

避難行動シミュレーション可視化のための アバターの基本動作の補間と合成

Interpolation and Composition of Basic Behaviours for Visualizing of a Multi-Agent Evacuation Simulation

金子雄太[†], 飯島 正[‡]

Yuta Kaneko[†], and Tadashi Iijima[‡]

[†]慶應義塾大学大学院 理工学研究科

[‡]慶應義塾大学 理工学部

[†]Graduate School of Science and Technology, Keio Univ.

[‡]Faculty of Science and Technology, Keio Univ.

要旨

エージェントベースドシミュレーション手法に基づく避難行動シミュレーション結果を可視化するにあたり、各エージェントの振る舞いを三次元的なアバターとして表現することを試みている。その際、各エージェントの行動は各時点/各地点での基本動作の集合として、時間的かつ空間的な面で離散的に与えられる。そのため、滑らかな動作として再現するためには、基本動作の補間と合成といった基礎技術を確認する必要がある。ここでは筆者らが構築中である、基本動作の補間と合成の手法に関して報告する。

1. はじめに

地震や津波などの災害が発生する前に、避難訓練を行っておくことは非常に重要である。学校や自治体などが催している避難訓練に参加することは大切だが、決して十分な訓練が行われているとは限らない。

小中学校での避難訓練では、地震の発生を想定すると、まず、机の下に隠れて頭部を保護するという初期行動をとった後は、集団で隊列をなして避難場所（多くの場合、校庭に設定されている）へ移動するという行動訓練が行われることが多い。しかし、そこでは、地震に伴う屋内火災（火や煙の発生と流動）や、屋内設備施設の損壊や故障、障害物となる散乱した備品などは用意されない。想定しなければならぬケースがあまりにも多くて準備できないこと、実際に、それらを配備すると、被訓練者に危険が伴うこと、などがその理由である。

そこで著者らの研究プロジェクトの一環として、広域災害や屋内災害などに関する避難シミュレーションを避難訓練に導入する試みを行っている。その一つに、三次元仮想空間において、被訓練者が、屋内で火災に遭遇した時に避難行動を体験するというものがある。HMD(ヘッドマウントディスプレイ)を装着した被訓練者が三次元仮想空間に没入するタイプの実現を目指しているが、現時点では、現在普及している多くのロールプレイングゲームのように、通常のディスプレイを通して訓練を行なうタイプの実現も並行して進めている。これにより、多様な災害に応じた場合の、避難場所までの最適な（より適切な）移動経路、周囲に多くの避難者が行動している中での最適な（より適切な）行動を、自らの判断で導き出せるよう多くの状況を体験してもらう。

ここでは、実際に日常的に使っている現実の建物において、いろいろな災害の状況を三次元的に見えるように描画・再現する。建物に関する三次元的な構造情報は、三次元 CAD データから取得できる。火災の燃焼モデル、煙の流動、熱伝達等の物理シミュレーションに関しては、NIST が開発保守しているオープンソース・ソフトウェアとして無償配布されている FDS(Fire Dynamics Simulator)などで計算できる（その描画も必要であるが）。これに加えて、被訓練者の周囲のヒトの行動を再現するエージェント・ベースド・シミュレーションも重要である。これには、減災・防災計画立案のために、やはり著者らの研究プロジェクトの一環として構築してきた、避難行動シミュレーションを利用することが可能である。平面的（二次元的な）セル空間モデルもしくはネットワーク空間モデル上で、意思決定主体であるエージェントが事前の知識と、行動中に取得する情報にもとづいて、単位時間毎に（離散的に）なんらかの行動を選択し実行することを繰り返す。今回取り上げる対象は、このシミュレーション中のエージェントの行動を、仮想空間の中で再現する三次元的な可視化である。

この三次元仮想空間を防災訓練の被験者に体験してもらうことで防災意識を高めるには、視覚的なりアリティも重要な要素である。具体的には、3D 表現されたシミュレーション上の避難者の動きを、より人間らしく動かすことが望ましい。そこで問題となるのが、三次元的なアバターの動作が切り替わるタ

イミングで不連続なアニメーションとなってしまふ点である。避難シミュレーションから与えられるエージェントの情報は、時系列データとなっており、単位時間ごとにそのエージェントの位置、行動が記録されている。このシミュレーション結果をもとに三次元空間内のアバターを動かすと、現実ではありえないような動きをしてしまふ。そこで本研究では複数の基本動作を円滑に繋げて表示できる技術を構築する。具体的には「しゃがむ」、「走る」などの避難行動として考えられる基本動作を組み合わせ、一連の動作にしていき、人の行動シミュレーションの可視化におけるリアリティを向上させることを目的とする。そして、避難シミュレーション上の避難者の行動をモデリングツールで作成したアニメーションではなく、実際の人の動作として避難シミュレーションに適用できる動作データを取得し、将来的には、基本動作をカタログ化することで、シミュレーション構築者が使いたい行動を一連の動作として、提供することを旨とする。

2. 補間手法と関連研究

基本動作の補間や合成に関する研究としては、計算量と精度を最適化する LOD 制御手法を提案したもの [1] や合成するモーションの割合に応じて、ブレンド後のモーションに適したボーンウェイトを得る手法 [7] などが挙げられる。本研究での補間は人間の基本動作を連続させることである。つまり与えられるデータとしては、人間の各関節における回転量や三次元位置座標となる。

回転量の補間手法として、球面線形補間 (Spherical Linear Interpolation, Slerp) があげられる。Slerp はオイラー角表現や回転表現では実現不可能な、2 つの 3 次元回転の滑らかな補間を計算することができる。その特徴としては補間係数の変化と回転量の変化に線形の関係がある点があげられる。つまり補間係数を一定速度で変化させると、得られる回転速度も一定になり、得られる回転の軌跡が球面上で一定、つまり最短経路となることも保障されている。

しかしこの Slerp は 2 つの回転量の補間のみ適用可能な技術であり、3 つ以上の回転量の補間には使えない。このために、補間だけでなく動作を合成する場合にも問題が生じる。そこで、本研究では回転量を対数空間にもっていくことで、多段階的に動作を補完する場合や 3 つ以上の動作を合成する場合にも適用可能にしている。回転量を対数空間として扱ったものを、以下、対数クォータニオンと呼ぶ [4]。

3. 対数クォータニオンを用いた 3 次元回転量の補間

3.1. モーションデータ

ここでは筆者らが用いたモーションデータについて記述する。

モーションデータの代表的なファイル形式に BioVision 社が提唱した BVH フォーマットと Acclaim 社が提唱した ASF-AMC フォーマットなどがある。本研究では BVH フォーマットを使用する。

BVH フォーマットは、複数のボーンで構成されるスケルトンの動作を記述する。骨格情報と動作情報から構成される。骨格情報は、各関節の初期オフセット (ボーンの長さや初期方向)、関節の親子関係、および関節自由度の情報を記述している。動作情報は骨格情報部で定義した関節自由度に対応して、ルート位置と回転角の時系列情報を記述している。

BVH ファイルの読み込みと書き出しは図 1 のように行っている。動作 1 から動作 2 へと遷移する場合、まず動作 2 の MOTION 部の時系列データだけを読み込み、動作 1 の MOTION 部の最後のフレームから下の部分に追加する形で書き出している。

この時、合成する上で問題になるのが 2 つの動作データの ROOT の絶対位置である。動作 1 の最後の絶対位置と動作 2 の最初の絶対位置は全く異なる場所であるため、それらを合成した場合、遷移する部分で瞬間的に違う場所に移動してしまう。つまり、一連の動作として 2 つの動作が繋がらないということである。よって、Z-X 平面において、動作 2 の最初の絶対座標を動作 1 の最後の絶対座標に書き換えて、動作 2 の以後の絶対座標の差分を加えていくことで解決している。

3.2. オイラー角の問題とクォータニオン

BVH ファイルの動作情報はオイラー角で記述されている。オイラー角は方向をたった 3 つの数値でパラメータ化することや、これらの数値が全て角度であることなど、方向を示す他の表現にはない明らかな特徴を持つ。しかし、2 つの角を補間する際には非常に難しくなる。所定の方向を表すのに使うこと

```

動作①のBVHファイル
HIERARCHY
ROOT Hips
{
  OFFSET 0 0 0
  CHANNELS 6 Xposition Yposition Zposition
  Zrotation Yrotation Xrotation
  JOINT LeftHip
  {
    :
  }
}

MOTION
Frames: 252
Frame Time: 0.0333334
21.8895 75.0615 -25.58991 -0.2562 -15.7927 -11.2218
22.0365 -2.1757 -0.5207 1.3869 32.3084
:
2.1805 0.005 -66.5925 -31.0408 -45.7613 -0.1 -0.0474
41.5429 -14.2091 -0.3957 -14.4649 1.3956 -0.2599 1
14.1851 72.001 -166.2953 1.9143 -1.6996 -10.5776

動作②のBVHファイル
HIERARCHY
ROOT Hips
{
  OFFSET 0 0 0
  CHANNELS 6 Xposition Yposition
  Zposition Zrotation Yrotation Xrotation
  JOINT LeftHip
  {
    :
  }
}

MOTION
Frames: 246
Frame Time: 0.33334444
-0.33395386 1.9947815 -0.32540894 0.00626899
-0.33395386 1.9947815 -0.32540894 3.24141470
:
-0.33395386 1.9947815 -0.32540894 3.8520014 0
-0.33395386 1.9947815 -0.32540894 3.8999195 1
-0.33395386 1.9947815 -0.32540894 3.948272 2
-0.33395386 1.9947815 -0.32540894 3.9970756 3
    
```

図 1: BVH ファイルの読み込みと書き出し

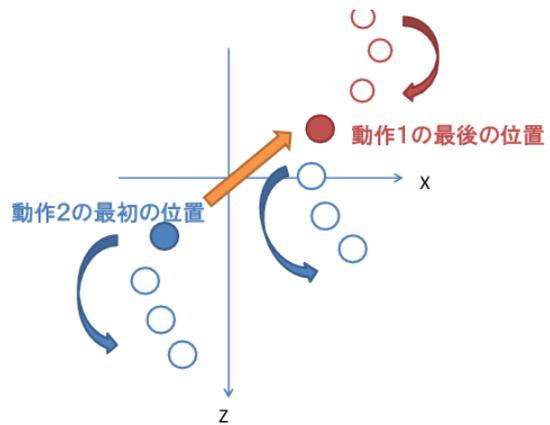


図 2: ROOT ノードの位置調整

のできる異なるオイラー角の 3 つの数値が複数存在していることは、時として、不便な場合がある。また、回転軸の親子関係の真ん中の軸が 90 度回転したときに、一番子の軸と一番親の軸が重なってしまい、3 つある回転の自由度が 2 つになってしまう。このジンバルロックという現象が補間を限りなく難しくしている。そこでこの問題を回避するために、クォータニオンという 3 つの虚数単位を持つ超複素数系の一つを使う [7]。

まず、オイラー角を h (Y 軸周りの回転), p (X 軸周りの回転), b (Z 軸周りの回転) とするとそれぞれに対するクォータニオンは以下ようになる。

$$q_h = [\cos(\frac{h}{2}) \ 0 \ \sin(\frac{h}{2}) \ 0] \tag{1}$$

$$q_p = [\cos(\frac{p}{2}) \ \sin(\frac{p}{2}) \ 0 \ 0] \tag{2}$$

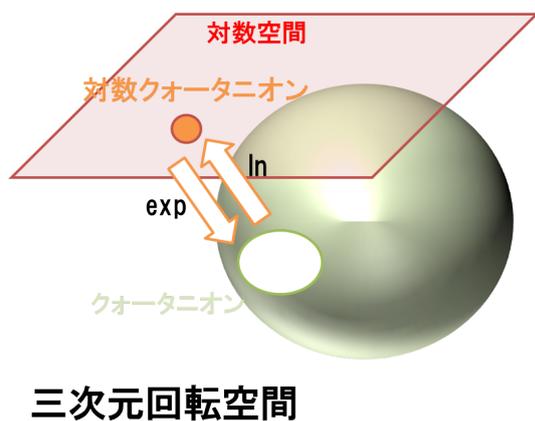
$$q_b = [\cos(\frac{b}{2}) \ 0 \ 0 \ \sin(\frac{b}{2})] \tag{3}$$

これらを正しい順番で連結することによりある関節ノードにおけるクォータニオンを求める。

3.3. 対数クォータニオン補間

対数クォータニオン (文献 [4]: CHapter8 パラメトリックポーズブレンド 向井智彦) は、3 成分はそれぞれ独立なので、回転クォータニオンのノルムを 1 とする必要がなく、各成分を別々に操作できる。また、最大の特徴として、2 つのクォータニオンの合成を、対数空間における加算で近似できる点がある。つまり、図 3 のように、2 つのクォータニオンの対数写像を求めた後に線形結合し、指数計算によってクォータニオンに逆写像すると、2 つの回転クォータニオンの合成回転の近似値が得られる。但し、対数空間での線形結合によるクォータニオンの計算はあくまでも近似であり、正しい値ではない。しかも、その近似誤差は、対数クォータニオンノルムに比例して大きくなるので、可能な限り小さなノルムの対数クォータニオンを用いることが、近似誤差を低減するために重要となる。

対数クォータニオンと補間誤差の関係のイメージを図 4 に示す。対数クォータニオンは三次元の値であるが、ここでは簡単化のために二次元で表現した。まず、2 つのクォータニオンが対数空間の A と B に写像されたとき、点線でしめす線形補間によってクォータニオンの補間を近似できる。しかし、C,D のような原点から離れた対数クォータニオンの計算では近似誤差が大きくなってしまふ。E,F のような近接の対数クォータニオンでも同様である。一方、A,B と G,H のような原点に近い対数クォータニオン同士の補間は、比較的小さな近似誤差となる。つまり原点に近い部分でクォータニオンを補間することが必要となる。



三次元回転空間

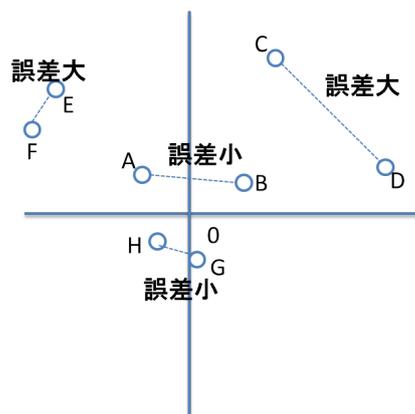


図 4: 対数クォータニオンの補間誤差 (文献 [4])

図 3: 対数クォータニオン (文献 [4])

クォータニオン $r=[r_x r_y r_z r_w]$ が与えられたとき，対数クォータニオンは三次元の値 $u=[u_x u_y u_z]$ となり，補間式は (4) で表される。

$$\sum_i \vec{r}_i = \vec{s} \exp\left(\sum_i \ln(\vec{s}^{-1} \vec{r}_i)\right) \quad (4)$$

これは，対数クォータニオンを算出する際に， \vec{s} が回転の原点に来るように座標変換し，補間の後に元の座標系に戻すという計算を行っている。そして，補間元のクォータニオンの差分が小さくなるように \vec{s} を決定し，対数クォータニオンのノルムを小さくすることで，近似誤差を小さくしている。

4. 実行結果

今回は基本動作のサンプルとして「歩く」動作と「ジャンプする」動作を使用した。2つの基本動作を補間技術によって滑らかに連結した結果を図5に示す。BVH データは，カーネギーメロン大学のモーションデータベース¹のものを使用した。

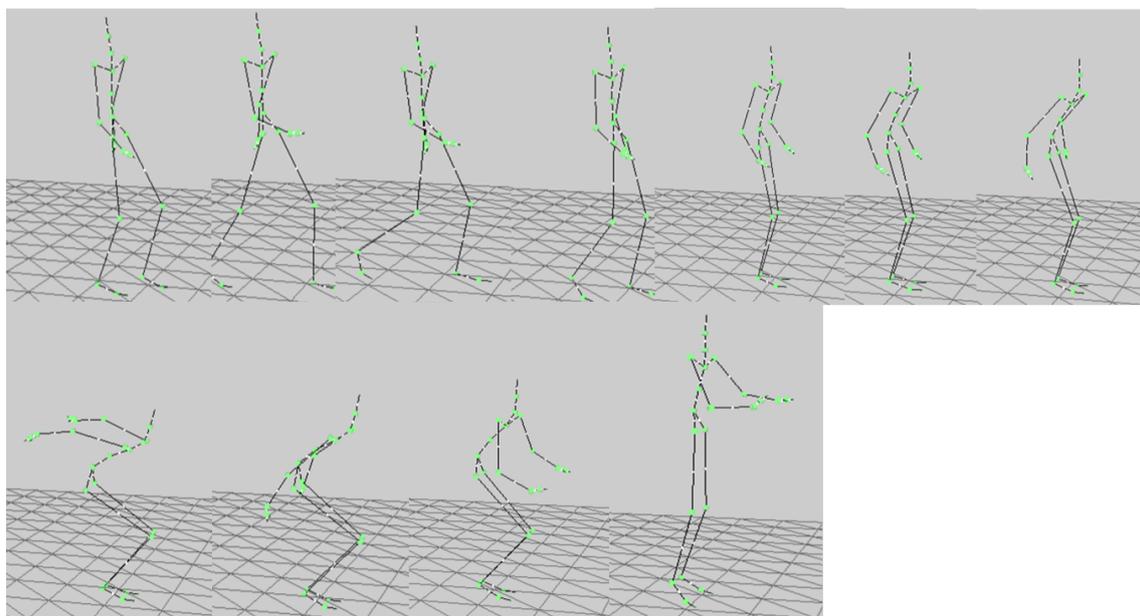


図 5: 「歩く」動作と「ジャンプする」動作の補完による連結例

¹CMU Graphics Lab Motion Capture Database <http://mocap.cs.cmu.edu/>

5. まとめ

本研究では、エージェント・ベースド・シミュレーション手法に基づく避難行動シミュレーションの三次元的なアバターの基本動作を、補間技術を使用して滑らかに連結させることを試みている。3次元の回転量に関しての有力な技術群は、本稿で紹介したもの以外は未だに報告されていない。しかし、動作合成に関してはベイズ理論に基づく非線形回帰法や最大事後確率推定など統計や機械学習の研究が応用されている。引き続き、既存研究の動向を追いつつ、補間や合成の精度を高める技術構築を行っていく。

参考文献

- [1] 向井智彦, 栗山茂: “テンソル疑似を用いた動作補間の多重解像度制御,” 電子情報通信学会, Vol.J91-D No.12, 2008.
- [2] 立川正人, 渡辺大地: “モーシヨンプレンドにおけるボーンウェイトの補間手法,” 情報処理学会, 第 71 回全国大会.
- [3] Jiannyuan Min and YEn-Lin Chen and Jinxiang Chai : “Interactive Generation og Human Animation with Deformable Motion Models,” ACM Transaction on Graphics,29(1), 2009.
- [4] 金森 由博, 井尻 敬, 堀田 一弘, 五十嵐 悠紀, 吉 雄介, 安田 廉, 山本 醜田, 向井 智彦, 梅谷 信行: “Computer Graphics Gems JP 2013/2014 – コンピュータグラフィックスの最前線,” 株式会社ボーンデジタル, 2013.
- [5] Jarmin Bruderlin Lance Wiliams : “IMotion Signal Processing,” SIGGRAPH 95proceedings og the 22nd annual conference on Computer graphics.
- [6] Fletcher Dunn Ian Parberry: “実例で学ぶ ゲーム 3D 数学,” オライリージャパン , 2008.
- [7] 金谷一朗: “3D-CG プログラマーのためのクォータニオン入門,” 工学社 , 2004.