

卒業論文要旨

二段階移動による四足歩行ロボットのフォーメーション制御

Formation Control of Four-Legged Robots by Two-Step Movements

システム工学群

人工知能システム論研究室 1250123 中山 太翔

1. はじめに

フォーメーション制御とは、複数のロボットが互いの持つ情報を交換して自発的に行動し、目標のフォーメーションを実現する制御のことである。フォーメーション制御には様々な応用があり、ビークル群の協調作業や人口衛星のフォーメーションフライトが例として挙げられる⁽¹⁾⁽²⁾。

本研究では、フォーメーションを形成するロボットとして、「四足歩行ロボット」を考える。ここでいう四足歩行ロボットとは、四本の足によって前後、左右の移動と旋回ができるロボットである。四足歩行ロボットは、地形に合わせて足先の高さを変更できるため、車輪型のロボットに比べて平坦でない場所での移動が容易であり、屋外での作業を効率的に行えると考えられる。

こうした四足歩行ロボットに対し、離散値の入力を用いたフォーメーション制御器が提案された⁽³⁾。文献⁽³⁾によると、入力の量子化により、結果として得られるフォーメーションは目標フォーメーションとは一致せず、両者の間の誤差は、量子化によって生じるロボットの動きの誤差（量子化誤差という）に依存する。それゆえ、量子化誤差が小さいほど目標フォーメーションに近いものが実現できる。

そこで本研究では、量子化誤差を減らすための1つのアプローチとして「二段階移動」によるフォーメーション制御を考え、そのための制御器の開発を目的とする。二段階移動とは、従来の1回移動（図1(a)）とは異なり、各時刻でのロボットの目標位置に対して、2回の移動によりその位置へ向かう（図1(b)）ことである。これにより、1回移動の場合に比べて量子化誤差を減らすことができる。

記法：実数の集合を $\mathbb{R}$ 、正の実数の集合を $\mathbb{R}_+$ とする。ベクトルと行列の無限大ノルムを $\|\cdot\|$ 、ベクトルのユークリッドノルムを $\|\cdot\|_2$ で表す。ベクトル $x_1, x_2, \dots, x_n \in \mathbb{R}^2$ と集合 $\mathbb{I} = \{i_1, i_2, \dots, i_m\} \subseteq \{1, 2, \dots, n\}$ に対し、 $[x_i]_{i \in \mathbb{I}} = [x_{i_1}^T \ x_{i_2}^T \ \dots \ x_{i_m}^T]^T \in \mathbb{R}^{2m}$ と定義する。

2. 先行研究の概要

準備として、離散値入力を用いた四足歩行ロボットのフォーメーション制御⁽³⁾について簡単に紹介する。

n 台の四足歩行ロボットからなる、二次元空間上のマルチロボット系を考える。ロボット i ($i \in \{1, 2, \dots, n\}$) を以下のようにモデル化する。

$$\begin{bmatrix} x_i(t+1) \\ \theta_i(t+1) \end{bmatrix} = \begin{bmatrix} x_i(t) \\ \theta_i(t) \end{bmatrix} + \begin{bmatrix} f(\theta_i(t), u_i(t)) \\ u_{i2}(t) \end{bmatrix} \quad (1)$$

ここで、 $x_i(t) \in \mathbb{R}^2$ と $\theta_i(t) \in (-\pi, \pi]$ はそれぞれロボットの位置と回転角度を表し、 f は運動を記述する関数を表す。 $u_i(t) = [u_{i1}(t) \ u_{i2}(t) \ u_{i3}(t)]^T$ は制御入力であり、 $u_{i1}(t), u_{i2}(t) \in \mathbb{R}$ は

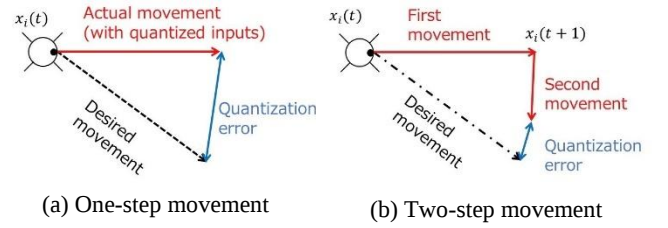


Fig. 1 Relation between the movements of a robot through quantized inputs and quantization errors

それぞれ直進、回転の速度に対応し、 $u_{i3}(t) \in \{0, 1\}$ は直進 ($u_{i3}(t) = 0$) と横移動 ($u_{i3}(t) = 1$) の切り替えを担う。ロボット i には以下の制御器が備えられているとする。

$$K_i: \begin{cases} \xi_i(t+1) = g_{i1} \left(\xi_i(t), [x_j(t) - x_i(t)]_{j \in \mathbb{N}_i}, \theta_i(t) \right) \\ u_i(t) = g_{i2} \left(\xi_i(t), [x_j(t) - x_i(t)]_{j \in \mathbb{N}_i}, \theta_i(t) \right) \end{cases} \quad (2)$$

ただし、 $\xi_i(t) \in \mathbb{R}^m$ は状態、 $[x_j(t) - x_i(t)]_{j \in \mathbb{N}_i}$ と $\theta_i(t)$ は入力、 $u_i(t)$ は出力、 g_{i1}, g_{i2} は制御器を定める関数である。集合 $\mathbb{N}_i \subset \{1, 2, \dots, n\}$ はロボット i が情報を得られるロボットの添え字集合を表す。また、出力 $u_{i1}(t), u_{i2}(t)$ は離散値に制限されると仮定する。具体的には、ステップ幅 $s \in \mathbb{R}_+$ に対し、 $u_{i1}(t) \in \{0, \pm s, \pm 2s, \dots\}$ 、 $u_{i2} \in \{0, \pm \pi/4, \pm \pi/2, \pm (3\pi)/4, \pi\}$ に制限されるとする。

ロボット間のネットワーク構造を時不変の有向グラフ $G = (\mathbb{V}, \mathbb{E})$ で表す。ここで、 $\mathbb{V} = \{1, 2, \dots, n\}$ は頂点集合、 $\mathbb{E}$ は辺集合であり、前者が n 台のロボットを、後者がロボット間の情報の流れを表す。このグラフ G を用いて、(2)式における集合 $\mathbb{N}_i = \mathbb{N}_i = \{j \in \mathbb{V} \mid (j, i) \in \mathbb{E}\}$ と定める。

ロボット j に対するロボット i の目標相対位置を $r_{ij} \in \mathbb{R}^2$ と表記する。このとき、以下の問題を考える。

問題 1：(1), (2)式からなるマルチロボット系に対し、ステップ幅 $s \in \mathbb{R}_+$ と目標相対位置 r_{ij} ($i, j = 1, 2, \dots, n$) が与えられるとする。すべての初期状態 $(x_i(0), \theta_i(0))$ ($i = 1, 2, \dots, n$) に対して、次式を満たす制御器 $K_1, K_2, \dots, K_n$ を求めよ。

$$\lim_{t \rightarrow \infty} (x_i(t) - x_j(t)) = r_{ij} \quad (3)$$

この問題では、入力 $u_{i1}(t), u_{i2}(t)$ ($i \in \mathbb{V}$) が離散値に制限されるため、(3)式を厳密に達成することはできない。それゆえ、(3)式を近似的に達成することを目指す。

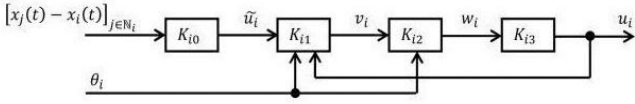


Fig. 2 Structure of the existing controller K_i

文献⁽³⁾では、問題 1 の解として、図 2 の制御器 K_i ($i \in \mathbb{V}$) が提案された。これは、 $K_{i0}, K_{i1}, K_{i2}, K_{i3}$ からなる。 K_{i0} は、全方向に移動できるロボットに対するフォーメーション制御器であり、以下の式で与えられる。

$$\tilde{u}_i(t) = -k \sum_{j \in \mathbb{N}_i} (x_i(t) - x_j(t) - r_{ij}) \quad (4)$$

ただし、 $\tilde{u}_i(t) \in \mathbb{R}^2$ は出力であり、ロボット i の時刻 t における (x_i の要素の方向の) 目標速度に対応する。また、 $k \in \mathbb{R}_+$ はゲインである。 K_{i1} は、入力が離散値に制限されることに伴う制御性能の低下を抑えるために、 K_{i0} の出力 $\tilde{u}_i(t)$ を $v_i(t) \in \mathbb{R}^2$ に修正する。 K_{i2} は、 $v_i(t)$ からロボット i の直進と回転方向の目標速度を求め、それを $w_i(t) \in \mathbb{R}^2$ として出力する。 K_{i3} は、 $w_i(t)$ に基づき、量子化器を用いて離散値の $u_i(t)$ を決定する。

3. 二段階移動への拡張

図 2 の制御器を、ロボットが二段階移動を行うように拡張する。文献⁽³⁾では、量子化誤差を抑制するために、図 2 の K_{i2} に飽和関数を導入して出力 $v_i(t)$ の大きさを制限していた。以下ではまず、二段階移動を考える動機付けとして、この飽和関数を用いないという仮定のもとで、 $v_i(t)$ と量子化誤差の関係性を明らかにする。

図 2 の K_{i3} での量子化による、目標速度 $v_i(t)$ とロボット i の (実際の) 速度の間の誤差を量子化誤差 $e_i(t) \in \mathbb{R}^2$ とする。このとき、 $\|v_i(t)\|$ と $\|e_i(t)\|$ に関して、つぎの結果を得た。

定理 1 : (1) 式と図 2 の制御器からなるフィードバック系に対し、ステップ幅 $s \in \mathbb{R}_+$ が与えられ、 K_{i2} における飽和関数はないものと仮定する。任意の $i \in \mathbb{V}$ 、 $t \in \{0, 1, \dots\}$ に対し、

$$\|e_i(t)\| \geq \sqrt{\|v_i(t)\| \left(\|v_i(t)\| - \frac{s}{2} \right) (1 - \cos \phi_i(t))} \quad (5)$$

が成り立つ。ここで、 $\phi_i(t) \in \mathbb{R}$ はベクトル $v_i(t)$ と $v_i(t) + e_i(t)$ のなす角である。

(5) 式において、任意の $i \in \mathbb{V}$ と $t \in \{0, 1, \dots\}$ に対し、 $1 - \cos \phi_i(t) \geq 0$ であることが示せる。また、ロボット i の 1 時刻での移動距離は目標速度の大きさ $\|v_i(t)\|$ に比例する。よって、定理 1 は $\|v_i(t)\| > s/2$ のとき、ロボット i の 1 時刻での移動距離が増加するほど量子化誤差も増加することを意味する。

ロボットの移動距離の増加に伴う量子化誤差の増加を抑えるために、図 2 の制御器において、 K_{i1} と K_{i2} を次式のように変更したものを提案する。

$$K_{i1}: \begin{cases} \xi_i(t+1) = e_i(t) \\ v_i(t) = \begin{cases} -\xi_i(t) & \text{if } t \in \{1, 3, \dots\}, \|e_i(t-1)\|_2 > \varepsilon \\ -\xi_i(t) + \tilde{u}_i(t) & \text{otherwise} \end{cases} \end{cases} \quad (6)$$

$$K_{i2}: w_i(t) = \begin{cases} [0 \ 0]^T & \text{if } t \in \{1, 3, \dots\}, \|e_i(t-1)\|_2 \leq \varepsilon \\ \begin{bmatrix} \|v_i(t)\|_2 \\ \arctan 2(v_{i2}(t), v_{i1}(t)) - \theta_i(t) \end{bmatrix} & \text{otherwise} \end{cases} \quad (7)$$

ただし、 $\varepsilon \in \mathbb{R}_+$ はある閾値、 $v_{i1}(t), v_{i2}(t)$ はそれぞれ $v_i(t)$ の第 1, 第 2 要素、 $\arctan 2$ は 4 象限の逆正接関数である。

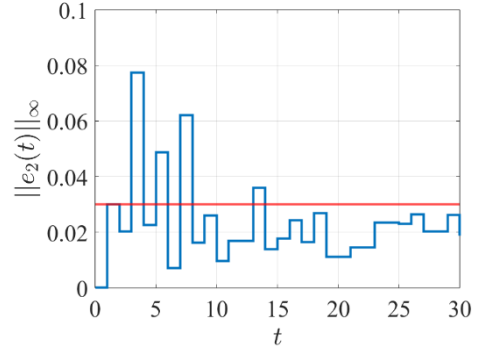


Fig. 3 Time evolution of $\|e_2(t)\|$

提案する制御器 K_i を用いたときのロボット i の動作は以下ようになる。まず、時刻 $t \in \{0, 2, \dots\}$ では、図 2 の従来の制御器を用いたときと同様に移動する。一方、時刻 $t \in \{1, 3, \dots\}$ では、(6), (7) 式より、前の時刻での量子化誤差の大きさ $\|e_i(t-1)\|_2$ が閾値 ε を超えた場合、 $-\xi_i(t) = -e_i(t-1)$ を目標速度 $v_i(t)$ として移動し、そうでない場合、停止する。こうした動作により、 $\|e_i(t)\|_2$ を ε 以下に抑えられると期待される。

提案制御器の性能に関して、以下の結果を得た。

定理 2 : (1), (6), (7) 式と図 2 の K_{i0}, K_{i3} からなるフィードバック系に対し、ステップ幅 $s \in \mathbb{R}_+$ が与えられるとする。ある $i \in \mathbb{V}$ と $t \in \{1, 3, \dots\}$ に対し、 $\|e_i(t)\|_2 > \varepsilon$ か $\|e_i(t)\| \gg s/2$ ならば、以下の不等式が成り立つ。

$$\|e_i(t+1)\| \leq 2\sqrt{1 - \cos\left(\frac{\pi}{8}\right)} \|e_i(t)\| \quad (8)$$

定理 2 は、 $e_i(t)$ と $e_i(t+1)$ 、すなわち 2 回目の移動の前後における量子化誤差の関係性を示している。 $e_i(t)$ が 2 回目の移動前の量子化誤差であることと $2\sqrt{1 - \cos(\pi/8)} \approx 0.55$ であることから、定理 2 は、 $\|e_i(t)\| \gg s/2$ のとき、2 回目の移動によって、量子化誤差を約 45% 以上減らせることを意味する。

以上の結果をシミュレーションにより検証する。提案制御器を用いてあるフォーメーション制御を行い、一例として、そのときの $\|e_2(t)\|$ の時間変化をプロットしたものを図 3 に示す。ここで、赤線は閾値 ε を表す。図を見ると、 $\|e_2(t)\|_2 \geq \|e_2(t)\| > \varepsilon$ となった後は量子化誤差が減少していることがわかる。また、時刻 $t = 3, 5$ からの $\|e_2(t)\|$ の減少率を計算すると、 $t = 3$ から 4 では約 71%、 $t = 5$ から 6 では約 86% の減少が確認できた。これは、定理 2 の妥当性を示している。

4. おわりに

本研究では、二段階移動による四足歩行ロボットのフォーメーション制御について検討した。量子化誤差がある閾値を超えたときに 2 回目の移動を行うような制御器を提案し、その性能を解析した。解析の結果、提案制御器を用いると、ある条件下では二段階移動を行わない場合に比べて量子化誤差をおよそ 45% 以上減らせることが示された。

参考文献

- (1) 石井. マルチエージェントシステムの制御—I 総論, システム/制御/情報, Vol. 57, No. 5, pp. 211–218, 2013.
- (2) 桜間. マルチエージェントシステムの制御—III 合意制御(1), システム/制御/情報, Vol. 57, No. 9, pp. 386–396, 2013.
- (3) S. Izumi, K. Shinagawa, X. Xin, and T. Yamasaki. Formation Control of Four-Legged Robots using Discrete-Valued Inputs, *IEEE Control Systems Letters*, Vol. 6, pp. 1088–1093, 2022.