

# 動きの汎化によるロボット操作システムの提案

池田憲弘

2010年1月xx日

# 目次

|       |   |    |
|-------|---|----|
| 第1章   | はじめに                                    | 3  |
| 1.1   | 背景                                      | 3  |
| 1.1.1 | 概要                                      | 3  |
| 1.1.2 | ロボットの動きを設計する方法                          | 3  |
| 1.1.3 | 学習を用いて状況に適した行動を行う方法                     | 3  |
| 1.1.4 | ユーザがロボットに対して教示を行う方法                     | 4  |
| 1.1.5 | 操作することによりロボットを動かす方法                     | 4  |
| 1.1.6 | まとめ                                     | 4  |
| 1.2   | 従来研究                                    | 5  |
| 1.3   | 本研究の着眼点                                 | 5  |
| 1.4   | 目的                                      | 6  |
| 第2章   | アプローチ                                   | 8  |
| 2.1   | 概要                                      | 8  |
| 2.2   | 教示と汎化                                   | 8  |
| 2.3   | 再生と操作                                   | 9  |
| 2.4   | 状況に応じた動きの選択                             | 10 |
| 第3章   | 「ロボットの動きに関する新しい表現方法」を用いた動きの汎化と汎化した動きの操作 | 12 |
| 3.1   | 概要                                      | 12 |
| 3.2   | 本論文で用いる空間の説明                            | 12 |
| 3.2.1 | 本論文で用いる空間の概要                            | 12 |
| 3.2.2 | 状態空間                                    | 13 |
| 3.2.3 | 動作空間                                    | 14 |
| 3.2.4 | 頻度空間                                    | 15 |
| 3.3   | 頻度空間を用いた動きの汎化                           | 15 |
| 3.3.1 | 概要                                      | 15 |
| 3.3.2 | 頻度空間の更新                                 | 16 |
| 3.3.3 | 頻度空間の更新と動きの汎化の関連                        | 17 |
| 3.4   | 頻度空間を用いた操作による動きの生成                      | 18 |
| 3.4.1 | 概要                                      | 18 |
| 3.4.2 | 動きの生成                                   | 18 |
| 3.4.3 | 操作による動きの変更                              | 19 |
| 3.5   | 頻度空間を用いた状況に応じた動きの選択                     | 19 |
| 3.5.1 | 概要                                      | 19 |
| 3.5.2 | 教示段階                                    | 19 |
| 3.5.3 | 再生段階                                    | 20 |
| 3.6   | ユーザによる操作と状況に応じた動きの選択                    | 20 |
| 3.7   | 本論文で作成したシステム                            | 20 |

|                                |           |
|--------------------------------|-----------|
| <b>第4章 実験</b>                  | <b>22</b> |
| 4.1 概要                         | 22        |
| 4.2 使用したロボットと操作デバイス            | 22        |
| 4.2.1 使用したロボット                 | 22        |
| 4.2.2 使用した操作デバイス               | 25        |
| 4.3 センサ, 操作デバイスを用いない実験         | 26        |
| 4.3.1 実験の説明                    | 26        |
| 4.3.2 実験の設定                    | 27        |
| 4.3.3 結果                       | 29        |
| 4.3.4 考察                       | 35        |
| 4.4 操作デバイスのみ用いる実験              | 35        |
| 4.4.1 実験の説明                    | 35        |
| 4.4.2 実験の設定                    | 36        |
| 4.4.3 結果                       | 40        |
| 4.4.4 考察                       | 46        |
| 4.5 センサのみ用いる実験                 | 46        |
| 4.5.1 実験の説明                    | 46        |
| 4.5.2 実験の設定                    | 47        |
| 4.5.3 結果                       | 51        |
| 4.5.4 考察                       | 57        |
| 4.6 センサ, 操作デバイスを用いる実験          | 57        |
| 4.6.1 実験の説明                    | 57        |
| 4.6.2 実験の設定                    | 59        |
| 4.6.3 結果                       | 63        |
| 4.6.4 考察                       | 71        |
| <b>第5章 おわりに</b>                | <b>72</b> |
| 5.1 本論文のまとめ                    | 72        |
| 5.2 今後の課題                      | 72        |
| 5.2.1 概要                       | 72        |
| 5.2.2 高次元空間の利用によるデータ量の増大に関する問題 | 72        |
| 5.2.3 教示・操作の容易さに関する評価実験を行っていない | 73        |
| 5.2.4 複数関節・複数入力を用いた実験を行っていない問題 | 73        |
| 謝辞                             | 74        |
| 参考文献                           | 74        |
| 研究業績                           | 77        |

# 第1章 はじめに

## 1.1 背景

### 1.1.1 概要

近年になり、家庭内や公共施設などの実環境下で作業を行うロボットが増え始めている [1]-[4]。例えば、警備ロボットや掃除ロボット、搬送ロボットなどが例として挙げられる [5][6]。

このようなロボットは、人がいる環境下で作業を行うことも多い。人がいる実環境下では、例えば「人がロボットの目の前を横切る」「信号機が青から赤に変わる」「目の前に物を積まれる」など、環境がめまぐるしく変化する。めまぐるしく変化する環境下で作業を行うためには多数のセンサが必要になる場合が多い。例として警備ロボットを挙げる。警備ロボットに必要なセンサとして、人や物との接触を感知するための圧力センサ、人の顔を認識するためのカメラ、物音を感知するためのマイクなどが挙げられる。

また、このようなロボットは、1つの簡単な作業のみではなく、複数の複雑な作業を求められることも多い。複数の複雑な作業を行うためには、ロボットが多数の自由度を持つ必要がある。例として災害救助用ロボット実環境下でさまざまな作業を行うロボットは、複雑なタスクや環境に適応するために多数の自由度やセンサが必要になる場合が多い。このようなロボットにタスクを行わせる方法として

- ロボットの動きを設計する方法 [7][8]
- 学習を用いて状況に適した行動を行う方法 [9]-[11]
- ユーザがロボットに対して教示を行う方法 [12][13]
- 操作することによりロボットを動かす方法 [14][15]

などが考えられる。

### 1.1.2 ロボットの動きを設計する方法

まず「ロボット設計者がロボットの動きを設計する方法」を考える。この方法は、主に産業用ロボットに用いられている。ロボットが状況に応じて適した行動を取ることが出来るように設計者がプログラミングを行う。ロボットはプログラムを元に自動的に動作する(図 1.1)。しかし、設計者は多くの状況に対応するために、多数の自由度やセンサを駆使し、動きを設計しなければならない。この方法を実環境下で動くロボットに適応することは非常に難しく、設計者の負担も大きい。また、ロボット設計者の想定出来なかった未知の環境では、正常に動作することが出来ないと考えられる。

### 1.1.3 学習を用いて状況に適した行動を行う方法

次に「学習を用いて状況に適した行動を行う方法」について考える。現在は機械学習などを用いた研究が進められている。この方法を用いることにより、設計者の負担の軽減することが出来る。また、未知の環境下でのロボットの正常動作も期待出来る。さらに、各状況に

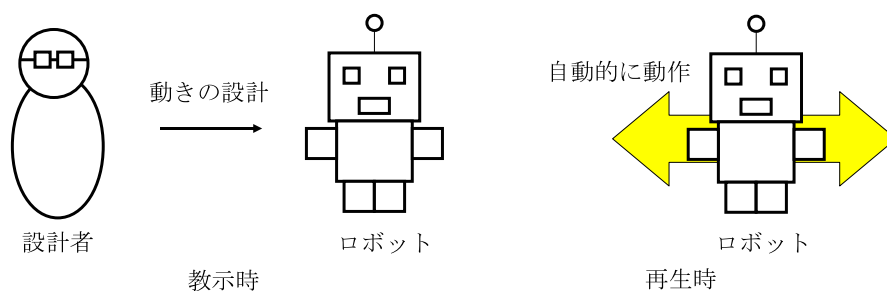


図 1.1: 設計者による動きの設計

対してロボットが能動的に行動を決めることも出来る．しかし，複雑な環境下で多自由度，多センサロボットに適用するまでには至っていない．

#### 1.1.4 ユーザがロボットに対して教示を行う方法

「ユーザがロボットに対して教示を行う方法」について考える．この方法は比較的古くから考えられており，少数生産を行う産業用ロボットなどに用いられている．この方法は，ユーザがロボットに対して動きを教え（教示），ロボットがその動きを再生することで実現される（図 1.2）．設計者が動きをプログラミングする必要がないため，設計者の負担を軽減することが出来る．しかし，教示していない動きについては行うことが出来ない．また，未知の状況では改めて教示し直す必要がある．

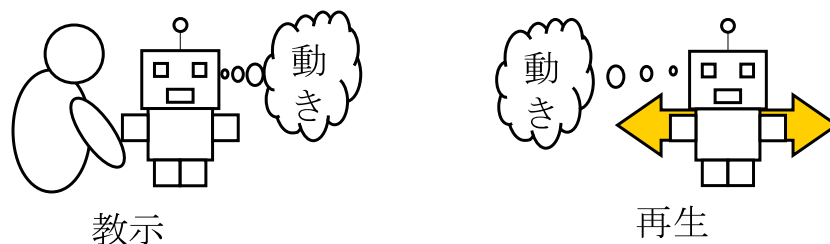


図 1.2: 教示と再生

#### 1.1.5 操作することによりロボットを動かす方法

最後に「操作することによりロボットを動かす方法」について考える（図 1.3）．ユーザが状況に応じて直接ロボットを操作することが出来れば，設計者が各状況を想定して動きを設計する必要がなくなる．これにより未知の状況への対応が容易となり，また設計者の負担も軽減することが出来る．しかし，リアルタイムでロボットを操作する必要があるため，ユーザの負担は増大する．

#### 1.1.6 まとめ

本研究では「ユーザがロボットに対して教示を行う方法」について着目する．

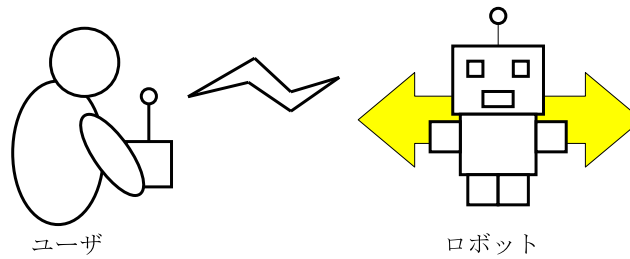


図 1.3: ユーザによる操作

## 1.2 従来研究

ここでは教示に関する従来研究について述べる。教示に関する研究は、大きく「教示」と「再生」に分けることが出来る。ロボットに対して、ユーザーが行わせたいタスクを教えることを「教示」という。ロボットがユーザーから教示された動きを元に動作することを「再生」という。

従来の教示手法として

- 直接教示
- 遠隔教示

などがある。「直接教示」はロボットに直接接触することで教示を行う。「遠隔教示」は遠隔地からデバイスを用いて教示を行う。

従来の再生手法として

- そのまま再生
- 状況に合わせて動きを修正

などがある。「そのまま再生」は、ロボットに教示した動きをなぞって動く。「状況に合わせて動きを修正」は、センサなどを用いて、状況に合わせて適する動きを選択しながら動く。

それぞれの手法についてさまざまな研究が行われている。

## 1.3 本研究の着眼点

従来研究には、教示手法に着目したものが多く、特に、よりユーザーの思い通りの教示を行うことを目的とした研究が多い。そのため、再生手法に着目した研究は少ない。そこで、本研究では再生手法に着目する。

まず、再生時にロボットに対して操作を行うことで、動きを変更可能なシステムを考える。例を図 1.4 に示す。この方法を用いることにより、ユーザーは再生時に自分の好ましい動きを選択出来るようになる。

このシステムを実現する方法として、設計者が操作方法を設計する方法が考えられる。例を、図 1.5 に示す。この例は、「レバー A を前後に」動かすことで、「力加減について」操作出来るように設計者が設計を行っていることを示している。これにより、ユーザーは再生時に動きを変更することが可能となる。

しかし、設計者が操作方法を設計する方法は、設計者が想定していない操作方法をユーザーが行うことは出来ないという問題点がある。例を図 1.6 に示す。図 1.6 のように設計者とユーザーで操作したい要素が異なる場合、ユーザーは設計者に操作方法の変更を要求する必要がある。そのため、ユーザーが希望する操作方法を自ら作成することが出来るシステムが求められる。

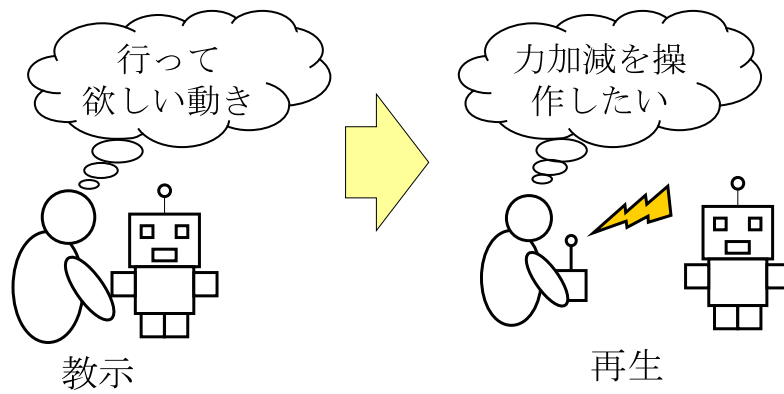


図 1.4: 再生時に操作を行う例

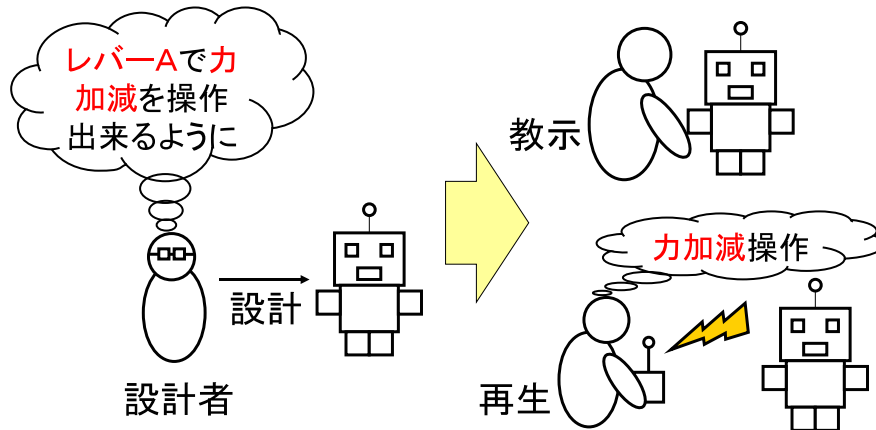


図 1.5: 設計者が操作方法を設計する場合

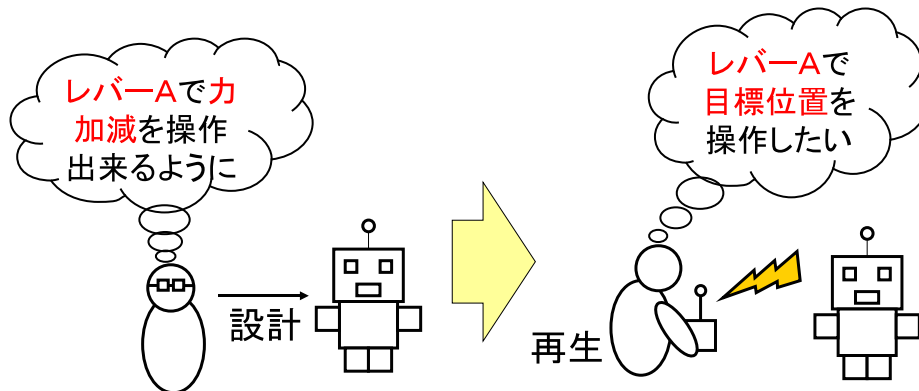


図 1.6: 設計者が操作方法を設計する問題点

## 1.4 目的

本研究では、教示と再生を用いたシステムに着目する。そして、再生時にユーザの操作によりロボットの動きを変えることが出来るシステムについて考える。特に、ユーザが「何について」「どのように」操作するかを決めることが出来るロボット操作システムの提案を行う。例を図 1.7 に示す。

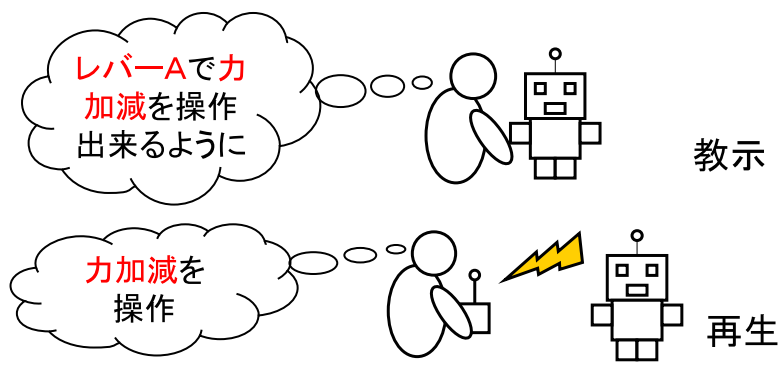


図 1.7: 目的の例



## 第2章 アプローチ

### 2.1 概要

本節ではアプローチの概要について説明する．本研究は，教示と再生を元にした操作システムを提案する．

本研究のアプローチの概要図を図 2.1 に示す．

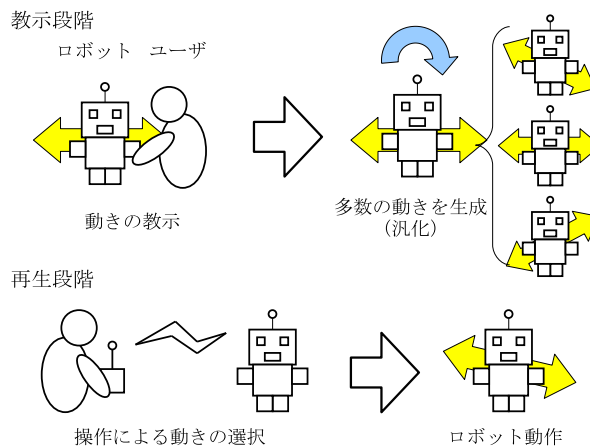


図 2.1: アプローチ概要図

アプローチを教示段階と操作段階の2段階に分ける．各段階の概要について説明する．

まず，教示段階について説明する．ユーザはロボットに対して教示を行う．ロボットは，ユーザから教示された動きを元に教示された動きに類する多数の動きを生成する．本研究では，これを動きの汎化とする．本研究における教示された動きに類する動きとは，例えば「手を振る動作」であれば「手を振る大きさの違う動き」や「手を振る速さの違う動き」など，教示した動きと似ているが多少異なる動きのことを表す．

次に，再生段階について説明する．ユーザはロボットが動きの再生を行っている間に操作を行う．教示段階で汎化した動きの中から操作により動きを選択する．

2.2 節で教示段階についての詳細説明を行う．2.3 節で再生段階についての詳細説明を行う．2.4 節では，教示段階・再生段階について，ユーザがロボットの動きを選択するだけでなく，状況に応じてロボットが動きを選択出来るように拡張するアプローチを示す．

### 2.2 教示と汎化

本節では教示段階についての詳細説明を行う．教示段階における手順を図 2.2 に示す．

ユーザはロボットに対して動きの教示を行う．また，ユーザはロボットに対して操作方法の教示を行う．動きの教示と操作方法の教示の概念図を図 2.3 に示す．動きの教示とは，例えば「手を振らせたい」「歩かせたい」など，ユーザがロボットに対して行わせたいタスクを教えることである．操作方法の教示とは，例えば教示した動きについて「力加減を操作

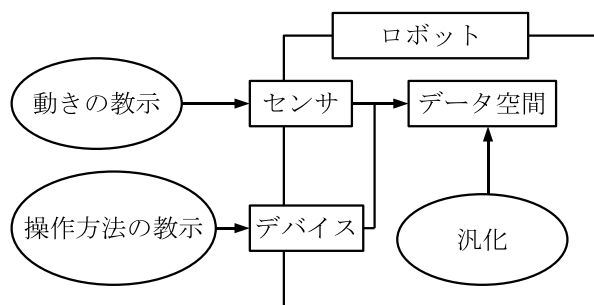


図 2.2: 教示の手順

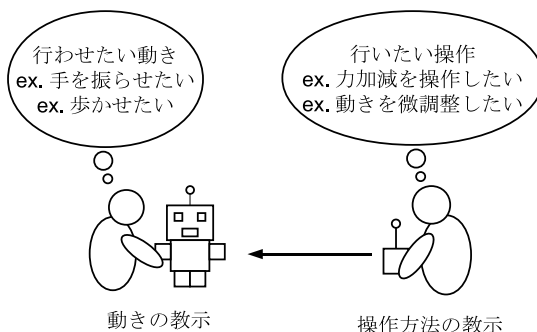


図 2.3: 動きの教示と操作方法の教示

しながらロボットを動作させたい」「動きを微調整しながらロボットを動作させたい」など、ユーザが操作したいものについてロボットに教えることである。

ロボットは教示された動きをセンサから取り込み、ロボット内部にもつデータ空間に記録する。また、ロボットは教示された操作方法をデバイスから取り込み、ロボット内部にもつデータ空間に記録する。これらのデータを元に動きの汎化を行う。動きの汎化により、教示した動きに類する動きが生成される。汎化された動きは、例えば「手を振る動作について汎化され、手を振る大きさの加減を操作出来る」「歩く動作について汎化され、足を踏み出す力強さを操作出来る」というように、教示した動きに類する動きの操作が可能となる。

## 2.3 再生と操作

本節では、再生段階についての詳細説明を行う。再生段階における手順を図 2.4 に示す。

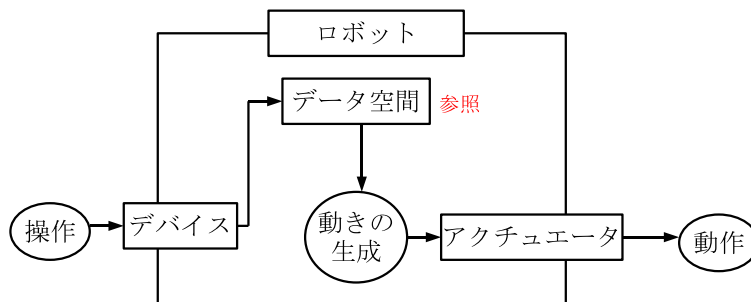


図 2.4: 再生の手順

動きの再生中にユーザはロボットに対して操作を行う。ロボットはユーザからの操作に関する情報を元にデータ空間を参照し、汎化された動きの中から1つの動きを生成する。生成

した動きをアクチュエータ（モーターやサーボなど、ロボットの出力にあたる）に渡し、動作する。

例えば、手を振る動きが汎化され、振る大きさについて操作出来るデータ空間があるとする。ユーザはロボットが手を振る動きを再生している最中に操作を行う。操作により、振る大きさが異なる手を振る動作を選択することが出来る。

## 2.4 状況に応じた動きの選択

本節では、2.2節、2.3節で説明したアプローチを拡張し、ユーザがロボットの動きを選択するだけでなく、状況に応じてロボットが動きを選択出来るようにするアプローチを示す。教示に関する手順を図 2.5 に示す。操作に関する手順を図 2.6 に示す。

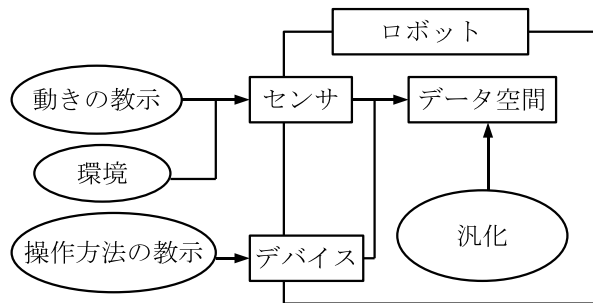


図 2.5: 教示の手順

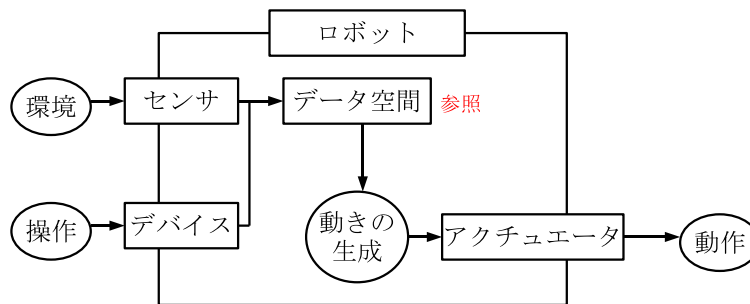


図 2.6: 再生の手順

図 2.2 および図 2.4 と異なる点は、ロボットが環境からの情報を取り込む点である。

教示段階について説明する。2.2節では、「手を振らせたい」「歩かせたい」など、ユーザがロボットに対して行わせたいタスクを教え、「力加減を操作したい」「動きを微調整するように操作したい」など、ユーザが操作したいものを教えている。本節では、これらに加えて「力加減を自動的に調節する」「動きの微調整を自動的に行う」など、ロボットが環境・状況に応じて動きの選択を自動的に行えるように環境からの情報を記録し、動きの汎化を行う点が2.2節と異なる。ユーザがロボットに対して行わせたいタスクを教えている最中、ロボットは周囲の環境やおかれている状況を記録する。これにより、どのような環境・状況でどのような教示が行われたかを記録する事が出来る。このデータを元に動きの汎化を行う。例えば「力加減の異なる動きを生成する」「動く速さが異なる動きを生成する」など、教示した動きに類する動きを多数生成し、「力加減を自動的に選択する」「動く速さを自動的に選択する」など、再生時にはその中から、環境・状況に応じて自動的に動きが選択されるように動きの汎化を行う。

再生段階について説明する。2.2 節では、ロボットが動きを再生している最中にユーザが操作を行うことで動きの選択を行っている。本節では、ユーザによる操作に加えて、ロボットがセンサから周囲の環境・状況を取り込み、その情報を元に動きの選択を自動的に行うことが出来るように拡張する。これにより、ユーザからの操作に関する情報と、ロボットが持つセンサから得ることが出来る環境・状況に関する情報を用いてロボットの動きが選択されることになる。

# 第3章 「ロボットの動きに関する新しい表現方法」を用いた動きの汎化と汎化した動きの操作

## 3.1 概要

本章では、2章で示したアプローチを実現する方法について述べる。アプローチでは、ロボットへの教示時にデータ空間を用いて動きの汎化を行い(図 2.2)，再生時にデータ空間を用いて動きの生成を行うこと(図 2.4)を説明した。本章では、まず本論文で用いるデータ空間について説明する。その後、データ空間を用いて動きの汎化を行う方法と、汎化されたデータ空間を用いて動きを生成する方法について説明する。最後に作成したシステムについて説明する。作成したシステムは全部で4つである。

3.2節で本論文で用いるデータ空間についての説明をする。3.3節でデータ空間を用いて動きの汎化を行う方法を説明する。3.4節で汎化されたデータ空間を用いて、ユーザによる操作から動きを選択し、生成する方法を説明する。3.5節では3.2節～3.4節を元に、頻度空間を用いて状況に応じて動きの選択を行う方法について説明する。3.6節で、ユーザによる操作と、状況に応じた動きの選択について説明を行う。3.7

## 3.2 本論文で用いる空間の説明

### 3.2.1 本論文で用いる空間の概要

本節ではデータ空間の説明を行う。本論文ではデータ空間として、室蘭工業大学情報工学科の2008年卒業論文「ロボットの動きに関する新しい表現方法」[16]内で定義されている「頻度空間」を改変して用いる。

頻度空間は運動軌道の生成を行うことを目的として作られたデータ空間である。ロボティクスにおける運動軌道とは、時間におけるロボットの動きの軌道である。ロボットの関節の位置を時系列で並べたものとして扱う。運動軌道により、ロボットの動きが表現される。

例として、人型ロボットの腕の上げ下げ運動(図 3.1)の軌道を示す。運動軌道の例を図 3.2に示す。図 3.2のように時間毎の腕の位置を表すことにより、腕の動きが表現出来る。

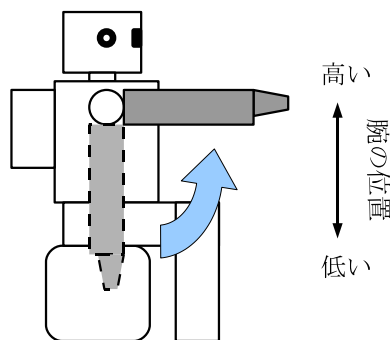


図 3.1: 人型ロボットの腕の上げ下げ運動

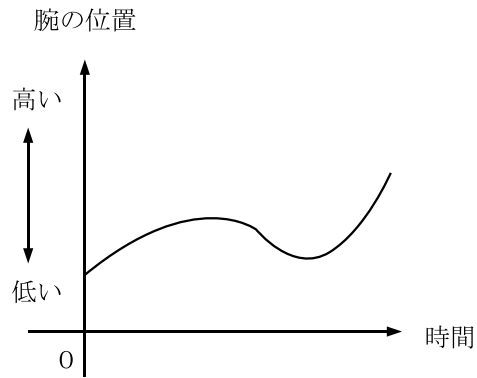


図 3.2: 運動軌道の例

図 3.2 の目標軌道の例を一般化すると図 3.3 のように各時間におけるロボットの各関節の位置となる。

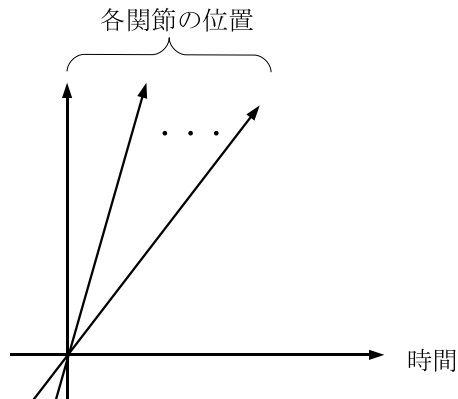


図 3.3: 運動軌道の概要図

選択頻度を用いた軌道生成は環境に適応的であり、また軌道を容易に作ることが出来る利点を持つ。本論文では動きを汎化し、汎化した動きを元に操作することを目的とする。そのため、運動軌道の生成方法については、容易に行うことが出来るものを選択した。

本論文では「ロボットの動きに関する新しい表現方法」[16]内で定義されている「頻度空間」を改変して用いるため、本論文で用いる頻度空間について改めて定義する。頻度空間を定義するために、まず本論文における状態空間、動作空間についてそれぞれ説明する。以下の順番で説明を行う。

- 状態空間
- 動作空間
- 頻度空間

### 3.2.2 状態空間

ロボットが持つ各関節の位置と各センサの値を軸とする多次元空間を考える。本論文では、この多次元空間をロボットの状態空間と呼ぶ(図 3.4)。この空間によって、ある時点のロボットが置かれている状態を表すことが出来る。

ここで、状態空間におけるセンサについて考える。ロボティクスにおけるセンサとは、「ロボットにかけられている負荷」や「ロボットの姿勢」などのロボットの身体に関わる情報や、

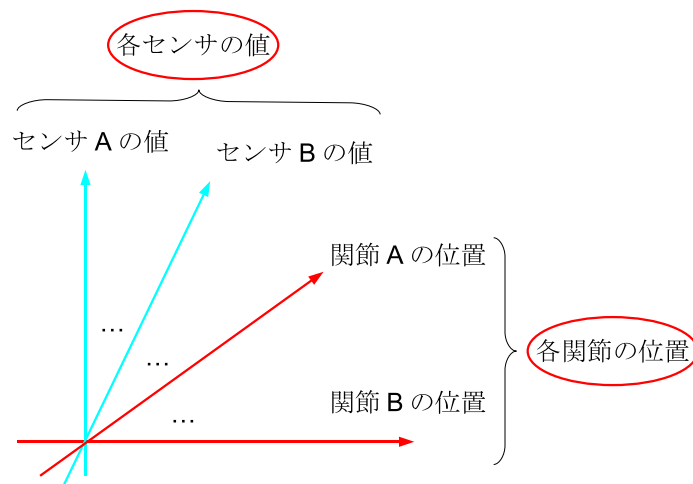


図 3.4: 状態空間の概念図

「周辺の温度」や「鳴っている音」などロボットが置かれている周囲の情報を認識する装置の意味で用いられている事が多い。本論文では上述したセンサの意味を拡張する。

ロボットからすると人間も環境内に存在するものの1つである。そのため、本論文では人間からロボットに対して送られる情報は、ロボットが認識出来る環境に関する情報の1つであると考えられる。そこで本論文では、ロボットの体に関わる情報とロボットの周囲の情報の他に、人間からの操作に関する情報についてもセンサとする(図 3.5)。

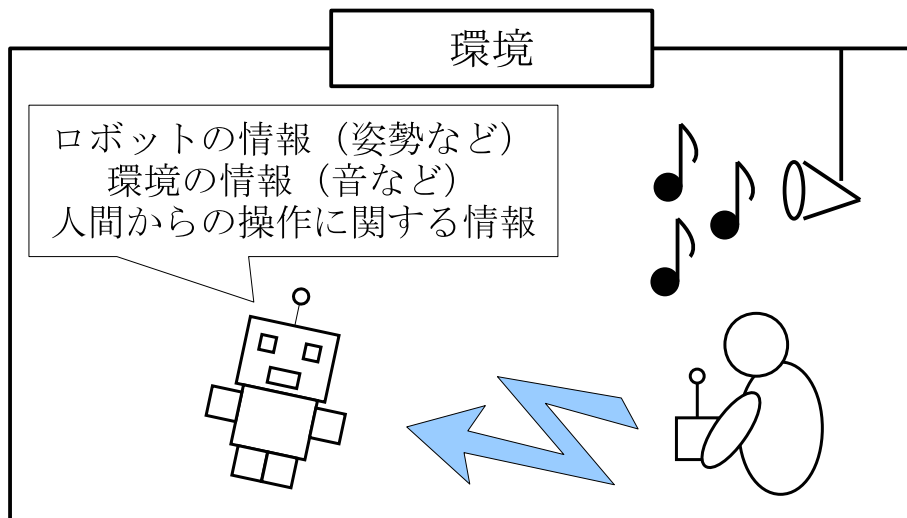


図 3.5: 本論文におけるセンサの概念図

### 3.2.3 動作空間

状態空間に時間を加えた空間を考える。本論文ではこの空間を動作空間と呼ぶ(図 3.6)。動作空間は時間毎のロボットの状態を保持出来るため、ロボットの動きや状態の変化を表すことが出来る。

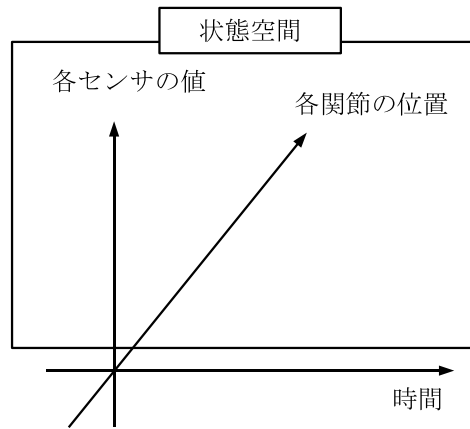


図 3.6: 動作空間の概念図

### 3.2.4 頻度空間

#### 頻度空間の概要

頻度空間は動作空間に選択頻度というパラメータを加えた空間である。選択頻度とは、動作空間に「選択のされやすさ」という概念を考えたものである。頻度空間と選択頻度の概念図を図 3.7 に示す。

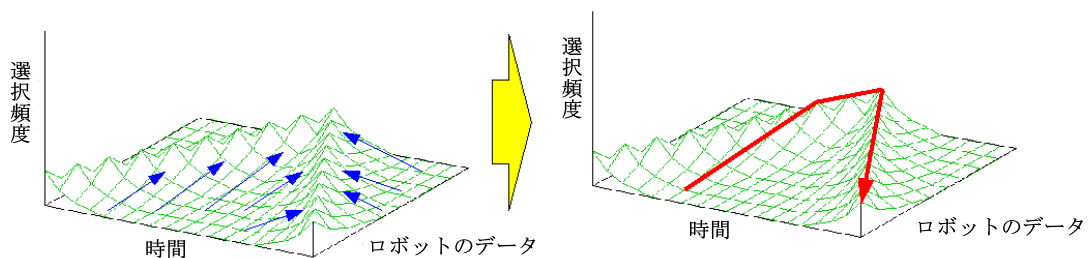


図 3.7: 頻度空間と選択頻度の概念図

図 3.7 のようにロボットのデータ（状態空間を構成する各要素）、時間、選択頻度からなる空間が頻度空間である。ロボットのデータを関節の位置と仮定する。ある時間の関節の位置が頻度空間のどこにあっても、時間の経過と共に選択頻度の高い場所へ徐々に移動する。この時、選択頻度が高い地点の関節の位置は選択されやすい地点であるといえる。

#### 頻度空間の詳細説明

まず、動作空間を構成する各軸をそれぞれ一定間隔で区切り、離散的な空間として表す。そして、区切りにより出来たマスセルをセルと定義する（図 3.8）。ここで、各セルに対して選択頻度を与える。これを頻度空間とする。

## 3.3 頻度空間を用いた動きの汎化

### 3.3.1 概要

本節では、3.2 節で定義した頻度空間を用いて動きの汎化を行う方法について述べる。動きの汎化は教示段階で行う。ユーザから教示された情報を頻度空間に保存し、頻度空間を更



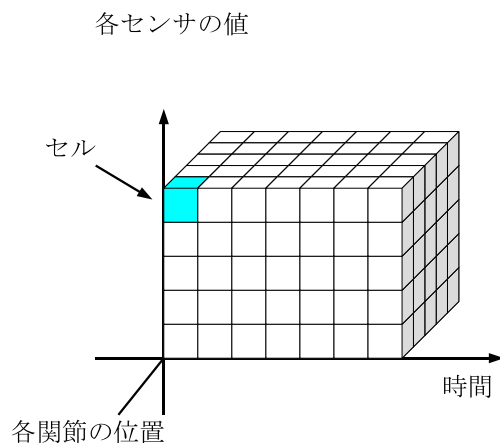


図 3.8: 動作空間の離散化とセルの概念図

新することで動きの汎化を行う．3.3.2 句で頻度空間の更新方法について述べる．3.3.3 句では，3.3.2 句で示した頻度空間の更新によってなぜ動きが汎化されるのかについて述べる．

### 3.3.2 頻度空間の更新

頻度空間の更新の概要図を図 3.9 に示す．

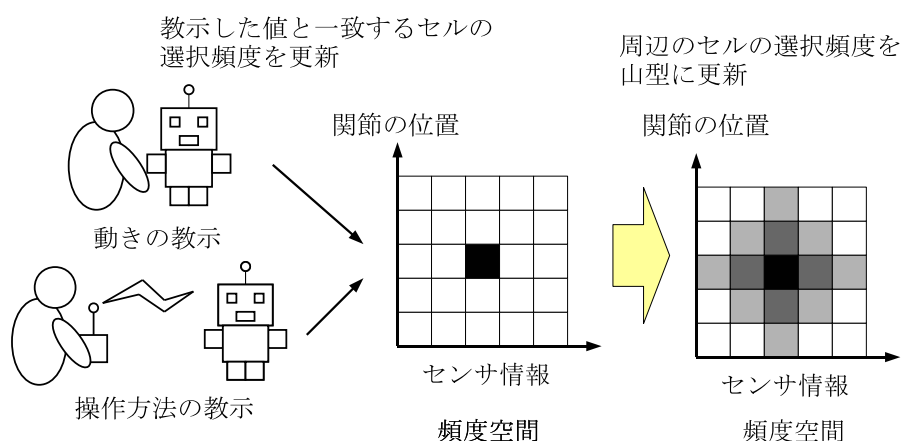


図 3.9: ある時間における動きの汎化の手法

頻度空間の更新は各時間毎に教示された情報を元に行われる．図 3.9 は各時間でどのように動きの汎化が行われるかを示している．ユーザはロボットに対して動きの教示と操作方法の教示を行う．動きの教示と操作方法の教示についてはすでに 2 章で解説している．ロボットは各関節の位置と各センサの値（ここでは操作に関する情報）を用いて頻度空間の更新を行う．取得した各関節の位置・各センサの値と一致するセルを 1 つ見つけ、そのセルを中心に山形に選択頻度を更新する．

各時間における更新式は以下ようになる．

$$x_i \leftarrow x_i + c \prod_{j=1}^n (\beta_j)^{\delta_{ij}} \quad (3.1)$$

$i$  は頻度空間に存在する各セルに割り当てられた数字である． $j$  は頻度空間に存在する各軸のうち、状態空間を構成するのに必要な各軸に対して割り当てた数字である． $c$  は更新の中心の加算値である．更新の中心となるのは取得した各関節の位置・各センサの値と一致する

セルである．この点を中心として山形に選択頻度が更新される． $\beta_j$  は軸  $j$  方向における割引率である．これは，取得した各関節の位置・各センサの値と一致するセルを中心として，軸  $j$  方向に 1 マス離れる毎に更新の中心の加算値  $c$  からどのくらい割り引かれるかを示す． $\delta_{ij}$  はセル  $i$  における軸  $j$  方向の更新の中心からの距離である．

更新式を用いて頻度空間を更新する例を図 3.10 に示す．ある時刻の関節の位置が 3 でセン

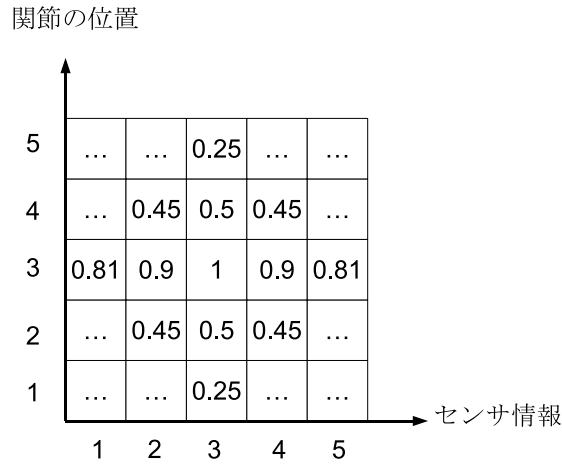


図 3.10: 各時間毎の頻度空間の更新の例

サ情報の値が 3 であったとする．この時，更新の中心は頻度空間の中でこれらの値を示すセルである． $c = 1$  とすると，関節の値が 3・センサ情報の値が 3 であるセルの選択頻度が +1 される．また， $j = 1$  をセンサ情報軸とし， $\beta_1 = 0.9$  とすると，センサ情報軸方向に中心から 1 セル離れる毎に  $c$  から 0.9 倍された値が選択頻度に加算される．同じように， $j = 2$  を関節の位置軸とし， $\beta_2 = 0.5$  とすると，関節の位置軸方向に中心から 1 セル離れる毎に  $c$  から 0.5 倍された値が選択頻度に加算される．例えば，センサ軸方向に 1 セル離れ，関節の位置軸方向に 1 セル離れているセル（センサ情報軸が 2，関節の位置軸が 2 であるセルなど）は  $1 * 0.9 * 0.5 = 0.45$  となる．

このような更新を各時間毎に行う．例を図 3.11 に示す．ここではロボットのデータ軸を仮

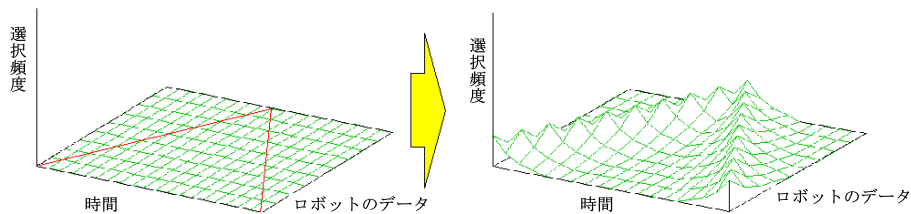


図 3.11: 頻度空間の更新の例

にロボットの関節の位置を表す軸とする．左図の赤線のように関節の位置が遷移したとすると，右図のように山が連なっているような形で頻度空間が更新される．

### 3.3.3 頻度空間の更新と動きの汎化の関連

3.3.2 句では，頻度空間の更新は教示した動きと各時間のセンサ値（ここでは操作に関する情報とする）を中心に選択頻度が山型に更新されることを示した．これにより，教示した動きだけではなく，教示した動きに近い動きについても再生時に選択される可能性が自動的

に上昇する．つまり教示した動きだけではなく，頻度空間の更新により教示した動きに近い動きについても再生できるようになる．

また，頻度空間の更新は，複数回行うことが出来る．教示を複数回行うことで，頻度空間の更新が複数回行われる．ユーザはロボットに対して複数回の異なる動きの教示を行う．これにより，1つの頻度空間にさまざまな動きが蓄積される．教示した複数の動きに類する動きについては，それぞれの動きについて頻度空間の更新方法に基づき更新されるため，教示した各動きの間の動きについては自動的に補完されるといえる．

以上より，教示した各動きの補完は頻度空間の更新によって行われるものであるといえる．そこで本論文では，頻度空間の更新によって，教示していない動きについても動きの補完を行うことを動きの汎化とする．また，頻度空間の更新が行われた空間を用いた動きを汎化された動きとする．

### 3.4 頻度空間を用いた操作による動きの生成

#### 3.4.1 概要

本節では，3.3節で説明した手法によって更新された頻度空間から動きを生成する方法について述べる．ユーザが操作を行うことにより，生成される動きを変えることが出来る．3.4.2句で動きを生成する手法について述べる．3.4.3句で操作により動きを変える方法について述べる．

#### 3.4.2 動きの生成

ここでは，十分に選択頻度が更新されている頻度空間があるものとして考える．動きの生成の概要図を図 3.12 に示す．

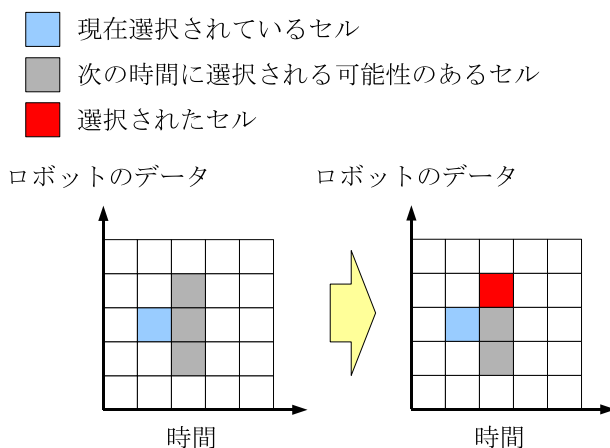


図 3.12: 動きの生成の概要図

まず，ロボットは現時刻のロボットのデータ（ここでは関節の位置とする）を取得し，頻度空間の中から取得した値と一致するセルを選択する．そして，次の時間の目標となるセルを選択する．選択の方法は，次の時間の中で最も選択頻度の高いセルとする．ただし，現在選択されているセルからの距離に制限を設け，その中から選択する（この距離の制限内に含まれるセルを図 3.12 内では次の時間に選択される可能性のあるセルとしている．）新しく選択されたセルにおけるロボットのデータ軸の値が次の時間に目標とする値となる．ここではロボットのデータを関節の位置としているので，新しく選択されたセルにおけるロボットの

データ軸の値が次の時間の関節の目標位置となる．これを繰り返すことで，次々と目標位置を算出することで，動きが生成される．例を 3.13 に示す．

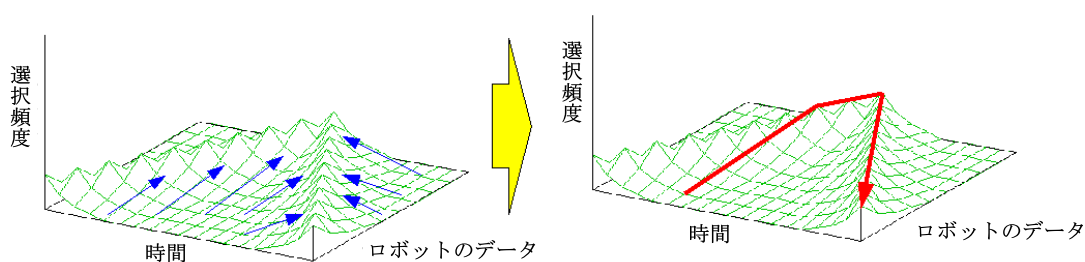


図 3.13: 動きの生成の例

図 3.12 内の左の頻度空間の青矢印の方向に向かうように次々とセルが選択される．その結果．例えば，図 3.12 内の左の頻度空間の赤矢印のようにセルを選択される．この時，各時間ごとに選択されたセルがロボットの軌道となる．

### 3.4.3 操作による動きの変更

3.4.2 句では，頻度空間内のロボットのデータ軸を関節の位置に関する軸として扱った．しかし，実際には状態空間を表すために必要なすべての軸がロボットのデータ軸である．その中に各センサの軸があり，本論文では操作に関する情報もセンサの 1 つとして扱っている．

ロボットのデータ軸として関節の位置と操作に関する情報が 1 軸ずつある場合を考える．この場合，ロボットの動きの生成は現時刻の関節の位置と現時刻のユーザによる操作に関する情報から次の時刻の関節の目標位置が決められる．ユーザは操作により，操作に関する情報軸の値を自由に変えることができる．そのため，ユーザはある程度自分の意思で選択されるセルを操作することが出来ことになる．これにより次の時刻の関節の目標位置も変わり，動きが変更されることになる．

上述した例は関節の位置と操作に関する情報が 1 軸ずつであるが，実際には関節の位置と操作に関する情報は 1 軸ずつとは限らない．この場合も上述した例と同じように，現時刻の各関節の位置と現時刻の各操作に関する情報から次の時刻の各関節の目標位置が決められる．

## 3.5 頻度空間を用いた状況に応じた動きの選択

### 3.5.1 概要

本節では，頻度空間を用いて状況に応じて動きの選択を行う方法について説明する．3.3 節と 3.4 節内ではセンサを操作に関する情報として扱った．しかし，頻度空間の定義ではセンサは操作に関する情報だけではなく，ロボットの体に関わる情報やロボットの周囲の情報などもセンサに含まれる．そこで，3.3 節と 3.4 節で用いられている「操作に関する情報」を「ロボットの体に関わる情報」や「ロボットの周囲の情報」と読み替えることで，状況に応じた動きの選択を行う手法を示す．手法を教示段階と再生段階に分けて説明する．

### 3.5.2 教示段階

教示段階では，「ロボットの体に関わる情報」や「ロボットの周囲の情報」により，動きの選択が行えるように動きの汎化を行う．動きの汎化は頻度空間の更新により行われる．頻度

空間の更新方法は3.3.2句で示した．頻度空間の更新により，教示した動きに類する動きが補完される．この時，ユーザによる「操作に関する情報」の代わりに「ロボットの体に関わる情報」や「ロボットの周囲の情報」をロボットが取得し，動きの汎化を行う．これにより，再生時に「ロボットの体に関わる情報」や「ロボットの周囲の情報」を元にロボットが動きの選択を行う．

### 3.5.3 再生段階

再生段階では，「ロボットの体に関わる情報」や「ロボットの周囲の情報」を用いて動きを選択し，動きの生成を行う．頻度空間を用いて，操作により場合の動きを選択する手法は3.4.2句で示した．ここでは操作により動きを選択するかわりに，「ロボットの体に関わる情報」や「ロボットの周囲の情報」を用いる場合を考える．

ロボットのデータ軸として関節の位置とロボットの周囲の情報の値を示す軸が1軸ずつある場合を考える．この場合，ロボットは現時刻の関節の位置とセンサの値（ロボットの周囲の情報）を読み取り，頻度空間の中から次の関節の目標位置を決める．次の時刻の関節の目標位置も変わり，動きが変更されることになる．

上述した例は関節の位置とセンサの値を示す軸が1軸ずつであるが，実際には関節の位置とセンサの値を示す軸は1軸ずつとは限らない．この場合も上述した例と同じように，現時刻の各関節の位置と現時刻の各センサの値から次の時刻の各関節の目標位置が決められる．

## 3.6 ユーザによる操作と状況に応じた動きの選択

3.3節および，3.4節では，センサをユーザの操作に関する情報として扱った．また，3.5では，頻度空間におけるセンサをロボットの体に関わる情報やロボットの周囲の情報として扱った．しかし，実際の頻度空間におけるセンサは「ロボットの体に関わる情報」「ロボットの周囲の情報」「ユーザの操作に関する情報」の全てである．よって，頻度空間を用いることにより，ユーザによる操作と，ロボットが置かれている状況の両方から動きの選択を行うことも可能である．

## 3.7 本論文で作成したシステム

本節では，作成したシステムについて説明する．システムの概要図を図3.14に示す．

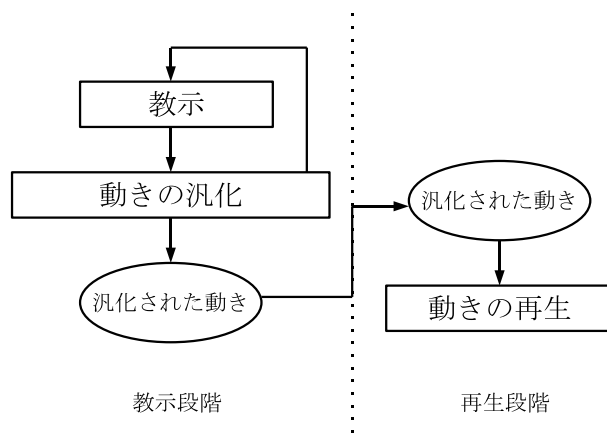


図 3.14: システムの流れ

まず，教示段階について説明を行う．教示段階の概要図を図 3.15 に示す．ユーザがロボットに対して動きの教示と，操作方法の教示を行う．ロボットは単位時間毎の関節の位置とセンサの値を取得し，システムへ送る．そして，取得した関節の位置とセンサの値を元に，頻度空間を更新することで動きの汎化を行う．頻度空間を更新する方法については 3.3 節で示した．ユーザによる動きの教示・操作方法の教示と動きの汎化は複数回行うことが出来る．

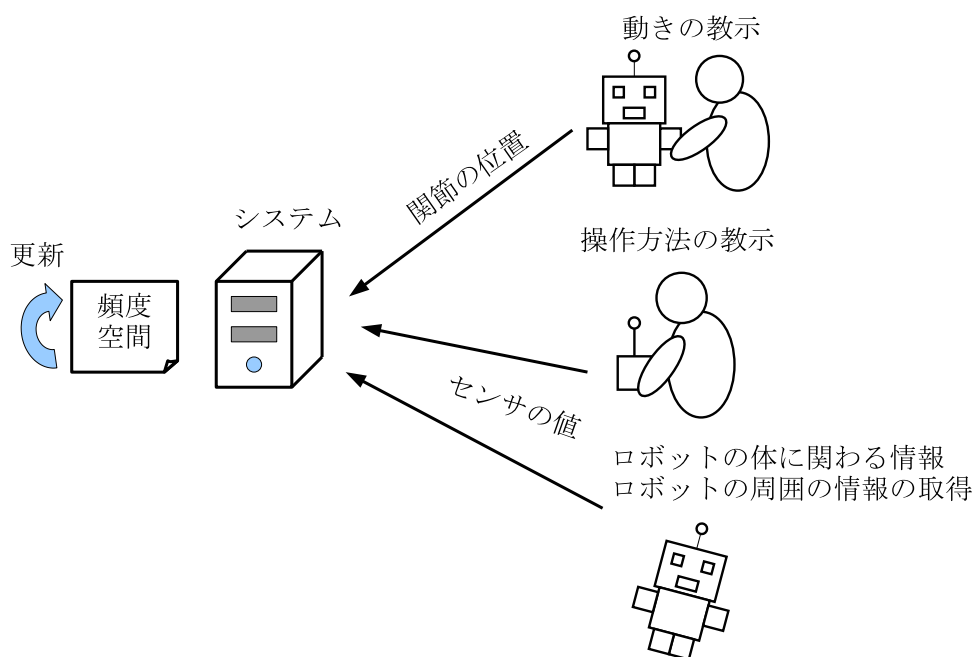


図 3.15: 教示段階の概要図

次に，再生段階について説明を行う．再生段階の概要図を図 3.16 に示す．ロボットは現時刻の関節の位置とセンサの値を取得し，システムへ送る．システムは現時刻の関節の位置とセンサの値から次の時刻の関節の目標位置を計算し，ロボットへ送る．現時刻の関節の位置とセンサの値から次の時刻の関節の目標位置を計算する方法については 3.4 節で示した．ロボットはシステムから送られてきた情報を元に，関節を目標位置まで動かす．関節の位置とセンサの値の取得から関節を目標位置まで動かすところまでを繰り返すことにより，動きが再生される．

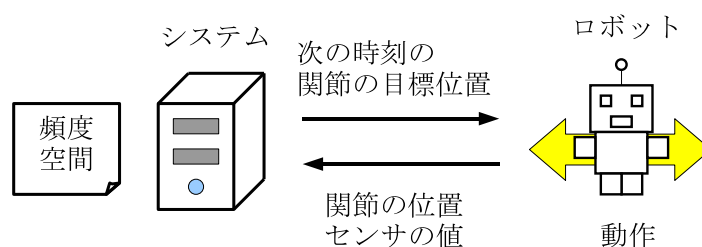


図 3.16: 再生段階の概要図

## 第4章 実験

### 4.1 概要

本章では，本研究で行った4つの実験の内容とその結果について述べる．さらに各実験について考察を行う．

行った実験は以下の通り．

- センサ，操作デバイスを用いない実験（実験1）
- センサのみ用いる実験（実験2）
- 操作デバイスのみ用いる実験（実験3）
- センサ，操作デバイスを用いる実験（実験4）

4.2節では本実験で使ったロボットと操作デバイスについての説明を行う．4.3節～4.6節ではそれぞれ実験1～実験4についての実験内容の説明と結果について述べる．

### 4.2 使用したロボットと操作デバイス

#### 4.2.1 使用したロボット

本研究ではロボットとして，SPC-001を用いた．これは，Speecys社により販売されている人型ロボットの1つであり，多数のサーボやセンサーを備える．SPC-001の外観を図4.1に示す．SPC-001の身体構造を図4.2に示す．SPC-001の基本的な仕様を表4.1に示す．

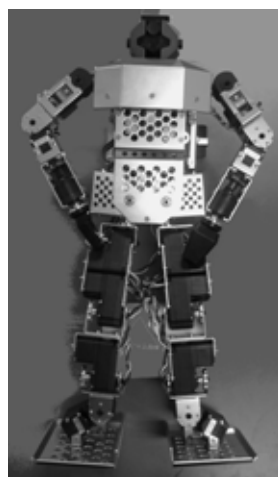


図 4.1: ロボットの外観

本研究では，関節可動部（サーボ）とGセンサーを用いる．

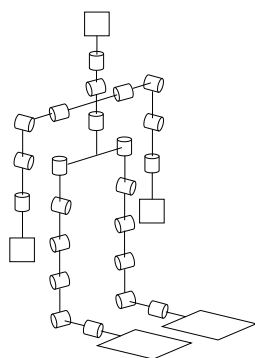


図 4.2: ロボットの身体構造

表 4.1: ロボットの仕様

|                     |                |
|---------------------|----------------|
| 名前                  | SPC-001        |
| 関節可動部自由度            | 2 3 自由度        |
| LED (眼部分)           | 3 色 * 2 セット    |
| 3 5 万画素 CMOS カラーカメラ | 1 個            |
| マイク入力               | 2 個            |
| 音声出力用スピーカ           | 1 個            |
| ジャイロセンサ             | 3 軸            |
| 3 軸 G センサ           | 1 セット          |
| 寸法                  | 約 50cm         |
| 重量                  | 3.7kg          |
| 無線 LAN              | IEEE802.11b 準拠 |

表 4.2: サーボの仕様

|           |                     |
|-----------|---------------------|
| 名前        | RS 6 0 1 CR         |
| 重量        | 93g                 |
| 動作速度      | 0 . 1 7 秒 / 6 0 度   |
| 可動範囲      | 2 4 0 度             |
| 角度センサの分解能 | 0 . 1 度             |
| 角度センサの取得値 | - 1 2 0 0 ~ 1 2 0 0 |

サーボはセンス情報として、角度・トルク・温度などを取得することが出来る。本研究ではサーボからのセンス情報として、角度情報のみを用いる。本研究に関連するサーボの仕様を表 4.2 に示す。

角度センサの取得値が表すロボットの状態は、サーボの付いている位置と方向により異なる。本研究では右肩・前後方向の 1 サーボと左肩・前後方向の 1 サーボを用いる。角度センサの取得値とロボットの状態の関係図を示す。右肩・前後方向と角度センサの取得値の関係図を図 4.3 に示す。左肩・前後方向と角度センサの取得値の関係図を図 4.4 に示す。

本論文では、角度センサの取得値に+1200 のバイアスを加えた値で表す。この場合の右肩・前後方向と角度センサの取得値の関係図を図 4.5 に示す。また、左肩・前後方向と角度センサの取得値の関係図を図 4.6 に示す。

G センサはロボット背面にあるセンサボードに実装されている。本研究に関連する G センサの仕様を表 4.3 に示す。



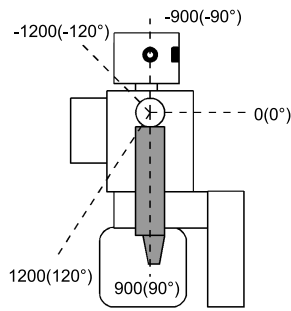


図 4.3: 本来の右肩・前後方向のサーボと角度センサ取得値の関係図

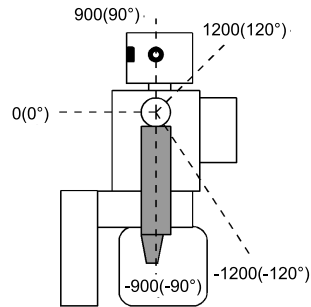


図 4.4: 本来の左肩・前後方向のサーボと角度センサ取得値の関係図

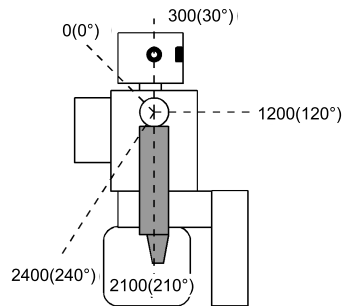


図 4.5: 本論文における右肩・前後方向のサーボと角度センサ取得値の関係図

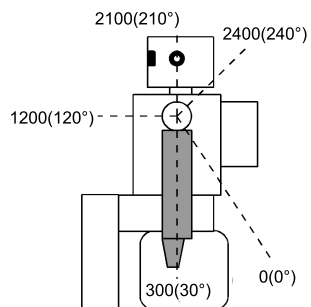


図 4.6: 本論文における左肩・前後方向のサーボと角度センサ取得値の関係図

本研究では、3軸存在するGセンサのうち、1軸のみを用いる。本研究で用いる軸のセンサ取得値はロボットが直立状態で0の値を示す。また、うつ伏せ静止状態で約90の値を示し、仰向け静止状態で約-90の値を示す。

表 4.3: G センサの仕様

|           |             |
|-----------|-------------|
| G センサの取得値 | - 477 ~ 477 |
|-----------|-------------|

#### 4.2.2 使用した操作デバイス

本研究では操作デバイスとして、EXTREME 3D PRO を用いた。これは Logitech 社が販売したジョイスティックである。EXTREME 3D PRO の外観を図 4.7 に示す。EXTREME 3D PRO の基本的な仕様を表 4.4 に示す。



図 4.7: EXTREME 3D PRO の外観

表 4.4: EXTREME 3D PRO の仕様

| 名前          | EXTREME 3D PRO |
|-------------|----------------|
| アナログ入力      | 3 自由度          |
| アナログ入力取得値   | 0 ~ 6 5 5 3 5  |
| ボタン入力       | 1 2 個          |
| 8 方向ハットスイッチ | 1 個            |
| スロットル       | 1 個            |

本研究ではアナログ入力のみを用いる。また、3 軸存在するアナログ入力のうち 1 軸のみを用いる。使用する軸は、レバーの前後方向への操作により値が変化する。使用するアナログ入力の軸と取得値の関係を図 4.8 に示す。

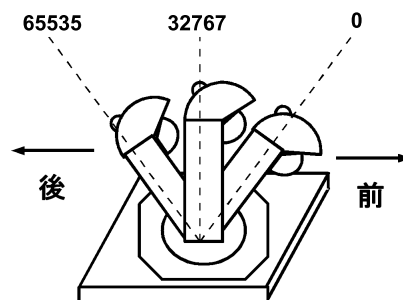


図 4.8: 使用するアナログ入力の軸と取得値の関係

本研究では、これらのロボット・ジョイスティックを用いて実験を行う。

## 4.3 センサ，操作デバイスを用いない実験

### 4.3.1 実験の説明

#### 実験の概要と目的

本節では，作成したシステムが正常に動作することを確認するための実験について説明する．特に，頻度空間が最低限の軸からなる場合についての実験を行う．これにより，動きの汎化が正常に行われ，動きの再生が正常に行われることを確認する．本実験の目的を達成するために，本節ではロボットに対して教示を行い，頻度空間を用いて動きを汎化する実験を行う．また，汎化された動きを用いて動きを再生する実験を行う．

#### 用いるロボットの部位

本実験では，ロボットが持つサーボを用いて実験を行う．サーボは現在の回転角度を取得することが出来る．そのため，サーボの回転角度をシステムにおける「関節の位置」として扱う．サーボは右肩・前後方向のサーボのみを用いる．

#### 実験で行うタスク

実験として，ロボットの腕の上げ下げを2回行わせるタスクを教示する．さらに，教示した動きを元にロボットを動作させる．教示方法は直接教示とする．直接教示とは，ロボットへ直接接触して教示する方法である．

タスクの概要を図 4.9 に示す．タスクの詳細を以下に示す．

1. ロボットは座った状態で実験を行う
2. ロボットの右腕の初期状態は，地面に対して垂直とする
3. 右腕を前方に向けて地面に水平になるまで上げる
4. 右腕を地面に垂直になるまで下げる
5. 右腕の上げ下げをもう一度行う

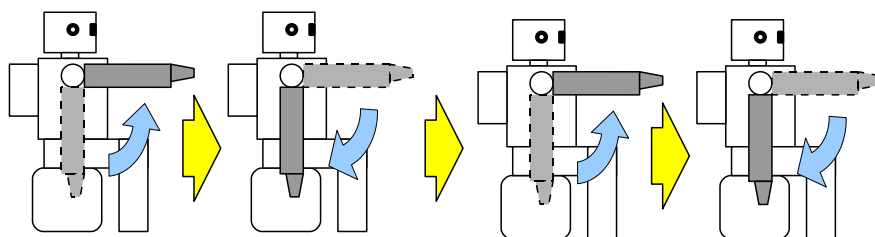


図 4.9: 実験 1・タスクの概要図

本実験では，3種類の腕の上げ下げの方法を試した．サーボの運動軌道がそれぞれ，正弦波・三角波・台形波となるように教示を行った．その後，各教示について，別々の頻度空間を用いて動きの汎化を行った．再生段階では，正弦波・三角波・台形波のそれぞれの運動軌道について，サーボの初期角度を替えて複数回の再生を行った．

#### 実験結果として

- 教示した動き

- タスクの教示により汎化された頻度空間
- ロボットの動作結果

を示す。

#### 4.3.2 実験の設定

まず、実験 1 全体に関連する設定を表 4.5 に示す。

表 4.5: 実験全体に関連する設定

|                  |         |
|------------------|---------|
| ロボットの教示・再生時間     | 10 秒    |
| サーボの角度データ取得タイミング | 0.1 秒刻み |
| サーボの角度データ取得回数    | 100 回   |
| 教示回数             | 1 回     |

ロボットの教示・再生時間を 10 秒とした。ロボットに対して動きを教示する際、時間毎のサーボの角度を取得するが、この取得タイミングを 0.1 秒毎とした。10 秒間教示を行い、0.1 秒毎にサーボの角度データを取得するので、サーボの角度データの取得回数は 100 回となる。また、教示回数は、1 回とした。運動軌道が正弦波・三角波・台形波のそれぞれの場合について 1 回ずつ教示し、それぞれ結果を出力した。運動軌道が正弦波・三角波・台形波となる理想の教示の例を図 4.10 に示す。

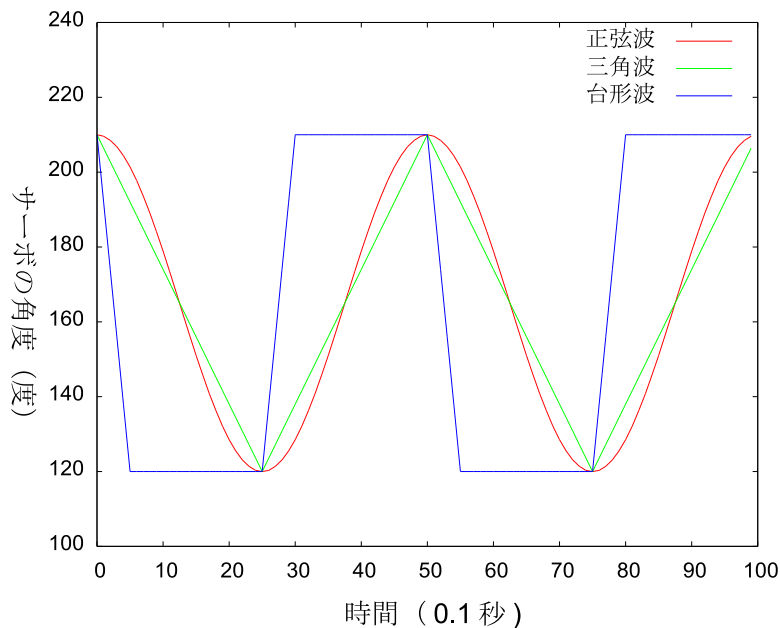


図 4.10: 理想の教示の例

動きの再生時には、正弦波・三角波・台形波のそれぞれについて複数回の再生を試した。その際、サーボの角度の初期位置を変えて実験を行った。

次に、頻度空間に関連する詳細の説明を行う。本実験における頻度空間は

1. サーボ角度軸
2. 時間軸

### 3. 選択頻度軸

からなる 3 次元空間である．頻度空間の定義に関連する設定を表 4.6 に示す．

表 4.6: 頻度空間の定義に関連する設定

|                                |         |
|--------------------------------|---------|
| サーボ角度軸におけるセルの刻み幅               | 1 度刻み   |
| サーボ角度軸の値                       | 0 ~ 239 |
| 時間軸におけるセルの刻み幅                  | 0.1 秒刻み |
| 時間軸の値                          | 0 ~ 99  |
| 中心となるセルの加算値                    | 1       |
| サーボ角度軸方向の割引率                   | 0.9     |
| サーボ角度軸方向において次の時間に選択される可能性のある距離 | 10 セル以下 |

#### サーボ角度軸におけるセルの刻み幅

パラメータ「サーボ角度軸におけるセルの刻み幅」は 1 度刻みとなっている．つまり，頻度空間はサーボ角度軸方向においては 1 度刻みのデータしか保持していない．これに対し，サーボにおける角度センサの分解能は 0.1 度刻みである．そこで，教示段階では，動きの教示時にロボットが取得した角度は小数点以下を切り捨てる．これにより，頻度空間で角度センサの値をサーボ角度軸の値として扱えるようにし，動きの汎化を行う．

また，再生時の動きを生成は，頻度空間を用いて次々と目標角度を計算することで行われる．しかし頻度空間はサーボ角度軸方向においては 1 度刻みのデータしか保持していないので，目標角度も 1 度刻みでしか計算することが出来ない．そのため，サーボにおける角度センサの分解能が 0.1 度刻みであっても 1 度刻みでしか動きの再生を行うことが出来ない点に注意する必要がある．

#### サーボの角度軸の値

上述したように，動きの教示時にロボットが取得した角度は，小数点以下を切り捨てる．そのため「サーボの角度軸の値」については，サーボにおける角度センサの取得値である 0 ~ 2400 を 1/10 した値となる．「サーボの角度軸の値」が 0 ~ 240 ではなく，0 ~ 239 となっているのは，表 4.7 のように設定したためである．サーボの角度軸の値が 239 の時の対応する角度が，1.1 度分となっている点に注意する．

表 4.7: 角度センサの値とサーボの角度軸の値の対応

| 角度センサの値     | サーボの角度軸の値 |
|-------------|-----------|
| 0 ~ 9       | 0         |
| 10 ~ 19     | 1         |
| 20 ~ 29     | 2         |
| ...         | ...       |
| 2380 ~ 2389 | 238       |
| 2390 ~ 2400 | 239       |

右腕・前後方向のサーボと頻度空間のパラメータ「サーボの角度軸の値」の関連図を図 4.11 に示す．

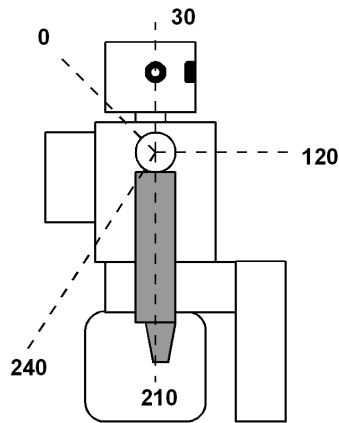


図 4.11: 右腕・前後方向のサーボと頻度空間のパラメータ「サーボの角度軸の値」の関連図

#### 中心となるセルの加算値

頻度空間の更新は、各時間ごとにある 1 つのセルを中心にして山型に更新される。その中心となるセルにおける選択頻度の加算値が「中心となるセルの加算値」というパラメータである。3.3 節の頻度空間の更新式の中では  $c$  というパラメータとして述べた。

#### サーボ角度軸方向の割引率

中心となるセル以外のセルは、中心から 1 セル離れる毎に一定の割引率で割り引かれた値が加算される。本実験では、頻度空間から時間軸と選択頻度軸を除くとサーボ角度軸のみである。そのため、選択頻度はサーボ角度軸方向に山型に更新される。この時の割引率が「サーボ角度軸方向の割引率」となる。3.3 節の頻度空間の更新式の中では  $\beta_j$  というパラメータとして述べた。

#### サーボ角度軸方向において次の時間に選択される可能性のある距離

再生段階では、現時刻のサーボの角度を元に次の時刻の目標角度を決定する。目標角度は、現時刻のサーボの角度から「一定の距離内」にあるセルの中で、最も選択頻度の高いセルが表すサーボ角度軸の値である。本実験では、頻度空間から時間軸と選択頻度軸を除くとサーボ角度軸のみである。そのため、「一定の距離内」はサーボ角度軸方向についてのみ考えるだけでよい。サーボ角度軸方向における「一定の距離内」がどのくらいの距離であるかを表したパラメータが「サーボ角度軸方向において次の時間に選択される可能性のある距離」である。例えば、ある時刻のサーボ角度軸の値が 15 だとする。このセルからサーボ角度軸方向に 10 セル以下の距離のセルが目標角度として選択される可能性がある。そのため、目標角度として選択させる可能性のあるセルは 5 ~ 25 の間となる。

### 4.3.3 結果

ここでは実験 1 の結果を示す。

## 動きの教示結果

動きをロボット（右腕・前後方向のサーボ）に教示した．教示した動きの軌道について，正弦波の軌道を図 4.12 に示す．同じように，三角波の軌道を図 4.13 に，台形波の軌道を図 4.14 に示す．

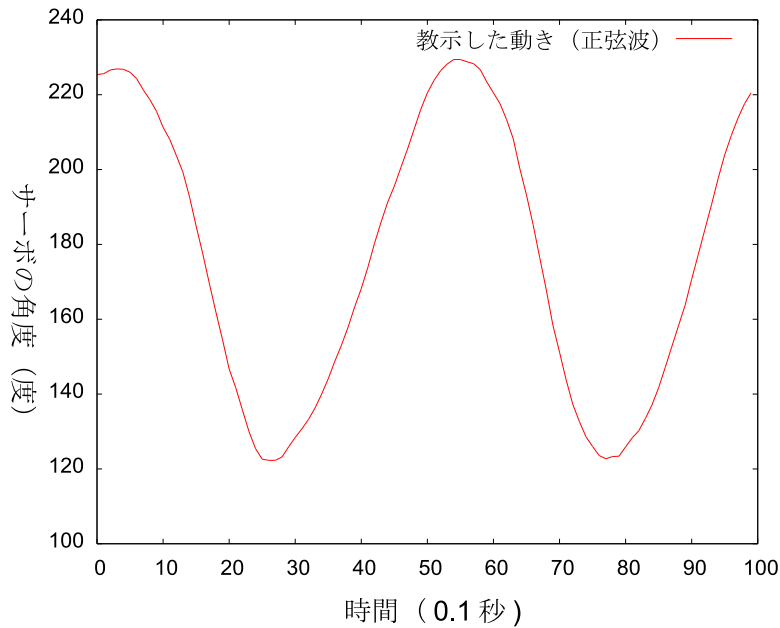


図 4.12: 教示した動き（正弦波）

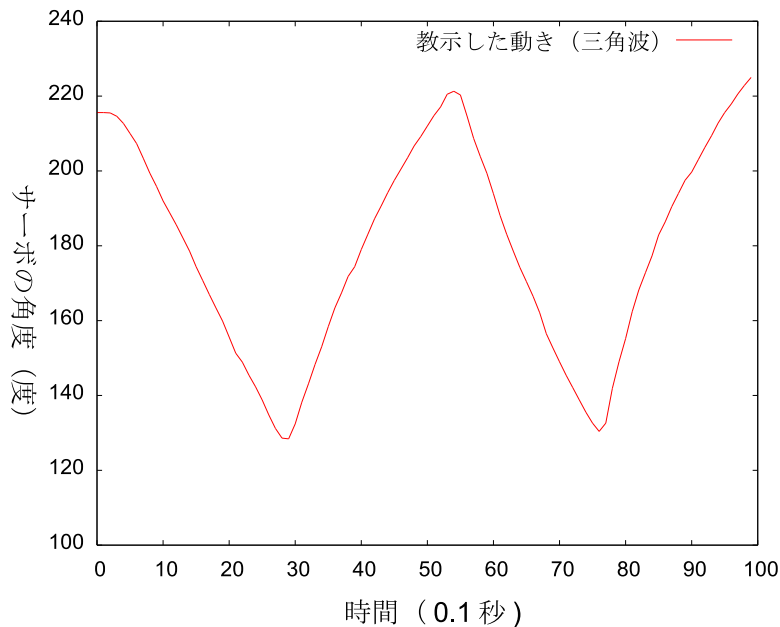


図 4.13: 教示した動き（三角波）

本実験では「右腕の上げ下げを 2 回行う」という教示を行った．この時のロボットのセンサの値は 210 度 120 度 210 度 120 度 210 度と変化するはずである（図 4.9 および図 4.5 参照）しかし図 4.13 の 50～60（0.1 秒）付近などでは，腕を下げる際に目標角度である 210 度の角度を超えている部分がある．また図 4.14 の 60～80（0.1 秒）付近などでは，腕を

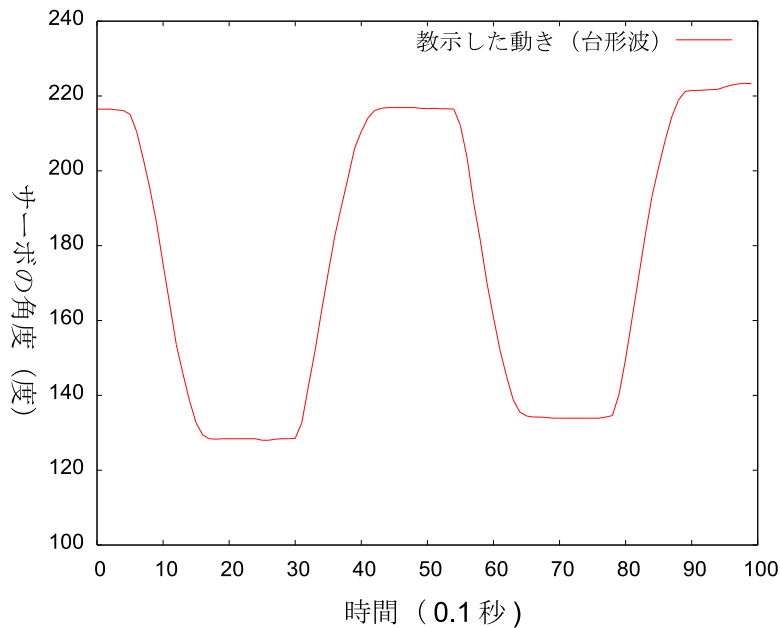


図 4.14: 教示した動き (台形波)

上げる際に目標角度である 120 度まで腕が上がりきっていない部分がある．このように目標角度を超えてしまったり，目標角度に足りない教示になってしまう原因として，「ロボットのハードウェア上の誤差」と「ユーザが教示した時に発生する誤差」が考えられる．

- ロボットのハードウェア上の誤差

例えば，ロボットの右腕が地面と水平であっても，ロボットのハードウェアに問題があり，サーボの角度の取得値が 1200 より少しずれてしまう場合がある．このような場合，たとえ正しく教示を行ったとしても，想定した値にならず，誤差が生じてしまう．これが「ロボットのハードウェア上の誤差」である．

- ユーザが教示した時に発生する誤差

ユーザは，目測で地面に対して垂直であるか，水平であることを確かめる．この時，目測を誤ると，誤差が発生する．例えば「ユーザがロボットの腕を水平まで上げきれていないにも関わらず，水平まで上げきったと勘違いして腕を下げ始める」といった場合に目標角度に足りない状態が発生する．これが「ユーザが教示した時に発生する誤差」である．

ロボットのハードウェア上の誤差については，実ロボットを用いて実験を行う限り必ず存在するものなので，本研究では問題としない．ユーザが教示した時に発生する誤差について，本実験ではユーザが教示した動きを元にロボットが正しく動作することを重視するため，ユーザの教示時のミスについては考慮しなくてもよい．

## 動きの汎化

頻度空間を用いて，教示した動きを汎化した．汎化した結果を示す．正弦波の動きの教示によって汎化された頻度空間を図 4.15 に示す．三角波の動きの教示によって汎化された頻度空間を図 4.16 に示す．台形波の動きの教示によって汎化された頻度空間を図 4.17 に示す．

図 4.12 および図 4.15 より教示した正弦波の動きを元に，頻度空間が山型に更新されていることが分かる．同じように三角波を教示した場合（図 4.13 および図 4.16 参照），台形波



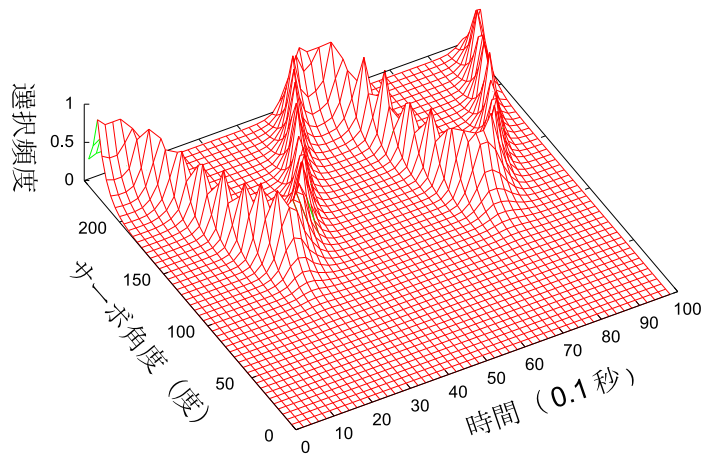


図 4.15: 汎化された頻度空間 (正弦波)

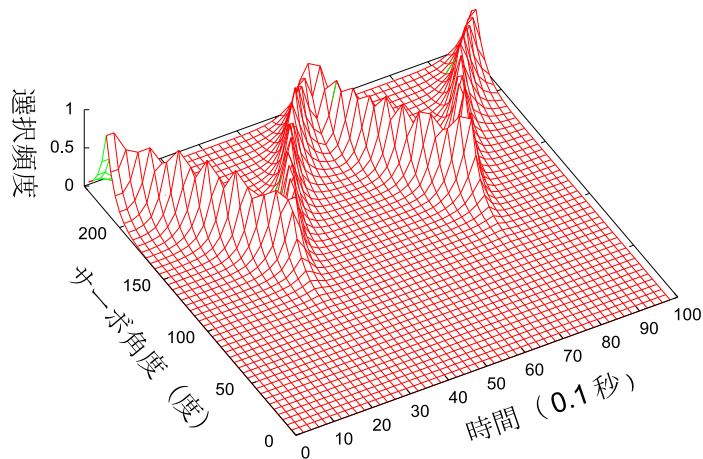


図 4.16: 汎化された頻度空間 (三角波)

を教示した場合 (図 4.14 および図 4.17 参照), のそれぞれについて, 教示した動きを元に頻度空間が山型に更新されていることが分かる.

#### 頻度空間を用いた動きの再生

汎化された頻度空間を用いて, ロボットを動作させた. 正弦波の動きを汎化した頻度空間を用いた場合の動作結果を図 4.18 に示す. 同じように, 三角波の動きを汎化した頻度空間を用いた場合の動作結果を図 4.19 に示す. 台形波の動きを汎化した頻度空間を用いた場合の動作結果を図 4.20 に示す.

図 4.18 の 10~20 (0.1 秒) 付近や, 図 4.19 の 40 (0.1 秒) 付近など, 動作結果のうちの一部が, 他の動作結果と異なる動きを示している部分がある. このような結果が出る原因とし

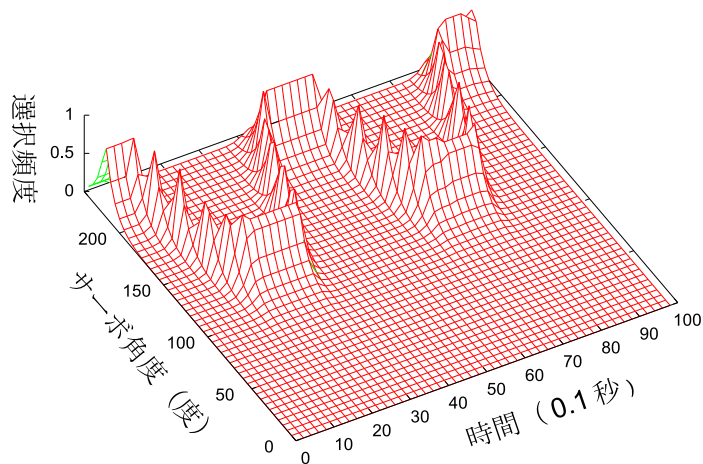


図 4.17: 汎化された頻度空間 (台形波)

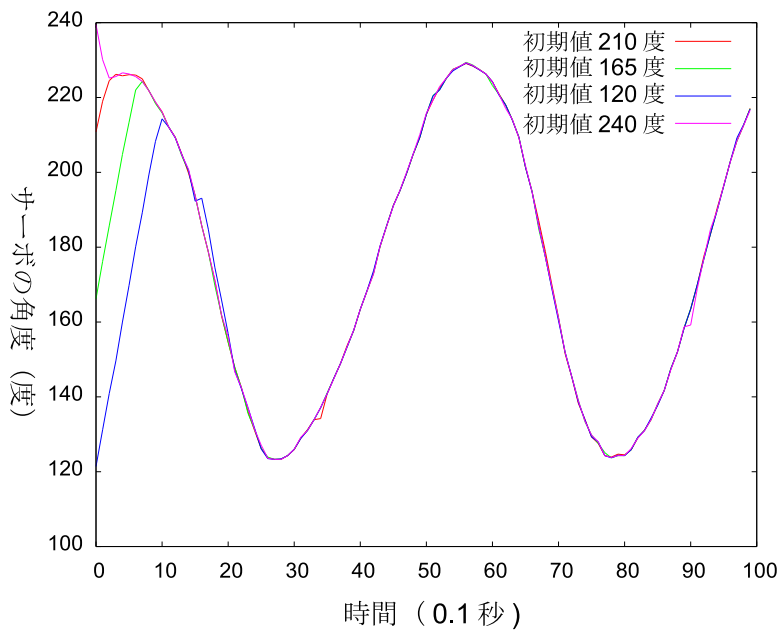


図 4.18: 動作結果 (正弦波)

て「動作生成時のセンサ値の取得ミス」と「動作結果取得時のセンサ値の取得ミス」の2つが考えられる。

- 動作生成時のセンサ値の取得ミス

これは、汎化された頻度空間を用いて動きを生成する時に発生する。ロボットは現在のサーボの角度を取得して、次の目標角度を決める。この時、現在のサーボの角度の取得を失敗すると、現在のサーボの角度を取得することが出来ない。そのため、次の目標角度を正しく決めることが出来ない。これにより、動きの生成にミスが生じる。これが「動作生成時のセンサ値の取得ミス」である。

- 動作結果取得時のセンサ値の取得ミス

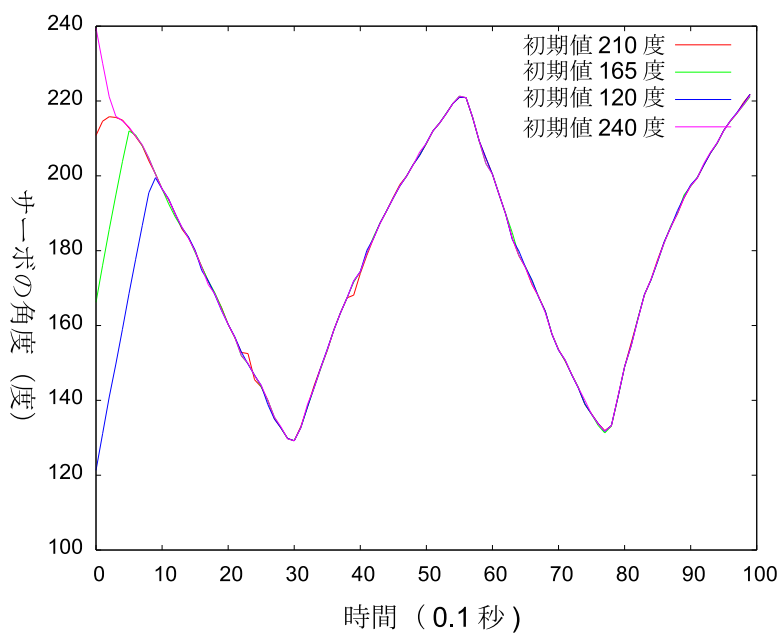


図 4.19: 動作結果 (三角波)

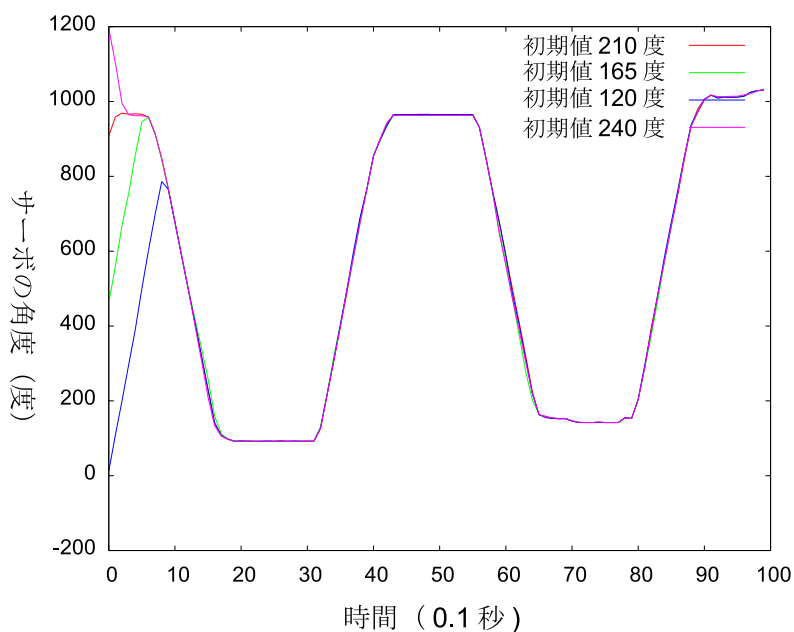


図 4.20: 動作結果 (台形波)

動作結果は、ロボット動作時の単位時間毎のサーボの角度を取得し、グラフ化している。そのため、サーボ角度の取得を失敗すると、エラー値がグラフの中に現れてしまう。これが「動作結果取得時のミス」である。

どちらの場合も原因はセンサ値の取得ミスであるため、実ロボットを用いる場合はかならず存在する問題である。問題を解決するために、エラー値と思われる取得値については除去するようにシステムに組み込んでいる。しかし、エラー値か正常値か判断を下すことの出来ない取得値も存在するため、すべてのエラー値を完全に除去するのは難しい。

動作生成時のセンサ値の取得ミスについては、サーボの角度を取得することは動きを生成するために必須である。そのため、この問題を回避することは出来ない。しかし、動作結果

取得時のセンサ値の取得ミスについては、動きの計測に関する問題であるため、サーボの角度センサの取得値を用いるのではなく、外部から人間が腕の上げる角度を測定するなどの方法で回避することが可能である。

#### 4.3.4 考察

4.12～4.14 および、4.18～4.20 より、正弦波・三角波・台形波のそれぞれについて、ほぼ教示した通りの動作を行っている事が分かる。また、サーボの角度の初期位置を変えた場合、ロボット（サーボ）は選択頻度の高い場所（ユーザが教示を行った軌道）に徐々に近づいて、最終的にはほぼユーザが教示した軌道に収束していることが分かる。

以上により、本論文で作成したシステムは、最低限の軸で構成される頻度空間において、正常に動作することが確認出来た。

### 4.4 操作デバイスのみ用いる実験

#### 4.4.1 実験の説明

##### 実験の概要と目的

本節では、作成したシステムが正常に動作することを確認するための実験について説明する。特に、再生時に操作によって動きの選択が可能であるか検証する。本実験の目的を達成するために、本節ではロボットに対して動きの教示と操作方法の教示を行い、頻度空間を用いて動きを汎化する実験を行う。また、汎化された動きを用いて動きを再生する実験を行う。

##### 用いるロボットの部位

本実験では、ロボットが持つサーボを用いて実験を行う。サーボは現在の回転角度を取得することが出来る。そのため、サーボの回転角度をシステムにおける「関節の位置」として扱う。サーボは右肩・前後方向のサーボのみを用いる。

##### 用いる操作デバイス

本実験では操作デバイスとして、4.2 節で紹介した EXTREME 3D PRO を用いて実験を行う。3軸存在するアナログ入力のうち1軸のみを用いる。

##### 実験で行うタスク

動きの教示として、ロボットの腕の上げ下げを2回行わせるタスクを教示する。教示方法は直接教示とする。直接教示とは、ロボットへ直接触れて教示する方法である。

タスクの概要を図 4.21 に示す。タスクの詳細を以下に示す。

1. ロボットは座った状態で実験を行う
2. ロボットの右腕の初期状態は、地面に対して垂直とする
3. 右腕を前方に向けて地面に水平になるまで上げる
4. 右腕を地面に垂直になるまで下げる
5. 右腕の上げ下げをもう一度行う

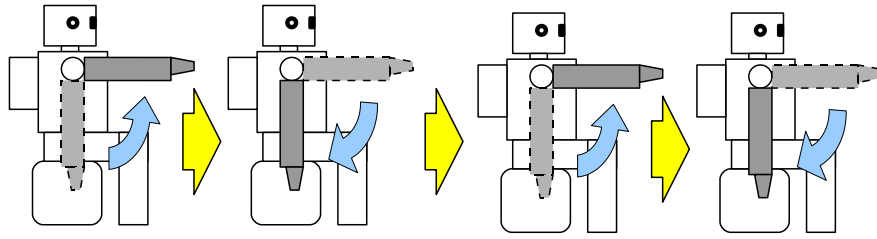


図 4.21: 実験 2・タスクの概要図

また、本実験では、操作方法の教示も行う。ユーザはロボットに対して動きの教示を行うのと同時に、操作方法の教示を行う。操作方法の教示は、再生時に、どのような操作をすると教示している動きを行うかを教示する。操作方法の教示は EXTREME 3D PRO を用いて行う。

本実験では、3種類の腕の上げ下げの方法を1回ずつ教示した。サーボの運動軌道がそれぞれ、正弦波・三角波・台形波となるように教示を行った。この際、操作方法は3回とも異なる操作を教示した。その後、3回の動きと操作方法の教示を元に、1つの頻度空間を用いて動きの汎化を行った。再生段階では、操作デバイスの入力を変えて複数回の再生を行った。

実験結果として

- 教示した動きと操作方法
- タスクの教示により汎化された頻度空間
- ロボットの動作結果

を示す。

#### 4.4.2 実験の設定

まず、実験 2 全体に関連する設定を表 4.8 に示す。

表 4.8: 実験全体に関連する設定

|                  |         |
|------------------|---------|
| ロボットの教示・再生時間     | 10 秒    |
| サーボの角度データ取得タイミング | 0.1 秒刻み |
| サーボの角度データ取得回数    | 100 回   |
| 教示回数             | 3 回     |

ロボットの教示・再生時間を 10 秒とした。ロボットに対して動きを教示する際、時間毎のサーボの角度を取得するが、この取得タイミングを 0.1 秒毎とした。10 秒間教示を行い、0.1 秒毎にサーボの角度データを取得するので、サーボの角度データの取得回数は 100 回となる。また、教示回数は、3 回とした。運動軌道が正弦波・三角波・台形波となるようにそれぞれ 1 回ずつ教示した。動きの教示と同時に、操作方法の教示も行う。本実験の目的は再生時に操作によって動きの選択が可能であるか検証することである。そのため、再生時に操作によって動きが選択出来るように操作方法の教示を行う必要がある。そこで、操作方法の教示は 3 回の教示とも操作デバイスの値が異なるように教示を行うように注意した。また、結果を分かりやすく示すために、本実験では、操作デバイスの値は出来るだけ変動しないように教示を行った。

運動軌道が正弦波・三角波・台形波となる理想の教示の例を図 4.22 に示す。

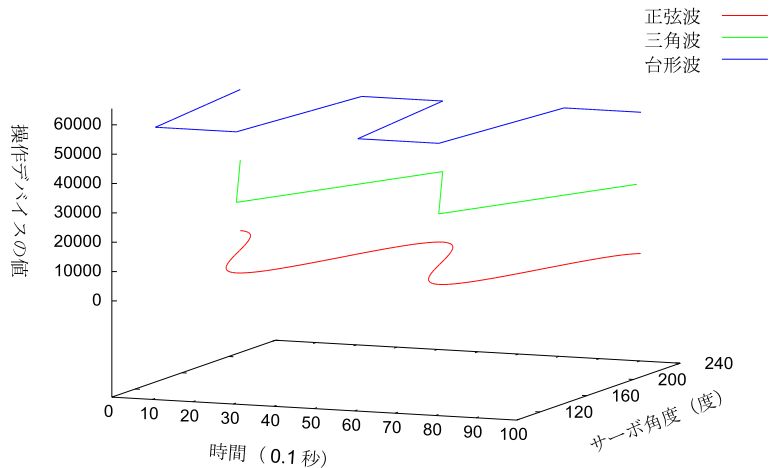


図 4.22: 理想の教示の例

動きの再生時には、操作デバイスからの入力を変えて複数回の再生を試した。次に、頻度空間に関連する詳細の説明を行う。本実験における頻度空間は

1. サーボ角度軸
2. 操作デバイス軸
3. 時間軸
4. 選択頻度軸

からなる 4 次元空間である。頻度空間の定義に関連する設定を表 4.9 に示す。

表 4.9: 頻度空間の定義に関連する設定

|                                 |         |
|---------------------------------|---------|
| サーボ角度軸におけるセルの刻み幅                | 1 度刻み   |
| サーボ角度軸の値                        | 0 ~ 239 |
| 操作デバイス軸におけるセルの刻み幅               | 4369 刻み |
| 操作デバイス軸の値                       | 0 ~ 14  |
| 時間軸におけるセルの刻み幅                   | 0.1 秒刻み |
| 時間軸の値                           | 0 ~ 99  |
| 中心となるセルの加算値                     | 1       |
| サーボ角度軸方向の割引率                    | 0.9     |
| 操作デバイス軸方向の割引率                   | 0.3     |
| サーボ角度軸方向において次の時間に選択される可能性のある距離  | 10 セル以下 |
| 操作デバイス軸方向において次の時間に選択される可能性のある距離 | 1 セル以下  |

#### サーボ角度軸におけるセルの刻み幅

パラメータ「サーボ角度軸におけるセルの刻み幅」は 1 度刻みとなっている。つまり、頻度空間はサーボ角度軸方向においては 1 度刻みのデータしか保持していない。これに対し、

サーボにおける角度センサの分解能は0.1度刻みである。そこで、教示段階では、動きの教示時にロボットが取得した角度は小数点以下を切り捨てる。これにより、頻度空間で角度センサの値をサーボ角度軸の値として扱えるようにし、動きの汎化を行う。

また、再生時の動きを生成は、頻度空間を用いて次々と目標角度を計算することで行われる。しかし頻度空間はサーボ角度軸方向においては1度刻みのデータしか保持していないので、目標角度も1度刻みでしか計算することが出来ない。そのため、サーボにおける角度センサの分解能が0.1度刻みであっても1度刻みでしか動きの再生を行うことが出来ない点に注意する必要がある。

#### サーボの角度軸の値

上述したように、動きの教示時にロボットが取得した角度は、小数点以下を切り捨てる。そのため「サーボの角度軸の値」については、サーボにおける角度センサの取得値である0～2400を1/10した値となる。「サーボの角度軸の値」が0～240ではなく、0～239となっているのは、表4.10のように設定したためである。サーボの角度軸の値が239の時の対応する角度が、1.1度分となっている点に注意する。

表 4.10: 角度センサの値とサーボの角度軸の値の対応

| 角度センサの値   | サーボの角度軸の値 |
|-----------|-----------|
| 0～9       | 0         |
| 10～19     | 1         |
| 20～29     | 2         |
| ...       | ...       |
| 2380～2389 | 238       |
| 2390～2400 | 239       |

右腕・前後方向のサーボと頻度空間のパラメータ「サーボの角度軸の値」の関連図を図4.23に示す。

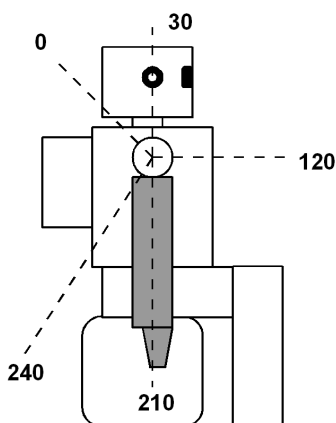


図 4.23: 右腕・前後方向のサーボと頻度空間のパラメータ「サーボの角度軸の値」の関連図

#### 操作デバイス軸におけるセルの刻み幅

パラメータ「操作デバイス軸におけるセルの刻み幅」は4369度刻みとなっている。このパラメータは、操作デバイスから取得した値を4369で割り、小数点以下を切り捨てた値が

頻度空間の操作デバイス軸におけるセルの値となっていることを示している。つまり、頻度空間は操作デバイス軸方向においては 4369 度刻みでしかデータを保持していない。そこで、教示時には上述したように、操作デバイスから取得した値を 4369 で割り、小数点以下を切り捨てた値を用いて頻度空間を更新する。再生時にも、動きの選択のために取得した操作デバイスの値は、4369 で割って小数点以下を切り捨てた値が用いられる。

刻み幅を 4369 とした理由は、EXTREME 3D PRO の取得値である 0 ~ 65535 の値をちょうど 15 段階で分割出来るためである。

### 操作デバイス軸の値

上述したように、EXTREME 3D PRO の取得値である 0 ~ 65535 の値を 15 段階で分割したため、頻度空間における操作デバイスの値は 0 ~ 14 の 15 段階となる。操作デバイスの取得値と頻度空間における操作デバイス軸の値の対応表を表 4.11 に示す。

表 4.11: 操作デバイスの取得値と頻度空間における操作デバイス軸の値の対応

| 操作デバイスの取得値    | 頻度空間における操作デバイス軸の値 |
|---------------|-------------------|
| 0 ~ 4368      | 0                 |
| 4369 ~ 8737   | 1                 |
| 8738 ~ 13106  | 2                 |
| ...           | ...               |
| 56797 ~ 61165 | 13                |
| 61166 ~ 65535 | 14                |

### 中心となるセルの加算値

頻度空間の更新は、各時間ごとにある 1 つのセルを中心にして山型に更新される。その中心となるセルにおける選択頻度の加算値が「中心となるセルの加算値」というパラメータである。3.3 節の頻度空間の更新式の中では  $c$  というパラメータとして述べた。

### 割引率

中心となるセル以外のセルは、中心から 1 セル離れる毎に一定の割引率で割り引かれた値が加算される。本実験では、頻度空間から時間軸と選択頻度軸を除くとサーボ角度軸と操作デバイス軸である。そのため、選択頻度はサーボ角度軸方向と操作デバイス軸方向に山型に更新される。この時の割引率が「サーボ角度軸方向の割引率」と「操作デバイス軸方向の割引率」なる。3.3 節の頻度空間の更新式の中では  $\beta_j$  というパラメータとして述べた。

### 次の時間に選択される可能性のある距離

再生段階において、次の時刻の目標角度を決定するために、現時刻のサーボの角度と現時刻の操作デバイスの値が用いられる。目標角度は、現時刻のサーボの角度から「一定の距離内」であり、かつ現時刻の操作デバイスの値から「一定の距離内」にあるセルの中で最も選択頻度の高いセルが選ばれる。このセルのサーボ角度軸の値が次の時刻の目標角度である。サーボ角度軸方向における「一定の距離内」がどのくらいの距離であるかを表したパラメータが、「サーボ角度軸方向において次の時間に選択される可能性のある距離」である。また、



操作デバイス軸方向における「一定の距離内」がどのくらいの距離であるかを表したパラメータが、「操作デバイス軸方向において次の時間に選択される可能性のある距離」である。

#### 4.4.3 結果

ここでは実験 2 の結果を示す。

##### 動きと操作方法の教示結果

動きをロボット（右腕・前後方向のサーボ）に教示した。また、操作デバイスである EXTREME 3D PRO を用いて操作方法についても教示した。教示した動きと操作方法の軌道を図 4.24 に示す。

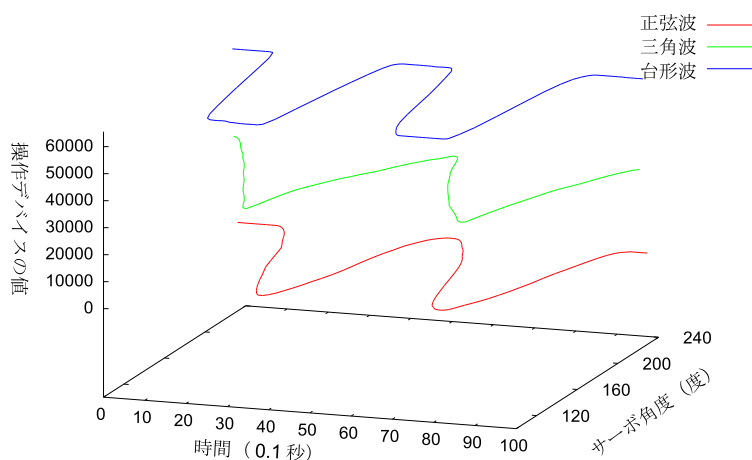


図 4.24: 教示した動きと操作方法

本実験では、操作デバイスを奥に倒しつつ、正弦波の動きの教示を行った。同じように、操作デバイスをニュートラルの位置のまま三角波の教示を行った。さらに、操作デバイスを手前に倒しつつ、台形波の教示を行った。操作デバイスは奥に倒すほど取得値が減り、手前に倒すほど取得値が増える（図 4.8 参照）。そのため、操作デバイスの値は正弦波が小さく、台形波が大きいう結果になる。

##### 動きの汎化

頻度空間を用いて、教示した動きを汎化した。汎化した結果を示す。動きの汎化の結果として、3 回の教示により汎化された頻度空間を示すが、本実験における頻度空間は、サーボ角度軸・操作デバイス軸・時間軸・選択頻度軸からなる 4 次元空間である。そのため、頻度空間のすべての結果を示すのは難しい。そこで、操作デバイス軸の値を固定し、残りのサーボ角度軸・時間軸・選択頻度軸で表される 3 次元空間を示す。

操作デバイス軸の値を 0 に固定した結果を 4.25 に示す。同じように、3 に固定した結果を 4.26 に、7 に固定した結果を 4.27 に、11 に固定した結果を 4.28 に、14 に固定した結果を 4.29 に示す。

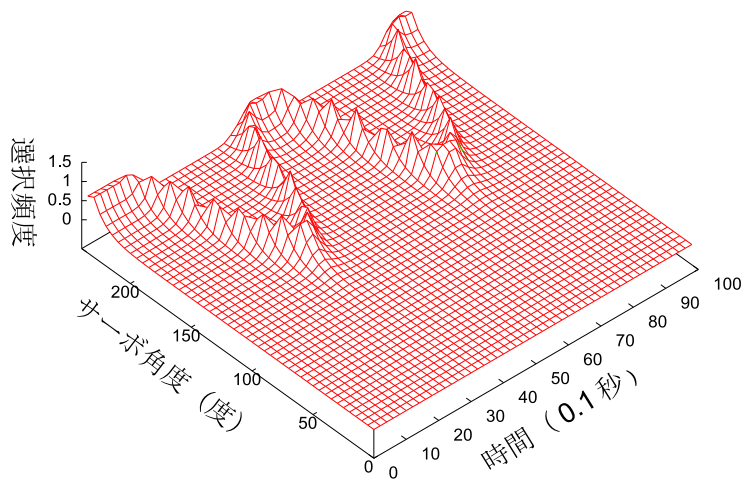


図 4.25: 汎化された頻度空間（操作デバイス軸の値を 0 で固定した空間）

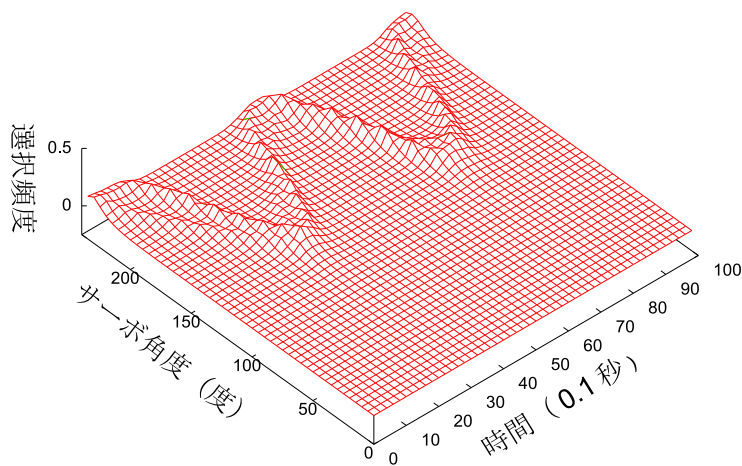


図 4.26: 汎化された頻度空間（操作デバイス軸の値を 3 で固定した空間）

操作デバイス軸の値を 0 で固定したときの頻度空間の様子である図 4.25 の結果を見ると、正弦波の形で頻度空間が更新されていることが分かる。これは、操作デバイスを奥に倒しつつ、正弦波の動きの教示を行ったためである。本実験で用いていた操作デバイスは奥に倒すほど、操作デバイスの取得値は小さくなる。その値を元に操作デバイス軸の値に変換し、頻度空間の更新を行ったため、操作デバイス軸の値が小さい場所では正弦波の動きの選択頻度が高くなっている。

操作デバイス軸の値を 7 で固定した時の頻度空間の様子である図 4.27 の結果を見ると、三角波の形で頻度空間が更新されていることが分かる。これは、操作デバイスをニュートラルのまま、三角波の動きの教示を行ったためである。操作を行っていない時の操作デバイスの取得値を操作デバイス軸の値に換算すると 7 である。そのため、操作デバイス軸の値が 7 で

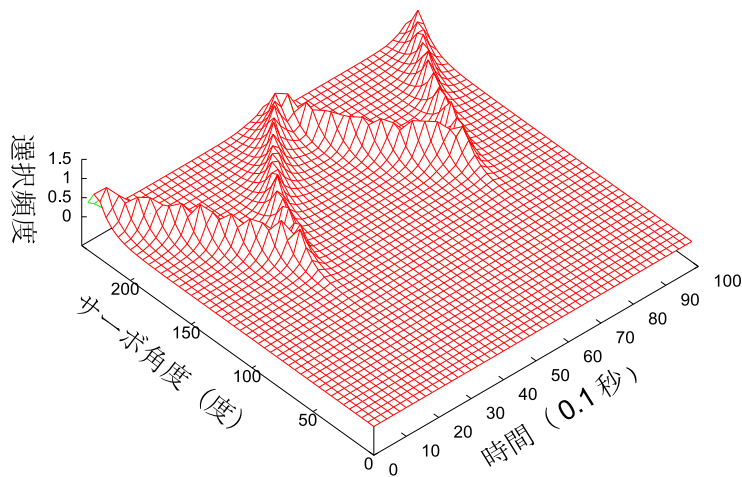


図 4.27: 汎化された頻度空間 (操作デバイス軸の値を 7 で固定した空間)

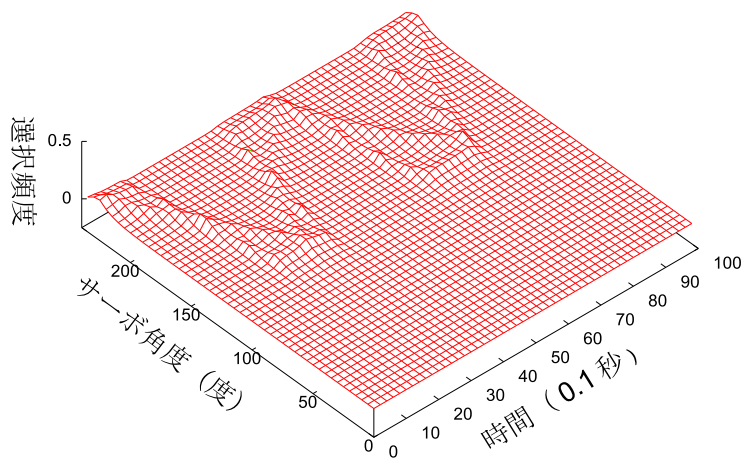


図 4.28: 汎化された頻度空間 (操作デバイス軸の値を 11 で固定した空間)

ある時には三角波の動きの選択頻度が高くなっている。

操作デバイス軸の値を 14 で固定した時の頻度空間の様子である図 4.29 の結果を見ると、台形波の形で頻度空間が更新されていることが分かる。これは、操作デバイスを手前に倒しつつ、台形波の動きの教示を行ったためである。本実験で用いていた操作デバイスは手前に倒すほど、操作デバイスの取得値は大きくなる。その値を元に操作デバイス軸の値に変換し、頻度空間の更新を行ったため、操作デバイス軸の値が大きい場所では台形波の動きの選択頻度が高くなっている。

操作デバイス軸の値を 3 で固定したときの頻度空間の様子である図 4.26 の結果を見ると、複数の動きが重なった頻度空間になっていることが分かる。この頻度空間は正弦波の動きと三角波の動きの両方の選択頻度が高い頻度空間となっている。この頻度空間は、操作デバイ

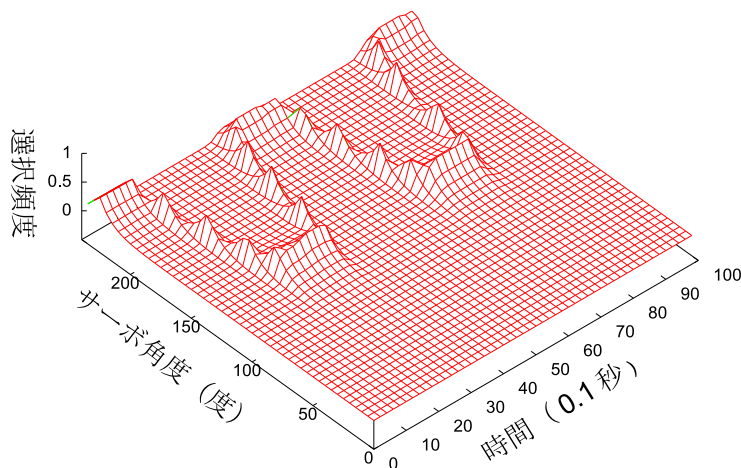


図 4.29: 汎化された頻度空間 (操作デバイス軸の値を 14 で固定した空間)

ス軸の値が 0 付近の空間と、操作デバイス軸の値が 7 付近の空間の中間にあたる空間である。操作デバイスの値が 0 付近では正弦波の動きの選択頻度が高い。また、操作デバイスの値が 7 付近では三角波の動きの選択頻度が高い。そのため、この頻度空間では両方の動きの選択頻度が上昇している。

操作デバイス軸の値を 11 で固定したときの頻度空間の様子である図 4.28 の結果を見ると、複数の動きが重なった頻度空間になっていることが分かる。この頻度空間は三角波の動きと台形波の動きの両方の選択頻度が高い頻度空間となっている。この頻度空間は、操作デバイス軸の値が 7 付近の空間と、操作デバイス軸の値が 14 付近の空間の中間にあたる空間である。操作デバイスの値が 7 付近では三角波の動きの選択頻度が高い。また、操作デバイスの値が 14 付近では台形波の動きの選択頻度が高い。そのため、この頻度空間では両方の動きの選択頻度が上昇している。

#### 頻度空間を用いた動きの再生

汎化された頻度空間を用いて、ロボットを動作させた。ユーザの操作によりロボットの動きを選択する。複数回の再生を行い、いくつかの操作をためした。図 4.30 で示した操作を行った結果を図 4.31 に示す。また、図 4.32 で示した操作を行った結果を図 4.33 に示す。図 4.34 で示した操作を行った結果を図 4.35 に示す。

図 4.35 の 60~70 (0.1 秒) 付近で、突然正弦波の動きとは関係ない動きをしている箇所がある。このような結果が出る原因として「動作生成時のセンサ値の取得ミス」と「他の動きに引き込まれた」場合の 2 つが考えられる。

- 動作生成時のセンサ値の取得ミス

これは、汎化された頻度空間を用いて動きを生成する時に発生する。ロボットは現在のサーボの角度と現在の操作デバイスの値を取得して、次の目標角度を決める。この時、現在のサーボの角度の取得を失敗すると、現在のサーボの角度を取得することが出来ない。そのため、次の目標角度を正しく決めることが出来ない。これにより、動きの生成にミスが生じる。これが「動作生成時のセンサ値の取得ミス」である。

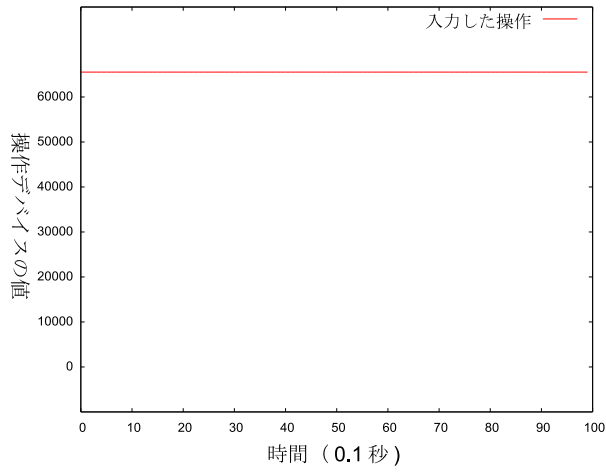


図 4.30: ロボットに対して行った操作

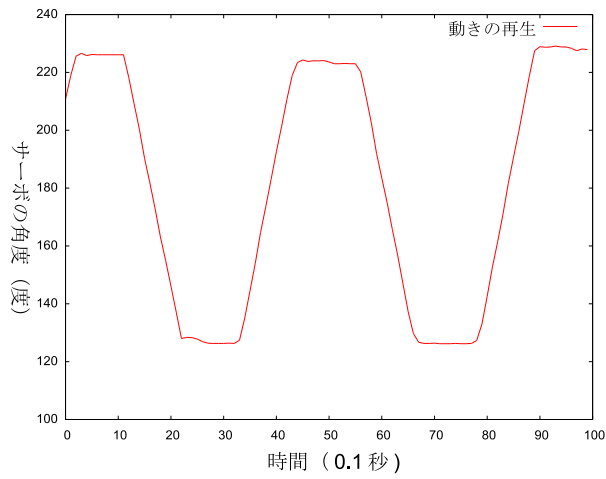


図 4.31: 台形波の動きを選択したときの動作結果

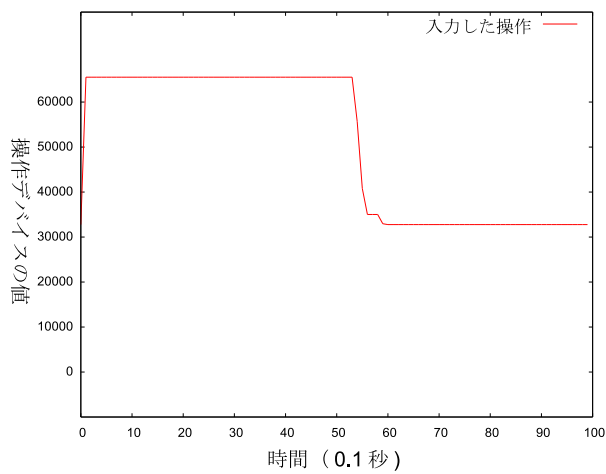


図 4.32: ロボットに対して行った操作

- 他の動きに引き込まれた

図 4.35 の 60~70 (0.1 秒) 付近の例を考える。この時、たまたま正弦波の動きを表す

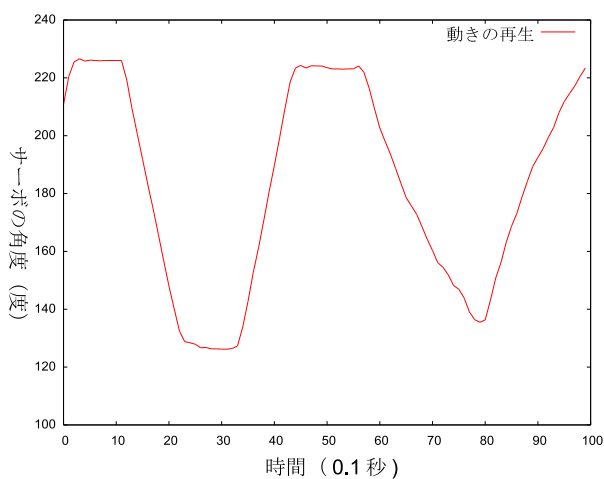


図 4.33: 台形波の動きから三角波の動きへ変更した時の動作結果

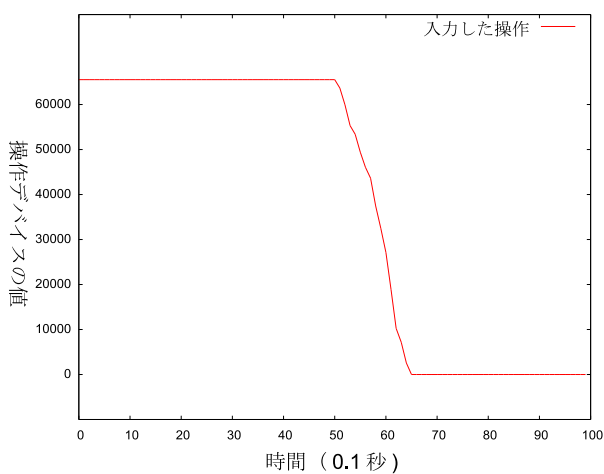


図 4.34: ロボットに対して行った操作

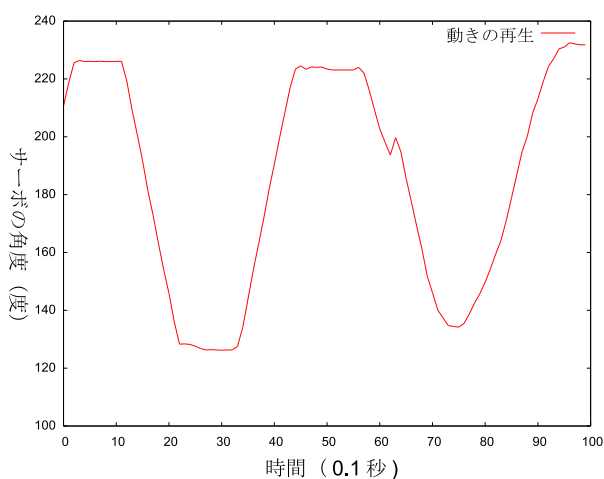


図 4.35: 台形波の動きから正弦波の動きへ変更した時の動作結果

セルの周辺のセルに選択頻度が高いセルがあり、そのセルの動きに引き込まれてしまった可能性がある。このように、選択したい動きと異なる動きに引き込まれる可能性が

ある．

動作生成時のセンサ値の取得ミスについては，原因はセンサ値の取得ミスであるため，実ロボットを用いる場合はかならず存在する問題である．問題を解決するために，エラー値と思われる取得値については除去するようにシステムに組み込んでいる．しかし，エラー値が正常値か判断を下すことの出来ない取得値も存在するため，すべてのエラー値を完全に除去するのは難しい．

他の動きに引き込まれてしまう問題については，ユーザが追加で教示を行い，行ってほしい動きの選択頻度を上げる方法が考えられる．

#### 4.4.4 考察

本論文で作成したシステムを用いることにより，動きの教示と操作方法の教示を行い，動きの汎化を行う．これにより，汎化された動きを操作することで，動きを変化させることが可能であることを示した．

## 4.5 センサのみ用いる実験

### 4.5.1 実験の説明

#### 実験の概要と目的

本節では，作成したシステムが正常に動作することを確認するための実験について説明する．特に，再生時にロボットの置かれている状況によって自動的に動きを変化させることが可能であるか検証する．本実験の目的を達成するために，本節ではロボットに対して動きの教示を行う．この際，ロボットは置かれている状況を認識し，動きの汎化に用いる．ここでは，頻度空間を用いて動きを汎化する実験を行う．また，汎化された動きを用いて動きを再生する実験を行う．

#### 用いるロボットの部位

本実験では，ロボットが持つサーボを用いて実験を行う．サーボは現在の回転角度を取得することが出来る．そのため，サーボの回転角度をシステムにおける「関節の位置」として扱う．サーボは右肩・前後方向のサーボのみを用いる．また，ロボットが持つGセンサも用いる．Gセンサにより，ロボットは自身が置かれている状況を認識する．

#### 実験で行うタスク

動きの教示として，ロボットの腕の上げ下げを2回行わせるタスクを教示する．教示方法は直接教示とする．直接教示とは，ロボットへ直接接触して教示する方法である．

タスクの概要を図 4.36 に示す．タスクの詳細を以下に示す．

1. ロボットは座った状態で実験を行う
2. ロボットの右腕の初期状態は，地面に対して垂直とする
3. 右腕を前方に向けて地面に水平になるまで上げる
4. 右腕を地面に垂直になるまで下げる
5. 右腕の上げ下げをもう一度行う

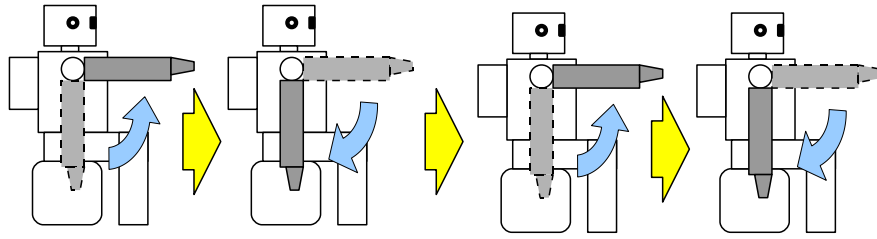


図 4.36: 実験 3・タスクの概要図

本実験では、3種類の腕の上げ下げの方法を1回ずつ教示した。この際、ロボットを前後方向（うつぶせ、仰向けの方向）に傾け、ロボットが置かれている状況を変えて3回の教示を行った。その後、3回の動きと教示されたときにロボットが置かれていた状況（Gセンサの値）を元に、1つの頻度空間を用いて動きの汎化を行った。再生段階では、ロボットが置かれている状況（うつぶせ状態か、仰向け状態か等）を変えて複数回の再生を行った。

実験結果として

- 教示した動きとそのときのロボットが置かれている状況（Gセンサの値）
- タスクの教示により汎化された頻度空間
- ロボットの動作結果

を示す。

#### 4.5.2 実験の設定

まず、実験 3 全体に関連する設定を表 4.12 に示す。

表 4.12: 実験全体に関連する設定

|                  |         |
|------------------|---------|
| ロボットの教示・再生時間     | 10 秒    |
| サーボの角度データ取得タイミング | 0.1 秒刻み |
| サーボの角度データ取得回数    | 100 回   |
| 教示回数             | 3 回     |

ロボットの教示・再生時間を 10 秒とした。ロボットに対して動きを教示する際、時間毎のサーボの角度を取得するが、この取得タイミングを 0.1 秒毎とした。10 秒間教示を行い、0.1 秒毎にサーボの角度データを取得するので、サーボの角度データの取得回数は 100 回となる。また、教示回数は、3 回とした。運動軌道が正弦波・三角波・台形波となるようにそれぞれ 1 回ずつ教示した。本実験の目的は、再生時にロボットが置かれている状況によって動きの選択が可能であるか検証することである。そのために、3 回の教示を行う時には、3 回の教示ともロボットが置かれている状況（Gセンサの値）が異なるようにして教示を行った。また、結果を分かりやすく示すために、本実験では、Gセンサの値は教示中は出来るだけ変動しないように注意して教示を行った。

運動軌道が正弦波・三角波・台形波となる理想の教示の例を図 4.37 に示す。

動きの再生時には、ロボットが置かれている状況を変えて複数回の再生を試した。

次に、頻度空間に関連する詳細の説明を行う。本実験における頻度空間は

##### 1. サーボ角度軸



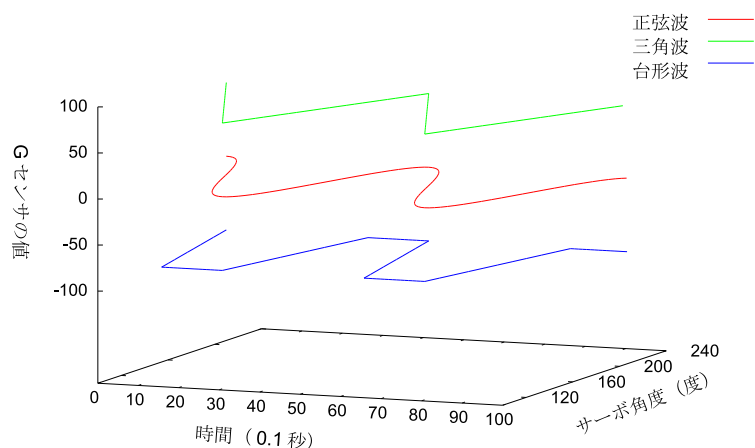


図 4.37: 理想の教示の例

2. G センサ軸
3. 時間軸
4. 選択頻度軸

からなる 4 次元空間である．頻度空間の定義に関連する設定を表 4.13 に示す．

表 4.13: 頻度空間の定義に関連する設定

|                                |         |
|--------------------------------|---------|
| サーボ角度軸におけるセルの刻み幅               | 1 度刻み   |
| サーボ角度軸の値                       | 0 ~ 239 |
| G センサ軸におけるセルの刻み幅               | 10 刻み   |
| G センサ軸の値                       | 0 ~ 19  |
| 時間軸におけるセルの刻み幅                  | 0.1 秒刻み |
| 時間軸の値                          | 0 ~ 99  |
| 中心となるセルの加算値                    | 1       |
| サーボ角度軸方向の割引率                   | 0.9     |
| G センサ軸方向の割引率                   | 0.5     |
| サーボ角度軸方向において次の時間に選択される可能性のある距離 | 10 セル以下 |
| G センサ軸方向において次の時間に選択される可能性のある距離 | 1 セル以下  |

#### サーボ角度軸におけるセルの刻み幅

パラメータ「サーボ角度軸におけるセルの刻み幅」は 1 度刻みとなっている．つまり，頻度空間はサーボ角度軸方向においては 1 度刻みのデータしか保持していない．これに対し，サーボにおける角度センサの分解能は 0.1 度刻みである．そこで，教示段階では，動きの教示時にロボットが取得した角度は小数点以下を切り捨てる．これにより，頻度空間で角度センサの値をサーボ角度軸の値として扱えるようにし，動きの汎化を行う．

また、再生時の動きを生成は、頻度空間を用いて次々と目標角度を計算することで行われる。しかし頻度空間はサーボ角度軸方向においては1度刻みのデータしか保持していないので、目標角度も1度刻みでしか計算することが出来ない。そのため、サーボにおける角度センサの分解能が0.1度刻みであっても1度刻みでしか動きの再生を行うことが出来ない点に注意する必要がある。

#### サーボの角度軸の値

上述したように、動きの教示時にロボットが取得した角度は、小数点以下を切り捨てる。そのため「サーボの角度軸の値」については、サーボにおける角度センサの取得値である0~2400を1/10した値となる。「サーボの角度軸の値」が0~240ではなく、0~239となっているのは、表4.14のように設定したためである。サーボの角度軸の値が239の時の対応する角度が、1.1度分となっている点に注意する。

表 4.14: 角度センサの値とサーボの角度軸の値の対応

| 角度センサの値   | サーボの角度軸の値 |
|-----------|-----------|
| 0~9       | 0         |
| 10~19     | 1         |
| 20~29     | 2         |
| ...       | ...       |
| 2380~2389 | 238       |
| 2390~2400 | 239       |

右腕・前後方向のサーボと頻度空間のパラメータ「サーボの角度軸の値」の関連図を図4.38に示す。

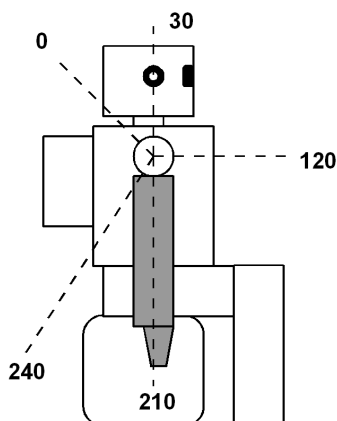


図 4.38: 右腕・前後方向のサーボと頻度空間のパラメータ「サーボの角度軸の値」の関連図

#### G センサ軸におけるセルの刻み幅

パラメータ「G センサ軸におけるセルの刻み幅」は10刻みとなっている。このパラメータは、G センサから取得した値を10で割り、小数点以下を切り捨てた値が頻度空間のG センサ軸におけるセルの値となっていることを示している。つまり、頻度空間はG センサ軸方向においては10刻みでしかデータを保持していない。そこで、教示時には上述したよう

に，G センサから取得した値を 10 で割り，小数点以下を切り捨てた値を用いて頻度空間を更新する．再生時にも，動きの選択のために取得した G センサの値は，10 で割って小数点以下を切り捨てた値が用いられる．

## G センサ軸の値

ロボットが持つ G センサの取得値と頻度空間における G センサ軸の値の対応表を表 4.15 に示す．

表 4.15: G センサの取得値と頻度空間における G センサ軸の値の対応

| G センサの取得値  | 頻度空間における G センサ軸の値 |
|------------|-------------------|
| -477 ~ -91 | 0                 |
| -90 ~ -81  | 1                 |
| -80 ~ -71  | 2                 |
| ...        | ...               |
| 80 ~ 89    | 18                |
| 90 ~ 477   | 19                |

G センサ軸におけるセルの刻み幅が 10 刻みであるにも関わらず，頻度空間における G センサ軸の値が 0 の時と 19 の時には 10 刻みになっていない．これは，本実験では，G センサの取得値はほとんどが -90 ~ 90 の範囲に収まるためである．本実験は G センサの値によりロボットの動きを変化させることが可能であるか検証することが目的である．そのため，使用頻度の少ない G センサの値については省略する．

## 中心となるセルの加算値

頻度空間の更新は，各時間ごとにある 1 つのセルを中心にして山型に更新される．その中心となるセルにおける選択頻度の加算値が「中心となるセルの加算値」というパラメータである．3.3 節の頻度空間の更新式の中では  $c$  というパラメータとして述べた．

## 割引率

中心となるセル以外のセルは，中心から 1 セル離れる毎に一定の割引率で割り引かれた値が加算される．本実験では，頻度空間から時間軸と選択頻度軸を除くとサーボ角度軸と G センサ軸である．そのため，選択頻度はサーボ角度軸方向と G センサ軸方向に山型に更新される．この時の割引率が「サーボ角度軸方向の割引率」と「G センサ軸方向の割引率」なる．3.3 節の頻度空間の更新式の中では  $\beta_j$  というパラメータとして述べた．

## 次の時間に選択される可能性のある距離

再生段階において，次の時刻の目標角度を決定するために，現時刻のサーボの角度と現時刻の G センサの値が用いられる．目標角度は，現時刻のサーボの角度から「一定の距離内」であり，かつ現時刻の G センサの値から「一定の距離内」にあるセルの中で最も選択頻度の高いセルが選ばれる．このセルのサーボ角度軸の値が次の時刻の目標角度である．サーボ角度軸方向における「一定の距離内」がどのくらいの距離であるかを表したパラメータが，「サーボ角度軸方向において次の時間に選択される可能性のある距離」である．また，G セ

ンサ軸方向における「一定の距離内」がどのくらいの距離であるかを表したパラメータが、「G センサ軸方向において次の時間に選択される可能性のある距離」である。

### 4.5.3 結果

ここでは実験 3 の結果を示す。

#### 動きと操作方法の教示結果

動きをロボット（右腕・前後方向のサーボ）に教示した。本実験では、ロボットを椅子に座らせた状態で、正弦波の動きの教示を行った。同じように、ロボットをうつぶせ方向に傾けつつ、三角波の教示を行った。さらに、ロボット仰向け方向に傾けつつ、台形波の教示を行った。教示した動きと教示時の G センサの値の軌道を図 4.39 に示す。

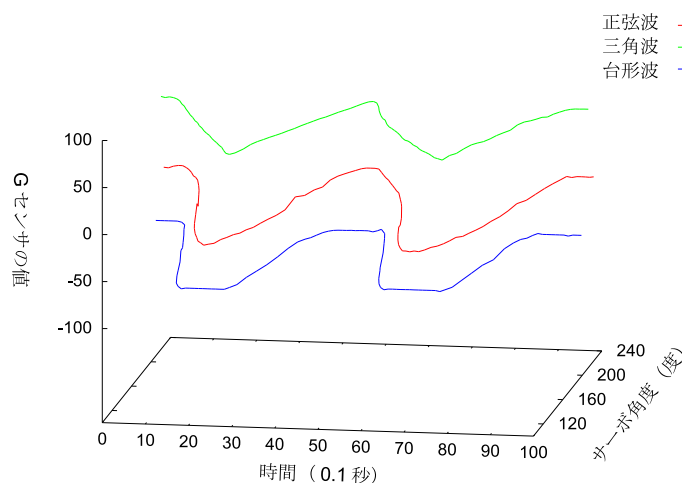


図 4.39: 教示した動きと教示時の G センサの値

#### 動きの汎化

頻度空間を用いて、教示した動きを汎化した。汎化した結果を示す。動きの汎化の結果として、3 回の教示により汎化された頻度空間を示すが、本実験における頻度空間は、サーボ角度軸・G センサ軸・時間軸・選択頻度軸からなる 4 次元空間である。そのため、頻度空間のすべての結果を示すのは難しい。そこで、G センサ軸の値を固定し、残りのサーボ角度軸・時間軸・選択頻度軸で表される 3 次元空間を示す。

G センサ軸の値を 4 に固定した結果を 4.40 に示す。同じように、6 に固定した結果を 4.41 に、9 に固定した結果を 4.42 に、12 に固定した結果を 4.43 に、16 に固定した結果を 4.44 に示す。

G センサ軸の値を 4 で固定したときの頻度空間の様子である図 4.40 の結果を見ると、台形波の形で頻度空間が更新されていることが分かる。これは、ロボットを仰向け方向に傾けつつ、台形波の動きの教示を行ったためである。本実験で用いた G センサは、ロボットを仰向け方向に傾けるほど、取得値は小さくなる。その値を元に G センサ軸の値に変換し、頻

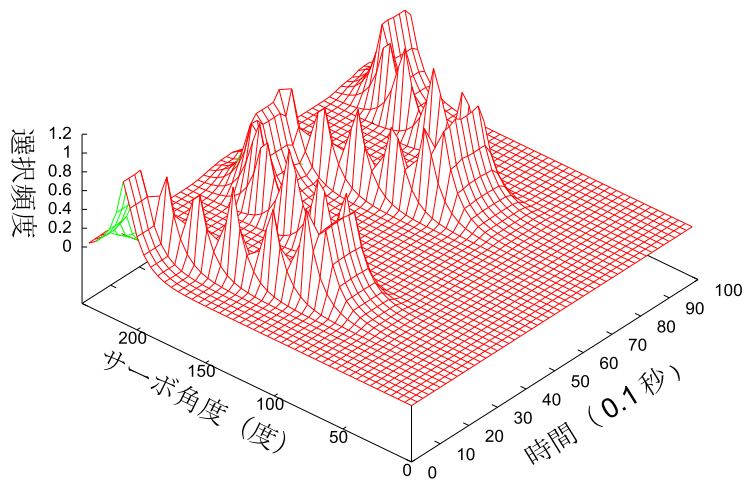


図 4.40: 汎化された頻度空間 (G センサ軸の値を 4 で固定した空間)

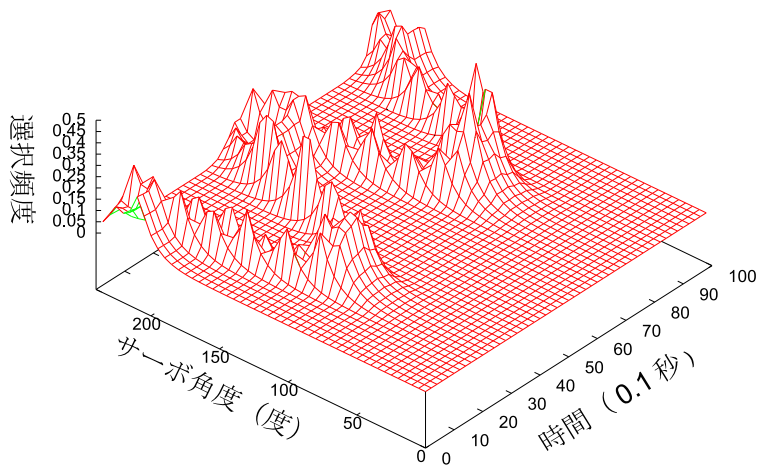


図 4.41: 汎化された頻度空間 (G センサ軸の値を 6 で固定した空間)

度空間の更新を行ったため、G センサ軸の値が小さい場所では台形波の動きの選択頻度が高くなっている。

G センサ軸の値を 9 で固定した時の頻度空間の様子である図 4.42 の結果を見ると、正弦波の形で頻度空間が更新されていることが分かる。これは、ロボットを椅子に座らせて、正弦波の動きの教示を行ったためである。ロボットの上体が地面に対して垂直の場合、G センサの取得値は 0 前後となる。この値を頻度空間における G センサ軸の値に変換すると、9 または 10 付近の値をとることになる。そのため、G センサ軸の値が 9 である時には正弦波の動きの選択頻度が高くなっている。

G センサ軸の値を 16 で固定した時の頻度空間の様子である図 4.44 の結果を見ると、三角波の形で頻度空間が更新されていることが分かる。これは、ロボットをうつぶせ方向に傾け

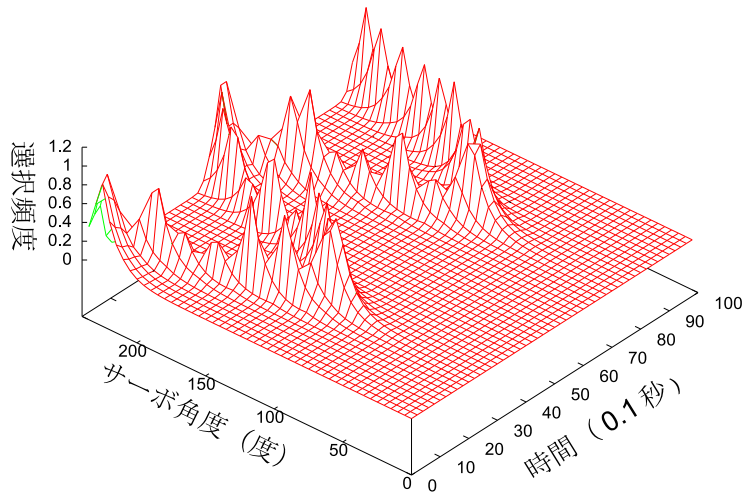


図 4.42: 汎化された頻度空間 (G センサ軸の値を 9 で固定した空間)

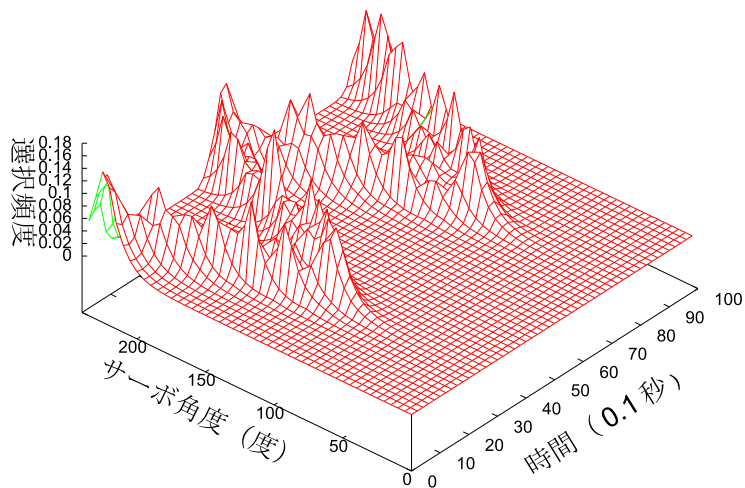


図 4.43: 汎化された頻度空間 (G センサ軸の値を 12 で固定した空間)

つつ、三角波の動きの教示を行ったためである。本実験で用いた G センサは、ロボットをうつぶせ方向に傾けるほど、取得値は大きくなる。その値を元に頻度空間における G センサ軸の値に変換し、頻度空間の更新を行ったため、G センサ軸の値が大きい場所では台形波の動きの選択頻度が高くなっている。

操作デバイス軸の値を 6 で固定したときの頻度空間の様子である図 4.41 の結果を見ると、図 4.40 や図 4.42 よりも複雑な頻度空間になっていることが分かる。ここでの複雑な頻度空間とは、どのような動きの教示により更新されたかが分かりにくい頻度空間のことか、または再生時にどのような動きが生成されるのか予想が付きにくい頻度空間のこととする。この頻度空間は台形波の動きと正弦波の動きの両方の選択頻度が高い頻度空間となっている。この頻度空間は、G センサ軸の値が 4 付近の空間と、G センサ軸の値が 9 付近の空間の中間に

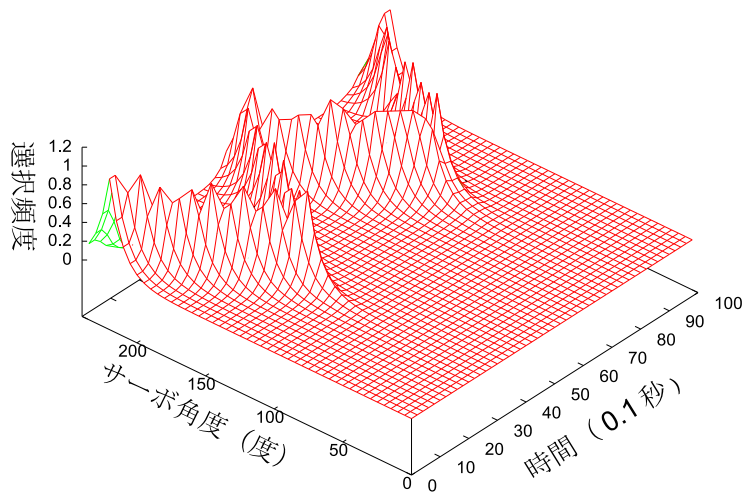


図 4.44: 汎化された頻度空間 (G センサ軸の値を 16 で固定した空間)

あたる空間である。G センサの値が 4 付近では台形波の動きの選択頻度が高い。また、G センサの値が 9 付近では正弦波の動きの選択頻度が高い。それにより、この頻度空間では両方の動きの選択頻度が上昇していると予想できる。

同じように G センサ軸の値を 16 で固定したときの頻度空間の様子である図 4.44 の結果を見ると、複雑な頻度空間になっていることが分かる。この頻度空間は三角波の動きと正弦波の動きの両方の選択頻度が高い頻度空間となっている。この頻度空間は、G センサ軸の値が 9 付近の空間と、G センサ軸の値が 16 付近の空間の中間にあたる空間である。G センサの値が 9 付近では正弦波の動きの選択頻度が高い。また、G センサの値が 16 付近では三角波の動きの選択頻度が高い。そのため、この頻度空間では両方の動きの選択頻度が上昇していると予想できる。

#### 頻度空間を用いた動きの再生

汎化された頻度空間を用いて、ロボットを動作させた。G センサの値によりロボットは動きを選択する。ロボットが置かれている状況 (G センサの値) を変えつつ、複数回の再生を行った。図 4.45 で示した G センサの値の変化によりロボットが動作した結果を図 4.46 に示す。また、図 4.47 で示した G センサの値の変化によりロボットが動作した結果を図 4.48 に示す。図 4.49 で示した G センサの値の変化によりロボットが動作した結果を図 4.50 に示す。

図 4.45 においては、ロボットの状態 (G センサの値) を意図的に変えている場所は無く、図 4.47 および図 4.49 においてもロボットの状態 (G センサの値) を意図的に変えているのは 5 秒付近のみである。それ以外の G センサの小さな値の変化は意図的なものではないので、なぜ起こるのか検証し、改善が必要であると思われる。この原因は以下の 2 つが考えられる。

- ロボットが発する振動

ロボットがサーボを動かす際、ある程度の振動が発生してしまう。これについてはロボットの構造上の問題であり、実ロボットを扱うかぎり回避するのは難しい。

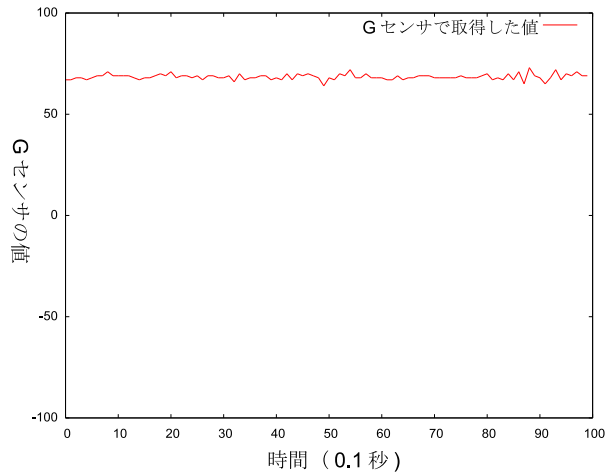


図 4.45: ロボットが取得した G センサの値

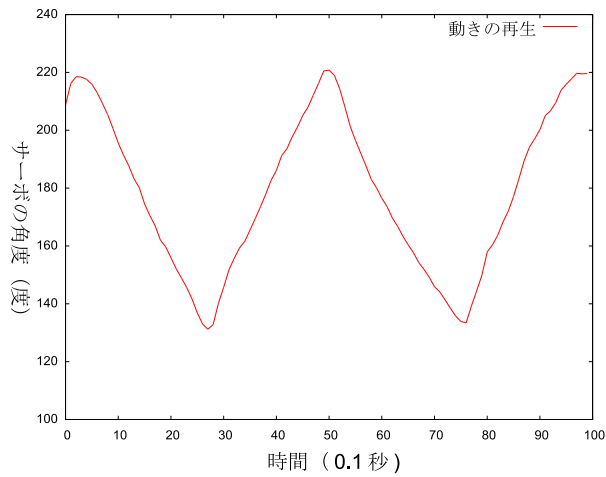


図 4.46: G センサの値が図 4.45 のように変化した時の動作結果

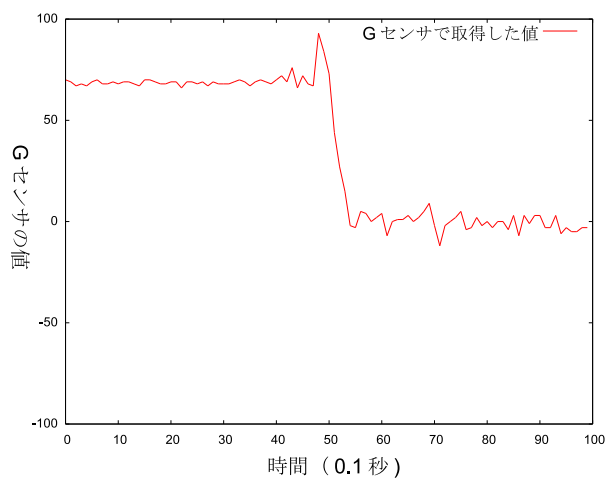


図 4.47: ロボットが取得した G センサの値

- 環境が発する振動

ロボット以外の外部要因により振動が発生し、ロボットに伝わってしまう可能性があ



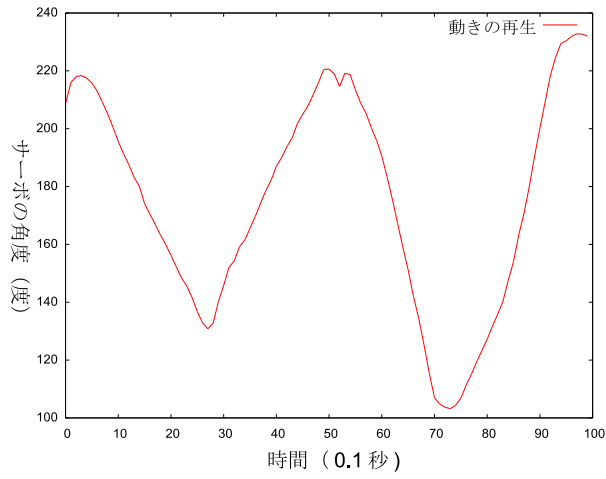


図 4.48: G センサの値が図 4.47 のように変化した時の動作結果

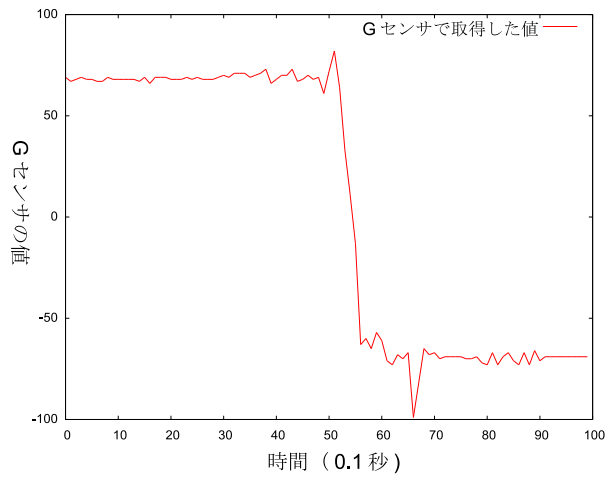


図 4.49: ロボットが取得した G センサの値

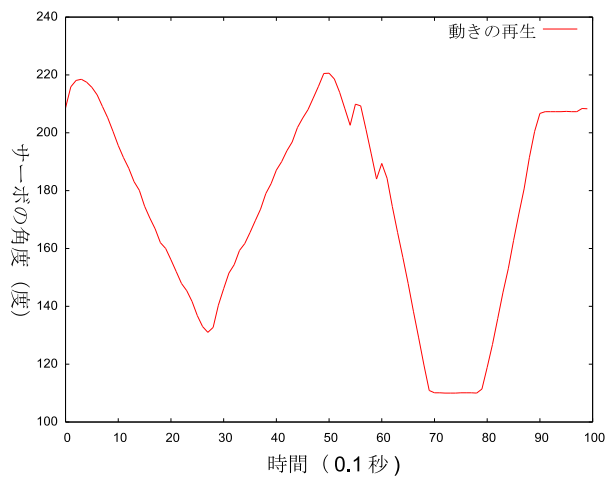


図 4.50: G センサの値が図 4.49 のように変化した時の動作結果

る．これを解決するためには，外部の振動が伝わりにくい環境を構築し，実験を行う必要がある．

図 4.48 の 50 (0.1 秒) 付近や 4.50 の 50~60 (0.1 秒) 付近で、突然想定していない動き (腕の上げ下げタスクと関係の無い動き) をしている箇所がある。このような結果が出る原因として「動作生成時のセンサ値の取得ミス」と「他の動きに引き込まれた」場合の 2 つが考えられる。

- 動作生成時のセンサ値の取得ミス

これは、汎化された頻度空間を用いて動きを生成する時に発生する。ロボットは現在のサーボの角度と現在の操作デバイスの値を取得して、次の目標角度を決める。この時、現在のサーボの角度の取得を失敗すると、現在のサーボの角度を取得することが出来ない。そのため、次の目標角度を正しく決めることが出来ない。これにより、動きの生成にミスが生じる。これが「動作生成時のセンサ値の取得ミス」である。

- 他の動きに引き込まれた

現在行っている動きとは関係が無いセルの中に、選択頻度が高いセルがあり、そのセルの動きに引き込まれてしまった可能性がある。

動作生成時のセンサ値の取得ミスについては、原因はセンサ値の取得ミスであるため、実ロボットを用いる場合はかならず存在する問題である。問題を解決するために、エラー値と思われる取得値については除去するようにシステムに組み込んでいる。しかし、エラー値か正常値か判断を下すことの出来ない取得値も存在するため、すべてのエラー値を完全に除去するのは難しい。

他の動きに引き込まれてしまう問題については、ユーザが追加で教示を行い、行ってほしい動きの選択頻度を上げる方法が考えられる。

#### 4.5.4 考察

本論文で作成したシステムを用いることにより、動きの教示とロボットが置かれている状態に関する情報 (本節では G センサの値) から、動きの汎化を行う。これにより、汎化された動きを元に、ロボットが動きを変化させることが可能であることを示した。

## 4.6 センサ、操作デバイスを用いる実験

### 4.6.1 実験の説明

#### 実験の概要と目的

本節では、作成したシステムが正常に動作することを確認するための実験について説明する。特に、再生時にユーザによる操作とロボットの置かれている状況によって動きの選択を行うことが出来るか検証する。本実験の目的を達成するために、本節ではロボットに対して動きの教示と操作方法の教示を行う。頻度空間を用いてこれらの教示を元に動きの汎化を行う。この際、ロボットは置かれている状況を認識し、これも動きの汎化に用いる。また、汎化された動きを用いて動きを再生する実験を行う。

#### 用いるロボットの部位

本実験では、ロボットが持つサーボを用いて実験を行う。サーボは現在の回転角度を取得することが出来る。そのため、サーボの回転角度をシステムにおける「関節の位置」として扱う。サーボは右肩・前後方向のサーボのみを用いる。また、ロボットが持つ G センサも用いる。G センサにより、ロボットは自身が置かれている状況を認識する。

## 用いる操作デバイス

本実験では操作デバイスとして、4.2 節で紹介した EXTREME 3D PRO を用いて実験を行う。3 軸存在するアナログ入力のうち 1 軸のみを用いる。

## 実験で行うタスク

動きの教示として、ロボットの腕の上げ下げを 2 回行わせるタスクを教示する。教示方法は直接教示とする。直接教示とは、ロボットへ直接触れて教示する方法である。

タスクの概要を図 4.51 に示す。タスクの詳細を以下に示す。

1. ロボットは座った状態で実験を行う
2. ロボットの右腕の初期状態は、地面に対して垂直とする
3. 右腕を前方に向けて地面に水平になるまで上げる
4. 右腕を地面に垂直になるまで下げる
5. 右腕の上げ下げをもう一度行う

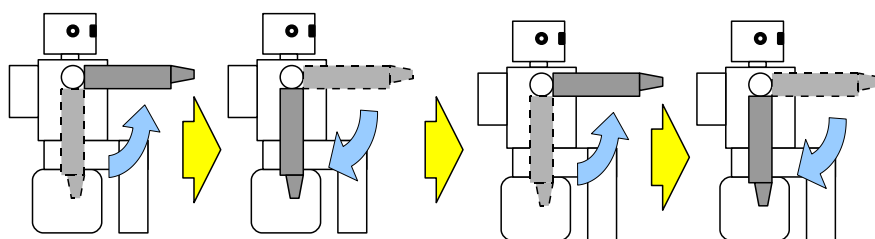


図 4.51: 実験 4・タスクの概要図

また、本実験では、操作方法の教示も行う。ユーザはロボットに対して動きの教示を行うのと同時に、操作方法の教示を行う。操作方法の教示は、再生時に、どのような操作をすると教示している動きを行うかを教示する。操作方法の教示は EXTREME 3D PRO を用いて行う。

本実験では、3 種類の腕の上げ下げの方法を 3 回ずつ教示した。サーボの運動軌道は、正弦波・三角波・台形波となるように教示を行った。教示した動きはすべて 1 つの頻度空間を用いて動きの汎化を行った。再生段階では、操作デバイスの入力とロボットが置かれている状況（うつぶせ状態か、仰向け状態か等）を変えて複数回の再生を行った。

実験結果として

- 教示した動き
- タスクの教示により汎化された頻度空間
- ロボットの動作結果

を示す。

表 4.16: 実験全体に関連する設定

|                  |         |
|------------------|---------|
| ロボットの教示・再生時間     | 10 秒    |
| サーボの角度データ取得タイミング | 0.1 秒刻み |
| サーボの角度データ取得回数    | 100 回   |
| 教示回数             | 9 回     |

#### 4.6.2 実験の設定

まず、実験 4 全体に関連する設定を表 4.16 に示す。

ロボットの教示・再生時間を 10 秒とした。ロボットに対して動きを教示する際、時間毎のサーボの角度を取得するが、この取得タイミングを 0.1 秒毎とした。10 秒間教示を行い、0.1 秒毎にサーボの角度データを取得するので、サーボの角度データの取得回数は 100 回となる。また、教示回数は、9 回とした。運動軌道が正弦波・三角波・台形波となるようにそれぞれ 3 回ずつ教示した。

運動軌道が正弦波・三角波・台形波となる理想の教示の例を図 4.52 に示す。

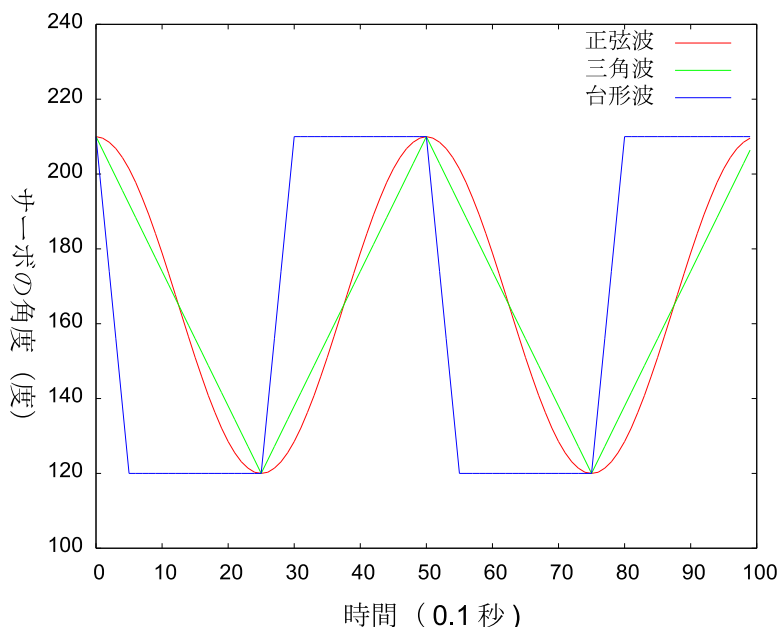


図 4.52: 理想の教示の例

9 回の教示を行う際、操作方法の教示（操作デバイスからの入力）とロボットが置かれている状況を毎回変えて教示した。動きの再生時には、操作デバイスからの入力と、ロボットが置かれている状況を変えて複数回の再生を試した。

次に、頻度空間に関連する詳細の説明を行う。本実験における頻度空間は

1. サーボ角度軸
2. 操作デバイス軸
3. G センサ軸
4. 時間軸
5. 選択頻度軸

からなる 5 次元空間である。頻度空間の定義に関連する設定を表 4.17 に示す。

表 4.17: 頻度空間の定義に関連する設定

|                                 |         |
|---------------------------------|---------|
| サーボ角度軸におけるセルの刻み幅                | 1 度刻み   |
| サーボ角度軸の値                        | 0 ~ 239 |
| 操作デバイス軸におけるセルの刻み幅               | 4369 刻み |
| G センサ軸におけるセルの刻み幅                | 10 刻み   |
| 操作デバイス軸の値                       | 0 ~ 14  |
| G センサ軸の値                        | 0 ~ 19  |
| 時間軸におけるセルの刻み幅                   | 0.1 秒刻み |
| 時間軸の値                           | 0 ~ 99  |
| 中心となるセルの加算値                     | 1       |
| サーボ角度軸方向の割引率                    | 0.9     |
| 操作デバイス軸方向の割引率                   | 0.3     |
| G センサ軸方向の割引率                    | 0.5     |
| サーボ角度軸方向において次の時間に選択される可能性のある距離  | 10 セル以下 |
| 操作デバイス軸方向において次の時間に選択される可能性のある距離 | 1 セル以下  |
| G センサ軸方向において次の時間に選択される可能性のある距離  | 1 セル以下  |

#### サーボ角度軸におけるセルの刻み幅

パラメータ「サーボ角度軸におけるセルの刻み幅」は1度刻みとなっている。つまり、頻度空間はサーボ角度軸方向においては1度刻みのデータしか保持していない。これに対し、サーボにおける角度センサの分解能は0.1度刻みである。そこで、教示段階では、動きの教示時にロボットが取得した角度は小数点以下を切り捨てる。これにより、頻度空間で角度センサの値をサーボ角度軸の値として扱えるようにし、動きの汎化を行う。

また、再生時の動きを生成は、頻度空間を用いて次々と目標角度を計算することで行われる。しかし頻度空間はサーボ角度軸方向においては1度刻みのデータしか保持していないので、目標角度も1度刻みでしか計算することが出来ない。そのため、サーボにおける角度センサの分解能が0.1度刻みであっても1度刻みでしか動きの再生を行うことが出来ない点に注意する必要がある。

#### サーボの角度軸の値

上述したように、動きの教示時にロボットが取得した角度は、小数点以下を切り捨てる。そのため「サーボの角度軸の値」については、サーボにおける角度センサの取得値である0~2400を1/10した値となる。「サーボの角度軸の値」が0~240ではなく、0~239となっているのは、表4.18のように設定したためである。サーボの角度軸の値が239の時の対応する角度が、1.1度分となっている点に注意する。

右腕・前後方向のサーボと頻度空間のパラメータ「サーボの角度軸の値」の関連図を図4.53に示す。

#### 操作デバイス軸におけるセルの刻み幅

パラメータ「操作デバイス軸におけるセルの刻み幅」は4369度刻みとなっている。このパラメータは、操作デバイスから取得した値を4369で割り、小数点以下を切り捨てた値が頻度空間の操作デバイス軸におけるセルの値となっていることを示している。つまり、頻度空間は操作デバイス軸方向においては4369度刻みでしかデータを保持していない。そこで、

表 4.18: 角度センサの値とサーボの角度軸の値の対応

| 角度センサの値     | サーボの角度軸の値 |
|-------------|-----------|
| 0 ~ 9       | 0         |
| 10 ~ 19     | 1         |
| 20 ~ 29     | 2         |
| ...         | ...       |
| 2380 ~ 2389 | 238       |
| 2390 ~ 2400 | 239       |

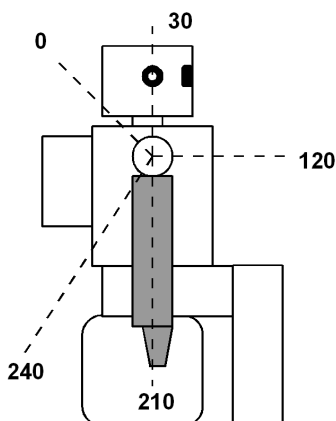


図 4.53: 右腕・前後方向のサーボと頻度空間のパラメータ「サーボの角度軸の値」の関連図

教示時には上述したように、操作デバイスから取得した値を 4369 で割り、小数点以下を切り捨てた値を用いて頻度空間を更新する。再生時にも、動きの選択のために取得した操作デバイスの値は、4369 で割って小数点以下を切り捨てた値が用いられる。

刻み幅を 4369 とした理由は、EXTREME 3D PRO の取得値である 0 ~ 65535 の値をちょうど 15 段階で分割出来るためである。

#### G センサ軸におけるセルの刻み幅

パラメータ「G センサ軸におけるセルの刻み幅」は 10 刻みとなっている。このパラメータは、G センサから取得した値を 10 で割り、小数点以下を切り捨てた値が頻度空間の G センサ軸におけるセルの値となっていることを示している。つまり、頻度空間は G センサ軸方向においては 10 刻みでしかデータを保持していない。そこで、教示時には上述したように、G センサから取得した値を 10 で割り、小数点以下を切り捨てた値を用いて頻度空間を更新する。再生時にも、動きの選択のために取得した G センサの値は、10 で割って小数点以下を切り捨てた値が用いられる。

#### 操作デバイス軸の値

上述したように、EXTREME 3D PRO の取得値である 0 ~ 65535 の値を 15 段階で分割したため、頻度空間における操作デバイスの値は 0 ~ 14 の 15 段階となる。操作デバイスの取得値と頻度空間における操作デバイス軸の値の対応表を表 4.19 に示す。

表 4.19: 操作デバイスの取得値と頻度空間における操作デバイス軸の値の対応

| 操作デバイスの取得値    | 頻度空間における操作デバイス軸の値 |
|---------------|-------------------|
| 0 ~ 4368      | 0                 |
| 4369 ~ 8737   | 1                 |
| 8738 ~ 13106  | 2                 |
| ...           | ...               |
| 56797 ~ 61165 | 13                |
| 61166 ~ 65535 | 14                |

## G センサ軸の値

ロボットが持つ G センサの取得値と頻度空間における G センサ軸の値の対応表を表 4.20 に示す。

表 4.20: G センサの取得値と頻度空間における G センサ軸の値の対応

| G センサの取得値  | 頻度空間における G センサ軸の値 |
|------------|-------------------|
| -477 ~ -91 | 0                 |
| -90 ~ -81  | 1                 |
| -80 ~ -71  | 2                 |
| ...        | ...               |
| 80 ~ 89    | 18                |
| 90 ~ 477   | 19                |

G センサ軸におけるセルの刻み幅が 10 刻みであるにも関わらず、頻度空間における G センサ軸の値が 0 の時と 19 の時には 10 刻みになっていない。これは、本実験では、G センサの取得値はほとんどが -90 ~ 90 の範囲に収まるためである。本実験は G センサの値によりロボットの動きを変化させることが可能であるか検証することが目的である。そのため、使用頻度の少ない G センサの値については省略する。

## 中心となるセルの加算値

頻度空間の更新は、各時間ごとにある 1 つのセルを中心にして山型に更新される。その中心となるセルにおける選択頻度の加算値が「中心となるセルの加算値」というパラメータである。3.3 節の頻度空間の更新式の中では  $c$  というパラメータとして述べた。

## 割引率

中心となるセル以外のセルは、中心から 1 セル離れる毎に一定の割引率で割り引かれた値が加算される。本実験では、頻度空間から時間軸と選択頻度軸を除くとサーボ角度軸と操作デバイス軸と G センサ軸である。そのため、選択頻度はサーボ角度軸方向と操作デバイス軸方向と G センサ軸方向に山型に更新される。この時の割引率が「サーボ角度軸方向の割引率」「操作デバイス軸方向の割引率」「G センサ軸方向の割引率」となる。3.3 節の頻度空間の更新式の中では  $\beta_j$  というパラメータとして述べた。

## 次の時間に選択される可能性のある距離

再生段階において、次の時刻の目標角度を決定するために、現時刻のサーボの角度と現時刻の G センサの値が用いられる。目標角度は、現時刻のサーボの角度から「一定の距離内」であり、かつ現時刻の操作デバイスの値から「一定の距離内」であり、かつ現時刻の G センサの値から「一定の距離内」にあるセルの中で最も選択頻度の高いセルが選ばれる。このセルのサーボ角度軸の値が次の時刻の目標角度である。サーボ角度軸方向における「一定の距離内」がどのくらいの距離であるかを表したパラメータが、「サーボ角度軸方向において次の時間に選択される可能性のある距離」である。また、操作デバイス軸方向における「一定の距離内」がどのくらいの距離であるかを表したパラメータが、「操作デバイス軸方向において次の時間に選択される可能性のある距離」である。また、G センサ軸方向における「一定の距離内」がどのくらいの距離であるかを表したパラメータが、「G センサ軸方向において次の時間に選択される可能性のある距離」である。

### 4.6.3 結果

ここでは実験 4 の結果を示す。

#### 動きと操作方法の教示結果

動きをロボット（右腕・前後方向のサーボ）に教示した。また、操作デバイスを用いて操作方法についても教示した。教示した動きを図 4.54～図 4.56 に示す。さらに、各動きを教示した時に「ユーザは操作デバイスを用いてどのような操作をしていたか」「ロボットはどのような状態にあったか」を図 4.57 に示す。

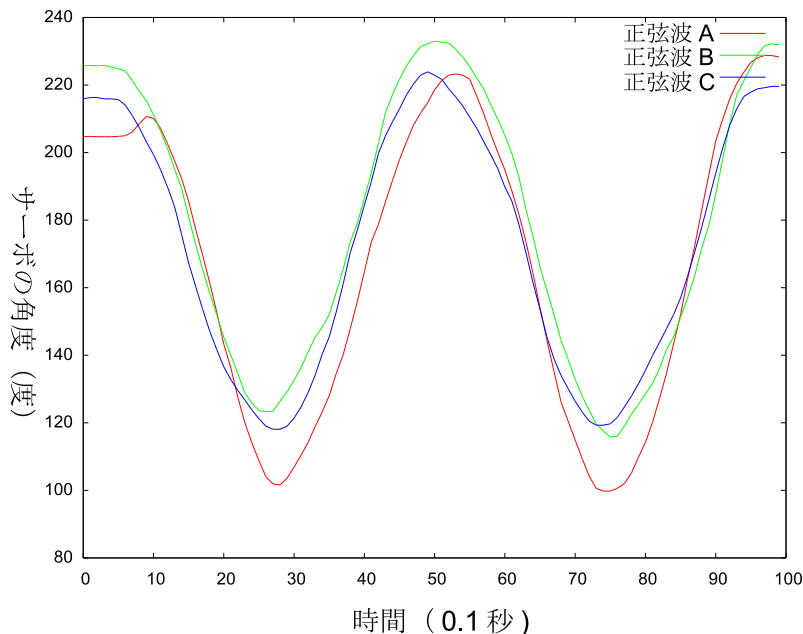


図 4.54: 教示した動き（正弦波）

例えば、正弦波 A の教示した動きの軌道は図 4.54 内で示されている。これに加えて、正弦波 A を教示したときの操作デバイスとロボットの状態は図 4.57 で示されている。正弦波 A の軌道の場合、ロボットが仰向けの状態で、操作デバイスは奥に倒されている時に教示されたことを表している。



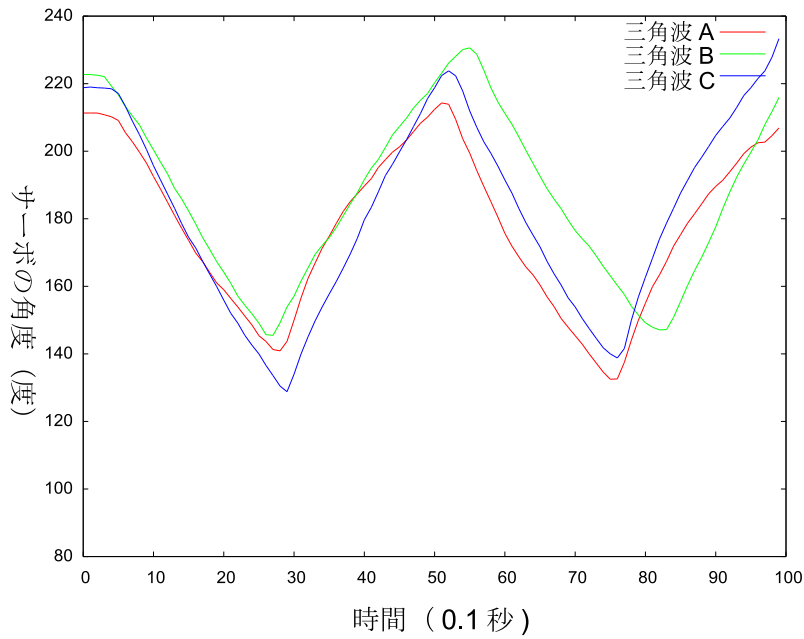


図 4.55: 教示した動き (三角波)

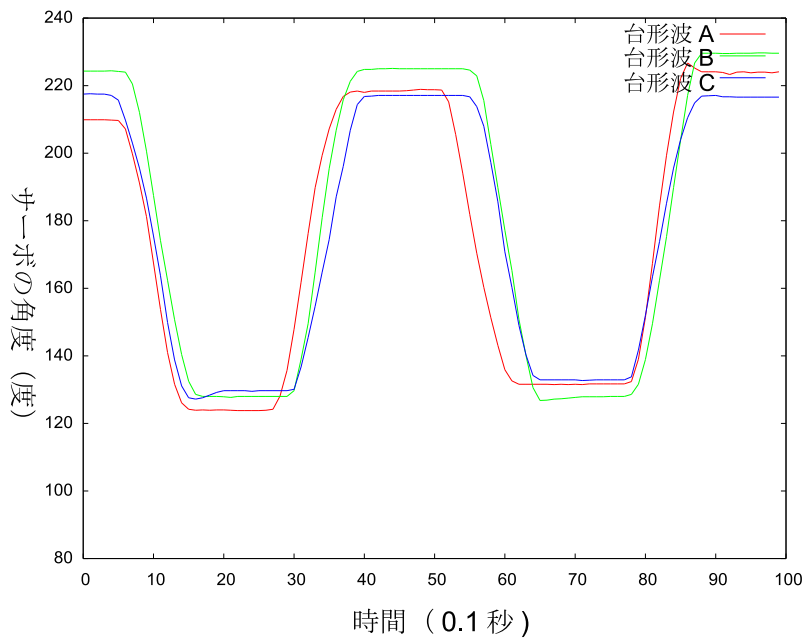


図 4.56: 教示した動き (台形波)

入力デバイスの状態は、「奥」が手前から奥に向かってレバーが倒されている状態を示す。おなじように「手前」が奥から手前に向かってレバーが倒されている状態を示す。「中」は操作デバイスのニュートラルの位置である。最大まで奥に向かって倒されている状態と最大まで手前に向かって倒されている状態のちょうど中間の位置である。

### 動きの汎化

頻度空間を用いて、教示した動きを汎化した。汎化した結果を示す。動きの汎化の結果として、9回の教示により汎化された頻度空間を示すが、本実験における頻度空間は、サーボ

|           |    | ロボットの状態 |       |       |
|-----------|----|---------|-------|-------|
|           |    | 仰向け     | 垂直    | うつぶせ  |
| 入力デバイスの状態 | 奥  | 正弦波 A   | 台形波 B | 三角波 C |
|           | 中  | 台形波 A   | 三角波 B | 正弦波 C |
|           | 手前 | 三角波 A   | 正弦波 B | 台形波 C |

図 4.57: 各教示を行った時の操作デバイスとロボットの状態

角度軸・操作デバイス軸・G センサ軸・時間軸・選択頻度軸からなる 5 次元空間である．そのため，頻度空間のすべての結果を示すのは難しい．そこで，操作デバイス軸・G センサ軸の値を固定し，残りのサーボ角度軸・時間軸・選択頻度軸で表される 3 次元空間を示す．

操作デバイスの値を 13 に，G センサ軸の値を 10 に固定した結果を 4.58 に示す．同じように，操作デバイスの値を 7 に，G センサの値を 9 に固定した結果を 4.59 に，操作デバイスの値を 7 に，G センサの値を 3 に固定した結果を 4.60 に，操作デバイスの値を 7 に，G センサの値を 6 に固定した結果を 4.61 に，操作デバイスの値を 10 に，G センサの値を 6 に固定した結果を 4.62 に示す．

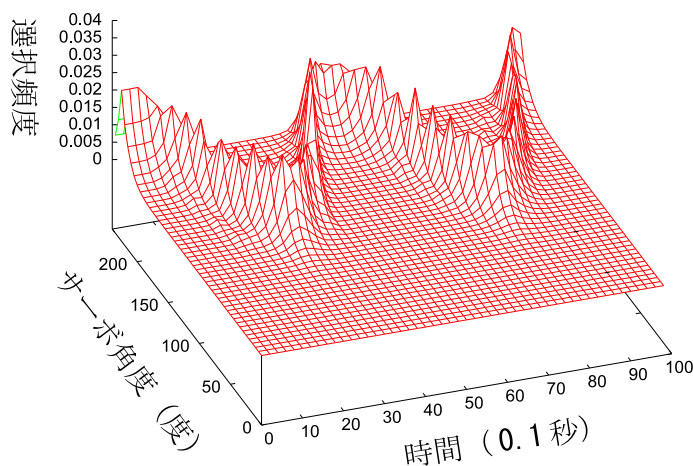


図 4.58: 汎化された頻度空間（操作デバイスの値を 13 に，G センサ軸の値を 10 に固定した頻度空間）

図 4.58 の結果を見ると，正弦波の形で選択頻度が更新されていることが分かる．操作デバイス軸の値が 13，G センサ軸の値が 10 であることは，操作デバイスは手前に倒され，ロボットの上部は地面と垂直に近いことを表している．この状態で行った教示は，正弦波の動きの教示である．そのため，正弦波の動きを表すセルの選択頻度が高くなっている．

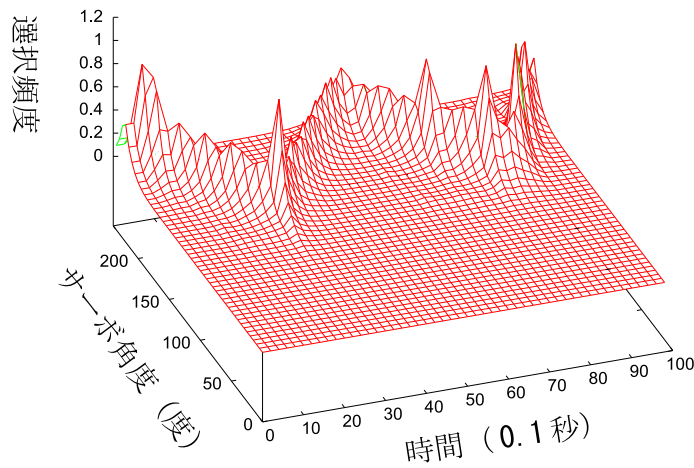


図 4.59: 汎化された頻度空間 (操作デバイスの値を 7 に, G センサの値を 9 に固定した頻度空間)

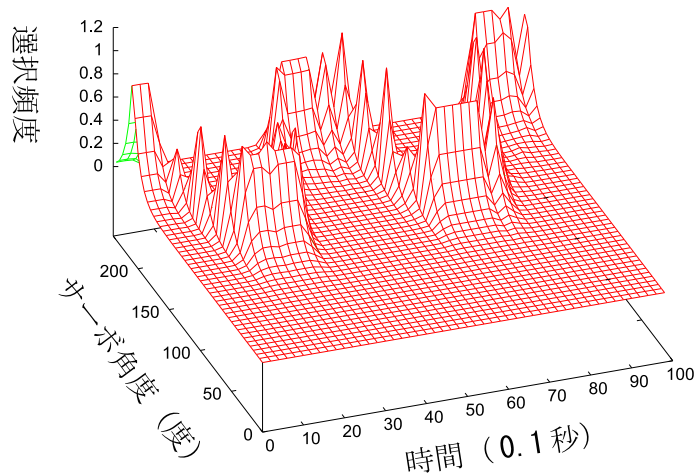


図 4.60: 汎化された頻度空間 (操作デバイスの値を 7 に, G センサの値を 3 に固定した頻度空間)

図 4.59 の結果を見ると, 三角波の形で頻度空間が更新されていることが分かる. 操作デバイスの値が 7, G センサの値が 9 であることは, 操作デバイスの入力ニュートラルの位置にあり, ロボットの上体は地面と垂直に近いことを表している. この状態で行った教示は, 三角波の動きの教示である. そのため, 三角波の動きを表すセルの選択頻度が高くなっている.

図 4.60 の結果を見ると, 台形波の形で頻度空間が更新されていることが分かる. 操作デバイスの値が 7, G センサの値が 3 であることは, 操作デバイスの入力ニュートラルの位置にあり, ロボットは仰向け方向に傾いていることを表している. この状態で行った教示は,

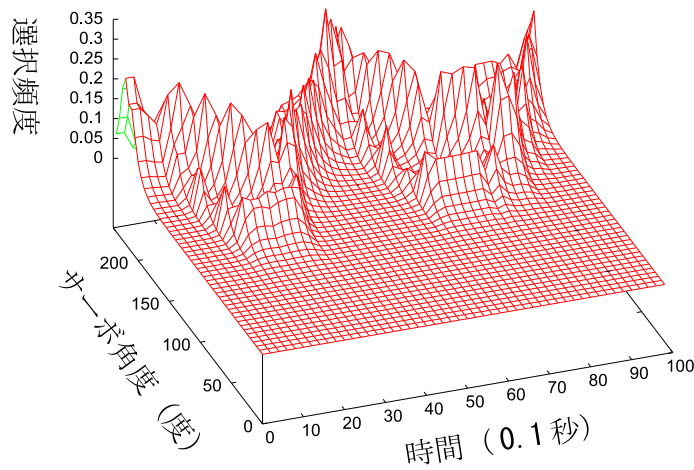


図 4.61: 汎化された頻度空間 (操作デバイスの値を 7 に, G センサの値を 6 に固定した頻度空間)

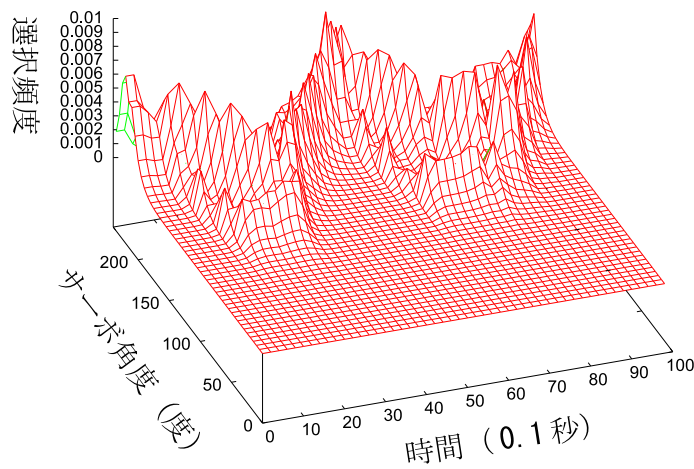


図 4.62: 汎化された頻度空間 (操作デバイスの値を 10 に, G センサの値を 6 に固定した頻度空間)

台形波の教示である．そのため，台形波の動きを表すセルの選択頻度が高くなっている．

図 4.61 の結果を見ると，一見ではどのような形で頻度空間が更新されているか分からない．操作デバイスの値が 7, G センサの値が 6 であることは，操作デバイスの入力ニュートラルの位置にあり，ロボットは地面に対して垂直の状態と仰向けの状態の中間の状態であることを表している．そのため，この頻度空間は台形波 A および三角波 B の影響を強く受けた頻度空間になっていることが分かる．

図 4.62 の結果を見ると，一見ではどのような形で頻度空間が更新されているか分からない．操作デバイスの値が 10 であることは，操作デバイスはニュートラルの位置と最大まで

手前に倒された位置の中間の位置にあることを表している．また，G センサの値が6 であることは，ロボットは地面に対して垂直の状態と仰向けの状態の中間の状態であることを表している．そのため，台形波 A，三角波 A，三角波 B，正弦波 B の影響を強く受けた頻度空間になっていることが分かる．

#### 頻度空間を用いた動きの再生

汎化された頻度空間を用いて，ロボットを動作させた．操作デバイスの値と G センサの値によりロボットは動きを選択する．ユーザによる操作とロボットが置かれている状況（G センサの値）を変えつつ，複数回の再生を行った．図 4.63 で示した操作デバイスの値の変化と，図 4.64 で示した G センサの値の変化によりロボットが動作した結果を図 4.65 に示す．また，図 4.66 で示した操作デバイスの値の変化と，図 4.67 で示した G センサの値の変化によりロボットが動作した結果を図 4.68 に示す．図 4.69 で示した操作デバイスの値の変化と，図 4.70 で示した G センサの値の変化によりロボットが動作した結果を図 4.71 に示す．

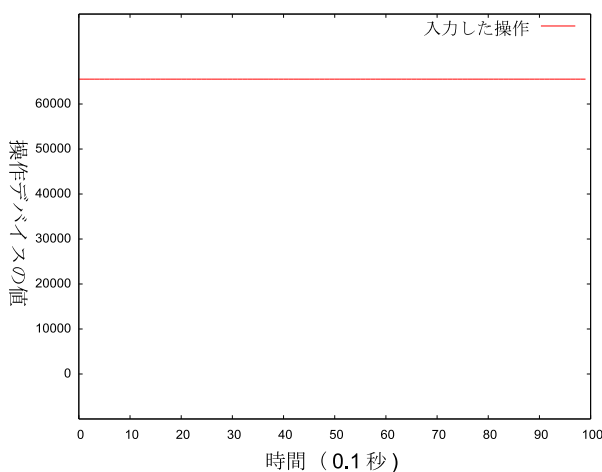


図 4.63: ロボットに対して行った操作

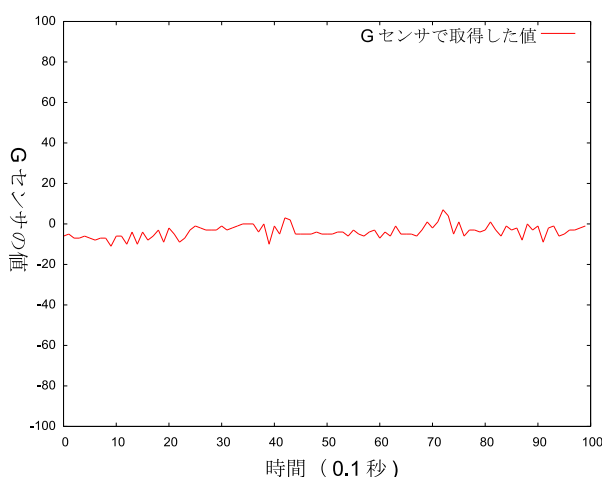


図 4.64: ロボットが取得した G センサの値

図 4.64 においては，ロボットの状態（G センサの値）を意図的に変えている場所は無く，図 4.67 および図 4.70 においてもロボットの状態（G センサの値）を意図的に変えているの

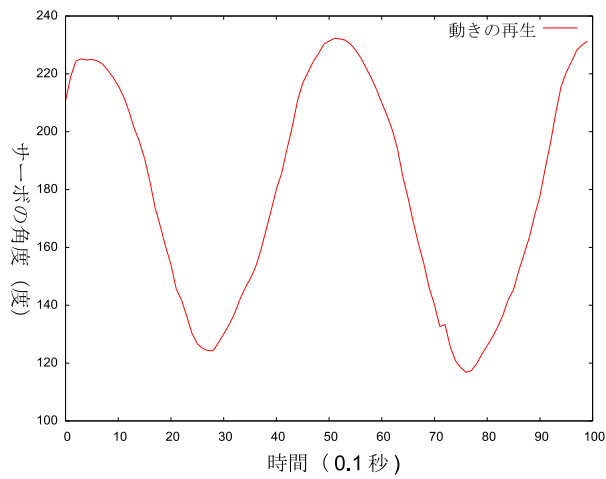


図 4.65: 図 4.63 と図 4.64 の値が与えられた時の動作結果

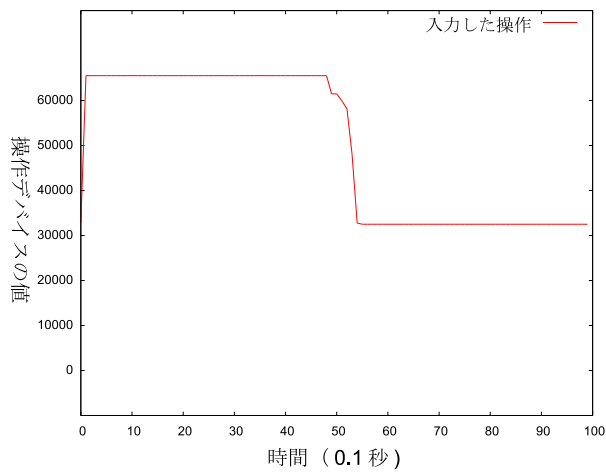


図 4.66: ロボットに対して行った操作

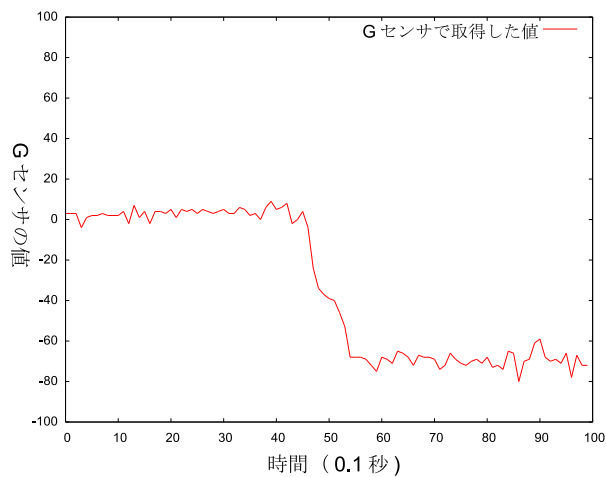


図 4.67: ロボットが取得した G センサの値

は 5 秒付近のみである．それ以外の G センサの小さな値の変化は意図的なものではないので，なぜ起こるのか検証し，改善が必要であると思われる．この原因は以下の 2 つが考えら

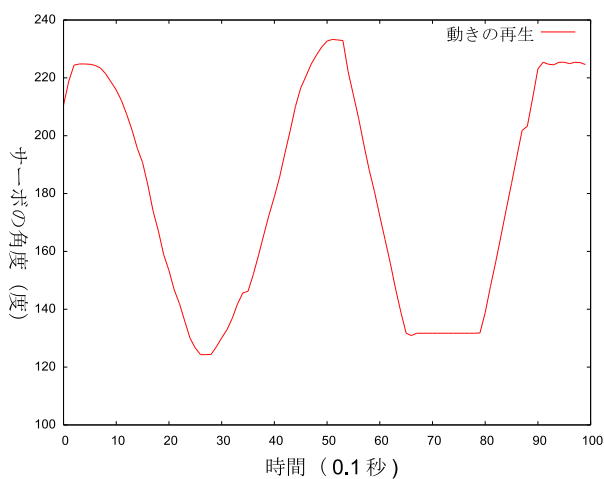


図 4.68: 図 4.66 と図 4.67 の値が与えられた時の動作結果

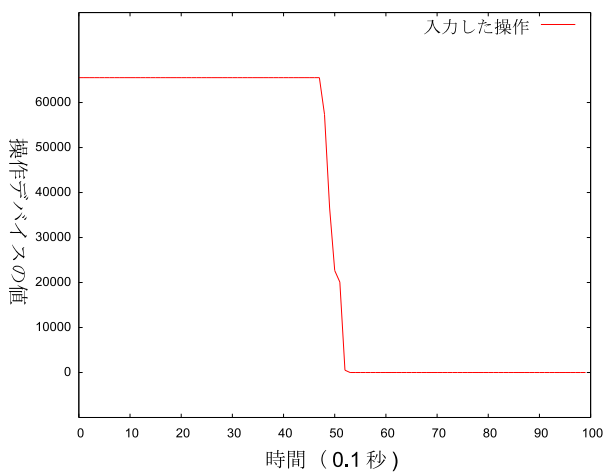


図 4.69: ロボットに対して行った操作

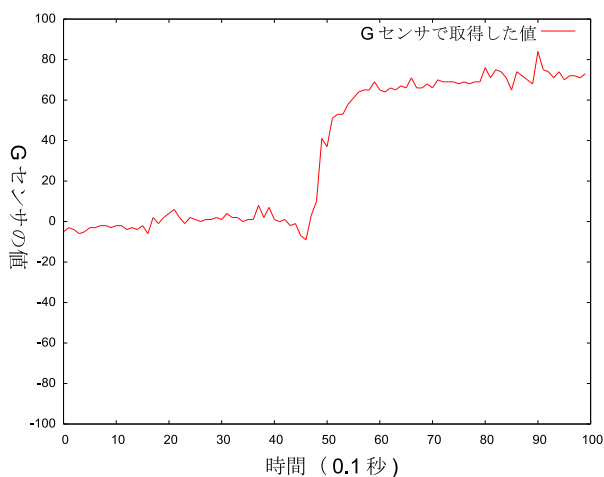


図 4.70: ロボットが取得した G センサの値

れる .

- ロボットが発する振動

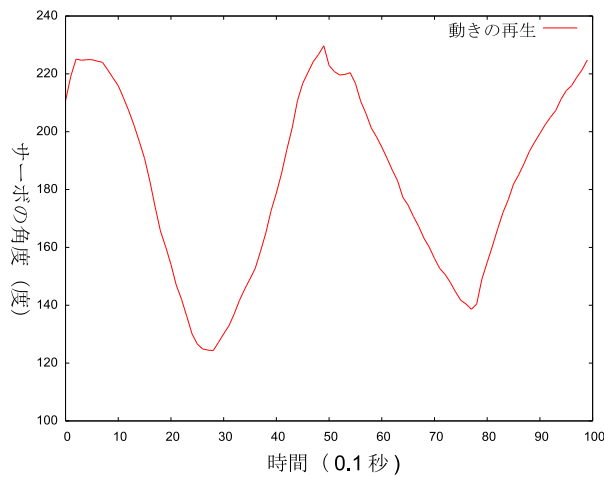


図 4.71: 図 4.69 と図 4.70 の値が与えられた時の動作結果

ロボットがサーボを動かす際、ある程度の振動が発生してしまう。これについてはロボットの構造上の問題であり、実ロボットを扱うかぎり回避するのは難しい。

- 環境が発する振動

ロボット以外の外部要因により振動が発生し、ロボットに伝わってしまう可能性がある。これを解決するためには、外部の振動が伝わりにくい環境を構築し、実験を行う必要がある。

#### 4.6.4 考察

本論文で作成したシステムを用いることにより、動きの教示と操作方法の教示、ロボットが置かれている状態に関する情報（本節ではGセンサの値）から、動きの汎化を行う。これにより、汎化された動きを元に、ユーザによる操作とGセンサの値からロボットの動きを変化させることが可能であることを示した。



## 第5章 おわりに

### 5.1 本論文のまとめ

本論文では教示と再生に着目し、再生時にユーザの操作によりロボットの動きを変えることが出来るシステムについて考えた。特に、ユーザが「何について」「どのように」操作するするかを決めることが出来るロボット操作システムの提案を行った。

そのために、システムを教示段階と再生段階に分けてアプローチを示した。教示段階では「手を振らせたい」「歩かせたい」など、ユーザがロボットに対して行わせたいタスクを教え、例えば「手を振る動作」であれば、「手を振る大きさの違う動き」や「手を振る速さの違う動き」など、教示した動きと似ているが多少異なる動きについては自動的に補完することで、動きの汎化を行うというアプローチを示した。再生段階では、デバイスを用いることでロボットに操作に関する情報を与え、補完した動きの中で自由に操作するというアプローチを示した。

このアプローチを実現するために、過去研究「ロボットの動きに関する新しい表現方法」[16]内で定義されている頻度空間を改変して用いた。教示段階では、頻度空間内にロボットが持つ各関節の値・ロボットが持つ各センサの値・ユーザの操作に関する情報などを保存し、再生時に操作によってロボットの動きが変わるように動きの汎化を行った。再生段階では、ロボットに対してリアルタイムで操作に関する情報を送ることで、任意にロボットの動きを変更するための手法を示した。

実験では、本論文で示したアプローチ・手法を用いたシステムを実ロボットを用いて実際に作り、正常に動作することを示した。

本論文のアプローチ・手法を用いることで、ユーザが「何について」「どのように」操作するするかを決めることが出来るロボット操作システムを実現した。

### 5.2 今後の課題

#### 5.2.1 概要

今後の課題として、以下の問題が挙げられる。

- 高次元空間の利用によるデータ量の増大
- 教示・操作の容易さに関する評価実験を行っていない
- 複数関節・複数入力を用いた実験を行っていない

#### 5.2.2 高次元空間の利用によるデータ量の増大に関する問題

本論文で述べた頻度空間は、ロボットが持つ各自由度・各センサごとに軸を持つ。そのため、ロボットが多自由度・多センサになるほど頻度空間のデータ量は指数関数的に増大する。しかし、データを保存可能な容量は有限である。また、データ量が指数関数的に増大することで、データを処理する時間も指数関数的に増大することが予想される。これが「高次元空間の利用によるデータ量の増大に関する問題」である。

この問題を解決する方法はいくつか考えられる．

- 頻度空間の整理や分割

多自由度・多センサを持つロボットであっても，1つのタスクを行うために使う自由度，センサは限られていることが多い．このような場合は頻度空間を整理して，必要な自由度やセンサからなる頻度空間を構築することである程度問題は解決される．

また，タスクは，元のタスクよりも小さな複数のタスクに分けることが出来る場合がある．タスクを分割することで，1度に使う自由度，センサを減らすことが出来る．この場合，複数のタスクを行うために頻度空間も複数個必要になるが，1度に使う自由度，センサを減らすことで頻度空間の次元数を抑えることが出来る．

- 頻度空間以外の運動軌道生成手法を用いる

本研究では頻度空間を用いて運動軌道を生成したが，それ以外にも，生物でリズムミク運動の生成に用いられていると考えられている神経振動子を用いた研究 [8] やカオストラクタを設計し，そのアトラクタへの引き込み現象を用いる研究 [17] などが行われているようである．これらの運動軌道の生成法を応用することも可能であるかもしれない．

### 5.2.3 教示・操作の容易さに関する評価実験を行っていない

本論文では，ユーザがロボットに対して動きと操作方法の教示を行い，再生時にロボットの操作を行うシステムを作成した．本論文ではこのシステムの教示・操作の容易さについては考えていないが，実際に運用する可能性を考えると，教示・操作の容易さの評価は行う必要がある．教示・操作がユーザにとって難しいものであると判断された場合，教示・操作を容易にする方法を考える必要がある．

### 5.2.4 複数関節・複数入力を用いた実験を行っていない問題

本論文では，提案した手法を実証するために必要な実験のみを行った．しかし，本論文で提案したシステムは，複数関節への操作，複数入力を用いた操作も可能である．複数関節への操作，複数入力を用いた操作を行う場合，上述したように「高次元空間の利用によるデータ量の増大に関する問題」が大きくなる．また，複数関節への教示・複数入力を用いた操作を行うことにより，1関節への教示・1入力を用いた操作と比べて教示・操作が難しくなることが考えられる．そのため，複数関節・複数入力を用いた実験を行い，問題の解決方法を考える必要がある．

# 謝辞

本論文を結ぶにあたり，日頃から様々な面で有益な御指導・御助言をいただきました倉重健太郎先生，畑中雅彦先生，本田泰先生，須藤秀紹先生，渡部修先生に深く感謝の意を表します．また多忙の中，査読を行なってくださった修士の木島康隆さん，学部4年の中南義典さん，宮崎愛央さんに感謝の意を表します．

## 参考文献

- [1] 小畑 昌之, 西田 健, 宮河 秀和, 近藤 隆史, 大川 不二夫, 「投機ゴミの回収を行う野外サービスロボットの開発」, 電気学会論文誌 C, Vol. 126, No. 7, pp. 840-848, (2006)
- [2] 藤田 善弘, 「パーソナルロボット R100」, 日本ロボット学会誌, Vol. 18, No. 2, pp. 198-199, (2000)
- [3] 柴田崇徳, 「人とロボットの身体的インタラクションを通じた主観的価値の創造 - アザラシ型メンタルコミットロボットの研究開発 - 」日本ロボット学会誌, Vol. 18, No. 2, pp. 200-203, (2000)
- [4] 藤田 善弘, 「チャイルドケアロボット PaPeRo」, 日本ロボット学会誌, Vol. 24, No. 2, pp. 162-163, (2006)
- [5] 下笹 洋一, 「警備ロボットの現状」, 映像情報メディア学会誌, Vol. 57, No. 1, pp. 79-82, (2003)
- [6] 吉泰 俊, 富沢 哲雄, 大矢 晃久, 油田 信一, 「親子型掃除ロボットの開発」, 日本機械学会関東支部第 11 期総会講演会公園論文集, pp. 303-304
- [7] 宮腰 清一, 多賀 徹太郎, 國吉 康夫, 長久保 晶彦, 「神経振動子を用いた三次元 2 足踏みシミュレーション - ヒューマノイドの実世界内行動を目指して - 」, 日本ロボット学会誌, Vol. 18, No. 1, pp.87-93, (2000)
- [8] 琴坂 信哉, Stefan Schaal, 「神経振動子を用いたロボットのリズム的な運動生成」, 日本ロボット学会誌, Vol. 19, No. 1, pp. 116-123, (2001)
- [9] 港 隆史, 浅田 稔, 「環境の変化に適応する移動ロボットの行動獲得」, 日本ロボット学会誌, Vol. 18, No. 5, pp. 706-712, (2000)
- [10] 本山 晴寿, 山科 亮太, 原 正之, 黄 健, 藪田 哲郎, 「強化学習によって獲得される芋虫型ロボットの前進行動形態に関する考察」, 日本機械学会論文集 ( C 編 ), Vol. 72, No. 723, pp. 3525-3532, (2006)
- [11] 高橋 泰岳, 浅田 稔, 「実ロボットによる行動学習のための状態空間の漸次的構成」, 日本ロボット学会誌, Vol. 17, No. 1, pp. 118-124, (1999)
- [12] 嶋田 憲司, 岡野 彰, 「グラフィックスによる対象レベルのロボット作業教示」, 情報処理学会第 33 回 ( 昭和 61 年後期 ) 全国大会, pp. 1573-1574, (1986)
- [13] 高橋 友一, 大原 秀一, 「プロダクションシステムによるロボット作業教示と実行システム」, 情報処理学会第 42 回 ( 平成 3 年前期 ) 全国大会, pp. 277-278, 1991
- [14] 田窪 朋仁, 西井 一敏, 井上 健司, 新井 健生, 「マリオネットシステムによるヒューマノイドロボットの全身遠隔操作」, 日本ロボット学会誌, Vol. 25, No. 3, pp. 457-465, (2007)

- [15] 舘 隆, 小森谷 清, 澤田 一哉, 井床 利之, 井上 幸三, 「HRP 遠隔操作プラットフォーム」, 日本ロボット学会誌, Vol. 19, No. 1, pp.16-27, (2001)
- [16] 幾世橋将文, 「ロボットの動きに関する新しい表現方法」, 室蘭工業大学卒業研究, (2007)
- [17] 岡田昌史, 中村仁彦, 「脳型処理情報を行う力学系の多項式設計法とそのヒューマノイドの全身運動生成への応用」, 日本ロボット学会誌, Vol. 22, No. 8, pp.1050-1060, (2004)

## 研究業績

[1] 池田 憲弘, 倉重 健太郎「教示における動きの汎化」, HAIシンポジウム 2009, 2D-2, (2009)