Finite-Horizon LQR Controller for Partially-Observed Boolean Dynamical Systems

Mahdi Imaniand Ulisses Braga-Neto

Department of Electrical and Computer Engineering,Texas A&M University, College Station, TX, USA

Abstract

This paper proposes an approach for finite-horizon control of partially-observed Boolean dynamical systems (POBDS) with uncertain continuous control input and infinite observation space. To cope with the partial observabiity of states, the proposed method first maps the POBDS to an unnormalized belief space. The nonlinear dynamics in this continuous belief space are linearized over a nominal trajectory. Then, the optimal feedback controller is derived, based on the well-known linear quadratic regulator (LQR), to push the system to follow the nominal trajectory. This nominal trajectory is computed in a planning stage before starting execution, and updated efficiently during execution, whenever the system is found to deviate from the nominal trajectory. We prove that, under mild regularization conditions, the proposed controller approaches the cost of the nominal trajectory as the linearization error approaches zero. The performance of the proposed controller is demonstrated by numerical experiments with a Melanoma gene regulatory network observed through noisy gene expression measurements.

Key words: Finite Horizon Control, Partially-Observed Boolean Dynamical Systems, Linear Quadratic Regulator, Gene Regulatory Networks.

1 Introduction

Partially-Observed Boolean Dynamical Systems (POBDS) are nonlinear, derivativeless dynamical systems that consist of a finite Boolean state process observed through an arbitrary noisy mapping to a possibly infinite measurement space [1,2,3,4,5]. Instances of POBDS abound in fields such as genomics [6], robotics [7], digital communication systems [8], and more. The POBDS model extends other well-known models for Boolean dynamical systems, such as Boolean Networks [6] and Probabilitistic Boolean Networks (PBN) [9], by allowing noisy and incomplete observations of the Boolean system state.

A POBDS with external inputs is a partially-observed Markov decision process (POMDP). Existing infinite-horizon controllers for general POMDPs include the Q_MDP method [10], approximate dynamic programming (ADP) techniques [11], Gaussian processes (GP) and reinforcement learning (RL) [12], and point-based methodologies [13,14]. An infinite-horizon state-feedback controller for POBDS, called V_BKF, has been proposed in [15]. This method is similar to the Q_MDP method in that the control policies by both Q_MDP and V_BKF are not computed in belief space, thus, they perform poorly in domains where repeated information gathering is necessary [14]. In addition, the combination of the GP and SARSA methods has been employed in [4] for control of POBDSs with uncertain control inputs. Finally, a point-based controller was proposed in [16] for POBDSs with infinite observation spaces.

Email addresses: m.imani88@tamu.edu (Mahdi Imani), ulisses@ece.tamu.edu (Ulisses Braga-Neto).

All previously mentioned methodologies have been designed for infinite-horizon problems and cannot be used to find a finite-horizon control policy. In this paper, we propose an efficient finite-horizon feedback controller for POBDSs with continuous control input and infinite measurement space. Our method resembles the optimal Linear Quadratic Gaussian (LQG) estimator and controller for linear systems with Gaussian noise [17]. The LQG method has been extended to general POMDPs with continuous state spaces in [18,19]. These methods require linearization of state and observation models over a nominal trajectory and approximating the uncertainty of the state and observation process with Gaussian distributions. However, POBDSs are nonlinear, derivativeless dynamical systems which cannot be linearized. To overcome this difficulty, in this paper first the POBDS is mapped to a fully-observable and differentiable space, known as *unnormalized belief space*, where the system can be linearized around a nominal trajectory computed by dynamic programming, after which a linear quadratic regulator (LQR) [20] is designed to keep the system close to the nominal trajectory during the execution process. An efficient way of updating the nominal trajectory during the execution process is proposed using the principle of optimality of dynamic programming. A similar approach has been proposed in [21], in the more restricted case of directly-observable systems with differentiable transition functions.

The proposed methodology is illustrated by applying it to the control of gene regulatory networks (GRN). Most of the existing techniques for control of GRN, both in the finite and infinite horizon cases, are based on PBNs [22,23,24,25]. However, the unrealistic assumptions of direct observability of the Boolean states and finiteness of the control space

limit the practical application of these methods. By contrast, the methodology proposed here allows indirect state observability and infinite measurement and control spaces.

The article is organized as follows. In Section 2, the POBDS model considered in this paper is introduced. The finite-horizon control problem for POBDS is formulated in Section 3. Then, in Section 4, the computation of a nominal trajectory is discussed. The proposed LQR-based controller, including linearization of the system over the nominal trajectory, replaning, and performance analysis, is introduced in Section 5. A numerical analysis of the performance of the proposed controller is performed in Section 6, using a Melanoma gene regulatory network. Finally, Section 7 contains concluding remarks.

2 POBDS Model

We consider a *state process* $\{\mathbf{X}_k; k = 0, 1, \ldots, T\}$ defined over a finite time interval of length $T + 1$, where $\mathbf{X}_k \in \{0, 1\}^d$ represents the state of the system at time k. The state is affected by a sequence of control inputs $\{\mathbf{u}_k; k = 0, 1, \ldots, T-1\}$, where $\mathbf{u}_k \in \mathbb{U} = [0, 1]^d$. The value $\mathbf{u}_k(i)$ is the probability that the state of the ith Boolean variable will be flipped at time $k + 1$, for $i = 1, \ldots, d$. The control input action is therefore uncertain. Notice that the case $\mathbf{u}_k(i) = 0$ corresponds to absence of control of the ith Boolean variable at time k. The states are assumed to be updated at each discrete time through the following nonlinear signal model:

$$\mathbf{X}_k \,=\, \mathbf{f}\left(\mathbf{X}_{k-1}\right) \,\oplus\, \boldsymbol{\beta}_k(\mathbf{u}_{k-1}) \,\oplus\, \mathbf{n}_k\,, \tag{1}$$

for $k = 1, 2, \ldots, T$, where $\mathbf{f} : \{0, 1\}^d \to \{0, 1\}^d$ is a Boolean function, called the *network function*, "$\oplus$" indicates componentwise modulo-2 addition, $\boldsymbol{\beta}_k(\mathbf{u}_{k-1}) \in \{0, 1\}^d$ is a Boolean noisy input vector, such that $P(\boldsymbol{\beta}_k(i) = 1) = \mathbf{u}_{k-1}(i)$, for $i = 1, \ldots, d$, and $\mathbf{n}_k \in \{0, 1\}^d$ is a Boolean transition noise vector, such that $P(\mathbf{n}_k(i) = 1) = p$, for $i = 1, \ldots, d$. The transition noise is assumed to be *zero mode*, i.e., $0 < p < 0.5$. The closer p is to 0.5, the more chaotic the system will be, while a value of p close to zero means that the state trajectories are nearly deterministic, being governed tightly by the network function. For simplicity, $\boldsymbol{\beta}_k$ and $\mathbf{n}_k$ are assumed to have independent components, and to be independent from the initial state $\mathbf{X}_0$. In addition, $\boldsymbol{\beta}_k$ and $\boldsymbol{\beta}_l$, and $\mathbf{n}_k$ and $\mathbf{n}_l$, are assumed to be independent for $k \neq l$.

In this paper, we consider the observation model

$$\mathbf{Y}_k = \mathbf{X}_k + \mathbf{w}_k\,, \tag{2}$$

for $k = 1, \ldots, T$, where $\mathbf{Y}_k$ is the observation at time k, and $\{\mathbf{w}_k; k = 0, 1, \ldots, T\}$ is zero-mean white Gaussian noise. We assume that the components of $\mathbf{w}_k$ are independent and have the same variance σ^2.

3 Finite-Horizon Control Problem

All that is available for decision making at the current time step k are the observations $\mathbf{Y}_{1:k} = (\mathbf{Y}_1, \ldots, \mathbf{Y}_k)$, and the control inputs applied to the system up to previous time step $\mathbf{u}_{0:k-1} = (\mathbf{u}_1, \ldots, \mathbf{u}_{k-1})$. Rather than storing these values, we introduce the *unnormalized belief state* vector $\boldsymbol{\rho}_k \in \mathbb{R}^{2^d}$, given by:

$$\boldsymbol{\rho}_k(i) \,=\, p(\mathbf{Y}_k, \mathbf{X}_k = \mathbf{x}^i \mid \mathbf{Y}_{1:k-1}, \mathbf{u}_{0:k-1})\,, \tag{3}$$

for $i = 1, \ldots, 2^d$ and $k = 0, 1, \ldots, T$, where $(\mathbf{x}^1, \ldots, \mathbf{x}^{2^d})$ is an arbitrary enumeration of the possible state vectors. Note that, for $k = 0$, we have $\boldsymbol{\rho}_0(i) = P(\mathbf{X}_0 = \mathbf{x}^i)$, $i = 1, \ldots, 2^d$. It is easy to verify that the unnormalized belief state satisfies the following recursion:

$$\boldsymbol{\rho}_k \,=\, T(\mathbf{Y}_k)\, M(\mathbf{u}_{k-1})\, \frac{\boldsymbol{\rho}_{k-1}}{||\boldsymbol{\rho}_{k-1}||_1}\,, \tag{4}$$

where $||\mathbf{v}||_1 = \sum_{i=1}^{d} |\mathbf{v}(i)|$, for a vector $\mathbf{v} \in R^d$, $M(\mathbf{u}_{k-1})$ is the controlled transition matrix of the state process, with entries

$$(M(\mathbf{u}_{k-1}))_{ij} \,=\, P(\mathbf{X}_k = \mathbf{x}^i \mid \mathbf{X}_{k-1} = \mathbf{x}^j, \mathbf{u}_{k-1})\,, \tag{5}$$

for $i, j = 1, \ldots, 2^d$, and the update matrix $T(\mathbf{Y}_k)$ is a diagonal matrix of size $2^d \times 2^d$ with diagonal entries

$$\big(T(\mathbf{Y}_k)\big)_{ii} = p(\mathbf{Y}_k \mid \mathbf{X}_k = \mathbf{x}^i) \,=\, \prod_{j=1}^{d} \phi(\mathbf{Y}_k(j) - \mathbf{x}^i(j)), \tag{6}$$

for $i = 1, \ldots, 2^d$, with $\phi(w) \,=\, (2\pi\sigma^2)^{-\frac{1}{2}}\, \exp\left(-\frac{w^2}{2\,\sigma^2}\right)$.

We desire to obtain a control policy to minimize the cost function [20]:

$$J \,=\, E\left[\sum_{k=0}^{T-1} h_k(\boldsymbol{\rho}_k, \mathbf{u}_k) + h_T(\boldsymbol{\rho}_T)\right]\,, \tag{7}$$

with

$$h_k(\boldsymbol{\rho}_k, \mathbf{u}_k) \,=\, \frac{||\mathbf{c}_k(\mathbf{u}_k) \circ \boldsymbol{\rho}_k||_1}{||\boldsymbol{\rho}_k||_1}\,, \quad h_T(\boldsymbol{\rho}_T) \,=\, \frac{||\mathbf{c}_T \circ \boldsymbol{\rho}_T||_1}{||\boldsymbol{\rho}_T||_1}\,, \tag{8}$$

where "$\circ$" denotes the Hadamard (componentwise) product of vectors, and $\mathbf{c}_k(\mathbf{u}_k) = (c_k(\mathbf{x}^1, \mathbf{u}_k), \ldots, c_k(\mathbf{x}^{2^d}, \mathbf{u}_k))$ and $\mathbf{c}_T = (c_T(\mathbf{x}^1), \ldots, c_T(\mathbf{x}^{2^d}))$ are cost vectors.

4 Nominal Trajectory

Finding the minimum of (7) using classical dynamic programming techniques is an intractable problem due to the infinite spaces for the control input and unnormalized belief. In this paper, a solution is provided by finding a desirable trajectory and forcing the system to follow it during the execution process.

First we compute a nominal state-feedback control policy

via the dynamical program:

$$\boldsymbol{\mu}^*_{0:T-1} = \underset{\boldsymbol{\mu}_{0:T-1}\in\Pi}{\operatorname{argmin}} \; E\Big[\sum_{k=0}^{T-1} c_k\left(\mathbf{X}_k, \boldsymbol{\mu}_k(\mathbf{X}_k)\right) \; + \; c_T\left(\mathbf{X}_T\right)\Big],$$
$$\text{s.t.}$$
$$\mathbf{X}_k \; = \; \mathbf{f}(\mathbf{X}_{k-1}) \oplus \boldsymbol{\beta}_k(\boldsymbol{\mu}_{k-1}(\mathbf{X}_{k-1})) \oplus \mathbf{n}_k\,,$$
$$\mathbf{X}_0 \; \sim \; P(\mathbf{X}_0)\,, \tag{9}$$

where $\boldsymbol{\mu}_k \in \mathbb{U}^{2^d}$ is the control policy at time k and $\Pi \in \mathbb{U}^{2^d\times T}$ is the space of all possible policies. This problem can be solved using backward dynamic programming [20] when the control input space is finite. Thus, we obtain an approximate solution $\boldsymbol{\mu}^{*q}_{0:T-1}$ by quantizing the control input space $\mathbb{U}$ into a finite set $\mathbb{U}^q = (\mathbf{u}^q_1, \ldots, \mathbf{u}^q_m)$, and computing the optimal policy using backward dynamic programming.

Next, we construct a nominal trajectory in the Boolean state space using the policy $\boldsymbol{\mu}^{*q}_{0:T-1}$. We consider the most probable sequence of Boolean states and control inputs for the nominal trajectory. Therefore, we take the state with the largest initial probability to be the starting point of the nominal trajectory:

$$\mathbf{X}^p_0 \; = \; \underset{\mathbf{x}^i\in\{\mathbf{x}^1,\ldots,\mathbf{x}^{2^d}\}}{\operatorname{argmax}} \; P(\mathbf{X}_0 = \mathbf{x}^i)\,. \tag{10}$$

In the case of a tie, the state with the smallest index is chosen. Using this initial state and the obtained control policy, the best control input at time 0 is $\mathbf{u}^p_0 = \boldsymbol{\mu}^{*q}_0(\mathbf{X}^p_0)$. We choose the most likely state as the next state of the nominal trajectory:

$$\mathbf{X}^p_1 \; = \; \underset{\mathbf{x}\in\{\mathbf{x}^1,\ldots,\mathbf{x}^{2^d}\}}{\operatorname{argmax}} \; P(\mathbf{X}_1 = \mathbf{x}^i \mid \mathbf{X}_0 = \mathbf{X}^p_0, \mathbf{u}_0 = \mathbf{u}^p_0)\,. \tag{11}$$

Now, since the process noise is zero mode, i.e., $0 < p < 1/2$, (1) implies that

$$\mathbf{X}^p_1 \; = \; \mathbf{f}(\mathbf{X}^p_0) \oplus \bar{\mathbf{u}}^p_0\,, \tag{12}$$

where $\bar{\mathbf{u}}^p_0$ is a Boolean vector, the ith component of which is equal to 1 if $\mathbf{u}^p_0 > 0.5$, and equal to 0, otherwise. Repeating this process from time 1 to T results in the nominal trajectory $\{\mathbf{X}^p_{0:T}, \mathbf{u}^p_{0:T-1}\}$.

Finally, we obtain a nominal trajectory in the unnormalized belief space, by assuming that the most probable measurements are observed over the entire Boolean state trajectory, i.e., $\mathbf{Y}^p_k = \mathbf{X}^p_k$, for $k = 1, \ldots, T$. Letting $\boldsymbol{\rho}^p_0(i) = P(\mathbf{X}_0 = \mathbf{x}^i), i = 1, \ldots, 2^d$, we can use (4) to obtain the next unnormalized belief vector

$$\boldsymbol{\rho}^p_1 \; = \; T(\mathbf{Y}^p_1)\, M(\mathbf{u}^p_0)\, \frac{\boldsymbol{\rho}^p_0}{||\boldsymbol{\rho}^p_0||_1}\,. \tag{13}$$

Repeating this process from time 1 to T results in the nominal trajectory $\{\boldsymbol{\rho}^p_{0:T}\}$ in unnormalized belief space.

The nominal trajectory $\{\mathbf{X}^p_{0:T}, \mathbf{u}^p_{0:T-1}, \mathbf{Y}^p_{1:T}, \boldsymbol{\rho}^p_{0:T}\}$ is uniquely specified and corresponds to the most probable sequence of Boolean states, control inputs and measurements. Its cost is near optimum, since it is based on the underlying near-optimal state-feedback policy $\boldsymbol{\mu}^{*q}_{0:T-1}$.

5 Proposed LQR-based Controller

In this section, we derive an output-feedback control policy by forcing the system to be near the nominal trajectory derived in the previous section. This is accomplished by linearizing the dynamics around the nominal trajectory and then applying in the classical closed-form Linear Quadratic Regulator (LQR) solution [20] to a quadratic performance index, as described below.

5.1 Linearization around Nominal Trajectory

First, notice that the distribution of each component of the measurement vector $\mathbf{Y}_k$ given all available information up to time $k-1$ is:

$$\begin{aligned} &p(\mathbf{Y}_k(j) \mid \mathbf{Y}_{1:k-1}, \mathbf{u}_{0:k-1}) \; = \; p(\mathbf{Y}_k(j) \mid \boldsymbol{\rho}_{k-1}, \mathbf{u}_{k-1}) \\ &= \; p(\mathbf{Y}_k(j) \mid \mathbf{X}_k(j){=}1)\mathbf{z}_k(j) \\ &+ \; p(\mathbf{Y}_k(j) \mid \mathbf{X}_k(j){=}0)(1-\mathbf{z}_k(j))\,, \end{aligned} \tag{14}$$

where

$$\mathbf{z}_k(j) = P(\mathbf{X}_k(j) = 1 \mid \boldsymbol{\rho}_{k-1}, \mathbf{u}_{k-1}), \tag{15}$$

for $j = 1, \ldots, d$. It is easy to verify that the vector $\mathbf{z}_k = (\mathbf{z}_k(1), \ldots, \mathbf{z}_k(d))$ is given by

$$\mathbf{z}_k \; = \; A\, M(\mathbf{u}_{k-1})\, \frac{\boldsymbol{\rho}_{k-1}}{||\boldsymbol{\rho}_{k-1}||_1}\,, \tag{16}$$

where $A = [\mathbf{x}^1 \cdots \mathbf{x}^{2^d}]$ is a $d \times 2^d$ matrix with all possible state vectors as columns.

Now, from (2), we have $\mathbf{Y}_k(j) \mid \mathbf{X}_k(j) = i \; \sim \; i + \mathbf{w}_k(j)$. Hence, (14) implies that

$$\begin{aligned} \mathbf{Y}_k(j) \mid \boldsymbol{\rho}_{k-1}, \mathbf{u}_{k-1} &\sim (1{+}\mathbf{w}_k(j))\mathbf{z}_k(j) + \mathbf{w}_k(j)(1{-}\mathbf{z}_k(j)) \\ &= \; \mathbf{z}_k(j) + \mathbf{w}_k(j)\,, \end{aligned} \tag{17}$$

for $j = 1, \ldots, d$. We thus generate a realization $\hat{\mathbf{w}}_k$ of the noise vector $\mathbf{w}_k$ and define the approximation

$$\hat{\mathbf{Y}}_k = \mathbf{z}_k + \hat{\mathbf{w}}_k\,. \tag{18}$$

Notice the symmetry between (2) and (18). We now plug the approximation $\hat{\mathbf{Y}}_k$ into (6) to obtain the diagonal matrix $F(\mathbf{z}_k, \hat{\mathbf{w}}_k)$ of size $2^d \times 2^d$ with diagonal entries

$$\begin{aligned} F(\mathbf{z}_k, \hat{\mathbf{w}}_k)_{ii} \; &= \; p(\mathbf{z}_k + \hat{\mathbf{w}}_k \mid \mathbf{X}_k = \mathbf{x}^i) \\ &= \; \prod_{j=1}^{d} \phi(\mathbf{z}_k(j) + \hat{\mathbf{w}}_k(j) - \mathbf{x}^i(j))\,, \end{aligned} \tag{19}$$

for $i = 1, \ldots, 2^d$. By virtue of (4), we arrive finally at the unnormalized belief state equation in $\boldsymbol{\rho}_k$:

$$\begin{aligned} \boldsymbol{\rho}_k \;&\approx\; \mathbf{g}(\boldsymbol{\rho}_{k-1}, \mathbf{u}_{k-1}, \hat{\mathbf{w}}_k) \\ &=\; F(\mathbf{z}_k, \hat{\mathbf{w}}_k)\, M(\mathbf{u}_{k-1})\, \frac{\boldsymbol{\rho}_{k-1}}{||\boldsymbol{\rho}_{k-1}||_1}\,. \end{aligned} \tag{20}$$

Notice that (20) is valid in a neighborhood of the unnormalized belief state $\boldsymbol{\rho}_{k-1}$ and control input $\mathbf{u}_{k-1}$ at time step $k-1$. Thus, assuming that $\boldsymbol{\rho}_{k-1}$ and $\mathbf{u}_{k-1}$ are close enough to $\boldsymbol{\rho}^p_{k-1}$ and $\mathbf{u}^p_{k-1}$, the system can be linearized as follows. For $k = 1, \ldots, T$,

$$\tilde{\boldsymbol{\rho}}_k \;\approx\; \mathbf{A}^p_{k-1}\, \tilde{\boldsymbol{\rho}}_{k-1} \;+\; \mathbf{B}^p_{k-1}\, \tilde{\mathbf{u}}_{k-1} \;+\; \mathbf{G}^p_{k-1}\, \hat{\mathbf{w}}_k\,, \tag{21}$$

where $\tilde{\boldsymbol{\rho}}_k = \boldsymbol{\rho}_k - \boldsymbol{\rho}^p_k$, $\tilde{\mathbf{u}}_k = \mathbf{u}_k - \mathbf{u}^p_k$, $\{\hat{\mathbf{w}}_k, k = 1, \ldots, T\}$ is a zero mean i.i.d. Gaussian noise sequence, and matrices $\mathbf{A}^p_{k-1} \in \mathbb{R}^{2^d \times 2^d}$, $\mathbf{B}^p_{k-1} \in \mathbb{R}^{2^d \times d}$ and $\mathbf{G}^p_{k-1} \in \mathbb{R}^{2^d \times d}$ are

$$\mathbf{A}^p_{k-1} = \frac{\partial\, \mathbf{g}(\boldsymbol{\rho}_{k-1}, \mathbf{u}_{k-1}, \hat{\mathbf{w}}_k)}{\partial \boldsymbol{\rho}_{k-1}} \,|_{\boldsymbol{\rho}_{k-1}=\boldsymbol{\rho}^p_{k-1}, \mathbf{u}_{k-1}=\mathbf{u}^p_{k-1}, \hat{\mathbf{w}}_k=\mathbf{0}}\,, \tag{22}$$

$$\mathbf{B}^p_{k-1} = \frac{\partial \mathbf{g}(\boldsymbol{\rho}_{k-1}, \mathbf{u}_{k-1}, \hat{\mathbf{w}}_k)}{\partial \mathbf{u}_{k-1}} \,|_{\boldsymbol{\rho}_{k-1}=\boldsymbol{\rho}^p_{k-1}, \mathbf{u}_{k-1}=\mathbf{u}^p_{k-1}, \hat{\mathbf{w}}_k=\mathbf{0}}\,, \tag{23}$$

$$\mathbf{G}^p_{k-1} = \frac{\partial \mathbf{g}(\boldsymbol{\rho}_{k-1}, \mathbf{u}_{k-1}, \hat{\mathbf{w}}_k)}{\partial \hat{\mathbf{w}}_k} \,|_{\boldsymbol{\rho}_{k-1}=\boldsymbol{\rho}^p_{k-1}, \mathbf{u}_{k-1}=\mathbf{u}^p_{k-1}, \hat{\mathbf{w}}_k=\mathbf{0}}\,. \tag{24}$$

5.2 Linear Quadratic Regulator (LQR)

The goal is to select control inputs to force the system to follow the nominal trajectory during the execution process. To reach this goal, the errors $\tilde{\boldsymbol{\rho}}_k$ and $\tilde{\mathbf{u}}_k$ should be as small as possible over the interval $k = 1, \ldots, T$. This can be achieved by minimizing the following quadratic cost function:

$$J^c \;=\; E\left[\sum_{k=1}^{T} \left(\tilde{\boldsymbol{\rho}}_k^T\, \mathbf{W}^{\boldsymbol{\rho}}_k\, \tilde{\boldsymbol{\rho}}_k \;+\; \tilde{\mathbf{u}}_{k-1}^T\, \mathbf{W}^{\mathbf{u}}_k\, \tilde{\mathbf{u}}_{k-1}\right)\right], \tag{25}$$

where $\mathbf{W}^{\boldsymbol{\rho}}_{1:T}, \mathbf{W}^{\mathbf{u}}_{1:T}$ are positive definite weight matrices. Here, we use the values $\mathbf{W}^{\boldsymbol{\rho}}_k = \mathbf{I}_{2^d}/||\boldsymbol{\rho}^p_k||_1^2$ and $\mathbf{W}^{\mathbf{u}}_k = \mathbf{I}_d$.

The LQR provides the optimal solution for linear stochastic systems with quadratic cost function, such as the one defined by (21) and (25), through the following linear feedback form

$$\tilde{\mathbf{u}}_k \;=\; -\,\mathbf{L}^p_k\, \tilde{\boldsymbol{\rho}}_k \;=\; -\,\mathbf{L}^p_k(\boldsymbol{\rho}_k - \boldsymbol{\rho}^p_k)\,, \tag{26}$$

where the linear feedback gain $\mathbf{L}^p_k$ is given by

$$\mathbf{L}^p_k \;=\; \left(\mathbf{W}^{\mathbf{u}}_k \;+\; (\mathbf{B}^p_k)^T\, \mathbf{S}^p_k\, \mathbf{B}^p_k\right)^{-1} (\mathbf{B}^p_{k-1})^T\, \mathbf{S}^p_k\, \mathbf{A}^p_{k-1}\,. \tag{27}$$

The matrix $\mathbf{S}^p_k$ in (29) can be obtained using backward iteration of the dynamic Riccati equation,

$$\begin{aligned} \mathbf{S}^p_{k-1} \;=\; (\mathbf{A}^p_k)^T \Big(&\mathbf{S}^p_k \;-\; \mathbf{S}^p_k\, \mathbf{B}^p_k \\ &\left(\mathbf{W}^{\mathbf{u}}_k \;+\; (\mathbf{B}^p_k)^T\, \mathbf{S}^p_k\, \mathbf{B}^p_k\right)^{-1} (\mathbf{B}^p_k)^T\, \mathbf{S}^p_k \Big)\, \mathbf{A}^p_k + \mathbf{W}^{\boldsymbol{\rho}}_k\,, \end{aligned} \tag{28}$$

with an initial condition $\mathbf{S}^p_T = \mathbf{W}^{\boldsymbol{\rho}}_T$. The proposed control policy is then given by:

$$\mathbf{u}^L_k \;=\; \mathbf{u}^p_k \;+\; \tilde{\mathbf{u}}_k \;=\; \mathbf{u}^p_k \;-\; \mathbf{L}^p_k(\boldsymbol{\rho}_k - \boldsymbol{\rho}^p_k)\,, \tag{29}$$

for $k = 0, \ldots, T$. Note that this control policy is in the traditional form of prediction plus gain times update.

5.3 Replanning Process

The performance of the feedback controller in (29) strongly depends on the accuracy of the linearized model. The latter is a good approximation of the nonlinear system as long as the system is close enough to the nominal trajectory during the execution process. However, the stochasticity of state, measurement, and control processes precludes the possibility of seeing the nominal trajectory during execution, and the accumulation of linearization errors may push the system unnormalized belief state and control input from the nominal trajectory, rendering the linearized model inaccurate. In this case, fast recomputation of a new nominal trajectory, and pushing the system toward it are essential.

We build the replanning process based on *the principle of optimality*, which states that segments of the dynamic programming solution are themselves optimal for their subproblems [20]. This principle allows the use of the previously obtained dynamic programming solution up to the current time. Replanning to obtain a new nominal trajectory for the remaining horizon is triggered whenever the deviations

$$\epsilon^{\boldsymbol{\rho}}_{k-1} \;=\; \frac{1}{2}\left\| \frac{\boldsymbol{\rho}_{k-1}}{||\boldsymbol{\rho}_{k-1}||_1} - \frac{\boldsymbol{\rho}^p_{k-1}}{||\boldsymbol{\rho}^p_{k-1}||_1} \right\|_1, \tag{30}$$

and

$$\epsilon^{\mathbf{u}}_{k-1} \;=\; \frac{1}{d}\left\| \mathbf{u}_{k-1} - \mathbf{u}^p_{k-1} \right\|_1. \tag{31}$$

exceed prespecified values $0 \le \gamma^{\boldsymbol{\rho}}, \gamma^{\mathbf{u}} < 1$, respectively. Since the replanning process is a forward-backward process and its computation depends on the remaining horizon, a good strategy for values of γ^{ρ} and γ^{u} is to make them large at early iterations, and reduce them as the end of the horizon approaches. The rate of reduction depends on the available computational resources and time constraints, as small values trigger replanning more often.

The detailed procedure of the proposed replanning methodology is presented in Algorithm 1. Notice that the original nominal trajectory is obtained by calling this subroutine with $k = 1$.

Notice that, as a fixed input to Algorithm 1, $\boldsymbol{\mu}^{*q}_{0:T-1}$ needs to be computed only once, before the execution process,

Algorithm 1 PLANNER ($\boldsymbol{\mu}^{*q}_{k-1:T-1}, \boldsymbol{\rho}_{k-1}$)

1: Initialization:
- Initial State: $\mathbf{X}^p_{k-1} = \text{argmax}_{\mathbf{x}^i : i\in\{1,\ldots,2^d\}} \boldsymbol{\rho}_{k-1}(i)$.
- Initial Unnormalized belief: $\boldsymbol{\rho}^p_{k-1} = \boldsymbol{\rho}_{k-1}$.

For $r = k, \ldots, T$, do:
- Control Input: $\mathbf{u}^p_{r-1} = \boldsymbol{\mu}^{*q}_{r-1}(\mathbf{X}^p_{r-1})$.
- Compute $\mathbf{A}^p_{r-1}$ and $\mathbf{B}^p_{r-1}$ from (??)-(??).
- Predicted State: $\mathbf{X}^p_r = \mathbf{f}(\mathbf{X}^p_{r-1}) \oplus \bar{\boldsymbol{\beta}}_r(\mathbf{u}^p_{r-1})$.
- Unnormalized Belief:

$$\boldsymbol{\rho}^p_r = T(\mathbf{X}^p_r)\, M(\mathbf{u}^p_{r-1})\, \frac{\boldsymbol{\rho}^p_{r-1}}{||\boldsymbol{\rho}^p_{r-1}||_1}.$$

2: Set $\mathbf{W}^{\boldsymbol{\rho}}_k = \mathbf{I}_{2^d}/||\boldsymbol{\rho}^p_k||^2_1$ and $\mathbf{W}^{\mathbf{u}}_k = \mathbf{I}_d$

3: Set $\mathbf{S}^p_T = \mathbf{W}^{\boldsymbol{\rho}}_T$.

4: For $r = T, \ldots, k+1$, do:

$$\mathbf{S}^p_{r-1} = (\mathbf{A}^p_r)^T \Big(\mathbf{S}^p_r - \mathbf{S}^p_r \mathbf{B}^p_r \Big(\mathbf{W}^{\mathbf{u}}_r + (\mathbf{B}^p_r)^T \mathbf{S}^p_r \mathbf{B}^p_r\Big)^{-1} (\mathbf{B}^p_r)^T \mathbf{S}^p_r \Big) \mathbf{A}^p_r + \mathbf{W}^{\boldsymbol{\rho}}_r$$

5: OUTPUT:

$$\mathbf{A}^p_{k:T-1}, \mathbf{B}^p_{k:T-1}, \mathbf{S}^p_{k:T-1}, \boldsymbol{\rho}^p_{k-1:T}, \mathbf{u}^p_{k-1:T-1}, \mathbf{W}^{\boldsymbol{\rho}}_{k:T}, \mathbf{W}^{\mathbf{u}}_{k:T}$$

and can be reused for computation of a new nominal trajectory, Hence, the replanning process is fast, making the proposed methodology suitable for real-time applications.

5.4 Proposed Controller

Before starting the execution process, the dynamic programming solution for the underlying Boolean dynamical system with the quantized control space is computed. Then, the planner (Algorithm 1) computes the original nominal trajectory, the parameters of the linearized model, and the required parameters for the LQR controller. During the execution process, the LQR controller computes the control input to make the system close to the nominal trajectory. When deviation of the unnormalized belief state and control input from the nominal trajectory is detected, the planner computes a new nominal trajectory and the required parameters for the remaining horizon. The procedure is summarized in Figure 1 and Algorithm 2.

The following theorem, the proof of which is in the Appendix, shows that the cost of the LQR-POBDS control policy approaches the cost of the nominal policy when the linearization approximation errors approach zero over a long horizon T.

Theorem 1 *Let J be the expected cost of the proposed control policy in (29),*

$$J^L = E\left[\sum_{k=0}^{T-1} h_k(\boldsymbol{\rho}_k, \mathbf{u}^L_k) + h_T(\boldsymbol{\rho}_T)\right], \qquad (32)$$

while J^p is the cost of the nominal policy,

$$J^p = \sum_{k=0}^{T-1} h_k(\boldsymbol{\rho}^p_k, \mathbf{u}^p_k) + h_T(\boldsymbol{\rho}^p_T) \qquad (33)$$

Let $\mathbf{e}_k \in \mathbb{R}^{2^d}$ be the linearization error at time step k,

$$\mathbf{e}_k = \tilde{\boldsymbol{\rho}}_k - (\mathbf{A}^p_{k-1}\tilde{\boldsymbol{\rho}}_{k-1} + \mathbf{B}^p_{k-1}\tilde{\mathbf{u}}_{k-1} + \mathbf{G}^p_{k-1}\hat{\mathbf{w}}_k), \quad (34)$$

for $k = 1, \ldots, T$. Under mild regularity conditions,

$$J^L = J^p + O\left(\sum_{k=1}^{T} ||E[\mathbf{e}_k]||_1\right). \qquad (35)$$

In particular, if $E[\mathbf{e}_k] \approx \mathbf{0}$, for $k = 0, \ldots, T$ then $J^L \approx J^p$.

The previous theorem shows that the proposed policy achieves the desired low cost of the nominal policy as the linearization errors become negligible. Since these mainly arise from deviations of the execution trajectory from the nominal one, the replaning process is essential to keep the system continually close to the nominal trajectory during the execution process, especially in the case of longer horizons. The regularity conditions to achieve the result in Theorem 1 are mild; namely, it is required that the cost function be smooth and that $\tilde{\boldsymbol{\rho}}_k$ be a well-behaved random vector with sufficiently short tails. This ensures that the expectation of the higher order terms of the Taylor series approximation of $h_k(\boldsymbol{\rho}_k, \mathbf{u}^L_k)$ converge to zero. A result similar to Theorem 1 appears in [21, Thm 3], for the case of directly-observable systems with differentiable transition functions.

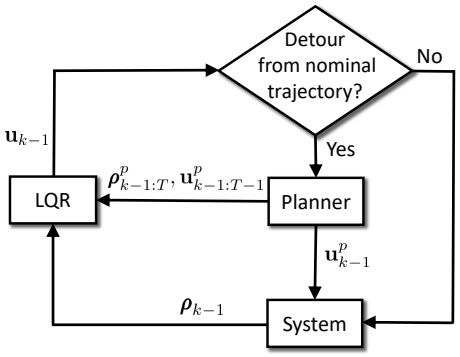

Fig. 1. Proposed LQR-POBDS Controller.

6 Numerical Experiments

In this section, we demonstrate the performance of the proposed controller via numerical experiments using a Boolean gene regulatory network involved in metastatic Melanoma [25]. The network contains 7 genes: WNT5A, pirin, S100P, RET1, MART1, HADHB and STC2. The regulatory relationship for this network is shown in Figure 2 and the Boolean function is presented in Table 1. The ith output binary string specifies the output value for ith input gene(s) in a binary representation. For example, the last row of Table 1 specifies the value of STC2 at the current time step k from different pairs of (pirin,STC2) values at the previous time step $k-1$:

$$\begin{aligned}
&(\text{pirin} = 0, \text{STC2} = 0)_{k-1} \rightarrow \text{STC2}_k = 1\\
&(\text{pirin} = 0, \text{STC2} = 1)_{k-1} \rightarrow \text{STC2}_k = 1\\
&(\text{pirin} = 1, \text{STC2} = 0)_{k-1} \rightarrow \text{STC2}_k = 0\\
&(\text{pirin} = 1, \text{STC2} = 1)_{k-1} \rightarrow \text{STC2}_k = 1
\end{aligned}$$

Algorithm 2 LQR-POBDS: Proposed Controller.

1: Initialization:
- Initial distribution: $\boldsymbol{\rho}_0(i) = P\left(\mathbf{X}_0 = \mathbf{x}^i\right)$, $i = 1, \ldots, 2^d$.
- Quantized Control Input: $\mathbb{U}^q = (\mathbf{u}_1^q, \ldots, \mathbf{u}_m^q)$.

2: Computation of optimal policy for underlying BDS.
- Terminal Cost: $J_T^{*q} = c_T$.

  For $k = T-1, \ldots, 0$, do:
- Cost Computation: for $j = 1, \ldots, 2^d$, do:

$$J_k^{*q}(j) = \min_{\mathbf{u}\in\mathbb{U}^q} \left[c_k(\mathbf{x}^j, \mathbf{u}) + \sum_{i=1}^{2^d} \left(M(\mathbf{u})\right)_{ij} J_{k+1}^{*q}(i) \right] .$$

- Policy Computation: for $j = 1, \ldots, 2^d$, do:

$$\boldsymbol{\mu}_k^{*q}(j) = \operatorname*{argmin}_{\mathbf{u}\in\mathbb{U}^q} \left[c_k(\mathbf{x}^j, \mathbf{u}) + \sum_{i=1}^{2^d} \left(M(\mathbf{u})\right)_{ij} J_{k+1}^{*q}(i) \right] .$$

3: Compute the nominal trajectory:

$$\mathbf{A}^p_{1:T-1}, \mathbf{B}^p_{1:T-1}, \mathbf{S}^p_{1:T-1}, \boldsymbol{\rho}^p_{0:T}, \mathbf{u}^p_{0:T-1}, \mathbf{W}^{\boldsymbol{\rho}}_{1:T}, \mathbf{W}^{\mathbf{u}}_{1:T-1} \leftarrow \text{PLANNER}\left(\boldsymbol{\mu}^{*q}_{0:T-1}, \boldsymbol{\rho}_0\right)$$

4: Execution Step:
- Initial control input: $\mathbf{u}_0 = \mathbf{u}_0^p$.

  For $k = 2, \ldots, T$, do:
- Update Step:

$$\boldsymbol{\rho}_{k-1} = T(\mathbf{Y}_{k-1})\, M(\mathbf{u}_{k-2})\, \boldsymbol{\rho}_{k-2} / ||\boldsymbol{\rho}_{k-2}||_1$$

- Computation of the linear feedback gain $\mathbf{L}^p_{k-1}$:

$$\mathbf{L}^p_{k-1} = \left(\mathbf{W}^{\mathbf{u}}_{k-1} + (\mathbf{B}^p_{k-1})^T \mathbf{S}^p_{k-1} \mathbf{B}^p_{k-1}\right)^{-1} (\mathbf{B}^p_{k-1})^T \mathbf{S}^p_{k-1} \mathbf{A}^p_{k-1}$$

- Control Input Selection:

$$\mathbf{u}^L_{k-1} = \mathbf{u}^p_{k-1} - \mathbf{L}^p_{k-1} \left(\boldsymbol{\rho}_{k-1} - \boldsymbol{\rho}^p_{k-1}\right)$$

- $\epsilon^{\boldsymbol{\rho}}_{k-1} = \frac{1}{2} \left\| \frac{\boldsymbol{\rho}_{k-1}}{||\boldsymbol{\rho}_{k-1}||_1} - \frac{\boldsymbol{\rho}^p_{k-1}}{||\boldsymbol{\rho}^p_{k-1}||_1} \right\|_1$, $\epsilon^{\mathbf{u}}_{k-1} = \frac{1}{d} \|\mathbf{u}_{k-1} - \mathbf{u}^p_{k-1}\|_1$.
- If $\left(\epsilon^{\boldsymbol{\rho}}_{k-1} > \gamma^{\boldsymbol{\rho}} \text{ or } \epsilon^{\mathbf{u}}_{k-1} > \gamma^{\mathbf{u}}\right)$
  - Compute a new nominal trajectory:

$$\mathbf{A}^p_{k:T-1}, \mathbf{B}^p_{k:T-1}, \mathbf{S}^p_{k:T-1}, \boldsymbol{\rho}^p_{k-1:T}, \mathbf{u}^p_{k-1:T-1}, \mathbf{W}^{\boldsymbol{\rho}}_{k:T}, \mathbf{W}^{\mathbf{u}}_{k:T-1} \leftarrow \text{PLANNER}\left(\boldsymbol{\mu}^{*q}_{k-1:T-1}, \boldsymbol{\rho}_{k-1}\right)$$

  - Control input: $\mathbf{u}_{k-1} = \mathbf{u}^p_{k-1}$.

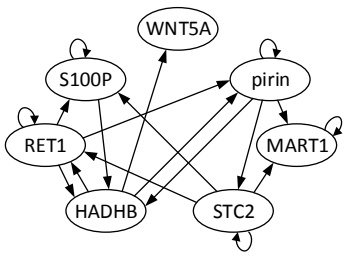

Fig. 2. Melanoma Gene Regulatory Network.

In the study conducted in [26], the expression of WNT5A was found to be highly discriminatory between cells with properties typically associated with high metastatic competence versus those with low metastatic competence. Hence, an intervention that blocked the WNT5A protein from activating its receptor could substantially reduce WNT5As ability to induce a metastatic phenotype. The study presented in [27] suggests that this can be achieved indirectly through the control of the activity of other genes. For more information, see [25].

Table 1
Boolean functions for the Melanoma Boolean network using a binary string notation (see text).

Genes	Input Gene(s)	Output
WNT5A	HADHB	10
pirin	prin, RET1,HADHB	00010111
S100P	S100P,RET1,STC2	10101010
RET1	RET1,HADHB,STC2	00001111
MART1	pirin,MART1,STC2	10101111
HADHB	pirin,S100P,RET1	01110111
STC2	pirin,STC2	1101

In summary, the goal of control is to prevent the WNT5A gene to be upregulated. This implies defining states in which WNT5A is active as undesirable states. In the numerical experiment below, either the RET1 or the HADHB gene is used as the control gene to reduce the activation of WNT5A. Thus, the intervention space is one of:

$$\begin{aligned} \mathbb{U}_{\text{RET1}} &= 0 \times 0 \times 0 \times [0\ 1] \times 0 \times 0 \times 0 \\ \mathbb{U}_{\text{HADHB}} &= 0 \times 0 \times 0 \times 0 \times 0 \times [0\ 1] \times 0\,. \end{aligned} \tag{36}$$

We have used the following cost function:

$$c_k(\mathbf{x}^i, \mathbf{u}) = \begin{cases} 5 + ||\mathbf{u}||_1 & \text{if } \mathbf{x}^i(1) = 1, \\ ||\mathbf{u}||_1 & \text{otherwise,} \end{cases} \tag{37}$$

for $k = 1, \ldots, T-1$, with terminal cost

$$c_T(\mathbf{x}^i) = \begin{cases} 5 & \text{if } \mathbf{x}^i(1) = 1, \\ 0 & \text{otherwise,} \end{cases} \tag{38}$$

where $\mathbf{x}^i(1)$ is the transcriptional state of WNT5A in state $\mathbf{x}^i$, for $i = 1, \ldots, 2^d$. This cost function was selected to penalize the application of control and the expression of WNT5A. Other cost functions could be used to accomplish the same objective; the choice in (37) and (38) was made for its simplicity.

The cost reported during the numerical experiment is

$$J = \sum_{k=0}^{T-1} \frac{||\mathbf{c}_k(\mathbf{u}_k) \circ \boldsymbol{\rho}_k||_1}{||\boldsymbol{\rho}_k||_1} + \frac{||\mathbf{c}_T \circ \boldsymbol{\rho}_T||_1}{||\boldsymbol{\rho}_T||_1}\,, \tag{39}$$

where $\boldsymbol{\rho}_k$ and $\mathbf{u}_k$ are the unnormalized belief state and control input at time step k, respectively.

The experiments were performed on a PC with an Intel Core i7- 4790 CPU@3.60 GHz clock and 16 GB of RAM. In all numerical experiments, we assume the same fixed set of values for the system parameters, summarized in Table 2.

Table 2
Parameter values for numerical experiments using the Malenoma gene regulatory network.

Parameter	**Value**
Time horizon T	50, 100
Number of genes d	7
Initial distribution $P(\mathbf{X}_0 = \mathbf{x}^i)$, $i = 1, \ldots, 128$	1/128
Control genes	RET1, HADHB
Control space $\mathbb{U}_{\text{RET1}}, \mathbb{U}_{\text{HADHB}}$	Equation (36)
Quantization level of intervention space m	10
Cost functions	Equations (37) and (38)
Transition noise intensity p	0.01, 0.05
Standard deviation σ	0.3, 0.5
Replanning thresholds $\gamma_{\boldsymbol{\rho}} = 7\,\gamma_{\mathbf{u}}$	0.1, 0.25, 0.40
Value Iteration threshold ϵ^{VI}	10^{-8}

In the first experiment, the average performance of the proposed LQR-POBDS controller is examined. Both RET1 and HADHB genes are considered for the intervention process using 500 different trajectories with length $T = 100$. The results of the LQR-POBDS are compared with four different controllers: Perseus-POBDS [16], PBVI-POBDS [16], V_BKF [15] and Q_MDP [10]. PBVI-POBDS and Perseus-POBDS are output feedback controllers designed for POBDS with infinite observation spaces, and V_BKF and Q_MDP are state-feedback controllers. The policies for all state and output feedback controllers are obtained based on the quantized intervention space. The stopping criterion threshold for the output-feedback controllers is set to be 0.05 and the sample sizes for Perseus-POBDS and PBVI-POBDS are set to be $50,000$ and 2048 respectively (see [16]). Finally, the replanning parameters for the LQR-POBDS controller are set to $\gamma_{\boldsymbol{\rho}} = 7\gamma_{\mathbf{u}} = 0.1$.

The average cost and computational time of the various controllers are presented in Table 3. It is clear that LQR-POBDS has the minimum average cost among different controllers. This can be justified by the fact that unlike the other controllers, which are developed to find the infinite-horizon control policy, LQR-POBDS is designed to deal with the finite-horizon control problem. Furthermore, it can be seen that the output-feedback controllers (Perseus-POBDS and PBVI-POBDS) perform better than state-feed back controllers (V_BKF and Q_MDP). This is due to the fact that the policy obtained by the output-feedback controllers take into account the uncertainty of the measurement, as opposed to the policy obtained by the state-feedback controllers.

Comparing computational time, it is clear that V_BKF and Q_MDP are very fast, but do not perform well, specially in the presence of large noises. LQR-POBDS displays smaller running times than the output-feedback controllers. The reason for the huge computational complexity of Perseus-POBDS and PBVI-POBDS is that both methods are based on point-based techniques, which try to approximate the whole continuous belief space with a finite number of samples.

As expected, the performance of all methods decreases as the intensity of the noise increases. This reduction in performance is lass visible for LQR-POBDS in comparison to other controllers. The reason is that during the execution process, the replanning process makes LQR-POBDS capable of changing its trajectory adaptively and efficiently. Comparing different control genes, it is clear that RET1 has better performance in reducing the activation of WNT5A in all cases.

Figure 3 displays the average cost over time for the system under control of RET1 and HADHB genes obtained by the proposed LQR-POBDS method, for $p = 0.01$ and $\sigma = 0.3$. It can be seen in Figure 3 that the minimum cost is obtained for the system under control of RET1. The system under control of HADHB also has lower cost on average in comparison to the system without control. The high cost in early iterations for the system under control is due to the uniform prior distribution considered for the initial state distribution, which makes the nominal trajectory more unreliable early, and as a result degrades the performance of control. As time goes on and more measurements are observed, the belief gets richer and as a result the decision in the unnormalized belief state becomes more accurate.

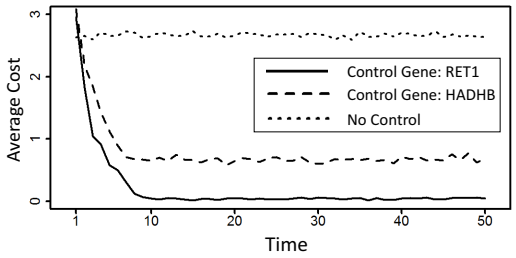

Fig. 3. Average cost under control of RET1 and HADHB genes and without control.

The state transition and the activity of the RET1 control gene, over one run of the simulation, are displayed in Figure 4. The states are numbered based on their order in Table 1 from 1 to 128. The states $1 - 64$ are states in which WNT5A is 0 (OFF) and $65 - 128$ are undesirable states in which WNT5A is 1 (ON). These two sets are separated by a horizontal line in top plot of Figure 4. The vertical dash lines in Figure 4 specify the time steps at which the replanning procedure has been performed. It can be seen that only a few undesirable states are seen after 100 time steps and, in all these occasions, the replanning procedure is executed to regenerate a new nominal trajectory and force the system to follow it. In addition, one can see more activations of the control input in time steps in which the system observes undesirables states.

The effect of having different replanning parameters on the performance of our proposed LQR-POBDS controller is examined next. Table 4 displays average results for the system under control of RET1 gene over a $T = 50$ time horizon. It can be seen that the average cost of the system increases as replanning rates grows. The reason is that small replanning rates make the nominal trajectory closer to the actual system behavior during execution process, increasing the validity of linearization process, and helping to achieve better performance of control. One can see that

Table 3
Average results for different methods.

		$p = 0.01$				$p = 0.05$			
		RET1		HADHB		RET1		HADHB	
Observation Noise	**Method**	**Cost**	**Time**	**Cost**	**Time**	**Cost**	**Time**	**Cost**	**Time**
$\sigma = 0.3$	LQR-POBDS	10.9	490.9	44.4	503.4	24.9	560.7	59.2	578.9
	Perseus-POBDS	29.7	6993.29	67.5	7080.5	50.2	7019.4	83.4	7022.8
	PBVI-POBDS	32.1	7385.6	69.3	7090.6	52.9	6927.8	88.7	6994.9
	V_BKF	33.3	2.1	75.4	2.1	59.6	2.2	99.7	2.1
	Q_MDP	33.2	2.1	75.8	2.1	61.3	2.1	101.0	2.1
$\sigma = 0.5$	LQR-POBDS	17.9	599.3	56.7	604.3	34.9	654.6	73.2	661.1
	Perseus-POBDS	38.4	7120.4	78.9	7119.7	59.4	7277.9	97.9	7223.3
	PBVI-POBDS	40.2	7230.2	82.9	7200.1	62.4	7211.0	100.3	7271.9
	V_BKF	57.3	2.1	96.1	2.1	78.6	2.1	111.7	2.1
	Q_MDP	57.5	2.1	96.5	2.1	77.4	2.1	109.8	2.1

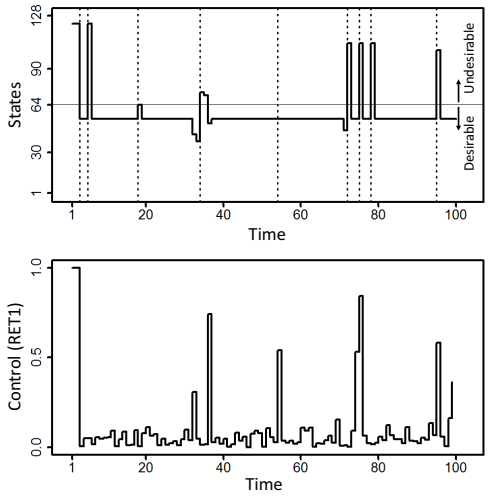

Fig. 4. State transition and the activity of the RET1 control gene over one run of the simulation.

Table 4
Average results for the LQR-POBDS controller under different replanning rates.

p	σ	$\gamma_{\boldsymbol{\rho}} = 7\gamma_{\mathbf{u}}$	Time	N_{Replan}	Cost
0.01	0.3	0.1	520	7.6	10.7
		0.25	432	6.4	13.8
		0.40	375	4.7	17.4
	0.5	0.10	583	10.1	17.2
		0.25	502	8.0	23.1
		0.40	415	6.1	26.7
0.05	0.3	0.10	554	10.7	23.9
		0.25	509	8.7	28.4
		0.40	452	7.3	39.1
	0.5	0.10	669	13.8	34.6
		0.25	602	11.3	43.8
		0.40	544	9.1	56.1

higher costs are obtained under larger process and measurement noise. Large noise intensities increase the chance of deviation from the nominal trajectory and make the control process more challenging. This effect can also be seen in the higher average number of replanning step executions, specified by N_{Replan} in Table 4. Finally, the close relationship between the number of performed replanning steps and the computational time of LQR-POBDS controller can be observed in Table 4. This highlights the importance of care in choosing appropriate replanning rates.

7 Conclusion

We proposed a methodology for finite-horizon control of partially-observed Boolean dynamical systems (POBDS) with uncertain continuous intervention and an infinite observation space. The POBDS is mapped to unnormalized belief space and the system is linearized over a nominal trajectory, which is obtained by dynamic programming methods before starting the execution process and updated efficiently afterward. The Linear Quadratic Regulator (LQR) is employed to force the system to stay close to the nominal trajectory during the execution process. The proposed controller was applied to a Boolean network model of Melanoma gene regulatory network observed through noisy gene expression data. The results demonstrate the ability of the proposed controller in reducing the average number of observed undesirable states of POBDS over a finite horizon.

Acknowledgment

The authors acknowledge the support of the National Science Foundation, through NSF award CCF-1320884.

Appendix: Proof of Theorem 1

Expanding the cost function $h_k(\boldsymbol{\rho}_k, \mathbf{u}_k^L)$ around the nominal trajectory $(\boldsymbol{\rho}_k^p, \mathbf{u}_k^p)$ using the multivariate Taylor's the-

orem, and using the fact that $\tilde{\mathbf{u}}_k = -\mathbf{L}_k^p \tilde{\boldsymbol{\rho}}_k$, leads to

$$h_k(\boldsymbol{\rho}_k, \mathbf{u}_k^L) = h_k(\boldsymbol{\rho}_k^p, \mathbf{u}_k^p) + \mathbf{C}_k \tilde{\boldsymbol{\rho}}_k + \tau(\tilde{\boldsymbol{\rho}}_k), \tag{40}$$

where $\tilde{\boldsymbol{\rho}}_k = \boldsymbol{\rho}_k - \boldsymbol{\rho}_k^p$,

$$\mathbf{C}_k = \left.\frac{\partial h_k(\boldsymbol{\rho}_k, \mathbf{u}_k)}{\partial \boldsymbol{\rho}_k}\right|_{\substack{\boldsymbol{\rho}_k=\boldsymbol{\rho}_k^p \\ \mathbf{u}_k=\mathbf{u}_k^p}} - \left.\frac{\partial h_k(\boldsymbol{\rho}_k, \mathbf{u}_k)}{\partial \mathbf{u}_k}\right|_{\substack{\boldsymbol{\rho}_k=\boldsymbol{\rho}_k^p \\ \mathbf{u}_k=\mathbf{u}_k^p}} \times \mathbf{L}_k^p, \tag{41}$$

and $\tau(\tilde{\boldsymbol{\rho}}_k)$ contains higher-order differentials of h_k and powers of $\tilde{\boldsymbol{\rho}}_k$. Provided that h_k is sufficiently smooth and that the higher-order moments of $\tilde{\boldsymbol{\rho}}_k$ are uniformly bounded, then $\tau(\tilde{\boldsymbol{\rho}}_k)$ vanishes faster than the first moment of $\tilde{\boldsymbol{\rho}}_k$. In particular, we have $E[\tau(\tilde{\boldsymbol{\rho}}_k)] = O(||E[\tilde{\boldsymbol{\rho}}_k]||_1)$.

Hence, taking expectations on both sides of (40) gives

$$\begin{aligned} E[h_k(\boldsymbol{\rho}_k, \mathbf{u}_k^L)] &= h_k(\boldsymbol{\rho}_k^p, \mathbf{u}_k^p) + \mathbf{C}_k E[\tilde{\boldsymbol{\rho}}_k] + E[\tau(\tilde{\boldsymbol{\rho}}_k)] \\ &= h_k(\boldsymbol{\rho}_k^p, \mathbf{u}_k^p) + O(||E[\tilde{\boldsymbol{\rho}}_k]||_1). \end{aligned} \tag{42}$$

Now, Notice that

$$\begin{aligned} \tilde{\boldsymbol{\rho}}_k &= \mathbf{A}_{k-1}^p \tilde{\boldsymbol{\rho}}_{k-1} + \mathbf{B}_{k-1}^p \tilde{\mathbf{u}}_{k-1} + \mathbf{G}_{k-1}^p \hat{\mathbf{w}}_k + \mathbf{e}_k \\ &= (\mathbf{A}_{k-1}^p - \mathbf{B}_{k-1}^p \mathbf{L}_{k-1}^p) \tilde{\boldsymbol{\rho}}_{k-1} + \mathbf{G}_{k-1}^p \hat{\mathbf{w}}_k + \mathbf{e}_k. \end{aligned} \tag{43}$$

Let $\mathbf{D}_k = I$, $\mathbf{D}_{k-1} = \mathbf{A}_{k-1}^p - \mathbf{B}_{k-1}^p \mathbf{L}_{k-1}^p$, and $\mathbf{D}_{k_2:k_1} = \mathbf{D}_{k_2} \times \cdots \times \mathbf{D}_{k_1}$ for $k_2 > k_1$. Iterating (43) yields

$$\tilde{\boldsymbol{\rho}}_k = \sum_{r=1}^{k} \mathbf{D}_{k:r} \mathbf{G}_{r-1}^p \hat{\mathbf{w}}_r + \sum_{r=1}^{k} \mathbf{D}_{k:r} \mathbf{e}_r, \tag{44}$$

where we used the fact that $\tilde{\boldsymbol{\rho}}_0 = \boldsymbol{\rho}_0 - \boldsymbol{\rho}_0^p = \mathbf{0}$. Hence,

$$E[\tilde{\boldsymbol{\rho}}_k] = \sum_{r=1}^{k} \mathbf{D}_{k:r} E[\mathbf{e}_r], \tag{45}$$

since the noise $\hat{\mathbf{w}}_r$ is zero-mean. Substituting this into (42),

$$E[h_k(\boldsymbol{\rho}_k, \mathbf{u}_k^L)] = h_k(\boldsymbol{\rho}_k^p, \mathbf{u}_k^p) + O\left(\sum_{r=1}^{k} ||E[\mathbf{e}_r]||_1\right). \tag{46}$$

Repeating this process for $k = 1, \ldots, T$ and adding the results yields (35).

References

[1] U. Braga-Neto, "Optimal state estimation for Boolean dynamical systems," in *Signals, Systems and Computers (ASILOMAR), 2011 Conference Record of the Forty Fifth Asilomar Conference on*, pp. 1050–1054, IEEE, 2011.

[2] M. Imani and U. Braga-Neto, "Maximum-likelihood adaptive filter for partially-observed Boolean dynamical systems," *IEEE Transactions on Signal Processing*, vol. 65, no. 2, pp. 359–371, 2017.

[3] M. Imani and U. Braga-Neto, "Particle filters for partially-observed Boolean dynamical systems," *Automatica*, 2017.

[4] M. Imani and U. Braga-Neto, "Control of gene regulatory networks with noisy measurements and uncertain inputs," *IEEE Transactions on Control of Network Systems*, 2018.

[5] M. Imani and U. Braga-Neto, "Optimal state estimation for Boolean dynamical systems using a Boolean Kalman smoother," in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 972–976, IEEE, 2015.

[6] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *Journal of theoretical biology*, vol. 22, no. 3, pp. 437–467, 1969.

[7] A. Roli, M. Manfroni, C. Pinciroli, and M. Birattari, "On the design of Boolean network robots," in *Applications of Evolutionary Computation*, pp. 43–52, Springer, 2011.

[8] D. G. Messerschmitt, "Synchronization in digital system design," *Selected Areas in Communications, IEEE Journal on*, vol. 8, no. 8, pp. 1404–1419, 1990.

[9] I. Shmulevich, E. R. Dougherty, and W. Zhang, "From Boolean to probabilistic Boolean networks as models of genetic regulatory networks," *Proceedings of the IEEE*, vol. 90, no. 11, pp. 1778–1792, 2002.

[10] A. R. Cassandra and L. P. Kaelbling, "Learning policies for partially observable environments: Scaling up," in *Machine Learning Proceedings 1995: Proceedings of the 12th International Conference on Machine Learning*, 1995.

[11] W. B. Powell, *Approximate Dynamic Programming: Solving the curses of dimensionality*, vol. 703. John Wiley & Sons, 2007.

[12] Y. Engel, S. Mannor, and R. Meir, "Reinforcement learning with Gaussian processes," in *Proceedings of the 22nd international conference on Machine learning*, pp. 201–208, ACM, 2005.

[13] J. Pineau, G. Gordon, S. Thrun, *et al.*, "Point-based value iteration: An anytime algorithm for POMDPs," in *IJCAI*, vol. 3, pp. 1025–1032, 2003.

[14] G. Shani, J. Pineau, and R. Kaplow, "A survey of point-based POMDP solvers," *Autonomous Agents and Multi-Agent Systems*, pp. 1–51, 2013.

[15] M. Imani and U. Braga-Neto, "State-feedback control of partially-observed Boolean dynamical systems using RNA-seq time series data," in *American Control Conference (ACC), 2016*, pp. 227–232, IEEE, 2016.

[16] M. Imani and U. Braga-Neto, "Point-based methodology to monitor and control gene regulatory networks via noisy measurements," *Submitted to IEEE transactions on Control Systems Technology*, 2017.

[17] K. J. Åström and P. Kumar, "Control: A perspective," *Automatica*, vol. 50, no. 1, pp. 3–43, 2014.

[18] J. Van Den Berg, P. Abbeel, and K. Goldberg, "LQG-MP: Optimized path planning for robots with motion uncertainty and imperfect state information," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 895–913, 2011.

[19] R. Platt Jr, R. Tedrake, L. Kaelbling, and T. Lozano-Perez, "Belief space planning assuming maximum likelihood observations," in *Proceedings of the Robotics: Science and Systems Conference, 6th*, 2010.

[20] D. P. Bertsekas, *Dynamic programming and optimal control*. Athena Scientific Belmont, MA, 1995.

[21] M. Rafieisakhaei, S. Chakravorty, and P. Kumar, "A near-optimal separation principle for nonlinear stochastic systems arising in robotic path planning and control," *arXiv preprint arXiv:1705.08566v1*, 2017.

[22] A. Datta, A. Choudhary, M. L. Bittner, and E. R. Dougherty, "External control in Markovian genetic regulatory networks," *Machine learning*, vol. 52, no. 1-2, pp. 169–191, 2003.

[23] R. Pal, A. Datta, and E. R. Dougherty, "Optimal infinite-horizon control for probabilistic Boolean networks," *Signal*

Processing, IEEE Transactions on, vol. 54, no. 6, pp. 2375–2387, 2006.

[24] R. Pal, A. Datta, and E. R. Dougherty, "Robust intervention in probabilistic Boolean networks," *IEEE Transactions on Signal Processing*, vol. 56, no. 3, pp. 1280–1294, 2008.

[25] E. R. Dougherty, R. Pal, X. Qian, M. L. Bittner, and A. Datta, "Stationary and structural control in gene regulatory networks: basic concepts," *International Journal of Systems Science*, vol. 41, no. 1, pp. 5–16, 2010.

[26] M. Bittner *et al.*, "Molecular classification of cutaneous malignant melanoma by gene expression profiling," *Nature*, vol. 406, no. 6795, pp. 536–540, 2000.

[27] A. T. Weeraratna, Y. Jiang, G. Hostetter, K. Rosenblatt, P. Duray, M. Bittner, and J. M. Trent, "Wnt5a signaling directly affects cell motility and invasion of metastatic melanoma," *Cancer cell*, vol. 1, no. 3, pp. 279–288, 2002.