

平成22年度 修士論文

バーチャルクリーチャのための
関節のシミュレーションモデル

電気通信大学大学院 博士前期課程

電気通信学研究科 知能機械工学専攻

横井研究室

0934053

花岡 優介

主任指導教員 : 横井 浩史 教授

指導教員 : 下条 誠 教授

目次

1	序論	1
1.1	研究背景	1
1.2	本研究の目的	2
1.3	物理シミュレータ	3
1.3.1	ニュートンオイラーの運動方程式に基づく多体剛体運動 シミュレータ	4
1.3.2	ロボット制御のシミュレータ	5
1.4	関連研究	6
1.5	解析法の物理シミュレータ	8
1.5.1	LCPの反復解法	13
1.5.2	拘束条件の相補性	14
1.5.3	ばねダンパモデルの扱い	16
2	提案	18
2.1	可動域の指定	19
2.2	可動域の内外判定	22
2.3	速度の拘束	23
3	実装	26
3.1	可動域の設定	26
3.2	現在の関節姿勢の計算	29
3.3	可動域内にあるかの判定	31
3.4	近傍姿勢の計算	34
3.5	拘束軸の計算	34
3.6	関節可動域の拘束力の計算	36

3.7	拘束力の計算	37
4	評価	39
4.1	提案手法の有効性の確認	39
4.1.1	人の関節可動域の計測	40
4.1.2	動きの評価	41
4.1.3	結果	41
4.2	キャラクタモデルと物理シミュレーション	46
4.2.1	結果	46
5	結論	48
5.1	今後の展望	48
	参考文献	49
	謝辞	51

目次

1.1	物理エンジンによるシミュレーション	2
1.2	不自然な姿勢をとるキャラクタ	3
1.3	球関節の表現	7
1.4	球関節可動域範囲	7
1.5	SwingTwist角が一意に決まらない例	8
1.6	時崎らの球関節の仕様	8
1.7	物理シミュレータ	9
1.8	剛体間の拘束座標軸	10
1.9	等式拘束	12
1.10	相補性拘束	12
2.1	関節可動域拘束の手順	18
2.2	スイングツイスト角座標系	20
2.3	関節可動域	21
2.4	関節可動域の例	22
2.5	拘束軸の取り方	24
2.6	剛体の接触	25
3.1	スイングツイスト角座標系	27
3.2	指定できない可動域	28
3.3	拘束の判定	32
3.4	ヤコビアンの中身	36
4.1	可動域を入れた人型モデル	39
4.2	マーカの配置	41
4.3	計測した関節可動域	42

4.4	関節可動域の組み込み	42
4.5	バーチャルヒューマンでの関節可動域表現	43
4.6	ポインタを動かした時の様子(拘束あり)	44
4.7	ポインタを動かした時の様子(拘束なし)	45
4.8	くまモデル(右), 物理モデル(左)	46
4.9	くまモデルの動き, 可動域拘束あり(左), なし(右)	47

第1章

序論

1.1 研究背景

コンピュータゲームやVR(バーチャルリアリティ)のアプリケーションにおいて3Dキャラクタの使用が増えている。キャラクタは外見や動きなどによってユーザをバーチャル世界に引き込むことや、ゲーム内の世界観を表現する役割を持っている。

これらのキャラクタの動きの多くはキーフレームアニメーションによって生成されている。この方法は、キャラクタの一連の動きに対して適当な間隔でその時刻での姿勢を指定し、時刻に応じて姿勢間の補間をすることで動作の生成をする手法である。この手法では、キャラクタの姿勢は、一つ一つ人が指定しなければならない。そのため、動作全体を指定するために非常に手間がかかる。また、キーフレームアニメーションでは予め用意した動作をつなぎ合わせたり、補間したりして再生することしかできない。このため多様な動作を作り出すためには、たくさんの動作を用意せねばならず、大変な手間がかかってしまう。

近年コンピュータゲームでは、任天堂のWiiやMicrosoftのKinectのような身体性を持った入力インターフェースが使用されるようになっており、ユーザの入力に自然に反応するキャラクタを実現するためには、多様なインタラクションに対応した多様な動作のリアルタイム生成が必要となっている。そこでリアルタイムに動作生成をおこなう手法として物理シミュレーションを用いる手法[1]が期待されている。物理シミュレータはキャラクタを剛体と関節によってモデル化している。このキャラクタの関節にトルクを与えることによって動作生成をおこなうことができる。

ところで、人など生物の関節には骨や腱の構造によって関節の動きに制約

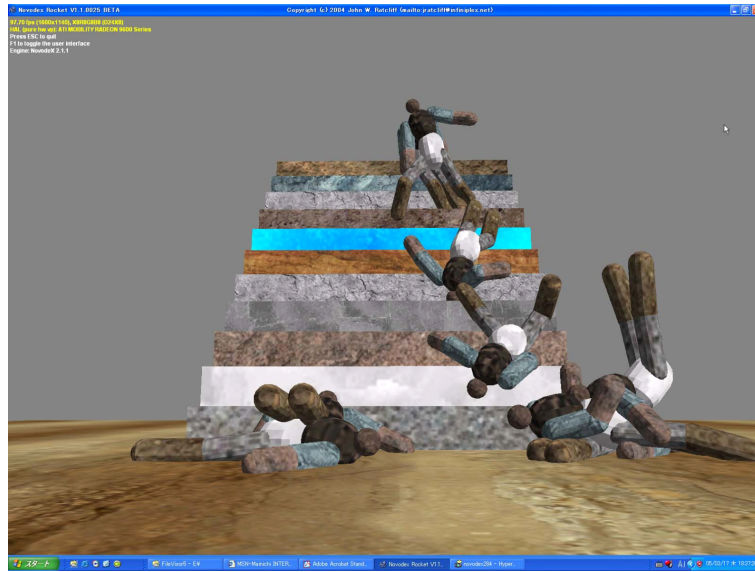


図1.1 物理エンジンによるシミュレーション

がある．この制約を満たさない姿勢は不自然に見える．そこで，シミュレータ内で可動域を再現することによってキャラクターが不自然な姿勢を取ることが防ぐことが考えられる．

生物の関節は非常に複雑な可動域を持っている．例えば人間の肩関節は複数の骨や腱によって作られている関節である．このような複雑な動きをする関節はリアルタイムシミュレータでは球関節によって近似されている．本来肩の関節は並進運動と回転運動を行う複合関節である．しかし，肩の動きはもともと回転の3自由度によるものなので，球関節として表現しても大きな誤差はない．しかし，従来の物理シミュレータでは球関節に対して複雑な可動域を設定することができず，球関節に近似した肩関節の可動域を再現することができなかった．そのため，キャラクターは不自然な姿勢をとってしまうことがあった．

1.2 本研究の目的

剛体と関節によって表現されるバーチャルクリーチャが不自然な姿勢を取らないようにするためには，複雑な動きをする関節に対して，関節の可動

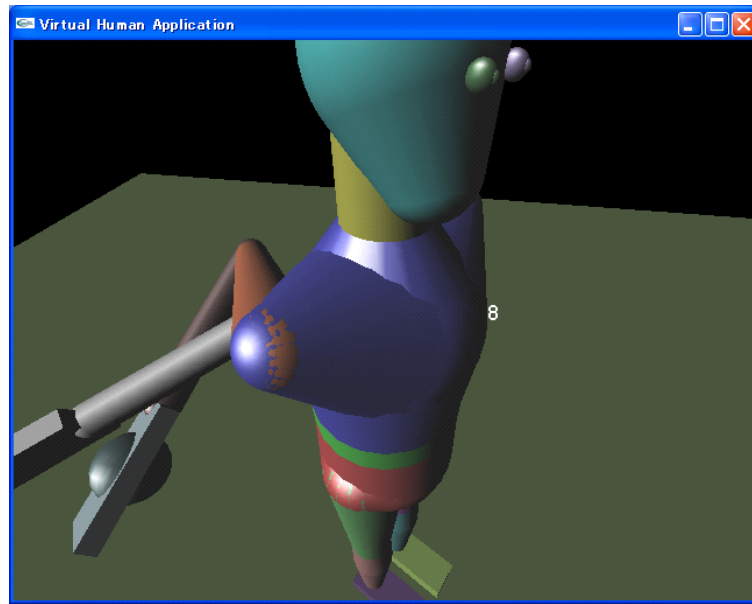


図1.2 不自然な姿勢をとるキャラクタ

域を再現すること必要となる。

本研究ではリアルタイム物理シミュレータにおいて、球関節に複雑な可動域を指定し可動域を拘束する手法を提案する。キャラクタの関節可動域を再現し、物理シミュレーションされたキャラクタが、不自然な姿勢をとらないようにすることを目的とする。

1.3 物理シミュレータ

物理シミュレータは、現実世界の物理法則に基づいて運動を計算することで、バーチャル世界にある物体に、現実世界と同様の振る舞いをさせるものである。対象や目的などに応じて、モデル化や計算の手法が異なる様々なシミュレータが提案されている。例えば、機構部品が破損しないことを確認するためには、有限要素法を用いた応力のシミュレータが、水の流れを予測するためには、ナビエ・ストークス方程式と連続の式でモデル化した流体の運動のシミュレーションが用いられる。キャラクタのように関節でつながった剛体を扱うものは複数の剛体を扱う多体動力学シミュレータと呼ばれる。多体

動力学シミュレータには、ラグランジュの運動方程式を数値積分するもの、ニュートンオイラーの運動方程式を数値積分するものなどがある。ラグランジュの運動方程式を剛体と関節の接続関係から自動的に計算するのは少し難しいため、任意の機構について自動的にシミュレーションを行うためにはニュートンオイラーの運動方程式を数値積分する手法が用いられることが多い。本研究が対象とするキャラクタのシミュレーションでも、ニュートンオイラーの運動方程式を数値積分するシミュレータが用いられることが多い。

1.3.1 ニュートンオイラーの運動方程式に基づく多体剛体運動シミュレータ

ニュートンオイラーの運動方程式は、拘束の無い剛体の運動を表すものである。しかし、キャラクタ動作の生成のための多体の剛体運動のシミュレーションでは、接触や関節といった運動拘束の下での運動をシミュレーションする必要がある。そこで、各剛体に拘束条件を満たすような拘束力が生じたと考え、拘束力を求めた後、ニュートンオイラーの運動方程式に拘束力を加えて数値積分を行うことでシミュレーションを進める。

剛体と関節を自由に配置した対象をシミュレーションをすることができる。このようなシミュレータは拘束条件を満たすような力(拘束力)の計算方法によって次のように分類することができる。

1. ペナルティ法 [4]

ペナルティ法では接触点や関節などの拘束の位置で剛体と剛体の間をばねダンパモデルでつなぎ、ばねダンパモデルが物体に加える力を拘束力としてシミュレーションをおこなう。バネダンパの伸びに基づいて拘束力を計算するため、拘束力を求めるためには、その時刻での剛体の位置・速度が分からなければならない。このため、数値積分前の時刻の拘束力に基づいて数値積分を行わざるを得ず、前進積分になってしまう。このため、ばねダンパモデルの伸びが大きくなり拘束力が大きくなると積分誤差が大きくなり、積分誤差が系のエネルギーを増すよう

に働くため、シミュレーションの安定性が損なわれる。そのため、高速に更新しなければならず時間刻みを大きくとることができない。そのため精度よくシミュレーションすることができるが計算量を増やすことができない。

2. 解析法 [5]

解析法のシミュレータでは剛体間の拘束力を運動方程式と拘束の条件式を連立させることで求める。多数の剛体が互いに拘束される場合、連立させるべき式の数に拘束の数に比例して多くなるため、厳密解を求めるには非常に時間がかかる。そのため、ガウスザイデル法などの繰り返し計算によって解を求めることでリアルタイム性を確保することが多い。ペナルティ法に比べてシミュレーションステップあたりの計算量は多くなる。しかし、拘束力を計算する際の拘束条件の式を、数値積分後の位置・速度について立てることで、後退積分とすることができ、系のエネルギーを減らす側に積分誤差が出るようにすることができる。このため、更新周期による安定性の低下が少ないためシミュレーション1ステップの時間刻みを大きく取ることができる。そのため単位時間あたりの計算量を減らすことができ、リアルタイムで多くの物体を扱う場合など計算量を減らす必要がある場合に用いられる。

解析法の多体動力学シミュレータの拘束の条件式は上手く定式化すると、線形相補性条件とすることができ、拘束力を求める問題は線形相補性問題(LCP)に帰着することができる。例えばSmithらの制作したOpen Dynamics Engine[7]や、長谷川らによって制作されたSpringhead2 [6]、Erlebenらの制作したOpenTissue[8]などで使用されている。本研究の実装では解析法の多体動力学シミュレータであるSpringhead2を用いる。

1.3.2 ロボット制御のシミュレータ

ロボットの動きのシミュレーションにはラグランジュの運動方程式によるシミュレータが広く使われている。これはロボットの機構ごとにラグランジュ

の運動方程式を用いて一般化座標の式を立ててシミュレーションをおこなうものである。この手法ではロボットの自由度の数だけの変数で式を立ててシミュレーションをおこなうことができるため、拘束力を求める必要がなく、少ない計算量でシミュレーションをおこなうことができる。しかし、ロボットの機構が外部に接触する場合など拘束が増加する場合には拘束力を別途外力として計算する必要がある。また、機構ごとにラグランジュの運動方程式が異なるため機構ごとに運動方程式を立て直す必要がある。

1.4 関連研究

本研究では、動力学シミュレータにおいて球関節に複雑な可動域拘束を与える。そのため、球関節に複雑な可動域拘束を指定する研究や、動力学シミュレータで球関節に可動域を与える研究が関連研究としてあげられる。

山根ら [11] は、ヒューマンフィギュアの全身運動生成のための協応構造化インターフェースの中で、球関節の可動範囲を設定している。

この手法は、図1.3のように関節の方向を表す2つのパラメータとひねり角を表す1つのパラメータを用いて、球面関節可動範囲を表現している。このときの d_i^0 は関節の中心であり、 a_i は d_i^0 に鉛直な平面に含まれるベクトルで、回転角を γ_i とした時に $\sin(\frac{\gamma_i}{2})$ を大きさとして持っている。またひねり角を (α) と定義している。この定義での球面関節が取りえるすべての範囲は、 d_i^0 を軸とする半径1高さ 2π の円柱として表現される(図1.4)。この円柱の範囲のうち可動範囲を、三角柱の集合として定義与える。

拘束条件の決定では、 $a_y - a_z$ 平面上の可動域を三角形の集合として表し、現在の姿勢の x, y を含む三角形が存在するかを調べる。存在していない場合には可動域範囲外となる。存在していた場合、さらにひねり角を調べて可動域範囲に入っているのかを調べる。

可動域範囲外にあるときには、可動域範囲内戻す必要がある。戻す位置として現在の姿勢に最も近い可動域内の姿勢が理想だが計算量が多くなってしまふ。そのため関節の中心である d_i^0 に戻す速度を加えることで拘束を行っている。

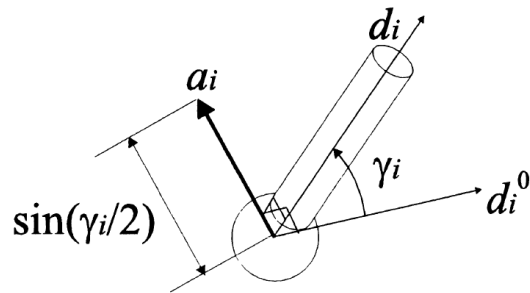


図 1.3 球関節の表現

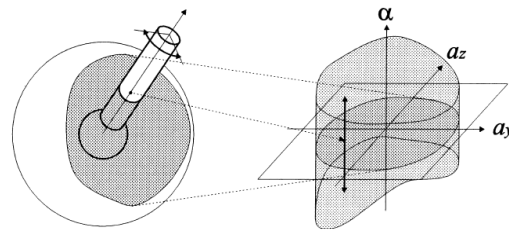


図 1.4 球関節可動域範囲

山根らの研究では，CGアニメーション制作ツールを構築することを目的として研究されているため動力学は考慮されていない．この他にも芹澤ら [12] や Paoloら [13] がモーションエディタやアニメーション制作ソフトにおいて球関節に可動域拘束を与える手法を研究している．

時崎ら [14] は動力学シミュレーションでの球関節の可動域拘束を実現している．可動域はオイラー角の一種であるスイングツイスト角を用いて与えられる．このスイングツイスト角で関節角度を一意に表すためにそれぞれの角度に対して定義域を設ける．関節の姿勢が一意に決まらない例として図 1.5 が挙げられる．(A) の姿勢から，(β) を -90° ，(γ) を 180° ，(α) を 180° と回転させると (D) の姿勢になる．(A) と (D) は指定する角度は異なるが同じ姿勢になる．

また，スイング角が $0^\circ, 180^\circ$ のときには特異点となりスイング方位角によらずに同一の姿勢をとる．そのため関節で結ばれた 2 剛体の姿勢から，スイング方位角を一意に定めることができず，スイング方位角で可動域を拘

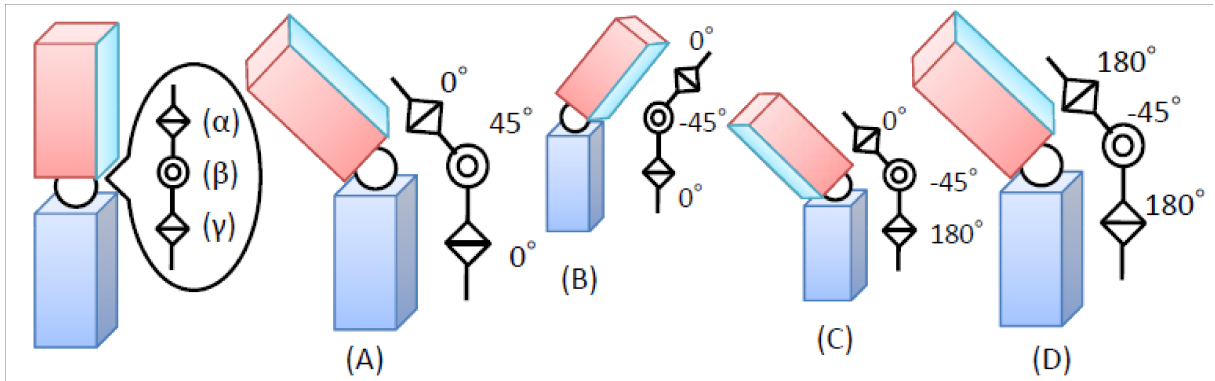


図 1.5 SwingTwist 角が一意に決まらない例

束することができない．そこで時崎らはスイング方位角は考えずに，スイング角とツイスト角だけに可動域拘束を加える手法を提案している．この可動域指定では，スイング角，ツイスト角それぞれにしか可動域を指定できないため，図 1.6 のような円錐形での可動域指定となる．

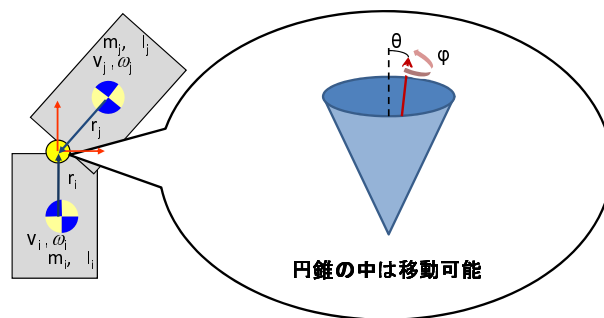


図 1.6 時崎らの球関節の仕様

1.5 解析法の物理シミュレータ

本研究ではニュートンオイラー運動方程式で剛体の運動を計算する解析法の多体動力学シミュレーションを扱う．運動方程式は，剛体に加わる力が剛体の加速度を決定することを意味するので，剛体の運動は力を求めることで計算できる．剛体に加わる力には，重力やバネダンパの力など，現在の

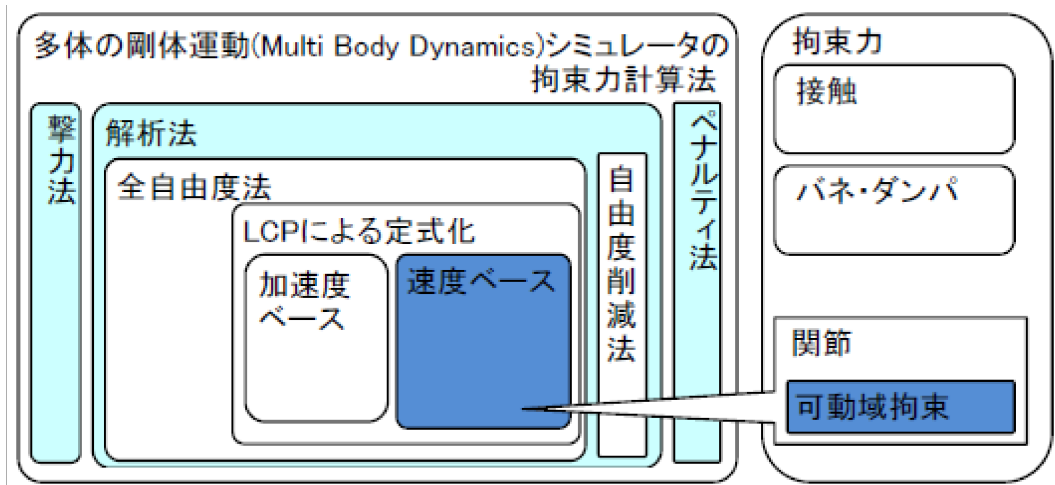


図1.7 物理シミュレータ

状態から直接求まるものもあるが、接触や、関節にかかる力などは簡単に求めることができない。これは、接触点に働く力や関節にかかる力は、接触点で2剛体が互いに侵入しないこと、2剛体の関節の位置がずれないことといった接触や関節を維持するように働くためである。接触や関節などは、2剛体が互いに侵入しない、位置がずれないなどのように、2剛体の間の幾何的な拘束条件と捉えることができる。そのため、このような力は拘束力と呼ばれる。関節の拘束条件は等式で表すことができる。接触や摩擦の拘束条件は、不等式と条件分岐を含む式となるが、上手く変形すると線形相補性条件の形で定式化することができる。そのため、拘束力を求める問題は、線形相補性問題に帰着させて計算することができる。ばねやダンパによる力は、状態から求まる力だが、後退積分の枠組みに組み込むことで安定性を確保するために、拘束条件として、物理シミュレータに組み込むことが提案されている [15]。

拘束条件は、2つの剛体の運動を拘束する。拘束条件と2剛体の運動を記述するため、ここで次のように座標系を定義する。図1.8のうち、オレンジの座標系が剛体の座標系、緑の座標系が拘束座標系となる。このとき、各座標軸につき、並進運動と回転運動を持つため速度、拘束力の変数はそれぞれ6変数になる。剛体の座標系では、剛体のニュートンオイラーの運動方

程式は次のように書ける .

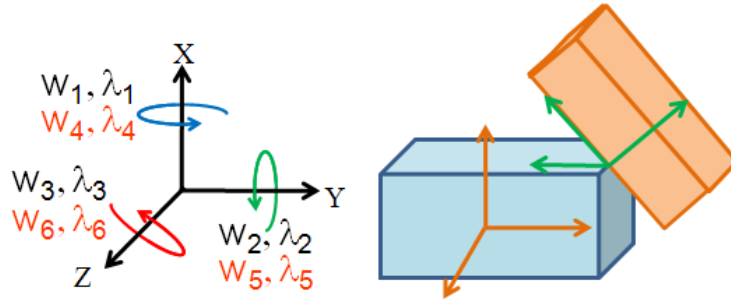


図 1.8 剛体間の拘束座標軸

$$\begin{bmatrix} m_i E_{33} & 0 & 0 & 0 \\ 0 & I_i & 0 & 0 \\ 0 & 0 & E_{33} & 0 \\ 0 & 0 & 0 & I_j \end{bmatrix} \begin{bmatrix} \dot{v}_i \\ \dot{\omega}_i \\ \dot{v}_j \\ \dot{\omega}_j \end{bmatrix} = \begin{bmatrix} f_{ci} \\ \tau_{ci} \\ f_{cj} \\ \tau_{cj} \end{bmatrix} + \begin{bmatrix} f_{ei} \\ \tau_{ei} \\ f_{ej} \\ \tau_{ej} \end{bmatrix} \quad (1.1)$$

$$M \dot{u} = f_c + f_e \quad (1.2)$$

ここで m : 剛体の質量 , I : 慣性テンソル , v : 並進速度 , w : 角速度 , $f_c \tau_c$: 拘束力 , $f_e \tau_e$: その他の外力 , E_{33} : 単位行列である . 2 剛体を表したために , 添え字 i, j をつける . 式 1.2 は , この式を行列を用いて纏めて書いたものである . 剛体の質量・慣性力を M , 並進速度・角速度を u , 拘束力 f_c , その他の外力が f_e となる . 同様に , n 個の剛体の運動方程式は次のように書ける .

$$M = \begin{bmatrix} M_1 & & & & \\ & M_1 & & & \\ & & \dots & & \\ & & & & M_n \end{bmatrix}, \mathbf{u} = \begin{bmatrix} v_1 \\ w_1 \\ v_2 \\ w_2 \\ \vdots \\ v_n \\ w_n \end{bmatrix}, \mathbf{f}_c = \begin{bmatrix} f_{c1} \\ \tau_{c1} \\ f_{c2} \\ \tau_{c2} \\ \vdots \\ f_{cn} \\ \tau_{cn} \end{bmatrix}, \mathbf{f}_e = \begin{bmatrix} f_{e1} \\ \tau_{e1} \\ f_{e2} \\ \tau_{e2} \\ \vdots \\ f_{en} \\ \tau_{en} \end{bmatrix} \quad (1.3)$$

$$M_i = \begin{bmatrix} m_i \mathbf{E}_{33} & 0 \\ 0 & \mathbf{I}_i \end{bmatrix} \quad (1.4)$$

$$M\dot{\mathbf{u}} = \mathbf{f}_c + \mathbf{f}_e \quad (1.5)$$

$$\mathbf{u}[t+1] = \mathbf{u}[t] + M^{-1}\Delta t\mathbf{f}_e + M^{-1}\Delta t\mathbf{f}_c \quad (1.6)$$

この運動方程式と拘束力を計算する拘束の条件式から剛体の速度を計算する。

t は離散化された時刻を表し、時刻 t の値を $[t]$ で表記する。また、 h をシミュレーションの積分ステップとする。また、 λ は拘束座標系(図1.8オレンジの座標系)での拘束力となる。シミュレータは各時刻 t において以下の手順に示す計算がおこなわれる。

1. 拘束力 $\lambda[t]$ の計算(詳細は後述する)
2. 速度の更新

$$\mathbf{u}[t+1] = \mathbf{u}[t] + M^{-1}(\mathbf{f}_e[t] + \mathbf{f}_c[t])h \quad (1.7)$$

3. 位置・姿勢の更新

時刻 t における剛体の位置を $p[t]$ 、姿勢を表すクォータニオンを $q[t]$ とし、剛体の慣性テンソルを I とすると、

$$p[t+1] = p[t] + v[t+1]h \quad (1.8)$$

$$q[t+1] = q[t] + I^{-1}\omega[t+1]h \quad (1.9)$$

このような手順で剛体の位置・姿勢を更新していきシミュレーションを進めていく。剛体の速度を計算するためには拘束力を求める必要がある。運動方程式だけでは速度と拘束力の2変数を求めることができない。そこで運動方程式に加えて拘束条件を考える必要がある。

拘束条件には関節や接触などが考えられる。関節は拘束する軸と自由に動く軸を作るため、 $w_i = 0$ (拘束する軸)、 $\lambda_i = 0$ (自由に動く軸)のように等式条件であらわすことができる。これに対して接触は、剛体同士がめり込む方向に力が働くときには侵入できないため速度が0になり拘束力が働く。また、剛体が離れる方向に力が働く時には自由に運動できるため拘束力が0になり速度が出る。そのため、 $w > 0 \cap \lambda = 0$ または $w = 0 \cap \lambda \geq 0$ のように2つの不等式で表すことができる。この2つの不等式は片方が動く時にはもう一方の値が0になっており、実質的な変数の数を一つに制限する式になっている。このような条件は線形相補性条件と呼ばれ、解法が知られている。

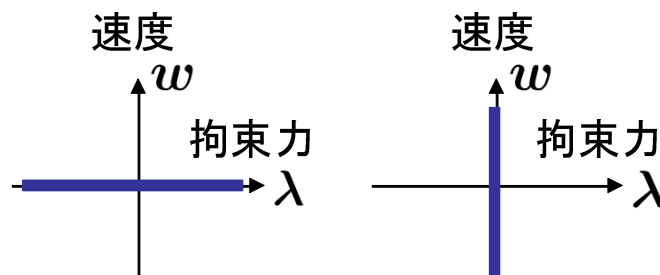


図1.9 等式拘束

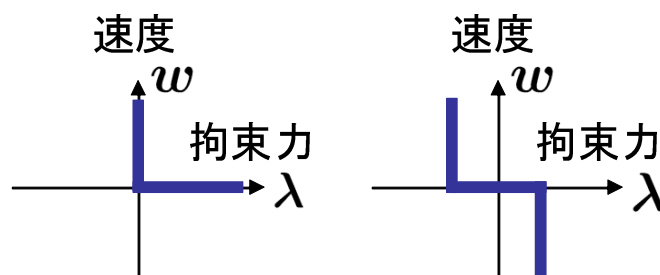


図1.10 相補性拘束

解析法のシミュレータではこの拘束の条件式と式 1.6 の運動方程式を連立させ、繰り返し計算によって計算することで拘束力 λ を求める。このとき式 1.6 では速度と拘束力の座標系が拘束条件の座標系と異なるためこのままでは運動方程式と拘束条件式を連立することができない。そこで、次のように式変形をおこなう。

$$M\dot{u} = f_c + J\lambda$$

$$JM\dot{u} = Jf_c + \lambda \quad (1.10)$$

$$M\dot{w} = Jf_c + \lambda \quad (1.11)$$

J は剛体の座標系から拘束条件の座標系へ変換するヤコビアンであり、 $w = Ju$ 、 $f_c = J^T\lambda$ となる。このように座標変換することで運動方程式と線形相補性条件 (LCP) を各成分ごとに連立して解くことができる。

1.5.1 LCP の反復解法

LCP として表された拘束の条件式はガウスザイデル法を用いて反復解法で求められることができる。以下、本節ではガウスザイデル法による解法を解説する。拘束条件を組み込んだ運動方程式は次のようになる。

$$\dot{w} = M^{-1}Jf_c + M^{-1}\lambda \quad (1.12)$$

$$w[t+1] = M^{-1}\lambda + w[t] + M^{-1}Jf_c \quad (1.13)$$

$$w[t+1] = A\lambda + b \quad (1.14)$$

このとき A は定数行列であり、 b は定数ベクトルである。また、 w や λ は式 1.6 の u や f_c を拘束座標系へ変換するヤコビアン J によって座標変換したものである。ガウスザイデル法ではこの式を次のように変形する。

$$w[t+1] = (D - F)\lambda + b \quad (1.15)$$

$$D\lambda = F\lambda + w[t+1] - b \quad (1.16)$$

$$\lambda = D^{-1}F\lambda + D^{-1}(w[t+1] - b) \quad (1.17)$$

$$(A = D - F)$$

このとき D は対角行列のような簡単な行列とする。式 1.17 を念頭に拘束力 λ についての漸化式

$$\lambda^{i+1} = D^{-1}F\lambda^i + D^{-1}(w[t+1] - b) \quad (1.18)$$

を考える。この漸化式が収束する場合、 λ は、式 1.17 を満たす。このことから、漸化式が収束する場合、 λ^n が、求めたい λ に近づいていくと考えられる。この漸化式は、行列 A が正定値の場合、収束することが知られており、行列 A はその定義から正定となる。そこで、この漸化式を計算することで、 λ を求める。

この解法は繰り返し計算によって近似解を求めるものであり、拘束力の計算精度は反復回数に依存するものとなる。LCP 条件では、 λ についての不等式制約がある。これを反映させるために、 λ^{i+1} を求めるまえに、 λ^i が不等式制約を満たすか確かめ、満たさない場合は、条件をみたすような λ^{i+1} を求め、これを式 17 に代入して、 λ^{i+1} を求める。例えば図 1.10(左) のような接触などで使われる相補性拘束の場合、 $\lambda^i < 0$ のときに $\lambda^i = 0$ とする。これにより拘束力が求められ、この拘束力から剛体の速度が計算できる。

1.5.2 拘束条件の相補性

本節では、拘束条件を等式条件か相補性条件として記述できることを示し、拘束力計算が LCP に帰着できることを示す。剛体の拘束運動を LCP へ帰着する方法を述べる。剛体群に作用する拘束力は等式拘束と相補性拘束に分けることができる。等式拘束は主に関節による剛体の連結を表現するために用いられる。相補性拘束は主に剛体同士の接触の表現に用いられる。

ここで等式拘束および相補性拘束によって拘束される速度ベクトルを w_c とおく。また、これに対応する拘束力ベクトルを λ とおく。これらに適切なヤコビアン J を用いて

$$w_c[t] = J[t]v[t] \quad (1.19)$$

と表現できる。このとき $v[t]$ はある時刻 t のときの剛体の速度を表す。

剛体の運動は常微分方程式である Newton-Euler の運動方程式に基づき以下のように記述される。

$$M\dot{v} = f[t] + J[t]^T \lambda[t] \quad (1.20)$$

ここで M は質量行列、 $f[t]$ は外力とコリオリ項を含むベクトルである。また、拘束条件は次のように記述される。

$$w_c[t] = 0 \text{ (等式拘束)} \quad (1.21)$$

$$w_c[t] \geq 0, \lambda[t] \geq 0, w_c[t]^T \lambda[t] = 0 \text{ (相補性拘束)} \quad (1.22)$$

ただし、ベクトルに関する不等号は、各成分について不等号が成り立つことを意味する。

この拘束条件を用いて 1.5 節で述べたように、拘束力、速度、位置の順に計算していく。この拘束条件 (式 1.21, 1.22) を 1.14 に代入することで拘束座標系での拘束力を求めることができる。

これらをまとめると次のようになり、LCP に帰着される。

$$w_c[t+1] = A\lambda[t] + b \quad (1.23)$$

$$A = J[t]M^{-1}J[t]^T h \quad (1.24)$$

$$b = J[t]\{v[t] + M^{-1}f[t]h\} \quad (1.25)$$

$$w_c[t+1] = 0 \text{ (等式拘束)} \quad (1.26)$$

$$w_c[t+1] \geq 0, \lambda[t] \geq 0, w_c[t+1]^T \lambda[t] = 0 \text{ (相補性拘束)} \quad (1.27)$$

1.5.3 ばねダンパモデルの扱い

田崎ら [15] はばねダンパモデルを LCP へ帰着させ組み込みをおこなっている．小節 1.5.2 で定義した w_c に，ばねダンパが作用する速度変数として w_s を加えて考える． w_s についても式 1.19 と同様にヤコビアンを用いて，

$$w_s[t] = J_s[t]v[t] \quad (1.28)$$

と表現できる．また，これに対応する拘束力(すなわちばね・ダンパ力)を λ_s とする．ばね・ダンパの力と速度の関係式は時刻 t におけるばねの変位 $q_s[t]$ を用いて，

$$\begin{aligned} \lambda_s[t] &= -Kq_s[t+1] - Dw_s[t+1] \\ &\approx -K(q_s[t+1] + w_s[t+1]h) - Dw_s[t+1] \end{aligned} \quad (1.29)$$

と記述することができ，これを $w_s[t+1]$ について変形すると，

$$w_s[t+1] = -(D + Kh)^{-1}\lambda_s[t] - (D + Kh)^{-1}Kq_s[t] \quad (1.30)$$

ここで改めて w, J, λ を以下のように定義しなおす．

$$w = \begin{bmatrix} w_c \\ w_s \end{bmatrix}, J = \begin{bmatrix} J_c \\ J_s \end{bmatrix}, \lambda = \begin{bmatrix} \lambda_c \\ \lambda_s \end{bmatrix} \quad (1.31)$$

と表記することで，小節 2.2.1 と同様に，

$$\begin{bmatrix} w_c[t+1] \\ w_s[t+1] \end{bmatrix} = A \begin{bmatrix} \lambda_c[t] \\ \lambda_s[t] \end{bmatrix} + b \quad (1.32)$$

$$A = J[t]M^{-1}J[t]^T h \quad (1.33)$$

$$b = J[t]\{v[t] + M^{-1}f[t]h\} \quad (1.34)$$

となる．ここで式 1.30 を式 1.32 に代入して w_s を消去し，残ったものを右辺に移項させると，

$$\begin{bmatrix} \boldsymbol{w}_c[t+1] \\ \mathbf{0} \end{bmatrix} = \tilde{\boldsymbol{A}} \begin{bmatrix} \boldsymbol{\lambda}_c[t] \\ \boldsymbol{\lambda}_s[t] \end{bmatrix} + \tilde{\boldsymbol{b}} \quad (1.35)$$

$$\tilde{\boldsymbol{A}} = \boldsymbol{J}[t]\boldsymbol{M}^{-1}\boldsymbol{J}[t]^T\boldsymbol{h} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (\boldsymbol{D} + \boldsymbol{K}h)^{-1} \end{bmatrix} \quad (1.36)$$

$$\tilde{\boldsymbol{b}} = \boldsymbol{J}[t]\boldsymbol{M}^{-1}\boldsymbol{J}[t]^T\boldsymbol{h} + \begin{bmatrix} \mathbf{0} \\ (\boldsymbol{D} + \boldsymbol{K}h)^{-1}\boldsymbol{K}q_s \end{bmatrix} \quad (1.37)$$

を得る。上式より、ばね・ダンパ拘束は、係数行列・係数ベクトル $\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{b}}$ を修正することで定義されるLCPとして表現できる。

第2章

提案

本研究ではリアルタイム物理シミュレータにおいて、回転の3自由度を持つ球関節に生物の関節のような可動域制約の組み込む。この章では解析法の物理シミュレータにおける可動域拘束の実現手法を提案する。本節ではその手順の全体像を説明する。

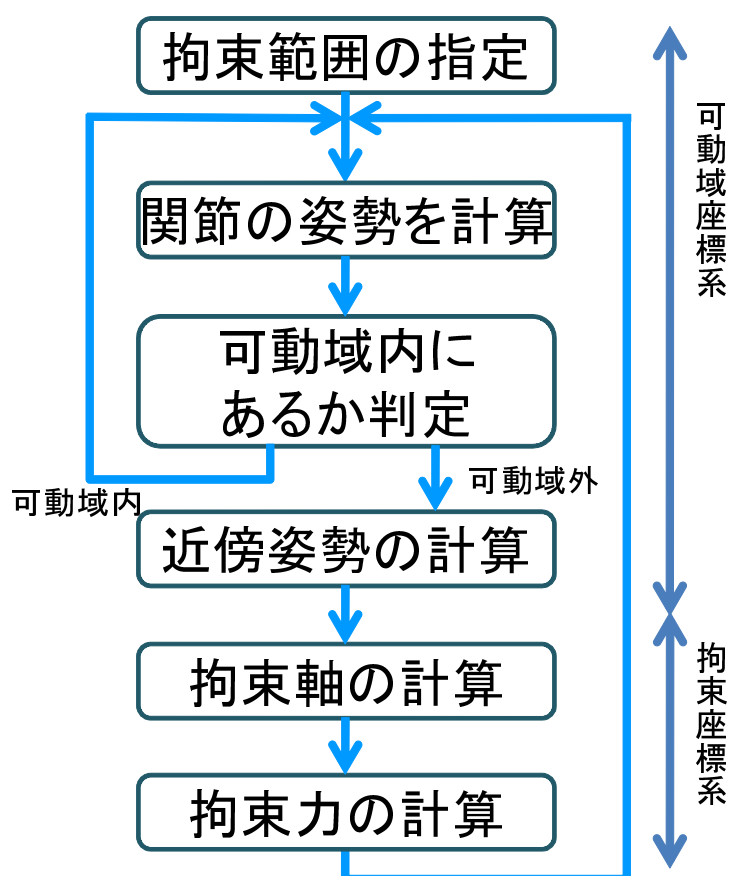


図 2.1 関節可動域拘束の手順

まず可動域を何らかの形で表現しておき、関節の現在の姿勢がこの範囲

に入っているのかどうかを判定する。入っていないときには可動域の中に戻す。本研究では関節の可動範囲を表現するために関節でつながれた2剛体の相対姿勢を表す座標系を用意し、この座標系内の自由曲面で囲まれた閉領域として可動範囲を指定する。次に現在の2剛体の姿勢に対応する可動域座標系上の点を求める。求めた点が指定した可動範囲を表す閉領域の中に入っているかどうかを確認する。可動範囲を表す閉領域の外にあった場合には閉領域内で最も現在の姿勢を表す点に近い点を探し、その点へ戻す。以上の操作で可動域内に関節の姿勢を戻すことはできるが、関節で結ばれた2剛体の相対速度が可動域の外へ出ようとする方向に働いている場合、関節にはこの速度を打ち消すような拘束力が働くと考えられる。この拘束力の方向は可動域の境界を表す自由曲面の法線方向になる。自由曲面の法線方向は曲面状の点の位置によって異なる。そこで拘束力を1変数で表すために、拘束力を新たに用意して、拘束力の計算をおこなう。新たに拘束力座標系を取ることで1.5.1節(図1.10)で述べたような相補性拘束として、この拘束を扱うことができる。これにより、可動域拘束のための拘束力の計算を他の拘束力と同様にシミュレータの拘束力計算部分でおこなうことができるようになる。図2.1は上記の計算手順をまとめたものである。以下では図2.1に沿って上から順に可動域拘束の実現手法を説明していく。

2.1 可動域の指定

本研究では球関節に可動域制約を組み込む。球関節は回転方向の3自由度を持つ。この3自由度に対して可動域の設定をするためにオイラー角の一種であるスイングツイスト座標系(図2.2)を用いる。このスイングツイスト座標系を用いるのは、関節の可動域を関節の倒れこみであるスイング方位角、スイング角と関節軸周り回転であるツイスト角に分けて考えることができ、可動域を表現した際に直感的に分かりやすいためである。また、肩関節などといった生き物の関節の可動域境界は、スイング方位角とスイング角の2自由度では相関が強く、スイング方位角とスイング角の2次元内の自由曲線の形式で詳細な指定ができることが望ましい。スイングツイスト座

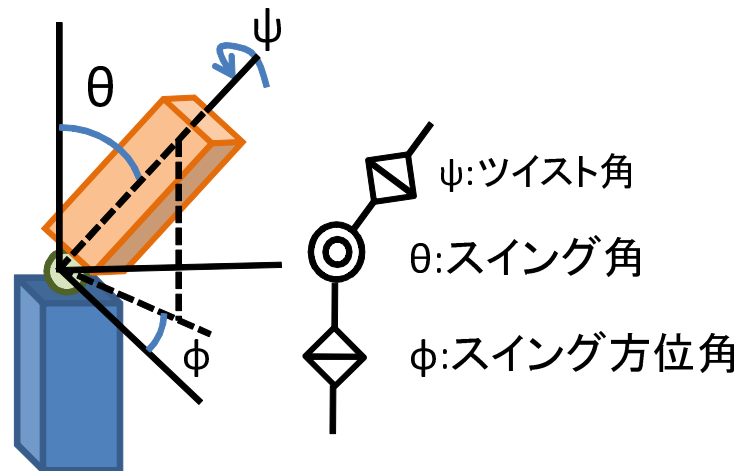


図2.2 スイングツイスト角座標系

標系での各成分の角度は，図2.2のように倒れる方向(スイング方位角) ϕ ，倒れこみ角(スイング角) θ ，2剛体間の姿勢を表現するひねり角(ツイスト角) ψ となる．このとき剛体の位置の動きはスイング方位角，スイング角によって定まり，その位置での姿勢がツイスト角で表される．

可動域を指定する際にはまず関節の倒れこみの限度を指定するためにスイング方位角，スイング角について可動域の指定をおこなう．このときスイング方位角，スイング角を極座標表示することで直感的に可動域を指定できるようにした(図2.3)．

図2.3はスイング方位角・スイング角の極座標表示で示す．座標平面上の1点がスイング方位角・スイング角の組(ϕ, θ)に対応する．ここでスイング角については，原点が $\theta = 0^\circ$ ，最も外側の円が $\theta = 180^\circ$ に対応する．また，スイング方位角については，原点から右向き半直線上の点が $\phi = 0^\circ$ に対応するものとし， $0^\circ \leq \phi < 360^\circ$ の範囲で反時計回りに偏角をとるものとする．関節可動域は，極座標表示されたスイング方位角・スイング角平面上のエルミートスプライン曲線によって指定する．具体的には，曲線の通過点(図2.3紫の点)を複数定義し，通過時の曲線の傾きを指定することによって可動域を定義する．

ツイスト角 ψ の可動域指定は，スイング方位角・スイング角の可動域の定

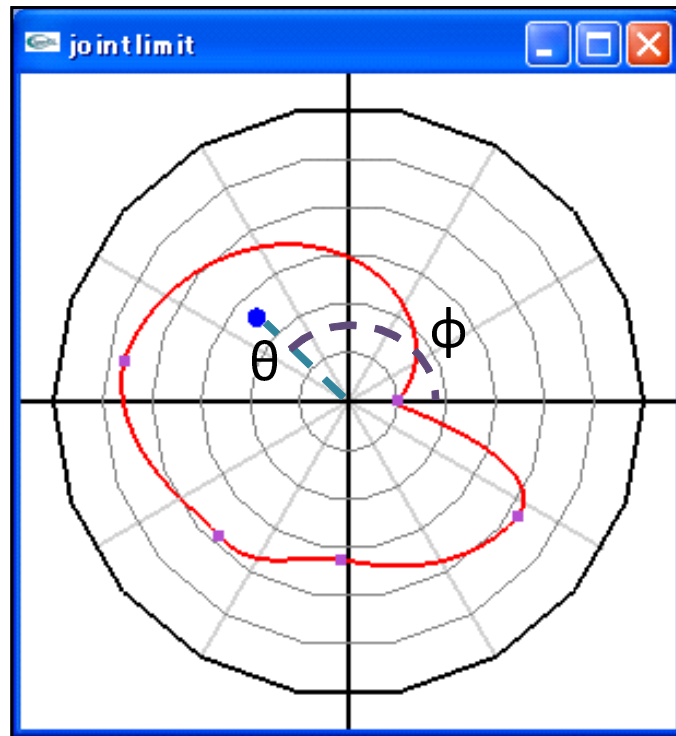


図2.3 関節可動域

義において指定したスプライン曲線の各通過点に対し，関節が対応するスイング角・スイング方位角をとった場合のツイスト角の最大値・最小値を指定するという形でおこなう．スイング角が $\theta = 0^\circ, 180^\circ$ となる場合，関節はスイング方位角によらず同じ状態をとるため，ツイスト角の可動域もスイング方位角によらず一定の値をとる必要がある．一方前述のように可動域の境界上の通過点についてツイスト角可動域を指定した場合，一般にはスイング方位角に依存して異なるツイスト角可動域を指定することができる．このことから，スイング角が $\theta = 0^\circ, 180^\circ$ の場合に対応するツイスト角の可動域は別に指定するものとした．以上のように指定された値を用い，関節が任意のスイング方位角・スイング角をとった際のツイスト角の可動域は，それぞれ指定された値から線形補間することによって決定するものとした．このようなツイスト角の可動域は，スイング角・スイング方位角を極座標表示した平面に直交する座標軸を加え，図2.4のような円筒座標とすることで表現することができる．

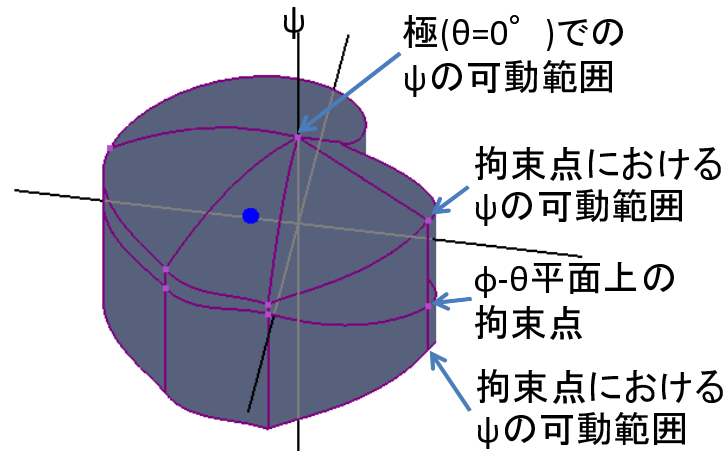


図 2.4 関節可動域の例

2.2 可動域の内外判定

拘束力の有無を決定するためには、現在の関節の姿勢が可動域の中にあるか外にあるかを調べる必要がある。スイングツイスト座標系で指定された可動域の内外にあるかを判定するため、まず現在の関節の姿勢をスイングツイスト座標系に変換する。計算した姿勢に対してまずスイング方位角、スイング角が可動域に入っているかを確認する。次にツイスト角が可動域内にあるかを確認する。

スイング方位角、スイング角については、直前のステップにおける関節の姿勢を表す点から現在のステップにおける姿勢を表す点を結ぶ線分と、可動域を表す曲線の間には交点があるかを調べた。交点がない場合には可動域内を移動しているため拘束力の発生はない。交点がある場合には現在の姿勢が可動域の外に移動するため、関節姿勢を可動域内に保つように拘束力を発生する必要がある。

関節の姿勢が可動域外に出ていた場合は関節可動域内の姿勢で、現在の姿勢から最も近い姿勢（近傍姿勢）を求め、関節の姿勢を可動域内に引き戻す。このとき最近傍の判断はスイングツイスト座標系でのユークリッド距離に基づいておこなうものとした。この最近傍姿勢は、内外判定のときに

算出された交点の周辺となる．このことから，算出された交点を微小区間に区切って繰り返し探索することで可動域内の最近傍姿勢を求めた．このようにして求めた可動域内の再近傍姿勢へ関節の姿勢を修正することで関節姿勢を可動域内に保つ．

ツイスト角可動域の内外判定では，スイング方位角，スイング角について関節姿勢が可動域内にあった場合には，現在のスイング方位角，スイング角におけるツイスト角の可動域を計算し現在のツイスト角がその範囲にあるかを確認した．また，スイング方位角，スイング角について関節姿勢が可動域外であった場合には，スイング方位角・スイング角について可動域内の最近傍姿勢を算出したのち，最近傍姿勢に対応するツイスト角の可動域について内外判定をおこなった．

2.3 速度の拘束

関節姿勢が可動域外にある場合には次のステップで可動域の外側に向かう速度が働かないように拘束力を与える必要がある．解析法の物理シミュレータにおいて拘束力はLCPで定式化された拘束条件を解くことにより計算される(1.5節)．例えば1軸の関節であれば可動軸に対する拘束条件は容易に定式化され，拘束力の計算をおこなうことができる．しかし球関節の可動域制限など複数の可動軸に対する可動域拘束では，拘束時の関節姿勢によって拘束力の働く方向が異なるため，拘束条件が複数の座標変数に依存するものとなってしまう，1変数に関するLCPとして定式化することができない．このことから，拘束条件を1軸に対するLCPとして定式化できるような座標軸をその都度算出して用いることを考える．

例えば図2.5のような2軸の平面に三角形の可動域がある状況を考える．関節姿勢を e_1 の軸と e_2 の軸で表現した場合，可動域拘束は，この座標系では $Ae_1 + Be_2 < C$ (A, B, C : 定数) のように複数の軸に対する拘束条件となってしまう，LCPによる定式化となっていない．そこで，拘束条件をLCPで定式化できるようにするため座標変換をおこなう．図2.5の e_3 軸のように可動域と垂直になる新たな座標軸を取り，これを拘束軸と呼ぶ．この拘束軸に関して

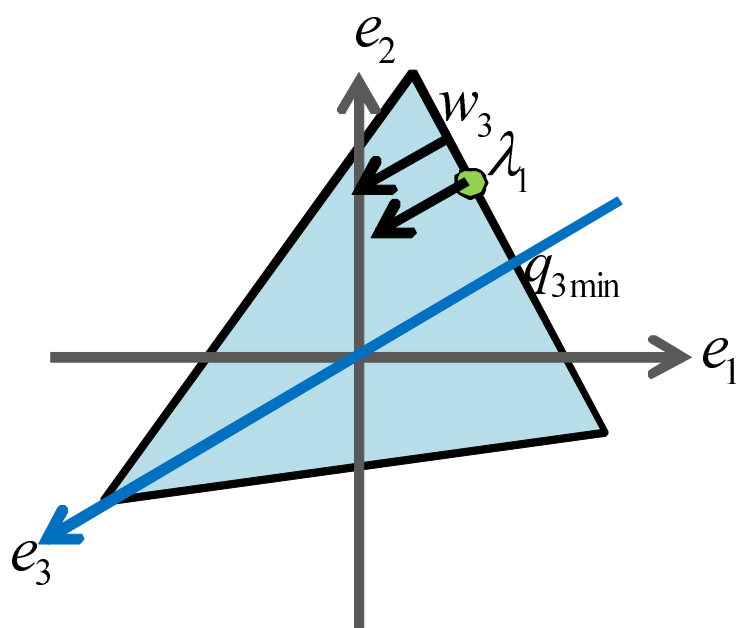


図 2.5 拘束軸の取り方

拘束条件を記述すると $e_3 < C$ という単純な式となり、拘束力と速度の関係を線形相補性問題に定式化することができる。このように関節姿勢を拘束軸を既定の一つとする座標系へ座標変換することで、関節可動域制限を剛体の接触と同様に線形相補性問題として扱うことができる。

このような座標系の変換はヤコビアンを書き換えることでおこなうことができる。拘束力の計算は拘束の座標系上でおこなわれる。このような計算は剛体の座標系における速度を拘束の座標軸における速度に変換するヤコビアンを用いることでおこなわれている。

$$M\dot{u} = f_c + f_e \quad (2.1)$$

$$M\dot{u} = f_c + J^t\lambda \quad (2.2)$$

$$J\dot{u} = M^{-1}Jf_c + M^{-1}\lambda \quad (2.3)$$

$$\dot{w} = M^{-1}Jf_c + M^{-1}\lambda \quad (2.4)$$

式 2.1 は剛体の座標系 (図 2.6 黒) での運動方程式であり、式 2.2 の λ は拘束の座標系 (図 2.6 赤) 上で見た拘束力である。このとき、ヤコビアン J は剛体の座標

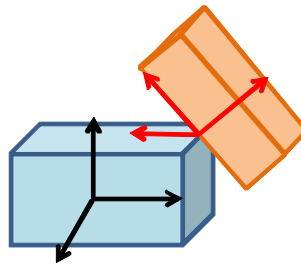


図2.6 剛体の接触

系から拘束の座標系への変換ヤコビアンであり，変換ヤコビアンを書き換えることで可動域拘束を記述する座標軸を変化させ，任意の座標軸を用いるようにすることができる．

剛体同士の運動を拘束するために必要な座標軸は本来並進運動と回転運動の6軸であるが，球関節では並進運動はもともと拘束されているため，球関節の可動域制限を計算する場合には回転3自由度のみを考えれば十分である．そのため，ヤコビアン J は最大でも 3×3 行列となる．実際には，ヤコビアンによる座標変換がおこなわれた後には，拘束力を作用させるべき拘束軸のみが残れば十分である．本手法ではスイング方位角・スイング角による拘束と，ツイスト角による拘束の2つに分けて考えるため，このヤコビアンはスイング方位角・スイングによる拘束，ツイスト角による拘束のどちらかに拘束力が発生する場合には1行，両方ともに拘束力が発生する場合には2行となる．

第3章

実装

本研究では解析法の物理シミュレータ (Springhead2) へ提案手法による可動域拘束の組み込みをおこなった。本章では実装した球関節の可動域拘束の詳細について以下の順に説明する。

まず、可動域の境界を設定するための方法として、指定された可動域境界の通過点の補間方法を述べる。次に関節の現在の姿勢の計算および、その姿勢が指定した可動域内にあるかの判定法について述べる。さらに、可動域の外にある場合に可動域内の最近傍姿勢を求める方法について述べる。そののち、拘束力の座標変換をおこなうヤコビアン の算出法、およびヤコビアンを使った拘束力の計算法について述べる。最後に解析法のシミュレータ本来のヤコビアンについて説明し関節可動域拘束の計算をするためのヤコビアンと他の拘束ヤコビアン の扱いについて述べる。

3.1 可動域の設定

可動域の境界となる曲線の指定は、曲線の通過点を指定することによっておこなうものとした。ここでは、各通過点から曲線を求める補間に利用した関数や式について説明する。

本研究で実装した方法とは別に検討した補間方法について先に述べる。本手法において関節可動域は図 2.3 に示すスイング方位角・スイング角を極座標で表示した平面内の閉曲線によって指定する。このとき閉曲線を定義するために指定する曲線の通過点をここでは拘束点と呼ぶ。拘束点の個数は任意であるが、スイング方位角が 0° になる位置に 1 点指定し、拘束点の数を n 点とする。指定した拘束点の間を補間して可動域とする。補間する方法としてまず、スイング角をスイング方位角の関数として定義するエルミー

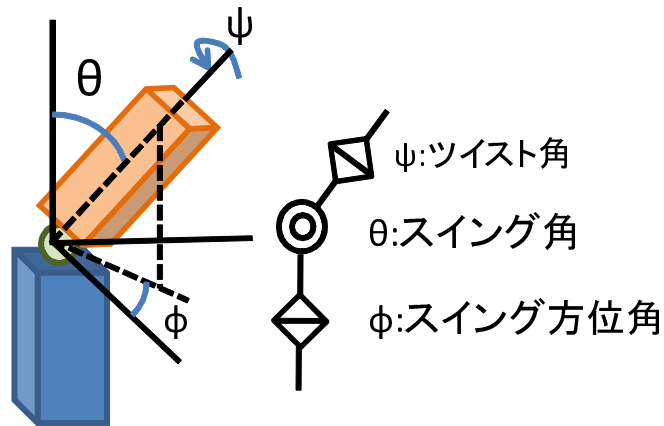


図 3.1 スイングツイスト角座標系

トスプラインによる可動域の指定を考えた．このとき補間によって得られる曲線は次の式で表すことができる．

$$\theta_{lim} = A\phi^3 + B\phi^2 + C\phi + D \quad (3.1)$$

$$A = 2\phi_{i+1} - 2\phi_1 + \dot{\phi}_{i+1} + \dot{\phi}_1 \quad (3.2)$$

$$B = 3\phi_1 - 3\phi_{i+1} - 2\dot{\phi}_1 - \dot{\phi}_{i+1} \quad (3.3)$$

$$C = \dot{\theta}_i \quad (3.4)$$

$$D = \theta_i \quad (3.5)$$

このとき i は拘束点の番号 ($1 \leq i \leq n-1$) である．このように定義された可動域境界では，一つのスイング方位角に対して必ず一つのスイング角が対応する．そのため，図 3.2 のような可動域を指定することができない．

次に拘束点の間を線形に補間する手法を考えた．この手法では，拘束点で可動域の境界線が微分不能になり安定にシミュレーションできない．そのため拘束点付近で可動域の境界を表す直線同士を微小の円で接続することで滑らかな可動域境界を定義するものとした．このような指定方法では図 3.2 のような関節可動域を指定することができるが，生物の関節は曲線の可動域を持っているため直線の組み合わせによる可動域指定では多数の拘束点を必要としてしまう．

以上の検討から，本研究では拘束点の間を媒介変数を用いたエルミート

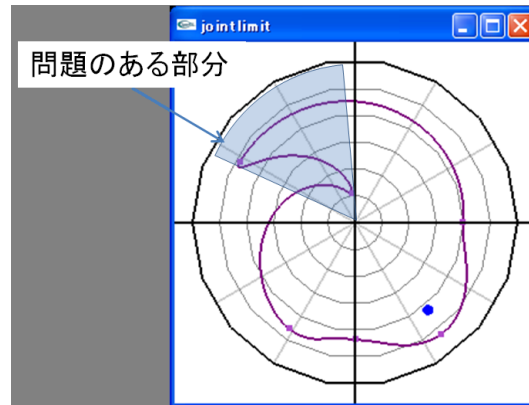


図3.2 指定できない可動域

スプラインによって補間したものを可動域拘束の境界として用いることとした。指定した拘束点数を n 点として、各拘束点の番号を i とする。ある拘束点とそこから補間によりつながる拘束点のスイング方位角、スイング角、ツイスト角はそれぞれ $(\phi_i, \theta_i, \psi_i)$ 、 $(\phi_{i+1}, \theta_{i+1}, \psi_{i+1})$ とおける。このときそれぞれの拘束点の間は次の式で補間する。

$$\phi_{lim} = A_\phi t^3 + B_\phi t^2 + C_\phi t + D_\phi \quad (3.6)$$

$$A_\phi = 2\phi_i - 2\phi_{i+1} + \dot{\phi}_i + \dot{\phi}_{i+1} \quad (3.7)$$

$$B_\phi = 3\phi_{i+1} - 3\phi_i - 2\dot{\phi}_{i+1} - \dot{\phi}_i \quad (3.8)$$

$$C_\phi = \dot{\phi}_i \quad (3.9)$$

$$D_\phi = \phi_i \quad (3.10)$$

$$\theta_{lim} = A_\theta t^3 + B_\theta t^2 + C_\theta t + D_\theta \quad (3.11)$$

$$A_\theta = 2\theta_i - 2\theta_{i+1} + \dot{\theta}_i + \dot{\theta}_{i+1} \quad (3.12)$$

$$B_\theta = 3\theta_{i+1} - 3\theta_i - 2\dot{\theta}_{i+1} - \dot{\theta}_i \quad (3.13)$$

$$C_\theta = \dot{\theta}_i \quad (3.14)$$

$$D_\theta = \theta_i \quad (3.15)$$

このとき t は媒介変数であり、 $(0 \leq t < 1)$ を満たす。 $\dot{\phi}_i, \dot{\theta}_i$ は $t=0$ のときの微分係数、 $\dot{\phi}_{i+1}, \dot{\theta}_{i+1}$ は $t=1$ のときの微分係数である。拘束点の区間ごとにこの式

を用いて補間をおこない，得られる曲線をスイング方位角，スイング角平面の可動域の境界とした．この補間によって得られる曲線を可動域曲線と呼ぶ．

ツイスト角の可動域についてはスイング方位角，スイング角平面上の各拘束点に対して可動域の最大値と最小値を与えるものとした．ここから，可動域内の各点について，スイング方位角・スイング角に対応するツイスト角可動域の値を補間によって求める必要がある．可動域曲線上の点については，拘束点で挟まれた各区間について，両端の拘束点でのツイスト角可動域から線形補間することによって決定するものとした．このときの補間の式は次のようになる．

$$\psi_{lim} = (1-t)\psi_i + t\psi_{i+1} \quad (3.16)$$

次に可動域曲線で囲まれた可動域内部の各点について対応するツイスト角可動域を求める補間について考える．スイング角 $\theta = 0^\circ, 180^\circ$ の場合に対応して，スイング方位角によらないツイスト角可動域を指定しておくことが必要である．これを用い，可動域内の任意のスイング方位角・スイング角におけるツイスト角の可動域制限の値 ψ_{lim} は， $\theta = 0^\circ$ に対応するツイスト角の可動域 ψ_c と式3.16によって求めた可動域曲線上のツイスト角の可動域制限との間を2次式で補間することで計算するものとした．

$$\psi_{lim} = \left(1 - \left(\frac{\theta_{now}}{\theta_{lim}}\right)^2\right)\{(1-t)\psi_i + t\psi_{i+1}\} + \left(\frac{\theta_{now}}{\theta_{lim}}\right)^2 \psi_c \quad (3.17)$$

ただし， θ_{now} は現在のスイング角の角度である．

3.2 現在の関節姿勢の計算

指定した可動域に対して，関節の現在の姿勢が可動域の内側に入っているかを判定するために，関節の姿勢をスイングツイスト座標系で求める．今回実装の対象とした物理シミュレータでは関節の姿勢をクォータニオンで表しているため，クォータニオンからスイングツイスト座標系での角度のを

計算する．ここで求めたスイングツイスト座標系での関節角度を用いて関節が可動域内にあるかの判定をおこなう．求める関節角度はスイング方位角，スイング角，ツイスト角の3つでありそれぞれの計算方法は次のようになる．

まずスイング角を計算する．スイング方位角，スイング角，ツイスト角がすべて0°になる姿勢を関節の基準姿勢とし，基準姿勢における関節子剛体座標系 z 軸の向きの単位ベクトルを v_l とする．基準姿勢から現在の姿勢へ変換する行列 J_c は次の通りである．

現在の姿勢への変換ヤコビアン J_c は次のように計算できる．

$$J_{cEz} = Xjrel \begin{bmatrix} 0; & 0 & 0 & 1 \end{bmatrix} Xjrel^* \quad (3.18)$$

$$J_{cEx} = Xjrel \begin{bmatrix} 0; & 1 & 0 & 0 \end{bmatrix} Xjrel^* \quad (3.19)$$

$$J_{cEy} = J_{cEz} \times J_{cEy} Jc = \begin{bmatrix} J_{cEx} & J_{cEy} & J_{cEz} \end{bmatrix}^T \quad (3.20)$$

ただし $Xjrel$ は関節を作る2剛体の相対姿勢を表すクォータニオンであり， $Xjrel^*$ は $Xjrel$ の共役クォータニオンである．この行列 J_c と v_l より関節のスイング角は次式で求められる．

$$\theta = \arccos(v_l \cdot J_{cEz}) \quad (3.21)$$

スイング方位角 ϕ は現在の関節座標系の z 軸と，ワールド座標系を関節の基準姿勢における子剛体の座標系へ移す行列 L_{center} (3×3 行列)から次のように計算できる．

$$P = J_{cEz} L_{center} \quad (3.22)$$

$$\phi = \text{atan}\left(\frac{P_y}{P_x}\right) \quad (3.23)$$

ただし ϕ は $0 \leq \phi < 360^\circ$ とする．

次にツイスト角を計算する．剛体座標系における z 軸基底ベクトル ${}^t \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$ を e_z とおく．関節を構成する2つの剛体間の相対的な回転を表すクォータニオンを $Q = {}^t \begin{bmatrix} w_q & v_q \end{bmatrix}$ とすると，ツイスト角は次のように計算される．

$$\mathbf{v}_p = \frac{0.5 \cdot \mathbf{e}_z + \mathbf{J}_{cEz}}{|0.5 \cdot \mathbf{e}_z + \mathbf{J}_{cEz}|} \times \mathbf{e}_z \quad (3.24)$$

$$\mathbf{w}_p = \sqrt{1 - |\mathbf{v}_p|^2} \quad (3.25)$$

$$\begin{bmatrix} w_\psi \\ \mathbf{v}_\psi \end{bmatrix} = \mathbf{PQ} \quad (3.26)$$

$$\psi = \begin{cases} \arccos(w_\psi) & (\mathbf{v}_\phi \text{の } z \text{ 成分} > 0) \\ -\arccos(w_\psi) & (\mathbf{v}_\phi \text{の } z \text{ 成分} < 0) \end{cases} \quad (3.27)$$

ただし, $\mathbf{P} = {}^t [\mathbf{w}_p \ \mathbf{v}_p]$ である.

3.3 可動域内にあるかの判定

前節の手法によって求めた関節のスイング方位角, スイング角, ツイスト角を用いて関節の現在の姿勢が指定した可動域の中にあるかを調べる. 本実装では, 1ステップ前に可動域内だった場合と, 1ステップ前に可動域を違反していた場合とで場合分けして可動域の違反の有無を調べる手法とした. これは可動域を違反した場合に可動域を戻す位置を計算するための目安となる姿勢を求めるためである. この姿勢を求める方法については次節で説明する. 本節では可動域内にあるかどうかの判定方法について述べ, 次節で関節姿勢の引き戻し先となる可動域内の位置の計算について述べる.

1ステップ前で可動域を違反していなかった場合, スイング方位角, スイング角を極座標とした平面において, 1ステップ前の関節の姿勢から現在の関節の姿勢へ向かうベクトルを計算する. このベクトルと可動域曲線との間に交点があるかを計算する. 交点がない場合には, 可動域の中で移動しているため, 関節姿勢は可動域を違反していない. 交点を持つ場合には可動域の中から外へ姿勢が変移しているため可動域を違反している. (図3.3のベクトルA)

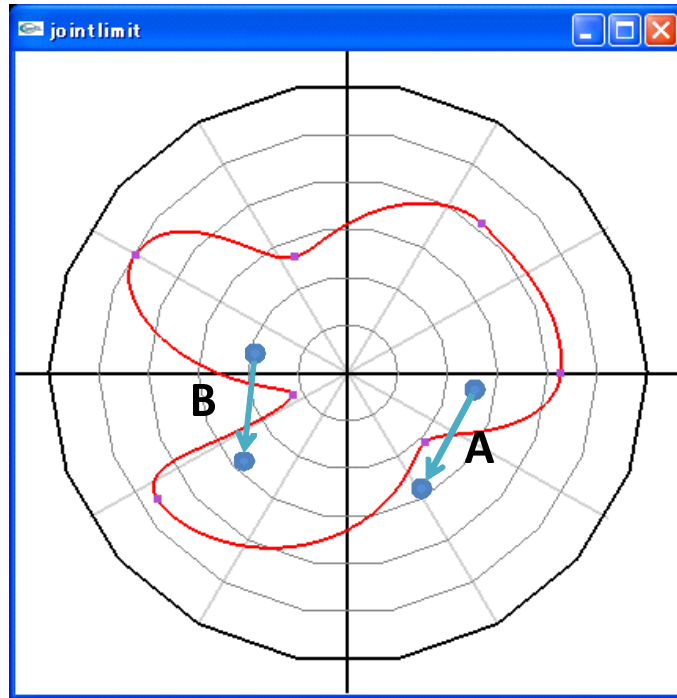


図3.3 拘束の判定

1ステップ前のスイング方位角，スイング角を ϕ_{bef}, θ_{bef} ，現在のスイング方位角，スイング角を ϕ_{now}, θ_{now} ，前のステップのスイング方位角，スイング角を $\phi_{last}, \theta_{last}$ とすると，二つを結ぶベクトルは次のように表せる．

$$\theta = \alpha\phi + \beta \tag{3.28}$$

$$\text{ただし } \alpha = \frac{\theta_{now} - \theta_{last}}{\phi_{now} - \phi_{last}} \tag{3.29}$$

$$\beta = \frac{\phi_{now}\theta_{last} - \theta_{now}\phi_{last}}{\phi_{now} - \phi_{last}} \tag{3.30}$$

スイング方位角・スイング角は式3.6,3.11によって計算される．式3.28にスイング方位角，スイング角を代入すると，関節姿勢の変化を表すベクトルと可動域曲線の交点を求める方程式は， t の3次式となり，式3.31，3.32のように表される．

$$A_{\theta_i}t^3 + B_{\theta_i}t^2 + C_{\theta_i}t + D_{\theta_i} = \alpha(A_{\phi_i}t^3 + B_{\phi_i}t^2 + C_{\phi_i}t + D_{\phi_i}) + \beta \quad (3.31)$$

$$(A_{\theta_i} - \alpha A_{\phi_i})t^3 + (B_{\theta_i} - \alpha B_{\phi_i})t^2 + (C_{\theta_i} - \alpha C_{\phi_i})t + (D_{\theta_i} - \alpha D_{\phi_i}) = 0 \quad (3.32)$$

ただし i は拘束点間の区間の番号である。また、可動域曲線を表す媒介変数の定義から $0 < t < 1$ とする。すべての拘束点間の区間において3次方程式を計算し解を求める。すべての拘束区間においてこの方程式を解く。解がない場合には可動域の内部での移動であり、可動域を違反していないと判定する。解がある場合にはベクトルと可動域曲線との間に交点が存在する。ここで1ステップ前には可動域内であったので、関節の姿勢が可動域の内側から外側へ移動してたことを意味しており、可動域を違反していると判定される。また、交点が複数見つかる場合には図 3.3B のように拘束曲線を突き抜けて可動域へ戻っていることが考えられるため、同様に可動域拘束を違反していると判定する。この場合、1ステップ前の状態に最も近い交点に接したときに拘束力を加えこの交点付近に戻るようにする必要がある。

前のステップで拘束していた場合には現在のスイング方位角と拘束曲線のスイング方位角の一致する点を求めその点のスイング角と現在の姿勢のスイング角の大小を比べて可動域を違反しているかを判定した。現在のスイング方位角を ϕ_{now} とすると次の3次方程式を解くことで媒介変数が求められる。

$$A_{\phi_i}t^3 + B_{\phi_i}t^2 + C_{\phi_i}t + D_{\phi_i} = \phi_{now} \quad (3.33)$$

これにより求められた媒介変数を式 3.11 に代入することで可動域曲線上のスイング角を求め、求めたスイング角と現在のスイング角の値を比べ拘束の有無の判定をした。

ツイスト角の可動域違反判定は、スイング方位角・スイング角の可動域違反がない場合には式 3.17 でツイスト角可動域を求め、関節の現在のツイスト角の値と比べて可動域を違反していないか調べる。スイング方位角、スイング角で可動域違反がある場合には、式 3.16 を用いて、関節姿勢が可動域内へ引きもどされた位置のスイング方位角・スイング角に対応するツイス

ト角可動域を計算する．計算された可動域を用いて，現在の姿勢が可動域に入っているのかを確認した．

3.4 近傍姿勢の計算

関節の現在の姿勢が可動域拘束を違反している場合，可動域内となるよう関節姿勢を戻す．本節ではその計算法を述べる．関節姿勢の引き戻しの目標姿勢は，可動域を違反した現在の姿勢から最も近い可動域内の姿勢であり，可動域曲線上の姿勢となる．この姿勢を近傍姿勢と呼ぶ．

近傍姿勢は1ステップ前の関節の姿勢から現在のステップの関節の姿勢の間にあり，式3.32で求めた交点の近傍に存在する．そこで，可動域の違反が新たに発生したステップにおいては，可動域違反判定の過程で求められる交点に対応する可動域曲線の媒介変数 t を求め，これを基準として t を微小変化させた点を新たに求め，スイング方位角・スイング角平面で現在の関節姿勢に対応する点 t のユークリッド距離が極小値となる姿勢を求め近傍姿勢とした．また，1ステップ前ですでに可動域を違反していた場合には，1ステップ前における近傍姿勢から媒介変数 t を微小変化させユークリッド距離が極小となる姿勢を求めた．

ツイスト角の可動域については，1自由度の可動域であるため，可動域を違反している場合には，可動域範囲の境界を近傍姿勢とした．

3.5 拘束軸の計算

ヒンジジョイントのような1軸の関節に対する可動域制限では，拘束条件が線形相補性問題(LCP)として定式化されるため解析法のシミュレータにそのまま組み込むことができる．

しかし球関節の可動域拘束では，座標系を変換しない場合可動域の拘束条件が2変数に渡ったものになるためLCPとして定式化できず，そのまま解析法のシミュレータに組み込むことはできない．そこで拘束条件を1変数についてのLCPの拘束条件を用いて定式化できるような拘束軸を計算し，こ

の軸について拘束条件を立てシミュレータへの組み込みをおこなう。ここではこの拘束軸の計算について述べる。

現在の姿勢のスイング角・スイング方位角を θ_n, ψ_n とし、可動域内の近傍姿勢のスイング角・スイング方位角を θ_l, ψ_l とすると、拘束軸 \tanLine は次のように計算できる。

$$\tanLine = \frac{(\theta_l - \theta_n, \phi_l - \phi_n, 0)}{|(\theta_l - \theta_n, \phi_l - \phi_n, 0)|} J_c \quad (3.34)$$

ここで J_c は関節を基準姿勢から現在の姿勢へ変換するヤコビアンである。拘束法の計算を、このように求めた拘束軸を基底の一つとする座標系上でおこなうことによって、可動域拘束をLCPで定式化することができる。

実際には、拘束軸を計算して座標変換を行うのではなく、解析法のシミュレーションにおいて剛体の座標系から関節の座標系に変換するためのヤコビアン J を書き換えることで、関節の座標系が拘束軸を含む座標系となるように実装をおこなった。

次にヤコビアン J の書き換えについて詳述する。拘束法のシミュレーションでは、剛体の座標系をヤコビアン J で変換して関節の座標系を求め、関節の座標系上で拘束条件を記述することで計算をおこなっている。また、関節の現在の姿勢は、標準姿勢における関節姿勢を現在の関節姿勢に変換するヤコビアン J_c によって表現される。

剛体の座標系から関節の座標系へ変換するヤコビアン J は 6×6 行列だが、左上の 3×3 (並進成分) と右下の 3×3 (回転成分) 以外には値は入っていない。また並進方向の運動を表す部分は球関節では固定されているため使用されない。そのためヤコビアン J の左下の 3×3 の成分 (以下 J_w とする) のみを書きかえればよい。また、関節の座標系を変換するヤコビアン J_c は J_w に対応する回転成分を表す 3×3 行列となる。

J_c の1行目の成分が関節のスイング角、2行目の成分がスイング方位角、3行目の成分がツイスト角の拘束軸をつくる座標系となるよう J_c を構成する必要がある。関節の座標系をこの3種類の拘束軸すべてを基底とする座標系とすることで、同時に拘束力の計算をおこなうことができる。以上より可動域拘束のための書き換え後のヤコビアン J_{cn} を式3.35のように計算する。

$$J = \begin{pmatrix} \boxed{\text{並進}} & 0 \\ 0 & \boxed{\text{回転}} \end{pmatrix} \rightarrow J_w$$

図3.4 ヤコビアンの中身

$$J_{cn} = \frac{(\theta_{now} - \theta_{lim}, \psi_{now} - \psi_{lim})}{|(\theta_{now} - \theta_{lim}, \psi_{now} - \psi_{lim})|} J_c \quad (3.35)$$

ヤコビアン J_{cn} を用いて，剛体の座標系から，各拘束軸を基底とする関節座標系への変換をおこなうことができる．

3.6 関節可動域の拘束力の計算

可動域を違反している場合には近傍姿勢へ戻し，可動域を違反する方向に速度が生じないような拘束力を与える．この拘束力は前節で求めた拘束軸を用いることでLCPの拘束条件に基づいて計算することができる．ここではこの拘束軸上で与えた拘束力について述べる．

関節姿勢の可動域外への侵入量に比例した速度によって関節を可動域内へ引き戻す．比例係数 c は適切に定める必要がある．このような速度が実現するよう拘束力の計算をおこなう．このとき拘束力と速度の関係式は次のようになる．

$$w = A\lambda + b \quad (3.36)$$

$$A = 0 \quad (3.37)$$

$$b = c * diff \quad (3.38)$$

ここで $diff$ は侵入量である．

式3.36は，侵入量 $diff$ を0とするよう拘束力をPD制御していると考えることができる．このとき係数 A, b は $diff$ に対するバネ・ダンパ係数である． A, b に適切なバネダンパ係数を設定することにより，可動域境界の壁を軟ら

かいバネダンパで構成することも可能である。これにより、関節姿勢を常に可動域内にとどめようとする「硬い可動域」だけでなく、強い力を加えると比例して可動域外に出ることができる「軟らかい可動域」を実現することもできる。

3.7 拘束力の計算

関節座標系を拘束軸ベースのものに変換するヤコビアンと、拘束法のシミュレータで一般に用いられる拘束力を計算するためのヤコビアンとの関係と取り扱いについて述べる。

一般に剛体同士の相対運動は並進と回転の計6自由度を持つ。接触や関節など剛体同士の拘束を計算する場合、これら6自由度のうち拘束すべき自由度を選んで拘束力を作用させる必要がある。例えば球関節を実現するための拘束条件は並進方向の相対運動を0とするものであり、3自由度を拘束する条件となる。さらに球関節の姿勢をPD制御する場合は、回転方向の3自由度に対する拘束条件として書かれる。

本手法の実装対象としたシミュレータでは、従来、6変数のそれぞれについて拘束条件の対象となる自由度であるかをチェックし、拘束条件の対象となっている自由度について拘束力を計算していた。このとき、関節を実現する拘束条件と関節姿勢のPD制御を実現する拘束条件は別個に取り扱い、それぞれについて拘束すべき自由度のチェックと拘束力の計算を行っていた。

しかし、ヤコビアンによる関節座標系への変換後、拘束力を計算するためには、変換後の座標空間には拘束にかかわる自由度のみが残れば十分である。そのため、剛体の相対運動の6自由度すべてをヤコビアンによって変換し、変換後の6自由度の関節座標系について各拘束条件ごとに拘束自由度のチェックと拘束力の計算を行う方法は無駄が多い。

このことから、拘束すべき自由度を並べた空間への変換を行うようヤコビアンを構成することで、ヤコビアンの計算を必要最低限の自由度についてのみ行うようにするとともに、関節を実現する拘束条件と関節姿勢のPD制御を実現する拘束条件を同時に一つのヤコビアンで取り扱えるようにし

た．すなわち，関節の実現と関節姿勢のPD制御はそれぞれ並進と回転の各3自由度ずつ，計6自由度についての拘束となる．このことから，両方の拘束に必要となる関節座標系は6自由度の空間であり，同じく6自由度を持つ剛体座標系からの変換は 6×6 のヤコビアンによってなされることとなる．

一方，関節の可動域制限を実現する拘束条件のためには，姿勢のPD制御とは別個に，拘束軸に基づく1ないし2自由度の変数が必要となる．関節可動域に関する拘束は関節姿勢が可動域を違反した場合にのみ必要となるため，可動域拘束に関する変換を関節座標系へ変換するヤコビアンに組み込んだ場合，可動域を違反した場合のみヤコビアンの次元を 7×6 ないし 8×6 とする必要が生じる．このように次元が可変の行列を扱うことは計算量の増加を招くため，本実装では可動域拘束のための拘束軸に基づく座標系への変換は別個のヤコビアンを用意しておこなうものとした．

第4章

評価

第2章で提案し，第3章で実現方法を示した，複雑な関節可動域拘束の物理シミュレータへの組み込み手法の有効性とキャラクタ動作への効果を確認するために評価をおこなった．有効性の確認では，人の肩関節の可動域を物理シミュレーション内のバーチャルヒューマンの型関節に適用し，可動きが安定に再現できるか確かめた．また，キャラクタ動作への効果の確認では，可動域を組み込むことで，不自然な姿勢をとることができるかどうかと，自然な動作を生成できるかどうかを確かめた．

4.1 提案手法の有効性の確認

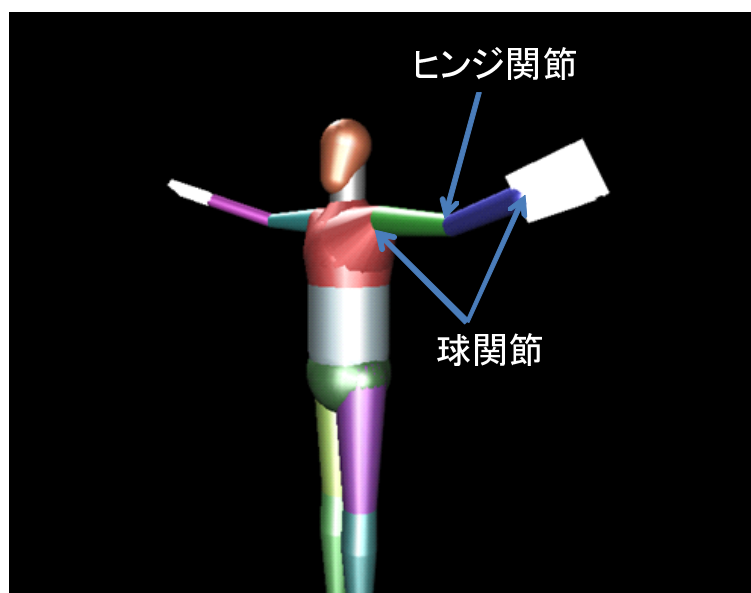


図4.1 可動域を入れた人型モデル

提案手法が期待通り可動域拘束を実現できるかどうか確認するため，図4.1に示すバーチャルヒューマンの肩関節に対して可動域制約の組み込み動作確認をおこなう実験をおこなった．このバーチャルヒューマンは17個の剛体でできており，可動域拘束の対象となる腕の部分では，肩と手首に球関節を用い，肘にはヒンジ関節を用いている．実験では，まず人の肩関節の可動域を計測し，これに近い可動域をバーチャルヒューマンの型関節に組み込むことで，人の肩関節のような複雑な可動域拘束を設定できることを確認する．次に拘束の有無によるバーチャルヒューマンの腕の動きの違いを確かめ，可動域を入れることで，人に近い動きを生成できるようになるかどうか確かめた．

4.1.1 人の関節可動域の計測

人の肩関節の可動域を光学式3次元位置計測装置OPTOTRAKを用いて計測した．図4.2のように肩に3点(裏に1点)と肘に4点(裏に1点)のマーカを配置した．このとき肩のマーカは体の正面と上，後に配置し，肘のマーカは肘を1周するように90度刻みで配置してデータの取得をおこなった．OPTOTRAKでは，マーカが体の影に隠れたり，マーカの向きがOPTOTRAKと逆向きになる場合などには，そのマーカの位置を計測することができなくなる．そこで，7点のマーカのうちで取得できたマーカの位置に応じて関節の向きを推定した．推定は次のような手順でおこなった．まず，肘のマーカから肘の中心位置の推定をおこなう．このとき肘のマーカは3点必要となる．次に肩のマーカの位置と，肘の中心位置から肩の中心位置の推定をした．肘の中心位置と肩の中心位置を結ぶベクトルが関節の向きとなる．また，データ取得をする際に，肩を真横に伸ばし，肘が手前に曲がる姿勢を初期姿勢とした．取得して求めたベクトルの向きのを初期姿勢のベクトルの向きとの差分を求め，この差分をスイングツイスト座標系で表した．

ここで取得したデータ点をスイングツイスト角座標系でプロットしたものを図4.3に示す．

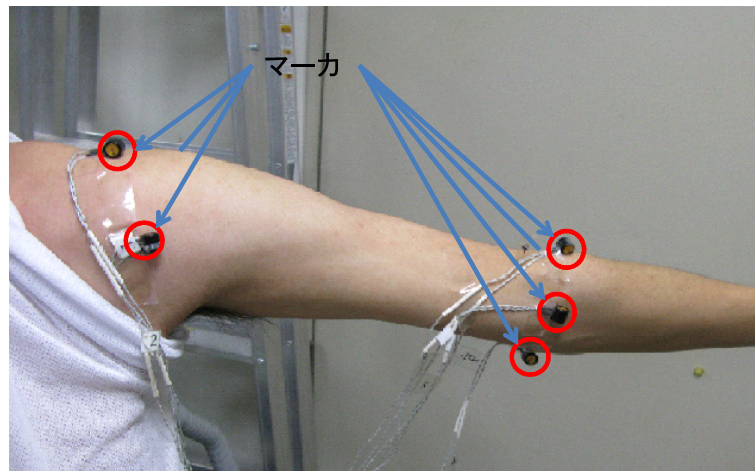


図4.2 マーカの配置

4.1.2 動きの評価

4.1.1節で取得した人の肩関節の可動域を図4.1のバーチャルヒューマンの肩関節に組み込む．球関節の可動域の設定は計測した可動域（図4.3）の外周に合わせて拘束点を指定し曲線で補間した．図4.4に指定した拘束点と可動範囲を示す．また，この可動域を図4.5のようにキャラクタモデル上に表示し，可動域の確認をした．

このバーチャルヒューマンの手とポインタをばねダンパでつなぎ肩の周りを一回転するようにポインタを動かしたときの腕の軌道を記録した．

4.1.3 結果

肩関節は前に広い可動域があり後側にはあまり可動範囲がない．そのため手を前から後ろに動かしたときに可動域の有無で動きに違いが出た．本研究で作った可動域拘束を与えたときには，図4.6のように無理な姿勢を取らずに腕を回転させることができた．一方可動域拘束を与えない場合は，図4.7の4~6コマのように不自然な姿勢をとることがあった．

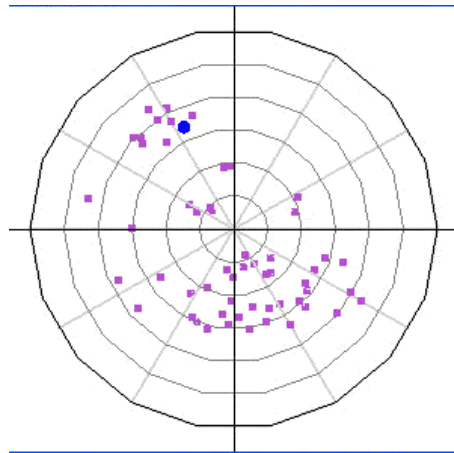


図 4.3 計測した関節可動域

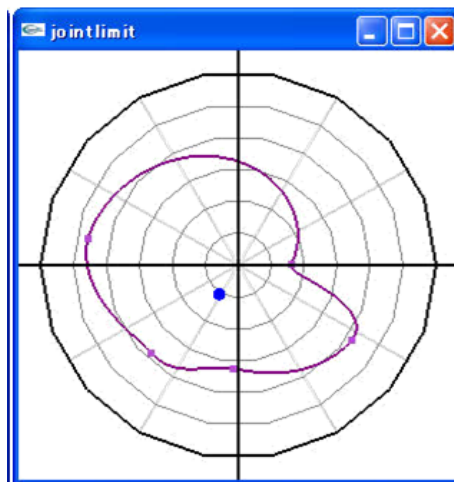


図 4.4 関節可動域の組み込み

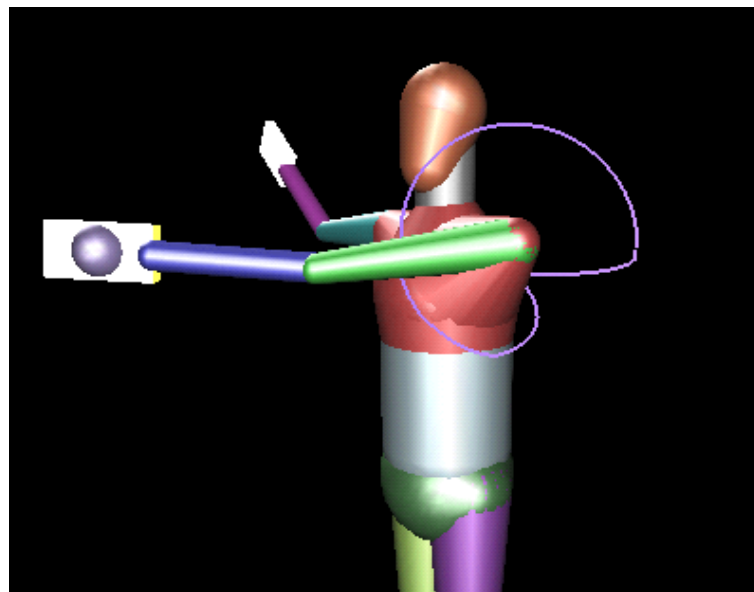


図 4.5 バーチャルヒューマンでの関節可動域表現

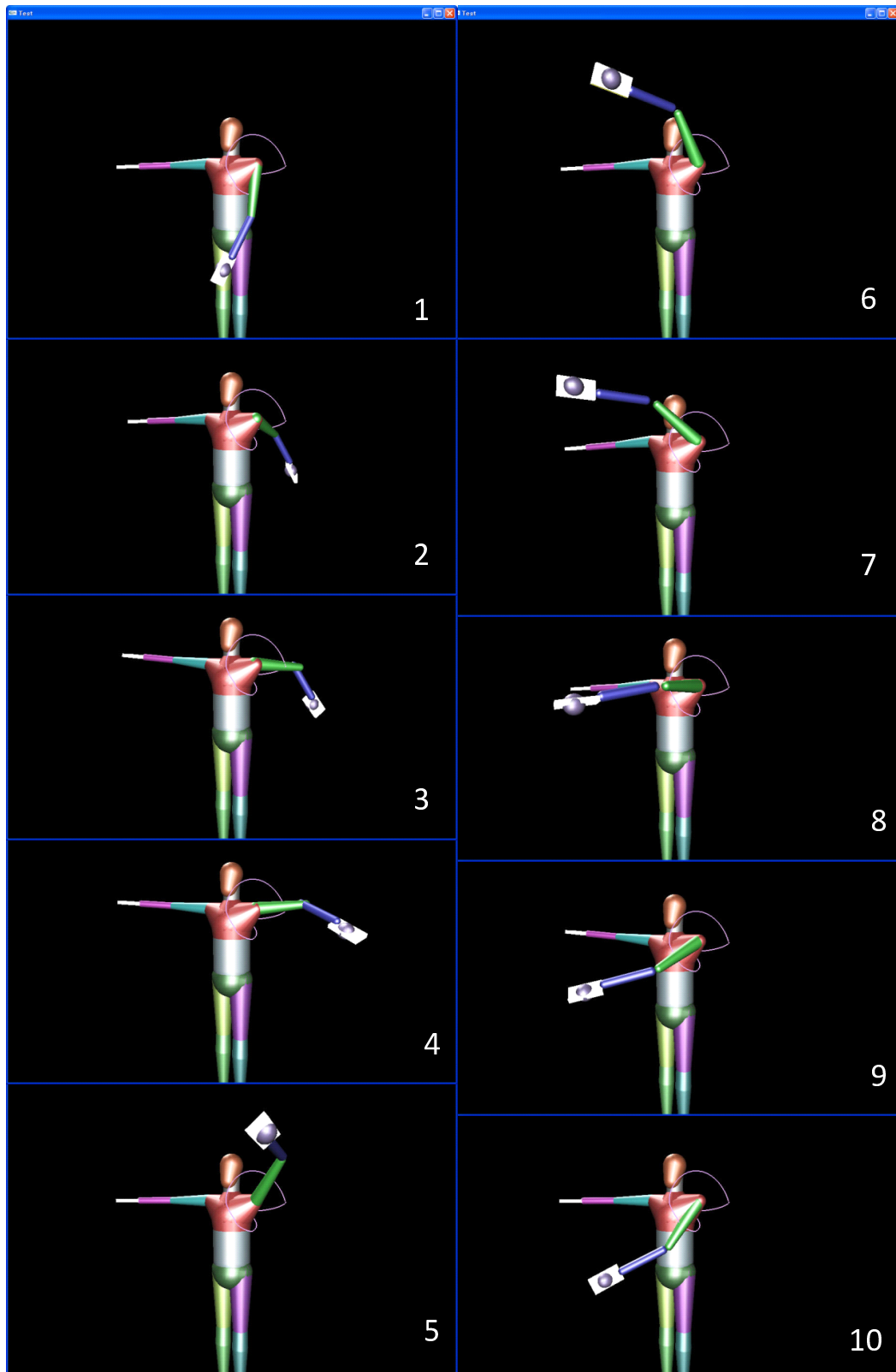


図4.6 ポインタを動かした時の様子(拘束あり)

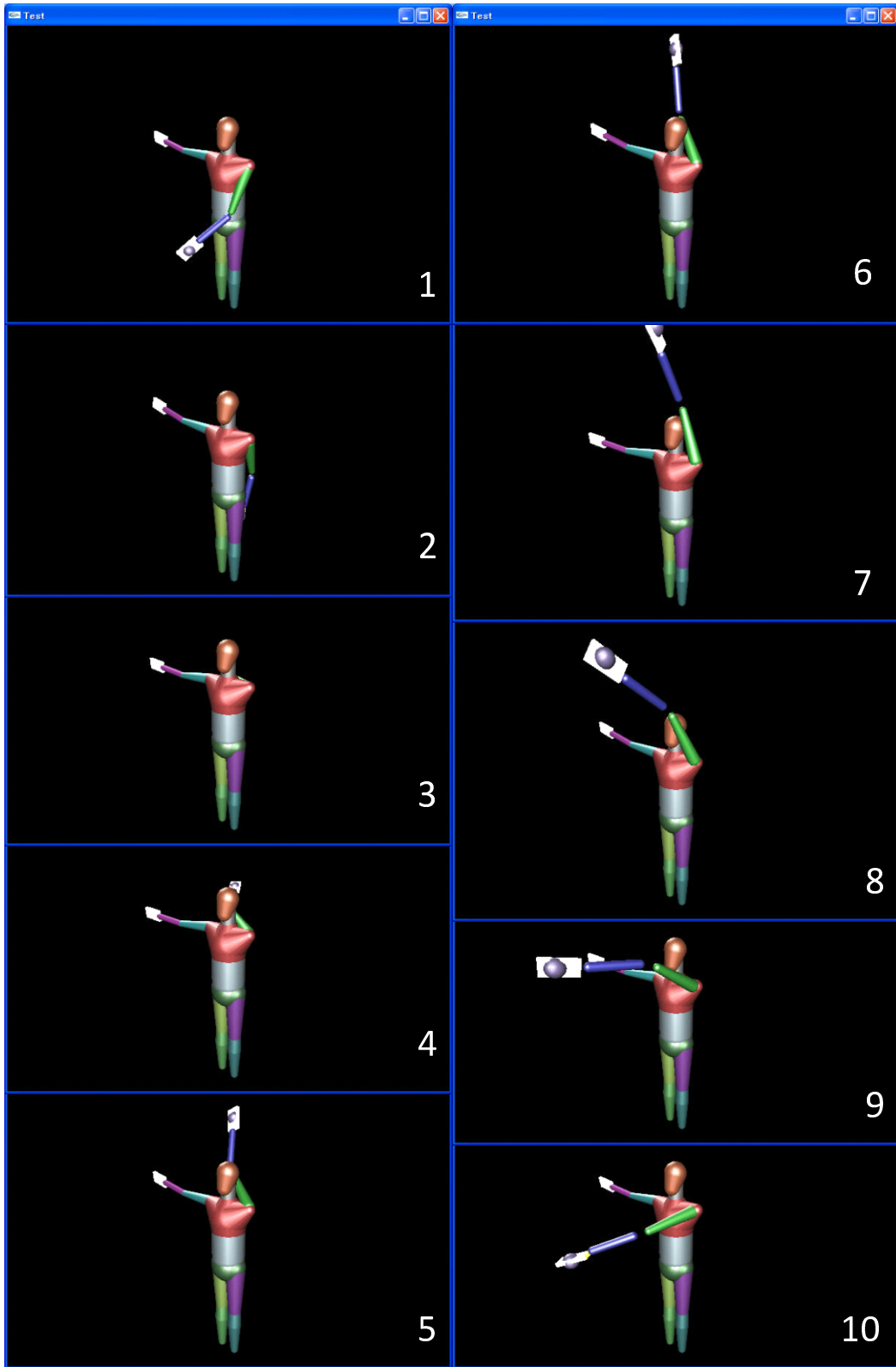


図4.7 ポインタを動かした時の様子(拘束なし)

4.2 キャラクタモデルと物理シミュレーション

三武らの制作したくま型のキャラクタ[17]に可動域拘束を加え、動きを確認した。このキャラクタは逆運動力学によって関節角の目標値を計算し、ばねダンパによって関節を動かし動作生成がされている。可動域の制限とばねダンパによるPD制御を同時におこなうことができるのか、また動きに変化が出るかどうかについて評価した。

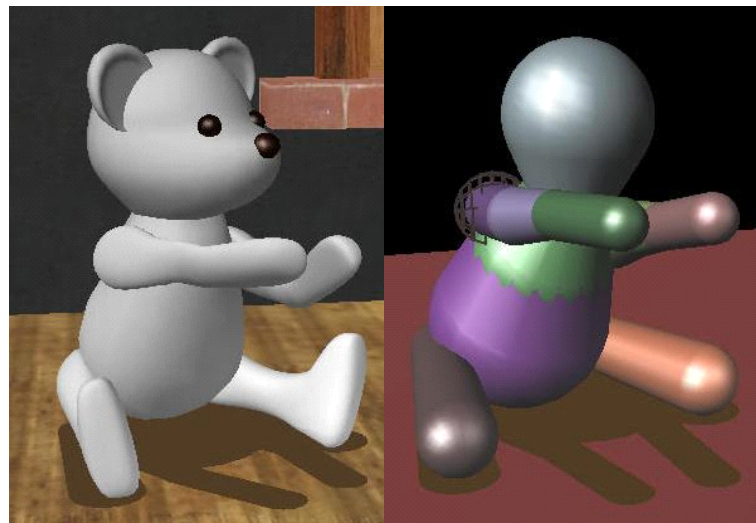


図4.8 くまモデル(右),物理モデル(左)

実験では、キャラクタが手でポインタを追いかけるように設定し、ポインタを右から左に動かした際のキャラクタの動きを観察した。

4.2.1 結果

図4.9に生成された動作を示す。逆運動学に基づく関節角のPD制御を行っている場合でも可動域拘束ができていることが分かる。また、可動域拘束がある場合とない場合で腕ののびし方などに違いがみられた。可動域拘束の有無で計算周期には変化が見られなかったため、計算量をあまり増やすことなく実装できていることが分かった。

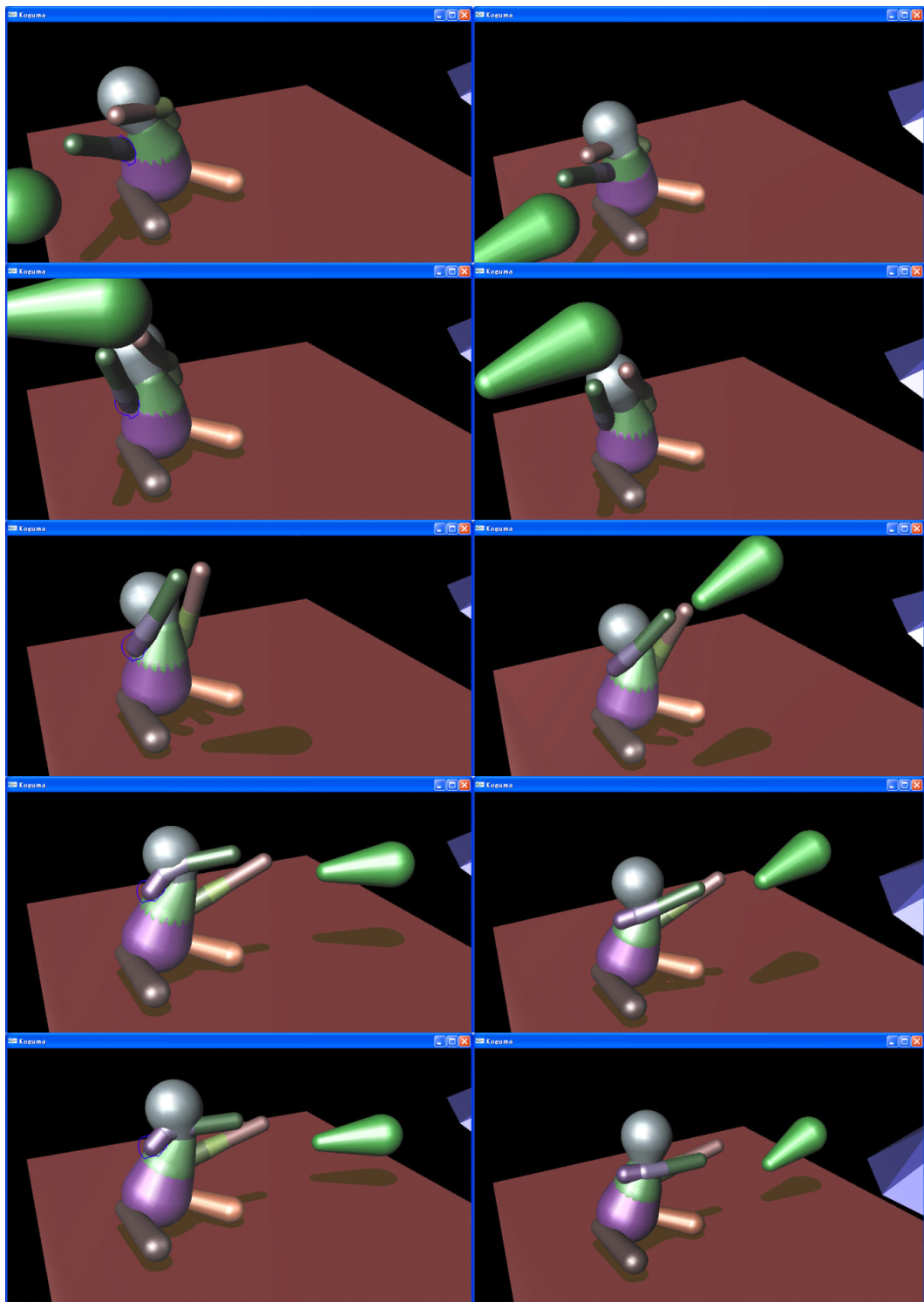


図 4.9 くまモデルの動き，可動域拘束あり(左)，なし(右)

第5章

結論

本論文では解析法の物理シミュレータにおいて、複雑な可動域拘束を実現できる関節モデルについて考え、拘束を指定する座標系と別に拘束力を計算するための座標系を用意することで拘束条件を線形相補性条件に帰着させ、物理シミュレータの拘束条件の一種として可動域拘束を物理シミュレータに組み込んだ。シミュレータへの組み込みの際には物理シミュレータで用いられている拘束条件と同様な条件を用いることで計算量をあまり増やさずに可動域拘束を実装することができた。また、実際に人の肩関節の可動域を取得し、バーチャルヒューマンの肩関節へ組み込みをおこなった。このバーチャルヒューマンをシミュレータ上で動かし不自然な姿勢を取らず、実際の人間の動きに近い動きが生成できることを確かめた。また、ばねダンパによる力と同時に可動域拘束による拘束力を出した場合にも可動域拘束をすることができ、逆運動学とPD制御を用いて動作生成を行うキャラクタモデルに適用できることを確かめた。

5.1 今後の展望

本論文では球関節を対象としたため、回転の3自由度についての可動域拘束をおこなった。可動域の拘束軸が計算できれば回転運動だけでなく並進運動や、並進と回転運動の混ざった関節に対しても可動域拘束ができ、より正確に生物の関節の再現ができると考えられる。

参考文献

- [1] Michael McKenna, David Zeltzer , ”Dynamic simulation of autonomous legged locomotion” , ”SIGGRAPH ’90”,1990
SIGGRAPH 90
- [2] Havok: ”Havok Physics”, <http://www.havok.com/>,1998-
- [3] AGEIA Technologies Inc.: ”PhysX SDK”, <http://www.ageia.com/>,2005-
- [4] M. Moore, J. Wilhelms. “ Collision Detection and Response for Computer Animation ”, Computer Graphics(SIGGRAPH 88), Vol. 22, pp. 289-298, August 1988.
- [5] D. Baraff. “ Analytical methods for dynamic simulation of non-penetrating rigid bodies ”, Computer Graphics Proceedings(SIGGRAPH 89) Vol. 23, pp. 223-232, 1989.
- [6] Shoichi Hasegawa , Naoki Okada , Jiro Baba , Yuuichi Tazaki , Hiroshi Ichikawa , Akihiko Shirai , Yasuharu Koike , Makoto Sato . ”Springhead: Open source haptic software for virtual worlds with dynamics simulations”, ”Proceedings of the 4th International Conference EuroHaptics 2004”,pp . 6.5-6.7 , 2004 .
- [7] Russell Smith. ”Open Dynamics Engine”, <http://www.ode.org/>, 2001-
- [8] Kenny Erleben. ”WikiTissue”, <http://www.opentissue.org/>
- [9] 長谷川 晶一”物理シミュレーションの現状と未来” , ”エンタテインメントコンピューティング 2006” , 2006
- [10] ”Springhead Wiki” , <http://springhead.info/wiki/index.php?Springhead Wiki Top>

- [11] 山根 克, 中村 仁彦. "ヒューマンフィギュアの全身運動生成のための協応構造化インターフェース", "日本ロボット学会誌", Vol. 20, No 3, pp. 335-343, 2002.
- [12] 芹澤 隆史, 柳田 康幸. "バーチャルフィギュアを用いたアバター動作の直感的オーサリング", "日本バーチャルリアリティ学会大会論文集", Vol. 12, pp. 2A2-5, 2007.
- [13] Paolo Baerlocher, Ronan Boulic. "Parametrization and range of motion of the ball-and-socket joint", "Deformable avatars", pp. 180-191, 2000.
- [14] 時崎 崇. "4足多関節剛体モデルのシミュレーションによる起き上がりの動作生成" "卒業論文", 2007.
- [15] 田崎 勇一, 長谷川 晶一. "拘束法の動力学シミュレータのための安定なバネダンパモデル", "情報処理学会情報処理学会研究報告. グラフィクスとCAD研究会報告", Vol.2006, No91, pp.55-60, 2007.
- [16] F. A. Potra M. Anitescu. "A time-stepping method for stiff multibody dynamics with contact and friction." "Intl. J. for Numerical Methods in Engineering 2002", pp. 55:753-784, 2002.
- [17] 三武 裕玄, 青木 孝文, 長谷川 晶一, 佐藤 誠. "精緻なフィジカルインタラクションにおいて生物らしさを実現するバーチャルクリーチャの構成法" "エンタテインメントコンピューティング 2010" セッション 3B, 2010.

謝 辞

本研究を遂行するにあたり、適切な御指導と御助言を頂きました電気通信大学知能機械工学科横井浩史教授に深く感謝いたします。本研究を遂行するにあたり、適切な御指導と御助言を頂きました東京工業大学精密工学研究所長谷川晶一准教授に深く感謝致します。また、研究全体を通して助言を頂いた、東京工業大学佐藤研究室三武裕玄様に深く感謝致します。また、ともに研究活動に励んだ長谷川研究室の卒業生、院生の皆様に感謝致します。