2015 American Control Conference
Palmer House Hilton
July 1-3, 2015. Chicago, IL, USA

Computational Methods for MIMO Flat Linear Systems: Flat Output Characterization, Test and Tracking Control

Sze Zheng Yong [a] Brian Paden [a] Emilio Frazzoli [a]

***Abstract*— This paper is concerned with the study of flat outputs for multiple-input-multiple-output (MIMO) controllable linear time-invariant discrete- and continuous-time systems in state-space representation. Leveraging the equivalence of flatness to an estimation-theoretic system property known as strong observability, a quick computational test is developed for ascertaining if an output candidate is flat and the subspace of flat outputs can be constructively characterized. Moreover, we propose a computational method to find differentially (continuous-time), as well as non-causal and causal difference (discrete-time) flat outputs via a system transformation into a special control canonical form. Finally, design principles for flatness-based trajectory planning and tracking control for discrete-time systems are presented, which to our best knowledge, have yet to be successfully demonstrated.**

I. Introduction

A system property known as flatness first introduced in [1] is widely used for analyzing and synthesizing controllers for nonlinear dynamical systems. Flatness-based techniques have been developed and successfully applied to many applications, and is employed even for linear time-invariant systems as a versatile technique for solving trajectory planning, feedforward and set point control problems.

Literature review. Fliess et al. [1], [2] have shown that linear systems are flat if and only if the system is controllable and that flat outputs are not unique. Moreover, they established that all flat outputs are in the linear form without outlining a test for checking if any of these candidates are flat outputs. Using polynomial matrices, Lévine and Nguyen [3] provided an indirect test by providing necessary conditions for flat outputs in multiple-input-multiple-output (MIMO) linear time-invariant continuous-time systems and proposed a trajectory planning approach based on these matrices. In contrast, Sira-Ramírez and Agrawal [4], and Ben Abdallah et al. [5] provided a means of constructing one flat output for MIMO linear time-invariant continuous-time systems and demonstrated their use for flatness-based trajectory planning and tracking control.

On the other hand, less attention has been given to linear time-invariant discrete-time systems. Sira-Ramírez and Agrawal [4] has discussed this class of systems as a counterpart to continuous-time systems, thus enabling the construction of one specific MIMO flat output. However, the equivalent notion of differential flatness for discrete-time systems is a non-causal version of difference flatness, which would render any such flatness based feedback control unimplementable. This problem was identified in [4] and a solution for only single-input-single-output (SISO) systems was presented. Hence, the characterization of MIMO causal difference flat outputs and its application to trajectory planning and tracking control remain an open problem.

Contributions. First, the equivalence of flatness to an estimation-theoretic system property known as strong observability (e.g., [6], [7]) is demonstrated. Consequently, previously established computational techniques can be made available for checking the flatness of an output candidate and for constructively characterizing the subspace of flat outputs, which may facilitate a 'wise' choice of flat outputs for certain applications. In addition, we provide a computational approach to find MIMO differentially (continuous-time) as well as non-causal and causal difference (discrete-time) flat outputs. Finally, we demonstrate that causal difference flatness can be capitalized on for trajectory planning and tracking control of linear time-invariant discrete-time systems using an example of an autonomous helicopter landing on a slope.

II. Problem Formulation

We consider controllable (hence flat [1], [2]) linear time-invariant (LTI) systems in the state-space representation:

Continuous-time:

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{1}$$

Discrete-time:

$$x[k+1] = Ax[k] + Bu[k], \tag{2}$$

where $x(t), x[k] \in \mathbb{R}^n$ and $u(t), u[k] \in \mathbb{R}^m$ are the state and input vectors at time t and k, respectively. The matrices A and B are known, and without loss of generality, we assume $\mathrm{rank}(B) = m$. (Otherwise, we can retain the linearly independent columns and the "remaining" inputs still affect the system in the same way).

The main objective of this paper is to characterize the subspace of flat outputs (as will be formally defined in Definitions 1, 2 and 3) for systems (1) and (2). As is shown in [1, p.1334], all flat outputs[1], $y(t) \in \mathbb{R}^p$, for LTI continuous-time systems are of the following linear form:

$$y(t) = Cx(t) + \textstyle\sum_{i=0}^{r} D_i \frac{d^i}{dt^i}(u(t)), \tag{3}$$

where r is finite. Analogously, for discrete-time systems, we also consider linear flat outputs, $y[k] \in \mathbb{R}^p$:

$$y[k] = Cx[k] + \textstyle\sum_{i=0}^{r} D_i u[k+i], \tag{4}$$

$$y[k] = Cx[k] + \textstyle\sum_{i=0}^{r} D_i u[k-i], \tag{5}$$

This research was conducted jointly with Aurora Flight Sciences and supported by the Office of Naval Research under ONR STTR Contract #N00014-11-C-0458, Program manager Dr. Marc Steinberg.

[a] The authors are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA, USA (email: {szyong, bapaden, frazzoli}@mit.edu).

[1] Note that the dimension of the flat output vector is shown to be equal to the dimension of the input vector, i.e., $p = m$ (cf. [1], [2]). This matches our observation in Theorem 2.

where the former flat outputs are non-causal and the latter are causal. Note that causality of the flat outputs are not necessary for many applications such as trajectory generation but would be necessary when used as feedback in controller design problems. In our search for causal flat outputs, we additionally assume that the discrete-time system is causal/reversible, with the implication that A is invertible [8]. Thus, the problem of characterizing flat outputs of (1) and (2) is equivalent to finding the matrices C and $\{D^i\}_{i=0}^r$ such that the outputs given by (3), (4) and (5) are flat.

The objective of this paper is to develop computational tools for MIMO linear time-invariant flat systems (both with continuous- and discrete-time dynamics). Specifically, we outline procedures to test if an output is flat, to automatically find flat outputs and to enable flatness based trajectory planning and tracking control.

III. Preliminary Material

Flatness has been primarily defined for continuous-time systems [1] in which the states and inputs can be represented as functions of some variables known as flat outputs and a finite number of their derivatives. On the other hand, flatness for discrete-time systems can be defined with the existence of functions involving a finite number of forward or backward values. Thus, there are three possible notions of flatness, which we define next.

Definition 1 (Differential Flatness (Continuous-Time)). *A linear time-invariant continuous-time system* (1) *is differentially flat if there exists an output $y(t)$ (of the linear form in* (3)*) such that the states $x(t)$ and inputs $u(t)$ are representable as functions of the outputs y and a finite number of its derivatives with respect to time, $y^{(i)}(t), i = 1, \ldots, q$.*

Definition 2 (Forward Difference Flatness (Non-Causal, Discrete-Time)). *A linear time-invariant discrete-time system* (2) *is forward difference flat if there exists an output $y[k]$ (of the linear non-causal form in* (4)*) such that the states $x[k]$ and inputs $u[k]$ are representable as functions of the outputs $y[k]$ and a finite number of its forward values, $y[k+i], i = 1, \ldots, q$.*

Definition 3 (Backward Difference Flatness (Causal, Discrete-Time)). *A linear time-invariant discrete-time system* (2) *is backward difference flat if there exists an output $y[k]$ (of the linear causal form in* (5)*) such that the states $x[k]$ and inputs $u[k]$ are representable as functions of the outputs $y[k]$ and a finite number of its backward values, $y[k-i], i = 1, \ldots, q$.*

A further definition that is used throughout the paper is the multiple-input-multiple-output (MIMO) invariant zeros. To this end, using bilateral Laplace and Z-transforms of the system dynamics (1), (2) and the output equations (3), (4) and (5), we first obtain the system matrix $S(\lambda)$, upon which the definition of invariant zeros is based, as

$$S(\lambda) := \begin{bmatrix} A - \overline{\lambda} I & B \\ C & \sum_{i=0}^{r} \lambda^i D_i \end{bmatrix}, \tag{6}$$

where λ represents the differential, the forward difference and the backward difference operators, respectively, whereas we have $\overline{\lambda} = \lambda$ for differential and forward difference flatness and $\overline{\lambda} = 1/\lambda$ for backward difference flatness.

Definition 4 (Invariant Zeros). *The invariant zeros λ of the system matrix $S(\lambda)$ in* (6) *are defined as the finite values of λ for which the matrix $S(\lambda)$ drops rank, i.e.,*

$$\mathrm{rk}(S(\lambda)) < \mathrm{nrank}(S),$$

where $\mathrm{nrank}(S)$ denotes the normal rank (maximum rank over $\lambda \in \mathbb{C}$) of $S(\lambda)$.

Moreover, we define a system property known as *strong observability* that is used in the estimation theory literature (e.g., [6], [7]), which will prove useful in determining system flatness later in this paper.

Definition 5 (Strong observability). *A linear system is* strongly observable*, if $y[k] = 0\,(k \in \mathbb{N}), y(t) = 0\,(t > 0)$ implies $x[k] = 0\,(k \in \mathbb{N}), x(t) = 0\,(t > 0)$ irrespective of the input and the initial state.*

A helpful result concerning strong observability is given in the next theorem (see, e.g., [6], [7] for details):

Theorem 1 (Strong Observability). *A linear system is* strongly observable *if and only if the system has no invariant zeros and has full normal rank, i.e., $\mathrm{rk}(S(\lambda)) = \mathrm{nrank}(S) = n + m$ for all $\lambda \in \mathbb{C}$.*

IV. Flatness and Strong Observability

By Definitions 1 and 2, it is clear that differential and difference flatness requires that the inputs $u(t), u[k] \in \mathbb{R}^m$ and states $x(t), x[k] \in \mathbb{R}^n$ can be recovered as functions of the flat outputs $y(t), y[k] \in \mathbb{R}^p$ and their derivatives and forward/backward values. Thus, this problem is very much like the estimation problem of states and unknown inputs based on observations/outputs (see, e.g., [7]); hence techniques from the latter field are also applicable for this problem. Specifically, the following necessary and sufficient condition for flatness is equivalent to the system property known as strong observability as defined in Definition 5.

Theorem 2 (Necessary and Sufficient Condition). *The outputs $y(t) \in \mathbb{R}^p$ and $y[k] \in \mathbb{R}^p$ in* (3), (4) *and* (5) *are flat outputs of* (1) *and* (2) *if and only if the overall system $(A, B, C, \{D_i\}_{i=0}^r)$ is strongly observable, or equivalently, has no invariant zeros and has full normal rank, i.e.,*

$$\mathrm{rk}(S(\lambda)) = n + m, \quad \forall \lambda \in \mathbb{C}, \tag{7}$$

where the system matrix $S(\lambda)$ is given in (6)*. Furthermore, assuming no redundancy, the number of flat outputs is equal to the number of inputs (i.e., $p = m$), which agrees with the fact that the input and flat output dimensions match [1], [2].*

Proof. It is straightforward to observe that the systems (1) and (2), along with the corresponding flat output candidates (3), (4) and (5) can be written as

$$S(\lambda) \begin{bmatrix} X(\lambda) \\ U(\lambda) \end{bmatrix} = \begin{bmatrix} 0 \\ Y(\lambda) \end{bmatrix}, \tag{8}$$

using bilateral Laplace and Z-transforms, respectively, with λ representing the differential, the forward difference and the backward difference operators, respectively. Thus, $X(\lambda)$ and $U(\lambda)$ can be uniquely determined if and only if $\text{rk}(S(\lambda)) = n + m$ for all $\lambda \in \mathbb{C}$, i.e., the overall system $(A, B, C, \{D_i\}_{i=0}^r)$ has no invariant zeros.

Moreover, for the rank condition to hold, we must have $p \geq m$ and if we assume the absence of redundancy, i.e., $S(\lambda)$ is square, then we have $p = m$. ■

The following corollary can be directly deduced from the rank condition for $S(\lambda)$:

Corollary 1. *The overall system* $(A, B, C, \{D_i\}_i^r)$ *must be both controllable and observable. Hence, the system* (1) *and* (2) *must be controllable.*

Proof. For differential and forward flatness ($\overline{\lambda} = \lambda$), note that the PBH test [9, pp. 135-137] for controllability and observability are embedded in the rank condition in (7). For causal/reversible discrete-time systems (i.e., A is invertible), it can be observed that controllability and observability of the time-reversed system are necessary for the rank condition in (7) to hold for backward (causal) difference flat outputs, and that controllability and observability are equivalent to reachability and constructibility [8]. ■

The necessary and sufficient condition in Theorem 2 provides a means to test if an output is flat (cf. Section V-A) and to characterize the subspace of flat outputs by requiring that the system matrix $S(\lambda)$ has no invariant zeros, i.e., by enforcing the constraints that the numerators are constants with all coefficients of multiples of λ being set to zero (cf. Section VI-A). The characterization of the flat output subspace also provides a possibility of selecting special flat outputs such that other tasks such as collision checking in the trajectory planning application becomes easier or is not required. This will be discussed in the context of an application example in Section VI-A.

Next, we construct the expressions for $u(t), u[k] \in \mathbb{R}^m$ and $x(t), x[k] \in \mathbb{R}^n$ as functions of the flat outputs and their derivatives and forward/backward values, respectively (cf. Definitions 1, 2 and 3), assuming that the necessary and sufficient condition in Theorem 2 is satisfied.

Theorem 3. *Suppose Theorem 2 holds. Then, the transfer function matrices* $\overline{\lambda}I - A$ *and* $F(\lambda) := C(\overline{\lambda}I - A)^{-1}B + \sum_{i=0}^r \lambda^i D_i$ *have no zeros for all* $\lambda \in \mathbb{C}$*; and the states* $X(\lambda)$ *and inputs* $U(\lambda)$ *can be obtained as functions of* $Y(\lambda)$*:*

$$U(\lambda) = (F(\lambda))^{-1} Y(\lambda), \tag{9}$$

$$\begin{aligned} X(\lambda) &= (\overline{\lambda}I - A)^{-1} B U(\lambda) \\ &= (\overline{\lambda}I - A)^{-1} B (F(\lambda))^{-1} Y(\lambda); \end{aligned} \tag{10}$$

thus, $u(t), u[k] \in \mathbb{R}^m$ *and* $x(t), x[k] \in \mathbb{R}^n$ *can be found via the inverse of the differential and forward/backward difference operators (i.e., via inverse Laplace and Z-tranforms) as functions of only the flat outputs and their derivatives and forward/backward values, respectively.*

Proof. The absence of zeros for the matrices $(\overline{\lambda}I - A)$ and $F(\lambda)$ follows from the following rank equality:

$$\begin{aligned} &\text{rk}\begin{bmatrix} A - \overline{\lambda}I & B \\ C & \sum_{i=0}^r \lambda^i D_i \end{bmatrix} \\ &= \text{rk}\begin{bmatrix} I & 0 \\ C(\overline{\lambda}I - A)^{-1} & I \end{bmatrix} \begin{bmatrix} A - \overline{\lambda}I & B \\ C & \sum_{i=0}^r \lambda^i D_i \end{bmatrix} \\ &= \text{rk}\begin{bmatrix} A - \overline{\lambda}I & B \\ 0 & C(\overline{\lambda}I - A)^{-1}B + \sum_{i=0}^r \lambda^i D_i \end{bmatrix} \\ &= \text{rk}(A - \overline{\lambda}I) + \text{rk}(C(\overline{\lambda}I - A)^{-1}B + \textstyle\sum_{i=0}^r \lambda^i D_i), \end{aligned}$$

since from the above and Theorem 2, $\text{rk}(A - \overline{\lambda}I) = n$ and $\text{rk}(F(\lambda)) = \text{rk}(C(\overline{\lambda}I - A)^{-1}B + \sum_{i=0}^r \lambda^i D_i) = m$, i.e., $A - \overline{\lambda}I$ and $F(\lambda)$ have full rank for all $\lambda \in \mathbb{C}$ such that $\text{rk}(S(\lambda)) = n + m$.

Furthermore, from the premultiplication of (8) on both sides by $\begin{bmatrix} I & 0 \\ C(\overline{\lambda}I - A)^{-1} & I \end{bmatrix}$, it is straightforward to show that $U(\lambda)$ and $X(\lambda)$ can be found as in (9) and (10). Since we have shown that both $(\overline{\lambda}I - A)$ and $F(\lambda)$ have no zeros, it follows that $U(\lambda)$ and $X(\lambda)$ have no poles. Thus, using the inverse of the differential and forward/backward difference operators (i.e., inverse Laplace and Z-tranforms), $u(t), u[k] \in \mathbb{R}^m$ and $x(t), x[k] \in \mathbb{R}^n$ are functions of only the flat outputs and their derivatives and forward/backward values, as required by the definitions for differential and difference flatness (cf. Definitions 1, 2 and 3). ■

Remark 1. *The expressions for* $U(\lambda)$ *and* $X(\lambda)$ *in* (9) *and* (10) *(cf. Theorem 3) trivially satisfy the conditions for flat output characterization in [3, Theorem 1].*

Moreover, a useful lemma can be found to obtain a subspace of flat outputs from a known flat output.

Lemma 1. *Suppose the output* $y(t), y[k] \in \mathbb{R}^m$ *is flat for some matrices* C *and* $\{D_i\}_{i=0}^r$*. Then, for any invertible matrix,* $R \in \mathbb{R}^{m \times m}$*, the output* $y'(t), y'[k]$ *with matrices* $C' := RC$ *and* $\{D_i'\}_{i=0}^r := \{RD_i\}_{i=0}^r$ *is also flat.*

Proof. Since R is invertible, using the following equality:

$$\begin{aligned} \text{rk}\begin{bmatrix} A - \overline{\lambda}I & B \\ C & \sum_{i=0}^r \lambda^i D_i \end{bmatrix} &= \text{rk}\begin{bmatrix} I & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} A - \overline{\lambda}I & B \\ C & \sum_{i=0}^r \lambda^i D_i \end{bmatrix} \\ &= \text{rk}\begin{bmatrix} A - \overline{\lambda}I & B \\ RC & \sum_{i=0}^r \lambda^i R D_i \end{bmatrix}, \end{aligned}$$

the corollary follows directly since the transformed outputs satisfy Theorem 2. ■

The above lemma has the immediate implication that if we know one set of flat outputs, then we have obtained a whole family or subspace of flat outputs (a.k.a. endogenous transformation in [10]). Moreover, if the set of flat outputs are positions/configurations, the transformed flat outputs with any coordinate transformation are also flat outputs since rotation matrices are invertible and it is straightforward to see that translation of flat outputs preserves their flatness.

While it well known that an LTI system is flat if and only if it is controllable [1, Theorem 2], it is useful to characterize the subspace of flat outputs in the following way:

Proposition 1 (Existence). *If a linear system is controllable, then there exists a set or family of flat outputs with some matrices C and $\{D_i\}_{i=0}^r$.*

Proof. The existence of flat outputs for controllable systems can be proven by construction, which will be described in detail in Section V-B. In addition, a family of flat outputs also exists by Lemma 1. ■

V. Computational Approach for Flat Output Test and Construction

In this section, we focus on computational methods for testing if an output is flat, and for construction/finding of flat outputs for linear controllable systems. In particular, we leverage existing control system algorithms (e.g., Control System Toolbox), which are readily available, among others, in the MATLAB® commercial software package (The MathWorks Inc., Natick, MA).

A. Flat Output Test

As briefly mentioned in the previous section, the necessary and sufficient condition in Theorem 2 provides a direct test for an output candidate/guess. For example, given a linear system with $r = 1$ and given A, B, C, D_0 and D_1, the *differential and forward difference flatness test* can be carried out with MATLAB command `[z,nrank]=tzero([A-s*eye(n),B; C,D0+s*D1])` to test if the system matrix $S(\lambda)$ given in (6), with `s=tf('s')` representing λ,

(i) has no invariant zeros (i.e., `z=` $\emptyset$) and
(ii) has full normal rank (i.e., `nrank=` $n+m$).

The satisfaction of the two conditions above guarantees that the output candidate is flat. Otherwise, the output candidate is not flat. A more concrete example to illustrate the flat output test is with $A = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ and $B = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$. It can be verified that with the outputs with $C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$ $(r = \emptyset)$ or with $C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$, $D_0 = D_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$ $(r = 1)$ are flat; but the output with $C = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ $(r = \emptyset)$ is not flat because there is an invariant zero at `z=` -1.

On the other hand, to test *backward difference flatness* for a linear discrete-time system (e.g., with $r = 1$), we use the MATLAB command `[z, nrank]= tzero([A-1/s*eye(n),B;C,D0+s*D1])` to test if the above conditions (i) and (ii) are satisfied, from which the flatness of the output candidate is verified.

B. Flat Output Construction

The fundamental assumption imposed in this section is that the linear system given by (1) or (2) is controllable (a necessary condition by Corollary 1). Thus, the system can be transformed into the control canonical form, which in general is not unique for multiple-input and multiple-output (MIMO) systems [11]. Thus, the flat outputs based on these different forms are also not unique.

A specific differentially and forward difference flat output can be found in [4, pp. 84–85,157–158], which is based on the first canonical form given in [11, Eq. (8) and (9)] but no backward difference flat output was given. In this section, we provide an alternative flat output using the 'special' canonical form in [11, Eq. (13), (14) and (16)], which transforms the system into a 'chain of integrators' and hence a convenient system representation for finding flat outputs. In fact, we will show that with this special canonical form (also considered in [5]), the differentially, the forward and backward difference flat outputs can be readily constructed.

Let T be the transformation matrix that transforms the state vector $x(t), x[k]$ into $\tilde{x}(t) = T^{-1}x(t), \tilde{x}[k] = T^{-1}x[k]$ (for details about the construction of the transformation matrix, the reader is referred to [11]). With this transformation, the transformed state and input matrices, $\tilde{A} := TAT^{-1}$ and $\tilde{B} := TB$, have the following structure

$$\left[\, \tilde{A} \,\middle|\, \tilde{B} \,\right] = \left[\begin{array}{cccccccccc|ccc} 0 & 1 & \dots & 0 & & & & & & & 0 & \dots & 0 \\ \vdots & & \ddots & \vdots & & & & & & & \vdots & & \vdots \\ 0 & 0 & \dots & 1 & & & & & & & 0 & \dots & 0 \\ * & * & \dots & * & \dots & * & * & \dots & * & & 1 & \dots & * \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & & \vdots & \ddots & \vdots \\ & & & & & 0 & 1 & \dots & 0 & & & & 0 \\ & & & & & \vdots & & \ddots & \vdots & & & & \vdots \\ & & & & & 0 & 0 & \dots & 1 & & & & 0 \\ * & * & \dots & * & \dots & * & * & \dots & * & & 0 & \dots & 1 \end{array}\right], \quad (11)$$

where the $*$'s in the matrix represent possible nonzero elements and each nonzero row of $\tilde{B}$ corresponds to the 'controllability' index of each chain of integrators which we denote as $\gamma_i, i = 1, \dots, m$. An algorithm for the system transformation into control canonical form described above is readily available on MATLAB central [12].

The key observation is that the above canonical form readily allows for finding matrices $\tilde{C}$ and $\{\tilde{D}_i\}_{i=0}^r$. Rather than by (tedious) algebraic manipulation, it can be easily verified by inspection (similar to the single-input and single-output (SISO) case, see, e.g., [1]) that one specific differentially and forward difference flat output can be found with

$$\tilde{C} := CT^{-1} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & & & & & & & & & \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & & \vdots & & \vdots \\ & & & & & & & 1 & 0 & 0 & \dots & 0 \end{bmatrix}, \quad (12)$$

i.e., each row of $\tilde{C}$ is of the form $\begin{bmatrix} 0, \dots, 0, 1, 0, \dots, 0 \end{bmatrix}$ with the '1' in the $(1 + \sum_{i=1}^{j-1} \gamma_i)$-th position for the j-th row ($j = 1, \dots, m$); and $\{\tilde{D}_i\}_{i=0}^r = \{D_i\}_{i=0}^r = 0$ for any finite r. Then, since strong observability is preserved under similarity transformation T, which can be observed from the following:

$$\begin{aligned} &\text{rk} \begin{bmatrix} \tilde{A} - \overline{\lambda} I & \tilde{B} \\ \tilde{C} & \sum_{i=0}^r \lambda^i \tilde{D}_i \end{bmatrix} \\ &= \text{rk} \begin{bmatrix} T & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} A - \overline{\lambda} I & B \\ C & \sum_{i=0}^r \lambda^i D_i \end{bmatrix} \begin{bmatrix} T^{-1} & 0 \\ 0 & I \end{bmatrix} \\ &= \text{rk} \begin{bmatrix} A - \overline{\lambda} I & B \\ C & \sum_{i=0}^r \lambda^i D_i \end{bmatrix}, \end{aligned}$$

it follows that $C = \tilde{C}T$ and $\{D_i\}_{i=0}^{r} = 0$ provides one flat output for the original linear system (1) and (2). Thus, this construction approach proves the existence of a flat output in Proposition 1. Note, however, that in practice, it is easier to plan a trajectory using the flat outputs in the transformed coordinates (in control canonical form), as will be discussed in Section VI-B.

Similarly, it can observed by simple bookkeeping that a specific backward difference flat output can be obtained with $\{\tilde{D}_i\}_{i=1}^{r} = 0$, as well as $\tilde{C}$ and $\tilde{D}_0$ given by

$$[\,\tilde{C}\,|\,\tilde{D}_0\,] = \left[\begin{array}{cccccccccc|ccc} * & * & \dots & * & \dots & * & * & \dots & * & 1 & \dots & * \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ * & * & \dots & * & \dots & * & * & \dots & * & 0 & \dots & 1 \end{array}\right], \quad (13)$$

whose rows ($j = 1, \dots, m$) are essentially all $(1+\sum_{i=1}^{j-1}\gamma_i)$-th rows of $[\,\tilde{A}\,|\,\tilde{B}\,]$; and in the original states, the backward difference flat output can be found with $C = \tilde{C}T$ and $D_0 = \tilde{D}_0$. This systematic approach for obtaining backward difference flat outputs of controllable linear time-invariant systems is, to our best knowledge, never been considered.

VI. Application to Trajectory Planning and Tracking Control

We now show that we can leverage differential and difference flatness for trajectory planning and tracking control using an example of a linearized helicopter model given by

$$\begin{aligned} \ddot{x} &= -b_x\dot{x} - g\theta,\ \ddot{y} = -b_y\dot{y} + g\phi,\ \ddot{z} = -\mu\dot{z} + \mu w_{ref}, \\ \ddot{\theta} &= -2\zeta_\theta\omega_\theta\dot{\theta} - \omega_\theta^2\theta + \omega_\theta^2\theta_{ref}, \\ \ddot{\phi} &= -2\zeta_\phi\omega_\phi\dot{\phi} - \omega_\phi^2\phi + \omega_\phi^2\phi_{ref}, \end{aligned} \quad (14)$$

with states $\vec{x} = \begin{bmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} & \theta & \dot{\theta} & \phi & \dot{\phi} \end{bmatrix}^\top$ and inputs $\vec{u} = \begin{bmatrix} \theta_{ref} & \phi_{ref} & w_{ref} \end{bmatrix}^\top$.

First, we show that a subspace of flat outputs can be described by some equality and inequality constraints and how that can be used to select a particular flat output that may lead to 'easier' trajectory planning. Then, we present design principles for a backward difference flat output based tracking controller, using the example of a discretized version of the helicopter model in (14).

A. Subspace of Flat Outputs and Selection

Computer algebra systems such as Mathematica (Wolfram Research, Inc., Champaign, IL), MATLAB Symbolic Math Toolbox, etc. can be used to describe a subspace of flat outputs. Suppose that we seek a subspace of flat outputs with $C = \begin{bmatrix} c_{11} & 0 & c_{13} & 0 & c_{15} & 0 & c_{17} & 0 & c_{19} & 0 \\ c_{21} & 0 & c_{23} & 0 & c_{25} & 0 & c_{27} & 0 & c_{29} & 0 \\ c_{31} & 0 & c_{33} & 0 & c_{35} & 0 & c_{37} & 0 & c_{39} & 0 \end{bmatrix}$ and $r = \emptyset$. Using any symbolic math package, we can define each non-zero element of C, of the matrices A and B corresponding to the helicopter model (14) and the λ operator as symbols, and compute $X(\lambda)$ using (10). By Theorem 3, $X(\lambda)$ must not contain poles (such that $x(t)$ can be expressed as a function of only the flat outputs and their derivatives). Hence, the characteristic/pole polynomial in λ of $X(\lambda)$ must be a non-zero constant, which implies that all coefficients of the polynomial must be equal to zero. From this observation, a large family/subspace of flat outputs can be described by all C matrices as shown above that satisfy:

$$\begin{aligned} &c_{13}c_{27}c_{39} - c_{13}c_{29}c_{37} - c_{17}c_{23}c_{39} + c_{17}c_{29}c_{33} \\ &\qquad + c_{19}c_{23}c_{37} - c_{19}c_{27}c_{33} = 0, \\ &-c_{17}c_{23}c_{39} + c_{17}c_{29}c_{33} + c_{19}c_{23}c_{37} - c_{19}c_{27}c_{33}) = 0, \\ &c_{12}c_{23}c_{37} - c_{12}c_{27}c_{33} - c_{13}c_{22}c_{37} + c_{13}c_{27}c_{32} \\ &+c_{17}c_{22}c_{33} - c_{17}c_{23}c_{32} + c_{11}c_{23}c_{39} - c_{11}c_{29}c_{33} \\ &-c_{13}c_{21}c_{39} + c_{13}c_{29}c_{31} + c_{19}c_{21}c_{33} - c_{19}c_{23}c_{31} = 0, \\ &b_x(c_{12}c_{23}c_{37} - c_{12}c_{27}c_{33} - c_{13}c_{22}c_{37} + c_{13}c_{27}c_{32} \\ &+c_{17}c_{22}c_{33} - c_{17}c_{23}c_{32}) + b_y(c_{11}c_{23}c_{39} - c_{11}c_{29}c_{33} \\ &-c_{13}c_{21}c_{39} + c_{13}c_{29}c_{31} + c_{19}c_{21}c_{33} - c_{19}c_{23}c_{31}) = 0, \\ &c_{11}c_{23}c_{32} - c_{11}c_{22}c_{33} + c_{12}c_{21}c_{33} - c_{12}c_{23}c_{31} \\ &\qquad -c_{13}c_{21}c_{32} + c_{13}c_{22}c_{31} \neq 0. \end{aligned}$$

Specifically, it can be verified that $c_{11} = c_{22} = c_{33} = 1, c_{12} = c_{13} = c_{17} = c_{19} = c_{21} = c_{23} = c_{27} = c_{29} = c_{31} = c_{32} = c_{37} = c_{39} = 0$ satisfy the above constraints (and also the flat output test in Section V-A); so do $c_{11} = \cos\gamma, c_{13} = \sin\gamma, c_{22} = 1, c_{31} = -\sin\gamma, c_{33} = \cos\gamma, c_{12} = c_{17} = c_{19} = c_{21} = c_{23} = c_{27} = c_{29} = c_{32} = c_{37} = c_{39} = 0$ for any angle γ. The first set of elements for C corresponds to the flat outputs being the positions x, y and z and the second set to the positions in a rotated coordinates axes $x' = x\cos\gamma + z\sin\gamma$, $y' = y$ and $z' = -x\sin\gamma + z\cos\gamma$. This is not surprising since the rotation matrix for rotating the coordinate axes is invertible and by Lemma 1, the second set of flat outputs can be obtained from the first.

This has a good implication on trajectory planning in the presence of obstacles (in our case a sloped landing site) because we can now *select* the flat outputs x', y' and z' with γ being the slope. With this, no collision checking routine is needed for x' and y', while for z', a clever choice of basis functions such that $z'(t)$ does not exceed the final landing $z'(T)$ where T is the desired landing time can also lead to a reduced number of collision checks, resulting in a potential decrease in computational cost for trajectory planning.

B. Discrete-Time Trajectory Planning and Tracking

For linear continuous-time systems, differentially flat outputs have been demonstrated to provide a means for trajectory planning and trajectory tracking (see, e.g., [3], [4]), where for tracking higher derivatives of the flat outputs are assumed to be available/measured. On the other hand, for linear discrete-time systems, the forward difference flat outputs can only be used for trajectory planning but not trajectory tracking due to non-causality. However, as we shall demonstrate, the backward difference flat outputs can be used for both trajectory planning and tracking, with the latter only requiring that a finite number of previous/backward values of the flat outputs is stored in memory.

To the best of our knowledge, trajectory planning and flatness based tracking control has not been demonstrated for discrete-time systems. Thus, the following trajectory planning and tracking control approach using backward difference flatness is novel. Given that the approach is computational in nature, this difference flatness based trajectory planning and tracking control is best illustrated with an example, in which the principles for designing such feedback control can be easily understood and applied to other examples. For this purpose, the system we consider is an equivalent discretized version of the helicopter model in (14) assuming zero-order hold (using conversion algorithms involving matrix exponentials [13], [14]), with $b_x = 0.05$, $b_y = 0.05$, $\zeta_\theta = 0.2329$, $\zeta_\phi = 0.707$, $\omega_\theta = 0.5747$, $\omega_\phi = 0.6843$, $\mu = 0.4711$ and sample time $\Delta t = 0.1s$. To find the backward difference flat output vector, we apply the computational approach in Section V-B. The transformation matrix T can be found from the state transformation procedure in [11], [12] and using (13), $C = \tilde{C}T$ and $D_0 = \tilde{D}_0$ can be computed; hence, one set of flat outputs is given by $y[k] := \begin{bmatrix} y_1[k] & y_2[k] & y_3[k] \end{bmatrix}^\top = Cx[k] + D_0 u[k]$ where

$$\begin{aligned} y_1[k] &= -3.1366x_1[k] - 0.7788x_4[k] + 0.8912x_7[k] \\ &\quad + 0.0629x_8[k] + u_1[k], \\ y_2[k] &= 2.2905x_2[k] + 0.5674x_5[k] + 0.6473x_9[k] \\ &\quad + 0.0441x_{10}[k] + u_2[k], \\ y_3[k] &= 0.2173x_3[k] + 0.0315x_6[k] + u_3[k]. \end{aligned}$$

With these flat outputs, the flatness test in Section V-A can be used to show that Conditions (i) and (ii) are indeed satisfied.

However, we observed that trajectory planning and tracking control procedure is much simpler in the transformed states, i.e., $x_d[k] = Tx[k]$ and $y_d[k] = \tilde{C}x_d[k] + \tilde{D}_0 u[k]$, and is thus the recommended approach. This is because it can be easily found (by inspection or by computing (9) and (10) using computer algebra packages (e.g., Mathematica and MATLAB Symbolic Math Toolbox) that the transformed states are simple functions of the flat outputs:

$$\begin{aligned} &x_{d,1}[k] = y_{d,1}[k-4], \quad x_{d,5}[k] = y_{d,2}[k-4], \\ &x_{d,2}[k] = y_{d,1}[k-3], \quad x_{d,6}[k] = y_{d,2}[k-3], \\ &x_{d,3}[k] = y_{d,1}[k-2], \quad x_{d,7}[k] = y_{d,2}[k-2], \\ &x_{d,4}[k] = y_{d,1}[k-1], \quad x_{d,8}[k] = y_{d,2}[k-1], \\ &x_{d,9}[k] = y_{d,3}[k-2], \quad x_{d,10}[k] = y_{d,3}[k-1]. \end{aligned} \tag{15}$$

The inputs are also simple functions of the flat outputs since they can be obtained from $u[k] = \tilde{D}_0^{-1}(y_d[k] - \tilde{C}x_d[k])$, where $\tilde{D}_0$ is always invertible by construction for linear controllable systems. In this example, we find

$$\begin{aligned} u_1[k] &= y_{d,1}[k] - 3.9653y_{d,1}[k-1] + 5.8994y_{d,1}[k-2] \\ &\quad -3.9028y_{d,1}[k-3] + 0.9687y_{d,1}[k-4], \\ u_2[k] &= y_{d,2}[k] - 3.8983y_{d,2}[k-1] + 5.6999y_{d,2}[k-2] \\ &\quad -3.7048y_{d,2}[k-3] + 0.9032y_{d,2}[k-4], \\ u_3[k] &= y_{d,3}[k] - 1.9540y_{d,3}[k-1] + 0.9540y_{d,3}[k-2]. \end{aligned} \tag{16}$$

Next, to plan trajectory that brings the helicopter from hover to a landing state with non-zero attitude within a given time $T = N\Delta t$, i.e., from $x[0] = [-5, -8, -18.35, 0, 0, 0, 0, 0, 0, 0]^\top$ to $x[N] = [0, 0, 0, 0, 0, 0, 0, 0, -0.2618, 0]^\top$, or equivalently in transformed coordinates, $x_d[0] = 10^4 \times [1.5683,\ 1.5683,\ 1.5683,\ 1.5683,\ -1.8324,\ -1.8324,\ -1.8324, -1.8324, -0.3987, -0.3987]^\top$ and $x_d[N] = [0,\ 0,\ 0,\ 0, -54.7504, 4.6754, 5.1253, -53.1067,\ 0,\ 0]^\top$, it suffices to generate a polynomial trajectory for $y_d[k]$ with respect to the time step k subject to the initial and final state constraints. There are 8 initial and final conditions corresponding to $y_{d,1}[k]$, $y_{d,2}[k]$ and 4 constraints corresponding to $y_{d,3}[k]$ which determine the minimal degree of the interpolating polynomials, as follows:

$$y^*_{d,1}[k] = \textstyle\sum_{i=0}^{7} a_i \left(\frac{k+1}{N}\right)^i, \ y^*_{d,2}[k] = \sum_{i=0}^{7} b_i \left(\frac{k+1}{N}\right)^i,$$
$$y^*_{d,3}[k] = \textstyle\sum_{i=0}^{3} c_i \left(\frac{k+1}{N}\right)^i,$$

where the coefficients a_i, b_i and c_i are chosen such that the initial and final condition constraints on $y^*_d[k]$ ($*$ denotes the planned reference trajectory) corresponding to (15) are satisfied. To obtain the minimum landing time subject to input constraints given by $|u_1[k]| \le 0.4363\rho$, $|u_2[k]| \le 0.5236\rho$ and $|u_3[k]| \le 10.1626\rho$ for all $k \in [0, N]$, where $\rho < 1$ is a safety margin (inverse of reserve factor) such that there remains some control margin for rejecting tracking error, we perform a line search to find the smallest N such that the inputs in (16) do not violate their respective constraints. We choose $\rho = \frac{2}{3}$ and the resulting smallest N is obtained as 147, which corresponds to a landing time of $T = N\Delta t = 14.7s$.

Finally, since the control inputs formulation in (16) is causal, we can design a difference flatness based tracking controller as follows:

$$\begin{aligned} u_1[k] &= y^*_{d,1}[k] - 3.9653y_{d,1}[k-1] + 5.8994y_{d,1}[k-2] \\ &\quad -3.9028y_{d,1}[k-3] + 0.9687y_{d,1}[k-4] - \alpha_1 e_1[k-1] \\ &\quad -\alpha_2 e_1[k-2] - \alpha_3 e_1[k-3] - \alpha_4 e_1[k-4], \\ u_2[k] &= y^*_{d,2}[k] - 3.8983y_{d,2}[k-1] + 5.6999y_{d,2}[k-2] \\ &\quad -3.7048y_{d,2}[k-3] + 0.9032y_{d,2}[k-4] - \beta_1 e_2[k-1] \\ &\quad -\beta_2 e_2[k-2] - \beta_3 e_2[k-3] - \beta_4 e_2[k-4], \\ u_3[k] &= y^*_{d,3}[k] - 1.9540y_{d,3}[k-1] + 0.9540y_{d,3}[k-2] \\ &\quad -\xi_1 e_3[k-1] - \xi_2 e_3[k-2], \end{aligned} \tag{17}$$

where $e_j[k] = y_{d,j}[k] - y^*_{d,j}[k]$ for $j = 1, 2, 3$ and all k. Equating the above tracking control law with (16), we obtain $e_1[k] + \sum_{i=1}^{4} \alpha_i e_1[k-i] = 0$, $e_2[k] + \sum_{i=1}^{4} \beta_i e_2[k-i] = 0$, $e_3[k] + \sum_{i=1}^{2} \xi_i e_3[k-i] = 0$,

which are error dynamics for each flat output which can be designed to be stable via pole placement. For this example, the poles for the 4th order error systems, $e_1[k]$ and $e_2[k]$, are first designed in the Laplace-domain as two pairs of complex poles with the following damping ratios and natural frequencies: $\zeta_1 = 0.975$, $\omega_{n,1} = 0.725$ and $\zeta_2 = 0.975$, $\omega_{n,2} = 0.725$ and converted to its Z-domain counterpart using $z = e^{s\Delta t}$ [15, Section 6.2]. Similarly, for $e_3[k]$, we choose a pair of complex poles with damping ratio $\zeta = 0.9$ and natural frequency $\omega_n = 1.1$ in Laplace-domain and convert the poles to Z-domain. As a result, we obtain $\alpha_1 = \beta_1 = -3.6022$, $\alpha_2 = \beta_2 = 4.8700$, $\alpha_3 = \beta_3 = -2.9287$, $\alpha_4 = \beta_4 = 0.6610$, $\xi_1 = -1.8094$ and $\xi_2 = 0.8204$.

Figure 1 shows a comparison of the trajectories of the helicopter system with perturbed initial conditions that result from using the nominal inputs (OL), the tracking control law (CL) and the planned trajectory (Ref), generated with initial conditions assumed to be known exactly. As expected, we observe that the open-loop trajectory deviates from the planned reference trajectory whereas with the tracking control law, initial condition errors are asymptotically rejected, i.e., the planned trajectory is asymptotically tracked, as desired. We can also observe the tracking control law in action when compared to the nominal planned inputs in Figure 2.

Remark 2. *Flat outputs of a continuous-time linear system are observed to be no longer flat outputs of its discretized counterpart although both models are equivalent under the assumption of zero-order hold for the inputs. Thus, while differentially flat outputs typically have physical significance, there is in general no obvious physical interpretation of difference flat outputs of discretized systems.*

VII. Conclusion

This paper presented various computational tools for multiple-input-multiple-output flat linear time-invariant discrete- and continuous-time systems. We first showed the equivalence of flatness to the absence of invariant zeros and thus, an estimation-theoretic system property known as strong observability. Consequently, a quick test is presented for ascertaining the flatness of an output candidate and the subspace of flat outputs is characterized. In addition, the system transformation into a special control canonical form can be exploited for finding differentially, forward difference and backward difference flat outputs. Finally, we demonstrate, for the first time, the use of backward difference flatness for trajectory planning and tracking control of linear time-invariant discrete-time systems.

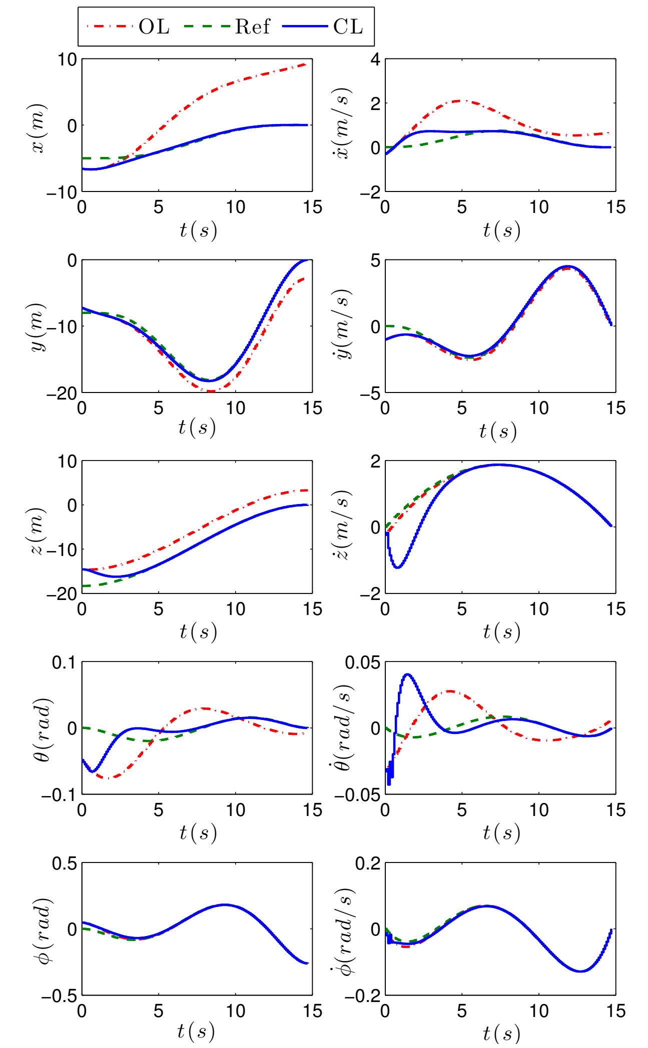

Fig. 1: A comparison of the open-loop (OL) trajectory with nominal inputs, the planned (Ref) trajectory and the closed-loop (CL) trajectory using the tracking control law in (17); initial conditions are perturbed by a random vector.

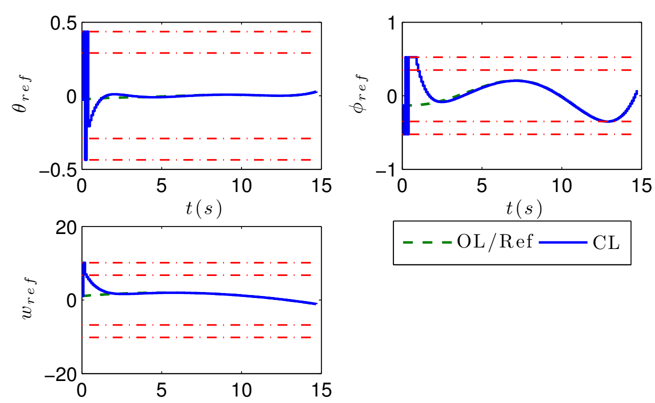

Fig. 2: A comparison of the nominal inputs (OL/Ref) with the inputs resulting from the tracking control law in (17); input constraints (with and without safety margin) are also depicted with red dash-dotted lines.

References

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

[2] M. Fliess, J. Lévine, P. Martin, and P. Rouchon. A Lie-Bäcklund approach to equivalence and flatness of nonlinear systems. *IEEE Transactions on Automatic Control*, 44(5):922–937, May 1999.

[3] J. Lévine and D.V. Nguyen. Flat output characterization for linear systems using polynomial matrices. *Systems & Control Letters*, 48(1):69–75, 2003.

[4] H. Sira-Ramírez and S.K. Agrawal. *Differentially Flat Systems*. Automation and Control Engineering. Taylor & Francis, 2004.

[5] M. Ben Abdallah, M. Ayadi, and M. Benrejeb. Flatness-based control of MIMO linear systems. In *International Multi-Conference on Systems, Signals and Devices*, pages 1–6, March 2009.

[6] M.L.J. Hautus. Strong detectability and observers. *Linear Algebra and its Applications*, 50:353–368, 1983.

[7] S.Z. Yong, M. Zhu, and E. Frazzoli. A unified framework for simultaneous input and state estimation of linear discrete-time stochastic systems. *Automatica*, 2014. Provisionally accepted. Available from: `http://arxiv.org/abs/1309.6627`.

[8] M. Fliess. Reversible linear and nonlinear discrete-time dynamics. *IEEE Transactions on Automatic Control*, 37(8):1144–1153, 1992.

[9] T. Kailath. *Linear Systems*. Prentice-Hall information and system sciences series. Prentice Hall International, 1998.

[10] P. Martin, R.M. Murray, and P. Rouchon. Flat systems. In *European Control Conference*, pages 211–264, 1997.

[11] D.G Luenberger. Canonical forms for linear multivariable systems. *IEEE Transactions on Automatic Control*, 12(3):290–293, 1967.

[12] S. Sivaramakrishnan. Multi-input controllable canonical form using `MI_ctrb`, 2008. Retrieved from `http://www.mathworks.com/matlabcentral/fileexchange/21663-multi-input-controllable-canonical-form/content/MI_ctrb.m` on July 16, 2014.

[13] R.A. DeCarlo. *Linear systems: a state variable approach with numerical implementation*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1989.

[14] C. Van Loan. Computing integrals involving the matrix exponential. *IEEE Transactions on Automatic Control*, 23(3):395–404, 1978.

[15] G.F. Franklin, J.D. Powell, and M.L. Workman. *Digital control of dynamic systems*. Addison-Wesley world student series. Addison Wesley Longman, 1998.