

# Gradient-based optimization techniques for the design of static controllers for Markov jump linear systems with unobservable modes

Alessandro N. Vargas<sup>1,\*</sup>, Daiane C. Bortolin<sup>2</sup>, Eduardo F. Costa<sup>2</sup> and João B. R. do Val<sup>3</sup>

<sup>1</sup>*Universidade Tecnológica Federal do Paraná, UTFPR,  
Av. Alberto Carazzai 1640, 86300-000 Cornélio Procópio-PR, Brazil.*

<sup>1</sup>*Basque Center for Applied Mathematics, BCAM,  
Alameda de Mazarredo 14, E-48009 Bilbao, Vizcaya, Spain*

<sup>2</sup>*Universidade de São Paulo, USP,  
Depto. Matemática Aplicada e Estatística, C.P. 668, 13560-970, São Carlos-SP, Brazil.*

<sup>3</sup>*Universidade Estadual de Campinas, UNICAMP,  
FEEC-DT, Av. Albert Einstein 400, 13083-852 Campinas-SP, Brazil.*

## SUMMARY

The paper formulates the static control problem of Markov jump linear systems, assuming that the controller does not have access to the jump variable. We derive the expression of the gradient for the cost motivated by the evaluation of ten gradient-based optimization techniques. The numerical efficiency of these techniques is verified by using the data obtained from practical experiments. The corresponding solution is used to design a scheme to control the velocity of a real-time DC Motor device subject to abrupt power failures. Copyright © 2013 John Wiley & Sons, Ltd.

Received ...

KEY WORDS: Optimal control; Stochastic systems; Optimization algorithms; DC Motor device.

## 1. INTRODUCTION

Markov Jump Linear Systems (MJLS) comprise a class of stochastic systems with a strong appeal to represent systems subject to abrupt variations in its structure. In the last two decades, it has been a subject of intensive investigation both in the theoretical and applications front, see for instance the monograph [9] and the papers [5, 6, 7, 8, 12, 13, 14, 15, 16, 24, 25, 30, 32] for an account.

In the MJLS literature, specifically in the control design, most of the results assume that the controller has complete and instantaneous access to the Markov state, but this assumption can fail in many real-time applications because the task of monitoring the Markovian mode requires a built-in sensor or a similar measurement instrument that might be expensive and difficult or even impossible to implement. In this case, a reasonable strategy is to use controllers whose implementation is irrespective of the Markov state. The design of optimal control for systems that do not have access to the Markovian mode is the central theme of this paper.

\*Correspondence to: Alessandro N. Vargas, UTFPR, Av. Alberto Carazzai 1640, 86300-000 Cornélio Procópio-PR, Brazil. E-mail: avargas@utfpr.edu.br

Contract/grant sponsor: Brazilian agencies FAPESP and CNPq; and Spanish agency Fundación Carolina; contract/grant number: FAPESP 03/06736-7, 04/06947-0; CNPq 471557/2009-9, 304856/2007-0, 304429/2007-4, 306466/2010-4; and Fundación Carolina – Programa “Movilidad de profesores e investigadores Brasil-España. C.2010”.

The control of MJLS with unobservable modes is studied in the papers [12], [13], and [25]. Our result improves upon the ones from [13] and [25] in that their solutions are suboptimal (guaranteed cost) for the  $H_2$ -norm while we seek here the optimal value. Our context is similar to the one in [12] except that the authors of [12] considered time-varying gains while we are seeking for a unique static gain. The necessary optimal condition presented here characterizes completely whether the optimization algorithm reaches a local minimum or a saddle point. Our investigation is also motivated by a practical application for a DC Motor device (Section 4).

The MJLS considered in this paper is as follows. Let  $(\Omega, \mathcal{F}, P)$  be a fixed filtered probability space, and consider the system

$$x_{k+1} = A_{\theta_k} x_k + B_{\theta_k} u_k, \quad \forall k \geq 0, x_0 \in \mathcal{R}^r, \theta_0 \sim \pi_0, \quad (1)$$

where  $x_k$  and  $u_k$ ,  $k \geq 0$  are processes taking values respectively in  $\mathcal{R}^r$  and  $\mathcal{R}^s$ . The process  $\{\theta_k\}$  represents a discrete-time homogeneous Markov chain and takes values in the finite set  $\mathcal{S} := \{1, \dots, \sigma\}$ . The state of the system is formed by the pair  $(x_k, \theta_k)$ , and  $u_k$  is the control. The matrices  $A_{\theta_k}$  and  $B_{\theta_k}$ ,  $k \geq 0$ , are given with compatible dimensions.

To measure the performance of the system (1), we consider a standard  $N$ -th horizon quadratic cost [12]

$$J_N(x_0, \pi_0) := E_{x_0, \pi_0} \left[ \sum_{k=0}^N x_k' Q_{\theta_k} x_k + u_k' R_{\theta_k} u_k \right], \quad (2)$$

where  $E_{x_0, \pi_0}[\cdot] \equiv E[\cdot | x_0, \pi_0]$  represents the expected value operator, and  $Q_{\theta_k}$  and  $R_{\theta_k}$  are positive semidefinite matrices.

Although the design of optimal controllers with no mode observation in (2) can be dealt with the theory of dynamic programming with imperfect state information, for the problem we are dealing with the dynamic programming strategy would lead to a nonlinear and high-dimensional solution for the optimization problem (see the information state in [4], [18]), which turns the solution prohibitive to be implemented in practice. Seeking for simplicity and aiming at practical control applications, we assume that the control law is in the linear static state-feedback format with no mode observation as follows.

$$u_k = Gx_k, \quad k \geq 0. \quad (3)$$

Notice that the controller  $u_k$  has complete access to the variable  $x_k$ , but it does not have any information about the value of  $\theta_k$ .

The optimization control problem we deal with is that of finding some matrix  $G$  that minimizes (2) subject to (1) and (3). Formally, if we let  $J_N(G)$  be the cost (2) for a given  $G$ , then we recast the optimization control problem as follows.

$$G^* = \arg \min_G J_N(G). \quad (4)$$

To the best of the authors' knowledge, there is no method to compute the optimal solution for the control problem in (4). A drawback for finding the optimal solution of (4) is the fact that the nonlinear functional  $J_N(G)$  may be non-convex (see Section 4). A tentative method to overcome this difficulty, aiming at the optimal solution, is to employ optimization techniques borrowed from the literature, although these techniques are able to guarantee stationary points only (i.e., local minimum or saddle points).

The main contribution of this paper is twofold. First, we derive the expression of the gradient of the optimization problem in (4). Second, we recall some optimization techniques from the literature to compare their efficiency on achieving the solution of (4) for a particular control problem. In fact, a real-time controller implements the result of (4) to control the velocity of a DC Motor device when it is subject to abrupt failures driven by a Markov chain.

The paper is organized as follows. In Section 2, we introduce the notation, problem formulation, and the main results. In Section 4, we deal with a practical application of the derived results for a DC Motor device. Finally, Section 5 presents concluding remarks.

## 2. DEFINITIONS, BASIC CONCEPTS, AND RESULTS

Let  $\mathcal{R}^r$  denote the usual  $r$ -th dimensional Euclidean space, and let  $\mathcal{M}^{r,s}$  ( $\mathcal{M}^r$ ) represent the linear space formed by all  $r \times s$  ( $r \times r$ ) real matrices. Let  $\mathcal{S}^r$  represent the normed linear subspace of  $\mathcal{M}^r$  of symmetric matrices such as  $\{U \in \mathcal{M}^r : U = U'\}$ , where  $U'$  denotes the transpose of  $U$ . Consider also  $\mathcal{S}^{r0}$  ( $\mathcal{S}^{r+}$ ) its closed (open) convex cone of positive semidefinite (definite) matrices  $\{U \in \mathcal{S}^r : U \geq 0$  ( $> 0$ ) $\}$ . Let  $\mathcal{S} := \{1, \dots, \sigma\}$  be a finite set, and let  $\mathbb{M}^{r,s}$  denote the linear space formed by a number  $\sigma$  of matrices such that  $\mathbb{M}^{r,s} = \{U = (U_1, \dots, U_\sigma) : U_i \in \mathcal{M}^{r,s}, i \in \mathcal{S}\}$ ; also  $\mathbb{M}^r \equiv \mathbb{M}^{r,r}$ . Moreover, we set  $\mathbb{S}^r = \{U = (U_1, \dots, U_\sigma) : U_i \in \mathcal{S}^r, i \in \mathcal{S}\}$ , and we write  $\mathbb{S}^{r0}$  ( $\mathbb{S}^{r+}$ ) when  $U_i \in \mathcal{S}^{r0}$  ( $\in \mathcal{S}^{r+}$ ) for all  $i \in \mathcal{S}$ .

We employ the ordering  $U > V$  ( $U \geq V$ ) for elements of  $\mathbb{S}^r$ , meaning that  $U_i - V_i$  is positive definite (semi-definite) for all  $i \in \mathcal{S}$ , and similarly for other mathematical relations. In addition, with  $U \in \mathbb{S}^r$  and  $V \in \mathcal{S}^r$ , the product  $UV$  represents the set  $(U_1V, \dots, U_\sigma V)$ . Define the inner product on the space  $\mathbb{M}^{r,s}$  as

$$\langle U, V \rangle = \sum_{i=1}^{\sigma} \text{tr}\{U'_i V_i\}, \quad \forall V, U \in \mathbb{M}^{r,s}, \quad (5)$$

and the Frobenius norm  $\|U\|_2^2 = \langle U, U \rangle$ .

If  $f : \mathcal{M}^{s,r} \mapsto \mathcal{R}$  is a differentiable function on the domain  $\mathcal{M}^{s,r}$ , we denote its partial derivative by  $\partial f(G)/\partial G$  whenever  $G \in \mathcal{M}^{s,r}$ . Let  $\text{tr}\{\cdot\}$  denote the trace operator. We now recall some derivative rules for the trace operator. Considering  $U, V, Z$ , and  $G$  as matrices with compatible dimensions, we have [23, Sec. 10.3.2]

$$\frac{\partial \text{tr}\{UGV\}}{\partial G} = U'V', \quad \frac{\partial \text{tr}\{UG'V\}}{\partial G} = VU, \quad \frac{\partial \text{tr}\{UGVG'Z\}}{\partial G} = U'Z'GV' + ZUGV. \quad (6)$$

## 2.1. Parameters and operators

Associated with the system (1)-(2), we define  $A \in \mathbb{M}^r$ ,  $B \in \mathbb{M}^{r,s}$ ,  $H \in \mathbb{M}^{r,q}$ ,  $Q \in \mathbb{S}^{r0}$ ,  $R \in \mathbb{S}^{s+}$  and  $F \in \mathbb{S}^{r0}$ . The transition probability matrix is denoted by  $\mathbb{P} = [p_{ij}]$ , for all  $i, j \in \mathcal{S}$ . The state of the Markov chain at a certain time  $k$  is determined according to an associated probability distribution  $\pi(k)$  on  $\mathcal{S}$ , namely,  $\pi_i(k) := \text{Pr}(\theta_k = i)$ . Considering the column vector  $\pi(k) = [\pi_0(k), \dots, \pi_\sigma(k)]'$ , the state distribution of the chain,  $\pi(k)$ , is defined as  $\pi(k) = (\mathbb{P}')^k \pi(0)$ . In addition, we define the operators  $\mathcal{D} = \{\mathcal{D}_i, i \in \mathcal{S}\} : \mathbb{S}^{n0} \mapsto \mathbb{S}^{n0}$  and  $\mathcal{E} = \{\mathcal{E}_i, i \in \mathcal{S}\} : \mathbb{S}^{n0} \mapsto \mathbb{S}^{n0}$ , respectively, as

$$\mathcal{D}_i(U) := \sum_{j=1}^{\sigma} p_{ji} U_j, \quad \mathcal{E}_i(U) := \sum_{j=1}^{\sigma} p_{ij} U_j, \quad \forall i \in \mathcal{S}, \forall U \in \mathbb{S}^{n0}. \quad (7)$$

Let us define the conditional second moment matrix of the system state  $x_k$ ,  $k \geq 0$ , as

$$X_i(k) = \text{E}[x_k x_k' \mathbb{1}_{\{\theta_k=i\}}], \quad \forall i \in \mathcal{S}, \forall k \geq 0, \quad (8)$$

where  $\mathbb{1}_{\{\cdot\}}$  stands for the Dirac measure. Using this definition, we can write the identity [9, p. 31]

$$\begin{aligned} & \text{E}_{x_0, \pi_0}[x_k'(Q_{\theta_k} + G'R_{\theta_k}G)x_k] \\ &= \sum_{i=1}^{\sigma} \text{tr}\{(Q_i + G'R_iG)\text{E}_{x_0, \pi_0}[x_k x_k' \mathbb{1}_{\{\theta_k=i\}}]\} = \langle Q + G'RG, X(k) \rangle, \quad \forall k \geq 0. \end{aligned}$$

Thus, the  $N$ -th horizon cost function  $J_N(x_0, \pi_0)$  as in (2) can be written equivalently as

$$J_N(G) = \sum_{k=0}^N \langle Q + G'RG, X(k) \rangle. \quad (9)$$

To evaluate precisely the cost  $J_N(G)$  as in (9), let us set  $X(k) = \{X_1(k), \dots, X_\sigma(k)\} \in \mathbb{S}^{n_0}$ ,  $k \geq 0$ , and notice that it satisfies the recurrence [9, Prop. 3.1]

$$X(k+1) = \mathcal{D}((A + BG)X(k)(A + BG)'), \quad \forall k \geq 0, \quad (10)$$

with  $X_i(0) = \pi_i(0)x_0x'_0$  for each  $i \in \mathcal{S}$ .

Finally, to complete the definition of recurrences required in the next results, let us consider the sets  $W(k) \in \mathbb{S}^{r_0}$ ,  $k = 0, \dots, N$ , generated as follows.

$$W(k+1) = (A + BG)' \mathcal{E}(W(k))(A + BG), \quad k = 0, \dots, N-1, \quad \text{and} \quad W(0) = Q + G'RG. \quad (11)$$

## 2.2. Main results

The proof of the next result is given in Appendix.

### Lemma 2.1

For each  $k = 0, \dots, N$ , there holds

$$\frac{\partial \langle Q + G'RG, X(k) \rangle}{\partial G} = 2 \left( \sum_{j=0}^{\sigma} R_j G X_j(k) + \sum_{\ell=0}^{k-1} \sum_{i=0}^{\sigma} B'_i \mathcal{E}_i(W(k-1-\ell))(A_i + B_i G) X_i(\ell) \right).$$

The next result is an immediate consequence of Lemma 2.1 and the expression for the cost in (9).

### Theorem 2.1

Let  $\varphi : \mathcal{M}^{s,r} \mapsto \mathcal{M}^{s,r}$  be the gradient of the cost  $J_N(G)$  as in (9). Then it satisfies

$$\frac{\partial J_N(G)}{\partial G} = \varphi(G), \quad (12)$$

where

$$\varphi(G) := 2 \sum_{k=0}^N \left( \sum_{j=0}^{\sigma} R_j G X_j(k) + \sum_{\ell=0}^{k-1} \sum_{i=0}^{\sigma} B'_i \mathcal{E}_i(W(k-1-\ell))(A_i + B_i G) X_i(\ell) \right), \quad (13)$$

and  $X(k) \in \mathbb{S}^{r_0}$  and  $W(k) \in \mathbb{S}^{r_0}$  satisfy (10) and (11), respectively.

The next result is immediate from Theorem 2.1 and [22, Coro. p. 185].

### Corollary 2.1 (Necessary optimal condition)

If  $\bar{G} \in \mathcal{M}^{s,r}$  is a local minimum, then  $\varphi(\bar{G}) = 0$ .

## 3. METHODOLOGY

The aim of this section is to describe the methodology we use to evaluate the necessary optimal condition of Corollary 2.1. For this purpose, let us consider the gradient of (9), evaluated at a point  $\mathbf{G}$ , as

$$\varphi(\mathbf{G}) = \left. \frac{\partial J_N(G)}{\partial G} \right|_{G=\mathbf{G}}. \quad (14)$$

We focus our study on conjugate gradient and quasi-Newton methods [2, 3, 10, 21, 22, 26, 31], and all of these algorithms are based on the following three steps.

*Step 1.* Choose  $\epsilon > 0$  and some initial point  $\mathbf{G}_0$ . Set  $k = 0$ .

*Step 2.* Find an appropriate descent direction  $\mathbf{d}_k$  and compute the scalar  $\alpha_k$  such that

$$\alpha_k := \arg \min_{\alpha > 0} J_N(\mathbf{G}_k + \alpha \mathbf{d}_k).$$

*Step 3.* Set  $\mathbf{G}_{k+1} = \mathbf{G}_k + \alpha_k \mathbf{d}_k$  and  $k = k + 1$ . Return to Step 2 if  $\|\varphi(\mathbf{G}_k)\| \geq \epsilon$ .

Notice that *Steps 1-3* produce a sequence of points  $\mathbf{G}_0, \mathbf{G}_1, \dots, \mathbf{G}_k, \dots$ , and hopefully we can choose a subsequence  $\mathbf{G}_{n_0}, \mathbf{G}_{n_1}, \dots, \mathbf{G}_{n_k}, \dots$  from it such that

$$\varphi(\mathbf{G}_{n_k}) \rightarrow 0 \quad \text{as } k \rightarrow \infty. \quad (15)$$

An accumulation point  $\mathbf{G}_\infty := \lim_{k \rightarrow \infty} \mathbf{G}_{n_k}$  satisfies the necessary optimal condition for (9) (Corollary 2.1), i.e.,

$$\varphi(\mathbf{G}_\infty) = 0.$$

As a consequence,  $\mathbf{G}_\infty$  realizes a local minimum or a saddle point for (9). Notice that a local minimum may coincide with the global one, and in this case we have  $\mathbf{G}_\infty = \mathbf{G}^*$ .

We select in our analysis the following ten optimization algorithms due to their wide use in practice, good speed of convergence, and general acceptance in the literature:

- Steepest descent (SD), see [2, Sec. 8.5], [22, Sec. 8.6];
- Davidon-Fletcher-Powell (DFP), see [2, Sec. 8.6], [31, Sec. 5.1];
- Fletcher-Reeves (FR), see [2, Sec. 8.6], [22, p. 278];
- Zangwill (Z), see [2, Sec. 8.6];
- Broyden-Fletcher-Goldfarb-Shanno (BFGS), see [31, Sec. 5.4.1];
- Polak-Ribière (PR), see [22, p. 278], [31, Sec. 4.2.1];
- Hestenes-Stiefel (HS), see [31, Sec. 4.2.1];
- Perry (P), see [21], [26];
- Dai-Yuan (DY), see [11];
- Liu-Storey (LS), see [20].

#### *Remark 3.1*

The expression of the gradient function  $\varphi(\cdot)$  as in (13) is the key to evaluate the conjugate gradient and quasi-Newton methods (SD), (DFP), (FR), (Z), (BFGS), (PR), (HR), (P), (DY), and (LS). The sequence of descent directions  $(\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_k, \dots)$  in *Step 2* requires the computation of the gradient  $\varphi(\mathbf{G}_k)$  for every point  $\mathbf{G}_k \in \mathcal{M}^{s,r}$ ,  $k \geq 0$ , c.f. [2, 3, 22, 31].

## 4. OPTIMIZATION METHODS TO CONTROL A DC MOTOR DEVICE

The main goal of this section is to illustrate the efficiency of the ten selected optimization algorithms (SD), (DFP), (FR), (Z), (BFGS), (PR), (HR), (P), (DY), and (LS), and to apply them in a scheme that controls the velocity of a real-time DC Motor device subject to abrupt failures. We evaluate the *Steps 1-3* for each algorithm, seeking for the optimal solution of the underlying Markovian controller to implement in the DC Motor device. As a matter of fact, all of the algorithms converge to the same point  $\mathbf{G}_\infty$ , and this point is used in practice to control the speed of the DC Motor device. The next section presents the equipments used in the experiments.

### 4.1. DC Motor device

The practical experiments are performed in the DC Motor Module 2208, made up by Datapool Eletronica Ltda, Brazil, using a National Instruments USB-6008 data acquisition card to set a transmission link with the computer (Fig. 1(a) and Fig. 1(b)). The control strategy is implemented physically in a computer, and the Matlab software is responsible to read the data from the respective acquisition card, process it, and return to the card an output signal. The experiments are conducted with a sampling period of 15.93 milliseconds approximately, and slight variations from this value may occur along the time stages. This laboratory testbed was used previously by some of the authors in a time-varying feedback experiment, see [33].

The main idea of this project is to design three scalar values  $g_1$ ,  $g_2$ , and  $g_3$ , to implement the state-feedback strategy into the computer (Fig. 1(c)).

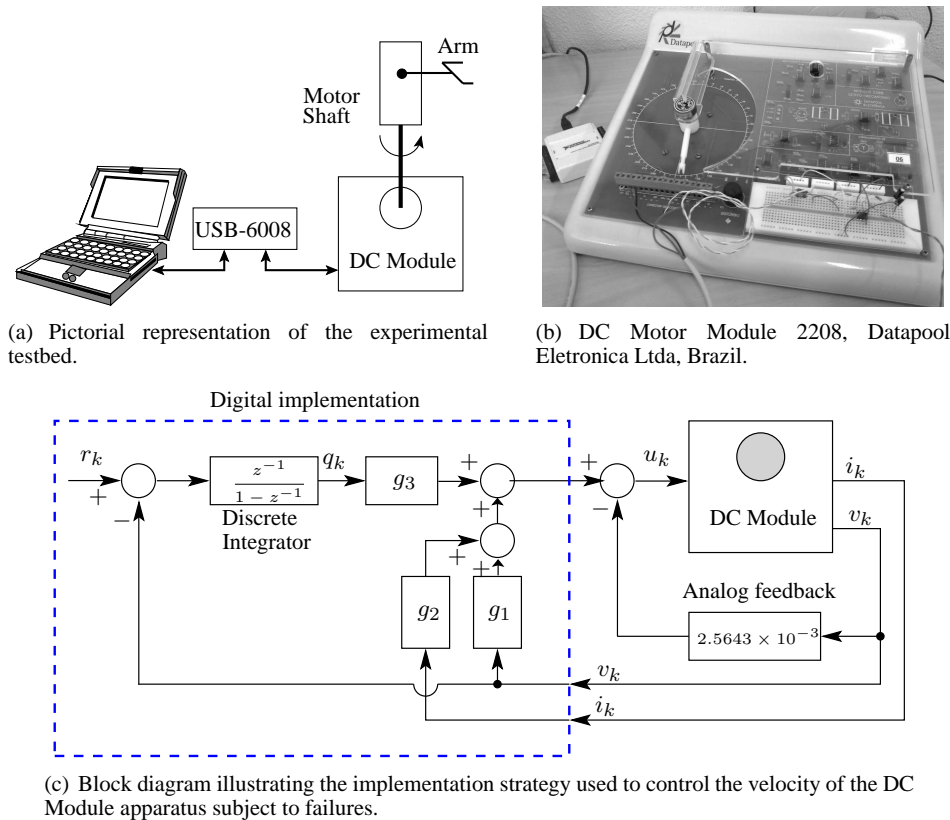


Figure 1. Laboratory DC Motor testbed used to perform the experiments of Section 4.

The angular velocity of the DC Motor and the electrical current consumed by it are represented here by  $v_k$  and  $i_k$ ,  $k \geq 0$ , respectively. Recall that DC Motors can be completely characterized by both variables  $v_k$  and  $i_k$  [19], [28], [29]. To measure  $v_k$  physically, we use the voltage range of  $0 \sim 5V$  via the manufacturer-provided tachogenerator; and to measure  $i_k$ , we connect a shunt resistor in series with the motor associated with a pre-amplifier signal stage to convert the corresponding current to voltage. To reduce the noise produced in the pre-amplifier stage, we implement a first-order analog filter. Notice that a discrete integrator is used to minimize the error between the reference signal  $r_k$  and the velocity signal  $v_k$ , as suggested in [27, Sec. 10.7.3]. A built-in analog inner loop is used to improve stability.

Abrupt failures on the power transmitted to the shaft play an important role in the speed of motors, and this fact motivates us to adjust the apparatus in order to impose power failures therein. Namely, we force the DC motor device to run under three distinct operation modes, i.e., the normal, low, and medium power modes, and these switching modes are programmed to occur according to a homogeneous Markov chain.

Abrupt failures on the power transmitted to the shaft play an important role in the speed of motors, motivating us to adjust the apparatus in order to impose power failures therein, as follows. We changed the functionality of a manufacturer-based potentiometer of the module that regulates the power level driven by the device, in such a manner that now it can be adjusted by the computer. In our experiment, the voltage signal of the potentiometer was chosen between three distinct voltage levels, producing different power modes for the DC motor device, namely the normal (full), low, and medium power modes. The switching between the voltage signal (therefore the modes) is performed by the computer, which simulates a homogeneous Markov chain by generating at each time instant

a random variable  $\lambda_k$  with uniform distribution in the interval  $[0, 1]$ , and recursively calculates

$$\theta_{k+1} = 1 + \mathbb{1}_{\lambda_k \geq p_{\theta_k,1}} + \mathbb{1}_{\lambda_k \geq p_{\theta_k,1} + p_{\theta_k,2}},$$

with initial condition  $\theta_0 = 1$ .

By setting the system state as  $x_k \equiv [v_k \ i_k \ q_k]'$  (where  $q_k$  represents the integrative term written as a discrete sum), we are able to model the DC Motor device subject to failures as the following discrete-time Markov jump linear system:

$$x_{k+1} = A_{\theta_k} x_k + B_{\theta_k} u_k + \Gamma_{\theta_k} r_k, \quad k \geq 0, \tag{16}$$

where the parameters

$$A_i = \begin{bmatrix} a_{11}^{(i)} & a_{12}^{(i)} & 0 \\ a_{21}^{(i)} & a_{22}^{(i)} & 0 \\ a_{31}^{(i)} & 0 & a_{33}^{(i)} \end{bmatrix}, \quad B_i = \begin{bmatrix} b_1^{(i)} \\ b_2^{(i)} \\ 0 \end{bmatrix}, \quad \Gamma_i = \begin{bmatrix} 0 \\ 0 \\ \gamma^{(i)} \end{bmatrix}, \quad i = 1, 2, 3.$$

are given in Table I. At the  $k$ -th stage, the system operates in normal (full) power mode when  $\theta_k = 1$ , in low power mode when  $\theta_k = 2$  or in medium power mode when  $\theta_k = 3$ .

The sequence  $\{r_k\}$  on  $\mathcal{R}$  denotes the tracking reference signal, and  $\{u_k\}$  on  $\mathcal{R}$  stands for the controller, which is defined in the linear state-feedback format, with  $G := [g_1 \ g_2 \ g_3]$ , as

$$u_k = Gx_k, \quad \forall k \geq 0. \tag{17}$$

#### 4.2. Numerical evaluations

In the numerical evaluations, we set  $N = 13$ ,  $\pi_0 = [1 \ 0 \ 0]'$ ,  $x_0 = [0.8 \ 0.22 \ 0.012]'$ ,

$$\mathbb{P} = \begin{bmatrix} 0.89999 & 0.1 & 0.00001 \\ 0.05 & 0.85 & 0.1 \\ 0.08 & 0.22 & 0.7 \end{bmatrix}, \quad Q_i = 10^{-3} \times \begin{bmatrix} 7.169327 & 0 & 0 \\ 0 & 26.971929 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

and  $R_i = 0.3025$ ,  $i = 1, 2, 3$ . The matrix  $\mathbb{P}$  was chosen in such a manner that, from normal mode, immediate visit of low power mode is much less probable than the medium one, and that retrieving from low power mode has high probability to involve the medium power mode. Note that the probabilities of changing between modes are relatively high (in view of the sampling time of about 15 milliseconds) to simulate a non-reliable device.

The design objective of this project is twofold. First, we design a controller, seeking an optimal solution for the regulator control problem stated in (4). For this purpose, we set  $r_k \equiv 0$  in (16), substitute (17) into (16), and use the expressions in (9) and (10) to evaluate the optimization

Table I. Parameters of the discrete-time MJLS representing a real DC motor device as in Section 4.

Parameters	$i = 1$	$i = 2$	$i = 3$
$a_{11}^{(i)}$	-0.479908	-1.60261	0.634617
$a_{12}^{(i)}$	5.1546	9.1632	0.917836
$a_{21}^{(i)}$	-3.81625	-0.5918697	-0.50569
$a_{22}^{(i)}$	14.4723	3.0317	2.48116
$a_{31}^{(i)}$	0.139933	0.0740594	0.386579
$a_{33}^{(i)}$	-0.925565	-0.43383	0.0982194
$b_1^{(i)}$	5.87058212	10.285129	0.7874647
$b_2^{(i)}$	15.50107	2.2282663	1.5302844
$\gamma^{(i)}$	0.11762727	-0.1328741	0.1632125

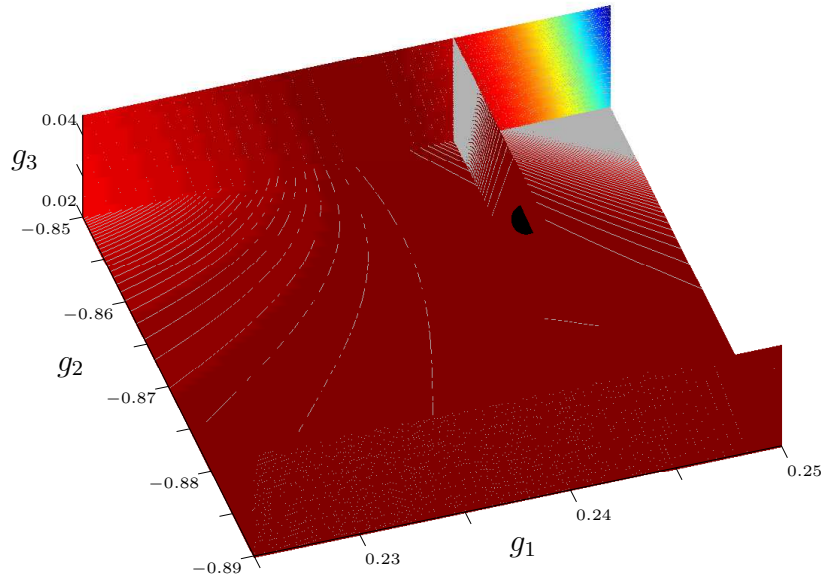


Figure 2. Three dimensional contour plot of the cost  $J_N(G)$  with  $G = [g_1 \ g_2 \ g_3]$ . The contour levels of the cost  $J_N(G)$ , represented by white lines, show that the cost  $J_N(G)$  is non-convex, and the black sphere in the center of the figure represents a point for a local minimum of  $J_N(G)$ .

algorithms (SD), (DFP), (FR), (Z), (BFGS), (PR), (HS), and (P) according to the *Steps 1-3* with initial point  $\mathbf{G}_0 = [0.24 \ -0.9 \ -0.018]$ . All of these algorithms converge successfully to the same point  $\mathbf{G}_\infty$  given by

$$\mathbf{G}_\infty = [0.242565 \ -0.866996 \ 0.035999]. \quad (18)$$

One can check that  $\varphi(\mathbf{G}_\infty) \simeq 0$ , so that  $\mathbf{G}_\infty$  is a candidate for a local minimum according to Corollary 2.1.

To evaluate the efficiency of the optimization algorithms, we check the number of iterations required by each of them to converge to the stationary point  $\mathbf{G}_\infty$  within a tolerance of  $\epsilon = 10^{-5}$  (i.e.,  $\|\varphi(\mathbf{G}_\infty)\| < \epsilon$ ). Despite the fact that the number of iterations required for the convergence vary drastically from one method to another, a relevant conclusion we can take is that all of the algorithms converges successfully to the same point of minimum (Table II). In addition, the (BFGS) algorithm is the quickest one to reach a local minimum point, while (SD) and (PR) are the slowest ones. The efficiency of the (BFGS) method is confirmed by the literature [10].

Another important conclusion we draw from the numerical evaluations is that the cost  $J_N(G)$ , for this example, is non-convex (Fig. 2). This prevents us to conclude that a stationary point obtained from the optimization algorithms is a global minimum one. Thus, for the static control problem of MJLS with unobservable modes as in (4), we are limited to assure local minimum or saddle points only.

In this project in particular, the numerical evaluations we perform indicate that  $\mathbf{G}_\infty$  represents a local minimum for the cost  $J_N(G)$  (the point  $\mathbf{G}_\infty$  is represented in Fig. 2).

In addition,  $\mathbf{G}_\infty$  is a stabilizing gain in the mean square sense [9, Th. 3.9, p. 36]. This stabilizing property is confirmed in practice by experimental data, as described in the next section.

The second main objective of this project is to implement in practice the state-feedback controller  $u_k = \mathbf{G}_\infty x_k$ , but now considering non-null values for  $r_k$ . This is illustrated in the next section.

#### 4.3. Experimental results

In this practical control project, we set the model (16) and (17) with  $G = \mathbf{G}_\infty$  and  $r(k)$  obeying a sawtooth wave signal. The idea of designing a controller for the regulation problem (i.e.,  $r(k) \equiv 0$ ), and then applying it to track a reference signal (i.e.,  $r(k) \neq 0$ ) is purposeful in practice to improve



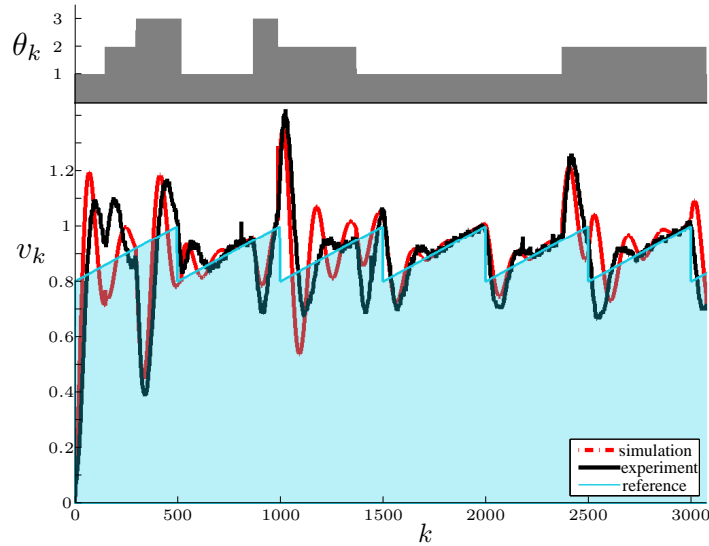


Figure 3. Trajectory of the velocity of the DC Motor device for some realization of the Markov chain  $\{\theta_k\}$ . When failures occur, the velocity of the DC Motor device suffers an impressive disturbance but the system rapidly return to a stable behavior to follow the sawtooth reference signal (colored in light blue).

attenuation of disturbances while keeping fast transient response, see [1], [17], and [29] for further details regarding deterministic systems.

It is worthy to point out that the failures in the DC Motor device impose relevant disturbances on the velocity (Fig. 3), but  $\mathbf{G}_\infty$  engenders a stable tracking behavior for the velocity and confirms the property of attenuation and fast transient response for the controller (17) with  $G = \mathbf{G}_\infty$ .

As mentioned earlier, abrupt power failures on the DC Motor device are responsible to sudden changes in the velocity of the rotor. This may expose the system to unexpected symptoms [19]. In order to verify how the DC Motor reacts to abrupt power failures, we perform eight hundreds distinct experiments with the power failures driven by the underlying Markov chain. We observe

Table II. Results obtained from an evaluation of ten selected optimization algorithms according to the control problem of a DC Motor device subject to failures, as described in Section 4. The results indicate that the BFGS algorithm is the quickest in the convergence to a local minimum.

Method	Num. Iter.	$\ \varphi(\mathbf{G}_k)\ $	$J_N(\mathbf{G}_k)$
(SD)	49192	$9.977636 \times 10^{-5}$	$7.50173125 \times 10^{-2}$
(DFP)	16	$9.598819 \times 10^{-5}$	$7.50173126 \times 10^{-2}$
(FR)	13120	$9.978498 \times 10^{-5}$	$7.50173116 \times 10^{-2}$
(Z)	26	$4.689837 \times 10^{-5}$	$7.50173114 \times 10^{-2}$
(BFGS)	9	$5.100781 \times 10^{-5}$	$7.50173110 \times 10^{-2}$
(PR)	49181	$9.982912 \times 10^{-5}$	$7.50173125 \times 10^{-2}$
(HS)	5099	$9.842895 \times 10^{-5}$	$7.50173122 \times 10^{-2}$
(P)	385	$8.265199 \times 10^{-5}$	$7.50173117 \times 10^{-2}$
(DY)	61	$6.052468 \times 10^{-5}$	$7.50173111 \times 10^{-2}$
(LS)	1892	$3.990148 \times 10^{-5}$	$7.50173111 \times 10^{-2}$

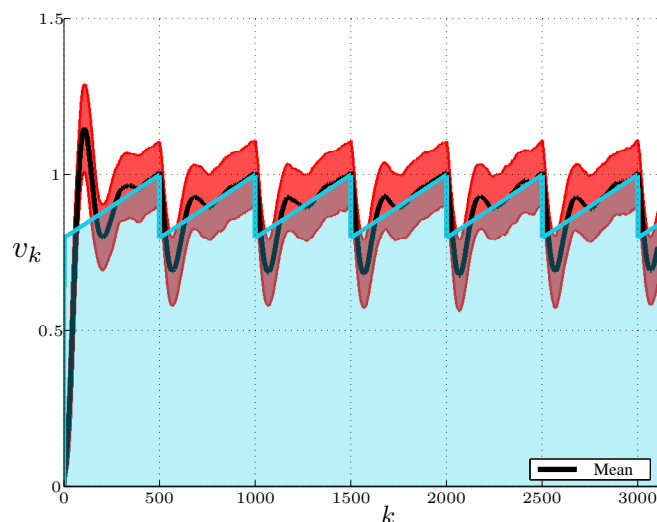


Figure 4. Representation of the velocity of the DC Motor, obtained from practical experiments, for eight hundreds distinct realizations of the underlying Markovian failure process. The mean of the velocity  $v_k$  is represented by the black straight line, bounded from above and below by its standard deviation (colored in red). The mean of the velocity  $v_k$  follows appropriately the sawtooth signal reference  $r_k$ .

that the mean value of the velocity follows the sawtooth signal reference with success even in this scenario of failures (Fig. 4). The standard deviation of the velocity from its mean is not of great amount, and indeed it is bounded, thus indicating that the stochastic system is stable, c.f. [9, Ch. 3], [34].

Our last conclusion in this project is that, even in the real scenario of failures, the designed controller with  $G_\infty$  proposed by the optimization algorithms is able to drive, with success, the mean value of the DC Motor speed to track the reference sawtooth wave signal.

## 5. CONCLUDING REMARKS

The main theoretical result in this paper is the evaluation of the gradient for the static control problem of MJLS with unobservable mode, see Theorem 2.1. To design a real-time controller for a DC Motor device, we collected the data from the experiments to generate the corresponding expression for the gradient. Using this gradient, we evaluated distinct optimization methods available in literature, to check their efficiency, in the design of the controller. Such controller was implemented in the laboratory testbed as described in Section 4.

We implemented eight hundreds distinct experiments by imposing power failures on the motor, in which the controlled system successfully tracked the reference signal (Fig. 4)

The novelty of this paper is as follows. We use the expression of the gradient  $\varphi(\cdot)$ , as derived in Theorem 2.1, into these ten selected optimization algorithms to make a comparison of their efficiency for a specific example. In fact, the example under investigation is based on the data collected from a real-time DC Motor device subject to abrupt failures. We verified that all of the algorithms converge to the same solution (Table II), and then we implement this numerical solution in a real-time controller to control the velocity of the corresponding DC Motor device.

### 5.1. Appendix

#### Proof

(Proof of Lemma 2.1). To prove the main result, it is necessary to introduce some auxiliary results.

To begin with, notice from the formulas (6) that we can write

$$U, V \in S^r \Rightarrow \frac{\partial \text{tr}\{U(A_i + B_i G)V(A_i + B_i G)'\}}{\partial G} = 2B_i'U(A_i + B_i G)V, \quad i = 1, \dots, \sigma. \quad (19)$$

Let us now turn our attention to the recurrence (10). If we set  $k = 1$  in (10), one can verify that

$$X_{i_1}(1) = \sum_{i_0=1}^{\sigma} p_{i_0 i_1}(A_{i_0} + B_{i_0} G)X_{i_0}(0)(A_{i_0} + B_{i_0} G)', \quad i_1 = 1, \dots, \sigma.$$

With  $k = 2$  in (10), we have

$$X_{i_2}(2) = \sum_{i_1=1}^{\sigma} \sum_{i_0=1}^{\sigma} p_{i_0 i_1} p_{i_1 i_2}(A_{i_1} + B_{i_1} G)(A_{i_0} + B_{i_0} G) \times X_{i_0}(0)(A_{i_0} + B_{i_0} G)'(A_{i_1} + B_{i_1} G)', \quad i_2 = 1, \dots, \sigma.$$

Proceeding similarly with  $k = \ell + 1$  in (10), we obtain

$$X_{i_{\ell+1}}(\ell + 1) = \sum_{i_{\ell}=1}^{\sigma} \cdots \sum_{i_0=1}^{\sigma} \left( p_{i_0 i_1} \cdots p_{i_{\ell} i_{\ell+1}}(A_{i_{\ell}} + B_{i_{\ell}} G) \cdots (A_{i_0} + B_{i_0} G) \times X_{i_0}(0)(A_{i_0} + B_{i_0} G)' \cdots (A_{i_{\ell}} + B_{i_{\ell}} G)' \right), \quad i_{\ell+1} = 1, \dots, \sigma. \quad (20)$$

Combining (5) and (20), we obtain the identity

$$\begin{aligned} \frac{\partial \langle Q + G'RG, X(\ell + 1) \rangle}{\partial G} &= \sum_{i_{\ell+1}=1}^{\sigma} \frac{\partial}{\partial G} \text{tr} \{ (Q_{i_{\ell+1}} + G'R_{i_{\ell+1}}G)X_{i_{\ell+1}}(\ell + 1) \} \\ &= \sum_{i_{\ell+1}=1}^{\sigma} \cdots \sum_{i_0=1}^{\sigma} p_{i_0 i_1} \cdots p_{i_{\ell} i_{\ell+1}} \left[ \frac{\partial}{\partial G} \text{tr} \{ (Q_{i_{\ell+1}} + G'R_{i_{\ell+1}}G) \right. \\ &\quad \left. \times (A_{i_{\ell}} + B_{i_{\ell}} G) \cdots (A_{i_0} + B_{i_0} G)X_{i_0}(0)(A_{i_0} + B_{i_0} G)' \cdots (A_{i_{\ell}} + B_{i_{\ell}} G)' \} \right]. \quad (21) \end{aligned}$$

On the other hand, the derivative chain rule [23, Sec. 10.3.1] states that

$$\frac{\partial \langle Q + G'RG, X(\ell + 1) \rangle}{\partial G} = \frac{\partial \overbrace{\langle Q + G'RG, X(\ell + 1) \rangle}^{\text{variable}}}{\partial G} + \frac{\partial \overbrace{\langle Q + G'RG, X(\ell + 1) \rangle}^{\text{fixed}}}{\partial G}. \quad (22)$$

The first expression in the right-hand size of the equality (22) is identical to (see (6))

$$\sum_{i=1}^{\sigma} \frac{\partial \text{tr} \{ \overbrace{Q_i + G'R_i G}^{\text{variable}}, \overbrace{X_i(\ell + 1)}^{\text{fixed}} \}}{\partial G} = \sum_{i=1}^{\sigma} 2R_i G X_i(\ell + 1). \quad (23)$$

To evaluate the second term in the right-hand size of (22), we start with (21) taking  $Q_{i_{\ell+1}} + G'R_{i_{\ell+1}}G$  as a fixed term. The derivative chain rule will be useful in this calculation. Indeed, the idea behind the derivative chain rule is to consider (21) with  $(A_{i_0} + B_{i_0} G)$  as variable and all of the other terms fixed, and after this we take  $(A_{i_1} + B_{i_1} G)$  as variable and all of the other terms fixed, and so on until the evaluation of the term  $(A_{i_{\ell}} + B_{i_{\ell}} G)$  is accomplished.

Let us now start this procedure. Assume that  $U \in S^r$  and  $V \in S^r$  are fixed and defined in (19) as

$$\begin{aligned} U &= (A_{i_1} + B_{i_1} G)' \cdots (A_{i_{\ell}} + B_{i_{\ell}} G)'(Q_{i_{\ell+1}} + G'R_{i_{\ell+1}}G)(A_{i_{\ell}} + B_{i_{\ell}} G) \cdots (A_{i_1} + B_{i_1} G), \\ V &= X_{i_0}(0). \end{aligned}$$

Thus the term inside the brackets of (21) equals

$$\frac{\partial}{\partial G} \text{tr}\{U(A_{i_0} + B_{i_0}G)V(A_{i_0} + B_{i_0}G)'\},$$

which yields

$$2B'_{i_0} \left[ (A_{i_1} + B_{i_1}G)' \cdots (A_{i_\ell} + B_{i_\ell}G)' (Q_{i_{\ell+1}} + G'R_{i_{\ell+1}}G) \right. \\ \left. \times (A_{i_\ell} + B_{i_\ell}G) \cdots (A_{i_1} + B_{i_1}G) \right] (A_{i_0} + B_{i_0}G)X_{i_0}(0).$$

Substituting this expression into (21), we obtain

$$\sum_{i_0=1}^{\sigma} 2B'_{i_0} \left[ \sum_{i_1=1}^{\sigma} p_{i_0 i_1} (A_{i_1} + B_{i_1}G)' \cdots \sum_{i_\ell=1}^{\sigma} p_{i_{\ell-1} i_\ell} (A_{i_\ell} + B_{i_\ell}G)' \right. \\ \left. \times \sum_{i_{\ell+1}=1}^{\sigma} p_{i_\ell i_{\ell+1}} (Q_{i_{\ell+1}} + G'R_{i_{\ell+1}}G) (A_{i_\ell} + B_{i_\ell}G) \cdots (A_{i_1} + B_{i_1}G) \right] (A_{i_0} + B_{i_0}G)X_{i_0}(0).$$

Notice that the term inside the brackets is identical to  $\mathcal{E}_{i_0}(W(\ell))$ . Hence, when  $(A_{i_0} + B_{i_0}G)$  is variable and all of the other terms remain fixed, we get that

$$\frac{\partial \langle Q + G'RG, X(\ell + 1) \rangle}{\partial G} = \sum_{i_0=1}^{\sigma} 2B'_{i_0} \mathcal{E}_{i_0}(W(\ell))(A_{i_0} + B_{i_0}G)X_{i_0}(0).$$

Let us now assume that the term  $(A_{i_1} + B_{i_1}G)$  is variable and all of the others are fixed. Since (21) can be rewritten as

$$\sum_{i_{\ell+1}=1}^{\sigma} \cdots \sum_{i_1=1}^{\sigma} p_{i_1 i_2} \cdots p_{i_\ell i_{\ell+1}} \left[ \frac{\partial}{\partial G} \text{tr}\{ (Q_{i_{\ell+1}} + G'R_{i_{\ell+1}}G) \right. \\ \left. \times (A_{i_\ell} + B_{i_\ell}G) \cdots (A_{i_1} + B_{i_1}G)X_{i_1}(1)(A_{i_1} + B_{i_1}G)' \cdots (A_{i_\ell} + B_{i_\ell}G)' \} \right],$$

one can repeat the previous reasoning, taking  $(A_{i_1} + B_{i_1}G)$  as variable and all of the other terms fixed, to show that

$$\frac{\partial \langle Q + G'RG, X(\ell + 1) \rangle}{\partial G} = \sum_{i_1=1}^{\sigma} 2B'_{i_1} \mathcal{E}_{i_1}(W(\ell - 1))(A_{i_1} + B_{i_1}G)X_{i_1}(1).$$

Finally, summing up the elements from this argument, we obtain

$$\frac{\partial \langle \overbrace{Q + G'RG}^{\text{fixed}}, \overbrace{X(\ell + 1)}^{\text{variable}} \rangle}{\partial G} = \sum_{i_0=1}^{\sigma} 2B'_{i_0} \mathcal{E}_{i_0}(W(\ell))(A_{i_0} + B_{i_0}G)X_{i_0}(0) \\ + \sum_{i_1=1}^{\sigma} 2B'_{i_1} \mathcal{E}_{i_1}(W(\ell - 1))(A_{i_1} + B_{i_1}G)X_{i_1}(1) \\ \vdots \\ + \sum_{i_\ell=1}^{\sigma} 2B'_{i_\ell} \mathcal{E}_{i_\ell}(W(0))(A_{i_\ell} + B_{i_\ell}G)X_{i_\ell}(\ell). \quad (24)$$

The desired result then follows from (22), (23), and (24).  $\square$

## REFERENCES

1. E. Assunção, C. Q. Andrea, and M. C. M. Teixeira.  $H_2$  and  $H_\infty$ -optimal control for the tracking problem with zero variation. *IET Control Theory Appl.*, 1(3):682–688, 2007.
2. M. S. Bazaraa, H. D. Sherali, and C. M. Shetty. *Nonlinear Programming: Theory and Algorithms*. Wiley-Interscience, 3rd edition, 2006.
3. D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1999.
4. D. P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, 2007.
5. E. F. Costa and J. B. R. do Val. On the detectability and observability of continuous-time Markov jump linear systems. *SIAM J. Control Optim.*, 41(4):1295–1314, 2002.
6. E. F. Costa, J. B. R. do Val, and M. D. Fragoso. A new approach to detectability of discrete-time Markov jump linear systems. *SIAM J. Control Optim.*, 43(6):2132–2156, 2005.
7. E. F. Costa, A. N. Vargas, and J. B. R. do Val. Quadratic costs and second moments of jump linear systems with general Markov chain. *Math. Control Signals Systems*, 23(1):141–157, 2011.
8. O. L. V. Costa and W. L. de Paulo. Indefinite quadratic with linear costs optimal control of Markov jump with multiplicative noise systems. *Automatica*, 43:587–597, 2007.
9. O. L. V. Costa, M. D. Fragoso, and R. P. Marques. *Discrete-Time Markovian Jump Linear Systems*. Springer-Verlag, New York, USA, 2005.
10. Y.-H. Dai. Convergence properties of the BFGS algorithm. *SIAM J. Optim.*, 13(3):693–701, 2002.
11. Y. H. Dai and Y. Yuan. A nonlinear conjugate gradient method with a strong global convergence property. *SIAM J. Optim.*, 10(1):177–182, 1999.
12. J. B. R. do Val and T. Başar. Receding horizon control of jump linear systems and a macroeconomic policy problem. *J. Econom. Dynam. Control*, 23:1099–1131, 1999.
13. J. B. R. do Val, J. C. Geromel, and A. P. Gonçalves. The  $H_2$  control for jump linear systems: Cluster observations of the Markov state. *Automatica*, 38:343–349, 2002.
14. V. Dragan and T. Morozan. Exponential stability in mean square for a general class of discrete-time linear stochastic systems. *Stoch. Anal. Appl.*, 26(3):495–525, 2008.
15. J. C. Geromel, A. P. C. Gonçalves, and A. R. Fioravanti. Dynamic output feedback control of discrete-time Markov jump linear systems through linear matrix inequalities. *SIAM J. Control Optim.*, 48(2):573–593, 2009.
16. M. Khanbaghi, R. P. Malhame, and M. Perrier. Optimal white water and broke recirculation policies in paper mills via jump linear quadratic control. *IEEE Trans. Automat. Control*, 10(4):578–588, 2002.
17. A. Kojima and S. Ishijima. LQ preview synthesis: Optimal control and worst case analysis. *IEEE Trans. Automat. Control*, 44(2):352–357, 1999.
18. P. R. Kumar and P. Varaiya. *Stochastic Systems: Estimation, identification and adaptive control*. Prentice-Hall, Inc., New Jersey, 1986.
19. W. Leonhard. *Control of Electrical Drives*. Springer-Verlag, New York, 3rd edition, 2001.
20. Y. Liu and C. Storey. Efficient generalized conjugate gradient algorithms, part 1: Theory. *J. Optim. Theory Appl.*, 69:129–137, 1991.
21. I. E. Livieris and P. Pintelas. Globally convergent modified Perry’s conjugate gradient method. *Appl. Math. Comput.*, 218(18):9197–9207, 2012.
22. D. G. Luenberger and Y. Ye. *Linear and Nonlinear Programming*. Springer, 3rd edition, 2010.
23. H. Lütkepohl. *Handbook of Matrices*. John Wiley & Sons, Chichester, 1996.
24. N. Meskin and K. Khorasani. A geometric approach to fault detection and isolation of continuous-time Markovian jump linear systems. *IEEE Trans. Automat. Control*, 55(6):1343–1357, 2010.
25. R. C. L. F. Oliveira, A. N. Vargas, J. B. R. do Val, and P. L. D. Peres. Robust stability,  $H_2$  analysis and stabilisation of discrete-time Markov jump linear systems with uncertain probability matrix. *Internat. J. Control*, 82(3):470–481, 2009.
26. A. Perry. A modified conjugate gradient algorithm. *Oper. Res.*, 26(6):1073–1078, 1978.
27. C. L. Phillips and R. D. Harbor. *Feedback Control Systems*. Prentice Hall, 3rd edition, 1996.
28. A. Rubaai and R. Kotaru. Online identification and control of a DC motor using learning adaptation of neural networks. *IEEE Trans. Industry Appl.*, 36(3):935–942, 2000.
29. M. Ruderman, J. Krettek, F. Hoffmann, and T. Bertram. Optimal state space control of DC motor. In *Proc. 17th IFAC World Congress*, pages 5796–5801, Seoul, Korea, 2008.
30. A. A. G. Siqueira and M. H. Terra. A fault tolerant manipulator robot based on  $H_2$ ,  $H_\infty$ , and mixed  $H_2/H_\infty$  Markovian controls. *IEEE/ASME Trans. Mechatronics*, 14(2):257–263, 2009.
31. W. Sun and Y.-X. Yuan. *Optimization Theory and Methods: Nonlinear Programming*. Springer, 2006.
32. J. B. R. Do Val and E. F. Costa. Stabilizability and positiveness of solutions of the jump linear quadratic problem and the coupled algebraic Riccati equation. *IEEE Trans. Automat. Control*, 50(5):691–695, 2005.
33. A. N. Vargas, E. F. Costa, and J. B. R. do Val. On the control of Markov jump linear systems with no mode observation: application to a DC motor device. *Int. J. Robust Nonlinear Control*, 23(10):1136–115, 2013.
34. A. N. Vargas and J. B. R. do Val. Average cost and stability of time-varying linear systems. *IEEE Trans. Automat. Control*, 55:714–720, 2010.