

# ハイブリッドオートマトンを用いた ヒューマノイドロボットの周期的動作の生成

高橋秀行 (大阪大学), 小林啓吾 (九州工業大学), 潮俊光 (大阪大学)

## Periodic Motion Generation for Humanoid Robot with Hybrid Automata

\*Hideyuki Takahasi(Osaka University) and Keigo Kobayashi(Kyushu Institute of Technology)  
and Toshimitsu Ushio (Osaka University)

**Abstract** A method for reconstruction of periodic motions based on Lyapunov functions has been proposed. In this paper, we propose a hybrid state net which is an extension of the state net, in order to perform smooth transitions between periodic motions such as a walk and a squat so that we realize a complicated action series. This technique is mounted in Humanoid Robot HOAP-1 and the validity is verified.

**Key words** Motion generation, Humanoid Robot, Hybrid Automaton

### 1 はじめに

これまでに多くのヒューマノイドロボットの行動生成に関する研究が行われて来た [1][2]. 歩行などの安定した周期軌道を非線形力学系のリミットサイクルとしてとらえ, ヒューマノイドロボットの周期的な行動パターンを生成する研究として, 足立 [3][5] は周期的な行動パターンの低次元化された時系列データを, リアプノフ関数を用いて, 3次元空間内における閉曲線として表現する方法を提案している.

一方, ヒューマノイドロボットの動作間の遷移を表した図式表現として状態ネットが提案されている [6]. 状態ネットは静止した姿勢間の遷移関係を表しており, 周期的動作を表現するには適していない.

本稿では, 歩行やスクワットなどの周期的な行動間で滑らかな遷移をおこない, 組み合わせにより複雑な行動系列を実現するために, ステートネットを拡張した, ハイブリッドステートネットを提案し, それに基づく行動系列生成法を提案する. 実際に本手法を HOAP-1 に実装し, その有効性を検証する.

### 2 行動生成

本章では周期的な行動パターンから非線形力学系を作成し, 歩行やスクワットのような周期的な行動を生成する手法 [1][2][3] を紹介する. ヒューマノイドロボットの行動パターンの時系列データ  $M \in R^{n \times m}$  を元に周期的な行動を生成する運動系を作成する. 元となるデータ  $M$  は行動パターンの 1 周期分の時系列データを時間間

隔  $\Delta t(s)$  で記録したものである.

$$M = \begin{bmatrix} \theta_1[1] & \theta_1[2] & \cdots & \theta_1[m] \\ \theta_2[1] & \theta_2[2] & \cdots & \theta_2[m] \\ \vdots & \vdots & & \vdots \\ \theta_n[1] & \theta_n[2] & \cdots & \theta_n[m] \end{bmatrix} \quad (1)$$

$M$  の列数  $m$  と周期  $T(s)$  との間には  $(m+1)\Delta t = T$  の関係が成り立つ. ここで  $\theta_i[j]$  は  $i$  番目の関節の時刻  $(j-1)\Delta t(s)$  のセンサ情報を表す. ヒューマノイドロボットの自由度は大きくこのままでは  $M$  の次元が大きいため, 次のように特異値分解により低次元化を行う.

$$M = USV^T, \quad (2)$$

$$U = [U_1|U_2] \in R^{n \times n}, \quad (3)$$

$$S = \begin{bmatrix} S_1 & 0 \\ 0 & S_2 \end{bmatrix} \in R^{n \times m}, \quad (4)$$

$$S_1 = \text{diag}\{s_1, s_2, \dots, s_r\} \in R^{r \times r}, \quad (5)$$

$$S_2 = \text{diag}\{s_{r+1}, s_{r+2}, \dots, s_n\} \in R^{(n-r) \times (m-r)}, \quad (6)$$

$$V^T = \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \in R^{m \times m}, \quad (7)$$

$s_r \gg s_{r+1}$  なら

$$M \approx U_1 S_1 V_1^T \quad (8)$$

と表すことができる. ここで  $U_1 \in R^{n \times r}$ ,  $S_1 \in R^{r \times r}$ ,  $V_1^T \in R^{r \times m}$  であり,  $n \times m$  のデータ  $M$  から  $r \times m$  に低次元化を行ったデータ  $V_1^T$  が得られる. また  $U_1 S_1$  は  $m$  次元のベクトルから  $n$  次元のセンサ情報に復元する復元関数となる. 次にこの低次元化したデータ  $V_1^T$  にフィッティングする閉曲線を求める. このとき本稿では [5]

で提案した手法を用いる．まず関数  $f(t)$  をフーリエ級数を用いて導出する．なお，ここでは十分大きな整数  $l$  を用いて近似を行っている．

$$f(t) = \sum_{k=0}^{\infty} \{a_k \cos(k\omega t) + b_k \sin(k\omega t)\} \quad (9)$$

$$\sum_{k=0}^l \{a_k \cos(k\omega t) + b_k \sin(k\omega t)\}$$

そして低次元データ  $V^T$  から最小二乗法を用いて係数  $a_k, b_k (k = 0, \dots, l)$  の推定値は

$$\Lambda = (\Phi^T \Phi)^{-1} \Phi^T V_1 \quad (10)$$

$$= [a_0, a_1, \dots, a_l, b_1, \dots, b_l]^T$$

となる．但し  $\Phi$  は

$$\Phi = \begin{bmatrix} 1 & \cos(t_1) & \dots & \cos(lt_1) & \sin(t_1) & \dots & \sin(lt_1) \\ 1 & \cos(t_2) & \dots & \cos(lt_2) & \sin(t_2) & \dots & \sin(lt_2) \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 1 & \cos(t_m) & \dots & \cos(lt_m) & \sin(t_m) & \dots & \sin(lt_m) \end{bmatrix} \quad (11)$$

であり， $t_j = \frac{(j-1)2\pi}{m}$  とする．最後にチェビシェフ多項式を用いて以下のように関数  $F_1, F_2$  を定める．

$$f(t) = \sum_{k=0}^l \{a_k \cos(k\omega t) + b_k \sin(k\omega t)\} \quad (12)$$

$$= \sum_{k=0}^l a_k T_k(\cos(\omega t)) + \sin(\omega t) \sum_{k=1}^l b_k U_{k-1}(\cos(\omega t))$$

$$\equiv F_1(\cos(\omega t)) + \sin(\omega t) F_2(\cos(\omega t))$$

さらに，この周期関数  $f(t)$  の位相を表す変数  $x = [x_1 \ x_2]^T = [\cos(\omega t) \ \sin(\omega t)]^T$  と  $r$  次元空間での座標を表す  $r$  次元ベクトル  $y$  からなる  $(r+2)$  次元ベクトル  $X = [x^T \ y^T]^T$  に関する次の微分方程式で記述される運動系を作成する [5]．

$$\dot{X} = \omega \begin{bmatrix} 0 & 1 \\ -1 & 0 \\ \frac{-\partial F_1(x_2)}{\partial x_2} - x_1 \frac{\partial F_2(x_2)}{\partial x_2} & F_2(x_2) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \alpha \begin{bmatrix} x_1(1-x_1^2-x_2^2) \\ x_2(1-x_1^2-x_2^2) \\ F_1(x_2) + x_1 F_2(x_2) - y \end{bmatrix} \quad (13)$$

$$\equiv f(X) + g(X)$$

ここで  $\alpha > 0$  は収束率であり，このとき次のリアプノフ関数  $V(x)$  が存在し，この運動系は式 (12) の安定なリミットサイクルに収束する．

$$V(x) = \begin{bmatrix} x_1^2 + x_2^2 - 1 \\ F_1(x_2) + x_1 F_2(x_2) - y \end{bmatrix} \quad (14)$$

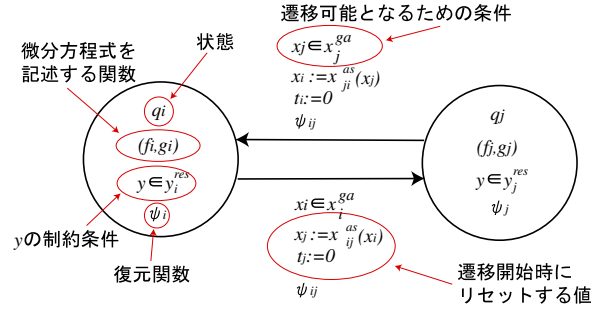


図 1: ハイブリッドステートネット

### 3 ハイブリッドステートネット

周期的な動作間 (静止状態も含む) の遷移が可能であるときに，その遷移を図式に表す方法としてハイブリッドステートネットを提案する．形式的にはハイブリッドステートネットは  $(Q, X, D, R, L_t, \Phi, A, G, A_s)$  の 9 項で記述できる．ハイブリッドステートネットの各項目を表 1 に示す．

Q	状態を表す集合
X	変数の集合
D	微分方程式を記述する ベクトル場の組 $(f, g)$ の集合
R	各状態での $y$ の制約条件の集合
$L_T$	各状態のローカルな時間の集合
$\Phi$	各状態，アークでの復元関数の集合
A	状態間のアークの集合
G	各アークの遷移可能な位相条件の集合
$A_s$	遷移時に次の状態の位相を求める関数の集合

表 1: ハイブリッドステートネットの項目

$Q$  は状態を表す記号の集合であり  $Q = \{q_1, q_2, \dots\}$  のように定め，各状態はそれぞれロボットの周期的な行動パターンの名称に対応している． $X = \{X_1, X_2, \dots\}$  は変数の有限集合であり， $q_i$  の状態  $X_i = [x_i^T \ y^T]^T$  とする． $x_i = [x_{i1} \ x_{i2}]^T$  であり，状態  $q_i$  の位相を表す変数である．また  $y$  は  $r$  次元空間での座標を表す  $r$  次元ベクトルであり， $y = [y_1 \ y_2 \ \dots \ y_r]^T$  となる． $D$  はベクトル場の組  $(f_i, g_i)$  を要素としてもち，この組を用いてヒューマノイドロボットの軌道を生成する．生成方法は 3.1 節で説明する． $R = \{y_1^{res}, y_2^{res}, \dots\}$  は状態  $q_i$  での  $r$  次元ベクトル  $y$  の制約条件の集合であり，状態  $q_i$  では  $y \in y_i^{res}$  を常に満たすような行動しかとれない．ここで  $y_i^{res}$  は  $n$  次元のセンサ空間の可動範囲から求めた  $r$  次元空間上での  $y$  の存在できる領域である． $L_T = \{t_1, t_2, \dots\}$  は

各状態のローカルな時間の集合であり，状態  $q_i$  の時間を  $t_i$  と表す． $\Phi = \{\Phi_1, \Phi_2, \dots, \Phi_{12}, \dots\}$  は  $r$  次元ベクトル  $y$  から  $n$  次元のセンサ情報に復元する関数の集合である．ここで  $\Phi_i, \Phi_j$  はそれぞれ状態  $q_i, q_j$  の復元関数， $\Phi_{ij}$  は状態  $q_i$  から  $q_j$  への遷移中の復元関数である． $A = \{a_{12}, a_{21}, \dots\}$  はアークを表す集合であり，状態  $q_i$  から  $q_j$  へのアークを  $a_{ij}$  とし，各行動間の遷移を表す．状態間の遷移はアーク毎に定めている遷移時間  $\epsilon_{ij}$  がかかる． $x_{ij}^{gu}$  は状態  $q_i$  から状態  $q_j$  への遷移が可能な位相条件を表し， $x_i \in x_{ij}^{gu}$  の時は遷移命令の入力後すぐに遷移が行われるが， $x_i \notin x_{ij}^{gu}$  であるなら，遷移可能となるまで遷移を行わない． $G$  を  $x_{ij}^{gu}$  の集合とおく． $x_{ij}^{as}$  は状態  $q_i$  から  $q_j$  に遷移するときに  $q_j$  の初期位相を定める関数で， $x_{ij}^{as}$  の集合を  $A_s$  とおく．

### 3.1 行動間の遷移

ハイブリッドステートネットを用いた状態  $q_i$  から状態  $q_j$  の遷移の手順を示す．状態  $q_i$  においてベクトル場の組  $(f_i, g_i)$  を基に安定な周期軌道をもつ微分方程式  $\Sigma^i$  を構成する．

$$X_i = \begin{bmatrix} x_i \\ y \end{bmatrix} \quad (15)$$

$$\Sigma^i : \dot{X}_i = f_i(X_i) + g_i(X_i) \quad (16)$$

ここで図 2 に示すように，状態  $q_i$  のローカルな時刻  $t_i = t_{i1}$  に遷移開始命令が入力されたが，命令時の状態  $q_i$  の位相  $x_i \notin x_{ij}^{gu}$  であったため， $x_i \in x_{ij}^{gu}$  となる時刻  $t_i = t_{i2}$  のときに遷移が開始する．

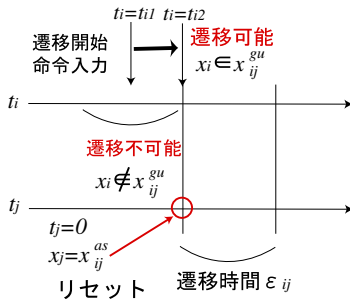


図 2: 状態遷移

次に遷移開始時に状態  $q_j$  のローカルな時刻  $t_j = 0$ ，位相  $x_j = x_{ij}^{as}$  にリセットが行われる．また遷移中は状態  $q_i$  のリミットサイクルから状態  $q_j$  のリミットサイクルに収束するため，次の二つの微分方程式で表される運動系を用いる．

$$X_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix} \quad X_j = \begin{bmatrix} x_j \\ y_j \end{bmatrix} \quad (17)$$

$$\dot{\Sigma}^i : \dot{X}_i = f_i(X_i) - g_i(X_i) \quad (18)$$

$$\Sigma^j : \dot{X}_j = f_j(X_j) + g_j(X_j) \quad (19)$$

ここで運動系  $\dot{\Sigma}^i$  は状態  $q_i$  のリミットサイクルから不安定となる運動系であり， $\Sigma^j$  は状態  $q_j$  の周期軌道を求める安定な運動系である．さらに重み関数

$$w_{ij}(t_j) = \frac{1}{2} \left( 1 - \cos\left(\frac{\pi t_j}{\epsilon_{ij}}\right) \right) \quad (20)$$

を用いることにより，状態  $q_i$  から状態  $q_j$  に遷移を行う運動系の微分方程式

$$X_{ij} = \begin{bmatrix} w_{ij}(t_j)x_i + (1 - w_{ij}(t_j))x_j \\ y \end{bmatrix} \quad (21)$$

$$\Sigma^{ij} : \dot{X}_{ij} = w_{ij}(t_j)(f_i(X_i) - g_i(X_i)) + (1 - w_{ij}(t_j))(f_j(X_j) + g_j(X_j)) \quad (22)$$

が得られ，時間の経過と共に運動系  $\dot{\Sigma}^i$  が不安定となり運動系  $\Sigma^j$  が安定となることにより運動系の遷移が起こる．ここで  $\epsilon_{ij}$  はヒューマノイドロボットの ZMP が安定となるように定める．遷移中の  $r$  次元ベクトル  $y$  から  $n$  次元のセンサ空間への復元は，それぞれの運動系  $i, j$  からの復元関数  $\psi_i, \psi_j$  を組み合わせて次のように定める．

$$\psi_{ij} = \omega_{ij}(t_j)\psi_i(y_i(t_i)) + (1 - \omega_{ij}(t_j))\psi_j(y_j(t_j)) \quad (23)$$

最後に  $t_j = \epsilon_{ij}$  となると遷移が終了し，状態  $q_j$  の微分方程式運動系による運動系  $\Sigma^j$  により周期軌道を行う．

## 4 実験

提案する手法を実機による実験によって確認する．ここでは，ヒューマノイドロボット HOAP-1(富士通オートメーション(株))を用いた．このロボットの全長は 0.438[m]，重量は 5.67[kg]．20 の関節を持ち，各腕は 4 つの自由度を持つ．制御用のホストコンピュータの CPU は Pentium 1GHz で，OS は RT-Linux，使用言語は C++を用いた．HOAP-1 の制御は USB ケーブルを介してホストコンピュータから直接行った．

まず初めに静止，歩行，スクワット，足踏みの時系列データを元に各行動を低次元化し，図 3 で示すハイブリッドステートネットを形成した．ここで  $q_o, q_w, q_s, q_l$  はそれぞれ静止，歩行，スクワット，足踏みに対応する状態であり，アーク  $a_{ow}$  は静止から歩行への遷移を表すアークと定める．

制約条件の集合  $R$  は  $n$  次元のセンサ空間での可動領域を各状態の復元関数の逆関数

$$S_1^{-1}U_1^T \quad (24)$$

によって  $r$  次元空間に射影し求めた．また遷移可能な位相条件の集合  $G$  は ZMP 規範により，脚底の範囲を半分狭めた支持脚領域でも ZMP が安定となる位相条件を  $G$  と定めた．遷移開始時に次の状態の初期位相を定める関数の集合  $A_s$  は遷移開始時において，式 (12) より求めた評価関数

$$(y - f_j(t_j))^T (y - f_j(t_j)) + (\dot{y} - \frac{df_j(t_j)}{dt})^T (\dot{y} - \frac{df_j(t_j)}{dt}) \quad (25)$$

を最小とする時の位相を  $x_j$  と定めた．

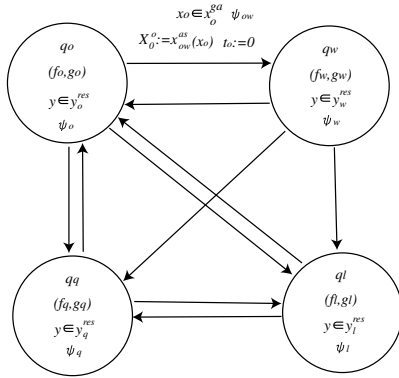


図 3: 作成したハイブリッド状態ネット

まず初期状態は静止の状態  $q_o$  とし，微分方程式の初期値は次のように定めた．

$$X_o = [0 \ 1 \ y_o]^T \quad (26)$$

ここで  $y_o$  は式 (12) より  $y_o = f(0)$  で求めた  $r$  次元のベクトルである．図 3 で示すハイブリッド状態ネットを用いて遷移を行っていき，状態を  $q_o \rightarrow q_w \rightarrow q_s \rightarrow q_l \rightarrow q_o$  と遷移させていった．各状態，アークの復元関数を用いて  $n$  次元のセンサ空間に復元を行い，さらに ZMP 修正を行うことにより，静止状態から歩行，スクワット，足踏みの動作を行った後静止する動作が転倒することなく実現できた．動作の様子を 4 に示す．

## 5 おわりに

本稿では，周期的な行動パターンの低減化によって歩行やスクワットのような周期的な行動を生成する運動系を作成し，ZMP 規範による軌道修正と遷移の開始タイミング修正を行った．また，ハイブリッド状態ネットを提案し，本手法により HOAP-1 の制御を行った．

## 参考文献

[1] 中村仁彦: ロボットの脳を創る; 岩波書店, (2003)

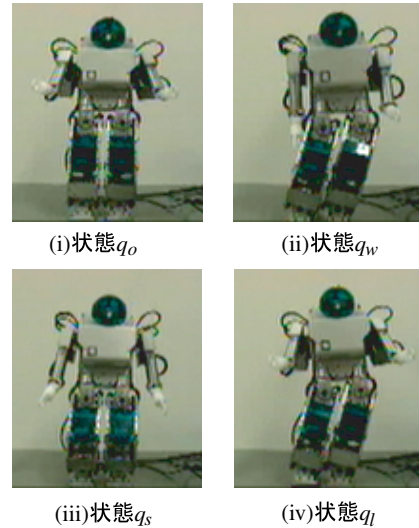


図 4: 実験結果

[2] M. Okada and Y. Nakamura: Polynomial design of the nonlinear dynamics for the brain-like information processing of whole body motion; in *Proceedings of the IEEE ICRA2002*, pp.1410-1415, May (2002)

[3] M. Adachi, T. Ushio, and S. Yamamoto: Synthesis of hybrid systems with limit cycles using piecewise quadratic Lyapunov functions; in *16th workshop on Circuits and Systems*, (2003)

[4] L. O. Chua and D. N. Green: Synthesis of nonlinear periodic systems; *IEEE Trans. Circuits and Syst.*, vol.21, p.286-294, March (1974)

[5] M. Adachi, T. Ushio, and S. Yamamoto: Application of Lyapunov Function Based Synthesis of Non-smooth Limit Cycles to Motion Generation for Humanoid Robots; *Proc. 2nd Int. Symp. Adaptive Motion of Animals and Machines, SaP-II*, p.249-254, (2003)

[6] F. Kanehiro, M. Inaba, H. Inoue, and S. Hirai: Developmental Realization of Whole-Body Humanoid Behaviors Based on State Net Architecture Containing Error Recovery Functions; *Proc. of the First IEEE-RAS International Conference on Humanoid Robots*, (2000)