

# 線形ブレンドスキニングにおける補助骨の姿勢最適化

向井 智彦<sup>1,a)</sup>

**概要:** 実時間アプリケーションにおける CG キャラクタの皮膚変形表現には、線形ブレンドスキニング法が広く利用されている。さらに、補助骨と呼ばれる冗長なリグを導入することで、スキニングアニメーションの品質を改善するとともに、筋肉の隆起などの複雑な皮膚変形を表現する技法が実用化されている。しかし、補助骨のリギングにはデザイナーの熟練と試行錯誤が必要であり、膨大な作業コストが必要となっている。本論文では、スケルトン姿勢と目標皮膚形状の組み合わせを例示データとして与えることで、最適な補助骨姿勢とスキニングウェイトを推定する手法を提案する。

## 1. はじめに

ビデオゲームをはじめとする実時間 CG アプリケーションでは、キャラクタの皮膚変形アニメーションを線形ブレンドスキニング法 (Linear Blend Skinning, LBS 法) によって計算することが多い [1]。この手法は実装が容易で計算コストも小さいことから、多くのグラフィクスエンジンが採用している。ただし、線形ブレンドスキニング法には、関節の屈曲や骨周りのひねり回転にしたがって皮膚が縮んだり凹んだりするような、candy-wrapper 効果や elbow-collapse 効果と呼ばれる不具合を生じることが知られている。そこで、アニメーションデータやインバースキネマティクス等で駆動される骨に加えて、形状歪みを補正するための冗長な骨を追加し、スキニングの品質を改善する技法が普及している [2], [3], [4]。なお、本論文では前者を駆動骨、後者を補助骨と呼んで区別する。

補助骨の利用の模式図を図 1 に示す。補助骨を用いない場合には、図 1(a) に示すように、駆動骨のみの姿勢に基づいてメッシュを変形する。その際、スキニングウェイトの設定次第では、関節付近に elbow-collapse 効果による大きな凹みを生じる。一方、図 1(b) ではひじの凹みを補正するための補助骨と、上腕筋の隆起を模倣するための補助骨を追加している。そして、駆動骨の姿勢にしたがって 2 つの補助骨を移動、回転したうえで、駆動骨と補助骨の両方の姿勢にもとづいてスキニング形状を計算する。このように、補助骨は駆動骨の姿勢に応じて手続的に操作されることが多く、その制御モデルは簡単な数式やスクリプトとして記述されることが多い。

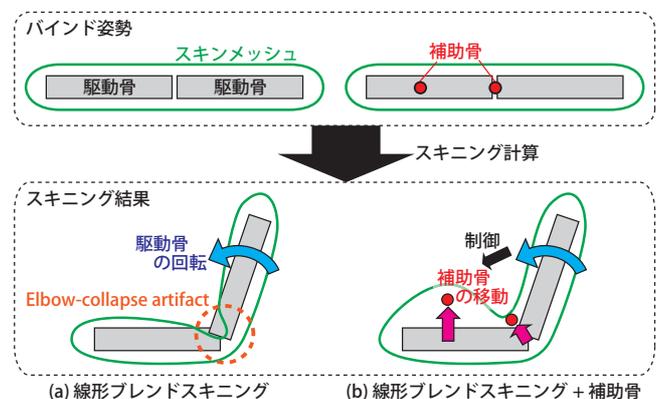


図 1 補助骨を用いた線形ブレンドスキニング

補助骨を用いたスキニングアニメーション生成技術の導入に際しては、基本的にキャラクタセットアップの工程にのみ変更を生じる。つまり、駆動骨を対象とするアニメーターの作業工程、および線形ブレンドスキニングを利用する一般的なグラフィクスパイプラインには影響を及ぼさない。また、補助骨の姿勢制御に要する追加計算コストは、スキニングの品質向上に十分見合うものと判断されることが多い。このように、補助骨の利用は標準的な実時間アニメーション制作工程との親和性が高く、導入コストや計算コストの面でも優れることから、国内外の開発現場で広く普及している。

ただし、補助骨のセットアップには熟練したリグデザイナーの試行錯誤を要する。補助骨の制御モデルは低次の多項式によって記述されることが多いが、いずれの駆動骨の姿勢を入力パラメータとして、何次の多項式を記述し、その係数をどのように決定すべきか、非技術者であるリグデザイナーが検討しなければならない。そのため補助骨セットアップ作業は、スキニング結果を見ながらの試行錯誤に

<sup>1</sup> 東海大学情報通信学部情報メディア学科  
Takanawa, Minato-ku, Tokyo 108-8619, Japan  
a) tmki@acm.org

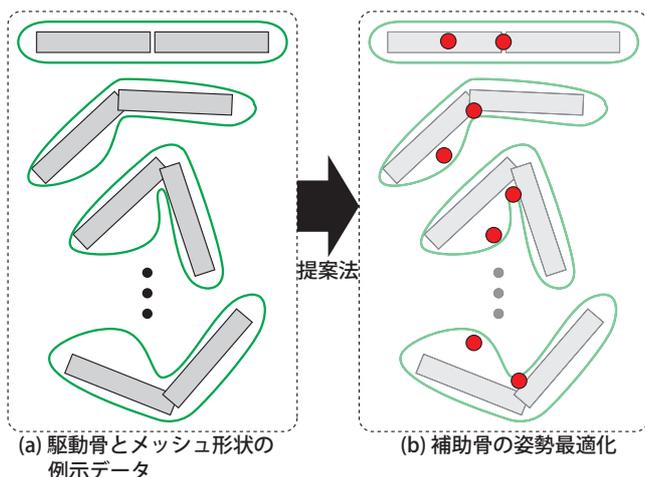


図 2 例示データを用いた補助骨姿勢の最適化

頼らざるを得ず、大きな作業負担となっている。

そこで本研究では、例示データを用いて補助骨のセットアップを半自動化するシステムを開発している。具体的には、ユーザーが駆動骨の姿勢と目標スキン形状の組み合わせを複数作成したうえで、導入する補助骨の数などの少数の計算パラメータを指定することで、最適な補助骨の配置と制御モデル、およびスキンウェイトを自動推定する手法について検討している。その中でも本報告では、与えられた例示データに対する最適な補助骨姿勢とスキンウェイトを推定する方法について報告する。なお、補助骨制御モデルの例示ベース推定については、今後の研究課題として位置づける。

## 2. 関連研究

線形ブレンドスキニングの改良法がいくつか提案されている。例えば、スキニングのウェイトを多次元化することで、骨の増加を最小化しつつ品質を向上する技術が提案されている [5]。また、骨の運動の表現形式として、アフィン変換行列ではなく双四元数を用いることで、LBS法の不具合を低減するスキニング法も提案されている [6]。同様に、関節の姿勢を複数の数学モデルを組み合わせることで、スキニング品質の向上を図る方法も提案されている [7], [8]。いずれの手法も、線形ブレンドスキニングより優れた結果を与えるが、筋肉の隆起や誇張された皮膚変形については考慮されていない。

基本的な補助骨のアイデアは、Morhら [2] によって提案された。この手法では、candy-wrapper 効果を低減するために、駆動骨の中間に挿入した補助骨のスケーリング成分を制御するアルゴリズムを提案している。さらに、ゲーム制作現場においては、補助骨の回転や平行移動も操作することによって、LBS法の不具合を改善するだけでなく、筋肉の隆起などの多様なスキン変形を実現する技法が発展している [3], [4]。しかし、補助骨の設計にはリグデザイナー

の試行錯誤を要する課題があり、何らかの自動化手法が求められている。例示メッシュ形状をもとにスキンウェイトを自動推定する手法も多数提案されているが、いずれも駆動骨や補助骨の姿勢は事前に与えられることを前提としている [2], [5], [9], [10], [11]。

本研究は、メッシュ形状の例示データ群をもとに骨姿勢を推定する点で、スキニング分解法と呼ばれるアルゴリズムと関連が深い。スキニング分解は、与えられた頂点アニメーションを近似するように、骨の配置と運動、そしてメッシュ各頂点のスキンウェイトを求める手法である。文献 [12] で発表されたこのアイデアは、次元削減法を用いて分解処理を高速に行う手法 [13] や、骨の姿勢行列を剛体変換に制限する方法 [14]、および階層化されたスケルトン骨格構造を推定する手法 [15] などに発展している。ただし、いずれの手法も頂点アニメーションをスキンアニメーションに変換すること目的としており、例示データに駆動骨の情報を含まない点で本研究と異なる。

また、駆動骨の姿勢から皮膚形状への直接写像を計算する手法も提案されている [16]。この手法では、駆動骨のスケルトン構造とアニメーション、および皮膚変形頂点アニメーションの例示データをもとに、駆動骨の回転量から各頂点座標への写像を表す線形回帰モデルを推定する。このとき、写像モデルは頂点ごとに決定されるため、デザイナーによる調整は特に高解像度メッシュでは困難である。一方、本研究では比較的少数の補助骨のみを扱うため、システムの推定結果を手作業で編集することも比較的容易である。

## 3. 問題設定

$D$  個の駆動骨で構成されるスケルトンについて、 $d$  番目の駆動骨のグローバル姿勢行列およびバインド姿勢行列を  $\mathbf{G}_d$ ,  $\mathbf{B}_d$ ,  $d = \{1, \dots, D\}$  と表すとき、スキニング行列は  $\mathbf{S}_d = \mathbf{B}_d^{-1} \mathbf{G}_d$  と定義される。ここで、バインド姿勢メッシュの  $j$  番目の頂点座標を  $\bar{\mathbf{v}}_j$ , スキンウェイトを  $\{w_{j,d}\}$  と表すとき、スキンメッシュの頂点座標は  $\mathbf{v}_j = \sum_d w_{j,d} \bar{\mathbf{v}}_j \mathbf{S}_d$  で計算される。ただし、 $\sum_d w_{j,d} = 1, \forall j$  を満たす。さらに、 $H$  個の補助骨を加え、そのスキニング行列とウェイトをそれぞれ  $\hat{\mathbf{S}}_h$ ,  $\{\hat{w}_{j,h}\}$ ,  $h = \{1, \dots, H\}$  と表すとき、スキンメッシュの頂点座標  $\mathbf{v}_j$  は次式で計算される。

$$\mathbf{v}_j = \sum_d w_{j,d} \bar{\mathbf{v}}_j \mathbf{S}_d + \sum_h \hat{w}_{j,h} \bar{\mathbf{v}}_j \hat{\mathbf{S}}_h. \quad (1)$$

ただし、 $\sum_d w_{j,d} + \sum_h \hat{w}_{j,h} = 1, \forall j$  を満たす。このとき、本研究の目標は、 $N$  組のサンプルメッシュ形状と駆動骨のグローバル姿勢行列の例示データ  $\{\bar{\mathbf{v}}_{j,n}\}$ ,  $\{\hat{\mathbf{G}}_{d,n}\}$ ,  $n = \{1, \dots, N\}$  が与えられたとき、各サンプルメッシュとスキンメッシュとの二乗誤差を最小化するようなスキンウェイト  $\{w_{j,d}\}$ ,  $\{\hat{w}_{j,h}\}$  と補助骨のスキニング行列  $\{\hat{\mathbf{S}}_{h,n}\}$

を求める問題である。すなわち、次の制約付き最小二乗問題が定義される。

$$\min_{\{w_{j,d}, \{\hat{w}_{j,h}\}, \{\hat{\mathbf{S}}_{h,n}\}} \sum_n \sum_j \left| \tilde{\mathbf{v}}_{j,n} - \sum_d w_{j,d} \tilde{\mathbf{v}}_j \mathbf{B}_d^{-1} \tilde{\mathbf{G}}_{d,n} - \sum_h \hat{w}_{j,h} \tilde{\mathbf{v}}_j \hat{\mathbf{S}}_{h,n} \right|_2 \quad (2)$$

$$\begin{aligned} \text{subject to } & \hat{\mathbf{S}}_{h,n} = \hat{\mathbf{R}}_{h,n} \hat{\mathbf{T}}_{h,n}, \quad \forall h, n \\ & w_{j,d} \geq 0, \quad \hat{w}_{j,h} \geq 0, \quad \forall j, d, h \\ & \sum_d w_{j,d} + \sum_h \hat{w}_{j,h} = 1, \quad \forall j \\ & \sum_d |w_{j,d}|_0 + \sum_h |\hat{w}_{j,h}|_0 \leq K, \quad \forall j \end{aligned}$$

なお、 $|\cdot|_0$  と  $|\cdot|_1$  はそれぞれ  $L^0$  ノルムと  $L^1$  ノルムを表す。ここでスキニング行列に関する1つめの剛体制約条件は、せん断変換などの回転と平行移動以外の座標変換は、一般的なアニメーションエンジンではサポートされないために課している。また、2つめ以降の制約条件は、それぞれ非負制約、partition of unity 制約、そしてスパース制約である。このスパース制約は、各頂点においてゼロ以外のスキニングウェイトを持つ骨の最大数を  $K$  以下に制限する。これは、LBS の計算式1において実際にブレンド対象となる行列の数に上限を設け、計算を高速化するために導入するものである。なお、一般的なグラフィクスエンジンでは  $K = 4$  とする場合が多い。

#### 4. 提案アルゴリズム

提案アルゴリズムの処理手順を、Algorithm 1 の擬似コードにまとめる。提案法では、駆動骨のスキニング行列と例示メッシュ形状のみが与えられた、補助骨が存在しない状態から処理を開始する。まず、駆動骨に応じたスキニング結果が例示メッシュ形状を近似するように、後述のスキニングウェイト更新アルゴリズムを用いて駆動骨に対するスキニングウェイトを初期化する。次に、補助骨を逐次的に1つずつ追加しながら、そのスキニング行列とスキニングウェイトを最適化する。そして全ての補助骨が追加されたら、Block coordinate descent アルゴリズム [14] を用いて、式2に示す制約付き最適化問題の近似解を求める。このアルゴリズムでは、各補助骨のスキニング行列とスキニングウェイトをそれぞれ順番に更新する。すなわち、スキニングウェイトを最適化するには全ての補助骨を固定し、またある1つの補助骨のスキニング行列を更新する際には、スキニングウェイトと残る  $H-1$  の補助骨を固定する。そして、スキニングウェイトと各補助骨のスキニング行列を順番に繰り返し更新することで、目的関数が単調減少するような収束計算を行う。

本章の以降では、スキニングウェイトの更新処理（擬似コードの2, 6, 11行目）、スキニング行列の更新処理（5, 10

#### Algorithm 1 提案アルゴリズム

---

**Input:**  $\{\tilde{\mathbf{v}}_j\}, \{\mathbf{B}_d\}, \{\tilde{\mathbf{v}}_{d,n}\}, \{\tilde{\mathbf{G}}_{d,n}\}, H$   
**Output:**  $\{\hat{\mathbf{S}}_{h,n}\}, \{w_{j,d}\}, \{\hat{w}_{j,h}\}$   
1:  $\{\hat{\mathbf{S}}_{h,n}\} = I, \forall h, n, \quad \{\hat{w}_{j,h}\} = 0, \forall j, h$   
2: 駆動骨のみに対するスキニングウェイト  $\{w_{j,d}\}$  の初期化  
3: **repeat**  
4: 新しい補助骨の追加  
5: スキニング行列  $\{\hat{\mathbf{S}}_{h,n}\}$  の更新  
6: スキニングウェイト  $\{w_{j,d}\}$  と  $\{\hat{w}_{j,h}\}$  の更新  
7: 影響の小さい補助骨の削除  
8: **until** 追加された補助骨数が  $H$  に到達  
9: **repeat**  
10: スキニング行列  $\{\hat{\mathbf{S}}_{h,n}\}$  の更新  
11: スキニングウェイト  $\{w_{j,d}\}$  と  $\{\hat{w}_{j,h}\}$  の更新  
12: **until** 誤差が収束もしくは最大反復回数に到達

---

行目)、そして補助骨の追加処理(4行目)に分けて詳細を説明する。

##### 4.1 スキニングウェイトの更新

スキニングウェイト  $\{w_{j,d}\}$  と  $\{\hat{w}_{j,h}\}$  は、式2における全てのスキニング行列  $\{\hat{\mathbf{S}}_{h,n}\}$  の値を固定したうえで最適化する。その結果、次の制約条件付き最小二乗問題が、頂点ごとに定義される。

$$\min_{\mathbf{w}_j} \left\| [\tilde{\mathbf{A}} \hat{\mathbf{A}}] \mathbf{w}_j - \mathbf{b} \right\|_2 \quad (3)$$

$$\text{subject to } \mathbf{w}_j \geq 0, \quad |\mathbf{w}_j|_1 = 1, \quad |\mathbf{w}_j|_0 \leq K, \quad \forall j$$

ここで、

$$\begin{aligned} \mathbf{w}_j &= \begin{bmatrix} w_{j,1} & \cdots & w_{j,D} & \hat{w}_{j,1} & \cdots & \hat{w}_{j,H} \end{bmatrix}^T \in \mathbb{R}^{(D+H)} \\ \tilde{\mathbf{A}} &= \begin{bmatrix} (\tilde{\mathbf{v}}_j \tilde{\mathbf{S}}_{1,1})^T & \cdots & (\tilde{\mathbf{v}}_j \tilde{\mathbf{S}}_{D,1})^T \\ \vdots & \ddots & \vdots \\ (\tilde{\mathbf{v}}_j \tilde{\mathbf{S}}_{1,N})^T & \cdots & (\tilde{\mathbf{v}}_j \tilde{\mathbf{S}}_{D,N})^T \end{bmatrix} \in \mathbb{R}^{3N \times D} \\ \hat{\mathbf{A}} &= \begin{bmatrix} (\tilde{\mathbf{v}}_j \hat{\mathbf{S}}_{1,1})^T & \cdots & (\tilde{\mathbf{v}}_j \hat{\mathbf{S}}_{H,1})^T \\ \vdots & \ddots & \vdots \\ (\tilde{\mathbf{v}}_j \hat{\mathbf{S}}_{1,N})^T & \cdots & (\tilde{\mathbf{v}}_j \hat{\mathbf{S}}_{H,N})^T \end{bmatrix} \in \mathbb{R}^{3N \times H} \\ \mathbf{b} &= \begin{bmatrix} \tilde{\mathbf{v}}_{j,1} & \cdots & \tilde{\mathbf{v}}_{j,N} \end{bmatrix}^T \in \mathbb{R}^{3N}. \end{aligned}$$

ただし、このような  $L^0$  ノルム項を含む最適化問題は NP 困難である。したがって、本研究ではアドホックアルゴリズム [14] を用いて近似解を計算する。まず、 $L^0$  ノルム項を目的関数から除外し、その結果得られる二次計画問題を解く。その解が  $L^0$  ノルム制約を満たさない場合には、ウェイトが大きい順に  $K$  個の補助骨を選択し、他のウェイトはゼロにする。そして、選択された  $K$  個の補助骨について再度二次計画問題を解くことで、スキニングウェイトの更新値を得る。

##### 4.2 スキニング行列の更新

スキニング行列の更新ステップにおいては、ある1つ

の補助骨のスキニング行列のみ更新する。つまり、残る  $H - 1$  の補助骨のスキニング行列とスキンウェイトを固定する。その結果、次式に示す weighted absolute orientation problem が、各例示データについて定義される。

$$\min_{\hat{\mathbf{R}}_{h,n}, \hat{\mathbf{T}}_{h,n}} \sum_j \left| \hat{\mathbf{v}}_{j,n} - \sum_d w_{j,d} \tilde{\mathbf{v}}_j \tilde{\mathbf{S}}_{d,n} - \sum_{i, i \neq h} \hat{w}_{j,i} \bar{\mathbf{v}}_j \hat{\mathbf{R}}_{i,n} \hat{\mathbf{T}}_{i,n} - \hat{w}_{j,h} \bar{\mathbf{v}}_j \hat{\mathbf{R}}_{h,n} \hat{\mathbf{T}}_{h,n} \right|_2 \quad (4)$$

$$\text{subject to } \hat{\mathbf{R}}_{h,n}^T \hat{\mathbf{R}}_{h,n} = \mathbf{I}, \quad \det \hat{\mathbf{R}}_{h,n} = 1, \quad \forall h, n.$$

このとき、最適な回転変換行列  $\hat{\mathbf{R}}_{h,n}$  と平行移動変換行列  $\hat{\mathbf{T}}_{h,n}$  は、先行研究 [14] で提案されている閉形式アルゴリズムによって解析的に計算される。

#### 4.2.1 補助骨の逐次的な追加

例示データからの誤差が大きい頂点を補正するように、補助骨を1つずつ逐次的に追加する。まず、頂点ごとに例示データとスキンメッシュの残差平方和を求め、その値が最大となる頂点を探索する。残差平方和の計算を次式に示す。

$$\arg \max_j \sum_n \left| \hat{\mathbf{v}}_{j,n} - \sum_d w_{j,d} \tilde{\mathbf{v}}_j \tilde{\mathbf{S}}_{d,n} - \sum_h \hat{w}_{j,h} \bar{\mathbf{v}}_j \hat{\mathbf{S}}_{h,n} \right|_2. \quad (5)$$

次に、探索された誤差最大の頂点とその一近傍頂点の例示位置に関して、Absolute orientation problem [17] を解くことで、それらの運動を最もよく近似する剛体変換を求める。そして、その剛体変換をスキニング行列とするような新しい補助骨を追加する。最後に、スキンウェイト  $\{w_{j,d}\}$  と  $\{\hat{w}_{j,h}\}$  を二次計画法により最適化したうえで、追加された全ての補助骨のスキニング行列  $\{\mathbf{S}_{h,n}\}$  を Block coordinate descent アルゴリズムを用いて更新する。なお、スキンウェイトを更新した結果、スキン変形にほとんど影響を及ぼさない補助骨は排除する。具体的には、4つ以下の頂点にだけ影響している補助骨は削除される。この一連の処理を補助骨が指定数に達するまで反復する。

## 5. 実験結果

単純な線形ブレンドスキニングでは再現不可能な皮膚形状データを作成するため、Autodesk Maya で提供されている Muscle 機能による仮想筋肉が組み込まれたキャラクターを利用した。テストモデルはモンスターキャラクターの脚部であり、スケルトンは5つの駆動骨で構成されている。そして、スケルトンの姿勢にしたがって11個の疑似筋肉が膨張、収縮することで、約700頂点で構成された表皮のメッシュを変形する。本実験では、駆動骨の回転自由度を可動範囲内で20度毎に離散化し、腿関節の振りの2自由度を6段階と5段階、腿のひねりを9段階、ひざの曲げを5段階、足首の曲げは5段階にそれぞれ変化させることで、計

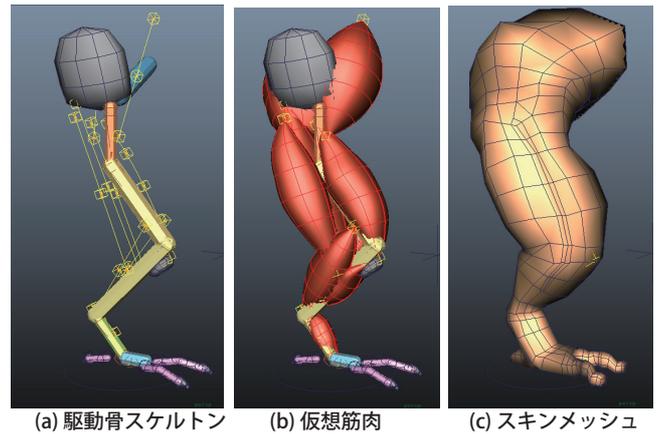


図3 例示データ生成用キャラクタモデル

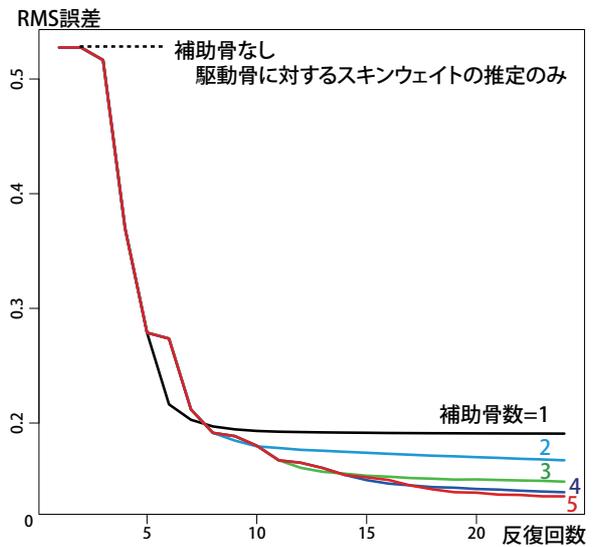


図4 補助骨の数および反復回数と近似誤差

6750 対の例示データを生成した。

補助骨の数および計算反復回数と、例示データに対する近似精度の関係を図4にまとめる。本実験では、頂点当たりの平方根平均二乗誤差 (RMS 誤差) を用いて近似精度を評価した。グラフに示すように、補助骨の追加にしたがって近似精度が向上しており、補助骨が4つの場合と5つの場合では差異が少なく、誤差がほぼ収束していることが確認できる。また、計算の反復によって誤差が単調減少しており、提案手法の計算安定性を示している。

次に、補助骨数を1つから4つまで変化させた際の、補助骨姿勢の変化を図5にまとめる。それぞれ中央の図がバインド姿勢を示しており、左右の図はそれぞれ脚を伸展した結果と屈曲した結果を示している。いずれの結果においても、補助骨 a, b, c は腿、ひざ、足首付近、補助骨 d は太もも付近に配置していることがわかる。これらは、各駆動関節付近に生じる LBS 法の不具合を低減する補助骨と、太ももの筋肉の隆起に応じた皮膚変形を与える補助骨にそれぞれ対応している。また、補助骨数の増加に伴って、a, b, c, d の優先度で補助骨が追加されていることがわかる。こ

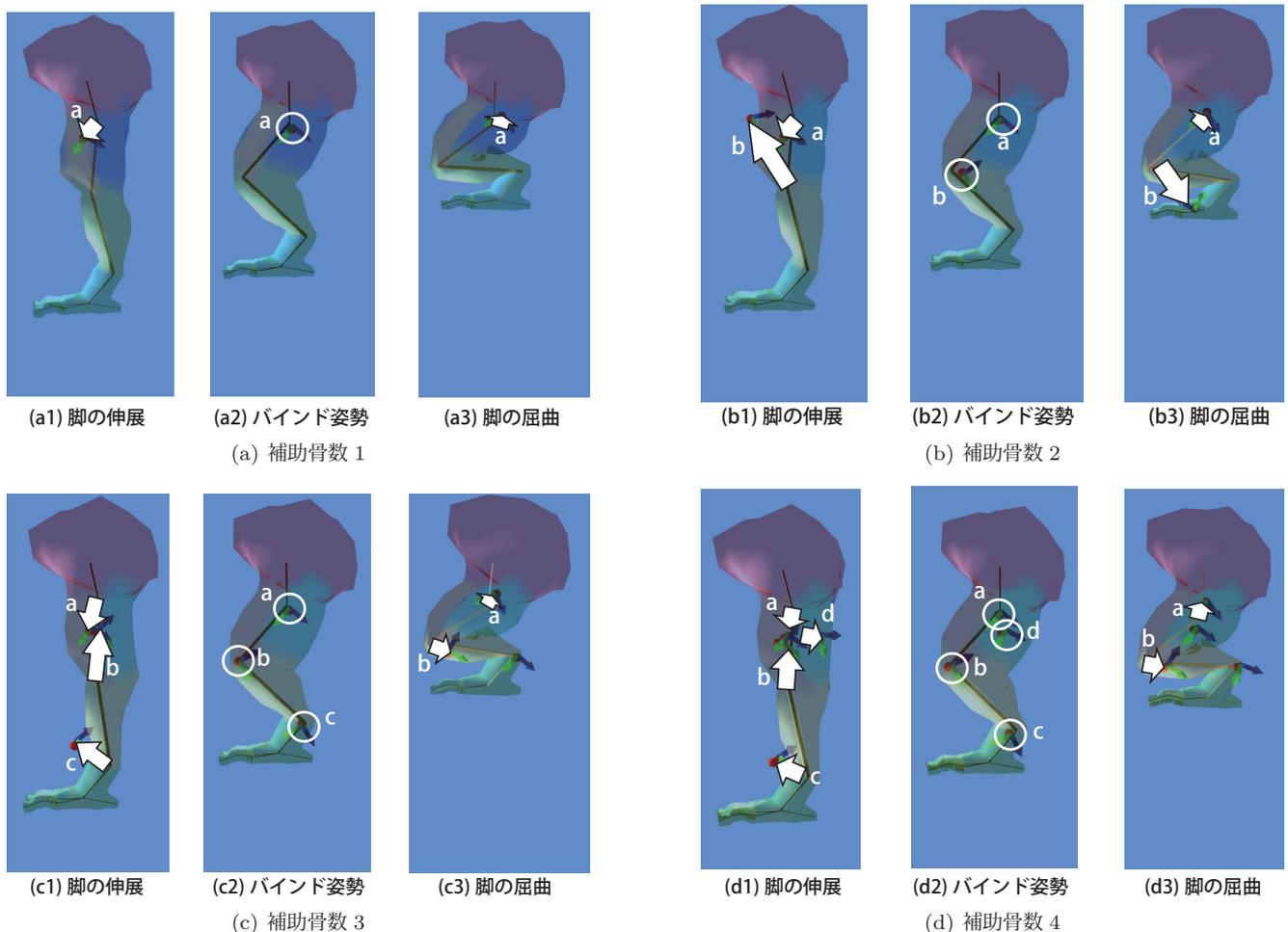


図 5 スケルトンの姿勢に応じた補助骨の移動

れは、例示データに対する近似精度向上のためには、まず腿付け根付近の歪みを低減した上で、太ももの筋肉の挙動を再現することが有効であると推定された結果を示している。また、補助骨 b と c の移動量が、新たな補助骨が追加されるにしたがって減少する様子も確認できる。これは、補助骨が不足している場合には、各補助骨が影響する頂点の数を増やした上で補助骨の変位を大きくし、かつスキんウェイトを相対的に小さく調整するような最適化が行われたためである。一方、十分な数の補助骨が追加された場合には、各補助骨の影響範囲を最小化して比較的大きなスキんウェイトを与え、かつ補助骨当たりの変位が小さくなるように最適化されたことを示している。

最適化に要した計算時間は、Core i7-4770 3.4 GHz CPU (論理コア数 8) と 16 GB RAM を搭載した計算機上で計測した。なお、現在の実装では、最適化計算の大部分を Intel Threading Building Blocks を用いて並列化している。まず、図 5(a) に示す補助骨数 1 の場合、補助骨の追加ステップ、スキんウェイトの更新、そしてスキニング行列の更新それぞれについて、1 ステップあたり 0.17 マイクロ秒、0.51 マイクロ秒、0.17 マイクロ秒を要した。本実験では最適化計算の反復回数を 20 に固定したため、全体で約 15 秒

を要した。一方、図 5(d) に示す補助骨数 4 の場合は、同計測結果は 0.17 マイクロ秒、0.82 マイクロ秒、0.72 マイクロ秒となり、全体では約 32 秒を要した。

## 6. 議論とまとめ

本論文では、例示データを用いて補助骨の姿勢とスキんウェイトを調整し、線形ブレンドスキニングの品質を改善する手法を提案した。提案システムでは、例示データおよび補助骨の数と反復計算の終了条件パラメータを指定するだけで、最適な補助骨を半自動的に推定できる。今後は、提案手法で最適化された補助骨の姿勢をもとに、駆動骨をパラメータとする補助骨制御モデルの推定に取り組む。その際、システムが推定した制御モデルはデザイナーによって修正されることを前提として、可能な限り単純な多項式やスクリプトによってモデル化することを検討している。

提案システムを用いた制作では、例示データ作成に要する作業量がボトルネックになることが予想される。つまり、手作業による補助骨セットアップのコストと比較しても、例示データの制作に要するコストのほうが大きくなる懸念がある。ただし、有限要素法などを用いた柔軟物体の変形シミュレーション [18], [19] など、リアルタイム計算

は不可能ながら、高品質な皮膚変形を計算できるシステムが開発されている。また、レンジスキャナを用いて人体表皮形状を3次元計測する手法も急速に発展している [20]。このように、今後は大量の皮膚変形データを効率的に量産できる体制の普及も見込まれ、補助骨セットアップの例示データ作成コストを大きく低減できることが期待される。

一方、皮膚変形データの量産体制の発展にともない、Pose-space deformation (PSD) 法 [21], [22] をはじめとする形状補間法の実時間アプリケーション応用も考えられる。実際、すでにいくつかのゲームタイトルでの採用例も見られ、将来的な発展も期待されている。しかしながら、PSD 法はアプリケーションランタイムに例示データを保持することを要求するため、メモリやストレージ容量が限られたモバイルデバイス等には依然として不向きである。また、PSD 法の計算量は一般的に例示データ量に比例して増加する。さらに、データ通信を伴う環境ではデータ量低減に対する要求が強い。したがって、将来的にも補助骨のような軽量の procedural アルゴリズムへの需要は大きいと考えている。

## 謝辞

本研究は、学校法人東海大学総合研究機構「研究スタートアップ支援」の援助を受けて行ったものである。

## 参考文献

- [1] Magnenat-Thalmann, N., Laperrière, R. and Thalmann, D.: Joint-Dependent Local Deformations for Hand Animation and Object Grasping, *Proceedings on Graphics interface '88*, pp. 26–33 (1988).
- [2] Mohr, A. and Gleicher, M.: Building Efficient, Accurate Character Skins from Examples, *ACM Transactions on Graphics*, Vol. 22, No. 3, pp. 562–568 (2003).
- [3] Parks, J.: Helper Joints: Advanced Deformations on RunTime Characters, *Game Developers Conference 2005* (2005).
- [4] Kim, J. and Kim, C.-H.: Implementation and Application of the Real-Time Helper-Joint System, *Game Developer Conference 2011* (2011).
- [5] Wang, X. C. and Phillips, C.: Multi-weight Enveloping: Least-Squares Approximation Techniques for Skin Animation, *Proc. of ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 129–138 (2002).
- [6] Kavan, L., Collins, S., Zara, J. and O'Sullivan, C.: Skinning with Dual Quaternions, *Proc. of ACM SIGGRAPH Symposium on Interactive 3D Graphics 2007*, pp. 39–46 (2007).
- [7] Jacobson, A. and Sorkine, O.: Stretchable and Twistable Bones for Skeletal Shape Deformation, *ACM Transactions on Graphics*, Vol. 30, No. 6, p. Article 165 (2011).
- [8] Kavan, L. and Sorkine, O.: Elasticity-Inspired Deformers for Character Articulation, *ACM Transactions on Graphics*, Vol. 31, No. 6, p. Article 196 (2012).
- [9] Baran, I. and Popović, J.: Automatic Rigging and Animation of 3D Characters, *ACM Transactions on Graphics*, Vol. 26, No. 3, p. Article 72 (2007).
- [10] Jacobson, A., Baran, I., Popović, J. and Sorkine, O.: Bounded Biharmonic Weights for Real-Time Deformation, *ACM Transactions on Graphics*, Vol. 30, No. 4, p. Article 78 (2011).
- [11] Miller, C., Arikian, O. and Fussell, D. S.: Frankenrigs: Building Character Rigs From Multiple Sources, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 17, No. 8, pp. 1060–1070 (2011).
- [12] James, D. L. and Twigg, C. D.: Skinning Mesh Animations, *ACM Transactions on Graphics*, Vol. 24, No. 3, pp. 399–407 (2005).
- [13] Kavan, L., Sloan, P.-P. and O'Sullivan, C.: Fast and Efficient Skinning of Animated Meshes, *Computer Graphics Forum*, Vol. 29, No. 2, pp. 327–336 (2010).
- [14] Le, B. H. and Deng, Z.: Smooth Skinning Decomposition with Rigid Bones, *ACM Transactions on Graphics*, Vol. 31, No. 6, p. Article 199 (2012).
- [15] Le, B. H. and Deng, Z.: Robust and Accurate Skeletal Rigging from Mesh Sequences, *ACM Transactions on Graphics*, Vol. 33, No. 4 (2014).
- [16] Park, S. I. and Hodgins, J. K.: Data-Driven Modeling of Skin and Muscle Deformation, *ACM Transactions on Graphics*, Vol. 27, No. 3, p. Article 96 (2008).
- [17] Horn, B. K. P.: Closed-form Solution of Absolute Orientation Using Unit Quaternions, *Journal of Optical Society of America A*, Vol. 4, No. 4, pp. 629–642 (1987).
- [18] Li, D., Sueda, S., Neog, D. R. and Pai, D. K.: Thin Skin Elastodynamics, *ACM Transactions on Graphics*, Vol. 32, No. 4, p. 49 (2013).
- [19] Fan, Y., Litven, J. and Pai, D. K.: Active Volumetric Musculoskeletal Systems, *ACM Transactions on Graphics*, Vol. 33, No. 4, p. 152 (2014).
- [20] Neumann, T., Varanasi, K., Hasler, N., Wacker, M., Magnor, M. and Theobalt, C.: Capture and Statistical Modeling of Arm-muscle Deformations, *Computer Graphics Forum*, Vol. 32, No. 2, pp. 285–294 (2013).
- [21] Lewis, J. P., Cordner, M. and Fong, N.: Pose Space Deformation: A Unified Approach to Shape Interpolation and Skeleton-Driven Deformation, *Proc. of SIGGRAPH 2000*, pp. 165–172 (2000).
- [22] Kurihara, T. and Miyata, N.: Modeling Deformable Human Hands from Medical Images, *Proc. of ACM SIGGRAPH/Eurographics Symposium on Computer Animation 2004*, pp. 355–363 (2004).