

# 自然言語からの仮想人間の 複合動作アニメーションの生成

國分 剛    森脇 晃    寺崎 卓也    尾下 真樹

九州工業大学 情報工学部 システム創生情報工学科 (旧・制御システム工学科)  
〒820-8502 福岡県飯塚市川津 680-4  
E-mail: kokusan@cg.ces.kyutech.ac.jp

我々は、オブジェクト指向データベースを用いて動作データを管理し、入力された自然言語による文章からアニメーションを生成するシステムを開発している。本論文では、今回行ったシステムの改良として「ーしながら～する。」や「ーして～する。」といった、複数の動作を並列・連続して行うような入力文に対応した手法を提案する。本手法では、自然言語入力文中の動作間の時間関係を表す要素に注目した意味解析を行い、各動作をどのような組み合わせ・順序で実行するという情報を持つ動作時間テーブルを生成する。その動作時間テーブルにもとづき、前後の動作の類似姿勢の足の状態に注目した動作ブレンディングを行うことで、自然なアニメーションを生成する。

キーワード： キャラクタアニメーション、複合動作、意味解析、動作ブレンディング

## Generating Animation of Composite Motions from Natural Language Text

Tsuyoshi KOKUBU    Akira MORIWAKI    Takuya TERASAKI    Masaki OSHITA

Department of Control Engineering and Science,  
Faculty of Computer Science and System Engineering, Kyushu Institute of Technology  
680-4 Kawazu, Iizuka-Shi, Fukuoka, 820-8502, Japan  
E-mail: kokusan@cg.ces.kyutech.ac.jp

We have been developing an animation system that converts a natural-language input text to an animation. In this paper, we present new methods to handle input texts that include a series of serial and parallel motions. Our system retrieves the temporal relationship between verbs in an input text and generates a time table which represents the execution-schedule of corresponding motions. A continuous animation is then generated based on the time table by using a motion blending technique.

Keywords: Character Animation, Composite Motions, Semantic Interpretation, Motion Blending.

## 1. はじめに

近年、映画やTV等における、3DCGアニメーションの利用が普及化している。3DCGアニメーションでは、キャラクターの動きの元となる動作データが使われる。しかし、映画等で必要となる動作のデータは膨大な数となり、動作の作成作業には多くの手間隙がかかる。そこで、一度使用した動作データを再利用したいという要求がある。しかし、多種多様な動作データを、従来の階層型ファイルシステム、単純なテーブル方式のリレーショナルデータベース等で管理することは困難である。

また、最近では、アニメーションの作成ツールの高度化、多機能化にともない、ツールを使うために必要な専門知識も多く、一般人には扱いにくいものとなっている。この問題に対し、誰でも手軽に扱える作成ツールの要求がある。

本研究室では、大量の動作データを管理し、入力文より動作データを検索することで、アニメーションの生成を行うシステムの開発を行っている[7]。これまでに、オブジェクト指向の継承を使った、キャラクターデータの階層的な管理手法を提案した。また、自然言語学の格フレームという概念を基に、動作の特徴的な情報を持つ動作フレームを定義した。動作フレームを各動作データに対応させることにより、大量の動作データを効率的に管理する手法を提案した。これらの手法を用いてキャラクター、動作のデータベースを構築し、動作データベースを用いて自然言語の文章を入力とした、アニメーション生成システムの開発を行った。本システムでは、自然言語で記述された文章を与えると、文章にあった動作データ、キャラクターをデータベースより検索する。検索されたデータを用いて自動的にアニメーションの生成を行うことを可能としている。

しかし、これまでのシステムにおいては複数の動作を並列に行う複合動作に対応していなかった。また、連続した動作の繋ぎ目において姿勢がワープしてしまうという問題があった。これらを解決するため、本研究では動作と時間・依存関係を表す「動作時間テーブル」と、動作間に推移動作を生成し複合動作を生成する「動作ブレンディング」を取り入れたシステムを提案する。

第2章ではこれまでのシステムについて述べる。また、第3章では本研究の概略を述べ、第4章、第5章でその内容について詳しく述

べる。第6章で実験と考察を行い、第7章で本論文のまとめを行う。

## 2. システム構成

図1に全体のシステム構成と全体の処理の流れを示す。本システムでは、入力文を与えると、文に応じたアニメーションを生成する。まず、入力文章の自然言語処理を行い、動作データ検索に必要なキー情報を抽出する。データベース内の動作データの候補に対して、キー情報と、動作データに付随した動作フレームを比較、評価することにより、最適な動作データを検索する。出力された動作データは、キー情報の示す関連キャラクター、アイテムとまとめて、イベントデータとして出力される。最終的に、このイベントデータよりアニメーションが生成される。

本システムでは、キャラクター、動作をデータベースで管理する手法として、オブジェクト指向を用いた手法を提案している。次に、動作データを検索する手法として、自然言語処理の格フレームを利用した。また、入力文の自然言語解析処理について、ト書き風の文章に特化した手法を用いる。本章では、これら3つの手法について、それぞれ述べる。

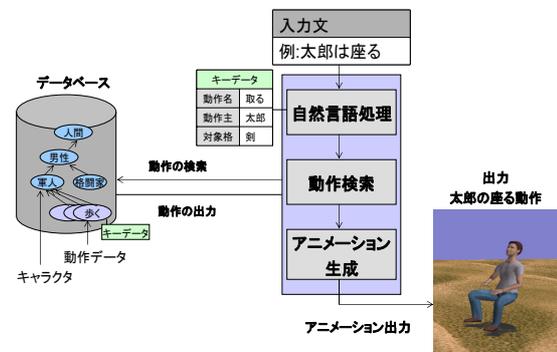


図1 システムの構成図

### 2.1 継承によるキャラクター、動作の管理手法

本研究では、オブジェクト指向によるキャラクター、動作データの管理手法を提案している。本システムでは、各キャラクターをそれぞれ一つのオブジェクトとして扱い、継承を用いて階層的に管理する。各キャラクターは、動作データを自身のメソッドとして扱う。動作データをオーバーライドさせることで、動作データを上位、下位のキャラクター間で共有することが可能となり、キャラクターの追加が容易となる。

## 2.2 格フレームによる動作データ検索手法

動作検索において、大量で多種多様な動作の中から、名前等の情報より最適な動作を検索するのは非常に困難である。本研究では、動作検索において、動作が持つ意味的な情報を、検索条件として与えることでこの問題を解決する。

本システムでは、格フレームの概念を基に、動作検索のキー情報として、動作の意味的な情報を表す動作フレームを定義する。動作フレームは各動作データに対して、前もって付随情報として与えられる。格フレームとは、フィルモア[1]によって提案された、文を意味的な構造として表現するための手段である。格フレームは各動詞に対して取り得る深層格と、それに関する制約を記述する。深層格は、文章の表層の中で伴う格助詞によって区別されるもの(これを表層格という)とは違い、文章中の動詞、形容詞との意味的な関係を捉えた格である。

表 1 格フレームの深層格

動作主	与えられた動作を引き起こすもの
道具格	道具としての役割
対象格	移動や変化する対象物
源泉格	対象物の移動における起点
目標格	対象物の移動における終点
経験者格	ある心理事象を体験するもの
場所格	ある出来事が起こる場所および位置
時間格	ある出来事が起こる時間

表 2 動作フレームの評価項目

	評価項目	評価項目の説明
	動作名	実行する動作の名前
	キャラクタ	動作を実行するキャラクタ
オブジェクト条件	道具格の直接指定	データベース内のアイテム
	道具格のサイズ	アイテムの幅、高さ、奥行き
	道具格の重さ	アイテムの重さ
	対象格の直接指定	データベース内のオブジェクト
	対象格のサイズ	オブジェクトの幅、高さ、奥行き
姿勢条件	対象格の重さ	オブジェクトの重さ
	初期姿勢	キャラクタの動作開始時姿勢
	位置、向きの変化量	動作によるキャラクタの位置、向きの変化量
	エフェクタの源泉位置	キャラクタのエフェクタの源泉位置
	エフェクタの目標位置	キャラクタのエフェクタの目標位置
	ファジィ測度	ファジィ測度 (0~1 の実数)

フィルモアの定義した深層格は8つある(表1)。本研究では、動作データを表現するために、8つの深層格から、経験者格、場所格、時間格の3つを除いた5つの深層格を選び、本研究で使える形に再定義した(表2)。

本システムでは、自然言語で記述された入力文章の自然言語処理を行い、解析結果として深層格に基づくキー情報を得る。このキー情報と、動作フレームとの比較、評価を行い、最適な動作データを検索する。

評価方法として、動作フレーム、キー情報の持つ深層格ごとに項目をわけて比較を行い、最終的に、以下のような、各評価項目の重みつき2乗総和を求める。この評価が最も高い動作フレームに対応した動作データを最適な動作として出力する。

$E_0, E_1, E_2, \dots, E_{10}$  各評価項目の評価点

$w_0, w_1, w_2, \dots, w_n$  各評価項目の比重値

$E$  総合評価点

$n$  評価項目の数

$$E = \sum_{k=0}^n w_k E_k^2 = w_0 E_0^2 + w_1 E_1^2 + w_2 E_2^2 + \dots + w_n E_n^2$$

## 2.3 アニメーションに特化した自然言語解析

通常、自然言語処理では、形態素解析、構文解析、意味解析、文脈解析の4段階の処理を行う[4]。

これに対し、本研究では、形態素解析、構文解析に、既存のシステムとして奈良先端科学技術大学の「南瓜」[6]を使い、意味解析に、本システムに特化した独自の手法を用いている。文脈解析については、本システムでは対応が困難であるため、現在はあまり考慮していない。

一般の意味解析では、本研究と同様に、主に、格フレームを使った解析手法が用いられる。しかし、あらゆる文章に対応させようとすると、文中の単語の意味情報を得るために、膨大な量の辞書が必要となる。また、この手法では、表層格という格助詞に応じて決められる格より、格フレームを用いて深層格を決定する。しかし、表層格から深層格は一様に決めることは一般には難しい。

このような問題に対して、本研究ではまず、扱う文をト書き風の簡単な文に注目し、主にアニメーションに関わる動詞のみを考慮する。これより、扱う動詞は、表3のように限定される。

表 3 本システムで対応する動詞の種類

動詞の種類		例	対応
動作動詞	外的動作動詞	「走る」「食べる」	○
	内的動作動詞	「思う」「考える」	×
状態動詞		「ある」「知っている」「含む」	×
変化動詞		「着く」「止まる」	△

また、本システムで扱う入力文中の自立語は、ほとんどが、アニメーションシーンにおけるキャラクタやアイテム等を指す。シーン情報を使用することにより、大規模な名詞の辞書を用いることなく、文節内の自立語に意味情報を付加させることを可能とする。これらより、単純なルールを用いて、表層格より深層格を決定することが可能となると考える。

### 3. 複合動作の生成

これまでのシステムでは、複数の動作を再生する際に、動作データを単純に並べて再生しており、動作を並列に再生することができなかった。また、動作間の姿勢の差を考慮していなかったため、動作が変わる際に姿勢がワープしてしまうという問題があった。本研究では、この問題を解決するためにシステムを改良した。

図 2 に改良後のシステム構成を示す。自然言語解析部において、新たに時間関係に注目した意味解析を行い、抽出された時間制約条件を出力する。この時間制約条件と検索された動作データから、動作時間テーブルを作成する。それに基づいて複合動作のアニメーションを生成するという流れになっている。

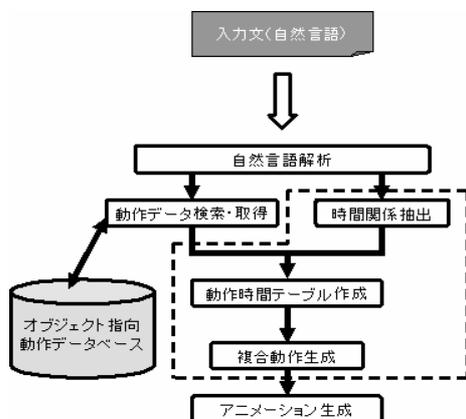


図 2 改良後のシステム構成

### 3.1 複合動作

本研究における複合動作とは、複数の動作が時間的に重なりを持つものを示す。また、複合動作アニメーションを生成する際に、動作間をなめらかに接続することも同時に行っている。図 3 にその例を示す。

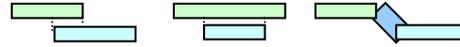


図 3 複合動作の例

### 3.2 動作時間テーブル

本研究では、複合動作の生成を実現するために、動作データと動作の時間関係を持つ動作時間テーブルを定義した。動作時間テーブルの各要素は、動作データと前の要素からの時間関係、アニメーション生成の際の動作のブレンド情報を持つ。例を 4.3 節の図 5 で示す。

次の 4 章では自然言語入力から動作時間テーブルの生成について、5 章では複合動作アニメーション生成のための動作ブレンド手法について説明する。

## 4. 自然言語からの複合動作スケジューリング

本章では、自然言語入力から動作時間テーブルを生成するための方法として、自然言語入力からの時間制約条件抽出と、時間制約条件からの動作時間テーブル生成について説明する。

自然言語からのアニメーション生成として馬場らの研究[3]がある。このシステムではキャラクタの移動に注目し、動作の時間関係とオブジェクトの配置関係についての条件を設定しているが、本研究では 1 人のキャラクタが連続して動作を行う部分に注目して研究を行った。

### 4.1 時間制約条件

入力文から動作の時間関係を設定する際に、その条件を表すと考えられる要素が複数ある場合が考えられる。動作間の適切な関係を設定するために、それらの要素ごとに時間制約条件として抽出する。図 4 に時間制約条件の構造を示す。

#### 時間制約条件の構造

- ・ 関連する動作：どの動作間の条件か（動作1、動作2）
- ・ 時間制約条件：動作同士の時間関係（順番に再生、開始時間について、終了時間について、同時に再生）
- ・ 時間条件：固定値、最大値固定、最小値固定、範囲のどれか

図 4 時間制約条件

### 4.2 時間制約条件抽出ルール

本研究では、入力文から時間制約条件を抽出するルールを設定した。自然言語入力からすべての時間関係を明示的に表すのは不可能なので、相対時間の数値を適当に与えてアニメーション生成部に一部判断を委ねている。また、現時点では入力文における動作の関係を1度に2動作までに限定することによって、自然言語解析部分では単純なルールのみを採用している。以下にそのルールを示す。

動作となる動詞を持つ文節に次の要素が含まれていた場合、その動作を動作1、動作1の文節の係り先にある動作を動作2として、この2動作間の条件を以下のように設定する。

(ア) 接続助詞「て」、読点

- ・ 動作を順番に再生し、負の値の相対時間

(イ) 句点

- ・ 動作を順番に再生し、正の値の相対時間

(ウ) 接続助詞「ながら」「つつ」

- ・ 二つの動作を同時に行う

現時点では、動作の含まれる文節についてのみ注目しているため、条件の競合は起こらないようになっている。しかし、今後の課題として、「～の前に」や「しばらくして～」といった動作の時間に関する副詞による条件の抽出を考えており、競合する条件を、優先度を用いて判別を行う予定である。

### 4.3 動作時間テーブルの生成

自然言語入力から抽出した時間制約条件と、既存システムの解析結果から検索した動作データを用いて動作時間テーブルを生成する。動作テーブルの各要素は、4.1節で示した時間制約条件によって決定する。そのルールを以下に示す。ブレンド方法については、5.2節で示し

ている。

- ・ 動作を順番に再生し、負の値の相対時間  
ブレンド方法：Tradition  
時間指定：最低値～0までの範囲
- ・ 動作を順番に再生し、正の値の相対時間  
ブレンド方法：Connection  
時間指定：+の固定値
- ・ 二つの動作を同時に行う  
ブレンド方法：Composition、  
時間指定：範囲指定

入力からの例を図 5に示す。

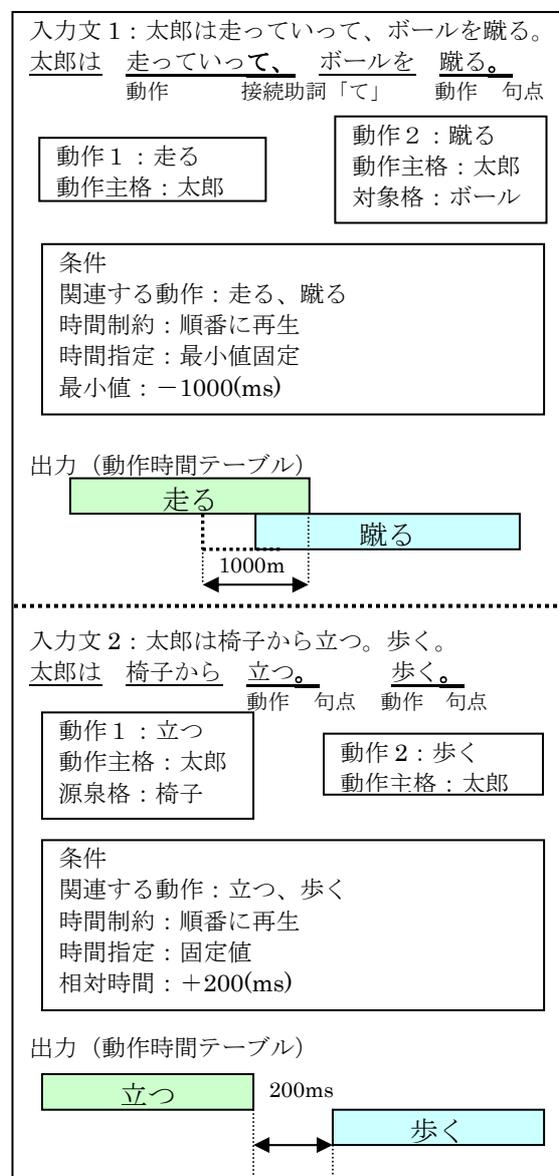


図 5 入力から動作時間テーブル生成までの例

## 5. 動作時間テーブル複合動作アニメーション生成

### 5.1 全体の流れ

本研究における複合動作アニメーションは、連結対象となる複数の動作において、類似した姿勢を検索し、その姿勢間を接続する推移動作を生成することで実現する。本手法では、まず、自然言語解析結果から作成された動作時間テーブルと動作データを入力として受け取る。次に、動作時間テーブルに付随しているテーブル同士の時間・依存関係の情報に基づき、連結対象動作から類似した姿勢を検索する。ここで類似姿勢の各動作データ内の時間に基づき、動作時間テーブルを再構成する。そして、再構成された動作時間テーブルに基づき、動作間に推移動作を生成し、滑らかな複合動作を生成する。

### 5.2 ブレンディング手法の分類

ブレンディング手法は数多く研究されているが、手法の名称が統一されていない。そのため本研究では、複合動作を生成する際に必要となるブレンディング手法を3つに分類し名称をつけた。本論文では“Connection”と“Transition”について述べる。

- Connection  
動作間が一定時間空く場合の手法
- Transition  
第1動作の一部と第2動作の一部が重なる場合の手法
- Composition  
ある動作の一部を別の動作に組み替える手法

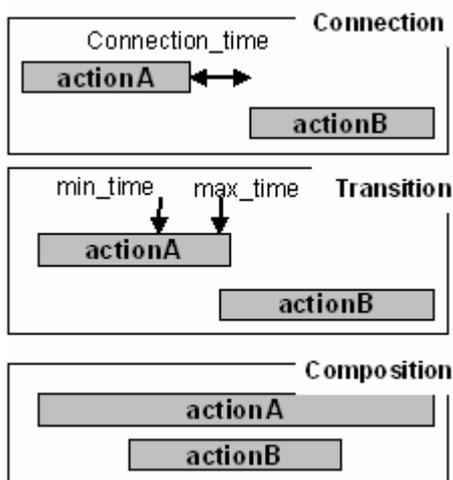


図6 ブレンディング手法の分類

### 5.3 ブレンディング手法

本研究では、検索した類似姿勢を用いて動作間に推移動作を生成し、その類似姿勢の動作データ内の時間と推移時間の長さの設定次第でセグメントの開始時間を前後にずらす(図7)。これらの操作によって“動作時間テーブル”を再構成する。本手法では、入力として得た“動作時間テーブル”を元に動作マッチング[5]により類似姿勢を取得する。“Connection”と“Transition”の違いは、姿勢間のブレンドに用いる姿勢の検索方法にある。“Connection”の場合、第1動作の終了姿勢を取得し、そしてその姿勢と類似する姿勢を第2動作から検索し、取得した2姿勢を類似姿勢とする。“Transition”の場合、動作時間テーブルは第2動作が第1動作と重なる範囲の最小値と最大値を持つ。そしてその最小値から最大値の範囲をひとつの動作と考える。また第2動作において、先頭から最小値と最大値の時間範囲の分をひとつの動作と考え、以上で得られた2つの動作を比較し、最も姿勢間の距離が近いものを類似姿勢とする。

次に、取得した類似姿勢を用いて滑らかな推移動作を図でセグメント2に当たる部分に生成する。本研究のブレンディングは、取得した2姿勢の関節角度を時間経過に伴いブレンド比率を変化させていくことで混ぜ合わせることである。

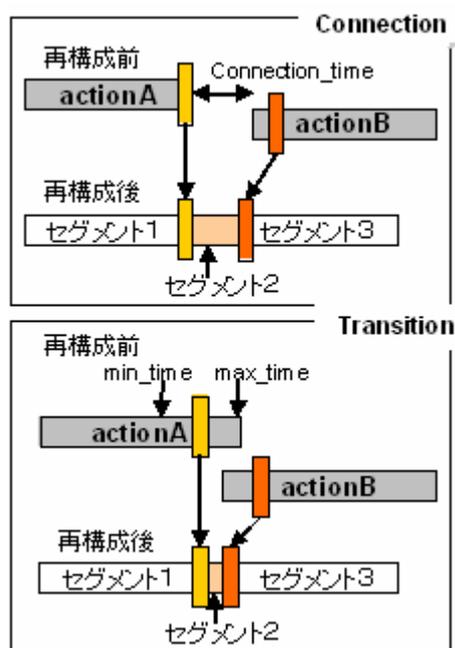


図7 姿勢取得から動作時間テーブルの再構成例



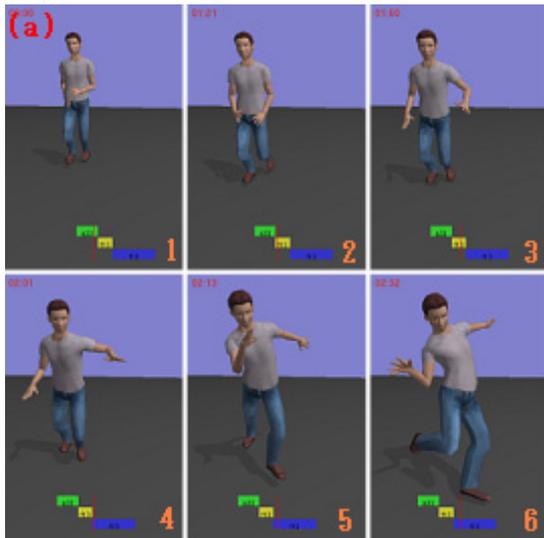


図 9 「走って、ボールを蹴る」

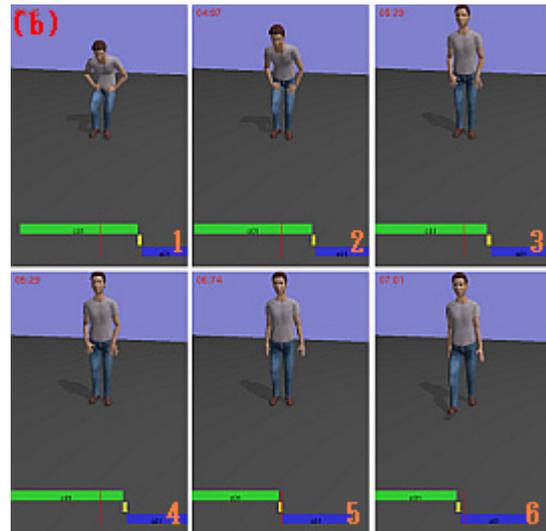


図 10 「椅子から立つ。歩き出す」

生成されてしまうものがあった。これは、類似姿勢取得アルゴリズムにおいて、類似姿勢と判断する基準に姿勢同士の距離しか考慮されていないために起こる問題だと考えられる。入力文1の Transition の場合は、第1動作と第2動作の動作スピードが重要になってくる。つまりその動作スピードに応じて、推移動作の長さも変化させなくてはならない。今回は手動で0.3sに設定した結果、自然なアニメーションとなった。入力文2の Connection の例では、第1動作と第2動作の間に0.2sの時間を設け、アニメーションを生成した。しかし、姿勢2において片足が地面から浮いているために、第1動作と第2動作の間に長い時間を設けてしまうと、非常にゆっくりと右足を上げる様な不自然なアニメーションが生成されてしまった。動作間に長い時間を設ける場合は、姿勢1から姿勢2に直接推移するような動作を生成せず、間に決まった姿勢を挿入し、それを中継して姿勢2に滑らかに推移する手法も考えられる。

## 7. まとめ

本論文では、オブジェクト指向データベースを用いた自然言語からのアニメーション生成システムの拡張として、時間関係に注目した意味解析による動作時間テーブルを用いた動作スケジューリング手法と、複合動作アニメーション生成のための動作ブレンド手法を提案した。

今後の課題として、自然言語処理部分における時間を表す副詞に対応した条件の抽出と、動

作ブレンド部分における“Composition”に対応した手法の開発、類似姿勢アルゴリズムの改良による、多様な状況の再現が挙げられる。

## 謝辞

本研究の一部は文部科学省・科学研究費(課題番号16200005)による支援を受けた。

## 参考文献

- [1] チャールズ J. フィルモア-著, 田中春美, 船城道雄-訳, 「格文法の定理」, 三省堂, 1975
- [2] Charles Rose, Brian Guenter, Bobby Bodenheimer, Michael F. Cohen, “Efficient Generation of Motion Transitions using Spacetime Constraints”, SIGGRAPH 1996
- [3] Hiromi BABA, Tsukasa NOMA, Naoyuki OKADA, “Visualization of Temporal and Spatial Information in Natural Language Descriptions”, IEICE Transaction on Information and Systems, Vol. E79-D, No. 5, pp. 591-599, May 1996.
- [4] 乾健太郎, 白井清昭, “例文を使って文を解析しよう”, 情報処理学会論文誌, 41(7), pp.763-768, July 2000
- [5] Lucas Kovar, Michael Gleicher, “Automated Extraction and Parameterization of Motions in Large Data Sets”, SIGGRAPH 2004
- [6] 奈良先端科学技術大学院大学自然言語処理学講座, 日本語係り受け解析機「南瓜」, <http://cl.aist-nara.ac.jp/>
- [7] 寺崎卓也, 上川千絵, 尾下真樹, “オブジェクト指向による動作データの管理手法と自然言語からのアニメーション生成システム”, Visual Computing 2004