ファジィ理論と最適化手法によるＬＱＲ制御の重み行列決定手法

○市原　秀晃[*1]　入江　寿弘[*2]
新宮　清志[*3]

キーワード：多目的数値最適化、線形制御、最適レギュレータ、ロボットアーム

１．はじめに

我々は障害者の自立援助を目的に自分の手が届かない範囲での軽い作業が可能なロボットアームの研究を行ってきた。その制御系にLQI制御を採用したが、制御性の向上を図る為に重み行列の最適化が必要であると感じ、文献1）によるファジィ理論を用いたLQR制御系の重み行列の決定方法を検討した。しかしながら、探索精度的にも時間的にも最良の結果が得られていなかった。そこで高次元でも解探索精度が優れる最適化手法であるArtificial Bee Colony（ABC）アルゴリズムを用いることで、解探索の探索範囲の拡大、探索精度向上、時間短縮を狙う旨の研究を行い、一定の成果を得た[2)]。しかしながら、前研究段階ではABC以外の最適化手法を採用した場合での本アルゴリズム評価があまり行えていなかった。これを踏まえ、本研究は別の最適化手法（PSOを用いる）とABCとを比較、評価し、ABCの探索性能をより詳しく見ていくものである。2、3、4章を通してファジィ理論とABCを用いたLQR制御の重み行列決定手法についての解説を行い、5章にて本アルゴリズムを適用するロボットアームのモデルを示す。6章ではABCとPSOを比較する実験方法、7章にその結果を示す。

２．LQI 制御系

ロボットアームの制御系に用いたLQI制御の基本原理を示す。まず、n次元、m入力、p出力、可制御、可観測な線形時不変システム

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \end{aligned} \tag{1}$$

を用いる。目標値rとの誤差を積分器に入力したときの出力x_rを状態変数として加えた拡大方程は式(2)のようになる。

$$\dot{x}_a = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} x_a + \begin{bmatrix} B \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ I \end{bmatrix} r(t) \tag{2}$$

評価関数、制御入力はそれぞれ式(3)、(4)となる。

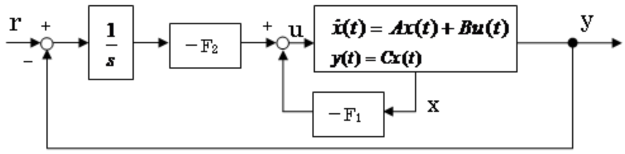
$$J = \int_0^\infty \left[x_a^T(t) Q x_a(t) + u^T(t) R u(t) \right] dt \tag{3}$$

$$u(t) = \begin{bmatrix} -F_1 & -F_2 \end{bmatrix} \begin{bmatrix} x^T & x_r^T \end{bmatrix}^T \tag{4}$$

$$PA^\# + A^{\#T} P - PB^\# R^{-1} B^{\#T} P + Q = 0 \tag{5}$$

$$A^\# = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix}, \qquad B^\# = \begin{bmatrix} B \\ 0 \end{bmatrix} \tag{6}$$

よって$A^\#$、$B^\#$について重み行列 Q、R を決定し、式(5)のリカッチ方程式を解けば良い。LQI 制御系は外乱に強く、ロバストなサーボ系が設計可能であり、携行型ロボットアームの制御系に採用している。図−1 に LQI 制御系のブロック図を示す。



図−１　LQI 制御系のブロック図[2)]

３．ファジィ理論による重み行列の決定

制御応答を評価基準にするために、制御応答形状をファジィ理論によって定式化していく。制御系の設計者が制御応答において特に注目を置くのは、行き過ぎ量、応答時間、整定時間、制御入力の上、下限であると考えられる。よって、これらをファジィ化する。以降、添字i、jは出力番号、入力番号を意味する。今回、メンバシップ関数の形状は全て同じものを使用し、図−2 のようなものとした。

（１）行き過ぎ量

ピークオーバーシュートM_p^iに関するメンバシップ関数を、式(7)のように定義する。$d_i(M_p)$は右スプレッドを表わし、開始時間から整定時間までの応答によって、メンバシップ値を決定する。

メンバシップ関数の横軸は$y_i(t) - M_p^i$である。

$$\mu M_p^i(y_i) = \begin{cases} 1, & y_i(t) - M_p^i \le 0 \\ 1 - \dfrac{y_i(t) - M_p^i}{d_i(M_p)}, & 0 \le y_i(t) - M_p^i \le d_i(M_p) \\ 0, & y_i(t) - M_p^i \ge d_i(M_p) \end{cases} \tag{7}$$

（２）応答時間

応答時間T_r^iに関するメンバシップ関数を、式(8)のように定義する。ここで、RT_iは応答$y_i(t)$が目標値の95%に最初に達する時間、$d_i(T_r)$は右スプレッドを意味する。文献1)では、三角型メンバシップ関数を用いているが、設定した応答時間T_r^iよりも応答速度の早い解はメンバシップ値が0になってしまう。よって本研究では、要求よりも応答の早い解についても許容させるため、図-2のような形とした。メンバシップ関数の横軸は$RT_i - T_r^i$である。

$$\mu T_r^i(RT_i) = \begin{cases} 1, & RT_i - T_r^i \le 0 \\ 1 - \dfrac{RT_i - T_r^i}{d_i(T_r)}, & 0 \le RT_i - T_r^i \le d_i(T_r) \\ 0, & RT_i - T_r^i \ge d_i(T_r) \end{cases} \tag{8}$$

（３）整定時間

整定時間T_s^iに関するメンバシップ関数を、式(9)のように定義する。ここで、ST_iは応答$y_i(t)$が目標値の±5%以内に収まる最小時間、$d_i(T_s)$は右スプレッドを意味する。こちらも応答時間と同様の理由により、図-2のような形とした。メンバシップ関数の横軸は$ST_i - T_s^i$である。

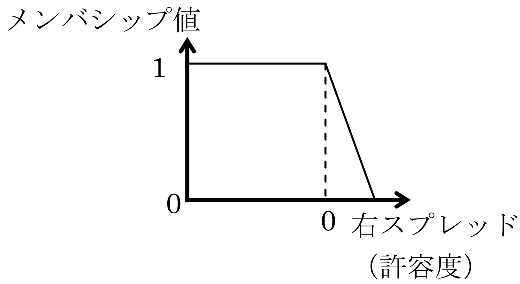
$$\mu T_s^i(ST_i) = \begin{cases} 1, & ST_i - T_s^i \le 0 \\ 1 - \dfrac{ST_i - T_s^i}{d_i(T_s)}, & 0 \le ST_i - T_s^i \le d_i(T_s) \\ 0, & ST_i - T_s^i \ge d_i(T_s) \end{cases} \tag{9}$$

（４）制御入力

制御入力の下限$\underline{\beta}_j$、上限$\overline{\beta}_j$に関するメンバシップ関数を式(10)、(11)に示す。ここで$d_j(\underline{\beta})$、$d_j(\overline{\beta})$は入力の下限、上限に対する右スプレッドを意味する。メンバシップ関数の横軸はそれぞれ$\underline{\beta}_j - u_j(t)$、$u_j(t) - \overline{\beta}_j$である。ここまでが制御応答と制御入力に対してのファジィ化の流れである。以降は重み行列についてのファジィ化を行う。

$$\mu\underline{\beta}_j(u_j) = \begin{cases} 1, & \underline{\beta}_j - u_j(t) \le 0 \\ 1 - \dfrac{\underline{\beta}_j - u_j(t)}{d_i(\underline{\beta})}, & 0 \le \underline{\beta}_j - u_j(t) \le d_j(\underline{\beta}) \\ 0, & \underline{\beta}_j - u_j(t) \ge d_j(\underline{\beta}) \end{cases} \tag{10}$$

$$\mu\overline{\beta}_j(u_j) = \begin{cases} 1, & u_j(t) - \overline{\beta}_j \le 0 \\ 1 - \dfrac{u_j(t) - \overline{\beta}_j}{d_i(\overline{\beta})}, & 0 \le u_j(t) - \overline{\beta}_j \le d_j(\overline{\beta}) \\ 0, & u_j(t) - \overline{\beta}_j \ge d_j(\overline{\beta}) \end{cases} \tag{11}$$



図−２　メンバシップ関数

（５）最適重み行列探索のための評価関数

解探索を行うために、式(7)、(8)、(9)、(10)、(11)より評価関数を式(12)のように定義する。

$$\begin{aligned} \min\ & f(x) \\ &= (3p+2m) - \{\mu M_p^1(y_1) + \cdots + \mu M_p^p(y_p) \\ &\quad + \mu T_r^1(RT_1) + \cdots + \mu T_r^p(RT_p) + \mu T_s^1(ST_1) \\ &\quad + \cdots + \mu T_s^p(ST_p) + \mu\underline{\beta_1}(u_1) + \cdots + \mu\underline{\beta_m}(u_m) \\ &\quad + \mu\overline{\beta_1}(u_1) + \cdots + \mu\overline{\beta_m}(u_m)\} \end{aligned} \tag{12}$$

評価関数の最良値を0とするために、右辺第一項からメンバシップ値を引く形に定義した。p出力には行き過ぎ量、応答、整定時間の3つ、m入力には上限、下限の2つが関係しているので、第一項のような形とした。例えば今回適用するロボットアームの場合は3出力2入力であるので、第一項は10となる。設計者の要望を満足する制御応答形状が得られると第二項は10に近づいていき、評価値は最小で0となる。

４．ABCアルゴリズム

ABCアルゴリズム[4]はミツバチの餌場探索を元にしたメタヒューリスティクスである。PSOやDEなどの最適化手法と比べ、局所解に陥りにくく、高次元多峰性関数の最適化に強いとされている。ベンチマークテスト等でも、優れた結果が得られている[5]。ABCアルゴリズムではミツバチの巣の中には「Employed bee」、「Onlooker bee」、「Scout bee」の3つのグループがあると定義している。また餌場は最適化問題の変数を、蜜の量は解の適合度を表している。ABCアルゴリズムによる最適化の手順は次の①～④で、これを指定回数繰り返して最適化を行う。その流れを図-3に示す。

①パラメータの初期化および、ランダムに探索点を決定する。

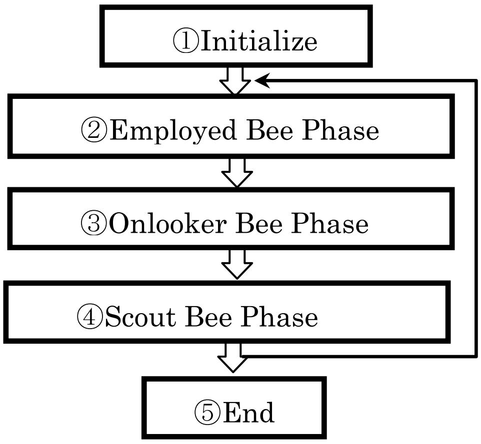
②Employed beeが他のミツバチの持つ餌場の情報を元に周辺により良い餌場がないか探索する。

③Onlooker beeがルーレット選択しEmployed beeに

蜜の量の多い餌場を優先して探索を行わせる

④周辺にそれ以上良い餌場が見つからなくなったらその餌場周辺の探索を止め、Scout bee がランダムに新しい餌場を探して記憶する。

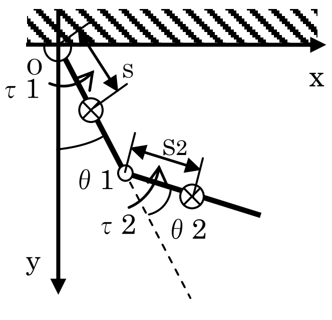
また、アルゴリズムの詳細についてはここでは省くが、文献 4）や 5）などで紹介されている。



図−３　ABC アルゴリズムの最適化手順 2)

５．実機への適用

アームの状態方程式を導出する。図−4 のようなアームを簡略化したものを考える。ここで、アームの各リンクは、質量m_1、m_2と重心まわりの慣性モーメントI_1、I_2、回転軸からリンクの重心までの距離S_1、S_2、全長l_1、l_2を持つ。ラグランジュの運動方程式を用いて、この系の線形化した状態方程式（直線状）を求め、製作した実機の各パラメータを代入したものを以下に示す。



図−４　２リンクアームモデル

$$A=\begin{pmatrix}0 & 1 & 0 & 0\\ -79 & -3.3 & 73 & 7.2\\ 0 & 0 & 0 & 1\\ 91 & 7.2 & -244 & -20\end{pmatrix},B=\begin{pmatrix}0 & 0\\ 326 & -720\\ 0 & 0\\ -720 & 2033\end{pmatrix} \qquad (13)$$

$$C=\begin{pmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\end{pmatrix},x=\begin{bmatrix}\theta_1 & \dot{\theta}_1 & \theta_2 & \dot{\theta}_2\end{bmatrix}^T$$

この状態方程式（13）を用いて、ファジィ理論を用いた制御系の設計を行う。

６．実験手順

ABC と PSO を比較するためのパラメータ条件は文献 4）を参照し決定している。ABC についての詳しいパラメータは Employed bee＝50、Onlooker bee＝50、limit=Ns*D＝300、Iterations＝5000 としている。また、PSO については、population size＝100、ω＝0.6、ϕ_1、ϕ_2＝1.8、Iterations＝5000、v_{max}は探索領域の上限から下限までの距離の 50%で、$v_{min}=-v_{max}$としている。本研究では重み行列 R は 0.01 で固定することとし、Q のみ 1〜10000 の探索範囲で最適化を行う。これは Q と R でトレードオフの関係があるためである。この条件で各 10 回最適化を行い、最適化手法の評価を行っていく。実機のリンク目標角度は共に 0.1745[rad]とし、モータへの電圧入力の上限下限は±12[V]とする。要求する制御応答形状のパラメータとファジィ理論に用いる許容度のパラメータは表−1、2 に示す。

表−１　ピークオーバーシュートと応答時間、整定時間の中心と許容度

	M_p^1	M_p^2	T_r^1	T_r^2	T_s^1	T_s^2
中心	0.1745	0.1745	0.5	0.5	1.0	1.0
許容度	0.02	0.02	0.5	0.5	0.5	0.5

表−２　入力上限、下限の中心、許容度

	$\underline{\beta_1}$	$\overline{\beta_1}$	$\underline{\beta_2}$	$\overline{\beta_2}$
中心	-12	12	-12	12
許容度	2	2	2	2

７．実験結果、考察

まずは ABC、PSO を用いた最適化中の評価関数値推移を図−5 に示す。比較の為に Iterations が 300 回になった時点までの結果で表記している。図−5 では PSO の方が最適解を短時間で探索できていることが分かる。前研究段階では PSO のパラメータの調整が満足に行われておらず、ABC と比較すると探索精度が劣っていたが、パラメータ調整後は飛躍的に探索精度が向上された。加えて、本アルゴリズムの場合次元数は 6 であり、高次元の最適化に強い ABC を用いずとも、PSO の最適化で十分な効果が得られていると考えられる。また、本実験ではオリジナルの ABC を使用し最適化を行っているが、文献 5）などでは改良型の ABC も発表されている。よって、改良型の ABC を用いることにより PSO よりも優れた結果が得られる可能性もあり、今後の課題となっている。ここで、PSO のv_{max}の数値を文献 6）を参照し、探索領域の上限から下限までの距離の 15%に変更し、最適化を行った。その結果を図-6 に示す。速度を 50%に設定した場合と比較して、15%の場合には局所解に陥り、探索性能が極端に落ちていることがわかる。本アルゴリズムでは LQR の制御対象が変

更された場合や制御系設計者の制御応答形状の要求により評価関数の特性などが逐次変化していくので、それに応じて、PSO のパラメータを調整する必要性があると考えられる。

PSO で求められた最適解の一つを表-3 に示す。この重み行列より得たステップ応答を図-7、8 に示す。リンク 1、2 共にオーバーシュートもなく、応答、整定時間も要求を満たしている。

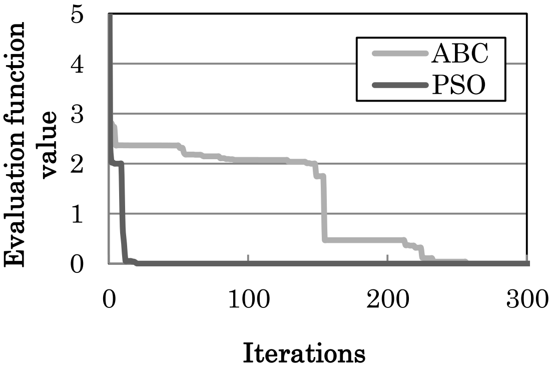


図-５　評価関数値の推移比較（ABC－PSO）

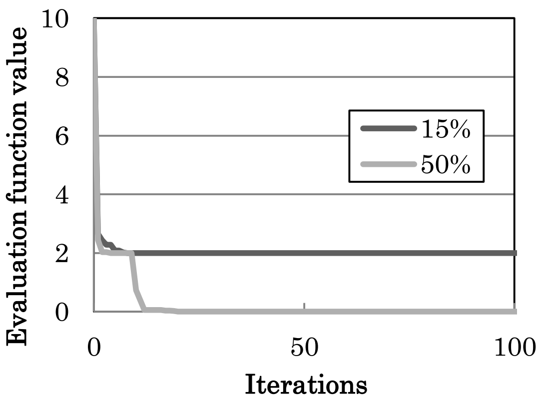


図-６　評価関数値の推移比較
［PSO; $v_{max}(15\% - 50\%)$］

８．まとめ

本アルゴリズムにおける最適化手法は PSO で十分な効果が得られている。但し、評価関数各々で PSO のパラメータ設定を行う必要がある。一方 ABC アルゴリズムであれば設定するパラメータは蜂の数と反復回数程度なので、評価関数の特性に依らない最適化が行えると考えられる。

また、今後は評価関数の変化と PSO の探索精度の相関関係を明らかにすることが課題になる。

表-３　設計結果（PSO）

q_{11}	q_{22}	q_{33}	q_{44}	q_{55}	q_{66}
187.7	1	198.6	1	8741.3	10000

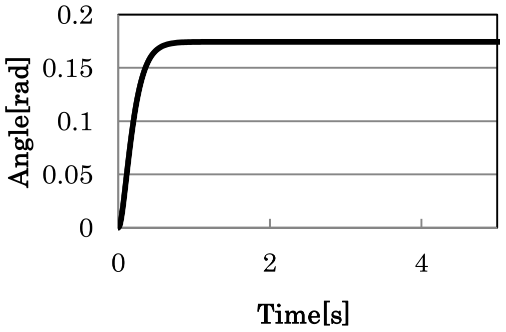


図-７　リンク１のステップ応答

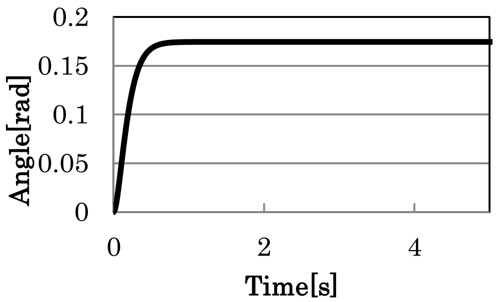


図-８　リンク２のステップ応答

［参考文献］

1)　荻野剛二郎・大林徳二　「LQ レギュレータのファジィ理論による設計法」　電子情報通信学会論文誌　A74 巻 7 号　pp.1023-1030　1991

2)　市原秀晃・入江寿弘・新宮清志　「ファジィ理論と ABC アルゴリズムによる LQI 制御系設計-携行型ロボットアームの制御-」　計算工学講演会論文集　vol.17　ROMBUNNO.F-7-4　2012

3)　野波健蔵・西村秀和・平田光男　「Matlabによる制御系設計」　東京電機大学出版局　pp. 96-97　1998

4)　Dervis Karaboga・Bahriye Akay「A comparative study of Artificial Bee Colony algorithm」Applied Mathematics and Computation　Vol.3　pp.1980-1987　2009

5)　宇谷明秀　「ABCアルゴリズムによる高次元多峰性関数の最適化」　日本建築学会シンポジウムソフトコンピューティングの最前線講演論文集　pp.11-20　2011

6)　飯村伊智郎・中山茂　「高次元関数最適化における Artificial Bee Colonyアルゴリズムの探索性能評価」　システム制御情報学会論文誌　vol.24　No.4　pp.97-99　2011

*1　日本大学大学院理工学研究科精密機械工学専攻
*2　日本大学理工学部精密機械工学科　教授
*3　日本大学理工学部海洋建築工学科　特任教授