, 2.5, 5, 7.5, 10$, and 12.5 sec. The target state is $[x_d, y_d, \theta_d] = [1000 \text{ mm}, 1000 \text{ mm}, 0]$.
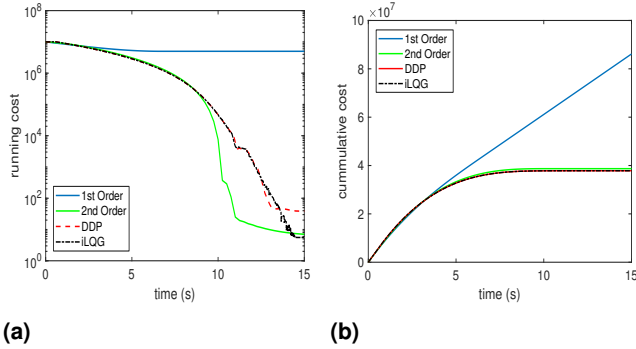

Figure 4. Fig. 4a plots the running state cost; Fig. 4b plots the integrated (cumulative) cost, including the control cost. DDP and iLQG obtain the same cumulative cost, with slightly different trajectories (see Fig. 3). Second-order needle variation actions demonstrate improved convergence to the target over DDP and iLQG, despite optimizing over one single action at each iteration.

5 Simulation Results

The proposed synthesis method based on (11) is implemented on three underactuated examples—the differential drive cart, a 3D kinematic rigid body and a dynamic model of an underwater vehicle. The kinematic systems of a 2D differential drive and a 3D rigid body are controllable using first-order Lie brackets of the vector fields and help demonstrate Theorem 1. The underactuated dynamic model of a 3D rigid body serves to compare controls in (11) and (12), as well as make comparisons to other control techniques, in a more sophisticated environment. In all simulation results, we start with default control $v = 0$ and an objective function of the

form

$$J(x(t)) = \frac{1}{2} \int_{t_o}^{t_f} \|\vec{x}(t) - \vec{x}_d(t)\|_Q^2 dt + \frac{1}{2} \|\vec{x}(t_f) - \vec{x}_d(t_f)\|_{P_1}^2,$$

where $\vec{x}_d$ is the desired state-trajectory, and $Q = Q^T \geq 0$, $P_1 = P_1^T \geq 0$ are metrics on state error.

5.1 2D Kinematic Differential Drive

We use the differential drive system to demonstrate that first-order controls shown in (12) that are based only on the first-order sensitivity of the cost function (3) can be insufficient for controllable systems, contrary to controls shown in (11) that guarantee decrease of the objective for systems that are controllable using first-order Lie brackets (see Theorem 1).

The system states are its coordinates and orientation, given by $s = [x, y, \theta]^T$, with kinematic $(g = 0)$ dynamics

$$f = r \begin{bmatrix} cos(\theta) & cos(\theta) \\ sin(\theta) & sin(\theta) \\ \frac{1}{L} & -\frac{1}{L} \end{bmatrix} \begin{bmatrix} u_R \\ u_L \end{bmatrix},$$

where $r = 3.6$ cm, $L = 25.8$ cm denote the wheel radius and the distance between them, and u_R, u_L are the right and left wheel control angular velocities, respectively (these parameter values match the specifications of the iRobot Roomba). The control vectors h_1, h_2 and their Lie bracket term $[h_1, h_2] = 2\frac{r^2}{L}[-sin(\theta), -cos(\theta)]^T$ span the state space ($\mathbb{R}^3$). Therefore, from Theorem 1, there always exist controls that reduce the cost to first or second order.

Fig. 3 and 4 demonstrate how first-order needle variation, second-order needle variation, iLQG, and DDP (Todorov and Li 2005; Tassa et al. 2014) perform on reaching a nearby target. We implement the iLQG and DDP algorithms to
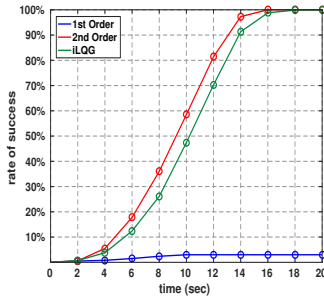
Figure 5. Convergence success rates of first- (12) and second-order (11) needle variation controls for the kinematic differential drive model. Simulation runs: 1000.

(a) **(b)**

Figure 6. Differential drive using second-order needle variation actions in the presence of obstacles. Fig. 6b shows the deviation from the nominal trajectory that is the solution to the no-obstacle task. The system performs two maneuvers to avoid each obstacle. These are evident in the angle deviation (compare to Fig. 3c).

generate offline trajectory optimization solutions using the publicly available software.[5] Actions based on first-order needle variations (12) do not generate solutions that turn the vehicle, but rather drive it straight until the orthogonal displacement between the system and the target location is minimized. Actions based on second-order needle variations (11), on the other hand, converge successfully. The solutions differ from the trajectories computed by iLQG and DDP, despite using the same simulation parameters.

We note the fact that, besides the computational benefits, single-action approaches appear to be rich in information and perform comparably to offline schemes that attempt to minimize the objective by computing different control responses over the entire horizon. Given that the solutions of iLQG and DDP are very similar, and the fact that DDP is slower than iLQG due to expanding the dynamics to second order, we use only the iLQG algorithm as a means of comparison for the rest of the simulations presented in this work. The results in Fig. 3 based on second-order needle variations are generated in real time in MATLAB and approximately forty times faster than the iLQG implementation.

Fig. 5 shows a Monte Carlo simulation that compares convergence success using first- and second-order needle variations controls shown in (12) and (11), respectively, and iLQG. We sampled over initial coordinates $x_0, y_0 \in [-1500, 1500]$ mm using a uniform distribution and keeping only samples for which the initial distance from the origin exceeded $L/5$; $\theta_0 = 0$ for all samples. Successful samples are defined by being within $L/5$ from the origin with an angle $\theta < \pi/12$ within 60 seconds using feedback sampling rate of 4 Hz. Results are generated using $Q = \text{diag}(10, 10, 1000)$, $P_1 = \text{diag}(0, 0, 0)$, $T = 0.5$ s, $R = \text{diag}(100, 100)$ for (12), $R = \text{diag}(0.1, 0.1)$ for (11), $\gamma = -15$, $\lambda = 0.1$ and saturation limits on the angular velocities of each wheel $\pm 150/36$ mm/s for each control approach.[6] As shown in Fig. 5, the system always converges to the target using second-order needle variation actions, matching the theory.

5.1.1 Convergence with Obstacles Next, we illustrate the performance of the algorithm in the presence of obstacles. In all simulations, obstacles are considered in the objective in the form of a penalty function. In Fig. 6, we test the system in the same task as Fig. 3 in the presence of two obstacles, indicated with red spheres. In comparison with Fig. 3f, it is worth noting the two angle peaks, corresponding to each
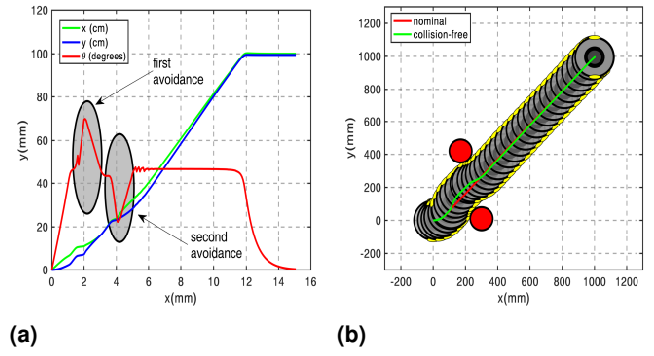
obstacle. After passing the obstacles, the system recovers the same angle profile.

Fig. 7 shows more complicated maneuvers using controls from (11). The controller, without relying on a motion planner, is able to generate collision-free trajectories and safely converge to the target in all cases. These simulations also demonstrate another aspect of Algorithm 1. The differential drive always drives up to an obstacle and then narrowly maneuvers around, instead of preparing a turn earlier on. This behavior is to be expected of needle variation actions that instantly reduce the cost.

We next use the more complicated scenario of Fig. 7d to evaluate the second-order expansion of the objective, shown in (9), across the state-space (see Fig. 8). The first- and second-order mode insertion gradients are computed based on the second-order needle variation controls from (11). States are sampled in the space for x and y in increments of 5 mm, with $\theta = 0$ everywhere. These results correspond to $\lambda = 0.001$. The horizontal discontinuity that appears around $y = 750$ mm is believed to be due to the effect of the penalty functions. As Fig. 8 indicates, the change in cost is always negative, verifying Theorem 1, even in the presence of obstacles.

We further use a Monte Carlo simulation of 500 trials on the initial conditions to test convergence success (Fig. 9). We sample initial conditions $[x, y]$ from a uniform distribution in [-200 mm, 1000 mm] $\times$ [-400 mm, 800 mm], where $\theta_o = 0$ in all cases. All trials converged within 25 seconds.

Last, we test the differential drive in the presence of moving obstacles (Fig. 10). The controller is again able to avoid collision and converge to the target, without relying on additional motion planning techniques. The feedback rate used is 20 Hz and the trajectory of the obstacles is known to the agent throughout the time horizon. In these simulations, $T = 0.3$ s.

5.2 3D Kinematic Rigid Body

The underactuated kinematic rigid body is a three dimensional example of a system that is controllable with first-order Lie brackets. To avoid singularities in the state space, the orientation of the system is expressed in quaternions (Titterton and Weston 2004; Kuipers 1999). The
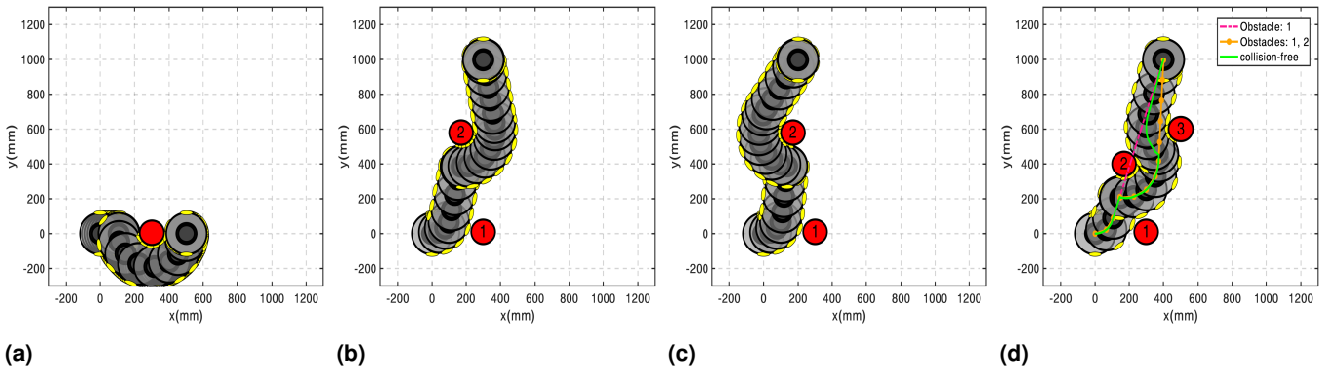
Figure 7. Trajectories of the differential drive in the presence of obstacles. Fig. 7d compares the solution to the trajectories generated when considering only a) obstacle 1, and b) obstacles 1 and 2, both of which collide with the obstacles. Simulations run in real time in MATLAB.
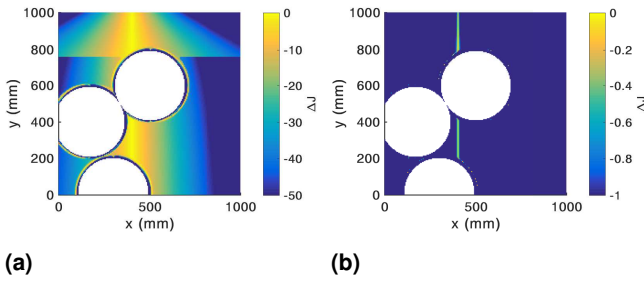

Figure 8. Cost reduction ΔJ, modeled after (9), for sampled x and y in the presence of obstacles. The first- and second-order mode insertion gradients are evaluated with the controls from (11). Figures 8a and 8b are identical, but shown over a different range to illustrate that even when looking at small variations of the first-order mode insertion gradient, the second-order method is reliably negative. The bright vertical line in Fig. 8b is vertically aligned with the target located at [400 mm, 1000 mm], where first-order solutions are singular. No data are sampled inside the white circles, as these indicate the infeasible occupied region.
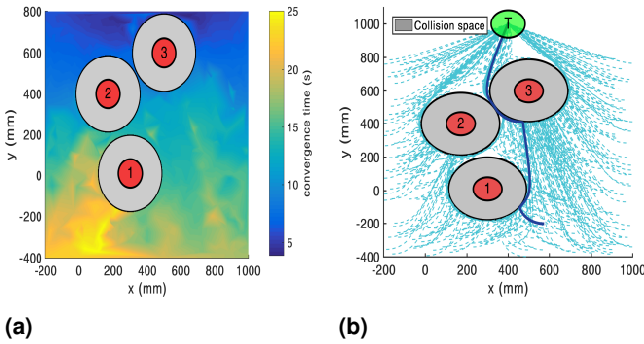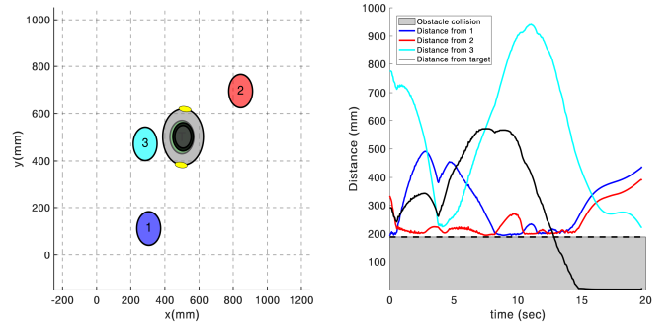

Figure 10. Performance of second-order needle variation actions in the presence of three moving obstacles. The left figure shows a snapshot of the simulation; the right figure plots the distance of the agent from each object and the target, where the gray area indicates the threshold minimum distance to avoid collision with the obstacles. The controller converges to the target in a collision-free manner (the simulation runs in real time in MATLAB). For visualization, watch Extension 2.

Figure 9. Performance of second-order needle variation actions in the presence of static obstacles. The controller is able to converge to the target for all 500 trials and avoid collisions. Fig. 9a is an interpolated heat map that indicates the time to convergence as a function of initial position; Fig. 9b shows the trajectories followed by the center of mass of the agent. The gray area indicates the collision space, taking into account the width of the differential drive. (the simulation runs in real time in MATLAB). For visualization, watch Extension 1.

states are $s = [x, y, z, q_0, q_1, q_2, q_3]$, where $b = [x, y, z]$ are the world-frame coordinates and $q = [q_0, q_1, q_2, q_3]$ are unit quaternions. Dynamics $f = [\dot{b}, \dot{q}]^T$ are given by

$$\dot{b} = R_q v, \tag{16}$$

$$\dot{q} = \frac{1}{2} \begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix} \omega, \tag{17}$$

where v and ω are the body frame linear and angular velocities, respectively (da Cunha 2015). The rotation matrix for quaternions is

$$R_q = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q1^2 - q_2^2 + q_3^2 \end{bmatrix}.$$

The system is kinematic: $v = F$ and $\omega = T$, where $F = (F_1, F_2, F_3)$ and $T = (T_1, T_2, T_3)$ describe respectively the surge, sway, and heave input forces, and the roll, pitch, and yaw input torques. We render the rigid body underactuated by removing the sway and yaw control authorities ($F_2 = T_3 = 0$).

The four control vectors span a four dimensional space. First order Lie bracket terms add two more dimensions to span the state space ($\mathbb{R}^6$) (the fact that there are seven states

in the model of the system is an artifact of the quaternion representation; it does not affect controllability).

The vectors $h_1, h_2, [h_2, h_3]$ span $\mathbb{R}^3$ associated with the world frame coordinate dynamics $\dot{x}, \dot{y}$, and $\dot{z}$. Similarly, vectors h_3, h_4, and $[h_4, h_3]$ also span $\mathbb{R}^3$ associated with the orientation. Thereby, control vectors and their first-order Lie brackets span the state space and, from Theorem 1, optimal actions shown in (11) will always reduce the cost function (3).

To verify this prediction, we present the convergence success of the system on 3D motion. Using Monte Carlo sampling with uniform distribution, initial locations are randomly generated such that $x_0, y_0, z_0 \in [-50, 50]$ cm keeping only samples for which the initial distance from the origin exceeded 6 cm. We regard as a convergence success each trial in which the rigid body is within 6 cm to the origin by the end of 60 seconds at any orientation. Results are generated at a sampling rate of 20 Hz using $Q = 0$, $P_1 = \text{diag}(100, 200, 100, 0, 0, 0, 0)$, $T = 1.0$ s, $\gamma = -50000$, $\lambda = 10^{-3}$, $R = 10^{-6} \text{diag}(1, 1, 100, 100)$ for (11), and $R = \text{diag}(10, 10, 1000, 1000)$ for controls in (12). Controls are saturated at ± 10 cm/s for the linear velocities and ± 10 rad/s for the angular ones. Using 280 simulations over 24 seconds, 80% satisfy the success criterion within 12 seconds and 100% of trajectories satisfy the success criterion within 20 seconds. None of the simulations converge for the first-order needle variation controls, because they cannot displace the system in the $\hat{y}$ direction.

5.3 Underactuated Dynamic 3D Fish

We represent the three dimensional rigid body with states $s = [b, q, v, \omega]^T$, where $b = [x, y, z]$ are the world-frame coordinates, $q = [q_0, q_1, q_2, q_3]$ are the quaternions that describe the world-frame orientation, and $v = [v_x, v_y, v_z]$ and $\omega = [\omega_x, \omega_y, \omega_z]$ are the body-frame linear and angular velocities. The rigid body dynamics are given by $\dot{b}$ and $\dot{q}$ shown in (16) and (17) and

$$M\dot{v} = Mv \times \omega + F,$$
$$J\dot{\omega} = J\omega \times \omega + T,$$

where the (experimentally determined) effective mass and moment of inertia of the rigid body are given by $M = \text{diag}(6.04, 17.31, 8.39)$ g and $J = \text{diag}(1.57, 27.78, 54.11)$g·cm², respectively. This example is inspired by work in Mamakoukas et al. (2016); Postlethwaite et al. (2009) and the parameters used for the effective mass and moment of inertia of a rigid body correspond to measurements of a fish. The control inputs are $F_2 = T_3 = 0$ and $F_3 \geq 0$.

The control vectors only span a four dimensional space and, since they are state-independent, their Lie brackets are zero vectors. However, the Lie brackets containing the drift vector field g (that also appear in the MIH expression) add from one to four (depending on the states) independent vectors such that control solutions in (11) guarantee decrease of the cost function (3) for a wider set of states than controls in (12).

Simulation results based on Monte Carlo sampling are shown in Fig. 11. Initial coordinates x_0, y_0, z_0 are generated using a uniform distribution in $[-100, 100]$ cm,
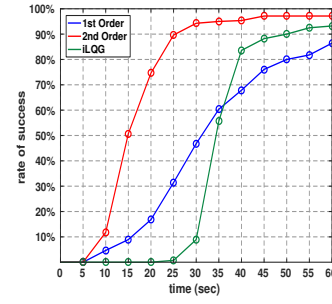

Figure 11. Convergence success rates of first- and second-order needle variation controls—(12) and (11), respectively—and iLQG for the underactuated *dynamic* vehicle model. Simulation runs: 280
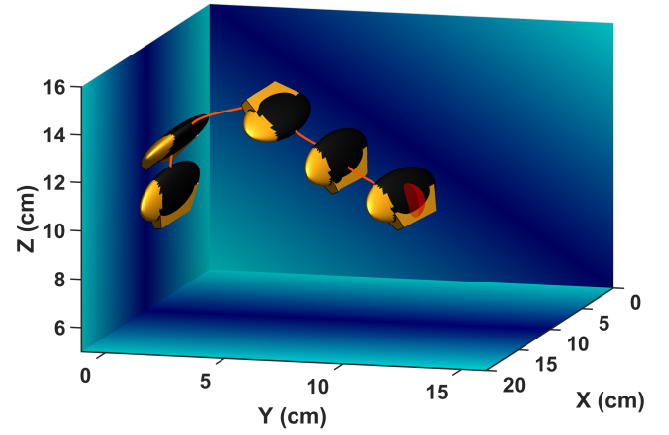

Figure 12. Snapshots of a parallel displacement maneuver using an underactuated dynamic vehicle model with second-order controls given by (11); first-order solutions (12) are singular throughout the simulation. Animation of these results is available in Extension 3.

discarding samples for which the initial distance to the origin is less than 15 cm. Successful trials are the ones for which, within a simulation window of 60 seconds, the system approached within 5 cm to the origin (at any orientation) and whose magnitude of the linear velocities is, at the same time, less than 5 cm/s. Results are generated at a sampling rate of 20 Hz using $T = 1.5$ s, $P_1 = 0$, $Q = \frac{1}{200}\text{diag}(10^3, 10^3, 10^3, 0, 0, 0, 0, 1, 1, 1, 2 \cdot 10^3, 10^3, 10^3)$, $\gamma = -5$, $R = \text{diag}(10^3, 10^3, 10^6, 10^6)$ for (12), $R = \frac{1}{2}\text{diag}(10^{-6}, 10^{-6}, 10^{-3}, 10^{-3})$ for (11), and $\lambda = 10^{-4}$. The same control saturations ($F_1 \in [-1, 1]$ mN, $F_3 \in [0, 1]$ mN, $T_1 \in [-0.1, 0.1]\,\mu$N·m, and $T_2 \in [-0.1, 0.1]\,\mu$N·m) are used for all simulations of the dynamic 3D fish. As shown in Fig. 11, controls computed using second-order needle variations converge faster than those based on first-order needle variations, and 97% of trials converge within 60 seconds.

Both methods converge over time to the desired location; as the dynamic model of the rigid body tumbles around and its orientation changes, possible descent directions of the cost function (3) change and the control is able to push the system to the target. Controls for the first-order needle variation case (12) are singular for a wider set of states than second-order needle variation controls (11) and, for this reason, they benefit more from tumbling. In a 3D parallel maneuver task, only second-order variation controls
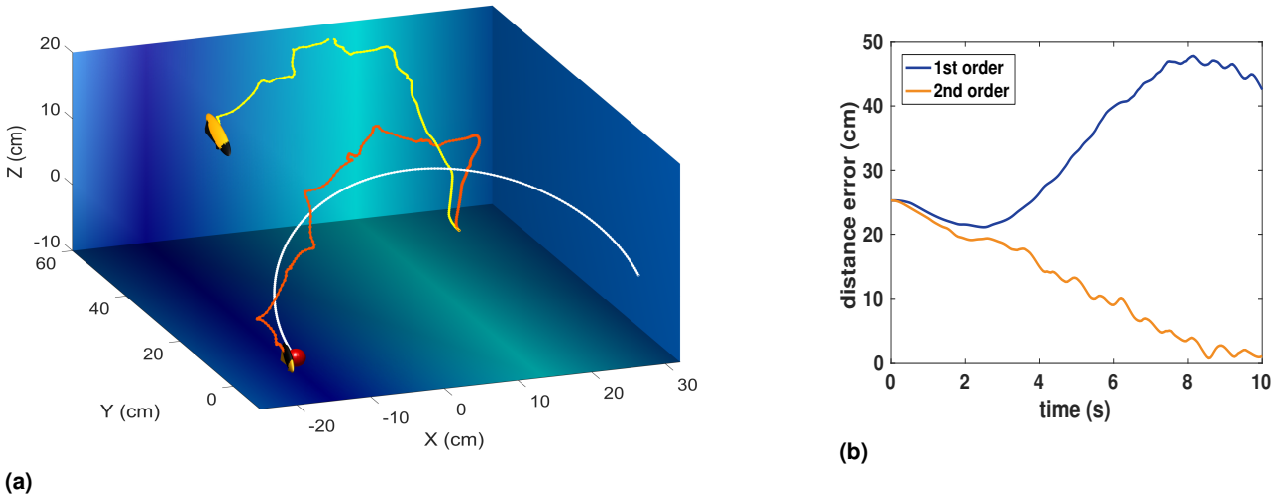
(a)

(b)

Figure 13. Tracking performance of the same system in the presence of +10 cm/s $\hat{y}$ fluid drift. The yellow system corresponds to first-order needle variation actions; the red one to second order. The target trajectory (red ball) is indicated with white traces over a 10-second simulation. Fig. 13b shows the error distance as a function of time, clearly demonstrating the advantage of the second-order approach. Animation of these results is available in Extension 4.

(11) manage to provide control solutions through successive heave and roll inputs, whereas controls based on first-order sensitivities (12) fail (see Fig. 12).

As controls in (11) are non-singular for a wider subset of the configuration state space than the first-order solutions in (12), they will provide more actions over a period of time and keep the system closer to a time-varying target. Fig. 13a demonstrates the superior trajectory tracking behavior of controls based on (11) in the presence of +10 cm/s $\hat{y}$ fluid drift. The trajectory of the target is given by $[x,y,z]=[\cos(\frac{3t}{10})(20+10\cos(\frac{t}{5}))$, $\sin(\frac{3t}{10})(20+10\cos(\frac{t}{5}))$, $10\sin(\frac{2t}{5})]$, with $T{=}2$ s, $\lambda{=}0.01$, $Q = \text{diag}(10,10,10,0,0,0,0,0,0,0,1,1,0.1)$, $\gamma{=}{-}50000$, $P_1{=}\text{diag}(10,10,10,0,0,0,0,0,0,0,0,0,0)$, $R{=}\text{diag}(10^3,10^3,10^6,10^6)$ for (12), and $R{=}\text{diag}(10,10,10^4,10^4)$ for (11). The simulation runs in real time using a C++ implementation on a laptop with Intel® Core™ i5-6300HQ CPU @2.30GHz and 8GB RAM.

The drift is known for both first- and second-order systems and accounted for in their dynamics in the form of $\dot{b} = \dot{b} + \dot{b}_{\text{drift}}$, where $\dot{b}_{\text{drift}}$ is a vector that points in the direction of the fluid flow. Simulation results demonstrate superior tracking of second-order needle variation controls that manage to stay with the target, whereas the system that corresponds to first-order needle variation controls is being drifted away by the flow.

We also tested convergence success of the +10 cm/s $\hat{y}$ drift case. Initial conditions x, y, z are sampled uniformly from a 30 cm radius from the origin, discarding samples for which the initial distance is less than 5 cm. We consider samples to be successful if, during 60 seconds of simulation, they approached the origin within 5 cm. Out of 500 samples, controls based on second-order variations converged 91% of the time (with average convergence time of 5.87 s), compared to 89% for first-order actions (with average convergence time of 9.3 s). Simulation parameters are $T{=}1$ s, $\gamma{=}{-}25000$, $Q{=}10^{-3}\text{diag}(10,10,10,0,0,0,0,1,1,1,1,1,1)$, $P_1{=}\text{diag}(100,100,100,0,0,0,0,\frac{1}{2},\frac{1}{2},\frac{1}{2},0,0,0)$, $\lambda{=}$

10^{-4}, $R{=}\text{diag}(0.1,0.1,10^4,10^4)$ for (12), and $R{=}\frac{1}{2}\text{diag}(10^{-5},10^{-5},1,1)$ for (11).

6 Conclusion

This paper presents a needle variation control synthesis method for nonlinearly controllable systems that can be expressed in control affine form. Control solutions provably exploit the nonlinear controllability of a system and, contrary to other nonlinear feedback schemes, have formal guarantees with respect to decreasing the objective. By optimally perturbing the system with needle actions, the proposed algorithm avoids the expensive iterative computation of controls over the entire horizon that other NMPC methods use and is able to run in real time for the systems considered here.

Simulation results on three underactuated systems compare first-order needle variation controls, second-order needle variation controls, and iLQG controls and demonstrate the superior convergence success rate of the proposed feedback synthesis. Because second-order needle variation actions are non-singular for a wider set of the state space than controls based on first-order sensitivity, they are also more suitable for time-evolving objectives, as demonstrated by the trajectory tracking examples in this paper. Second order needle variation controls are also calculated at little computational cost and preserve control effort. These traits, demonstrated in the simulation examples of this paper, render feedback synthesis based on second- and higher-order needle variation methods a promising alternative feedback scheme for underactuated and nonlinearly controllable systems.

In the future, we wish to generalize Theorems 1 and 2 to guarantee solutions for all controllable systems. Further, we are interested in showing that the second-order control responses (11) can be applied over the entire horizon, and not only as a needle action. This would create a second-order continuous feedback scheme like iLQG, but with the formal guarantees for controllable systems. To test the feasibility

of the algorithm, we are planning on conducting underwater experiments with a fish-robot.

Notes

1. In this work, we consider second-order sensitivity with respect to an action centered at one single application time τ. It is also possible to consider the second-order sensitivity with respect to two application times τ_i and τ_j in the same iteration. Assuming that the entire control curve is a descent direction over the time horizon for second-order needle variation solutions, as we have proved is the case for first-order needle variation methods in recently submitted work Mamakoukas et al. (2018), multiple, second-order needle actions at different application times would still decrease the objective. On the other hand, searching for two application times would slow down the algorithm and was not preferred in this work.

2. If the control vectors span the state space $\mathbb{R}^N$, the costate vector $\rho \in \mathbb{R}^N$ cannot be orthogonal to each of them. Therefore, for first-order controllable (fully actuated) systems, there always exist controls for which the cost can be reduced to first order.

3. The application time of the inserted action is typically chosen to correspond to the most negative first-order mode insertion gradient.

4. Although control responses are constructed from a second-order Taylor series approximation of the objective, the iterated sequence is guaranteed to decrease the real cost function at each iteration. If the real cost function is convex with respect to the state (in the unconstrained sense), the iterated sequence will converge towards the only minimizer. Using a sufficient descent condition, the algorithm is guaranteed to converge to a point where the unconstrained derivative of the objective is zero (i.e., $D_x J(x)=0$), which, from Assumption 4, only happens at the global minimizer.

5. Available at http://www.mathworks.com/matlabcentral/fileexchange/52069-ilqg-ddp-trajectory-optimization.

6. The metric on control effort is necessarily smaller for (11), due to parameter λ. The parameter is chosen carefully to ensure that control solutions from (11) and (12) are comparable in magnitude.

Acknowledgment

References

Allgöwer, F., Findeisen, R. and Nagy, Z. K. (2004), 'Nonlinear model predictive control: From theory to application', *Journal of the Chinese Institute of Chemical Engineers* **35**(3), 299–315.

Ames, A. D., Grizzle, J. W. and Tabuada, P. (2014), Control barrier function based quadratic programs with application to adaptive cruise control, *in* 'IEEE Conference on Decision and Control (CDC)', pp. 6271–6278.

Ansari, A. R. and Murphey, T. D. (2016), 'Sequential action control: Closed-form optimal control for nonlinear and nonsmooth systems', *IEEE Transactions on Robotics* **32**(5), 1196–1214.

Aseev, S. and Veliov, V. (2014), 'Needle variations in infinite-horizon optimal control', *Variational and Optimal Control Problems on Unbounded Domains* **619**, 1–17.

Becker, A. and Bretl, T. (2010), Motion planning under bounded uncertainty using ensemble control., *in* 'Robotics: Science and Systems (RSS)'.

Borrmann, U., Wang, L., Ames, A. D. and Egerstedt, M. (2015), 'Control barrier certificates for safe swarm behavior', *Analysis and Design of Hybrid Systems* **48**(27), 68–73.

Boskovic, J. D., Li, S.-M. and Mehra, R. K. (1999), Intelligent control of spacecraft in the presence of actuator failures, *in* 'IEEE Conference on Decision and Control (CDC)', pp. 4472–4477.

Bouadi, H., Bouchoucha, M. and Tadjine, M. (2007*a*), 'Modelling and stabilizing control laws design based on backstepping for an uav type-quadrotor', *IFAC Proceedings* **40**(15), 245–250.

Bouadi, H., Bouchoucha, M. and Tadjine, M. (2007*b*), 'Sliding mode control based on backstepping approach for an UAV type-quadrotor', *World Academy of Science, Engineering and Technology* **26**(5), 22–27.

Boyd, S. and Vandenberghe, L. (2004), *Convex Optimization*, Cambridge University Press, Cambridge, U.K.

Brockett, R. W. (1982), Control theory and singular Riemannian geometry, *in* 'New Directions in Applied Mathematics', Springer, pp. 11–27.

Brockett, R. W. (1983), 'Asymptotic stability and feedback stabilization', *Differential Geometric Control Theory* **27**(1), 181–191.

Brown, S. C. and Passino, K. M. (1997), 'Intelligent control for an acrobot', *Journal of Intelligent & Robotic Systems* **18**(3), 209–248.

Bullo, F., Leonard, N. E. and Lewis, A. D. (2000), 'Controllability and motion algorithms for underactuated Lagrangian systems on Lie groups', *IEEE Transactions on Automatic Control* **45**(8), 1437–1454.

Bullo, F. and Lynch, K. M. (2001), 'Kinematic controllability for decoupled trajectory planning in underactuated mechanical systems', *IEEE Transactions on Robotics and Automation* **17**(4), 402–412.

Caldwell, T. M. and Murphey, T. D. (2011), 'Switching mode generation and optimal estimation with application to skid-steering', *Automatica* **47**(1), 50–64.

Caldwell, T. M. and Murphey, T. D. (2016), 'Projection-based iterative mode scheduling for switched systems', *Nonlinear Analysis: Hybrid Systems* **21**, 59–83.

Chen, N., Song, F., Li, G., Sun, X. and Ai, C. (2013), 'An adaptive sliding mode backstepping control for the mobile manipulator with nonholonomic constraints', *Communications in Nonlinear Science and Numerical Simulation* **18**(10), 2885–2899.

Choudhury, P. and Lynch, K. M. (2004), Trajectory planning for kinematically controllable underactuated mechanical systems, *in* 'Algorithmic Foundations of Robotics V', Springer, pp. 559–575.

Chow, W. (1940/1941), 'Über Systeme von linearen partiellen Differentialgleichungen erster Ordnung.', *Mathematische Annalen* **117**, 98–105.

URL: *http://eudml.org/doc/160043*

da Cunha, A. E. C. (2015), Benchmark: Quadrotor attitude control., *in* 'Applied Verification for Continuous and Hybrid Systems (ARCH)', pp. 57–72.

Deng, M., Inoue, A. and Sekiguchi, K. (2008), 'Lyapunov function-based obstacle avoidance scheme for a two-wheeled mobile robot', *Journal of Control Theory and Applications* **6**(4), 399–404.

Dmitruk, A. and Osmolovskii, N. (2014), 'On the proof of Pontryagin's maximum principle by means of needle variations', *arXiv preprint arXiv:1412.2363* .

Egerstedt, M., Wardi, Y. and Axelsson, H. (2006), 'Transition-time optimization for switched-mode dynamical systems', *IEEE Transactions on Automatic Control* **51**(1), 110–115.

El-Nagar, A. M., El-Bardini, M. and EL-Rabaie, N. M. (2014), 'Intelligent control for nonlinear inverted pendulum based on interval type-2 fuzzy PD controller', *Alexandria Engineering Journal* **53**(1), 23–32.

Escareño, J., Salazar, S., Romero, H. and Lozano, R. (2013), 'Trajectory control of a quadrotor subject to 2D wind disturbances', *Journal of Intelligent & Robotic Systems* **70**(1–4), 51–63.

Fan, T. and Murphey, T. D. (2016), Online feedback control for input-saturated robotic systems on Lie groups, *in* 'Robotics: Science and Systems (RSS)'.

Fang, Y., Dixon, W., Dawson, D. and Zergeroglu, E. (2003), 'Nonlinear coupling control laws for an underactuated overhead crane system', *IEEE/ASME Transactions on Mechatronics* **8**(3), 418–423.

Fierro, R., Lewis, F. L. and Lowe, A. (1999), 'Hybrid control for a class of underactuated mechanical systems', *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* **29**(6), 649–654.

Fliess, M., Lévine, J., Martin, P. and Rouchon, P. (1995), 'Flatness and defect of non-linear systems: introductory theory and examples', *International Journal of Control* **61**(6), 1327–1361.

Garavello, M. and Piccoli, B. (2005), 'Hybrid necessary principle', *SIAM Journal on Control and Optimization* **43**(5), 1867–1887.

Harris, C. J., Moore, C. G. and Brown, M. (1993), *Intelligent Control: Aspects of Fuzzy Logic and Neural Nets*, Vol. 6, World Scientific, Singapore.

Hsu, D., Kindel, R., Latombe, J.-C. and Rock, S. (2002), 'Randomized kinodynamic motion planning with moving obstacles', *International Journal of Robotics Research* **21**(3), 233–255.

Jakubczyk, B. (2001), 'Introduction to geometric nonlinear control; controllability and Lie bracket', *Mathematical Control Theory* **1**(2), 107–168.

Junkins, J. L. and Thompson, R. C. (1986), 'An asymptotic perturbation method for nonlinear optimal control problems', *Journal of Guidance, Control, and Dynamics* **9**(4), 391–396.

Khalil, H. K. (2002), *Nonlinear Systems*, Prentice-Hall.

Kokotovic, P. V. (1992), 'The joy of feedback: Nonlinear and adaptive', *IEEE Control Systems* **12**(3), 7–17.

Kolmanovsky, I. and McClamroch, N. H. (1995), 'Developments in nonholonomic control problems', *IEEE Control Systems* **15**(6), 20–36.

Kuipers, J. B. (1999), *Quaternions and Rotation Sequences*, Princeton Univ. Press, Princeton, NJ.

La Valle, S. M. (2011), 'Motion planning', *IEEE Robotics & Automation Magazine* **18**(2), 108–118.

Lafferriere, G. and Sussmann, H. (1991), Motion planning for controllable systems without drift, *in* 'Robotics and Automation', IEEE, pp. 1148–1153.

Lafferriere, G. and Sussmann, H. J. (1993), A differential geometric approach to motion planning, *in* 'Nonholonomic Motion Planning', Springer, pp. 235–270.

Lamiraux, F. and Laumond, J.-P. (2000), 'Flatness and small-time controllability of multibody mobile robots: Application to motion planning', *IEEE Transactions on Automatic Control* **45**(10), 1878–1881.

Laumond, J.-P., Sekhavat, S. and Lamiraux, F. (1998), 'Guidelines in nonholonomic motion planning for mobile robots', *Robot Motion Planning and Control* **229**, 1–53.

LaValle, S. M. and Kuffner Jr, J. J. (2001), 'Randomized kinodynamic planning', *International Journal of Robotics Research* **20**(5), 378–400.

Lin, Y., Wu, D., Wang, X., Wang, X. and Gao, S. (2014), 'Lift path planning for a nonholonomic crawler crane', *Automation in Construction* **44**, 12–24.

Lynch, K. M., Shiroma, N., Arai, H. and Tanie, K. (2000), 'Collision-free trajectory planning for a 3-DOF robot with a passive joint', *The International Journal of Robotics Research* **19**(12), 1171–1184.

Mamakoukas, G., Kalinowska, A., MacIver, M. A. and Murphey, T. D. (2018), 'Continuous feedback control using needle variations for nonlinear and hybrid systems', *Robotics and Automation Letters (RA-L)* . Submitted.

Mamakoukas, G., MacIver, M. A. and Murphey, T. D. (2016), Sequential action control for models of underactuated underwater vehicles in a planar ideal fluid, *in* 'American Control Conference (ACC)', pp. 4500–4506.

Mamakoukas, G., MacIver, M. A. and Murphey, T. D. (2017), 'Feedback synthesis for controllable underactuated systems using sequential second order actions', *arXiv preprint arXiv:1709.01947* .

McMickell, M. B. and Goodwine, B. (2007), 'Motion planning for nonlinear symmetric distributed robotic formations', *The International Journal of Robotics Research* **26**(10), 1025–1041.

Morbidi, F. and Prattichizzo, D. (2007), 'Sliding mode formation tracking control of a tractor and trailer-car system.', *Robotics: Science and Systems (RSS)* **3**, 126–133.

Morgansen, K. A., Duidam, V., Mason, R. J., Burdick, J. W. and Murray, R. M. (2001), Nonlinear control methods for planar carangiform robot fish locomotion, *in* 'International Conference on Robotics and Automation (ICRA)', Vol. 1, pp. 427–434.

Murphey, T. D. and Burdick, J. W. (2006), 'The power dissipation method and kinematic reducibility of multiple-model robotic systems', *IEEE Transactions on Robotics* **22**(4), 694–710.

Murray, R. M., Li, Z. and Sastry, S. S. (1994), *A Mathematical Introduction to Robotic Manipulation*, CRC press, chapter 18.

Murray, R. M. and Sastry, S. S. (1993), 'Nonholonomic motion planning: Steering using sinusoids', *IEEE Transactions on Automatic Control* **38**(5), 700–716.

Murray, W. (2010), 'Newton-type methods', *Wiley Encyclopedia of Operations Research and Management Science* .

Nakazono, K., Ohnishi, K., Kinjo, H. and Yamamoto, T. (2008), 'Vibration control of load for rotary crane system using neural network with GA-based training', *Artificial Life and Robotics* **13**(1), 98–101.

Nocedal, J. and Wright, S. J. (2006), *Numerical Optimization*, Springer, chapter 3.

Perruquetti, W. and Barbot, J.-P. (2002), *Sliding Mode Control in Engineering*, Marcel Dekker, New York.

Pontryagin, L., Boltyanskii, V., Gamkrelidze, R. and Mishchenko, E. (1962), 'The mathematical theory of optimal processes', *New York: Interscience* .

Postlethwaite, C. M., Psemeneki, T. M., Selimkhanov, J., Silber, M. and MacIver, M. A. (2009), 'Optimal movement in the prey strikes of weakly electric fish: a case study of the interplay of body plan and movement capability', *Journal of The Royal Society Interface* **6**(34), 417–433.

Prajna, S., Jadbabaie, A. and Pappas, G. J. (2007), 'A framework for worst-case and stochastic safety verification using barrier certificates', *IEEE Transactions on Automatic Control* **52**(8), 1415–1428.

Rashevsky, P. (1938), 'About connecting two points of a completely nonholonomic space by admissible curve', *Uch. Zapiski Ped. Inst. Libknechta* **2**, 83–94.

Rathinam, M. and Murray, R. M. (1998), 'Configuration flatness of Lagrangian systems underactuated by one control', *SIAM Journal on Control and Optimization* **36**(1), 164–179.

Reyhanoglu, M., van der Schaft, A., McClamroch, N. H. and Kolmanovsky, I. (1996), Nonlinear control of a class of underactuated systems, *in* 'IEEE Conference on Decision and Control (CDC)', pp. 1682–1687.

Rimon, E. and Koditschek, D. E. (1992), 'Exact robot navigation using artificial potential functions', *IEEE Transactions on Robotics and Automation* **8**(5), 501–518.

Ross, I. M. and Fahroo, F. (2004), 'Pseudospectral methods for optimal motion planning of differentially flat systems', *IEEE Transactions on Automatic Control* **49**(8), 1410–1413.

Rossmann, W. (2002), *Lie groups: An Introduction Through Linear Groups*, Vol. 5, Oxford University Press.

Rouchon, P., Fliess, M., Levine, J. and Martin, P. (1993), Flatness and motion planning: The car with n trailers, *in* 'European Control Conference (ECC)', pp. 1518–1522.

Roy, B. and Asada, H. (2007), 'Closed loop control of a gravity-assisted underactuated snake robot with application to aircraft wing-box assembly', *Robotics: Science and Systems (RSS)* **25**(5).

Sastry, S. (2013), *Nonlinear Systems: Analysis, Stability, and Control*, Vol. 10, Springer.

Schnabel, R. B. and Eskow, E. (1990), 'A new modified Cholesky factorization', *SIAM Journal on Scientific and Statistical Computing* **11**(6), 1136–1158.

Seto, D. and Baillieul, J. (1994), 'Control problems in super-articulated mechanical systems', *IEEE Transactions on Automatic Control* **39**(12), 2442–2453.

Shaikh, M. S. and Caines, P. E. (2007), 'On the hybrid optimal control problem: theory and algorithms', *IEEE Transactions on Automatic Control* **52**(9), 1587–1603.

Shammas, E. and de Oliveira, M. (2012), An analytic motion planning solution for the snakeboard., *in* 'Robotics: Science and Systems (RSS)', MIT Press, p. 297.

Strichartz, R. S. (1987), 'The Campbell-Baker-Hausdorff-Dynkin formula and solutions of differential equations', *Journal of Functional Analysis* **72**(2), 320–345.

Sussmann, H. J. (1991), Two new methods for motion planning for controllable systems without drift, *in* 'European Control Conference (ECC)', pp. 1501–1506.

Tanner, H. G., Loizou, S. and Kyriakopoulos, K. J. (2001), Nonholonomic stabilization with collision avoidance for mobile robots, *in* 'International Conference on Intelligent Robots and Systems', Vol. 3, pp. 1220–1225.

Tassa, Y., Mansard, N. and Todorov, E. (2014), Control-limited differential dynamic programming, *in* 'IEEE International Conference on Robotics and Automation (ICRA)', IEEE, pp. 1168–1175.

Teel, A. R., Murray, R. M. and Walsh, G. C. (1995), 'Nonholonomic control systems: from steering to stabilization with sinusoids', *International Journal of Control* **62**(4), 849–870.

Titterton, D. and Weston, J. L. (2004), *Strapdown Inertial Navigation Technology*, 2nd edn, IET.

Todorov, E. and Li, W. (2005), A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems, *in* 'Proceedings of the American Control Conference (ACC)', pp. 300–306.

Toussaint, G. J., Basar, T. and Bullo, F. (2000), Tracking for nonlinear underactuated surface vessels with generalized forces, *in* 'IEEE International Conference on Control Applications', pp. 355–360.

Utkin, V. I. (1992), *Sliding Modes in Control and Optimization*, Springer-Verlag.

Wang, L., Ames, A. D. and Egerstedt, M. (2017), 'Safety barrier certificates for collisions-free multirobot systems', *IEEE Transactions on Robotics* .

Wu, G. and Sreenath, K. (2016), 'Safety-critical geometric control for systems on manifolds subject to time-varying constraints', *IEEE Transactions on Automatic Control, in review* .

Xu, R. and Özgüner, Ü. (2008), 'Sliding mode control of a class of underactuated systems', *Automatica* **44**(1), 233–241.

Xu, X., Tabuada, P., Grizzle, J. W. and Ames, A. D. (2015), 'Robustness of control barrier functions for safety critical control', *the IFAC Conference on Analysis and Design of Hybrid Systems* **48**(27), 54–61.

Appendix A   Index to multimedia Extensions

The multimedia extensions to this article can be found online by following the hyperlinks from www.ijrr.org.

Table of Multimedia Extensions

Extension	Type	Description
1	Video	Collision-free convergence in the presence of static obstacles from random initial conditions
2	Video	Collision-free convergence in the presence of moving obstacles
3	Video	Parallel maneuver of 3D dynamic fish.
4	Video	Underactuated tracking of dynamic 3D Fish in the presence of drift.

Appendix B   Derivations and Proofs

B.1   Derivation of the Mode Insertion Hessian

Consider switched systems that are defined by dynamics

$$\dot{x}(t)=f\big(x(t),\Lambda,t\big)=\begin{cases} f_1\big(x(t),t\big)\,, & T_0\leq t<\tau_1 \\ f_2\big(x(t),t\big)\,, & \tau_1\leq t<\tau_1+\lambda_1 \\ f_1\big(x(t),t\big)\,, & \tau_1+\lambda_1\leq t<\tau_2 \\ f_3\big(x(t),t\big)\,, & \tau_2\leq t<\tau_2+\lambda_2 \\ f_1\big(x(t),t\big)\,, & \tau_2+\lambda_2\leq t<\tau_3 \\ \quad\vdots\quad\quad\vdots\quad\quad\vdots \\ f_1\big(x(t),t\big)\,, & \tau_L+\lambda_L\leq t<T_F \end{cases}$$

$$(18)$$

$$\text{subject to: } x(T_0)=x_0,$$

where T_0 is the initial time, T_F is the final time, $x_0:\mathbb{R}\mapsto\mathbb{R}^N$ is the initial state, L is the number of injected dynamics, $\tau=\{\tau_1,\tau_2,\cdots,\tau_L\}\in\mathbb{R}^L$ is a monotonically increasing set of switching times, $\Lambda=\{\lambda_1,\lambda_2,\cdots,\lambda_L\}\in\mathbb{R}^L$ is a set of control durations, $f_1:\mathbb{R}\mapsto\mathbb{R}^N$ specify the default dynamics, and $f_i:\mathbb{R}\mapsto\mathbb{R}^N$ describe the ith injected dynamics. The switching times are assumed to be fixed.

We note that, while the system dynamics f depend on the set of control durations Λ, the same is not true for the individual switch mode dynamics f_i. In addition, we refer to individual elements in the set Λ as either Λ_i or λ_i. We measure the performance of the system with the integral of the Lagrangian, $\ell(\cdot)$, and a terminal cost $m(\cdot)$, similar to (3).

$$J(\Lambda)=\int_{T_0}^{T_F}\ell(x(t))\mathrm{d}t+m(x(T_F)). \qquad (19)$$

We can re-write the dynamics using step functions, such that

$$\begin{aligned} f(x(t),\Lambda,t) =& \big[1(t-T_0)-1(t-\tau_1^-)\big]f_1\big(x(t),t\big) \\ &+\big[1(t-\tau_1^+)-1\big(t-(\tau_1+\lambda_1)^-\big)\big]f_2\big(x(t),t\big) \\ &+\big[1\big(t-(\tau_1+\lambda_1)^+\big)-1(t-\tau_2^-)\big]f_1\big(x(t),t\big) \\ &+... \\ &+\big[1\big(t-(\tau_{L-1}+\lambda_{L-1})^+\big)-1(t-\tau_L^-)\big]f_1\big(x(t),t\big) \\ &+\big[1(t-\tau_L^+)-1\big(t-(\tau_L+\lambda_L)^-\big)\big]f_L\big(x(t),t\big) \\ &+\big[1\big(t-(\tau_L+\lambda_L)^+\big)-1(t-T_F)\big]f_1\big(x(t),t\big). \end{aligned}$$

The superscripts $+$ and $-$ help avoid ambiguity at the switching times. We use directional derivatives to differentiate, where the slot derivative $D_i\mathcal{F}(\cdot,\cdot)$ is the partial derivative of a function $\mathcal{F}$ with respect to its ith argument. That is, $D_x\mathcal{F}$ indicates the derivative of $\mathcal{F}$ with respect to x and is the same as $\frac{\partial\mathcal{F}}{\partial x}$. Further, $D_{i,j}\mathcal{F}(\cdot,\cdot)$ denotes the second partial of a function $\mathcal{F}$ with respect to its first and second arguments. The step-function form of the dynamics makes it straightforward to compute the partial derivatives

$D_1f\big(x(t),\Lambda,t\big)$ and $D_2f\big(x(t),\Lambda,t\big)$. Specifically,

$$\begin{aligned} D_1f\big(x(t),\Lambda,t\big) =& \big[1(t-T_0)-1(t-\tau_1^-)\big]D_1f_1\big(x(t),t\big) \\ &+\big[1(t-\tau_1^+)-1\big(t-(\tau_1+\lambda_1)^-\big)\big]D_1f_2\big(x(t),t\big) \\ &+\big[1\big(t-(\tau_1+\lambda_1)^+\big)-1(t-\tau_2^-)\big]D_1f_1\big(x(t),t\big) \\ &+... \\ &+\big[1\big(t-(\tau_{L-1}+\lambda_{L-1})^+\big)-1(t-\tau_L^-)\big]D_1f_1\big(x(t),t\big) \\ &+\big[1(t-\tau_L^+)-1\big(t-(\tau_L+\lambda_L)^-\big)\big]D_1f_L\big(x(t),t\big) \\ &+\big[1\big(t-(\tau_L+\lambda_L)^+\big)-1(t-T_F)\big]D_1f_1\big(x(t),t\big) \end{aligned}$$

and

$$\begin{aligned} D_2f\big(x(t),\Lambda,t\big)=\Big\{&\delta\big(t-(\tau_k+\lambda_k)^-\big)f_k(x(t),t) \\ &-\delta\big(t-(\tau_k+\lambda_k)^+\big)f_1(x(t),t)\Big\}_{k=1}^L, \quad (20) \end{aligned}$$

where $\delta(\cdot)$ is the Dirac delta functions. Using variational calculus,

$$DJ(\Lambda)\cdot\theta=\int_{T_0}^{T_F}D\ell(x(r))\cdot z(r)\mathrm{d}r+Dm\big(x(T_F)\big)\cdot z(T_F) \qquad (21)$$

where $z(t):\mathbb{R}\mapsto\mathbb{R}^{N\times 1}$ is the variation of $x(t)$ due to the variation, θ, in Λ. Also,

$$\begin{aligned} \dot{z}(t)&=\frac{\partial}{\partial t}\frac{\partial x(t)}{\partial\Lambda}=\frac{\partial}{\partial\Lambda}\frac{\partial x(t)}{\partial t}=\frac{\partial}{\partial\Lambda}\dot{x}(t)=\frac{\partial}{\partial\Lambda}f\big(x(t),\Lambda,t\big) \\ &=D_1f\big(x(t),\Lambda,t\big)\cdot z(t)+D_2f\big(x(t),\Lambda,t\big)\cdot\theta, \\ &\qquad\text{subject to: } z(0)=\frac{\partial}{\partial\Lambda}x(0)=0. \end{aligned}$$

Define $A(t)\triangleq D_1f\big(x(t),\Lambda,t\big)$ and $B(t)\triangleq D_2f\big(x(t),\Lambda,t\big)$. Therefore, $\dot{z}$ is

$$\dot{z}(t)=A(t)\cdot z(t)+B(t)\cdot\theta$$
$$\text{subject to: } z(0)=0.$$

The above differential equation has the solution

$$z(t)=\int_{T_0}^{t}\Phi(t,r)B(r)\cdot\theta\mathrm{d}r, \qquad (22)$$

where $\Phi(t,r)$ is the state transition matrix corresponding to $A(t)$. Substituting $z(\cdot)$ in $DJ(\Lambda)\cdot\theta$,

$$\begin{aligned} DJ(\Lambda)\cdot\theta=&\int_{T_0}^{T_F}D\ell\big(x(r)\big)\int_{T_0}^{r}\Phi(r,s)B(s)\cdot\theta\mathrm{d}s\mathrm{d}r \\ &+Dm\big(x(T_F)\big)\int_{T_0}^{T_F}\Phi(T_F,s)B(s)\cdot\theta\mathrm{d}s. \end{aligned}$$

Switching the order of integration in the first-integral,

$$DJ(\Lambda)\cdot\theta = \int_{T_0}^{T_F}\int_s^{T_F} D\ell\big(x(r)\big)\Phi(r,s)B(s)\cdot\theta\,\mathrm{d}r\,\mathrm{d}s$$
$$+\int_{T_0}^{T_F} Dm\big(x(T_F)\big)\Phi(T_F,s)B(s)\cdot\theta\,\mathrm{d}s$$
$$=\int_{T_0}^{T_F}\Big[\underbrace{\int_s^{T_F} D\ell\big(x(r)\big)\Phi(r,s)\,\mathrm{d}r}$$
$$+\underbrace{\int_{T_0}^{T_F} Dm\big(x(T_F)\big)\Phi(T_F,s)}_{\rho(s)^T}\Big]B(s)\,\mathrm{d}s\cdot\theta.$$

Then,

$$\rho(t)=\Phi(T_F,t)^T Dm(x(T_F))^T+\int_t^{T_F}\Phi(r,t)^T D\ell(x(r))^T\,\mathrm{d}r,$$

where $\rho(t)$ is the solution to the backwards differential equation:

$$\dot{\rho}(t)=-D_1 f(x(t),\Lambda,t)^T\rho-D\ell(x(t))$$
$$\text{subject to: }\rho(T_F)=Dm(x(T_F))^T. \tag{23}$$

To avoid confusion, it is important to explain the notation used in the remaining of the derivation. We use θ to represent first-order and η to represent second-order perturbations to control durations Λ, respectively. We use subscripts to refer to the perturbation acting on a specific (single) duration. For example, θ_i indicates the perturbation that takes place with respect to the ith control duration, λ_i. We index the order of perturbations with a superscript, so that θ^j indicates the jth (in order) perturbation to the set of control durations Λ. Therefore, θ_1^2 indicates the perturbation that acts on the first control duration λ_1 and that is associated with the second perturbation.

We write

$$\frac{\partial}{\partial\Lambda}(DJ(\Lambda)\cdot\theta^1)=\frac{\partial}{\partial\Lambda}(\int_{T_0}^{T_N} D\ell(x(r))\cdot z^1(r)\,\mathrm{d}r$$
$$+Dm(x(T_F))\cdot z^1(T_F)),$$

and, using the product rule, we compute

$$D^2 J(\Lambda)\cdot(\theta^1,\theta^2)+DJ(\Lambda)\cdot\eta=\int_{T_0}^{T_F} D^2\ell(x(r))\cdot\big(z^1(r),z^2(r)\big)$$
$$+D\ell(x(r))\cdot\zeta(r)\,\mathrm{d}r$$
$$+D^2 m(x(T_F))\cdot(z^1(T_F),z^2(T_F))$$
$$+Dm\big(x(T_F)\cdot\zeta(T_F), \tag{24}$$

where θ^1 and θ^2 are two first-order variations of Λ, η is a second-order variation of Λ and $\zeta(t)$ is the second-order variation of $x(t)$. Parameter $\dot{\zeta}(t)$ is found by taking the

second-order switching time derivative of $\dot{x}(t)$:

$$\dot{\zeta}(t)=\frac{\partial^2}{\partial\Lambda^2}\dot{x}(t)=\frac{\partial}{\partial\Lambda}\dot{z}^1(t)$$
$$=\frac{\partial}{\partial\Lambda}\big(D_1 f(x(t),\Lambda,t)\cdot z^1(t)+D_2 f\big(x(t),\Lambda,t\big)\cdot\theta^1\big)$$

such that

$$\dot{\zeta}(t)=A(t)\cdot\zeta(t)+B(t)\cdot\eta$$
$$+\big(z^1(t)^T\ \ \theta^{1T}\big)\begin{pmatrix} D_1^2 f(x(t),\Lambda,t) & D_{1,2}f(x(t),\Lambda,t) \\ D_{2,1}f(x(t),\Lambda,t) & D_2^2 f(x(t),\Lambda,t)\end{pmatrix}\begin{pmatrix}z^2(t) \\ \theta^2\end{pmatrix}$$
$$\text{subject to:}\zeta(0)=\frac{\partial^2}{\partial\Lambda^2}x(0)=0.$$

Define

$$C(t)\triangleq\begin{pmatrix} D_1^2 f\big(x(t),\Lambda,t\big) & D_{1,2}f\big(x(t),\Lambda,t\big) \\ D_{2,1}f\big(x(t),\Lambda,t\big) & D_2^2 f\big(x(t),\Lambda,t\big)\end{pmatrix},$$

and notice that $\dot{\zeta}(t)$ is linear with respect to $\zeta(t)$ and therefore $\dot{\zeta}(t)$ has solution

$$\zeta(t)=\int_{T_0}^t\Phi(t,r)\Big[B(r)\cdot\eta+\big(z^1(r)^T\theta^{1T}\big)C(r)\begin{pmatrix}z^2(r) \\ \theta^2\end{pmatrix}\Big]\mathrm{d}t.$$

Substituting $\zeta(t)$ into (24), we see that

$$D^2 J(\Lambda)\cdot(\theta^1,\theta^2)+DJ(\Lambda)\cdot\eta=\int_{T_0}^{T_F}\Big[z^1(r)^T D^2\ell(x(r))z^2(r)$$
$$+D\ell(x(r))\int_{T_0}^r\Phi(r,s)\Big[B(s)\cdot\eta+(z^1(s)^T\theta^{1T})C(s)\begin{pmatrix}z^2(s) \\ \theta^2\end{pmatrix}\Big]\mathrm{d}s\Big]\mathrm{d}r$$
$$+z^1(T_F)^T D^2 m(x(T_F))z^2(T_F)+Dm(x(T_F))\cdot$$
$$\int_{T_0}^{T_F}\Phi(T_F,s)\Big[B(s)\cdot\eta+(z^1(s)^T\theta^{1T})C(s)\begin{pmatrix}z^2(s) \\ \theta^2\end{pmatrix}\Big]\mathrm{d}s.$$

Note that $DJ(\Lambda)\cdot\eta$ equals $\int_{T_0}^{T_F} D\ell(x(r))\int_{T_0}^r\Phi(r,s)B(s)\cdot\eta\,\mathrm{d}s\mathrm{d}r+Dm(x(T_F))\int_{T_0}^{T_F}\Phi(T_F,s)B(s)\cdot\eta\,\mathrm{d}s$, which is clear from (21) and (22). Therefore, this leaves

$$D^2 J(\Lambda)\cdot(\theta^1,\theta^2)=\int_{T_0}^{T_F}\Big[z^1(r)D^2\ell(x(r))z^2(r)+D\ell(x(r))$$
$$\int_{T_0}^\tau\Phi(r,s)(z^1(s)^T\theta^{1T})C(s)\begin{pmatrix}z^2(s) \\ \theta^2\end{pmatrix}\mathrm{d}s\Big]\mathrm{d}r$$
$$+z^1(T_F)^T D^2 m(x(T_F))z^2(T_F)$$
$$+Dm(x(T_F))\int_{T_0}^{T_F}\Phi(T_F,s)\big(z^1(s)^T\theta^{1T}\big)$$
$$C(s)\begin{pmatrix}z^2(s) \\ \theta^2\end{pmatrix}\mathrm{d}s.$$

Split the integral over $\mathrm{d}r$, move $D\ell\big(x(r)\big)$ and $Dm(x(T_F))$ into their respective integrals and switch the order of

integration of the double integral:

$$
= \int_{T_0}^{T_F} z^1(r)^T D^2 \ell(x(r)) z^2(r) dr
$$

$$
+ \int_{T_0}^{T_F} \int_s^{T_F} D\ell(x(r)) \Phi(r,s) \big(z^1(s)^T \theta^{1^T} \big) C(s) \begin{pmatrix} z^2(s) \\ \theta^2 \end{pmatrix} dr ds
$$

$$
+ z^1(T_F)^T D^2 m(x(T_F)) z^2(T_F)
$$

$$
+ \int_{T_0}^{T_F} Dm\big(x(T_F)\big) \Phi(T_F,s) \big(z^1(s)^T, \theta^{1^T} \big) C(s) \begin{pmatrix} z^2(s) \\ \theta^2 \end{pmatrix} ds.
$$

We combine the integrals over ds, and notice that $\rho(r)^T$, in (23), enters the equations. Furthermore, we switch the dummy variable s to r and put everything back under one integral:

$$
= \int_{T_0}^{T_F} z^1(r)^T D^2 \ell(x(r)) z^2(r) + \rho(r)^T \big(z^1(r)^T \theta^{1^T} \big) C(r) \begin{pmatrix} z^2(r) \\ \theta^2 \end{pmatrix} dr
$$

$$
+ z^1(T_F)^T D^2 m(x(T_F)) z^2(T_F).
$$

Expand $C(\cdot)$ back out,

$$
= \int_{T_0}^{T_F} z^1(r)^T D^2 \ell(x(r)) z^2(r)
$$

$$
+ \rho(r)^T \big[z^1(r)^T D_1^2 f(x(r),\Lambda,r) z^2(r) \big]
$$

$$
+ \rho(r)^T \big[z^1(r)^T D_{1,2} f(x(r),\Lambda,r) \theta^2 \big]
$$

$$
+ \rho(r)^T \big[\theta^{1^T} D_{2,1} f(x(r),\Lambda,r) z^2(r) \big]
$$

$$
+ \rho(r)^T \big[\theta^{1^T} D_2^2 f(x(r),\Lambda,r) \theta^2 \big] dr
$$

$$
+ z^1(T_F)^T D^2 m(x(T_F)) z^2(T_F).
$$

Switching to index notation, where $\rho_k(\cdot)$ is the kth component of $\rho(\cdot)$ and $f^k(\cdot,\cdot,\cdot)$ is the kth component of $f(\cdot,\cdot,\cdot)$,

$$
= \int_{T_0}^{T_F} z^1(r)^T D^2 \ell(x(r)) z^2(r)
$$

$$
+ z^1(r)^T \sum_{k=1}^n \rho_k(r) D_1^2 f^k(x(r,\Lambda,r) z^2(r)
$$

$$
+ z^1(r)^T \sum_{k=1}^n \rho_k(r) D_{1,2} f^k(x(r),\Lambda,r) \theta^2
$$

$$
+ \theta^{1^T} \sum_{k=1}^n \rho_k(r) D_{2,1} f^k(x(r),\Lambda,r) z^2(r)
$$

$$
+ \theta^{1^T} \sum_{k=1}^n \rho_k(r) D_2^2 f^k(x(r),\Lambda,r) \theta^2 dr
$$

$$
+ z^1(T_F)^T D^2(x(T_F)) z^2(T_F).
$$

Rearrange the terms allows $D^2 J(\Lambda) \cdot (\theta^1, \theta^2)$ to be partitioned into the summation of parts P_1, P_2, P_3 given by

$$
P_1 = \int_{T_0}^{T_F} z^1(r)^T \Big[D^2 \ell(x(r)) + \sum_{k=1}^n \rho_k(r) D_1^2 f^k(x(r),\Lambda,r) \Big]
$$

$$
z^2(r) dr + z^1(T_F)^T D^2 m(x(T_F)) z^2(T_F),
$$

$$
P_2 = \int_{T_0}^{T_F} \theta^{2^T} \sum_{k=1}^n \rho_k(r) D_{2,1} f^k(x(r),\Lambda,r) z^1(r) +
$$

$$
\theta^1 \sum_{k=1}^n \rho_k(r) D_{2,1} f^k(x(r),\Lambda,r) z^2(r) dr,
$$

$$
P_3 = \int_{T_0}^{T_F} \theta^{1^T} \sum_{k=1}^n \rho_k(r) D_2^2 f^k(x(r),\Lambda,r) \theta^2 dr
$$

Looking at P_1 first, let

$$
g(r) = D^2 \ell(x(r)) + \sum_{k=1}^n \rho_k(r) D_1^2 f^k(x(r),\Lambda,r).
$$

Then,

$$
P_1 = \int_{T_0}^{T_F} z^1(r)^T g(r) z^2(r) dr + z^1(T_F) D^2 m(x(T_F)) z^2(T_F).
$$

Substituting (22) for $z^{\cdot}(\cdot)$, results in

$$
= \int_{T_0}^{T_F} \Big[\int_{T_0}^r \Phi(r,s) B(s) \theta^1 ds \Big]^T g(r) \int_{T_0}^r \Phi(r,\omega) B(\omega) \theta^2 dw dr
$$

$$
+ \Big[\int_{T_0}^{T_F} \Phi(T_F,s) B(s) \theta^1 ds \Big]^T D^2 m(x(T_F)) \int_{T_0}^{T_F} \Phi(T_F,w)
$$

$$
B(w) \theta^2 dw.
$$

The integrals may be specified as follows:

$$
= \int_{T_0}^{T_F} \int_{T_0}^r \int_{T_0}^r \theta^{1^T} B(s)^T \Phi(r,s)^T g(r) \Phi(r,w) B(w) \theta^2 ds dw dr
$$

$$
+ \int_{T_0}^{T_F} \int_{T_0}^{T_F} \theta^1 B(s)^T \Phi(T_F,s)^T D^2 m(x(T_F)) \Phi(T_F,w) B(w) \theta^2
$$

$$
ds dw.
$$

Note that the volume of the triple integral is given by $r = max(s,w)$. Therefore, the order of integration may be switched to:

$$
= \int_{T_0}^{T_F} \int_{T_0}^{T_F} \int_{max(s,w)}^{T_F} \theta^{1^T} B(s)^T \Phi(r,s)^T g(r) \Phi(r,w) B(w) \theta^2 dr
$$

$$
ds dw
$$

$$
+ \int_{T_0}^{T_F} \int_{T_0}^{T_F} \theta^1 B(s)^T \Phi(T_F,s)^T D^2 m(x(T_F)) \Phi(T_F,w) B(w) \theta^2
$$

$$
ds dw.
$$

We combine the double integral with the triple integral and rearrange the terms so that only the ones depending on r are inside the internal integral:

$$= \int_{T_0}^{T_F}\int_{T_0}^{T_F} B(s)^T \Big[\int_{max(s,w)}^{T} \Phi(r,s)^T g(r)\Phi(r,w)\mathrm{d}r$$

$$+ \Phi(T_F,s)^T D^2 m\big(x(T_F)\big)\Phi(T_F,w)\Big] B(w)\mathrm{d}s\mathrm{d}w$$

$$\cdot (\theta^1,\theta^2).$$

Let

$$\Omega(t)= \int_{t}^{T_F} \Phi(r,t)g(r)\Phi(r,t)\mathrm{d}r + \Phi(T_F,t)^T D^2 m\big(x(T_F)\big)\Phi(T_F,t)$$

where $\Omega(t)\in\mathbb{R}^{n\times n}$ is the integral curve to the following differential equation

$$\dot{\Omega}(t) = -A(t)^T \Omega(t) - \Omega(t)A(t) - g(t)$$

$$= -A(t)^T \Omega(t) - \Omega(t)A(t) - D^2\ell\big(x(t)\big)$$

$$- \sum_{k=1}^{n} \rho_k(t) D_1^2 f^k\big(x(t),\Lambda,t\big)$$

$$\text{subject to:} \Omega(T_F)=D^2 m\big(x(T_F)\big).$$

Then, depending on the relationship between s and w, P_1 becomes

$$P_1 = \begin{cases} \int_{T_0}^{T_F}\int_{T_0}^{T_F} B(s)^T \Omega(s)\Phi(s,w)B(w)\mathrm{d}s\mathrm{d}w\cdot(\theta^1,\theta^2) & s>w \\ \int_{T_0}^{T_F}\int_{T_0}^{T_F} B(s)^T \Phi(w,s)^T \Omega(w)B(w)\mathrm{d}s\mathrm{d}w\cdot(\theta^1,\theta^2) & s<w \\ \int_{T_0}^{T_F}\int_{T_0}^{T_F} B(s)^T \Omega(s)B(w)\mathrm{d}s\mathrm{d}w\cdot(\theta^1,\theta^2) & s=w, \end{cases}$$

P_1 is a scalar and equal to its transpose, therefore

$$P_1 \stackrel{s\leq w}{=} \int_{T_0}^{T_F}\int_{T_0}^{T_F} B(w)^T \Omega(w)\Phi(w,s)B(s)\mathrm{d}s\mathrm{d}w\cdot(\theta^2,\theta^1)$$

Use i and j to index θ^1 and θ^2 respectively, where θ indicate the variations of Λ and $i,j=1,...,L$. Integrating the δ-functions in $B(s)$ and $B(w)$ will pick out times $s=\tau_i+\lambda_i$ and $w=\tau_j+\lambda_j$ such that $P_{1_{ij}}$ is given by

$$P_{1_{ij}} \begin{cases} \stackrel{i\geq j}{=} \Big[f_i\big(x(\tau_i+\lambda_i),t\big)-f_1\big(x(\tau_i+\lambda_i),t\big)\Big]^T \\ \qquad \Omega(\tau_i+\lambda_i)\Phi(\tau_i+\lambda_i,\tau_j+\lambda_j) \\ \qquad \Big[f_j\big(x(\tau_j+\lambda_j),t\big)-f_1\big(x(\tau_j+\lambda_j),t\big)\Big]\cdot(\theta_i^1,\theta_j^2) \\ \stackrel{i\leq j}{=} \Big[f_j\big(x(\tau_j+\lambda_j),t\big)-f_1\big(x(\tau_j+\lambda_j),t\big)\Big]^T \\ \qquad \Omega(\tau_j+\lambda_j)\Phi(\tau_j+\lambda_j,\tau_i+\lambda_i) \\ \qquad \Big[f_i\big(x(\tau_i+\lambda_i),t\big)-f_1\big(x(\tau_i+\lambda_i),t\big)\Big]\cdot(\theta_i^1,\theta_j^2) \\ \stackrel{i=j}{=} \Big[f_i\big(x(\tau_i+\lambda_i),t\big)-f_1\big(x(\tau_i+\lambda_i),t\big)\Big]^T \Omega(\tau_i+\lambda_i) \\ \qquad \Big[f_i\big(x(\tau_i+\lambda_i),t\big)-f_1\big(x(\tau_i+\lambda_i),t\big)\Big]\cdot(\theta_i^1,\theta_i^2) \end{cases}$$

Taking the limit $\Lambda\to 0$, $\lim_{\Lambda\to 0} P_{1_{ij}}$ becomes

$$\lim_{\Lambda\to 0} P_{1_{ij}} \begin{cases} \stackrel{i\geq j}{=} [f_i(x(\tau_i),t)-f_1(x(\tau_i),t)]^T \Omega(\tau_i)\Phi(\tau_i,\tau_j) \\ \qquad [f_j(x(\tau_j),t)-f_1(x(\tau_j),t)]\cdot(\theta_i^1,\theta_j^2) \\ \stackrel{i\leq j}{=} [f_j(x(\tau_j),t)-f_1(x(\tau_j),t)]^T \Omega(\tau_j)\Phi(\tau_j,\tau_i) \\ \qquad [f_i(x(\tau_i),t)-f_1(x(\tau_i),t)]\cdot(\theta_i^1,\theta_j^2) \\ \stackrel{i=j}{=} [f_i(x(\tau_i),t)-f_1(x(\tau_i),t)]\Omega(\tau_i) \\ \qquad [f_i(x(\tau_i),t)-f_1(x(\tau_i),t)]\cdot(\theta_i^1,\theta_i^2). \end{cases}$$

Now consider P_2, where

$$D_{2,1}f^k(x(t),\Lambda,t) = \Big\{ \delta\big(t-(\tau_a+\lambda_a)^-\big) D_1 f_a^k(x(t),t)^T$$

$$- \delta\big(t-(\tau_a+\lambda_a)^+\big) D_1 f_1^k(x(t),t)^T \Big\}_{a=1}^{L}.$$

Choose again the ith index of θ^1 and the jth index of θ^2, where $i,j=1,...,L$. This corresponds to the ith index of $z^1(t)$ and the jth index of $z^2(t)$, where the kth index of $z^{\cdot}(\cdot)$ is

$$z_k^{\cdot}(t)= \int_{T_0}^{t} \Phi(t,r)\Big[\delta\big(r-(\tau_k+\lambda_k)^-\big) f_k\big(x(r),r\big)$$

$$- \delta\big(r-(\tau_k+\lambda_k)^+\big) f_1\big(x(r),r\big)\Big]\mathrm{d}r\theta_k^{\cdot}, \qquad (25)$$

Specifying these indexes allows us to revert back to matrix representation for $\rho(\cdot)$ and $f(\cdot,\cdot,\cdot)$. Thus,

$$P_{2_{ij}} = \int_{T_0}^{T_F} \theta_j^2 \rho(r)^T \big[\delta\big(r-(\tau_j+\lambda_j)^-\big) D_1 f_j(x(r),r)$$

$$- \delta\big(r-(\tau_j+\lambda_j)^+\big) D_1 f_1(x(r),r)\big] z_i^1(r)$$

$$+ \theta_i^1 \rho(r)^T \big[\delta\big(r-(\tau_i+\lambda_i)^-\big) D_1 f_i(x(r),r)$$

$$- \delta\big(r-(\tau_i+\lambda_i)^+\big) D_1 f_1(x(r),r)\big] z_j^2(r)\mathrm{d}r.$$

Integrating over the δ-functions picks out the times for which the δ-functions' arguments are zero:

$$= \theta_j^2 \rho\big((\tau_j+\lambda_j)^-\big) D_1 f_j\big(x(\tau_j+\lambda_j)^-,(\tau_j+\lambda_j)^-\big) z_i^1\big((\tau_j+\lambda_j)^-\big)$$

$$- \theta_j^2 \rho\big((\tau_j+\lambda_j)^+\big) D_1 f_1\big(x(\tau_j+\lambda_j)^+,(\tau_j+\lambda_j)^+\big) z_i^1\big((\tau_j+\lambda_j)^+\big)$$

$$+ \theta_i^1 \rho\big((\tau_i+\lambda_i)^-\big)^T D_1 f_i\big(x(\tau_i+\lambda_i)^-,(\tau_i+\lambda_i)^-\big) z_j^2\big((\tau_i+\lambda_i)^-\big)$$

$$- \theta_i^1 \rho\big((\tau_i+\lambda_i)^+\big)^T D_1 f_1\big(x(\tau_i+\lambda_i)^+,(\tau_i+\lambda_i)^+\big) z_j^2\big((\tau_i+\lambda_i)^+\big).$$

The indexes i and j relate in three possible ways: $i<j$, $i=j$, or $i>j$. The first and last case are the same, which is based on the fact that partial derivatives (with respect to perturbations indexed with i and j) commute.

Recall that τ is a set of monotonically increasing times. Therefore, if $i>j$, then $\tau_i+\lambda_i>\tau_j+\lambda_j$. Given (25), $z_k^{\cdot}(t)$ is non-zero only after time $t=(\tau_k+\lambda_k)^-$. In other words, the state does not change up until the first injected control and so the state perturbation z will be zero for all times prior to the perturbations to the control duration. Consequently, because $t_j+\lambda_j<t_i+\lambda_i$, given that $i>j$ and so $z_i^1(\tau_j+\lambda_j)=0$.

Therefore, the first two terms of $P_{2_{ij}}$ are zero and

$$
\overset{i\geq j}{=}\theta_i^1\rho\big((\tau_i+\lambda_i)^-\big)^T D_1 f_i\big(x(\tau_i+\lambda_i)^-,(\tau_i+\lambda_i)^-\big)
$$
$$
\Phi\big((\tau_i+\lambda_i)^-,(\tau_j+\lambda_j)^-\big)\Big[f_j\big(x(\tau_j+\lambda_j)^-,(\tau_j+\lambda_j)^-\big)
$$
$$
-f_1\big(x(\tau_j+\lambda_j)^+,(\tau_j+\lambda_j)^+\big)\Big]\theta_j^2-\theta_i^1\rho\big((\tau_i+\lambda_i)^+\big)^T
$$
$$
D_1 f_1\big(x(\tau_i+\lambda_i)^+,(\tau_i+\lambda_i)^+\big)\Phi\big((\tau_i+\lambda_i)^-,(\tau_j+\lambda_j)^-\big)
$$
$$
\Big[f_j\big(x(\tau_j+\lambda_j)^-,(\tau_j+\lambda_j)^-\big)
$$
$$
-f_1\big(x(\tau_j+\lambda_j)^+,(\tau_j+\lambda_j)^+\big)\Big]\theta_j^2.
$$

Omitting the no longer useful superscripts $+$ and $-$, we see that

$$
\overset{i\geq j}{=}\theta_i^1\rho\big(\tau_i+\lambda_i\big)^T D_1 f_i\big(x(\tau_i+\lambda_i),\tau_i+\lambda_i\big)\Phi\big(\tau_i+\lambda_i,\tau_j+\lambda_j\big)
$$
$$
\Big[f_j\big(x(\tau_j+\lambda_j),\tau_j+\lambda_j\big)-f_1\big(x(\tau_j+\lambda_j),\tau_j+\lambda_j\big)\Big]\theta_j^2
$$
$$
-\theta_i^1\rho\big(\tau_i+\lambda_i\big)^T D_1 f_1\big(x(\tau_i+\lambda_i),\tau_i+\lambda_i\big)\Phi\big(\tau_i+\lambda_i,\tau_j+\lambda_j\big)
$$
$$
\Big[f_j\big(x(\tau_j+\lambda_j),\tau_j+\lambda_j\big)-f_1\big(x(\tau_j+\lambda_j),\tau_j+\lambda_j\big)\Big]\theta_j^2
$$

$$
\overset{i\geq j}{=}\theta_i^1\rho\big(\tau_i+\lambda_i\big)^T
$$
$$
\Big[D_1 f_i\big(x(\tau_i+\lambda_i),\tau_i+\lambda_i\big)-D_1 f_1\big(x(\tau_i+\lambda_i),\tau_i+\lambda_i\big)\Big]
$$
$$
\Phi\big(\tau_i+\lambda_i,\tau_j+\lambda_j\big)
$$
$$
\Big[f_j\big(x(\tau_j+\lambda_j),\tau_j+\lambda_j\big)-f_1\big(x(\tau_j+\lambda_j),\tau_j+\lambda_j\big)\Big]\theta_j^2.
$$

Taking the limit $\Lambda\to 0$,

$$
\lim_{\Lambda\to 0}P_2\overset{i\geq j}{=}\rho(\tau_i)^T[D_1 f_i(x(\tau_i),\tau_i)-D_1 f_1(x(\tau_i),\tau_i)]
$$
$$
\Phi(\tau_i,\tau_j)[f_j(x(\tau_j),\tau_j)-f_1(x(\tau_j),\tau_j)]\cdot(\theta_i^1,\theta_j^2).
$$

Now consider the $i=j$ case. Because $i=j$, the perturbations θ^1 and θ^2 are equivalent—in the sense that they are both perturbations to the same control duration λ_i—and therefore $z^1(t)$ and $z^2(t)$ are also equivalent. So,

$$
P_{2_{ij}}\overset{i=j}{=}2\theta_i^2\rho\big((\tau_i+\lambda_i)^-\big)^T D_1 f_i\big(x(\tau_i+\lambda_i)^-,(\tau_i+\lambda_i)^-\big)
$$
$$
z_i^1\big((\tau_i+\lambda_i)^-\big)-2\theta_i^2\rho\big((\tau_i+\lambda_i)^+\big)^T
$$
$$
D_1 f_1\big(x(\tau_i+\lambda_i)^+,(\tau_i+\lambda_i)^+\big)z_i^1\big((\tau_i+\lambda_i)^+\big)
$$

Substituting in for $z_i^1(\cdot)$,

$$
=2\theta_i^2\rho\big((\tau_i+\lambda_i)^-\big)^T D_1 f_i\big(x(\tau_i+\lambda_i)^-,(\tau_i+\lambda_i)^-\big)
$$
$$
\int_{T_0}^{(\tau_i+\lambda_i)^-}\Phi\big((\tau_i+\lambda_i)^-,r\big)\cdot\big[\delta\big(r-(\tau_i+\lambda_i)^-\big)f_i(x(r),r)
$$
$$
-\delta\big(r-(\tau_i+\lambda_i)^+\big)f_1(x(r),r)\big]\mathrm{d}r\theta_i^1
$$
$$
-2\theta_i^2\rho\big((\tau_i+\lambda_i)^+\big)^T D_1 f_1\big(x(\tau_i+\lambda_i)^+,(\tau_i+\lambda_i)^+\big)
$$
$$
\int_{T_0}^{(\tau_i+\lambda_i)^+}\Phi\big((\tau_i+\lambda_i)^+,r\big)\cdot\big[\delta\big(r-(\tau_i+\lambda_i)^-\big)f_i(x(r),r)
$$
$$
-\delta\big(r-(\tau_i+\lambda_i)^+\big)f_1(x(r),r)\big]\mathrm{d}r\theta_i^1.
$$

This time the arguments of the δ-functions are zero at the upper bounds of their integrals. So,

$$
=2\theta_i^2\rho\big((\tau_i+\lambda_i)^-\big)^T D_1 f_i\big(x(\tau_i+\lambda_i)^-,(\tau_i+\lambda_i)^-\big)\cdot
$$
$$
\frac{1}{2}\Phi\big((\tau_i+\lambda_i)^-,(\tau_i+\lambda_i)^-\big)f_i\big(x(\tau_i+\lambda_i)^-,(\tau_i+\lambda_i)^-\big)\theta_i^1
$$
$$
-2\theta_i^2\rho\big((\tau_i+\lambda_i)^+\big)^T D_1 f_1\big(x(\tau_i+\lambda_i)^+,(\tau_i+\lambda_i)^+\big)\cdot
$$
$$
\Big[\Phi\big((\tau_i+\lambda_i)^+,(\tau_i+\lambda_i)^-\big)f_i\big(x(\tau_i+\lambda_i)^-,(\tau_i+\lambda_i)^-\big)
$$
$$
-\Phi\big((\tau_i+\lambda_i)^+,(\tau_i+\lambda_i)^+\big)\frac{1}{2}f_1\big(x(\tau_i+\lambda_i)^+,(\tau_i+\lambda_i)^+\big)\Big]\theta_i^1.
$$

Recall that $\Phi\big((\tau_i+\lambda_i)^-,(\tau_i+\lambda_i)^-\big)=\Phi\big((\tau_i+\lambda_i)^+,(\tau_i+\lambda_i)^+\big)=I$ and that $\Phi(\cdot,\cdot)$ is a continuous operator, such that $\Phi\big((\tau_i+\lambda_i)^+,(\tau_i+\lambda_i)^-\big)=I$. Therefore, omitting the no longer helpful $-$ and $+$ super-scripts,

$$
=\rho(\tau_i+\lambda_i)^T\Big[D_1 f_i\big(x(\tau_i+\lambda_i),\tau_i+\lambda_i\big)f_i\big(x(\tau_i+\lambda_i),\tau_i+\lambda_i\big)
$$
$$
-2D_1 f_1\big(x(\tau_i+\lambda_i),(\tau_i+\lambda_i)\big)f_i\big(x(\tau_i+\lambda_i),(\tau_i+\lambda_i)\big)
$$
$$
+D_1 f_1\big(x(\tau_i+\lambda_i),(\tau_i+\lambda_i)\big)f_1\big(x(\tau_i+\lambda_i),(\tau_i+\lambda_i)\big)\Big]\cdot(\theta_i^1,\theta_i^2).
$$

Taking the limit $\Lambda\to 0$,

$$
\lim_{\Lambda\to 0}P_2\overset{i=j}{=}\rho(\tau_i)^T[D_1 f_i(x(\tau_i),\tau_i)f_i(x(\tau_i),\tau_i)
$$
$$
-2D_1 f_1(x(\tau_i),(\tau_i))f_i(x(\tau_i),(\tau_i))
$$
$$
+D_1 f_1(x(\tau_i),(\tau_i))f_1(x(\tau_i),(\tau_i))]\cdot(\theta_i^1,\theta_i^2).
$$

Finally, P_3. Start with $D_2^2 f^k\big(x(r),\lambda,r\big)$. For $i=j$,

$$
D_2^2 f^k\big(x(r),\lambda,r\big)_{ij}=\Big(\frac{\partial}{\partial\Lambda_i}\delta\big(r-(\tau_i+\lambda_i)^-\big)\Big)f_i^k\big(x(r),r\big)
$$
$$
-\Big(\frac{\partial}{\partial\Lambda_i}\delta\big(r-(\tau_i+\lambda_i)^+\big)f_1^k\big(x(r),r\big)\Big),
$$

and, for $i\neq j$, $D_2^2 f^k\big(x(r),\lambda,r\big)_{ij}=0$. Revert back to matrix representation of $\rho(\cdot)$ and $f(\cdot,\cdot)$. For $i=j$, using chair rule on $D_2^2 f^k\big(x(r),\Lambda,r\big)_{ij}$ results in:

$$
D_2^2 f^k\big(x(r),\Lambda,r\big)_{ij}=-\dot{\delta}\big(r-(\tau_i+\lambda_i)^-\big)f_i^k\big(x(r),r\big)
$$
$$
+\dot{\delta}\big(r-(\tau_i+\lambda_i)^+\big)f_1^k\big(x(r),r\big).
$$

Then,

$$
P_3=\int_{T_0}^{T_F}\Big[-\rho(r)^T\dot{\delta}\big(r-(\tau_i+\lambda_i)^-\big)f_i(x(r),r)
$$
$$
+\rho(r)^T\dot{\delta}\big(r-(\tau_i+\lambda_i)^+\big)f_1(x(r),r)\Big]\mathrm{d}r\cdot(\theta_i^1,\theta_i^2).
$$

Using integration by parts,

$$
\begin{aligned}
=&\Big[-\rho(r)^T\delta\big(r-(\tau_i+\lambda_i)^-\big)f_i(x(r),r)\Big|_{T_0}^{T_F}\\
&+\rho(r)^T\delta\big(r-(\tau_i+\lambda_i)^+\big)f_1(x(r),r)\Big|_{T_0}^{T_F}\\
&\int_{T_0}^{T_F}\Big[\dot{\rho}(r)^Tf_i(x(r),r)\\
&+\rho(r)^TD_1f_i(x(r),r)\dot{x}(t)+\rho(r)^TD_2f_i(x(r),r)\Big]\cdot\\
&\delta\big(r-(\tau_i+\lambda_i)^-\big)\mathrm{d}r-\int_{T_0}^{T_F}\Big[\dot{\rho}(r)^Tf_1(x(r),r)\\
&+\rho(r)^TD_1f_1(x(r),r)\dot{x}(t)+\rho(r)^TD_2f_1\big(x(r),r\big)\Big]\\
&\delta\big(r-(\tau_i+\lambda_i)^+\big)\mathrm{d}r\Big]\cdot(\theta_i^1,\theta_i^2).
\end{aligned}
$$

Integrating over the δ-functions picks out the times for which the δ-functions' arguments are zero:

$$
\begin{aligned}
=&\Big[\dot{\rho}\big((\tau_i+\lambda_i)^-\big)f_i\big(x\big((\tau_i+\lambda_i)^-\big),(\tau_i+\lambda_i)^-\big)\\
&-\dot{\rho}\big((\tau_i+\lambda_i)^+\big)f_1\big(x\big((\tau_i+\lambda_i)^+\big),(\tau_i+\lambda_i)^+\big)\\
&+\rho\big((\tau_i+\lambda_i)^-\big)^TD_1f_i\big(x\big((\tau_i+\lambda_i)^-\big),(\tau_i+\lambda_i)^-\big)\dot{x}\big((\tau_i+\lambda_i)^-\big)\\
&-\rho\big((\tau_i+\lambda_i)^+\big)^TD_1f_1\big(x\big((\tau_i+\lambda_i)^+\big),(\tau_i+\lambda_i)^+\big)\dot{x}\big((\tau_i+\lambda_i)^+\big)\\
&+\rho\big((\tau_i+\lambda_i)^-\big)^TD_2f_i\big(x\big((\tau_i+\lambda_i)^-\big),(\tau_i+\lambda_i)^-\big)\\
&-\rho\big((\tau_i+\lambda_i)^+\big)^TD_2f_1\big(x\big((\tau_i+\lambda_i)^+\big),(\tau_i+\lambda_i)^+\big)\Big]\cdot(\theta_i^1,\theta_i^2).
\end{aligned}
$$

Using (23),

$$
\begin{aligned}
=&\Big[\Big[-\rho\big((\tau_i+\lambda_i)^-\big)^TD_1f_i\big(x\big((\tau_i+\lambda_i)^-\big),(\tau_i+\lambda_i)^-\big)\\
&-D\ell\Big(x\big((\tau_i+\lambda_i)^-\big)\Big)\Big]\cdot f_i\big(x\big((\tau_i+\lambda_i)^-\big),(\tau_i+\lambda_i)^-\big)\\
&-\Big[-\rho\big((\tau_i+\lambda_i)^+\big)^TD_1f_1\big(x\big((\tau_i+\lambda_i)^+\big),(\tau_i+\lambda_i)^+\big)\\
&-D\ell\Big(x\big((\tau_i+\lambda_i)^+\big)\Big)\Big]\cdot f_1\big(x\big((\tau_i+\lambda_i)^+\big),\lambda_i,(\tau_i+\lambda_i)^+\big)\\
&+\rho\big((\tau_i+\lambda_i)^-\big)^TD_1f_i\big(x\big((\tau_i+\lambda_i)^-\big),(\tau_i+\lambda_i)^-\big)\\
&f_i\big(x\big((\tau_i+\lambda_i)^-\big),(\tau_i+\lambda_i)^-\big)-\rho\big((\tau_i+\lambda_i)^+\big)^T\\
&D_1f_1\big(x\big((\tau_i+\lambda_i)^+\big),(\tau_i+\lambda_i)^+\big)f_1\big(x\big((\tau_i+\lambda_i)^+\big),(\tau_i+\lambda_i)^+\big)\\
&+\rho\big((\tau_i+\lambda_i)^-\big)^TD_2f_i\big(x\big((\tau_i+\lambda_i)^-\big),(\tau_i+\lambda_i)^-\big)\\
&-\rho\big((\tau_i+\lambda_i)^+\big)^TD_2f_1\big(x\big((\tau_i+\lambda_i)^+\big),(\tau_i+\lambda_i)^+\big)\Big]\cdot(\theta_i^1,\theta_i^2).
\end{aligned}
$$

Canceling out terms,

$$
\begin{aligned}
=&\big[-D\ell\big(x\big((\tau_i+\lambda_i)^-\big)\big)\big(f_i\big(x\big((\tau_i+\lambda_i)^-\big),(\tau_i+\lambda_i)^-\big)\\
&-f_1\big(x\big((\tau_i+\lambda_i)^+\big),(\tau_i+\lambda_i)^+\big)\big)\\
&+\rho\big((\tau_i+\lambda_i)^-\big)^T\big(D_2f_i\big(x\big((\tau_i+\lambda_i)^-\big),(\tau_i+\lambda_i)^-\big)\\
&-D_2f_1\big(x\big((\tau_i+\lambda_i)^+\big),(\tau_i+\lambda_i)^+\big)\big)\big]\cdot(\theta_i^1,\theta_i^2).
\end{aligned}
$$

Then, taking $\Lambda\to0$ and omitting the superscripts,

$$
\begin{aligned}
\lim_{\Lambda\to0}P_3=&[D\ell(x(\tau_i))(f_i(x(\tau_i),\tau_i)-f_1(x(\tau_i),\tau_i))\\
&+\rho(\tau_i)^T(D_2f_i(x(\tau_i),\tau_i)-D_2f_1(x(\tau_i),\tau_i))]\cdot\\
&(\theta_i^1,\theta_i^2).
\end{aligned}
$$

Therefore, for $i\neq j$,

$$
\begin{aligned}
\lim_{\Lambda\to0}D^2J=&\Big[[f_i(x(\tau_i),\tau_i)-f_1(x(\tau_i),\tau_i)]^T\Omega(\tau_i)\\
&+\rho(\tau_i)^T[D_1f_i(x(\tau_i),\tau_i)-D_1f_1(x(\tau_i),\tau_i)]\Big]\cdot\\
&\Phi(\tau_i,\tau_j)[f_j(x(\tau_j),\tau_j)-f_1(x(\tau_j),\tau_j)]
\end{aligned}
$$

and, for $i=j$,

$$
\begin{aligned}
\lim_{\Lambda\to0}D^2J=&\Big[f_i\big(x(\tau_i),\tau_i\big)-f_1\big(x(\tau_i),\tau_i\big)\Big]^T\Omega(\tau_i)\\
&\Big[f_i\big(x(\tau_i),\tau_i\big)-f_1\big(x(\tau_i),\tau_i\big)\Big]\\
&+\rho(\tau_i)^T\Big[D_1f_i\big(x(\tau_i),\tau_i\big)f_i\big(x(\tau_i),\tau_i\big)\\
&-2D_1f_1\big(x(\tau_i),\tau_i\big)f_i\big(x(\tau_i),\tau_i\big)\\
&+D_1f_1\big(x(\tau_i),\tau_i\big)f_1\big(x(\tau_i),\tau_i\big)+D_2f_i\big(x(\tau_i),\tau_i\big)\\
&-D_2f_1\big(x(\tau_i),\tau_i\big)\Big]\\
&-D\ell(x(\tau_i))(f_i(x(\tau_i),\tau_i)-f_1(x(\tau_i),\tau_i)).
\end{aligned}
$$

Given dynamics of the form (1), the MIH (for $i=j$) takes the form in (6).

B.2 Proof of Proposition 5

Proof. The following analysis shows the algebraic dependence of the MIH expression on the first-order Lie brackets $[h_i,h_j]$ and $[g,h_i]$ and proves Proposition 5 if either: 1) $\rho^T[h_i,h_j]\neq0$ or 2) $\rho^T[g,h_i]\neq0$, as guaranteed by Proposition 4.

Consider controls such that $u_j=v_i\,\forall\,j,i\neq k$ and $v_k=0$ and expresses the MIH expression (6) as

$$
\frac{d^2J}{d\lambda_+^2}=u^T\mathcal{G}u-u_k((D_xl_1)h_k-\rho^T[g,h_k]),
$$

where $\mathcal{G}_{ij}=0\,\forall\,i,j\in[1,M]\backslash\{k\}$, $\mathcal{G}_{ik}=\mathcal{G}_{ki}=\frac{1}{2}[h_i,h_k]$, and $\mathcal{G}_{kk}=h_k^T\Omega h_k+\rho^TD_xh_k\cdot h_k$. The matrix $\mathcal{G}$ is shown to be either indefinite or negative semidefinite if there exists a Lie bracket term $[h_i,h_k]$ such that $\rho^T[h_i,h_k]\neq0$. From Proposition (4), there exist i, $j\in[1,M]$ such that either $\rho^T[h_i,h_j]\neq0$ or $\rho^T[g,h_i]\neq0$. Let $k\in[1,M]$ be an index chosen such that either $\rho^T[h_i,h_k]\neq0$ or $\rho^T[g,h_i]\neq0$ for some $i\in[1,M]\setminus\{k\}$.

We use summation notation to express the MIH as

$$\frac{d^2 J}{d\lambda^2} = \Big(\sum_{i=1}^{M} h_i(u_i - v_i)\Big)^T \Omega \sum_{j=1}^{M} h_j(u_j - v_j)$$

$$+\rho^T \Big[\frac{\partial g}{\partial x} g + \frac{\partial g}{\partial x}\sum_{j}^{M} h_j u_j + \sum_{j}^{M}\frac{\partial h_j}{\partial x} u_j g$$

$$+\sum_{j}^{M}\frac{\partial h_j}{\partial x} u_j \sum_{j}^{M} h_j u_j + \frac{\partial g}{\partial x} g + \frac{\partial g}{\partial x}\sum_{i}^{M} h_i v_i$$

$$+\sum_{i}^{M}\frac{\partial h_i}{\partial x} v_i g + \sum_{i}^{M}\frac{\partial h_i}{\partial x} v_i \sum_{i}^{M} h_i v_i - 2\frac{\partial g}{\partial x} g$$

$$-2\frac{\partial g}{\partial x}\sum_{j}^{M} h_j u_j - 2\sum_{i}^{M}\frac{\partial h_i}{\partial x} v_i g - 2\sum_{i}^{M}\frac{\partial h_i}{\partial x} v_i \sum_{j}^{M} h_j u_j\Big]$$

$$-\frac{\partial \ell}{\partial x}\Big(\sum_{i=1}^{M} h_i(u_i - v_i)\Big),$$

which can be simplified to

$$= \Big(\sum_{i=1}^{M} h_i(u_i - v_i)\Big)^T \Omega \sum_{j=1}^{M} h_j(u_j - v_j)$$

$$+\rho^T \Big(-\frac{\partial g}{\partial x}\sum_{j}^{M} h_j u_j + \sum_{j}^{M}\frac{\partial h_j}{\partial x} u_j g + \sum_{j}^{M}\frac{\partial h_j}{\partial x} u_j \sum_{j}^{M} h_j u_j$$

$$+\frac{\partial g}{\partial x}\sum_{i}^{M} h_i v_i - \sum_{i}^{M}\frac{\partial h_i}{\partial x} v_i g + \sum_{i}^{M}\frac{\partial h_i}{\partial x} v_i \sum_{i}^{M} h_i v_i$$

$$-2\sum_{i}^{M}\frac{\partial h_i}{\partial x} v_i \sum_{j}^{M} h_j u_j\Big) - \frac{\partial \ell}{\partial x}\Big(\sum_{i=1}^{M} h_i(u_i - v_i)\Big).$$

Rearranging the expression into quadratic and linear terms in the control input, we rewrite the MIH expression (6) as

$$= \Big(\sum_{i=1}^{M} h_i(u_i - v_i)\Big)^T \Omega \sum_{j=1}^{M} h_j(u_j - v_j)$$

$$+\rho^T \Big(\sum_{j}^{M}\frac{\partial h_j}{\partial x} u_j \sum_{j}^{M} h_j u_j + \sum_{i}^{M}\frac{\partial h_i}{\partial x} v_i \sum_{i}^{M} h_i v_i$$

$$-2\sum_{i}^{M}\frac{\partial h_i}{\partial x} v_i \sum_{j}^{M} h_j u_j\Big) - \frac{\partial \ell}{\partial x}\Big(\sum_{i=1}^{M} h_i(u_i - v_i)\Big)$$

$$+\rho^T \Big(-\frac{\partial g}{\partial x}\sum_{j}^{M} h_j u_j + \sum_{j}^{M}\frac{\partial h_j}{\partial x} u_j g + \frac{\partial g}{\partial x}\sum_{i}^{M} h_i v_i$$

$$-\sum_{i}^{M}\frac{\partial h_i}{\partial x} v_i g\Big).$$

Further considering controls such that $u_j = v_i \; \forall \; j, i \neq k$ and $v_k = 0$,

$$= (h_k u_k)^T \Omega(h_k u_k) + \rho^T \mathcal{D} - \frac{\partial \ell}{\partial x}(h_k u_k)$$

$$+\rho^T \Big(-\frac{\partial g}{\partial x} h_k u_k + \frac{\partial h_k}{\partial x} u_k g\Big),$$

where u_k is the kth control input and $\mathcal{D}$ is

$$\mathcal{D} = \sum_{j}^{M}\frac{\partial h_j}{\partial x} u_j \sum_{j}^{M} h_j u_j + \sum_{i}^{M}\frac{\partial h_i}{\partial x} v_i \sum_{i}^{M} h_i v_i$$

$$-2\sum_{i}^{M}\frac{\partial h_i}{\partial x} v_i \sum_{j}^{M} h_j u_j$$

$$= \Big(\sum_{j \neq k}^{M}\frac{\partial h_j}{\partial x} u_j\Big)\sum_{j \neq k}^{M} h_j u_j + \Big(\frac{\partial h_k}{\partial x} u_k\Big)\sum_{j \neq k}^{M} h_j u_j$$

$$+\Big(\sum_{j \neq k}^{M}\frac{\partial h_j}{\partial x} u_j\Big)h_k u_k$$

$$+\frac{\partial h_k}{\partial x} u_k h_k u_k + \Big(\sum_{i \neq k}^{M}\frac{\partial h_i}{\partial x} v_i\Big)\sum_{i \neq k}^{M} h_i v_i$$

$$-2\Big(\sum_{i \neq k}^{M}\frac{\partial h_i}{\partial x} v_i\Big)\sum_{j \neq k}^{M} h_j u_j - 2\Big(\sum_{i \neq k}^{M}\frac{\partial h_i}{\partial x} v_i\Big)h_k u_k$$

$$= u_k \sum_{j \neq k}^{M}\frac{\partial h_k}{\partial x} h_j u_j - u_k\Big(\sum_{j \neq k}^{M}\frac{\partial h_j}{\partial x} u_j h_k\Big) + \frac{\partial h_k}{\partial x} u_k h_k u_k$$

$$= u_k\Big[\sum_{j \neq k}^{M} u_j\Big(\frac{\partial h_k}{\partial x} h_j - \frac{\partial h_j}{\partial x} h_k\Big)\Big] + \frac{\partial h_k}{\partial x} u_k h_k u_k$$

$$= u_k\Big[\sum_{j \neq k}^{M} u_j [h_j, h_k]\Big] + \frac{\partial h_k}{\partial x} u_k h_k u_k,$$

where terms cancel because $u_j = v_i \; \forall \; j, i \neq k$. We use the property $x^T A x = x^T\big(\frac{1}{2}(A + A^T)\big)x$ and we write $\mathcal{D}$ in a matrix form,

$$= u^T \begin{pmatrix} 0 & \dots & \frac{1}{2}[h_1, h_k] & \dots & 0 \\ \vdots & \ddots & \frac{1}{2}[h_2, h_k] & \iddots & \vdots \\ \frac{1}{2}[h_1, h_k] & \frac{1}{2}[h_2, h_k] & \frac{\partial h_k}{\partial x} h_k & \frac{1}{2}[h_{M-1}, h_k] & \frac{1}{2}[h_M, h_k] \\ \vdots & \iddots & \frac{1}{2}[h_{M-1}, h_k] & \ddots & \vdots \\ 0 & \dots & \frac{1}{2}[h_M, h_k] & \dots & 0 \end{pmatrix} u.$$

The dotted entries in the matrix represent zero terms. Combining all terms, the MIH can be written as

$$\frac{d^2 J}{d\lambda^2} = u^T \mathcal{G} u - \frac{\partial \ell}{\partial x}(h_k u_k) + \rho^T\Big(-\frac{\partial g}{\partial x} h_k u_k + \frac{\partial h_k}{\partial x} u_k g\Big) \tag{26}$$

$$= u^T \mathcal{G} u - u_k\Big(\frac{\partial \ell}{\partial x} h_k - \rho^T[g, h_k]\Big), \tag{27}$$

where

$$\mathcal{G} = \begin{pmatrix} 0 & \dots & \frac{1}{2}\rho^T[h_1, h_k] & \dots & 0 \\ \vdots & \ddots & \frac{1}{2}\rho^T[h_2, h_k] & \iddots & \vdots \\ \frac{1}{2}\rho^T[h_1, h_k] & \frac{1}{2}\rho^T[h_2, h_k] & \mathcal{C}_1 & \frac{1}{2}\rho^T[h_{M-1}, h_k] & \frac{1}{2}\rho^T[h_M, h_k] \\ \vdots & \iddots & \frac{1}{2}\rho^T[h_{M-1}, h_k] & \ddots & \vdots \\ 0 & \dots & \frac{1}{2}\rho^T[h_M, h_k] & \dots & 0 \end{pmatrix},$$

and $\mathcal{C}_1 = h_k^T \Omega h_k + \rho^T \frac{\partial h_k}{\partial x} h_k$.

Given that $\frac{dJ}{d\lambda_+} = 0$, then, by Proposition 3, $\rho^T h_i = 0 \; \forall \; i \in [1, M]$. In addition, by Proposition 1, $\rho \neq 0$ and, by Proposition 4, there exist $i, j \in [1, M]$ such that $\rho^T[h_i, h_j] \neq 0$ or $\rho^T[g, h_i] \neq 0$. It is more convenient

to consider two cases that capture all possible scenarios: 1) $\rho^T[h_i,h_j] \neq 0$ and 2) $\rho^T[h_i,h_j] = 0$ (which implies $\rho^T[g,h_i] \neq 0$, by Proposition 3).

Case 1. $\rho^T[h_i,h_j] \neq 0$.

Let $\mathcal{G}_{[i,j]}$ denote a 2×2 matrix obtained from $\mathcal{G}$ by deleting all but its ith and j^{th} row and i^{th} and j^{th} column

$$\mathcal{G}_{[i,j]} = \begin{bmatrix} \mathcal{G}_{ii} & \mathcal{G}_{ij} \\ \mathcal{G}_{ji} & \mathcal{G}_{jj} \end{bmatrix},$$

where $\mathcal{G}_{ij} = \mathcal{G}_{ji}$ because $\mathcal{G}$ is symmetric. The principal minors of $\mathcal{G}$ of order 2 are given by $\Delta_2 = \det(\mathcal{G}_{[i,j]}) = \mathcal{G}_{ii}\mathcal{G}_{jj} - \mathcal{G}_{ij}^2\mathcal{G}_{ji} \, \forall \, i \neq j \in [1,M]$.

Consider first the diagonal terms of $\mathcal{G}_{[i,j]}$. We note that, because $i \neq j$ and $\mathcal{G}_{ii} = 0 \, \forall \, i \neq k$, then either $\mathcal{G}_{ii} = 0$ or $\mathcal{G}_{jj} = 0$. Therefore, $\Delta_2 = -\mathcal{G}_{ij}\mathcal{G}_{ji} \, \forall \, i \neq j \in [1,M]$. Next, consider the off-diagonal elements. We note that $\mathcal{G}_{ij} = 0 \, \forall \, i, j \in [1,M] \setminus \{k\}$. Given that $\mathcal{G}_{ik} = \mathcal{G}_{ki} = \frac{1}{2}\rho^T[h_i,h_k] \, \forall \, i \in [1,M] \setminus \{k\}$, we have $\Delta_2 = 0 \, \forall \, i \in [1,M] \setminus \{k\}$ and $\Delta_2 = -\frac{1}{4}(\rho^T[h_i,h_k])^2$, otherwise. We summarize these cases as follows

$$\Delta_2 = \begin{cases} 0 & \forall \, i, j \in [1,M] \setminus \{k\} \\ -\mathcal{G}_{ij}^2 = -\frac{1}{4}(\rho^T[h_i,h_k])^2 \leq 0 & \text{otherwise.} \end{cases}$$

If there exists $i \in [1,M]$ such that $\rho^T[h_i,h_j] \neq 0$, there is at least one negative second-order principal minor. Therefore, $\mathcal{G}$ is indefinite and so there exist controls $u \in \mathbb{R}^M$ such that $u^T\mathcal{G}u < 0$.

Choose $u \in \mathbb{R}^M$ such that $u^T\mathcal{G}u < 0$ and let $u_k \in \mathbb{R}$ represent the kth element of u. If $u_k\big(\rho^T[g,h_k]-D_xlh_k\big) \leq 0$, then

$$u^T\mathcal{G}u < 0 \implies u^T\mathcal{G}u + u_k\big(\rho^T[g,h_k]-D_xlh_k\big) < 0.$$

Else, if $u_k\big(\rho^T[g,h_k]-D_xlh_k\big) > 0$, choose $u' = -u$ so that

$$u'_k\big(\rho^T[g,h_k]-D_xlh_k\big) = -u_k\big(\rho^T[g,h_k]-D_xlh_k\big) < 0$$

and

$$u'^T\mathcal{G}u' + u'_k\big(\rho^T[g,h_k]-D_xlh_k\big)$$
$$= u^T\mathcal{G}u - u_k\big(\rho^T[g,h_k]-D_xlh_k\big) < 0.$$

Therefore, if $\rho^T[h_i,h_j] \neq 0$, there always exists $u \in \mathbb{R}^M$ such that $\frac{d^2J}{d\lambda^2} < 0$.

Case 2. $\rho^T[h_i,h_j] = 0$.

If $\rho^T[h_i,h_j] = 0 \, \forall \, i, j \in [1,M]$, then, shown in Proposition 4, there exists $i \in [1,M]$ such that $\rho^T[g,h_i] \neq 0$. For $\rho^T[h_i,h_j] = 0$, the MIH becomes

$$u_k^2\big(h_k^T\Omega h_k + \rho^T D_x h_k\big) + u_k\big(\rho^T[g,h_k]-D_xlh_k\big), \quad (28)$$

which is a quadratic expression of the form ax^2+bx+c. Quadratic expressions become negative if and only if $a \leq 0$ or $b^2-4ac > 0$. Therefore, (28) takes negative values if and only if, for some time $t \in [t_o,t_o+T]$,

1. $h_k^T\Omega h_k + \rho^T D_x h_k \leq 0$ OR

2. $\rho^T[g,h_k]-D_xlh_k \neq 0$.

We consider the second condition: $\rho^T[g,h_k]-D_xlh_k \neq 0$. Because $\frac{dJ}{d\lambda_+} = 0 \, \forall \, u \in \mathbb{R}^M, \, \forall \, t \in [t_o,t_o+T]$, then

$$\begin{aligned} \frac{dJ}{d\lambda_+} = 0 &\implies \rho^T h_i = 0 \, \forall \, i \in [1,M], \, \forall \, t \in [t_o, t_o+T] \\ &\implies \rho^T h_k = 0 \, \forall \, t \in [t_o, t_o+T] \\ &\implies \rho^T h_k = 0 \text{ for } t = t_o+T \\ &\quad \text{AND } \dot{\rho}^T h_k = 0 \, \forall \, t \in [t_o, t_o+T] \\ &\implies D_x m h_k = 0 \text{ for } t = t_o+T \\ &\quad \text{AND} \\ &\quad (-D_xl-\rho^T D_x f_2)h_k = 0 \, \forall \, t \in [t_o, t_o+T] \\ &\implies (x-x_d)^T P_1 h_k = 0 \text{ for } t = t_o+T \\ &\quad \text{AND } (-(x-x_d)^T Q - \rho^T D_x f_2)h_k = 0 \\ &\quad \forall \, t \in [t_o, t_o+T]. \end{aligned}$$

Consider positive-definite weight matrices $Q = \delta P_1 \succ 0$, where δ is a scale factor. Then,

$$\begin{aligned} (x-x_d)^T P_1 h_k = 0|_{t_o+T} &\Leftrightarrow (x-x_d)^T \delta P_1 h_k = 0|_{t_o+T} \\ &\Leftrightarrow (x-x_d)^T Q h_k = 0|_{t_o+T}, \end{aligned}$$

and

$$\begin{aligned} \frac{dJ}{d\lambda_+} = 0 &\implies D_x m h_k = 0 \\ &\Leftrightarrow D_xlh_k = 0 \text{ AND } \rho^T D_x f_2 h_k = 0. \end{aligned}$$

Then, $\rho^T[g,h_k]-D_xlh_k = \rho^T[g,h_k] \neq 0$. Therefore, there exist control solutions $u \in \mathbb{R}^M$ such that the MIH expression becomes negative.

B.3 Derivation of equation (11)

In the following derivation, we treat the first- and second-order mode insertion gradient terms separately.

Associate f_1 with default control v and f_2 with injected control u, such that

$$\begin{aligned} f_1 &\triangleq f(x(t),v(t)) = g(x(t)) + h(x(t))v(t) \\ f_2 &\triangleq f(x(t),u(t)) = g(x(t)) + h(x(t))u(t) \end{aligned}$$

For simplicity, we drop the arguments as necessary. For the mode insertion gradient, the update step is straightforward

$$\frac{\partial}{\partial u}\frac{dJ}{d\lambda_+} = \frac{\partial}{\partial u}\rho^T(f_2-f_1) = \frac{\partial}{\partial u}\rho^T h(u-v) = \rho^T h, \quad (29)$$

$$\frac{\partial^2}{\partial u^2}\frac{dJ}{d\lambda_+} = 0. \quad (30)$$

The update step on the MIH is more complicated and so we divide the MIH expression into three parts

$$\frac{d^2J}{d\lambda_+^2} = \mathcal{A}_1 + \mathcal{A}_2 + \mathcal{A}_3,$$

where the terms $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$ are given by the following set of equations

$$\begin{aligned} \mathcal{A}_1 &= (f_2-f_1)^T \Omega (f_2-f_1) \\ \mathcal{A}_2 &= \rho^T(D_x f_2 \cdot f_2 + D_x f_1 \cdot f_1 - 2D_x f_1 \cdot f_2) \\ \mathcal{A}_3 &= -D_xl \cdot (f_2-f_1). \end{aligned}$$

Let $l_2 = \frac{d^2 J}{d\lambda_+^2}$. Using the Gâteux derivative,

$$\frac{\partial l_2}{\partial u} = \frac{\partial l_2(u+\epsilon\eta)}{\partial \epsilon}\Big|_{\epsilon=0} = \frac{\partial \mathcal{A}_1(\cdot)}{\partial \epsilon} + \frac{\partial \mathcal{A}_2(\cdot)}{\partial \epsilon} + \frac{\partial \mathcal{A}_3(\cdot)}{\partial \epsilon}\Big|_{\epsilon=0}.$$

Then,

$$\begin{aligned}
\frac{\partial \mathcal{A}_1}{\partial u} &= \frac{\partial}{\partial \epsilon}\mathcal{A}_1(u+\epsilon\eta)\Big|_{\epsilon=0} \\
&= \frac{\partial}{\partial \epsilon}\left([h((u+\epsilon\eta)-v)]^T\Omega\,[h((u+\epsilon\eta)-v)]\right)\Big|_{\epsilon=0} \\
&= (h\eta)^T\Omega\,(h\cdot(u+\epsilon\eta-v)) + (h(u+\epsilon\eta-v))^T\Omega\,h\eta\Big|_{\epsilon=0} \\
&= \eta^T h^T\Omega\,hu - \eta^T h^T\Omega\,hv + u^T h^T\Omega\,h\eta - v^T h^T\Omega\,h\eta \\
&= \left(u^T h^T(\Omega^T+\Omega)h - v^T h^T(\Omega^T+\Omega)h\right)\eta
\end{aligned}$$

$$\begin{aligned}
\frac{\partial \mathcal{A}_2}{\partial u} &= \frac{\partial}{\partial \epsilon}\mathcal{A}_2(u+\epsilon\eta)\Big|_{\epsilon=0} \\
&= \frac{\partial}{\partial \epsilon}\rho^T\left(D_x f_1\cdot f_1 - 2\,D_x f_1\cdot f_2 + D_x f_2\cdot f_2\right)\Big|_{\epsilon=0} \\
&= \frac{\partial}{\partial \epsilon}\rho^T\left(D_x(g+hv)\cdot(g+hv)\right. \\
&\qquad -2D_x(g+hv)(g+h(u+\epsilon\eta)) \\
&\qquad \left.+D_x(g+h(u+\epsilon\eta))(g+h(u+\epsilon\eta))\right)\Big|_{\epsilon=0} \\
&= \rho^T\left(-2\,D_x(g+hv)h\eta + D_x(h\eta)\cdot(g+hu)\right. \\
&\qquad \left.+D_x(g+hu)\cdot h\eta\right) \\
&= \rho^T\left(-2D_x(g+hv)h\eta + D_x(h\eta)g + D_x(h\eta)hu\right. \\
&\qquad \left.+D_x gh\eta + D_x(hu)h\eta\right) \\
&= \rho^T\left(-D_x gh\eta - 2D_x(hv)h\eta + D_x(h\,\eta)\cdot g\right. \\
&\qquad \left.+D_x(h\,\eta)\cdot hu + D_x(hu)\cdot h\eta\right) \\
&= \rho^T\left(-D_x g\cdot h\eta - 2D_x(\sum_{k=1}^m h_k u_{d_k})\cdot h\eta\right. \\
&\qquad +D_x(\sum_{k=1}^m h_k\eta_k)\cdot g + D_x(\sum_{k=1}^m h_k\eta_k)hu \\
&\qquad \left.+D_x(\sum_{k=1}^m h_k u_{2_k})\cdot h\eta\right) \\
&= -\rho^T D_x gh\eta - 2v^T(\sum_{k=1}^n(D_x h_k)\rho_k)h\eta \\
&\qquad +\eta^T(\sum_{k=1}^n(D_x h_k)\rho_k)g + \eta^T(\sum_{k=1}^n(D_x h_k)\rho_k)hu \\
&\qquad +u^T(\sum_{k=1}^n(D_x h_k)\rho_k)h\eta \\
&= -\rho^T D_x gh\eta - 2v^T(\sum_{k=1}^n(D_x h_k)\rho_k)\cdot h\eta \\
&\qquad +g^T(\sum_{k=1}^n(D_x h_k)\rho_k)^T\eta + u^T h^T\cdot(\sum_{k=1}^n(D_x h_k)\rho_k)^T\eta \\
&\qquad +u^T(\sum_{k=1}^n(D_x h_k)\rho_k)\cdot h\eta
\end{aligned}$$

$$\begin{aligned}
&= \left[-\rho^T D_x g\cdot h - 2v^T(\sum_{k=1}^n(D_x h_k)\rho_k)\cdot h\right. \\
&\qquad +g^T(\sum_{k=1}^n(D_x h_k)\rho_k)^T + u^T h^T\cdot(\sum_{k=1}^n(D_x h_k)\rho_k)^T \\
&\qquad \left.+u^T(\sum_{k=1}^n(D_x h_k)\rho_k)\cdot h\right]\cdot\eta.
\end{aligned}$$

Last,

$$\begin{aligned}
\frac{\partial \mathcal{A}_3}{\partial u} &= \frac{\partial}{\partial \epsilon}\mathcal{A}_3(u+\epsilon\eta)\Big|_{\epsilon=0} \\
&= -\frac{\partial l}{\partial x}\frac{\partial}{\partial \epsilon}\left(g+h(u+\epsilon\eta)-g-hv\right)\Big|_{\epsilon=0} \\
&= -\frac{\partial l}{\partial x}h\eta
\end{aligned}$$

Therefore,

$$\begin{aligned}
\frac{\partial l_2}{\partial u} &= u^T\left[h^T(\Omega^T+\Omega)h + h^T(\sum_{k=1}^n(D_x h_k)\rho_k)^T\right. \\
&\qquad \left.+(\sum_{k=1}^n(D_x h_k)\rho_k)h\right] - v^T\left[h^T(\Omega^T+\Omega)h\right. \\
&\qquad \left.+2(\sum_{k=1}^n(D_x h_k)\rho_k)h\right] - \rho^T D_x g\cdot h \\
&\qquad +g^T(\sum_{k=1}^n(D_x h_k)\rho_k)^T - \frac{\partial l}{\partial x}h.
\end{aligned}$$

Solving for the minimizer, $\frac{\partial l_2}{\partial u}^T = 0$, we get

$$\begin{aligned}
\frac{\partial l_2}{\partial u}^T &= 0 \Rightarrow \left[h^T(\Omega^T+\Omega)h + h^T(\sum_{k=1}^n(D_x h_k)\rho_k)^T\right. \\
&\qquad \left.+(\sum_{k=1}^n(D_x h_k)\rho_k)h\right]u = \left[h^T(\Omega^T+\Omega)h\right. \\
&\qquad \left.+2h^T(\sum_{k=1}^n(D_x h_k)\rho_k)^T\right]v + D_x g^T\rho h \\
&\qquad -(\sum_{k=1}^n(D_x h_k)\rho_k)g + h^T D_x l^T \\
&\Rightarrow u = \left[h^T(\Omega^T+\Omega)h + h^T(\sum_{k=1}^n(D_x h_k)\rho_k)^T\right. \\
&\qquad \left.+(\sum_{k=1}^n(D_x h_k)\rho_k)h\right]^{-1}\left[h^T(\Omega^T+\Omega)h\right. \\
&\qquad \left.+2h^T(\sum_{k=1}^n(D_x h_k)\rho_k)^T\right]v + D_x g^T\rho h \\
&\qquad -(\sum_{k=1}^n(D_x h_k)\rho_k)g + h^T D_x l^T \qquad (31)
\end{aligned}$$

The terms shown in (29) and (31), together with a penalty term for the control, are the gradient and Hessian terms used for the Newton update step that appears in (11).