

確率ブーリアンネットワークの 解析と制御のための計算手法

小林 孝一^{†1} 平石 邦彦^{†1}

確率ブーリアンネットワーク (PBN) は、遺伝子ネットワークに代表される生体ネットワークのモデルの一つであり、盛んに研究が行われている。本論文では、PBN の可到達性および安全性の解析と最適制御のための計算手法を提案する。最適制御では、整数計画法を用いた手法を提案する。可到達性および安全性の解析では、確率モデル検査ツールを用いた手法を提案する。

Computational Techniques for Analysis and Control of Probabilistic Boolean Networks

KOICHI KOBAYASHI^{†1} and KUNIHICO HIRAISHI^{†1}

A probabilistic Boolean network (PBN) is one of the models of biological networks such as gene regulatory networks, and has been extensively studied. In this paper, computational techniques for reachability/safety analysis and optimal control are proposed. In optimal control, an integer programming-based method is proposed. In reachability/safety analysis, a model checking approach is proposed.

1. はじめに

システムバイオロジーの主要な研究課題の一つとして、遺伝子ネットワークや代謝ネットワークに代表される生体ネットワークのモデリング、解析、制御に関する研究が挙げられる。一般に、生体ネットワークは高い非線形をもつ高次元の常微分または偏微分方程式によって

表現できるが、解析や制御を考える場合は簡便なモデルを利用することが適切である。とくに、制御問題に関しては、ブーリアンネットワーク、ハイブリッドシステムが利用されている^{1),3),4),12),13)}。しかしながら、問題を解くための計算時間の影響により、ハイブリッドシステムの適用は低次元の対象に制限される^{3),4)}。一方、ブーリアンネットワークはブール関数を用いて遺伝子などの相互作用を表現したモデルである¹¹⁾。したがって、単純化され過ぎているという問題点はあるが、大規模システムを対象とすることができるという長所があり、盛んに研究が行われている^{2),10),14),17)}。さらに、実際の遺伝子ネットワークは雑音などの影響により確率的に動作していると考えられることから、ブール関数が確率的に選択される確率ブーリアンネットワークが提案されている¹⁶⁾。そこで本論文では、大規模システムへの適用および実用性を考慮し、確率ブーリアンネットワークの解析および制御を考える。

確率ブーリアンネットワークの解析および制御はこれまでに多くの研究が行われている^{6),8),9),15)}。しかしながら、いずれの手法においても、制御入力を求めるためには、 n 次元の状態に対し、 2^n 個のノードをもつ状態遷移図 (離散時間マルコフ連鎖) を計算する必要がある。したがって、既存手法を大規模システムへ直接適用することは困難である。

本論文では、状態遷移図を計算せずに解析や制御をおこなう新しい計算手法を提案する。制御では最適制御問題を考え、評価関数の上下限を基に制御性能を評価することとする。まず、与えられているブール関数を用いて、線形不等式拘束をもつ線形状態方程式を導出する。このモデルを用いることで、与えられている評価関数の下限を最小化する制御入力を求める問題は、整数線形計画 (ILP) 問題に帰着される。さらに、得られた制御入力を用いたときに達成可能な評価関数の上限の最小値は、ILP 問題を繰り返し解くことで得られる。提案手法を用いることで、状態遷移図の計算が困難な大規模システムに対しても、短い制御時間であれば制御入力の計算が可能となる。

解析においては、可到達性解析問題と安全性検証問題を考える。本論文では、確率時間分岐相論理^{7),24)}を用いて仕様を記述し、確率モデル検査ツール PRISM²¹⁾を用いて実装する手法を提案する。直感的な手法としては、状態遷移図を基に実装する手法が考えられるが、本論文ではブール関数を直接用いる手法を提案する。

表記: \mathcal{R} は実数の集合を表す。 $\{0, 1\}^n$ は要素が 0 および 1 からなる n 次元ベクトルを表す。 I_n および $0_{m \times n}$ は、それぞれ n 次元単位行列、 $m \times n$ の零行列を表す。とくにサイズを明示する必要がない場合には、 I_n を I 、 $0_{m \times n}$ を 0 と簡単に表記する。また、命題あるいは対応する 0-1 変数 a, b に対し、 $\neg a$ は a の否定、 $a \wedge b$ は a と b の論理積、 $a \vee b$ は a と b の論理和をそれぞれ表す。

^{†1} 北陸先端科学技術大学院大学 情報科学研究科
School of Information Science, Japan Advanced Institute of Science and Technology

2. 確率ブリアンネットワーク

まず、本論文で扱う確率ブリアンネットワーク (PBN) の概要を説明する。

次の PBN

$$x(k+1) = f(x(k)) \quad (1)$$

を考える。ここで $x \in \{0, 1\}^n$ は状態、 $k = 0, 1, 2, \dots$ は離散時刻、 $f: \{0, 1\}^n \rightarrow \{0, 1\}^n$ は適当なブール関数である。また、状態 x の第 i 要素を x_i 、ブール関数 f の第 i 要素のブール関数を $f^{(i)}$ と表記する。確定的なブリアンネットワークでは、ブール関数 f は一つだけ与えられることから、 $x(k)$ を定めることで $x(k+1)$ は確定的に定まる。一方、PBN においては、 $f^{(i)}$ の候補が複数個あり、各時刻で確率的に独立に選択される。ここで、 $f^{(i)}$ の候補を $f_j^{(i)}$ 、 $j = 1, 2, \dots, l(i)$ とし、 $f_j^{(i)}$ が選択される確率を

$$c_j^{(i)} = \text{Prob}(f^{(i)} = f_j^{(i)})$$

と表記する。このとき

$$\sum_{j=1}^{l(i)} c_j^{(i)} = 1 \quad (2)$$

が成立する。

【例題 1】 簡単な例として、3 状態の PBN を考える²²⁾。ブール関数 f は

$$f^{(1)} = \begin{cases} f_1^{(1)} = x_3(k), & c_1^{(1)} = 0.8, \\ f_2^{(1)} = \neg x_3(k), & c_2^{(1)} = 0.2, \end{cases}$$

$$f^{(2)} = f_1^{(2)} = x_1(k) \wedge \neg x_3(k), \quad c_1^{(2)} = 1.0,$$

$$f^{(3)} = \begin{cases} f_1^{(3)} = x_1(k) \wedge \neg x_2(k), & c_1^{(3)} = 0.7, \\ f_2^{(3)} = x_2(k), & c_2^{(3)} = 0.3, \end{cases}$$

として与えられているとする。 $l(1) = 2, l(2) = 1, l(3) = 2$ であり、(2) 式が成立することが確認できる。つぎに、状態の遷移について考える。 $x(0) = [0 \ 0 \ 0]^T$ とする。このとき

$$\text{Prob}(x(1) = [0 \ 0 \ 0]^T \mid x(0) = [0 \ 0 \ 0]^T) = 0.8,$$

$$\text{Prob}(x(1) = [1 \ 0 \ 0]^T \mid x(0) = [0 \ 0 \ 0]^T) = 0.2$$

を得る。この例の場合、状態の候補は $2^3 = 8$ 通りであり、それぞれの状態からの遷移を状

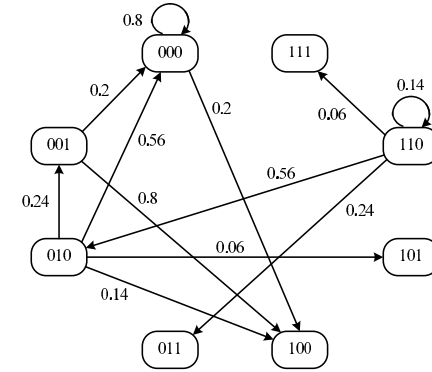


図 1 State transition diagram

態遷移図で表現すると、Fig. 1 を得る。Fig. 1 において、各ノードの 3 桁の数字は状態変数の要素 x_1, x_2, x_3 、辺の数字は遷移確率を表す。また、煩雑となることを避けるため、 $x(k) = [0 \ 0 \ 0]^T, [0 \ 0 \ 1]^T, [0 \ 1 \ 0]^T, [1 \ 1 \ 0]^T$ からの状態遷移のみを記した。

PBN の制御における既存手法では、状態遷移図を用いて制御入力を計算している。しかしながら、状態遷移図のノード数は 2^n (n は PBN の状態数) であることから、状態遷移図の計算は指数時間の計算量が必要であり、小規模の問題しか扱うことができない。そこで、本論文では状態遷移図を計算しない新しい方法を提案する。

3. 問題設定

本節では、いくつかの準備を行ったのち、考える問題を設定する。

3.1 制御入力をもつ確率ブリアンネットワーク

(1) 式に制御入力を付加した PBN

$$x(k+1) = f(x(k), u(k)) \quad (3)$$

を考える。ここで $u \in \{0, 1\}^m$ は制御入力、 $f: \{0, 1\}^n \times \{0, 1\}^m \rightarrow \{0, 1\}^n$ は適当なブール関数であり、各要素のブール関数は確率的に独立に選択されることとする。ブール関数を選択する確率などの記号は 2 節と同様とする。なお、遺伝子ネットワークの場合、制御入力は各時刻で任意に制御可能な遺伝子の発現量に対応する。

ブール関数の第 i 要素および時刻 k に対し、 $\bar{j}(i, k) \in \{1, 2, \dots, l(i)\}$ が与えられている

とする．このとき

$$\pi_{\bar{j}}(k) := \prod_{i=1}^n c_{\bar{j}(i,k)}^{(i)}$$

を定義する． $\pi_{\bar{j}}(k)$ はブール関数

$$\left[f_{\bar{j}(1,k)}^{(1)} \quad f_{\bar{j}(2,k)}^{(2)} \quad \cdots \quad f_{\bar{j}(n,k)}^{(n)} \right]^T$$

が時刻 k において選択される確率を意味する．さらに

$$\pi_{\bar{j}}(k_1, k_2) := \prod_{q=k_1}^{k_2} \pi_{\bar{j}}(q)$$

を定義する． $\pi_{\bar{j}}(k_1, k_2)$ は時間区間 $[k_1, k_2]$ において， $\bar{j}(i, k)$ により定まるブール関数の組み合わせが選択される確率を意味する．

[例題 2] 例題 1 の PBN を再び考える． $\bar{j}(1,0) = 1, \bar{j}(2,0) = 1, \bar{j}(3,0) = 2$ とすると， $\pi_{\bar{j}}(0) = c_1^{(1)} \cdot c_1^{(2)} \cdot c_2^{(3)} = 0.8 \cdot 1 \cdot 0.3 = 0.24$ を得る．さらに， $\bar{j}(1,1) = 2, \bar{j}(2,1) = 1, \bar{j}(3,1) = 1$ および $\bar{j}(1,2) = 1, \bar{j}(2,2) = 1, \bar{j}(3,2) = 1$ とすると， $\pi_{\bar{j}}(1) = 0.2 \cdot 1 \cdot 0.7 = 0.14$ および $\pi_{\bar{j}}(2) = 0.8 \cdot 1 \cdot 0.7 = 0.56$ が得られる．また $\pi_{\bar{j}}(0,2) = \pi_{\bar{j}}(0) \cdot \pi_{\bar{j}}(1) \cdot \pi_{\bar{j}}(2) = 0.24 \cdot 0.14 \cdot 0.56 = 0.018816$ が得られる．

3.2 制御問題

まず，最適制御問題を定式化する．PBN(3) に対し，次の問題を考える．

[問題 1] (最適制御問題) 制御入力をもつ PBN (3) が与えられているとする．また，初期状態 $x(0) = x_0$ ，要素が正の実数である重みベクトル $Q, Q_f \in \mathcal{R}^{1 \times n}$ ， $R \in \mathcal{R}^{1 \times m}$ ，スカラー ρ ($0 \leq \rho \leq 1$)，制御時間 N が与えられているとする．このとき，次の 2 つの問題の解を求めよ．

Problem A: 制約条件

$$\pi_{\bar{j}}(0, N-1) \geq \rho \quad (4)$$

を満足し，評価関数

$$J = \sum_{i=0}^{N-1} \{Qx(i) + Ru(i)\} + Q_f x(N) \quad (5)$$

の下限を最小化する制御入力 $u(0), u(1), \dots, u(N-1)$ を求めよ．

Problem B: Problem A において得られた制御入力を PBN (3) に印加した閉ループ系を考える．このとき，評価関数の上限の最小値 \bar{J}_s を求めよ．

以下では，Problem A において得られた評価関数の下限の最小値を \underline{J}_s と表記する．簡単のため，線形の評価関数を考えるが，二次形式の評価関数を考えることも可能である．また，オフセットベクトル $x_d \in \{0, 1\}^n$ を用いて， $x(i)$ を $\hat{x}(i) := x(i) - x_d$ に置き換えてもよい．この場合，評価関数の状態に関する項は絶対値を考える必要がある．

ここで，不等式制約 (4) の必要性を説明する．(4) 式がない場合，問題 1 は PBN を不確かなシステムとみなした上で，最良性能と最悪性能を導出していると考えられる．このとき，最良性能と最悪性能が達成される確率が極めて小さい場合が考えられ，結果として，評価が粗くなる可能性がある．この問題を回避するために，(4) 式を付加している．

3.3 解析問題

つぎに，解析問題を定式化する．本論文では，可到達性解析問題と安全性検証問題を考える．まず，PBN(3) に対し，出力 $y(k) = Cx(k) \in \{0, 1\}^p$ を考える．ただし， $y_i = x_j$ ， $i = 1, 2, \dots, p$ ， $j \in \mathcal{J} \subseteq \{1, 2, \dots, n\}$ が成立すると仮定する．本論文では，出力は観測量を意味しないことに注意されたい．このとき，次の問題を考える．

[問題 2] (可到達性解析問題) 制御入力をもつ PBN (3) および出力 $y(k) = Cx(k)$ が与えられているとする．また，初期状態 $x(0) = x_0$ ，制御時間 N ，目標出力 y_f が与えられているとする．このとき，時刻 $k = 0, 1, \dots, N$ において， $y(k) = y_f$ が 1 回以上成立する最大の確率 P_{\max} を求めよ．

標準的な可到達性解析では，終端時刻のみに着目し， $y(N) = y_f$ が成立するかどうかを議論する．本論文では，時刻 $k = 0, 1, \dots, N$ のどこかで， $y(k) = y_f$ が成立すればよいこととする．また，PBN は確率的な振る舞いをもつことから， $y(k) = y_f$ が成立するかどうかを判定するのではなく，成立する最大の確率を求めることとする．

さらに，可到達性解析問題とは逆の安全性検証問題を以下に与える．

[問題 3] (安全性検証問題) 制御入力をもつ PBN (3) および出力 $y(k) = Cx(k)$ が与えられているとする．また，初期状態 $x(0) = x_0$ ，制御時間 N ，危険な出力 y_u が与えられているとする．このとき，時刻 $k = 0, 1, \dots, N$ において， $y(k) = y_u$ が 1 回以上成立する確率が 0 であるかどうかを判定せよ．この確率が 0 であれば，PBN は安全であると呼ぶ．

4. 最適制御問題の解法

本節では，問題 1 の最適制御問題の解法を導出する．まず，いくつかの準備を行ったの

ち、Problem A と Problem B の解法をそれぞれ与える。

4.1 準備

整数計画法を用いるためには、ブール関数を実数体上の関数に変換する必要がある。このとき、次の補題¹⁸⁾を用いる。

[補題 1] 0-1 変数 δ_1, δ_2 を考える。このとき、次の関係が成立する。

- (i) $\neg\delta_1$ は $1 - \delta_1$ と等価である。
- (ii) $\delta_1 \vee \delta_2$ は $\delta_1 + \delta_2 - \delta_1\delta_2$ と等価である。
- (iii) $\delta_1 \wedge \delta_2$ は $\delta_1\delta_2$ と等価である。

補題 1 から、任意のブール関数は実数体上の多項式に等価的に変換可能であることが理解できる。例えば、ブール関数 $\delta_1 \vee \neg\delta_2$ は $\delta_1 + (1 - \delta_2) - \delta_1(1 - \delta_2) = 1 - \delta_2 + \delta_1\delta_2$ と等価である。

また、0-1 変数の積は次の補題⁵⁾を用いることで線形化が可能である。

[補題 2] 0-1 変数 $\delta_j \in \{0, 1\}$, $j \in \mathcal{J}$ が与えられているとする。ここで \mathcal{J} は適当な添字集合である。このとき、 $z = \prod_{j \in \mathcal{J}} \delta_j$ は 2 組の線形不等式

$$\sum_{j \in \mathcal{J}} \delta_j - z \leq |\mathcal{J}| - 1, \quad -\sum_{j \in \mathcal{J}} \delta_j + |\mathcal{J}|z \leq 0$$

と等価である。ここで $|\mathcal{J}|$ は添字集合 \mathcal{J} の要素数である。

補題 1 と補題 2 から、任意のブール関数は線形関数と線形不等式の組に等価的に変換可能であることが理解できる。詳細は^{5),18)}を参照されたい。

4.2 Problem A の解法

まず、問題 1 の Problem A の解法を考える。

補題 1 を用いると、(3) 式におけるブール関数の候補 $f_j^{(i)}(x(k), u(k))$ と等価な実数体上の多項式が得られる。得られた多項式を $\hat{f}_j^{(i)}(x(k), u(k))$ と表記する。このとき、 $\hat{f}_j^{(i)}(x(k), u(k))$ を用いた次のシステム

$$x_i(k+1) = \sum_{j=1}^{l(i)} \left\{ \delta_{i,j}(k) \hat{f}_j^{(i)}(x(k), u(k)) \right\} \quad (6)$$

を考える。ここで $i = 1, 2, \dots, n$ および $\delta_{i,j}(k) \in \{0, 1\}^1$ である。また、次の拘束条件

$$\sum_{j=1}^{l(i)} \delta_{i,j}(k) = 1 \quad (7)$$

を課すこととする。つぎに、 $c_j^{(i)}, \delta_{i,j}(k)$ を用いて

$$L_i := \begin{bmatrix} \ln c_1^{(i)} & \ln c_2^{(i)} & \cdots & \ln c_{l(i)}^{(i)} \end{bmatrix},$$

$$\delta_i(k) := \begin{bmatrix} \delta_{i,1} & \delta_{i,2} & \cdots & \delta_{i,l(i)} \end{bmatrix}^T \in \{0, 1\}^{l(i)}$$

を定義し、さらに

$$L := \begin{bmatrix} L_1 & L_2 & \cdots & L_n \end{bmatrix},$$

$$\delta(k) := \begin{bmatrix} \delta_1(k) & \delta_2(k) & \cdots & \delta_n(k) \end{bmatrix}^T \in \{0, 1\}^l$$

を定義する。ここで $l := \sum_{i=1}^n l(i)$ である。このとき、次の補題が成立する。

[補題 3] Problem A は次の問題と等価である。

Problem C:

$$\text{find } u(k), \delta(k), \quad k = 0, 1, \dots, N-1$$

$$\text{min Cost function (5)}$$

$$\text{subject to System (6), } x(0) = x_0,$$

$$\text{Inequality constraint } \sum_{k=0}^{N-1} L\delta(k) \geq \ln \rho,$$

$$\text{Equality constraint (7)}$$

(証明) 等式制約 (7) を満足する $\delta(k)$ を定めると、各時刻でのブール関数が一意に定まる。このとき、各時刻であるブール関数が選択される確率の自然対数は $\ln \pi_j(k) = L\delta(k)$ として得られる。さらに

$$\ln \pi_j(0, N-1) = \sum_{k=0}^{N-1} \ln \pi_j(k) = \sum_{k=0}^{N-1} L\delta(k)$$

が成立することから、Problem C の不等式制約は Problem A の不等式制約 (4) に対応する。したがって、Problem A と Problem C は等価である。

補題 3 では、不等式制約 (4) を線形不等式として表現するために、確率の自然対数を用いている。また、システム (6) は多項式型非線形システムであるが、補題 2 を用いることで、

次の線形状態方程式と線形不等式の組

$$\begin{cases} x(k+1) = Ax(k) + B_1u(k) + B_2\delta(k) + B_3z(k), \\ Cx(k) + D_1u(k) + D_2\delta(k) + D_3z(k) \leq E \end{cases} \quad (8)$$

で表現可能である．ここで $z(k) \in \{0, 1\}^p$ は補助変数であり， p はブール関数に依存して決定される．以下では表記の簡単化のため， $B := [B_1 \ B_2 \ B_3]$ ， $v(k) := [u^T(k) \ \delta^T(k) \ z^T(k)]^T$ とする．また， $x(0) \in \{0, 1\}^n$ ， $v(k) \in \{0, 1\}^{m+l+p}$ のもとで， $x(k)$ は自動的に 0-1 変数となることに注意されたい．したがって，以下では初期状態を除き $x(k) \in \mathcal{R}^n$ とする．

(8) 式を用いて，Problem C をさらに変換する．なお，Problem C の等式制約 (7) は，(8) 式の線形不等式に組み込まれていることとする．まず，(8) 式の状態方程式の一般解

$$x(k) = A^k x_0 + \sum_{i=1}^k A^{i-1} B u(k-i)$$

から

$$\bar{x} = \bar{A}x_0 + \bar{B}\bar{v} \quad (9)$$

を得る．ここで $\bar{x} := [x^T(0) \ x^T(1) \ \dots \ x^T(N)]^T$ ， $\bar{v} := [v^T(0) \ v^T(1) \ \dots \ v^T(N-1)]^T$ であり

$$\bar{A} = \begin{bmatrix} I \\ A \\ A^2 \\ \vdots \\ A^N \end{bmatrix}, \quad \bar{B} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ B & 0 & \dots & 0 \\ AB & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ A^{N-1}B & \dots & AB & B \end{bmatrix}$$

である．また，(8) 式の線形不等式から

$$\bar{C}\bar{x} + \bar{D}\bar{v} \leq \bar{E} \quad (10)$$

を得る．ここで

$$\begin{aligned} \bar{C} &= [\text{block-diag}(C, C, \dots, C) \ 0], \\ \bar{D} &= \text{block-diag}(D, D, \dots, D), \\ \bar{E} &= [E^T \ E^T \ \dots \ E^T]^T \end{aligned}$$

である．つぎに，Problem C の $\sum_{k=0}^{N-1} L\delta(k) \geq \ln \rho$ を考える． $\delta(k) = Wv(k)$ ， $W =$

$[0_{l \times m} \ I_l \ 0_{l \times p}]$ から， $\sum_{k=0}^{N-1} L\delta(k) \geq \ln \rho$ は

$$-\bar{L}\bar{W}\bar{v} \leq -\ln \rho \quad (11)$$

として変形できる．ここで $\bar{L} = [L \ \dots \ L]$ ， $\bar{W} = \text{block-diag}(W, \dots, W)$ である．また，評価関数 (5) は

$$J = \bar{Q}\bar{x} + \bar{R}\bar{v} \quad (12)$$

となる．ここで $\bar{Q} = [Q \ \dots \ Q \ Q_f]$ ， $\bar{R} = [R \ \dots \ R]$ である．(9) 式を (10) 式および (12) 式に代入すると，次の定理を得る．

《定理 1》 Problem C は次の整数線形計画 (ILP) 問題と等価である．

Problem D:

$$\begin{aligned} &\text{find } \bar{v} \in \{0, 1\}^{(m+l+p)N}, \\ &\text{min } (\bar{R} + \bar{Q}\bar{B})\bar{v} + \bar{Q}\bar{A}x_0, \\ &\text{subject to } \begin{bmatrix} \bar{C}\bar{B} + \bar{C} \\ -\bar{L}\bar{W} \end{bmatrix} \bar{v} \leq \begin{bmatrix} \bar{E} - \bar{C}\bar{A}x_0 \\ -\ln \rho \end{bmatrix} \end{aligned}$$

Problem D は ILOG CPLEX¹⁹⁾ などの適当なソルバーを用いることで解くことができる．

4.3 Problem B の解法

つぎに，問題 1 の Problem B の解法を考える．評価関数の上限の最小値 \bar{J}_s を導出するために，本論文では，次の二分法に基づくアルゴリズムを提案する．

Algorithm 1:

Step 1: $a = \underline{J}_s$ ，十分大きな正の定数 b ，十分小さな正の定数 ε を与える．

Step 2: $c = (a + b)/2$ とおく．

Step 3: 次の ILP 問題を解く．

Problem E:

$$\begin{aligned} &\text{given } u(0), u(1), \dots, u(N-1), \\ &\text{find } \bar{v} \in \{0, 1\}^{(m+l+p)N}, \\ &\text{min } (\bar{R} + \bar{Q}\bar{B})\bar{v} + \bar{Q}\bar{A}x_0, \\ &\text{subject to } \begin{bmatrix} \bar{C}\bar{B} + \bar{C} \\ -\bar{L}\bar{W} \\ -(\bar{R} + \bar{Q}\bar{B}) \end{bmatrix} \bar{v} \leq \begin{bmatrix} \bar{E} - \bar{C}\bar{A}x_0 \\ -\ln \rho \\ \bar{Q}\bar{A}x_0 - c \end{bmatrix} \end{aligned}$$

Step 4: Problem E が可解であれば, $a = c$ と更新する. 非可解であれば, $b = c$ と更新する.

Step 5: $|a - b| < \varepsilon$ が成立するならば, $\bar{J}_s = a$ として終了する. 成立しなければ, Step 2 へ.

$u(0), u(1), \dots, u(N-1)$ が与えられているので, Problem E における 0-1 変数の実際の次元は $(l+p)N$ である. また, Problem E の不等式制約 $-(\bar{R} + \bar{Q}\bar{B})\bar{v} \leq \bar{Q}\bar{A}x_0 - c$ は, (12) 式から $J \geq c$ を意味する. したがって, Algorithm 1 から, 評価関数の上限の最小値 \bar{J}_s を求めることができる. なお, 評価関数を二次形式で与えた場合, ILP 問題ではなく整数二次計画問題を解く必要がある. この問題は例えば ILOG CPLEX を用いることで解くことが可能である.

5. 解析問題の解法

つぎに, 問題 2 の可到達性解析問題および問題 3 の安全性検証問題の解法を説明する. 準備として時相論理の基礎的事項を説明したのち, モデル検査ツール PRISM⁽²¹⁾ による実装方法を説明する.

5.1 時相論理の基礎的事項

命題論理や述語論理では命題の真偽値は時不変であるが, 時相論理では命題の真偽値の時間的変化を扱う. 本論文では PBN が離散時間システムであるので, 時相論理でも離散時間の場合を考える. 分岐時間時相論理 (Computation Tree Logic, CTL)⁽²³⁾ を説明したのち, CTL を拡張した確率 CTL (Probabilistic CTL, PCTL)^{(7),(24)} を説明する.

CTL では, 命題論理で用いる論理演算子 ($\neg, \wedge, \vee, \rightarrow, \leftrightarrow$ など) の他に, 時相演算子および経路限量子を用いる. 時相演算子は, F (将来いつか), G (今後常に), X (次の時刻で), U (ある時間まで) から構成される. 経路限量子は, 「すべての実行可能な経路に対して」を表す A, および「ある実行可能な経路に対して」を表す E から構成される.

つぎに, CTL の論理式を以下の構成規則により定義する. あわせて状態式および経路式も定義する.

- (1) 命題変数, 命題定数は状態式である.
- (2) ϕ, ψ を状態式とすると, $\neg\phi, \phi \wedge \psi, \phi \vee \psi, \phi \rightarrow \psi, \phi \leftrightarrow \psi$ は状態式である.
- (3) ϕ を経路式とすると, $E\phi, A\phi$ は状態式である.
- (4) ϕ, ψ を状態式とすると, $X\phi, F\phi, G\phi, \phi U\psi$ は経路式である.

(5) 以上が状態式と経路式のすべてであり, 状態式が CTL の論理式のすべてである.

PCTL では, CTL の論理式に確率の概念を導入している. すなわち CTL の論理式 ϕ に対し, 論理式 $\mathcal{P}_{\geq p}(\phi), \exists \in \{\leq, <, \geq, >\}, p \in [0, 1]$ を付加する. たとえば, $\mathcal{P}_{\leq p}(\phi)$ は, 論理式 ϕ が確率 p 以下で成立するならば真, そうでなければ偽を意味する. また以下では, PCTL を可到達性解析問題, 安全性検証問題に利用するために, 時相演算子 F (将来いつか) を改良した時相演算子 $F^{\leq N}$ (時刻 N までのいつか) を用いることとする.

PCTL の論理式を用いると, 安全性検証問題は $\mathcal{P}_{\leq 0}(F^{\leq N}[y(k) = y_u])$ の真偽を確かめる問題といえかえることができる. ここで $[y(k) = y_u]$ は, 時刻 k において $y(k) = y_u$ が成立すれば真, そうでなければ偽となる論理式を表す. また, PCTL の論理式を用いることで, 可到達性解析問題, 安全性検証問題以外のさまざまな問題を記述することが可能である.

5.2 PRISM による実装

PBN および PCTL の論理式を, 確率的モデル検査ツール PRISM を用いて実装する方法を説明する. PRISM は, 対象として連続時間マルコフ連鎖, 離散時間マルコフ連鎖, マルコフ決定過程を考え, これらの対象に対して, PCTL 論理式で記述された仕様を満足するかどうかを判定するツールである.

まず, PRISM 上で PBN をどのようにモデリングするかを考える. 直感的な方法の一つは, Fig. 1 の状態遷移図すなわち離散時間マルコフ連鎖を直接, PRISM の言語で記述することである. しかしながら, この方法では状態遷移図を設計者側が計算する必要があり, 大規模なシステムでは困難である. そこで, ブール関数を直接用いる手法を考える. 例として, 制御入力を付加した PBN を考える. ブール関数 f は

$$f^{(1)} = \begin{cases} f_1^{(1)} = x_3(k) \wedge \neg u(k), & c_1^{(1)} = 0.8, \\ f_2^{(1)} = \neg x_3(k) \wedge \neg u(k), & c_2^{(1)} = 0.2, \end{cases}$$

$$f^{(2)} = f_1^{(2)} = x_1(k) \wedge \neg x_3(k), \quad c_1^{(2)} = 1.0,$$

$$f^{(3)} = \begin{cases} f_1^{(3)} = x_1(k) \wedge \neg x_2(k), & c_1^{(3)} = 0.7, \\ f_2^{(3)} = x_2(k) \vee u(k), & c_2^{(3)} = 0.3, \end{cases}$$

として与えられているとする. ここで $u(k) \in \{0, 1\}^1$ である. 補題 1 を適用すると各ブール関数は実数体上の関数に変換できる. このとき, この PBN を記述した PRISM のソースコードを以下に示す.

```

01: mdp
02:   const int c=0;
03:   module PBN1
04:     x1:[0..1] init 0;
05:     [PBN1] c=0 -> 0.8:(x1'=x3*(1-u)) + 0.2:(x1'=(1-x3)*(1-u));
06:   endmodule
07:   module PBN2
08:     x2:[0..1] init 0;
09:     [PBN1] c=0 -> 1.0:(x2'=x1*(1-x3));
10:   endmodule
11:   module PBN3
12:     x3:[0..1] init 0;
13:     [PBN1] c=0 -> 0.7:(x3'=x1*(1-x2)) + 0.3:(x3'=x2+u-x2*u);
14:   endmodule
15:   module input
16:     u:[0..1] init 0;
17:     [PBN1] u = 0 -> (u' = 0);
18:     [PBN1] u = 0 -> (u' = 1);
19:     [PBN1] u = 1 -> (u' = 0);
20:     [PBN1] u = 1 -> (u' = 1);
21:   endmodule

```

1 行目では、システムがマルコフ決定過程であること、すなわち決定すべき動作（入力）が含まれていることを宣言している。2 行目では、定数 $c = 0$ を定義している。3~6 行目では、 $f^{(1)}$ をモデリングしている。4 行目では、 x_1 が 0-1 変数であり、初期値が 0 であることを記述している。5 行目では、 $c = 0$ であれば、次の時刻の状態 x_1 が、確率 0.8 で $x_3(1-u)$ 、確率 0.2 で $(1-x_3)(1-u)$ に遷移することを記述している。なお、 $c = 0$ は定数であることから、この遷移は各時刻で必ず生起することに注意されたい。7~10 行目および 11~14 行目も同様に、状態 x_2 および x_3 の遷移を記述している。PBN では、ブール関数の要素毎に離散確率分布を与えていることから、モジュールを個別に記述する必要があることに注意されたい。5, 9, 13 行目の [PBN1] はモジュール PBN2, PBN3 は PBN1 と同期していることを意味する。最後に、16~20 行目では、制御入力を非決定性の変数とし

て記述している。PRISM では、制御入力の初期値を指定する必要があることに注意されたい（16 行目）。したがって、制御入力が任意の 0-1 変数である場合、初期値だけはソースコードを設計者側で修正する必要がある。

以上が PRISM を用いた PBN のモデリング方法の例である。一般の PBN に対しても、同様の手順でモデリングができることは明らかである。

つぎに、PCTL の論理式の、PRISM 上での表現方法を説明する。例として、安全性検証問題を記述する PCTL 論理式 $P_{\leq 0}(F^{\leq N}[y(k) = y_u])$ を考える。 $y = [y_1 \ y_2]^T$ 、 $y_u = [0 \ 1]^T$ とすると、PRISM 上では $P_{\leq 0}[F^{\leq N}(y_1=0) \& (y_2=1)]$ と記述すればよい。また、 $P_{\min} = ?[F^{\leq N}(y_1=0) \& (y_2=1)]$ （時刻 N までに $y_1 = 0$ および $y_2 = 1$ が 1 回以上成立する最小の確率を求めよ、という意味）と記述し、 P_{\min} が 0 かどうかを判定してもよい。

可到達性解析問題では、条件を満足する最大の確率を求めるため、PCTL の論理式で直接記述することはできない。二分法を用いて PCTL 論理式の真偽を繰り返し確認することで、 P_{\max} を導出することがある。しかし、PRISM では $P_{\max} = ?[F^{\leq N}(y_1=0) \& (y_2=1)]$ （時刻 N までに $y_1 = 0$ および $y_2 = 1$ が 1 回以上成立する最大の確率を求めよ、という意味）と記述すればよい。

以上が、PBN のモデリングおよび解析に対する PRISM を用いた実装方法である。PRISM を用いた実装方法は他にも考えられることから、上記の手法はその中の一つであると解釈できる。より良い方法が存在するかもしれないが、状態遷移図を用いずブール関数を直接用いてモデリングおよび解析を行う、という本論文の目的は達成していることを強調しておく。

6. おわりに

本論文では、確率ブーリアンネットワークの解析および制御のための新しい計算手法を提案した。最適制御問題では、評価関数の上下限により制御性能を評価することとし、問題を整数線形計画 (ILP) 問題に帰着させた。可到達性解析問題および安全性検証問題では、確率時間分岐相論理 (PCTL) により仕様を記述した上で、確率モデル検査ツール PRISM を用いて実装する手法を提案した。既存の計算手法では、 n 状態の PBN に対して 2^n 個のノードからなる状態遷移図を計算する必要があった。提案した手法では、少なくとも設計者が状態遷移図を計算する必要がなく、簡便な手法であるといえる。なお、紙面の都合上、数値例は省略する。

最後に、今後の課題について述べる。最適制御問題すなわち ILP 問題の計算時間が長く

なる場合があることから，計算時間の低減は今後の重要な課題である．このとき，SAT（充足可能性問題）ソルバー（例えば zChaff⁽²⁰⁾）を用いることが一つの解決策であり，今後検討していく．解析問題においては，PCTL では可到達性解析問題および安全性検証問題以外の仕様も記述可能であり，システム生物学の視点に基づくより実用的な問題設定を考える必要がある．さらに，実際の遺伝子ネットワークのモデルに適用することも重要である．

本研究は文部科学省科学研究費補助金（若手研究（B）No. 20760278，基盤研究（C）No. 21500009）による支援を受けて行われています．記して感謝いたします．

参 考 文 献

- 1) T. Akutsu, M. Hayashida, W.-K. Ching, and M. K. Ng, Control of Boolean networks: Hardness results and algorithms for tree structured networks, *Journal of Theoretical Biology*, vol. 244, pp. 670–679, 2007.
- 2) M. Aldana, Boolean dynamics of networks with scale-free topology, *Physica D*, vol. 185, pp. 45–66, 2003.
- 3) S. Azuma, E. Yanagisawa, and J. Imura, Controllability Analysis of Biosystems Based on Piecewise Affine Systems Approach, *IEEE Trans. on Automatic Control*, vol. 53, no. 1, pp. 139–152, 2008.
- 4) C. Belta, J. Schug, T. Dang, V. Kumar, G. J. Pappas, H. Rubin, and P. Dunlap, Stability and reachability analysis of a hybrid model of luminescence in the marine bacterium *Vibrio fischeri*, *Proc. 40th IEEE Conf. on Decision and Control*, pp. 869–874, 2001.
- 5) T. M. Cavalier, P. M. Pardalos, and A. L. Soyster, Modeling and integer programming techniques applied to propositional calculus, *Computer & Operations Research*, vol. 17, no. 6, pp. 561–570, 1990.
- 6) W.-K. Ching, S.-Q. Zhang, Y. Jiao, T. Akutsu, N.-K. Tsing, and A.S. Wong, Optimal control policy for probabilistic Boolean networks with hard constraints, *IET Systems Biology*, vol. 3, no. 2, pp. 90–99, 2009.
- 7) F. Ciesinski and M. Groesser, On Probabilistic Computation Tree Logic, *Validation of Stochastic Systems, LNCS 2925*, pp. 147–188, Springer, 2004.
- 8) A. Datta, A. Choudhary, M. L. Bittner, and E. R. Dougherty, External Control in Markovian Genetic Regulatory Networks, *Machine Learning*, vol. 52, pp. 169–191, 2003.
- 9) A. Datta, A. Choudhary, M. L. Bittner, and E. R. Dougherty, External control in Markovian genetic regulatory networks: the imperfect information case, *Bioinformatics*, vol. 20, pp. 924–930, 2004.
- 10) A. Fauré, A. Naldi, C. Chaouiya, and D. Thieffry, Dynamical analysis of a generic

- Boolean model for the control of the mammalian cell cycle, *Bioinformatics*, vol. 22, pp. 124–131, 2006.
- 11) S. A. Kauffman, Metabolic stability and epigenesis in randomly constructed genetic nets, *Journal of Theoretical Biology*, vol. 22, pp. 437–467, 1969.
 - 12) K. Kobayashi, J. Imura, and K. Hiraishi, Polynomial-Time Controllability Analysis of Boolean Networks, *Proc. 2009 American Control Conf.*, pp. 1694–1699, 2009.
 - 13) C. J. Langmead and S. K. Jha, Symbolic Approaches to Finding Control Strategies in Boolean Networks, *Journal of Bioinformatics and Computational Biology*, vol. 7, no. 2, pp. 323–338, 2009.
 - 14) S. Martin, Z. Zhang, A. Martino, and J.-L. Faulon, Boolean dynamics of genetic regulatory networks inferred from microarray time series data, *Bioinformatics*, vol. 23, pp. 866–874, 2007.
 - 15) R. Pal, A. Datta, M. L. Bittner, and E. R. Dougherty, Optimal Infinite-Horizon Control for Probabilistic Boolean Networks, *IEEE Trans. on Signal Processing*, vol. 54, pp. 2375–2387, 2006.
 - 16) I. Shmulevich, E. R. Dougherty, S. Kim, and W. Zhang, Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks, *Bioinformatics*, vol. 18, pp. 261–274, 2002.
 - 17) I. Shmulevich and W. Zhang, Binary analysis and optimization-based normalization of gene expression data, *Bioinformatics*, vol. 18, pp. 555–565, 2002.
 - 18) H. P. Williams, *Model building in mathematical programming*, 4th ed., John Wiley & Sons, Ltd, 1999.
 - 19) <http://www.ilog.com/products/cplex/>
 - 20) <http://www.princeton.edu/~chaff/zchaff.html>
 - 21) <http://www.prismmodelchecker.org/>
 - 22) 阿久津：遺伝子ネットワークの離散モデルと制御，計測と制御，vol. 47, no. 8, pp. 664–669, 2008.
 - 23) 田中監修，磯部ら著：ソフトウェア科学基礎，近代科学社，2009.
 - 24) 萩谷，山本著：化学系・生物系の計算モデル，共立出版，2009.