Single integration optimization of linear time-varying switched systems

T. M. Caldwell and T. D. Murphey

Abstract— **This paper considers the switching time optimization of time-varying linear switched systems subject to quadratic cost—also potentially time-varying. The problem is formulated so that only a single set of differential equations need to be solved prior to optimization. Once these differential equations have been solved, the cost may be minimized over arbitrary number of modes and mode sequences without requiring additional simulation. The number of matrix multiplications needed to compute the gradient grows linearly with respect to the number of switching times, resulting in fast execution even for high dimensional optimizations. Lastly, the differential equations that need to be simulated are as smooth as the system's vector fields, despite the fact that the optimization itself is nonsmooth. Examples illustrate the technique and its efficiency, including a comparison with other standard techniques.**

I. Introduction

This paper considers the linear quadratic switching time optimization (LQSTO) problem—the problem of optimizing a quadratic cost subject to the constraint of linear switching dynamics where the mode order is known ahead of time—and formulates an efficient way of computing optimal switching times. The primary result of this paper is showing that a single set of differential equations may be solved prior to optimization so that during optimization (e.g., during computation of the gradient) no differential equations need to be solved. This is in contrast to the optimization approaches in [6], [7], [14], [15], which do not make full use of the linearity of the switched system and therefore solve a differential equation at every step of the descent algorithm.

Others have investigated the specific case of linear switching time control. Giua et al. [6], [7] present an optimal control law that finds regions of the state space where switches should occur. Their approach assumes the linear modes are time-invariant as well as stable, neither of which are requirements in the present work. Xu and Antsaklis [14], [15] derive non-linear, switching time gradient and Hessian calculations of the cost for steepest descent and Newton's method. The iterative descent-based approach as introduced in Xu and Antsaklis' work form the basis for the work in this paper. Egerstedt et al. [5] propose a computationally more efficient adjoint-based gradient calculation, which only requires solving two differential equations: the state and co-state. In [3], [10], [11] the adjoint calculation for the Hessian was identified, which also requires solving a constant number of differential equations.

Xu and Antsaklis [14], [15] also consider the time-invariant version of the LQSTO problem. Analogous to the linear quadratic regulator problem of optimal controls (see [1]), Xu and Antsaklis found that the gradient depends on the state and a differential equation with solution specifying the linear relationship between the state and co-state. By only considering time-invariant switched systems, the evolution of the state is dictated by the matrix exponential, which allows one to avoid solving the state differential equation numerically. However, under Xu and Antsaklis' formulation, the linear relation of the state and co-state must still be numerically solved for at every step of a descent algorithm.

We present a formulation with two important differences. We refer to our formulation as *algebraic*[1] *linear quadratic switching time optimization* (ALQSTO). First, ALQSTO makes no assumption about the time-variance of the modes. Therefore, ALQSTO does not exclude many important linear systems. For example, power systems are often linear time-varying [8], [12]. Also, the linearization about a trajectory of a non-linear system, in general, is time-varying. Second, ALQSTO generates a single set of differential equations to be solved such that these differential equations are independent of the switching times as well as the assumed mode sequence. Therefore, these differential equations must be solved only once and off-line from the optimization. By doing so, the state and Riccati equation may be computed algebraically from the solutions of these differential equations and moreover, the optimality equation (i.e., $\nabla J(\mathcal{T}) = 0$) does not require any additional differential equations to be solved. For this reason, the optimization complexity does not depend on the differential equations—they are only computed once.

This paper is organized as follows:[2] Section II reviews switched systems. Section III, presents optimality conditions for linear switched systems with quadratic cost. The algebraic linear quadratic switching time optimization (ALQSTO) problem is proposed in Section IV. Differential equations are derived such that the state and a Riccati relation may be computed from solutions to these differential

This material is based upon work supported by the National Science Foundation under award CCF-0907869 as well as the Department of Energy (DOE) Office of Science Graduate Fellowship Program (DOE SCGF). The DOE SCGF Program was made possible in part by the American Recovery and Reinvestment Act of 2009. The DOE SCGF program is administered by ORISE (which is managed by ORAU under DOE contract number DE-AC05-06OR23100). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF, DOE, ORAU, or ORISE.

T. M. Caldwell and T. D. Murphey are with the Department of Mechanical Engineering, Northwestern University, 2145 Sheridan Road Evanston, IL 60208, USA E-mail: caldwelt@u.northwestern.edu ; t-murphey@northwestern.edu

[1]We will use the term *algebraic* to refer to optimization procedures that only need addition and multiplication to compute descent directions.

[2]This paper has been submitted to the IEEE journal, Transactions on Automatic Control. That version includes a discussion on reference tracking.

equations for any choice of switching times. Section V shows how to apply the presented formulation to solve the LQSTO problem using steepest-based methods.

II. Review of Switched Systems

Switched systems are a class of hybrid systems that evolve over a sequence of continuous modes of operation, where the modes transition in a discrete manner. The times for which the transitions occur are referred to as switching times. We denote the set of strictly monotonically increasing switching times as $\mathcal{T} = \{T_1, T_2, \ldots, T_{N-1}\}$, where N is the number of modes.

In this paper, the mode sequence is assumed to be known, and therefore, the optimization is only over the switching times. Mathematically, a switched system may be defined as [3], [5], [13], [15]:

$$\dot{x}(t) = \overline{f}\Big(x(t), \mathcal{T}, t\Big) = \begin{cases} f_1\Big(x(t), t\Big) & , \quad T_0 \le t < T_1 \\ f_2\Big(x(t), t\Big) & , \quad T_1 \le t < T_2 \\ \quad \vdots & \quad \vdots \quad \quad \vdots \\ f_N\Big(x(t), t\Big) & , \quad T_{N-1} \le t < T_N \end{cases}$$
$$\text{subject to: } x(T_0) = x_0 \tag{1}$$

where T_0 is the initial time, T_N is the final time, x_0 is the initial state, and $f_i : \mathbb{R}^n \times \mathbb{R} \mapsto \mathbb{R}^n$ is the vector field corresponding to the i^{th} mode in the assumed mode sequence.

III. Optimality Conditions for Linear Switched Systems with Quadratic Cost

According to [3], [5], the optimal switching times of the cost functional,

$$J(\mathcal{T}) = \int_{T_0}^{T_N} \ell\Big(\tau, x(\tau)\Big)\, \mathrm{d}\tau + m\Big(x(T_N)\Big),$$

must satisfy the following equations:[3]

$$\dot{x}(t) = \overline{f}\Big(t, x(t), \mathcal{T}\Big), \quad \text{s.t.: } x(T_0) = x_0 \tag{2a}$$

$$\dot{\rho}(t) = -D_2\overline{f}\Big(t, x(t), \mathcal{T}\Big)^T \rho(t) - D_2\ell\Big(t, x(t)\Big)^T, \quad \text{s.t.: } \rho(T_N) = Dm\Big(x(T_N)\Big)^T \tag{2b}$$

$$0 = \rho^T(T_i)\Big[f_i\Big(T_i, x(T_i)\Big) - f_{i+1}\Big(T_i, x(T_i)\Big)\Big], \quad \forall i \in (1, \ldots, N-1). \tag{2c}$$

The evolution of the state is given by Eq (2a) and the evolution of the co-state is given by Eq (2b). Then, given some set of optimal switching times, the corresponding optimal state and co-state must satisfy Eq (2c). This set of equations forms a necessary condition for optimality. We refer to Eq (2c) as the set of optimality equations. Also, note that the right hand side of the optimality equations is the gradient of the cost, $\nabla J(\mathcal{T})$.

For this paper, we restrict our focus to linear time-varying switched systems,

$$\overline{f}\Big(t, x(t), \mathcal{T}\Big) = \overline{A}(t, \mathcal{T})x(t)$$

where $\overline{A}(t, \mathcal{T}) = A_i(t)$ for $T_{i-1} \le t < T_i$. Moreover, we choose a quadratic cost function such that

$$\begin{aligned} &\ell(t, x(t)) := \tfrac{1}{2}\, x(t)^T Q(t) x(t) \text{ and} \\ &m(x(T_N)) := \tfrac{1}{2}\, x(T_N)^T P_1 x(T_N). \end{aligned} \tag{3}$$

where Q and P_1 are symmetric positive semi-definite. Then, the necessary condition, Eq (2), becomes

$$\dot{x}(t) = \overline{A}(t, \mathcal{T})x(t), \quad \text{s.t.: } x(T_0) = x_0 \tag{4a}$$

$$\dot{\rho}(t) = -\overline{A}(t, \mathcal{T})^T \rho(t) - Q(t)x(t), \quad \text{s.t.: } \rho(T_N) = P_1 x(T_N). \tag{4b}$$

$$0 = \rho^T(T_i)[A_i(T_i) - A_{i+1}(T_i)]x(T_i), \quad \forall i \in (1, \ldots, N-1). \tag{4c}$$

Notice that Eqs (4a) and (4b) make up a two-point boundary value problem. As in the LQR from classical controls, we will find a linear mapping between x and ρ in order to transform the problem into an initial value problem.

A. The Linear Switched System Analog to the Riccati Equation from LQR

In the following theorem, the state, $x(t)$, and co-state, $\rho(t)$, are shown to be linearly dependent through a Riccati relation[4].

Theorem 1: For optimizing a quadratic cost function described by Eq (3) and subject to a linear time-varying switched system of the form in Eq. (4a), the state, $x(t)$, and co-state, $\rho(t)$, are linearly dependent:

$$\rho(t) = P(t)x(t),$$

where $P(t)$ is the solution to the following linear differential equation:

$$\dot{P}(t) = -\overline{A}(t, \mathcal{T})^T P(t) - P(t)\overline{A}(t, \mathcal{T}) - Q(t), \quad \text{s.t.: } P(T_N) = P_1. \tag{5}$$

Proof: The proof follows from manipulation of $x(t)$ and $\rho(t)$ using the state-transition matrix. Begin with the solution to $\dot{x}$ (see Eq (4a)): $x(t) = \overline{\Phi}(t, T_0)x_0$, where $\overline{\Phi}(t, T_0)$ is the state-transition matrix corresponding to $\overline{A}(t, \mathcal{T})$ (i.e. $\frac{d}{dt}\overline{\Phi}(t, T_0) = \overline{A}(t, \mathcal{T}) \cdot \overline{\Phi}(t, T_0)$). Furthermore, the integral form of the adjoint equation, $\rho(t)$, is

$$\rho(t) = \overline{\Phi}(T_N, t)^T P_1 x(T_N) + \int_t^{T_N} \overline{\Phi}(\tau, t)^T Q(\tau) x(\tau)\, \mathrm{d}\tau.$$

[3]The notation D is the slot derivative. For a function g, $Dg(\cdot)$ is the partial derivative of g with respect to its single argument. Similarly, $D_i g(\cdot, \cdot)$ is the partial of g with respect to the i^{th} argument.

[4]We use the term, Riccati relation, as it is used in optimal control theory—as the variable specifying the linear mapping between the state and the co-state. Here, the Riccati relation is the solution to a linear differential equation and should not be confused with the 2nd-order differential equations studied by the mathematician Jacopo Riccati.

Plugging in for $x(t)$,

$$= \overline{\Phi}(T_N,t)^T P_1 \overline{\Phi}(T_N,T_0)x_0 + \int_t^{T_N} \overline{\Phi}(\tau,t)^T Q(\tau)\overline{\Phi}(\tau,T_0)x_0 \mathrm{d}\tau.$$

Noting $\overline{\Phi}(T_N,T_0) = \overline{\Phi}(T_N,t)\overline{\Phi}(t,T_0)$,

$$= \overline{\Phi}(T_N,t)^T P_1 \overline{\Phi}(T_N,t)\overline{\Phi}(t,T_0)x_0 + \int_t^{T_N} \overline{\Phi}(\tau,t)^T Q(\tau)\overline{\Phi}(\tau,t)\overline{\Phi}(t,T_0)x_0 \,\mathrm{d}\tau.$$

Pulling $x(t) = \overline{\Phi}(t,T_0)x_0$ to the right and denoting the matrix integral equation as $P(t)$

$$= \underbrace{\Big[\overline{\Phi}(T_N,t)^T P_1 \overline{\Phi}(T_N,t) + \int_t^{T_N} \overline{\Phi}(\tau,t)^T Q(\tau)\overline{\Phi}(\tau,t)\,\mathrm{d}\tau\Big]}_{P(t)} x(t).$$

The differential form of $P(t)$ is Eq (5). Furthermore, $x(t)$ and $\rho(t)$ are linearly dependent through $P(t)$. ■

B. Linear Quadratic Switched System Optimality Condition

Using the linear dependence of $\rho(t)$ and $x(t)$, the optimality equations become:

$$0 = x(T_i)^T P(T_i)[A_i(T_i) - A_{i+1}(T_i)]x(T_i), \quad \forall i \in (1,\ldots,N-1). \tag{6}$$

The optimality equations provides $N-1$ equations and $N-1$ switching time unknowns such that the problem of finding the optimal switching times may be formulated as a root-finding problem where $x(T_i)$ and $P(T_i)$ may be calculated separately (i.e. two separate initial value problems). Eq (6), along with Eqs (4a) and (5) together make up the necessary condition for optimality for the linear quadratic switching time optimization problem.

IV. Algebraic Linear Quadratic Switching Time Optimization (ALQSTO)

As seen in the previous section, LQ switching time optimization is a root-finding problem that depends on the solutions to differential equations. Suppose the problem may be reformulated such that all of the differential equations may be solved *off-line* from an optimization routine.

Consider the optimization problem characterized by the quadratic cost function described by Eq (3) and subject to the linear switched system of Eq (4a). We hypothesize that an equivalent optimization problem exists that depends on the solutions to a set of differential equations, where once the solutions have been obtained, no additional integration is necessary in order to solve the optimization problem using standard techniques (e.g. steepest descent, Newton's method, etc). These differential equations are switching time and mode sequence invariant and because they need only be solved once, we shall refer to them as the **single integration differential equations (SIDE)**. Furthermore, we shall refer to this equivalent optimization problem as the **Algebraic Linear Quadratic Switching Time Optimization (ALQSTO)** problem for it relies solely on addition and multiplication once the SIDE have been solved for.

In order for the hypothesis to be true, the state, $x(t)$, and Riccati relation, $P(t)$, (i.e. the solutions to Eq (4a) and Eq (5)) must be computable from the solutions to the SIDE for any arbitrary, valid set of switching times. Furthermore, this computation must not require solving any additional differential equations. Then, the optimality equations, Eq (6), may also be computed without solving additional differential equations. The rest of this section is devoted to finding the SIDE as well as the associated computations of $x(t)$ and $P(t)$.

A. Computing $x(t)$

As stated, the SIDE for computing $x(t)$ are the state-transition matrices for each $A_i(t)$. Using $\Phi^i(\cdot,\cdot)$ to denote the STM corresponding to A_i, the value of x at the i^{th} switching time is

$$x(T_i) = \overline{\Phi}(T_i,T_0)x_0 = \Big[\prod_{j=i}^{1} \Phi^j(T_j,T_{j-1})\Big]x_0. \tag{7}$$

The product is from $j = i$ down to $j = 1$ so that the sequence of matrix multiplications is correct. Now, suppose each $\Phi^j(t,T_0)$ has been solved for all times $t \in [T_0,T_N]$ and for all $j = 1,\ldots,N$ and stored in memory. Then, $\Phi^j(T_j,T_{j-1})$ is $\Phi^j(T_j,T_0)\Phi^j(T_{j-1},T_0)^{-1}$. It follows that $x(t)$ may be computed using only addition and multiplication for any arbitrary choice of switching times.

B. Computing $P(t)$

We wish to find an analogous operator to the STM to be the SIDE for computing $P(t)$. We refer to such an operator as the *adjoint-transition matrix*, or ATM, and denote $\Psi^i : \mathbb{R} \times \mathbb{R} \to \mathbb{R}^{n\times n}$, $i = 1,\ldots,N$ as the ATM corresponding to each linear mode. We also consider the ATM that corresponds to $\overline{A}(t,\mathcal{T})$, which we label as $\overline{\Psi}$. We will find that $\overline{\Psi}$ may be calculated from the Ψ^i for arbitrary switching times using only matrix multiplication and addition. We will also find that the operation,

$$P(t) = \overline{\Psi}(t,T_N) \circ P_1, \tag{8}$$

gives the solution to Eq (5). First, we define the ATM and its properties and second, use the properties to calculate $P(t)$ from Ψ^i, $i = 1,\ldots,N$ and the terminal condition P_1.

*Definition 1: Consider the linear systems defined by $A(t)$, associated STM, $\Phi(t,\tau)$, and cost function defined by $Q(t)$. Then, the **adjoint-transition matrix (ATM)**, $\Psi(\tau,t) : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}^{n\times n}$, is defined as the solution to*

$$\tfrac{d}{d\tau}\Psi(\tau,t) = -A(\tau)^T\Psi(\tau,t) - \Psi(\tau,t)A(\tau) - Q(\tau) \quad \text{subject to: } \Psi(t,t) = 0_{n\times n} \tag{9}$$

Note that the arguments of $\Psi(\tau,t)$ and $\Phi(t,\tau)$ are switched. This notational choice is due to how the state and adjoint are solved in opposing directions.

The following theorem defines the properties of the ATM that will be used to compute $P(t)$ in a similar manner the properties of the STM are used to compute $x(t)$.

Theorem 2: The ATM, Ψ, characterized by $A(t)$ and associated STM, $\Phi(t,\tau)$, and cost function defined by $Q(t)$ has the following properties:

$$\begin{aligned}
&1.\quad \Psi(\tau,t) = \int_{\tau}^{t} \Phi(s,\tau)^T Q(s)\Phi(s,\tau)\,\mathrm{d}s \\
&2.\quad \Psi(\tau,t) = -\Phi(t,\tau)^T \Psi(t,\tau)\Phi(t,\tau) \\
&3.\quad \begin{aligned} P(\tau) &= \Psi(\tau,t)\circ P(t) \\ &= \Psi(\tau,t) + \Phi(t,\tau)^T P(t)\Phi(t,\tau) \end{aligned} \\
&4.\quad \begin{aligned} \Psi(t_1,t_3) &= \Psi(t_1,t_2)\circ\Psi(t_2,t_3) \\ &= \Psi(t_1,t_2) + \Phi(t_2,t_1)^T\Psi(t_2,t_3)\Phi(t_2,t_1), \end{aligned}
\end{aligned} \tag{10}$$

where $t,\tau,t_1,t_2,t_3 \in \mathbb{R}$ are arbitrary times.

Proof: We prove each property separately.

Property 1: Property 1 may be derived directly from the definition of the ATM. Take the derivative of $\Psi(\tau,t)$ with respect to τ:

$$\begin{aligned}
&\frac{d}{d\tau}\Psi(\tau,t) = \frac{d}{d\tau}\Big[\int_{\tau}^{t}\Phi(s,\tau)^T Q(s)\Phi(s,\tau)\,\mathrm{d}s\Big] \\
&= \int_{\tau}^{t}\frac{d}{d\tau}\left[\Phi(s,\tau)^T Q(s)\Phi(s,\tau)\right]\mathrm{d}s - \Phi(\tau,\tau)^T Q(\tau)\Phi(\tau,\tau) \\
&= \int_{\tau}^{t}[-A(\tau)^T\Phi(s,\tau)^T Q(s)\Phi(s,\tau) \\
&\quad -\Phi(s,\tau)^T Q(s)\Phi(s,\tau)A(\tau)]\mathrm{d}s - Q(\tau) \\
&= -A(\tau)^T\Psi(\tau,t) - \Psi(\tau,t)A(\tau) - Q(\tau).
\end{aligned}$$

Clearly, property 1 is in the integral form of the ATM. Further, notice that property 1 is consistent with the boundary condition, $\Psi(t,t) = 0$.

Property 2: Start with property 1:

$$\begin{aligned}
\Psi(\tau,t) &= \int_{\tau}^{t}\Phi(s,\tau)^T Q(s)\Phi(s,\tau)\,\mathrm{d}s \\
&= -\Phi(t,\tau)^T\int_{t}^{\tau}\Phi(s,t)^T Q(s)\Phi(s,t)\,\mathrm{d}s\Phi(t,\tau) \\
&= -\Phi(t,\tau)^T\Psi(t,\tau)\Phi(t,\tau)
\end{aligned}$$

Properties 3 and 4 are proven using the Fundamental Theorem of ODEs [9] by showing that for each property, the left-hand side, $\mathcal{L}$, and right-hand side, $\mathcal{R}$, share the same differential equation and initial condition.

Property 3: The left and right sides of property 3 are $\mathcal{L}(\tau) = P(\tau)$ and $\mathcal{R}(\tau) = \Psi(\tau,t) + \Phi(t,\tau)^T P(t)\Phi(t,\tau)$. The initial conditions are $\mathcal{L}(t) = P(t)$ and $\mathcal{R}(t) = \Psi(t,t) + \Phi(t,t)P(t)\Phi(t,t)$. Simplifying $\mathcal{R}(t)$ by noting that $\Psi(t,t) = 0$ and $\Phi(t,t) = \Phi(t,t) = I$, results in $\mathcal{R}(t) = P(t)$. Now, to check the differential equations:

$$\frac{d}{d\tau}\mathcal{L}(\tau) = \frac{d}{d\tau}P(t) = -A(\tau)^T P(\tau) - P(\tau)A(\tau) - Q(\tau), \tag{11}$$

and

$$\begin{aligned}
\tfrac{d}{d\tau}\mathcal{R}(\tau) &= \tfrac{d}{d\tau}\Big[\Psi(\tau,t) + \Phi(t,\tau)^T P(t)\Phi(t,\tau)\Big] \\
&= -A(\tau)^T\Psi(\tau,t) - \Psi(\tau,t)A(\tau) - Q(\tau) \\
&\quad -A(\tau)^T\Phi(t,\tau)^T P(t)\Phi(t,\tau) \\
&\quad -\Phi(t,\tau)^T P(t)\Phi(t,\tau)A(\tau).
\end{aligned}$$

However, $\Psi(\tau,t) = P(\tau) - \Phi(t,\tau)^T P(t)\Phi(t,\tau)$, so

$$\frac{d}{d\tau}\mathcal{R}(\tau) = -A(\tau)^T P(\tau) - P(\tau)A(\tau) - Q(\tau),$$

thus satisfying the Fundamental Theorem of ODEs and proving property 3.

Property 4: The left and right sides of property 4 are $\mathcal{L}(t_1) = \Psi(t_1,t_3)$ and $\mathcal{R}(t_1) = \Psi(t_1,t_2) + \Phi(t_2,t_1)^T \cdot\Psi(t_2,t_3)\Phi(t_2,t_1)$. Furthermore, the initial conditions are $\mathcal{L}(t_3) = \Psi(t_3,t_3) = 0$ and

$$\begin{aligned}
\mathcal{R}(t_3) &= \Psi(t_3,t_2) + \Phi(t_2,t_3)^T\Psi(t_2,t_3)\Phi(t_2,t_3) \\
&= \int_{t_3}^{t_2}\Phi(s,t_3)^T Q(\tau)\Phi(s,t_3)\mathrm{d}s \\
&= +\Phi(t_2,t_3)^T\int_{t_2}^{t_3}\Phi(s,t_2)^T Q(\tau)\Phi(s,t_2)\mathrm{d}s\Phi(t_2,t_3) \\
&= -\int_{t_2}^{t_3}\Phi(s,t_3)^T Q(\tau)\Phi(s,t_3)\mathrm{d}s \\
&\quad +\int_{t_2}^{t_3}\Phi(s,t_3)^T Q(\tau)\Phi(s,t_3)\mathrm{d}s = 0.
\end{aligned}$$

Now, the equivalence of the differential equations of $\mathcal{L}$ and $\mathcal{R}$ are checked:

$$\begin{aligned}
&\tfrac{d}{dt_1}\mathcal{L}(t_1) = \tfrac{d}{dt_1}\Psi(t_1,t_3) \\
&\quad = -A(t_1)^T\Psi(t_1,t_3) - \Psi(t_1,t_3)A(t_1) - Q(t_1), \text{ and} \\
&\tfrac{d}{dt_1}\mathcal{R}(t_1) = \tfrac{d}{dt_1}\Big[\Psi(t_1,t_2) + \Phi(t_2,t_1)^T\Psi(t_2,t_3)\Phi(t_2,t_1)\Big] \\
&\quad = -A(t_1)^T\Psi(t_1,t_2) - \Psi(t_1,t_2)A(t_1) - Q(t_1) \\
&\quad\quad -A(t_1)^T\Phi(t_2,t_1)^T\Psi(t_2,t_3)\Phi(t_2,t_1) \\
&\quad\quad -\Phi(t_2,t_1)^T\Psi(t_2,t_3)\Phi(t_2,t_1)A(t_1).
\end{aligned}$$

Recall, $\Psi(t_1,t_2) = \Psi(t_1,t_3) - \Phi(t_2,t_1)^T\Psi(t_2,t_3)\Phi(t_2,t_1)$, so

$$\frac{d}{dt_1}\mathcal{R}(t_1) = -A(t_1)^T\Psi(t_1,t_3) - \Psi(t_1,t_3)A(t_1) - Q(t_1).$$

The left-hand and right-hand sides of property 4 share the same differential equation and initial conditions, completing the proof of the properties. ■

The ATM may be solved from its differential form, Eq (9). The ATM's integral form, property 1, is useful in proofs. The second property specifies the operation of the ATM on the adjoint such that the operation transitions the adjoint from one time to another (i.e. $P(\tau) = \Psi(\tau,t)\circ P(t)$). Property 4 specifies how an ATM operates on another ATM. Property 4 will be useful for computing $\overline{\Psi}$ from the set $\{\Psi_i\}_i^N$.

Now to prove the ATM is the operator that satisfies Eq (8).

Theorem 3: The operator that satisfies Eq (8) is $\overline{\Psi}(\tau,t)$, where $\overline{\Psi}(\tau,t)$ is characterized by two copies of the linear system defined by $\overline{A}(t,\mathcal{T})$.

Proof: Using property 3 and recalling that $P(T_N) = P_1$,

$$\overline{\Psi}(t,T_N)\circ P_1 = \overline{\Psi}(t,T_N) + \overline{\Phi}(T_N,t)^T P_1 \overline{\Phi}(T_N,t),$$

which is the solution to Eq (5). ∎

Using property 4 of ATM and defining $\Psi^i := \Psi_{A_i,A_i}$ for $i = 1,\dots,N$, $\overline{\Psi}(\cdot,\cdot)$ is computed as follows:

$$\begin{aligned}\overline{\Psi}(T_j,T_i) &= \overline{\Psi}(T_j,T_{j+1})\circ\cdots\circ\overline{\Psi}(T_{i-1},T_i)\\ &= \textstyle\sum_{k=j+1}^{i}\overline{\Phi}(T_{k-1},T_j)^T\Psi^k(T_{k-1},T_k)\overline{\Phi}(T_{k-1},T_j).\end{aligned} \tag{12}$$

Then, using property 3,

$$\begin{aligned}P(T_i) &= \overline{\Psi}(T_i,T_N)\circ P_1\\ &= \overline{\Psi}(T_i,T_N) + \overline{\Phi}(T_N,T_i)^T P_1\overline{\Phi}(T_N,T_i).\end{aligned} \tag{13}$$

Now, we conduct a similar numerical exercise as done when computing $x(t)$. Suppose, $\Psi^i(t,T_N)$ has been solved for all $i = 1,\dots,N$ and $t\in[T_0,T_N]$. Consequently, by noting that

$$\begin{aligned}&\Psi^k(T_{k-1},T_k) = \Psi^k(T_{k-1},T_N)\circ\Psi^k(T_N,T_k)\\ &=\Psi^k(T_{k-1},T_N)+\Phi^k(T_k,T_{k-1})^T\Psi^k(T_N,T_k)\Phi^k(T_k,T_{k-1})\\ &=\Psi^k(T_{k-1},T_N)-\Phi^k(T_k,T_{k-1})^T\Psi^k(T_k,T_N)\Phi^k(T_k,T_{k-1}),\end{aligned}$$

then, $\Psi(T_j,T_i)$, Eq (12), may be computed without solving any additional differential equations for all $i,j = 1,\dots,N$. Additionally, this result implies the calculation of $P(T_i)$ (see Eq (13)) also does not require solving any extra differential equations.

C. Example

In the following example, we solve an LQSTO problem using the ALQSTO formulation. The problem set up comes from the example in [4].

Example 1: Suppose a switch system is described by the two linear vector fields:

$$\begin{aligned}\text{Mode } \sigma = 1: &\quad f_1\Big(x(t)\Big) = A_1x(t) \quad \text{and}\\ \text{Mode } \sigma = 2: &\quad f_2\Big(x(t)\Big) = A_2x(t),\end{aligned}$$

where

$$A_1 = \begin{pmatrix}-1 & 0\\ 1 & 2\end{pmatrix} \quad \text{and} \quad A_2 = \begin{pmatrix}1 & 1\\ 1 & -2\end{pmatrix}.$$

Further suppose that at initial time $T_0 = 0$, the system is in mode $\sigma = 1$ with initial configuration $x_0 = (1,1)^T$. It is also known that the system transitions between modes 1 and 2 on 5 occasions (i.e. $\Sigma = \{1,2,1,2,1,2\}$) before the conclusion of the time interval at $T_N = 1$. The goal is to find the switching times that optimize the cost function with Lagrangian described by $Q = \text{diag}(2,2)$ and zero terminal cost.[5]

In Mathematica, we calculate the optimality conditions, Eq (6), with variable switching times. The process to do so begins by solving for $\{\Phi^i(t,T_0)\}$ and $\{\Psi^i(t,T_N)\}$ $\forall t\in[T_0,T_N]$ and storing in memory. To elaborate, we solve $\frac{\partial}{\partial t}\Phi^i(t,T_0) = A_i(t)\Phi(t,T_0)$, s.t. $\Phi^i(T_0,T_0)$ and Eq (9) for each A_i respectively using Adam's method for numerically solving differential equations and store in memory the discrete data points chosen from the method's adaptive time stepping. The value of $\{\Phi^i(t,T_0)\}$ and $\{\Psi^i(t,T_N)\}$ is then given by a third-order polynomial interpolation of the four data points surrounding the time t. In this example, a total of 162 data points were saved to fully constitute $\{\Phi^i(t,T_0)\}$ and $\{\Psi^i(t,T_N)\}$ for all $t\in[T_0,T_N]$.

Following, we compute $\{x(T_i)\}$ and $\{P(T_i)\}$ for variable switching times according to Eqs (7) and (13), respectively. By obtaining the optimality conditions in this way transforms the LQSTO problem into an ALQSTO problem. Under this formulation, the optimal switching times are given by finding the switching times that are the roots to the optimality condition. In Mathematica, we conduct Newton's method with a trust region, starting with an evenly distributed set of initial switching times. Doing so results in the locally optimal switching times, $\mathcal{T}^\star = (0.100, 0.297, 0.433, 0.642, 0.767)^T$, which have an associated locally optimal cost of 2.252. The algorithm converges after six iterations to within 10^{-8} of $\|\nabla J(\mathcal{T})\| = 0$.

V. Implementation and Complexity

For ALQSTO, we assume the STM, $\Phi^i(t,T_0)$, and ATM, $\Psi^i(T_N,t)$, corresponding to the N modes have been previously solved for and saved in memory for all $t\in[T_0,T_N]$. Then, following from the previous sub-section, the computation of $x(t)$ and $P(t)$ (and consequently the optimality equations) relies simply on memory calls and matrix algebra. For this reason, the number of differential equations one needs to solve at each iteration of the descent direction is 0.

The complexity of ALQSTO may instead be discussed in terms of the number of matrix multiplications.

Theorem 4: Given $\Phi^i(t,T_0)$ and $\Psi^i(t,T_N)$ for all $i = 1,\dots,N$ and $t\in[T_0,T_N]$, the number of matrix multiplications needed to compute $\{x(T_i)\}_{i=1}^{N-1}$, $\{P(T_i)\}_{i=1}^{N-1}$ and the right hand side of the optimality equations, Eq (6) for ALQSTO, is $7(N-1)$.

Proof: When computing $x(T_i)$, begin with $x(T_0) = x_0$. Then, recursively calculate

$$x(T_i) = \Phi^i(T_i,T_{i-1})x(T_{i-1}) \tag{14}$$

where

$$\Phi^i(T_i,T_{i-1}) = \Phi^i(T_i,T_0)\Phi^i(T_{i-1},T_0)^{-1}. \tag{15}$$

Therefore, each additional mode requires two extra multiplications and a matrix inverse. However, $\Phi^i(t,T_0)^{-1}$ may too be stored in memory for $t\in[T_0,T_N]$ if desired.

Similarly, for $P(T_i)$, begin with $P(T_N) = P_1$ and recursively calculate

$$\begin{aligned}P(T_i) = &\Psi^{i+1}(T_i,T_{i+1})\\ &+\Phi^{i+1}(T_{i+1},T_i)^T P(T_{i+1})\Phi^{i+1}(T_{i+1},T_i).\end{aligned}$$

[5] We use $diag()$ as a compact representation of the diagonal matrix.

Recall $\Psi^{i+1}(T_i, T_{i+1}) = \Psi^{i+1}(T_i, T_N) - \Phi^{i+1}(T_{i+1}, T_i)^T \cdot \Psi^{i+1}(T_{i+1}, T_N)\Phi^{i+1}(T_{i+1}, T_i)$ and that $\Phi^{i+1}(T_{i+1}, T_i)$ has already been calculated using Eq (15) for $\{x(T_i)\}$. Therefore,

$$P(T_i) = \Psi^{i+1}(T_i, T_N) + \Phi^{i+1}(T_{i+1}, T_i)^T [P(T_{i+1}) - \Psi^{i+1}(T_{i+1}, T_N)]\Phi^{i+1}(T_{i+1}, T_i). \quad (16)$$

Thus, each additional mode requires two additional multiplications to compute $P(t)$. Furthermore, each of the $N-1$ optimality equations, Eq (6), require three multiplications. In total, these computations require $7(N-1)$ matrix multiplications. ■

A. Implementation by Steepest Descent

Eqs (14) and (16) in the proof of Theorem 4 show how $x(T_i)$ and $P(T_i)$ should be calculated for executional efficiency. These equations are easily implemented in a numerical optimization routine. In Algorithm 1, we present solving the ALQSTO problem using steepest descent. The steepest descent direction is the negative gradient of the cost:

$$\nabla J(\mathcal{T}) = \{x(T_i)^T P(T_i)[A_i(T_i) - A_{i+1}(T_i)]x(T_i)\}_1^{N-1}, \quad (17)$$

which equals the 0 vector when at a local optimum (i.e. satisfies Eq (6)).

Algorithm 1 *ALQSTO* Steepest Descent: $sd()$

Arguments:

Σ is the mode sequence
$\mathcal{T}_0$ is the set of initial switching times
$\Phi^i(t, T_0)$ are the STM $\forall i \in \{1, \ldots, N\}$ and $t \in [T_0, T_N]$
$\Psi^i(t, T_N)$ are the ATM $\forall i \in \{1, \ldots, N\}$ and $t \in [T_0, T_N]$

$\mathcal{T}^*$ = $sd(\Sigma, \mathcal{T}_0, \Phi^i(t, T_0), \Psi^i(t, T_N))$**:**

(1) $k = 0$; $\mathcal{T}_k = \mathcal{T}_0$; $x(T_0) = x_0$; $P(T_N) = P_1$
while $\|\nabla J(\mathcal{T}_k)\|$ is not near 0 **do**
  (2) Recursively solve for $\{x(T_i)\}_{i=1}^{N-1}$ with switching times $\mathcal{T}_k$ (see Eq (14))
  (3) Recursively solve for $\{P(T_i)\}_{i=1}^{N-1}$ with switching times $\mathcal{T}_k$ (see Eq (16))
  (4) Calculate the gradient, $\nabla J(\mathcal{T}_k)$ (see Eq (17))
  (5) Update the switching times: $\mathcal{T}_{k+1} = \mathcal{T}_k - \gamma \nabla J(\mathcal{T}_k)$, for some step size γ
  (6) $k = k + 1$
end while

If one needs to calculate the cost (e.g. for choosing the step size from a line search as in [2]), then notice that

$$\begin{aligned} J(\mathcal{T}) &= \int_{T_0}^{T_N} \frac{1}{2} x(\tau)^T Q(\tau) x(\tau)\, \mathrm{d}\tau + \frac{1}{2} x(T_N)^T P_1 x(T_N) \\ &= \tfrac{1}{2} x_0^T \Big[\int_{T_0}^{T_N} \overline{\Phi}(\tau, T_0)^T Q(\tau) \overline{\Phi}(\tau, T_0)\, \mathrm{d}\tau \\ &\quad + \overline{\Phi}(T_N, T_0)^T P_1 \overline{\Phi}(T_N, T_0) \Big] x_0 \\ &= \tfrac{1}{2} x_0^T \left[\overline{\Psi}(T_0, T_N) \circ P_1 \right] x_0 \\ &= \tfrac{1}{2} x_0^T \left[\overline{\Psi}(T_0, T_1) \circ P(T_1) \right] x_0. \end{aligned}$$

The computation, $\overline{\Psi}(T_0, T_1) \circ P(T_1)$ should be calculated from the already calculated $P(T_1)$ (from step 3) using Eq (16). Then, the number of additional matrix multiplications to calculate the cost is 4.

B. Complexity Comparison with the Literature

The literature has found that each step of steepest descent for switching time optimization requires solving a constant number of differential equations. In [3], [5], the gradient requires solving the state, Eq (2a), forward in time, and the co-state, Eq (2b), backward in time. This approach works for non-linear switched systems. In this discussion, we refer to this steepest descent calculation as the "differential gradient switching time optimization," or DGSTO. Moreover, in [14], [15], Xu and Antsaklis present an approach which solves for the Riccati relation, Eq (5), and calculates the state using the matrix exponential. Their approach assumes the modes are linear time-invariant. We shall refer to this technique as "Xu and Antsaklis switching time optimization," or XASTO.

Each step of the descent of DGSTO and XASTO requires solving an additional constant number of differential equations. In comparison, each step of descent for ALQSTO requires solving zero differential equations, which implies the ALQSTO descent direction calculation time is invariant to any choice of differential equation solver. In comparison, the execution time of DGSTO and XASTO depends on the number of steps a variable (or fixed) ODE integrator requires to solve their respective differential equations.

It is of note, however, that DGSTO generalizes to non-linear switched systems. In fact, ALQSTO's restriction to quadratic costs that are subject to linear switched systems is the reason why ALQSTO does not require solving any differential equations during the optimization.

Furthermore, the complexity of ALQSTO comes at a price; ALQSTO trades computational complexity for memory demands. Depending on the application, storing the STM and ATM over the full time interval for each mode may be too costly, in which case, XASTO or DGSTO may be preferable.

C. Example

We compare the execution times of steepest descent for ALQSTO, XASTO and DGSTO in an example.

Example 2: Consider the bi-modal system in Example 1. For this example, however, the number of modes, N, is varied and the execution time to calculate a single gradient is compared between ALQSTO and DGSTO. The mode sequence begins with mode 1 and alternates between modes 1 and 2 over $N-1$ occasions.

Fig.1 compares the calculation times of the gradient for ALQSTO, XASTO and DGSTO for up to 100 modes. Fig.2 shows the gradient calculation time for ALQSTO for up to 1000 modes. The initial computation time of 0.016 seconds for calculating the SIDE is not factored into the results for ALQSTO. The results were simulated in Mathematica on a 2.26 GHz PC. Mathematica's AbsoluteTime[] function was used to time the gradient calculations. The differential equations for DGSTO and XASTO were solved using

NDSolve[] with method "ExplicitRungeKutta"—an adaptive time stepping method.

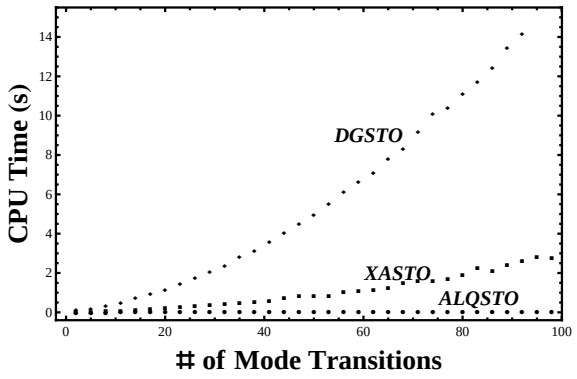

Fig. 1. Comparison of the execution times for one calculation of the gradient for ALQSTO, XASTO and DGSTO. At 100 modes, one gradient calculation of ALQSTO executes in 0.046 seconds.

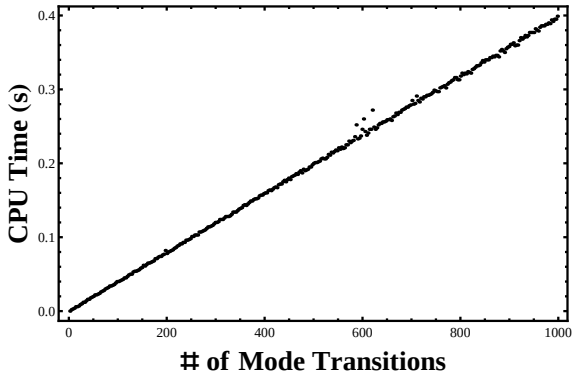

Fig. 2. The execution times for one calculation of the steepest descent direction for ALQSTO. The number of modes is varied between 1 and 1000.

Remark: A few remarks on the results

1) For this example, at 100 modes, Fig.1 shows that XASTO is greater than one order of magnitude slower than ALQSTO, while DGSTO is greater than two orders of magnitude slower.
2) A different choice of ODE integrator may result in a quicker execution of XASTO and DGSTO. For instance, an integrator with a fixed step size would result in a linear relation between the number of modes and execution time. However, such an integrator would lose accuracy, especially near the switching times due to the state and co-state being non-differentiable at those times. The choice of step size would likely depend on the number of modes and the stability of the integration.

   This sort of discussion is irrelevant for ALQSTO. The accuracy of ALQSTO depends on the numerical solver used to solve for $\{\Phi^i(t, T_0)\}$ and $\{\Psi^i(t, T_N)\}$. Since the numerical integration of these STM and ATM are as smooth as each of the modes, ALQSTO has additional numerical robustness. Furthermore, since the integration is only done once and off-line from the optimization, the integration may be as slow as needed.

VI. Conclusion

This paper considers the switching time optimization of linear time-varying switched systems subject to quadratic cost. We formulate the problem such that a set of differential equations, which we refer to as single integration differential equations (SIDE), may be solved once and off-line from the optimization routine and that once the SIDE have been solved, no more integration is necessary to solve the optimization problem using standard techniques (e.g. steepest descent, Newton's method, etc.). We refer to this formulation as algebraic linear quadratic switching time optimization, or ALQSTO. Once the SIDE have been solved, the computation of the gradient for an arbitrary set of switching times requires an $O(N)$ number of matrix multiplications, resulting in fast execution. Furthermore, since the SIDE are as smooth as the switched system's vector fields, the ALQSTO formulation has additional robustness to numerical errors compared with solving the state and co-state directly.

References

[1] B. D. O. Anderson and J. B. Moore. *Optimal Control: Linear Quadratic Methods*. Dover Publications, INC, 1990.

[2] L. Armijo. Minimization of functions having lipschitz continuous first-partial derivatives. *Pacific Journal of Mathematics*, 16:1–3, 1966.

[3] T. M. Caldwell and T. D. Murphey. Switching mode generation and optimal estimation with application to skid-steering. *Automatica*, 47:50–64, 2011.

[4] M. Egerstedt, S.-I. Azuma, and H. Axelsson. Transition-time optimization for switched-mode dynamical systems. *IEEE Transactions on Automatic Control*, 51:110–115, 2006.

[5] M. Egerstedt, Y. Wardi, and F. Delmotte. Optimal control of switching times in switched dynamical systems. *IEEE Conference on Decision and Control*, 3:2138–2143, 2003.

[6] A. Giua, C. Seatzu, and C. Van Der Mee. Optimal control of autonomous linear systems switched with a pre-assigned finite sequence. *In Proc. of the 2001 IEEE ISIC*, pages 144–149, 2001.

[7] A. Giua, C. Seatzu, and C. Van Der Mee. Optimal control of switched autonomous linear systems. *IEEE Conference on Decision and Control*, 40:2472–2477, 2001.

[8] A. J. Goldsmith and S.-G. Chua. Variable-rate variable-power mqam for fading channels. *IEEE Transactions on Communications*, 45(10):1218–1230, 1997.

[9] M. W. Hirsch and S. Smale. *Differential Equations, Dynamical Systems, and Linear Algebra*. Academic Press, New York, 1974.

[10] E. R. Johnson and T. D. Murphey. Second order switching time optimization for time-varying nonlinear systems. *IEEE Conference on Decision and Control*, 48:5281–5286, 2009.

[11] E. R. Johnson and T. D. Murphey. Second-order switching time optimization for non-linear time-varying dynamic systems. *IEEE Transactions on Automatic Control*, 2011. (In Press).

[12] M. J. Neely. Energy optimal control for time varying wireless networks. *IEEE Transactions on Information Theory*, 52(7):2915–2934, 2006.

[13] A. Schild, X. C. Ding, M. Egerstedt, and J. Lunze. Design of optimal switching surfaces for switched autonomous systems. *IEEE Conference on Decision and Control*, 48:5293–5298, 2009.

[14] X. Xu and P. J. Antsaklis. An approach for solving general switched linear quadratic optimal control problems. *IEEE Conference on Decision and Control*, 40:2478 – 2483, 2001.

[15] X. Xu and P. J. Antsaklis. Optimal control of switched autonomous systems. *IEEE Conference on Decision and Control*, 41:4401 – 4406, 2002.