非線形システムに対する入力から状態までの
近似線形化法とAcrobotの制御

山田 功*1, 湯澤 宏介*2
新田 浩二*2, 木下 宏一*2

Approximate Input-state Feedback Linearization for
Non-linear Systems and Control for the Acrobot

Kou YAMADA*1, Atsuyuki YUZAWA,
Kouji NITTA and Wataru KINOSHITA

*1Department of Mechanical System Engineering, Gunma University,
1-5-1 Tenjincho, Kiryu-shi, Gunma, 376-8515 Japan

In this paper, we consider a design method of approximate feedback linearization for non-linear systems to which the exact linearization method is not applicable. We adopt a two-step procedure to solve the approximate linearization. First, a state transformation matrix is settled, so that the non-linear system is transformed approximately into the controllable canonical form. Second, a standard non-linear linearization method is used to transform the controllable canonical form into a stable linear system. Finally, the application of the proposed method to the Acrobot, which is known as a system to which the exact linearization method cannot be applied, is shown to illustrate the effectiveness of the proposed method.

Key Words: Non-linear System, Approximate Input-state Feedback Linearization, Non-horonomic System, Acrobot

1. まえがき

本稿では、非線形システムに対する近似線形化法について検討する。非線形システムの線形化問題は、非線形システムの制御問題において、重要性が一つとなっている。広く用いられている近似線形化法に、テイラー展開の一次近似を用いて平衡点周りで近似線形化する方法がある。この方法は、手法が簡潔であること、ほとんどの非線形システムに適用できることから多くの非線形システムに適用されている。近似の精度が平衡点の近傍では十分な場合が多く、平衡点近傍の狭い動作領域しか持たない非線形システムに対しては有効である。しかし、動作領域が広い非線形システムに対しては、有効ではない。そのため、この方法は動作領域が広く非線形性の強いロボットなどの機械システムには適用できない場合も多いため、ロボットなどの機械システムに対しては、非線形フィードバックを用いた厳密な線形化法が提案されている。この方法は、近似を用いていないので、線形化されたシステムに対して設計された補償器は、平衡点の近傍のみならず状態全体で有効となる。したがって、この方法は、動作領域が広く非線形性の強い非線形システムに対しても適用可能である。しかし、この方法は、非線形システムを線形化するために、ロボットなどの機械システムの運動方程式が持つ構造的な特徴を利用しているので、この方法が適用可能な非線形システムは、限られている。そのため、この方法が適用できない非線形システムもある。たとえば、この方法は、移動ロボットや倒立振子には適用できない。

状態フィードバックのみで線形化できない非線形システムでは、座標変換と非線形フィードバックを用いた厳密な線形化手法が提案されている。この方法も近似を用いていないので動作領域の広い非線形シス템に対して適用可能である。また、非線形フィードバックのみで線形化できない非線形システムに対しても適用可能なので、これらの非線形システムに対して、この方法は強力な道具となっている。この方法は、次元が1次または2次の非線形システムに対しては比較的容易に適用できるが、次元が3次以上の非線形システムに対しては、必ずしも適用できるとは限らない。

このように、動作領域が広く非線形性が強い3次以上の非線形システムに有効な厳密な線形化法は、発表されていない。すなわち、動作領域が広く非線形性が
強い3次以上の非線形システムに対しては、非線形システムに対する近似線形化法に頼らざるをえない。座標変換と非線形フィードバックを通じて厳密な線形化ができない非線形システムに対しては、厳密な線形化法の近似的解法であるp次近似線形化法とExtended Linearization法が提案されている。参考文献のp次近似線形化法は、3次以上の非線形システムに対して、近似に1次近似線形化しか達成できない。また、参考文献のExtended Linearization法は、局所極を持つ非線形システムに対しては、適用ができないなどの問題点がある。

本稿では、これまで発表されてきた近似線形化法と異なり、設計がしやすく、厳密な線形化法が適用できる非線形システムに対しても適用可能な入力から状態までの近似線形化法を提案する。まず、本稿で検討する問題を定式化する。つぎに、座標変換と非線形フィードバックを通じた近似線形化法を提案する。提案方法は、2段階の手順から構成される。はじめに、制御対象を座標変換により、近似的に形式的かつ制御正準形に変換する。つぎに、形式的な制御正準形に書き換えたシステムを、非線形フィードバックを用いて安定な線形システムに変換する方法を採用する。提案方法を動作領域が広く非線形性の強いAcrobotの倒立制御に適用し、提案方法の有効性を数値例により検証する。提案方法は、テイラー展開の一次近似を用いていないので、動作領域が広く非線形性の強い非線形システムに対しても適用できる。提案方法が適用可能な条件は、それほど厳しいものではなく、多くの非線形システムに対して適用できる。

本稿で使用する記号を以下にまとめる。

\( R \) 実数空間
\( R^n \) n次元の実ベクトル空間
\( R^{n \times m} \) n行 m列の実行列
\( 0_{n \times m} \) n行 m列の零行列

2. 問題の記述

次式で表される入力の非線形システムについて考える。

\[
\frac{dx}{dt} = f(x) + b(x)u
\]  \( (1) \)

ただし、

\[
x = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^T \in R^n
\]  \( (2) \)

は状態、\( u \in R \) は制御入力とする。また \( f(x), b(x) \) は、

(1), (2) 式に矛盾しないサイズの非線形関数であり、\( x \)

に関して連続微分可能であるとする。一般性を失うことなく、原点 \( x = 0 \) が平衡点であるとする。また、

\[
f(x) \] の要素のうち \( x_i (i = 1, \ldots, n) \) の項式でない要素 \( \eta(x) \) は、\( i = 1, \ldots, n \) のうち少なくとも一つの \( i \) に対して、

\[
\frac{\partial \eta(x)}{\partial x_i} \bigg|_{x_i=0} < \infty,
\]  \( (3) \)

が成り立つと仮定する。

これらの仮定のもとで(1)式の非線形システムは、

\[
\frac{dx}{dt} = A(x) + b(x)u
\]  \( (4) \)

と書き直すことができる。ここで、(3) 式の仮定から、

\( A(0) \) の各要素は有界であることに注意する。

本稿では、(4) 式で表される非線形システムに対し、座標変換と非線形フィードバック制御を用いた入力から状態までの近似線形化法を検討する。すなわち、座標変換と非線形フィードバック制御を用いて、(4) 式を近似的に

\[
\frac{dz}{dt} = A_z z + b_z v
\]  \( (5) \)

なる安定な状態方程式に変換することを考える。ただし、\( A_z \in R^{n \times n} \) の固有値のすべては開左半平面にあるとし、\( b_z \in R^n \), \( v \in R \) は新しい制御入力とする。

3. 近似線形化法

ここでは、入力から状態までの新しい近似線形化法を提案する。提案する近似線形化法は、2段階の手順から構成される。はじめに、制御対象を、近似的に形式的な制御正準形に変換する。つぎに、形式的な制御正準形に対して、非線形フィードバックを用いて、(5) 式で表される安定な状態方程式に変換する。まず、座標変換を行うことにより、(4) 式で表される非線形システムを近似的に形式的な制御正準形に変換する方法を与える。次式で表される座標変換を考える。

\[
x = T(x)z
\]  \( (6) \)

ここで \( T(x) \) は、

\[
T(x) = \begin{bmatrix} b(x) & A(x)b(x) & \cdots & A^{n-1}(x)b(x) \\ a_1(x) & a_2(x) & \cdots & a_{n-1}(x) \end{bmatrix}
\]  \( (7) \)
 där, $a_i(x)(i = 0, \ldots, n)$ は、

$$
\det(sI - A(x)) = a_n(x)s^n + a_{n-1}(x)s^{n-1} + \cdots + a_0(x)
$$

(8)

である。ここで、一般性を失うことなく、$a_n(x) = 1$ である。また、座標変換に伴う近似としては

$$
dT(x)/dt = T^{-1}(x)x \approx 0 \forall t, x
$$

(9)

を採用する。なお、平衡点近傍では $x \equiv 0$ が成り立つことから、(9) 式が常に成り立つ。またゆっくりした動作しか要求されない非線形システムに対しても $dT(x)/dt$ の各要素が小さな値を持つことから (9) 式が成り立つことに注意する。

(9) 式の近似のもとで、(6) 式の座標変換を (4) 式で表される非線形システムに適用すると、座標変換されたシステムに関して、つぎの定理が成り立つ。

定理 1 つぎの条件

$$
det T(x) \neq 0
$$

(10)

を満足する領域において、(9) 式の近似のもとで、(6) 式で表される座標変換を (4) 式の非線形システムに対して適用すると、

$$
\frac{dz}{dt} = T^{-1}(x)A(x)T(x)z + T^{-1}(x)b(x)u = A_c(x)z + b_c(x)u
$$

(11)

となる。ただし、$A_c(x), b_c(x)$ は、それぞれ

$$
A_c(x) = \begin{bmatrix}
0 & 1 & 0 & \cdots & 0 \\
\vdots & \ddots & \ddots & \cdots & \vdots \\
0 & \ddots & \ddots & \ddots & \vdots \\
-a_0(x) & -a_1(x) & \cdots & -a_{n-1}(x) & 1
\end{bmatrix} \in R^n
$$

(12)

$$
b_c(x) = \begin{bmatrix}
0 \\
\vdots \\
0 \\
a_0(x) \\
a_1(x)
\end{bmatrix}^T \in R^n
$$

(13)

である。

定理 1 の証明には、つぎの補題が必要である。

補題 1 $A(x) \in R^{n \times n}(x)$ に対して $\det(sI - A)$ を

$$
\Phi(x) = \det(sI - A(x)) = s^n + a_{n-1}(x)s^{n-1} + \cdots + a_0(x)
$$

(14)

とおく。このとき

$$
\Phi(A(x)) = A(x)^n + a_{n-1}(x)A^{n-1}(x) + \cdots + a_0(x)I = 0
$$

(15)

が成り立つ。

(証明) 行列式の定義(15)(16)から,

$$
(sI - A(x)) \cdot \text{adj}(sI - A(x)) = \text{adj}(sI - A(x)) \cdot (sI - A(x)) = \Phi(x)I
$$

(16)

が成り立つ。したがって、明らかに

$$
\Phi(A(x)) = 0
$$

(17)

を満足する。以上のことから補題 1 は証明された。

注意 1 補題 1 の (15) 式は、$A \in R^{n \times n}$ のときにはケーリーハミルトンの定理(15)(16)としてよく知られている。

補題 1 は、$A(x)$ が非線形要素を含んでいたとしてもケーリーハミルトンの定理が成り立つことを示したものであることに注意すること。

補題 1 を用いて定理 1 を証明する。

(証明) (9) 式の近似と (10) 式の仮定から

$$
\frac{d\dot{x}}{dt} = \frac{dT(x)}{dt}T^{-1}(x)x + T(x)\frac{dz}{dt}
$$

(18)

が成り立つ。 (10) 式の仮定のもとで (18) 式は

$$
\frac{dz}{dt} = A_c(x)z + b_c(x)u
$$

(19)

と書きかえられる。ここで、$A_c(x), b_c(x)$ はそれぞれ

$$
A_c(x) = T^{-1}(x)A(x)T(x)
$$

(20)

$$
b_c(x) = T^{-1}(x)b(x)
$$

(21)

である。したがって、残された問題は、(20), (21) 式の $A_c(x), b_c(x)$ がそれぞれ (12), (13) 式となることを示すことである。

$$
T(x) = \begin{bmatrix}
e_1(x) & \cdots & e_n(x)
\end{bmatrix}
$$

(22)

と記述する。ここで,

$$
e_i(x) = \sum_{j=0}^{n-i} a_{i+j}(x)A^j(x)b(x)
$$

(23)

で、
非線形システムに対する入力から状態までの近似線形化法とAcrobotの制御

である。補題1から $A(x) e_1(x)$ は,

$$
A(x) e_1(x) = (A^n(x) + a_{n-1}(x)A^{n-1}(x) + \cdots + a_1(x)A(x)) b(x) \\
= (A^n(x) + a_{n-1}(x)A^{n-1}(x) + \cdots + a_0(x)) b(x) \\
- a_0(x) b(x) \\
= -a_0 b(x)
$$

となる。$A(x) e_1(x)$ の計算と同様にして、$A(x) e_i(x) (i = 2, \ldots, n)$ は,

$$
A(x) e_2(x) = 
\begin{bmatrix}
1 & 0 \\
\vdots & \vdots \\
0 & 0 \\
-a_1(x) & 0
\end{bmatrix}
$$

$A(x) e_n(x)$ は,

$$
A(x) e_n(x) = 
\begin{bmatrix}
1 & 0 \\
\vdots & \vdots \\
0 & 0 \\
-a_n(x) & 1
\end{bmatrix}
$$

となる。以上のことをまとめて

$$
A(x) T(x)
= 
\begin{bmatrix}
0 & 1 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & \vdots & \vdots & \cdots & 0 \\
-a_0(x) & -a_1(x) & \cdots & -a_{n-1}(x) & 1
\end{bmatrix}
$$

が成り立つ。式(10)の仮定と(25)式から,

$$
A_t(x) = T^{-1}(x) A(x) T(x)
= 
\begin{bmatrix}
0 & 1 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & \vdots & \vdots & \cdots & 0 \\
-a_0(x) & -a_1(x) & \cdots & -a_{n-1}(x) & 1
\end{bmatrix}
$$

が得られる。

また、$b(x)$ は (24) 式から,

$$
b(x) = e_n(x) \\
= T(x) \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}^T
$$

と表される。(10) 式の仮定、(21) 式と(27) 式から $b_2(x)$ は,

$$
b_2(x) = T^{-1}(x) b(x) \\
= \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}^T
$$

と表される。以上のことから、定理1は証明された。

つきに、(9) 式の近似のもとで (6) 式の座標変換を施された (11) 式のシステムを、非線形フィードバックを用いて安定な線形システムに変換することを考える。

(11) 式に対して

$$
u(t) = - \begin{bmatrix} -a_0(x) + \lambda_0 & \cdots & -a_{n-1}(x) + \lambda_{n-1} \end{bmatrix} z + \nu \\
= -F_2(x) z + \nu
$$

で制御入力を与える。このとき、(11) 式のシステムは,

$$
\frac{dz}{dt} = 
\begin{bmatrix}
0 & 1 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & \vdots & \vdots & \cdots & 0 \\
-\lambda_0 & -\lambda_1 & \cdots & -\lambda_{n-1} & 1
\end{bmatrix} z + \begin{bmatrix} 0 \\
\vdots \\
\vdots \\
1 \\
1
\end{bmatrix} \nu
$$

と書き換えられる。ただし、$\nu$ は、新しい制御入力、$A_t$, $b_s$ はそれぞれ,

$$
A_t = \begin{bmatrix}
0 & 1 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & \vdots & \vdots & \cdots & 0 \\
-\lambda_0 & -\lambda_1 & \cdots & -\lambda_{n-1} & 1
\end{bmatrix}
$$

$$
b_s = b_s(x) = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}^T
$$

である。また、$\lambda_i (i = 1, \ldots, n-1)$ は,

$$
\lambda_i (i = 1, \ldots, n-1) = 0
$$

\[\]
の根がすべて開左半平面に配置されるように選ぶものとする。式のようなシステムは、明らかに線形システムである。$A_r$の固有値は$(33)$式の根と一致し、$A_r(i=0,\ldots,n-1)$が$(33)$式の根が開左半平面に存在するように選ばれていることから、$(30)$式が安定であることにも確認される。以上のことから、座標変換行列$T(x)$、制御入力$u(t)$をそれぞれ$(7)$、$(29)$式で与え、$(9)$式の近似を行うことにより、$(4)$式の非線形システムを線形で安定な状態方程式$(5)$式に変換することができる。

つぎに、提案方法が適用可能であるための条件を明らかにする。定理1の議論から、提案方法が適用可能であるための必要十分条件は、$(7)$式の座標変換行列$T(x)$が正則であることである。ここで、$(7)$式中の行列

$$
\begin{bmatrix}
    a_1(x) & a_2(x) & \cdots & a_{n-1}(x) & 1 \\
    a_2(x) & 1 & & & \\
    \vdots & & & & \\
    a_{n-1}(x) & & & & 1 \\
    1 & & & & 0
\end{bmatrix}
$$

は明らかに正則である。このことより、提案方法が適用可能であるための必要十分条件は、

$$
\begin{bmatrix}
    b(x) & A(x)b(x) & \cdots & A^{n-1}(x)b(x)
\end{bmatrix}
$$

が正則であることである。

注意2以上のことから、本稿の提案方法は、非線形システムが原点に極を持たないという仮定をせずに議論している。そのため、本稿の提案方法は、非線形システムが原点極を持つか否かに関わらず、

$$
\begin{bmatrix}
    b(x) & A(x)b(x) & \cdots & A^{n-1}(x)b(x)
\end{bmatrix}
$$

が正則であるならば、適用可能することに注意する。

なお、すべての状態$x$に対し$det(T(x))\neq 0$の条件が成り立たない非線形システムであったとしても、平衡点近傍において$det(T(x))\neq 0$が成り立つ範囲が狭い場合には、つきのようにすることにより提案方法が適用できる。本提案方法の問題点は、$det(T(x))=0$のときには、$(29)$式の制御入力$u(t)$が計算できないことに、$det(T(x))=0$に近いとき、$(29)$式の制御入力が大きな値となり、その極限として$det(T(x)=0$のときは制御入力$u(t)$が無限大となっていると解釈することができる。無限大の制御入力$u(t)$を用いることはできない。そこで、制御入力$u(t)$の後に飽和要素を加え、制御入力に制限を加えることを考え、この方法を用いることに、$det(T(x)=0$の場合には、許容された上限（または下限）を制御入力$u(t)$とすることで、制御入力$u(t)$が計算できないという問題点が解決される。たとえば、ある時間$det(T(x)=0$が成り立ったとしても、$det(T(x)=0$が成り立つ範囲が狭いという仮定から、許容された上限（または下限）を制御入力$u(t)$とすることで、ほとんどの場合に$det(T(x)=0$が成り立つ領域から抜けて出$c(T(x))\neq 0$が成り立つようになり、$(29)$式を用いて制御入力$u(t)$が計算できる。このように、すべての状態$x$に対して$det(T(x)\neq 0$の条件が成り立たない非線形システムに対しても、飽和要素を加えることで、本稿の提案方法は適用可能となる。4.で、この考えに基づき、すべての状態$x$に対し$det(T(x)\neq 0$が成り立たず、原点極を持つAcrobotの倒立制御に、本提案方法が適用できることを数値例により検証する。

4. Acrobotの倒立制御への適用

ここでは、3.で述べた近似線形化法が、動作領域を広く非線形性の強いAcrobotの倒立制御設計問題に有効であることを数値例により検証する。

Acrobotの機構を図1に示す。図1においてAcrobotは肘関節部にアクチュエータを持って、肩関節部にはアクチュエータを持たないことに注意する。図1のAcrobotの運動方程式は次式のように記述される。

$$
\begin{align*}
    d_1(t)q_1(t) + d_2(t)q_2(t) + h_1(t) + \Phi_1(t) \\
    d_3(t)q_1(t) + d_4(t)q_2(t) + h_2(t) + \Phi_2(t) \\
    u(t)
\end{align*}
$$

ただし、$q_1(t)$は$y$軸とリンク1間の角度、$q_2(t)$はリンク1とリンク2間の角度、$q_1(t)$、$q_2(t)$はそれぞれ$q_1(t)$、$q_2(t)$の角速度、$\dot{q}_1(t)$、$\dot{q}_2(t)$はそれぞれ$q_1(t)$、$q_2(t)$の角加速度、$u(t)$は制御入力（肘関節部の入力トルク）とし、

$$
\begin{align*}
    d_1(t) &= m_1q_1^2 + m_2(\dot{q}_1^2 + \dot{q}_2^2 + 2\dot{q}_1\dot{q}_2 \cos(q_2(t))) + h_1(t) \\
    d_2(t) &= m_2q_2^2 + h_2(t) \\
    d_1(t) &= m_2q_2^2 + l_1 \dot{q}_2 (\sin(q_2(t))) + l_2
\end{align*}
$$

Fig. 1 The Acrobot

－165－
\[d_{21}(t) = m_2 \left\{ I_{l_2}^2 + l_1 l_2 \cos(q_2(t)) \right\} + l_2\]
\[h_1(t) = -m_2 l_1 l_2 \sin(q_2(t)) \dot{q}_2(t) + 2m_2 l_1 l_2 \sin(q_2(t)) \dot{q}_2(t) \dot{q}_1(t)\]
\[h_2(t) = m_2 l_2 \sin(q_2(t)) \dot{q}_2(t)^2\]
\[\phi_1(t) = (m_1 l_1 + m_2 l_2) \cos(q_1(t)) + m_2 l_2 \cos(q_1(t) + q_2(t))\]
\[\phi_2(t) = m_2 l_2 \cos(q_1(t) + q_2(t))\]

であり、\(m_1, m_2\) はそれぞれリンク1, 2の質量、\(l_1, l_2\) はそれぞれリンク1, 2の長さ、\(l_1, l_2\) はそれぞれリンク1, 2のリンク端から重心までの長さ、\(l_1, l_2\) はそれぞれリンク1, 2の重心周りの慣性モーメントとする。シミュレーションで使用したパラメータをTable 1にまとめると、

<table>
<thead>
<tr>
<th>(m_1)</th>
<th>(m_2)</th>
<th>(l_1)</th>
<th>(l_2)</th>
<th>(l_1)</th>
<th>(l_2)</th>
<th>(g)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0.5</td>
<td>1</td>
<td>0.083</td>
</tr>
</tbody>
</table>

提案方法を用いて、図1のAcrobotを倒立させる

\[x = \begin{bmatrix} x_1(t) & x_2(t) & x_3(t) & x_4(t) \end{bmatrix}^T \]
\[= \begin{bmatrix} q_1(t) & q_2(t) & \dot{q}_1(t) & \dot{q}_2(t) \end{bmatrix}^T \quad (35)\]

とする。このとき、図1のAcrobotを倒立させる問題は、

\[\lim_{t \to \infty} x = 0 \quad (36)\]

とする制御問題を等価になる。

状態変数を（35）式とおくと、（34）式で表されるAcrobotの運動方程式は、（4）式の形式に書き換えることができる。ただし、

\[A(x) = [A_{ij}(x)], (i = 1, \ldots, 4; j = 1, \ldots, 4) \quad (37)\]

\[A_{11}(x) = 0, A_{12}(x) = 0, A_{13}(x) = 1, A_{14}(x) = 0\]
\[A_{21}(x) = 0, A_{22}(x) = 0, A_{23}(x) = 0, A_{24}(x) = 1\]
\[A_{31}(x) = \frac{\sin(x_1(t))}{x_1(t)} \left\{ \frac{I_{l_2}^2 + l_1 l_2 \cos(x_2(t))}{x_1(t)} \right\} \]
\[A_{32}(x) = \frac{l_1 l_2 m_2 \sin(x_2(t))}{x_2(t)} \left\{ \frac{\left( l_2 + l_2^2 \cos(x_2(t)) \right)^2}{x_2(t)} \right\} \]
\[A_{33}(x) = \frac{l_1^2 l_2^2 m_2^2 \sin(x_2(t))}{(l_1 + l_1^2 m_1 + l_2 m_2)(l_2 + l_2^2 m_2)} \cos(x_2(t)) x_3(t) \]
\[A_{34}(x) = 0\]
\[A_{41}(x) = 0\]
\[A_{42}(x) = \frac{g \sin(x_1(t))}{x_1(t)} \left\{ \frac{(l_1 + l_1^2 m_1 + l_2 m_2)(l_2 + l_2^2 m_2)}{x_1(t)} \right\} \]
\[A_{43}(x) = 0\]
\[A_{44}(x) = 0\]

\[b(x) = [b_1(x)](i = 1, \ldots, 4) \quad (38)\]

\[b_1(x) = 0, b_2(x) = 0\]
\[b_3(x) = \frac{l_1 l_2^2 m_2 + l_1 l_2 m_2 \cos(x_2(t))}{x_1(t)} \left\{ \frac{(l_1 + l_1^2 m_1 + l_2 m_2)(l_2 + l_2^2 m_2)}{x_1(t)} \right\} \]
\[b_4(x) = \frac{l_1 l_2 m_2 \sin(x_2(t))}{x_2(t)} \left\{ \frac{(l_1 + l_1^2 m_1 + l_2 m_2)(l_2 + l_2^2 m_2)}{x_2(t)} \right\} \]

となる。このとき \(A(0)\) のすべての要素は有界であることが確認できる。

つきに、座標変換行列 \(T(x)\) と制御入力 \(u(t)\) を与える。\(\det(sl - A(x))\) は、

\[\det(sl - A(x)) = s^4 + a_3(x)s^3 + a_2(x)s^2 + a_1(x)s + a_0(x)\]

と表される。ただし、

\[a_0(x) = 24500 [49 \sin(2x_1(t)) (882509 \sin(x_2(t)) \]
\[260289 - 50000 \cos(2x_1(t))^2x_1(t)x_2(t) \]
\[-181225 \sin(3x_2(t)) + 5 \sin(x_1(t)) \]
\[343039 \]
\[\sin(2x_1(t)) - 100000 \sin(4x_2(t)) \]

\[a_3(x) = \frac{\sin(x_1(t))}{x_1(t)} \left\{ \frac{(l_1 + l_1^2 m_1 + l_2 m_2)(l_2 + l_2^2 m_2)}{x_1(t)} \right\} \]
\[a_2(x) = \frac{\left( l_2 + l_2^2 \cos(x_2(t)) \right)^2}{x_2(t)} \]

\[a_1(x) = \frac{l_1^2 l_2^2 m_2^2 \cos(x_2(t))}{(l_1 + l_1^2 m_1 + l_2 m_2)(l_2 + l_2^2 m_2)} \cos(x_2(t)) x_3(t) \]
\[a_0(x) = 0\]
非線形システムに対する入力から状態までの近似線形化法と Acrobot の制御

\[ a_1(x) = \frac{-100000\sin^2(x_2(t))x_3(t)\{1 - 127289 + \frac{[260289 - 50000\cos^2(x_2(t))]x_2(t)}{50000\cos(2x_2(t))}\}}{x_3(t)} + x_4(t) \]

\[ a_2(x) = \frac{20[245x_2(t)\sin(x_1(t))\{-299 + 100\}}{\{-260289 + 50000\cos(2x_2(t))\}x_1(t)x_2(t)\cos(2x_2(t)) + x_1(t)[-49\cos(x_1(t))]} \]

\[ + 500\sin(2x_2(t)) - 50[133 \sin(x_2(t)) + 500\sin(2x_2(t))] - 50[133 \sin(x_2(t)) + 500\sin(2x_2(t))]d(x_3(t) + x_4(t))] \]

\[ a_3(x) = \frac{50000\sin^2(2x_1(t))x_3(t)}{-260289 + 50000\cos(2x_2(t))} \]

である。式 (37), (38) 式を用い、座標変換行列 \( T(x) \) を (7) 式で与える。制御入力が (29) 式で与える。ただし、(31) 式の \( A \) の固有値が 3.05 の 4 重根となるように \( \lambda_2 \) は 3.32 および \( \lambda_0 = 86.54, \lambda_1 = 113.5, \lambda_2 = 55.81, \lambda_3 = 12.2 \) とし、\( v = 0 \) とする。

前節での議論から、本稿の提案方法が適用可能であるためには、det \( T(x) \neq 0 \) を満足しなくてはいけない、det \( T(x) = 0 \) の条件を解析的に求めることが難しい。そこで、det \( T(x) \) が、\( x_1(t), x_2(t), x_3(t), x_4(t) \) のみで定まることが知られていることから、det \( T(x) \) の値を求めるシミュレーションを数多く行った。その結果、ほとんどすべての場合に det \( T(x) \neq 0 \) を満たすことが明らかにされた。さらに、ある状態 \( x \) で det \( T(x) = 0 \) を満たしたとしても、その状態が連続的に続く可能性が低いことが確認された。紙面の都合上、図 2 に、\( x_1(t) = q_1(t) = 0, x_2(t) = q_2(t) = 0 \) のみの、det \( T(x) \) の値を示す。図 2 において、det \( T(x) \) の最大値は -0.7056 であるので、\( x_1(t) = q_1(t) = 0, x_2(t) = q_2(t) = 0 \) の場合には、det \( T(x) = 0 \) とはならないことが確認される。

Fig. 2  det \( T(x) \)(x_1(t) = q_1(t) = 0, x_2(t) = q_2(t) = 0)

\[ -10^7 \leq u(t) \leq 10^7 \] (39)

の範囲内で与える方法を採用することにより、提案方法が Acrobot の制御系設計に適用可能であることを検証する。初期状態を \( x_1(0) = q_1(0) = \pi[\text{rad}], x_2(0) = q_2(0) = 0.1[\text{rad}], x_3(0) = 0, x_4(0) = 0[\text{rad/sec}], x_4(0) = 0[\text{rad/sec}] \) のとき、\( q_1(t), q_2(t) \) の時間応答と制御入力 \( u(t) \) をそれぞれ図 3、図 4 に示す。ただ

Fig. 3 The responses of \( q_1(t) \) and \( q_2(t) \)(-10^7 \leq u(t) \leq 10^7)

Fig. 4 The responses of \( u(t) \)(-10^7 \leq u(t) \leq 10^7)

し、図 3 の実線は \( q_1(t) \) の時間応答、破線は \( q_2(t) \) の時間応答をそれぞれ表す。図 3 には、15[sec] 付近に近い \( u(t) \) の値が Acrobot の制御可能であることが示されている。また、図 4 には、かなり大きな制御入力が加えられている時間があることから、det \( T(x) \) が 0 の近傍になる場合があるので、すべての場合において、制御入力が (39) 式の範囲内であることが示されている。このことから、すべての状態 \( x \) で det \( T(x) \) が 0 を満たし続けることが制御が実現できることが確認される。
なお、上記のシミュレーションと同じ設定で、制御入力 $u(t)$ の制限を

$$-13000 \leq u(t) \leq 13000$$

(40)

としたときの Acrobot の倒立制御問題を考える。このとき、図 4 から、制御入力は、(40) 式で制限される時間帯があることになる。$q_1(t), q_2(t)$ の時間応答を図 5 に示す。ここで、図 5 の実線は $q_1(t)$ の時間応答、破

立倒立制御に有効であることが数値例で示され、提案方法は、動作領域が広く非線形性の強い非線形システムに対しても適用できることが明らかにされた。

未発表ながら、重要なご指摘をいただきました査読委員に感謝いたします。本稿の一部は、竹川研究助成の援助を受けた。ここに記し、感謝の意を表したい。

文献

(4) 石島、鳥、石、山下、三平、渡辺: 非線形システム論，計測自動制御学会 (1993)
(5) 三平、古田: 三輪バイラの振動子の制御，第 3 回自動制御連携講演会第 31 回，371/372 (1987)
(6) 小島、山口、坂本: ロボットによる倒立振子の安定化制御，機械学会論文集 (C), 57-544, 3890/3894 (1991)
(7) 助、小島、金、小林: ファジィ理論に基づく倒立振り子制御ロボットの振り上げ機械の実装と開発，機械学会論文集 (C), 58-555, 3307/3312 (1992)
(10) S. Sastry: Nonlinear systems analysis, stability, and control, Springer (1999)
(11) 高木、黒沢、空間線形化による非線形制御系の設計，システム/制御情報，33-9, 454/461 (1989)
(15) 須美, 须藤: システム制御のためのマトリックス理論, 計測自動制御学会 (1978)