

# 接触による力学的反応を自動生成するアバター

杉森 健<sup>1,a)</sup> 三武 裕玄<sup>1,b)</sup> 佐藤 裕仁<sup>1,c)</sup> 小栗 賢章<sup>1,d)</sup> 長谷川 晶一<sup>1,e)</sup>

受付日 2020年1月30日, 採録日 2020年9月10日

**概要:** VR メタバースや VTuber などのように, モーションキャプチャデバイスを用いてリアルタイムにアバターを操作する機会が増えるにつれて, アバター同士接触することでコミュニケーションをとったり, アバターと物体が衝突する機会が増えている. しかし, 現状ではアバター同士やアバターと物体が接触しても貫通してしまうか, 不自然な動作になってしまう. そこで本研究では物理シミュレーションを用いて自然な接触動作を自動生成する. その際に問題となる追従遅れをフィードフォワード制御により解決する. 提案手法により, これまでできなかったアバター同士やアバターと物体の自然な接触が可能であることを示す. さらに, 提案手法ではこれまで一般的であった接触している振りの演技が不要になることで, 演者の負担が軽減されることを示す.

**キーワード:** アバター, キャラクタ, モーションキャプチャ, 物理シミュレーション

## Virtual Avatar Automatically Generates Physical Response to Contact

KEN SUGIMORI<sup>1,a)</sup> HIRONORI MITAKE<sup>1,b)</sup> HIROHITO SATO<sup>1,c)</sup> KENSHO OGURI<sup>1,d)</sup>  
SHOICHI HASEGAWA<sup>1,e)</sup>

Received: January 30, 2020, Accepted: September 10, 2020

**Abstract:** As opportunities to operate avatars in real time using a motion capture device such as VR metaverse and VTuber increases, the opportunities of communication by contacting avatars and the opportunities of collisions between avatars and objects increase. However, at present, even if an avatar and an avatar or an avatar and objects come into contact with each other, they will penetrate or behave unnaturally. Therefore, in this research, natural contact motion is automatically generated using physical simulation. The tracking delay which is a problem at that time is solved by feedforward control. We show that the proposed method allows for natural contact between an avatar and an avatar or an avatar and objects, which was previously impossible. Furthermore, we show that the proposed method reduces the burden on the performers by eliminating the need to perform touching pretending, which has been common in the past.

**Keywords:** avatar, character, motion capture, physical simulation

### 1. 背景と目的

CG キャラクタはゲーム, 映画, アニメなど多くのエンタテインメントコンテンツに利用されている. CG キャラクタにはキャラクター性を表現する魅力的な動作が求めら

れ, 手作りのアニメーションだけでなくモーションキャプチャが用いられてきた.

画像認識の発展や高性能なカメラ・センサの価格低下にともない, モーションキャプチャ技術は家庭でも利用可能となった. それにともなってモーションキャプチャを操作入力として用いるインタラクティブコンテンツも多く登場し, 特に, プレイヤの分身となる CG キャラクタ, すなわちアバターのリアルタイムでの身体動作にも用いられるようになっていく. 近年急速に普及しつつある VR 用ヘッドマウントディスプレイ (HMD) もその多くが HMD と両手用コントローラの位置姿勢検出機能を持ち, VR 世界内

<sup>1</sup> 東京工業大学  
Tokyo Institute of Technology, Yokohama, Kanagawa 226-8503, Japan

a) sugimori.k.ac@m.titech.ac.jp

b) mitake@pi.titech.ac.jp

c) satoh.h.aa@m.titech.ac.jp

d) oguri.k.aa@m.titech.ac.jp

e) hase@pi.titech.ac.jp

でのアバターのリアルタイム操作を可能としている。

こうした背景のもとで、リアルタイムモーション入力アバターの振舞いを他人に見せることをともなうエンタテインメントが数多く誕生した。主な例としてキャラクターの姿で演技し動画配信などを行うバーチャル YouTuber (VTuber) やバーチャルアイドル、VR 空間でアバター同士のコミュニケーションを楽しむ VRChat などの VR メタバースがある。複数人でのオンライン対戦が可能な VR ゲームなどもその一例といえる。

アバターの動作する範囲に物体や他人のアバターが存在すれば、接触が発生する。複数人参加型の VR ゲームでは参加者同士が武器を打ち合えるものがあるし、VR メタバースでは撫でる・ハグするといった接触コミュニケーションが現実の人間同士よりも気楽に行われる場合もあることが指摘されている。VTuber においては視聴者が番組に干渉する手段としてバーチャルアイテムを購入しキャラクターに投げつける投げ銭システムが普及して大きな収益源となっているほか、複数の VTuber が同じバーチャルスタジオで共演する際に身体的接触をともなう場面も見られる。

しかし、これらの事例の多くで、接触対象となる物や人はアバターを操作するプレイヤーのいる実空間には存在しないため、プレイヤーが何も考えずにアバターを操作すると貫通したり、物体が衝突したのに無反応であるといった違和感のある映像が生み出されるという問題がある。

接触・衝突の影響が十分考慮されない環境下で違和感のない映像を得るために、アバターの操作者が接触・衝突が起きた「振り」を演技することが行われている。しかし、これには高度な技能が必要であり、多くの演者に期待できるものではない。また、演者は本来の動作をする以外に接触とその影響を意識しながら演技する必要が生じるため負担が大きい。さらに、たとえ熟練した演者であっても予測不可能な動きをする相手との接触では貫通を回避することは不可能に近い。このことから接触・衝突の作用は演技に頼るのではなくアバターや物体の動作に対する計算により実現することが望ましい。

物理シミュレーションを用い、アバターや物体を形状付き剛体モデルで表現し計算すれば接触・衝突の作用を考慮した力学的な動作を自動で行わせることができる。

実際に物理シミュレーションが用いられる例として、VTuber と物体との接触がある。VTuber (アバター) と物体が接触した際はバーチャル物体のみを一方的に跳ね返すようになっている場合が多い。アバター側の動作制御が不要で演者の演技どおりに動く利点もある一方で、重い物体が衝突しても一方的に跳ね返す様子は不自然であり、またアバター同士の接触では貫通を防止できないという問題がある。

これらの問題を解決する手法として操作者の動作入力を目標角度とする PD 制御 (比例・微分制御) が考えられる

が、重い物体の衝突に対し柔軟になるよう PD 係数を小さくすると演者が素早い演技をした際に遅れて追従したり行き過ぎたりする副作用が発生する。そのような状態で狙った演技を行うには演者に新たな技能を要求するうえに、腕を急停止させる動作などは依然不可能となるため演技表現の幅が狭まってしまう。

以上の背景から、本論文では演者の技能によらずに違和感なく物体や他のアバターへの接触が表現できるアバターの制御手法を提案する。ここまでに述べた課題を解消するため、本研究ではアバターの制御手法が満たすべき要求として以下の3点をあげる。

『自然な接触映像』 接触の影響が考慮された動作を自動で行うことで、接触時および接触後に違和感のない映像を生み出すこと

『演者の負担軽減』 接触した振りの演技に関する演者の負担を軽減すること

『非接触時の追従性』 接触時以外の演技に影響を与えないよう、接触時以外は演者の演技に素早く追従すること

## 2. 関連研究

キャラクターが接触・衝突したときの反応動作を自動生成できる研究や製品を紹介する。

### 2.1 データベースを用いた反応動作生成

外力が加わった際の反応生成では、事前に用意した反応アニメーションのデータベースから外力の向き・強さに応じたアニメーションを適用する手法が提案されている [1], [2]。しかし、データベースを用意する手間がある。また、アバター同士でのインタラクションでは押される側のみでなく押す側の力学的反応も生成する必要がある。

### 2.2 物理シミュレーションを用いた反応動作生成

#### アニメーションと物理シミュレーション

キャラクターのアニメーションと物理シミュレーションを使用することで接触時のキャラクターの反応動作を自動生成する研究は複数存在する。Shapiro らの研究 [3] や Zordan らの研究 [4] では接触がない場合はアニメーションを元にした運動学コントローラを使用し、接触がある場合は物理シミュレーションによる力学コントローラを使用するといった手法により反応動作を作成している。他の手法として、予期せぬ外力が加わった際に、外力による影響を大きく受ける関節のみを動力学制御する手法 [5]、運動学制御と動力学制御の切替えではなく、常時動力学制御を適用して、接触時に関節の PD 係数を弱めることで追従性と反応動作を両立させた研究 [6], [7]、モーションデータと外力から関節の角加速度を直接計算する手法 [8] なども提案されている。これらの研究ではアニメーションデータを用いてキャラクターを動かしており、リアルタイムで動作するアバターに対

しては試みていない. さらに, 物理シミュレーションから運動学に切り替えるタイミングを判断する必要があり, リアルタイムで動作するアバターに使用するのは難しい.

物理を考慮したアバター

モーションキャプチャでリアルタイムにアバターを動かしつつ, 物理シミュレーションを使用することで接触時のキャラクターの反応動作を自動生成する試みに Nguyen らの研究 [9] がある. この研究ではアバターの各関節の角度と位置を PD 制御することでバーチャル物体と衝突した場合の動きを生成している. バーチャル物体と柔軟に衝突するには PD 係数を小さい値にしなければならない. しかし PD 係数を小さくするとモーションキャプチャに遅れて追従してしまう. PD 係数を大きくすることで追従性は向上するが, 一方で接触時の反応は固く不自然なものとなる. フィードフォワード制御による追従遅延の解消

PD 係数が小さいとアニメーションに遅れて追従してしまうという問題を解決する方法にフィードフォワード制御とフィードバック制御を併用する手法が提案されている. フィードフォワード制御によりアニメーションに追従するためのトルクを関節に加えることで, フィードバック制御の PD 係数が小さくとも正確な追従ができる. フィードフォワードのトルクを求めるには, 逆動力学計算を解く方法 [10], 別のシミュレーションシーンで PD 係数を大きく設定したうえで PD 制御を行いその際に発揮されたトルクをフィードフォワードトルクとして適用する方法 [11], [12] などが提案されている. これらの研究では, アニメーションデータで動かすキャラクターを対象としている. 本研究では, Rubens らの研究 [11] や Tokizaki らの研究 [12] と同様の手法を使用し, モーションキャプチャを用いたリアルタイムに操作されるアバターに適用することで『演者の負担軽減』を図る. また, Rubens らの研究 [11] や Tokizaki らの研究 [12] は手法の提案に焦点を当てており生成される映像の被験者実験による評価や遅延の計測は行っていない. そこで, 本研究では生成される映像が『自然な接触映像』の要求を満たすかを被験者実験により確認するとともに, 『非接触時の追従性』の要求を満たすかを遅延の計測によって評価する.

2.3 製品

物理シミュレーションと各関節の PD 制御により入力した動作データに追従するようにキャラクターを動かす PuppetMaster [13] という製品がある. 力学的な反応動作を自動生成でき, 入力にモーションキャプチャを使用することも可能である. しかし, PuppetMaster では 2.2 節で説明したように追従性と反応動作を両立することが難しい.

通常 VR ゲームでは操作しているアバターと物が接触した場合貫通してしまうが, 接触した場合に貫通しないゲームとしてソードオブガルガンチュア [14] という VR ゲーム

がある. このゲームではプレイヤーは VR 空間で近接武器や盾などを用いた剣戟アクションが楽しめる. 剣戟アクションをよりリアルにするために, ゲーム内で武器どうしが衝突した場合は貫通しないように処理される. しかし, アバター同士が接触した場合は貫通してしまう.

3. 提案手法

提案手法は, 衝突判定と接触力の計算が可能な剛体物理シミュレーションを用いることで, 接触の影響が考慮された動作をアバターに自動で行わせるものである. これにより違和感のある映像が出力される問題を解消するとともに, 演者が貫通の回避や接触に対する反応のタイミング調整などを意識する必要をなくし, 演者の負担を軽減する. また, 追従遅れの問題を解決するためにフィードフォワード制御を用いる.

提案システムの構成を図 1 に示す. 物理シミュレーションされたアバターをモーションキャプチャに追従させるために, 関節角の PD 制御 (比例・微分制御) を用いる. このとき, 接触時の柔らかな反応と, 非接触時の素早い動作への追従性の両立のために 2 つの物理シミュレーションを用い, それぞれ異なる係数の PD 制御を行う. 追従トルク計算用シミュレーションでは大きな係数の PD 制御を行い, 入力動作への素早い追従に必要な関節トルクを計算する. このとき接触の影響は考慮しない. メインシミュレーションでは追従用トルクを加味しつつ小さな係数の PD 制御を行うことで非接触時には遅れなく追従し, 接触時には柔らかく変形しつつゆるやかに演者のポーズに戻る挙動を実現する.

加えて, 柔らかい PD 制御の PD 係数を演者の持つコントローラによって調整可能とする. 係数が一定だと接触時の反応がつねに柔らかいものになってしまう. しかし, 人の身体の柔軟性はつねに一定ではなく, 筋肉に力を込め身体を硬くして衝撃に耐えることもできる. 接触反応におけるアバター身体の硬さを演者の意思で変化させられるよう

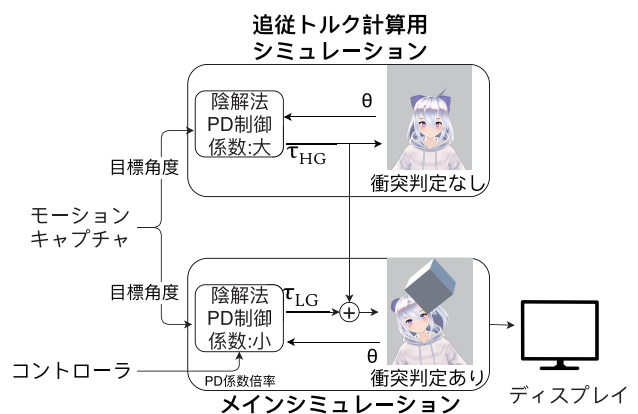


図 1 システム構成

Fig. 1 System structure.



図 2 アバターの衝突判定形状の例

Fig. 2 An example of collision detection shape of an avatar.

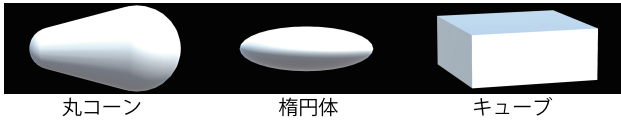


図 3 衝突判定用幾何プリミティブの例

Fig. 3 Examples of primitive shape for collision detection.

にすることはキャラクタを表現する上でも役に立つ。

なお、2つのPD制御を用いる同様の手法は先行研究[11]で提案されているが、本提案ではより安定な陰解法PD制御[12]を用いることで追従トルク計算用のPD制御により大きな係数を使用可能にする。また、文献[11],[12]ともにあらかじめ取得した動作データベースへの適用を想定したもので、リアルタイムモーションキャプチャによるアバター制御に用いる利点を示すことも本論文の目的である。

以降の節では手法の詳細を述べる。また、4章の評価で用いた実装例で使用したパラメータも述べる。

### 3.1 アバターの身体物理モデル

本研究ではアバターの3DCGモデルとして一般的なボーン入りメッシュモデルを想定し、アバターのボーン構造に即して物理シミュレーション用の剛体関節モデルを構築する。すなわち、各ボーンに対応する位置に剛体を配置し、ボーンの回転中心にあたる位置に関節を配置して、親子関係にあるボーンに対応する剛体どうしを接続する。ただしすべてのボーンに対応する剛体を作るのではなく、モーションキャプチャにより直接駆動されるボーンのみを対象とし、髪やスカートなどのいわゆる「揺れもの」に相当するボーンは除外する。また、関節は3自由度の球関節を用いる。

各剛体には衝突判定用の形状を割り当てる。ポリゴンメッシュでもよいが、幾何プリミティブを用いると計算量を抑えられる。実装例で用いた形状を図2に示す。図3に示す3種類の幾何プリミティブ(丸コーン・回転楕円体・直方体)、特に丸コーンを主に用いた。丸コーンは直径の異なる2球体を接平面でつなげたもので、主に関節位置に球が位置するようにして用いた。衝突判定形状はアバターの外形から極力はみ出さないように調整した。

剛体の質量は、人体各部の質量比[15]を参考にしたアバターの全体重に対する各剛体の質量比に基づき決定する。実装例では全体重を50kgとし、質量比は表1のとおりと

表 1 各部位の質量比

Table 1 Mass ratio of each body part.

部位	頭	首	肩	上腕	前腕
質量	0.057	0.013	0.019	0.038	0.019
部位	手	胸(上部)	胸(下部)	腹	腰
質量	0.0095	0.057	0.057	0.13	0.13
部位	大腿	下腿	足		
質量	0.11	0.057	0.019		

した。また、剛体の重心や慣性テンソルは、衝突判定形状をもとに密度一定と仮定して計算した。

### 3.2 PD制御による入力動作への追従

追従トルク計算用シミュレーションとメインシミュレーションのどちらでも、入力動作を目標とするPD制御を行って剛体関節モデルを入力動作に追従させる。なお、以下で関節角度・剛体姿勢はクォータニオン、位置・姿勢・角速度は3次元ベクトルである。

関節*i*に対するPD制御のP係数 $P_i$ 、D係数 $D_i$ 、目標角度 $q_i^{goal}$ 、目標角速度 $\omega_i^{goal}$ 、現在の関節角度 $q_i$ 、角速度 $\omega_i$ に対し、関節トルク $\tau_i$ のPD制御を式(1)に示す。

$$\tau_i = P_i Rot(q_i^{goal} q_i^{-1}) - D_i(\omega_i - \omega_i^{goal}) \quad (1)$$

ただし $Rot(q)$ はクォータニオンを対応する回転ベクトルに変換する関数であり、以下のように定義する。

$$Rot((w, x, y, z)) := 2 \arccos(w) \frac{(x, y, z)}{|(x, y, z)|}$$

また、ルートボーンである腰剛体は位置・姿勢をPD制御する。腰剛体の位置に対するPD制御のP係数 $P_{wf}$ 、D係数 $D_{wf}$ 、姿勢に対するPD制御のP係数 $P_{w\tau}$ 、D係数 $D_{w\tau}$ 、目標位置 $p_w^{goal}$ 、目標姿勢 $q_w^{goal}$ 、目標角速度 $\omega_w^{goal}$ 、現在の位置 $p_w$ 、速度 $v_w$ 、姿勢 $q_w$ 、角速度 $\omega_w$ に対し、腰剛体に対する力 $f_w$ とトルク $\tau_w$ のPD制御を式(2)、(3)に示す。

$$f_w = P_{wf}(p_w^{goal} - p_w) - D_{wf}(v_w - \dot{p}_w^{goal}) \quad (2)$$

$$\tau_w = P_{w\tau} Rot(q_w^{goal} q_w^{-1}) - D_{w\tau}(\omega_w - \omega_w^{goal}) \quad (3)$$

実際には、提案手法では通常のPD制御ではなく陰解法によるPD制御を用いる。陰解法PD制御はPD制御を式(4)のように表現して関節・接触などを実現する他の拘束条件との連立方程式とし、その解として関節角速度 $\omega_i[n+1]$ および関節トルク $\tau_i[n]$ を得る手法である。通常のPD制御と比べて非常に大きなPD係数を設定しても発振・発散せず安定に動作する特長を持つ。詳細は文献[12]を参照されたい。

$$\tau_i[n] = P_i(\Delta q_i[n] + \omega_i[n+1]\Delta t) - D_i(\omega_i[n+1] - \omega_i^{goal}[n+1]) \quad (4)$$

$$\text{ただし } \Delta \mathbf{q}_i = \text{Rot}(\mathbf{q}_i^{\text{goal}} \mathbf{q}_i^{-1})$$

各関節の  $P_i$ ,  $D_i$  は重い身体部位を駆動する関節ほど大きな値とする。先行研究 [9] を参考に、各関節により動かされる身体部位の T ポーズにおける慣性モーメントに比例するものとした。具体的には次のとおりである。木構造の剛体関節モデルにおいて関節  $i$  より葉側に剛体  $\{s_0^i, s_1^i, \dots, s_N^i\}$  があるものとする。関節  $i$  の回転中心位置を  $\mathbf{c}_i$ 、剛体  $s$  の質量を  $M(s)$ 、キャラクタが T ポーズをとった際の剛体  $s$  の重心位置を  $\text{CoM}(s)$  としたとき、 $P_i$  は式 (5) により決定する。

$$P_i = P_{\text{ratio}} \sum_{k=0}^N M(s_k^i) |\mathbf{c}_i - \text{CoM}(s_k^i)|^2 \quad (5)$$

$P_{\text{ratio}}$  はキャラクタの身体全体の P 係数に一律でかかる倍率である。大きな値にすれば硬く素早い追従になり、小さな値にすれば衝突に対し柔らかく反応するようになる。追従トルク計算用シミュレーションとメインシミュレーションではこの倍率を異なる値とする。詳しくは次節で述べる。関節によっては  $P_i$  が小さくなりすぎるため、 $P_i \leq P_{\text{min}}$  ならば  $P_i = P_{\text{min}}$  とする。さらに  $D_i$  を臨界減衰比の式を参考とした式 (6) により決定する。

$$D_i = 2 \sqrt{P_i \sum_{k=0}^N M(s_k^i) |\mathbf{c}_i - \text{CoM}(s_k^i)|^2} \quad (6)$$

### 3.3 フィードフォワード制御とフィードバック制御

アバター物理モデルを動作に追従するための追従用トルクを計算し、そのトルクを関節に加えて駆動する。追従用トルクはモーションキャプチャ入力のみによって決まり、制御対象であるメインシミュレーション内身体モデルの姿勢のフィードバックを受けないので、これはフィードフォワード制御である。フィードフォワード制御を、柔らかい PD 制御によるフィードバック制御と組み合わせることで、モーションキャプチャへの素早い追従と接触・衝突時の柔らかい反応動作を両立する。

追従用トルクの計算は、メインシミュレーションとは別に用意したシミュレーションで行う (図 1 参照)。追従トルク計算用シミュレーションでは衝突判定を無効化し、大きい PD 係数の PD 制御を行い、PD 制御に要したトルク (式 (4) の  $\boldsymbol{\tau}$ ) を求める。これが追従用トルク  $\boldsymbol{\tau}_{HG}$  である。

メインシミュレーションでは、衝突判定を有効化した上で、各関節に追従用トルク  $\boldsymbol{\tau}_{HG}$  と、弱い PD 制御によるトルク  $\boldsymbol{\tau}_{LG}$  の両方を加えて動作させる。接触発生時にはモーションキャプチャ入力から大きく逸脱したのち、弱い PD 制御によりゆっくりと入力姿勢に戻される。

実装例における PD 係数の値について述べる。メインシミュレーションの PD 係数は、各関節は 3.2 節の方法で  $P_{\text{ratio}} = 30$ ,  $P_{\text{min}} = 10$  として求めた。腰

は  $P_{wf} = 50000$  [N・m/rad],  $D_{wf} = 5000$  [N・m・s/rad],  $P_{w\tau} = 5000$  [N・m/rad],  $D_{w\tau} = 500$  [N・m・s/rad] という大きい値として接触の影響を受けづらくした。これは接触により腰が大きく動くと足が滑って不自然な動作となりがちなためである。また、追従トルク計算用シミュレーションは、P 係数・D 係数をともにメインシミュレーションの 1,000 倍とした。

### 3.4 アバターの体の硬さの変化

人の身体の硬さは一定ではなく、主動筋・拮抗筋の同時収縮により身体を硬くすることができる。アバターでも同様に身体の硬さを変化させることができれば、同じ接触力に対しても、リラックス時は柔らかく反応し、身構えているときは硬く反応するなど異なる表現ができる。提案手法では、図 1 のメインシミュレーションにおける PD 制御が接触時の反応を担う。したがって、この PD 制御の係数に一律で倍率  $k_{\text{stiff}}$  をかけ、 $k_{\text{stiff}}$  を演者が操作できるようにする。

実装例では、Oculus Touch のアナログ入力ボタンを用いた。ボタンは人差し指・中指の位置にあり、指の握り込み度合いを 0 から 1 のアナログ値で取得できる。0 (まったく握り込んでいない) のときに  $k_{\text{stiff}} = 1$ , 1 (完全に握り込んでいる) のときに  $k_{\text{stiff}} = 9$  となるようにした。

なお、力のかけ具合をより直接的に取得する方法として筋電計が考えられる。しかし実際に試したところ、演者が動いただけで反応してしまい、身体の硬さを意図どおりに操作するという点では必ずしも使いやすくなかったため、評価用の実装例では使用していない。

### 3.5 アバター身体への重力作用の無効化

提案手法の物理シミュレーション内では、物体のみに重力を適用し、アバターには重力を適用しない。

アバターの身体に重力を適用すると追従性が低下する。これは重力に対抗するトルクを余分に計算する必要が生じて関節の追従トルクに誤差が生じるためと考えられる。一方で仮にアバターの物理モデルに重力を適用したとしても、提案手法では追従トルク計算の過程で体重を支える力が加味されるため、結局は重力が存在しないかのように動作するはずである。また演者の実身体にすでに重力が作用しており、演者への追従という点からもシミュレーション内で重力を新たに加える必要はない。ゆえに、そもそもアバターの身体に重力を適用しないことで問題が解決する。

## 4. 評価

本章ではまず、提案手法による映像の例示により提案手法の有用性を示す (4.2 節)。次に提案手法が 1 章の 3 点の要求を満たすことを確認するために他の手法との比較実験を行い提案手法を評価する— 4.3 節で『非接触時の追従

性』の要求に対する評価, 4.4 節で『自然な接触映像』の要求に対する評価, 4.5 節で『演者の負担軽減』の要求に対する評価を行う。

物理シミュレーションを使用して, モーションキャプチャによりアバターを動かす手法として, 以下の4つについて比較を行った。

**ONEWAY** モーションキャプチャをそのままアバターに適用, 物体が衝突してもアバターはいっさい影響を受けず物体のみ一方的に動かされる状態

**PD-WEAK** モーションキャプチャを目標角度とし, 各関節を PD 制御 (PD 係数小:  $P_{ratio} = 30, P_{min} = 10$ )

**PD-STRONG** モーションキャプチャを目標角度とし, 各関節を PD 制御 (PD 係数大:  $P_{ratio} = 2100, P_{min} = 10$ )

**PLIANT** 提案手法

ONEWAY は現行の VTuber など広く使われる手法を想定した例である。アバターと物が衝突しても貫通しないが, アバター同士の衝突ではどちらのアバターも適用されたポーズから動かず貫通してしまう。PD-STRONG の PD 係数は追従性に支障がでない程度の大きい値に設定した。PD-WEAK の PD 係数は接触が生じたとき PLIANT と同じ反応が得られるように PLIANT のメインシミュレーション側の PD 係数と同じ値に設定した。

#### 4.1 評価に用いた実装

開発は広く使用されているゲームエンジンである Unity 上で行った。物理シミュレーションには陰解法 PD 制御を標準機能として搭載する Springhead [16] を用いた。

操作入力デバイスとして OculusRift および OculusTouch を用い, 頭と両手の位置姿勢を検出した。頭と両手の位置姿勢から逆運動学計算により全身の関節角を推定し, モーションキャプチャ入力として用いた。逆運動学計算には Unity のアセットである FinalIK [17] を用いた。

#### 4.2 生成された映像の例

実際に提案手法を用いて様々な演技を行い映像を作成した。結果を付録映像 6 秒から 1 分 7 秒, および図 4 に示す。映像のアバターはすべて提案手法により人が演じている。それぞれの映像の下部に体の硬さとして  $k_{stiff}$  を表示している。体の硬さが表示されていない側のアバターは  $k_{stiff} = 1$  である。

- (1) ハイタッチ アバター (左) の手とアバター (右) の手が接触することで, 両方のアバターが動いていることが分かる。
- (2) 弱い平手打ち, (3) 強い平手打ち 演者がコントローラによりアバター (左) の体の硬さを調整することで異なる反応動作を生成することができた。
- (4) 握手 この映像ではアバター (左) の右手とアバター

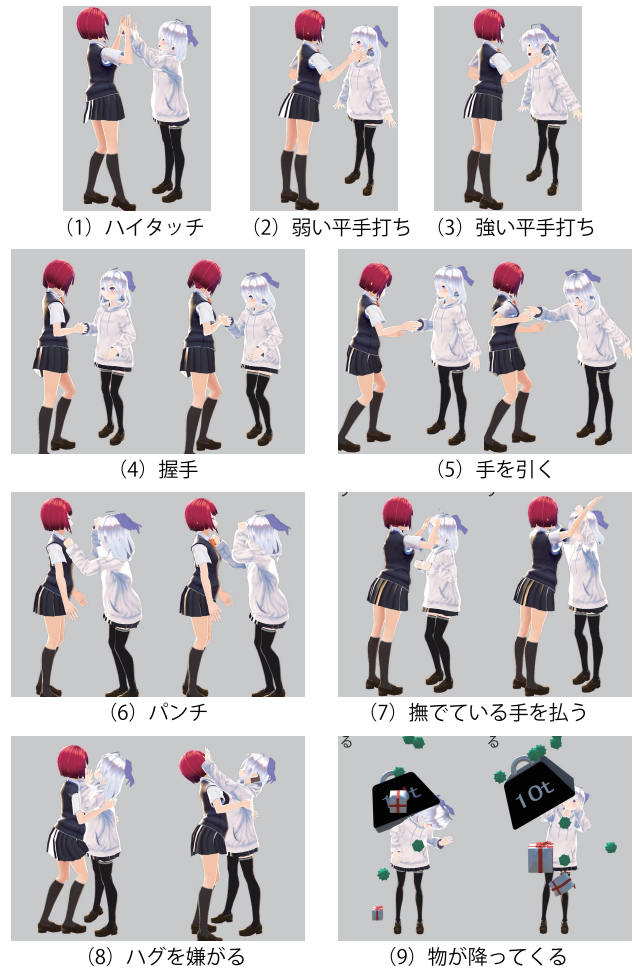


図 4 生成された映像の例 (付録映像 6 秒参照)  
 Fig. 4 Examples of generated movies (See appendix video after 6 seconds).

- (右) の右手を位置拘束用ばねでつなげている。位置拘束用ばねの P 係数は  $100000 [N \cdot m/rad]$ , D 係数は  $10000 [N \cdot m \cdot s/rad]$  である。アバター (左) の右手を動かせば, アバター (右) の右手も連動するため, アバター (右) は相手の手の動きにあわせようとする必要がなく容易に演技ができた。
- (5) 手を引く 『握手』と同様の位置拘束用ばねで手同士をつなげている。アバター (左) は体を硬くすることでアバター (右) より強い力を出し, アバター (右) の身体を大きく動かすことができた。
- (6)パンチ 弱くパンチする映像である。
- (7) 撫でている手を払う アバター (左) の演者が両手で撫でていた最中に, アバター (右) の演者が手を払う動作をする。体を硬くしているアバター (右) の動作がより優先されることが分かる。
- (8) ハグを嫌がる アバター (左) はハグをしようとするが, アバター (右) がそれを嫌がり押し返そうとする映像である。このような複雑な接触が生じる映像も作成できた。

(9)物が降ってくる 10tと書かれた黒い重りの質量は10kg, 緑の球は1kg, 赤いリボン付きのキューブは3kgである. 重たい物がアバターに衝突すると大きく動き, 軽い物がアバターに衝突してもあまり動かないといった反応動作が得られた. 最初は体の硬さは小さく, 黒い重りが当たると大きく動くが, 途中から体の硬さを大きくすることで耐えるような動作ができていくことが分かる.

### 4.3 追従性の評価

1章であげた『非接触時の追従性』の要求が満たしているかどうかを評価する. 追従性の尺度として, 同一の入力動作を与えた場合の, 各時刻  $t$  [s] における, 入力動作における右手の目標位置とアバターの右手の位置との差分  $d$  [m], および入力動作における目標関節角度とアバターにおいて実現した各関節角度を計測した.

与えた入力動作を付録映像1分11秒および図5, 図6に示す. 提案手法が手先位置計測と逆運動学(IK)を組み合わせた環境で利用されることを想定し, IKを用いて右手を移動させる動作とした. 具体的には, 黄色の球の位置から赤色の球の位置へ定速(5[m/s])で移動し運動前後では静止するような目標位置を, 右手で追従するように, 腰から上の上半身の各関節角度をIKにより計算した. この動作では, 右腕が伸びきった状態から開始し, 主に右肩関節と右肘関節が連動して手先を移動させ, 再び右腕が伸びきった姿勢で静止する. IKの計算にはFinalIK[17]を用いた. IKの追従誤差は十分少なく, 付録映像および図6のとおり, 入力動作の右手はほぼ定速(5[m/s])でおおむね直線的に動作した.

なお, 本評価で追従性の基準とする入力動作はIK計算後のものであり, IK自体による追従誤差は含まれない.

#### 4.3.1 結果と考察

目標位置からの差分  $d$  [m] のグラフを図7に, 目標関節



図5 追従性評価用入力動作(付録映像1分11秒)

Fig. 5 Input motion for trackability evaluation (See appendix video after 1m11s).



図6 追従性評価用入力動作(頭上視点)

Fig. 6 Input motion for trackability evaluation (Top view).

角度とアバターの関節角度のグラフを図8に示す. 入力動作に含まれる上半身の関節のうち, 図8では右肩関節, 右肘関節, 胸部の関節の3関節のみグラフで示した. 右肩, 右肘は入力動作で大きく動き, 手法間でも顕著な差が見られた関節であり, 胸部の関節は入力動作での動きは大きくないものの手法間での差が見られた関節である. また, 関節角度としては, 親関節からの相対回転を示すオイラー角( $\theta_x, \theta_y, \theta_z$ )のうち鉛直軸方向の回転を表し運動の主な方向であった  $\theta_y$  をグラフに示した.  $\theta_x, \theta_z$  は動作量が少ないため省略したが,  $\theta_y$  と同様の傾向が見られた.

図7, 図8からPLIANT, PD-STRONG, PD-WEAK

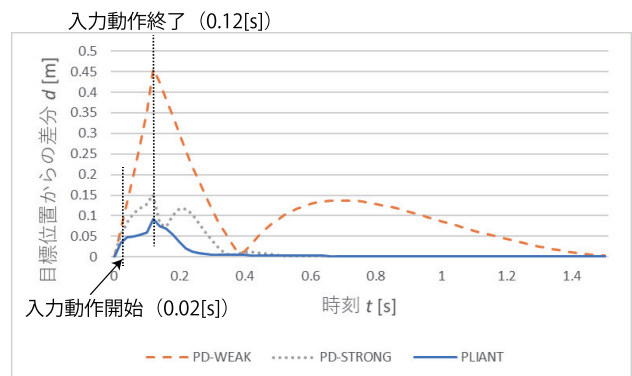


図7 「右手の」目標位置からの差分  
Fig. 7 Difference from target position.

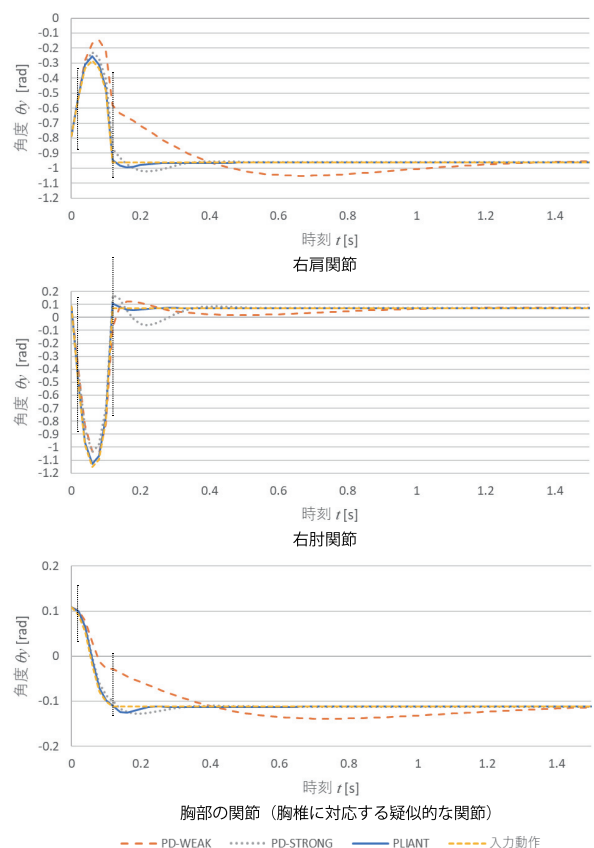


図8 各関節の目標角度・関節角度  
Fig. 8 Target angle · joint angle of each joint.

の順に追従性が良いことが分かる。さらに、付録映像1分14秒を見るとPD-WEAKではアバターの右手が赤い球で停止できずオーバシュートしてしまっている。この状態では演技に支障をきたす。

図7, 図8を見ると, PLIANTでも目標位置, 角度になるまでに少し時間がかかっていることが分かる。この原因は, PLIANTでもPD制御であるため遅延が生じ, オーバシュートが生じているためだと考えられる。また遅延はほぼなく, オーバシュートも0.2[s]程度で収まっている。そのため, PLIANTでは十分に『非接触時の追従性』の要求が満たされているといえる。

PD-WEAKは『非接触時の追従性』の要求を満たさなかった一方で, PD-STRONGの追従性はPLIANTとほぼ同等である。そこで以降では, PLIANTとPD-STRONG, および原理的に正確な動作が可能なONEWAYを比較対象としてさらに評価を行う。

#### 4.4 作成された映像の自然さの評価

##### 4.4.1 目的

1章であげた『自然な接触映像』の要求が満たしているかどうかを評価する。ONEWAY, PD-STRONGによる作成映像とPLIANTによる作成映像の自然さを比較する。作成映像を被験者に見せ, どちらの手法がより自然か5段階評価で選んでもらい, 選んだ理由を書いてもらった。アンケートに使用した映像(付録映像1分19秒および図9)は以下の4つの状況を想定した映像である。付録映像2分2秒のように貫通している映像は明らかに不自然であるため, ONEWAYでは貫通しないよう演技してきた映像を用いた。

『10kg』 10kgのキューブが秒速4.1mで後頭部にぶつかる。

『1kg』 1kgのキューブが秒速4.1mで後頭部にぶつかる。

『撫でる』 頭を撫でる。

『ハグ』 ハグ。

大きな力が加わる状況として『10kg』, 小さな力が加わる状況として『1kg』を選定した。さらに, 応用先として考えられるVRメタバースやVTuberの配信などで頻繁に見られる状況として『撫でる』と『ハグ』を選定した。



図9 アンケート用映像(付録映像1分19秒)(PLIANTのみ)

Fig. 9 Movies for questionnaire (See appendix video after 1 m19s) (only PLIANT).

##### 4.4.2 方法

『10kg』と『1kg』では比較条件を揃えるため, モーションキャプチャ入力は同一の静止ポーズとした。ONEWAYによる映像(付録映像2分9秒参照)はアバターが動かず明らかに不自然であるため比較対象から外した。

『撫でる』と『ハグ』の白髪のキャラクター(右)はアニメーションで動かし, 赤髪のアバター(左)は筆者が操作して映像を作成した。白髪のキャラクター(右)はまったく動かない静止状態では不自然であるため, 身じろぎするようなアニメーションを適用した。

『ハグ』については, アバターの腰が動くと言いつづけてしまうという問題が生じた(詳細は5章で述べる)。この問題に対応するためにPLIANTとPD-STRONGではこの動作のみキャラクターの足剛体と下腿剛体に対して位置姿勢のPD制御を追加で行い, 足が滑らないようにした。

『撫でる』と『ハグ』ではアンケートに使用した映像は手法ごとに演技したものであり, 同一の状況を演技したとしても, 頭を撫でる位置が違うといった演技の小さな差に着目されると手法の違いを評価できないおそれがあった。そのため, 被験者には接触に注目して自然かどうか判断するように伝えた。映像は何度でも見てよいとした。

被験者にはどちらが提案手法であるかは伝えていない。付録映像に掲載したアンケート用映像(1分19秒)では上部に手法名を記述したが実際のアンケートでは識別用のA, Bのみ表示した。A, BどちらをPLIANTまたは比較対象の手法にするかはランダムに決定した。

被験者は20代から30代の男性10名であった。

##### 4.4.3 結果と考察

アンケート結果を図10に示す。PLIANTに対する比較対象となった手法をカッコ内に記述した。

図10からPLIANTとONEWAYによる映像の比較では, PLIANTによる映像のほうがより自然と答える人数の方が多かった。PLIANTを選んだ理由として「接触しているように見えるため」という意見が多く, PLIANTの映像では両アバターとも接触した際に動くため人数が多かったと考えられる。PLIANTとPD-STRONGによる映

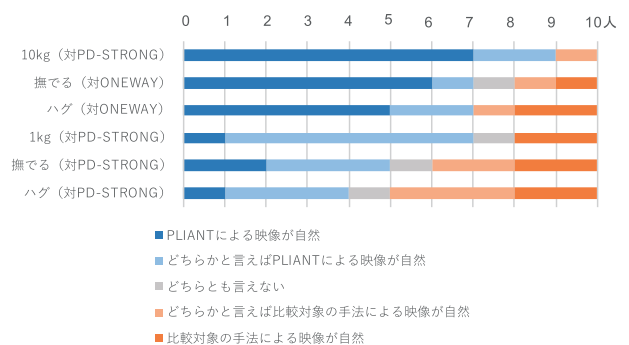


図10 自然さのアンケート結果

Fig. 10 result of naturalness evaluation.



像の比較では『10 kg』、『1 kg』では PLIANT による映像がより自然と答える人数の方が多かった。しかし、アバター同士接触が生じる『撫でる』、『ハグ』では PLIANT と PD-STRONG で人数差はほとんど見られなかった。その理由は、『撫でる』、『ハグ』はどちらのアバターも大きく動かされない接触であるためだと思われる。アバター同士接触が生じる場合でも、より大きく動くことが期待される 4.2 節の (3) 強い平手打ちなどでは、PLIANT による映像のほうがより自然と答える人数の方が多くなると考えられる。

『10 kg』で PD-STRONG は明らかに PLIANT よりも反応動作が小さい。PD-STRONG で反応動作を大きくする方法としてキューブの質量を大きくする方法が考えられる。しかし、この方法では質量が違うことで反応動作が現実とは違う。さらに、PD 係数や質量などを現実を基準にできず、試行錯誤することではしか決定できない。したがって、キューブの質量を大きくする方法は比較対象としなかった。

以上のように、PLIANT による映像は ONEWAY, PD-STRONG による映像に対しより自然であるか、同等に自然であることが示唆された。このことから、PLIANT は『自然な接触映像』の要求を ONEWAY, PD-STRONG 以上に満たすと考えられる。

## 4.5 演技の容易さの評価

### 4.5.1 目的

1 章であげた『演者の負担軽減』の要求が満たしているかどうかを評価する。この評価実験では被験者には演者になってもらい、アバターを操作して演技してもらった。提案手法 PLIANT に対する比較対象としては、現在広く使用されており、「接触した振り」の演技が必要な手法である ONEWAY とした。

### 4.5.2 方法

被験者に演技してもらう動作は以下の 3 つである。

『ぶつかる』 キューブがアバターにぶつかる。

『撫でる』 動くキャラクタを撫でる。

『叩く』 色が変わったパネルを素早く叩く。

上記動作それぞれについて以下の手順で実験を行った。

手順 (1) 動作例、自然さの指標として手本映像 (付録映像 2 分 14 秒および図 11) を見せる。

手順 (2) アバター制御手法の特徴の簡単な説明。

手順 (3) 演技をしてもらう、アバターの映像を記録する。

手順 (4) アバターの映像が自然か演者が判断する。不自然なら手順 (3) へ、自然なら終了。

簡単に演技できるかの指標として、手順 (3) を実行した回数  $n$  を測った。実験終了後に感想を聞き記録した。提案手法が応用される場合に利用者は手法の特徴を知ったうえで適した演技を行うことが想定される。そのため本実験では

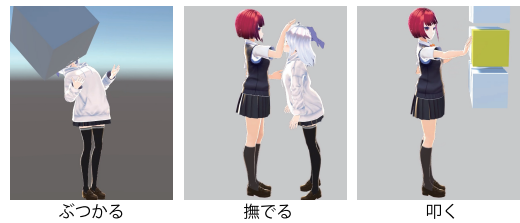


図 11 手本映像 (付録映像 2 分 14 秒参照)

Fig. 11 Model movie (See appendix video after 2m14s).

手順 (2) において「一方的に跳ね返す手法 (ONEWAY)」, 「お互いに影響を受ける手法 (PLIANT)」という特徴を伝えた。今回の実験に使用する動作には PLIANT の 3.4 節の体の硬さを変化させる入力が必要ないため、演者が誤って入力してしまわないように  $k_{stiff} = 1$  の状態で機能を除いた。実験では順序効果が予想されたため ONEWAY と PLIANT のどちらを先にするかをランダムに決定した。実験では 5 名に演者になってもらった。

### 自然さについて

アバターの映像の自然さについては手順 (1) の手本映像 (付録映像 2 分 14 秒および図 11) の自然さ以上の映像が撮れたら自然とし終了とする。今回演者の腰や足の姿勢を取得しておらず、アバターの腰や足の姿勢を思いどおりに動かせないため、腰や足の動きは自然さの対象に含めない。さらに、5 章で示すようにアバターの頭も演者が思いどおりに動かすことが難しい場合がある。その影響が顕著な『叩く』では頭の動作は自然さの対象には含めないよう伝えた。

### 動作の選定について

『演者の負担軽減』の要求が満たされているかを示すために、ONEWAY だと演者が接触している振りの演技が必要となるような動作を選定した。具体的には、アバターに大きな反応動作が求められる『ぶつかる』を選定した。さらに、予測不可能な相手の動きに合わせて動かなければならない『撫でる』を選定し、慌てている状況でアバター操作が正確にできなくなるような動作として『叩く』を選定した。

### 比較する手法について

本節の実験では、PLIANT の有効性を示すため、広く使用されている ONEWAY と PLIANT を比較する。PD-WEAK は 4.3.1 項で『非接触時の追従性』の要求を満たさなかったため比較対象としなかった。

PD-STRONG を比較対象としなかったのは、次の理由による。4.4 節の結果から、自然な動作を生成するためには、被験者は貫通を避け、状況にあわせて大きな反応動作が生成されるよう演技する必要があることが分かる。貫通は、アバター同士や固定物体など物理シミュレーションされていない物体とアバターが接触した場合に、ONEWAY で生じる。大きな反応動作は、アバターに大きな外力が加わる

場合に生じ、PLIANTとPD-WEAKでは自動生成され、ONEWAYとPD-STRONGでは演技が必要となる。以上から、PD-STRONGでは、同一の入力に対して、貫通についてはPLIANTと、大きな反応動作についてはONEWAYと同様の動作が生成されることが分かる。

一方、対象動作では、『撫でる』と『叩く』ではONEWAYで貫通が生じる可能性があり、『ぶつかる』で大きな反応動作が必要となる。このため、PD-STRONGを用いて本節の実験を行った場合、『撫でる』と『叩く』ではPLIANTと同様、『ぶつかる』ではONEWAYと同様の結果が得られることが予測できる。そこで、被験者の負担の低減を考えPD-STRONGを省いた。

#### 4.5.3 結果と考察

実験の結果として回数  $n$  を図 12 に示す。3つの動作すべてで、ONEWAYの方がPLIANTより多いか、同じ回数を要した。また  $t$  検定の結果『叩く』については回数に有意差がみられた。

被験者の感想では、「ONEWAYでは相手の動きを予想しながら演技しないとイケないので非常に疲れる」、「PLIANTでは叩くことに集中できた」などの意見が得られた。

『ぶつかる』においてONEWAYではキューブにぶつかった後演技しなければならない。被験者Bの1回目（付録映像2分18秒）を見ると、ONEWAYではキューブにぶつかったときの動作が小さく不自然としてみう1回演技をしたことが分かる。一方、PLIANT側は反応動作が自動生成されるので1回で終了している。

『撫でる』においてONEWAYでは、被験者Eの1回目（付録映像2分41秒）のようにアバター（左）の手とアバター（右）の頭が貫通してしまう。さらに被験者Eは2回で終了しているが2回目（付録映像2分57秒）でもONEWAY側は貫通してしまっている。一方、PLIANT側は1回目も2回目も貫通していないことが分かる。

『叩く』においてONEWAYではキューブと手が被験者Bの2回目（付録映像3分23秒）のように貫通してしま

うが、PLIANTでは貫通しない。被験者Cの1回目（付録映像3分51秒）では貫通しないように意識した結果、PLIANT側よりも動作が遅いことが分かる。

以上より、PLIANTはONEWAYよりも『演者の負担軽減』の要求を満たす手法であるといえる。

## 5. 課題

提案手法では付録映像4分18秒のように動かない白い壁を押しした場合に胸関節が曲がってしまいアバターが仰け反るが、演者は仰け反っていることに気付かないという問題がある。壁を押しときに押し返される挙動は力学的に正しい。しかし、演者の意図と反する場合がある。これを解決する方法として演者の視点を改善する手法が考えられる。今回演者の視点はモーションキャプチャの頭の位置としたが、アバターの視点にすることでアバターの頭が動いたこととの理解につながる可能性がある。しかし、試したところ演者の動作と無関係に視点が動くため酔うことが分かった。また、視点が動いただけでアバターの頭が動いていると理解するのが難しかった。

付録映像1分20秒のように頭に重いものが当たった場合に足が動かないのは不自然である。このような状況では片足が前に出ること、倒れないようにバランスを保とうとする動作が理想的であると思われる。体の重心位置を用いて片足を前に出すか判断するようなバランス制御をすると改善するだろう。

今回使用したアバターの身体物理モデルではルートボーンである腰が動いてしまうとアバターの全身が動いてしまい、足が滑ってしまい不自然な動作になってしまうという問題がある。4.2節の(8)ハグを嫌がるではこの問題が発生している。

## 6. 結論

提案手法では物理シミュレーションを用いて自然な接触動作を自動生成し、その際に問題となる追従遅れをフィードフォワード制御により解決した。付録映像により、提案手法によりこれまでできなかったアバター同士やアバターと物の自然な接触表現が可能であることを示した。さらに、提案手法はアバター制御手法が満たすべき3点の要求を満たすことを評価実験により示した。

## 参考文献

- [1] Arikan, O., Forsyth, D.A. and O'Brien, J.F.: Pushing people around, *ACM SIGGRAPH/5th Eurographics Symposium on Computer Animation*, pp.59-66 (2005).
- [2] Oshita, M.: Motion-capture-based avatar control framework in third-person view virtual environments, *ACE'06*, p.2 (2006).
- [3] Shapiro, A., Pighin, F. and Faloutsos, P.: Hybrid control for interactive character animation, *Proc. 11th Pacific Conference on Computer Graphics and Applications*,

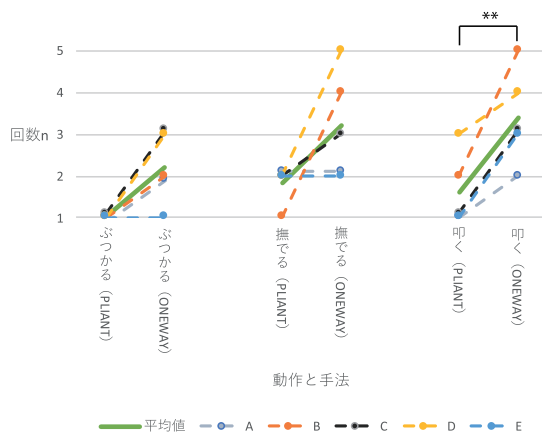


図 12 演技にかかる回数  $n$ , \*\*:  $p < 0.01$

Fig. 12 Number of takes, \*\*:  $p < 0.01$ .

- pp.455–461 (Oct. 2003).
- [4] Zordan, V.B., Majkowska, A., Chiu, B. and Fast, M.: Dynamic response for motion capture animation, *ACM Trans. Graph.*, Vol.24, No.3, pp.697–701 (July 2005).
  - [5] Nguyen, N.H., Arista, R., Liu, C.K. and Zordan, V.: Adaptive dynamics with hybrid response, *SIGGRAPH Asia 2012 Technical Briefs, SA'12*, Association for Computing Machinery (2012).
  - [6] Zordan, V.B. and Hodgins, J.K.: Motion capture-driven simulations that hit and react, *Proc. 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA'02*, pp.89–96, Association for Computing Machinery (2002).
  - [7] Electronic, P., Odest, A., Jenkins, O., Mcguire, M. and Colledge, W.: Dynamo: Dynamic, data-driven character control with adjustable balance, *Sandbox Symposium* (2006).
  - [8] Oshita, M. and Makinouchi, A.: A dynamic motion control technique for human-like articulated figures, *Comput. Graph. Forum*, Vol.20 (2001).
  - [9] Nguyen, N., Wheatland, N., Brown, D., Parise, B., Liu, C.K. and Zordan, V.: Performance capture with physical interaction, *Proc. 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA '10*, pp.189–195, Eurographics Association (2010).
  - [10] Yin, K.K., Cline, M.B. and Pai, D.K.: Motion perturbation based on simple neuromotor control models, *Proc. 11th Pacific Conference on Computer Graphics and Applications*, pp.445–449 (2003).
  - [11] Nunes, R.F., Vidal, C.A., Cavalcante-Neto, J.B. and Zordan, V.B.: Simple feedforward control for responsive motion capture-driven simulations, Bebis, G., Boyle, R., Parvin, B., Koracin, D., Remagnino, P., Porikli, F., Peters, J., Klosowski, J., Arns, L., Chun, Y.K., Rhyne, T.-M. and Monroe, L. (Eds.), *Advances in Visual Computing*, pp.488–497, Berlin, Heidelberg, Springer Berlin Heidelberg (2008).
  - [12] Tokizaki, T., Tazaki, Y., Mitake, H. and Hasegawa, S.: Pliant motion: Integration of virtual trajectory control into lcp based physics engines, *SIGGRAPH '09: Posters, SIGGRAPH '09*, p.10:1, ACM (2009).
  - [13] RootMotion: unity assetstore puppetmaster, available from (<https://assetstore.unity.com/packages/tools/physics/puppetmaster-48977>) (accessed 2020-01-25).
  - [14] gumi, YOMUNECO: Swords of gargantua tesseract abyss, available from (<https://www.gargantuavr.com/?lang=ja>) (accessed 2020-01-25).
  - [15] 国立研究開発法人産業技術総合研究所：人体特殊文献データベース，入手先 (<https://unit.aist.go.jp/hiri/dhrg/ja/dhdb/properties/segment/k-07.html>) (参照 2019-01-11).
  - [16] 長谷川晶一，三武裕玄，田崎勇一：動作行動開発のための物理エンジン springhead, *日本ロボット学会誌*, Vol.30, No.9, pp.841–848 (2012).
  - [17] RootMotion: Final ik, available from (<https://assetstore.unity.com/packages/tools/animation/final-ik-14290>) (accessed 2020-01-25).



杉森 健 (正会員)

1996年生。2019年東京工業大学工学部情報工学科卒業。同年同大学大学院情報通信系修士課程入学，現在に至る。インタラクティブキャラクターの研究に従事。



三武 裕玄 (正会員)

1984年生。2006年東京工業大学工学部情報工学科卒業。2008年同大学院知能システム科学専攻修士課程修了。同年日本学術振興会特別研究員。2011年同専攻博士課程修了，博士(工学)。同年東京工業大学精密工学研究所助教，現在に至る。インタラクティブキャラクターの研究に従事。



佐藤 裕仁 (学生会員)

1993年生。2016年東京工業大学工学部情報工学科卒業。2018年同大学大学院情報通信系修士課程修了。同年同大学同系博士後期課程入学，現在に至る。インタラクティブキャラクターの研究に従事。



小栗 賢章 (学生会員)

1994年生。2019年東京工業大学工学部電気電子工学科卒業。同年同大学大学院情報通信系修士課程入学，現在に至る。インタラクティブキャラクターの研究に従事。



長谷川 晶一 (正会員)

1974年生。1999年東京工業大学大学院知能システム科学専攻修士課程修了。同年ソニー株式会社入社。2000年東京工業大学精密工学研究所助手。2007年電気通信大学知能機械工学科准教授。2010年東京工業大学精密工学研究所准教授，現在に至る。バーチャルリアリティ，ハプティクス，ヒューマンインタフェースの研究に従事。博士(工学)。