

*Electrical and Computer Engineering*

*ECE Technical Reports*

---

*Purdue Libraries*

*Year 2009*

---

# A Study of the Discrete-Time Switched LQR Problem

Wei Zhang\*

Jianghai Hu<sup>†</sup>

Alessandro Abate<sup>‡</sup>

\*Purdue University, zhang70@purdue.edu

<sup>†</sup>Purdue University, jianghai@purdue.edu

<sup>‡</sup>Stanford University, aabate@stanford.edu

This paper is posted at Purdue e-Pubs.

<http://docs.lib.purdue.edu/ecetr/384>

# A Study of the Discrete-Time Switched LQR Problem

Wei Zhang, Jianghai Hu and Alessandro Abate

## Abstract

This paper studies the discrete-time switched LQR (DSLQR) problem based on a dynamic programming approach. One contribution of this paper is the analytical characterization of both the value function and the optimal hybrid-control strategy of the DSLQR problem. Their connections to the Riccati equation and the Kalman gain of the classical LQR problem are also discussed. Several interesting properties of the value functions are derived. In particular, we show that under some mild conditions, the family of finite-horizon value functions of the DSLQR problem is homogeneous (of degree 2), uniformly bounded over the unit ball, and converges exponentially fast to the infinite-horizon value function. Based on these properties, efficient algorithms are proposed to solve the finite-horizon and infinite-horizon DSLQR problems. More importantly, we establish conditions under which the strategies generated by the algorithms are stabilizing and suboptimal. These conditions are derived explicitly in terms of subsystem matrices and are thus very easy to verify. The proposed algorithms and the analysis provide a systematic way of solving the DSLQR problem with guaranteed closed-loop stability and suboptimal performance. Simulation results indicate that the proposed algorithms can efficiently solve not only specific but also randomly generated DSLQR problems, making the NP-hard problems numerically tractable.

## I. INTRODUCTION

Switched systems arise naturally in many engineering fields, such as power electronics [1], [2], embedded systems [3], [4], manufacturing [5], and communication networks [6], etc. Incorporating the switching behavior in the model and controller structures offers much greater freedom and more possibilities for capturing complex system dynamics, achieving stabilization and improving the overall performance of the feedback systems. In the last decade or so, the stability and stabilizability of switched systems have been extensively studied [7], [8], [9], [10]. Many theoretical and numerical tools have been developed for the stability analysis of various switched systems. These stability results have also led to some controller synthesis algorithms that ensure stability of some simple switched systems [11], [12], [13], [14]. However, for many engineering applications, ensuring the stability is only the first step rather than the ultimate design goal. How to design a control strategy that not only stabilizes a given switched system, but also optimizes certain design criteria is an even more meaningful research problem.

The focus of this paper is on the optimal discrete-time linear quadratic regulation problem for switched linear systems, hereby referred to as the DSLQR problem. The goal is to develop a computationally appealing algorithm to

This work was partially supported by the National Science Foundation under Grant CNS-0643805

W. Zhang and J. Hu are with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47906. Email:{zhang70,jianghai@purdue.edu}.

A. Abate is with the Department of Aeronautics and Astronautics, Stanford University, CA 94305. Email:{aabate@stanford.edu}.

construct an optimal or suboptimal feedback strategy that minimizes a given quadratic cost function. The problem is of fundamental importance both in theory and practice and has challenged researchers for many years. The bottleneck mostly lies in the determination of the optimal switching strategy. Many methods have been proposed to tackle this problem, most of which are in a divide-and-conquer manner. Algorithms for optimizing the switching instants for a fixed mode sequence have been developed for general switched systems in [15] and for switched systems with autonomous dynamics in [16]. Although an algorithm for updating the switching sequence is discussed in [16], finding the best switching sequence is still an NP-hard problem, even for switched linear systems.

This paper studies the DSLQR problem from the dynamic programming (DP) perspective. The last few years have seen increasing interest in using DP to solve various optimal control problems of switched systems. In [17], Xu and Antsaklis used DP to study the continuous-time switched LQR problem and developed an algorithm to find the suboptimal switching instants and continuous control for a fixed switching sequence. In [18], Rantzer and Johansson derived lower and upper bounds for the value function of the quadratic optimal control problem of piecewise affine systems; these bounds were then used to construct a suboptimal control strategy. A discrete-time version of this problem was studied by Bemporad *et al.* in [19], [20], where the value function and the optimal control law were proved to be piecewise quadratic and piecewise linear, respectively. Based on these structural properties, an algorithm based on multi-parametric programming was developed to compute the optimal feedback control law. More recently, Lincoln and Rantzer developed a general relaxation procedure in [21] to tackle the curse of dimensionality of dynamic programming. This procedure was also employed to study the infinite-horizon DSLQR problem in [21], [22] and the quadratic optimal control problem of continuous-time switched homogeneous systems in [23].

One contribution of this paper is the analytical characterization of both the value function and the optimal control strategies for general DSLQR problems. In particular, we show that the value function of the DSLQR problem is the pointwise minimum of a finite number of quadratic functions. These quadratic functions can be exactly characterized by a finite set of positive semidefinite (p.s.d.) matrices, which can be obtained recursively using the so-called *Switched Riccati Mapping*. Explicit expressions are also derived for both the optimal switching law and the optimal continuous control law. Both of them are in the state-feedback form and are homogeneous on the state space. Furthermore, the optimal continuous control is shown to be piecewise linear with different Kalman-type feedback gains within different conic regions of the state space. Although other researchers have also suggested a piecewise affine structure for the optimal feedback control ([19], [20], [24]), the analytical expression of the optimal feedback gain and in particular its connection with the Kalman gain and the Riccati equation of the classical LQR problem have not been explicitly presented.

Another contribution of this paper is the derivation of various properties of the value functions of the DSLQR problem. In particular, it is proved that under some mild conditions, the family of the finite-horizon value functions of the DSLQR problem is homogeneous (of degree 2), uniformly bounded over the unit ball, and converges exponentially fast to the infinite-horizon value function. More importantly, the exponential convergence rate of the value iteration is characterized analytically in terms of the subsystem matrices. This provides an efficient way

of terminating the value iterations, especially for high-dimensional state spaces. The above results, especially the convergence-rate characterization, have not been adequately investigated in the literature.

The last contribution of this paper is the design and analysis of various efficient algorithms for solving the optimal and suboptimal DSLQR problems. The key idea is to use convex optimization to identify and remove the matrices that are redundant in terms of characterizing the optimal and suboptimal strategies. This is in line with the approaches of Neuro-dynamic programming ([25]) and approximate dynamic programming ([21]), both of which try to simplify the computations by finding compact representations of the value functions up to certain numerical relaxations. Compared with the previous work, our distinction mostly lies on the analysis of these algorithms. We establish conditions under which the strategies generated by the proposed algorithms are stabilizing and suboptimal. More importantly, these conditions are derived explicitly in terms of subsystem matrices and are very easy to verify. Therefore, the proposed algorithms, together with the analysis, provide a systematic way of solving the DSLQR problem with guaranteed closed-loop stability and suboptimal performance. Simulation results indicate that the proposed algorithms can efficiently solve not only specific but also randomly generated DSLQR problems, making the NP-hard problems numerically tractable.

This paper is organized as follows. In Section II, the DSLQR problem is formulated. The value function of the DSLQR problem is derived in a simple analytical form in Section III. Various interesting properties of the value functions are derived in Section IV. These properties are then used in Sections V and VI to develop optimal and suboptimal algorithms for solving the DSLQR problems. Finally, some concluding remarks are given in Section VII.

## II. PROBLEM FORMULATION

Consider the discrete-time switched linear system described by:

$$x(t+1) = A_{v(t)}x(t) + B_{v(t)}u(t), \quad t \in T_N \triangleq \{0, \dots, N-1\}, \quad (1)$$

where  $x(t) \in \mathbb{R}^n$  is the continuous state,  $v(t) \in \mathbb{M} \triangleq \{1, \dots, M\}$  is the discrete mode,  $u(t) \in \mathbb{R}^p$  is the continuous control and  $T_N$  is the control horizon with length  $N$  (possibly infinite). The integers  $n$ ,  $M$  and  $p$  are all finite and the control  $u$  is unconstrained. The sequence of pairs  $\{(u(t), v(t))\}_{t=0}^{N-1}$  is called the *hybrid control sequence*. For each  $i \in \mathbb{M}$ ,  $A_i$  and  $B_i$  are constant matrices of appropriate dimensions and the pair  $(A_i, B_i)$  is called a subsystem. This switched linear system is time invariant in the sense that the set of available subsystems  $\{(A_i, B_i)\}_{i=1}^M$  is independent of time  $t$ . We assume that there is no internal forced switching, i.e., the system can stay at or switch to any mode at any time instant. At each time  $t \in T_N$ , denote by  $\xi_{t,N} \triangleq (\mu_{t,N}, \nu_{t,N}) : \mathbb{R}^n \rightarrow \mathbb{R}^p \times \mathbb{M}$  the *hybrid control law* of system (1), where  $\mu_{t,N} : \mathbb{R}^n \rightarrow \mathbb{R}^p$  is called the *continuous control law* and  $\nu_{t,N} : \mathbb{R}^n \rightarrow \mathbb{M}$  is called the *switching control law*. A sequence of hybrid control laws over the horizon  $T_N$  constitutes an *N-horizon feedback policy*:  $\pi_N \triangleq \{\xi_{0,N}, \xi_{1,N}, \dots, \xi_{N-1,N}\}$ . If system (1) is driven by a feedback policy  $\pi_N$ , then the closed-loop dynamics is governed by

$$x(t+1) = A_{\nu_{t,N}(x(t))}x(t) + B_{\nu_{t,N}(x(t))}\mu_{t,N}(x(t)), \quad t \in T_N. \quad (2)$$

For a given initial state  $x(0) = z \in \mathbb{R}^n$ , the performance of the feedback policy  $\pi_N$  can be measured by the following cost functional:

$$J_{\pi_N}(z) = \psi(x(N)) + \sum_{t=0}^{N-1} L(x(t), \mu_{t,N}(x(t)), \nu_{t,N}(x(t))), \quad (3)$$

where  $\psi : \mathbb{R}^n \rightarrow \mathbb{R}^+$  and  $L : \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{M} \rightarrow \mathbb{R}^+$  are called the *terminal cost function* and the *running cost function*, respectively. In this paper, the functions  $\psi$  and  $L$  are assumed to take the following quadratic forms:

$$\psi(x) = x^T Q_f x, \quad L(x, u, v) = x^T Q_v x + u^T R_v u, \quad \forall x \in \mathbb{R}^n, u \in \mathbb{R}^p, v \in \mathbb{M},$$

where  $Q_f = Q_f^T \succeq 0$  is the terminal-state weighting matrix, and  $Q_v = Q_v^T \succeq 0$  and  $R_v = R_v^T \succ 0$  are the running weighting matrices for the state and the control, respectively, for subsystem  $v \in \mathbb{M}$ . When the control horizon  $N$  is infinite, the terminal cost will never be incurred and the objective function, which might be infinite, becomes:

$$J_{\pi_\infty}(z) = \sum_{t=0}^{\infty} L(x(t), \mu_{t,\infty}(x(t)), \nu_{t,\infty}(x(t))). \quad (4)$$

For a possibly infinite positive integer  $N$ , denote by  $\Pi_N$  the set of all admissible  $N$ -horizon policies, i.e., the set of all sequences of functions  $\pi_N = \{\xi_{0,N}, \dots, \xi_{N-1,N}\}$  with  $\xi_{t,N} : \mathbb{R}^n \rightarrow \mathbb{R}^p \times \mathbb{M}$  for  $t \in T_N$ . The goal of this paper is to find the optimal policy  $\pi_N^*$  that minimizes the quadratic cost function defined in (3) or (4). This problem is a natural extension of the classical LQR problem to the switched linear system case and is thus called the *Discrete-time Switched LQR problem*, hereby referred to as the DSLQR problem.

*Problem 1 (DSLQR problem):* For a given initial state  $z \in \mathbb{R}^n$  and a possibly infinite positive integer  $N$ , find the  $N$ -horizon policy  $\pi_N \in \Pi_N$  that minimizes  $J_{\pi_N}(z)$  subject to the dynamic equation (2).

*Remark 1:* With the quadratic cost function (3), there always exists a solution to the finite-horizon DSLQR problem. We assume that the optimal solution also exists in the infinite-horizon case. However, for both finite and infinite horizons, the optimal solution may not be unique.

To solve Problem 1, for each time  $t \in T_N$ , we define the value function  $V_{t,N} : \mathbb{R}^n \rightarrow \mathbb{R}$  as:

$$V_{t,N}(z) = \inf_{\substack{u(j) \in \mathbb{R}^p, v(j) \in \mathbb{M} \\ t \leq j \leq N-1}} \left\{ \psi(x(N)) + \sum_{j=t}^{N-1} L(x(j), u(j), v(j)) \right\} \text{ subject to eq. (1) with } x(t) = z. \quad (5)$$

The  $V_{t,N}(z)$  so defined is the minimum cost-to-go starting from state  $z$  at time  $t$ . The minimum cost for the DSLQR problem with an initial condition  $x(0) = x_0$  is simply  $V_{0,N}(x_0)$ . Due to the time-invariant nature of the switched system (1), its value function depends only on the number of remaining time steps, i.e.,

$$V_{t,N}(z) = V_{t+m,N+m}(z),$$

for all  $z \in \mathbb{R}^n$  and all integers  $m \geq -t$ . In the rest of this paper, when no ambiguity arises, we will denote by  $V_k(z) \triangleq V_{N-k,N}(z)$  and  $\xi_k \triangleq \xi_{N-k,N}$  the value function and the hybrid control law, respectively, at time  $t = N - k$  when there are  $k$  time steps left. With the new notations, the  $N$ -horizon policy  $\pi_N$  can also be written as  $\pi_N = \{\xi_N, \dots, \xi_1\}$ . For any positive integer  $k$ , the newly introduced  $\xi_k$  can be thought of as the first step of a  $k$ -horizon policy.

By a standard result of Dynamic Programming [26], for any finite integer  $N$ , the value function  $V_N$  can be obtained recursively using the one-stage *value iteration*:

$$V_{k+1}(z) = \inf_{u,v} \{L(z, u, v) + V_k(A_v z + B_v u)\}, \forall z \in \mathbb{R}^n,$$

with initial condition  $V_0(z) = \psi(z)$ ,  $\forall z \in \mathbb{R}^n$ . Denote by  $V_\infty(\cdot)$  the pointwise limit (if it exists) of the sequence of functions  $\{V_k(\cdot)\}_{k=0}^\infty$  generated by the value iterations. It is well known [26] that even if  $V_\infty(z)$  exists, it may not always coincide with the infinite-horizon value function. To emphasize its substantial difference from the finite-horizon value function, the infinite-horizon value function is specially denoted by  $V^*(z)$ , i.e.,  $V^*(z) = \inf_{\pi_\infty \in \Pi_\infty} J_{\pi_\infty}(z)$ .

### III. ANALYTICAL CHARACTERIZATION OF THE FINITE-HORIZON VALUE FUNCTION

For any fixed switching sequence, the switched linear system can be viewed as a linear time-varying system. Theoretically, the finite-horizon DSLQR problem can be solved using dynamic programming by enumerating all the possible switching sequences. Clearly, this approach is not practically feasible as its complexity grows exponentially fast as  $N$  increases. Fortunately, for the DSLQR problem, such enumerations can be avoided and the value functions can be computed in a rather efficient way. The efficient computation relies on the particular analytical structure of the value function, which will be derived in this section.

We first review some important results of the classical discrete-time LQR problem. Such a problem can be viewed as a special case of the DSLQR problem with  $M = 1$ . In this special case, denote by  $(A, B)$  the system matrices and by  $Q$  and  $R$  the state and control weighting matrices, respectively. It is well known that when  $N$  is finite, the value functions of this LQR problem are of the following quadratic form:

$$V_k(z) = z^T P_k z, \quad k = 0, \dots, N, \quad (6)$$

where  $\{P_k\}_{k=0}^N$  is a sequence of p.s.d. matrices satisfying the Difference Riccati Equation (DRE):

$$P_{k+1} = Q + A^T P_k A - A^T P_k B (R + B^T P_k B)^{-1} B^T P_k A, \quad (7)$$

with initial condition  $P_0 = Q_f$ . Denote by  $\mathcal{A}$  the *positive semidefinite cone* ([27]), namely, the set of all symmetric p.s.d. matrices. Some results of the classical LQR problem are summarized in the following lemma.

*Lemma 1 ([28], [29]):* Let  $\{P_k\}_{k=0}^N$  be generated by the DRE (7), then

- 1) For each  $k = 0, \dots, N-1$ , if  $P_k \in \mathcal{A}$ , then  $P_{k+1} \in \mathcal{A}$ .
- 2) If  $(A, B)$  is stabilizable, then  $V_k(z) \rightarrow V^*(z)$  for all  $z \in \mathbb{R}^n$  as  $k \rightarrow \infty$ .
- 3) Let  $Q = C^T C$ . If  $(A, B)$  stabilizable and  $(C, A)$  detectable, then the optimal trajectory of the LQR problem is exponentially stable.

In general, when  $M \geq 2$ , the value function  $V_k(z)$  is no longer of a simple quadratic form as in (6). Nevertheless, the notion of the DRE can be generalized to the Switched LQR problems. The DRE (7) can be viewed as a mapping

from  $\mathcal{A}$  to  $\mathcal{A}$  depending on the matrices  $(A, B, Q, R)$ . We call this mapping the *Riccati Mapping* and denote by  $\rho_i : \mathcal{A} \rightarrow \mathcal{A}$  the Riccati Mapping of subsystem  $i \in \mathbb{M}$ , i.e.,

$$\rho_i(P) = Q_i + A_i^T P A_i - A_i^T P B_i (R_i + B_i^T P B_i)^{-1} B_i^T P A_i. \quad (8)$$

*Definition 1:* Let  $2^{\mathcal{A}}$  be the power set of  $\mathcal{A}$ . The mapping  $\rho_{\mathbb{M}} : 2^{\mathcal{A}} \rightarrow 2^{\mathcal{A}}$  defined by:

$$\rho_{\mathbb{M}}(\mathcal{H}) = \{\rho_i(P) : \text{for some } i \in \mathbb{M} \text{ and } P \in \mathcal{H}\}$$

is called the *Switched Riccati Mapping* (SRM) associated with Problem 1.

In words, the SRM maps a **set** of p.s.d. matrices to another **set** of p.s.d. matrices and each matrix in  $\rho_{\mathbb{M}}(\mathcal{H})$  is obtained by taking the classical Riccati mapping of some matrix in  $\mathcal{H}$  through some subsystem  $i \in \mathbb{M}$ .

*Definition 2:* The sequence of sets  $\{\mathcal{H}_k\}_{k=0}^N$  generated iteratively by  $\mathcal{H}_{k+1} = \rho_{\mathbb{M}}(\mathcal{H}_k)$  with initial condition  $\mathcal{H}_0 = \{Q_f\}$  is called the *Switched Riccati Sets* (SRSs) of Problem 1.

The SRSs always start from a singleton set  $\{Q_f\}$  and evolve according to the SRM. For any finite  $N$ , the set  $\mathcal{H}_N$  consists of  $M^N$  p.s.d. matrices. An important fact about the DSLQR problem is that its value functions are completely characterized by the SRSs.

*Theorem 1:* For  $k = 0, \dots, N$ , the value function for the DSLQR problem at time  $N - k$ , i.e., with  $k$  time steps left, is

$$V_k(z) = \min_{P \in \mathcal{H}_k} z^T P z. \quad (9)$$

Furthermore, for  $z \in \mathbb{R}^n$  and  $k = 1, \dots, N$ , if we define

$$(P_k^*(z), i_k^*(z)) = \arg \min_{(P \in \mathcal{H}_{k-1}, i \in \mathbb{M})} z^T \rho_i(P) z, \quad (10)$$

then the optimal hybrid control law at state  $z$  and time  $t = N - k$  is  $\xi_k^*(z) = (\mu_k^*(z), \nu_k^*(z))$ , where  $\mu_k^*(z) = -K_{i_k^*(z)}(P_k^*(z))z$  and  $\nu_k^*(z) = i_k^*(z)$ . Here,  $K_i(P)$  is the Kalman gain for subsystem  $i$  with matrix  $P$ , i.e.,

$$K_i(P) \triangleq (R_i + B_i^T P B_i)^{-1} B_i^T P A_i. \quad (11)$$

*Proof:* The theorem can be proved by induction. It is obvious that for  $k = 0$  the value function is  $V_0(z) = z^T Q_f z$ , satisfying (9). Now suppose equation (9) holds for some  $k \leq N - 1$ , i.e.,  $V_k(z) = \min_{P \in \mathcal{H}_k} z^T P z$ . We shall show that it is also true for  $k + 1$ . By the principle of dynamic programming and noting that  $V_k(\cdot)$  represents the value function at time  $N - k$ , the value function at time  $N - (k + 1)$  can be recursively computed as

$$\begin{aligned} V_{k+1}(z) &= \inf_{i \in \mathbb{M}, u \in \mathbb{R}^p} [z^T Q_i z + u^T R_i u + V_k(A_i z + B_i u)] \\ &= \inf_{i \in \mathbb{M}, P \in \mathcal{H}_k, u \in \mathbb{R}^p} [z^T Q_i z + u^T R_i u + (A_i z + B_i u)^T P (A_i z + B_i u)] \\ &= \inf_{i \in \mathbb{M}, P \in \mathcal{H}_k, u \in \mathbb{R}^p} [z^T (Q_i + A_i^T P A_i) z + u^T (R_i + B_i^T P B_i) u + 2z^T A_i^T P B_i u]. \end{aligned} \quad (12)$$

Since the quantity inside the bracket is quadratic in  $u$ , the optimal  $u^*$  can be easily found to be

$$u^* = -(R_i + B_i^T P B_i)^{-1} B_i^T P A_i z = -K_i(P)z, \quad (13)$$

where  $K_i(P)$  is the matrix defined in (11). Substituting  $u^*$  into (12), we obtain

$$V_{k+1}(z) = \inf_{i \in \mathbb{M}, P \in \mathcal{H}_k} \left[ z^T \left( Q_i + A_i^T P A_i - A_i^T P B_i K_i(P) \right) z \right] = \min_{i \in \mathbb{M}, P \in \mathcal{H}_k} z^T \rho_i(P) z.$$

Observing that  $\{\rho_i(P) : i \in \mathbb{M}, P \in \mathcal{H}_k\} = \rho_{\mathbb{M}}(\mathcal{H}_k) = \mathcal{H}_{k+1}$ , we have  $V_{k+1}(z) = \min_{P \in \mathcal{H}_{k+1}} z^T P z$ . In addition, let  $P_k^*(z)$  and  $i_k^*(z)$  be defined by (10). Then it can easily be seen from the above derivation that  $(-K_{i_{k+1}^*(z)}(P_{k+1}^*(z))z, i_{k+1}^*(z))$  is the optimal decision at time  $N - (k + 1)$  that achieves the minimum cost  $V_{k+1}(z)$ . ■

*Remark 2:* The piecewise quadratic structure of the value function has been proved in [20] for piecewise affine hybrid systems and has also been suggested in [21] for infinite-horizon DSLQR problems. However, the analytical expression for the value function and in particular its connection to the Kalman gain and the Riccati equation of the classical LQR problem have not been explicitly presented. Furthermore, from a computation point of view, Theorem 1 indicates that under our formulation, the value function over the entire state space can be exactly characterized by a finite number of p.s.d. matrices, which excludes the need of discretizing the state space as in [24], [30], [31].

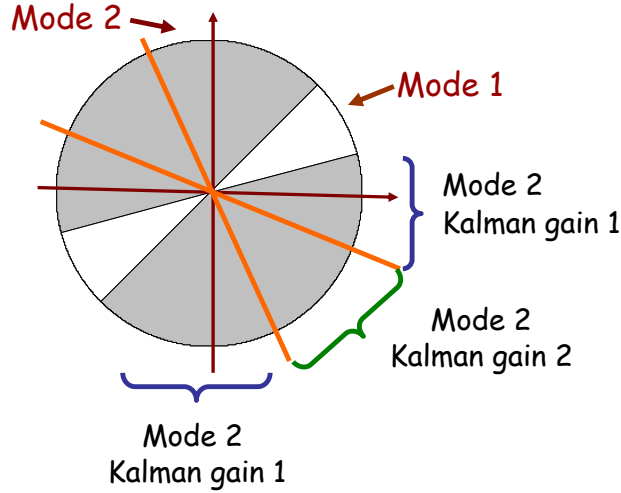


Fig. 1. Typical optimal decision regions of a two-switched system, where mode 1 is optimal within the white region and mode 2 is optimal within the gray region. The optimal mode region is further divided into smaller conic regions, each of which corresponds to a different Kalman gain.

Compared with the discrete-time LQR problem, the value function of the DSLQR problem is no longer a single quadratic function; it becomes the pointwise minimum of a finite number of quadratic functions. At each time step, instead of having a single Kalman gain for the entire state space, the optimal state feedback gain becomes state dependent. Furthermore, the minimizer  $(P_k^*(z), i_k^*(z))$  of equation (10) is radially invariant, indicating that



at each time step all the points along the same radial direction have the same optimal hybrid control law. These interesting properties are illustrated in Fig. 1 using an example in  $\mathbb{R}^2$  with 2 subsystems. At each time step, the state space is decomposed into two homogeneous regions: the white region and the gray region, which are called the *optimal switching regions*. Within the white region, one mode, say mode 1, is optimal; within the gray region, the other mode, mode 2, is optimal. Furthermore, the states within the same optimal switching region may have different optimal feedback gains (Kalman gains). This is illustrated in Fig. 1 by the further division of the gray region into smaller conic regions, each of which correspond to a different Kalman gain. It is worth mentioning that in a higher dimensional state space, the decision regions are still cones; however, these cones may not be convex and the manifolds defining the boundaries between adjacent cones may be complicated. A salient feature of the DSLQR problem is that all these complex decision regions are completely encoded in a finite number of matrices in the switched Riccati sets  $\{\mathcal{H}_k\}_{k=0}^N$ .

Theorem 1 and the above discussion have made it clear that the key for solving the DSLQR problem is the computation of the SRSs  $\{\mathcal{H}_k\}_{k=0}^N$ . Although analytical formulas are available for evaluating the matrices in these SRSs, a direct computation is almost impossible because  $|\mathcal{H}_k|$  grows exponentially fast as  $k$  increases. Nevertheless, the particular structure of the value function derived in Theorem 1 provides us a clear view of what information is necessary for making the optimal decision and, in turn, enable us to avoid many redundant computations. It is the basis of the efficient algorithms to be discussed in Sections V and VI.

#### IV. PROPERTIES OF THE VALUE FUNCTIONS

In this section, we will derive various important properties of the family of finite-horizon value functions  $\{V_N(z)\}_{N \geq 0}^\infty$  and the infinite-horizon value function  $V^*(z)$ . These properties are crucial in the design and analysis of the efficient algorithms for solving the DSLQR problems.

We first introduce some notations to be used throughout the subsequent discussions. Denote by  $I_n$  the identity matrix of dimension  $n$ . Let  $\|\cdot\|$  be the 2-norm of a given matrix or vector. Let  $\mathbb{Z}^+$  be the set of all nonnegative integers. Denote by  $\lambda_{\min}(\cdot)$  and  $\lambda_{\max}(\cdot)$  the smallest and the largest eigenvalue of a p.s.d. matrix, respectively. Define  $\lambda_Q^- = \min_{i \in \mathbb{M}} \{\lambda_{\min}(Q_i)\}$  and  $\lambda_f^+ = \lambda_{\max}(Q_f)$ . Denote by  $x_{z,N}^*(t)$  for  $0 \leq t \leq N$  an optimal trajectory originating from  $z$  at time 0 and denote by  $(u_{z,N}^*(t), v_{z,N}^*(t))$  the corresponding optimal hybrid control sequence.

##### A. Homogeneity

An immediate consequence of Theorem 1 is the homogeneity of the finite-horizon value function  $V_N$ .

*Lemma 2 (Homogeneity of  $V_N(z)$ ):*  $V_N(\lambda z) = \lambda^2 V_N(z)$ , for any  $z \in \mathbb{R}^n$ ,  $\lambda \in \mathbb{R}$  and  $N \in \mathbb{Z}^+$ .

Although the explicit expression of  $V_N(z)$  is available for any finite horizon  $N$ , little is known about the infinite-horizon value function  $V^*(z)$ . Let  $(u, v)$  be the hybrid control sequence generated by an infinite-horizon policy  $\pi_\infty$  with initial condition  $x(0) = z$ . Then the cost  $J_{\pi_\infty}(z)$  can be expressed in terms of  $(u, v)$  as:

$$J_\infty(z, u, v) = \sum_{j=0}^{\infty} L(x(j), u(j), v(j)).$$

It follows easily from the linearity of the system and the quadratic structure of the running cost that for any given mode sequence  $v$ , the function  $J_\infty(z, u, v)$  is quadratic jointly in the state and control, i.e.,

$$J_\infty(\lambda z, \lambda u, v) = \lambda^2 J_\infty(z, u, v), \quad \forall \lambda \in \mathbb{R}, \lambda \neq 0, \quad (14)$$

where  $\lambda u \triangleq \{\lambda u(0), \lambda u(1), \dots\}$ . Equality (14) also holds when either side is infinite. Since  $V^*(z)$  can be written as  $V^*(z) = \inf_{(u,v)} J_\infty(z, u, v)$ , it is also homogeneous.

*Lemma 3 (Homogeneity of  $V^*(z)$ ):*  $V^*(\lambda z) = \lambda^2 V^*(z)$ , for any  $z \in \mathbb{R}^n$  and any nonzero real number  $\lambda$ .

*Proof:* Let  $z \in \mathbb{R}^n$  and  $\lambda \in \mathbb{R}, \lambda \neq 0$  be arbitrary. Immediately from (14), we know  $V^*(\lambda z)$  is infinite whenever  $V^*(z)$  is infinite. The desired equality holds. When both  $V^*(z)$  and  $V^*(\lambda z)$  are finite, we have

$$V^*(\lambda z) = \inf_{u,v} J_\infty(\lambda z, u, v) = \inf_{u,v} J_\infty(\lambda z, \lambda u, v) = \inf_{u,v} \lambda^2 J_\infty(z, u, v) = \lambda^2 V^*(z).$$

■

The properties of the value functions presented in the rest of this section are based on the following stabilizability condition of the switched system (1).

(A1) At least one subsystem is stabilizable.

## B. Boundedness

*Proposition 1:* Under assumption (A1), there must exist a finite constant  $\beta$  such that  $V_k(z) \leq \beta \|z\|^2$ , for all  $k \in \mathbb{Z}^+$  and  $z \in \mathbb{R}^n$ . Furthermore, if the stabilizable subsystem is  $(A_i, B_i)$  and  $F$  is any feedback gain for which  $\bar{A}_i \triangleq A_i - B_i F$  is stable, then one possible choice of  $\beta$  is given by:

$$\beta = (\|Q_f\| + \|Q_i + F^T R_i F\|) \cdot \left( \sum_{j=0}^{\infty} \|\bar{A}_i^j\|^2 \right) < \infty. \quad (15)$$

*Proof:* Suppose subsystem  $(A_i, B_i)$  is stabilizable. Let  $\{P_k^{(i)}\}_{k=0}^{\infty}$  be the sequence of matrices generated by the Riccati mapping using only subsystem  $i$ , i.e.,  $P_{k+1}^{(i)} = \rho_i(P_k^{(i)})$  with  $P_0^{(i)} = Q_f$ . Since the switched system (1) can stay in subsystem  $(A_i, B_i)$  all the time, the value function of the DSLQR problem must be no greater than the value function of the LQR problem for the subsystem  $(A_i, B_i)$ , i.e.,  $V_k(z) \leq z^T P_k^{(i)} z$  for all  $k \in \mathbb{Z}^+$  and  $z \in \mathbb{R}^n$ . Thus, it suffices to show that the  $\beta$  given in (15) is an upper bound of the 2-norm of all the matrices in  $\{P_k^{(i)}\}_{k=0}^{\infty}$ . Let  $F$  be a feedback gain for which  $\bar{A}_i = A_i - B_i F$  is stable. Define  $\{\tilde{P}_k^{(i)}\}_{k=0}^{\infty}$  iteratively by

$$\tilde{P}_{k+1}^{(i)} = Q_i + \bar{A}_i^T \tilde{P}_k^{(i)} \bar{A}_i + F^T R_i F, \quad \text{with } \tilde{P}_0^{(i)} = Q_f. \quad (16)$$

In the above equation, if  $F = K_i(\tilde{P}_k^{(i)})$  for each  $k$ , where  $K_i(\cdot)$  is defined in (11), then  $\tilde{P}_k^{(i)}$  would coincide with  $P_k^{(i)}$ . In other words,  $\tilde{P}_k^{(i)}$  defines the quadratic energy cost of using the stabilizing feedback gain  $F$  instead of the time-dependent optimal Kalman gain of the  $k$ -horizon LQR problem. By a standard result of the Riccati equation theory (Theorem 2.1 in [28]), we have  $P_k^{(i)} \preceq \tilde{P}_k^{(i)}$  for all  $k \geq 0$ . Thus, it suffices to show  $\|\tilde{P}_k^{(i)}\| \leq \beta$  for each

$k \geq 0$ . By (16), we have

$$\begin{aligned}
\tilde{P}_k^{(i)} &= \tilde{P}_0^{(i)} + \sum_{j=1}^k (\tilde{P}_j^{(i)} - \tilde{P}_{j-1}^{(i)}) = \tilde{P}_0^{(i)} + \sum_{j=0}^{k-1} (\bar{A}_i^T)^j (\tilde{P}_1^{(i)} - \tilde{P}_0^{(i)}) (\bar{A}_i)^j \\
&= Q_f + \sum_{j=0}^{k-1} (\bar{A}_i^T)^{j+1} Q_f (\bar{A}_i)^{j+1} + \sum_{j=0}^{k-1} (\bar{A}_i^T)^j (Q_i - Q_f + F^T R_i F) (\bar{A}_i)^j \\
&\leq (\bar{A}_i^T)^k Q_f (\bar{A}_i)^k + \sum_{j=0}^{\infty} (\bar{A}_i^T)^j (Q_i + F^T R_i F) (\bar{A}_i)^j
\end{aligned}$$

Thus,  $\|P_k^{(i)}\| \leq \|\tilde{P}_k^{(i)}\| \leq (\|Q_f\| + \|Q_i + F^T R_i F\|) \left( \sum_{j=0}^{\infty} \|\bar{A}_i^j\|^2 \right)$ . Note that the formula of the geometric series does not directly apply here, as the 2-norm of a stable matrix may not be strictly less than 1 in general. However, it is shown in Chapter 5 of [32] that  $\lim_{k \rightarrow \infty} \|\bar{A}_i^k\|^{1/k} = \rho(\bar{A}_i) < 1$ , where  $\rho(\cdot)$  denotes the spectral radius of a given matrix, we know that  $\|\bar{A}_i^j\| < (1 - \epsilon)^j$  for some small  $\epsilon > 0$  and all large  $j$ . Therefore,  $\sum_{j=0}^{\infty} \|\bar{A}_i^j\|^2 < \infty$  and the proposition is proved. ■

### C. Exponential Stability of the Optimal Trajectory

In view of part 3) of Lemma 1, to ensure the stability of the optimal trajectory, it is natural to assume that each subsystem is stabilizable and detectable. Unfortunately, such a natural extension does not hold in the DSLQR case.

As an example, consider the following DSLQR problem:

$$\begin{aligned}
A_1 &= \begin{bmatrix} 0 & 2 \\ 0 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 0.5 \\ 0.5 & 0 \end{bmatrix}, \quad Q_1 = \begin{bmatrix} 100 & 0 \\ 0 & 0 \end{bmatrix}, \quad Q_2 = \begin{bmatrix} 0 & 0 \\ 0 & 100 \end{bmatrix}, \\
x_0 &= \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad Q_f = 0, \text{ and } B_i = R_i = 0, \quad i = 1, 2.
\end{aligned} \tag{17}$$

Let the horizon  $N$  be arbitrary (possibly infinite) and let  $x^*(\cdot)$  be the optimal trajectory of this DSLQR problem with initial condition  $x^*(0) = x_0$ . Notice that both  $A_1$  and  $A_2$  are stable and each subsystem is stabilizable and detectable. However, it can be easily verified that  $x^*(t) = [0, 1]^T$  if  $t$  is even and  $x^*(t) = [2, 0]^T$  otherwise.

To ensure the stability of the optimal trajectory, we introduce the following assumption.

$$(A2) \quad Q_i \succ 0, \forall i \in \mathbb{M}.$$

*Theorem 2:* Under assumptions (A1) and (A2), the  $N$ -horizon optimal trajectory originating from  $z$  at time  $t = 0$ , namely,  $x_{z,N}^*(\cdot)$ , satisfies the following inequalities:

$$\|x_{z,N}^*(t)\|^2 \leq \frac{\beta}{\lambda_Q^-} \gamma^t \|z\|^2, \text{ for } t = 1, \dots, N-1, \text{ and } \|x_{z,N}^*(N)\|^2 \leq \frac{\beta \zeta^2}{\lambda_Q^-} \gamma^{N-1} \|z\|^2, \tag{18}$$

where  $\beta$  is defined in Proposition 1,

$$\gamma = \frac{1}{1 + \lambda_Q^- / \beta} < 1 \quad \text{and} \quad \zeta = \max_{i \in \mathbb{M}} \|A_i - B_i K_i(Q_f)\|. \tag{19}$$

In other words, the optimal trajectory is exponentially stable with decay rate  $\gamma$ .

*Proof:* For simplicity, for  $t = 0, 1, \dots, N$ , define  $\tilde{x}(t) \triangleq x_{z,N}^*(t)$  and  $\tilde{V}_{N-t} \triangleq V_{N-t}(x_{z,N}^*(t))$ . Denote by  $(\tilde{u}(\cdot), \tilde{v}(\cdot))$  the optimal hybrid control sequence corresponding to  $x_{z,N}^*(\cdot)$ . For  $t = 1, \dots, N$ , we have

$$\begin{aligned} \tilde{V}_{N-(t-1)} - \tilde{V}_{N-t} &= L(\tilde{x}(t-1), \tilde{u}(t-1), \tilde{v}(t-1)) \geq \tilde{x}(t-1)^T Q_{\tilde{v}(t-1)} \tilde{x}(t-1) \\ &\geq \lambda_Q^- \|\tilde{x}(t-1)\|^2 \geq \frac{\lambda_Q^-}{\beta} \tilde{V}_{N-(t-1)} \geq \frac{\lambda_Q^-}{\beta} \tilde{V}_{N-t}. \end{aligned}$$

Hence, we have  $\tilde{V}_{N-t} \leq \frac{1}{1+\lambda_Q^-/\beta} \tilde{V}_{N-(t-1)}$  for  $t = 1, \dots, N$ . Therefore,  $\tilde{V}_{N-t} \leq \left(\frac{1}{1+\lambda_Q^-/\beta}\right)^t \tilde{V}_N$ . Obviously, for  $t \leq N-1$ ,  $\tilde{V}_{N-t} \geq \tilde{x}(t)^T Q_{\tilde{v}(t)} \tilde{x}(t) \geq \lambda_Q^- \|\tilde{x}(t)\|^2$ . Thus,

$$\|\tilde{x}(t)\|^2 \leq \frac{1}{\lambda_Q^-} \tilde{V}_{N-t} \leq \frac{1}{\lambda_Q^-} \left(\frac{1}{1+\lambda_Q^-/\beta}\right)^t \tilde{V}_N \leq \frac{\beta}{\lambda_Q^-} \left(\frac{1}{1+\lambda_Q^-/\beta}\right)^t \|z\|^2 = \frac{\beta}{\lambda_Q^-} \gamma^t \|z\|^2. \quad (20)$$

For  $t = N$ , by Theorem 1, we have that  $\tilde{x}(N) = (A_i - B_i K_i(Q_f)) \cdot \tilde{x}(N-1)$  for some  $i \in \mathbb{M}$ . Therefore,  $\|\tilde{x}(N)\|^2 \leq \zeta^2 \|\tilde{x}(N-1)\|^2$ , where  $\zeta$  is defined in (19), and then the desired result follows from (20). ■

#### D. Exponential Convergence of Value Iteration

The main goal of this subsection is twofold: (i) to establish easy-to-check conditions under which  $V_N(z) \rightarrow V^*(z)$  exponentially fast as  $N \rightarrow \infty$ ; (ii) to derive the convergence rate in terms of the subsystem matrices. Some classical results on the convergence of value iterations of general dynamic DP problems can be found in [26]. Most of these results require either a discount factor with magnitude strictly less than 1 or that  $\psi(z) \leq V^*(z)$  for all  $z \in \mathbb{R}^n$ . Neither is true for the general DSLQR problems with nontrivial terminal costs. A more recent convergence result is given by Rantzer in [22], where the abovementioned assumptions are replaced with some other conditions on  $V^*(z)$ . Since the infinite-horizon value function  $V^*(z)$  of the DSLQR problem is usually unknown, the conditions in [22] are not easy to check. In view of these limitations, a further study of the convergence of the value iterations in the DSLQR problems is necessary.

By part 2) of Lemma 1, for the classical LQR problem, if the system is stabilizable, then the value iteration converges to the infinite-horizon value function. For the DSLQR problem, however, Assumption (A1) alone is not enough to ensure the convergence of the value functions. In fact, the value function may not converge even if all the subsystems are stabilizable. For example, consider the DSLQR problem with matrices defined by (17) except that  $Q_f$  is the identity matrix of dimension 2. Although each subsystem is stable, it can be easily seen that  $V_N(x_0)$  is 2 if  $N$  is an odd number and is 1 otherwise. Thus, the limit of  $V_N(x_0)$  as  $N \rightarrow \infty$  does not exist. This example indicates that a stronger condition than (A1) is needed to guarantee the convergence for the DSLQR problem.

In the following we shall show that the value iteration will converge exponentially fast if both (A1) and (A2) are satisfied. The following lemma provides a bound for the difference between two value functions with different horizons and is the key in proving the convergence result.

*Lemma 4:* Let  $N_1$  and  $N_2$  be positive integers such that  $N_1 > N_2$ . For any  $z \in \mathbb{R}^n$ , the difference between the  $N_1$ -horizon value function and the  $N_2$ -horizon value function can be bounded as follows:

$$V_{N_1-N_2}(x_{z,N_1}^*(N_2)) - \psi(x_{z,N_1}^*(N_2)) \leq V_{N_1}(z) - V_{N_2}(z) \leq V_{N_1-N_2}(x_{z,N_2}^*(N_2)) - \psi(x_{z,N_2}^*(N_2)). \quad (21)$$

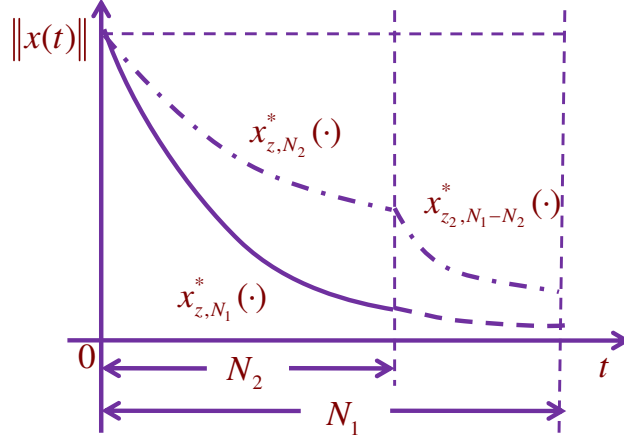


Fig. 2. Illustrating the proof of Lemma 4, where the dashdot line represents the trajectory  $\tilde{x}(\cdot)$  and the solid line represents the trajectory  $\hat{x}(\cdot)$ .

*Proof:* Let  $z_2 = x_{z,N_2}^*(N_2)$ . Define a new  $N_1$ -horizon trajectory  $\tilde{x}(\cdot)$  as

$$\tilde{x}(t) = \begin{cases} x_{z,N_2}^*(t), & t \leq N_2 \\ x_{z_2,N_1-N_2}^*(t - N_2), & N_2 < t \leq N_1 \end{cases} \quad (22)$$

As shown in Fig. 2 (the dashdot line),  $\tilde{x}(\cdot)$  is obtained by first following the  $N_2$ -horizon optimal trajectory and then the  $(N_1 - N_2)$ -horizon optimal trajectory. Let  $(\tilde{u}(\cdot), \tilde{v}(\cdot))$  be the hybrid controls corresponding to  $\tilde{x}$ . Then by the definition of the value function, we have

$$\begin{aligned} V_{N_1}(z) &\leq \sum_{t=0}^{N_1-1} L(\tilde{x}(t), \tilde{u}(t), \tilde{v}(t)) + \psi(\tilde{x}(N_1)) \\ &= \sum_{t=0}^{N_2-1} L(x_{z,N_2}^*(t), u_{z,N_2}^*(t), v_{z,N_2}^*(t)) + \\ &\quad \sum_{t=0}^{N_1-N_2-1} L(x_{z_2,N_1-N_2}^*(t), u_{z_2,N_1-N_2}^*(t), v_{z_2,N_1-N_2}^*(t)) + \psi(x_{z_2,N_1-N_2}^*(N_1 - N_2)) \\ &= V_{N_2}(z) - \psi(x_{z,N_2}^*(N_2)) + V_{N_1-N_2}(z_2) = V_{N_2}(z) - \psi(x_{z,N_2}^*(N_2)) + V_{N_1-N_2}(x_{z,N_2}^*(N_2)) \end{aligned} \quad (23)$$

Equation (23) describes exactly the second inequality in (21). To prove the first one, define an  $N_2$ -horizon trajectory  $\hat{x}(\cdot)$  as the solid line in Fig. 2 by taking the first  $N_2$  steps of  $x_{z,N_1}^*$ , i.e.,  $\hat{x}(t) = x_{z,N_1}^*(t)$  for  $0 \leq t \leq N_2$  and let  $(\hat{u}(\cdot), \hat{v}(\cdot))$  be the corresponding hybrid control sequence. Then

$$\begin{aligned} V_{N_2}(z) &\leq \sum_{t=0}^{N_2-1} L(\hat{x}(t), \hat{u}(t), \hat{v}(t)) + \psi(\hat{x}(N_2)) \\ &= \sum_{t=0}^{N_2-1} L(x_{z,N_1}^*(t), u_{z,N_1}^*(t), v_{z,N_1}^*(t)) + \psi(x_{z,N_1}^*(N_2)) \\ &= V_{N_1}(z) - V_{N_1-N_2}(x_{z,N_1}^*(N_2)) + \psi(x_{z,N_1}^*(N_2)), \end{aligned} \quad (24)$$

where the last step follows from the Bellman's principle of optimality, namely, any segment of an optimal trajectory must be the optimal trajectory joining the two end points of the segment. The desired result follows from (23) and (24).  $\blacksquare$

With a nontrivial terminal cost, the  $N$ -horizon value function  $V_N(z)$  may not be monotone as  $N$  increases. Nevertheless, by Lemma 4, the difference between the value functions  $V_{N_1}(z)$  and  $V_{N_2}(z)$  can be bounded by some quadratic functions of  $x_{z,N_1}^*(N_2)$  and  $x_{z,N_2}^*(N_2)$ . By Theorem 2, we know both quantities converge to zero as  $N_1$  and  $N_2$  grow to infinity. This will guarantee that by choosing  $N_1$  and  $N_2$  large enough, the upper and lower bounds in (21) can be made arbitrarily small. The convergence of the value iteration can thus be established.

*Theorem 3:* Under assumptions (A1) and (A2),  $V_N(z)$  converges exponentially fast for each  $z \in \mathbb{R}^n$  as  $N \rightarrow \infty$ . Furthermore, the convergence is uniform over the unit ball in  $\mathbb{R}^n$  and for any  $N_1 > N_2$ , the difference between the  $N_1$ -horizon value function and the  $N_2$ -horizon value function is bounded above by

$$|V_{N_1}(z) - V_{N_2}(z)| \leq \alpha \gamma^{N_2} \|z\|^2, \quad (25)$$

where  $\alpha = \max\{1, \frac{\zeta^2}{\gamma}\} \cdot \frac{(\beta + \lambda_f^+)\beta}{\lambda_Q^-}$ , with  $\beta$ ,  $\gamma$  and  $\zeta$  defined in (15) and (19).

*Proof:* By Theorem 2, for any  $z \in \mathbb{R}^n$  we have  $\|x_{z,N_2}^*(N_2)\|^2 \leq \frac{\zeta^2 \beta}{\lambda_Q^- \gamma} \gamma^{N_2} \|z\|^2$  and  $\|x_{z,N_1}^*(N_2)\|^2 \leq \frac{\beta}{\lambda_Q^-} \gamma^{N_2} \|z\|^2$ . Hence,

$$\begin{aligned} V_{N_1-N_2}(x_{z,N_2}^*(N_2)) &\leq \beta \|x_{z,N_2}^*(N_2)\|^2 \leq \frac{\zeta^2 \beta^2}{\lambda_Q^- \gamma} \gamma^{N_2} \|z\|^2, \\ \psi(x_{z,N_2}^*(N_2)) &\leq \lambda_f^+ \|x_{z,N_2}^*(N_2)\|^2 \leq \frac{\lambda_f^+ \zeta^2 \beta}{\lambda_Q^- \gamma} \gamma^{N_2} \|z\|^2, \\ V_{N_1-N_2}(x_{z,N_1}^*(N_2)) &\leq \beta \|x_{z,N_1}^*(N_2)\|^2 \leq \frac{\beta^2}{\lambda_Q^-} \gamma^{N_2} \|z\|^2, \\ \psi(x_{z,N_1}^*(N_2)) &\leq \lambda_f^+ \|x_{z,N_1}^*(N_2)\|^2 \leq \frac{\lambda_f^+ \beta}{\lambda_Q^-} \gamma^{N_2} \|z\|^2. \end{aligned}$$

Thus, by Lemma 4 we have

$$|V_{N_1}(z) - V_{N_2}(z)| \leq \max\{1, \frac{\zeta^2}{\gamma}\} \cdot \frac{(\beta + \lambda_f^+)\beta}{\lambda_Q^-} \gamma^{N_2} \|z\|^2.$$

Since  $\gamma < 1$  and the upper bound in the above equation is independent of  $N_1$ , the value function converges exponentially fast for each fixed  $z$ . In addition, the convergence is obviously uniform over the unit ball.  $\blacksquare$

Assumptions (A1) and (A2) together imply the exponential convergence of the value iteration. In general, the limiting function  $V_\infty(z)$  may not coincide with the infinite-horizon value function  $V^*(z)$ . The following Theorem shows that the two functions agree for the DSLQR problem.

*Theorem 4:* Under assumptions (A1) and (A2),  $V_\infty(z) = V^*(z)$  for each  $z \in \mathbb{R}^n$ .

*Proof:* For a finite  $N$ , we know that

$$V_N(z) = \sum_{t=0}^{N-1} L(x_{z,N}^*(t), u_{z,N}^*(t), v_{z,N}^*(t)) + \psi(x_{z,N}^*(N)).$$

By the optimality of  $V^*(z)$ , we have

$$\begin{aligned} V^*(z) &\leq \sum_{t=0}^{N-1} L(x_{z,N}^*(t), u_{z,N}^*(t), v_{z,N}^*(t)) + V^*(x_{z,N}^*(N)) \\ &= V_N(z) - \psi(x_{z,N}^*(N)) + V^*(x_{z,N}^*(N)). \end{aligned}$$

By Theorem 3 and Theorem 2, as  $N \rightarrow \infty$ ,  $V_N(z) \rightarrow V_\infty(z)$ ,  $\psi(x_{z,N}^*(N)) \rightarrow 0$  and  $V^*(x_{z,N}^*(N)) \rightarrow 0$ . Therefore,  $V^*(z) \leq V_\infty(z)$ . We now prove the other direction. Notice that by (A2) we must have  $V^*(z) = \inf_{\pi_\infty \in \Pi_\infty^s} J_{\pi_\infty}(z)$ , where  $\Pi_\infty^s$  denotes the set of all the infinite-horizon stabilizing policies. Let  $\pi_\infty$  be an arbitrary policy in  $\Pi_\infty^s$  and let  $\hat{x}(\cdot)$  and  $(\hat{u}(\cdot), \hat{v}(\cdot))$  be the corresponding trajectory and the hybrid control sequence, respectively. Since  $\hat{x}(t) \rightarrow 0$  as  $t \rightarrow \infty$ , for any  $\epsilon > 0$ , there always exists an  $N_1$  such that  $\psi(\hat{x}(t)) \leq \epsilon$  for all  $t \geq N_1$ . Hence, for all  $N \geq N_1$ ,

$$V_N(z) \leq \sum_{t=0}^{N-1} L(\hat{x}(t), \hat{u}(t), \hat{v}(t)) + \psi(\hat{x}(N)) \leq \sum_{t=0}^{N-1} L(\hat{x}(t), \hat{u}(t), \hat{v}(t)) + \epsilon \leq J_{\pi_\infty}(z) + \epsilon.$$

Let  $N \rightarrow \infty$ , we have  $V_\infty(z) \leq J_{\pi_\infty}(z) + \epsilon$ ,  $\forall \pi_\infty \in \Pi_\infty^s$ . Thus,  $V_\infty(z) \leq V^*(z) + \epsilon$  and the theorem is proved as  $\epsilon$  is arbitrary.  $\blacksquare$

*Remark 3:* The convergence of value iterations has been extensively studied and many results are available [21], [26]. Compared with the previous work, our convergence result derived specially for the DSLQR problem has several distinctions. It allows general terminal cost, which is especially important for finite-horizon DSLQR problems. In addition, the convergence conditions are expressed in terms of the subsystem matrices rather than the infinite-horizon value function, and thus become much easier to verify. Finally, by Theorem 3, for a given tolerance on the optimal cost, the required number of iterations can be computed before the actual computation starts. This provides an efficient means to stop the value iterations with guaranteed suboptimal performance.

## V. EFFICIENT EXACT SOLUTION IN FINITE HORIZON

As discussed at the end of Section III, the main challenge for solving the DSLQR problem lies in the exponential growth of  $|\mathcal{H}_k|$ . However, as indicated by (9), in terms of computing the value function, we only need to keep the matrices in  $\mathcal{H}_k$  that give rise to the minimum of (9) for at least one  $z \in \mathbb{R}^n$ . In other words, although  $\mathcal{H}_k$  is exponentially large, only a small portion of its matrices may be useful for computing the value function. Therefore, we can remove all the other “redundant” matrices to simplify the computation without causing any error. This is the key idea of our efficient algorithm.

### A. Algebraic Redundancy and Equivalent Subsets

To formalize the above idea, we introduce a few definitions.

*Definition 3 (Algebraic Redundancy):* A matrix  $\hat{P} \in \mathcal{H}$  is called (algebraic) *redundant* if for any  $z \in \mathbb{R}^n$ , there exists a matrix  $P \in \mathcal{H}$  such that  $P \neq \hat{P}$  and  $z^T P z \leq z^T \hat{P} z$ .

If  $\hat{P} \in \mathcal{H}$  is redundant, then  $\mathcal{H}$  and  $\mathcal{H} \setminus \{\hat{P}\}$  will define the same value function. In this sense, these two sets are equivalent.

*Definition 4 (Equivalent Sets of p.s.d Matrices):* Let  $\mathcal{H}$  and  $\hat{\mathcal{H}}$  be two sets of p.s.d matrices. The set  $\mathcal{H}$  is called *equivalent* to  $\hat{\mathcal{H}}$ , denoted by  $\mathcal{H} \sim \hat{\mathcal{H}}$ , if  $\min_{P \in \mathcal{H}} z^T P z = \min_{\hat{P} \in \hat{\mathcal{H}}} z^T \hat{P} z, \forall z \in \mathbb{R}^n$ .

Therefore, two sets of p.s.d. matrices are equivalent if they define the same value function of the DSLQR problem. To ease the computation, we are interested in finding an equivalent subset of  $\mathcal{H}_k$  with as few elements as possible.

*Definition 5 (Minimum Equivalent Subset (MES)):* Let  $\mathcal{H}$  and  $\hat{\mathcal{H}}$  be two sets of symmetric p.s.d matrices.  $\hat{\mathcal{H}}$  is called an *equivalent subset* of  $\mathcal{H}$  if  $\hat{\mathcal{H}} \subseteq \mathcal{H}$  and  $\hat{\mathcal{H}} \sim \mathcal{H}$ . Furthermore,  $\hat{\mathcal{H}}$  is called a *minimum equivalent subset* (MES) of  $\mathcal{H}$  if it is the equivalent subset of  $\mathcal{H}$  with the fewest elements. Note that the MES of  $\mathcal{H}$  may not be unique. Denote by  $\Gamma(\mathcal{H})$  one of the MESs of  $\mathcal{H}$ .

The following lemma provides a test for the equivalent subsets of  $\mathcal{H}_k$ .

*Lemma 5:*  $\hat{\mathcal{H}}$  is an equivalent subset of  $\mathcal{H}$  if and only if: (i)  $\hat{\mathcal{H}} \subseteq \mathcal{H}$ ; (ii)  $\forall P \in \mathcal{H}$  and  $\forall z \in \mathbb{R}^n$ , there exists a  $\hat{P} \in \hat{\mathcal{H}}$  such that  $z^T \hat{P} z \leq z^T P z$ .

*Proof:* Straightforward. ■

*Remark 4:* Lemma 5 can be used as an alternative definition of the equivalent subset. Although the original definition is conceptually simpler, the conditions given in this lemma provide a more explicit characterization of the equivalent subset.

### B. Computation of (Minimum) Equivalent Subsets

To simplify the computation at each step  $k$ , we shall prune out as many redundant matrices as possible and obtain an equivalent subset of  $\mathcal{H}_k$  as close as possible to  $\Gamma(\mathcal{H}_k)$ . However, testing whether a matrix in  $\mathcal{H}_k$  is redundant or not is itself a challenging problem. Geometrically, any p.s.d. matrix  $\hat{P}$  defines uniquely an ellipsoid in  $\mathbb{R}^n$ :  $\{x \in \mathbb{R}^n : x^T \hat{P} x \leq 1\}$ . It can be easily verified that  $\hat{P} \in \mathcal{H}_k$  is redundant if and only if its corresponding ellipsoid is completely contained in the union of all the ellipsoids corresponding to the matrices in  $\mathcal{H}_k \setminus \{\hat{P}\}$ . Since the union of ellipsoids is not convex in general, there is no efficient way to verify this geometric condition or equivalently the condition used in Definition 3. Nevertheless, a sufficient condition for a matrix to be redundant can be easily obtained and is given in the following lemma.

*Lemma 6:*  $\hat{P}$  is redundant in  $\mathcal{H}_k$  if there exist nonnegative constants  $\alpha_1, \dots, \alpha_{|\mathcal{H}_k|-1}$  such that  $\sum_{i=1}^{|\mathcal{H}_k|-1} \alpha_i = 1$  and  $\hat{P} \succeq \sum_{i=1}^{|\mathcal{H}_k|-1} \alpha_i P^{(i)}$ , where  $\{P^{(i)}\}_{i=1}^{|\mathcal{H}_k|-1}$  is an enumeration of  $\mathcal{H}_k \setminus \{\hat{P}\}$ .

*Proof:* Straightforward. ■

For given  $\hat{P}$  and  $\mathcal{H}_k$ , the condition in Lemma 6 can be easily verified using various existing convex optimization algorithms [27]. Although Lemma 6 may not identify all the redundant matrices, it can usually eliminate a large portion of them. Based on this lemma, an efficient algorithm (Algorithm 1) is developed to compute an ES for any given set  $\mathcal{H}_k$ . In words, the algorithm simply removes all the matrices that satisfy the condition of Lemma 6 and return the set of the remaining matrices.

Algorithm 1 in general may not return a MES of  $\mathcal{H}_k$ . In fact, when the dimension of the state space is two, there exists an alternative approach that can identify all the redundant matrices and obtain the exact MES of  $\mathcal{H}_k$ . By the homogeneity of the value function, it suffices, in  $\mathbb{R}^2$ , to consider only the points on the unit circle for



**Algorithm 1**

- 1) Denote by  $P^{(i)}$  the  $i^{th}$  matrix in  $\mathcal{H}_k$ . Set  $\mathcal{H}_k^{(1)} = \{P^{(1)}\}$ .
- 2) For each  $i = 2, \dots, |\mathcal{H}_k|$ , if  $P^{(i)}$  satisfies the condition in Lemma 6 with respect to  $\mathcal{H}_k$ , then  $\mathcal{H}_k^{(i)} = \mathcal{H}_k^{(i-1)}$ ; otherwise  $\mathcal{H}_k^{(i)} = \mathcal{H}_k^{(i-1)} \cup \{P^{(i)}\}$ .
- 3) Return  $\mathcal{H}_k^{(|\mathcal{H}_k|)}$ .

testing the redundancy of the matrices in  $\mathcal{H}_k$ . Let  $z(\theta) \triangleq [\cos(\theta), \sin(\theta)]^T$  for  $\theta \in [0, 2\pi)$ . Let  $\{P^{(l)}\}_{l=1}^{|\mathcal{H}_k|}$  be an enumeration of  $\mathcal{H}_k$ . The computation of  $\Gamma(\mathcal{H}_k)$  can be achieved iteratively as follows. Initially, let  $\mathcal{H}_k^{(1)}$  be a set consisting of only one matrix  $P^{(1)}$ . Clearly,  $\mathcal{H}_k^{(1)} = \Gamma(\{P^{(1)}\})$ . At step  $l$ , suppose we have obtained a set  $\mathcal{H}_k^{(l)} = \Gamma(\{P^{(1)}, \dots, P^{(l)}\})$ , namely, the MES of the first  $l$  matrices in  $\mathcal{H}_k$ . The matrices in  $\mathcal{H}_k^{(l)}$  will partition the whole state space into a number of conic regions, within each of which the minimizer  $\arg \min_{P \in \mathcal{H}_k^{(l)}} z(\theta)^T P z(\theta)$  does not depend on  $\theta$ . Each of these regions can be represented by a connected interval of  $\theta$ . Let  $[\theta_{i-1}, \theta_i]$  represent the  $i^{th}$  conic region and let  $P_l^{(i)}$  be the minimizer over this region, namely,  $P_l^{(i)} = \arg \min_{P \in \mathcal{H}_k^{(l)}} z(\theta)^T P z(\theta)$  for all  $\theta \in (\theta_{i-1}, \theta_i]$ . For each  $i \leq |\mathcal{H}_k^{(l)}|$ , we can compare  $z(\theta)^T P^{(l+1)} z(\theta)$  with  $z(\theta)^T P_l^{(i)} z(\theta)$ . If the former is bigger for all  $\theta \in (\theta_{i-1}, \theta_i]$ ,  $P_l^{(i)}$  is still the minimizer of the  $i^{th}$  region. If the latter is always bigger within  $(\theta_{i-1}, \theta_i]$ , then  $P^{(l+1)}$  becomes the new minimizer over the  $i^{th}$  region. If none of these two is true, then we can further divide the interval  $(\theta_{i-1}, \theta_i]$  into some subintervals and record the minimizing matrix in each subinterval. After comparing  $P^{(l+1)}$  with all the matrices in  $\mathcal{H}_k^{(l)}$ , we end up with a new set of intervals and the corresponding minimizing matrices. These minimizing matrices will constitute  $\mathcal{H}_k^{(l+1)}$ , namely, the MES of the first  $l+1$  matrices in  $\mathcal{H}_k$ . If  $P^{(l+1)}$  is redundant with respect to the first  $l+1$  matrices in  $\mathcal{H}_k$ , it cannot beat any  $P_l^{(i)}$  within the corresponding region and  $\mathcal{H}_k^{(l+1)}$  will be exactly the same as  $\mathcal{H}_k^{(l)}$ . On the other hand, if a matrix in  $\mathcal{H}_k^{(l)}$  becomes redundant after considering  $P^{(l+1)}$ , then it will be replaced by  $P^{(l+1)}$  within its minimizing interval and will not be included in  $\mathcal{H}_k^{(l+1)}$ . In this way, we can consider one more matrix in  $\mathcal{H}_k$  at each step and eventually obtain the MES of the whole set  $\mathcal{H}_k$ . The implementation details of the above procedure is summarized in Algorithm 2.

*Remark 5:* It is rather difficult to extend the idea of Algorithm 2 to higher dimensional state spaces because there is no efficient way to characterize the boundaries between adjacent switching regions on the unit sphere. Thus, when  $n > 2$ , we usually still use Algorithm 1 to compute an equivalent subset of  $\mathcal{H}_k$  with not necessarily minimum but sufficiently small number of matrices.

### C. Overall Algorithm in Finite Horizon

We have developed two algorithms to prune out the redundant matrices in  $\mathcal{H}_k$ . A natural question is whether the matrices removed at earlier steps will affect the value iterations later on. This question can be easily answered using the Bellman's optimality principle. Notice that the value iteration at step  $k+1$  only depends on  $V_k(z)$  and that removing the redundant matrices will only change the representation of  $V_k(z)$ , not its actual value. These two facts guarantee that the redundant matrices removed at step  $k$  will not affect any value functions at later steps. The

**Algorithm 2**

- 1) Let  $\Theta^{(1)} = \{0, 2\pi\}$  and  $\mathcal{H}_k^{(1)} = \{P^{(1)}\}$ .
- 2) For  $l \leq |\mathcal{H}_k| - 1$ , given a partition of  $[0, 2\pi]$ ,  $\Theta^{(l)} = \{\theta_0, \dots, \theta_{n_l}\}$  with  $0 = \theta_0 \leq \theta_1 \leq \dots \leq \theta_{n_l} = 2\pi$ , and an ordered set of matrices  $\mathcal{H}_k^{(l)} = \{P_l^{(1)}, \dots, P_l^{(n_l)}\}$ , so that for each  $i = 1, \dots, n_l$ ,  $z(\theta)^T P_l^{(i)} z(\theta) \leq z(\theta)^T P^{(m)} z(\theta)$  for all  $\theta \in [\theta_{i-1}, \theta_i]$  and all  $1 \leq m \leq l$ .
- 3) Compute  $\Theta^{(l+1)}$  and  $\mathcal{H}_k^{(l+1)}$  as follows:
 

**for**  $i = 1$  to  $n_l$  **do**  
    $m = 1$  and  $P_{min}(\theta) = \arg \min_{P \in \{P_l^{(i)}, P^{(l+1)}\}} z(\theta)^T P z(\theta)$   
   **if**  $P_{min}(\theta) = P^{(l+1)}$  for all  $\theta \in [\theta_{i-1}, \theta_i]$  **then**  
      $P_{l+1}^{(m)} = P^{(l+1)}$  and  $m = m + 1$   
   **else if**  $P_{min}(\theta) = P_l^{(i)}$  for all  $\theta \in [\theta_{i-1}, \theta_i]$  **then**  
      $P_{l+1}^{(m)} = P_l^{(i)}$  and  $m = m + 1$   
   **else**  
     Find  $d1$  and  $d2$  such that  $\theta_{i-1} < d1 \leq d2 < \theta_i$  and  $P_{min}(\theta)$  is a constant matrix over intervals  $[\theta_{i-1}, d1]$ ,  $[d1, d2]$  and  $[d2, \theta_i]$ .  
     **if**  $d1=d2$  **then**  
        $P_{l+1}^{(m)} = P_{min}(\theta_{i-1})$ ,  $P_{l+1}^{(m+1)} = P_{min}(d1)$  and  $m = m + 2$   
     **else**  
        $P_{l+1}^{(m)} = P_{min}(\theta_{i-1})$ ,  $P_{l+1}^{(m+1)} = P_{min}(d1)$ ,  
        $P_{l+1}^{(m+2)} = P_{min}(d2)$  and  $m = m + 3$   
     **end if**  
   **end if**  
   **end for**  
    $n_{l+1} = m - 1$
- 4) If  $l < |\mathcal{H}_k|$ , let  $l = l + 1$  and repeat steps 2 and 3, otherwise define  $\hat{\mathcal{H}}_k$  as the set consisting of all the distinct matrices in  $\mathcal{H}_k^{(l)}_{k=0}$  and return  $\hat{\mathcal{H}}_k$ .

following lemma uses this property to embed the ES algorithms in the value iteration. Its basic idea is to remove the redundant matrices after each value iteration and then apply the next value iteration based on the obtained equivalent subset with fewer matrices.

*Lemma 7 (ES Iteration):* Let the sequence of sets  $\{\hat{\mathcal{H}}_k\}_{k=0}^N$  be generated by

$$\hat{\mathcal{H}}_0 = \mathcal{H}_0, \text{ and } \hat{\mathcal{H}}_{k+1} = \text{Algo}(\rho_{\mathbb{M}}(\hat{\mathcal{H}}_k)) \text{ for } 0 \leq k \leq N - 1, \quad (26)$$

where  $\text{Algo}(\mathcal{H})$  denotes the equivalent subset of  $\mathcal{H}$  returned by Algorithm 1 or 2. Then  $\hat{\mathcal{H}}_k \sim \text{Algo}(\mathcal{H}_k)$ .

*Proof:* As explained in the preceding paragraph. A more direct proof can be found in [33]. ■

---

**Algorithm 3** (Algo for Finite-horizon DSLQR Problems)

---

- 1) **Initialization:** Set  $\hat{\mathcal{H}}_0 = Q_f$ .
- 2) **Equivalent Subset Iteration:** Compute  $\{\hat{\mathcal{H}}_k\}_{k=0}^N$  using the iteration (26).
- 3) **Value Function:** The value function at time  $t = N - k$  is given by:  $V_k(z) = \min_{P \in \hat{\mathcal{H}}_k} z^T P z$ .
- 4) **Optimal Strategy:** The optimal  $N$ -horizon policy  $\pi_N^* = \{\xi_N^*(z), \dots, \xi_1^*(z)\}$  is given by:

$$\xi_k^*(z) = \left( -K_{i_k^*(z)}(P_k^*(z))z, i_k^*(z) \right),$$

$$\text{where } \left( P_k^*(z), i_k^*(z) \right) = \arg \min_{P \in \hat{\mathcal{H}}_{k-1}, i \in \mathbb{M}} z^T \rho_i(P) z.$$


---

In summary, to solve the DSLQR problem, we start with the singleton set  $\mathcal{H}_0 = \{Q_f\}$ . Then the SRM is applied to obtain  $\mathcal{H}_1 = \rho_{\mathbb{M}}(\mathcal{H}_0)$ . Some matrices in  $\mathcal{H}_1$  may be redundant. After removing them using Algorithm 1 or 2, we will have  $\hat{\mathcal{H}}_1 = \text{Algo}(\mathcal{H}_1)$ . Next, we should apply the SRM to  $\hat{\mathcal{H}}_1$ , and repeat the whole process until the end of the horizon. This way, we can obtain a sequence of sets  $\{\hat{\mathcal{H}}_k\}_{k=0}^N$ . By Lemma 7,  $\{\hat{\mathcal{H}}_k\}_{k=0}^N$  define the exact value functions of the DSLQR problem. By Theorem 1, the optimal strategies can also be computed based on  $\{\hat{\mathcal{H}}_k\}_{k=0}^N$ . This procedure of solving the finite-horizon DSLQR problem is summarized in Algorithm 3. A distinctive feature of this algorithm is that it computes the exact optimal control strategy without any approximation. Compared with the strategy of enumerating all the possible switching sequences, this algorithm is potentially much more efficient because a large portion of the matrices in  $\mathcal{H}_k$  may end up being removed during the iterations. Furthermore, if the state dimension is  $\mathbb{R}^2$ , all the redundant matrices will be pruned out. Therefore, in this case, our algorithm achieves the minimal complexity in computing the exact optimal strategy of a finite-horizon DSLQR problem.

#### D. Numerical Examples

1) : We first consider the following simple DSLQR problem for which an analytical solution is available for verification purpose.

$$A_1 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad Q_1 = \begin{bmatrix} 100 & 0 \\ 0 & 0 \end{bmatrix}, \quad Q_2 = \begin{bmatrix} 0 & 0 \\ 0 & 100 \end{bmatrix},$$

$$Q_f = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad B_1 = B_2 = 0, \quad R_1 = R_2 = 1, \quad \text{and} \quad N = 10;$$

It can be easily seen that the optimal mode sequence for the initial state  $x_0^{(1)} = [1, 0]^T$  is  $\{2, 1, 2, 1, \dots, 2, 1\}$  and the corresponding optimal cost is 1. If the initial state is  $x_0^{(2)} = [0, 1]^T$ , then the optimal cost remains the same, but the optimal mode sequence would be  $\{1, 2, 1, 2, \dots, 1, 2\}$ . Let

$$\chi_1 = \{r \cdot [\cos(\theta), \sin(\theta)]^T \in \mathbb{R}^2 : r \geq 0 \text{ and } \theta \in [-\pi/4, \pi/4) \cup [3\pi/4, 5\pi/4)\} \text{ and } \chi_2 = \mathbb{R}^2 \setminus \chi_1.$$

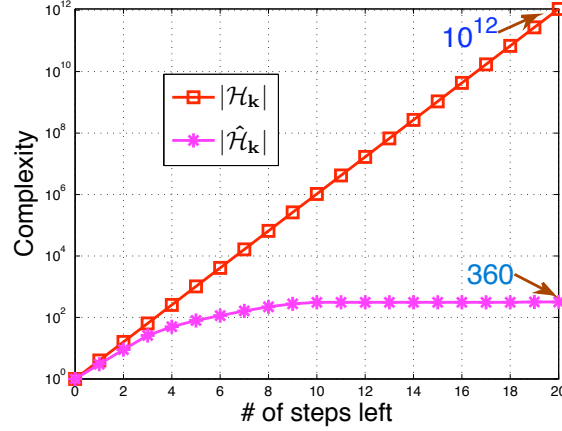


Fig. 3. Complexity of Algorithm 3 for Example V-D.2

By the symmetry of the problem, it can be easily seen that the optimal feedback law is:  $\xi_k^*(x) = (\cdot, 1)$ , if  $x \in \chi_2$  and  $\xi_k^*(x) = (\cdot, 2)$  otherwise.

With the analytical solution in mind, we now demonstrate how to obtain the same result by carrying out Algorithm 3. Initially, we have  $\hat{\mathcal{H}}_0 = \{Q_f\} = I_2$ . Taking the SRM yields

$$\mathcal{H}_1 = \rho_{\mathbb{M}}(\hat{\mathcal{H}}_0) = \{[100, 0; 0, 1], [1, 0; 0, 100]\}.$$

Apparently, none of the two matrices are redundant. Thus,  $\hat{\mathcal{H}}_1 = \text{Algo}(\rho_{\mathbb{M}}(\hat{\mathcal{H}}_0)) = \mathcal{H}_1$ . Proceeding one more step, we have  $\rho_{\mathbb{M}}(\hat{\mathcal{H}}_1) = \{[1, 0; 0, 100], [100, 0; 0, 1], [100, 0; 0, 100], [100, 0; 0, 100]\}$ . Obviously, the last two matrices are redundant. Thus,  $\hat{\mathcal{H}}_2 = \{[1, 0; 0, 100], [100, 0; 0, 1]\}$ . Continuing this process, we have,  $\hat{\mathcal{H}}_k = \{[1, 0; 0, 100], [100, 0; 0, 1]\}$ , for all  $k \leq N$ . Then, using Step 4) of Algorithm 3, the same optimal policy as discussed in the last paragraph can be obtained.

This example shows that although the original SRSs  $\{\mathcal{H}_k\}_{k=0}^N$  grow exponentially fast, their equivalent subsets  $\{\hat{\mathcal{H}}_k\}_{k=0}^N$  can be made rather small and the optimal solution can be easily found using Algorithm 3. For more complex problems, analytical solutions are usually impossible to obtain. However, in many cases, Algorithm 3 can still eliminate many redundant computations and characterize the exact optimal strategy efficiently.

2) : We next consider a more general example with the following matrices:

$$A_1 = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}, A_2 = \begin{bmatrix} 2 & 1 \\ 0 & 0.5 \end{bmatrix}, A_3 = \begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix}, A_4 = \begin{bmatrix} 3 & 1 \\ 0 & 0.8 \end{bmatrix},$$

$$B_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, B_2 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, B_3 = B_1, B_4 = B_2,$$

$$Q_i = Q_f = I_2, R_i = 1, i = 1, \dots, 4, \text{ and } N = 20.$$

This problem can not be solved analytically. However, it can still be efficiently solved using Algorithm 3. As shown in Fig. 3, compared with the brute-force solution with combinatorial complexity of the order  $10^{12}$ , Algorithm 3

requires at most 360 matrices to completely characterize the exact optimal strategy.

## VI. SUBOPTIMAL CONTROL OF DSLQR PROBLEMS

While Algorithm 3 can efficiently solve some DSLQR problems, it may still fail in many other cases. In fact, many DSLQR problems require a prohibitively large number of matrices to characterize their *exact* optimal solutions. Fortunately, suboptimal strategies are often acceptable in practice. In this section, we shall explore the opportunity to further simplify the computation by allowing some small error on the optimal cost.

### A. Numerical Redundancy and $\epsilon$ -Equivalent Subsets

We first generalize the redundancy and ES concepts to allow some error in representing the value functions.

*Definition 6 (Numerical Redundancy):* A matrix  $\hat{P} \in \mathcal{H}_k$  is called (numerically)  $\epsilon$ -redundant with respect to  $\mathcal{H}_k$  if

$$\min_{P \in \mathcal{H}_k \setminus \hat{P}} z^T P z \leq \min_{P \in \mathcal{H}_k} z^T (P + \epsilon I_n) z, \text{ for any } z \in \mathbb{R}^n.$$

*Definition 7 ( $\epsilon$ -ES):* The set  $\mathcal{H}_k^\epsilon$  is called an  $\epsilon$ -Equivalent-Subset ( $\epsilon$ -ES) of  $\mathcal{H}_k$  if

$$\mathcal{H}_k^\epsilon \subset \mathcal{H}_k \quad \text{and} \quad \min_{P \in \mathcal{H}_k} z^T P z \leq \min_{P \in \mathcal{H}_k^\epsilon} z^T P z \leq \min_{P \in \mathcal{H}_k} z^T (P + \epsilon I_n) z, \text{ for any } z \in \mathbb{R}^n.$$

Removing the  $\epsilon$ -redundant matrices may introduce some error for the value function; but the error is no larger than  $\epsilon$  for  $\|z\| \leq 1$ . To simplify the computation, for a given tolerance  $\epsilon$ , we want to prune out as many  $\epsilon$ -redundant matrices as possible. Similar to Lemma 6, the following lemma provides a sufficient condition for testing the  $\epsilon$ -redundancy for a given matrix.

*Lemma 8:*  $\hat{P}$  is  $\epsilon$ -redundant with respect to  $\mathcal{H}_k$  if there exist nonnegative constants  $\alpha_1, \dots, \alpha_{|\mathcal{H}_k|-1}$  such that  $\sum_{i=1}^{|\mathcal{H}_k|-1} \alpha_i = 1$  and  $\hat{P} + \epsilon I_n \succeq \sum_{i=1}^{|\mathcal{H}_k|-1} \alpha_i P^{(i)}$ , where  $\{P^{(j)}\}_{j=1}^{|\mathcal{H}_k|-1}$  is an enumeration of  $\mathcal{H}_k \setminus \{\hat{P}\}$ .

Algorithms 1 and 2 can be easily modified to compute an  $\epsilon$ -ES for a given set  $\mathcal{H}_k$ . Denote the modified algorithms as  $Algo_\epsilon(\cdot)$ , whereas  $Algo(\cdot)$  denotes the original ones. In other words,  $Algo_\epsilon(\mathcal{H}_k)$  is the  $\epsilon$ -ES of  $\mathcal{H}_k$  returned by the modified algorithms. Similar to (26), we can embed the algorithm  $Algo_\epsilon(\cdot)$  in the value iteration by defining the sets  $\{\mathcal{H}_k^\epsilon\}_{k=0}^N$  iteratively as:

$$\mathcal{H}_0^\epsilon = \mathcal{H}_0, \text{ and } \mathcal{H}_{k+1}^\epsilon = Algo_\epsilon(\rho_{\mathbb{M}}(\mathcal{H}_k^\epsilon)), \text{ for } 0 \leq k \leq N-1. \quad (27)$$

The above iteration computes a sequence of relaxed SRSs  $\{\mathcal{H}_k^\epsilon\}_{k=0}^N$ . Using the formulas in Theorem 1, these sets  $\{\mathcal{H}_k^\epsilon\}_{k=0}^N$  also define a sequence of “approximate” value functions and the corresponding feedback policies. Specifically, define  $V_k^\epsilon(z) = \min_{P \in \mathcal{H}_k^\epsilon} z^T P z$ . For  $k = 1, \dots, N$ , let  $\xi_k^\epsilon(\cdot)$  be the feedback law generated by  $V_{k-1}^\epsilon$ , namely,

$$\xi_k^\epsilon(z) = (\mu_k(z), \nu_k(z)) = \arg \min_{(u,v)} \{L(z, u, v) + V_{k-1}^\epsilon(A_v z + B_v u)\}. \quad (28)$$

Following a similar argument as in the proof of Theorem 1, one can easily obtain:

$$\xi_k^\epsilon(z) = \left( -K_{i_k^\epsilon(z)}(P_k^\epsilon(z))z, i_k^\epsilon(z) \right), \quad \text{where } \left( P_k^\epsilon(z), i_k^\epsilon(z) \right) = \arg \min_{P \in \mathcal{H}_{k-1}^\epsilon, i \in \mathbb{M}} z^T \rho_i(P)z. \quad (29)$$

Let  $\pi_N^\epsilon = \{\xi_N^\epsilon, \dots, \xi_1^\epsilon\}$  be the  $N$ -horizon policy generated by  $\{V_k^\epsilon\}_{k=0}^{N-1}$ . Recall that  $\pi_N^* = \{\xi_N^*, \dots, \xi_1^*\}$  denotes the optimal policy generated by the *exact* value functions  $\{V_k\}_{k=0}^{N-1}$ . Typically,  $\pi_N^\epsilon$  is much easier to compute than  $\pi_N^*$  because  $\mathcal{H}_k^\epsilon$  contains much fewer matrices than both  $\mathcal{H}_k$  and  $\text{Algo}(\mathcal{H}_k)$ . However, the relaxation  $\text{Algo}_\epsilon(\cdot)$  introduces an error and this error propagates through the iteration (27). Therefore, to take advantage of the simplicity of  $\pi_N^\epsilon$ , it must be ensured that  $J_{\pi_N^\epsilon}(z)$ , namely, the actual cost associated with  $\pi_N^\epsilon$ , does not deviate too far from the optimal cost  $V_N(z)$ .

### B. Performance Analysis of $\pi_N^\epsilon$

The goal of this subsection is to derive conditions under which the feedback policy  $\pi_N^\epsilon$  is stabilizing and suboptimal. A general  $N$ -horizon policy  $\pi_N$  is called  $\delta$ -suboptimal over a set  $E$  if for any initial state  $x_0 \in E$ , the cost under  $\pi_N$  is within the  $\delta$ -neighborhood of the optimal cost, i.e.,  $|J_{\pi_N}(x_0) - V_N(x_0)| \leq \delta$ . Let  $x_{z,N}^*(\cdot)$  be the optimal trajectory defined in Section IV. Similarly, denote by  $x_{z,N}^\epsilon(\cdot)$  the  $N$ -horizon state trajectory driven by  $\pi_N^\epsilon$  with initial condition  $x_{z,N}^\epsilon(0) = z$ . Define  $V_k^\epsilon(z) = \min_{P \in \mathcal{H}_k^\epsilon} z^T P z$  and

$$\tilde{V}_{k+1}^\epsilon(z) = \min_{u,v} \{L(z, u, v) + V_k^\epsilon(A_v z + B_v u)\}. \quad (30)$$

Following easily from (29), we have

$$\tilde{V}_{k+1}^\epsilon(z) = \min_{P \in \rho_M(\mathcal{H}_k^\epsilon)} z^T P z.$$

According to (27) and the definition of the  $\epsilon$ -ES, we have

$$\tilde{V}_{k+1}^\epsilon(z) \leq V_{k+1}^\epsilon(z) \leq \tilde{V}_{k+1}^\epsilon(z) + \epsilon \|z\|^2. \quad (31)$$

Two important inequalities that are frequently used throughout this subsection are given in the following lemma.

*Lemma 9:* Under assumptions (A1) and (A2), for any integer  $N \geq 0$ , we have

$$V_N(z) \leq V_N^\epsilon(z) \leq V_N(z) + \epsilon \eta \|z\|^2 \quad \text{and} \quad \tilde{V}_N^\epsilon(z) \leq V_N(z) + \epsilon(\eta - 1) \|z\|^2, \quad (32)$$

where  $\eta = \frac{1+(\beta/\lambda_Q^- - 1)\gamma}{1-\gamma}$ .

*Proof:* See Appendix I. ■

As discussed in Section IV-C, under assumptions (A1) and (A2), the optimal trajectory  $x_{z,N}^*(\cdot)$  is exponentially stable. Intuitively speaking, this property should also hold for  $x_{z,N}^\epsilon(\cdot)$  when  $\epsilon$  is sufficiently small. We now derive an upper bound of  $\epsilon$  that guarantees the stability of  $x_{z,N}^\epsilon(\cdot)$ . The following lemma is the key in deriving this upper bound.

*Lemma 10:* Under assumptions (A1) and (A2), the trajectory  $x_{z,N}^\epsilon(\cdot)$  satisfies

$$\begin{aligned} \|x_{z,N}^\epsilon(t)\|^2 &\leq \left(\gamma + \frac{\epsilon\gamma\eta}{\beta}\right)^t \left(\frac{\beta + \epsilon\eta}{\lambda_Q^-}\right) \|z\|^2, \text{ for } t = 1, \dots, N-1, \\ \text{and } \|x_{z,N}^\epsilon(N)\|^2 &\leq \left(\gamma + \frac{\epsilon\gamma\eta}{\beta}\right)^{N-1} \left(\frac{\zeta^2(\beta + \epsilon\eta)}{\lambda_Q^-}\right) \|z\|^2. \end{aligned} \quad (33)$$

where  $\beta, \gamma, \zeta$  and  $\lambda_Q^-$  are the same constants as defined in the last section.

*Proof:* In this proof, we denote  $x_{z,N}^\epsilon(\cdot)$  by  $\hat{x}(\cdot)$  and assume the corresponding hybrid control sequence is  $(\hat{u}(\cdot), \hat{v}(\cdot))$ . By (28), (30) and (31), for each  $t = 1, \dots, N$ , we have

$$\begin{aligned} V_{N-(t-1)}^\epsilon(\hat{x}(t-1)) - V_{N-t}^\epsilon(\hat{x}(t)) &\geq \tilde{V}_{N-(t-1)}^\epsilon(\hat{x}(t-1)) - V_{N-t}^\epsilon(\hat{x}(t)) \\ &= L(\hat{x}(t-1), \hat{u}(t-1), \hat{v}(t-1)) \geq \lambda_Q^- \|\hat{x}(t-1)\|^2 \geq \frac{\lambda_Q^-}{\beta} V_{N-(t-1)}^\epsilon(\hat{x}(t-1)) \\ &\geq \frac{\lambda_Q^-}{\beta} \left( V_{N-(t-1)}^\epsilon(\hat{x}(t-1)) - \epsilon\eta \|\hat{x}(t-1)\|^2 \right) \geq \frac{\lambda_Q^-}{\beta} V_{N-t}^\epsilon(\hat{x}(t)) - \frac{\lambda_Q^- \epsilon\eta}{\beta} \|\hat{x}(t-1)\|^2. \end{aligned}$$

Therefore, for  $t = 1, \dots, N$ ,

$$\begin{aligned} V_{N-t}^\epsilon(\hat{x}(t)) &\leq \gamma \left[ V_{N-(t-1)}^\epsilon(\hat{x}(t-1)) + \frac{\lambda_Q^- \epsilon\eta}{\beta} \|\hat{x}(t-1)\|^2 \right] \leq \left[ \gamma \left( 1 + \frac{\epsilon\eta}{\beta} \right) \right] V_{N-(t-1)}^\epsilon(\hat{x}(t-1)) \\ &\leq \left( \gamma + \frac{\epsilon\gamma\eta}{\beta} \right)^t V_N^\epsilon(z) \leq \left( \gamma + \frac{\epsilon\gamma\eta}{\beta} \right)^t (\beta + \epsilon\eta) \|z\|^2. \end{aligned}$$

Here, the second inequality follows from the fact that  $V_k^\epsilon(z) \geq \lambda_Q^- \|z\|^2$  for  $k \geq 0$ . Using this fact again yields

$$\|\hat{x}(t)\|^2 \leq \left( \gamma + \frac{\epsilon\gamma\eta}{\beta} \right)^t \left( \frac{\beta + \epsilon\eta}{\lambda_Q^-} \right) \|z\|^2, \text{ for } t = 1, \dots, N-1.$$

For  $t = N$ , following the same argument as in the proof of Theorem 2, we have

$$\|\hat{x}(N)\|^2 \leq \left( \gamma + \frac{\epsilon\gamma\eta}{\beta} \right)^{N-1} \left( \frac{\zeta^2(\beta + \epsilon\eta)}{\lambda_Q^-} \right) \|z\|^2$$

With Lemma 10, the following theorem follows immediately.

*Theorem 5:* Under (A1) and (A2), if  $\epsilon < \frac{(1-\gamma)\beta}{\gamma\eta}$ , the policy  $\pi_N^\epsilon$  is stabilizing.

We now derive an upper bound for the actual cost associated with the policy  $\pi_N^\epsilon$

*Theorem 6:* Under assumptions (A1) and (A2),  $J_{\pi_N^\epsilon}(z) \leq V_N(z) + \epsilon(\eta - 1)\|z\|^2$ . for any  $z \in \mathbb{R}^n$  and  $N \geq 0$ .

*Proof:* Let  $\hat{x}(\cdot)$  and  $(\hat{u}(\cdot), \hat{v}(\cdot))$  be the same as in the proof of Lemma 10. By (28) and (30), we have  $L(\hat{x}(t), \hat{u}(t), \hat{v}(t)) = \tilde{V}_{N-t}^\epsilon(\hat{x}(t)) - V_{N-(t+1)}^\epsilon(\hat{x}(t+1))$  for each  $t = 0, \dots, N-1$ . Therefore,

$$\begin{aligned} J_{\pi_N^\epsilon}(z) &= \sum_{t=0}^{N-1} L(\hat{x}(t), \hat{u}(t), \hat{v}(t)) + \psi(\hat{x}(N)) \\ &= \sum_{t=0}^{N-1} [\tilde{V}_{N-t}^\epsilon(\hat{x}(t)) - V_{N-(t+1)}^\epsilon(\hat{x}(t+1))] + \psi(\hat{x}(N)) \\ &= \tilde{V}_N^\epsilon(z) + \sum_{t=1}^{N-1} [\tilde{V}_{N-t}^\epsilon(\hat{x}(t)) - V_{N-t}^\epsilon(\hat{x}(t))] + [\psi(\hat{x}(N)) - V_0^\epsilon(\hat{x}(N))]. \end{aligned}$$

Since by definition  $\psi(z) = V_0^\epsilon(z)$  and  $\tilde{V}_{N-t}^\epsilon(z) \leq V_{N-t}^\epsilon(z)$  for any  $z \in \mathbb{R}^n$  and  $t = 1, \dots, N-1$ , we have

$$J_{\pi_N^\epsilon}(z) \leq \tilde{V}_N^\epsilon(z) \leq V_N(z) + \epsilon(\eta - 1)\|z\|^2.$$

*Remark 6:* Notice that the error function  $\epsilon(\eta - 1)\|z\|^2$  does not depend on the horizon  $N$ . This property plays a crucial role in deriving the suboptimal policies for the infinite-horizon DSLQR problems.

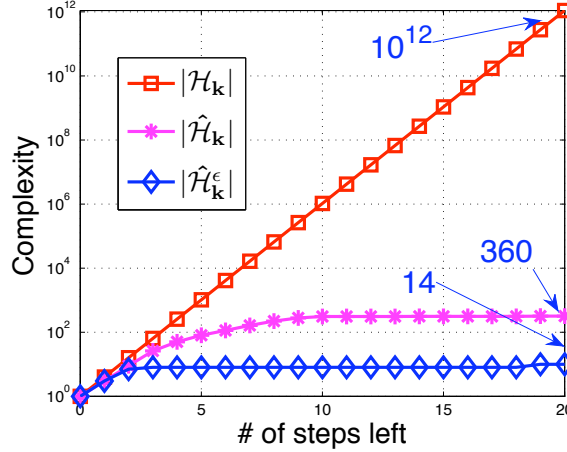


Fig. 4. Complexity comparison between Algorithm 3 and Algorithm 4 with  $\delta = 10^{-3}$ .

*Corollary 1:* Under the same conditions as in Theorem 6,  $\pi_N^\epsilon$  is  $\delta$ -suboptimal over the unit ball if  $\epsilon \leq \frac{\delta}{\eta - 1}$ .

Based on our analysis in this subsection, Algorithm 3 can be easily modified to compute a  $\delta$ -suboptimal policy within the unit ball.

---

**Algorithm 4** (Suboptimal Control in Finite Horizon)

---

- 1) **Initialization:** Specify an error tolerance  $\delta$ . Let  $\epsilon = \frac{\delta}{\eta - 1}$  and set  $\mathcal{H}_0^\epsilon = Q_f$
- 2) **Approximate Subset Iteration:** Perform iteration (27) over the whole horizon  $N$ .
- 3) **Suboptimal Strategy:** The suboptimal  $N$ -horizon policy  $\pi_N^\epsilon = \{\xi_N^\epsilon(x), \dots, \xi_1^\epsilon(x)\}$  is given by:

$$\xi_k^\epsilon(x) = \left( -K_{i_k^\epsilon(x)}(P_k^\epsilon(x))x, i_k^\epsilon(x) \right),$$

$$\text{where } \left( P_k^\epsilon(x), i_k^\epsilon(x) \right) = \arg \min_{P \in \mathcal{H}_{k-1}^\epsilon, i \in \mathbb{M}} x^T \rho_i(P)x.$$


---

### C. Example V-D.2 Revisited

For comparison, we test Algorithm 4 using the same example as described in Section V-D.2. As shown in Fig. 4, instead of characterizing the optimal solution exactly using 360 matrices, with the relaxation  $\delta = 10^{-3}$ , we can obtain a  $\delta$ -suboptimal strategy using only 14 matrices. It is worth mentioning that for many other DSLQR problems, Algorithm 3 may still suffer from combinatorial complexity. In these cases, relaxing the accuracy using Algorithm 4 becomes necessary.

### D. Extension to Large or Infinite Horizon

The numerical redundancy has greatly simplified the computation of each step of the value iteration. However, the overall computation may still grow out of hand when the horizon  $N$  is very large or even infinite. The convergence



property of the value iterations derived in Section IV-D becomes crucial in dealing with these cases, because it allows us to terminate the iterations at some early steps instead of carrying out the iterations over the whole horizon.

It is natural to solve the infinite-horizon case in a divide-and-conquer manner, namely, by applying Algorithm 4 to a reasonably large size of subhorizon,  $m$ , and then extending the obtained strategy periodically. We now show that, by choosing proper  $m$  and  $\epsilon$ , such a periodic policy can indeed achieve an arbitrary suboptimal performance. Let  $\hat{\pi}_m^\epsilon = \{\hat{\xi}_m^\epsilon, \dots, \hat{\xi}_1^\epsilon\}$  be the  $m$ -horizon policy returned by Algorithm 4 with  $Q_f = 0$ . It follows from Theorem 6 that

$$J_{\hat{\pi}_m^\epsilon}(z) \leq V_m^0(z) + \epsilon(\eta - 1)\|z\|^2 \leq V^*(z) + \epsilon(\eta - 1)\|z\|^2, \quad (34)$$

where  $V_m^0(z)$  denotes the  $m$ -horizon value function with  $Q_f = 0$ . For  $m \geq 2$ , let  $\pi_\infty^{\epsilon, m}$  be the periodic extension of the first  $m - 1$  terms of  $\hat{\pi}_m^\epsilon$ , i.e.<sup>1</sup>,

$$\pi_\infty^{\epsilon, m} = \{\hat{\xi}_m^\epsilon, \dots, \hat{\xi}_2^\epsilon, \hat{\xi}_m^\epsilon, \dots, \hat{\xi}_2^\epsilon, \dots\}. \quad (35)$$

We first establish conditions under which the specially constructed policy  $\pi_\infty^{\epsilon, m}$  is stabilizing.

*Theorem 7:* Under assumptions (A1) and (A2), if  $\epsilon < \frac{(1-\gamma)\beta}{\gamma\eta}$  and  $m > \frac{\ln \lambda_Q^- - \ln(\beta + \epsilon\eta)}{\ln(\beta\gamma + \epsilon\gamma\eta) - \ln\beta} + 1$ , then  $\pi_\infty^{\epsilon, m}$  is exponentially stabilizing.

*Proof:* Denote by  $\hat{x}(\cdot)$  the trajectory generated by the policy  $\pi_\infty^{\epsilon, m}$  with initial condition  $\hat{x}(0) = z$ . Let

$$c_m = \left( \gamma + \frac{\epsilon\gamma\eta}{\beta} \right)^{m-1} \left( \frac{\beta + \epsilon\eta}{\lambda_Q^-} \right). \quad (36)$$

It can be easily verified that under our assumptions,  $c_m$  is strictly smaller than 1. By inequality (33), we have  $\|\hat{x}(k(m-1))\|^2 \leq c_m \|\hat{x}((k-1)(m-1))\|^2$  for all  $k \geq 1$ . Thus,  $\|\hat{x}(\cdot)\|^2$  must decrease by a factor of  $c_m < 1$  in every  $m - 1$  steps. It follows that the policy  $\pi_\infty^{\epsilon, m}$  is exponentially stabilizing. ■

We now derive a bound for the error between the actual cost  $J_{\pi_\infty^{\epsilon, m}}(z)$  and the optimal cost  $V^*(z)$ .

*Theorem 8:* Under the same conditions as in Theorem 7, we have

$$V^*(z) \leq J_{\pi_\infty^{\epsilon, m}}(z) \leq V^*(z) + \frac{c_m\beta + \epsilon(\eta - 1)}{1 - c_m} \|z\|^2, \quad (37)$$

where  $c_m$  is defined in (36).

*Proof:* Obviously,  $V^*(z) \leq J_{\pi_\infty^{\epsilon, m}}(z)$  as  $\pi_\infty^{\epsilon, m}$  is an infinite-horizon policy. Let  $\hat{x}(\cdot)$  be the system trajectory generated by the policy  $\pi_\infty^{\epsilon, m}$  starting from  $\hat{x}(0) = z$ . Define  $z_i = \hat{x}(i \cdot (m-1))$  for  $i = 0, 1, \dots$ . Let  $\tilde{\pi} \triangleq \{\hat{\xi}_m^\epsilon, \dots, \hat{\xi}_2^\epsilon\}$  be the first  $m - 1$  terms of  $\hat{\pi}_m^\epsilon$ . Then by (34),

$$J_{\pi_\infty^{\epsilon, m}}(z) = \sum_{i=0}^{\infty} J_{\tilde{\pi}}(z_i) \leq \sum_{i=0}^{\infty} J_{\hat{\pi}_m^\epsilon}(z_i) \leq \sum_{i=0}^{\infty} [V^*(z_i) + \epsilon(\eta - 1)\|z_i\|^2].$$

By inequality (33),  $\|z_i\|^2 \leq c_m^i \|z\|^2$ , where  $c_m < 1$  is defined in (36). Therefore,  $J_{\pi_\infty^{\epsilon, m}}(z) \leq V^*(z) + \frac{c_m\beta + \epsilon(\eta - 1)}{1 - c_m} \|z\|^2$  for any initial state  $z$ . ■

<sup>1</sup>As can be seen from Lemma 10, by using only the first  $m - 1$  terms of  $\hat{\pi}_m^\epsilon$  in constructing  $\pi_\infty^{\epsilon, m}$ , we can obtain a better bound for the convergence of the closed-loop trajectory.

With the above result, we can easily derive a lower bound for  $m$  that guarantees the  $\delta$ -suboptimality of  $\pi_\infty^{\epsilon, m}$  for an arbitrary  $\delta > 0$ .

*Corollary 2:* Suppose that the conditions in Theorem 7 hold. For any  $\delta > 0$ , if we further have  $\epsilon < \frac{\delta}{\eta-1}$  and

$$m > m_{\infty}^{\delta, \epsilon} \triangleq \frac{\ln[\delta - \epsilon(\eta - 1)]\lambda_Q^- - \ln(\beta + \delta)(\beta + \epsilon\eta)}{\ln(\beta\gamma + \epsilon\gamma\eta) - \ln\beta} + 1, \quad (38)$$

then the policy  $\pi_\infty^{\epsilon, m}$  is  $\delta$ -suboptimal over the unit ball.

*Proof:* The proof follows immediately from Lemma 10 and Theorem 8.  $\blacksquare$

For a given tolerance  $\delta$  on the optimal cost, we only need to perform  $m_{\infty}^{\delta, \epsilon}$  steps of the approximate value iterations (27). The obtained value functions  $\{V_k^\epsilon(z)\}_{k=0}^{m_{\infty}^{\delta, \epsilon}}$  characterize the  $m_{\infty}^{\delta, \epsilon}$ -horizon feedback policy  $\hat{\pi}_m^\epsilon$  whose  $m_{\infty}^{\delta, \epsilon} - 1$  steps can be used periodically to construct an infinite-horizon policy  $\pi_\infty^{\epsilon, m}$ . By Corollary 2, such a periodic policy is guaranteed to be  $\delta$ -suboptimal over the unit ball. This idea can also be used when the horizon is large but finite. Denote by  $[\pi_\infty^{\epsilon, m}]_N$  the  $N$ -horizon truncation of the policy  $\pi_\infty^{\epsilon, m}$ , i.e.,  $[\pi_\infty^{\epsilon, m}]_N(t) = \pi_\infty^{\epsilon, m}(t)$  for  $t = 0, \dots, N-1$ . Similar performance bound as in Theorem 8 can be derived for  $[\pi_\infty^{\epsilon, m}]_N(t)$ .

*Theorem 9:* Under the same conditions as in Theorem 7, for any  $N \geq m$ , we have

$$V_N(z) \leq J_{[\pi_\infty^{\epsilon, m}]_N}(z) \leq V_N(z) + \left[ \frac{c_m\beta + \epsilon(\eta - 1)}{1 - c_m} + \lambda_f^+ c_m^{N_m} \right] \|z\|^2, \quad (39)$$

where  $c_m$  is defined in (36) and  $N_m = \lfloor N/(m-1) \rfloor$ .

*Proof:* Denote by  $\hat{x}(\cdot)$  the closed-loop trajectory generated by the policy  $[\pi_\infty^{\epsilon, m}]_N$ . Let  $\tilde{\pi}$  and  $z_i$  be the same as in the proof of Theorem 8. Then by (34),

$$J_{[\pi_\infty^{\epsilon, m}]_N}(z) - \psi(\hat{x}(N)) \leq \sum_{i=0}^{N_m+1} J_{\tilde{\pi}}(z_i) \leq \sum_{i=0}^{N_m+1} J_{\hat{\pi}_m^\epsilon}(z_i) \leq \sum_{i=0}^{N_m+1} [V_m^0(z_i) + \epsilon(\eta - 1)\|z_i\|^2]$$

Notice that  $V_m^0(z) \leq V_N(z)$ ,  $V_m^0(z_i) \leq V^*(z_i)$  and  $V^*(z_i) \leq \beta\|z_i\|^2 \leq \beta c_m^i \|z\|^2$ , by adding some small positive terms, we have

$$J_{[\pi_\infty^{\epsilon, m}]_N}(z) - \psi(\hat{x}(N)) \leq V_N(z) + \sum_{i=1}^{\infty} \beta c_m^i \|z\|^2 + \sum_{i=0}^{\infty} \epsilon(\eta - 1) c_m^i \|z\|^2. \quad (40)$$

By our hypotheses, we have  $c_m < 1$ . Thus,  $J_{[\pi_\infty^{\epsilon, m}]_N}(z) - \psi(\hat{x}(N)) \leq V_N(z) + \frac{c_m\beta + \epsilon(\eta-1)}{1-c_m} \|z\|^2$ . Considering  $\psi(\hat{x}(N)) \leq \lambda_f^+ \|\hat{x}(N)\|^2 \leq \lambda_f^+ c_m^{N_m} \|z\|^2$ , the desired result is proved.  $\blacksquare$

*Corollary 3:* Suppose the conditions in Theorem 7 hold. For any  $\delta > 0$ , if we further have  $\epsilon < \frac{\delta}{\eta-1}$  and

$$N \geq m > m_N^{\delta, \epsilon} \triangleq \frac{\ln[\delta - \epsilon(\eta - 1)]\lambda_Q^- - \ln(\beta + \delta + \lambda_f^+)(\beta + \epsilon\eta)}{\ln(\beta\gamma + \epsilon\gamma\eta) - \ln\beta} + 1, \quad (41)$$

then the  $N$ -horizon policy  $[\pi_\infty^{\epsilon, m}]_N$  is  $\delta$ -suboptimal over the unit ball.

*Remark 7:* In deriving (41) from (39), we have replaced  $c_m^{N_m}$  by its upper bound 1. As a result, the bound in (41) does not depend on  $N$ . Its main difference from (38) is the  $\lambda_f^+$  term which accounts for the final cost.

From the above analysis, for large or infinite  $N$ , a  $\delta$ -suboptimal  $N$ -horizon policy can be obtained as follows. First, find the largest  $\epsilon$  that satisfies all the conditions in Corollary 2. Second, let  $m = m_{\infty}^{\delta, \epsilon}$  or  $m = m_N^{\delta, \epsilon}$  depending on whether  $N$  is infinite or not. Third, compute the  $m$ -horizon suboptimal policy  $\hat{\pi}_m^\epsilon$  using Algorithm 4 with

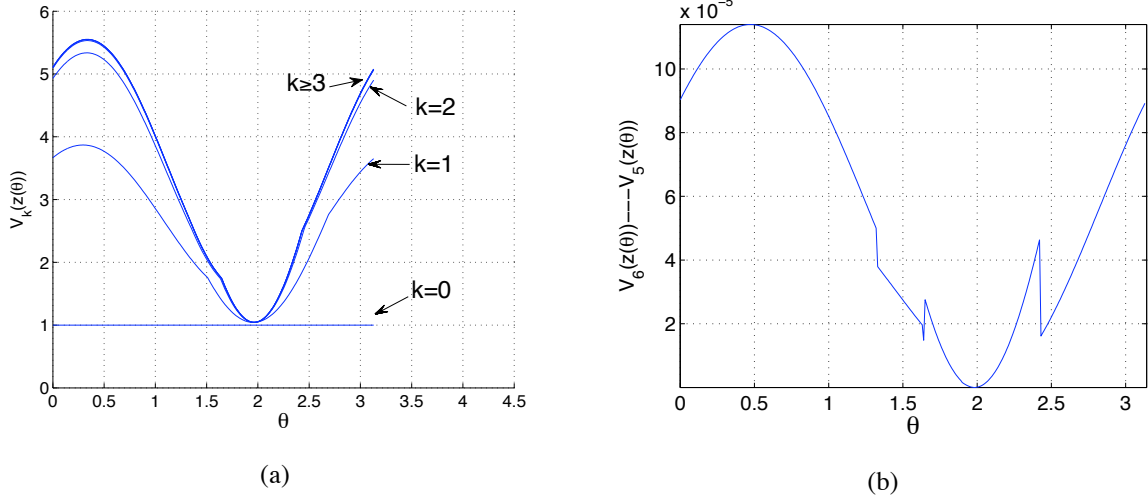


Fig. 5. Convergence results for Ex VI-E.1. (a) Convergence of the Value function. (b) Difference between the last two iterations.

$Q_f = 0$ . Finally, use  $\hat{\pi}_m^\epsilon$  to construct  $\pi_\infty^{\epsilon,m}$  based on (35) and keep the first  $N$  steps of  $\pi_\infty^{\epsilon,m}$  to obtain an  $N$ -horizon policy  $[\pi_\infty^{\epsilon,m}]_N^2$ . By Corollary 2 or 3,  $[\pi_\infty^{\epsilon,m}]_N$  is guaranteed to be  $\delta$ -suboptimal over the unit ball. The above procedure of constructing the suboptimal control policy is summarized in Algorithm 5. Note that in this procedure, we have assumed that  $N > m$ . If this is not the case, we should still use Algorithm 4 to carry out the approximate iterations (27) for the whole horizon  $N$ .

---

**Algorithm 5** (Large or infinite Horizon Suboptimal Control)

---

- 1) **Initialization:** Specify an error tolerance  $\delta$ . Let  $\epsilon = \max\{\frac{\delta}{\eta-1}, \frac{\beta(1-\gamma)}{\gamma\eta}\}$ .
  - 2) **# of iterations steps:** If  $N = \infty$ , let  $m = m_\infty^{\delta,\epsilon}$ ; otherwise, let  $m = m_N^{\delta,\epsilon}$ . If  $N \leq m$ , stop and turn to Algorithm 4.
  - 3) **m-horizon Policy:** Calculate the  $m$ -horizon suboptimal policy  $\hat{\pi}_m^\epsilon$  using Algorithm 4 with  $Q_f = 0$ .
  - 4) **Horizon Extension:** Construct  $\pi_\infty^{\epsilon,m}$  from  $\hat{\pi}_m^\epsilon$  using (35) and keep its first  $N$  terms to obtain  $[\pi_m^\infty]_N$ .
- 

*Remark 8:* The analytical bounds  $m_\infty^{\delta,\epsilon}$  and  $m_N^{\delta,\epsilon}$  derived in (38) and (41) may be conservative for some applications. An alternative approach is to start from a smaller value for  $m$  in Step 2) of Algorithm 5 and gradually increase its value until the performance saturates. Our analysis guarantees that this tentative procedure can eventually reach any pre-specified suboptimal performance by gradually increasing  $m$ .

### E. More Examples

<sup>2</sup>If  $N$  is infinite, the policy  $[\pi_\infty^{\epsilon,m}]_N$  would be the same as  $\pi_\infty^{\epsilon,m}$

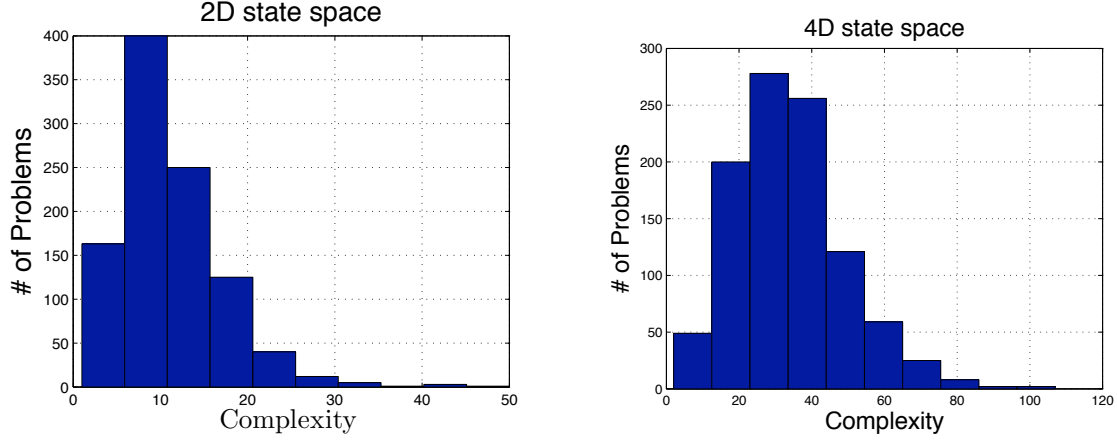


Fig. 6. Complexity distributions of the random examples.

1) : First consider a simple DSLQR problem with control horizon  $N = 1000$  and two second-order subsystems:

$$A_1 = \begin{bmatrix} 2 & 1 \\ 0 & 1 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 2 & 1 \\ 0 & 0.5 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}.$$

Suppose that the state and control weights are  $Q_1 = Q_2 = I_2$  and  $R_1 = R_2 = 1$ , respectively. Both subsystems are unstable but controllable. Algorithm 5 is applied to solve this DSLQR problem. With  $\delta = 10^{-3}$ , the upper bound of the required number of iterations is  $m_N^{\delta, \epsilon} = 56$ , while as observed in the simulation, the value function already converges in 6 steps. Since  $V_k(z)$  is homogeneous and symmetric, in Fig. 5, we plot the evolution of the value functions on the upper half of the unit circle, i.e. the points of the form  $z(\theta) = [\cos(\theta), \sin(\theta)]^T$  with  $\theta \in [0, \pi]$ . The number of matrices in  $\mathcal{H}_k^\epsilon$  at each step  $k$  is listed in Table I. It can be seen that  $|\mathcal{H}_k^\epsilon|$  is indeed very small and stays at the maximum value 5 as opposed to growing exponentially as  $k$  increases.

TABLE I

$|\mathcal{H}_k^\epsilon|$  FOR EXAMPLE VI-E.1

$k$	1	2	3	4	5	6
$ \mathcal{H}_k^\epsilon $	2	4	5	5	5	5

2) *Random Examples*: To further demonstrate its effectiveness, Algorithm 5 is tested by two sets of randomly generated DSLQR problems. The first set consists of 1000 two-dimensional DSLQR problems with 10 subsystems. The second set consists of 1000 four-dimensional DSLQR problems with 4 subsystems. For both sets, the control horizon  $N$  is infinite and  $\delta = 10^{-3}$ . All of these problems are successfully solved by Algorithm 5 and the distributions of the complexity, namely, the maximum numbers of matrices required for characterizing the suboptimal policy, are plotted in Fig. 6. It can be seen from the figure that all of the two-dimensional problems require less than 50 matrices and a majority of them only need less than 15 matrices. However, a majority of the four-dimensional problems need about 40 matrices and some of them may need more than 100 matrices. The complexity

of Algorithm 5 depends heavily on the state dimension. In a higher dimensional state space, a larger relaxation  $\delta$  is usually needed in order to retain a high computational speed.

## VII. CONCLUSION

We have proved that the value function of the DSLQR problem is piecewise quadratic and can be characterized by a finite number of p.s.d. matrices in the switched Riccati sets  $\mathcal{H}_k$ . These matrices can be obtained analytically through the switched Riccati mapping. The main challenge of solving the DSLQR problem is on the exponential growth of  $|\mathcal{H}_k|$ . Three types of simplifications have been proposed to overcome this difficulty. First, some matrices in  $\mathcal{H}_k$  are algebraically redundant and can be directly removed without affecting the value function and the optimal strategy at all. Second, many matrices in  $\mathcal{H}_k$  are numerically redundant in the sense that removing them will only incur a small error on the value function. Third, under some mild conditions, the value function converges exponentially fast to the infinite-horizon value function. Thus, we can terminate the value iteration at some early steps with satisfactory numerical performance. Efficient algorithms based on one or more of the above ideas are developed to achieve various design goals. Analytical conditions have been derived to guarantee the stability and suboptimality of the obtained policy. The results of this paper can be used to study many other problems of the switched linear systems, such as the switched Kalman filtering problem, the switched LQG problem, and the switched receding horizon control problem, etc. All of these will be our future research directions.

## APPENDIX I

### PROOF OF LEMMA 9

*Lemma 11:* With the same notations as in Section VI, we have

$$V_N^\epsilon(z) \leq V_N(z) + \epsilon \sum_{t=0}^{N-1} \|x_{z,N}^*(t)\|^2. \quad (42)$$

*Proof:* By definition,  $V_0^\epsilon(z) = V_0(z)$ . Thus, the desired inequality holds for  $N = 0$ . Now suppose it is true for a general  $N \geq 0$ , we shall show it is also the case for  $N + 1$ . Substituting (42) into (30) with  $k = N$ , we have

$$\tilde{V}_{N+1}^\epsilon \leq \min_{u,v} \{L(z, u, v) + V_N(A_v z + B_v u) + \sum_{t=0}^{N-1} \epsilon \|x_{A_v z + B_v u, N}^*(t)\|^2\} \quad (43)$$

Let  $(\hat{u}, \hat{v}) = \xi_{N+1}^*(z) = (\mu_{N+1}^*(z), \nu_{N+1}^*(z))$ , i.e.,  $(\hat{u}, \hat{v})$  is the first step of the  $(N + 1)$ -horizon optimal policy at state  $z$ . Thus, we have  $A_{\hat{v}} z + B_{\hat{v}} \hat{u} = x_{z, N+1}^*(1)$ . By Bellman's principle of optimality, we know that the  $N$ -horizon optimal trajectory starting from  $x_{z, N+1}^*(1)$  coincides with the last  $N$  steps of the  $(N + 1)$ -horizon optimal trajectory originating from  $z$ . Therefore, under this  $(\hat{u}, \hat{v})$ , we have  $x_{A_{\hat{v}} z + B_{\hat{v}} \hat{u}, N}^*(t) = x_{z, N+1}^*(t+1)$  for each  $t = 0, \dots, N-1$ . In addition, by the definition of  $\xi_{N+1}^*$ , we also have  $L(z, \hat{u}, \hat{v}) + V_N(A_{\hat{v}} z + B_{\hat{v}} \hat{u}) = V_{N+1}(z)$ . Notice that this  $(\hat{u}, \hat{v})$  is just one choice of all the possible hybrid controls in (43), hence,

$$\tilde{V}_{N+1}^\epsilon \leq V_{N+1}(z) + \epsilon \sum_{t=1}^N \|x_{z, N+1}^*(t)\|^2.$$

Then it follows from (31) that

$$V_{N+1}^\epsilon(z) \leq \tilde{V}_{N+1}^\epsilon(z) + \epsilon \|z\|^2 \leq V_{N+1}(z) + \epsilon \sum_{t=0}^N \|x_{z,N+1}^*(t)\|^2.$$

Thus, the inequality also holds for  $N + 1$ . ■

*Proof:* [Proof of Lemma 9] By Theorem 2 and some simple computations, we have  $\sum_{t=0}^N \|x_{z,N+1}^*(t)\|^2 \leq \eta \|z\|^2$  with  $\eta = \frac{1+(\beta/\lambda_Q^- - 1)\gamma}{1-\gamma}$ . The desired result then follows directly from Lemma 11 and inequality (31). ■

## REFERENCES

- [1] T. Geyer, G. Papafotiou, and M. Morari. Model Predictive Control in Power Electronics: A Hybrid Systems Approach. In *IEEE Conference on Decision and Control*, Seville, Spain, December 2005.
- [2] A.G. Beccuti, G. Papafotiou, and M. Morari. Optimal Control of the Boost dc-dc Converter. In *IEEE Conference on Decision and Control*, Seville, Spain, December 2005.
- [3] W. Zhang, J. Hu, and Y.-H. Lu. Optimal Power Modes Scheduling Using Hybrid Systems. In *Proceedings of the American Control Conference*, New York City, NY, July 2007.
- [4] W. Zhang and J. Hu. Dynamic buffer management using optimal control of hybrid systems. *Automatica*, (7):1831–1840, Jul 2008.
- [5] C. G. Cassandras, D. L. Pepyne, and Y. Wardi. Optimal control of a class of hybrid systems. *IEEE Transactions on Automatic Control*, 46(3):398–415, 2001.
- [6] J. Hespanha, S. Bohacek, K. Obraczka, and J. Lee. Hybrid modeling of TCP congestion control. In Maria Domenica Di Benedetto and Alberto Sangiovanni-Vincentelli, editors, *Hybrid Systems: Computation and Control*, number 2034 in LNCS, pages 291–304. Springer-Verlag, Berlin, March 2001.
- [7] D. Liberzon and A. S. Morse. Basic problems in stability and design of switched systems. *IEEE Control Systems Magazine*, 19(5):59–70, 1999.
- [8] Daniel Liberzon. *Switching in Systems and Control*. Systems & Control: Foundations & Applications. Birkhauser, Boston, 2003.
- [9] R. DeCarlo, M. Branicky, S. Pettersson, and B. Lennartson. Perspectives and results on the stability and stabilizability of hybrid systems. *Proceedings of IEEE, Special Issue on Hybrid Systems*, 88(7):1069–1082, 2000.
- [10] C. Cai, A.R. Teel, and R. Goebel. Smooth Lyapunov functions for hybrid systems part I: Existence is equivalent to robustness. *IEEE Transactions on Automatic Control*, (7):1246–1277, 2007.
- [11] S. Pettersson. Synthesis of switched linear systems. In *IEEE Conference on Decision and Control*, pages 5283– 5288, Maui, HI, Dec 2003.
- [12] A. Bemporad, A. Giua, and C. Seatzu. Synthesis of switched linear systems. In *IEEE Conference on Decision and Control*, pages 5283– 5288, Maui, HI, Dec 2003.
- [13] J. Hespanha and A. S. Morse. Switching between stabilizing controllers. *Automatica*, Nov. 2002.
- [14] J.-W. Lee and P. P. Khargonekar. Optimal output regulation for discrete-time switched and markovian jump linear systems. *SIAM J. Control Optim.*, 47(1):40–72, 2008.
- [15] X. Xu and P.J. Antsaklis. Optimal control of switched systems based on parameterization of the switching instants. *IEEE Transactions on Automatic Control*, 49(1):2–16, 2004.
- [16] M. Egerstedt, Y. Wardi, and F. Delmotte. Optimal control of switching times in switched dynamical systems. *IEEE Transactions on Automatic Control*, 51(1):110–115, 2006.
- [17] X. Xu and P. Antsaklis. A dynamic programming approach for optimal control of switched systems. In *Proceedings of the IEEE Conference on Decision and Control*, pages 1822–1827, Sydney, Australia, December 2000.
- [18] A. Rantzer and M. Johansson. Piecewise linear quadratic optimal control. *IEEE Transactions on Automatic Control*, Apr. 2000.
- [19] A. Bemporad, A. Giua, and C. Seatzu. Synthesis of state-feedback optimal controllers for continuous-time switched linear systems. In *Proceedings of the IEEE Conference on Decision and Control*, pages 3182– 3187, 2002.
- [20] F. Borrelli, M. Baotic, A. Bemporad, and M. Morari. Dynamic programming for constrained optimal control of discrete-time linear hybrid systems. *Automatica*, 41:1709–1721, October 2005.

- [21] B. Lincoln and A. Rantzer. Relaxing dynamic programming. *IEEE Transactions on Automatic Control*, 51(8):1249–1260, August 2006.
- [22] A. Rantzer. Relaxed dynamic programming in switching systems. *IEE Proceedings - Control Theory & Applications*, 153(5):567 – 574, September 2006.
- [23] M. Rinehart, M. Dahleh, and I. Kolmanovsky. Optimal control of switched homogeneous systems. In *Proceedings of the American Control Conference*, pages 1377–1382, New York City, NY, Jul 2007.
- [24] C. Seatzu, D. Corona, A. Giua, and A. Bemporad. Optimal control of continuous-time switched affine systems. *IEEE Transactions on Automatic Control*, 51:726– 741, May 2006.
- [25] D. P. Bertsekas and J. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, Sep. 1996.
- [26] D.P. Bertsekas. *Dynamic Programming and Optimal Control*, volume 2. Athena Scientific, 2 edition, 2001.
- [27] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.
- [28] P. E. Caines and D. Q. Mayne. On the discrete time matrix Riccati equation of optimal control. *Internation Journal of Control*, 12(5):785–794, 1970.
- [29] Siew Chan, Goodwin, and G. Kwai Sin. Convergence properties of the Riccati difference equation in optimal filtering of nonstabilizable systems. *IEEE Transactions on Automatic Control*, 29(2):110–118, 1984.
- [30] Sven Hedlund and Anders Rantzer. Convex dynamic programming for hybrid systems. *IEEE Transactions on Automatic Control*, 47(9):1536–1540, sep 2002.
- [31] S. Hedlund and A. Rantzer. Optimal control of hybrid system. In *Proceedings of the IEEE Conference on Decision and Control*, volume 4, pages 3972–3977, Phoenix, AZ, December 1999.
- [32] R. Horn and C. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.
- [33] Wei Zhang and Jianghai Hu. On Optimal Quadratic Regulation for Discrete-Time Switched Linear Systems. In *International Workshop on Hybrid Systems: Computation and Control*, pages 584–597, St Louis,MO,USA, 2008.