Second-Order Switching Time and Magnitude Optimization for Impulsive Hybrid Systems

Yoke Peng Leong[1] and Todd Murphey[2]

***Abstract*— This paper presents a method to perform second-order impulsive hybrid system optimization that optimizes a cost functional over mode transition times and impulse magnitudes simultaneously. The derivation of the first-order and second-order derivatives of the cost functional with respect to switching times and impulse magnitudes is presented. An adjoint formulation is utilized to compute the derivatives for a faster convergence at a lower computational cost. An example in robotics illustrates the effectiveness of this optimization technique when measurement noise is present.**

I. INTRODUCTION

Switched system optimization is of interest to numerous recent works [1]–[12]. Most work [1]–[3], [5] focuses on optimizing a cost functional with respect to a finite number of times at which the dynamics of a hybrid system switches. They assume that the dynamical mode sequence and the number of switches are known. Other works [9]–[12] study a switching time optimization that also focuses on mode sequence generation. [6]–[8] consider switching time optimization for impulsive switched systems.

This paper extends results in [7] which study an optimization over both impulse times (switching times) and impulse magnitudes of an impulsive non-hybrid system. We develop a second-order optimization technique that simultaneously computes the optimal impulse times and magnitudes of an impulsive hybrid system assuming a finite number of impulses and a known dynamical mode sequence.

In this paper, the first-order and second-order adjoint formulations for the derivatives of a cost functional with respect to hybrid transition times and impulse magnitudes are derived. We show that the derivatives with respect to impulse times, the derivatives with respect to impulse magnitudes and the cross derivatives between impulse times and impulse magnitudes share the same first-order and second-order adjoint operators. Hence, to calculate the gradient and the Hessian required at every second-order optimization step, only two backward integrations of adjoint operators are necessary, independent of the total number of impulses. This result facilitates the implementation of a second-order optimization method (e.g., Newton's method) which has the advantage of quadratic convergence [13]. Convergence plots in Fig. 3 in Section VI show that this technique is effective even when noisy measurement data are present.

[1]California Institute of Technology, Pasadena, CA 91125, USA yleong@caltech.edu

[2]Northwestern University, Evanston, IL 60201, USA t-murphey@northwestern.edu

The rest of this paper is organized as follows: Section II explains the notation used in this paper. The optimization problem is defined in Section III. Optimization methods are discussed in Section IV. Section V computes the first-order, second-order and cross derivatives of a cost functional. An example is presented in Section VI illustrating the optimization technique and its performance. Section VII summarizes the findings of this work and discusses future research directions beyond this paper.

II. NOTATION

A trajectory $x(x_0, \tau_1, \tau_2, ..., \tau_N, \delta_1, \delta_2, ..., \delta_N, t)$ is abbreviated as $x(t)$, and $x_i(t)$ refers to a segment in $x(t)$ when $t \in [\tau_i, \tau_{i+1}]$.

To represent derivatives, $Df(\cdot)$ is used. The derivative of a function $f(\cdot)$ with respect to the n-th argument is written as $D_n f(arg_1, arg_2, ...)$, and the derivative of a function $f(\cdot)$ with respect to arg is written as $D_{arg} f(\cdot)$.

An operator M applied to U is written as $M \circ U$.

III. PROBLEM DEFINITION

Consider a n-dimensional nonlinear impulsive hybrid system governed by a sequence of N dynamical modes:

$$\dot{x}(t) = f_i(x(t), t) \text{ for } \tau_i < t < \tau_{i+1} \tag{1}$$

where $i \in \{1, 2, 3, ..., N\}$ and $x(\tau_i^+) = x_{i,0}$. Each $f_i(\cdot)$ is at least C^2 in x and C^1 in t. These dynamical modes are separated by $N-1$ distinct impulses for the total time horizon from $\tau_1 = t_0$ to $\tau_{N+1} = t_f$. The i-th impulse time is τ_i, and the impulse magnitude at τ_i is δ_i. At each impulse time, M out of a total of n states experience impulses, and thus, δ_i is a M-length vector. Note that δ_i is not a function of time, t, and δ and τ are independent of each other. The dynamical mode sequence and the number of impulses are assumed known.

The goal is to simultaneously solve for the $N-1$ unknown impulse times and the M unknown impulse magnitudes at each τ_i by minimizing a cost function which is defined as

$$J(\cdot) = \int_{t_0}^{t_f} l(x(s), s) ds \tag{2}$$

where $l(\cdot)$ is an incremental error metric that is C^2 in x and integrable in s. For the example in Section VI, we assume $l(\cdot) = \frac{1}{2}(x(s) - x_r(s))^{\mathrm{T}}(x(s) - x_r(s))$ which corresponds to performing a least-squares estimation [14]. The model trajectory is represented by $x(\cdot)$, and the reference trajectory, $x_r(\cdot)$, is a smoothed continuous measurement signal that is C^1 in s.

IV. OPTIMIZATION METHOD

The optimization methods considered in this paper start with first-order iterations (i.e., steepest descent) and then transition to Newton's method. Newton's method, which is second order, converges quadratically [13]. At each iteration, we choose a descent direction $z_k = -[H]^{-1}[DJ(\cdot)]^T$. H is a positive definite matrix and $DJ(\cdot)$ is the gradient defined as follows:

$$DJ(\cdot) = (D_{\tau_1}J(\cdot), ..., D_{\tau_i}J(\cdot), D_{\delta_1}J(\cdot), ..., D_{\delta_i}J(\cdot)).$$

In steepest descent, $H = I$ where I is the identity matrix. In Newton's method, H is the Hessian of the form

$$H = D^2J(\cdot) = \begin{pmatrix} D_\tau^2 J(\cdot) & D_\tau D_\delta J(\cdot) \\ D_\delta D_\tau J(\cdot) & D_\delta^2 J(\cdot) \end{pmatrix}.$$

When $D^2J(\cdot)$ is not positive definite, a quasi-Newton's method, which follows [13], is implemented. The Hessian is decomposed into a matrix containing eigenvalues, λ, and a matrix with corresponding eigenvectors, P. The eigenvalues which are close to zero or negative are replaced with one. Then, the Hessian is reconstructed using the original matrix of eigenvectors, P, and the modified matrix of eigenvalues, λ^*, such that $H = P\lambda^* P^{-1}$. This modification results in using steepest descent in eigenvector directions with negative eigenvalues and Newton's method in eigenvector directions with positive eigenvalues.

After a descent direction is calculated, the largest value of $\epsilon_k \in (0, 1]$ is chosen such that the updated impulse time, $\tau_{i,k+1} = \tau_{i,k} + \epsilon_k z_k$, will maintain the specified ordering of modes in time. The Armijo line search algorithm is then performed to further reduce the step size as needed without changing the impulse times' order [15]. This algorithm uses a backtracking line search to ensure a sufficient decrease for convergence.

V. DERIVATIVES OF COST FUNCTION

In this section, first-order, second-order and cross derivatives of a cost functional, $J(\cdot)$, with respect to impulse times and magnitudes are derived using an adjoint formulation. These derivatives are required in $DJ(\cdot)$ and $D^2J(\cdot)$. The results are stated below and brief sketches of the proofs are provided. The full proofs may be found in [16].

A. First-Order Derivatives of Cost Function

Lemma 1: The first-order derivative of a cost function, $J(\cdot)$, with respect to an impulse time, τ_i, is

$$\begin{aligned} D_{\tau_i}J(\cdot) &= \psi(t_f, \tau_i) \circ X^i + l(x(\tau_i^-), \tau_i^-) - l(x(\tau_i^+), \tau_i^+) \\ X^i &= (f_{i-1}(x(\tau_i^-), \tau_i^-) - f_i(x(\tau_i^+), \tau_i^+)) \end{aligned} \tag{3}$$

where the first-order adjoint operator, $\psi(\cdot)$, is found by integrating the following differential equation backwards along τ from t_f:

$$\begin{aligned} \frac{\partial}{\partial \tau}\psi(t, \tau) &= -D_1 l(x(\tau), \tau) - \psi(t, \tau) \circ D_1 f(x(\tau), \tau) \\ \psi(t, t) &= 0. \end{aligned} \tag{4}$$

Proof: Rewrite (1) in integral form and differentiate the equation with respect to τ_i:

$$\begin{aligned} D_{\tau_i}x(t) =& f_{i-1}(x(\tau_i^-), \tau_i^-) - f_i(x(\tau_i^+), \tau_i^+) \\ &+ \int_{\tau_i^+}^t D_1 f_i(x(s), s) \circ D_{\tau_i}x(s)\ ds. \end{aligned} \tag{5}$$

Rewrite (5) in differential form:

$$\begin{aligned} \frac{\partial}{\partial t}D_{\tau_i}x(t) &= D_1 f_i(x(t), t) \circ D_{\tau_i}x(t) \\ D_{\tau_i}x(\tau_i) &= f_{i-1}(x(\tau_i^-), \tau_i^-) - f_i(x(\tau_i^+), \tau_i^+). \end{aligned} \tag{6}$$

This is a linear differential equation of the form:

$$\begin{aligned} \dot{z}(t) &= (D_1 f_i(x(t), t)) \circ z(t) \\ z(t_0) &= z_0 \end{aligned} \tag{7}$$

where z_0 is the initial condition. Equation (6) can be represented as a state transition matrix operating on an initial condition:

$$D_{\tau_i}x(t) = \Phi(t, \tau_i) \circ D_{\tau_i}x(\tau_i) \tag{8}$$

where $\Phi(\cdot)$ is the state transition matrix for the linearization. When $t \geq \tau_i$, (8) is the result. When $t < \tau_i$, changes in τ_i will not affect $x(t)$ because $x(t)$ is in the past relative to τ_i. Therefore,

$$\begin{aligned} D_{\tau_i}x(t) &= \begin{cases} 0, & t < \tau_i \\ \Phi(t, \tau_i) \circ X^i, & t \geq \tau_i \end{cases} \\ X^i &= (f_{i-1}(x(\tau_i^-), \tau_i^-) - f_i(x(\tau_i^+), \tau_i^+)). \end{aligned} \tag{9}$$

Next, take the derivative of (2) with respect to τ_i and substitute (9) into the equation:

$$\begin{aligned} D_{\tau_i}J(\cdot) &= \int_{\tau_i}^{t_f} D_1 l(x(s), s) \circ D_{\tau_i}x(s)\ ds \\ &\quad + l(x(\tau_i^-), \tau_i^-) - l(x(\tau_i^+), \tau_i^+) \qquad \text{(10a)} \\ &= \int_{\tau_i}^{t_f} D_1 l(x(s), s) \circ \Phi(s, \tau_i) \circ X^i\ ds \\ &\quad + l(x(\tau_i^-), \tau_i^-) - l(x(\tau_i^+), \tau_i^+) \\ &= \int_{\tau_i}^{t_f} D_1 l(x(s), s) \circ \Phi(s, \tau_i)\ ds \circ X^i \\ &\quad + l(x(\tau_i^-), \tau_i^-) - l(x(\tau_i^+), \tau_i^+). \qquad \text{(10b)} \end{aligned}$$

The last two terms in (10a) come from applying Leibniz's rule. The integrand is integrated from τ_i to t_f because $D_{\tau_i}x(t) = 0$ up until $t = \tau_i$ as stated in (9). In (10b), X^i can be taken out of the integral because it does not depend on s. Define

$$\psi(t, \tau) = \int_\tau^t D_1 l(x(s), s) \circ \Phi(s, \tau)\ ds. \tag{11}$$

Take the derivative of (11) with respect to τ and evaluate (11) at τ result in (4). Substitute (11) into (10b):

$$D_{\tau_i}J(\cdot) = \psi(t_f, t) \circ X^i + l(x(\tau_i^-), \tau_i^-) - l(x(\tau_i^+), \tau_i^+).$$

■

Lemma 2: The first-order derivative of a cost function, $J(\cdot)$, with respect to an impulse magnitude, δ_i, is

$$D_{\delta_i} J(\cdot) = \psi(t_f, \tau_i) \circ \Delta^i$$
$$\Delta^i = D_{\delta_i} x(\tau^+). \tag{12}$$

Proof: Rewrite (1) in integral form, and differentiate the equation with respect to δ_i to obtain

$$D_{\delta_i} x(t) = D_{\delta_i} x(\tau_i^+) + \int_{\tau_i^+}^{t} D_1 f_i(x(s), s) \circ D_{\delta_i} x(s)\ ds. \tag{13}$$

Rewrite (13) in differential form:

$$\frac{\partial}{\partial t} D_{\delta_i} x(t) = D_1 f_i(x(t), t) \circ D_{\delta_i} x(t) \tag{14}$$

where the initial condition is $D_{\delta_i} x(\tau_i^+)$. Again, (14) is a linear differential equation in the form of (7). Rewrite (14) as a state transition matrix operating on an initial condition:

$$D_{\delta_i} x(t) = \Phi(t, \tau_i) \circ D_{\delta_i} x(\tau^+).$$

$D_{\delta_i} x(\cdot)$ is found for the case when $t \geq \tau_i$. When $t < \tau_i$, changes in δ_i will not affect $x(t)$ because $x(t)$ is in the past relative to δ_i at τ_i. Hence,

$$D_{\delta_i} x(t) = \begin{cases} 0, & t < \tau_i \\ \Phi(t, \tau_i) \circ \Delta^i, & t \geq \tau_i \end{cases}$$
$$\Delta^i = D_{\delta_i} x(\tau^+). \tag{15}$$

Next, take the derivative of (2) with respect to δ_i, and substitute (11) and (15) into the equation:

$$\begin{aligned} D_{\tau_i} J(\cdot) &= \int_{\tau_i}^{t_f} D_1 l(x(s), s) \circ D_{\delta_i} x(s)\ ds \\ &= \int_{\tau_i}^{t_f} D_1 l(x(s), s) \circ \Phi(s, \tau_i) \circ \Delta^i\ ds \\ &= \int_{\tau_i}^{t_f} D_1 l(x(s), s) \circ \Phi(s, \tau_i)\ ds \circ \Delta^i \\ &= \psi(t_f, \tau_i) \circ \Delta^i. \end{aligned} \tag{16}$$

■

As shown, only a single integration is required to compute the first-order derivatives of a cost function with respect to all impulse times and magnitudes. At this point, impulse times and magnitudes can be estimated using first-order optimization methods such as steepest descent. To implement a second-order method for quadratic convergence, the second-order and cross derivatives of a cost function with respect to impulse times and magnitudes are useful, discussed next.

B. Second-Order Derivatives of Cost Function

Theorem 1: The second-order derivative of a cost function, $J(\cdot)$, with respect to an impulse time, τ_i, is

$$\begin{aligned} &D_{\tau_j} D_{\tau_i} J(\cdot) \\ &\quad = D_1 l(x(\tau_i^-), \tau_i^-) \circ (D_{\tau_j} x(\tau_i^-) - D_{\tau_j} x_r(\tau_i^-)\ \delta_i^j) \\ &\qquad - D_1 l(x(\tau_i^+), \tau_i^+) \circ (D_{\tau_j} x(\tau_i^+) - D_{\tau_j} x_r(\tau_i^+)\ \delta_i^j) \\ &\qquad - D_1 l(x(\tau_i^+), \tau_i^+) \circ X^i\ \delta_i^j + \psi(t_f, \tau_i) \circ X^{i,j} \\ &\qquad + \Omega(t_f, \tau_i) \circ (\Phi(\tau_i, \tau_j) \circ X^j, X^i) \end{aligned}$$

where δ_i^j is the Kronecker delta, and $X^{i,j}$ has the form

$$X^{i,j} = \begin{cases} D_1 f_i(x(\tau_i^+), \tau_i^+) \circ f_i(x(\tau_i^+), \tau_i^+) \\ \quad + D_1 f_{i-1}(x(\tau_i^-), \tau_i^-) \circ f_{i-1}(x(\tau_i^-), \tau_i^-) \\ \quad - 2 D_1 f_i(x(\tau_i^+), \tau_i^+) \circ f_{i-1}(x(\tau_i^-), \tau_i^-) \\ \quad + D_2 f_i(x(\tau_i^+), \tau_i^+) - D_2 f_{i-1}(x(\tau_i^-), \tau_i^-), & i = j \\ (D_1 f_{i-1}(x(\tau_i^-), \tau_i^-) - D_1 f_i(x(\tau_i^+), \tau_i^+)) \circ \\ \quad \Phi(\tau_i, \tau_j) \circ X^j, & i > j. \end{cases}$$

The second-order adjoint operator, $\Omega(\cdot)$, is found by integrating (17) backwards along τ from t_f:

$$\begin{aligned} \frac{\partial}{\partial \tau} \Omega(t, \tau) &= -D_1^2 l(x(\tau), \tau) - \psi(t, \tau) \circ D_1^2 f_i(x(\tau), \tau) \\ &\quad - D_1 f_i(x(\tau), \tau)^T \circ \Omega(t, \tau) - \Omega(t, \tau) \circ D_1 f_i(x(\tau), \tau) \\ \Omega(t, t) &= 0. \end{aligned} \tag{17}$$

Note that different from (9), the derivative of $x(\tau_i)$ with respect to τ_j is

$$D_{\tau_j} x(\tau_i) = \begin{cases} 0, & i < j \\ f(x(\tau_i), \tau_i), & i = j \\ \Phi(\tau_i, \tau_j) \circ X^j, & i > j. \end{cases}$$

When $i = j$, the derivative of $x(\tau_i)$ is taken with respect to its argument τ_i, resulting in $f(x(\tau_i), \tau_i)$.

Proof: Take the derivative of (9) with respect to τ_j and apply fundamental theorem of calculus. The derivation is similar to $D_{\tau_i} x(\cdot)$'s, and thus, only the result is stated:

$$\begin{aligned} &D_{\tau_j} D_{\tau_i} x(t) \\ &\qquad = \Phi(t, \tau_i) \circ X^{i,j} + \phi(t, \tau_i) \circ (\Phi(\tau_i, \tau_j) \circ X^j, X^i) \\ &\phi(t, \tau) = \int_{\tau_i}^{t} \Phi(t, s) D_1^2 f_i(x(s), s) \circ (\Phi(s, \tau), \Phi(s, \tau))\ ds. \end{aligned} \tag{18}$$

Next, take the derivative of (10a) with respect to τ_j, and substitute (9) and (18) into the equation:

$$\begin{aligned} &D_{\tau_j} D_{\tau_i} J(\cdot) \\ &\quad = \frac{\partial}{\partial \tau_j} \Big(\int_{\tau_i^+}^{t_f} D_1 l(x(s), s) \circ D_{\tau_i} x(s)\ ds \\ &\qquad + l(x(\tau_i^-), \tau_i^-) - l(x(\tau_i^+), \tau_i^+) \Big) \\ &\quad = D_1 l(x(\tau_i^-), \tau_i^-) \circ (D_{\tau_j} x(\tau_i^-) - D_{\tau_j} x_r(\tau_i^-)\ \delta_i^j) \\ &\qquad - D_1 l(x(\tau_i^+), \tau_i^+) \circ (D_{\tau_j} x(\tau_i^+) - D_{\tau_j} x_r(\tau_i^+)\ \delta_i^j) \\ &\qquad - D_1 l(x(\tau_i^+), \tau_i^+) \circ D_{\tau_j} x(\tau_i^+)\ \delta_i^j \\ &\qquad + \int_{\tau_i^+}^{t_f} D_1 l(x(s), s) \circ D_{\tau_j} D_{\tau_i} x(s) \\ &\qquad\quad + D_1^2 l(x(s), s) \circ (D_{\tau_j} x(s), D_{\tau_i} x(s))\ ds \\ &\quad = D_1 l(x(\tau_i^-), \tau_i^-) \circ (D_{\tau_j} x(\tau_i^-) - D_{\tau_j} x_r(\tau_i^-)\ \delta_i^j) \\ &\qquad - D_1 l(x(\tau_i^+), \tau_i^+) \circ (D_{\tau_j} x(\tau_i^+) - D_{\tau_j} x_r(\tau_i^+)\ \delta_i^j) \\ &\qquad - D_1 l(x(\tau_i^+), \tau_i^+) \circ D_{\tau_j} x(\tau_i^+)\ \delta_i^j \end{aligned}$$

$$
\begin{aligned}
&+\int_{\tau_i^+}^{t_f} D_1 l(x(s),s)\circ\Phi(s,\tau_i)\circ X^{i,j}\ ds\\
&+\int_{\tau_i^+}^{t_f} D_1 l(x(s),s)\circ\phi(s,\tau_i)\circ(\Phi(\tau_i,\tau_j)\circ X^j, X^i)\\
&\quad+D_1^2 l(x(s),s)\circ(\Phi(s,\tau_i)\circ\Phi(\tau_i,\tau_j)\circ X^j,\\
&\qquad\Phi(s,\tau_i)\circ X^i)\ ds\\
&=D_1 l(x(\tau_i^-),\tau_i^-)\circ(D_{\tau_j}x(\tau_i^-)-D_{\tau_j}x_r(\tau_i^-)\ \delta_i^j)\\
&\quad-D_1 l(x(\tau_i^+),\tau_i^+)\circ(D_{\tau_j}x(\tau_i^+)-D_{\tau_j}x_r(\tau_i^+)\ \delta_i^j)\\
&\quad-D_1 l(x(\tau_i^+),\tau_i^+)\circ X^i\ \delta_i^j+\psi(t_f,\tau_i)\circ X^{i,j}\\
&\quad+\Omega(t_f,\tau_i)\circ(\Phi(\tau_i,\tau_j)\circ X^j, X^i) \qquad (19)
\end{aligned}
$$

where $\Omega(\cdot)$ is defined as

$$
\begin{aligned}
\Omega(t,\tau)=\int_\tau^t & D_1 l(x(s),s)\circ\phi(s,\tau))\\
&+D_1^2 l(x(s),s)\circ(\Phi(s,\tau),\Phi(s,\tau))\ ds. \qquad (20)
\end{aligned}
$$

Take the derivative of (20) with respect to τ and evaluate (20) at τ result in (17).

First two terms in (19) are the derivatives of $l(x(\tau_i^-),\tau_i^-)$ and $l(x(\tau_i^+),\tau_i^+)$ from (10a) with respect to τ_i. Because $l(x(\tau_i^-),\tau_i^-)$ and $l(x(\tau_i^+),\tau_i^+)$ are evaluated at τ_i^+ and τ_i^-, the reference trajectory, $x_r(\cdot)$, and the model trajectory, $x(\cdot)$, explicitly depend on τ_i. Therefore, both the derivatives of $x_r(\cdot)$ and $x(\cdot)$ present when taking the derivatives of $l(x(\tau_i^-),\tau_i^-)$ and $l(x(\tau_i^+),\tau_i^+)$ with respect to τ_i. ∎

Theorem 2: The second-order derivative of a cost function, $J(\cdot)$, with respect to an impulse magnitude, δ_i, is

$$
\begin{aligned}
D_{\delta_j}D_{\delta_i}J(\cdot)&=\psi(t_f,\tau_i)\circ\Delta^{i,j}+\Phi(\tau_i,\tau_j)^T\circ\Omega(t_f,\tau_i)\\
\Delta^{i,j}&=D_{\delta_j}D_{\delta_i}x(\tau_i^+).
\end{aligned}
$$

Proof: Take the derivative of (15) with respect to δ_j and apply fundamental theorem of calculus. The derivation is omitted because it is straightforward, and the result is

$$
D_{\delta_j}D_{\delta_i}x(t)=\Phi(t,\tau_i)\circ\Delta^{i,j}+\phi(t,\tau_i)\circ(\Phi(\tau_i,\tau_j)\circ\Delta^j,\Delta^i). \qquad (21)
$$

Take the derivative of (16) with respect to δ_j and substitute (15) and (21) into the equation:

$$
\begin{aligned}
&D_{\delta_j}D_{\delta_i}J(\cdot)\\
&=\frac{\partial}{\partial\delta_j}\left(\int_{\tau_i}^{t_f} D_1 l(x(s),s)\circ D_{\delta_i}x(s)\ ds\right)\\
&=\int_{\tau_i}^{t_f} D_1 l(x(s),s)\circ D_{\delta_j}D_{\delta_i}x(s)\\
&\quad+D_1^2 l(x(s),s)\circ(D_{\delta_j}x(s),D_{\delta_i}x(s))\ ds\\
&=\int_{\tau_i}^{t_f} D_1 l(x(s),s)\circ\Phi(s,\tau_i)\circ\Delta^{i,j}\ ds\\
&\quad+\int_{\tau_i}^{t_f} D_1 l(x(s),s)\circ\phi(s,\tau_i)\circ(\Phi(\tau_i,\tau_j)\circ\Delta^j,\Delta^i)\\
&\qquad+D_1^2 l(x(s),s)\circ(\Phi(s,\tau_i)\circ\Phi(\tau_i,\tau_j)\circ\Delta^j,\\
&\qquad\Phi(s,\tau_i)\circ\Delta^i)\ ds\\
&=\psi(t_f,\tau_i)\circ\Delta^{i,j}+\Omega(t_f,\tau_i)\circ(\Phi(\tau_i,\tau_j)\circ\Delta^j,\Delta^i).
\end{aligned}
$$

∎

C. Cross Derivatives of Cost Function

Theorem 3: The cross derivative of $J(\cdot)$ with respect to an impulse time and an impulse magnitude is

$$
D_{\delta_j}D_{\tau_i}J(\cdot)=\begin{cases}\Omega(t_f,\tau_j)\circ(\Delta^j,\Phi(\tau_j,\tau_i)\circ X^i), & \tau_i<\tau_j\\ D_1 l(x(\tau_i^-),\tau_i^-)\circ D_{\delta_j}x(\tau_i^-)\\ \ -D_1 l(x(\tau_i^+),\tau_i^+)\circ D_{\delta_j}x(\tau_i^+)\\ \ +\psi(t_f,\tau_i)\circ\Delta X^{i,j}\\ \ +\Omega(t_f,\tau_i)\circ(\Phi(\tau_i,\tau_j)\circ\Delta^j,X^i), & \tau_i\geq\tau_j\end{cases}
$$

$$
\Delta X^{i,j}=\begin{cases}0, & \tau_i<\tau_j\\ -D_1 f_i(x(\tau_i^+),\tau_i^+)\circ\Delta^j, & \tau_i=\tau_j\\ (D_1 f_{i-1}(x(\tau_i^-),\tau_i^-)\\ \ -D_1 f_i(x(\tau_i^+),\tau_i^+))\circ\Phi(\tau_i,\tau_j)\circ\Delta^j, & \tau_i>\tau_j.\end{cases}
$$

Proof: Take the derivative of (5) with respect to δ_j:

$$
\begin{aligned}
&D_{\delta_j}D_{\tau_i}x(t)\\
&\quad=\begin{cases}\Phi(t,\tau_j)\circ\Delta X^{i,j}\\ \ +\phi(t,\tau_j)\circ(\Delta^j,\Phi(\tau_j,\tau_i)\circ X^i), & \tau_i<\tau_j\\ \Phi(t,\tau_i)\circ\Delta X^{i,j}\\ \ +\phi(t,\tau_i)\circ(\Phi(\tau_i,\tau_j)\circ\Delta^j,X^i), & \tau_i\geq\tau_j.\end{cases}
\end{aligned} \qquad (22)
$$

Take the derivative of (10a) with respect to δ_j, and substitute (9), (15), and (22) into the equation:

$$
\begin{aligned}
&D_{\delta_j}D_{\tau_i}J(\cdot)\\
&=\frac{\partial}{\partial\delta_j}(\int_{\tau_i^+}^{t_f} D_1 l(x(s),s)\circ D_{\tau_i}x(s)\ ds\\
&\qquad+l(x(\tau_i^-),\tau_i^-)-l(x(\tau_i^+),\tau_i^+))\\
&=D_1 l(x(\tau_i^-),\tau_i^-)\circ D_{\delta_j}x(\tau_i^-)\\
&\quad-D_1 l(x(\tau_i^+),\tau_i^+)\circ D_{\delta_j}x(\tau_i^+)\\
&\quad+\int_{\tau_i^+}^{t_f} D_1 l(x(s),s)\circ D_{\delta_j}D_{\tau_i}x(s)\\
&\qquad+D_1^2 l(x(s),s)\circ(D_{\delta_j}x(s),D_{\tau_i}x(s))\ ds\\
&=D_1 l(x(\tau_i^-),\tau_i^-)\circ D_{\delta_j}x(\tau_i^-)\\
&\quad-D_1 l(x(\tau_i^+),\tau_i^+)\circ D_{\delta_j}x(\tau_i^+)\\
&\quad+\int_{\tau_i^+}^{t_f} D_1 l(x(s),s)\circ\Phi(s,\tau_i)\circ\Delta X^{i,j}\ ds\\
&\quad+\int_{\tau_i^+}^{t_f} D_1 l(x(s),s)\circ\phi(s,\tau_i)\circ(\Phi(\tau_i,\tau_j)\circ\Delta^j,X^i))\\
&\qquad+D_1^2 l(x(s),s)\circ(\Phi(s,\tau_i)\circ\Phi(\tau_i,\tau_j)\circ\Delta^j,\\
&\qquad\Phi(s,\tau_i)\circ X^i)\ ds\\
&=D_1 l(x(\tau_i^-),\tau_i^-)\circ D_{\delta_j}x(\tau_i^-)\circ\partial\delta_j\\
&\quad-D_1 l(x(\tau_i^+),\tau_i^+)\circ D_{\delta_j}x(\tau_i^+)\circ\partial\delta_j\\
&\quad+\psi(t_f,\tau_i)\circ\Delta X^{i,j}+\Omega(t_f,\tau_i)\circ(\Phi(\tau_i,\tau_j)\circ\Delta^j,X^i).
\end{aligned}
$$

This is the proof when $\tau_i\geq\tau_j$. Follow similar steps for the case when $\tau_i<\tau_j$.

Note that the full form of $\Delta X^{i,j}$ is

$$
\begin{aligned}
\Delta X^{i,j}=&D_1 f_{i-1}(x(\tau_i^-),\tau_i^-)\circ D_{\delta_j}x(\tau_i^-)\\
&-D_1 f_i(x(\tau_i^+),\tau_i^+)\circ D_{\delta_j}x(\tau_i^+).
\end{aligned}
$$

When $\tau_i < \tau_j$, both $D_{\delta_j} x(\tau_i^+) = 0$ and $D_{\delta_j} x(\tau_i^-) = 0$ because $x(\cdot)$ at time τ_i does not depend on δ_j at time τ_j (a future time relative to τ_i). When $\tau_i = \tau_j$, only $D_{\delta_j} x(\tau_i^-) = 0$ because $x(\cdot)$ right before τ_i does not depend on δ_j. Considering different cases of $D_{\delta_j} x(\tau_i^-)$ and $D_{\delta_j} x(\tau_i^+)$, each case of $D_{\delta_j} D_{\tau_i} J(\cdot)$ and $\Delta X^{i,j}$ is derived. ■

To compute the second-order derivatives of a cost function with respect to any number of impulse times and magnitudes, only two integrations (for $\Phi(\cdot)$ and $\Omega(\cdot)$) are required in addition to an integration for $\psi(\cdot)$ which is computed when calculating the gradient of the cost function.

As shown, all elements of the gradient and the Hessian of the cost function are computed independent of the total number of impulses. Accordingly, this optimization method has the benefit of zero increase in the computational complexity when the number of impulses are increased.

VI. EXAMPLE

In this section, the second-order impulsive hybrid system optimization is implemented to localize geometrical surface features on a surface traced by a three-revolute-joint robotic finger (Fig. 1). This surface has three segments whereby each segment corresponds to a dynamical mode, f_i. The trajectory, $x(t)$, includes joint angles $(\theta_1(t), \theta_2(t), \theta_3(t))$ and joint angular velocities $(\omega_1(t), \omega_2(t), \omega_3(t))$. Impulses are observed in the velocity trajectories when the finger reaches the boundary of each surface segment. The hybrid optimization was implemented to localize the surface segments by identifying the transition times and impulse magnitudes assuming that the segment sequence and the total number of segments are known.

Two types of measurement trajectories were considered: without noise and with noise (Fig. 2). To generate a noisy trajectory, a measurement trajectory is simulated and sampled at 0.001 second intervals. Random noise sampled from a Gaussian distribution with mean, $\mu = 0$, and standard deviation, σ, is added at each time step. Then, the noisy trajectory is smoothed using a Gaussian filter (with standard deviation, $\sigma_G = 5$) before a spline is used to generate the continuous measurement trajectory required by the optimization formulation.

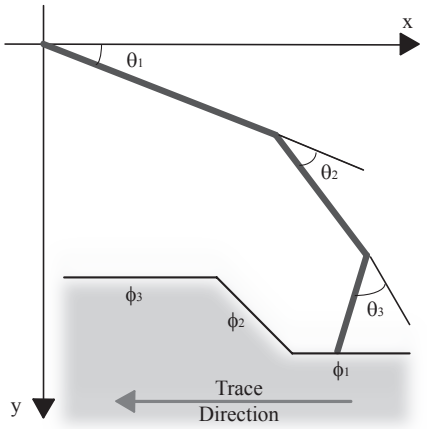

Fig. 1. Configuration of the robotic finger model used in simulation.

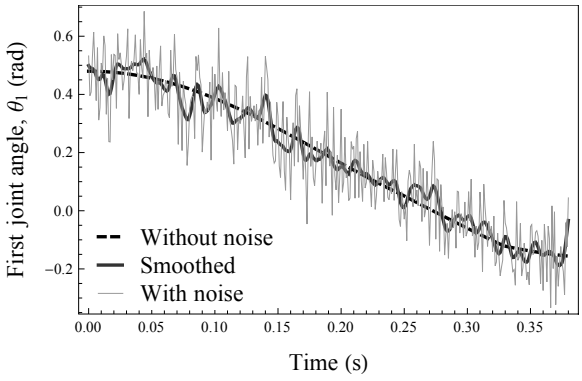

Fig. 2. Comparison of a measurement trajectory without noise, a measurement trajectory with noise ($\sigma = 0.1$ rad $(5.73°)$), and a smoothed measurement trajectory with noise for the first joint.

Both type of simulated measurement trajectories have impulse times, $\tau = (0.247, 0.329)$, impulse magnitudes $\delta_1 = (0.249, -0.058, 4.074)$ (in the velocity) at the first impulse time, and impulse magnitudes $\delta_2 = (-1.100, 0.102, 0.378)$ at second impulse time. The total time horizon is from $\tau_1 = 0$ to $\tau_4 = t_f = 0.380$ seconds. Both algorithms for measurement trajectories with noise and without noise are initialized to $\tau = (0.270, 0.350)$, $\delta_1 = (0.230, -0.030, 4.070)$, and $\delta_2 = (-0.900, 0.200, 0.410)$. The optimization algorithm is terminated when the norm of the gradient, $||DJ(\cdot)|| < 10^{-5}$.

A. Convergence

The optimization algorithm took sixteen iterations to converge when using measurement trajectories without noise. The convergence plots for the logarithm of the norm of cost gradient (Fig. 3) show a quadratic convergence when the Newton's method was applied after ten initial first-order iterations. Fig. 3 also shows that varying the noise level did not drastically change the total number of iterations for convergence. The average number of iterations taken is seventeen when σ of the added noise is 0.1 rad $(5.73°)$.

B. Measurement Noise

The effect of measurement noise on the standard deviation of the estimates was investigated using the Monte Carlo method. For each noise level, thirty random simulations are computed. The noise level is quantified by the standard deviation of the noise added, σ. The largest σ considered is 0.1 rad $(5.73°)$. A noise level above $\sigma = 0.1$ rad results in frequent instability during optimization.

The optimization algorithm shows a linear increase in the standard deviation of estimates when the standard deviation of noise is increased (Fig. 4). Note that in Fig. 4, the trend lines go through the origin, and thus reflect that the estimates are unbiased. Also, note that the estimates at the second impulse time are more sensitive to the noise than the estimate at the first impulse time. This is expected

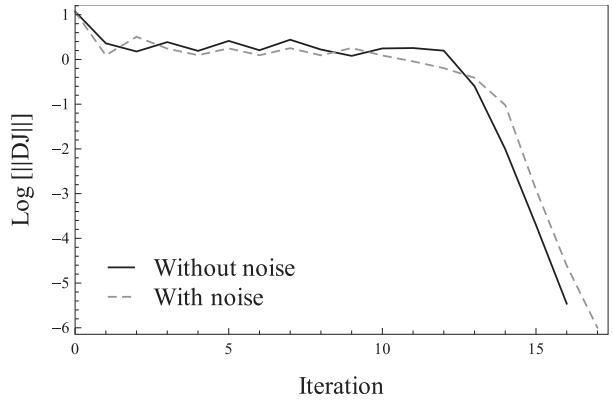

Fig. 3. Convergence plots for the logarithm of the norm of cost gradient of a finger model when the measurement trajectories are (a) without noise and (b) with noise. First ten iterations use steepest descent, and the remaining iterations use Newton's method.

because the derivatives in the gradient and the Hessian are calculated using backward integration. The estimate at the second impulse time includes fewer points, and thus are more prone to be affected by outliers in the noise. On the other hand, the estimate at the first impulse time include more points, and hence the effect of outliers in the noise is reduced.

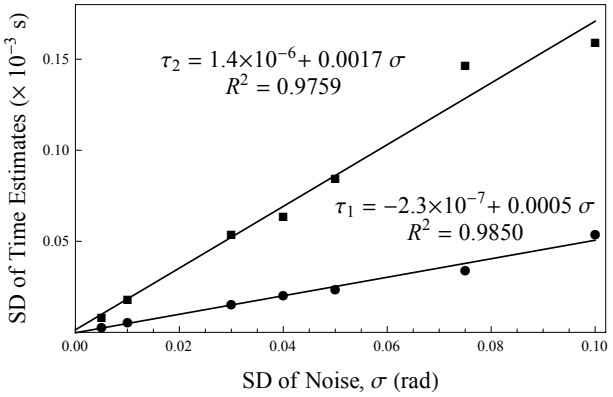

Fig. 4. Effect of the standard deviation (SD) of measurement noise on the SD of the estimated impulse times.

VII. CONCLUSION

This paper introduces a second-order adjoint-based impulsive hybrid system optimization to optimize the transition times and impulse magnitudes of an impulsive hybrid system. This is an extension of the results in [7] to a different class of impulsive systems. The adjoint formulation allows the efficient computation of the first-order, second-order and cross derivatives of a cost functional with respect to impulse times and impulse magnitudes as shown in Section V. In Section VI, an application of this optimization technique in surface feature localization shows that this technique is effective even when measurement noise is present.

In this paper, the number of impulses and the dynamical mode sequence are assumed to be known a priori. When this assumption does not hold, a relaxation of the hybrid optimization, a similar technique to the one in [9], can be implemented to compute the number of impulses and mode sequence before performing the optimization. Another approach that does not rely on relaxation is to extend the result in [10] to allow for impulses in the system. This is an area of future research.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under award IIS-1018167. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] T. M. Caldwell and T. D. Murphey, "An adjoint method for second-order switching time optimization," in *IEEE Int. Conf. on Decision and Control (CDC)*, 2010, pp. 2155 –2162.

[2] E. Johnson and T. D. Murphey, "Second-order switching time optimization for nonlinear time-varying dynamic systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 8, pp. 1953–1957, 2011.

[3] M. Egerstedt, Y. Wardi, and F. Delmotte, "Optimal control of switching times in switched dynamical systems," in *IEEE Int. Conf. on Decision and Control (CDC)*, 2003, pp. 2138 – 2143.

[4] M. Egerstedt, S. Azuma, and H. Axelsson, "Transition-time optimization for switched-mode dynamical systems," *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 110–115, 2006.

[5] X. Xu and P. J. Antsaklis, "Optimal control of switched autonomous systems," in *IEEE Int. Conf. on Decision and Control (CDC)*, 2002, pp. 4401 – 4406.

[6] M. Travers, L. Pao, and T. D. Murphey, "Data association with impulse optimization," in *IEEE Int. Conf. on Decision and Control (CDC)*, 2010, pp. 2204–2209.

[7] M. Travers, T. D. Murphey, and L. Pao, "Impulsive data association with an unknown number of targets," in *Hybrid Systems: Computation and Control*, 2011, pp. 261–270.

[8] X. Xu and P. J. Antsaklis, "Optimal control of hybrid autonomous systems with state jumps," in *American Controls Conf. (ACC)*, vol. 6, June 2003, pp. 5191 – 5196.

[9] T. Caldwell and T. Murphey, "Switching mode generation and optimal estimation with application to skid-steering," *Automatica*, vol. 47, no. 1, pp. 50 – 64, 2011.

[10] T. Caldwell and T. D. Murphey, "Projection-based switching time optimization," in *American Controls Conf. (ACC)*, 2012.

[11] M. Alamir and S. A. Attia, "On solving optimal control problems for switched hybrid nonlinear systems by strong variations algorithms," in *6th IFAC Symposium on Nonlinear Control Systems*, 2004, pp. 558–563.

[12] S. A. Attia, M. Alamir, and C. C. de Wit, "On solving optimal control problems for switched hybrid nonlinear systems by strong variations algorithms," in *16th IFAC World Congress*, 2005, pp. 558–563.

[13] C. Kelley, *Iterative Methods for Optimization*. Society for Industrial Mathematics, 1994.

[14] Y. Bar-shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. Wiley-Interscience, 2001.

[15] L. Armijo, "Minimization of functions having lipschitz continuous first partial derivatives," *Pacific Journal of Mathematics*, vol. 16, no. 1, pp. 1–3, 1966.

[16] Y. P. Leong, "Surface feature detection based on proprioception of a robotic finger during haptic exploration," M. S. Thesis, Northwestern University, Jun 2012.