

2008年度 卒業論文

3DCGにおけるタコの腕と脚の動きの  
自動生成に関する研究

指導教員：渡辺 大地 講師

メディア学部 ゲームサイエンスプロジェクト  
学籍番号 M0105040  
石塚 真貴男

2008年度 卒業論文概要

論文題目

3DCGにおけるタコの腕と脚の動きの  
自動生成に関する研究

メディア学部

学籍番号：M0105040

氏名

石塚 真貴男

指導  
教員

渡辺 大地 講師

キーワード

3DCG、モーション、タコ、  
軟体類、インバースキネマティクス

近年3DCGを用いたゲーム、アニメーションの発展は目覚しく、ハードウェアの進化に伴い、よりリアルなモーションやモデリングを表現することが可能となった。人間同様、動物の動きも研究されしばしば3DCGの中に登場する。本論文では中でもタコが移動する際の動きに注目した。タコの腕・脚の動きに関する研究は少なく3Dのタコのモデルに腕・脚の動きつけるという作業は、実際のタコの観察に多くの時間を割いているのが現状である。また関節がなく、そのほとんどが筋肉でできているタコの腕・脚はとても複雑な動きをし、このような器官の動きの特徴を把握しアニメーションにするという作業はモーションデザイナーの知識・経験に基づく熟練度に依存し、制作時間やコストが大きくなる。本研究ではタコの腕に擬似的な関節をいれ、これにインバース・キネマティクスの発見的方法 (Heuristic method) を用いることで任意の位置まで自動で腕を動かす手法を提案する。タコの腕という自由な曲線を制御するために、動作に最低限必要だと考えられる節の関節群を設定しそれぞれを擬似ターゲットの位置まで動かすことでタコの腕の動作を制御した。状況によって変わる一連の移動動作を自動生成したコンテンツをC++で開発し、実際に手付けで生成したモーションまたは実写映像と自動生成で作成したモーションのクオリティ、作業量で比較し実用性を検証した。

# 目次

第1章	はじめに	1
1.1	研究の背景と目的	1
1.2	本論文の構成	3
第2章	タコの生態学的な移動手法の分析	4
2.1	タコの生態と腕・脚の種別	4
2.2	状況によるタコの移動手法の分類	5
第3章	本研究の腕・脚の動きの生成手法	8
3.1	使用したタコの3Dモデルの設定	8
3.1.1	使用したタコの腕の3Dモデル	8
3.1.2	インバース・キネマティクス	10
3.1.3	擬似ターゲットの設定	13
3.2	腕の移動	15
3.3	歩行・遊泳動作への対応	16
3.3.1	歩行動作	16
3.3.2	遊泳動作	18
3.4	擬似ターゲットの移動量の変化	20
3.5	移動ルートの設定と動作の対応	20
第4章	結果と考察	21
4.1	自動生成したタコの移動モーション	21
4.2	現状での問題点	23
第5章	まとめ	24
	謝辞	25
	参考文献	26

# 目 次

2.1	一連の歩行動作	6
2.2	一連の遊泳動作	7
3.1	腕の 3D モデル	9
3.2	使用したタコのモデル	10
3.3	発見的解法	12
3.4	アーム $i$ が誤差最小に達した場合	13
3.5	関節群ごとに異なる擬似ターゲット位置へ収束	14
3.6	歩行動作における 1 本の腕の動き	18
3.7	遊泳動作における 1 本の腕の動き	19
4.1	歩行動作 1	22
4.2	歩行動作 2	22
4.3	歩行動作 3	22
4.4	歩行動作 4	22
4.5	歩行動作 5	22
4.6	歩行動作 6	22
4.7	歩行動作 7	22
4.8	腕を持ち上げる	23
4.9	腕を振り下ろす	23
4.10	流線型を維持する	23
4.11	根元部分を持ち上げる	23

# 第 1 章

## はじめに

### 1.1 研究の背景と目的

近年の3次元コンピュータグラフィックス(以下3DCG)技術の発展により、3DCGを用いたアニメーションやゲームの制作はより活発なものになってきている[1]。またハードウェアの進歩によってキャラクターのより複雑なモーションや、ディテールにこだわったモデリングをすることが可能になった。それに伴いゲームだけでなく映画、テレビなど多くの映像作品で本物と見間違えるようなリアルな3DCGの動物が見られるようになった。3DCGにおける姿かたちのリアルな表現のみならず、馬や猫のような4足歩行の動物の動きの研究は大きく進歩した。特に歩行に関してはヒトに関する手法[2][3]はもとより多くの動物の歩行に関するアルゴリズムが明確になっている。しかしヒトや4足歩行の動物以外の軟体生物や虫といった複雑なメカニズムを持つ生物の動きに関する研究[4][5]は少ない。実際に動きを制作する際にはモーションデザイナーは逐一実際の映像を見て観察しCGのモデルにモーションをつけるという作業をしなければならない。本研究が対象とするタコの腕の動き方も同様である。また関節がなく、そのほとんどが筋肉でできているタコの腕はとても複雑な動きをし、どの方向にも伸ばしたり曲げたりできる。このような器官の特徴を把握しアニメーションにするという作業はモーションデザイナーの知識・経験に基づく熟練度に依存し、制作時間やコストが大きくかかる。

タコの腕を動かす手法として考えられるものにキーフレームアニメーション [6][7] とモーションキャプチャ [8]、ダイナミクス [9][10] がある。キーフレームアニメーションとは、フレームごとに位置や形状が変化するオブジェクトの情報を任意のキーとなるフレームにだけ登録しアニメーションを表現する手法である。キーフレーム間における位置や形状は、ファンクションカーブというオブジェクトの状態変化をあらわす線を基にして、コンピュータが計算する。ファンクションカーブを編集することで、キーフレーム間の動きが変化し、オブジェクトの加速や減速を表現できる。この方法では前述したように、モーションデザイナーの経験にアニメーションのクオリティが依存することに加え、任意の位置に腕を移動させる動きを自動で生成するということが不可能である。モーションキャプチャとは人体や動物にセンサーを設置し、動きを数値化しコンピュータに取り込む技術である。リアルタイムで画面上の CG キャラクターを動かしたり、人体の動作や表情をアニメーションの素材として活用したり、運動生理学などの研究の場で人体の動作の解析に利用されるなど、さまざまな現場で応用されている。しかし動物相手に制作者の求める動きをさせるのはそれだけで骨の折れる作業である。また実際には、制作者が求める動きを動物がしたとしても、必ずしも正確にそのデータが測定されているわけではない。そのためデータを補正するという作業もしなければならない。ダイナミクスとは物理シミュレーションとも呼ばれ、キャラクターなどの動きを任意に生成するキーフレーム法に対して、物理法則に基づいた動きを生成する手法である。例えばキャラクターがボールを投げるといったシーンの場合、ボールがキャラクターの手から離れ床でバウンドしてやがて静止するまでの動きは、実際の物理法則をあてはめて制作することができる。その他の例として、ばねの動き、花火の表現における微小な粒子（パーティクル）の動きなどがダイナミクスによって生成される。しかしタコのような生態を正確に記述することは困難であり、汎用的に実用できる段階には至っていない。一般的にこれらの手法を用いることでアニメーションを制作することは可能となるが、キーフレームアニメーションと同様に精度はモーションデザイナーの知識・経験に基づく熟練度

に依存してしまい、制作時間とコストが大きくなってしまふ。

本研究ではタコの腕の動きを自動で制御する手法を提案する。物に巻きついたり、蛸壺の中に入りこんだりと、さまざまな動作がある中で、本研究では特に移動する際の動きに注目した。移動の際における腕の動きを、地面に接地しているかどうかの違いを考慮した上で分析し、遊泳、歩行といった一連の動きを自動化する手法を提案する。それによりモーションデザイナーの作業を軽減する。本手法ではタコの腕に擬似的な関節をいれ、関節1つ1つを上下左右に回転させることで腕を動かす。任意の位置まで腕を移動させる手法としてインバース・キネマティクスを用いる。インバース・キネマティクスには大きく3つの代表的なアプローチがある。代数的に解く方法、反復によって解く方法、発見的に解く方法である。それぞれの考え方にのっとった計算方法がいくつも考案されている。本研究ではこの中でも発見的方法 (heuristic method) を用いたタコの腕の制御を行った。発見的方法の利点は他の2つに比べ収束するスピードが格段に早く、また関節の数に計算の複雑さが依存しないという点である。タコの腕の曲線を表現するためには多くの関節を必要とする。そのような関節体をリアルタイムで制御するには最も適した方法であるといえる。またタコの腕を比較的自由に制御するために、動作に最低限必要だと考えられる節を3つの関節群に分ける。分割した3つの関節群をそれぞれ擬似ターゲットの位置まで動かすことでタコの腕の動作を制御する。これらの機能を Fine Kernel ToolKit System[11] を利用して実装し、自動生成したタコの腕の動きを実写映像のタコの腕の動きと比較し本研究の有効性を検証した。

## 1.2 本論文の構成

第2章ではタコの生態学的な概要と、状況によるタコの移動手法の分析した調査結果を述べる。第3章では動作の実装手法を述べる。第4章では、実際に生成したタコの動作と既存手法とのクオリティを比較検証した考察を述べる。最後に第5章で、本研究の成果と今後の課題を述べる。

## 第 2 章

# タコの生態学的な移動手法の分析

本章ではタコの生態学的な特徴と観察結果からの移動手法について述べる。

### 2.1 タコの生態と腕・脚の種別

タコは学術的に「軟体動物門 頭足綱 八腕形上目」に分類される。体の構造としては、丸く大きな部分は胴体となり、足の付け根部分が頭となる。頭から足が生えていることになるので頭足類に分類される。そして長い触手状の 8 本の部位だが、学術的に「腕」と表現されることが多い。タコは眼を正面とし、左右に 4 対の腕を持っており、これらの腕は前から第 1 腕、第 2 腕、第 3 腕、第 4 腕と呼ばれている [12]。しかし近年 Thomas ら [13] の研究によりタコには 6 本の腕と 2 本の脚を持っているということが明かされた。タコの腕と脚には、骨や関節といったものは存在せず、柔軟な筋肉で構成されている。そのためどのような形にも曲げることができる。また多少の伸縮もすることが可能である。タコの特徴として一般的に考えられるのは吸盤の存在である。吸盤の中央に小さな口のような凹みがあり、この容積を筋肉の収縮で増減することで圧着する。これによりタコは腕を絡めることなく、物を引き寄せたり、吸い付けたりすることができる。

## 2.2 状況によるタコの移動手法の分類

実際のタコの動きを観察した結果から移動方法を大きく2つに分類した。タコの移動方法としては、遊泳動作と歩行動作が挙げられる。これら2つの動作であるが、タコの体が接地しているかいないかの違いにより腕の動かし方が変わる。まずタコが地面などに接している場合、タコは地面を這うように歩行動作をする。移動するために各腕で地面を押し推進力を得る。Thomasの研究では移動の際に第4腕で推進するために地面を押し、第1腕から第3腕で周りの探査および推進のサポートをする。また同様に、タコが海底を這う動作から泳ぐ動作に移行するとき、腕で地面を押し動作に使用されている。移動方向としては、主に眼の向いている正面方向となるが、前後左右に移動することも可能である。このとき各腕の動きとしては、先端よりも主に根元に近い部分で地面をけるなどの動作をする。先端に伸びるほど、根元の動作の影響を受けていると考えられる。図.2.1は対象物を観察し、実際にキーフレームアニメーションによって表現した一連の動きである。

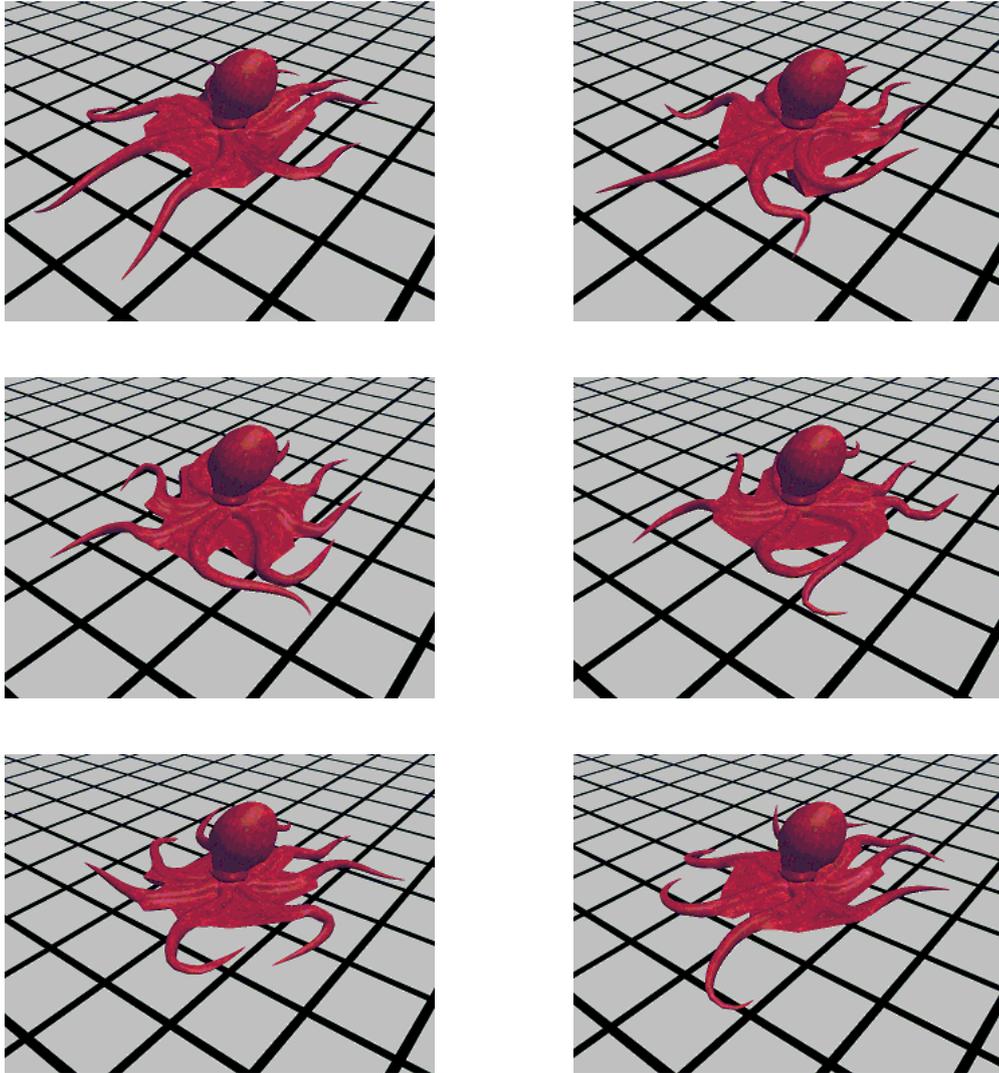


図 2.1: 一連の歩行動作

次にタコが地面に接していない、海中を漂っている場合について述べる。このときタコは腕を上下に動かし遊泳動作をする。移動するために各腕で水をかき泳ぐ。移動方向としては、主に頭の先の上方向へ向かう。各腕の動きとしては、1度腕を広げた状態から水をかき、一直線な流線型となる。その後、中央に近い部分から持ち上げていき動作を繰り返す。図.2.2 は歩行動作と同様に遊泳動作を実際にアニメーションとして表現した一連の動きである。

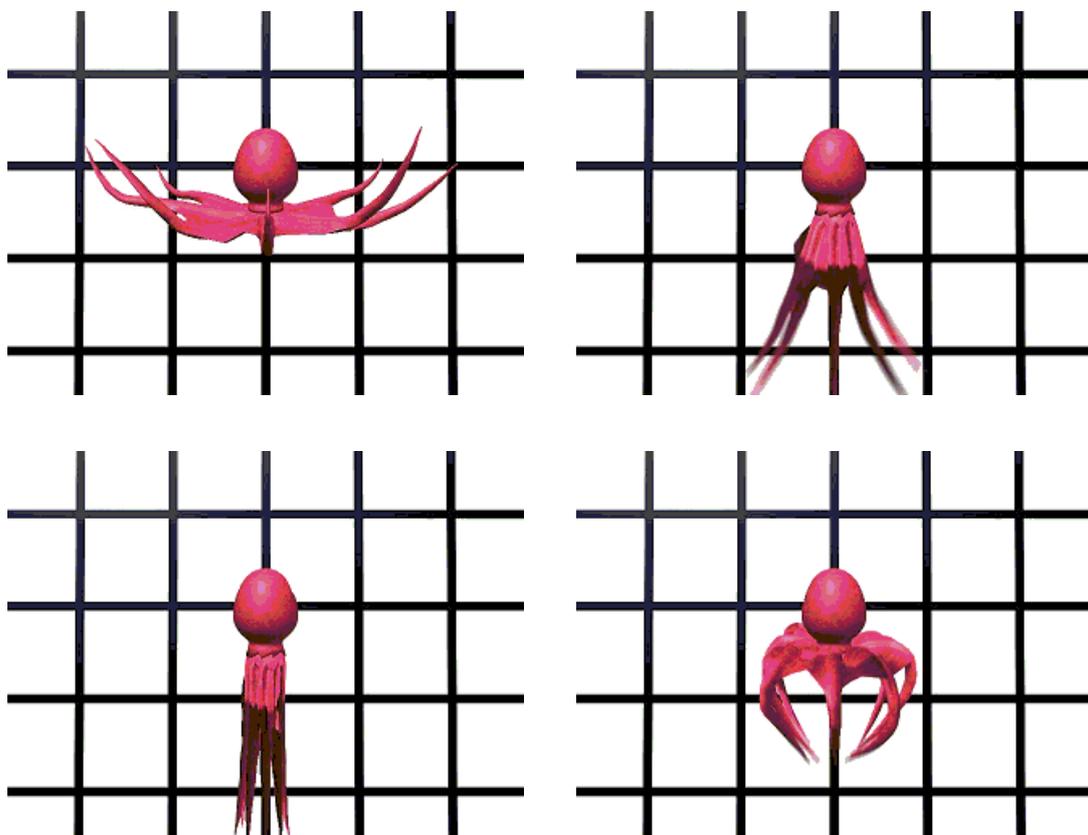


図 2.2: 一連の遊泳動作

## 第 3 章

# 本研究の腕・脚の動きの生成手法

本章では、提案手法について詳細に述べる。3.1 節では実装する 3D モデルの設定について述べる。3.2 節では腕の移動について述べる。3.3 節では擬似ターゲットを用いた移動動作への対応について述べる。

### 3.1 使用したタコの 3D モデルの設定

3.1.1 節では使用したタコの腕の 3D モデルについて述べる。3.1.2 節では実装に用いたインバース・キネマティクスについて述べ、3.1.3 節で擬似ターゲットについて述べる。

#### 3.1.1 使用したタコの腕の 3D モデル

実装に用いたタコの 3D モデルについて述べる。まずタコの腕に擬似的な関節を入れる。関節は腕の付け根から先端にかけて等比率で関節の幅が狭まるようにする。また本研究では関節の肉付けを簡易的に球体で表現している。本研究ではタコの腕の各関節を付け根側から  $P_i (i = 1, 2, \dots, n)$  とした。この時  $n$  は 3 の倍数である。本研究では  $n$  を 12 程度と想定した。また  $P_i$  を中心とする球の直径を  $w_i$  とするときそれぞれの球体の直径は式 (3.1) となる。タコの腕は先端ほど細くなって

いる。タコの腕は千切れたり、その後再生したりするため、明確な長さ、太さは個々によって変化するが本研究では考慮しない。

$$w_i = w_0(1 - 0.01i) \quad (3.1)$$

また  $P_i$  と  $P_{i+1}$  の幅、つまり節の長さを  $l_i$  とすると、それぞれの節の長さは式 (3.2) となる。

$$l_i = \frac{w_i}{2} \quad (3.2)$$

こうすることで腕はまっすぐにしたとき円錐形となり、また先端にいくほどよく曲がるという特性も考慮できる。図 3.1 は 1 本の腕のモデルを示す。

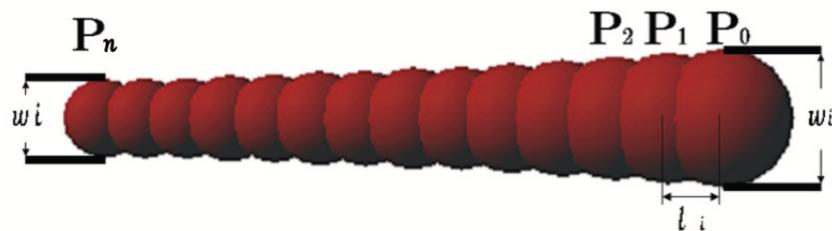


図 3.1: 腕の 3D モデル

$P_0$  はタコの頭の下とし、地面と平行に設定した。その腕を体の 8 分割である 45 度ずつ回転させ、8 本の腕として設定した。図 3.2 は使用したタコのモデルである。

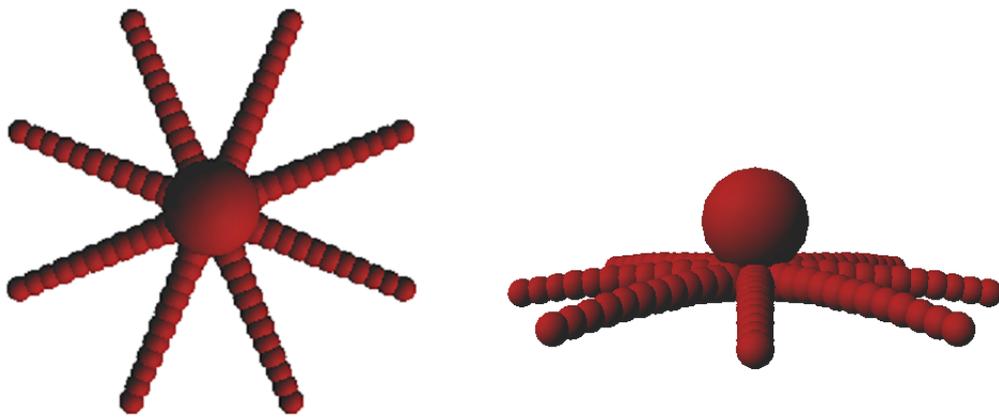


図 3.2: 使用したタコのモデル

また腕の関節には付け根から先端に向けてそれぞれ階層構造を持たせる。階層構造とは、人間の太ももを動かせば、その下にある足先全体はつられて動くように上の階層が動けば下の階層も追従する構造である。つまりタコの腕では根元部分の関節が回転すればそれより先端方向の関節がつられて動く構造である。

### 3.1.2 インバース・キネマティクス

逆運動学、インバース・キネマティクス (以下 IK とする)[14][15][16] とは人の腕や、ロボットのアームなどの関節を持つ物体の先端を任意の位置に移動させたいとき、どのくらい関節を曲げればその位置に到達するかを算出して関節物体を移動させる手法である。IK には大きく 3 つのアルゴリズムが存在する。ひとつは代数的解法 [17][18] である。アームの関節を  $\theta$  回転させるとアームの先端の位置が  $X$  に移動するという関数  $X = f(\theta)$  があるとき、逆関数  $\theta = f^{-1}(X)$  を求め回転する角度  $\theta$  を求める方法である。もうひとつは反復による解法 (Iterative Solution) である。これはヤコビアンという行列を用いてアームの回転と先端の位置を関数で表し、その逆関数を求めることで回転する角度を算出する。これら 2 つの方法は次

元数やアームの数が増えると式が非常に複雑になったり、解が無限に存在してしまうということが起こる。しかしアームの回転する範囲を制限するなど、いくつか条件 [19] を設定することで、人やロボットなど広い場面で取り入れることができる。最後が発見的方法 (Heuristic Method) である。本手法では、この発見的方法を基本とした。全部で節が  $i$  個ある関節体があり、先端の節を  $L_n$ 、 $L_n$  と  $L_{n-1}$  の関節を  $P_{n-1}$  とし、関節体全体の先端の位置を  $P_n$ 、先端を向かわせたい位置を  $A$  とする。ここで  $P_n$  と  $A$  の誤差について考えると、誤差は式 (3.3) になる。

$$E = |A - P_n| \quad (3.3)$$

そこでまず節  $L_n$  についてだけ誤差を最小にする。そのためには  $P_{n-1}$  を関節として節を角度ぶんだけ回転してやればよい。回転軸は  $P_n - P_{n-1}$  と  $A - P_{n-1}$  の外積ベクトルとする。図 3.3 は関節体の初期状態と、節  $L_n$  について、回転すべき角度を示す。角度は 2 つのベクトル  $P_n - P_{n-1}$ 、 $A - P_{n-1}$  の内積から求めることができる。

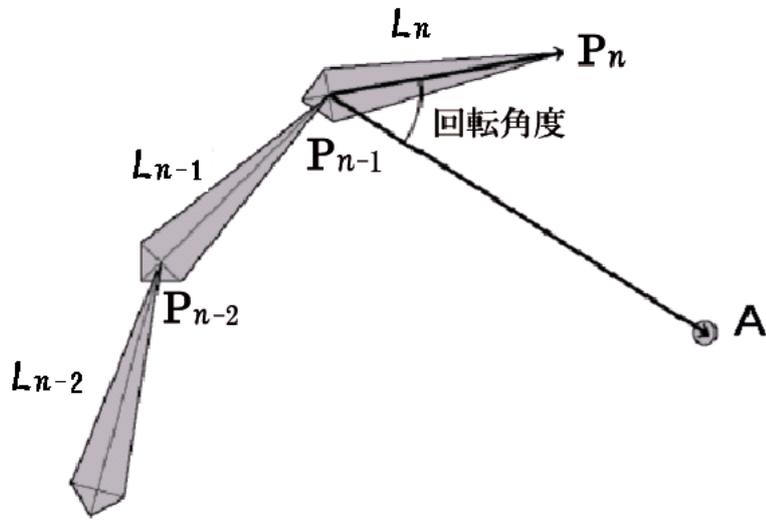


図 3.3: 発見的解法

次は節  $L_{n-1}$  についてみる。 $L_n$  の場合と同様に 2 つのベクトル  $P_n - P_{n-2}$ 、 $A - P_{n-2}$  の外積ベクトルを回転軸とし、内積から角度を求める。これを順に根元の節  $L_1$  まで繰り返して誤差を減らす。この誤差が 0 になればゴールに達したことになる。求めた結果と異なる場合は続けて節  $L_n$  から繰り返せばよい。図 3.4 は節  $L_n$  が誤差最小になったときの関節体の状態と、節  $L_{n-1}$  が回転すべき角度を示す。

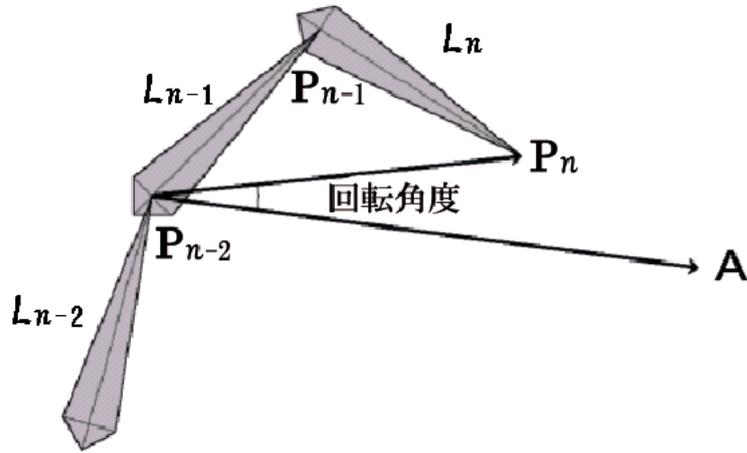


図 3.4: アーム  $i$  が誤差最小に達した場合

### 3.1.3 擬似ターゲットの設定

タコの腕は関節を持つ生物に比べ関節の制限がないため自由に動くことができ制御が難しいため、観察の結果から動作に最低限必要だと考えられる節の関節群を3つに分割し、各腕に移動用の擬似ターゲットを設定した。擬似ターゲットは  $Q_0$ 、 $Q_1$ 、 $Q_2$ 、 $Q_3$  とする。擬似ターゲット  $Q_0$ 、 $Q_1$ 、 $Q_2$ 、 $Q_3$  はそれぞれ  $Q_0$ 、 $Q_1$ 、 $Q_2$ 、 $Q_3$  を原点とした相対的な座標系を持つ。それをローカル座標とする。また各腕の関節は  $P_1$  から  $P_{\frac{n}{3}}$  をグループ1、 $P_{\frac{n}{3}+1}$  から  $P_{\frac{2}{3}n}$  をグループ2、 $P_{\frac{2}{3}n+1}$  から  $P_n$  をグループ3の3つの関節群に分け、グループ1を  $Q_1$  へ、グループ2を  $Q_2$  へ、グループ3を  $Q_3$  へ収束させる。擬似ターゲット  $Q_0$  から  $Q_1$  の幅を  $H_1$  とし、球の直径  $w_i$  から幅は式 (3.4)、 $Q_1$  から  $Q_2$  の幅を  $H_2$  とし、 $H_2$  は式 (3.5)、 $Q_2$  から  $Q_3$  の幅を  $H_3$  とし、 $H_3$  は式 (3.6) とした。

$$H_1 = \sum_{k=0}^{\frac{n}{3}} \frac{w_k}{2} \quad (3.4)$$

$$H_2 = \sum_{k=\frac{n}{3}+1}^{\frac{2}{3}n} \frac{w_k}{2} \quad (3.5)$$

$$H_3 = \sum_{k=\frac{2}{3}n+1}^n \frac{w_k}{2} \quad (3.6)$$

図 3.5 は 3 つの擬似ターゲットとそれぞれに対応する関節を示したものである。

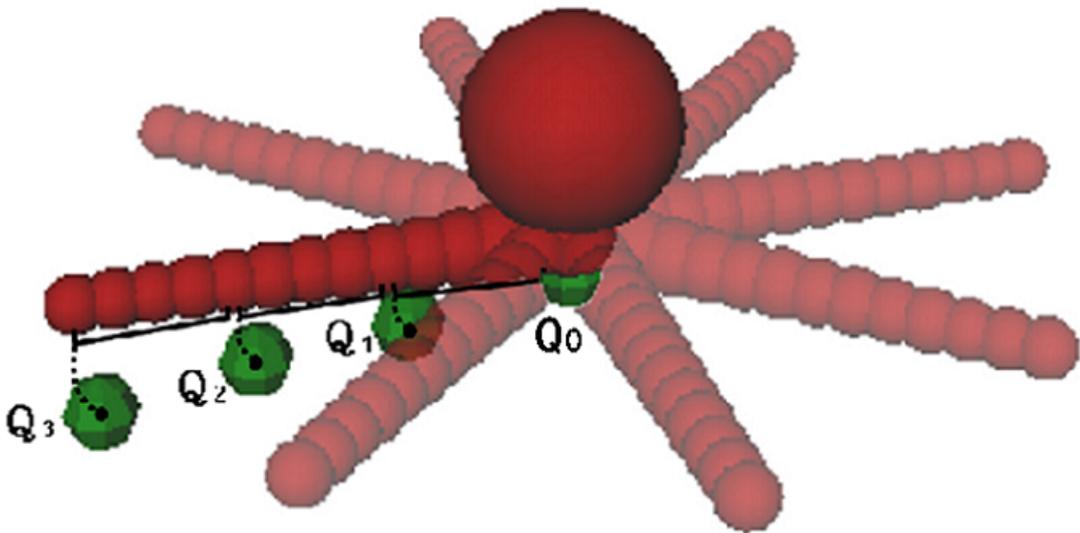


図 3.5: 関節群ごとに異なる擬似ターゲット位置へ収束

4つの擬似ターゲットに対しても付け根部分から先端に向けてそれぞれ階層構造を持たせる。そうすることで根元付近の動作が腕全体に影響する。擬似ターゲットの移動はローカル座標系の軸を回転軸とする3次元回転を組み合わせ制御する。

3次元回転にはヘッド角、ピッチ角、バンク角がある。具体的な角度の設定は3.3節で述べる。ヘッド角が上下の動き、ピッチ角が左右の動き、バンク角がねじれる動きを表す。擬似ターゲットの移動にはヘッド角とピッチ角を使用する。

## 3.2 腕の移動

腕の各関節  $P_i$  を回転させることで腕の形状を変える。本研究ではタコの吸盤がどちらをむいているかなどの姿勢を考慮せず、ただの曲線と考える。そこで任意の擬似ターゲット  $Q_1$ 、 $Q_2$ 、 $Q_3$  の位置まで腕を動かす方法として前述のIKの発見的方法を基に行う。発見的方法では一つの関節を回転させる際、各関節  $P_i$  において誤差が最小になる角度  $\xi_i$  ぶん回転する。この手法では先端の位置にある関節から計算が行われるため多関節体になればなるほど、先端に近い関節が優先的に動き、根元に近い関節はそれを補助するように動くような挙動となる。仮に先の位置にある関節角を変化させた際に、先端部分が指定の位置に、より近づいてしまうと、根元部分の関節の変化する角度が小さくなってしまう。つまりタコの腕のように多関節であればあるほど、根元部分の関節の変化は少なくなる。またこれらの手法を用いても期待する解が得られるとは限らない。そこで本研究では、IKの発見的方法によって得られた各関節の回転角度  $\xi_i$  に各関節ごとにある割合  $a_i$  を設定し、回転の大きさを調節した。割合  $a_i$  はユーザによって調節できるようにし、 $(0 < a_i \leq 1)$  を満たす範囲とする。これにより、期待する解が得られるとは限らないという問題を回避し、好きな関節を優先的に動かせるようにした。割合  $a_i$  の値が0に近ければ、根元部分の関節の変化が大きく、 $a_i$  の値が1に近ければ先端部分の関節の変化が大きくなってしまう [20]。各関節  $P_i$  が回転する角度を  $\psi_i$  とし、各関節の回転角度は式 (3.7) となる。

$$\psi_i = a_i \xi_i \quad (0 < a_i \leq 1) \quad (3.7)$$

このようにIKを用い各関節  $P_i$  は対応する擬似ターゲットに収束する。対応す

る擬似ターゲットと各関節群の距離を適当な長さまで縮め任意の動作をし移動動作を行う。以降の動作の種別においては後で述べるが、これによってタコの移動動作の動きが実装可能となる。

### 3.3 歩行・遊泳動作への対応

前述した手法を使いタコの腕の動作を表現していく。歩行動作・遊泳動作における擬似ターゲットとの対応を示す。3.3.1節で歩行動作、3.3.2節で遊泳動作の実装の手法を説明し、3.4節で動作の移動量の設定について説明する。最後に3.5節でそれらをあわせた移動ルートの設定と動作の対応を説明する。

#### 3.3.1 歩行動作

歩行動作は擬似ターゲットのローカル座標系においてヘッド角、ピッチ角の回転を使用する。タコの歩行動作はヒトの歩行動作と同様に繰り返しの運動なので移動動作の1周期を $2\pi$ と設定する。擬似ターゲットの移動は、時間を $t$ とした場合に、 $Q_1$ のピッチ角 $\alpha_1$ に式(3.8)、 $Q_2$ のピッチ角 $\alpha_2$ に式(3.9)、 $Q_3$ のピッチ角 $\alpha_3$ に式(3.10)を与える。

$$\alpha_1 = \begin{cases} \frac{5}{18}t & (0 \leq t < \frac{\pi}{2}) \\ \frac{5}{36}\pi - \frac{5}{6}(t - \frac{\pi}{2}) & (\frac{\pi}{2} \leq t < \pi) \\ -\frac{5}{18}\pi - \frac{5}{18}(t - \pi) & (\pi \leq t < \frac{3}{2}\pi) \\ -\frac{5}{12}\pi + \frac{5}{6}(t - \frac{3}{2}\pi) & (\frac{3}{2}\pi \leq t < 2\pi) \end{cases} \quad (3.8)$$

$$\alpha_2 = \begin{cases} -\frac{4}{3}t & (0 \leq t < \frac{\pi}{2}) \\ -\frac{2}{3}\pi + \frac{20}{9}(t - \frac{\pi}{2}) & (\frac{\pi}{2} \leq t < \pi) \\ \frac{4}{9}\pi + \frac{2}{9}(t - \pi) & (\pi \leq t < \frac{3}{2}\pi) \\ \frac{5}{9}\pi - \frac{10}{9}(t - \frac{3}{2}\pi) & (\frac{3}{2}\pi \leq t < 2\pi) \end{cases} \quad (3.9)$$

$$\alpha_3 = \begin{cases} \frac{1}{3}t & (0 \leq t < \frac{\pi}{2}) \\ \frac{\pi}{6} - \frac{2}{3}(t - \frac{\pi}{2}) & (\frac{\pi}{2} \leq t < \pi) \\ -\frac{\pi}{6} - \frac{1}{3}(t - \pi) & (\pi \leq t < \frac{3}{2}\pi) \\ -\frac{\pi}{3} + \frac{2}{3}(t - \frac{3}{2}\pi) & (\frac{3}{2}\pi \leq t < 2\pi) \end{cases} \quad (3.10)$$

上記のように擬似ターゲットを回転し制御することで、タコの腕の動きを制御する。また回転角度に応じて擬似ターゲットの1つ上の階層の擬似ターゲットに近づける。そのため擬似ターゲットの幅の長さに係数  $e$  をかけることで幅の長さを変化する。係数  $e$  の式は (3.11) とし、 $Q_0$  から  $Q_1$  の幅の変位を式 (3.12)、 $Q_1$  から  $Q_2$  の幅の変位を式 (3.13)、 $Q_2$  から  $Q_3$  の幅の変位を式 (3.14) とする。

$$e = - \left| \frac{1}{2\pi} F \right| \quad (F = \alpha_1, \alpha_2, \alpha_3) \quad (3.11)$$

$$H_1 = e \sum_{k=0}^{\frac{n}{3}} \frac{w_k}{2} \quad (3.12)$$

$$H_2 = e \sum_{k=\frac{n}{3}+1}^{\frac{2n}{3}} \frac{w_k}{2} \quad (3.13)$$

$$H_3 = e \sum_{k=\frac{2n}{3}+1}^n \frac{w_k}{2} \quad (3.14)$$

以上の手法により図 3.6 のようにタコの腕の曲線に近い動きになる。

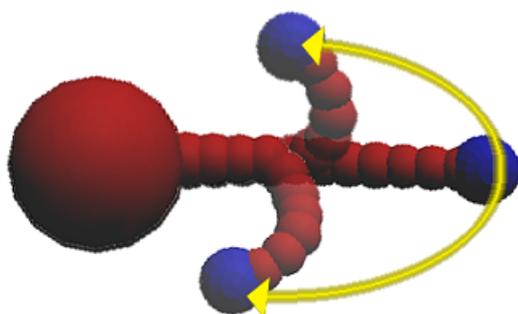


図 3.6: 歩行動作における 1 本の腕の動き

これら 3 つの擬似ターゲットの動作を基準とし、第 1 腕、第 2 腕、第 3 腕、第 4 腕の性質を考慮し、各腕の回転を調整する。これらの動きを基本の動作として、歩行動作の実装が可能となった。

### 3.3.2 遊泳動作

遊泳動作は擬似ターゲットのローカル座標系においてヘッド角を回転させる。遊泳動作も繰り返しの運動なので移動動作の 1 周期を  $2\pi$  と設定する。擬似ターゲットの移動は、時間を  $t$  とした場合に、 $Q_1$  のヘッド角  $\beta_1$  に式 (3.15)、 $Q_2$  のヘッド角  $\beta_2$  に式 (3.16)、 $Q_3$  のヘッド角  $\beta_3$  に (3.17) の式を与える。

$$\beta_1 = \begin{cases} \frac{1}{9}t & (0 \leq t < \frac{\pi}{2}) \\ \frac{\pi}{18} - (t - \frac{\pi}{2}) & (\frac{\pi}{2} \leq t < \pi) \\ -\frac{4}{9}\pi - \frac{25}{18}(t - \pi) & (\pi \leq t < \frac{3}{2}\pi) \\ -\frac{\pi}{36} + \frac{1}{18}(t - \frac{3}{2}\pi) & (\frac{3}{2}\pi \leq t < 2\pi) \end{cases} \quad (3.15)$$

$$\beta_2 = \begin{cases} \frac{1}{6}t & (0 \leq t < \frac{\pi}{2}) \\ \frac{1}{12}\pi - \frac{2}{9}(t - \frac{\pi}{2}) & (\frac{\pi}{2} \leq t < \pi) \\ -\frac{\pi}{36} - \frac{4}{9}(t - \pi) & (\pi \leq t < \frac{3}{2}\pi) \\ -\frac{\pi}{4} + \frac{1}{2}(t - \frac{3}{2}\pi) & (\frac{3}{2}\pi \leq t < 2\pi) \end{cases} \quad (3.16)$$

$$\beta_3 = \begin{cases} \frac{1}{6}t & (0 \leq t < \frac{\pi}{2}) \\ \frac{\pi}{12} - \frac{1}{6}(t - \frac{\pi}{2}) & (\frac{\pi}{2} \leq t < \pi) \\ -\frac{2}{3}(t - \pi) & (\pi \leq t < \frac{3}{2}\pi) \\ -\frac{\pi}{3}\pi + \frac{2}{3}(t - \frac{3}{2}\pi) & (\frac{3}{2}\pi \leq t < 2\pi) \end{cases} \quad (3.17)$$

上記のように擬似ターゲットを回転し制御することで、タコの腕の動きを制御する。遊泳動作においては第1腕から第4腕全てに動作が適用される。図3.6が遊泳動作における1本の腕の動きである。

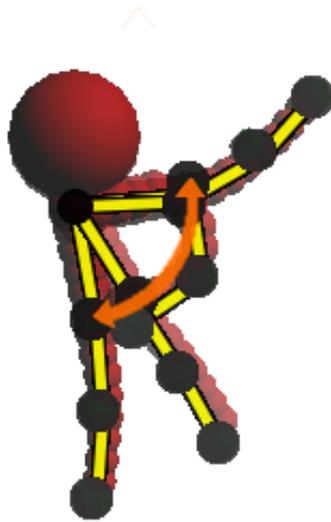


図 3.7: 遊泳動作における1本の腕の動き

基本動作においてピッチ角で制御することはほぼ必要ない。3.5節で説明する目標地点と動作の対応にピッチ角の回転を加味する。

### 3.4 擬似ターゲットの移動量の変化

これまでの手法によりタコの動作は実現可能だが一定の動作を繰り返してしまう。そこで8本の腕の動きにばらつきを持たせるために擬似ターゲットの移動量に変化をもたせる。この変化量を  $R(j, k)$  とする。ただし  $R(j, k)$  は  $j \leq r < k$  を満たす一様分布乱数  $r$  を返す関数とする。3.3節で設定した遊泳動作、歩行動作の各回転角度  $\alpha_1, \alpha_2, \alpha_3, \beta_1, \beta_2, \beta_3$  に変化量  $R(j, k)$  を加え、8本の腕の動きにばらつきをもたせる。ばらつきを持たせることでタコの8本の腕の動きのよりリアルな動作を行う。

### 3.5 移動ルートの設定と動作の対応

本研究ではタコの移動ルートを Bézier 曲線 [21] で入力した。Bézier 曲線の式は (3.18) となる。

$$C(s) = \sum_{i=1}^n C_i s^i (1-s)^{n-i} \mathbf{P}_i \quad (3.18)$$

本研究では制御点を四つ設け、これらの位置情報に変更を加えることによりタコの進行ルートを設定している。B を制御点とした場合、3次 Bézier 曲線は以下の式 (3.19) により決定する。

$$C(s) = (1-s)^3 \mathbf{B}_0 + 3s(1-s)^2 \mathbf{B}_1 + 3s^2(1-s) \mathbf{B}_2 + s^3 \mathbf{B}_3 \quad (3.19)$$

タコモデルを Bézier 曲線上にそって移動するように設定し、速度は以下の式 (3.20)、加速度は以下の式 (3.21) からそれぞれ求める。

$$V(s) = (1-s)^2 (\mathbf{B}_1 - \mathbf{B}_0) + 2s(1-s) (\mathbf{B}_2 - \mathbf{B}_1) + s^2 (\mathbf{B}_3 - \mathbf{B}_2) \quad (3.20)$$

$$\mathbf{A}(s) = (1-s) (\mathbf{B}_2 - 2\mathbf{B}_1 + \mathbf{B}_0) + s (\mathbf{B}_3 - 2\mathbf{B}_2 + \mathbf{B}_1) \quad (3.21)$$

# 第 4 章

## 結果と考察

### 4.1 自動生成したタコの移動モーション

本章では、第 3 章で述べた手法を実装した結果を示すと同時に、本手法の有用性の検証を行う。図 4.1、4.2、4.3、4.4、4.5、4.6、4.7 はタコが歩行動作する際の一連の動きである。一連の動作の図は IK を使い擬似ターゲットに向き動く動作をするものである。タコを歩行動作で移動させる場合、本研究で自動制御した腕の動きは、目的位置への移動として比較的リアルで、スムーズな動きが実現できた。

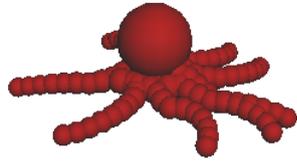


図 4.1: 歩行動作 1



図 4.2: 歩行動作 2

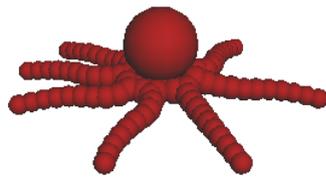


図 4.3: 歩行動作 3

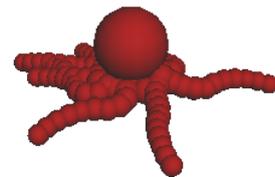


図 4.4: 歩行動作 4

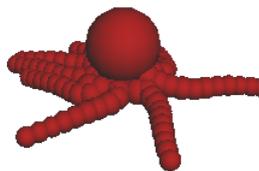


図 4.5: 歩行動作 5

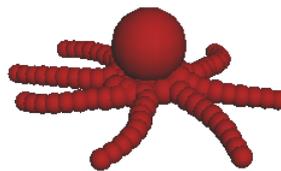


図 4.6: 歩行動作 6

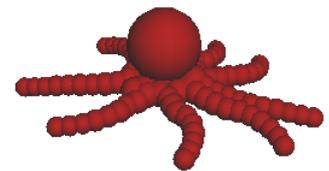


図 4.7: 歩行動作 7

図 4.8、4.9、4.10、4.11 は海底を漂う遊泳動作のときの一連の動作結果である。一連の動作の図は歩行と同様に IK を使い擬似ターゲットに向き動く動作をしたものである。図は第 2 章で述べたタコの動きのキーポーズの順になっている。



図 4.8: 腕を持ち上げる

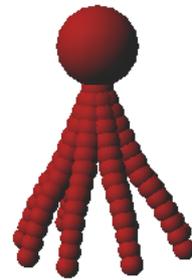


図 4.9: 腕を振り下ろす



図 4.10: 流線型を維持する

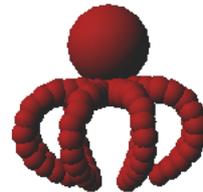


図 4.11: 根元部分を持ち上げる

## 4.2 現状での問題点

今回の研究ではタコの腕の関節群を3分割し、分割した関節群をIKを使い擬似ターゲットに向き、擬似ターゲットのヘッド角とピッチ角を制御することでタコの腕の動作をあらわした。しかし更なるタコの動きの滑らかさや繊細さを出すためには、ねじるといった動作や、吸盤の向きといったものを考慮しなければならない。そのためタコの腕自身の回転として、ヘッド角とピッチ角、バンク角であらわすことが必要になってくる。現状ではIKを使うことで、腕のねじりという動作は考慮していないため改善の余地がある。また今回の実装結果では遊泳動作と歩行動作の切り替わりについて考慮していない。このつながりを改善することでさらなるタコの動作の向上になると考える。

## 第 5 章

### まとめ

本研究のベースとなった IK の発見的方法は一般的にクネクネとした挙動をし、人などの関節物体を動かすには不向きな手法である。しかしこれをタコの腕に応用し、腕に IK のグループを 3 つ作り動かすことでタコらしい動きが実現できた。またこの発見的方法は目的の位置への収束が速く、リアルタイムアプリケーションには効果的である。タコの脚を 1 本動かすのではなく、8 本連動させて動かさなければならないため、タコのような多関節体では有効であったといえる。今回の研究では移動する際の腕の動きにのみ焦点を当てたが、ものをつかむ、絡みつく、捕食するといったその他の動作を自動化することで、より実用的なタコのモーション生成が可能となる。

# 謝辞

本論文を締めくくるにあたり、ご指導頂きました本校メディア学部の渡辺大地講師に心より深く感謝いたします。また、本研究を進めるにあたり、相談に乗っていただいた研究室のメンバーおよび、多大な影響を与えてくださったゾウの鼻の動きに関する研究、研究者水野様に深く深く感謝いたします。最後に、私を支えてくれた家族と、全ての友人に感謝いたします。

## 参考文献

- [1] 東京工科大学クリエイティブラボ, 「デジタルアニメマニユアル 2008」, 東京工科大学編, 2008.
- [2] Petros Faloutsos, Michiel van de Panne, Demetri Terzopoulos, "Composable Controllers for Physics-Based Character Animation", SIGGRAPH, 2001.
- [3] Harold C. Sun, Dimitris N. Metaxas, "Automating gait generation", SIGGRAPH, 2001.
- [4] 水野照章, "3DCG におけるゾウの鼻の動きに関する研究", 東京工科大学, 2007.
- [5] 土永まり子, "3DCG における鳥のはばたきモーションの自動生成プログラム", 東京工科大学, 2006.
- [6] 「CGWORLD vol.32」, 骨と筋肉をまじめに考える 32-52, ワークスコーポレーション, 2001.
- [7] 尾沢直志, 「アニメ作画の仕組み キャラに命を吹き込もう!」, ワークスコーポレーション, 2004.
- [8] Adam Kirk, James F. O'Brien, David A. Forsyth, "Skeletal Parameter Estimation from Optical Motion Capture Data", SIGGRAPH, 2004.

- [9] デジタル映像表現編集委員会, 「デジタル映像表現-CGによる映像制作-」, 画像情報教育振興協会 (CG-ARTS 協会), 2004.
- [10] アイザック・ビクター・カーロウ, 「コンプリート 3DCG -デジタル映像に関わるすべての人に」, 株式会社エムディエヌコーポレーション, 2001.
- [11] 渡辺大地, FK Tool Kit System , <<http://fktoolkit.sourceforge.jp/>>.
- [12] 奥谷 喬司, 「泳ぐ貝、タコの愛 軟体動物のふしぎな生態」, 晶文社,1991.
- [13] David Thomas Octopuses have two legs and six arms,  
<<http://www.telegraph.co.uk/news/newsttopics/howaboutthat/2547597/Octopuses-have-two-legs-and-six-arms.html>>.
- [14] 久保田一郎, 宇治社中, インバースキネマティクス, 「C MAGAZINE」, 110-115,7 月号, 2001.
- [15] Jeff Landar, Game Developer Magazine,Sep 1998 Sep,1998 Nov,  
<<http://darwin3d.com/>>.
- [16] Dante Treglia, 中本 浩, 川西 裕幸, 「Game Programming Gems 3 日本語版」, 株式会社ボーンデジタル,2004.
- [17] Monacha D and Zhu Y, A Fast Algorithm and System for the Inverse Kinematics of General Serial Manipulators, "IEEE International Conference on Robotics and Automation", 1994.
- [18] Chin K, Closed-Form and Generalized Inverse Kinematic Solutions for Animating the Human Articulated Structure,"Bachelor 's Thesis in Computer Science",Curtin University of Technology, 1996.
- [19] Welman Chris, B.Sc.Simon Fraser University , "Inverse Kinematics and Geometric Constraints for Articulated Figure Manipulation",1989.

- [20] 滝沢崇, "3DCG における馬の代表的な 4 種の走法のなめらかな移り変わりの表現", 東京工科大学, 2005.
- [21] 鳥谷浩志 千代倉弘明, 「3 次元 CAD の基礎と応用」, 共立出版株式会社, 1991.