Quotients of Probabilistic Boolean Networks

Rui Li ♦, Member, IEEE, Qi Zhang ♦, and Tianguang Chu ♦

Abstract—A probabilistic Boolean network (PBN) is a discrete-time system composed of a collection of Boolean networks between which the PBN switches in a stochastic manner. This article focuses on the study of quotients of PBNs. Given a PBN and an equivalence relation on its state set, we consider a probabilistic transition system that is generated by the PBN; the resulting quotient transition system then automatically captures the quotient behavior of this PBN. We therefore describe a method for obtaining a probabilistic Boolean system that generates the transitions of the quotient transition system. Applications of this quotient description are discussed, and it is shown that for PBNs, controller synthesis can be performed easily by first controlling a quotient system and then lifting the control law back to the original network. A biological example is given to show the usefulness of the developed results.

Index Terms—Optimal control, probabilistic Boolean networks (BNs), probabilistic transition systems, quotient, stabilization.

I. INTRODUCTION

Mathematical modeling of biological systems is a valuable avenue for understanding complex biological systems and their behaviors. One powerful approach to modeling biological systems is through a Boolean model, where each system component is characterized with a binary variable. Boolean network (BN) modeling can capture the system’s behavior without the need for much kinetic detail, making it a practical choice for systems where enough kinetic information may not be at disposal. A BN is typically placed in the form of a (deterministic) nonlinear system (with a finite state-space); while interestingly, based on an algebraic state representation approach, the Boolean dynamics can be exactly mapped into the standard discrete-time linear dynamics [1]. This formal simplicity makes it relatively easy to formulate and solve classical control-theoretic problems for BNs, and thereby has stimulated a great many interesting subsequent developments in this area [2]–[19]. For some recent work on the analysis and control of BNs based on other approaches, see, e.g., [20]–[22].

A probabilistic BN (PBN) is a stochastic extension of the classical BN. It can be considered as a collection of BNs endowed with a probability structure describing the likelihood with which a constituent network is active. PBNs possess not only the appealing properties of BNs, such as requiring few kinetic parameters, but also are able to cope with uncertainties, both in the experimental data and in the model selection [23]. The algebraic state representation has also proved a powerful framework for studying control-related problems in PBNs. Examples of recent studies based on the algebraic representation approach include investigations of network robustness and synchronization [24]–[26], controllability and stabilizability [27]–[31], observability and detectability [32]–[34], optimal control [35], just to quote a few.

It is a well-known fact that the analysis of control systems and synthesis of controllers become increasingly difficult as the dimension of the system gets larger. It is then desirable to have a methodology that reduces the size of control systems while preserving the properties relevant for analysis or synthesis. Quotient systems can be seen as lower dimensional models that may still contain enough information about the original system. A stability analysis of BNs based on a quotient map was presented in [36] and [37], where it was shown that the stability of the original BN can be inferred from the analysis of a specific quotient dynamics. Our recent work described a process for obtaining quotients of BNs [38]. A relation-based transformation strategy was introduced, which is able to transform a BN expressed in algebraic form into a quotient Boolean system suited for use. This article focuses on the study of quotients of PBNs. Given a PBN, together with an equivalence relation on the state set, we consider a probabilistic transition system $\mathcal{T}$ that is generated by the PBN. The equivalence relation then naturally induces a partition of the state-space of $\mathcal{T}$, and the corresponding quotient system fully captures the quotient dynamics of the PBN concerned. We therefore develop a probabilistic Boolean system that produces the transitions of the quotient transition system. As an application of this quotient description, we apply the proposed technique to solve two typical control problems, namely the stabilization and optimal control problems. The results show us that through the use of an appropriately defined relation, the proposed quotient system can indeed preserve the system property relevant to control design. Consequently, synthesizing controllers for a PBN can be done easily by first designing control policies on the quotient, and then inducing the control policies back to the original network.

II. NOTATION AND PRELIMINARIES

A. Notation

The symbol $\delta_i^k$ denotes the $i$th $k \times 1$ canonical basis vector (all entries of $\delta_i^k$ are 0 except for the $i$th one, which is 1). $\Delta_k$ denotes the set consisting of the canonical vectors $\delta_1^k, \ldots, \delta_k^k$, and $\mathcal{L}^{k\times r}$ denotes the set of all $k \times r$ matrices, whose columns are canonical basis vectors of length $k$. Elements of $\mathcal{L}^{k\times r}$ are called logical matrices (of size $k \times r$). A $(0,1)$-matrix is a matrix with all entries either 0 or 1. The $(i,j)$-entry of a matrix $A$ is denoted by $A_{ij}$. Given two $(0,1)$-matrices $A$ and $B$ of the same size, by $A \leq B$ we mean that if $A_{ij} = 1$, then $B_{ij} = 1$ for every $i$ and $j$. The meet of $A$ and $B$, denoted by $A \land B$, is the $(0,1)$-matrix, whose $(i,j)$-entry is $(A \downarrow B)_{ij}$. The (left) semitensor product [1] of two matrices $C$ and $D$ of sizes $k_1 \times r_1$ and $k_2 \times r_2$, respectively, denoted by $C \ltimes D$, is defined by $C \ltimes D = (C \otimes I_{r_1}(r_2)) (D \otimes I_{k_2} (r_1))$, where $\otimes$ is the Kronecker product of matrices, and $I_{r_1}$ and $I_{k_2}$ are the identity matrices of orders $r_1$ and $k_2$, respectively, with $l$ being the least common multiple of $r_1$ and $k_2$. 
B. Probabilistic Boolean Networks

A PBN is described by the following stochastic equation:

\[ X(t+1) = f_{b(t)}(X(t), U(t)) \]

where \( X(t) = \{X_1(t), \ldots, X_n(t)\} \subseteq \{1,0\}^n \) is the state, \( U(t) = \{U_1(t), \ldots, U_m(t)\} \subseteq \{1,0\}^m \) is the control, \( \{\delta(t): t=0,1,\ldots\} \) is a stochastic process consisting of independent and identically distributed (i.i.d.) random variables taking values in a finite set \( \mathbb{S} = \{1, \ldots, S\} \), and \( f_i \) (\( i=1, \ldots, S \)) are the Boolean functions from \( \{1,0\}^n \) to \( \{1,0\}^n \). By performing a matrix expression of Boolean logic and using the semitensor product, model (1) can be cast in a form similar to a random jump linear system with i.i.d. jumps.

To be more precise, we let \( x(t) = x_1(t) \times \cdots \times x_i(t) \) and \( u(t) = u_{11}(t) \times \cdots \times u_{ij}(t) \), where \( x_i(t) = \{X_i(t), \neg X_i(t)\} \) and \( u_{ij}(t) = \{U_{ij}(t), \neg U_{ij}(t)\} \). Then, it is shown that the PBN (1) satisfies the following algebraic description:

\[ x(t+1) = F_{b(t)} \times u(t) \times x(t) \]

where \( x(t) \in \Delta_N, u(t) \in \Delta_M \), and \( F_i \in \mathcal{L}^{N \times N \times M} \) for each \( i=1, \ldots, S \). With these logical matrices \( F_1, \ldots, F_S \) and positive real constants \( \lambda_1, \ldots, \lambda_S \), such that \( P = \sum_{i=1}^S \lambda_i F_i \) and \( \sum_{i=1}^S \lambda_i = 1 \). Let \( \theta(t) \) be the i.i.d. process with the probability that \( \theta(t) = i \) equal to \( \lambda_i \) for all \( t \geq 0 \). Then, the PBN described in (2) has as its transition probability matrix, the matrix \( P \).

In order to investigate quotients of (2), we first recall that every equivalence relation \( R \subseteq \Delta_N \times \Delta_N \) can be viewed as induced by a logical matrix \( C \in N \times \text{columns} \) and full row rank, by saying

\[ (x,x') \in R \iff Cx = Cx'. \]

The matrix \( C \) is easily derived from the matrix representation of \( R \).

Indeed, let \( \mathcal{A}_R \) be the \( N \times N \) matrix with entries

\[ (\mathcal{A}_R)_{ij} = \begin{cases} 1 & \text{if } (\delta_N, \delta_N') \in R \\ 0 & \text{otherwise.} \end{cases} \]

If \( C \) is a matrix having the same set of distinct rows as \( \mathcal{A}_R \), but with no rows repeated, then it must be a logical matrix of full row rank and fulfilling condition (4) (see [45, Lemma 4.6], where it is shown that such a \( C \) is a logical matrix with no zero rows, hence of full row rank, and (4) holds for that \( C \)). Note that, for an equivalence relation \( R \subseteq \Delta_N \times \Delta_N \) induced by a matrix \( C \in \mathbb{C}^{N \times N} \) of full row rank, the quotient set \( \Delta_N / R \) has cardinality \( N \), and the correspondence \( [x] \mapsto Cx \) gives a bijection between \( \Delta_N / R \) and \( \Delta_N \).

We now consider quotients of (2). The PBN (2) naturally generates a probabilistic transition system \( T(\Sigma) = (\Delta_N, \Delta_M, \rightarrow) \), where the transition relation \( \rightarrow \) is defined as follows: for \( a \in \Delta_N \), \( u \in \Delta_M \), and \( \mu \in \text{Dist}(\Delta_N) \)

\[ a \xrightarrow{u} \mu \iff \mu(x) = x^T P(u)a \text{ for all } x \in \Delta_N, \]

where \( x^T P(u)a \) is just the transition probability of \( \Sigma \) moving from \( a \) to \( x \) under input \( u \), since it coincides with the \((i,j)\)-entry of \( P(u) \) when \( x = \delta_N^i \) and \( a = \delta_N^j \). The abovementioned definition of \( \rightarrow \) then says that, for each state \( a \in \Delta_N \) and any \( u \in \Delta_M \), the probability of \( T(\Sigma) \) transitioning to the next state \( x \) is exactly the same as the probability of \( \Sigma \) transitioning from \( a \) to \( x \). Clearly, the transition system \( T(\Sigma) \) generated in this way is reactive. In view of the following discussion, we mention that the converse of this fact is also true. Indeed, given a reactive transition system \( T' = (\Delta_N, \Delta_M, \rightarrow') \), for each \( u \in \Delta_M \) define \( P'(u) \) to be the \( N \times N \) matrix with \((i,j)\)-entry \( P'(u)_{ij} = \mu(\delta_N^i \xrightarrow{u} \delta_N^j) \), where \( \mu \) is the unique probability distribution on \( \Delta_N \), such that \( \delta_N^j \xrightarrow{u} \mu \). Set

\( T \) is. Indeed, it is possible that there are two states in a class, say \([q]\), which have different probabilities of transitioning to some equivalence class under a given action, say \( \alpha \), thus, violating the abovementioned condition 2). Then, the action \( \alpha \) is not executable in \([q]\) and, consequently, the quotient transition system \( T/R \) is not reactive (recall that a necessary condition for \( T/R \) to be reactive is that every action is executable in every \([q]\) \( \in Q/R \)).

In Section III, we will use a similar framework to study quotients of a PBN.

III. Construction of Quotients

Let us consider a PBN, described by

\[ \Sigma: x(t+1) = F_{\theta(t)} \times u(t) \times x(t), x \in \Delta_N, u \in \Delta_M. \]

As assumed above, \([\theta(t)]\) is an i.i.d. process taking finitely many values \( 1, \ldots, S \) with associated probabilities \( p_1, \ldots, p_S \), and \( F_i \in \mathcal{L}^{N \times N \times M} \) for each \( 1 \leq i \leq S \). We define a column-stochastic matrix

\[ P = p_1 F_1 + p_2 F_2 + \cdots + p_S F_S \]

and for each \( u \in \Delta_M \), let

\[ P(u) = P \times u. \]

The \((i,j)\)-entry of \( P(u) \) then gives the transition probability of \( \Sigma \) from its state \( \delta_N^i \) to state \( \delta_N^j \) when input \( u \) is applied (e.g., [1]). The abovementioned matrix \( P \) is called the transition probability matrix of \( \Sigma \) [30]. Note that any column-stochastic matrix \( P \) of size \( N \times N \) can be interpreted as the transition probability matrix of a PBN of the form (2). Indeed, every column-stochastic matrix is a convex combination of logical matrices (cf. the algorithms in [43] and [44]), there exist logical matrices \( F_1, \ldots, F_S \) and positive reals \( \lambda_1, \ldots, \lambda_S \), such that \( P = \sum_{i=1}^S \lambda_i F_i \) and \( \sum_{i=1}^S \lambda_i = 1 \). Let \( \theta(t) \) be the i.i.d. process with the probability that \( \theta(t) = i \) equal to \( \lambda_i \) for all \( t \geq 0 \). Then, the PBN described in (2) has as its transition probability matrix, the matrix \( P \).

We now consider quotients of (2). The PBN (2) naturally generates a probabilistic transition system \( T(\Sigma) = (\Delta_N, \Delta_M, \rightarrow) \), where the transition relation \( \rightarrow \) is defined as follows: for \( a \in \Delta_N \), \( u \in \Delta_M \), and \( \mu \in \text{Dist}(\Delta_N) \)

\[ a \xrightarrow{u} \mu \iff \mu(x) = x^T P(u)a \text{ for all } x \in \Delta_N, \]

where \( x^T P(u)a \) is just the transition probability of \( \Sigma \) moving from \( a \) to \( x \) under input \( u \), since it coincides with the \((i,j)\)-entry of \( P(u) \) when \( x = \delta_N^i \) and \( a = \delta_N^j \). The abovementioned definition of \( \rightarrow \) then says that, for each state \( a \in \Delta_N \) and any \( u \in \Delta_M \), the probability of \( T(\Sigma) \) transitioning to the next state \( x \) is exactly the same as the probability of \( \Sigma \) transitioning from \( a \) to \( x \). Clearly, the transition system \( T(\Sigma) \) generated in this way is reactive. In view of the following discussion, we mention that the converse of this fact is also true. Indeed, given a reactive transition system \( T' = (\Delta_N, \Delta_M, \rightarrow') \), for each \( u \in \Delta_M \) define \( P'(u) \) to be the \( N \times N \) matrix with \((i,j)\)-entry \( P'(u)_{ij} = \mu(\delta_N^i \xrightarrow{u} \delta_N^j) \), where \( \mu \) is the unique probability distribution on \( \Delta_N \), such that \( \delta_N^j \xrightarrow{u} \mu \). Set

1We note that some authors use the terminology “reactive” for a probabilistic transition system where there is at most one (but perhaps no) transition on a given action from a given state.
$P' = [P'(\delta_1^1) \cdots P'(\delta_8^8)]$. Then, $P'$ is column-stochastic [since each $P'(u)$ is], and the system $T'$ can be considered as generated by a PBN whose transition probability matrix is $P'$.

Let $\mathcal{R}$ be an equivalence relation on $\Delta_N$, and consider the quotient transition system $T(\Sigma)/\mathcal{R} = (\Delta_N/\mathcal{R}, \Delta_M, R)$. For the analysis to remain in the Boolean context, we expect that the transitions of $T(\Sigma)/\mathcal{R}$ are also generated by a Boolean system\(^2\) of the form (2). By the abovementioned argument, this is the case exactly when $T(\Sigma)/\mathcal{R}$ is reactive, or equivalently, when

$$\sum_{a^t \in [a]} x^\top P(u)a = \sum_{a^t \in [a]} x^\top P(u)a' \quad \forall u \in \Delta_M \quad \forall b \in \Delta_N/\mathcal{R},$$

(5)

(that is, for any control action, states in the same class have the same transition probabilities to any equivalence class). We therefore restrict our attention to those $\mathcal{R}$ satisfying (5). The following theorem gives a method for constructing a probabilistic Boolean system that generates the transitions of $T(\Sigma)/\mathcal{R}$.

**Theorem 1:** Consider a PBN $\Sigma$ as in (2), and let $P(u)$ be as in (3). Suppose that $\mathcal{R}$ is an equivalence relation on $\Delta_N$ induced by a matrix $C \in \mathbb{L}^{N \times N}$ of full row rank, and that property (5) holds. Let $\tilde{C} \in \mathbb{L}^{N \times N}$ be such $\tilde{C} \leq C^\top$, and for each $u \in \Delta_M$ define $\tilde{P}(u)$ to be the $\tilde{N} \times \tilde{N}$ matrix given by $\tilde{P}(u) = CP(u)\tilde{C}$. Then,

1. Each $\tilde{P}(u)$ is column-stochastic.
2. Let

$$\Sigma_{\mathcal{R}} : x_R(t+1) = \tilde{P}_{\theta(t)} \times x_R(t), \quad x_R \in \Delta_N \cup \Delta_M,$$

be a probabilistic Boolean system that has $\tilde{P} = [\tilde{P}(\delta_{ij}^1) \cdots \tilde{P}(\delta_{ij}^8)]$ as its transition probability matrix. For any $a, a' \in \Delta_N$ and any $u \in \Delta_M$, the transition probability of $\Sigma_{\mathcal{R}}$ from $Ca$ to $Ca'$ under the input $u$ is equal to the transition probability of $\Sigma$ moving from $a$ to the equivalence class $[a'] = \{x \in \Delta_N : Cx = Ca'\}$ when $u$ is applied.

**Proof:** We first claim that for all $u \in \Delta_M$ and $a, a' \in \Delta_N$, we have

$$\sum_{a^t \in [a]} x^\top P(u)a = (q')^\top \tilde{P}(u)q$$

where $q = Ca$ and $q' = Ca'$. To see this, suppose that $q = \delta_i^j, q' = \delta_i^{j'}, C\delta_i^j = \delta_i^{j'}$. Then

$$(q')^\top \tilde{P}(u)q = (\tilde{P}(u))_{ij} = \sum_{k=1}^{N} (\sum_{k=1}^{N} (C)_{ik}(P(u))_{k}) (\tilde{C})_{ij} = \sum_{k=1}^{N} (C)_{ik}(P(u))_{ks}.$$  

The last equality follows since $(\tilde{C})_{ij} = 1$ exactly when $l = s$. Noting the equivalence

$$(C)_{ik} = 1 \iff C\delta_i^j = \delta_i^{j'} = q' = Ca' \iff (\delta_i^{j'}, a') \in \mathcal{R} \iff \delta_i^j \in [a'],$$

we get (7) equal to

$$\sum_{\{k, \delta_i^j \in [a']\}} (P(u))_{ks} = \sum_{\delta_i^j \in [a']} (\delta_i^j)^\top P(u)\delta_i^j.$$  

(8)

Since $\tilde{C}$ (being logical) has full row rank, the transpose $\tilde{C}^\top$ does not contain zero columns, so such $\tilde{C}$ must exist.

---

\(^2\)In the following, we use the term "probabilistic Boolean system" to refer to a stochastic system of the form (2), where $N$ and $M$ are not restricted to be powers of 2.

\(^3\)Since $\tilde{C}$ (being logical) has full row rank, the transpose $\tilde{C}^\top$ does not contain zero columns, so such $\tilde{C}$ must exist.
The state transition diagram of $\Sigma_R$ defined in Example 1.

![State transition diagram of the PBN in Example 1. A solid arrow represents the transition by the input $\delta_2^1$ and a dashed-arrow represents the transition by the input $\delta_2^3$. The number associated with each arrow denotes the probability of the state transition given the input.](image)

The state transition diagram of $\Sigma_R$ whose transition probability matrix is given by $P = [P(\delta_2^1) \ P(\delta_2^3)]$ is shown in Fig. 2. It is clear from the figure that $\Sigma_R$ is indeed a quotient of the original network, which does not distinguish between states related by $\mathcal{R}$.

Theorem 1 enables us to obtain a quotient Boolean system once an equivalence relation satisfying (5) is found. For the rest of this section, we will discuss the issue of computing equivalence relations, which allow the construction of quotient Boolean systems. More precisely, we consider the following problem, given a PBN $\Sigma$ and an equivalence relation $\mathcal{R}$ on $\Delta_N$, determine the maximal (with respect to set inclusion) equivalence relation $\mathcal{S} \subseteq \Delta_N \times \Delta_N$, such that $\mathcal{R} \subseteq \mathcal{S}$ and condition (5) holds. Here, the relation $\mathcal{S}$ may be interpreted as a preliminary classification of the states of $\Sigma$ and we focus on finding the maximal equivalence relation since in many cases we want the size of a quotient system to be as small as possible. The following theorem suggests a way of deriving such an equivalence relation. Due to space limitations, the proof of the theorem is not provided here but can be found in [46].

**Theorem 2:** Let $\Sigma$ be a PBN described by (2), and let $\mathcal{S}$ be an equivalence relation on $\Delta_N$. Define a sequence of relations $\mathcal{R}_k$ by

$$\mathcal{R}_1 = \mathcal{S} \quad \text{and} \quad \mathcal{R}_{k+1} = \left( \bigcap_{i \in [k]} \mathcal{S}_{u,i} \right) \cap \mathcal{R}_k$$

where $\mathcal{S}_{u,i}$ is the relation on $\Delta_N$ defined by $(a, a') \in \mathcal{S}_{u,i}$ if and only if $\sum_{x \in [b]} x^i P(u)a = \sum_{x \in [b]} x^i P(u)a'$ for all $[b] \in \Delta_N / \mathcal{R}_k$, with the matrix $P(u)$ given by (3).

Then,

1. The sequence of relations $\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_k, \ldots$ satisfies $\mathcal{R}_1 \supseteq \mathcal{R}_2 \supseteq \cdots \supseteq \mathcal{R}_k \supseteq \cdots$.
2. There is an integer $k^*$ such that $\mathcal{R}_{k^*+1} = \mathcal{R}_{k^*}$.
3. $\mathcal{R}_{k^*}$ is nonempty and is the maximal equivalence relation on $\Delta_N$, such that $\mathcal{R}_{k^*} \subseteq \mathcal{S}$ and property (5) holds.

Recall that a relation $\mathcal{R} \subseteq \Delta_N \times \Delta_N$ can be represented by a $(0,1)$-matrix of size $N \times N$, whose $(i,j)$-entry is 1 if and only if $(\delta_N^i, \delta_N^j) \in \mathcal{R}$. For the sake of applications, it is convenient to reformulate the abovementioned theorem in terms of $(0,1)$-matrices.

**Corollary 1:** Suppose that $\mathcal{S}$ is an equivalence relation on $\Delta_N$ represented by a matrix $A_S$. For each $u \in \Delta_M$, let $P(u)$ be as in (3). Define a sequence of $(0,1)$-matrices by

$$A_1 = A_S \quad \text{and} \quad A_{k+1} = A_k \wedge B_{k,1} \wedge \cdots \wedge B_{k,M}$$

where $B_{k,l} (l = 1, 2, \ldots, M)$ are $N \times N (0,1)$-matrices whose $(i,j)$-entry is 1 if and only if the $i$th and $j$th columns of $A_k P(\delta_N^l)$ are identical. Then, there is an integer $k^*$, such that $A_{k^*+1} = A_{k^*}$, and $A_{k^*}$ is the matrix representing the maximal equivalence relation on $\Delta_N$ that is contained in $\mathcal{S}$ and satisfies property (5).

The proof of this corollary can also be found in [46].

**Example 2:** Consider again the PBN in Example 1. If we let $\mathcal{S}$ be the equivalence relation determined by the partition $\mathcal{P} = \{\{\delta_2^1, \delta_2^3\}, \{\delta_2^2, \delta_2^4\}\}$, then $A_1 = \text{diag}(1, J_3, J_4)$, and a direct computation from Corollary 1 yields $A_2 = A_1 = \text{diag}(1, J_2, 1, J_4)$, which is precisely the matrix representing the relation given in Example 1. Hence, the relation $\mathcal{R}$ presented in Example 1 is the maximal equivalence relation contained in $\mathcal{S}$, which satisfies condition (5). We mention that here it is easy to check directly that the obtained $\mathcal{R}$ is indeed maximal. Specifically, note that any equivalence relation contained in $\mathcal{S}$ corresponds to a refinement of the partition $\mathcal{P} = \{\{\delta_2^1, \delta_2^3\}, \{\delta_2^2, \delta_2^4\}\}$. Since, for $u \in \Delta_2$, $(\delta_2^1)^T P(u)\delta_2^1 = (\delta_2^3)^T P(u)\delta_2^3 \neq 0$ while $(\delta_2^2)^T P(u)\delta_2^2 = 0$, condition (5) does not hold for any equivalence relation corresponding to a refinement of $\mathcal{P}$, in which $\delta_2^1$ and $\delta_2^3$ belong to the same block. On the other hand, we observed in Example 1 that the relation $\mathcal{R}$ produced by the partition $\{\{\delta_2^1, \delta_2^3\}, \{\delta_2^2\}\}$ fulfills (5); thus, it is the maximal equivalence relation, which is contained in $\mathcal{S}$ and satisfies (5).

To conclude, we would like to point out that the proposed method for generating a quotient of a PBN is a natural extension of the approach presented in [38] for constructing a quotient of a deterministic BN. Recall that a deterministic BN is described by $\Sigma': x(t + 1) = F \cup u(t) \times x(t), x \in \Delta_N, u \in \Delta_M, F \in \mathcal{R}^{N \times N}$.

This can be seen as a special case of (2), with $\theta(t)$ having a constant value with probability 1 for all $t \geq 0$. So the results of this section apply at once. For $u \in \Delta_M$, let $F(u)$ be defined as $P(u)$ in Theorem 1 with $P(u)$ in place of $F(u) := F \cup u$. We note that $F(u)$ has all nonnegative integer entries, and since it is column-stochastic by Theorem 1(a), every column contains exactly one nonzero entry and the nonzero entry equals 1, i.e., $F(u)$ is a logical matrix. Also, recall that the $(i,j)$-entry of $P(u)$ defined in Theorem 1 is equal to the probability with which the original network reaches the equivalence class $\{x : x(t) = \delta_N^l\}$ from an arbitrary but fixed state in $\{x : Cx = \delta_N^l\}$ when $u$ is applied (cf. Remark 1). Translated to the deterministic setting, this means that $(F(u))_{i,j} = 1$ if and only if there is a one-step transition of $\Sigma'$ from a state in $\{x : Cx = \delta_N^l\}$ to a state in $\{x : Cx = \delta_N^l\}$ under input $u$. The quotient system $x_{R}(t+1) = F \times u(t) \times x_{R}(t)$ given by Theorem 1, where $F = [F(\delta_N^1), \ldots, F(\delta_N^M)]$, then coincides precisely with the one presented in [38] Theorem 1, in which a state $\delta_N^l$ can make a transition to another state $\delta_N^k$ by applying an input exactly when that input drives $\Sigma'$ from some state in $\{x : Cx = \delta_N^l\}$ to some state in $\{x : Cx = \delta_N^k\}$.

IV. CONTROL DESIGN VIA QUOTIENTS

This section illustrates the application of quotient systems for control design. We consider two typical control problems in PBNs and show how the problems can be solved through the use of a quotient Boolean system.

A. Stabilization

Consider a PBN $\Sigma$ as in (2) and let $P(u)$ be as in (3), which gives the (one-step) transition probabilities of $\Sigma$ under input $u \in \Delta_M$. A (time-invariant) feedback controller is given by a map $\tilde{u} : \Delta_N \rightarrow \Delta_M$ so that if the present state is $x \in \Delta_N$, then the controller selects the control input $U(x) \in \Delta_M$, resulting in the matrix $P(U(x))$ that determines the one-step transition probabilities. Observe that when the present state is, say, $\delta_N^l$, only the transition probabilities of leaving $\delta_N^l$ are relevant and are given by the $l$th column of the matrix $P(U(\delta_N^l))$. We use $P_{il}$ to denote the matrix obtained by stacking such columns,
i.e., the $i$th column of $P_r$ is the $i$th column of $P(U(\delta_N))$. It is easy to see that the evolution of $\Sigma$ under the control of the state feedback controller $U : \Delta_N \rightarrow \Delta_M$ is governed by the matrix $P_r$, i.e., the transition probability from $a \in \Delta_N$ to $b \in \Delta_N$ after $k$ steps is given by $b^T P_r^k a$. Let $M \subseteq \Delta_N$ be a target set of states. The Boolean system $\Sigma$ is stabilized to $M$ with probability 1 by $U : \Delta_N \rightarrow \Delta_M$, if for every initial state $x_0 \in \Delta_N$, there exists an integer $\tau$, such that $k \geq \tau$ implies $\sum_{x \in \Delta_N} x^T P_r^k x_0 = 1$ (e.g., [47] and [30]). The following result shows that we can easily derive a stabilizing controller for $\Sigma$ on the basis of a stabilizing controller for its quotient system.

**Proposition 1:** Consider a PBN $\Sigma$ as given in (2). Let $M \subseteq \Delta_N$ and let $S$ be the equivalence relation on $\Delta_N$ determined by the partition $\{M, \Delta_N - M\}$. Suppose that $R$ is an equivalence relation on $\Delta_N$ induced by a full row rank matrix $C \in \mathbb{C}^{N \times N}$, $R \subseteq S$, and (5) holds. Suppose $\Sigma_R$ is defined as in Theorem 1 and let $M_R = \{Cx : x \in M\}$. Then,

a) There exists a control law $U : \Delta_N \rightarrow \Delta_M$ that stabilizes $\Sigma$ to $M$ with probability 1 if and only if there exists a control law $U_R : \Delta_N \rightarrow \Delta_M$ that stabilizes $\Sigma_R$ to $M_R$ with probability 1.

b) If the controller $x_0 \rightarrow U(x_0)$ stabilizes $\Sigma_R$ to $M_R$ with probability 1, then the controller given by $x \rightarrow U(x) = U_R(Cx)$ stabilizes $\Sigma$ to $M$ with probability 1.

For the proof of Proposition 1 we need the following lemma adapted from [48] (a complete proof of this lemma can also be found in [46]).

**Lemma 1:** Consider a PBN as in (2). Let $M \subseteq \Delta_N$, and let $M^*$ be the last term of the sequence $M_0 = M, M_i = M_{i-1} \cap A(M_{i-1}), i = 1, \ldots, \ell$ where $A(M_{i-1}) = \{\delta \in \Delta_N : \sum_{x \in M_{i-1}} x^T P_u(x) = 1\}$ for some $u \in \Delta_M$, and the value of $\ell$ is determined by the condition $M_{\ell+1} = M$. Define the sequence $Z_j$ according to

$$Z_j = \left\{ \delta \in \Delta_N : \sum_{x \in Z_{j-1}} x^T P_u(x) = 1 \right\}, j \geq 1.$$ 

Then $Z_j \supseteq Z_{j-1}$, and the PBN can be stabilized to $M$ with probability 1 by a feedback $U : \Delta_N \rightarrow \Delta_M$ if and only if $Z_0 = \Delta_N$ for some $\ell \geq 1$.

**Proof of Proposition 1:** (a) Let $M_i$ and $Z_j$ be as in Lemma 1. Let $M_R$ be the last term of the sequence $M_0 = M_R, M_i = M_{i-1} \cap A(M_{i-1}), i = 1, \ldots, \ell$ where $A(M_{i-1}) = \{\delta \in \Delta_N : \sum_{x \in M_{i-1}} x^T P_u(x) = 1\}$ for some $u \in \Delta_M$, and the value of $\ell$ is determined by the condition $M_{\ell+1} = M$. Define the sequence $Z_j$ according to

$$Z_j = \left\{ \delta \in \Delta_N : \sum_{x \in Z_{j-1}} x^T P_u(x) = 1 \right\}, j \geq 1.$$ 

We show that for $j \geq 0$

$$x \in Z_j \iff Cx \in Z_j,$$ 

(10)

First, we claim that

$$x \in M_i \iff Cx \in \tilde{M}_i.$$ 

(11)

Indeed, if $Cx \in \tilde{M}_0$, then there exists $x' \in M$ such that $Cx = Cx'$, and hence $(x, x') \in R \subseteq S$, forcing $x \in M$ since $S$ is the equivalence relation yielded by the partition $\{M, \Delta_N - M\}$. This shows that $Cx \in M_0 \Rightarrow x \in M_0$. The converse implication is trivial. Assume by induction that $x \in M_{i-1} \iff Cx \in \tilde{M}_{i-1}$. Denoting $I(z) = \{x \in \Delta_N : Cx = z\}$ for $z \in \Delta_N$, which is nonempty since $C$ is supposed to have full row rank, then $M_{i-1}$ can be partitioned as the disjoint union $M_{i-1} = \bigcup_{z \in \tilde{M}_{i-1}} I(z)$. Indeed, the set $I(z), \ z \in \tilde{M}_{i-1},$ is clearly mutually disjoint, and for any $x \in \Delta_N, x \in M_{i-1}$ if and only if $Cx \in \tilde{M}_{i-1}$, and if only if $x \in I(z)$ for some $z \in \tilde{M}_{i-1}$. Suppose $x \in \Delta_N$ and $u \in \Delta_M$, and let $q = Cx$. Then

$$\sum_{b \in \tilde{M}_{i-1}} b^T P_u(x) = \sum_{z \in \tilde{M}_{i-1}} \sum_{b \in \tilde{M}_{i-1}} b^T P_u(x) = \sum_{z \in \tilde{M}_{i-1}} z^T \tilde{P}(u) q$$

where the second equality follows from (6) in the proof of Theorem 1. This immediately implies that $x \in A(M_{i-1})$ if and only if $Cx \in A(\tilde{M}_{i-1})$, and hence $x \in M_i$ if and only if $Cx \in M_i$.

The proof of (10) is easily obtained by induction on $j$. It follows from (11) that $x \in Z_0$ if and only if $Cz \in \tilde{Z}_0$, establishing the base step. The induction step is similar to that done in the proof of (11).

Since $C$ is of full row rank, we conclude from (10) that $Z_j = \Delta_N$ if and only if $Z_j = \Delta_N$, and the proof of (a) follows by Lemma 1.

(b) Define the matrix $P_{\tilde{U}}^k$ for $\Sigma_R$ in the same way as $P_r$ is defined for $\Sigma$. We first prove that for any $a \in \Delta_N, z \in \Delta_N$, and integer $k \geq 1$, we have

$$\sum_{x \in z} a^T P_{\tilde{U}}^k a = z^T \tilde{P}_{\tilde{U}}^k q$$

(12)

where $I(z) = \{x \in \Delta_N : Cx = z\}$ and $q = Ca$. The proof is by induction on $k$. Since $P_r a = P(U(a)) a$ by the construction of $P_r$, it follows from (6) in the proof of Theorem 1 that

$$\sum_{x \in z} a^T P_{\tilde{U}}^k a = \sum_{x \in z} a^T P(U(a)) a = z^T \tilde{P}(U(a)) q$$

and since $U(a) = U_R(Ca) = U_R(q)$, the above is equal to $z^T \tilde{P}(\tilde{U}(q)) q = z^T \tilde{P}_{\tilde{U}}^k q$. This gives (12) for $k = 1$. Assume as induction hypothesis that the statement holds for $k - 1$. Decomposing the $N \times N$ identity matrix as $\sum_{b \in \tilde{B}} b^T b$, we have

$$\sum_{x \in z} a^T P_{\tilde{U}}^k a = \sum_{x \in z} a^T P_U \left( \sum_{b \in \tilde{B}} b^T b \right) P_{\tilde{U}}^{k-1} a$$

$$= \sum_{x \in z} \sum_{b \in \tilde{B}} a^T P_{\tilde{U}} b P_{\tilde{U}}^{k-1} a$$

$$= \sum_{b \in \tilde{B}} a^T \sum_{x \in z} P_{\tilde{U}} b P_{\tilde{U}}^{k-1} a.$$ 

(13)

The last equality holds true since $\Delta_N$ is the disjoint union of the set $I(\delta_N) = \{x : Cx = \delta_N\}$, $i = 1, \ldots, N$. It follows from the case $k = 1$ that $\sum_{x \in z} a^T P_{\tilde{U}} b = z^T \tilde{P}_{\tilde{U}}^k \delta_N$ for all $b \in I(\delta_N)$, and the right-hand side of (13) is equal to the following expression:

$$\sum_{i=1}^{N} z^T \tilde{P}_{\tilde{U}}^k \delta_N^i = \left( \sum_{b \in \tilde{B}} b^T b \right) P_{\tilde{U}}^{k-1} a.$$ 

(14)

According to the induction hypothesis, we have for each $1 \leq i \leq N$

$$\sum_{b \in I(\delta_N)} b^T P_{\tilde{U}}^{k-1} a = (\delta_N^i)^T \tilde{P}_{\tilde{U}}^k q$$

and substituting this into (14) we get

$$\sum_{x \in z} a^T P_{\tilde{U}}^k a = \sum_{i=1}^{N} z^T \tilde{P}_{\tilde{U}}^k \delta_N^i (\delta_N^i)^T \tilde{P}_{\tilde{U}}^k q$$

$$= z^T \tilde{P}_{\tilde{U}}^k \left( \sum_{i=1}^{N} \delta_N^i (\delta_N^i)^T \right) \tilde{P}_{\tilde{U}}^k q = z^T \tilde{P}_{\tilde{U}}^k q$$

which is (12).
From the proof of (a), we know that $x \in M$ if and only if $Cx \in \mathcal{M}_R$, and consequently, we can write $M$ as the disjoint union $M = \bigcup_{x \in \mathcal{M}_R} I(x)$. The proof of (b) is now obvious. Suppose $x_0 \in \Delta_N$. Let $x_{R0}^* = Cx_0$. Then, for each integer $k \geq 1$ we have

$$\sum_{x \in \mathcal{M}} x^T P_{R}^k x_0 = \sum_{x \in \mathcal{M}_R} \sum_{x \in I(x)} x^T P_{R}^k x_0 = \sum_{x \in \mathcal{M}_R} x^T P_{R}^k x_{R0}^*$$

from which (b) follows immediately.

### B. Optimal Control

Let us consider the following optimal control problem, introduced in [49].

**Problem 1:** Consider a PBN as in (2). Given an initial state $x_0 \in \Delta_N$ and a finite time horizon $T \in \mathbb{Z}^+$, find a control policy, $u(t) = U(t, x(t))$ for $0 \leq t \leq T - 1$, that minimizes the cost functional

$$J = \mathbb{E} \left[ \sum_{t=0}^{T-1} l(u(t), x(t)) \right]$$

where $l(u, x)$ and $g(x)$ are real-valued functions defined on $\Delta_M \times \Delta_N$ and $\Delta_N$, respectively.

We show that the solution to Problem 1 can be found by considering the problem for a suitably chosen quotient system. To this end, let $S$ be the equivalence relation on $\Delta_N$, given by

$$(x, x') \in S \iff g(x) = g(x')$$

and $l(u, x) = l(u, x')$ for all $u \in \Delta_M$. (15)

We note that if $C \in \mathbb{L}^{N \times N}$ has full row rank and if the equivalence relation $\mathcal{R}$ induced by $C$ satisfies $\mathcal{R} \subseteq S$, then every $x \in \Delta_N$ can be written as $x = Cx$ for some $x \in \Delta_N$ and the function $g$ is constant on the set $\{x \in \Delta_N : Cx = x\}$. Hence, the map $g_R : \Delta_N \to \mathbb{R}$, defined by

$$g_R(z) = g(Cz) = g(x)$$

is well defined. For the same reason, the map $l_R : \Delta_M \times \Delta_N \to \mathbb{R}$ defined by

$$l_R(u, z) = l_R(u, Cx) = l(u, x)$$

is also well-defined. We can state the following proposition.

**Proposition 2:** Let $S$ be a PBN described by (2), and consider Problem 1 with given $x_0$ and $T$. Suppose that $S$ is the equivalence relation given by (15), $\mathcal{R} \subseteq \Delta_N \times \Delta_N$ is an equivalence relation induced by a full row rank matrix $C \in \mathbb{L}^{N \times N}$, $\mathcal{R} \subseteq S$, and (5) holds. Let $\mathcal{R}_S$ be the probabilistic Boolean system constructed in Theorem 1, and define $J_R = \mathbb{E} \left[ \sum_{t=0}^{T-1} l_R(u(t), x_R(t)) \right]$, where $g_R$ and $l_R$ are given by (16) and (17), respectively. Suppose that $(t, x_0) \to U_0(t, x_0)$ is an optimal control policy for $\mathcal{R}_S$. Moreover, let $J^*$ be the optimal value of the initial state $x_0$, and let $J_0$ be the optimal value of $J_R$ associated with $x_{R0}^* = Cx_0$. Then, $J^* = J_0$.

The proof of the proposition follows from the following two lemmas, whose proofs are omitted due to limited space and are available in the online version of the article in [46].

**Lemma 2:** Consider Problem 1 with given $x_0$ and $T$. Let $S$, $\mathcal{R}$, and $C$ be as in Proposition 2. Then, there exists an optimal control policy $(t, x_0) \to \hat{U}(t, x_0)$ with the property that $\hat{U}(t, x) = U(t, x)$ for all $0 \leq t \leq T - 1$ and all $x, x' \in \Delta_N$, such that $C_{x} = C_{x'}$.

**Lemma 3:** Let the notation be as in the statement of Proposition 2. If the initial states of $\mathcal{S}$ and $\mathcal{S}_R$ satisfy $C_{x_0} = x_{R0}^*$, and if the two control policies $(t, x_0) \to \hat{U}(t, x)$ and $(t, x_R) \to \hat{U}(t, x_R)$ satisfy $\hat{U}(t, x) = \hat{U}(t, x)$ for all $0 \leq t \leq T - 1$ and $x \in \Delta_N$, then the cost functionals $J$ and $J_R$ have the same value.
TABLE I

| Variable | Boolean Function |
|----------|------------------|
| $M_{ac}$ | $C_{ap} \land \neg R \land \neg R_{mn}$ |
| $P_{ac}$ | $M_{ac}$ |
| $B$ | $M_{ac}$ |
| $C_{ap}$ | $\neg G_{e}$ |
| $R$ | $\neg A \land \neg A_{mn}$ |
| $R_{mn}$ | $(-A \land \neg A_{mn}) \lor R$ |
| $A$ | $B \land L$ |
| $A_{mn}$ | $L \lor L_{m}$ |
| $L$ | $P_{ac} \land L_{e} \land \neg G_{e}$ |
| $L_{em}$ | $((L_{em} \land P_{ac}) \lor L_{e}) \land \neg G_{e}$ |

V. A BIOLOGICAL EXAMPLE

A Boolean model for the lac operon in Escherichia coli was identified in [50]. The model consists of 13 variables (one mRNA, five proteins, and seven sugars) denoted by $M_{ac}, P_{ac}, B, C_{ap}, R, A, A_{mn}, L, L_{e}, L_{em}, L_{mn},$ and $G_{e}$. The Boolean functions of the model are given in Table I. We assume that the concentration of extracellular lactose (indicated by $L_{e}$ and $L_{em}$) can be either low or medium,4 causing the model to appear random. We then arrive at a PBN consisting of two BNs. The first constituent BN is determined from Table I when $L_{e} = L_{em} = 0,$ and the second constituent BN is determined by setting $L_{e} = 0$ and $L_{em} = 1$. The two constituent BNs are assumed to be equally likely. The concentration level of extracellular glucose ($G_{e}$) acts as the control input. The algebraic representation of the PBN is as in (2), with $N = 1024$ and $M = 2$, and the selection probabilities given by $p_{1} = p_{2} = 0.5$. The matrices $F_{1}, F_{2} \in \mathbb{L}^{1024 \times 1024}$ are not presented explicitly due to their sizes.

1) Stabilization: When extracellular lactose is low, the lac operon model is known to exhibit two steady states [50], expressed in the canonical vector form as $\delta_{1024}^{0}$ and $\delta_{1024}^{1}$. Let $M = \{\delta_{1024}^{0}, \delta_{1024}^{1}\}$, and let $S$ be the equivalence relation produced by the partition $(M, \Delta_{1024} - M)$. Then, by following the procedure described in Section III, we obtain a quotient system $\Sigma_{R}$ with the transition probability matrix, given by

$$P = \begin{pmatrix} \delta_{1}^{0} & \delta_{1}^{1} & \ldots & \delta_{1}^{5} \\ \delta_{2}^{0} & \delta_{2}^{1} & \ldots & \delta_{2}^{5} \\ \vdots & \vdots & \ddots & \vdots \\ \delta_{5}^{0} & \delta_{5}^{1} & \ldots & \delta_{5}^{5} \end{pmatrix}.$$ 

$\delta_{1}^{0}$ with probability 1 via the feedback law $x_{k} \mapsto Kx_{k}$, where $K \in \mathbb{L}^{5 \times 5}$ has $\delta_{2}^{1}$ as the first and fourth columns and $\delta_{3}^{2}$ as its other columns. Proposition 1 then ensures that the feedback law $x \mapsto U(x) = Kx$ stabilizes the original PBN to the state $\delta_{1024}^{0}$ with probability 1. Specifically, this controller is given as $U(x) = \delta_{2}^{1}$ if $x \in L_{796}^{0} \cup L_{800}^{0} \cup L_{906}^{0} \cup L_{912}^{0} \cup L_{1024}^{0} \cup \{L_{1024}^{1}, L_{1024}^{2}, L_{1024}^{3}, L_{1024}^{4}, L_{1024}^{5}, L_{1024}^{6}, L_{1024}^{7}, L_{1024}^{8}\}$ and $(U(x)) = \delta_{2}^{1}$ otherwise. A similar argument can be made for finding a feedback controller that stabilizes the PBN to the state $\delta_{1024}^{0}$; the details are not repeated here.

2) Optimal Control: Assume that $T = 10$ and $x_{0} = \delta_{1024}^{0}$, and the functions $l(u, x)$ and $g(x)$ are given by $l(\delta_{1024}^{0}, x) = 1, l(\delta_{1024}^{1}, x) = 0, x \in \Delta_{1024}, g(\delta_{1024}^{0}) = \cdots = g(\delta_{1024}^{5}) = 3$, and $g(\delta_{1024}^{0}) = \cdots = g(\delta_{1024}^{5}) = 6$. Here, we mention that $\delta_{1024}^{0}, \ldots, \delta_{1024}^{5}$ are exactly the states corresponding to the lac operon being ON (cf. [50]). The abovementioned choice of $g(x)$ then indicates that ON states are more desirable. By proceeding as in Section IV-B, one can obtain a quotient system $\Sigma_{R}$ with the transition probability matrix, given by

$$\tilde{P} = \begin{pmatrix} \delta_{1}^{0} & \delta_{1}^{1} & \ldots & \delta_{1}^{5} \\ \delta_{2}^{0} & \delta_{2}^{1} & \ldots & \delta_{2}^{5} \\ \vdots & \vdots & \ddots & \vdots \\ \delta_{5}^{0} & \delta_{5}^{1} & \ldots & \delta_{5}^{5} \end{pmatrix}.$$ 

Note that the size of $\Sigma_{R}$ is less than 2.5% when compared to the original model. The matrix $C_{R} = \delta_{1024}^{25}$, and the induced functions $l_{R}$ and $g_{R}$ are defined by $l_{R}(\delta_{R}, x_{R}) = 1, l_{R}(\delta_{R}, x_{R}) = 0, x_{R} \in \Delta_{25}, g_{R}(\delta_{R}) = \cdots = g_{R}(\delta_{R}) = 6$, and $g_{R}(\delta_{R}) = \cdots = g_{R}(\delta_{R}) = 3$. It is not hard to see that the constant control $u = \delta_{25}^{5}$ is optimal for $\Sigma_{R}$, with the optimal cost $J_{R} = 5.9063$ (to which corresponds $x_{R} = \delta_{25}^{5}$). Thus, by virtue of Proposition 2, this constant input also solves the optimal control problem for the original PBN, and the optimal cost corresponding to the initial state $x_{0} = \delta_{1024}^{0}$ is $J = J_{R} = 5.9063$.

4Recall that the optimal control problem for $\Sigma_{R}$ can be solved by the following dynamic programming algorithm, which proceeds backward in time from $t = 10$ to $t = 0$ (e.g., [49]):

$$H(10, x_{R}) = g_{R}(x_{R}), x_{R} \in \Delta_{25}$$

$$H(t, x_{R}) = \min_{u \in \Delta_{25}} \left\{ l_{R}(u, x_{R}) + \sum_{\xi \in \Delta_{25}} H(t + 1, \xi)^{T} \tilde{P}(u) x_{R} \right\}$$

where $\tilde{P}(u) = \tilde{P} \times u$ for $u \in \Delta_{25}$. The optimal control law is obtained as

$$U_{R}(t, x_{R}) = \arg \min_{u \in \Delta_{25}} \left\{ l_{R}(u, x_{R}) + \sum_{\xi \in \Delta_{25}} H(t + 1, \xi)^{T} \tilde{P}(u) x_{R} \right\}.$$
REFERENCES

[1] D. Cheng, H. Qi, and Z. Li, *Analysis and Control of Boolean Networks: A Semi-Product Approach*. London, U.K.: Springer-Verlag, 2011.

[2] D. Laschov and M. Margaliot, “On Boolean control networks with maximal topological entropy,” *Automatica*, vol. 50, no. 11, pp. 2924–2928, 2014.

[3] N. Bof, E. Fornasini, and M. E. Valcher, “Output feedback stabilization of Boolean control networks,” *Automatica*, vol. 57, pp. 21–28, 2015.

[4] E. Fornasini and M. E. Valcher, “Fault detection analysis of Boolean control networks,” *IEEE Trans. Autom. Control*, vol. 60, no. 10, pp. 2734–2739, Oct. 2015.

[5] J. Lu, J. Zhong, C. Huang, and J. Cao, “On pinning controllability of Boolean control networks,” *IEEE Trans. Autom. Control*, vol. 61, no. 6, pp. 1658–1663, Jun. 2016.

[6] K. Zhang and L. Zhang, “Observability of Boolean control networks: A unified approach based on finite automata,” *IEEE Trans. Autom. Control*, vol. 61, no. 9, pp. 2733–2738, Sep. 2016.

[7] Y. Guo, Y. Ding, and D. Xie, “Invariant subset and set stability of Boolean networks under arbitrary switching signals,” *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 4209–4214, Aug. 2017.

[8] F. Li and Y. Tang, “Set stabilization for switched Boolean control networks,” *Math. Biosci.*, vol. 289, pp. 223–230, 2017.

[9] J. Liang, H. Chen, and Y. Liu, “On algorithms for state feedback stabilization of Boolean control networks,” *Automatica*, vol. 84, pp. 10–16, 2017.

[10] Y. Liu, B. Li, J. Lu, and J. Cao, “Pinning control for the disturbance decoupling problem of Boolean networks,” *IEEE Trans. Autom. Control*, vol. 62, no. 12, pp. 6595–6601, Dec. 2017.

[11] D. Cheng, C. Li, and H. Fu, “Observability of Boolean networks via set controllability approach,” *Syst. Control Lett.*, vol. 115, pp. 22–25, 2018.

[12] M. R. Rafimanzelat and F. Bahrami, “Attractor stabilizability of Boolean control networks,” *IEEE Trans. Autom. Control*, vol. 4, no. 2, pp. 319–324, Apr. 2020.

[13] B. Wang and J. Feng, “On detectability of probabilistic Boolean networks,” *Inf. Sci.*, vol. 483, pp. 383–395, 2019.

[14] Y. Wu, Y. Guo, and M. Toyoda, “Policy iteration approach to the infinite horizon average optimal control of probabilistic Boolean networks,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 7, pp. 2910–2924, Jul. 2021.

[15] P. Wang and Y. Guo, “Set stability of Boolean networks via quotient mappings,” in *Proc. 26th Chin. Control Decis. Conf.*, 2014, pp. 912–917.

[16] Y. Guo, “Stability of Boolean networks with state-triggered impulses based on a 2-D index model,” in *Proc. 37th Chin. Control Conf.*, 2018, pp. 1982–1987.

[17] R. Li, Q. Zhang, and T. Chu, “On quotients of Boolean control networks,” *Automatica*, vol. 125, 2021, Art. no. 109401.

[18] C. Baier and M. Kwiatkowska, “Domain equations for probabilistic processes,” *Math. Struct. Comput. Sci.*, vol. 10, no. 6, pp. 665–717, 2000.

[19] H. Hermans, A. Parma, R. Segala, B. Wachter, and L. Zhang, “Probabilistic logical characterization,” *Inf. Comput.*, vol. 209, no. 2, pp. 154–172, 2011.

[20] Y. Feng and L. Zhang, “When equivalence and bisimulation join forces in probabilistic automata,” in *Proc. 19th Int. Symp. Formal Methods*, 2014, pp. 247–262.

[21] L. Zhang and D. N. Jansen, “A space-efficient simulation algorithm on probabilistic automata,” *Inf. Comput.*, vol. 249, pp. 138–159, 2016.

[22] J. K. Ching, X. Chen, and N.-K. Tsing, “Generating probabilistic Boolean networks from a prescribable transition probability matrix,” *JET Syst. Biol.*, vol. 3, no. 6, pp. 453–464, 2009.

[23] K. Kobayashi and K. Hirota, “Design of probabilistic Boolean networks based on network structure and steady-state probabilities,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 8, pp. 1966–1971, Aug. 2017.

[24] R. Li, Q. Zhang, and T. Chu, “Reduction and analysis of Boolean control networks by bisimulation,” *SIAM J. Control Optim.*, vol. 59, no. 2, pp. 1033–1056, 2021.

[25] R. Li, Q. Zhang, and T. Chu, “Quotients of probabilistic Boolean networks,” 2021, arXiv:2107.14466.

[26] R. Li, M. Yang, and T. Chu, “State feedback stabilization for probabilistic Boolean networks,” *Automatica*, vol. 50, no. 4, pp. 1272–1278, 2014.

[27] R. Li, M. Yang, and T. Chu, “Feedback stabilization of a probabilistic Boolean network to a given set of states,” *Chin. J. Syst. Sci. Math. Sci.*, vol. 36, no. 3, pp. 371–380, 2016.

[28] A. Datta, A. Choudhary, M. L. Bittner, and E. R. Dougherty, “External control in Markovian genetic regulatory networks,” *Mach. Learn.*, vol. 52, no. 1–2, pp. 169–191, 2003.

[29] A. Veliz-Cuba and B. Stigler, “Boolean models can explain bistability in the Lac operon,” *J. Comput. Biol.*, vol. 18, no. 6, pp. 783–794, 2011.