

腕を有する2脚ロボットの制御に関する研究

Control of a Biped Robot that has Arms

甲斐田壮*, 熊谷正朗*, 江村超*

Sou Kaida*, Masaaki Kumagai*, Takashi Emura*

*東北大学

*Tohoku University

キーワード : 2脚ロボット (biped robot), 歩行ロボット (walking robot), 腕 (arm), モーメント (moment)

連絡先 : 〒980-8579 仙台市青葉区荒巻字青葉01 東北大学大学院工学研究科機械電子工学専攻
メカトロニクス講座メカトロニクス設計学分野

甲斐田壮, Tel.: (022)217-6969, Fax.: (022)217-6967, E-mail: kaida@emura.mech.tohoku.ac.jp

1. 緒言

今日, 盛んに2脚ロボットの研究が行われている. 将来, ロボットは人間の日常生活の手伝いや, 福祉活動などのために, 人間の生活環境に入り込んでくると考えられる. そうしたとき, 最適なロボットの移動手段の一つとして人間と同じ2脚移動機構が挙げられる. なぜなら, 人間の生活環境は2本脚で歩行する人間に適したつくりになっているからである.

しかしながら, 2脚移動機構は他のよく用いられている移動機構(車輪, クローラなど)に比べ安定性が低いという欠点があり, 安定性の向上が大きな課題となっている.

人間は歩行をする際, 脚の動きだけでバランスをとっているわけではなく体全体を用いて行っている. そこで今回, 歩行の安定性向上のため, 人間の腕振りに注目した. 人間は腕がないとバランスがとりにくく, 腕の振りにより発生する力をうまく用いて安定した歩行を行っている. これを2脚ロ

ボットに応用することで, 安定した歩行の実現を目指す.

2. Monroe Arm の特徴

江村研究室において以前より研究が行われている2脚ロボット(Monroe)へ, 新たに腕をつけることにより, より安定な歩行を目指すためMonroe用の腕の設計及び製作を行った. 今回設計した腕(Monroe Arm)は, クランクスライダを用い, 1自由度ながら2関節(肩関節, 肘関節)が可動する機構を有する. その機構をFig.1に示す. この機構により, 1自由度ながら歩行時の人間の腕振りに近い肘関節の動きも実現可能となっている. その腕振り軌跡をFig.2に示す.

また腕の利用を考える場合, なるべく大きな力を腕振りにより発生させられることが望まれる. 今回の機構では肘関節が動くことにより, 肩関節のみ可動の腕振りに比べ, より大きな慣性力及びモーメントを発生させることができる.

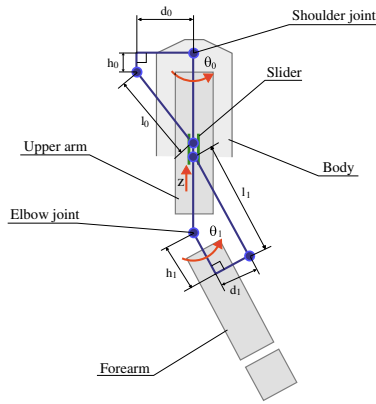


Fig. 1 Monroe Arm の機構

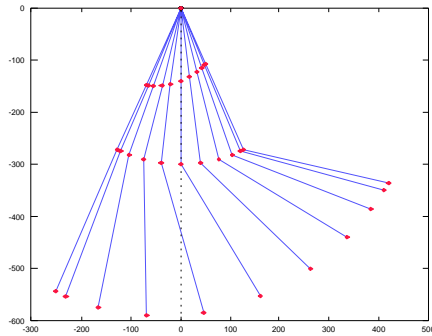


Fig. 2 腕振り軌跡 (Sagittal Plane)

本来はそれぞれの関節に自由度を持たせ制御したほうがより人間に近い腕振りを実現できると考えられる。しかし1自由度あたり最低一つのモータが必要となり、動かすための機構を含めるとかなりの重量となる。ロボット全体の制御を考えたとき、脚部モータのトルクの関係でなるべく軽量にする必要があり、今回腕の自由度は1自由度とした。

3. Monroe Arm の設計

設計した腕の組立図をFig.3に示す。スマートな形状にするため、モータを腕の中に組み込み、モータの回転運動をボールネジで直線運動に変換し、クランクスライダ機構によって関節を駆動している。フレームの材料は主にアルミを使用し、強度を必要とする箇所にはジュラルミンを使用している。腕の寸法は身長1600[mm]の人間をモデルとし、長さ600[mm]で設計しており、腕の重量は片腕あた

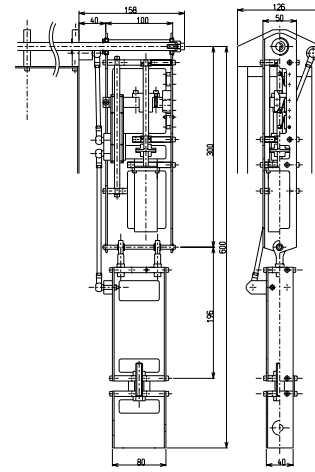


Fig. 3 腕の組立図

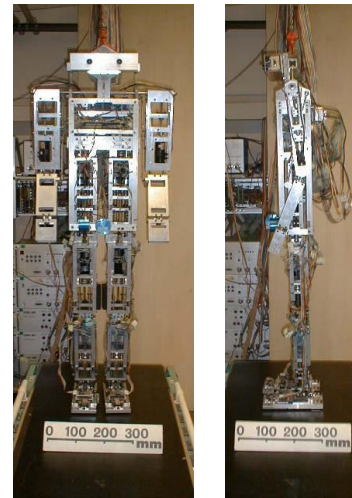


Fig. 4 腕装着時のMonroeの外観

り2.5[kg]と軽量に抑えられている。腕の可動範囲は、ロボット本体に対する上腕の角度(肩関節角度)が後方 $-25[\text{deg}]$ から前方 $25[\text{deg}]$ 、上腕に対する前腕の角度(肘関節角度)が $0[\text{deg}]$ から前方 $52.8[\text{deg}]$ となっている。

実際に製作した腕をMonroeに装着させたときの外観をFig.4に示す。肩幅は駆動機構を腕の中に収めたため若干大きめとなっており、人間の肩幅が約 $450[\text{mm}]$ であるのに対しMonroeは $540[\text{mm}]$ となっている。腕を含めたMonroeの全重量は $27[\text{kg}]$ である。

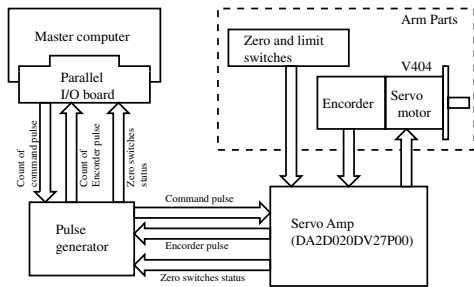


Fig. 5 ハードウェア構成

4. ハードウェアの構成

設計・製作したハードウェアの構成をFig.5に示す。腕の駆動には山洋電気製V404のサーボモータを用い、山洋電気製のDA2D020DV27P00サーボアンプを用いてモータの制御を行う。このアンプはアナログ入力(速度制御)とパルス列入力(位置制御)の2つを選択することができ、今回はパルス列入力を用いている。その入力パルスを生産するのがパルスジェネレータである。このパルスジェネレータはマスタコンピュータとPIOボード経由で接続されており、マスタコンピュータから指令された数のパルス列を生成しアンプへ出力する。またアンプから出力されるエンコーダの2相パルスをカウントして、マスタコンピュータにモータの回転角情報を提供する。これをもとにマスタコンピュータにおいて腕の運動学演算、軌道計画等の処理を行う。

5. 腕制御のソフトウェア

腕を制御するにあたり、腕の軌道計画、運動学演算等のプロセスが必要となる。それらのプロセスをまとめて一つのプログラムとすると非常に複雑になる。そこでMonroeの脚の制御同様、頻繁に書き換えるプログラム(軌道計画等)とそうでないもの(運動学演算等)とに分け、それぞれマルチタスクで実行する。

プロセス間のやりとりは変数をVariableSetに設定し、伝言板デバイス(message board device)上で

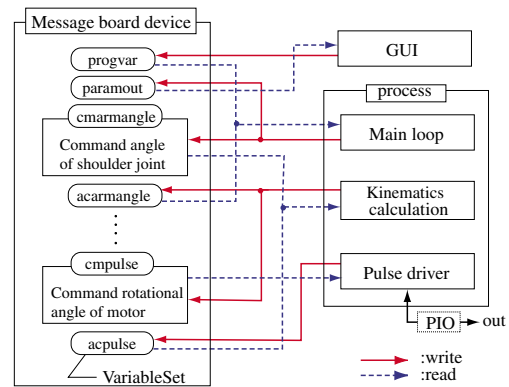


Fig. 6 マルチタスクによる腕の制御

行う³⁾。これにより腕の制御はメインループにおいて、VariableSetの肩関節角度の設定のみを行い、その変数が書き換えられた時点で他のプロセスが変数を受け取り運動学演算等の処理を行う。このプロセスの流れをFig.6に示す。

6. 足軌道の生成

歩行を行う際、足軌道の良し悪しは安定性を大きく左右する要因の一つである。その足軌道は今まで、単位時間に進む軌道の始点と終点を三角関数で補間し生成していた。しかしこの生成手法は、一周期通した軌道のイメージがつかみ辛く、意図したとおりの軌道を生成するのが困難であった。

そこで、図やイラスト作成によく用いられるアプリケーションのIllustrator(Adobe社)を用いて足の軌道を作成する。この方法は、Illustrator上で描きながら軌道を作ることができるので、視覚的にわかりやすく、また複雑な軌道も簡単に作ることができる。

6.1 Bézier曲線

Illustratorの曲線の多くは3次のBézier曲線で描かれる。 n 次のBézier曲線は $t(0 \leq t \leq 1)$ をパラメータとして次式で与えられる¹⁾。

$$P(t) = \sum_{i=0}^n B_{i,n}(t)C_i \quad (1)$$

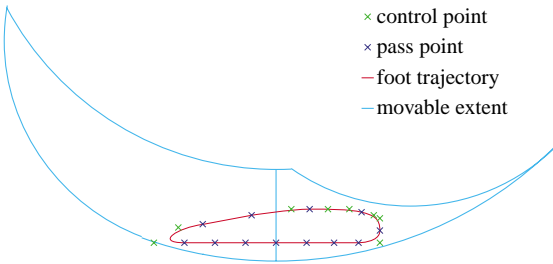


Fig. 7 Illustratorによる足軌道生成(矢状面)

ここで $B_{i,n}$ はBernstein基底関数(Bernstein basis function)と呼ばれるものであり, 次式で表される.

$$\begin{aligned}
 B_{i,n}(t) &= \binom{n}{i} (1-t)^{n-i} t^i \\
 &= \frac{n!}{i!(n-i)!} (1-t)^{n-i} t^i \quad (2)
 \end{aligned}$$

また, C_i は制御点(control point)と呼ばれる任意の位置ベクトルであり, C_i の配置によって曲線の形状が決定される.

6.2 軌道の生成

水平面もしくは矢状面に投影した足軌道をIllustrator上で描く. その際, 複数の3次Bézier曲線, 2次Bézier曲線, 直線(1次Bézier曲線)を組み合わせ, それらを連結させ一周期の足軌道とする(Fig.7). このとき, 分割の仕方, 及び制御点の位置により足の移動速度を調整することができる.

作成した軌道はEPSファイル形式で出力する. そしてこのファイルデータの制御点部分及び線情報のみを取り出し, 別ファイル(制御点ファイル)に出力する. この制御点ファイルをロボット制御時に読み込み, そこでBézier曲線を復元し足軌道を生成する. この一連の流れをFig.8に示す.

7. 着地時の衝撃力対策

歩行の際, 着地時の衝撃により安定性が低下するという現象が見られる. これは, 歩行速度が上がるにつれ顕著に現れ, この衝撃力の吸収は必要不

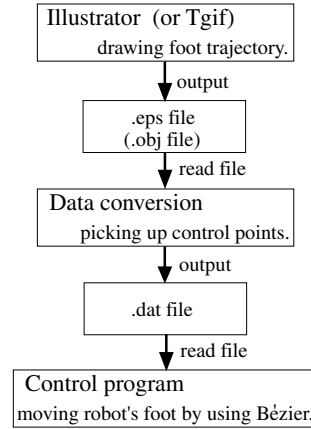


Fig. 8 足軌道生成の流れ

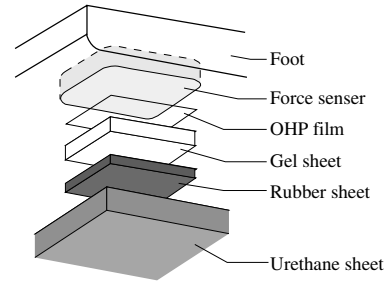


Fig. 9 衝撃吸収構造

可欠となる. その衝撃力を吸収緩和する対策を以下に述べる.

7.1 緩衝材による衝撃吸収

ハード的に緩衝材を用いて衝撃緩和を行う. 緩衝材は, ジェルテック社製の θ ゲルを使用する. θ ゲルとは, 自己保形, 復元性に優れたシリコンをベースとしたゲル状物質(ゲル)の機械的強度を飛躍的に高めた高強度ゲルである.

Monroeの足底には片足あたり6つの力センサが取り付けられており, 床反力はそれらの力センサで全て受け, ZMPなどの計算を行っている. 力センサは大きな撃力が加わるとセンサの材質上ゼロ点が狂うときがあり, これを防ぐことも考慮に入れ, 力センサと床との間に緩衝材を置く. その構造をFig.9に示す.

床と接する面にはウレタンシート(厚さ7[mm])

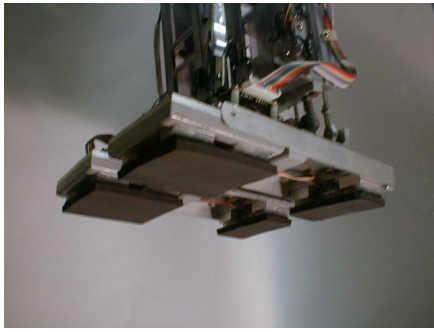


Fig. 10 緩衝材装着後の足底

を複数のゲルをつなぐようにして取り付けられている。ウレタンシートもまた衝撃吸収に優れた素材の一つである。歩行の際、足底と床面との間の摩擦によりゲルには剪断力が働く。一つのゲルでその力を受けると非常に大きな変位となり歩行時の不安定要素となる。そこで複数のゲルを結ぶことにより、この剪断力を複数のゲルで受け剪断方向の変位を抑えている。緩衝材装着後の足底をFig.10に示す。

緩衝材による衝撃吸収の度合いを調べるため、衝撃吸収実験を行った。実験は、重さ2.5[kg]の鉄の重りを高さ50[mm]の所から足底へ自由落下させる。そのときの衝撃力をFig.11に示す。緩衝材無し足底には1[s]、ゲルθ-6を利用した緩衝材着き足底には2[s]、ゲルθ-5の場合は3[s]にそれぞれ衝撃力を与えている。ゲルθ-6、ゲルθ-5はそれぞれゲルの種類であり、硬さや耐久性の観点よりゲルθ-6を緩衝材として採用した。この図から分かるように、緩衝材を足底に装着することにより衝撃力を半分以上に抑えることができる。

7.2 インピーダンス制御

衝撃力を吸収緩和する方法としてインピーダンス制御がある。脚先にFig.12のような、仮想バネ-質量-ダンパ系を置き、脚先にかかる力に対する足首の位置制御を行う²⁾。

この系の運動方程式は f [N]を接地力、 m_d [kg]を目標質量、 d_d [N·s/m]を目標減衰係数、 k_d [N/m]を

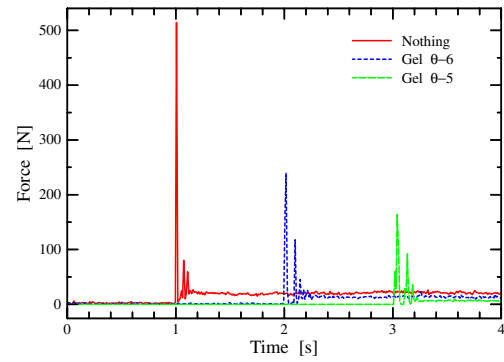


Fig. 11 衝撃吸収実験

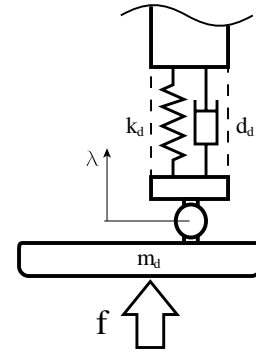


Fig. 12 インピーダンスモデル

目標バネ定数、 λ [m]を脚先変位量として次式で表される。

$$f = m_d \ddot{\lambda} + d_d \dot{\lambda} + k_d \lambda \quad (3)$$

ここで、

$$\begin{cases} \lambda_1 = \lambda \\ \lambda_2 = \dot{\lambda}_1 = \dot{\lambda} \end{cases} \quad (4)$$

とおくことにより、以下の式になる。

$$\begin{bmatrix} \dot{\lambda}_1 \\ \dot{\lambda}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k_d}{m_d} & -\frac{d_d}{m_d} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{f}{m_d} \end{bmatrix} \quad (5)$$

各定数は、経験より目標質量を100[kg]、目標減衰係数を9000[N·s/m]、目標バネ定数を20000[N/m]とした。

また常にインピーダンス制御を行っているとき、単脚支持期の支持脚は自重により脚長が短くなり、その結果安定性が落ちるとともに、遊脚の足付き位置が上がりかえって衝撃力を増大させる恐れがある。そこで着地時の衝撃力を吸収した後は元の脚長に戻す必要がある。

7.3 着地位置補正制御

着地時に大きな衝撃が発生する要因の一つとして、遊脚から支持脚に移行する足が床を蹴り込む現象があると考えられる。足軌道では床との高さが0[mm]になるところで足下げを止めている。しかし実際にはロボット座標系において、脚の剛性により支持脚の接地位置が上にあがり、それに伴い遊脚の着地位置も上昇する。そのため上昇した分、足が床を蹴り込むことになる。

これは、剛性のみならず上体が傾斜するときにも似たような現象が起きる。上体が前方や遊脚側へ倒れ込んだ場合に、遊脚の着地位置が上昇する。この上昇分を補正することにより衝撃力の低減を行った。そのモデルをFig.13に示し、補正值の算出方法を以下に述べる。

ロボット座標系において、床面(水平面)の法線ベクトルを (p, q, r) とするとそれらはロボットの姿勢角 (θ_p, θ_r) より次のように与えられる。

$$\begin{cases} p = 1 \\ q = \tan(\theta_p) \\ r = \tan(\theta_r) \end{cases} \quad (6)$$

床面の固定点より決まる定数を a とすると床面の式が以下のように求まる。

$$px + qy + rz + a = 0 \quad (7)$$

この式より、 $y-z$ 平面での座標 (l_y, l_z) におけるX軸方向の補正值 L_l は次式で求まる。

$$L_l = -\frac{1}{p}(ql_y + rl_z + a) \quad (8)$$

実際はロボットの剛性による変位を考慮し姿勢角 (θ_p, θ_r) にオフセット角 (ρ_p, ρ_r) を引いて床面の式を算出している。そして遊脚の足軌道のx座標を補正值に収束させると、足が床を蹴り込むことにより発生する衝撃力は減少し、歩行時の安定性が向上する。

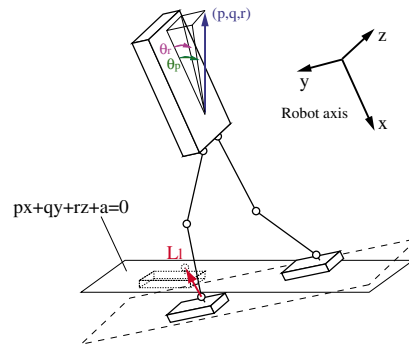


Fig. 13 着地位置補正制御

8. 歩行時の安定化制御

この章では、歩行時の安定性を向上させるための制御について説明を行う。

8.1 ZMPを用いた上体移動制御

歩行中、2脚ロボットの姿勢の崩れ方として2通りの崩れ方が顕著に見られた。一つ目は、ロボット全体が前につんのめる形で姿勢を崩すパターン。もう一つは、後方に倒れこみ支持脚から遊脚への移行が出来ないというパターンである。これらの崩れ方は、前後のバランスの悪さが引き起こすものである。

そこでZMPを用いた上体移動制御を行う。上体位置が前よりの場合、ロボットは前方へ倒れ、逆に上体位置が後よりの場合、ロボットは後方へ倒れ込む傾向がある。この傾向を利用し、倒れ込みをZMPで判断、それに応じ上体を前後へ移動させる。

両足底の力センサより得られる左右それぞれの足底ZMPから、ロボット全体のZMPを算出する。このZMPと両足首を結んだ直線との距離を算出し、その距離に応じて上体を前後に移動させる。

ロボット全体のZMP座標 (x_{zmp}, y_{zmp}) は、右左それぞれの足底ZMPの座標を $(x_r, y_r), (x_l, y_l)$ とし、右左それぞれにかかる加重を f_r, f_l とする、次式のように表される。

$$\begin{bmatrix} x_{zmp} \\ y_{zmp} \end{bmatrix} = \frac{f_r}{f_r + f_l} \begin{bmatrix} x_r \\ y_r \end{bmatrix} + \frac{f_l}{f_r + f_l} \begin{bmatrix} x_l \\ y_l \end{bmatrix} \quad (9)$$

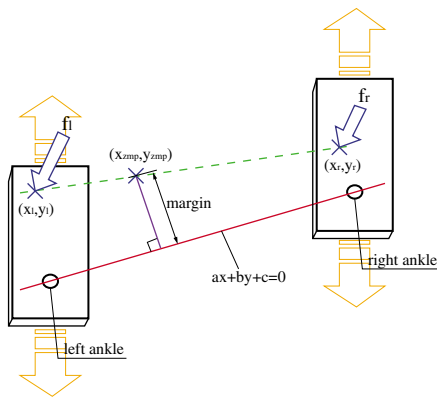


Fig. 14 上体移動制御

このZMPと両足首を結んだ直線との距離をmarginとおき、両足首を結んだ直線を(10)式とすると、(11)式によりmarginは求まる。

$$ax + by + c = 0 \quad (10)$$

$$\text{margin} = \frac{ax_{zmp} + by_{zmp} + c}{\sqrt{a^2 + b^2}} \quad (11)$$

このmarginが大きければ大きいほど安定性が低く、marginの値が低くなる方向に上体を前後させる。正確には足軌道を前後に移動させる。これらをFig.14に示す。

8.2 蹴り動作制御

支持脚だった脚が遊脚に移行する際は、ZMPをもう片方の脚の方へ移動させ、足底にかかる床反力を軽減させなくてはならない。しかし安定性を欠いた歩行の場合、ZMPの移動がうまくできず大きな床反力がかかったまま足上げ動作に移り、転倒する現象が起こる。この現象を抑えるため、遊脚に移行する直前に床面を蹴る動作を加える。

蹴り量は、足底にかかる床反力 f_a [N]と両足首間を結ぶ軸に直交し、床面に含まれる軸回りの倒れ込み角 r_a [deg](遊脚に移行する脚方向への倒れ込み正)により決定する(Fig.16)。仮想荷重を f_k [N]、変換係数を a_{af} とし、次式で f_k を求める。

$$f_k = -(f_a + a_{af} r_a) \quad (12)$$

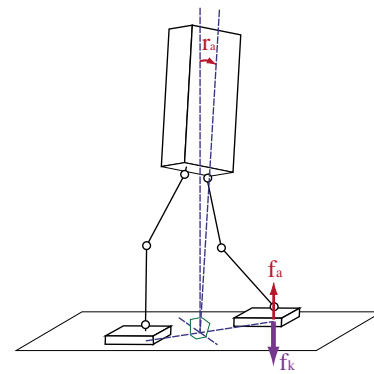


Fig. 15 蹴り動作制御

この仮想荷重 f_k を遊脚に移行する手前で、インピーダンス制御へ負の荷重として入力する。それにより足首位置は下向きに変位し、蹴り動作が実現する。

またこの蹴り動作が入る直前の単脚支持期のRoll傾斜角を用いて蹴り動作時の仮想バネ係数を調整する。Roll方向の揺れが大きい場合、その反動により次の単脚支持期の揺れも大きくなるので、仮想バネ係数の値を大きくし蹴り量を小さくすることにより、揺れを抑制する。

蹴りの動作は前方への勢いを生み、支持脚がロボットの上部を前に押し出すときの負担を軽くする効果も得られる。

8.3 歩幅調整制御

姿勢維持の制御は主に、姿勢が崩れるのを防止する役目大きい。しかし歩行の際、上体を常に目標の姿勢に保とうと制御を行っていても、着地の状況などによりどうしてもバランスを崩すことがある。そこでバランスを崩さないように制御を行うのではなく、バランスが崩れつつある状態からの復帰をさせる制御が必要となる。そういう制御の一つとして、歩行中の歩幅を調整する歩幅調整制御を行う。

歩幅調整制御は単脚支持期、ロボット上部が前方へ倒れ込んだ場合、遊脚の着地位置を前方へ移動させ歩幅を長くする。これにより転倒を防ぎ、両

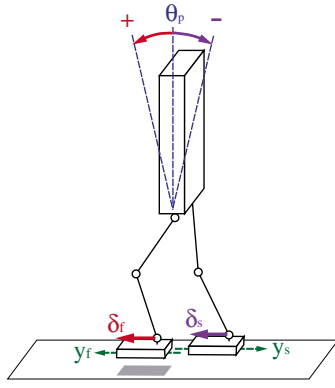


Fig. 16 歩幅調整制御

脚支持期において上体の姿勢を目標に復帰させる。逆に後方へ傾斜した場合は支持脚の移動量を少なくさせ歩幅を短くする。これにより後方への傾斜を抑制する。

移動量をそれぞれ遊脚と支持脚において δ_f, δ_s とし、上体のPitch傾斜角を θ_p とすると移動量は $\theta_p - \rho_p \geq 0$ のとき

$$\delta_f = g_h(\theta_p - \rho_p) \quad (13)$$

$\theta_p - \rho_p < 0$ のとき

$$\delta_s = -g_h(\theta_p - \rho_p) \quad (14)$$

なお g_h は傾斜ゲイン、 ρ_p はオフセット角。この移動量を遊脚、支持脚それぞれの足位置 y_f, y_s に加えることにより、歩幅調整制御が行われる。

9. 腕の利用

左右の腕を対称に振ることによりYaw軸回りのモーメントが発生する。この力を有効に利用することにより、歩行時の安定性を向上させる。

9.1 モーメント抑制制御

歩行速度が速くなるに従い、脚の動きにより発生するYaw軸回りのモーメントが大きくなる。このモーメントは足底と床面との間に滑りを生じさせ、進行方向が変化するとともに歩行時の安定性を大きく低減させる。

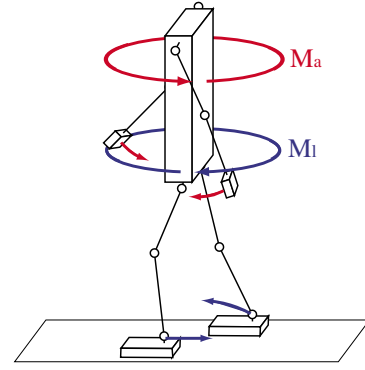


Fig. 17 Yaw軸回りに発生するMoment

そこで脚の動きに同調した腕振りをさせることにより、脚の動きで発生したYaw軸回りのモーメントをキャンセルする方向にモーメントを発生させる。これにより滑りによる安定性低下を抑制することができる(Fig.17)。

9.2 方向転換制御

歩行速度が上がり歩行距離が長くなるにつれ、ロボットの進む向きは直進を保つのが困難となり、左右のいずれかに誤差を生じ始める。今の実験環境は、ベルトコンベア上であり横幅が非常に狭い。進む向きが左右に誤差を生じた場合、その向きを補正しなければすぐにベルトコンベアの使用範囲から外れてしまう。これを防ぐために、熊谷が製作した2脚ロボット用ステレオビジョンによる目標物探査³⁾を利用して、ロボットの方向転換を行う。この方向転換の際、腕振りを変化させロボットの向きを変える方向に作用する力を発生させる。これにより足底に滑りが生じにくいスムーズな方向転換が可能となる。

方向転換は、歩行の一周期を通して行う(Fig.18)。画像の取得は一周期に一回両脚支持期の時に行い、画像処理により目標物の位置(中心からのずれ)を計算する。この位置を v_d とし、位置に比例した角度($\phi_r[deg], \phi_l[deg]$)を求める。

$$\begin{cases} \phi_r = a_c v_d \\ \phi_l = -a_c v_d \end{cases} \quad (15)$$

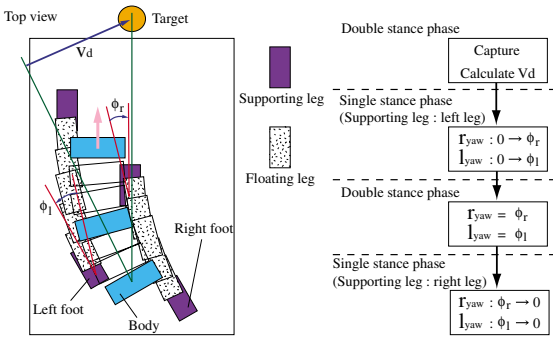


Fig. 18 方向転換制御

なお a_c は変換係数.

Fig.18はロボットの歩行を上から見ている図である. そして塗りつぶしの状態が両脚支持期である. 初めの単脚支持期(左足接地)において, それぞれ足底をyaw軸回りに回転させ $\phi_r[deg]$, $\phi_l[deg]$ に収束させる. そして次の単脚支持期(右足接地)において, yaw軸回りの角度を0[deg]に収束させ方向転換が終了する.

腕の動きは, 指令肩関節角度を右左それぞれ θ_{sr} , θ_{sl} とすると次のようになる.

$$\theta_{sr} = a_s \cos(\pi t) + \rho_{so} - b_g \phi_r \sin(2\pi t) \quad (16)$$

$$\theta_{sl} = -a_s \cos(\pi t) + \rho_{so} - b_g \phi_l \sin(2\pi t) \quad (17)$$

なお a_s は腕振り振幅, ρ_{so} はオフセット角, b_g は方向転換ゲイン, $t(0 \leq t \leq 1)$ は時間パラメータである.

第1項は, 足の動きにより発生するYaw軸回りに発生するモーメントをキャンセルするための項である. そして第3項が方向転換の際に利用するモーメントを発生させる項である. 方向転換制御時の腕振り波形の様子をFig.19に示す. この図におけるモーメントは腕振りにより発生するYaw軸回りのモーメントであり, 右回りが正となっている.

初めの3[s], 4[s]付近において進行方向を右へ方向転換させる制御が行われている. この時, 腕振りが変化し右回りのモーメントを大きく発生させている. このとき足のYaw角を変化させることにより足底の滑りを抑えた方向転換が行われる. 同様に7[s]から9[s]付近にかけては左への方向転換の様子である.

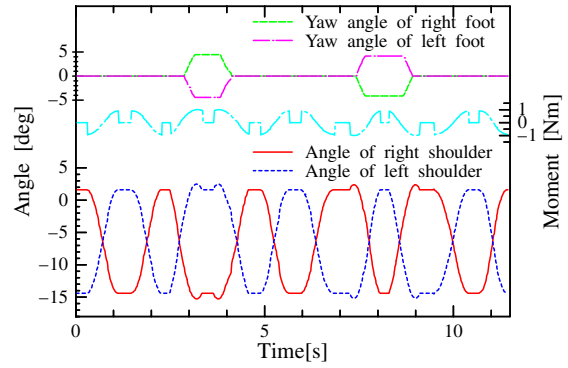


Fig. 19 方向転換時の腕振り波形

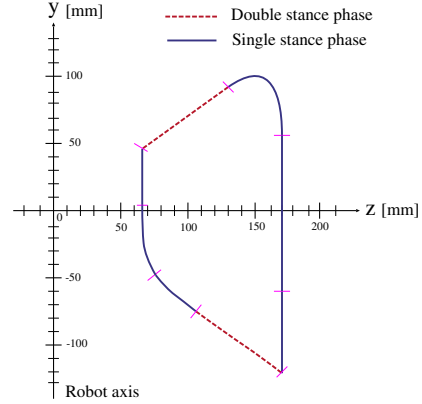


Fig. 20 水平面内における足軌道計画(右足)

10. 歩行実験

前章まで歩行時の制御等について述べてきた. それらの制御を用いてデューティ比5/8, ストライド長約330[mm], 周期2.3[s], 歩行速度に直すと約145[mm/s]の歩行を行わせる. このときの足軌道計画(右足)をFig.20に示す.

デューティ比が大きい場合, 両脚支持期において単脚支持期の運動エネルギーを消そうとする動きが生まれる. 一方デューティ比を向上させ値が小さくなるほど, 単脚支持期の運動エネルギーを次の動作にうまく受け継ぐ必要がある. そのことを意識し, 単脚支持期から両脚支持期への軌道の移行は滑らかに行われるよう軌道計画を行っている.

またこの軌道計画の特徴として, 両脚支持期においてロボットの重心位置が両足首を結んだ軸上をほぼ移動するように計画している. これにより

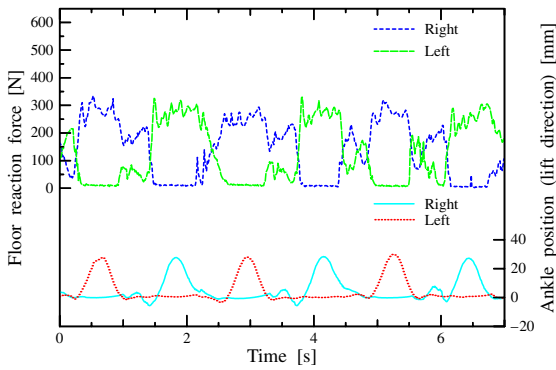


Fig. 21 床反力の時間変化

両足首を結んだ軸回りの倒れ込みを抑制することができ、安定性の向上が図られている。

10.1 実験結果

実際に歩行させたときの床反力の時間変化を Fig.21 に示す。この図より着地時の衝撃が非常に小さく抑えられているのが分かり、着地時の衝撃力対策が有効に働いていることが確認できる。

次に上体姿勢角の時間変化を Fig.22 に示す。Pitch 角は前方倒れ込みが正、Roll 角は右側倒れ込み正となっている。この図より $-5[deg]$ から $+5[deg]$ と大きめに姿勢角は推移しているが、発散せずに制御がなされている。

肩関節角と足Yaw角の時間変化を Fig.23 に示す。ロボットがほぼ直進したため、方向転換のための腕振り変化は殆ど見られないが、足Yaw角の変化が若干ではあるが見られ、方向転換制御が行われていることが分かる。

11. 結言

2脚ロボットに腕をつけるにあたり、ハードウェアの腕本体、サーボアンプ等の設計及び製作、制御のための運動学演算処理等を含むソフトウェアの作成を行った。それらの環境を用いて、安定性向上のための各種制御、及び腕振りで発生するYaw軸回りの力を利用した制御により、歩行速度約 $145[mm/s]$

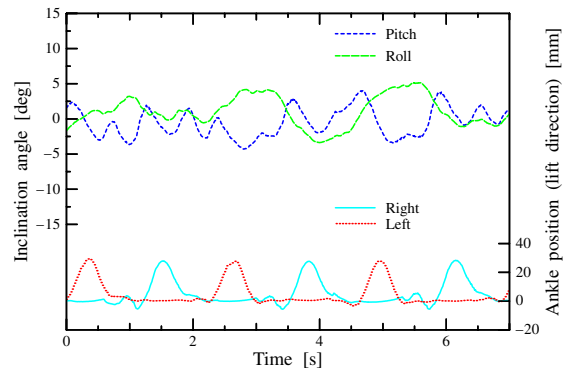


Fig. 22 上体姿勢角の時間変化

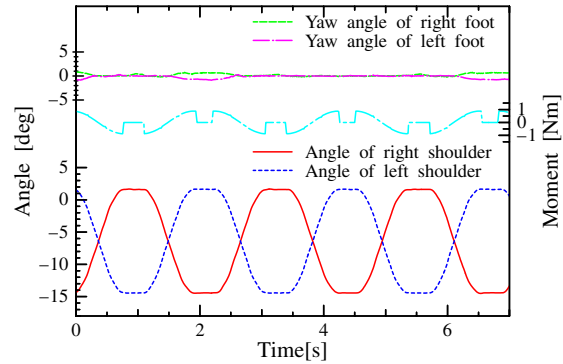


Fig. 23 肩関節角と足Yaw角の時間変化

の動歩行を実現した。

今後の課題として、実現した動歩行はまだ安定性が高いと言い切れる段階ではなく、数歩歩行した所で安定性を著しく崩すケースが見られる。今後さらなる安定性向上が課題である。

参考文献

- 1) 香美 義幸: プレイバック自立走行の基礎的研究, 東北大学大学院修士学位論文, 18/21(1997)
- 2) 森谷 慎司: 4脚ロボットの斜面歩行に関する研究, 東北大学機械電子工学科卒業論文, 29/30(1998)
- 3) 熊谷 正朗: 人型2脚ロボットの時変傾斜面への適応に関する研究, 東北大学大学院博士学位論文, 68/86 125/132(2000)