

## RVEにおけるモーションの補間と拘束

1H-10

村尾 高秋, 小出 昭夫

日本アイ・ビー・エム株式会社 東京基礎研究所

## 1 はじめに

従来のアニメーションは動作の生成に重点がおかれていた。そのため、インタラクティブな世界を実現しようとする、リアルタイムで動作を生成することが要求され、非常に膨大なCPU資源が必要とされた。我々の提案するRVE(Reactive Virtual Environment)では視点を変え、個々のオブジェクトの近未来の属性は予測されていると言う世界を前提にしている。例えば、仮想空間内のオブジェクトの動きは、個々の位置を決定するマトリクスの時系列データとして扱う。時系列データを補間することによって任意の時間でのオブジェクトの属性を取り出すことが可能となる。また、オブジェクトの位置を決定するマトリクスに拘束条件を与えることによって、地上を歩く、斜面を歩く、椅子に座るといった他の物体と干渉するようなモーションを高い計算負荷なく扱うことが可能になると考える。本報告ではRVEにおけるオブジェクトの属性の補間と拘束の指針とそれによってどのようなことが実現出来るようになるかについて述べて行く。

## 2 従来のアニメーションの問題点

従来のCGアニメーションの手法は、特定の物理現象をモデルを使ってシミュレートすると言う部分に主眼が置かれていた。このような手法でインタラクティブアニメーションを実現しようとする

- モーションの生成に膨大なCPU資源が必要になる。

- 現象によって必要とする手法、モデルが異なる

と言った問題が発生する。しかしながら、視点を変えると現在モーションキャプチャによって複雑な人間の動きを簡単に手に入れることが出来る。また、キーフレームアニメーション、インバースキネマティクスを利用したモーション編集などの手法によって作成された資産が

A Constrained motion of Reactive Virtual Environment  
Takaaki Murao  
Tokyo Research Laboratory, IBM Japan Ltd.

存在する。これらの資産をライブラリとしてインタラクティブな世界で活用して行くことが我々の主眼である。

## 3 RVEのモーション

RVEで扱うモーションは歩く、座る、走ると言った個々の動作を表す時系列データである。

## 3.1 モーションエンジン

RVEでは個々のモーションは関節の時系列データとして scheduled field に格納されてライブラリ化されている。モーションエンジンはこれらのデータから任意の時間の各関節の位置を生成する働きをする。

モーションエンジンは関節モデルの各関節の回転、位置情報を持つフィールドに接続して使う。

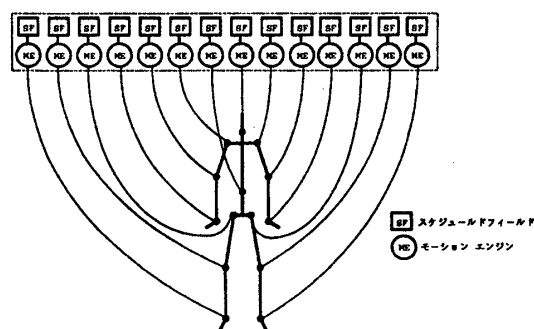


図1: モーションエンジン

モーションデータとして速度、加速度と言った値を持つことにより、自由落下、物体の反射といった簡単な物理現象も表現出来る。

モーションエンジンはモーションデータのレートとは関係なく任意の時刻の姿勢を簡単な補間によって提供する補間機能を持っている。こうすることによって任意のレートでの表示、インタラクションに対応可能になる。また、こうすることによってモーションデータのサンプルレートも任意で良くなる。

## 3.2 Behavior

RVEでは個々のモーシオンが時系列に並べる、あるいはインタラクションに対するモーシオンの切替を記述することによってオブジェクトのbehaviorが表現される。

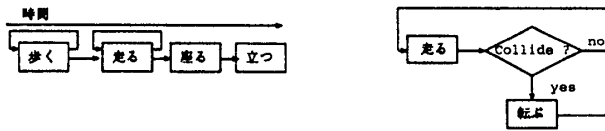


図 2: RVE の Behavior

このようにして RVE 内では各オブジェクトが自律的に動く世界が提供される。

## 4 拘束

RVE はオブジェクトの持つ属性を拘束する機能を提供する。こうすることによってユーザーはモーシオンに拘束を付加することによって Behaviour を記述出来る。すなわち、イベントコールバックシステムを使うことなくある程度の Behaviour を定義することが可能となる。

### 4.1 単純な拘束

RVE ではフィールドに範囲を指定出来る。範囲は最大値、最小値、平面、2次形式で表すことの出来る曲面で囲まれる領域によって指定する。(フィールドはベクター型である場合もある)

フィールドに与えられる値が境界を越えた場合、次のような動作をする。

- 境界に達した時点で再びフィールドに与えられる値が指定された範囲に入るまで値は固定される。
- ユーザーの指定した点とフィールドに与えられた点を結ぶ直線と境界との交点を値とする。

またフィールドに与えられた値が境界を越えた場合、イベントを発生して、ユーザーが独自に対応することも可能である。

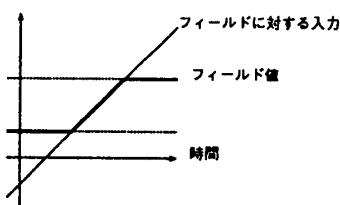


図 3: 単純な拘束

これを利用することによってシーングラフにおいて親子関係にあるオブジェクトの位置関係の拘束が可能となる。

### 4.2 より高度な拘束

前出の拘束法では平地、傾斜地を歩く人や、高さの違った椅子に座る人と言った動作を表現することは出来ない。こう言った Behavior に必要な拘束に共通するのは、拘束条件を与えるマトリクスと、拘束を受けるマトリクスが同一でないということである。この拘束は次のように実現出来る。まず、任意のオブジェクト間の相対位置を取得するマトリクスを定義し、拘束条件を与える。次に、値が境界に達した場合に拘束するマトリクスと拘束法を指定する。この方法では拘束条件を与えるマトリクスと拘束するマトリクスが違うため、任意の拘束法を許すことは出来ない。しかしながら、ワールド座標系の座標軸方向の並行移動のみに限定すると言った使い方でも有用な手法であると我々は考える。

## 5 今後の課題

今回は内容をモーシオンに限定したが拘束による Behavior の定義はモーシオンだけでなくオブジェクトの他の属性に対しても有効であると我々は考えている。また、複数の拘束が衝突した場合の処理についても拘束可能なものを限定すると言った手法で少ない負荷でより大きな効果が得られる世界を目指して行く。

## 6 最後に

今回の報告では RVE におけるモーシオンの取り扱いの概要を説明し、これに拘束を加えることによって behavior を記述出来ることを示した。

## 7 謝辞

最後になりましたが本論文を書くにあたって助言を頂いた 宮沢、吉田、谷口、小出 各氏に感謝の意を表します。

## 参考文献

- [1] 宮沢ら: 「Reactive Virtual Environment(RVE) システムの概要」, 情報処理学会第 52 回全国大会, 1H-08, 1996.
- [2] 吉田ら: 「RVE システムでの仮想環境記述言語」, 情報処理学会第 52 回全国大会, 1H-09 1996.