

OpenSim における RRA と CMC の検証 Verification of RRA and CMC in OpenSim

同志社大学大学院生命医科学研究科

家城 悠馬 伊藤 利明
Yuma Ieshiro Toshiaki Itoh

Graduate School of Life and Medical Sciences
Doshisha University

概要

PCで筋骨格モデルを表示し、様々な解析やシミュレーションが行えるフリーソフトにOpenSim[1][2]がある。本報告は、OpenSimの様々な解析ツールの中の、RRA(Residual Reduce Algorithm)とCMC(Computed Muscle Controls)に注目する。これらのツールを用いることで、神経信号を動作情報として人の動作のシミュレーションを行える。この際に、最適化問題を解く必要がある。しかし、RRAとCMCの動作保証などは未だ開発・発展段階であると思われる。ここでは、モーションキャプチャ装置を用いて取得したデータをもとに、これらツールの動作・処理結果について検証を報告する。解析例として床反力の効果を考慮し、最も単純な動作と思われる立ち上がり運動を対象にした。この動作に対し以上のツールを用い、その解析結果と筋電位計の結果を比較することで、これらの解析ツールの整合性について調べた。

1 OpenSim

OpenSimは、スタンフォード大学のデルブ教授らによって開発された様々な解析やシミュレーションが行えるフリーソフトである。これは、筋骨格構造を持ったモデルの構築と、それを用いた動力学計算が可能である。また、データを公開することにより共同で交換・テスト・解析・改善を行うことのできるシミュレーションライブラリを構築できる。そのためのコミュニティサイト[3]が非常に充実している。ソフトウェアの下層はC++で記述され、GUIはJavaで記述されている。また、プラグインにより拡張することも可能である。

2 数理モデル化

近年のコンピュータ技術・処理の発展に伴い、複雑な数理モデルを用いたシミュレーションが可能になっている。OpenSimは筋骨格モデルのシミュレーションを行うため、人間の体を様々な方法で数理モデル化している。本章では、OpenSimの筋骨格モデルの筋肉と関節部分の数理モデル化について述べる。

2.1 筋肉モデル

筋骨格モデルでは、筋肉を図1のようにモデル化する。これはHillの筋肉モデル[4][5]である。筋骨格モデルによる運動のシミュレーションを数値解法を用いて行うにあたって、筋肉モデルを収縮力を発生するアクチュエータとして考える。Hillの筋肉モデルは、以下の3つの部位から構成される。

- 収縮部 (Contractile Element : CE)
- 並列弾性部 (Parallel Elastic Element : PEE)
- 腱部 (Series Elastic Element : SEE)

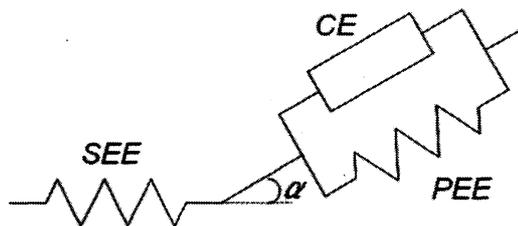


図 1: Hill の筋肉モデル

2.2 関節モデル

OpenSim は、筋骨格モデルの関節を表現するために multibody dynamics をモビライザーと共に用いる。これは、関節の表現やそれによる力の伝達、関節角度などの制約に関する分野を扱う。具体的には、セグメントと呼ばれる塊が筋骨格モデルの骨に相当し、それをモビライザーによって接続する。モビライザーには様々な種類があり、それにより人間の体の様々な複雑な関節を表現している。例として、足関節のような蝶番関節は PinJoint として定義されている。他にも、BallJoint や SliderJoint などがあり、それらを組み合わせたり制約を加えたりすることで、股関節のような臼関節や、膝関節のような蝶番関節と顆状関節の働きを併せ持った関節がモデル化されている。

2.3 筋骨格モデル

以上の筋肉モデルと関節モデルを用いて、OpenSim は図2のように筋骨格モデルを PC 上に構築する。このモデルは、以後述べる RRA と CMC の検証に用いた。

3 解析ツール

OpenSim には様々な解析ツールがある。ここでは、Inverse Kinematics (IK) Tool, Residual Reduce Algorithm (RRA) Tool, Computed Muscle Controls (CMC) Tool という解析ツールの詳細を述べる。

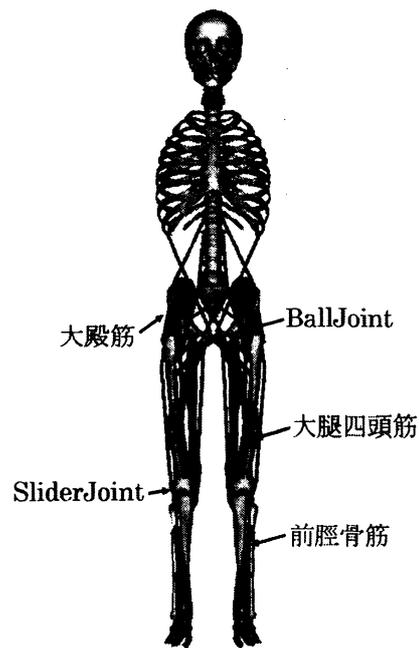


図 2: 使用する筋骨格モデル

解析の流れは、はじめにモーションキャプチャ装置で撮影したデータを IK Tool でモーションファイルに変換する。その後、この変換データに含まれる誤差を減らし、モーションファイルを改良するために RRA Tool を実行する。最後に、これらのデータを用いて CMC Tool を実行して、筋肉の刺激を求める。その流れを図 3 に示す。

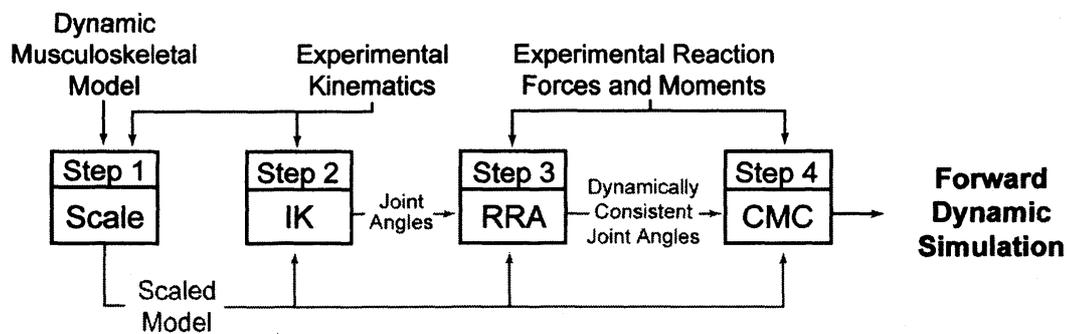


図 3: 解析の流れ ([1] より引用)

3.1 Inverse Kinematics

Inverse Kinematics(逆運動学)とは、多関節構造の末端部分の位置から全体の関節の角度を逆計算する手法のことである。OpenSim の IK Tool は、モーションキャプチャ装置で測定したマーカー

データを入力データとして、その測定した動作を再現するような関節角度を出力する。OpenSim では、逆運動学問題は (1) 式の最小 2 乗問題として扱う [1][8]。

$$\text{誤差} = \sum_{i=1}^{\text{markers}} \omega_i (x_i^{\text{subjects}} - x_i^{\text{model}})^2 + \sum_{j=1}^{\text{segments}} \omega_j (\theta_j^{\text{subjects}} - \theta_j^{\text{model}})^2 \quad (1)$$

ここで、 x_i^{subjects} と x_i^{model} は、被験者と筋骨格モデルの i 番目の筋肉を示し、 $\theta_j^{\text{subjects}}$ と θ_j^{model} は、被験者と筋骨格モデルの i 番目の関節の関節角度を示す。また、 ω_i と ω_j はそれぞれマーカーとセグメントの重みを示す。

3.2 Residual Reduce Algorithm

筋骨格モデルに対する IK の近似解は、測定された床反力やモーメントと運動学的に一致しないことが多い。これは、測定中の誤差による残差力が発生するためである。この不一致性を解決するために RRA Tool を用いる。OpenSim の RRA Tool では (2) 式により、残差力 F^{residual} を求める [1][9]。

$$F^{\text{external}} = \sum_{i=1}^{\text{segments}} m_i a_i - F^{\text{residual}} \quad (2)$$

ここで、 F^{external} は床反力から体重を引いたもの、 a_i は IK によって導き出されたモデルの i 番目のセグメントの質量中心における並進方向加速度、 m_i はこのセグメントの質量である。残差力とモーメントを減らすために、これらは動作の間中計算される。その後、平均された値に基づいて、残差の平均値を減らすようにモデルの調整を行う。

3.3 Computed Muscle Controls

CMC は、RRA が生成した動作を実現するための筋肉の刺激 $x(t)$ を計算する [1][9]。時間 t における筋肉の刺激は、 $0 < x(t) < 1$ で表される。 $x(t) = 0$ の時は筋肉が全く働いていないことを示す。一方、 $x(t) = 0.5$ の時は筋電図では 50% MVC を示し、筋生理学的には、全運動単位の 50% の筋肉が働いていることを示す。%MVC とは、最大の力を発揮しているときの筋電位の結果 EMG と、測定値 MVC を用いると以下のように表現される。

$$\%MVC = \frac{MVC}{EMG} \times 100$$

CMC ははじめに、RRA が生成した動作を実現するための加速度 \ddot{q}^{des} を求める。このとき、IK によって導き出されたデータ (q, \dot{q}) と実験データ $(q^{\text{exp}}, \dot{q}^{\text{exp}})$ を用いる。時間 $t+T$ における \ddot{q}^{des} を計算するために、シミュレーション時間 $t+T$ における $\ddot{q}^{\text{exp}} (= a_i)$ と、時間 t における位置と速度を用いて (3) 式を解く。また、(3) 式の変数関係を図 4 に示す [9][10]。

$$\ddot{q}^{\text{des}}(t+T) = \ddot{q}^{\text{exp}}(t+T) + k_v[\dot{q}^{\text{exp}}(t) - \dot{q}(t)] + k_p[q^{\text{exp}}(t) - q(t)] \quad (3)$$

ここで、 k_v と k_p はそれぞれ速度と位置のフィードバック係数である。次に、(3) 式で求めた加速度 \ddot{q}^{des} を発生する筋力 $f(t+T)$ を求め、その筋力を生成するような筋肉の刺激を計算する [9]。ここで、 $x(t)$ を用いて、筋力の範囲を $f(x_{\min}(t)) < f(t+T) < f(x_{\max}(t))$ と定める。この範囲内

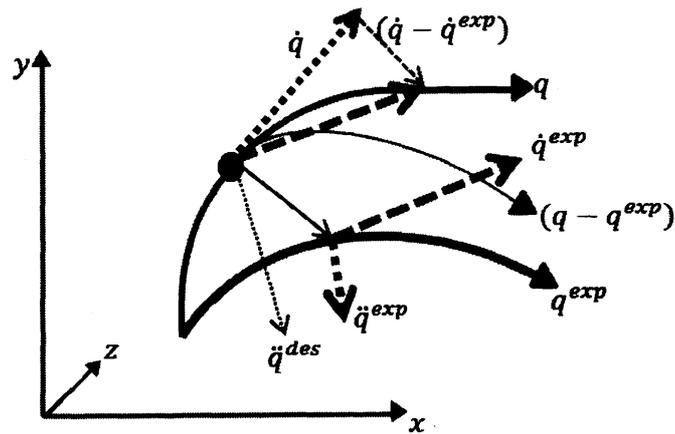


図 4: (3) 式の変数関係

で理想的な筋力 f^{des} を加速度 \ddot{q}^{des} により特定する。そのために、時間 $t+T$ で $f^{des}(t+T)$ を満たすような $x(t)$ を、最適化問題を用いて求める [11]。

次に、 T について説明する。 T は CMC を実行する際にユーザーが任意で決めることができるパラメータである。この T 毎に (3) 式を解いて \ddot{q}^{des} を求める。同様に、次に説明する最適化問題に関しても T 毎に計算を行う。今回の検証では、 $t = 2.0[s]$ 、 $T = 0.01, 0.02, 0.03, 0.04, 0.05[s]$ として計算を行った。

4 最適化問題

OpenSim は、数値解を求める際に最適化問題を用いる。使用する解法は主に”IPOPT”と”CFSQP”である。IPOPT はライセンスフリーのため、デフォルトは IPOPT である。CFSQP は使用の際に作成者の許可が必要である。

4.1 RRA と CMC の最適化問題

以上より RRA と CMC で最適化問題を解く必要がある。OpenSim では以下のように最適化問題を使いわける。

$$\min J_{slow} = \sum_{i=1}^{muscles} (f_i^*(x_i(t+T)))^2 + \sum_{j=1}^{segments} w_j (\ddot{q}_j^* - \ddot{q}_j)^2 \quad (4)$$

$$\begin{aligned} \min J_{fast} &= \sum_{i=1}^{muscles} (f_i^*(x_i(t+T)))^2 \\ \text{s.t.} \quad \ddot{q}_j^* &= \ddot{q}_j \\ f_i(x_{min}(t)) &< f_i^*(t+T) < f_i(x_{max}(t)) \end{aligned} \quad (5)$$

(4)と(5)はそれぞれ Slow target と Fast target と呼ばれている [12]. Slow target の目的関数 J_{slow} は, 筋力の合計と計算結果とモデルの間の加速度の差の合計を最小化するものである. 一方, Fast target の目的関数 J_{fast} は, 筋力の合計のみである. しかし, Fast target には制約がある. 一つは計算結果とモデルの加速度が一致すること, もう一つは $x(t)$ を用いて表された筋力の範囲内に, 求める筋力が存在することである.

このように目的関数や制約が異なるのは, 各々の役割が異なるからである. Slow target は主に RRA で使用される. RRA はモデルやデータの誤差を減らす役割があることは先程述べたが, 誤差を減らしたデータの出力も行っている. その際に, Slow target を解いてパラメータを算出する. 一方 Fast target は CMC で使用される. (4)と(5)を見ればわかるように, Fast target には制約があり複雑な最適化問題になっている. 制約があるため, その分出力結果の精度も良い. このように, RRA は CMC で使うデータの作成, CMC はそのデータを用いて筋肉の刺激を求める, という役割の違いから最適化問題を使い分ける.

5 シミュレーション

実際に測定したデータを用いて RRA と CMC の検証を行う. 本章では, 今回の検証におけるシミュレーションの内容, 使用した筋骨格モデルの詳細, 測定・計算環境について説明する.

5.1 シミュレーション内容

今回の検証では, 中腰の姿勢を保持した後に立ち上がる動作を測定対象とした. OpenSim による筋骨格モデルでの様子を図5に示す. このような動作に注目した理由として, OpenSim が発展途中であることが挙げられる. そのため, 簡単な日常動作を測定対象とした. また, 床反力の時間変化がどのように影響するのかを確認するために, 床反力の影響を受けやすい立ち上がる動作を選んだ. そのデータを OpenSim を用いて解析し, その結果と筋電位計の結果を比較することで OpenSim の整合性を検証する. また, CMC は床反力データを使用して解析を行っている. その影響を確かめるために, マーカーデータは同じだが床反力データが異なると OpenSim の解析の結果にどのように影響が表れるかを検証した.

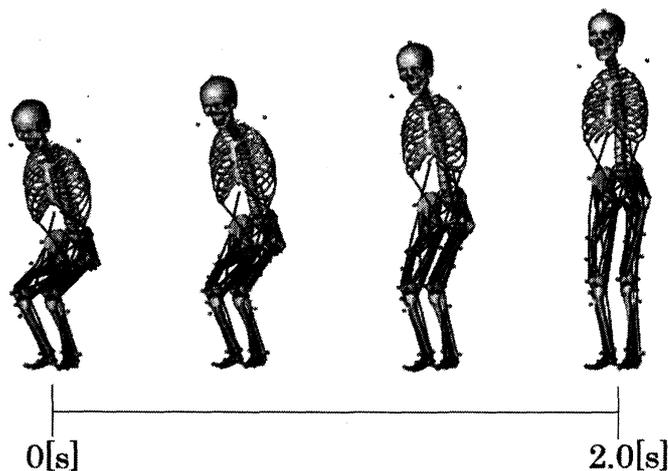


図 5: 測定の対象とした動作

動作測定のために、モーションキャプチャ装置を使用した。また、マーカーデータを撮影するカメラには株式会社ナックイメージテクノロジーの”Raptor-4”，床反力を測定する床反力計は Bertec Corporation の”FP4060”を使用した。測定したデータの編集には Motion Analysis Corporation の”EVaRT 5.0.4”を使用し、測定したデータを OpenSim に取り込めるようにデータの編集を行った。その後、編集を行ったデータを用いて解析ツールを実行する。表 1 に、今回使用したパソコンのスペックを示す。

表 1: 使用した PC のスペック

Windows Edition	Windows 7 Enterprise
プロセッサ	Intel(R) Core(TM) i7 CPU 3.33GHz
メモリ (RAM)	16GB
システム	64bit

5.2 データ形式

測定から、trc ファイルと force ファイルが得られる。trc ファイルには、各マーカーの座標データが含まれる。マーカーの位置によっては認識されていない時間があったり、違うマーカーと誤認識している場合がある。その際は EVaRT を用いて修正を行う。このデータは筋骨格モデルの大きさの調節や IK ツールで使用される。force ファイルには、床反力の大きさ [N]、起始位置、トルクが含まれている。

5.3 筋骨格モデル

今回使用した筋骨格モデルは簡単のため、腕のないモデルを使用した。そのため、中腰を保つときや立ち上がるときに腕の力を使わないように測定を行った。筋骨格モデルの詳細は以下の表 2 に示す。

表 2: 筋骨格モデルの詳細

身長	176.4cm
体重	64.7kg
マーカー数	39 個
セグメント数	12 個
筋肉数	92 個

6 結果

$T = 0.05$ と定めて CMC ツールを実行した結果、計算時間は 511 秒だった。実際の運動が 1.8 秒なので、0.1 秒の動作の解析に約 30 秒かかったことになる。また、データのサイズは 6,631kb だった。

6.1 T の影響

3.3 で説明した T による影響を図 6 に示す。図 6 は、大腿方形筋の刺激の時間変化である。明瞭のため、 $T = 0.01, 0.03, 0.05$ の結果のみを示している。細線が $T = 0.01$ 、点線が $T = 0.03$ 、太線が

$T = 0.05$ の結果である。図から、 T の値が解析結果に大きく影響を与えることがわかる。 $T = 0.01$ の場合、最大値である $x = 1$ をとった T 秒後には最小値である $x = 0$ になる箇所が多数あり、結果の考察に使用するのは困難である。 $T = 0.03$ の場合、どの時間で筋肉の刺激がピーク値を取るのかがある程度わかる結果になる。また、 $T = 0.05$ では $T = 0.03$ の結果と同様にピーク値の箇所が分かりやすい結果になる。これらの結果より、今回の検証では主に $T = 0.05$ を使用する。

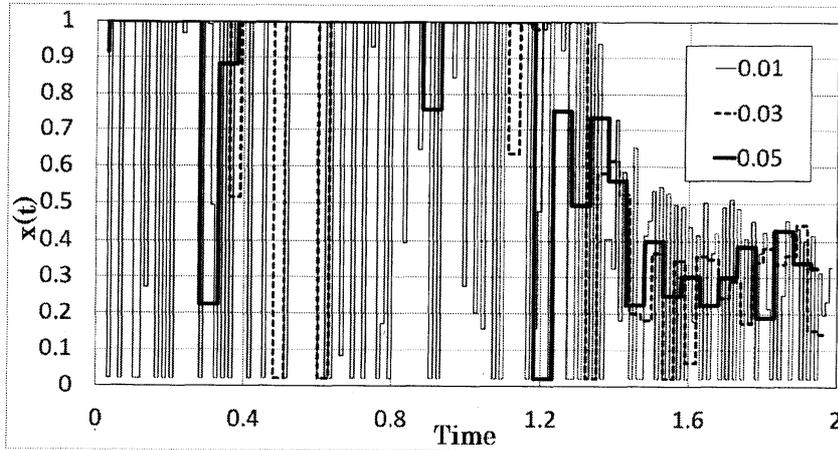


図 6: $T=0.01, 0.03, 0.05$ の結果の違い

6.2 結果の平滑化

T の数値を定めた後、出力結果の平滑化を行った。 $T = 0.05$ の出力結果は図 7 の太線のようにになる。図からわかるように、平滑ではない時間変化になっている。このような結果が出力される原因として、計算時間と書き出し時間の違いが考えられる。CMC ツールは、 T 毎に計算を行っているのだが、0.001 秒毎にデータの書き出しをしている。そのため、 T 毎にグラフの値が変化している。これを解決するために、計算時間と書き出し時間が一致していない部分のデータ（グラフの平らな部分）を削除した。その様子が図 7 の細線である。また、更に平滑化を進めるために移動平均法を用いた。その様子が図 7 の点線である。図 7 の点線のように修正したグラフと筋電位計の結果を比較することで考察を行う。

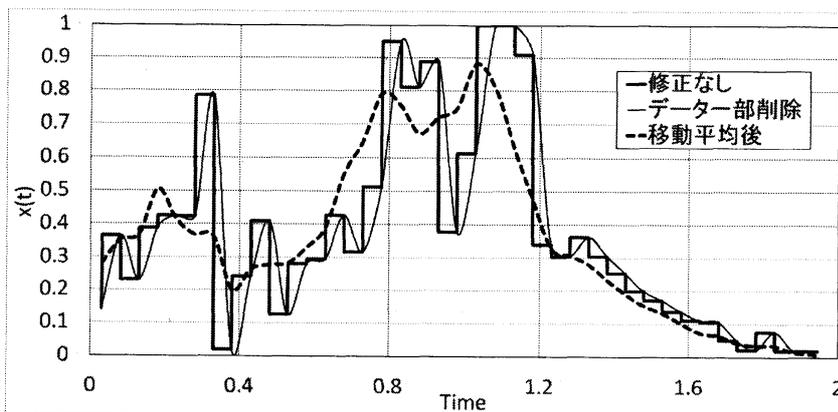


図 7: $T=0.05$ の出力結果の平滑化過程

6.3 CMCの結果と筋電位計の結果の比較

図8～図10に筋電位とCMCの結果を比較した図を示す。図8は前脛骨筋の刺激の時間推移を表した図とその筋肉の様子である。星印が示す筋肉が前脛骨筋である。前脛骨筋とは、脛骨の外側から内側の中足骨・楔状骨に付着している筋肉である[13]。主な働きは、足関節の背屈・内反である。今回の検証の動作においては、中腰の姿勢を保っている間に力を発揮していると考えられる。図8において、中腰の姿勢を保っている $Time = 0 \sim 0.8$ まで、 $x(t)$ と%MVCが共に高い値を示している。その後は値が低くなっていく。その理由として、立ち上がると共に足関節が底屈していくためだと考えられる。

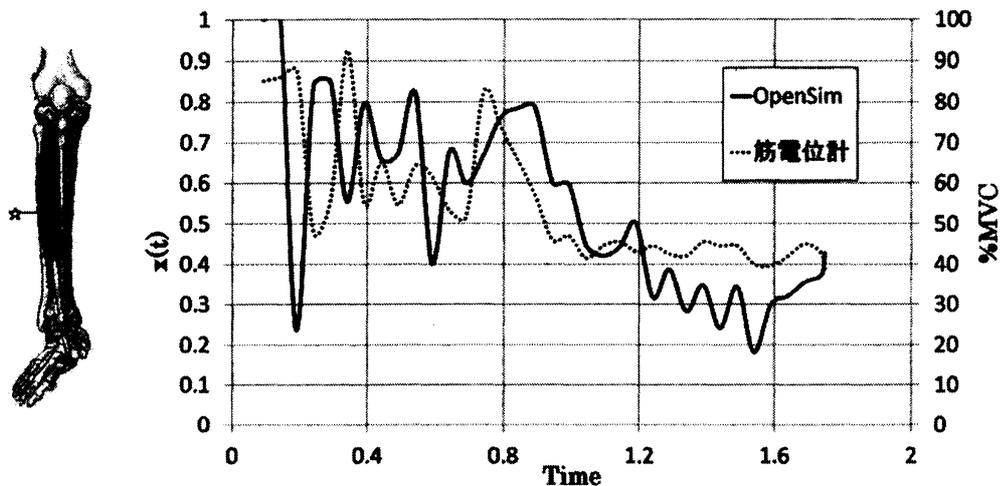


図8: 前脛骨筋の刺激の時間変化(左図は[14]より引用)

次に、図9は大殿筋の刺激の時間推移を表した図とその筋肉の様子である。大殿筋とは、後殿筋線の後方、仙骨・尾骨の外側縁、胸腰筋膜、仙結節靭帯らに付着している筋肉である[13]。主な働きは、股関節の伸展・外旋である。今回の検証の動作においては、中腰から立ち上がる際に力を発揮していると考えられる。図9において、中腰から立ち上がる $Time = 0.8 \sim 1.1$ 付近で高い値を示している。

次に、図10は大腿直筋の刺激の時間推移を表した図とその筋肉の様子である。大腿直筋とは、下前腸骨棘・大腿骨に付着している筋肉である[13]。主な働きは、膝関節の伸展である。今回の検証の動作においては、中腰の姿勢を保っている間に力を発揮していると考えられる。中腰の姿勢に関して膝関節に注目して考えると、膝関節を伸展させる大腿直筋と膝関節を屈曲させる半膜様筋や腓腹筋が互いに作用し合ってその姿勢を維持している。図10を見てみると、中腰の姿勢を保っている $Time = 0 \sim 0.8$ で高い値を示している。

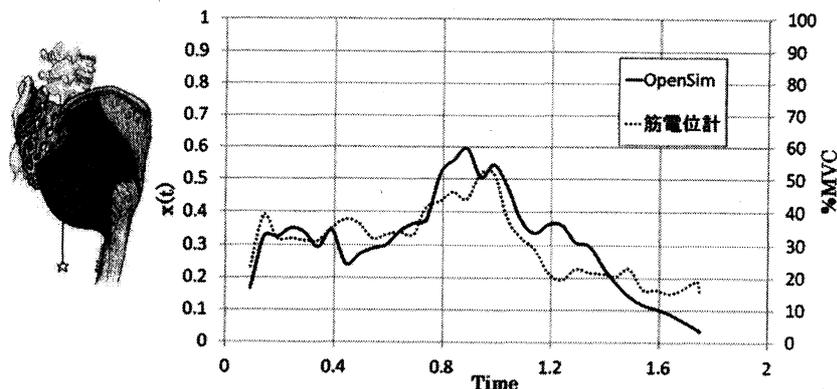


図 9: 大殿筋の刺激の時間変化 (左図は [14] より引用)

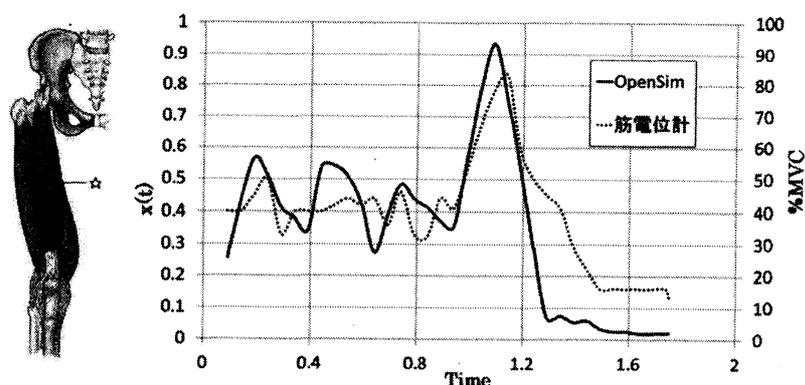


図 10: 大腿直筋の刺激の時間変化 (左図は [14] より引用)

6.4 床反力計がCMCに与える影響

床反力がどのようにCMCの結果に影響を与えるのかを調べるために、被験者に5kgと10kgと20kgの負荷をかけ、負荷のない場合と同様の動作の測定を行った。20kgの解析に関しては、RRAまでは解析を行うことができた。しかしCMCの解析中にエラーが発生し、解析を行うことができなかった。後述の考察でも述べるが、このようなアルゴリズムの改良もOpenSimのCMCの今後の課題であると考えられる。そのため、今回は5kgと10kgの負荷をかけた状態での解析を行う。

図11に示すように、重さによって動作が変わらないように測定を行った。図11は、右上前腸骨棘に付けたマーカータの時間変化を示している。太線は負荷なしの場合、細線は負荷が5kgの場合、点線が負荷が10kgの場合を表している。このように、マーカータはさほど変わらないが、床反力は異なるデータを用いて解析を行った。

その結果を、図12と図13に示す。図12と図13はそれぞれ、負荷が5kg、10kgの時の前脛骨筋におけるCMCの結果と筋電位の結果を比較した図である。実線がCMCの結果を示しており、点線が筋電位計の結果を示している。負荷が0kgの際は、図8のようになる。それぞれの図を見ても、CMCの結果と筋電位計の結果がよく似ていることがわかる。また、負荷が大きくなるにつれて筋肉の刺激も高い値を示すようになったことがわかる。前脛骨筋は、中腰の姿勢を保持する際に働く。負荷が多くなれば、その姿勢を保持することが困難になることは容易に想像できる。そのため、図における早い時間帯での筋肉の刺激が高い値を示したと考えられる。これらの結果から、OpenSimはマーカータが同じでも、床反力データが異なればその分筋肉の刺激が高い値を示すことがわかった。

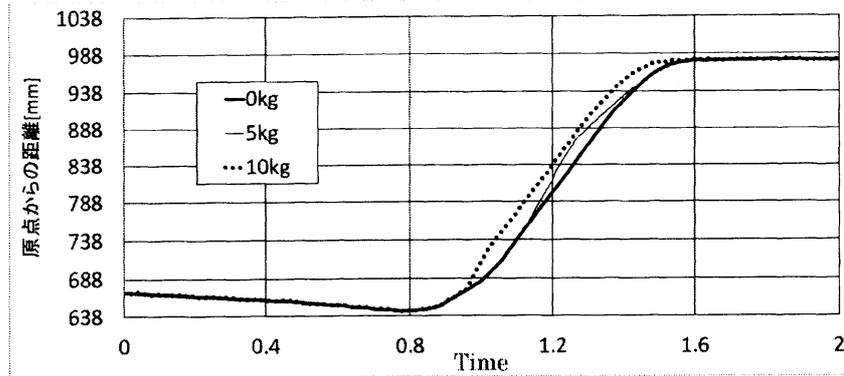


図 11: 右上前腸骨棘の時間変化

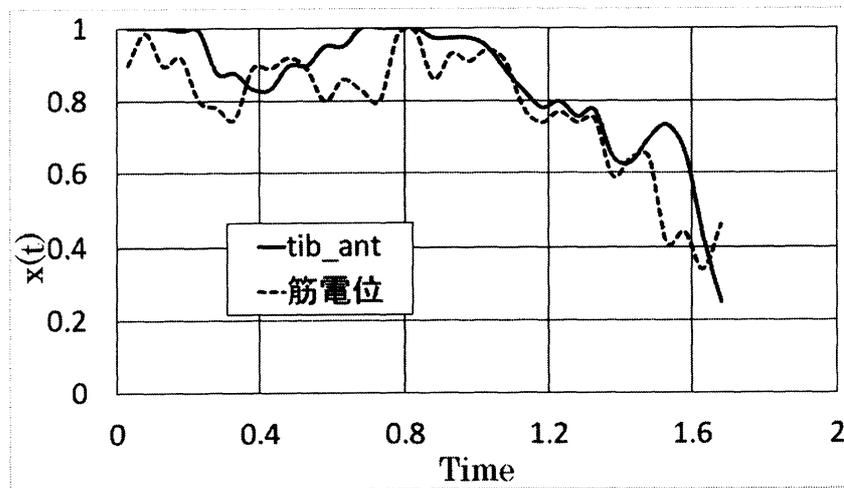


図 12: 前脛骨筋の刺激の時間変化 (負荷 5kg)

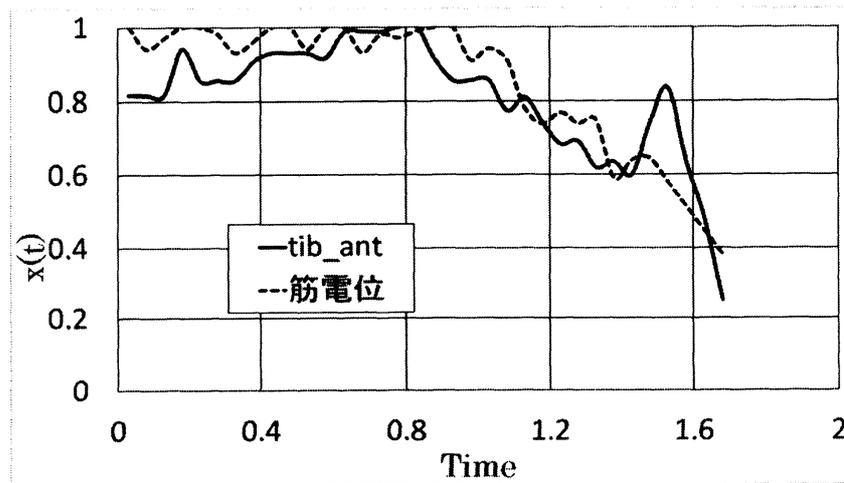


図 13: 前脛骨筋の刺激の時間変化 (負荷 10kg)

6.5 大腿四頭筋が複合筋であることを考慮した場合の比較検証

図10において、大腿前面に貼った筋電位の結果とCMCにおける大腿直筋の刺激の時間変化を比較した。しかし、図10からわかるように、大腿直筋付近にはいくつかの大きな筋肉が存在する。それらの筋肉(中間広筋・内側広筋・外側広筋)と大腿直筋を合わせた複合筋は、大腿四頭筋と呼ばれている。今回の検証では、表面筋電位計を使用したため、筋電位計の結果が大腿直筋のものかどうかはわからない。そのため、図10で使用した筋電位の結果と複合筋を用いて比較検証を行う。なお、筋電位計は大腿前面かつ外側に貼ったため、大腿直筋と外側広筋の複合筋を比較対象とした。

比較方法として、以下のような関係式を用いて算出した複合筋の刺激 x_{com} と筋電位の結果の最小二乗誤差が小さくなるよう反復試行を行った。また、中腰の姿勢の保持、そこから立ち上がる動作、直立姿勢の保持の三つに分けて最小二乗誤差を求めた。

$$x_{com} = x_{rect} \times A + x_{vas} \times B$$

このとき、 $A + B = 1$ となるように A と B の値を変えながら試行を行った。また、 x_{rect} は大腿直筋の刺激、 x_{vas} は外側広筋の刺激を表している。このようにして求めた複合筋の時間変化と筋電位との比較を図14に示す。図14の黒い太線、点線、灰色の太線はそれぞれ中腰の姿勢の保持、そこから立ち上がる動作、直立姿勢の保持における複合筋の刺激を表している。また、黒い細線は筋電位の結果である。このときの A と B の値を表3に示す。

図14を見てみると、図10よりも筋電位の結果と近い時間変化を示しているのがわかる。この検証により、CMCの結果を見る際には複合筋として考察を行った方が良い可能性が示された。これに関しては、その他の部位での検証を通して更に調べていく必要があると考えられる。

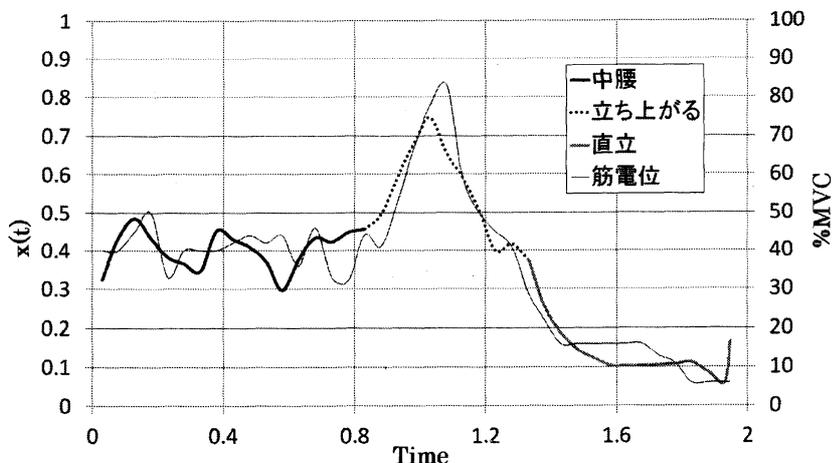


図14: 複合筋と筋電位計の結果の比較

表3: A と B の値

	A	B
中腰	0.38	0.62
立ち上がる	0.36	0.64
直立	0.72	0.28

7 考察

対象の動作における各筋肉の働きと解析結果を照らし合わせることで、図 8～図 10 のように OpenSim の CMC の結果と筋電位計の結果がおおよそ等しいことが示された。つまり、マーカデータと床反力データさえあれば筋電位計を使用せずに、CMC を用いることで筋電位の時間変化を調べることができることがわかった。加えて、マーカデータは極力変えずに床反力データのみが異なるデータの解析を行った結果、図 12 と図 13 に示すように CMC の結果に影響を与えることがわかった。

筋電位のデータを入手するには専用の機器が必要である。そのような機器がない場合でも筋電位の時間変化がわかることは、OpenSim を用いた動作解析が非常に効果的な手段であると考えられる。しかし、今回の検証では、簡単な日常動作のみ扱ったため、複雑な動作でも OpenSim の CMC が使用できなければマーカデータから筋電位の時間変化を推測する効果的な方法であるとは言い切れない。今後は、様々な動作に関しても CMC を実行し、さらに CMC の検証を行う必要がある。

また、図 6 で T による影響について示した。今回の検証では $T = 0.05$ を用いて解析を行ったが、これが最適な値かどうかは解析できていない。 $T = 0.01$ のような結果が出力される原因を調べ、 T の最適値を明確にする必要がある。それに加え、6.5 で示した複合筋を用いた解析についても更に検証を進める必要がある。

謝辞

本研究に関して、同志社大学の中村康雄教授と筑波大学の蔡東生教授から多大な協力を頂きました。

参考文献

- [1] Delp, S.L., et al., "OpenSim: open-source software to create and analyze dynamic simulations of movement", *IEEE Transactions on Biomedical Engineering*, 54(11) : p.1940-1950, 2007.
- [2] Miguel Christophy, "A Detailed Open-Source Musculoskeletal Model of the Human Lumbar Spine", Master's thesis, University of California at Berkeley, 2010.
- [3] OpenSim Community, <http://www.stanford.edu/group/opensim/support/index.html>
- [4] A. V. Hill, "The heat of shortening and the dynamic constants of muscle. Proceedings of the Royal Society of London" *Series B, Biological Sciences*, 126(843):136-195, 1938.
- [5] F. E. Zajac, "Muscle and tendon: Properties, models, scaling, and application to biomechanics and motor control", *Critical Reviews in Biomedical Engineering*, 17(4) : p.359-411, 1989.
- [6] 日本機械学会, "生体力学", p.31-51, 1997.
- [7] F. E. Zajac, "Muscle and tendon: Properties, models, scaling, and application to biomechanics and motor control" in J. Bourne editors, CRC Critical Rev. in Biomed., Vol.17, Issue 4, pp.359-411, 1989.
- [8] T. W. Lu and J. J. O'Connor, "Bone position estimation from skin marker coordinates using global optimization with joint constraints", *Journal of Biomechanics*, vol.32, p.129-134, 1999.
- [9] Ajay Seth and Tim Dorn , "Generating Simulations with OpenSim", OpenSim Advanced User and Developer Workshop March 2012, 2012.
- [10] D. G. Thelen and F. C. Anderson, "Using computed muscle control to generate forward dynamic simulations of human walking from experimental data", *Journal of Biomechanics*, vol.39, pp. 1107-1115, 2006.
- [11] Happee, R., 1994. "Inverse dynamic optimization including muscular dynamics, a new simulation method applied to goal directed movements", *Journal of Biomechanics*, vol.27, 953-960, 1994.
- [12] OpenSim User's Guide, <http://simtk-confluence.stanford.edu:8080/display/OpenSim/User%27s+Guide>
- [13] Richard L. Drake , et al., "グレイ解剖学原著第1版" , エルゼビア・ジャパン, 2011.
- [14] 筋肉.guide, <http://www.musculature.biz/>