ベイズ学習に基づく確率論理システムの最適化

豊田 充*・申 鉄龍*

Bayesian Learning Based Optimization for Stochastic Logical System

Mitsuru TOYODA* and Tielong SHEN*

The optimization problem of stochastic logical systems is studied in this paper. To deal with a system without knowledge of the objective function, a Bayesian optimization framework is extended with the learning algorithm called Gaussian process. Firstly, the regret bound, which represents the difference between the true optimal value and the achieved objective function value, is evaluated with exploiting the statistic features of Gaussian process. A numerical example is illustrated for the purpose of validation on the optimization algorithm afterward.

Key Words: stochastic system, logical system, learning control, Bayesian optimization, Gaussian process regression

1. はじめに

動的システム論において、ダイナミクスの表現は微分方程式あるいは差分方程式で与えられることが一般的であるが、論理システム理論¹⁾においては、システムの状態を論理変数として表現し、その時間と伴う遷移特性は遷移確率行列を用いて表わされる。一般的に、論理状態変数は有限個の離散状態からなる論理状態空間上で遷移し、その遷移確率は同じく論理変数の制御入力によって変化する。近年、このような論理システムの解析と設計手法に関する研究に注目が集まっている^{2)~4)}。また、最適制御手法に関する研究では、論理変数の状態と入力の関数を目標関数とし、状態の遷移確率の拘束の下、その期待値を最小にする論理制御手法に関する研究が報告されているが^{5)~7)}、文献8)によって示されたように、最適化のための繰り返し計算アルゴリズムを行列の同次操作によって表現可能なため、最適制御入力を求める計算の効率化を図ることが可能である。実際にこのアルゴリズムをガソリンエンジン制御に応用した研究⁷⁾では、RGF (Residual Gas Fraction, 残留ガス割合) と呼ばれる連続値物理量を離散化することで離散状態を定義し、実験データの統計解析によって状態遷移確率行列を同定した後、提案した論理システム最適化設計手法の実応用の可能性が示され、論理制御の枠組で連続域における動的システムの制御の有効性が示されている。

また、従来の動的システムの最適化の手法⁹⁾は、状態変数と

制御入力に依存するステージコストの値が既知であることが要求されるため、システムのモデリングと最適化のステップは分離しており、実プラントへの応用ではモデルベースの制御系設計に先立ってあらかじめ詳細な物理モデリングあるいは応答データによる同定を行なう必要がある^{10),11)}。この点に対して、ダイナミクスの拘束の下で定常状態における未知静的関数値の極値を学習しながら最適化する、極値探索法^{12)~14)}は効果的なアプローチであるが、対象となる未知関数の局所的勾配情報を用いたアルゴリズムであるため、局所解に陥る可能性がある。また、定常状態でダイナミクスが収束する平衡点での目的関数値を最適化するための手法であり、過渡状態すなわち状態変数が定常状態に至るまでのダイナミクスの挙動は目的関数に関与しておらず最適性は考慮されない。目的関数をパラメトリックに学習するという点からも、学習モデルが再現できる目的関数の範囲が狭められる。

一方、機械学習分野での最適化の手法として、静的関数の最適化のためのベイズ最適化の枠組みで、regret¹⁵⁾と呼ばれる、最適解とアルゴリズムの解による目的関数の差を考察した研究¹⁶⁾では、regret のオーダの評価を行っており、このアルゴリズムの応用研究として、太陽電池の出力最適化問題への活用例が報告されている¹⁷⁾。本研究ではこのアルゴリズムを論理システムの期待値型目的関数に拡張することで、1 ステップ予測制御アルゴリズムを構成する。また、前述のregret の評価方法を用いることで提案アルゴリズムの精度を評価する。

本論文の構成は以下のとおりである：2章では、論理システムに対して1 ステップ予測の形で目的関数を設定し、最適制御問題を定式化する。設定した問題に対して3章ではベイズ最適化アルゴリズムを構成し、アルゴリズムの最適化の精度の評価を行なう。最後に4章にて数値例を用いてアルゴリ

* 上智大学理工学部機能創造理工学科
東京都千代田区紀尾井町 7-1

* Department of Engineering and Applied Sciences, Sophia University, 7-1 Kioicho, Chiyoda-ku, Tokyo
(Received April 19, 2017)
(Revised June 21, 2017)

ズムの有用性を検証を行なう.

2. 準備

論理システムにおいて, そのダイナミクスは, t 時刻での状態 $x(t) = x^j$ と制御入力 $u(t) = u^l$ が与えられたときに $t+1$ 時刻で x^i 状態に遷移する条件付き確率, すなわち状態遷移確率

$$P_{ij}(u^l) = \mathcal{P}\{x(t+1) = x^i | x(t) = x^j, u(t) = u^l\} \quad (1)$$

によって表現される. 式中 $\mathcal{P}$ は確率を表わし, 論理状態空間は,

$$\mathcal{S} = \{x^1, x^2, \dots, x^{n_s}\}, \quad (2)$$

制御入力 u の集合も同様にして,

$$\mathcal{U} = \{u^1, u^2, \dots, u^{n_u}\} \quad (3)$$

と表わされる. 定義から遷移確率は $\sum_{i=1}^{n_s} P_{ij}(u^l) = 1, \forall l$ を満たす点に注意されたい. 以降では表式の便宜上, 入力 $u(t)$ に依存する遷移行列 P の縦ベクトルを簡略に

$$p_j(u^l) = \begin{bmatrix} P_{1j}(u^l) & P_{2j}(u^l) & \dots & P_{n_s j}(u^l) \end{bmatrix}^T \quad (4)$$

として表わす.

連続な状態変数 $x = [x_1, x_2, \dots, x_{n_x}]^T \in \mathbb{R}^{n_x}$ を以下のようにならび離散化し論理状態空間を定義する⁷⁾.

$$\mathcal{S} = \{x_1^{(1)}, x_1^{(2)}, \dots, x_1^{(s_1)}\} \times \dots \times \{x_n^{(1)}, x_n^{(2)}, \dots, x_n^{(s_n)}\} \quad (5)$$

ここで, すべての離散状態の数は $n_s = s_1 \times \dots \times s_n$ であるがこれを新たに, $x^1, x^2, \dots, x^{n_s}$ と置き換えた状態

$$\begin{cases} x^1 = [x_1^{(1)}, x_2^{(1)}, \dots, x_n^{(1)}]^T \\ x^2 = [x_1^{(1)}, x_2^{(1)}, \dots, x_n^{(2)}]^T \\ \vdots \\ x^{n_s} = [x_1^{(s_1)}, x_2^{(s_2)}, \dots, x_n^{(s_n)}]^T \end{cases} \quad (6)$$

と同一視する. 離散状態 x^i とその実数値 $[x_1^{(\cdot)}, x_2^{(\cdot)}, \dots, x_{n_x}^{(\cdot)}] \in \mathbb{R}^{n_x}$ は一対一に対応する点に注意されたい. これらの離散論理状態の時間的遷移を表わした図が **Fig. 1** である.

本稿では確率過程 x, u の t 離散時刻での値をそれぞれ $x(t), u(t)$ と括弧付きで記述するとし, 最小化問題

$$\min_{x \in \mathcal{S}} f(x) \quad (7)$$

を考える. 上式中, 未知の目的関数 $f(x)$ は以下によって特徴づけられるガウシアンプロセスからのサンプルであるとする.

(1) 各 x 上で平均

$$m(x) = \mathbb{E}[f(x)] = 0, \forall x \quad (8)$$

Fig. 1 Transition of logical state

である.

(2) 共分散関数

$$k(x, x') = \mathbb{E}[(f(x) - m(x))(f(x') - m(x')))] \quad (9)$$

で表現され, 各 x で有限な分散 $k(x, x) \leq 1$ をもつとする. 目的関数 $f(x)$ の各離散論理状態での値が未知であるため, t 時刻で観測

$$y(t) = f(x(t)) + \varepsilon(t) \quad (10)$$

から目的関数 $f(x)$ の学習を行ないながらの最適化を試みる. 式中 $\varepsilon(t)$ は $N(0, \sigma_\varepsilon^2)$ に従うガウシアンホワイトノイズであり, 観測外乱やモデル化誤差を表わす.

さらに regret, つまり設計した最適化アルゴリズムによって与えられる制御入力 $x_{\text{algorithm}}(t)$ と目的関数値が既知の場合の (7) の最適解 x^* との誤差を以下のようにして定義する:

$$r(t) = f(x_{\text{algorithm}}(t)) - f(x^*). \quad (11)$$

$$R(t) = \sum_{s=1}^t r(s). \quad (12)$$

このような探索学習と最適化手法として文献 16) の成果を以下のようにまとめることができる.

[アルゴリズム 2.1]¹⁶⁾

Step 1 t 時刻の状態 $x(t) = x^j$ を観測し, システム出力 $y(t) = f(x(t)) + \varepsilon(t)$ からガウシアンプロセスモデルを更新し, $\mu(t, x)$ と $v(t, x, x')$ を得る. ガウシアンプロセス学習は以下の手順で与えられる:

$$\begin{cases} \mu(t, x) = \kappa^T(t, x) (K(t) + I_{t \times t} \sigma_\varepsilon^2)^{-1} Y(t), \\ v(t, x, x') = k(x, x') \\ \quad - \kappa^T(t, x) (K(t) + I_{t \times t} \sigma_\varepsilon^2)^{-1} \kappa(t, x'), \end{cases} \quad (13)$$

上式の中のベクトル $\kappa(t, x)$ および行列 $K(t)$ は計測データとカーネル関数から

$$\kappa(t, x) = [k(x, x(1)), \dots, k(x, x(t))]^T, \quad (14)$$

$$Y(t) = [y(1), \dots, y(t)]^T \quad (15)$$

$$K(t) = \begin{bmatrix} k(x(1), x(1)) & \cdots & k(x(1), x(t)) \\ \vdots & \ddots & \vdots \\ k(x(t), x(1)) & \cdots & k(x(t), x(t)) \end{bmatrix} \quad (16)$$

として計算される．特に各点 $x^j \in \mathcal{S}$ での分散は， $\sigma^2(t, x^j) = v(t, x^j, x^j)$ として計算される．

Step 2 以下の目的関数に基づいて $x(t+1)$ を決定する：

$$x(t+1) = \arg \min_{x(t+1) \in \mathcal{S}} \left[\mu(t, x(t+1)) - \sqrt{\beta(t+1)\sigma(t, x(t+1))} \right], \quad (17)$$

式中， $\beta(t)$ は探索と最適化の重みを決める係数であり， $\sum_{t=1}^{\infty} \frac{1}{\varphi(t)} = 1, \varphi(t) > 0$ を満たす適当な数列 $\varphi(t)$ および正数 $0 < \delta < 1$ 用いて

$$\beta(t) = 2 \log(n_s \varphi(t) / \delta) \quad (18)$$

と決定する．

Step 3 $t := t+1$ として Step1 に戻り繰り返し．

ガウシアンプロセスによる学習に関して，入力系列 $\mathcal{X}(t) = \{x(1), \dots, x(t)\}$ と観測データ系列 $Y(t) = [y(1), \dots, y(t)]^T$ ，および各離散論理状態 $x^j \in \mathcal{S}$ 上での目的関数値真値をまとめたベクトルを

$$F = [f(x^1), f(x^2), \dots, f(x^{n_s})]^T \quad (19)$$

と定義する．このとき，ガウシアンプロセスに対応する情報ゲイン¹⁸⁾は

$$I(Y(t); F) = \frac{1}{2} \sum_{s=1}^t \log(1 + \sigma_\varepsilon^{-2} \sigma^2(s, x(s))) \quad (20)$$

として計算される．さらに $I(t)$ の最大値を

$$\gamma(t) = \max_{\mathcal{X}(t)} I(Y(t); F), \quad (21)$$

と定義したときに，以下の結果が成り立つ．

《定理 2.1》 ガウシアンプロセスがパラメータ l によって与えられる 2 乗指数カーネル

$$k(x, x') = \exp\left(\frac{1}{2l^2} \|x - x'\|^2\right) \quad (22)$$

により生成される場合，状態変数 x の次元 n_x を用いて

$$\gamma(t) = \mathcal{O}((\log t)^{n_x+1}) \quad (23)$$

が成り立つ．

《定理 2.2》 $\delta \in (0, 1)$ ， $\beta(t) = 2 \log(n_x t^2 \pi^2 / 6\delta)$ とする．このときに，アルゴリズムは以下の確率変数 $R(t)$ に関する確

率不等式を満たす．

$$\mathcal{P}\left\{R(t) \leq \sqrt{C_1 t \beta(t) \gamma(t)}\right\} \geq 1 - \delta, \quad (24)$$

式中の定数は， $C_1 = 8 / \log(1 + \sigma_\varepsilon^{-2})$ で与えられる．

3. 主な結果

本研究ではバイズ学習をモデル予測制御問題への応用を目的として， t 時刻での観測 $x(t) = x^j$ が得られたときに，1 ステップ予測した期待値型の目的関数

$$\begin{aligned} \min_{u(t) \in \mathcal{U}} J(u(t)), \\ J(u(t)) &= \mathbb{E}_{x(t+1)} \{f(x(t+1)) | x(t) = x^j, u(t)\} \\ &= \sum_{i=1}^{n_s} P_{ij}(u(t)) f(x^i) \\ &= p_j^T(u(t)) F \end{aligned} \quad (25)$$

を定義する．この 1 ステップ予測区間，つまりシステムの 1 ステップの過渡状態における期待値として目的関数を設定することで，各ステップごとでシステムが状態遷移する過渡状態の最適化を図る．

(19) 式で表わされた各離散状態に対応する目的関数のベクトル F が既知である場合は前述の最適化問題の解を厳密に求めることができるが，現実の問題設定で各離散論理状態における目的関数の値が制御系設計の段階で未知であることも多い．その場合，直接観測できない目的関数値を得られた観測データから推定しつつ，最小化する制御入力 $u(t)$ を求めることが要求される．また，そのアルゴリズムによって得られる目的関数の値と真の最適解との誤差を検証することが本研究の目的となる．

本問題設定では (5) 式で定義された，連続値状態変数を離散化した論理状態空間を対象とするため，各論理状態に対応する実数値を用いたガウシアンプロセスによる目的関数の学習や，それによるアルゴリズムの解析が可能となる．

本稿で提案する最適化・探索アルゴリズムは以下の手順にて計算される．

[アルゴリズム 3.1]

Step 1 t 時刻の状態 $x(t) = x^j$ を観測し，システム出力 $y(t)$ から目的関数 f を学習更新．ガウシアンプロセス学習によって新たに $\mu(t, x)$ と $v(t, x, x')$ を得る．状態 $\mathcal{S} = \{x^1, \dots, x^{n_s}\}$ 上の目的関数値のベクトル $M(t)$ と共分散行列 $V(t)$ を推定する． x がベクトルである場合には，

$$\begin{aligned} y(t) &= f(x(t)) + \varepsilon(t) \\ &= f(x_1(t), x_2(t), \dots, x_{n_x}(t)) + \varepsilon(t) \end{aligned} \quad (26)$$

なる多変数の関数として学習する．ここで， t 時刻までの入力 x と観測 y のデータを

$$\begin{cases} Y(t) = [y(1), \dots, y(t)]^T \\ \mathcal{X}(t) = \{x(1), \dots, x(t)\} \end{cases} \quad (27)$$

とおけば、ガウシアンプロセス学習はベイズの定理¹⁹⁾によって平均 $\mu(t, x)$ と分散 $v(t, x, x')$ の以下の更新式で与えられる:

$$\begin{cases} \mu(t, x) = \kappa^T(t, x) (K(t) + I_{t \times t} \sigma_\varepsilon^2)^{-1} Y(t), \\ v(t, x, x') = k(x, x') \\ \quad - \kappa^T(t, x) (K(t) + I_{t \times t} \sigma_\varepsilon^2)^{-1} \kappa(t, x'), \end{cases} \quad (28)$$

そして、以後の計算の簡便化のために、各離散点での平均と分散の推定を行列化して以下のように定義する.

$$M(t) = [\mu(t, x^1), \dots, \mu(t, x^{n_s})]^T, \quad (29)$$

$$V(t) = \begin{bmatrix} v(t, x^1, x^1) & \cdots & v(t, x^1, x^{n_s}) \\ \vdots & \ddots & \vdots \\ v(t, x^{n_s}, x^1) & \cdots & v(t, x^{n_s}, x^{n_s}) \end{bmatrix}. \quad (30)$$

Step 2 以下の目的関数に基づいて $u(t)$ を決定する:

$$\begin{aligned} u(t) &= \arg \min_{u \in \mathcal{U}} \left[\mathbb{E}_{x(t+1)} \{ \mu(t, x(t+1)) | x(t) = x^j \} \right. \\ &\quad \left. - \sqrt{\beta(t)} \sqrt{p_j^T(u) V(t) p_j(u)} \right] \\ &= \arg \min_{u \in \mathcal{U}} \left[p_j^T(u) M(t) \right. \\ &\quad \left. - \sqrt{\beta(t)} \sqrt{p_j^T(u) V(t) p_j(u)} \right], \end{aligned} \quad (31)$$

式中、 $\beta(t)$ は探索と最適化の重みを決める係数であり、 $\sum_{t=1}^{\infty} \frac{1}{\varphi(t)} = 1$, $\varphi(t) > 0$ を満たす適当な数列 $\varphi(t)$ および正数 $0 < \delta < 1$ 用いて

$$\beta(t) = 2 \log(n_s n_u \varphi(t) / \delta) \quad (32)$$

と決定する.

Step 3 $t := t + 1$ として Step1 に戻り繰り返す.

以下では学習の進展と目的関数の値の時間的な変化について考察する. 学習の進展に併せて学習による推定値が関数の取る真値に収束するという結果は必ずしも容易に得られないため、ここでは regret 問題設定に従い、最適化問題の regret, つまり制御による目的関数値と最適目的関数値の差を比較することで、最適性を議論することとする.

[補題 3.1] 各要素がガウシアンに従うベクトル確率変数 $x = [x_1, x_2, \dots, x_n]^T$ の平均と共分散行列がそれぞれ

$$M = \mathbb{E}\{x\} = \begin{bmatrix} \mathbb{E}\{x_1\} \\ \vdots \\ \mathbb{E}\{x_n\} \end{bmatrix} = \begin{bmatrix} m_1 \\ \vdots \\ m_n \end{bmatrix}, \quad (33)$$

$$V = \begin{bmatrix} v_{11} & \cdots & v_{1n} \\ \vdots & \ddots & \vdots \\ v_{n1} & \cdots & v_{nn} \end{bmatrix} \quad (34)$$

$$= \begin{bmatrix} \text{cov}(x_1, x_1) & \cdots & \text{cov}(x_1, x_n) \\ \vdots & \ddots & \vdots \\ \text{cov}(x_n, x_1) & \cdots & \text{cov}(x_n, x_n) \end{bmatrix},$$

$$\text{cov}(x_i, x_j) = \mathbb{E}\{(x_i - m_i)(x_j - m_j)\} \quad (35)$$

で与えられるとする. このとき、係数ベクトル $\alpha \in \mathbb{R}^n$ によるベクトル確率変数 x の各要素の重み付き和

$$z = \alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_n x_n = \alpha^T x \quad (36)$$

の平均 m_z と分散 σ_z^2 は以下で与えられる.

$$m_z = \alpha^T M, \quad \sigma_z^2 = \alpha^T V \alpha. \quad (37)$$

(証明) まず $n = 2$, $\alpha = [1, 1]^T$ の場合については、 x_1 と x_2 の同時確率分布 $p(x_1, x_2)$ を求め、 $p(z) = \int p(x_1, z - x_1) dx_1$ として畳み込み積分を計算すれば求まることが知られている. この事実とガウシアンスケーリングの性質、つまり与えられた実数 a, b について、 $z \sim N(0, 1)$ ならばスケーリング $z' = az + b$ は $z' \sim (b, a^2)$ に従うことを逐次的に用いると示せる. ■

以下では、上記の目的関数の時間変化を考察する. 基本的な方針は¹⁶⁾にあるように、ガウシアンに従う確率分布をチェビシェフの不等式²⁰⁾を用いて求める. つまり、 $r \sim N(0, 1)$ のとき、 $\mathcal{P}\{r > c\} = \frac{1}{2}e^{-c^2/2}$ である、という主張を利用して、ガウシアンプロセスによる推定値から真値が大きく外れてしまう確率を評価する.

[補題 3.2] $\sum_{t=1}^{\infty} \frac{1}{\varphi(t)} = 1$, $\varphi(t) > 0$ を満たす数列 $\varphi(t)$ を用いて $\beta(t) = 2 \log(n_s n_u \varphi(t) / \delta)$ とするとき、以下の不等式

$$\begin{aligned} & \left| \mathbb{E}_{x(t+1)} [f(x(t+1)) | x(t)] \right. \\ & \quad \left. - \mathbb{E}_{x(t+1)} [\mu(t, x(t+1)) | x(t)] \right| \\ & \leq \sqrt{\beta(t)} \sqrt{p_j^T(u(t)) V(t) p_j(u(t))}, \end{aligned} \quad (38)$$

$$\forall x(t) \in \mathcal{S}, \forall t \in \mathbb{Z}_{>0}$$

が成り立つ確率は $1 - \delta$ 以上である.

(証明) ガウシアンプロセスによる推定の性質²¹⁾から各 x^j に対して、

$$f(x^j) \sim N(\mu(t, x^j), \sigma^2(t, x^j)), \quad (39)$$

$$\text{cov}(f(x^i), f(x^j)) = v(t, x^i, x^j) \quad (40)$$

が成り立つ. 期待値型目的関数が

$$\mathbb{E}_{x(t+1)} \{ \mu(t, x(t+1)) | x(t) = x^j \} = p_j^T(u^l) M(t), \quad (41)$$

$$\mathbb{E}_{x(t+1)} \{ \mu(t, x(t+1)) | x(t) = x^j \} = p_j^T(u^l) F \quad (42)$$

と各離散論理状態 x における目的関数値の重み付き和で表わされることに注意して、補題 3.1 より、ある時刻 t で固定された $x(t) = x^j$, $u(t) = u^l$ に対して、

$$\frac{p_j^T(u^l)(M(t) - F)}{\sqrt{p_j^T(u^l)V(t)p_j(u^l)}} \sim N(0, 1) \quad (43)$$

となる。

ここで、事象 $A_{t,j,l}$ を

$$A_{t,j,l} = \left\{ \frac{|p_j^T(u^l)(M(t) - F)|}{\sqrt{p_j^T(u^l)V(t)p_j(u^l)}} \geq \sqrt{\beta(t)} \right\} \quad (44)$$

と定義すれば、チェビシェフの不等式から、

$$\mathcal{P}\{A_{t,j,l}\} \leq e^{-\beta(t)/2} \quad (45)$$

である。確率事象 A, B におけるブールの不等式 $P(A \cup B) \leq P(A) + P(B)$ から、

$$\begin{aligned} \mathcal{P}\left(\bigcup_{t=1}^T \bigcup_{j=1}^{n_s} \bigcup_{l=1}^{n_u} A_{t,j,l}\right) &\leq \sum_{t=1}^T \sum_{j=1}^{n_s} \sum_{l=1}^{n_u} \mathcal{P}(A_{t,j,l}) \\ &\leq n_s n_u \sum_{t=1}^T e^{-\beta(t)/2} \end{aligned} \quad (46)$$

であるので、式 (38) の成り立つ確率は

$$\begin{aligned} \mathcal{P}\left(\bigcap_{t=1}^T \bigcap_{j=1}^{n_s} \bigcap_{l=1}^{n_u} \overline{A_{t,j,l}}\right) &= 1 - \left(\bigcup_{t=1}^T \bigcup_{j=1}^{n_s} \bigcup_{l=1}^{n_u} A_{t,j,l}\right) \\ &\geq 1 - n_s n_u \sum_{t=1}^T e^{-\beta(t)/2} \geq 1 - \sum_{t=1}^T \frac{\delta}{\varphi(t)} \geq 1 - \delta \end{aligned} \quad (47)$$

となる。■

上の式変形中、 $\{A_{t,j,l}\}_{j,l}$ に重複があり、 $A_{t,j',l'} = A_{t,j,l}$ となる場合、つまり、 $p_{j'}(u^{l'}) = p_j(u^l)$ なる添字の組 $\{j, l\}$ と $\{j', l'\}$ が存在する場合は、式中 $n_s n_u$ の部分をより小さく厳密に評価できる点に注意されたい。

つぎに、目的関数の真値がわかっていると仮定した上での最適解 $u^*(t)$ は以下のように定義できる。

$$\begin{aligned} J(u(t)) &= \mathbb{E}_{x(t+1)} [f(x(t+1)) | x(t) = x_j] \\ &= p_j^T(u(t))F, \end{aligned} \quad (48)$$

$$u^*(t) = \arg \min_{u(t) \in \mathcal{U}} J(u(t)). \quad (49)$$

つまり設計した最適化アルゴリズムによって与えられる制御入力 $u_{\text{algorithm}}(t)$ と本来の最適値との誤差は以下のようにして計算できる。

$$r(t) = J(u_{\text{algorithm}}(t)) - J(u^*(t)). \quad (50)$$

この regret が小さければ目的関数を既知とした真の最適解に近いということから、アルゴリズム精度を評価することができる。

〔補題 3.3〕 補題 3.2 の仮定が満たされているとき、 $r(t)$ は

以下の不等式を満たす：

$$\mathcal{P}\{r^2(t) \leq 4\beta(t)\lambda_{\max}(V(t)), \forall t \in \mathbb{Z}_{>0}\} \geq 1 - \delta \quad (51)$$

(証明) 学習した目的関数から得られた解 $u(t)$ と真の最適解である $u^*(t)$ を比較すれば、アルゴリズムの構成から

$$\begin{aligned} p_j^T(u(t))M(t) - \sqrt{\beta(t)}\sqrt{p_j^T(u(t))V(t)p_j(u(t))} \\ \leq p_j^T(u^*(t))M(t) - \sqrt{\beta(t)}\sqrt{p_j^T(u^*(t))V(t)p_j(u^*(t))} \\ \leq p_j^T(u^*(t))F \end{aligned} \quad (52)$$

であるので、

$$\begin{aligned} r(t) &= \mathbb{E}_{x(t+1)} [f(x(t+1)) | x(t), u(t)] \\ &\quad - \mathbb{E}_{x(t+1)} [f(x(t+1)) | x(t), u^*(t)] \\ &= p_j^T(u(t))F - p_j^T(u^*(t))F \\ &\leq p_j^T(u(t))F \\ &\quad - \left(p_j^T(u(t))M(t) \right. \\ &\quad \left. - \sqrt{\beta(t)}\sqrt{p_j^T(u(t))V(t)p_j(u(t))} \right) \\ &\leq 2\sqrt{\beta(t)}\sqrt{p_j^T(u(t))V(t)p_j(u(t))} \end{aligned} \quad (53)$$

を得る。上式変形中、右辺 2 行目から 3 行目の変形で前述のアルゴリズムの構成による不等式 (52)、右辺 3 行目から 4 行目の変形で補題 3.2 の不等式 (38) で $x(t) = x_j$ として

$$p_j^T(u^l)(F - M(t)) \leq \sqrt{\beta(t)}\sqrt{p_j^T(u(t))V(t)p_j(u(t))} \quad (54)$$

を用いた。補題 3.2 を用いたことから得られた式は確率 $1 - \delta$ で成り立つ。したがって各時刻 t において、いかなる状態 $x(t) = x_j$ にあっても、

$$\mathcal{P}\{r^2(t) \leq 4\beta(t)p_j^T(u(t))V(t)p_j(u(t))\} \geq 1 - \delta \quad (55)$$

が成り立つ。つぎに、確率遷移行列の性質から、 $\max \|p_j(u^l)\|_2 = 1, \forall j, \forall l$ であることから、

$$p_j^T(u(t))V(t)p_j(u(t)) \leq \lambda_{\max}(V(t)), \forall j \quad (56)$$

となる。ただし上式中 $\lambda_{\max}(V(t))$ は $V(t)$ の最大固有値である。この事実を用いれば、

$$\begin{aligned} 4\beta(t)p_j^T(u(t))V(t)p_j(u(t)) \\ \leq 4\beta(t)\lambda_{\max}(V(t)) \end{aligned} \quad (57)$$

となって定理の主張が成り立つ。■

《定理 3.1》 regret の時間積分

$$R(t) = \sum_{s=1}^t r(s) \quad (58)$$

を定義する. 補題 3.2 の仮定を満たす $\beta(t)$ が選択されたとき, アルゴリズムの regret の積分の上限は確率 $1 - \delta$ で不等式

$$R(t) \leq \sqrt{C_1 n_s \beta(t) t \gamma(t)} \quad (59)$$

を満たす. 式中の定数 $C_1 = 8 / (1 + \sigma_\varepsilon^{-2})$ として与えられる. (証明)

$$\lambda_{\max}(V(t)) \leq \text{trace } V(t) = \sum_{i=1}^{n_s} \sigma^2(t, x^i) \quad (60)$$

である. ここで, Lemma 5.4¹⁶⁾ より

$$\sigma^2(s, x) \leq C_2 \sigma_\varepsilon^2 \log(1 + \sigma_\varepsilon^{-2} \sigma^2(s, x)), \quad (61)$$

$$C_2 = \sigma_\varepsilon^{-2} / \log(1 + \sigma_\varepsilon^{-2}) \quad (62)$$

が成り立つので,

$$4 \sum_{s=1}^t \beta(s) \lambda_{\max}(V(s)) \leq 8 \sigma_\varepsilon^2 C_2 \beta(t) n_s \gamma(t), \quad (63)$$

となる. ここで, コーシー・シュワルツの不等式から得られる不等式 $R^2(t) = (\sum_{s=1}^t r(s))^2 \leq t \sum_{s=1}^t r^2(s)$ および 補題 3.3 で得られた確率 $1 - \delta$ で成り立つ不等式 $r^2(t) \leq 4\beta(t) \lambda_{\max}(V(t))$ と併せて, $C_1 = 8 \sigma_\varepsilon^2 C_2$ と定義し直せば

$$R^2(t) \leq C_1 n_s \beta(t) t \gamma(t) \quad (64)$$

となるので, 両辺平方根を取れば定理の主張が満たされる. ■

この結果から, 本研究で提案したアルゴリズムは一定の確率の下で求められたオーダーに従うことが保証された.

[系 3.1] 2 乗指数カーネルでアルゴリズムを適用した際の regret は確率 $1 - \delta$ で以下の式を満たす.

$$R(t) = \mathcal{O}\left(\sqrt{n_s \beta(t) t (\log t)^{n_x+1}}\right). \quad (65)$$

4. 数 値 例

ここでは簡便な例題でアルゴリズムの有効性を検証する. 論理状態は Fig. 2 内 x 軸に示したとおり, $[-10, 10] \subset \mathbb{R}$ を 20 等分して

$$\mathcal{S} = \{x^1, \dots, x^{20}\} = \{-10, \dots, 10\} \quad (66)$$

と定義され, Fig. 2 に示された目的関数を最適化の対象とする.

ここで, t 時刻での離散論理状態が $x(k) = x^j$ であるとし

Fig. 2 Objective function $f(x)$

て, 1 ステップ予測型の目的関数を最小化する最適化問題

$$\begin{aligned} & \min_{u(t) \in \mathcal{U}} J(u(t)), \\ & J(u(t)) = \mathbb{E}_{x(t+1)} [f(x(t+1)) | x(t) = x_j] \\ & \text{subject to} \\ & P_{ij}(u^l) = \mathcal{P}\{x(t+1) = x^i | x(t) = x^j, u(t) = u^l\} \end{aligned} \quad (67)$$

を検討する.

この目的関数 $f(x)$ は, すべての x 上で平均がゼロ, 共分散関数が 2 乗指数カーネル

$$k(x, x') = \exp\left(\frac{1}{2}(x - x')^2\right) \quad (68)$$

に従うガウシアンプロセスからのサンプリングされたものである. この状態変数の遷移は遷移確率行列

$$P_{ij}(u^k) = \begin{cases} \frac{1}{2}, & \text{if } i = k \\ \frac{1}{2 \times 19}, & \text{if } i \neq k \end{cases} \quad (69)$$

$$\mathcal{U} = \{u^1, \dots, u^{20}\} \quad (70)$$

により与えられる. この遷移行列は, 制御入力 $u^k, k = 1, \dots, 20$ をシステムに印加した場合, 次回に x^k に遷移する確率が $1/2$ となり, そのほかには均等に低確率で遷移する, という意味である. 例として,

$$P(u^2) = \begin{bmatrix} 1/38 & 1/2 & \cdots & 1/38 \\ 1/38 & 1/2 & \cdots & 1/38 \\ \vdots & \vdots & \ddots & \vdots \\ 1/38 & 1/2 & \cdots & 1/38 \end{bmatrix} \quad (71)$$

となる. 問題設定から, 目的関数 $f(x)$ は x^{11} のときに最小値をとるため, t 時刻の $x(t)$ とは独立に最適解はすべての時刻 t において $u(t) = u^{11}$ である.

$\beta(t)$ を以下のように変化させ計算結果の比較を行なった.

- (1) $\beta(t) = 0$ つまり探索なしで計算した.
- (2) $\beta(t)$ を提案アルゴリズムで時間的に変化させた.
- (3) $\beta(t) = 100$, 探索の重みを大きく設定し計算した.

3 つの場合で計算を行ない平均値を計算した.

制御入力の時系列の 10 回分のサンプルパスを示したものが Fig. 3 である. 図から, 提案アルゴリズムに比べて $\beta(t) = 0$ の場合は探索は確率遷移のみに依っている一方, $\beta(t) = 100$ と大きく設定した場合では領域全体を探索しているようすが観察される.

さらに (67) の真の最適解 $u^*(t)$ による目的関数値と, アルゴリズムによって得られた $u(t)$ による目的関数値の差 regret を

$$r(t) = J(u(t)) - J(u^*(t)) \quad (72)$$

とおき, 時間変化をプロットしたものが Fig. 4 である. このとき, 定理 3.1 の仮定となっている, 補題 3.3 で与えられた確率不等式について, $\delta = 0.2$ として計算した 100 のサンプル

Fig. 3 Control input $u(t)$

パス上すべてで補題 3.3 中括弧内の不等式は成り立った、つまり、不等式左辺は確率 1 となって成立することが確認できた。この事実から、系 3.1 も同様に成立し、regret は (59) 式で与えられたオーダに従うことがわかった。なお、 δ を 0 に近い値から 1 まで段階的に変化させても確率 1 で成り立ったことから、本例題での $\beta(t)$ の選択は保守的で探索に多く重みを割いてしまっていると考えられる。

Fig. 4 では、 $\beta(t)$ を提案手法で変化させた場合と 100 に設定した場合は大抵ゼロに収束しているのに対して、 $\beta(t) \equiv 0$ つまり探索に重みを課さなかった場合では regret に偏差が発生しているようすがわかる。これは、探索が進まずに最適解への収束を図れなかったことが原因と考えられる。また、 $\beta(t) \equiv 100$ と重点的に探索を継続したものは、時間変化の初期段階で提案手法より大きい regret となっていることも確認できる。

Fig. 4 Regret $r(t)$

regret の時間的に積分した $R(t) = \sum_{s=1}^t r(s)$ をプロットしたものが Fig. 5 である。Fig. 4 から、 $\beta(t) = 0$ にした場合には初期段階での探索の割合が少ないため、 $t = 60$ 以前程度までは提案した $\beta(t)$ の更新則よりも優れているが、途中から提案方法よりも劣る値となっている。

本例題の数値計算結果では提案した $\beta(t)$ の設定手法が

$\beta = 0, 100$ のケースに比べて良好な結果を示したが、この結果は必然ではなく、定理の前提となった β の選び方によって regret の上限が一定の確率で保証されている、という点のみを定理は主張している点に留意されたい。

したがって、提案した $\beta(t)$ は入力の数 n_u および離散論理状態数 n_s で表わされる問題のスケールによって β の設定の基準とその regret のオーダの概算を与えるものである。また、補題 3.2 で示したとおり、本研究で検証した 1 ステップ型予測型目的関数に関しては、離散論理状態の数 n_s および入力の数 n_u が増加するに従って、真の目的関数値と学習によって推定した目的関数の値の誤差も増加する傾向にある。このことから、問題のスケール n_s および n_u に伴って探索の重み β を増加させる必要があるといえる。また、(26) 式で表わされた観測ノイズ ε の分散 σ_ε^2 の増減はガウシアンプロセスによる学習式 (28) で考慮されていることから、陽に β で重み付けをすることなく、目的関数式 (31) 内の $V(t)$ として反映される点に注目されたい。

Fig. 5 Integral of regret

5. ま と め

本研究では、確率論理システムにおける 1 ステップ予測制御問題を、ガウシアンプロセスによるベイズ学習の観点から考察した。提案手法はオンラインで目的関数を探索しながら、学習したデータを使って目的関数値の平均と分散の推定を行ない、制御入力決定を行なった。

例題では、探索の重みを固定した場合と提案したアルゴリズムに則り変化させた場合とで regret の変化を比較し、良好な結果を確認することができた。提案した目的関数の予測方法を用いることで複数ステップのモデル予測制御に拡張するといった一般化を今後は行なっていきたい。

参 考 文 献

- 1) H. Qi and D. Cheng: Logic and logic-based control, *Journal of Control Theory and Applications*, **6**-1, 26/36 (2008)
- 2) D. Cheng and H. Qi: Controllability and observability of boolean control networks, *Automatica*, **45**-7, 1659/1667 (2009)
- 3) D. Cheng and H. Qi: A linear representation of dynamics of boolean networks, *IEEE Transactions on Automatic Control*, **55**-10, 2251/2258 (2010)

- 4) I. Shmulevich, E.R. Dougherty, S. Kim and W. Zhang: Probabilistic boolean networks: A rule-based uncertainty model for gene regulatory networks, *Bioinformatics*, **18**-2, 261/274 (2002)
- 5) R. Pal, A. Datta and E.R. Dougherty: Optimal infinite-horizon control for probabilistic boolean networks, *IEEE Transactions on Signal Processing*, **54**-6, 2375/2387 (2006)
- 6) Y. Zhao, Z. Li and D. Cheng: Optimal control of logical control networks, *IEEE Transactions on Automatic Control*, **56**-8, 1766/1776 (2011)
- 7) Y. Wu and T. Shen: Policy iteration approach to control residual gas fraction in IC engines under the framework of stochastic logical dynamics, *IEEE Transactions on Control Systems Technology*, **25**-99, 1/8 (2016)
- 8) Y. Wu and T. Shen: An algebraic expression of finite horizon optimal control algorithm for stochastic logical dynamical systems, *Systems & Control Letters*, **82**, 108/114 (2015)
- 9) A.E. Bryson: *Applied optimal control: Optimization, estimation and control*, CRC Press (1975)
- 10) M. Kang, F. Tahir, T. Shen and T. Ohtsuka: MPC-based speed tracking control design for spark-ignition engines, *SICE Journal of Control, Measurement, and System Integration*, **8**-3, 201/208 (2015)
- 11) T. Shen, J. Zhang, X. Jiao, M. Kang, J. Kako and A. Ohata: *Transient Control of Gasoline Engines*, CRC Press (2015)
- 12) M. Krstić and H.-H. Wang: Stability of extremum seeking feedback for general nonlinear dynamic systems, *Automatica*, **36**-4, 595/601 (2000)
- 13) D. Nesic, A. Mohammadi and C. Manzie: A framework for extremum seeking control of systems with parameter uncertainties, *IEEE Transactions on Automatic Control*, **58**-2, 435/448 (2013)
- 14) M. Guay, E. Moshksar and D. Dochain: A constrained extremum-seeking control approach, *International Journal of Robust and Nonlinear Control*, **25**-16, 3132/3153 (2015)
- 15) P. Auer: Using confidence bounds for exploitation-exploration trade-offs, *Journal of Machine Learning Research*, **3**, 397/422 (2003)
- 16) N. Srinivas, A. Krause, S.M. Kakade and M.W. Seeger: Information-theoretic regret bounds for gaussian process optimization in the bandit setting, *IEEE Transactions on Information Theory*, **58**-5, 3250/3265 (2012)
- 17) H. Abdelrahman, F. Berkenkamp, J. Poland and A. Krause: Bayesian optimization for maximum power point tracking in photovoltaic power plants, *Proceedings of European Control Conference 2016*, 2078/2083 (2016)
- 18) 小西, 北川: 情報量規準, 朝倉書店 (2004)
- 19) J. Berkson: Bayes' theorem, *Annals of Mathematical Statistics*, **1**-1, 42/56 (1930)
- 20) G. Chen, G. Chen and S.H. Hsu: *Linear Stochastic Control Systems*, Probability and Stochastics Series, Taylor & Francis (1995)
- 21) C.E. Rasmussen and C.K.I. Williams: *Gaussian Processes for Machine Learning*, The MIT Press (2005)

=====

[著 者 紹 介]

豊 田 充

2016 年上智大学大学院理工学研究科理工学専攻博士前期課程修了。同年同博士後期課程入学, 現在に至る。研究趣味は確率システム制御, 最適制御およびその自動車パワートレイン制御応用。日本機械学会などの会員。

申 鉄 龍 (正会員)

1992 年上智大学大学院理工学研究科機械工学専攻博士後期課程修了。工学博士。86 年中国東北重型機械学院自動制御学科専任講師, 89 年留学のため来日。上智大学理工学部機械工学科制御工学講座助手, 同准教授を経て現在機能創造理工学科教授。研究趣味は制御理論および機械系, 電力系, パワートレイン制御における応用。IEEE, IEEJ, SAE などの会員。

=====