

混合座標法に基づく線状ソフトロボットの動的モデリング

Dynamic Modeling of Linear Soft Robots based on Mixed Coordinates

正 平井 慎一 (立命館大)

Shinichi HIRAI, Ritsumeikan University, hirai@se.ritsumei.ac.jp

This paper presents dynamic modeling of linear soft robots based on mixed coordinates. We focus on bendable yet inextensible linear soft robots acting on two-dimensional space. Modeling of such linear robots has been based on differential geometry coordinates, but it suffers from complex calculation in kinetic energy formulation. This paper combines differential geometry coordinates and Cartesian coordinates to reduce complex calculation. We show the formulation based on the mixed coordinates with simulation results.

Key Words: Soft Robots, Linear Robots, Modeling, Dynamics

1 緒言

本報告では、線状ソフトロボットの動的モデリングのために、混合座標法を提案する。一方向の寸法が支配的であり、他の二方向の寸法を無視することができる線状ソフトロボットのモデリング手法として、微分幾何座標が提案されている [1]。この手法は、三次元空間内の線状ソフトロボットの静的な安定形状を計算することができる。また、微分幾何座標を用いて、二次元空間内で曲げ変形を生じる線状ソフトロボットの動的なモデリング手法が提案されている [2]。動的なモデリングにおいては、運動エネルギーを解析的に求め、慣性行列を計算する必要がある。微分幾何座標を用いる手法においては、運動エネルギーを解析的に求める過程が煩雑になり、結果として慣性行列の計算に多大な計算負荷を生じた。そこで本報告では、微分幾何座標と直交座標を混合して用いる手法を提案する。これにより、運動エネルギーの計算における積分が簡単になり、慣性行列の計算が簡単になる。本原稿では、二次元空間内で運動する線状ソフトロボットを対象として、動的なモデリングの過程とシミュレーション結果を示す。

2 混合座標法

曲げ変形を生じ、伸縮変形を伴わない線状ソフトロボットが、二次元空間内で運動するとする。線状ソフトロボットの全長を L で表し、ボディに沿う一端からの距離が ξ である点を $P(\xi)$ で表す。時刻 t において、点 $P(\xi)$ における接線が水平線と成す角度を $\theta(\xi, t)$ と表す (図 1)。区間 $[0, L]$ に、一様に分布する n 個の節点 P_1, P_2, \dots, P_n を設定する。区間幅は $h = L/(n-1)$ で与えられる。節点 P_k における距離は $\xi_k = (k-1)h$ と表される。節点における角度を、 $\theta_k = \theta(\xi_k, t)$ と表す。関数 $\theta(\xi, t)$ は区間 $[\xi_k, \xi_{k+1}]$ で、 $\theta(\xi, t) = \theta_k(t)N_{k,k+1}(\xi) + \theta_{k+1}(t)N_{k+1,k}(\xi)$ と近似できる。ここで関数 $N_{i,j}(\xi)$ は、 $N_{i,j}(\xi_i) = 1$ と $N_{i,j}(\xi_j) = 0$ を満たす一次関数である。上式を時間微分すると、角速度 $\dot{\theta}(\xi, t) = \dot{\theta}_k(t)N_{k,k+1}(\xi) + \dot{\theta}_{k+1}(t)N_{k+1,k}(\xi)$ を得る。以降、表現を簡単にするために、必要でない限り引数の t を省略する。たとえば上記の式は

$$\theta(\xi) = \theta_k N_{k,k+1}(\xi) + \theta_{k+1} N_{k+1,k}(\xi) \quad (1)$$

$$\dot{\theta}(\xi) = \dot{\theta}_k N_{k,k+1}(\xi) + \dot{\theta}_{k+1} N_{k+1,k}(\xi) \quad (2)$$

と表す。

点 $P(\xi)$ の位置ベクトルは、0 から ξ までの積分を用いて表される。したがって、点 $P(\xi)$ の速度ベクトルは、0 から ξ までの積分を用いて表される。運動エネルギーの計算においては、速度の二乗を求める必要がある。積分区間 $[0, \xi]$ は、複数の区間 $[\xi_1, \xi_2], [\xi_2, \xi_3], \dots$ から成るため、速度の二乗の計算が煩雑になる。そこで、新しい変数

$$\mathbf{x}_k = \begin{bmatrix} x_k \\ y_k \end{bmatrix} = \int_0^{\xi_k} \begin{bmatrix} \cos \theta(u) \\ \sin \theta(u) \end{bmatrix} du \quad (3)$$

を導入する。このとき、区間 $[\xi_k, \xi_{k+1}]$ 内の点 $P(\xi)$ の位置ベク

トルは

$$\mathbf{x}(\xi) = \mathbf{x}_k + \int_{\xi_k}^{\xi} \begin{bmatrix} \cos \theta(u) \\ \sin \theta(u) \end{bmatrix} du \quad (4)$$

と表すことができる。点 $P(\xi)$ の速度は

$$\dot{\mathbf{x}}(\xi) = \dot{\mathbf{x}}_k + \int_{\xi_k}^{\xi} \begin{bmatrix} -\sin \theta(u) \\ \cos \theta(u) \end{bmatrix} \dot{\theta}(u) du \quad (5)$$

となる。

以降、 $i = k$ のとき $i' = k+1$ 、 $i = k+1$ のとき $i' = k$ と定義する。さらに、以下の関数

$$C_k(\xi) \triangleq \int_{\xi_k}^{\xi} \cos \theta(u) du$$

$$S_k(\xi) \triangleq \int_{\xi_k}^{\xi} \sin \theta(u) du$$

$$C_{k,i}^1(\xi) = \int_{\xi_k}^{\xi} \cos \theta(u) N_{i,i'}(u) du$$

$$S_{k,i}^1(\xi) = \int_{\xi_k}^{\xi} \sin \theta(u) N_{i,i'}(u) du$$

$$C_{k,i,j}^2(\xi) = \int_{\xi_k}^{\xi} \cos \theta(u) N_{i,i'}(u) N_{j,j'}(u) du$$

$$S_{k,i,j}^2(\xi) = \int_{\xi_k}^{\xi} \sin \theta(u) N_{i,i'}(u) N_{j,j'}(u) du$$

$$Q_{k,i,j}^2(\xi) = C_{k,i}^1(\xi) C_{k,j}^1(\xi) + S_{k,i}^1(\xi) S_{k,j}^1(\xi)$$

$$Q_{k,i,j,l}^3(\xi) = \{S_{k,i}^1(\xi) C_{k,j,l}^2(\xi) - C_{k,i}^1(\xi) S_{k,j,l}^2(\xi)\} \\ + \{S_{k,j}^1(\xi) C_{k,i,l}^2(\xi) - C_{k,j}^1(\xi) S_{k,i,l}^2(\xi)\}$$

と定積分

$$\bar{C}_{k,i}^1 = \int_{\xi_k}^{\xi_{k+1}} C_{k,i}^1(u) du$$

$$\bar{S}_{k,i}^1 = \int_{\xi_k}^{\xi_{k+1}} S_{k,i}^1(u) du$$

$$\bar{C}_{k,i,j}^2 = \int_{\xi_k}^{\xi_{k+1}} C_{k,i,j}^2(u) du$$

$$\bar{S}_{k,i,j}^2 = \int_{\xi_k}^{\xi_{k+1}} S_{k,i,j}^2(u) du$$

$$\bar{Q}_{k,i,j}^2 = \int_{\xi_k}^{\xi_{k+1}} Q_{k,i,j}^2(u) du$$

$$\bar{Q}_{k,i,j,l}^3 = \int_{\xi_k}^{\xi_{k+1}} Q_{k,i,j,l}^3(u) du$$

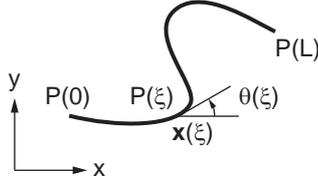


Fig.1 Coordinates of soft linear robot

を導入する。このとき、 $i, j, \in \{k, k+1\}$ に対して

$$\frac{\partial}{\partial \theta_j} C_{k,i}^1(\xi) = -S_{k,i,j}^2(\xi) \quad (6)$$

$$\frac{\partial}{\partial \theta_j} S_{k,i}^1(\xi) = C_{k,i,j}^2(\xi) \quad (7)$$

を得る。同様に、 $i, j, l \in \{k, k+1\}$ に対して

$$\frac{\partial}{\partial \theta_l} Q_{k,i,j}^2(\xi) = Q_{k,i,j,l}^3(\xi) \quad (8)$$

を得る。なお、積分 $C_k(\xi)$, $S_k(\xi)$, $C_{k,i}^1(\xi)$, $S_{k,i}^1(\xi)$, $C_{k,i,j}^2(\xi)$, $S_{k,i,j}^2(\xi)$ は、解析的に求めることができる。

積分区間 $[\xi_k, \xi]$ で、角速度は式 (2) で表される。点 $P(\xi)$ の速度を成分で書き表すと

$$\dot{x}(\xi) = \dot{x}_k - S_{k,k}^1(\xi)\dot{\theta}_k - S_{k,k+1}^1(\xi)\dot{\theta}_{k+1} \quad (9)$$

$$\dot{y}(\xi) = \dot{y}_k + C_{k,k}^1(\xi)\dot{\theta}_k + C_{k,k+1}^1(\xi)\dot{\theta}_{k+1} \quad (10)$$

となる。ここで、 $\mathbf{q}_k = [x_k, y_k, \theta_k]^T$ と定義する。物体の一般化座標は、 $\mathbf{q} = [\mathbf{q}_1^T, \mathbf{q}_2^T, \dots, \mathbf{q}_n^T]^T$ で与えられる。一般化座標は

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} - \begin{bmatrix} x_k \\ y_k \end{bmatrix} = \int_{\xi_k}^{\xi_{k+1}} \begin{bmatrix} \cos(u) \\ \sin(u) \end{bmatrix} du$$

を満たさなくてはならない。したがって、制約式

$$X_k = x_{k+1} - x_k - C_k(\xi_{k+1}), (k = 1, 2, \dots, n-1) \quad (11)$$

$$Y_k = y_{k+1} - y_k - S_k(\xi_{k+1}), (k = 1, 2, \dots, n-1) \quad (12)$$

の値は 0 でなくてはならない。

本報告では、微分幾何座標 θ_k と直交座標 x_k, y_k を用いて、線形ソフトロボットの变形と運動を定式化する。この手法を混合座標法と称する。

3 運動エネルギー

区間 $[\xi_k, \xi_{k+1}]$ における運動エネルギー T_k を計算する。線密度は一定であると仮定し、線密度を ρ で表す。区間 $[\xi_k, \xi_{k+1}]$ 内の点 $P(\xi)$ の速度は、 $\dot{\mathbf{q}}_k$ と $\dot{\mathbf{q}}_{k+1}$ の要素を用いて表されるので、式 (9) 式 (10) を用いると

$$\begin{aligned} & \int_{\xi_k}^{\xi_{k+1}} \{\dot{x}(\xi)^2 + \dot{y}(\xi)^2\} d\xi \\ &= \begin{bmatrix} \dot{\mathbf{q}}_k^T & \dot{\mathbf{q}}_{k+1}^T \end{bmatrix} \begin{bmatrix} H_{k,k} & H_{k,k+1} \\ H_{k+1,k} & \hat{H}_{k,k} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}_k \\ \dot{\mathbf{q}}_{k+1} \end{bmatrix} \end{aligned}$$

となる、ここで

$$\begin{aligned} H_{k,k} &= \begin{bmatrix} h & 0 & -\bar{S}_{k,k}^1 \\ 0 & h & \bar{C}_{k,k}^1 \\ -\bar{S}_{k,k}^1 & \bar{C}_{k,k}^1 & \bar{Q}_{k,k,k}^2 \end{bmatrix}, \\ H_{k,k+1} &= \begin{bmatrix} 0 & 0 & -\bar{S}_{k,k+1}^1 \\ 0 & 0 & \bar{C}_{k,k+1}^1 \\ 0 & 0 & \bar{Q}_{k,k,k+1}^2 \end{bmatrix}, \\ H_{k+1,k} &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -\bar{S}_{k,k+1}^1 & \bar{C}_{k,k+1}^1 & \bar{Q}_{k,k,k+1}^2 \end{bmatrix}, \\ \hat{H}_{k,k} &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \bar{Q}_{k,k+1,k+1}^2 \end{bmatrix} \end{aligned}$$

である。したがって、区間 $[\xi_k, \xi_{k+1}]$ における運動エネルギー T_k は

$$T_k = \frac{1}{2} \begin{bmatrix} \dot{\mathbf{q}}_k^T & \dot{\mathbf{q}}_{k+1}^T \end{bmatrix} J_k \begin{bmatrix} \dot{\mathbf{q}}_k \\ \dot{\mathbf{q}}_{k+1} \end{bmatrix} \quad (13)$$

と表すことができる。部分慣性行列 J_k は

$$J_k = \rho \begin{bmatrix} H_{k,k} & H_{k,k+1} \\ H_{k+1,k} & \hat{H}_{k,k} \end{bmatrix} \quad (14)$$

となる。

物体の運動エネルギーは

$$T = T_1 + T_2 + \dots + T_{n-1} = \frac{1}{2} \dot{\mathbf{q}}^T J \dot{\mathbf{q}} \quad (15)$$

で与えられる。慣性行列 J は三重ブロック帯行列

$$J = O_{3n \times 3n} \oplus J_1 \oplus J_2 \oplus \dots \oplus J_{n-1} \quad (16)$$

である。演算子 \oplus は、対応するブロックへの寄与を加算する演算を表す。すなわち、行列 J_k の 3×3 ブロック $\rho H_{k,k}$, $\rho H_{k,k+1}$, $\rho H_{k+1,k}$, $\rho \hat{H}_{k,k}$ を (k, k) , $(k, k+1)$, $(k+1, k)$, $(k+1, k+1)$ ブロックに加算する演算を表す。

4 ラグランジュの運動方程式

ラグランジアンは

$$\mathcal{L} = T - U + \sum_{k=1}^{n-1} (\lambda_k X_k + \mu_k Y_k) \quad (17)$$

で与えられる。ここで、 λ_k, μ_k は、制約 X_k, Y_k に対応するラグランジュの未定乗数である。ラグランジュの運動方程式は

$$\frac{\partial \mathcal{L}}{\partial \mathbf{q}} - \frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} = \mathbf{0} \quad (18)$$

となる。ポテンシャルエネルギー U と制約 X_k, Y_k は一般化速度 $\dot{\mathbf{q}}$ に依存しないので、上式は

$$\frac{\partial T}{\partial \mathbf{q}} - \frac{d}{dt} \frac{\partial T}{\partial \dot{\mathbf{q}}} - \frac{\partial U}{\partial \mathbf{q}} + \sum_{k=1}^{n-1} \left(\lambda_k \frac{\partial X_k}{\partial \mathbf{q}} + \mu_k \frac{\partial Y_k}{\partial \mathbf{q}} \right) = \mathbf{0} \quad (19)$$

と書くことができる。

運動エネルギーのラグランジュの運動方程式への寄与を求める。慣性行列 J は、 x_k, y_k に依存しないので

$$\frac{\partial T}{\partial x_k} = 0, \quad \frac{\partial T}{\partial y_k} = 0$$

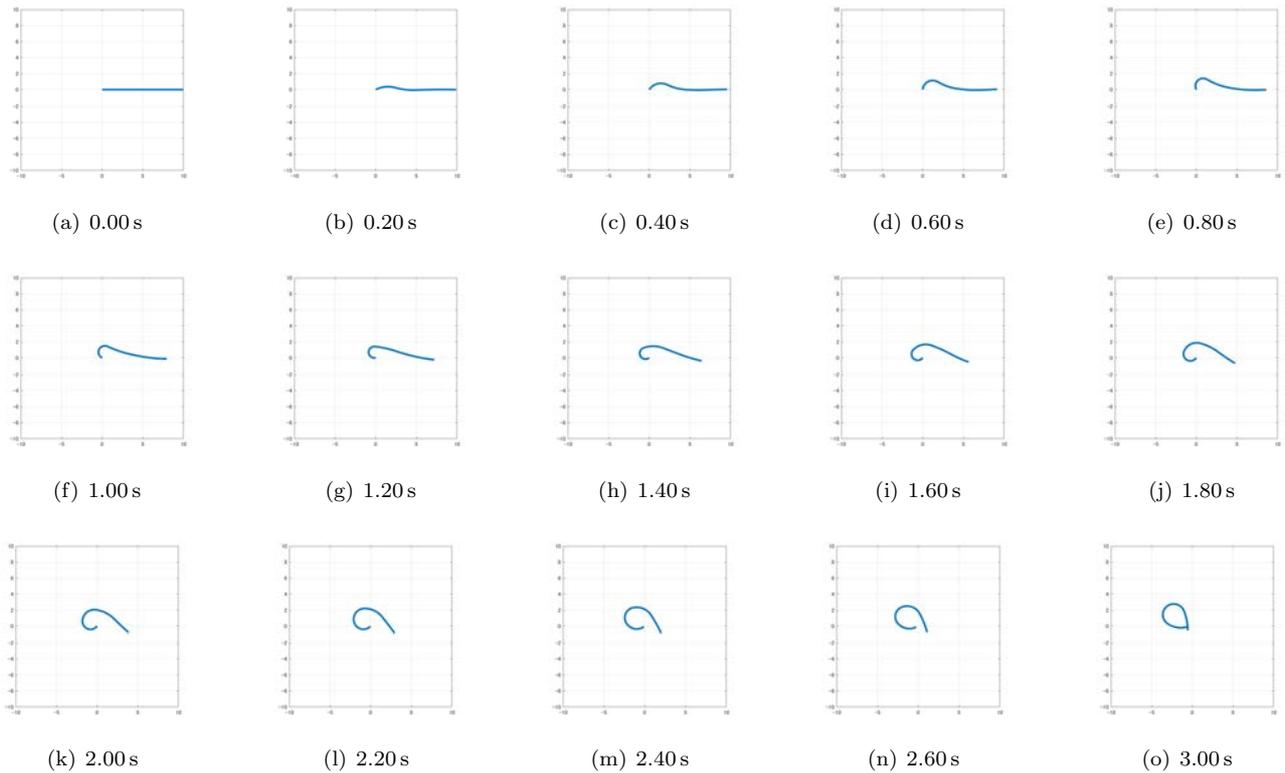


Fig.2 Simulation result

6 シミュレーション

式 (25) と式 (27) をまとめると

$$\begin{bmatrix} J & -A \\ -A^T & -A \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{f}_p \\ \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) \end{bmatrix} \quad (28)$$

となる。一般化座標 \mathbf{q} と一般化速度 $\dot{\mathbf{q}}$ の値が与えられると、左辺の行列と右辺のベクトルの値を計算することができる。したがって、上式を解くことにより、 $\ddot{\mathbf{q}}$ を求めることができる。この計算過程は常微分方程式の標準形であり、標準形を数値的に解くことにより、各時刻における \mathbf{q} と $\dot{\mathbf{q}}$ の値を求めることができる。

シミュレーションを通して線状ソフトロボットの变形と運動を求めた。全長 $L = 10$ 、線密度 $\rho = 0.1$ 、曲げ弾性 $R_{ela} = 1$ 、曲げ粘性 $R_{vis} = 2$ とおき、節点数を $n = 5$ として計算を進めた。線形曲げ弾性と線形曲げ粘性に起因する一般化力は、 $\mathbf{f}_p = -R_{ela}N\mathbf{q} - R_{vis}N\dot{\mathbf{q}}$ と表される。ここで、 $3n \times 3n$ 行列 N は、節点の座標により決まる。時間 $[0, 2]$ において、点 $P(0)$ の位置を固定する。これは、制約 $X_1 = 0, Y_1 = 0$ で与えられる。これらの制約式はまとめて、 $A_e^T \mathbf{q} = \mathbf{0}_2$ で与えられる。ここで

$$A_e^T = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \end{bmatrix}$$

である。時間 $[0, 1]$ において、点 $P(0)$ まわりに外部トルク $\tau = 50$ を加える。時間 $[1, 2]$ においては、このトルクを加えない。外力や外部トルクに対応する項を \mathbf{f}_e で表す。このとき、式 (28) は

$$\begin{bmatrix} J & -A & -A_e \\ -A^T & -A & -A_e \\ -A_e^T & -A_e & -A_e \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \lambda \\ \lambda_e \end{bmatrix} = \begin{bmatrix} \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{f}_p + \mathbf{f}_e \\ \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) \\ \mathbf{c}_e(\mathbf{q}, \dot{\mathbf{q}}) \end{bmatrix}$$

となる。項 $\mathbf{c}_e(\mathbf{q}, \dot{\mathbf{q}})$ は、追加した制約に対する制約安定化の式で、 $\ddot{\mathbf{q}}$ を含まない部分を表す。時間 $[2, 3]$ において、点 $P(0)$ の位置に対する制約を解放する。このとき、ソフトロボットの運動方程式は式 (28) で与えられる。図 2 にシミュレーション結果を示す。時刻 0 で線状ソフトロボットは自然状態にあり、点 $P(0)$ は原点

にある。時間 $[0, 1]$ で加えられる原点まわりのトルクで、線状ソフトロボットが变形する。このときの変形は、トルクの作用点 $P(0)$ の近くに限られる。時間 $[1, 2]$ で、ボディの右側が左側に移動し、結果としてボディ全体に曲げ変形が生じる。時間 $[2, 3]$ では、点 $P(0)$ における制約が解放されており、結果として線状ソフトロボットは变形しながら左向きに移動する。

シミュレーションは、Intel Core i5 2GHz、メモリ 8GB のノート PC 上の Matlab R2019b で実行した。計算時間は約 104 分であった。

7 結言

本報告では、混合座標法に基づき、線状ソフトロボットの運動方程式を定式化した。混合座標法を用いることにより、運動エネルギーの計算において生じる積分が簡単に表される。また、解析的に計算できる部分が多くなり、数値計算が単純になる。提案手法は、三次元空間内で運動する線状ソフトロボットの動的なモデリングに拡張できると考える。この拡張が今後の課題である。

謝辞

本研究は JSPS 科研費、新学術領域研究「ソフトロボット学」19H05337 の助成を受けた。

参考文献

- [1] H. Wakamatsu and S. Hirai, "Static Modeling of Linear Object Deformation Based on Differential Geometry", Int. J. Robotics Research, Vol. 23, No. 3, pp. 293–311, 2004.
- [2] H. Wakamatsu, K. Takahashi and S. Hirai, "Dynamic Modeling of Linear Object Deformation based on Differential Geometry Coordinates", Proc. IEEE Int. Conf. on Robotics and Automation, pp. 1028–1033, Barcelona, Spain, Apr. 18–22, 2005.