

ヒューマノイドのための短周期オンライン歩行軌道生成更新法

西 脇 光 一* 加賀美 聡*

Short Cycle Pattern Generation for Onlilne Walking Control of Humanoids

Koichi Nishiwaki* and Satoshi Kagami*

This paper presents a humanoid walking control system that generates and updates the motion pattern frequently, with a cycle time on the order of 20 [ms]. Frequently updating the motion pattern enables a robot to quickly respond to changes in the commanded walking direction and upper body posture with minimal delay. Our experiments also show that short cycle times are useful for building a walking system that provides motion status information and sensor feedback to the motion generation layer in order to maintain consistency of the overall dynamics. Using preview control theory, we generate horizontal torso trajectories that can realize desired ZMP trajectories. Trajectories for the free leg are generated using a method that allows for changes to the landing position while taking a step, and adjusting the foot placement to avoid large ZMP errors which may result from sudden changes in the desired motion. We present an implementation of the proposed method on the full-size humanoid HRP-2 and show experimental results designed to validate the overall functionality.

Key Words: Humanoid, Biped Walking, Preview Control, ZMP

1. はじめに

多自由度、分布質量系の複雑な動力学モデルを持つヒューマノイドの歩行の多くは、事前に動力学的安定な位置軌道を生成し、それをセンサフィードバックによりモデル世界と実世界の差を吸収しながら再生することによって実現されてきた（例えば文献 [1]~[3]）。そして近年、目標移動を満たす位置軌道をオンラインで生成・更新することで、オンラインで歩行をコントロールする方法が考案されてきている [4]~[11]。オンラインで歩行を制御することにより、ジョイスティック等を用いた遠隔操縦やオンラインでの環境観測の結果を用いた経路計画に基づく移動が実現されてきた。

しかしながら提案されてきたオンライン制御法の軌道更新の周期は1歩より長いため、目標入力から実現までの時間遅れが大きい。また台車を押す、ドアを開けるなどの動作をしながら歩行するためには、より短い周期で上体の運動を動力学的整合性をもって変更できることが望まれる。

そこで、本論文では、20 [ms] といった短い周期で歩行軌道を生成、更新する方法を提案する。さらに軌道生成が短周期で行われるという特徴を生かし、センサフィードバックにより修正された運動状態を反映した動力学的安定軌道生成を逐次行うよ

うにすることで、センサフィードバック層でなく、軌道生成層が実際の歩行における長期的安定化を担うことができることを示す。

また実行中の1歩の着地位置を変更するために必要となる、ロボットの性能限界を考慮して遊脚中に同軌道を変更する方法及び、近い将来の目標着地位置を変更することが転倒を引き起こすことを防ぐために着地位置を調整する方法について論じる。

2. 短周期オンライン歩行軌道生成更新システム

提案する短周期オンライン歩行軌道生成更新システムの概要を Fig. 1 に示す。従来1歩程度の周期であった歩行軌道生成を20 [ms] 程度に短周期化する。経路計画、歩容計画の手法の研究が進み、その計画周期は1歩より短くなってきているため [12]、軌道生成の周期を短くすることは自律移動システム全体の応答

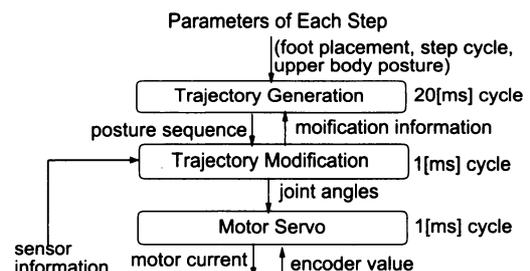


Fig. 1 Walking control system with short cycle pattern generation

原稿受付 2007年1月15日

*独立行政法人産業技術総合研究所

*National Institute of Advanced Industrial Science and Technology (AIST)

性能を向上させることとなる。また、ジョイスティック等による操縦を行う場合の即応性の向上にも貢献する。

また従来提案されてきたオンライン軌道生成法は、軌道生成は力学シミュレーションの世界で完結し、センサフィードバック運動修正層による運動変更の結果を反映することはなかった。そのため、運動修正層では、転倒を回避するために運動を元軌道から変更すると同時に、運動軌道が元軌道から大きく乖離しないように維持する必要がある。本論文では、運動修正層により変更された運動軌道を元に、変更後の軌道に接続する動力学的安定な運動軌道生成を高頻度で行うことにより、長期的な安定化は精密モデルを用いた軌道生成層が行い、センサフィードバックによる運動修正層は、反射的な短期安定化に集中させることができることを示す。重心集中モデルとして取り扱可能な2足歩行ロボットにおいては、文献[13][14]に代表されるように、オンラインで運動状態を推定し、簡略化ダイナミクスモデルを用いて、推定された運動状態から転倒せずに与えられた目標を達成する運動指令を生成する方法が提案されてきた。本論文で提案する手法は、高頻度で軌道を更新することにより、多自由度のダイナミクスモデルを用いて、同様の効果を実現するためのアプローチと位置づけることもできる。

さらに、実行中の1歩の着地位置を変更可能とするための遊脚軌道設計法として、遊脚の速度、加速度が制限範囲内となり、実行中の軌道に滑らかに接続する方法を開発し導入する。

また、近い将来の目標運動が変更可能となることから、例えば遊脚期中の着地位置をどの程度までなら変更しても転倒せずに実現できるかというような、変更可能範囲を考慮する必要がある。そこで着地位置目標について、与えられた目標に従う運動軌道を生成した後、転倒せずに実現可能かを評価し、実現不能であれば変更量を修正する方法を考案する。また、軌道生成はセンサフィードバックにより修正された運動状態を反映して行うため、この着地位置調整が外乱により崩れたバランスを回復するのにも効果があることを示す。

3. 予見制御を用いた短周期歩行軌道生成

3.1 予見制御理論を用いたZMP追従運動軌道生成

短周期軌道生成においては、目標ZMPに追従する運動軌道を生成する手法として、梶田らにより提案された予見制御を用いて目標ZMP軌道を実現する重心軌道を生成する方法[15]を用いる。この方法は、生成した軌道の始点においても目標のZMPを満たすように軌道を生成できるため、他の目標ZMP追従型運動軌道生成法[2][10][16]に比べ、短い運動軌道を高頻度で更新するのに適している。以下に同手法の概要を述べる。

ロボットが重心集中系であるとし、 z 軸を鉛直上向き、 $\mathbf{P} = [p_x, p_y, p_z]^T$ 、 $\mathbf{Q} = [q_x, q_y, q_z]^T$ をそれぞれZMPおよび重心の位置であるとする(Fig. 2)。ここで、 $\dot{q}_z = 0$ であると仮定するとZMPの x 成分 p_x は、次のように求められる。

$$p_x = q_x - \frac{q_z - p_z}{g} \ddot{q}_x, \quad (1)$$

ここで、 g は重力加速度である。さらに、 u_x を \ddot{q}_x の時間微分とすると、式(1)は、次のような動的システムとして表現することができる。

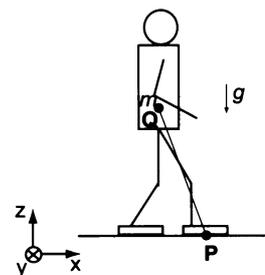


Fig. 2 Coordinate system for mass concentrated model

$$\frac{d}{dt} \begin{bmatrix} q_x \\ \dot{q}_x \\ \ddot{q}_x \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} q_x \\ \dot{q}_x \\ \ddot{q}_x \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u_x, \quad (2)$$

$$p_x = \begin{bmatrix} 1 & 0 & -\frac{q_z - p_z}{g} \end{bmatrix} \begin{bmatrix} q_x \\ \dot{q}_x \\ \ddot{q}_x \end{bmatrix}. \quad (3)$$

このシステムを時間間隔 T で離散化し、次のように表す。

$$\mathbf{q}_x(k+1) = \mathbf{A}\mathbf{q}_x(k) + \mathbf{B}u_x(k), \quad (4)$$

$$p_x(k) = \mathbf{C}\mathbf{q}_x(k). \quad (5)$$

ここで、エラーシステムで取り扱うため、

$$\begin{cases} \Delta u_x(i) = u_x(i) - u_x(i-1), \\ \Delta \mathbf{q}_x(i) = \mathbf{q}_x(i) - \mathbf{q}_x(i-1), \\ \Delta p_x^{ref}(i) = p_x^{ref}(i) - p_x^{ref}(i-1), \\ \mathbf{q}_x^*(i) = \begin{bmatrix} p_x(i) \\ \Delta \mathbf{q}_x(i) \end{bmatrix}, \end{cases} \quad (6)$$

とする。ここで、 $p_x^{ref}(i)$ は、目標ZMPの x 成分である。そして、評価関数を次のように設定する。

$$J = \sum_{i=k}^{\infty} \left\{ R_1 \left(p_x^{ref}(i) - p_x(i) \right)^2 + R_2 \Delta u_x^2(i) \right\}, \quad (7)$$

(ただし、 R_1, R_2 は正数)

ここで N ステップ先までの目標ZMPを考慮することとすると、評価関数を最小化する $\Delta u_x(k)$ は、予見制御理論により次の式で与えられる。

$$\Delta u_x(k) = -G_x q_x^*(k) - \sum_{j=1}^N G_p(j) \Delta p_x^{ref}(k+j). \quad (8)$$

ここで G_x と $G_p(j)$ は、 R_1, R_2 および $(q_z - p_z)/g$ から決定される係数である。

これを時間軸に沿って順次解くことにより、目標ZMPに追従する重心 \mathbf{Q} の運動軌道の x 成分 q_x が求められる(q_y についても同様に求められる)。

3.2 パラメータの事前計算による高速化

式 (8) のゲイン G_x , $G_p(j)$ を決定するためには Riccati 方程式を数値的に解く必要がある。しかし, G_x , $G_p(j)$ を決定するパラメータの中で歩行条件により変化する可能性があるのは, $z_h = q_z - p_z$ のみである。そこで, 取りうる可能性のあるすべての z_h について, G_x および $G_p(j)$ を事前に計算しておくこととした。実際には, 広めに見積もって, $z_h = 0.001$ [m] から 1 [m] を 0.001 [m] 刻みで事前に計算し, G_x , $G_p(j)$ の値を蓄積しておくことにした。ここで, 離散時間間隔 $T = 0.01$ [s], $R_1 = 1 \times 10^7$, $R_2 = 10$ とし, $G_p(j)$ は $t = 1.6$ [s] までの 160 ステップを計算した。

4. 短周期オンライン歩行軌道生成システムの構成と実装

4.1 システム構成

オンライン歩行軌道生成の流れを Fig. 3 に示す。歩行動作は, 両脚支持期の中央を境界として 1 歩単位に区切られ, 1 歩ごとの歩行のパラメータとして, 着地位置, 1 歩の時間, 上体運動軌道等が上位層から与えられる。以下, Fig. 3 の各層について説明する。

4.1.1 歩行軌道の設計

これより後の層では, 胴体の水平位置軌道を生成することにより動学的な安定な歩行軌道を生成するので, 本階層 (Fig. 3 (b)) では, 足の軌道, 目標 ZMP 軌道, 胴体の運動軌道の鉛直および姿勢の成分, 上体各関節の軌道をあらかじめ設計する。これらの軌道は, 1 歩ごとの歩行パラメータとして与えられた着地位置, 歩行周期, 上体運動等から設計される。

予見制御を用いた方法では, 将来の目標 ZMP を与える必要があり, そのため将来の目標軌道情報が必要となる。本論文では, 1.6 [s] (t_{p1}) 先までの情報を用いて予見制御を行い軌道を作ることとした。さらに後に述べるように重心集中モデルで取り扱う誤差を補償するために, 予見期間 0.7 [s] (t_{p2}) での予見制御を繰り返したので, 生成する軌道の時間に加えて 2.3 [s] ($t_{p1} + t_{p2}$) の軌道をこの階層で設計しておく必要がある。そこで次のような方法で, 軌道を生成することとした (Fig. 4)。もし上位層からの指令が十分な歩数であれば, 生成対象となる 1 歩に加えて必要な時間を満たすまでの数の歩数の軌道を設計する (Fig. 4 (a))。もし上位層からの指令の歩数が十分でなければ, 指令されている歩数すべての分の軌道を設計し, さらに停止のための期間を付け加え, それでも足りない場合は, その場で停止している軌道を付け加える (Fig. 4 (b), (c))。また生成する 1 歩が停止に到る 1 歩の場合は, その後に停止のための期間および 2.3 [s] の静止している軌道を付け加える (Fig. 4 (d))。ここで, 停止のための期間とは予見制御を用いて生成される胴体の水平運動が停止するのを待つために設けた期間で, この階層においては, その場に停止している軌道と同一の設計が行われるが, 実行時に途中で止められないという点で違う期間としている。

4.1.2 重心の位置, 速度, 加速度の計算

$\mathbf{X}(t)$ を時刻 t における胴体の位置姿勢, $\Theta(t)$ を時刻 t における全身の関節角度とし, $\mathbf{CM}(\mathbf{X}, \Theta)$ を重心の位置を胴体の位置姿勢, 関節角度から力学モデルを用いて求める関数であるとする。すると, 生成軌道の開始時刻 t_0 における重心の位置

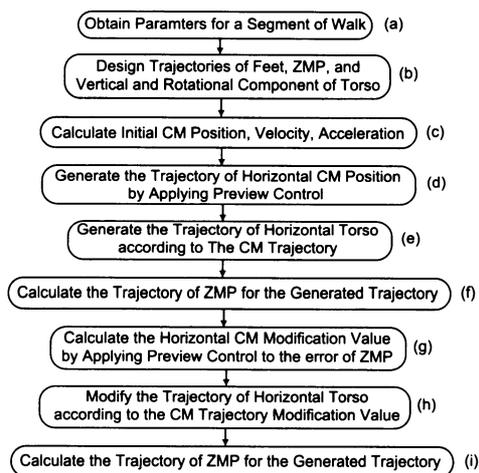


Fig. 3 Flow of walking pattern generation

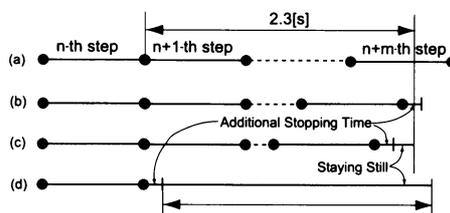


Fig. 4 Making a segment of walking for preparing a reference trajectory

$\mathbf{Q}(t_0)$ は, 次式のようになる。

$$\mathbf{Q}(t_0) = \mathbf{CM}(\mathbf{X}(t_0), \Theta(t_0)). \quad (9)$$

開始時刻における速度は数値的に,

$$\dot{\mathbf{Q}}(t_0) = \frac{\mathbf{Q}(t_0 + \Delta t) - \mathbf{Q}(t_0)}{\Delta t}. \quad (10)$$

として求められる。ここで, Δt は, 0.001 [s] とした。また, 重心の開始時刻における加速度については, 胴体並進速度と重心速度が正比例の関係にあると仮定して, その仮定の下では, 重心加速度と胴体加速度の比が, 重心速度と胴体速度の比と一致することから, 重心加速度の各成分 $\dot{Q}_i(t_0)$ について, 対応する胴体加速度の成分 $\ddot{X}_i(t_0)$ を用いて,

$$\dot{Q}_i(t_0) = \frac{\dot{Q}_i(t_0)}{\dot{X}_i(t_0)} \ddot{X}_i(t_0), \quad (11)$$

として求めることとした。ただし, $\dot{X}_i(t_0) \approx 0$ のときは, $\dot{Q}_i(t_0) = \ddot{X}_i(t_0)$ とした。

生成軌道の始点における胴体の位置, 速度, 加速度および, 関節角度は, 運動中は前回生成した軌道から, 運動開始時点においては, 実ロボットの姿勢から得ることができる (運動開始時の胴体速度, 加速度は 0)。センサフィードバックによる運動軌道修正層において, 修正した影響をどう取り扱うかについては, 5 章で述べる。

4.1.3 予見制御を用いた重心軌道の生成

軌道開始点での重心の位置, 速度, 加速度および目標 ZMP 軌道から予見制御を用いて, 目標重心軌道 (水平成分) を生成

する。後にもう一度予見制御を用いた軌道生成を行うので、ここでは、 $s + t_{p2}$ の長さの運動軌道を生成する (s は最終的に生成する運動軌道。ここでは、 0.02 [s])。

4.1.4 重心軌道から胴体軌道の生成

胴体軌道の水平成分 $\mathbf{X}_h(t)$ ($t = t_0$ から $t_0 + s + t_{p2}$) を生成された目標重心軌道の水平並進成分 $\mathbf{Q}_h(t)$ より決定する。ここでは、次式により $\mathbf{X}_h(t)$ を決定することとした。

$$\mathbf{X}_h(t) = \mathbf{Q}_h(t) - \mathbf{Q}_h(t_0) + \mathbf{X}_h(t_0). \quad (12)$$

4.1.5 ZMP 軌道の計算

予見制御による軌道生成は質量集中系を用いていること、計算された目標重心軌道を胴体の軌道に反映する際に近似を用いていることに起因する誤差を評価するため、質量分布系の動力学モデルを用いて、生成された運動軌道の ZMP 軌道を計算する。

4.1.6 予見制御に基づく重心修正量の計算

文献 [15] に述べられているように、重心集中モデルとして近似したことに起因する誤差の影響は、予見期間 0.7 [s] 程度の予見制御を ZMP の誤差に適用し、重心の修正量を求めることで、取り除くことができる。そこで、入力を $t = t_0$ から $t_0 + s + t_{p2}$ の ZMP の誤差として、 $t = t_0$ から $t_0 + s$ の水平重心位置修正量 $\Delta \mathbf{Q}_h$ を求める。軌道開始点における重心修正量に関する次の変数 $\Delta \mathbf{Q}_h(t_0)$, $\Delta \dot{\mathbf{Q}}_h(t_0)$ および、 $\Delta \ddot{\mathbf{Q}}_h(t_0)$ は、 0 とする。

4.1.7 胴体水平位置軌道の修正

胴体水平位置軌道 $\mathbf{X}_h(t)$ ($t = t_0$ から $t_0 + s$) に、求められた重心位置修正量を加算することにより、胴体水平位置軌道を修正する。

4.1.8 ZMP 軌道の計算

最後に、センサフィードバック修正層のリファレンスとして与えるために ZMP 軌道を分布質量モデルを用いて再び計算する。

4.2 HRP-2 への実装

前節で述べたシステムを HRP-2 [17] に実装した。HRP-2 の運動制御計算機を Pentium M 2.1 [GHz] を搭載した計算機ボードに置換し、その上でベクトル・行列演算に SIMD (SSE) 命令を用いることにより計算の高速化を図った。また、HRP-2 とともに提供される OpenHRP 制御ソフトウェア [18] は使用せず、今回提案する歩行制御システムの実装が関節角サーボの指令値を直接に生成する構成とした。その結果、搭載の計算機上で、歩行に必要な他の制御を行いながら、 10 [ms] のサンプリング周期で 20 [ms] の歩行軌道を生成するのに、 9.2 [ms] 程度の時間を要している。歩行軌道生成より優先度の高い制御は、関節角サーボ、センサフィードバック運動修正、軌道の補間・再生があり、すべて 1 [ms] 周期で実行されている。歩行軌道生成の各区間における計算時間を Table 1 に示す。Table 1 (a)~(i) は、Fig. 3 の記号に対応する。この表より、計算時間の約 $2/3$ が 0.72 [s] ($t_{p2} + s$) の軌道の ZMP の計算 (f) に費やされていることが分かる。ZMP は各サンプリング時刻における運動状態から計算するため、ZMP 軌道の計算時間は運動時間に比例す

Table 1 Calculation time for each section

section	(a)	(b)	(c)(d)	(e)	(f)	(g)	(h)	(i)
time [ms]	0.1	1.4	0.1	0.8	5.8	0.03	0.8	0.4

る。さらに、 t_{p2} は設計上固定の値なので、 s をこれより小さくしても全体の計算時間は大きくは変化しない。そこで、生成周期すなわち s を 20 [ms] とすることとした。

5. センサフィードバック修正の軌道生成への反映

センサフィードバックの結果を軌道生成層に反映する方法の一つとして、ZMP 誤差補償のためのセンサフィードバックとその結果の反映について述べる。いま、 $\mathbf{P}_s(t_i)$, $\mathbf{P}_m(t_i)$, $\Delta \mathbf{X}_h(t_i)$ をそれぞれ時刻 t_i における軌道生成の結果与えられた目標 ZMP の水平成分、力センサの値から計算された ZMP の水平成分、胴体位置の水平方向修正量としたとき、次式により目標 ZMP への追従を図ることができる [19]。

$$\Delta \mathbf{X}_h(t_i) = \Delta \mathbf{X}_h(t_{i-1}) + k_1(\mathbf{P}_m(t_i) - \mathbf{P}_s(t_i)). \quad (13)$$

この修正では、上体を水平方向に加速することにより、ZMP の誤差を補償するため、短期的には目標の ZMP に近づき、バランスを保つことができるが、実行軌道は元の軌道から離れていくため、長期的に見ると、生成した力学的安定軌道に修正量を加算した軌道は、力学的安定性を実現しない軌道となる。そのため、転倒回避のための何らかの制御がさらに必要となる (文献 [19] では、倒立振子の支持点変更問題として目標 ZMP を変更する方法を提案している)。筆者らは、生成された運動軌道から乖離しないように引き戻す項を加えた次式を用いて、歩行制御を実現してきた。

$$\Delta \mathbf{X}_h(t_{i+1}) = \Delta \mathbf{X}_h(t_i) + k_1(\mathbf{P}_m(t_i) - \mathbf{P}_s(t_i)) - k_2 \Delta \mathbf{X}_h(t_{i-1}). \quad (14)$$

(ただし、 $k_2 \ll 1$)

この引き戻し項は、動力学を直接考慮したものではなかった。また、前述の目標 ZMP を変更する方法も倒れる振子の復元を図るものであり、長期の安定性確保を考慮したものではない。今回、軌道生成周期を短くし、目標 ZMP に追従させるために胴体軌道を変更した結果を考慮した動力学安定軌道を生成することにより、この引き戻し項を不要とし、センサフィードバック修正層は目標 ZMP への追従に専念し、軌道生成層が力学モデルに基づき長期的安定を図るという枠組みを実現する。

軌道生成は前章で述べたように、センサフィードバック修正周期に比べて、無視できない時間を要するため、接続時点での修正量を生成開始時に予測する必要がある。いま、生成開始時刻を t_s 、接続の時刻を t_u として、接続時刻における胴体の水平位置修正量 $\Delta \mathbf{X}_h(t_u)$ を次のように見積もることとした。

$$\Delta \mathbf{X}_h(t_u) = \Delta \mathbf{X}_h(t_s) + \Delta \dot{\mathbf{X}}_h(t_s)(t_u - t_s). \quad (15)$$

さらに、 $\Delta \ddot{\mathbf{X}}_h(t_u)$ および、 $\Delta \ddot{\mathbf{X}}_h(t_u)$ を推定する必要があるが、 $\Delta \ddot{\mathbf{X}}_h(t_u)$ は 0 と見積もることとし、 $\Delta \dot{\mathbf{X}}_h(t_u)$ については 2 種の方法を試した。すなわち 0 と見積もる方法、 $\Delta \dot{\mathbf{X}}_h(t_s)$ と見積もる方法である。どちらの方法でも歩行を安定して継続することができたが、実験の結果適切な k_1 の値が、第 1 の方法では、 1×10^{-3} 、第 2 の方法では 2×10^{-4} となった (制御周期

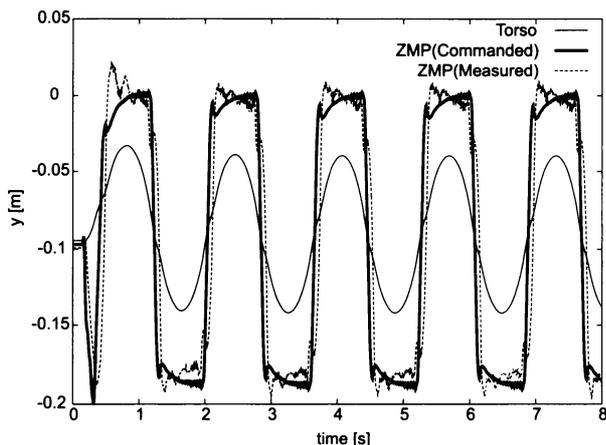


Fig. 5 Trajectories of the torso and ZMP while walking on the spot

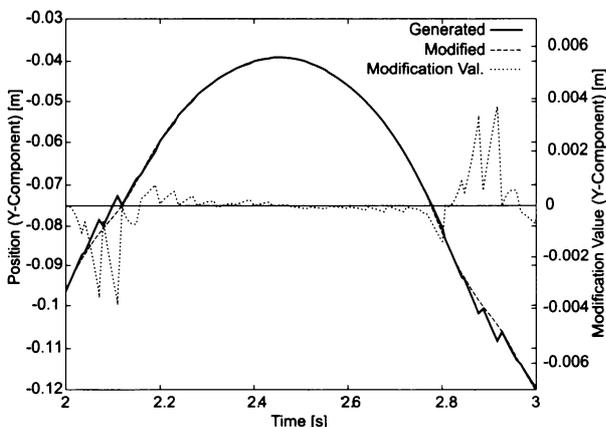


Fig. 6 Generated trajectory and executed one after sensor feedback

1 [ms]). これは、第2の方法では、修正の傾向を引き継いで軌道生成するのに対して、第1の方法では修正の傾向を考慮せずに接続する軌道を生成するので、修正ゲインを大きくとれる結果となったと考える。式(13)から分かるように $\Delta \dot{\mathbf{X}}_h(t_s)$ は、制御周期ごとの ZMP 誤差から決まるため大きく変動しやすい値である。そこで、今回は第1の方法を採用した。

Fig. 5 は、足踏み歩行における軌道生成の結果から予定される ZMP (センサフィードバックにおける目標 ZMP) と計測された ZMP, 胴体の位置の横方向成分を示している。Fig. 6 は、Fig. 5 と同じ歩行の一部を抜き出したもので、軌道生成層により生成された胴体軌道とセンサフィードバックにより修正された結果実行された胴体軌道、その差である胴体の水平方向修正量 $\Delta \dot{\mathbf{X}}_h(t)$ を拡大したものの横方向成分を示している。20 [ms] 毎の軌道生成において修正された胴体軌道に接続する軌道が生成されるため、定期的に修正量が0に近づいている。なお修正量が完全に0にならないのは、軌道生成開始時の修正量で接続時点の修正量を見積もっているため誤差が生じるためである。

6. 遊脚軌道の設計

本章では、遊脚期間中にその着地位置が変更された場合にも

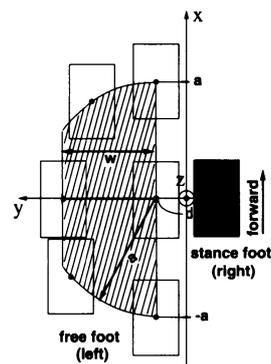


Fig. 7 Limitation of free foot transition

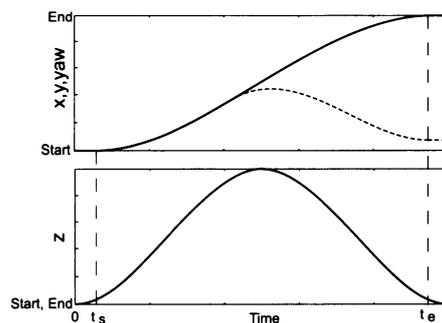


Fig. 8 Trajectories of a free foot

滑らかな軌道を生成する遊脚軌道の設計法について論じる。この遊脚軌道設計は Fig. 3 (b) の一部として実行されるものである。また本章での遊脚軌道設計には Fig. 7 に示される立脚足部位置を基準とした座標系を用い、水平面内での歩行を仮定する。

遊脚期となる以前の軌道生成においては、遊脚足部の並進回転各成分は、時刻に対する3次曲線を用いて、Fig. 8 に示すように設計する。z (鉛直) 成分は、開始時および中点で速度0となる曲線と中点および終了時で速度0となる曲線の二つの曲線で、x, y および旋回角は運動期間を t_s から t_e までに制限し、両時刻に速度0となる一つの曲線で構成する。運動期間を制限するのは、離地および着地を垂直に行うためである。

遊脚期間中の軌道生成においては、z 成分に関しては、遊脚期になる前に設計した軌道から変更しないこととし、x, y および旋回角は、接続点において位置、速度が連続で、時刻 t_e に速度0で目標位置に達する3次曲線として設計することとした (Fig. 8 の破線)。また着地位置を立脚足部から見て Fig. 7 の斜線部に示すように中立位置から半径 a の半円内かつ幅 w の範囲内に制限する。ただし両足間の最短距離 d は、両足の相対旋回角に比例して設定する。すると着地位置は、Fig. 7 で示される範囲内でのみ変化するので、遊脚期中にその軌道を変更しても脚間の干渉を考慮する必要は生じない。

x, y および旋回角の最大速度、最大加速度を制限することにより、遊脚関節の角速度、角加速度が過大となるのを防ぐこととした。これらの運動軌道は3次曲線であるので、最大速度および加速度を計算するのは容易である。そこで、これらを評価し設定した上限値を超過した場合には、着地位置を変更前の着地位置に近づけることとした。1周期前の軌道生成で速度、加

Table 2 Velocity and acceleration limits for a free foot

	x	y	yaw
Vel.	1.5[m/s]	0.50[m/s]	3.0[rad/s]
Acc.	15[m/s ²]	5.0[m/s ²]	30[rad/s ²]

速度上限を超過せず実現可能とされた変更前の着地位置へは、速度、加速度上限を超過せずに到達可能である（3次曲線で表現し、位置、速度の連続性を条件として曲線を接続しているの、着地位置を変更しない場合には、実行中の前回生成時の曲線と同一の形状となる）。そこで上限値を超過した場合には、着地位置を変更前の位置に一定量（現在の実装では、0.0025[m] および 0.025[rad]）近づけ、再度速度、加速度を評価する。この過程を繰り返すことにより、各成分の最大速度および加速度が上限値以下となる範囲で着地位置変更を反映することとした。HRP-2 への実装で採用した上限値を **Table 2** に示す。

7. バランス維持のための着地位置調整

遊脚期間中に着地位置を変更することは、近い将来の目標 ZMP を変更することとなる。その結果、予見制御において ZMP の追従性が十分とならないことがある。そこで予見制御を用いて生成した運動軌道において ZMP の目標からの誤差が大きい場合に、着地位置すなわち将来の目標 ZMP を調整して、生成した軌道の ZMP 誤差を目標範囲内に抑える方法を考える。

目標 ZMP は、片脚立脚期は立脚足部の中心に固定し、両脚支持中に次の立脚足部の中心に等速運動するように軌道設計することとした。そのため、将来どの時刻から目標 ZMP を変更できるかは、着地までの時間によって決まる。

l ステップ将来以降の目標 ZMP を δp_x^{ref} だけオフセットさせるということは、式 (6) より、エラーシステムでは、 $\Delta p_x^{ref}(l)$ を δp_x^{ref} だけ変更することである。そこで、式 (8) を用いると、 l ステップ将来以降の目標 ZMP を δp_x^{ref} だけオフセットさせた場合に時刻 0 に u_x がどれだけ変化するか ($(\delta u_x(0) (= \delta \Delta u_x(0)))$) は、次式により求められる。

$$\delta u_x(0) = G_p(l) \delta p_x^{ref}. \quad (16)$$

式 (4), (5) より、 $p_x(1)$ は、 $\delta u_x(0)$ により、次のように変化する。

$$\delta p_x(1) = \mathbf{CB} \delta u_x(0). \quad (17)$$

すなわち、 $\delta p_x(1)$ を実現するための δp_x^{ref} は、次のとおりとなる。

$$\delta p_x^{ref} = G_p(l)^{-1} (\mathbf{CB})^{-1} \delta p_x(1). \quad (18)$$

そこで、生成された軌道の ZMP の水平成分 ($\mathbf{P}^{gen}(1)$) の目標 ZMP の水平成分 ($\mathbf{P}^{ref}(1)$) からの誤差 ($\mathbf{P}^{err}(1)$) を次のとおり計算し、

$$\mathbf{P}^{err}(1) = \mathbf{P}^{gen}(1) - \mathbf{P}^{ref}(1) \quad (19)$$

誤差 ($|\mathbf{P}^{err}(1)|$) が設定した閾値 (p^{tor}) より大きければ、誤差を閾値以下に抑えるために、 $\delta \mathbf{P}(1)$ を次式により計算することとした。

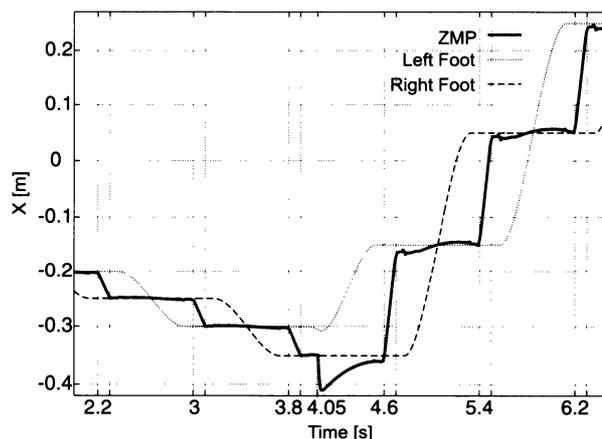


Fig. 9 Trajectories of feet and ZMP (without ZMP adjustment)

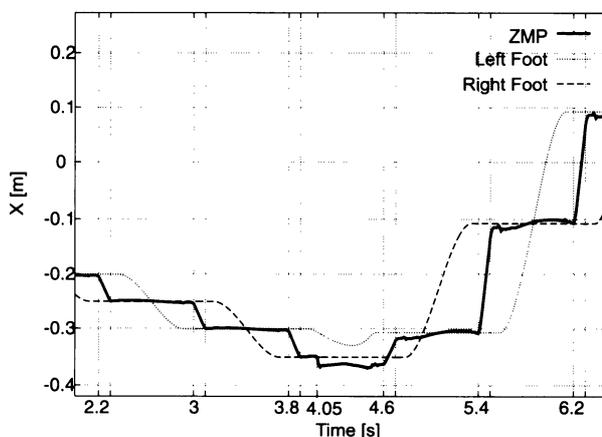


Fig. 10 Trajectories of feet and ZMP (with ZMP adjustment)

$$\delta \mathbf{P}(1) = - \left(\mathbf{P}^{err}(1) - \frac{p^{tor}}{|\mathbf{P}^{err}(1)|} \mathbf{P}^{err}(1) \right) \quad (20)$$

$\delta \mathbf{P}(1)$ と次の着地までの時間（ただし片脚支持期中、時刻 t_e 以降はさらに次の着地までの時間）より求められる l を用いて、式 (18) により、 δp_x^{ref} , δp_y^{ref} を求め、着地位置をこれらが実現されるように修正し、前章で述べた遊脚の最大速度、加速度制限を適用した後、軌道生成を再度行った。 p^{tor} は、HRP-2 への実装では、0.02[m] とした。

8. 着地位置調整機能の検証実験

8.1 目標移動の急激な変化に対する着地位置調整

目標移動を急激に変化させる実験として、**Fig. 9** および **Fig. 10** に示すように、片脚支持期中の時刻 4.05[s] に 1 歩 0.05[m] 後方への歩行から、1 歩 0.2[m] 前方への歩行に目標を変化させたときの実際の着地位置の変化および、生成された運動の ZMP の挙動を調べた（図中縦線の時刻が離地、着地時刻）。また目標 ZMP は、片脚支持期中は固定とし、各足の位置に一致するように座標系を設定した。なお本実験においては軌道生成の様子を調べるため、センサフィードバックは考慮せず軌道生成の結果とそこから逆動力学計算によって得られる ZMP を評価した。**Fig. 9** は、バランス維持のための着地位置調整を

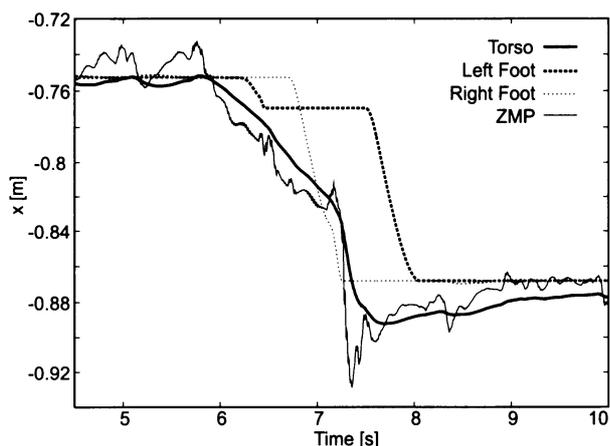


Fig. 11 Adjustment of foot placement according to the external force

行わなかった場合の結果である。時刻 4.05 [s] 以降、片脚支持期中であるにもかかわらず生成された運動の ZMP が大きく変化していることが分かる。それに対し、バランス維持のための着地位置調整を行った場合の結果を Fig. 10 に示す。着地位置調整の結果、着地位置目標変化への追従はさらに悪くなったが、ZMP の誤差の x 成分は、閾値として設定した 0.02 [m] 以内に抑えられている。

8.2 着地位置調整による外乱適応バランス維持

本オンライン歩行制御システムでは、センサフィードバックによって修正された運動状態が次の歩行軌道生成における始点の運動状態に反映されるため、センサフィードバックの結果、そのままの目標着地では安定して歩行を継続できない場合にも着地位置調整機能で同様に対応できる。

その場での足踏み歩行中に、時刻 6 [s] 付近でロボットを 1[m] 程度の高さで後ろ向きに押した際の足、胴体、生成された運動から計算された ZMP の軌跡を Fig. 11 に示す。ロボットが押された結果、ZMP 追従制御により胴体を後方に加速し、足が地面からはがれて、転倒するのを回避した。しかしその結果、胴体 (Fig. 11 の Torso) が後方に移動したため、目標着地位置をそのままにしておくと生成された軌道の ZMP (Fig. 11 の ZMP) も後方に移動してしまい長期的な安定性が確保できない。ここで、ZMP の誤差が着地位置調整の閾値を超えると着地位置すなわち将来の目標 ZMP が変更される。このケースでは、6.4 [s] 付近で着地する 1 歩の実行中に、着地位置が後方に変更され、続くもう 1 歩の着地位置変更と合わせて、後方に踏み出してバランスを維持する結果となった。着地位置調整が外乱からのバランス回復にも有効に働くことが確認できた。

9. おわりに

本論文では、20 [ms] とした短周期で歩行軌道を生成、更新するシステムの構成法について述べ、HRP-2 における実装を示した。短周期軌道生成により、運動目標変化の反映が高速化することに加え、運動軌道生成をセンサフィードバックの結果修正された現在の運動状態を考慮して行うことにより、軌道生成が力学モデルを用いた長期的安定化を担うことができることを

示した。

また、実行中の 1 歩の着地位置を変更するために必要となる遊脚軌道の設計法、および着地位置の急激な変化により目標 ZMP 追従が不可能になり転倒することを回避するための着地位置調節法について述べた。さらにこの将来の着地位置を調節して現在の ZMP を目標に近づける方法は、外乱に適応して着地位置を変化させ安定に歩行を継続させるのにも役立つことを述べ、これら提案手法についても HRP-2 への実装、実験を通じて有効性を確認した。

今後は、提案した転倒回避の短期的適応はセンサフィードバック層、長期的安定は軌道生成層の枠組みでの床面形状や外乱に対し頑健な歩行制御システムの構築を目指して、段差・傾斜地への足底部の着地適応、上体の姿勢角保持等の複数の短期的適応を含むシステム構成法の実現研究を行っていく。

参 考 文 献

- [1] K. Hirai: "Current and future perspective of Honda humanoid robot," Proc. of 1997 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.500-508, 1997.
- [2] J. Yamaguchi, S. Inoue, D. Nishino and A. Takanishi: "Development of a bipedal humanoid robot having antagonistic driven joints and three dof trunk," Proc. of the 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.96-101, 1998.
- [3] K. Nagasaka, M. Inaba and H. Inoue: "Walking pattern generation for a humanoid robot based on optimal gradient method," Proc. of 1999 IEEE International Conference on Systems, Man, and Cybernetics, pp.VI-908-VI-913, 1999.
- [4] S.A. Setiawan, S.H. Hyon, J. Yamaguchi and A. Takanishi: "Physical interaction between human and a bipedal humanoid robot—realization of human-follow walking—," Proc. of the 1997 IEEE International Conference on Robotics and Automation, pp.361-367, 1999.
- [5] K. Nishiwaki, T. Sugihara, S. Kagami, M. Inaba and H. Inoue: "Realtime generation of humanoid walking trajectory by mixture and connection of pre-designed motions—Online control by footprint specification—," Proc. of International Conference on Robotics and Automation, pp.4110-4115, 2001.
- [6] K. Yokoi, F. Kanehiro, K. Kaneko, K. Fujiwara, S. Kajita and H. Hirukawa: "A Honda humanoid robot controlled by AIST software," Proc. of the IEEE-RAS International Conference on Humanoid Robots, pp.259-264, 2001.
- [7] 金島義治, 菅原雄介, 安藤俊平, 佐藤基弘, 林憲王, 高西淳夫: "2 足ヒューマノイドロボットの準リアルタイムパターン生成に関する研究", 第 19 回日本ロボット学会学術講演会講演論文集, pp.987-988, 2001.
- [8] K. Nishiwaki, S. Kagami, Y. Kuniyoshi, M. Inaba and H. Inoue: "Online generation of humanoid walking motion based on a fast generation method of motion pattern that follows desired ZMP," Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.2684-2689, 2002.
- [9] K. Löffler, M. Gienger and F. Pfeiffer: "Sensor and control design of a dynamically stable biped robot," Proc. of IEEE International Conference on Robotics and Automation, pp.484-490, 2003.
- [10] K. Harada, S. Kajita, K. Kaneko and H. Hirukawa: "An analytical method on real-time gait planning for a humanoid robot," Proc. of IEEE-RAS/RSJ International Conference on Humanoid Robots, 2004.
- [11] T. Sugihara and Y. Nakamura: "A fast online gait planning with boundary condition relaxation for humanoid robots," Proc. of IEEE International Conference on Robotics and Au-

- tomation, pp.306-311, 2005.
- [12] J. Chestnutt and J. Kuffner: "A tiered planning strategy for biped navigation," Proc. of IEEE-RAS/RSJ International Conference on Humanoid Robots, 2004.
- [13] 下山勲: "竹馬形 2 足歩行ロボットの動的歩行", 日本機械学会論文集 (C 編), vol.48, no.433, pp.1445-1455, 1982.
- [14] M.H. Raibert: Legged Robots That Balance. MIT Press, 1986.
- [15] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi and H. Hirukawa: "Biped walking pattern generation by using preview control of zero-moment point," Proc. of IEEE International Conference on Robotics and Automation, pp.1620-1626, 2003.
- [16] S. Kagami, K. Nishiwaki, T. Kitagawa, T. Sugihara, M. Inaba and H. Inoue: "A fast generation method of a dynamically stable humanoid robot trajectory with enhanced ZMP constraint," Proc. of IEEE International Conference on Humanoid Robotics, 2000.
- [17] K. Kaneko, F. Kanehiro, S. Kajita, H. Hirukawa, T. Kawasaki, M. Hirata, K. Akachi and T. Isozumi: "Humanoid robot HRP-2," Proc. of IEEE International Conference on Robotics and Automation, pp.1083-1090, 2004.
- [18] 金広文男, 藤原清司, 原田研介, 金子健二, 梶田秀司, 横井一仁, 比留川博久: "HRP ヒューマノイドプラットフォームの運動制御システム", 日本ロボット学会創立 20 周年記念学術講演会予稿集, 1D33, 2002.
- [19] 長阪憲一郎, 稲葉雅幸, 井上博允: "体幹位置コンプライアンス制御を用いた人間型ロボットの歩行安定化", 第 17 回日本ロボット学会学術講演会予稿集, pp.1193-1194, 1999.



西脇光一 (Koichi Nishiwaki)

1974 年 5 月 5 日生。2002 年東京大学大学院工学系研究科機械情報工学専攻博士課程修了。博士 (工学)。同年 4 月東京大学大学院情報理工学系研究科科学技術振興特任教員。2004 年 4 月より産業技術総合研究所デジタルヒューマン研究センター研究員となり現在に至る。(日本ロボット学会正会員)



加賀美聡 (Satoshi Kagami)

1970 年 3 月 14 日生。1997 年東京大学大学院工学系研究科情報工学専攻博士課程修了。博士 (工学)。同年より日本学術振興会未来開拓学術研究推進事業「マイクロ・ソフトメカニクス統合体としての高度生体機能機械の研究」プロジェクトにより東京大学リサーチアソシエート。2001 年産業技術総合研究所入所。デジタルヒューマン研究ラボの主任研究員。2003 年改組によりデジタルヒューマン研究センター・ヒューマノイドインタラクションチームのチーム長。ヒューマノイドロボットの認識・計画・制御とそのリアルタイムシステム構成の研究に従事。(日本ロボット学会正会員)