

エンターテインメント
ロボットの自発的な行動選択

平成19年度

向井 淳

目次

第1章 序論	1
1.1 はじめに	1
1.2 本研究の目的	1
1.3 本研究のアプローチ	2
1.4 関連する研究分野	3
1.5 本研究の意義	4
1.6 本論文の構成	5
第2章 ロボットの行動と多様性	6
2.1 動作と行動	6
2.2 エンターテインメントロボットの行動と多様性	7
2.3 行動変数と行動パターンの多様性	9
2.4 ロボットの行動モデル	10
第3章 関連研究	13
3.1 行動および行動規則の獲得に関する研究	13
3.1.1 新しい行動の獲得	13
3.1.2 行動規則の獲得	14
3.1.3 本研究との関連	15
3.2 感情モデルによる行動選択に関する研究	15
3.2.1 感情を用いたロボット二台の協調システム	16
3.2.2 パロ	17
3.2.3 WE-4RII	19
3.2.4 QRIO	21
3.2.5 Kismet	23
3.2.6 本研究との関連	25
第4章 観測指向モデル	28
4.1 概要	28
4.2 システム構成	29
4.3 観測モジュールとコンテキスト	29
4.4 不定な観測	30
4.5 出力モジュールと秩序関数	31
4.6 秩序関数の更新	32

4.7	行動基準の変化	33
4.8	実験環境	34
4.9	実験結果と考察	37
4.9.1	典型的な実行結果	37
4.9.2	ランダム選択モデルとの比較	44
4.9.3	パラメータの検討	46
4.9.4	インタラクションの影響	50
4.10	本章のまとめ	51
第5章	属性変換モデル	54
5.1	概要	54
5.2	構成図	55
5.3	物体の観測	56
5.4	コンテキストの更新	57
5.5	実験と考察	58
5.5.1	実験環境	58
5.5.2	典型的な行動結果	58
5.5.3	ランダム選択モデルとの比較	62
5.5.4	パラメータの検討	62
5.6	本章のまとめ	65
第6章	コミュニケーションロボット Robovie による動作例	67
6.1	システム構成	67
6.2	人間からのフィードバック	68
6.3	実装した行動と動作シナリオ	69
6.4	実験結果	72
6.5	本章のまとめ	76
第7章	議論	77
7.1	行動パターンの多様性と分布の関係	77
7.2	観測指向モデルと属性変換モデルの関係	78
7.3	今後の課題	80
7.3.1	モデルの拡張	80
7.3.2	パラメータの動的なチューニング	81
7.3.3	人間への心的効果の検証	82
7.4	他の応用事例の可能性	82
第8章	まとめ	84

目次

3.1	太田の並列再帰型自己組織化マップ	14
3.2	Murphy らのロボット	16
3.3	パロ	17
3.4	パロの動作決定機構	18
3.5	WE-4RII	19
3.6	WE-4RII の心理モデル	20
3.7	QRIO	21
3.8	QRIO の行動の階層構造	22
3.9	Kismet	23
3.10	Kismet の動機モデル	24
3.11	Kismet の感情モデル	25
4.1	観測指向モデルの概略	29
4.2	更新プロセス	30
4.3	出力プロセス	32
4.4	シミュレーション環境	35
4.5	直線配置パターン	36
4.6	ランダム配置パターン	36
4.7	ロボットの方向の時間遷移	37
4.8	直線配置でのロボットの方向の時間遷移	39
4.9	ランダム配置でのロボットの方向の時間遷移	39
4.10	グループ数の頻度分布	40
4.11	特定の方向を向いていた期間の頻度分布	42
4.12	維持された行動変数の累積	42
4.13	行動変数が維持される時間の頻度分布	43
4.14	ランダム選択モデルの実行例	44
4.15	ランダムとの比較	45
4.16	システムの出力におけるランダムとの比較	45
4.17	コンテキストによる頻度の差	46
4.18	コンテキストの差によるグループ数の比較	47
4.19	誤り率と維持期間の頻度の関係	48
4.20	誤り率とグループ数の関係	49
4.21	インタラクションがない場合の動作例	50

4.22	インタラクションがない場合の頻度分布	52
4.23	インタラクションがない場合のグループ数の頻度分布	52
5.1	属性変換モデルの構成	55
5.2	属性変換モデルにおける注意を向ける物体の遷移	59
5.3	コンテキスト内の属性分布の遷移	60
5.4	属性変換モデルにおける物体への注意を維持する期間の頻度分布	61
5.5	維持された期間の累積	61
5.6	ランダムとの比較	62
5.7	コンテキストサイズによる挙動の変化	63
5.8	選択数 n による挙動の変化	64
6.1	ロボットシステムの構成図	68
6.2	コミュニケーションロボット Robovie	70
6.3	実験風景	71
6.4	ロボットに実装した行動の関係	71
6.5	青いブロックを指差す	73
6.6	オレンジ色のブロックを見せる	73
6.7	人間の命令に従ってコンテキストを上書きする	74
6.8	青いブロックを指差す	74
6.9	Robovie の注意の変化	75

表目次

4.1	秩序関数の特性	32
5.1	コンテキストサイズと最大属性の割合の関係	63
5.2	n と最大属性の割合の関係	65

第1章

序論

1.1 はじめに

近年のロボット技術の発達により、従来のような産業ロボットとは異なり、日常生活の場を自由に移動し人間とともに様々なタスクを遂行するロボットの実現が現実味を帯びてきた。日常生活の場において活動するロボットは、従来のような産業ロボットとは異なった問題を考えなければならない。たとえば産業ロボットが配置できる環境は決められており、周囲の環境で何が起こるかをあらかじめ想定しておくことができるが、日常生活の場で活動するロボットでは周囲の環境を完全に想定することが難しい。また、産業ロボットは、あらかじめ定められた行動を正確に遂行することが求められているが、日常生活の場で活動するロボットではむしろ、様々な行動を遂行する能力が求められている。

日常生活の場と一括りに述べたが、その目的には掃除や家事の補助など様々なものが考えられている。そうした目的のうち本研究ではエンターテインメントに焦点をあてる。エンターテインメントロボットの中には AIBO [SONY 99] のようにすでに販売されているものもあり、今後も日常生活の場で活動するロボットの重要な目的のひとつとなると考えられている。

エンターテインメントロボットを実現するにあたって、ロボットは自分の動作をどのように決定するかということが問題となる。エンターテインメント用途でないロボットではこれは大きな問題とはならないことが多い。なぜなら、ロボットの取るべき行動は論理的に決められるからである。たとえば掃除ロボットならば取るべき行動はごみを正しく認識して収集するという行動である。しかしながらロボットの目的がエンターテインメントである場合、ロボットの取るべき行動は論理的に決められるとは限らない。つまり、ごみがあれば回収をするといった単純な対応関係だけでは上手く行かない。むしろ、論理的な必然性によらない行動があることが楽しさの源となりうるだろう。とはいえ、ロボットが自分の行動を決定する場合には何らかのメカニズムが必要となる。

1.2 本研究の目的

本研究では、エンターテインメントロボットの行動決定メカニズムに焦点を当てる。特にロボットの行動パターンに注目し、行動パターンに多様性を与えることでロボットに思いもよらない行動を取らせることを目的とする。

エンターテインメントロボットでは行動の選択を論理的に推論するのが難しい場合がある。むしろエンターテインメントロボットにとって重要なのは実行する行動をいかに多様に変化させるかという点にあると言える。なぜなら、いつも同じ行動をするとインタラクションを行う人間は毎回同じ行動であると判断してしまうからである。その結果、人間はロボットの反応を単調に感じたり、飽きてしまうだろう。飽きさせないためには、ロボットが時として思いもよらない行動を取ることが望ましい。

従来のロボットでは、エンターテインメントを目的とするロボットであっても(たとえば PaPeRo[NEC 01] など)、ロボットの行動が典型的には if-then ルールに従って制御されている。つまり、ロボットは多数の条件部を持ち、その条件に一致したときに対応する行動を取り内部状態を変化させる。内部状態の変化によって条件部と対応する行動が変化し、一定の行動パターンが生まれるというものである。たとえばロボットが前進していて人間を発見したら手を挙げ「こんにちは」と発話することで人間に挨拶をする、というような行動が実現される。さらにその状況で人間がロボットに話しかけてきたら内部状態が変化し、移動を継続するのではなく人間との対話を行うように条件部と行動が切り替わる。

しかしこのアプローチでは思いもよらない行動をロボットに取らせるのは難しい。なぜなら、ロボットの行動はルールによって制御されているから、周囲の状況が決まればロボットの反応は一意に定まってしまうからである。人間は自分の働きかけに対してロボットがどう反応するかをすぐに覚えてしまう。

そこで考えられるのは、ロボットの行動シナリオを維持する内部状態とは別にロボットに内部状態を与え、それを複雑に変化させるということである。この内部状態のことをコンテキストと呼ぶ。コンテキストはセンサ入力やロボットの行動に影響を受け複雑に変化する。またコンテキストはロボットの行動に影響を与えることで、ロボットの行動パターンが複雑に変化する。さらにシナリオが同じ状態であってもコンテキストが同じであるとは限らないので、人間からの働きかけへの反応も様々に変化することができる。

ただし、コンテキストは単に変化をすればいいというわけではない。あまりにも頻繁に変化をするならばロボットの行動は無秩序なものとなり、エンターテインメント目的にはかえって適さない。一方、まったく変化しないのであれば最初から固定化されたルールを持つのと変わらない。大域的に見れば変化が起きる必要がある。

以上をまとめると、

- 局所的には安定し変化しないこと
- 大域的には変化する可能性があること

の二点が満たされることで、思いもよらない複雑な行動パターンが実現されると考えられる。

1.3 本研究のアプローチ

以上で述べた目的のため、本研究では二つの行動決定モデルを提案する。

1. **観測指向モデル:** 他のロボットの行動を観測することで複雑な行動パターンを生成するモデル

2. 属性変換モデル: 物体の属性情報から複雑な行動パターンを生成するモデル

本研究ではロボットの置かれる状況を大きく二つに分類する。第一は環境中に複数のロボットや人間が存在するという状況であり、この場合は観測指向モデルが対応する。第二は環境中には他のロボットや人間が存在しない状況であり、この場合は周囲に存在する物体の情報を用い、属性変換モデルにより複雑な行動パターンを実現する。

観測指向モデルでは、環境中に存在する他のロボットや人間の行動を観測し、現在の自分の行動と類似するものを選択的に自分の内部に取り込む。これによって、ほかのロボットと類似する行動を取ることで、ロボット群は互いに類似する行動によって局所的に安定する。ただし、観測指向モデルでは、エージェントの行動を観測した結果を意図的に間違えるようにしている。この誤りにより、システム内部には、現在の行動とは類似しないデータも僅かに導入される。この僅かな異種のデータはまれに大きな影響を持ち、最終的には大域的な変化を引き起こす。

属性変換モデルでは、環境に他のロボットの存在を想定しない。かわりに、環境中に存在する物体には色や大きさといった属性の情報をもつという構造を仮定する。属性変換モデルでは、物体の持つ属性のうち一部の属性を重視することで、安定的に注意を向ける物体を決定する。しかし、ロボットが重視する属性を時間的に変化させる。この変化によって、注意を向ける物体の変化が引き起こされる。

1.4 関連する研究分野

本研究に関連する研究分野は、大きく二つに分けられる。第一の分野はロボットの行動学習の研究であり、第二の分野はロボットの感情モデルに関する研究である。

ロボットの行動学習とは、機械学習の手法によってロボットが取るべき行動を学習する枠組みを提案する研究である。使用者がエンターテインメントロボットに飽きてしまうことを指摘する既存の研究では、ロボットが行動を学習し獲得することでこの問題に対処することが多い。なぜなら、ロボットが実行できる行動の種類が増えることで飽きることに対処できると考えられるからである。たとえば太田は再帰型自己組織化マップによってロボットへのセンサ入力と行動出力の関係を学習し、新しい行動の獲得に成功している[太田 03]。

しかしながら、このような研究では行動の実行はできるが、いつその行動を実行するかという点については考慮されていない。もしくは単純な条件に一致するという if-then ルールを獲得していくのみである。したがって、このままでは働きかけに対するロボットの反応を人間は覚えてしまい、行動パターンの多様性は実現されない。

第二の分野として感情モデルを挙げた。本研究の目的は、単純な if-then ルールとは異なるメカニズムによってロボットの行動選択を多様にすることであると言える。既存の研究でも心理学にもとづいた感情のモデルをロボット上に実装することで if-then ルールとは異なる行動決定機構はいくつも存在している。

たとえば、MIT で開発された Kismet [Breazeal 02] という顔ロボットは Ekman の感情モデル [Ekman *et al.* 94] にもとづいた感情モデルを持ち、表情を変化させる。また、Drive と呼ばれる動機モデルを持ち、人間に視線を向けるか、周囲にある玩具に視線を向ける

か、ということを選択できる。

また、QRIO [Sawada *et al.* 04] は内部に情動を司るモデルを持つ。具体的には motivation と呼ばれるパラメータを持ち、動的に行動を選択し実行する。

しかしながら、感情モデルをもつロボットに関する既存研究でも行動のバリエーションという点は考慮されていない。QRIO では motivation にもとづいて行動を決定できるが、ボールを蹴るという最初の motivation を QRIO 自身が獲得する仕組みはない。「ボールを蹴る」という motivation を人間が与えると、遂行するためのサブゴールとしてたとえば「ボールに近づく」といったことを選択することが可能であるという枠組みである。また、Kismet における動機システムは、周辺にある玩具に視線を向けるか人間に視線を向けるかということ調整するためのメカニズムであり、両方が視界内に存在するときには、どちらか一方を優先するようにあらかじめ組込まれている。

また、Kismet では、周辺に玩具が複数ある場合や、複数人の人間に囲まれたとき、どの玩具に注目するか、どの人間に視線を向けるか、ということは考慮されていない。いちばん目立つ玩具や人間を単純に選択してしまう。

1.5 本研究の意義

本研究の目的はエンターテインメントロボットの行動パターンに多様性を与えることであつた。

エンターテインメントロボットの行動に多様性を与えることを目的とした既存の研究では、ロボットが取ることのできる行動の種類を増やすことに焦点が当てられている。新しく獲得した行動を実行するためのルールについては詳しい検討はされておらず、単純なルールによって実行されることが想定されている。しかしながら、ロボットの行動に多様性を与えるという点からすると、行動の選択が単純なルールによって決定されるだけでは不十分である。

これに対し、本研究はロボットが新しい行動を獲得することは目的としない。ロボットが行動を選択するメカニズムを変更することで、ロボットの取りうる行動の種類は変わらなくても、ロボットが実行する行動のパターンを複雑に変化させることは可能である。多様な行動パターンを実現することにより、思いもよらない複雑な動きが実現される。行動そのものではなく行動パターンに着目することで多様性を実現するのが、既存の研究とは異なる本研究のアプローチである。

本研究では、ロボットの置かれる状況を2つに分けて考える。第一は周囲に人間や他のロボットが存在する状況であり、第二は周囲に人間や他のロボットが存在しないという状況である。そして、第一の状況に対しては他の人間やロボットの行動を観測し、その行動と類似する行動を取ることで複雑な行動パターンを実現する観測指向モデルを提案する。第二の状況に対しては、環境中に存在する物体の属性情報から複雑な行動パターンを実現する属性変換モデルを提案する。以上の2つのモデルにより、様々な状況下においてロボットは自発的に複雑な行動パターンを生成できる。

本研究の成果を既存の研究成果と組み合わせることでロボットのインタラクションはより多様になる。このことから、優れたエンターテインメントロボットの実現において重要

な成果ということができる。

1.6 本論文の構成

以下、2章では、本研究での議論を厳密にするために、特に混乱が生じやすい行動という語について定義を行い、本研究の目的について改めて論じる。そして3章では本研究の関連研究として、現在存在するエンターテインメントロボットの実例やその構成方法、行動にバリエーションを与えるための技術、および、感情モデルや注意機構にもとづくロボットの行動選択手法についての研究について述べる。4章では、観測指向モデルを提案し、その結果を検討する。5章では属性変換モデルを提案する。6章では、コミュニケーションロボット Robovie 上で属性変換モデルを実装した例を説明し、有効に動作していることを示す。7章では本研究で提案する2つのモデルについて有効性などの議論を行い、今後の課題を述べる。8章で本研究の結論を述べる。

第2章

ロボットの行動と多様性

本研究で目的とするのは、エンターテインメントロボットの行動に多様性を与えることである。

ところが、ロボットの「行動」という単語が指し示す範囲は広く、曖昧であり、また研究によっても定義が異なっていることも多い。また、エンターテインメントロボット自体についても明確な定義は存在しない。

そこで、本章ではまず、本研究で用いる「行動」という語の定義を行い、エンターテインメントを目的とするロボットについても定義する。そして、エンターテインメントロボットにおいて行動の多様性が必須であることを改めて論じ、本研究の目的について確認する。

2.1 動作と行動

ロボットは身体をもち、周囲の環境に様々な働きかけを行う能力を持つ。ロボットが取りうる環境への働きかけは、様々な観点から分類が可能である。たとえば歩いていると人間に出会って挨拶をして雑談をする、という場合、“歩いている”と“挨拶をする”と“雑談をする”というように分けられるだろう。このように分けられたものはさらに小さな構成要素に分けることができるかもしれない。“挨拶をする”ならば、“右手を挙げて”、“「こんにちは」と発話をする”というように切り分けることができよう。

このようにロボットが取りうる環境への働きかけを細かく分けていくと、切り分け不能な段階に到達する。この最小単位を本研究では**動作**(motion)と呼ぶ。動作はロボットを制御するプログラムが外部へ指令を出すにあたってのアトミックな構成単位である。

たとえば、腕に取り付けられた複数のモータに指令を送ることによって、ロボットは右腕を下方から上方へ動かすことができる。これは“右腕を上げる”という動作となる。ロボットの動作は、制御システム上では1つの指令として解釈される。“右腕を上げる”という動作は目的となる肩や肘などの関節の角度とモータの回転速度の指定に対応する。

ロボットの環境への働きかけは、モータの動きだけとは限らない。たとえば人間とインタラクションを行う能力を持つロボットでは、スピーカを有し音声を発話できる機能を有することがある(Robovie[神田 他 02]など)。また、LEDをいくつか備え、点滅や模様を様々に変化させる機能をもつロボットもある(AIBO[SONY 99]やIfbot[Kato *et al.* 04]など)。本研究では、ロボットの発話やLEDの点滅や模様も動作の一種とみなして考える。

たとえばロボットが発話をする場合は、あらかじめ録音または作成しておいた音声ファイルを指定して再生する。この場合、1回の発話が1つの動作である。LED群が表示する模様を決めるときも、個別のLEDの制御をするのではなく、制御プログラムの側では基本となる模様があつて、それを組み合わせたり点滅の速度を変更することで様々な模様を提示する。この場合には、基本となる模様が1つの動作に対応する。

このように、ロボットが環境に対して行う働きかけは動作によって構成される。しかし動作は働きかけの単位としては極めてプリミティブである。たとえば“挨拶をする”といった働きかけをする場合には、“右手を上げる”、“「こんにちは」と発話する”、“右手を下ろす”という動作の並びが実行される。このようにロボットが実行する動作の系列は、実際には一定のパターンによって区切られている。そこで、ロボットの動作のパターンに意味を付加したものを本研究では**行動(behavior)**と呼ぶ。

行動は一般に複数の動作のパターンによって構成されるが、単独の動作が行動となることもある。たとえば“右腕を上げる”という動作は、そのままでも“挨拶をする”という行動と解釈できる。

また、行動は意味的な単位であるため、実行される文脈が重要である。同一の動作であっても異なる行動となることがある。たとえば“右腕を上げる”という動作は、人間の前でやれば挨拶をするという行動であるとみなされるし、離れていれば注意を喚起するという行動であるとみなされよう。会話が終わるタイミングであれば別れの挨拶となる。

多くの場合、動作は制御システムを設計する際に必要となる概念である。一方、行動はシステムが実際に出力する動作系列を意味づけるものである。また、行動はシステム設計には陽に出現せず、暗黙のうちに動作決定メカニズムに埋め込まれていることもある。本研究では行動を選択するメカニズムを主に扱う。

2.2 エンターテインメントロボットの行動と多様性

エンターテインメントロボットはすでに多数が開発・発表されている。それだけでなく、エンターテインメントロボットには既に実用されているものや、AIBOのように市販されているケースもある。しかし、エンターテインメントロボットを構築するための技術的な課題はまだいくつか残されている。

まず、エンターテインメントロボットとはどのようなロボットかという定義について確認する。ロボットをエンターテインメントに用いるということはSFなどの文学でさかんに論じられてきたが、現実のロボットに対しては近年まで論じられてこなかった。ロボットをエンターテインメント用途に用いるというコンセプトは藤田らが提唱したのが最初である [Fujita *et al.* 97, 藤田 98]。その後、この概念が一般化するとともに、エンターテインメントを目的とするロボットを総称してエンターテインメントロボットという呼び方が定着している。

しかしながら、藤田らの提案ではエンターテインメントロボットの定義は行われていない。用途としてペットロボットやゲーム、テレプレゼンスなどを挙げているのみである。現在でも、人間を楽しませることを主たる目的とするロボットであるという共通認識はあるものの、エンターテインメントロボットの定義は存在しない。そこでまず、本研究にお

けるエンターテインメントロボットの定義を述べる。

本研究におけるエンターテインメントロボットは、主として**人間とのインタラクションを通じて人間に対して楽しみや快さなどの心的効果を与えることを目的としたロボット**と定義する。この定義に従えば、ペットロボットはエンターテインメントロボットに含まれる。また、パロ [柴田 99, Shibata *et al.* 01a] という医療介護ロボットもエンターテインメントロボットである。パロはアザラシ型のロボットで、患者がこのロボットとインタラクションをすることで心理的に癒やされるという効果を持つ。目的は医療介護であるが、人間への心的効果が目的なので本研究の定義ではエンターテインメントロボットに含まれる。一方で、展示会などでは舞いを披露するロボットのデモンストレーションなどがある。これも人間を楽しませることを目的としたロボットという意味ではエンターテインメントであると言えるが、一般の人間とのインタラクションを持たないため本研究の定義ではエンターテインメントロボットに含まれない。また、券売機のような単純な受付事務を行うロボットの場合は、人間とのインタラクションはあるが心的効果を目的としていないためエンターテインメントロボットではない。ただし、受付事務ロボットに代表されるタスク遂行に重きを置いたロボットであっても、人間の心理状態への影響を考慮したものがある。これらのロボットは、境界領域に位置する研究であると言える。

次にエンターテインメントロボットの要件をまとめる。90年代から、人間の生活環境で人間とともに活動するタイプのロボットは様々な研究されてきた。この状況を踏まえ、人間共存型ロボットの実現に向けた技術的な課題をまとめた報告が1998年に提出されている [人間共存型ロボット研究専門委員会 98]。この報告では、人間とロボットとのインタラクションについての要件を物理的、情動的、環境・経済的という4つの側面に分け、それぞれについて論じている。物理的な要件とは実際に握手などのインタラクションを行うためのメカニズム、情動的な要件とは周囲環境の認識などの技術、情動的な要件とは感情や情緒などを表出することによるインタラクション、環境・経済的な要件とは価格や利用環境などを意味している。このうち、エンターテインメントロボットは特に情動的な要件を重視しているロボットであると言えるだろう。

そして、この報告では、ロボットの側から情動的な働きかけを行うという場面において、

このなかで特に重要なポイントが、いかに多様な動作を生成するかである。画一的な動きでは、すぐに相手にそのパターンを読み取られ、相手に心的影響を持続して与えることはできなくなるからである。多様性をもたらす方法として、人間と同等の情緒表出メカニズム(生物学的自律系)を基にロボットを構成している例もあるが、現状の技術では、ジェスチャーをロボットのなかにモデルとしてすべて作り込む方法が取られている。最近様々なペットロボットが試作されているが、それらはすべて数十種類の動作パターンをあらかじめプログラムされているものである。人間に心地良く、魅力的で面白いといった心的影響を与えるために、そのパターンは試行錯誤を繰り返して作られていく。代表的な例は、プロのデザイナーによるディズニーのオーディオアニマトロニクスである。しかし、人間による動作の作り込みには限界があることは明らかであり、高い親和性が得られるようにするには、動作パターンの自動生成のメカニズムを検討する必要があるだろう。

と述べられている。つまり、情緒的な働きかけを実現するためには、ロボットの動作や行動が多様にならないといけないことが指摘されている。

エンターテインメントロボットは人間への心的効果が重要であるから、飽きられやすいとすれば大きな問題である。実際、ペットロボットに対して飽きられやすいという指摘はすでにくつかなされている ([太田 03][森 他 02])。

ただし、多様性が必要であるという指摘では、ロボットの環境に対する働きかけにおいて動作と行動を混同しているふしがある。もちろん、動作が多様であれば、実現される行動もまた多様となるとも考えられる。また、実行できる動作の種類が少なければ、実現される行動の種類も制限を受け、結果として多様性が実現できない可能性がある。とはいえ、“挨拶をする”という行動について“手を挙げる”あるいは“手を挙げて「こんにちは」と発話する”、“手を振る”といったように多数の動作パターンを持っていたとしても、“挨拶をする”という行動に変わりがないのであれば、人間への心的影響という観点からすると多様性は少ないと考えられる。つまり、エンターテインメントのような情緒的なインタラクションを目的とする場合、人間への心的効果が重要であり、そのために必要なのは動作ではなく行動の多様性である。

行動の多様性を実現するためのアプローチには二つのものが考えられる。第一のアプローチは、実行する行動の種類を増やすことである。しかし行動は意味的な単位なので、設計者が実装するのにも限界があり、多数の行動を用意するのは困難である。第二のアプローチは、行動を選択するルールをうまく構築することによって、行動の選択に変化を生じさせるというものである。このアプローチではロボットの取りうる行動の種類は増えないが、ロボットが実際に行う行動のパターンが様々に変化する。行動パターンが多様となることにより、人間にとって複雑に変化する行動が実現できる。つまり、動作レベルでのパターンの多様性を追求するアプローチと、行動レベルのパターンの多様性を追求するアプローチという差である。

また、第一のアプローチを取る既存の研究では、現在のところ獲得される行動はセンサ刺激の条件から対応する動作を実行するというパラダイムである (たとえば [太田 03] など)。このため人間の働きかけに対してロボットが複雑な反応を返すためには第二のアプローチは必要である。そこで本研究は第二のアプローチに注目し、多様な行動パターンの実現を目的とする。

2.3 行動変数と行動パターンの多様性

本研究では、行動パターンを多様化するために行動変数という概念を導入する。たとえば、“挨拶をする”という行動には、挨拶をする相手となる人間を選ぶ必要がある。同じように、多くの行動には対象となる人間や物体が存在する。このような行動の対象を本研究では行動変数と呼ぶ。

行動変数が増えることによって、ロボットは動作を変えるだけでなく、行動も僅かに変化させる。実際、既存研究でのロボットの行動は“人間に視線を向ける”といった単純なものであり、対象となる人間は、ロボットに最も近い位置にいるとか、視界内をスキャンして最初にプログラムが発見した人間であるとか、何らかの基準によってあらかじめ定

まっている [Breazeal 02]. これに対して本研究では行動を“視線を向ける”というテンプレートと“誰に”という行動変数に分解する. 行動変数を切り替えることによって行動が再構成され, 変化する. そのため“視線を向ける”という部分の意味は変わらないとしても, 対象が変化することによってインタラクションにおいては意味のある変化が生まれる.

ただし, 行動変数は行動の対象となる人間や物なので, その種類には限度がある. たとえば, 周囲に人間が2人しかいなければ“視線を向ける”という行動には2種類のバリエーションしか存在しない. つまり, 行動変数を導入しても行動の種類を増やすことには限界がある. しかしながら, 行動変数が様々に変化することによって, ロボットが実行する行動のパターンは複雑になる.

本研究では, ロボットは周囲の環境を観測することで行動変数を獲得・維持し, また変化させる. また, ロボットが保持する行動命令には行動変数を受け入れる余地を設け, 行動変数によって行動を再構成する. この結果として, システムに保持される行動の種類は少なくとも, 実際に実行される行動には様々な変化が生じる. また, 人間が同じ働きかけをしても行動変数によってロボットの反応は変化するから, ロボットの反応を人間が容易に推定できてしまうという問題にも対処が可能である.

以上をまとめる. 本研究の目的はエンターテインメントロボットに着目し, ロボットの取る行動の多様性を実現することである. この目的のため, 本研究では行動変数という概念を導入する. 行動変数とロボットが保持する行動と組み合わせることで, 実際に実行される行動パターンの多様化を目指す.

2.4 ロボットの行動モデル

ここでロボットが行動を選択するメカニズムを定式化する. 最も単純な定式化は, ロボットが外部刺激を受け取るとそれに応じて最適な行動を選択するというものであり, 次式で表現される.

$$b_t = \pi(s_t) \quad (2.1)$$

ここで s_t は時刻 t における外部刺激 (stimulation), b_t は時刻 t でロボットが取る行動 (behavior) を意味している. π は刺激を受け取り, 複数ある行動を決定する関数である. ただしこれでは状況の変化や記憶ができない.

これに対し, 何らかの内部状態を持つロボットは次のように表現できる.

$$\begin{cases} b_t = \pi(i_t, s_t) \\ i_{t+1} = U(i_t, s_t, b_t) \end{cases} \quad (2.2)$$

ここで i は内部状態 (internal state) であり, U は内部状態の更新関数を意味している. 通常の if-then ルールによって構成されるロボットでは, i はロボットが認識している状況を意味し, 対話シナリオが U によって表現されているとみなすことができる.

本研究では, 行動にパラメータを与えることで行動パターンの多様性を実現する. そこ

で時刻 t の行動変数を v_t として拡張すれば,

$$\begin{cases} B_t = \pi(i_t, s_t) \\ b_t = B_t(v_t) \\ i_{t+1} = U(i_t, s_t, b_t) \end{cases} \quad (2.3)$$

となる。ここで B_t は上の例では「挨拶をする」といった行動を意味するものであるが、行動変数はまだ与えられていない。 B_t と行動変数 v_t が組み合わさって具体的な行動 b_t が決定される。

さて、ロボットの行動変数は内部状態から決定されると考えることができる。ただし、先述したように内部状態 i や更新関数 U は単純な if-then ルールでも利用される概念なので、混乱のないよう、独立して記述する方が良いと思われる。行動変数に関する内部状態が既存研究のパラダイムにおける内部状態と独立しているならば、既存研究と組み合わせることも比較的容易であるという利点もある。そこで、通常の内部状態 i ではなく、行動変数に関する内部状態については本研究ではコンテキストと呼ぶことにする。コンテキストは次式のように更新される。

$$\begin{cases} v_t = V(c_t) \\ c_{t+1} = U_c(c_t, s_t, b_t) \end{cases} \quad (2.4)$$

このとき、ロボットの行動パターンはコンテキスト c 、コンテキストから行動変数を作成する関数 V 、コンテキストの更新関数 U_c によって特徴づけられる。そこでこの (c, P, U_c) の組を**モデル**と呼ぶ。

本研究では、以下に挙げる二つのモデルを提案する。

1. **観測指向モデル**: 他のロボットの行動を観測しあうというインタラクションから多様な行動変数を生成するモデル
2. **属性変換モデル**: 物体の属性情報から複雑な多様な行動変数を生成するモデル

さて、本研究で目的とする多様な行動パターンは、行動の系列 b_0, b_1, b_2, \dots が複雑であることを意味する。特に、本研究においては行動変数の系列 v_0, v_1, v_2, \dots が複雑に変化することで複雑な行動系列を実現する。

前章で述べたように、本研究で目的とする行動パターンの複雑さの条件は次の3つであった。

- 局所的には安定し、変化しないこと
- 大域的には変化する可能性があること

局所的な安定性は、時間的に隣接する行動変数、つまり v_t と v_{t+1} が同一であることが多いことを意味する。一方で大域的な変化とは、行動の系列が与えられたとき、ある時刻 t と時刻 u のあいだに十分な差があるときに v_t と v_u は類似しないことがあることである。もしくは次のように考えられる。大域的な変化が生じないときは行動変数の系列は同一の

ものだけで構成されるようになる。したがって、行動変数の系列全体で見たときに特定の行動変数のみに偏ることがなければ大域的な変化が生じていると言える。

最後に、変化するまでの期間は次のように考えられる。ある時刻 t から $t+n$ のあいだの期間で行動変数の系列 $v_t, v_{t+1}, \dots, v_{t+n}$ がすべて同一であるとき、その期間では同じ行動が維持されているという。また、一定の時間において同一の行動変数が維持される期間の出現回数が多ければ、その時間のあいだでは頻繁に行動変数の変化が起きている。したがって行動変数の系列全体に存在する期間の数から局所的な安定性を調べることができる。

以上で本研究の目的を定式化した。次章では、既存の研究について紹介し、本章で挙げた問題とどのような関係を持っているかを明らかにする。

第3章

関連研究

本章では、エンターテインメントロボットに関連する既存研究の概略を述べ、本研究との関連を明らかにする。まず、既存研究としてロボットが新しい行動を獲得する手法の研究について概略を述べ、これらの手法だけでは行動の多様性を実現できないことを指摘する。次に、感情や心理状態をモデル化しロボットに組み込むことで複雑な行動選択を可能にする既存研究について説明し、それらの研究について本研究の目的との関連を述べる。

3.1 行動および行動規則の獲得に関する研究

現在、ロボットの行動は設計者によってひとつずつ実装されている。しかし、ロボットの行動を設計するのは非常に手間のかかる作業である。また、手作業では作成される行動の種類にはどうしても限界があり、多様性をもたらすことは難しい。もしロボットが自律的に新しい行動を獲得できれば、こうした問題は解決されることが考えられる。

本節では、ロボットが新しい行動や行動規則を獲得することを目的とする既存の研究について概略を示す。まず行動そのものを作りだし獲得する研究について説明し、次に行動を実行するかどうかを決める行動規則を獲得する研究について述べる。そして、これらの研究と本研究との関係について述べる。

3.1.1 新しい行動の獲得

山田らは動物をしつけるとき用いられる古典的条件付けアルゴリズムを用いて AIBO に新しい行動を獲得させている [山田 他 02]。古典的条件付けとはパブロフの犬で知られる実験で用いられる手法である。犬に餌をやるときに必ずメトロノームの音を鳴らす。最初はメトロノームの音という条件刺激を与えられても、犬は無条件反応である“注意を向ける”という反応しか行なわない。しかし、条件刺激であるメトロノームの音と無条件刺激である餌を同時に犬に与え、餌に対応する無条件反応の唾液分泌を実行させることを何度も繰り返すと、本来関連のない条件刺激であるメトロノームの音と無条件反応である唾液分泌が関連づけられ、最後には、餌が与えられなくとも、条件刺激であるメトロノームの音が与えられただけで、学習により関連づけられた反応である唾液分泌が行われるようになる。

山田らはこのメカニズムを AIBO に実装した。無条件刺激としておしりを叩く行為を

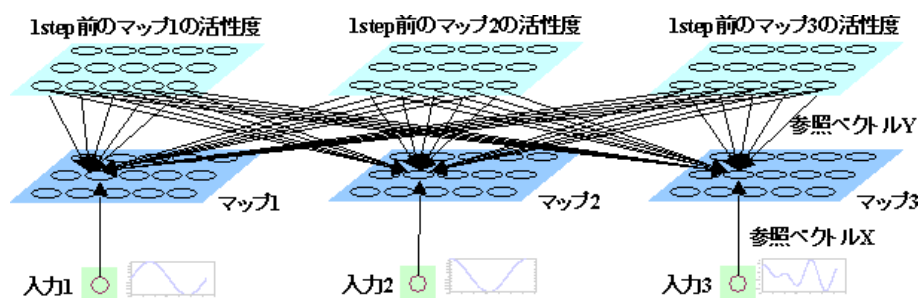


図 3.1: 太田の並列再帰型自己組織化マップ (in [太田 03])

行い，これに対応する無条件反応として AIBO は前進することとした．また条件刺激としてボールを見せ，最終的に AIBO にボールを見せることで前進するという条件反応を引き出す学習を行なった．

この実験からは条件反応によって新しく，ボールを見ると前に進むという行動を AIBO が獲得していると言える．

また，太田はエンターテインメントロボットが既に獲得した行動を変化させて自ら新しい行動を生成する行動進化ソフトウェアを提案した [太田 03]．太田のシステムでは，再帰型自己組織化マップを拡張した並列再帰型自己組織化マップを用い，入力軌道パターン間の相関関係を学習させている (図 3.1)．

複数の再帰型自己組織化マップが併置されている．図 3.1 におけるマップ 1，マップ 2，マップ 3 はそれぞれセンサに対応する．各マップは 1step 前のマップと結合しており，これによってセンサ間の関係や時系列の学習が可能となる．

太田は AIBO を用い実証実験を行なっている．実験では，AIBO にあらかじめ“お手”と“お座り”と“立ち”という 3 つの状態を学習させている．AIBO が座っている“お座り”状態が初期状態であり，人間が手を近づければ AIBO が左前脚を伸ばす“お手”を行う．また“お座り”の状態で頭を撫でると脚を伸ばして立ち上がる“立ち”状態に遷移し，“立ち”状態で手を近づけると“お座り”状態に遷移する．以上の状態遷移を自己組織化マップに学習させた．

このように学習させた後で AIBO を立たせて手を近づけたところ，最初に学習させた“お座り”に戻るといった状態にならず，脚を曲げずに左前脚だけを伸ばす“立ちながらお手”という新しい行動が生まれたことを太田は報告している．この新しい行動は，学習の結果として“お手”が距離センサと左前脚のマップによって決まることを学習した結果として生じるものであることが考察されている．

3.1.2 行動規則の獲得

本田は人間の身体動作とロボットの行動の対応関係を構築することで，人間からの働きかけに対する反応を学習するシステム HURMA を提案した [本田 04]．

HURMA は学習フェーズとクエリフェーズに分かれる．学習フェーズでは，ロボットは自分が何らかの行動を実行すると，そのときの人間の身体動作データを対応づけて記憶す

る。一方、クエリフェーズでは人間の身体動作を獲得し、これをクエリとして対応する行動を探索する。人間の身体動作は関節角情報として取得され、最長共通部分列(LCSS)によって過去の身体動作とのマッチングが調べられる。そしてマッチした身体動作が発見されると、対応する自分の行動を取り出し、実行する。

たとえば人間が両手を挙げるとロボットは“抱き付く”という行動を実行し、右手を差し出すと“握手する”という行動が実行されることが報告されている。

3.1.3 本研究との関連

エンターテインメントロボットにおいて、行動の種類が少ないことは問題である。また、設計者が自分で行動を実装するのにも限界がある。そこで、ここで挙げたように新しい行動をロボットに獲得させるという研究が行なわれている。

しかしながら、このような既存研究ではロボットの行動は非常に単純な構造となっている。つまり“手を差し出したらお手をする”といったように、単純な条件と実行される動作系列が強く結びつけられている。したがって、これらのアプローチによってロボットの取りうる行動の種類は増えるが、ロボットの行動の多様性の実現にはまだ足りない点があると考えられる。それは既存研究には、どの行動を取るべきかという行動選択における多様性が欠けているからである。

たとえば、太田の研究では、ロボットは初期状態で学習させた“手を差し出すと座る”という関係を放棄し、新しく“立ったまま右前脚を差し出してお手の動作をする”という行動が実行されている。本来であればロボットは“お手”か“お座り”のどちらにするかを選択する必要があるが、太田の研究では一方の関係だけが優先されているのである。

本研究では行動変数を様々に変化させることで複雑な行動パターンを実現させるというアプローチを取る。ただし、ロボットが持っている行動の種類は増えない。

したがって本研究とこれらの既存研究は相補的な関係にあると言える。すなわち、既存研究によってロボットは行動を獲得し、本研究の提案するモデルによって行動パターンに多様性を与える。このことでエンターテインメントロボットの行動をより複雑に変化できるようにになると考えられる。

3.2 感情モデルによる行動選択に関する研究

一般に、ロボットの行動は設計者があらかじめ設定した行動規則にのっとって決められる。しかし、エンターテインメントロボットなどでは単なる行動規則での設計が難しい場合が多く見られる。たとえばエンターテインメントロボットは人間に似た顔を持ち、何らかの表情を表出することが多い。表出する表情を決めるには、感情を内部状態として持つということが考えられる。

また、ロボットが勝手に行動するとき、どの行動を実行すればいいか、ということもあらかじめ決めておくことが難しい。そこで、ロボットに動機や欲求といった変数を与えることで行動を選択させるという方式が考えられる。

ロボットの内部に感情や欲求といった変数を与え、その内部状態をもとに行動を決定す



図 3.2: Murphy らのロボット (in [Murphy *et al.* 02])

るという手法は既にいくつも提案されている。以下ではその例として、まず感情状態と通常の制御ルールを組み合わせて複数ロボットの協調に用いた例を挙げる。また、感情状態にもとづいて表情などを表出する例としてアザラシ型ロボットのパロと顔ロボットの WE-4RII, 欲求にもとづく行動決定メカニズムの例として人型ロボットの QRIO と顔ロボットの Kismet を例にとり、説明する。最後に、これらの研究と本研究の関連について述べ、本章をまとめる。

3.2.1 感情を用いたロボット二台の協調システム

Murphy らは、感情にもとづいて2体のロボットで協調動作を行うシステムを提案している [Murphy *et al.* 02]。このシステムはパーティで給仕を行うというタスクを遂行するロボットである(図 3.2)。ただしロボットの役割は分かれており、一体は実際に飲料を運んでパーティの参加者に配るウェイターロボット、もう一体はウェイターロボットの飲料の残量が少なくなったら補充する補充ロボットである。

この研究の感情モデルは Frijda の感情の理論 [Frijda 86] にもとづいており、ウェイターロボットは HAPPY, CONFIDENT, CONCERNED, FRUSTRATED という4つの感情状態、補充ロボットは HAPPY, CONFIDENT, CONCERNED という3つの感情状態を持つ。感情状態の遷移は、ウェイターロボットの場合は飲料の残量や減少率、補充ロボットはウェイターロボットからのリクエストの有無によって決まっている。ロボットは、もともとの役割にあわせた行動選択規則に加えて、感情状態によって行動を変更する。

このロボットは次のように動作する。ウェイターロボットは最初は HAPPY 状態で周囲に給仕しているが、次第に保有する飲料の数が減っていく。すると感情が CONFIDENT に変更され、補充ロボットにコマンドを発行する。補充ロボットは何もリクエストがない状態が HAPPY で、その場合は待機しているが、ウェイターロボットからリクエストが発行されると移動を開始し、ウェイターロボットにトレイを運ぶ。補充が済むとウェイターロボットはまた HAPPY に戻り、補充ロボットは定位置に戻って待機する。しかし、



図 3.3: パロ (in [Wada *et al.* 05])

混雑状況などで補充ロボットがなかなか到達しないとウェイトロボットの感情状態は CONFIDENT から CONCERNED や FRUSTRATED に遷移する。するとウェイトロボットは hurry コマンドを発行する。hurry コマンドが発行されると、補充ロボットは安全走行速度から最大速度に変化して移動する。

このように Murphy らのシステムでは、感情は単なる状態遷移でしかなく、行動決定ルールの一部に HAPPY や FRUSTRATED という名前をつけているだけであるとも言える。この研究において Murphy らが目的としているのは、感情というメカニズムを導入することで循環的に依存するような複雑なタスクでも中央集中的な管理システムなしで適切に動作させることである。このため、本研究で目的とする多様な行動はまったく実現されない。

3.2.2 パロ

パロ [柴田 99] はアザラシ型の医療介護ロボットである。パロの写真を図 3.3 に示す。

ただし、医療目的といっても、パロは手術ロボットのような医者の作業を支援する機能は持たない。パロにできるのは、四肢を動かしたり、瞼を閉じたり、鳴き声を上げたりするといった単純な動作の組み合わせだけである。

パロは医療における作業支援を目的としているわけではなく、患者に心理的な効果を与えることを目的としているロボットである。パロの動作は感情の表出のみだが、人間がパロに対して撫でるなどの働きかけをすると何らかの反応を返す。このインタラクションの結果によって人間は癒しを得る。もともとは、患者が動物とインタラクションを行うことによって心理的に癒やされる効果を得るアニマル・セラピーと呼ばれる手法をパロはベースにしている。アニマル・セラピーにおける動物を動物型のロボットで代替した手法であ

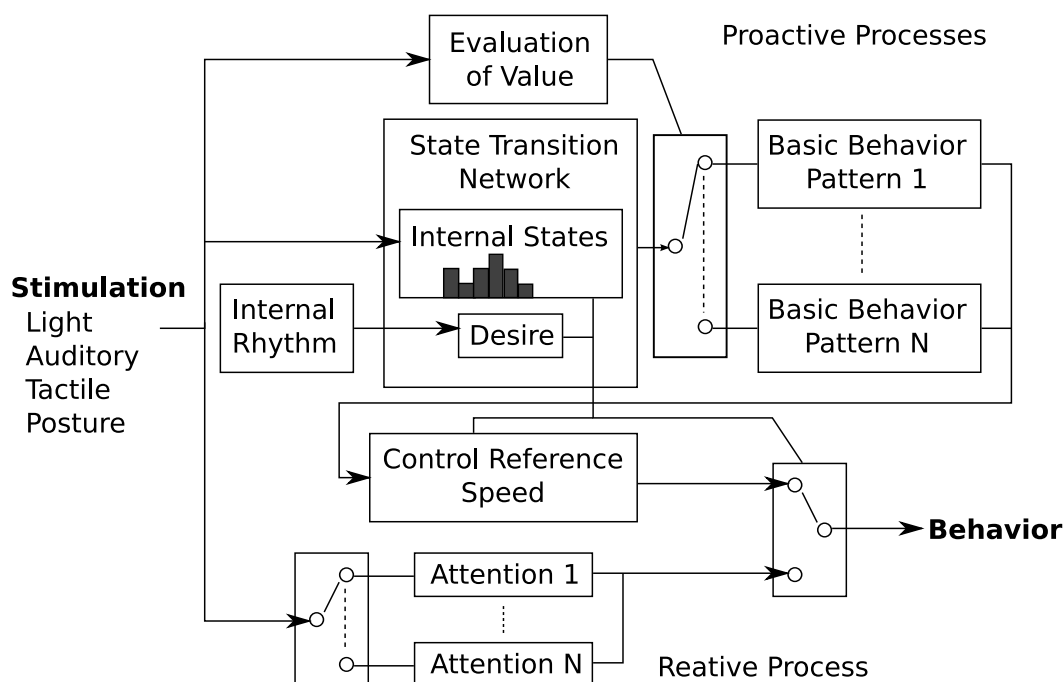


図 3.4: パロの動作決定機構 (in [Wada et al. 04])

り、ロボット・セラピーと呼ばれている。

パロは介護老人施設などの場所で実際に利用されており、ロボット・セラピーについても一定の効果があることが報告されている [Shibata et al. 01b][Wada et al. 04]. 特に、アレルギーなどの問題でアニマル・セラピーを適用できない患者でも同等の効果を与えることができるとして注目されている。

パロは、人間からは撫でられるといった働きかけを受ける。パロはこれを受けて、内部の感情状態を更新し、自分の次の動作を決定している。パロの動作決定機構を図3.4に示す。

パロの動作決定機構は、proactive, reactive, physiological という3種類のメカニズムを組み合わせることで動作を決定する。このうち reactive とは反射的な行動を意味する(図の下部の Ractive Processes に対応する)。たとえば、突然大きな音がすると音のした方向に注意を向けるという動作が生成されるといったものである。また、physiological の行動決定機構は生体的なリズムを表現している。たとえばしばらくすると眠るというようなことを実現している(図中の Internal Rhythm に対応する)。これらのメカニズムは非常に単純なものであり、残りの proactive のメカニズムがパロの動作を主に決定していると言える。

パロは内部状態として、いくつかの感情を持っており、感情にはそれぞれ「うれしい」「悲しい」などの名前と、現在その感情をどれくらい強く感じているかを示す値を持つ。この数値は人間の働きかけなどによって生じる外部からの刺激と、physiological のメカニズムによって生じる生体リズムによって変化する。パロは基本動作 (Basic Behavior Pattern) をいくつか備えている。これは「ヒレを振る」といったような単純な動作であり、感情状態によって実行される動作が選択される。さらに同じく感情状態によって、選択された動



図 3.5: WE-4RII (in [三輪 04])

作の繰り返し回数や速度が変化し、様々な動作が実現される。これらの動作の組み合わせにより、実際のアザランに類似した行動が実現されている。

このように、パロは様々な動作を実現することができるが、内部状態の変化は固定的である。つまり、人間の働きかけによって生じる内部状態への変化は固定されているため、人間の働きかけに対して同じように反応することになる。パロが目指している多様な反応は動作レベルのものであり、本研究における行動のバリエーションの問題は解決されない。

3.2.3 WE-4RII

WE-4RII は早稲田大学のヒューマノイドロボット研究所で開発された顔ロボットである [三輪 04]。WE-4RII の写真を図 3.5 に示す。

WE-4RII は人間の上半身を模した外見と自由度を有するロボットである。特に眉や口、瞼や目などで細かい制御が可能となっており、様々な表情表出が可能となっている。本研究にとって興味深いのは、内部に感情状態を備えており、その感情に従って表情を変化させるということである。

WE-4RII の感情モデルの構成を図 3.6 に示す。

WE-4RII の感情は Russell のモデル [Russell *et al.* 85] と Smith のモデル [Smith *et al.* 85] を組み合わせたものであり、快度、覚醒度、確信度からなる三次元の心理空間として表現される。この感情空間は、図 3.6 で示されるように分割され、それぞれ名前がつけられて

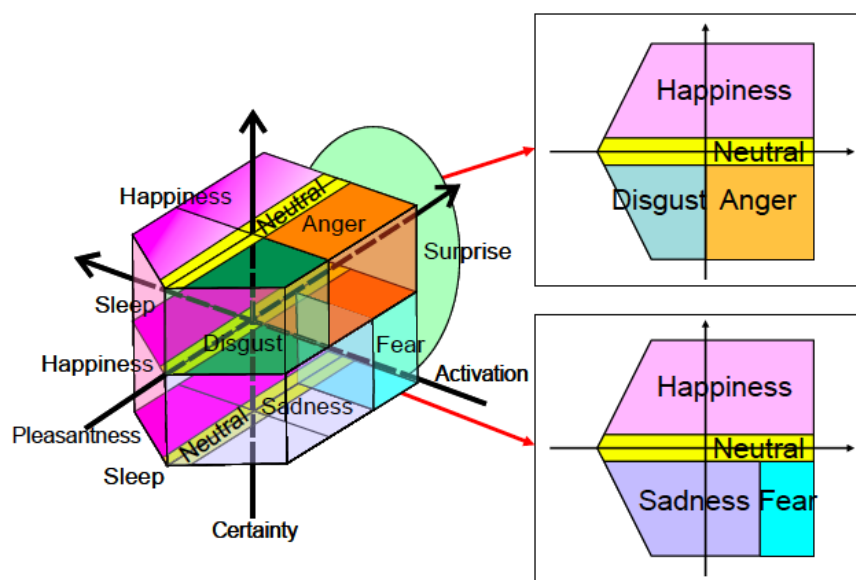


図 3.6: WE-4RII の心理モデル (in [三輪 04])

いる。WE-4RII の感情はこの感情空間内のベクトルとして表現され、外部刺激や内的な欲求によって感情空間内を移動し、感情の変化が表現される。そしてロボットは、現在の感情ベクトルが位置する区画に応じて自分の表情を変化させる。

WE-4RII では、3種類の方法でロボットに様々な反応を引き起こさせることが試みられている。

第一の方法は、感情に変化を与える方程式の係数を様々に変えるという方法である。この方法では、係数はあらかじめロボットに組み込まれているため、あとから変更をすることができない。ロボットは感情の変化に応じて「怒りやすい」などの個性を獲得できている。

第二の方法は、外部の刺激と快不快を結びつけさせることで、学習させるという方式である [Miwa et al. 03a][Miwa et al. 03b]。たとえば、緑色のボールを見せながらロボットをなでる。ロボットはなでられるのが快いという関係があらかじめ組み込まれているため感情が快い方向へ変化する。すると緑色のボールという外部刺激と、快いという感情との関係が学習されるようになる、という方式である。

第三の方法は、内的な欲求を定義するものである。人間に生得的な欲求である食欲や安全欲求などを定義し、また、より低次の欲求が満たされてはじめて高次の欲求が生まれるという Maslow の欲求階層論 [Maslow 70] に従ってモデル化している。ただし、このモデルは食欲や安全欲求のような生命の維持に関する欲求をモデル化するものであり、好奇心のような生命の維持と直接関係のない欲求が定義されない。そこで三輪は独自に、未知の刺激に遭遇したときに高まる探索欲求を定義している。

これらの方法により、ロボットは物体との関連を学習できる。しかしながら第一の方法はロボットにあらかじめ個性を組み込んでおく方法である。感情が変化するパターンは変化しないため、本研究の目的となる多様な行動パターンは実現されない。第二の方法は、



図 3.7: QRIO (in [Tanaka *et al.* 05])

言わばロボットの個性を後天的に変化させる方法であるが、対応関係は人間があらかじめ教示しなければならないし、教示した対応関係は変化することがない。第三の方法は、ロボットの内的な欲求を定義するものであるが、その欲求は食欲や未知の刺激への反応など、根源的ではあるものの、外部刺激から一意に定まる反射的なものに過ぎない。たとえば未知物体が発見されたとき、それに注意を向けるが、それを快不快のどちらに判断するかは、第二の方法によって人間から教示されなければならないのである。

つまり、WE-4RII は心理的なモデルを忠実に再現しているが、物体への探索欲求は、けっきょく新奇な物体に注目するという非常に単純なものである。また、快不快のような反応は人間が教示してやらなければならない。このため、物体に相対するときのロボットの反応は人間が教示したままであり、多様性は生まれない。また、環境中に複数の物体があるといった状況は想定していない。さらに困難な問題としては、視界内に快い物体と不快な物体が同時に存在したら、どのような表情を表出するべきなのか、また一方を無視するとした場合どちらを無視するか、といったような問題は想定していない。

3.2.4 QRIO

QRIO(SDR-4X) [黒木 03][Kuroki *et al.* 03] はソニーで開発された人型ロボットである (図 3.7)。

QRIO は舞いや走行、サッカーボールのシュートなど様々なデモンストレーションを実行できるロボットであるが、本節では行動を決定するためのメカニズムに注目する。QRIO

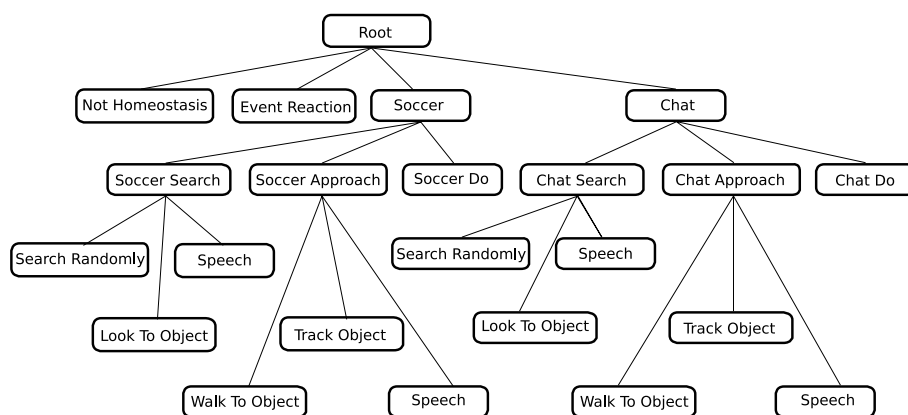


図 3.8: QRIO の行動の階層構造 (in [Sawada *et al.* 04])

の行動システムについては澤田らによって報告されている [Sawada *et al.* 04]. この報告によれば QRIO の行動は階層構造を取っており, QRIO は適切な動作系列を生成できる. QRIO の行動の階層構造を図 3.8 に示す.

行動の階層構造は意味的な構造になっている. つまり, 最上位のノードは行動の最終ゴールを意味しており, そのゴールを実現するサブゴールとして下位ノードが定義されている. たとえば最上位ノードが“サッカーボールを蹴る”だとすると, その一段下のノードは“ボールに近づく”, “足の位置を合わせる”, “蹴る”といったものとなり, さらに“ボールに近づく”の下は“ボールを発見”, “位置の同定”といったサブゴールに分解される.

澤田らはこれらをまとめて行動 (behavior) と呼んでいるが, 本研究の定義に従えば最上位のノードが行動であり, その下位ノードは行動を実現するための動作系列を表現したものと考えられる.

QRIO の行動は Activation Level (AL) というパラメータで決定される. AL はその行動を実行したい度合いを占めず抽象的な値であり, 実際には motivation (Mot) と releasing value (Rel) という 2 つの要素に分解される.

$$AL = W_{mot}Mot + (1 - W_{mot})Rel \quad (3.1)$$

ここで Mot はその行動を実行したいという内的な状態を表しており, Rel はその行動を実行した結果として生じるであろう刺激を意味している. つまり, Mot は行動をやりたいかどうかという動機であり, Rel は行動を実行した結果がどれほど素晴らしいかという期待感と解釈できる. この両者を W_{mot} によって重み付けした和から AL が決定され, 実行する行動を選択される.

ロボットには NOURISHMENT や FATIGUE, VITALTY といった内部状態が用意されており, それぞれ固有の値を持っている. ある動作を実行するときの, これらの内部状態の値の変化の重みつき総和が motivation となる. また, 内部状態から満足度を計算することができ, この満足度の増量の重みつき総和が releasing value となる.

澤田らは, 内部状態や行動によるパラメータや満足度の変化などを適切に設定することで, サッカーボールを蹴ってゴールに向けてシュートする, というタスクを QRIO が遂行



図 3.9: Kismet (in [Menzel 01])

できたことを報告している。澤田らはまた、この機構によって人間との対話も実現できるとしている。

Mot と *Rel* という2つの欲求により、QRIO は適切な行動を実行できる。そこで問題となるのは、初期の内部状態をどのように与え、外部刺激によってそれがどのように変化するのかというモデルである。しかしながら澤田らの報告ではその点は言及されていない。初期の動機を人間が与えることで、予定した動作を実行させるのである。外部刺激から内部状態が複雑に変化すればロボットの行動は多様に変化すると考えられるが、このシステムだけでは実現されない。

3.2.5 Kismet

Kismet は MIT で開発された顔ロボットである [Breazeal *et al.* 99] [Breazeal 02]。Kismet の写真を図 3.9 に示す。

Kismet は人間の幼児の行動を模したロボットである。周囲の画像を認識し、明るい色合いの物体や移動物体、人間の顔に注目して視線を追従する動作が組込まれている。また、心理学にもとづいた感情モデルを持ち、内部の感情や事前に設計された疲労度にもとづいて表情を変化させる。

Kismet とのインタラクションでは、人間は幼児に対して世話を焼く大人の役割を演じることとなる。人間が玩具を動かすと、その動きにつられて Kismet が視線を動かす、といったふるまいが実現され、これによってロボットと人間のための社会的なインタラクションが実現される。

Kismet の欲求モデルを図 3.10 に示す。

Kismet は3つの欲求を持っている。一つは社会欲求 (social drive) であり、この動機は人間と視線を合わせるという行動に結びついている。二つめは刺激欲求 (stimulation drive) であり、玩具に視線を向けるという行動に結びついている。三つめは疲労欲求 (fatigue drive) であり、何も反応しないという行動に結びついている。図 3.10 は、この欲求と行動の関係を示している。

Kismet の欲求は外部刺激とのフィードバックによって発生する。Kismet は画像の入力から、人間の顔を発見するアルゴリズムや、派手な色や動きを発見するアルゴリズムに

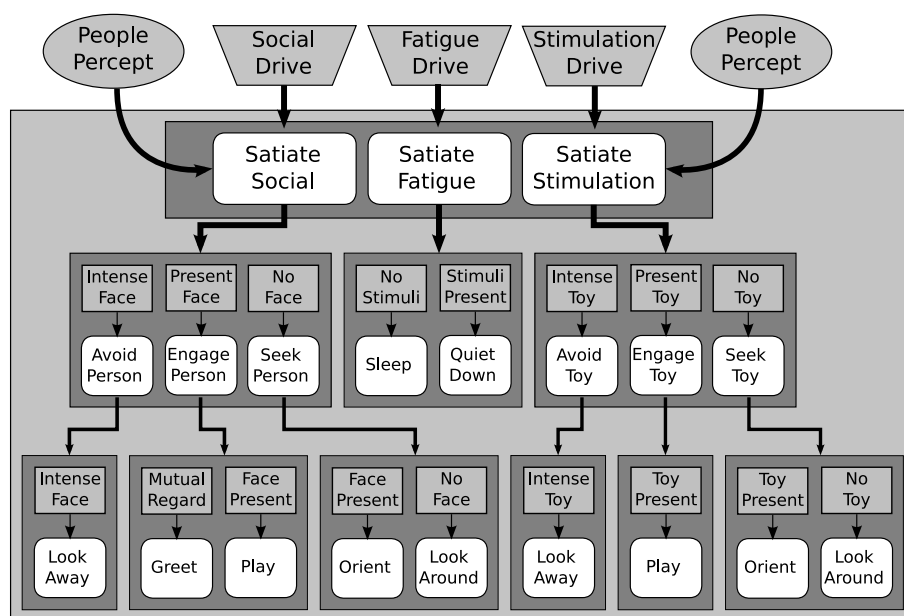


図 3.10: Kismet の動機モデル (in [Breazeal et al. 99])

よって特徴を抽出する。そして、抽出された特徴に対してその時点での欲求を用いて重みづけすることで注意を向ける対象を絞り込む。また、視界内の特徴に基づいて欲求はフィードバックを受ける。たとえば、人間がいなければ社会欲求は発生しないし、逆に人間が存在することで社会欲求は高まる。このように、周囲の環境から欲求が生じ、欲求によって行動を選択するというモデルになっている。

Kismet の感情モデルを図 3.11 に示す。

Kismet の感情は Ekman のモデル [Ekman et al. 94] に基づき覚醒度 (arousal)、valence、態度 (stance) という 3 つの次元をもつ感情空間で表現される。また、感情空間は図で示される区間に分割されており、現在の感情の位置に従って表出する感情を決定する。たとえば、valence が positive で覚醒度が中立の場合は幸福 (happiness) となる。Kismet の感情は現在の欲求と行動から影響を受ける。Kismet が最終的に出力する動作は、欲求によって決定される行動と感情状態によって決定される表情によって決まる。

以上のように、Kismet は感情モデルに従って表出する表情を決定し、欲求モデルに従って行動を決定する。この 2 つの要素によって様々な反応が引き出される。しかしながら、欲求は画像情報に基づいて決定されるから、たとえば人間と玩具がともに視界内に存在したとすると、どちらを優先するかは決まってしまう。また、視界内に複数の玩具があるときは、単純に色が派手で大きいものを選んでしまう。複数の人間がいるときは、先に発見された方が単純に選択される。Kismet の行動は様々に変化するが、環境が与えられれば一意に定まってしまうため、行動のバリエーションは実現できていないと言える。

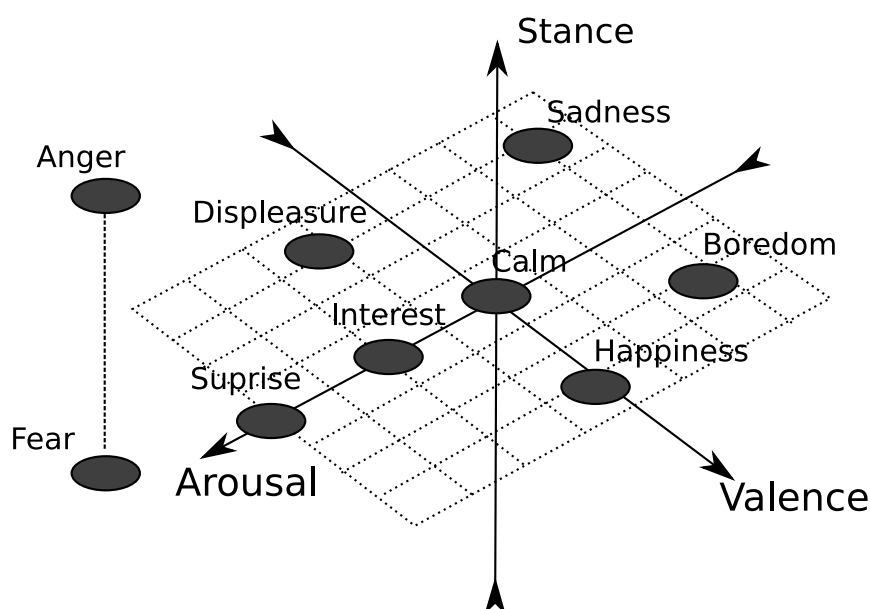


図 3.11: Kismet の感情モデル (in [Breazeal *et al.* 99])

3.2.6 本研究との関連

以上，感情や欲求といった生物的なモデルをロボットに導入した行動システムについていくつか紹介した．次にこれらの研究を総括し，本研究の意義について確認する．

これまで紹介してきたように，感情や欲求についての何らかの理論にもとづいてモデル化するという研究は数多くなされている．しかし，ここで紹介したように，感情を実現するためのアプローチは類似していないことがわかる．

そもそも人間の感情や欲求のようなメカニズムについては解明が進んでいない．現在でも様々な理論が提唱されているが，その妥当性は明らかでない．従って，ロボットの設計者は何らかの理論にもとづいて感情モデルを設計しているものの，どのモデルが妥当であるとされているかははっきりしていないというのが現状である．また，同じ理論をもとにしても実現方法がまるで違う例もある．

従って，ロボットに感情モデルを与える際に重要なのは根拠となる理論ではない．むしろ，感情モデルを導入することで何が実現されるかに着目する方が重要である．

たとえば Murphy らのシステムは複数台のロボットが給仕というタスクを行うことを目的としており，中央で集中管理しなくても複雑な依存関係を持った行動が問題なく遂行できるというものだった．Murphy らの感情モデルは非常に単純であるが，この目的に対しては有効である．

さて，感情モデルを導入するシステムの目的は，次の2つに分けることができる．

1. 頑健性
2. 多様性

ここで頑健性とは，周囲の環境が多少は変化しても正しく動作を決定できることを意味している．ロボットが人間の生活環境などで動作するという状況を考えると，ロボットの

動作環境はこれまでの産業ロボットと比較すると極めて乱雑で、何が起こるかあらかじめ完全に想定することはできない。そこで、内部状態として感情のようなパラメータを用いてロボットの動作を制御することで環境の複雑さに対処する。Murphyらのシステムはこの典型例である。またQRIOの行動選択システムもこれに該当する。Kismetの行動システムのうち、欲求にあたるのもこちらである。

QRIOでは、motivationやreleasing valueという変数を用いることが適切かは疑問の余地があるかもしれない。仮に、サッカーボールを蹴るという行動だけを実現したいのであれば、行動ツリーのなかから適切に選択する方法は他にもあるはずだからである。しかし、ロボットが直面する状況が様々なものに変化しても適切に動作を選択するためには、欲求という変数に沿って行動ルールを設計することが有効となる場合もあると考えられる。

Kismetの欲求システムは非常に単純で、社会欲求と刺激欲求と疲労欲求という3つで駆動される。これらの欲求は視界内の情報からフィードバックを受け、注意を向ける対象を決める。これによって、視界内に玩具と人間がともにあるとき、どちらの行動を選択するかという調整を目的とするメカニズムとして欲求が存在していると言える。また、調整メカニズムによって、玩具に注意を向けている行動から人間に視線を向ける行動へと次第に変化させることが可能である。

つまり、頑健性を目的とする感情モデルにおいて問題としているのは環境とタスクの複雑さである。取るべき行動が複数存在する状況や、環境が複雑すぎてどの行動を選択していいのかわかりづらい状況では、中間的な行動を実現したり、うまく一方の行動を選択するメカニズムとして感情を用いることは有効である。

ただし、頑健性を目的とした感情モデルの場合にはロボットの反応を多様に変化させることは難しい。たとえばKismetの視界内に人間がいて玩具もあるとき、人間と玩具のどちらを優先するかは、最終的にどちらの欲求がより強くなるかということによって決まる。欲求の上限はKismet内に組み込まれているのだから、どちらの欲求を優先するか決められない場合にはあらかじめ決められている方を選ぶことになる。このため外部状態が同じならロボットはつねに同じ行動を選択してしまうが、エンターテインメントロボットではこれが問題となる。何らかの手法で、頑健なだけでなく多様な反応を実現する必要がある。

二番目の多様性とは、感情によって動作を様々に選択し、あるいは組み合わせることを目的とするものである。パロ、WE-4RII、Kismetなどがこれにあたる。本研究も、ロボットの行動パターンの多様性を目的とする。しかし、これらの研究でも多様性についての検討がなされているが、実際には本研究で目的とする点は満たされていない。

パロの行動システムを例に取って説明する。パロの行動システムでは、尾を振るとかいった基本動作があらかじめいくつかが定義されている。そして現在の感情状態に従って基本動作を組み合わせ、あるいは実行速度や繰り返し回数を変化させ、多様な動作を実現するというものである。つまり、パロが実現しているのは感情状態に対応する動作系列を動的に作り出す機構である。パロの感情状態は単純に“怒り”や“悲しみ”や“喜び”といったシンボルではなく、その中間的な状態も考慮に入れなければならないものである。従って、個々の感情状態ごとに一つ一つ対応する動作を設計していくのは困難で、限界がある。そこで、パロでは基本動作を現在の感情状態に応じて組み合わせるアプローチを取っている。

しかし感情状態の遷移は決定的である。つまり、外部刺激と感情状態の変化は一对一の対応となっており、変化しない。同じなでられるという刺激でもどのような感情がどのように変化するかは時と場合によるだろう。心地良くなることもあれば、逆に怒ることもある。パロではこの問題に対処できず、感情状態の遷移は単調である。

Kismet や WE-4RII も同様である。WE-4RII では、感情状態の変化の仕方を変える方法も検討されている。とはいえ、その変化は“新奇な物体があれば注目する”といったものや“人間が学習させることで物体に対する態度を学習する”といったものである。また Kismet でも疲労度が蓄積することで次第に反応が鈍くなることが実現されている。しかし、いずれにせよ本研究で述べるような多様性は実現されていない。

感情状態の変化を複雑にするためには、単純なルールだけではない方法によって変化を引き起こす必要がある。本研究の目的はまさにこの問題を解決することである。本研究で提案するモデルでは、周囲の環境から影響を受けつつ、確率的な手法によって内部状態を複雑に変化させる。内部状態の複雑な変化によって、ロボットは自発的に様々な行動を選択し、複雑な行動パターンが実現される。

第4章

観測指向モデル

エンターテインメントロボットでは、ロボットと人間のインタラクションが重要である。そこで、そのインタラクションによって互いに影響を与えあい、複雑な行動を実現するものとして、本研究では観測指向モデルを提案する。

観測指向モデルでは環境中に複数のロボットや人間などが存在するという仮定を置く。そして、ロボットは自分自身や他のロボット、人間の動作を観測し、その影響を受けて行動変数を変化させる。すると、やがて環境中のロボットたちはグループを形成して類似した行動変数を生成し、安定する。ただし、観測には意図的に誤りが混入されている。この誤りのためロボットはまれにグループとは異なる行動を取ることがある。異なる行動はインタラクションを通じて他のロボットに影響を及ぼすことがあり、最終的には行動変数の大域的な変化が生じるというモデルである。

本章では、観測指向モデルの構成を説明し、シミュレーション環境においてモデルの特徴を調査する。

4.1 概要

観測指向モデルは、郡司らによる不定な観測にもとづくセルオートマトンのモデル [Gunji *et al.* 96, Gunji *et al.* 97] にもとづくロボットの行動モデルである。

郡司は、セルオートマトンの状態として $\{0, 1\}$ の二値のほかに、第三の値である α を導入した。ただし α は内部状態としては存在するが外部から観測できない特別な値である。このため、外部から観測するとオートマトンの状態は $0, 1$ の二値となる。

郡司らのモデルでも、用意された遷移規則と周囲の状態から各オートマトンが自己の次の状態を決定することを繰り返し、時間発展する点は通常のセルオートマトンと変わらない。しかし、 α のため、状態遷移をあとから観測しなおすと、その時点での遷移規則との間に矛盾が発生することがある。そして、矛盾が発見されると、その矛盾を解消するべく遷移規則を書き換える。このことで遷移規則は動的に変化する。このモデルによって、実際に時間発展させると初期の遷移規則によらずカオス的なパターンと安定なパターンが混ざった複雑なパターンが生成される。

観測指向モデルは、郡司らのモデルにおけるオートマトンをロボットに当てはめる。セルオートマトンが左右の状態から自己の状態を変化させるように、ロボットは互いの行動を観測しながら自己の状態を遷移させる。そして、自己の状態の時間発展によって複雑な

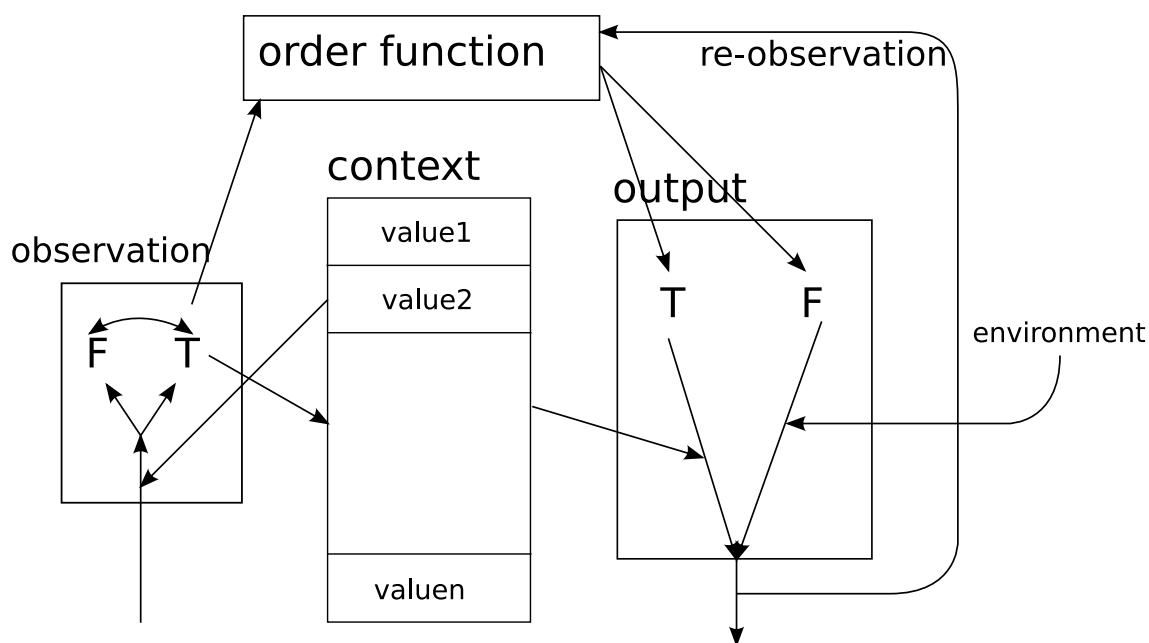


図 4.1: 観測指向モデルの概略

パターンを生成するモデルとした。

4.2 システム構成

観測指向モデルの基本的な構成を図 4.1 に示す。

観測指向モデルは「コンテキスト (Context)」「観測モジュール (Observation Module)」「出力モジュール (Output Module)」「秩序関数 (Order Function)」の4つのモジュールで構成される。この中で、秩序関数は郡司らのモデルにおける状態遷移関数に対応する。観測モジュールでは、センサデータを解釈してその結果を秩序関数に渡す。出力モジュールは秩序関数の戻り値を解釈して実際の行動変数を与える。コンテキストは郡司らのモデルにおいて真値に対応する関数に相当し、観測時のセンサデータと $\{0, 1\}$ の対応を与える。

次の各節で、個々のモジュールの説明を行う。

4.3 観測モジュールとコンテキスト

観測モジュールは周囲のロボットの行動を観測するモジュールである。観測モジュールの役割は二つある。第一はコンテキストと照合して周囲環境の状況を調べる。この観測結果に従って行動変数の決定が行なわれる。第二は観測結果に基づいてコンテキストの更新を行うこと。2章で述べたモデルの記述に従えば、観測モジュールはコンテキスト更新関数 U_c と、行動変数導出関数 P に相当する。

```

( 1) param = context[random()]
( 2) for b in observations do
( 3)   if similar(b, param) then
( 4)     add True to results
( 5)     context[random()] = b
( 6)   else
( 7)     add False to results
( 8)   end
( 9) end
(10) return results

```

図 4.2: 更新プロセス

一方、コンテキストは行動変数の集合であり、過去の観測結果を示す。すなわち、

$$c = \{p_0, p_1, \dots, p_{l_c}\} \quad (4.1)$$

ここで、コンテキストのサイズ l_c は一定としている。このためあまりにも古い観測結果は破棄されることになる。

観測結果を導く方法について説明する。観測モジュールはロボットの行動を観測し、その行動変数を推測する。またコンテキストからも行動変数をひとつ取り出す。そして観測したロボットの行動変数とコンテキストから取り出した行動変数を比較する。一致したか否かの真理値が観測結果となる。

コンテキストは過去の観測結果を蓄えるものである。そこで、観測結果が真であるときに観測された行動変数をコンテキストに追加する。ただしコンテキストサイズは有限であるから、新しい行動変数をひとつ追加するたびに古い行動変数をひとつ削除することになる。これがコンテキストの更新 U_c の手順となる。

コンテキストに追加されるとき観測結果は真なので、追加される行動変数はコンテキストの中身のどれかと一致している。従ってこの更新プロセスを繰り返すことにより、コンテキストの保持する行動変数はどれも似たものとなっていく。この結果、観測結果は安定し、出力する行動変数も安定したものとなる。

以上の観測プロセスを疑似コードで書くと図 4.2 となる。観測プロセスはモデルにおける行動変数生成関数 P の一部と更新関数 U_c の両方を担当するものであり、分けることはできない。つまり、観測のたびにコンテキストは更新される。

4.4 不定な観測

郡司らのモデルでは、個々のオートマトンの真の状態は $\{0, 1, \alpha\}$ の三値であるのに対して観測された値は $\{0, 1\}$ の二値でしかない。真の状態に存在する α は観測すると 0 か

1へと変換され、観測結果は真の状態と完全には一致しない。この観測の不定性のため、状態遷移を観測すると直前の状態遷移関数とは異なる状態遷移が起こっていることが観測される。そして直前の状態遷移関数と観測された状態遷移の不一致が起こると、状態遷移関数を変化させる。これによって状態遷移関数の変化が実現される。

観測指向モデルにおいても、観測モジュールによる変換に不定性が含まれる。観測モジュールではセンサ入力を真理値にいったん変換する。しかし、観測ごとにコンテキストに変化が起きるため、ある時点での観測結果が過去の観測結果と一致するとは限らない。

しかし、これだけでは不十分である。先述したようにコンテキストの保持する行動変数は次第に同一のものに収束する傾向がある。しかし、コンテキストの保持する行動変数が同一の内容になってしまうと、それ以降はあるセンサデータに対して常に同じ観測結果が得られることになる。つまりある時点から観測結果は完全に固定されてしまうため、大域的な可変性は実現されない。何らかの手段により、コンテキストが変化しなくなるを防ぐ仕組みが必要である。

観測指向モデルでは非常に単純な仕組みでこの問題に対処する。それは通常の結果である真理値を、単純にある確率 p_0 で真を偽に、偽を真に逆転させるというものである。このため、コンテキストの保持する行動変数と一致しないデータであってもコンテキストに追加されることになる。この結果、通常はコンテキストは収束する傾向を持つために安定した行動変数を保持するが、観測結果が逆転するためコンテキストの中身は完全には同一なものとはならず、大域的な可変性は失なわれない。このように観測結果が常に正しいものとは限らず、時として誤りを与える観測のことを不定な観測という。

4.5 出力モジュールと秩序関数

秩序関数は、郡司らのモデルにおける状態遷移関数に対応しており、観測結果を受け取り、出力モジュールへの行動指針として真理値を渡す。そして出力モジュールは秩序関数から渡される指針にもとづいて行動変数を生成し、出力する。秩序関数と出力モジュールは行動変数生成関数 P に相当する。

秩序関数において入力である観測結果と出力である行動指針のあいだの関係はテーブルによって保持されている。秩序関数に観測結果が入力されると、テーブルの対応するエントリが取り出され、出力する値が決定される。郡司らのモデルでは状態遷移を調べることで矛盾を発見していたが、観測指向モデルでも同様に、出力する行動変数から状態遷移の矛盾を発見し、秩序関数を更新する。

秩序関数の返り値は出力モジュールで解釈される。返り値が真の場合は、安定して行動するという意味に解釈される。したがって、コンテキストからランダムに1つの行動変数を取り出して出力する。逆に偽の場合は現在の行動変数とは無関係な行動をすることを意味する。そこで、周囲のセンサ情報からランダムに行動変数を生成し出力する。

以上の出力プロセスを疑似コードで書くと図4.3となる。order_functionはテーブルであり、観測結果に対応した行動指針を出力する。2-6行目が出力モジュールであり、random_generateは周囲のセンサ情報から行動変数をランダムに生成することを意味する。

```

(1) policy = order_function[observation_results]
(2) if policy then
(3)     return context[random()]
(4) else
(5)     return random_generate()
(6) end

```

図 4.3: 出力プロセス

表 4.1: 秩序関数の特性

入力	出力が <i>False</i>	出力が <i>True</i>
<i>FFF</i>	拡散	新しい基準の選択
<i>FFT</i>	新しい基準の探索	グループに加わる
<i>FTF</i>	他の基準の探索	現在の基準の保持
<i>FTT</i>	グループからの脱出	グループの維持
<i>TFF</i>	新しい基準の探索	グループに加わる
<i>TFT</i>	新しい基準の探索	グループに加わる
<i>TTF</i>	グループからの脱出	グループの維持
<i>TTT</i>	基準の遷移	基準の維持

以上から、秩序関数のふるまいの意味を考えることができる。各入力に対する出力の示す意味について、表 4.1 に示す。この表の入力部分で、*T* が真を、*F* が偽を表しており、中央が自分自身の観測結果、左右はそれぞれ他を観測して得られた結果を意味する。ここで「グループ」は、複数のロボットで一つの行動変数を共有している状態を指す。たとえば入力が *FFT* の場合で出力が偽の時、自分のほかに現在の行動変数に一致する行動をするロボットが存在しているが、それと無関係な行動変数を用いようとしている。すなわち新しい変数の探索をしていると考えられる。一方で入力は同じだが出力が真の時、コンテキストから行動変数を出力しようとしているが、そのコンテキストは自分の現在の行動とは一致せず、ほかのロボットの行動変数と一致している。つまり、現在の行動を捨ててほかのロボットと同じ行動しようということを意味する。したがって、そのロボットと同じグループに加わると解釈できるだろう。

4.6 秩序関数の更新

秩序関数は最終的に出力される行動変数を管理している。したがって、行動変数の局所的な安定性と大域的な可変性は、ともに秩序関数の更新メカニズムに強く依存している。

郡司らのモデルでは、観測の不定性にもとづいて状態遷移関数を変化させていた。つまり、状態遷移を観測することによって真の状態に存在する α が 0 か 1 に変化するため、

観測された状態遷移はそれまでの状態遷移と必ずしも一致するとは限らない。そこで、観測された状態遷移に一致するように状態遷移関数を推定し、これを新しい状態遷移関数とする。観測指向モデルでも、出力した行動変数をふたたび観測することにより、秩序関数の更新を行う。

具体的には以下の通りである。まず、出力部が出力しようとしている行動変数を、観測モジュールと同じ手法によって観測する。つまり、コンテキストのなかから行動変数を取り出し、出力しようとしている行動変数と比較して一致しているかどうかを調べる。一致しているならばコンテキストを更新する。このプロセスを自己観測と呼ぶ。

自己観測の結果は、その時の秩序関数の返り値と比較される。通常の場合にはこの結果は一致するが、異なることがある。つまり「本来は真だと思って行動変数を生成していたが実は偽だった」あるいは「偽のつもりで行動変数を生成していたが実際には真だった」ということが起こりうる。秩序関数が真を返しているとき、コンテキストの中身がすべて同じもののみで構成されていれば矛盾は発生しない。しかし不定な観測によって異なる行動変数がコンテキストに混入している場合、出力するときの行動変数と自己観測のときの行動変数では異なるものを選ばれることがある。その結果、自己観測の結果が秩序関数の返り値と一致しなくなる。また秩序関数の返り値が偽のとき、出力部は行動変数をランダムに生成する。ここでランダムに生成した行動変数はたまたまコンテキストと一致すると自己観測の結果は真になる。このような要因により、不一致が発生したときに、秩序関数のテーブルでその時に使われたエントリの出力値を反転させる。

ただし、これだけでは矛盾が発生せず、ロボットの行動変数がまったく変化しなくなることがある。たとえば周囲のロボットグループがすべて同じ行動をしていて、秩序関数も真を返すという状況を想定する。表4.1における $TTT \rightarrow T$ という状態である。この状況が維持されるとコンテキストもその行動変数で完全に統一されてしまう。すると、この状況から変化する要因が完全になくなってしまう。そこで、自己観測の際にも一定の確率 p_a による不定な観測を導入することで、そのような状況に対処する。

自己観測は図4.3において返される値を、図4.2における観測プロセスと同じアルゴリズムで計算することで実現される。自己観測の結果は図4.3における `policy` と一致するかどうか検査される。

4.7 行動基準の変化

本節で、観測指向モデルにおいて行動変数が変化するプロセスについて説明する。

観測指向モデルでは、コンテキストが行動変数を保持している。そして、観測モジュールで観測が行なわれることによりコンテキストは更新されるが、コンテキストからランダムに選択された行動変数に従って更新するデータは変化するため、コンテキストは類似する行動変数を集めるようになる。そして秩序関数が真を返すと、コンテキストから行動変数が選ばれて出力される。それにもとづいた行動はまた同じように観測される。これが繰り返されることでコンテキストはどんどん類似する行動変数を溜めてゆき、類似した行動変数がそのまま出力される。これによって安定した行動変数が維持されるようになる。

しかしながら、不定な観測によって、異種のデータもまれにコンテキスト内に入り込む

ようになる。といっても、コンテキストの残る大部分は類似するデータのままなのだから、その影響は大したことはなく、たいていはすぐに消えてしまう。ただし、秩序関数が組み合わさることにより、不定な観測から変化が生じることがある。自己観測において使われるコンテキスト内のデータとして不定な観測によって混入した異種データが偶然選ばれる場合、秩序関数の結果と実際に出力する行動変数のあいだに矛盾が生じるからである。

矛盾が起こると、秩序関数のふるまいが変わる。つまり、それまで安定的に真を返していたものが偽を返すようになる。すると、コンテキストとは異なる行動変数ばかりが出力されるようになる。この結果、ロボットの行動が無秩序化する。

観測される行動が様々に変化するため、不定な観測による影響からコンテキストには様々なデータが入ってくる。しばらくこれが継続されると、ランダムに生成される行動変数が、たまたま自己観測で一致してしまうことがある。すると、ロボットの行動はまた真に戻る。真に戻ると、コンテキストが収束する傾向の影響が強くなって、また安定した行動変数を出力するようになる。ただし、そのときの行動変数は、収束する前にコンテキストに含まれる行動変数のうちどれか、ということになる。このため、元の行動変数とは異なる行動変数で安定することがある。

このように通常は安定的に行動変数を出力しているが、不定な観測によって異種のデータが混入する。さらにこれが秩序関数に影響を与え、出力が無秩序化する。そしてしばらくするとふたたび安定的に戻るが、戻ったときには元の行動変数とは違うもので安定化することがある。この安定と無秩序の2つの状態を交互に行き来することで、局所的には安定しているが、大域的には変化しうる行動変数を実現されている。

4.8 実験環境

観測指向モデルの出力する行動変数の特性はシミュレーション環境で調査した。本節ではそのシミュレーション環境を説明する。

観測指向モデルでは、周囲のロボットや人間の行動を観測する。相互に観測しあうというインタラクションのなかから複雑な行動が創発されるというモデルである。そこで、シミュレーションでは環境中に同等のロボットを複数配置し、互いに観測しあうという状況を想定した。

シミュレーション環境を図4.4に示す。

この図のようにシミュレーション環境は全体で円形をしており、その内部に仮想的にロボットが一定のパターンで配置されている。図中の小円がロボットを示している。

ロボットには行動変数として「色」を与える。また、図から示されるようにシミュレーション環境を区切る壁は何色かの色で塗り分けられている。また、ロボットの行動は単純に、方向を変更するだけとする。図において小円から出ている線は、そのロボットの向いている方向を意味する。そして、出力する行動変数の色と一致する色のある方向へと向きを変える。ただしロボットは全方位カメラを備えており、全周囲の色情報を取得できるとする。またロボットは自分の向いている方向を連続的にしか変化させることができない。つまり、ぜんぜん違う方向に向きを変えたいとしても、急激に向きを変化させることはできず、現在の方向から少しずつ向きを変えていかなければならない。このため、あるロ

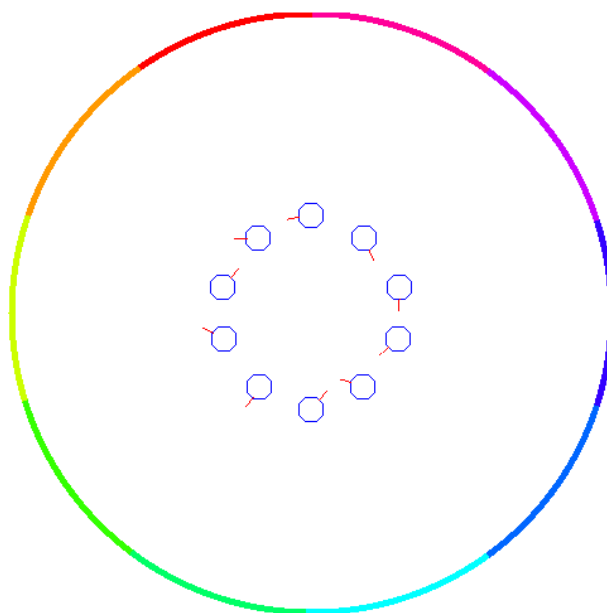


図 4.4: シミュレーション環境

ロボットがある時点である方向を向いていたとしても、それが移動途中であるのか、それとも目的地であるのかということにはわからない。

次にロボットの観測について述べる。ロボットは互いに行動を観測しあう関係にあるが、単純に自分と最も近い2つのロボットの行動を観測して行動変数を得る。つまり、図4.4の配置であれば、隣のロボットをそれぞれ観測するということになる。また、観測指向モデルでは最終的に人間などの行動を観測することを目的としているので、互いの行動を観測するとき、相手の行動変数のような内部状態のやりとりはせず、外部から得られる情報だけをもとにして観測を行う。そこで観測する相手が現在向いている方向の先にある壁の色を、その時点での相手の行動変数と推定し、これを観測モジュールに与える。先述したように、この行動変数は観測する相手の実際の行動変数とは必ずしも一致しないことがあるが、この観測の不確かさによって、より複雑な行動パターンが実現できると考えられる。

この環境においてロボットは自分のコンテキストを更新する。コンテキストが安定していると、そのコンテキストの色に対応する方向にロボットは自分の向きを変え、その方向に留まって安定する。しかしコンテキストを更新していくと、不定な観測と秩序関数の変化し、いったん無秩序な状態になったあとで大域的な変化が引き起こされる。したがってロボットは特定の方向をしばらく維持したあと、いったん出鱈目に方向を変化させつつ、しばらくするとほかの方向に向きを変えてふたたび安定する、というふるまいをすることが期待される。また、近い位置に存在するロボットの行動を観測するため、位置的に近いロボット同士が相互に影響を与えあい、同じ方向を向くことが多くなることが期待される。また、秩序関数が偽を返すときにはランダムに行動変数を生成するとしたが、この場合はランダムに一つの方向を選択し、ロボットからみてその方向に存在する色を行動変数として用いることとした。

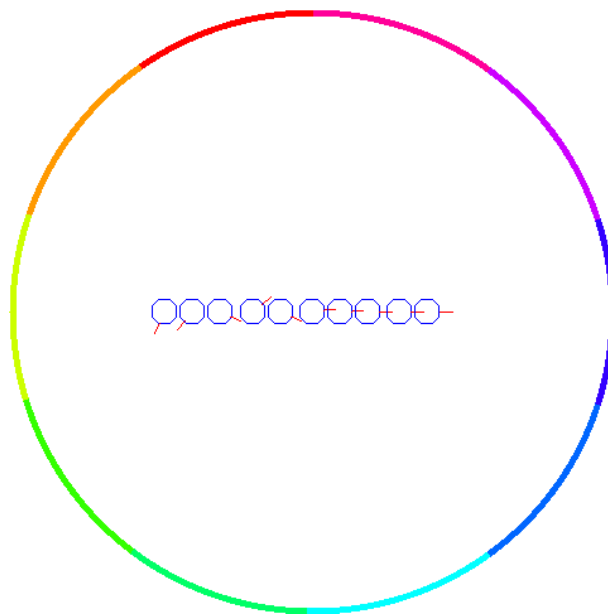


図 4.5: 直線配置パターン

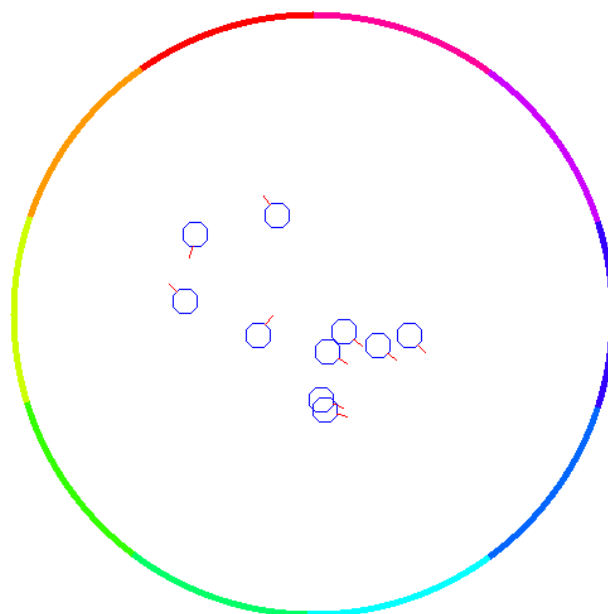


図 4.6: ランダム配置パターン

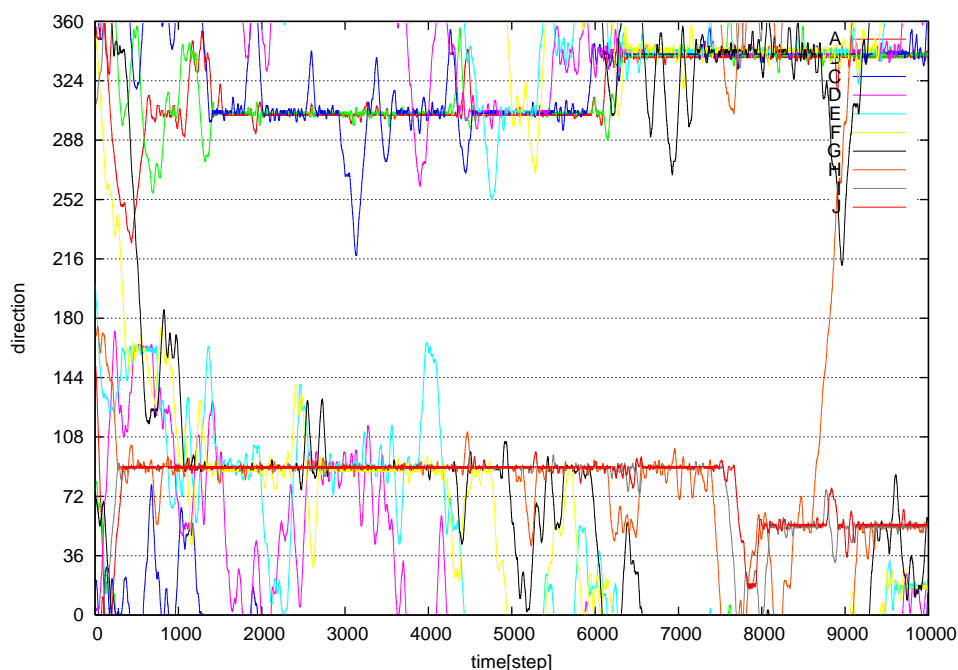


図 4.7: ロボットの方向の時間遷移

本シミュレーションでは、壁面は均等な幅の 10 色に塗り分けられているとした。また、ロボットは 10 体で一定のパターンで配置した。配置パターンは図 4.4 で示した円形配置の图案の他、直線配置(図 4.5)とランダム配置(図 4.6)を用意した。ただし、ランダム配置は毎回各ロボットの位置として中心からの距離と角度をランダムに与えて配置する方式であり、図はその一例である。

観測指向モデルでは、コンテキストのサイズと、意図的に導入された誤り率 p_o および p_a によって挙動は変化する。また、コンテキストのサイズは 8 個、 $p_o = 20\%$ 、 $p_a = 2\%$ とし、1 回の試行では 10,000 ステップ実行した。

4.9 実験結果と考察

4.9.1 典型的な実行結果

まず、動作させたときにロボットが実際にどのように自分の方向を変化させたかを調べる。図 4.7 は円形の配置のときの典型的な動作例である。この図において横軸は時間(ステップ)、縦軸は、その時点でそのロボットが向いている壁の方向を示している。また、点線は壁面の色の境界を示す。

たとえばロボット C の遷移を意味する青色の線に注目して図 4.7 を説明する。ロボット C は最初は 0-72 の方向で不安定に向きを変えていたが、1,200 ステップほどで 288-324 の方向に向きを定めた。2,000 ステップほどでいったん 0-36 の方向に向きを変えるが、また元の方向に向きを戻した。そのようにときどき異なる方向に向きを変えては戻るとい

動作を繰り返しつつ288-324の方向を維持していたが、6,000ステップでほかのロボットと同様に向きを変えた。そして試行の終了まで324-360の方向を維持した。

ロボット群全体としては、大きくわけて2つのグループに分けることができる。すなわち、72-108の方向を向いているグループと、288-324の方向を向いているグループである。また、ロボットDのようにどちらのグループにも属さず、この2つのグループの間を行き来しているロボットも存在している。ロボットは自分と位置の近いロボットの行動を観測するため、近隣のロボットたちによってグループが形成される。円形配置の場合、近いロボットとは隣のロボットのことだから、グループは円を構成する一部の円弧となる。グループを構成する円弧が2つあれば、その境界に位置するロボットは双方の影響を受け、2つのグループのあいだを行き来するようになると考えられる。

この結果からは、時間の経過によって特定の色の行動変数を共有するグループが消えることもわかる。実際、5,500ステップあたりから228-324のグループの一部のロボットがグループから離脱して他の方向に切り替える行動を示し、それに応じて6,300ステップほどでこの行動変数を保持するロボットはいなくなる。かわりに、先に離脱したグループとともに324-360の方向で安定する。一方、72-108の方向のグループも7,800ステップほどで消滅し、そのグループに属していたロボットは別の色に行動変数を変化させた。また、グループを構成するロボットの入れ替わりも発生している。

このように、ロボットは内部のコンテキストが安定しているために一定の方向を維持しているが、しばらくするとコンテキストが変化して別の方向に遷移することが確認された。また、ロボットは互いに影響を与えあうことで、同時に同じ方向を向くグループを形成していることが確認された。

ロボットの配置による影響を見るため、次に直線配置の典型的な例を図4.8に示す。軸などの意味は図4.7と同じである。

直線配置でも円形配置とほぼ似たような動作パターンとなっている。したがって観測指向モデルが生成する行動変数は、初期配置のような環境要因とは関係なく生成されることがわかる。

ただし、詳しく調べてみたところ配置による差は見い出された。直線配置の場合、回転対象である円形配置と異なりロボットが互いに観測しあう関係に偏りがあるからである。円形配置では観測するロボットは単に両隣のロボットとなっていた。直線の中央付近に位置するロボットも同様に両隣の行動を観測するが、右端のロボットには右隣のロボットが存在せず左隣の2体のロボットの行動を観測する。この偏りのため、直線配置では両端がそれぞれ独立したグループを構成しやすい。そこで、線の両端がそれぞれ異なるグループとなり、線の中央付近が両方のグループの影響を受けて変化するという状況が発生しやすくなった。この例では3,000ステップあたりで別個なグループがあらわれた。さらに8,000ステップあたりで3つめのグループが登場している。これは線の中央に位置するロボットたちが両方の影響を受けながらその中間的な位置を保った結果である。

さらに、ランダム配置の典型的な例を図4.9に示す。軸などの意味は図4.7と同じである。

ランダム配置では、直線配置や円形配置とはかなり異なる行動パターンが示されている。まず、特定の方向を向いたまま、周囲の影響をまったく受けず動かないロボットが存在していることがわかる。ほかの影響を受けるロボットもいることはいるが、この例では影響を受けないパターンが多い。ランダム配置の場合は配置パターンが毎回変わるが、

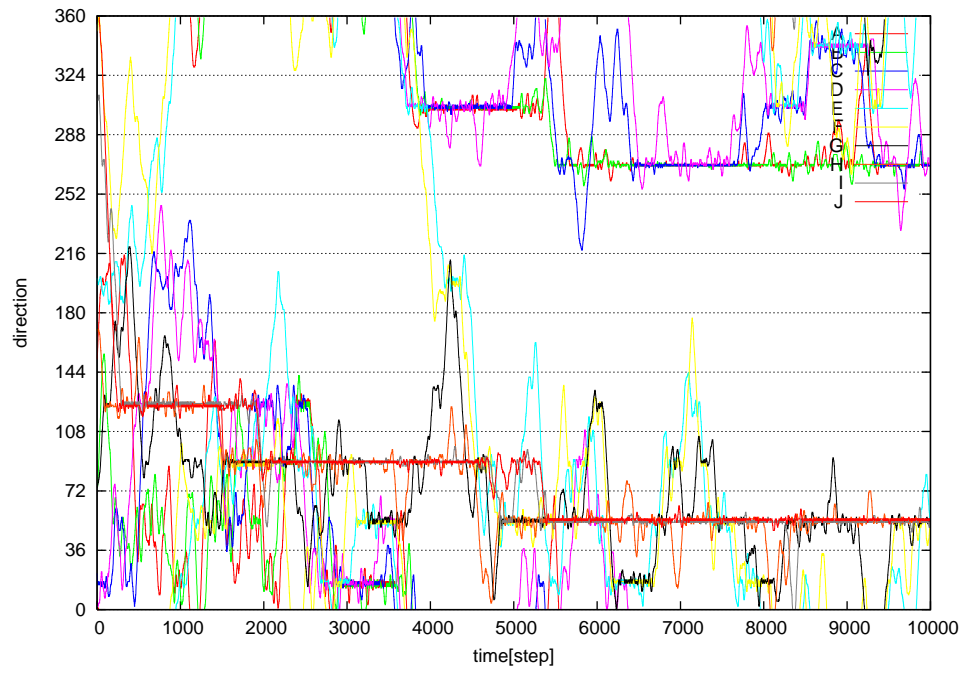


図 4.8: 直線配置でのロボットの方向の時間遷移

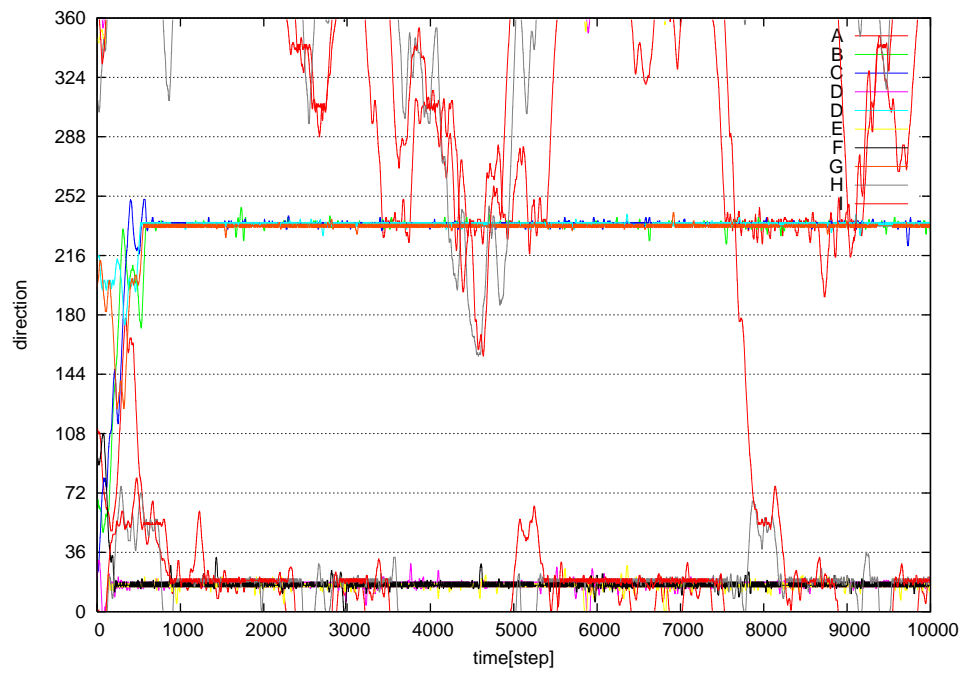


図 4.9: ランダム配置でのロボットの方向の時間遷移

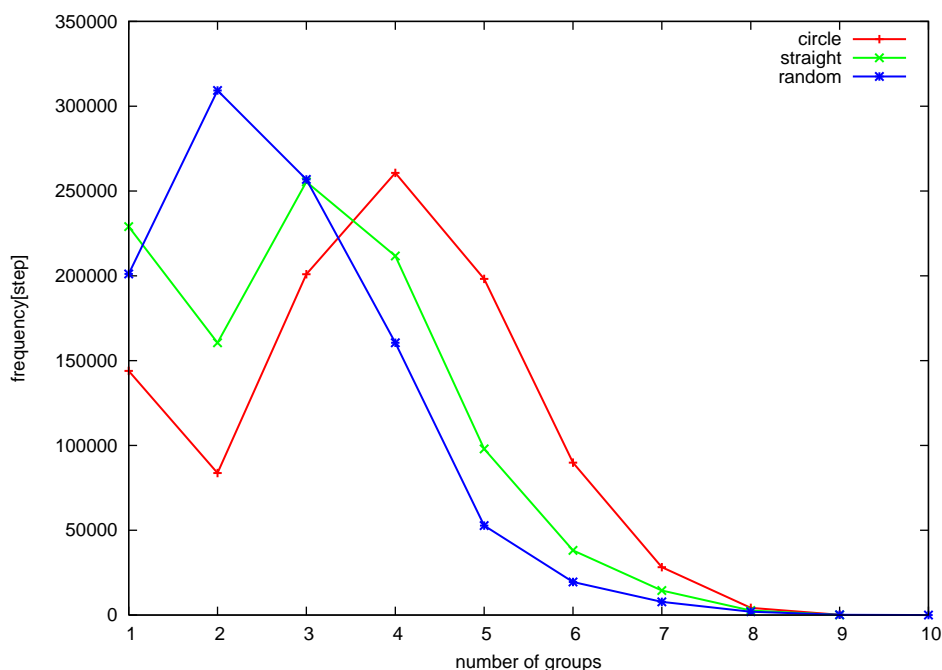


図 4.10: グループ数の頻度分布

そのときの配置パターンによっては3体のみが非常に近い位置に集まることがある。するとその3体は互いに互いだけを観測することになり、外部の影響をまったく受けなくなる。このような配置をされたロボットたちを孤立系と呼ぶことにする。孤立系では属するロボットの数が少なく、観測による影響力が強いため、新しい行動変数を獲得しづらくなる。そのため、いったん行動変数を獲得すると、獲得された行動変数は試行が終了するまで維持されることになる。図 4.9 では、216–252 の方向を向く孤立系が存在している。

このように、配置によってロボットが相互に影響を与えあう関係には何らかの差があることが示唆された。そこで、この差を比較するために、ロボット群が相互にどれだけ影響しあっているかを調べた。相互に影響を与えあう関係は、ロボットが実際に同じ色のある方向を向いているかどうかで判断できる。そこで、ロボットが同じ色のある方向を向いているとき同一のグループであるとみなし、ステップごとにロボット群がいくつのグループを形成したかを調べた。100 回の試行を行い、グループ数の頻度を調べた。ここで、ロボットが互いにまったく影響を与えあわないのであればグループ数は多くなりがちである。ただし偶然に一致することもあるから中間的な数のグループの頻度が最も高くなり、正規分布に従って頻度は減少すると考えられる。一方、観測指向モデルは相互に影響を与えあうから、それよりは少ないグループ数が最頻値となると考えられる。また、円形配置よりは直線配置の方がグループが分かれやすいという傾向が見られたから、直線配置の方がより多いグループ数に最頻値があらわれ、ランダム配置は周囲の影響を受けないグループが存在する可能性があるため、より多いグループ数に最頻値が表れることが予想される。

実際に頻度を調べ、プロットしたものを図 4.10 に示す。

結果は予想と逆であり、ランダム配置がもっとも少ないグループ数となり、直線配置がそれに続き、円形配置のグループ数は多い。ランダム配置では孤立する群と、その孤立群

を観測するために影響を受けるロボットによって、ロボット全体が分かれがちであるため、2グループが非常に多く出現するものと思われる。直線配置では両端のグループと、それに影響を受ける中間のロボットたちで3グループになることが多くなる。円配置の場合、対称形であるためにグループが強く形成される要因が弱く、結果としてグループ数が多くなるものと考えられる。ただし円形配置の場合でも、10ロボットで4グループが最頻値となるため、類似する傾向が強くなっていることが見てとれる。また、どの配置でもおおむね正規分布に従っているが、直線配置と円形配置の場合はグループ数が1の場合だけ多く、ランダム配置の場合はグループ数1の場合が少なく、どちらも正規分布から外れている点は興味深い。グループ数が1の場合は、偶然一致する確率を除くとすべてのロボットが相互に影響を与えあった結果、同一の行動変数を取るためであると考えられる。グループ数が1の場合の頻度が高くなっているということは、配置によって決まるグループ分けの影響を上回って相互の影響が全体に伝播しているということを示している。またランダム配置で減るのは、孤立系のロボットが存在する可能性があるため、全ロボットに影響を及ぼす状況が生まれにくいことを示している。

以上のように、互いに行動を観測しあうインタラクションのなかから複雑な行動パターンが実現できたことがわかった。しかし本研究での目的は、単に複雑な行動パターンだけではなく、行動パターンが短期的に安定し、また長期的には変化して、予測もつかない変化をするという点にある。そこで、実際にロボットが向いていた方向から行動変数の特性を調べる。

ロボットが行動変数を獲得すると、その色に対応した特定の方向を向くが、しばらくするとほかのロボットの影響で向きを変えることがある。このように、実際にロボットが特定の方向に連続して向いていたステップ数によって、ロボットの行動パターンを知ることができる。そこで100回の試行を行い、ロボットの向いていた方向を色の境界でわけ、特定の色の方向が維持されるステップ数を調べた。さらにそれをステップ数ごとに累積し、100回の試行のなかでどれくらいの頻度で出現するかを調べた。その頻度分布が図4.11である。ただし横軸は行動変数が維持されるステップ数、縦軸はその出現頻度(回数)であり、両対数プロットした。なお、各プロットはそれぞれのロボットの配置条件を示す。

ステップ数が30以下の場合、大きく上下に変動している。この、短い期間の場合には観測指向モデルによる特性よりはむしろ、色領域の大きさやロボットの回転速度といったシミュレーション環境に由来する要因の方が強いと考えられる。この部分はひとまず無視する。30-40ステップあたりで頻度はピークを迎え、あとはおおむね直線的に減少する。つまり、極めて長いステップ数の行動変数が存在している一方で、極めて短いステップ数の行動変数もそれなりの頻度で出現している。

また、配置条件による差はほとんどない。ただしランダム配置の場合は若干、長いものの頻度が高いようである。これは、孤立系を形成するロボットが存在しうるためであると考えられる。先述したように孤立系では行動変数は変化しづらくなるため、より長いものが多く出現すると考えられる。

さて、頻度分布からは極めて長い期間を行動変数が維持されていることが確認できているが、一方で短いステップ数の行動変数の頻度も高い。局所的な安定性が実現できているかどうかを実際に確認する必要がある。そこで期間の長さとは頻度とを掛け合わせ、その期間が実際の試行で存在したステップ数を計算した。そして、このステップ数を累積してプ

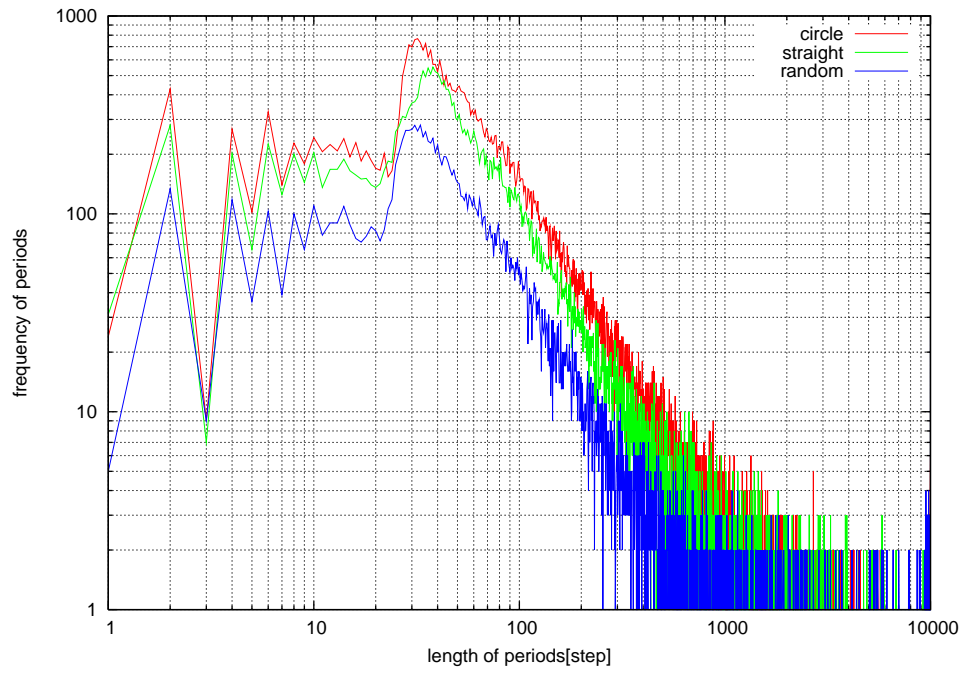


図 4.11: 特定の方向を向いていた期間の頻度分布

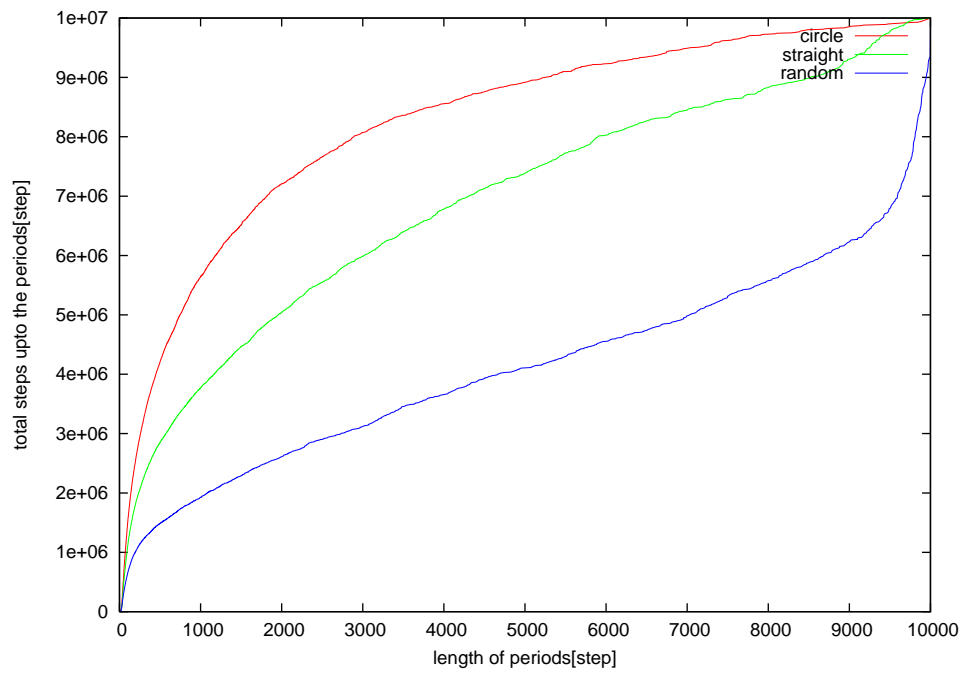


図 4.12: 維持された行動変数の累積

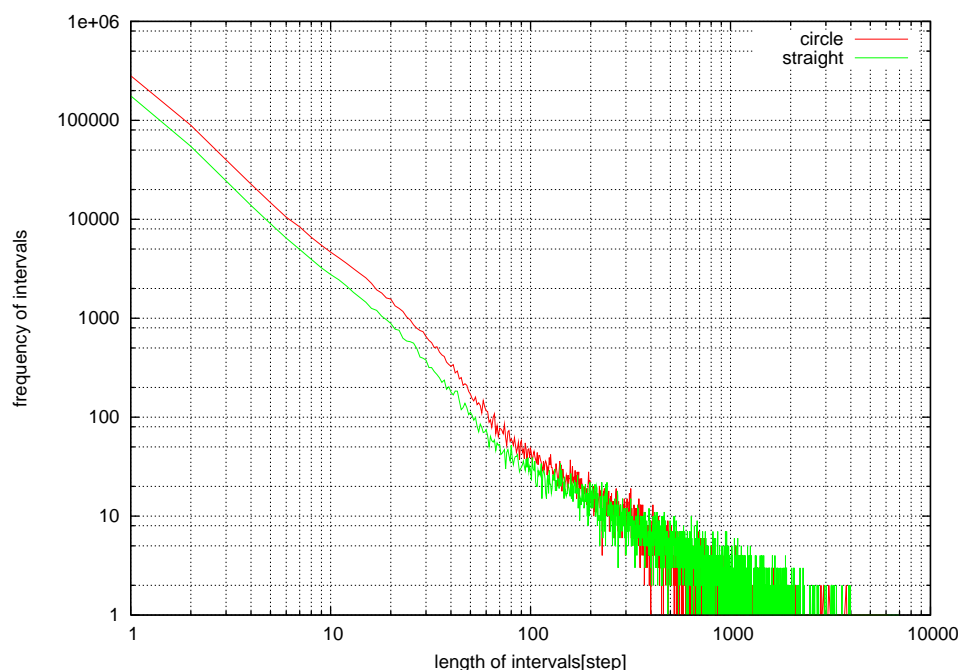


図 4.13: 行動変数が維持される時間の頻度分布

ロットした(図 4.12).

どのプロットでも期間が短い部分で急上昇し、それから増加量がだんだん減少していく傾向を示している。図 4.11 から 30–40 ステップが非常に高い頻度で出現していることがわかったが、短い期間での急上昇は高い頻度で出現していることを示している。ただし短い期間が全体に占める割合は高いものではない。全体の半分との交点を見ると、円形配置で 700 ステップほど、直線配置で 2,000 ステップほど、ランダム配置では 8,000 ステップほどとなっている。つまり、少なくとも全体の半分以上の時間はこのステップ数よりも長い期間を維持し安定しているということである。30–40 ステップの短い期間は頻出しているものの、全体を占める割合としてはそれほど多くないことを示している。また、円形配置と直線配置と異なり、ランダム配置では 8,000–10,000 ステップで急激に上昇しており、このような長い期間を維持することが多いことが示唆されている。ランダム配置の場合には孤立系が発生することがあり、孤立系に属するロボットが試行のほぼすべてを同一の行動変数で終えてしまうことがその理由であると考えられる。

さて、ここまで検討してきたのは実際にロボットが取った行動の履歴から得られた結果である。このため、どうしてもシミュレーション環境に依存する要素の影響を考慮に入れなければならない。そこで、秩序関数が真を返したときにロボットがコンテキストから出力した行動変数を調べることで、観測指向モデルではコンテキストがどのように変化するかがわかると考えられる。もちろん相互に観測しあう関係やランダム配置における孤立系の問題など、環境による要因を完全に除外することはできないが、ある程度は環境と独立した特性を見ることができよう。

そこで、同様に 100 回の試行を行い、同一の行動変数がどれくらい連続して出力されるかを調べ、そのステップ数ごとに頻度を計算する。この頻度分布を図 4.13 に示す。横軸

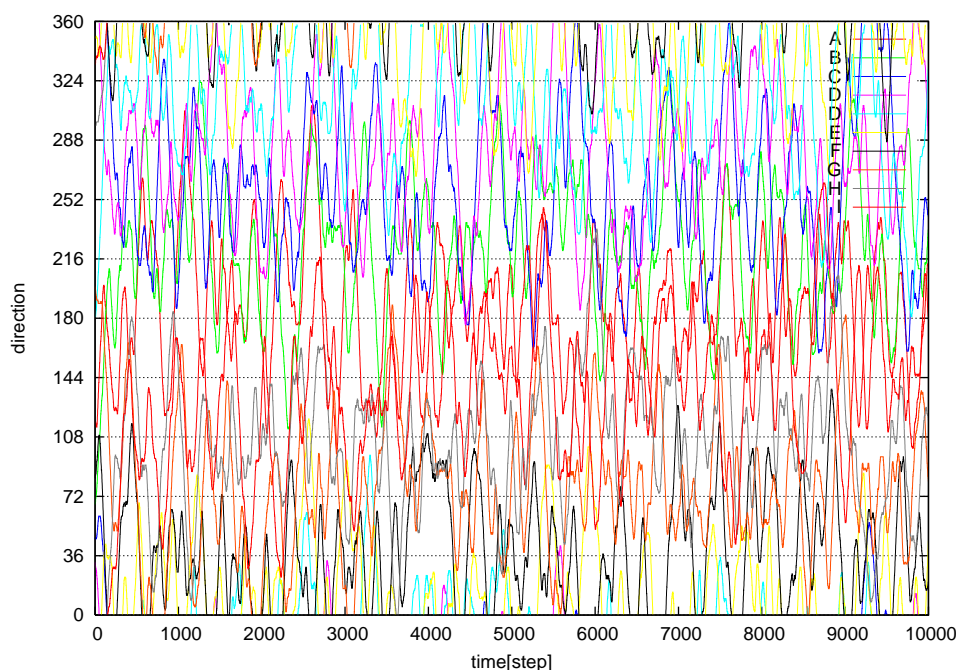


図 4.14: ランダム選択モデルの実行例

と縦軸は先程と同じであり、やはり両対数プロットした。

このグラフでも減少の仕方は図4.11と同じようにほぼ直線的になっている。つまり、ロボットが実際に向いた方向はロボットのコンテキストの中身に影響を受けたことによることがわかる。また、図4.11では30ステップ以下の頻度は安定せず上下していたが、このグラフではそれが見られず、行動変数が維持される時間は長さが1ステップのものが最も高い頻度で出現し、ほぼ単調に減少している。したがって、この上下の変動は環境による要因であることが示唆される。

4.9.2 ランダム選択モデルとの比較

次に、観測指向モデルとは異なる行動選択モデルを実装し、観測指向モデルと比較することで観測指向モデルの特性を検討する。このモデルでは、ステップごとに行動変数となる色を単純にランダムに選択する。このランダム選択モデルでロボットが実際に向いた方向は、次の図4.14のように、まったく無秩序なものである。

さて、このようなランダム選択モデルと観測指向モデルの違いを、ロボットが実際に特定の方向を維持する期間の頻度分布から比較した。図4.12と同様に、100回の試行を行い、頻度を計算した。ロボットを円形に配置した場合の観測指向モデルの結果とともにプロットしたものを図4.15に示す。

やはり期間が30ステップ以下の場合と同じように上下しており、この部分は環境に依存するものと考えられる。また、ランダムモデルでも30-100ステップの頻度が極めて多いという分布は似ているが、より急激に減少する。図からはランダムモデルでも100ステップを越える期間で維持される行動も存在していることがわかる。これはランダムに選

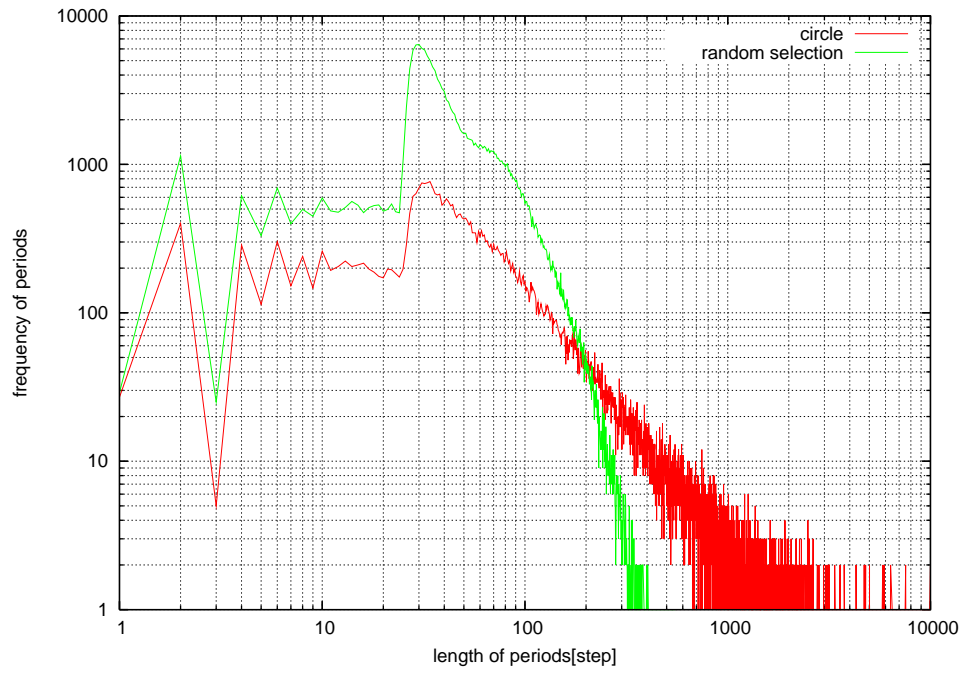


図 4.15: ランダムとの比較

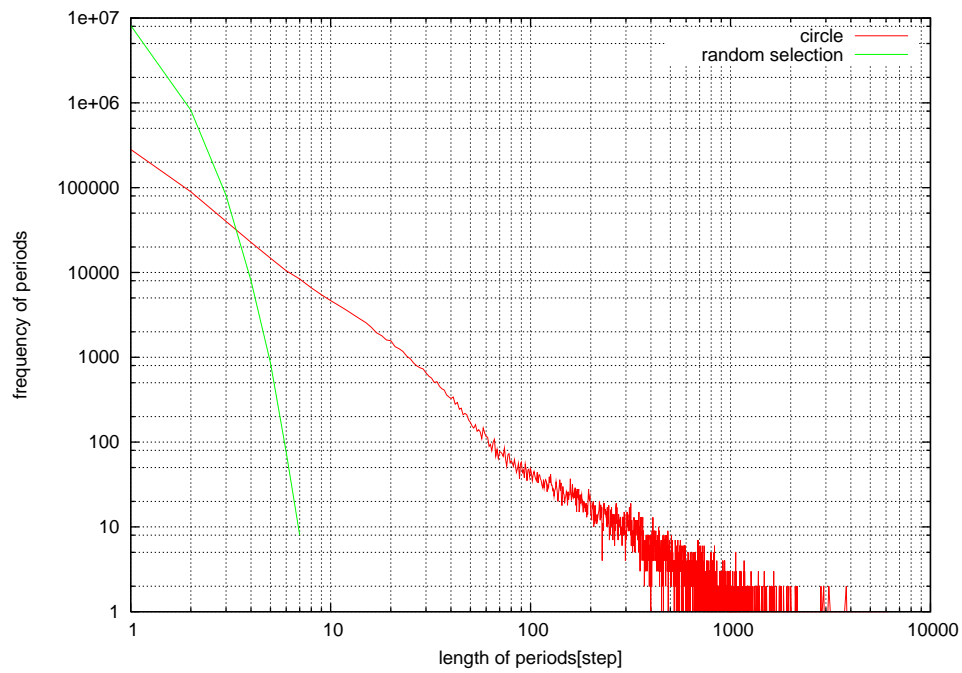


図 4.16: システムの出力におけるランダムとの比較

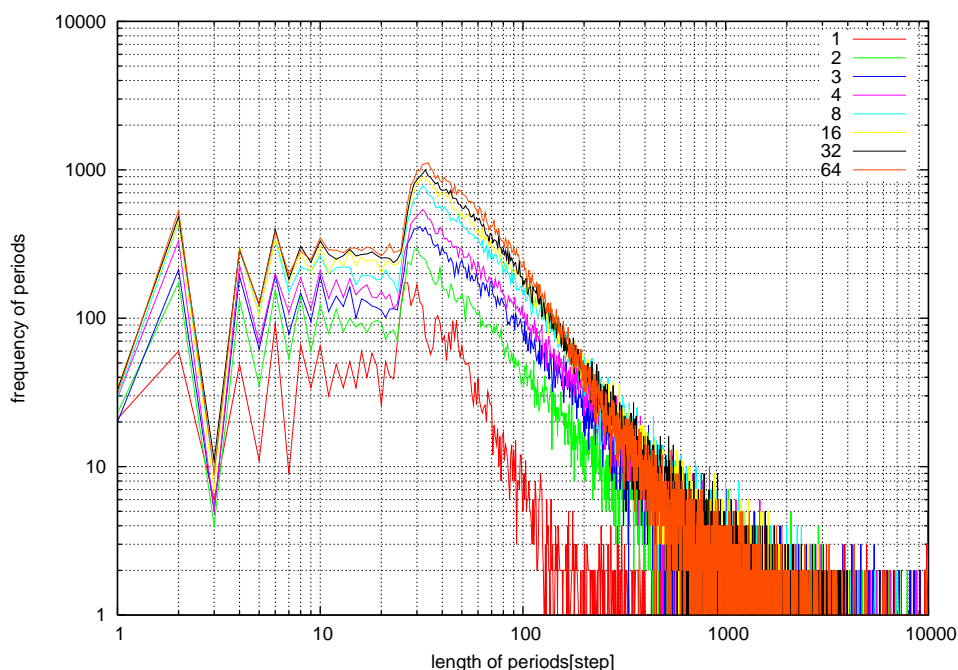


図 4.17: コンテキストによる頻度の差

択することで維持される期間と考えると長いように思われるが、これは特定の色の壁がある程度の幅を持っているため、通過するのに時間がかかったり、途中で方向転換をする可能性があるためであると考えられる。いずれにせよ、それより長いものについてはまったく存在しない。

また、システムが出力する行動変数を図4.16に示す。毎回ランダムに選択するとき、行動基準が n ステップ維持されたとすると、それが発生する確率は $(\frac{1}{c})^n$ となる (ただし c は環境中の色の数)。単純ランダムモデルにおける頻度分布は完全にこの式に従っており、10を超える長さの行動変数はまったく存在しない。まったく異なる分布であることがわかる。

4.9.3 パラメータの検討

観測指向モデルには、挙動に影響を及ぼすパラメータがいくつか存在する。特にコンテキストの大きさと、不定な観測による誤りの確率は挙動への影響が非常に強いと考えられる。そこでこれらのパラメータによる挙動の変化を調べた。次にその結果を示す。なお、特にことわらない限りパラメータとしてはコンテキストサイズは8、観測時の誤り率20%、自己観測時の誤り率2%とし、またロボットの配置は円形配置とした。また、統計データは100回の試行から得た。

まず、コンテキストサイズを1-64の範囲で変化させた。ロボットが実際に向いた方向からの行動変数を維持する期間の頻度分布をプロットさせた(図4.17)。

図からわかるように、1-3では急激に変化するがそれ以上では分布に大きな差は見られなかった。シミュレーション環境ではロボットは3体のロボットの行動を観測するので、

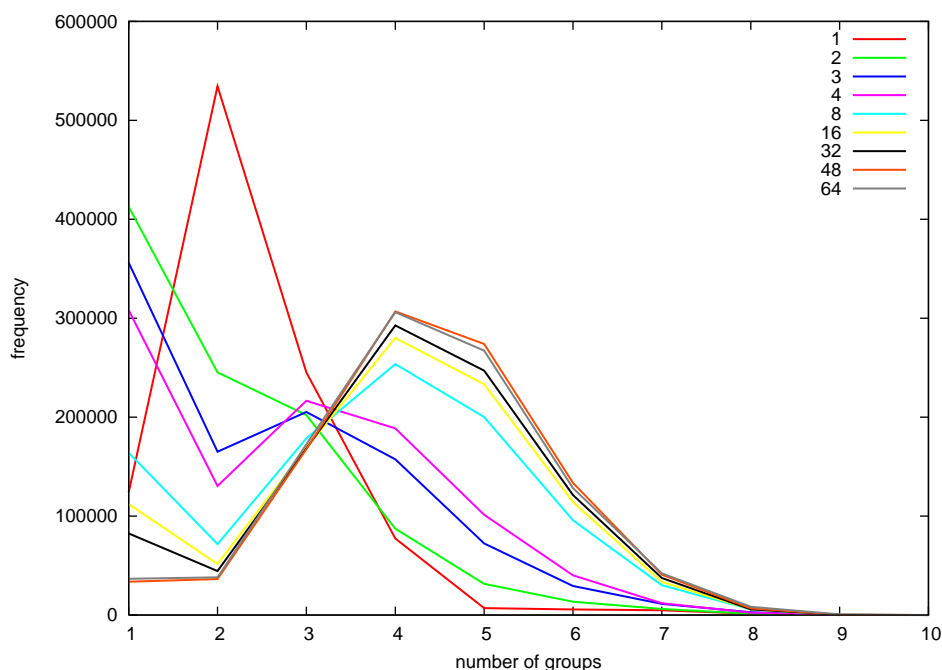


図 4.18: コンテキストの差によるグループ数の比較

たとえばコンテキストサイズが1のときは一回の観測のあいだに観測結果が上書きされる可能性がある。このため、他のロボットの行動を観測した影響が非常に強くなる。したがって、全体でひとつの行動変数に統一されやすくなり、互いに強く依存しあう状況となる。したがって群全体としてはむしろ安定し、短い分布はそれほど出現しなくなった。コンテキストサイズが大きくなると他のロボットへの依存度は弱くなり、短い期間の行動変数の頻度も増加する。

コンテキストサイズが大きすぎる場合、コンテキスト全体をひとつのデータに統一するのは困難である。逆にいったん統一されると、この傾向は安定である。したがって特定の行動変数が維持されるのではないかと推測されるが、実際にはそうになっていない。その理由としては、コンテキスト中に複数の行動変数を保持したまま安定しているためと考えられる。コンテキスト全体で同一の行動変数を保持することはできなくなるが、優勢なコンテキストが存在することで安定な行動が生成される。どの行動変数が優勢かは簡単には変化しないため比較的安定した行動変数が維持される。一方、コンテキストは同一の行動変数のみを保持するわけではないため、秩序関数が真のまま異なる行動変数が出力されることがある。このため、短い期間の頻度も大きい。

コンテキストサイズが大きすぎる場合、他のロボットが異なる行動をしていたとき、その行動変数が自分のコンテキストに入っても影響力が弱くなると考えられる。つまり、他のロボットが異なる行動をしていても許容しやすくなるという傾向があると考えられそうである。そこで、図 4.10 のようにロボット群がいくつのグループに分かれるかをステップ毎に調査した(図 4.18)。

コンテキストサイズが8以上になると、最頻値もグラフの概形もそれほど変わらない。しかしながら、グループ数が1のもの、つまり全てのロボットが同じ方向を向いていると

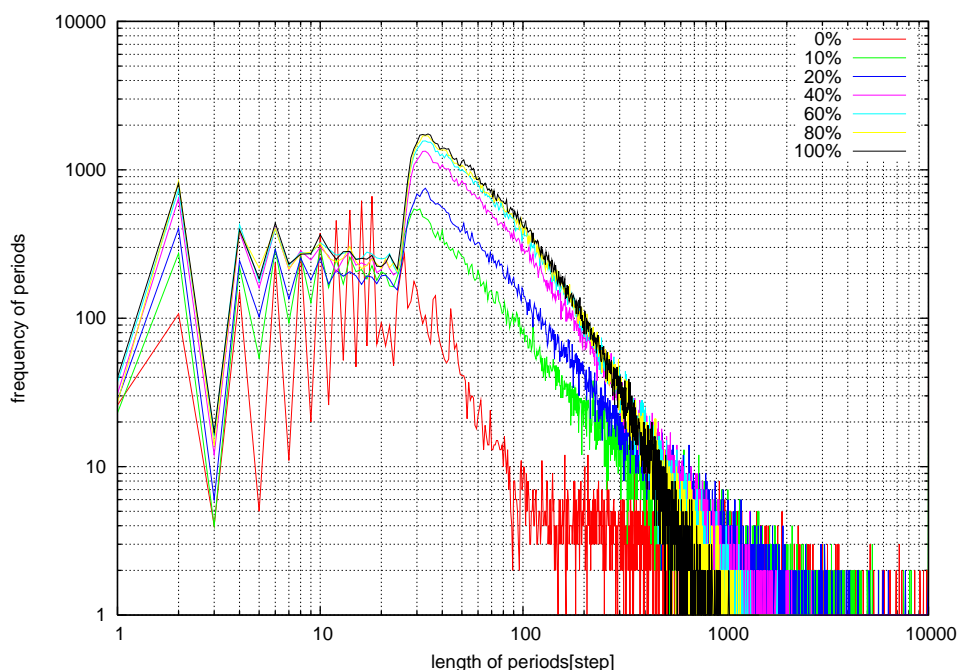


図 4.19: 誤り率と維持期間の頻度の関係

いう場合は減少する傾向にあることがわかる。つまり他のロボットが異なる行動をしていることを許容しやすくなる。

コンテキストサイズが8より小さい場合、最頻値となるグループ数はコンテキストが小さいほど小さくなる。これは他のロボットの行動に影響を受けやすいためであると考えられる。また、コンテキストサイズが1の場合の結果は極めて特異である。コンテキストサイズが1の場合は、サイズ2以上の場合と比較するとグループ数1の頻度は減少し、グループ数が2となることが多い。グループ数が1の場合が少ないのは、行動変数が順番にコンテキストに上書きするときの順序の影響であろうと考えられる。

次に観測時の誤り率を変化させたとき、同じようにロボットの行動から特定の方向が維持される期間の頻度分布を調べた(図4.19)。

頻度分布は0%の場合、0%より多く50%より小さい場合、50%以上の場合、の3種類のパターンに分かれた。次にそれぞれについて詳しく論じる。

0%のときは、これまで見てきたのとは異なる分布となっている。誤り率が0%のときは不定な観測を用いず、観測に誤りが生じないという状況である。このとき、異種の行動は観測時にまったく混入しないから、ロボットの行動はいったん安定するとそのままになってしまう。つまり、初期状態から安定するまでのあいだに短い期間の行動変数が生成され、いったん安定すると試行の最後までほとんど変化しない。

0%より多く、50%よりも小さいあいだは、多少の差こそあれ、図4.11とほぼ変わらない分布となっている。つまり、相互に影響を与えあうがときどき異なるデータが混入し、それが影響を及ぼすという経過を経ている。

50%以上の場合、誤り率があまりにも高いため、コンテキストは適切な行動変数を維持できない。たとえば誤り率が100%のときは、コンテキストが保持する行動変数とは一致

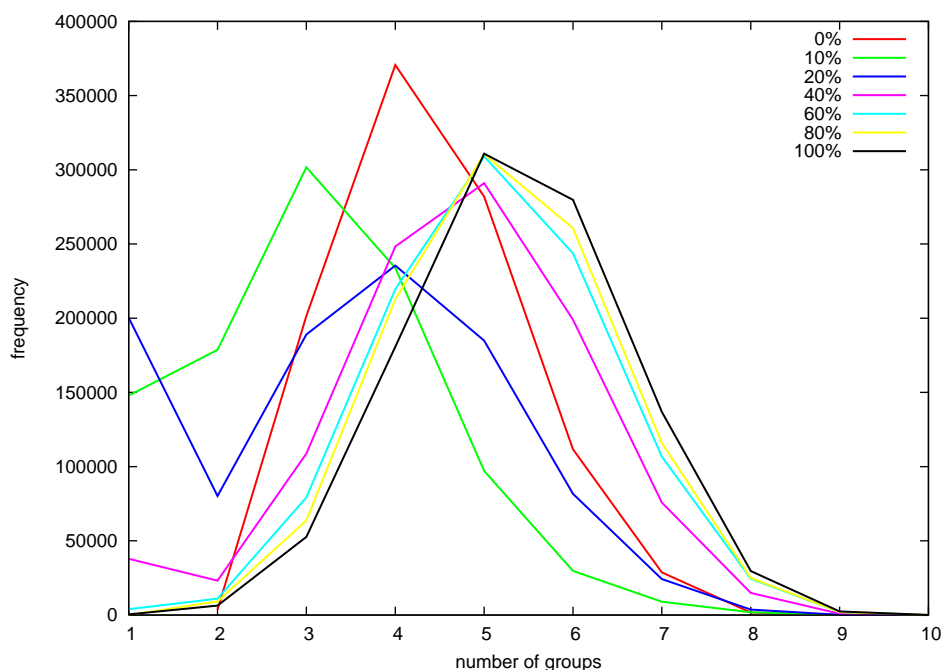


図 4.20: 誤り率とグループ数の関係

しないものしか選ばれない。この場合、コンテキストの中身が安定しないため、秩序関数が真を返しても、出力する行動変数は無秩序なままである。したがって長期的な安定が実現されず、ある短い期間以上のものは出てこなくなる。

次にグループ数の頻度の変化を調べた(図4.20)。

まず最頻値に注目すると、0%では4グループが最頻値だったが、10%ではいったん3に減少し、20%で4に戻り、それより大きいと5グループで安定した。0%の場合には隣接するロボットがそれぞれ異なる行動変数を保持していても互いに影響を与えあわない。これに対して誤りが少しでもあると他のグループの行動変数の影響を受けるため、整列しやすくなると考えられる。このためグループの統合は起こりやすくなり、グループ数は減少する。しかし誤り率が高くなると、コンテキストの中身が無秩序化する傾向が強まり、グループ数が増大する傾向を見せるものと考えられる。

次に、グループ数が1の場合に着目する。つまり、全体がでひとつの行動変数が保持される場合がどれくらいあったかという点、0%の場合はまったく存在していない。10%では150,000ステップほど存在し、20%で増大し200,000ステップほど存在した。そして、それより高いと逆に頻度は減少した。誤りがまったく存在しなければグループの統合は起こらず、同一の行動変数でロボット群全体が統一されることがないことは直観的に想像できるし、誤り率が極めて高くても同様に統一されることはない。その中間的なところである20%前後では、相互に影響を与えあうために行動変数が統一されやすくなったと解釈できる。ただし、上で説明した最頻値との間にはずれがある。これは次のように解釈できる。先述したように誤りが存在すると他のロボットの行動変数の影響を受ける。ただし10%の場合には誤り率が低いために他のロボットの影響は弱い。したがってロボット群全体で同一の行動変数を持つことは多くない。しかし、2つのグループに挟まれたロ

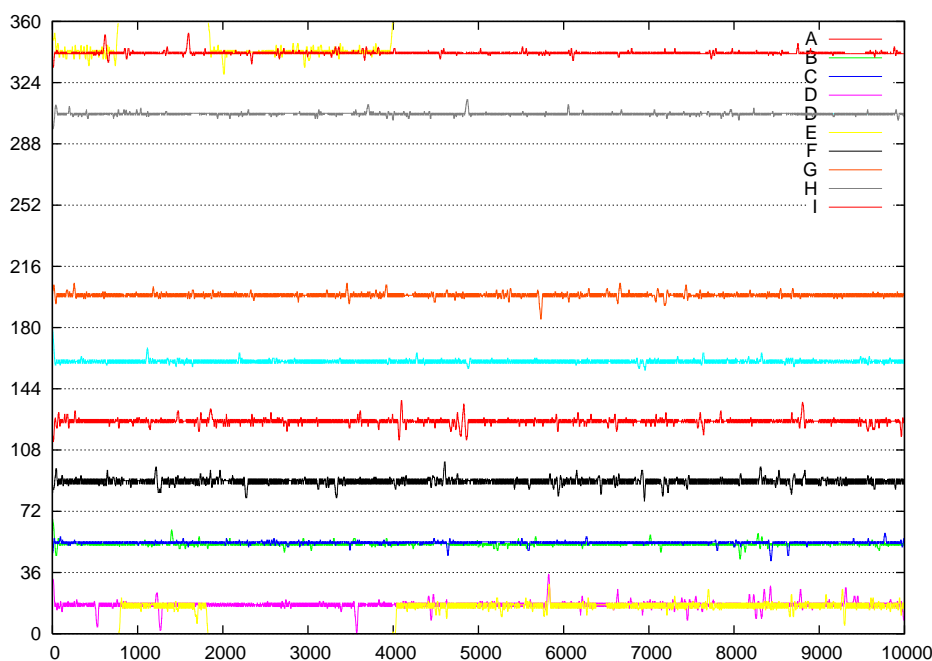


図 4.21: インタラクションがない場合の動作例

ロボットがどちらにも属さず独立した行動を示すことは抑制されるため、グループ数は結果として減少する傾向にある。誤り率が増加して20%になると、他のロボットの影響が強くなるためにグループの統合が発生して全体で同じ行動変数を保持することがより多くなるが、2つのグループに挟まれたロボットは両方の影響を受けて独立した行動を取る可能性がある。誤り率が20%を越すとコンテキスト内の異種データの割合が増大する。このため、全体で統一した行動を取りづらくなり、グループ数が1の場合の頻度が減少するものと考えられる。

4.9.4 インタラクションの影響

観測指向モデルは相互に観測しあうというインタラクションによって行動パターンが複雑に変化するというモデルである。そこで、システムの構成を変えずに、意図的に相互のインタラクションを行わせないことで行動パターンがどのように変化するかを調べた。

シミュレーション環境ではロボットはもっとも近くに位置するほかのロボット2体と自分自身の行動を観測しコンテキストを更新する。そこで、ほかのロボットではなく自分自身を3回観測することでシステム構成を変えずにインタラクションの存在しないロボットを実現した(図4.21)。

このように、互いにまったく影響を与えあわず、各ロボットが独立して特定の色を行動変数として選び、安定している。ロボットの行動が変化しづらいのは、自分自身の行動を三回ずつ観測するため、コンテキストへの依存性が強いためと考えられる。それでもまったく変化が引き起こされないわけではない。この例でも、2,000–4,000ステップほどのあいだだけ324–360の方向から0–36の方向に変化し、また324–360の方向に戻るロボット

がいる。ランダム配置において外部からの影響を受けない孤立系が存在すると述べたが、インタラクションが存在しない場合は各ロボットは単独で孤立系を構成しているのと同じであると考えられる。

さて、ほかの検討事項と同様に100回の試行を行いふるまいを調べた。まずロボットが特定の方向を向いた期間の頻度分布を調べた結果を図4.22に示す。図では比較のためにインタラクションが存在する場合の結果として図4.11の円形配置を同時にプロットした。

インタラクションが存在しない場合の分布はインタラクションが存在する場合の分布と顕著に異なっている。短い期間の分布はほとんどなく、長い期間の頻度が高い。つまり、ほとんどのロボットはまったく変化を起こさずそのまま試行を終えている場合が多いということを示している。つまり、ランダム配置における孤立系と同様にほとんど行動変数を変化させずに試行を終えていることがわかる。

次にグループ数の分布について比較した(図4.23)。この図では比較のために、図4.10における円形配置のプロットと、毎回ランダムに選択するモデルのものをプロットした。

このようにグループ数の分布は通常のものとは全く異なり、むしろランダムに近い。ロボットが自分の向いている向きを決めるときに周囲のほかのロボットの影響を受けないため、ランダムに選択されるのと同様であるからであると考えられる。

4.10 本章のまとめ

本章では、複数のロボットなどが存在するという状況を想定した。そしてその状況において互いの行動を観測しあうというインタラクションを実現し、インタラクションのなかから複雑な行動パターンを実現する観測指向モデルを提案した。

観測指向モデルは郡司らのセルオートマトンのモデルをベースにしている。郡司らのモデルでは、オートマトンが $\{0, 1\}$ 以外の第三の値である α を取りうるという状況を想定し、状態遷移関数を時間的に変化させて複雑な時間発展を実現していた。観測指向モデルはロボットを郡司らのモデルにおけるオートマトンになぞらえた。ただし、郡司らのモデルは $\{0, 1\}$ などの単純な離散値のみを扱うモデルであり、ロボットに対してそのまま表現できるものではない。観測指向モデルはセンサデータや行動変数を入力力として取り扱うよう郡司らのモデルを拡張したモデルである。

観測指向モデルでは、行動変数はコンテキストの中の値として表現される。コンテキストの中の値はロボットの行動を観測し、自分と類似するものを取り入れて更新されるため、互いに観測しあうことでロボットは相互に影響を与えあいながら同一の行動変数を保持しやすくなる。このことでロボット群は安定した行動変数を獲得する。

その一方で、観測には意図的に誤るという操作をつけ加えた。こうすることで、ロボットは自分と類似していなくても行動を取り入れることが可能となる。単に類似するものだけを取り入れるだけだとロボットの行動変数は特定のものに固定され変化しなくなるが、誤りを導入することでこれを回避した。この誤りにより、行動パターンの複雑な変化を実現した。

観測指向モデルを実装し、シミュレーション環境で動作を確認した。シミュレーション

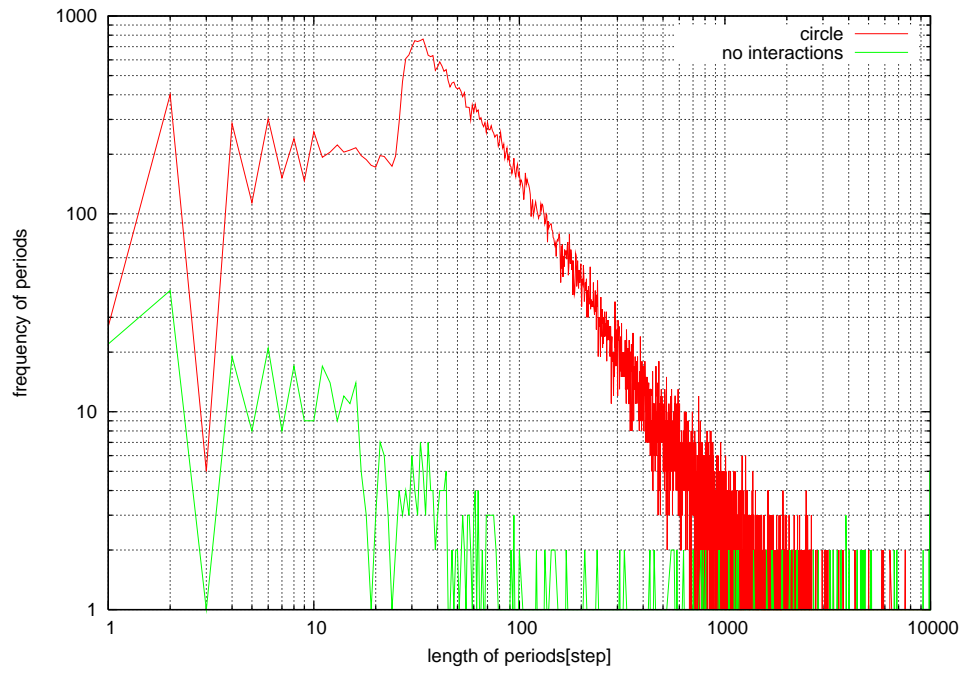


図 4.22: インタラクションがない場合の頻度分布

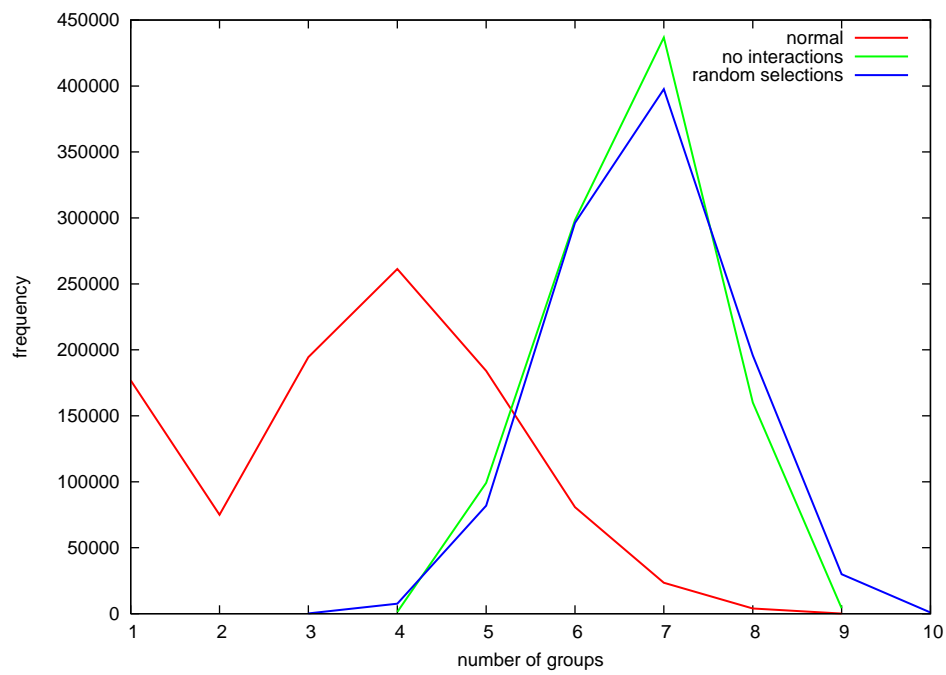


図 4.23: インタラクションがない場合のグループ数の頻度分布

環境では、特定の方向をしばらく維持しているが、時間の経過によってその方向が変化する、というように短期的な安定性と長期的な可変性をともに満たす行動をロボットが取っていることが確認された。また、位置的に近いロボットは互いの行動を観測しあうが、そのため実際に同じ行動変数を取りやすく、同一の方向を向きやすいことが確認された。すなわち、互いに影響を与えあうことでロボット群として安定した行動変数が維持されることが確認された。

また、シミュレーションでは、行動変数の維持される期間の長さについて頻度分布を調べた。この結果、ランダムに選択する場合とは異なり、非常に長い期間を維持する行動変数が存在する可能性があることが示された。

また、互いにインタラクションをしない場合と比較し、観測指向モデルではロボットが相互に観測しあうというインタラクションがあることから複雑な行動パターンが実現されていることを確認した。さらにコンテキストサイズや誤り率を様々に変化させ挙動の変化を調べた。これにより、観測の誤りがまったく存在しない場合には行動変数は変化しないことや、誤りが50%を越えると不安定化することがわかった。また、コンテキストサイズは行動の安定化よりは他のロボットが自分と異なる行動をしていることをどれだけ許容するかを示していることが示唆された。

以上により、提案する観測指向モデルによって、複数のロボットが存在し相互に影響を与えあう環境において、複雑な行動パターンを実現することが確かめられた。

次章では、環境内にロボットが単独で存在する場合においても複雑な行動パターンを実現するためのモデルを提案する。

第5章

属性変換モデル

本章では属性変換モデルによる多様な行動パターンの生成について議論する。

観測指向モデルでは、環境中にインタラクションを行う人間や、ほかのロボットが存在する状況を想定した。そして、環境中の人間やロボットの行動を観測することで多様な行動パターンを実現した。

しかし、エンターテインメントロボットが存在する環境は、かならずしもこの想定が成立するわけではない。ロボットが単独の場合、観測指向モデルは適用できない。

観測指向モデルでは、行動変数には特に前提を置かなかつた。シミュレーション環境では色を用いたが、実際には一致するかどうかの比較が可能であればデータの種類の問わない。

しかし、場合によっては行動変数に構造を与えると良い場面もあると考えられる。たとえば、ロボットが物体に注目しているとする。そのとき、単にその物体に注視しているというだけでなく、その物体の色であるとか、形状であるといった属性の情報を与えることで、より多様なインタラクションが可能になることも考えられる。

そこで周囲物体について、色や大きさといった属性とそれらに対応する値を持つという構造を仮定する。そしてこの仮定によって属性情報を介して多様な行動パターンが生成できると考えられる。本章で提案する属性変換モデルがそれである。

5.1 概要

属性変換モデルは、属性情報をもとに注意を向ける対象を切り替えるというモデルである。本節ではその基本的な考え方を説明する。

属性変換モデルでは、システムは周囲物体が色や大きさといった属性を持つと認識する。ただし、システムは複数ある属性のうち特定の属性しか重視しない。残りの属性は単純に無視し、ある属性の値だけを見て注意を向けるかどうかを決定する。さらに、重視する属性は時間的に変化すると考える。

たとえば、大きさが同じで色が赤と青と異なる2つのブロックが環境中に存在するとしよう。ここで「大きさ」と「色」が属性にあたる。システムは最初、色だけに注目しており、赤いブロックの方にだけ注意を向ける。ところが時間の変化によって、重視する属性が色から大きさに変化する。重視する属性が変化してもロボットの注意がすぐ変わるわけではなく継続されるから、それまで注意を向けていた赤いブロックの大きさが、注意を向

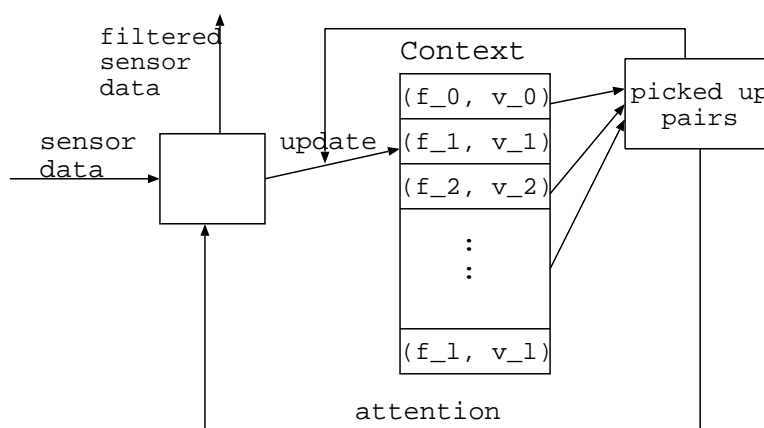


図 5.1: 属性変換モデルの構成

けるかどうかの基準となる。ところがどちらのブロックも同じ大きさなので、結果として両方のブロックに注意を向けるようになる。さらにここで属性がふたたび大きさから色に変化するとする。このとき、それまでの注意に従って赤と青という2種類の色情報が受け入れられるようになるが、どちらか一方が優先される。結果として(赤いブロックに注意が戻ることもあるが)青いブロックの方へ注意が遷移する。

属性変換モデルでは、この例における「大きさ」のように類似する値をもつ属性値をもつ物体の間でのみ、注意の変化が起こりうる。これを繰り返すことで、注意を向ける対象が局所的には安定しながら次々と変化していくというモデルである。

5.2 構成図

属性変換モデルの構成図を図 5.1 に示す。

属性変換モデルでも内部にコンテキストがあり、これがその時点での行動変数を保持している。ただし、構造を仮定しなかった観測指向モデルと異なり、コンテキストの各要素は属性名と値のペアという構造を有する。

属性変換モデルでは、周囲環境の物体の情報が入力される。この物体の情報はコンテキストの保持するデータをもとに行動変数に合致するかどうか判断され、合致すると判断されたものは注意を向ける物体であると判定する。

また、注意を向けると判断した物体の属性値はコンテキストに追加される。観測指向モデルと同様に、コンテキストの大きさはある有限の値であり、追加とともに既存のデータがひとつ破棄される。この操作によってコンテキストは古すぎるデータを捨てる。しかし、注意を向けると判断した物体の情報がコンテキストに追加されるので、それまでコンテキストが保持していたデータと類似するデータだけが追加され、安定した行動変数が維持される。しかしながら、コンテキストが保持する属性をうまく変化させることで、前節で説明したように重視する属性を変化させ、大域的な変化を引き起こす。

次に、観測における具体的な操作と、コンテキストの更新における具体的な操作について詳しく述べる。

5.3 物体の観測

属性変換モデルでは、物体は属性と値のペアの集合として表現される。すなわち、

$$obj_i = \{(f_0, v_0), (f_1, v_1), \dots, (f_m, v_m)\}. \quad (5.1)$$

となる。ここで f_j は「色」や「大きさ」のような属性の名前(または識別子)であり、 v_j はその f_j に対応する値である。ある物体について、たとえば大きさは変わらないはずなので、大きさ属性のペアが2つ以上あるといった場合は考えにくい。2つ以上の色からなる区分にわけられた物体、という複雑な状況も考えられるが、本研究ではこの問題は取り扱わない。すなわち、ある物体は同じ属性のペアを高々1つしか持たない。したがってある物体について属性 f_j が決まれば対応する v_j は一意に定まるとする。この関係について「物体 obj_i において属性 f_j は値 v_j を持つ」といい、 $v_j = value_i(f_j)$ と表記する。

属性変換モデルでは、コンテキストには行動変数ではなく属性と値のペアが蓄えられる。つまりコンテキスト c は、

$$c = \{(f_0, v_0), (f_1, v_1), \dots, (f_{l_c}, v_{l_c})\} \quad (5.2)$$

のような集合として表現できる。属性変換モデルでもコンテキストのサイズは有限の値であり、 l_c と表記する。

なお、ひとつの物体は複数の属性を同時に持てなかったが、コンテキストにはそのような制約はない。任意の $i \neq j$ に対して $f_i = f_j$ が成立する可能性がある。逆にコンテキストでは $value$ 関数は定義されず、要素には添字によってのみアクセスされる。そこで c の i 番目のペアを c_i と表記する。

以上の定式化をもとに具体的な観測のプロセスを説明する。ここではある物体 obj_i が周囲の環境にあるとする。観測のプロセスではこれをコンテキスト c と照合し、注意を向ける対象か否かを判定する。

観測の際は、コンテキスト全体と直接比較するのではなく、比較用のデータを別に構築する。このデータを「注意」と呼ぶ。観測指向モデルではコンテキストからランダムに一つの要素だけが選ばれて比較されていた。属性変換モデルでは、コンテキストから単一の要素だけを取り出すと、特定の属性にだけ偏りすぎる。一方コンテキスト全体から行動変数を構成すると変化が起こりづらい。そこでコンテキストの一部から注意を構成し、注意と物体を比較するとした。コンテキストを構成するペアからランダムに n 個のペアを選択したものを s と呼ぶ。そしてこの s から行動変数 A を構成する。

$$s = \{(f_0, v_0), (f_1, v_1), \dots, (f_n, v_n)\}, (f_i, v_i) \in s \Rightarrow (f_i, v_i) \in c \quad (5.3)$$

$$A = a(s). \quad (5.4)$$

ここで a は s から A を作る関数である。これは以下で述べる操作を行う。まず s を属性ごとにまとめる。すると属性と、対応する値の集合になる。そしてこの集合から以下のように属性値の範囲を作る。

- 集合の要素数が複数あるときは、その最大値と最小値を範囲とする

- 集合の要素数が一つだけのときは、その値の近傍を範囲とする

すなわち A は属性と値の範囲の集合となる。この A をもとに物体に注意を向けるか否かを決定する。これは単純で、物体 obj_i に対してすべての $(f_j, r_j) \in A$ について、 $value_i(f_j)$ と r_j を比較し、範囲に含まれるかどうかをチェックする。すべてのチェックが成立したものは注意に含まれるとみなされ、注意を向ける対象であると判断する。

$$obj_i \subset A \Leftrightarrow \forall (f_j, r_j) \in A, v_j \in r_j, v_j = value_i(f_j) \quad (5.5)$$

そして注意を向ける物体とわかったものを行動変数として出力する。

以上の、コンテキストサイズ l_c およびコンテキストから選択するペア数 n によって属性変換モデルの観測が定まる。

5.4 コンテキストの更新

次にコンテキストの更新プロセスについて説明する。観測指向モデルと同様に、属性変換モデルでもコンテキストは動的に維持され、行動パターンの複雑さに貢献する。

観測指向モデルでは、その時の行動変数と一致する行動変数だけをコンテキストに追加することで、コンテキストは類似するデータを保持しやすいという傾向を与えた。一方、一定の確率で誤ることで現在の行動変数とは一致しない行動変数を追加した。この誤りによって大域的な可変性を実現した。

属性変換モデルでも基本的には同様である。すなわち、その時の注意と一致する物体の属性データだけをコンテキストに追加することで、なるべく同じ物体の属性データが集まり、同様な注意が構成されるという傾向となる。

しかしながら、注意と一致するかどうか判断した物体は複数の属性をもつ。したがって、どの属性を選択するかを決定しなければならない。

属性変換モデルでは、コンテキストの保持する属性に偏りを与えることが重要である。たとえば、環境中に同じような大きさの、色の異なるブロックが1つずつあるとする。このとき、「色」という属性だけがコンテキストにあるときには色でしか判断を下さないため、安定して一方の物体にだけ注意を向けるという状況が実現される。つまり、安定した行動変数のためには、コンテキストは同じ物体に関するデータをなるべく多く集めるだけでなく、属性もなるべく同じものを集め、偏らせることが望ましい。

そこで、注意を構成するのに用いた s を利用する。 s はコンテキストからランダムに取り出された部分集合なので、コンテキスト内の分布を反映していると考えられるからである。具体的には s のなかでもっとも多く出現した属性を選択する。ただしもっとも多く出現した属性が複数ある場合は、そのなかからランダムに選択するとする。

しかし、それだけでは問題がある。というのは偏りが大きくなり、コンテキストが一つの属性しか持たなくなると、必ずその属性だけを選択するようになるからである。つまりひとつの属性だけが選択され、変化しなくなってしまう。すると行動変数がまったく変化しなくなり、多様な行動パターンが実現されない。コンテキストに含まれる属性は偏りを与えることが望ましいが、その偏りは変化することが望ましい。

これを実現するため、 s がすべて同じ属性だったときには、最も多い属性を選ぶのではなく、ランダムに属性を選択することにする。 s はコンテキストが含む属性の分布を反映するものだから、すべて同じ属性のときはコンテキストの大部分がその属性で占められている可能性が高い。これで、コンテキストが多く含む属性を選択しやすい傾向を持たせながら、ひとつの属性に統一させ変化がなくなることを防ぐ。

属性が選択されたら、ランダムに選択したコンテキストの1要素を削除し、新しい属性を追加する。

$$c_k \leftarrow (f(s), \text{value}_i(f(s))). \quad (5.6)$$

ここで f は上で定義した属性選択関数である。

5.5 実験と考察

5.5.1 実験環境

属性変換モデルの動作を調査するために、シミュレーション環境で動作を調べた。

シミュレーション環境では、環境中に obj_1 と obj_2 という2つの物体があることにした。各物体は1-5と名付けられた5つの属性からなり、それぞれの属性について値を持つ。 obj_1 と obj_2 は属性1については同じ値を持つが、残りの4つの属性では値を共有しない。このため、ロボットがほかの属性を重視して判断を下しているときは一方の物体にだけ注目するが、属性1を重視するときは両方の物体のデータがコンテキストに追加される。そのため、しばらく一方の物体に注意を向けているが、重視する属性が属性1になると両方の物体に注意を向け、さらにそこから別の属性を重視するように変化すると、また一方の物体に注意を向ける、という変化が起こることが期待される。

また、以下で試行を行う際は、コンテキストサイズ $c_l = 8$ 、コンテキストから選択する数 $n = 4$ とした。

5.5.2 典型的な行動結果

最初に、典型的な動作結果を図5.2に示す。

ここで横軸は時間、縦軸はその時注意を向けている物体の名前 (ID) を示す。図からわかるように、しばらく一方のオブジェクトに注意を向けているが、時間が経過すると注意が他方に変化するという切り替えが実際に起こっていることが確かめられた。たとえば、最初は obj_1 に注意を向けているが、400ステップほどでこの注意は終了し、 obj_2 に遷移している。この注意はそのまま1,100ステップほど持続し、いったん obj_1 に遷移する。そのあとも1,400ステップほどで obj_1 に注意が移り、示し400ステップほど過ぎて1,800ステップあたりでまた obj_2 に注意が移っている。このように注意を向ける物体は変化している。

属性変換モデルでは、システムは特定の属性を重視して判定を下すが、重視する属性そのものが時間的に変化する。そして、同じ属性値をもつ複数の物体があるとき、その同じ値をもつ属性を重視しているときに注意を向ける物体が移りかわる、というものであつ

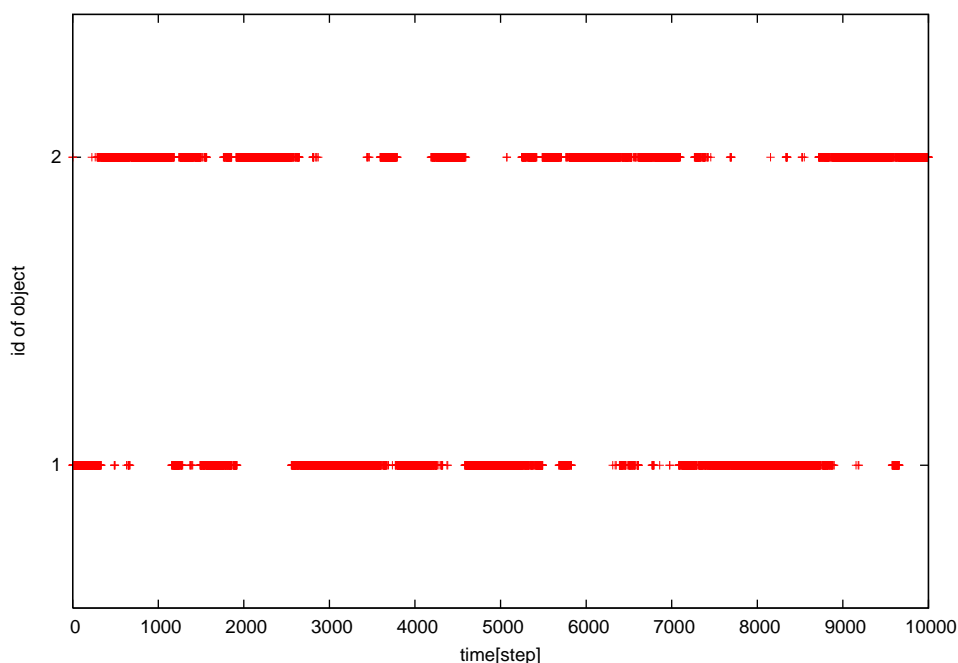


図 5.2: 属性変換モデルにおける注意を向ける物体の遷移

た。このため、共通する属性を重視するタイミングでは両方の物体に注意が向くというタイミングが存在する。実際、図 5.2 でも、 obj_1 のプロットと obj_2 のプロットは時間的に多少の重なりを持っていることが確認される。

次に、実際に注意を向ける物体が切り替わっているときのシステムの内部状態を調べることで、システムがどのような行動変数のもとに判断を下しているかを調べた。具体的には、各ステップにおいて、コンテキストを構成するペアのなかで、どの属性がどれくらい存在するかを数え上げ、その遷移を調べた。

実際に図 5.2 を実行させた場合の、2,000 ステップ前後のコンテキスト内の属性の分布を図 5.3 に示す。

この図でも横軸は時間を示している。また縦軸は、ある属性がコンテキスト中にいくつのペアを持つかという数を示す。各プロットは 5 つある属性に対応しており、上にあるほどコンテキスト内で大勢を占めているという意味となる。コンテキストサイズは 8 としたので、8 に達している時は、コンテキストはその属性しか含まないということを示している。

この図では、ほとんどすべての場合に、ある一つの属性がコンテキストの大部分を占めているということを示している。すなわち、システムは特定の属性を重視して判定を下すということが実現されている。また、ある属性がコンテキストの多数を占めている状態はしばらく持続するが、時間が経過するとコンテキストを占める割合は現象し、ほかの属性が大勢を占めるという状態に変化する。つまり、重視する属性が時間的に変化している様子が実現されていることがわかる。ある属性を多く持っているとき行動変数を構成するときその属性だけが選ばれる可能性が高まるため、数を減らす可能性が高まるためである。2,000 ステップ前後のように、二種類の属性が同等の量で存在している場合もあるが、こ

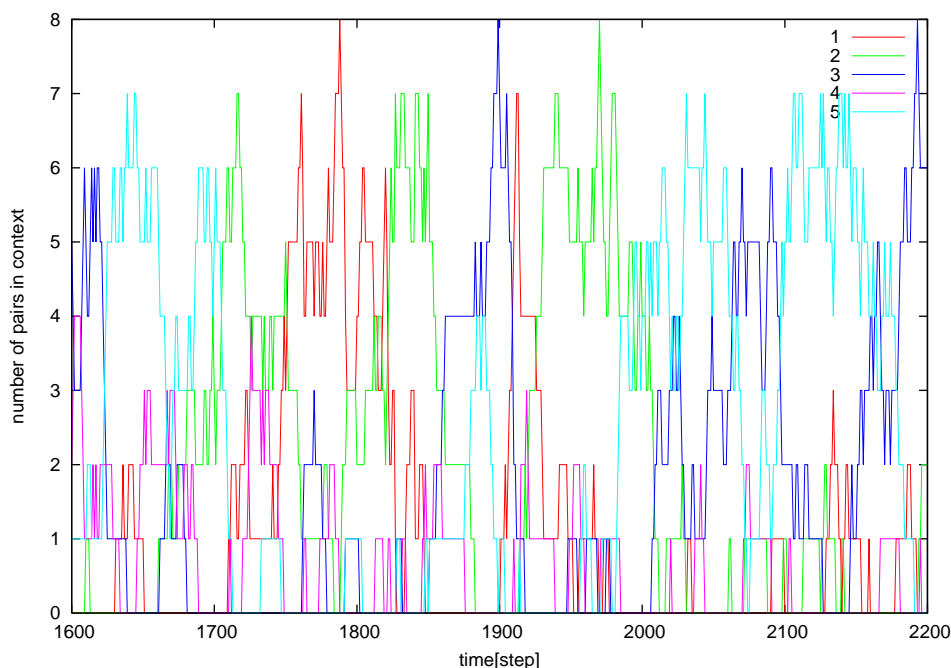


図 5.3: コンテキスト内の属性分布の遷移

れも長続きせず，一方の属性が優勢となる場合がほとんどとなる。

次に，実際に遷移が発生した 1,800 ステップ前後に注目する．このときは 1 番の属性がコンテキストの大勢を占め，行動変数に大きな影響を及ぼしている．ところで先述したように， obj_1 と obj_2 は属性 1 の値が共通しているとしている．したがってこの時には両方の物体に注意を向けやすくなっている．コンテキストは属性 1 だけでなくほかの属性も含んでいるが，ここでは obj_1 のデータと obj_2 のデータが混在している状況であると考えられる．ここで属性 1 以外の属性がたまたま多く選ばれて行動変数が構成されると，注意を向ける物体は一方の物体だけとなり，その物体のその属性値が数を増やす．グラフからは属性 1 の数が減って，次に属性 2 の数が大勢を占めるようになることがわかるが，ここでは obj_2 のデータが使われているため，注意を向ける物体が obj_1 から obj_2 に遷移したと考えられる．

以上から，属性を介して注意を向ける物体を変化させるという属性変換モデルの挙動が実現されることが確認された．次に，属性変換モデルによって実現される行動パターンを調べる．観測指向モデルと同様に，100 回の試行を行って物体への注意がどれくらい長く維持されるかを調べ，各期間ごとの頻度を計算した (図 5.4)．

観測指向モデル同様に，非常に裾野の広い分布となっている．このため，極めて長い期間を維持される行動変数が存在する一方で，短い期間のものも存在していることがわかる．このため，属性変換モデルと同様に複雑な行動パターンが生成できると考えられる．

各期間の長さや頻度を掛け合わせることで，その期間が全体に占めるステップ数を調べることができる．それを累積したのが図 5.5 である．

このグラフは，観測指向モデルの場合，とくに円形配置の場合と類似した増加の仕方をしており，全体の大半は長いステップ数の期間で占められていることがわかる．

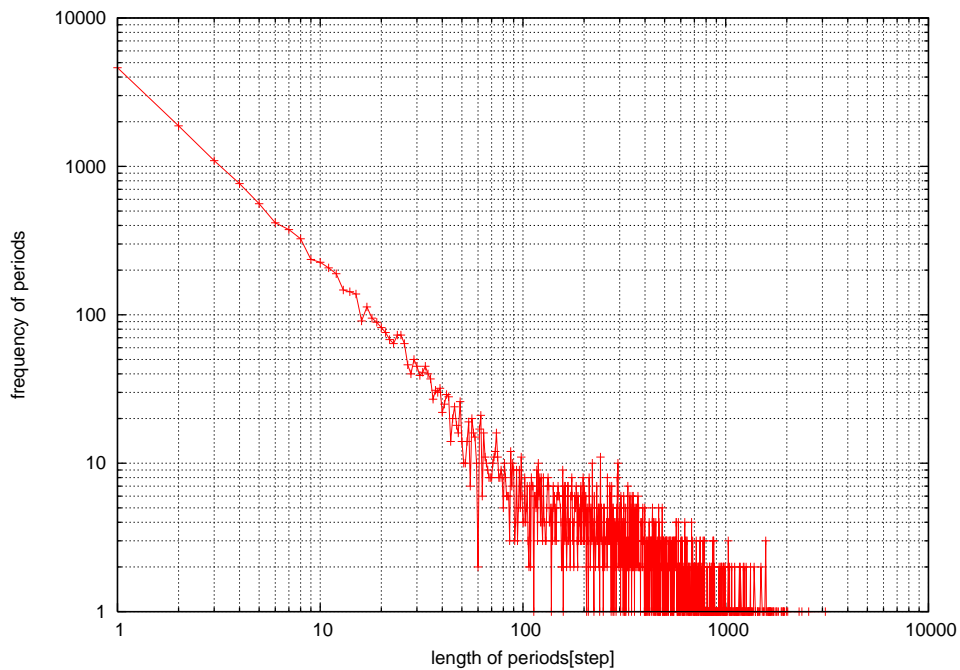


図 5.4: 属性変換モデルにおける物体への注意を維持する期間の頻度分布

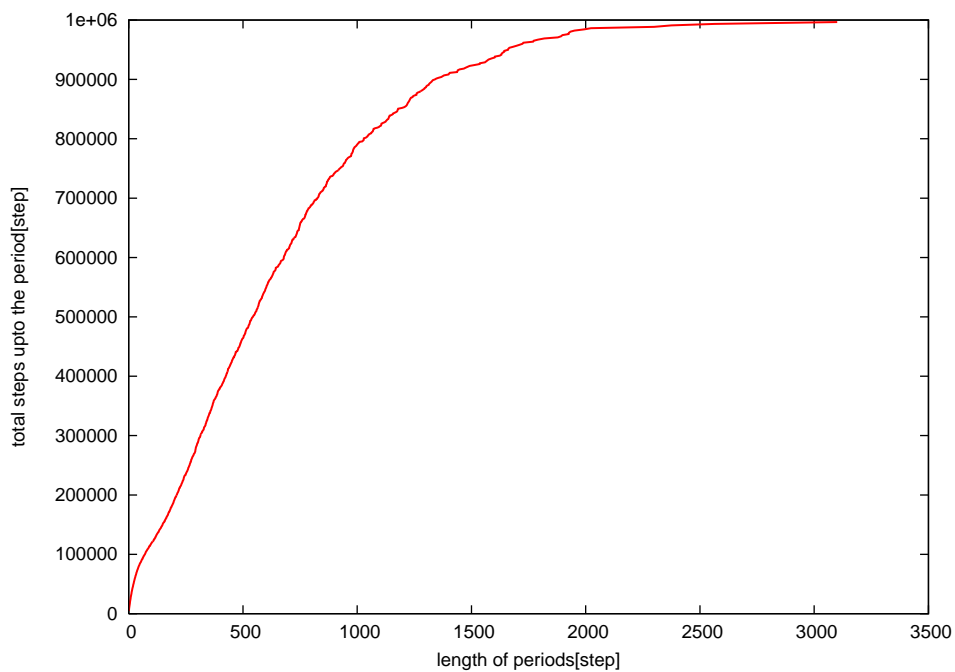


図 5.5: 維持された期間の累積

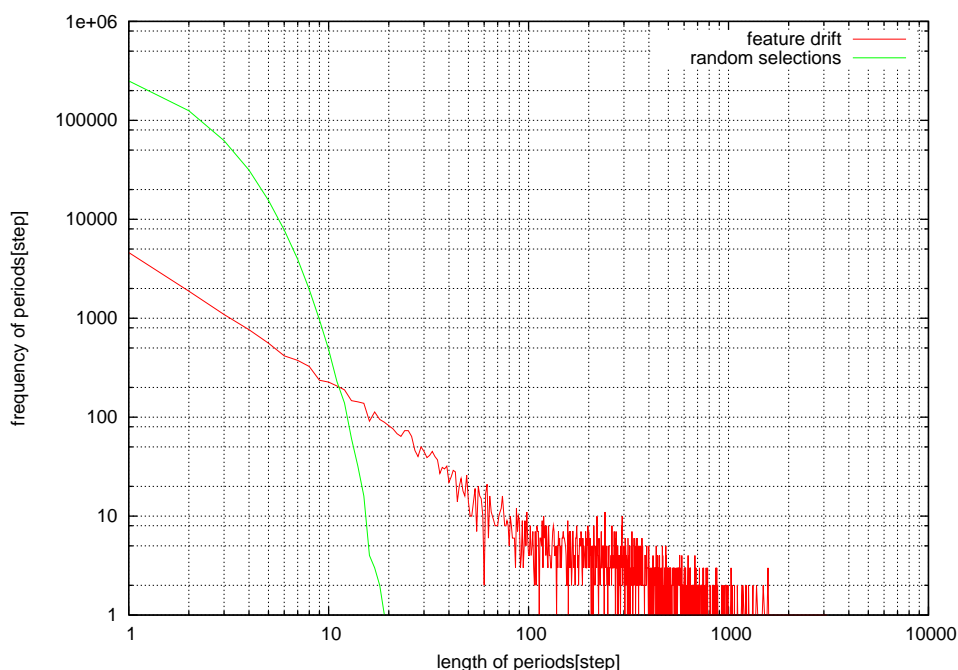


図 5.6: ランダムとの比較

5.5.3 ランダム選択モデルとの比較

以上から、属性変換モデルも観測指向モデルと同じような分布を持っていることがわかった。次に、これを単純に毎回ランダムに選択するモデルと比較する。ランダム選択モデルでは毎回2つある物体のうちどちらかを50%の確率で選択する。

一方の物体に注意を向ける期間の頻度分布で比較を行った(図5.6)。

図5.6からわかるように、ランダム選択モデルでは非常に短い期間の注意しか生成されず、長期間の注意はまったく出現しない。属性変換モデルも、観測指向モデルと同じように、単純にランダムな選択を繰り返すのとは明らかに異なるふるまいを示すことが明らかとなった。

5.5.4 パラメータの検討

次に、属性変換モデルの挙動を制御するパラメータについて検討した。検討した内容は、コンテキストサイズ l_c と、行動変数を構成するときに選択する s のサイズ n とした。

最初に、コンテキストサイズを変更させ、変化を見た。100回の試行を行い、注意が維持された期間の頻度分布を調べた結果を図5.7に示す。

コンテキストサイズが8と12の場合の概形は類似しているが、4の場合は少し異なる。300ステップほどで頻度が急激に減少し、1000ステップを越える期間の出現頻度はほとんど存在しない。これはコンテキストが小さいと保持されるペアはすぐに上書きされてしまうため、長期間同じ行動変数を維持するのが困難となる。したがって、ある長さを越えて維持される注意が存在しなくなると考えられる。

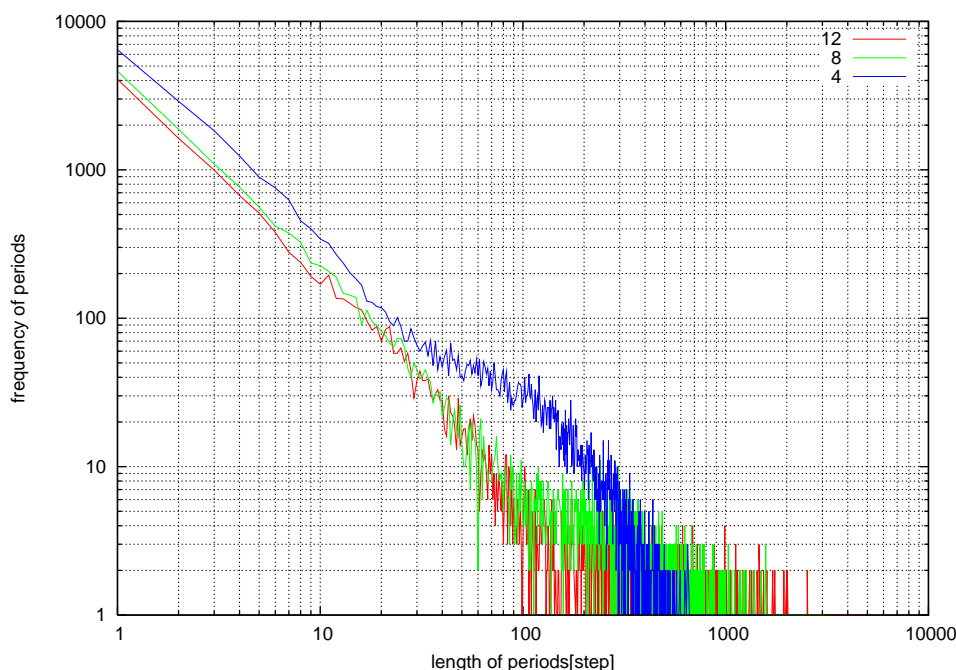


図 5.7: コンテキストサイズによる挙動の変化

表 5.1: コンテキストサイズと最大属性の割合の関係

コンテキストサイズ	4	8	12	16	32
平均割合 (%)	64.58	62.33	61.68	61.65	63.02
標準偏差 (%)	16.56	13.82	12.61	11.85	9.968

コンテキストサイズが8と12の場合は類似しているが、12の場合の方がより短い期間で頻度が減少し、かわりにより長い期間の頻度が増加している。つまりコンテキストが大きくなることで変化が生じづらくなっている。実際、コンテキストサイズが増えるほど、一回の更新による影響は小さくなり、変化が生じるまでに要するステップ数も増えていくと考えられる。

そこで、コンテキストの中身について調べた。各ステップでコンテキスト中で最もペア数の多い属性を調べ、そのペア数がコンテキスト全体に占める割合を調べた。100回の試行を行い、平均を取った。この値が高いほど、コンテキストは一つの属性の影響を強く受けていると言える。

計算結果を表 5.1 に示す。

この結果から、コンテキスト中で最大数の属性の割合は平均すると6割よりやや多いという程度であること、また予測に反して平均の割合はコンテキストサイズによらないことがわかった。また、標準偏差はコンテキストサイズが大きくなると減少している。ただし、コンテキストサイズが4の割合は0%、25%、50%、75%、100%の5通りしか取りえないから、平均割合である64.58%という値そのものとなることができない。つまりコンテキストサイズが大きくなることで、より平均の値に近い割合が表現されたということ

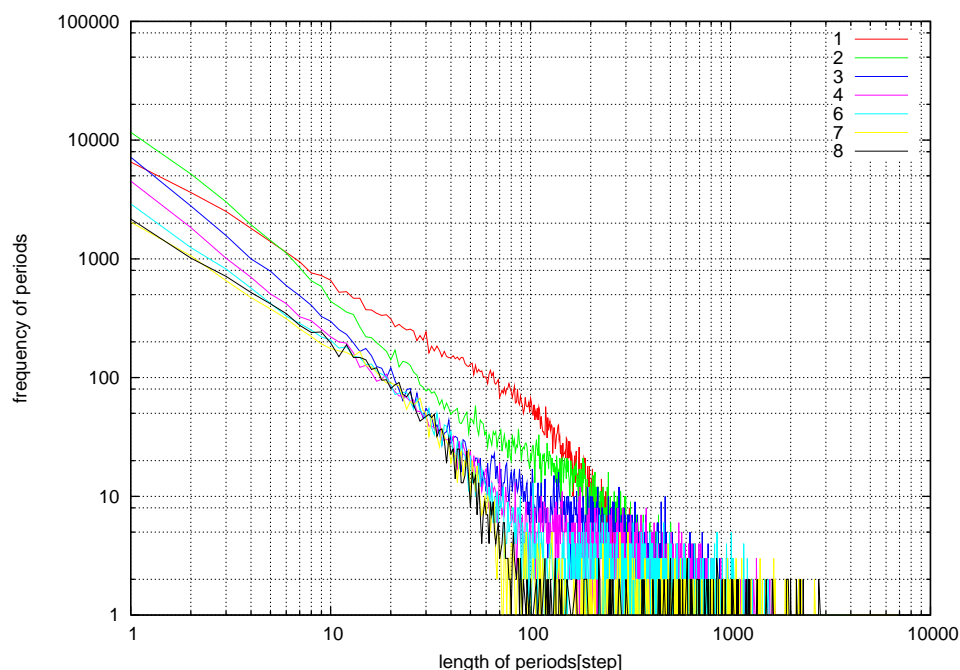


図 5.8: 選択数 n による挙動の変化

あると考えられる。

コンテキストサイズによって割合は変化しないが、システムの挙動は変化している。これはそのコンテキストに含まれる実際の個数が関係していると考えられる。つまりコンテキストサイズが8のとき、最も優勢な属性はコンテキストに5つ含まれている。一方、コンテキストサイズが32のときは割合は同じでも20個である。1ステップで上書きされるのは1つなので、仮に異なる属性で上書きするとコンテキストサイズが8の場合は最短で5ステップで済むが、32の場合は最短でも20ステップを要する。このため、重視される属性が変化するまでに要するステップ数に差が生まれるからと考えられる。

次に、 s のサイズ n を変化させた。同じく100回の試行を行い、注意が維持された期間の頻度分布を調べた結果を図5.8に示す。

図では隠れてしまっているが、 $n = 1$ のときは100ステップを越えたところで頻度が急激に減少し、300ステップを越えることはなかった。 $n = 1$ の場合、アルゴリズムの関係上、つねに属性がランダムに選択される。そのため特定の属性のみを重視しつづけることができず、同一の行動変数を長期間維持することが困難になっていると考えられる。また、 n は大きくなるにつれて100ステップから500ステップあたりまでの頻度は減少し、その分、長期の頻度が上昇した。 n が増大することにより、コンテキスト内の状況をより正確に反映できるため、重視される属性が容易には変化しなくなるためであると考えられることができる。

そこで、コンテキスト内で最も数の多い属性を含む割合を比較した。これを表5.2に示す。

このように n が大きくなると最も数の多い属性がコンテキストもまた大きくなる。実際、 n が大きいということはそれだけコンテキストの中身を正確に反映しやすくなること

表 5.2: n と最大属性の割合の関係

n	1	2	3	4	5	6	7	8
平均割合 (%)	39.80	46.05	55.92	62.33	67.97	71.70	75.19	77.09
標準偏差 (%)	9.733	11.22	13.62	13.85	13.85	13.20	12.27	11.72

いうことである。つまり、コンテキスト中に異なる属性があるとそれが取り出されてしまうため、最も多い属性で更新されやすくなり、コンテキストに占める割合が増大するものと考えられる。また、 n が小さい場合は、1つの属性のみがコンテキストを占めていなくても偶然同じ属性だけが選ばれ、結果として更新される属性がランダムに選ばれることが起こりやすくなる。逆に n が大きければそのようなことは起こりづらくなるため、平均の割合も増加する。

以上をまとめると次のようになる。コンテキストから行動変数を構成する際に使われる s のサイズ n は、コンテキストがもっとも多く含む属性の占める割合に影響を与える。高いほど高い割合となる。つまり n は特定の属性への偏りの度合いを決めるパラメータである。また、コンテキストサイズはこの割合には影響を及ぼさない。しかしコンテキストサイズが大きくなれば同じ割合でもペアの数が増えるから、結果としてより長期の注意が生成されやすくなる。つまりコンテキストサイズは変化しにくさを決めるパラメータとなる。

5.6 本章のまとめ

本章では、ロボットの周囲にある物体は色や大きさといった属性値によって認識されるという枠組みを想定した。そして、その枠組みのなかで、特定の属性を重視するように偏りを与えることから多様な行動パターンを生成する属性変換モデルを提案した。

属性変換モデルでも観測指向モデルと同様にコンテキストを持つ。コンテキストは属性と対応する値のペアとなっており、ステップごとにコンテキストから行動変数を構成する。行動変数はステップごとに微妙に変化するが、コンテキストの中身のペアは行動変数に一致した物体からしか取られないから安定的に同じ物体だけが行動変数に一致するものと判断される。

属性変換モデルではコンテキストになるべく同じ属性が多く集まる仕組みを持っているため、行動変数は特定の属性を重視したものとなる。しかしコンテキストがすべて同じ属性だけになってしまうと変化が起こらなくなってしまう。そこで、コンテキストが一つの属性を多く持ちすぎているときにはほかの属性を選択する仕組みを取り入れることで、変化する可能性も組込める。

属性変換モデルもシミュレーション環境で動作を確認した。この結果、複数ある属性のうちひとつを重視するという状態が持続するが、重視する属性が時間の経過によって交代する、ということが実現された。また重視する属性の時間的な変化によって、ロボットは一方の物体に注意を向けるが、時間が経過すると注意が変化した。すなわち局所的な安定性と大域的な可変性がそれぞれ満たされていることが確認された。

シミュレーションではまた、属性変換モデルの挙動を決定する様々な変数を変化させ、挙動の変化を調べた。この結果、行動変数を構成するために選択するペアの数 n によって、システムが特定の属性をどれだけ重視するかが決まる、コンテキストのサイズによって特定の属性を重視する期間の長さが決まるということがわかった。

次節では、実際のロボットに属性変換モデルを実装し動作させた例について述べる。

第6章

コミュニケーションロボット Robovie による動作例

前章までで、本研究で提案するエンターテインメントロボットのための行動モデルを説明した。本章では、提案する行動モデルを実ロボットに組み込み動作させることで、行動モデルが適切に動作することを確認する。

6.1 システム構成

本章で動作させるロボットのシステムには属性変換モデルを用いて、単体のロボットで複雑な行動パターンを生成する。

システムの構成は図 6.1 の通りである。

ロボットは“挨拶をする”とか“物体を指差す”とか“人間の方を見る”といった行動モジュールを持つ。そして、周囲の状況からルールにもとづいて次の行動を決定し、出力する。この点は従来研究におけるルールベースのシステムとまったく同じである。

違いは属性変換モデルを実装したモジュールにある。この属性変換モジュールは通常のセンサ入力を受け取り、行動の決定を行う協調モジュールに行動変数を与える。具体的には、センサ入力から物体を識別し、その物体の属性情報を取り出す。そして属性変換モデルにもとづいて注意を向ける物体かどうかを決定し、その情報を付加してロボットの行動を決定する。協調モジュールは元のセンサ入力は受け取らず属性変換の結果だけから行動を決定する。また、行動モジュールが保持する“挨拶する”とか“物体を指差す”といった行動には、行動の対象が決められておらず空白となっている。協調モジュールは属性変換モジュールが出力した注意情報にもとづいて選択した行動の空白を適切に埋め、実際に出力する行動をつくり出す。このメカニズムによって、ロボットの行動は属性変換の行動変数に間接的な影響を受けることになる。もちろんロボットの行動を動作させるときは生のセンサ入力は必要であるが、どの行動を選択するか、という点には生のセンサ入力は用いない。

ただし、このことによって協調モジュールが属性変換モデルからの影響を一方向的に受け取るという関係しかないと考えるのは誤りである。たとえばロボットが首を動かせば視界は変化し、センサ入力に変化が引き起こされ、属性変換モジュールにも影響を及ぼす。すなわち、協調モジュールもまた、実行する行動の変化によって間接的に属性変換モジュール

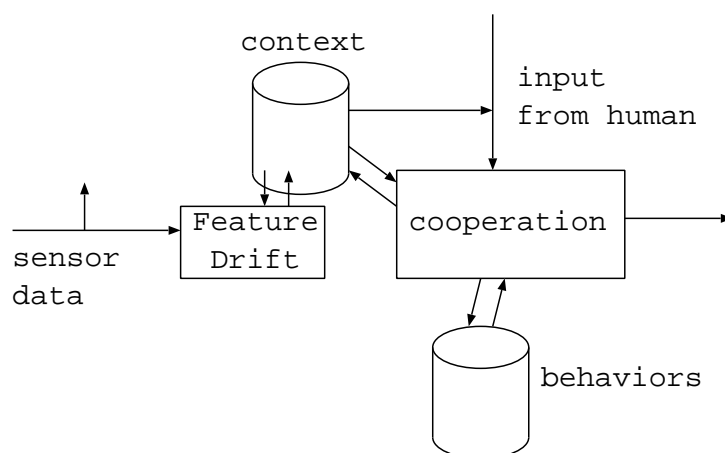


図 6.1: ロボットシステムの構成図

ルに影響を及ぼす。このフィードバックループによってロボットの多様な行動が実現されている。

6.2 人間からのフィードバック

本システムでは、人間とのインタラクションも考慮に入れた。つまり、たとえば人間からの問い合わせがあれば現在注目している物体を指差す、といった動作を実現する。

これは人間からの働きかけに応じて行動が駆動する例であるが、人間からの働きかけというのは、それに応じて単に内部状態を提示するという単なる問い合わせには限らない。人間からの働きかけによって協調モジュールだけでなく属性変換モジュールにも影響が及ぶことも考えられる。

たとえば人間が「これを見よ」という指示を出し、ロボットがそれに従って指示された物体に視線を移す、というシナリオを考える。この場合、人間の指示によって「物体に注目する」という行動が駆動され、しかもその行動の対象となる物体も変化する必要がある。

これを実現するにも様々な方法が考えられる。たとえば、その状況では属性変換を使わず、行動の空白を人間の行動に従って埋めるといった方式である。本システムではそれとは異なり、属性変換を用いたアプローチを取る。つまり、人間から「これを見よ」という指示が出されると、指示された物体を認識し属性情報を取り出して、その物体の属性情報によってコンテキストのすべてのペアを上書きする。これによって属性変換モデルの仕組みはそのままに注意を向ける物体をルール側から制御できる。

ロボットは複数の属性を持つから、ルールによって強制的にコンテキストを上書きする場合にもどの属性で上書きするかを決めなければならない。本システムでは単純に、コンテキストの通常更新と同じ方法を取ることにした。つまり、コンテキストから n 個のペアを取り出し s を構成する。そしてその中で最も多く含まれる属性を選ぶが、もし n 個のペアがすべて同じ属性であればランダムに選択する。ただし、将来的には人間からの指示の内容や状況にあわせて更新する行動を決定するというアプローチは考えられる。

この仕組みを用いると、最初は人間の指示に従って注意を生成するが時間の経過によってコンテキストの内容が変化し、ほかの物体に注意を切り替えてしまうかもしれない。というのは、属性変換モデルを用いているため注意を向ける物体は変化する可能性があり、いつ切り替わるかわからないからである。これは利用される状況によっては問題であるが、エンターテインメントという用途においてはむしろ有用なことである。人間が指示を出してもロボットはいずれ飽きてしまいほかに注意を向けてしまうという状況からは複雑なインタラクションが実現されると考えられる。

6.3 実装した行動と動作シナリオ

以上のシステムをコミュニケーションロボット Robovie [Kanda *et al.* 04] 上に実装した。Robovie は ATR 知能ロボティクス研究所¹ によって開発されたヒューマノイドロボットである (図 6.2)。Robovie は人間の上半身を模した形状をしており、2本の腕を持ち、腕はそれぞれ4自由度で動作する。また3自由度をもつ首を有し、その上の頭部には目のように左右に並べられたパンチルトカメラとマイク、スピーカを持つ。ほかにも肩に取り付けられたポールの先に全方位カメラを持ち、周囲の状況を認識できるほか、24個の超音波センサを持ち、腕、胸、肩、および頭にタッチセンサを持つ。

ただし本システムでは両眼からの画像入力のみを用いて物体および属性値の認識を行った。具体的には物体の属性は色と大きさの2種類とし、彩度と色相を用いて背景から物体を識別し、画像上の物体の大きさから大きさの情報を識別した。

実験風景を図 6.3 に示す。

ロボットがおり、脇に人間が立っているという状況である。ロボットの前には机があり、赤と青と色は異なるが同じ大きさのブロックが置かれている。ロボットはこのブロックを認識し、属性変換モデルに従って注意を向ける物体かどうかを判断する。

ロボットの行動は、“物体の方向を向く”、“物体を指差す”、“好き嫌いを判断する”の3つと、人間の「これを見よ」という指示に従ってコンテキストを強制的に上書きするものである。

“物体の方向を向く”という行動では、視界内の物体のなかから注意を向けている物体を探索する。存在していれば、それが視界の中央に位置するように首を動かす。もし人間が注意を向けている物体を動かせば、それに応じてロボットは首の向きを変え、自分が注意を向けている物体のある方向へ向きを変える。視界内に注意を向ける物体がなければ何もしない。

“物体を指差す”と“好き嫌いを判断する”という行動は人間の問い掛けに反応してのものである。“物体を指差す”は視界内に複数の物体が存在する場合にのみ実行される行動であり、人間の問い合わせに対して自分の注意を向けている物体を指差し、「これ、おもしろい」と発話することで自分の注意を向けている物体を人間に提示する。“好き嫌いを判断する”では視界内には物体がひとつしかない状況に実行される行動であり、人間の問い合わせに対して視界内に存在する物体がその時の行動変数と一致するかどうか判断する。一致するなら「面白い」と発話し、さもなくば「つまんない」と発話することでロボット

¹<http://www.irc.atr.co.jp/>



図 6.2: コミュニケーションロボット Robovie

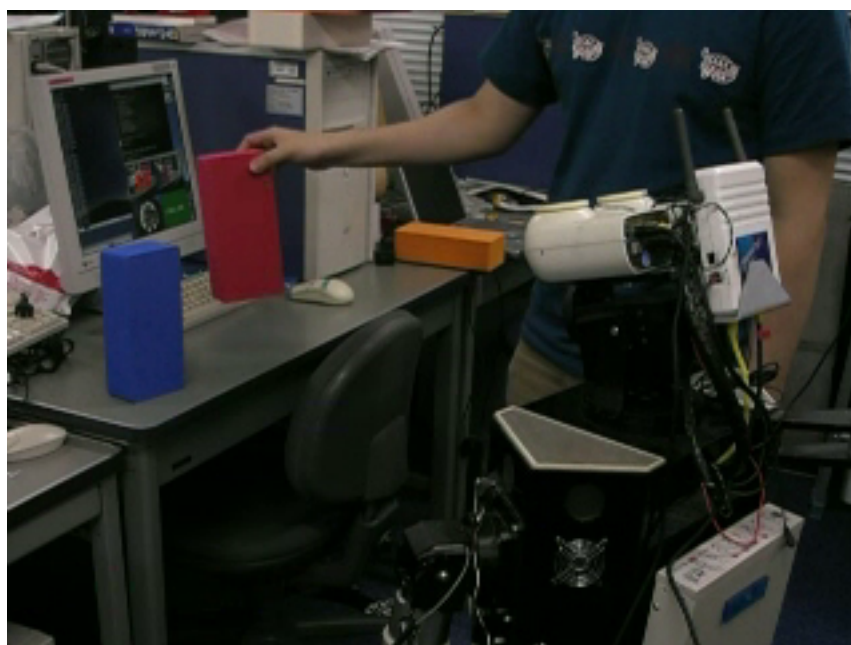


図 6.3: 実験風景

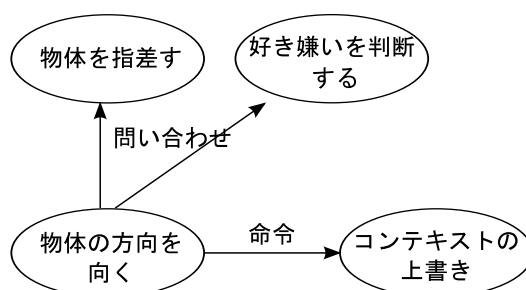


図 6.4: ロボットに実装した行動の関係

は自分の内部状態を人間に提示する。

「これを見よ」という人間の指示に従ってコンテキストを更新するという行動については、前節で説明した通りである。

今回のロボットの行動シナリオは非常に単純なものとした。ロボットは何もしていないときは“物体の方向を向く”行動を維持している。そして人間からの問い合わせがあると、視界内の状況に応じて“物体を指差す”か“好き嫌いを判断する”かの行動を取り、実行が完了するとまた“物体の方向を向く”に戻る。また人間からの命令があれば、それによってコンテキストを強制的に更新する。行動同士の関係を図 6.4 に示す。

このようにシナリオは非常に単純なものである。しかし周囲の環境の影響によってコンテキストは多様に変化するため、人間が同じ問い掛けをしてもロボットの反応は多様に変化することが期待される。

6.4 実験結果

本節では、実際に Robovie で動作させたときの結果を示す。

今回のシステムでは、コンテキストサイズ l_c は 8 とし、行動変数を構成するときに取り出す数 n を 3 とした。

以下ではまず、実際に動作させたときに起こったことを解説する。

まず最初は Robovie だけで人間はおらず、視界内には青と赤のブロックがどちらもあるという状態とした。この状態でしばらく放置し、Robovie にコンテキストを維持させた。

それから人間が Robovie の脇に立った。今回の実験では、人間の働きかけがない場合は Robovie の行動は“物体の方向を向く”である。そこで人間が赤いブロックを持って動かしたところ、Robovie は赤いブロックの方に注意を向けていたため、その動きに応じて Robovie は首の向きを変え、人間の動きに追従した(図 6.3)。人間はそのままブロックを動かし、Robovie の向かって右にあった赤いブロックを向かって左の位置に置いた。そしてすぐに人間は「どっちが好き？」と Robovie に尋ねた。赤いブロックはまた机の上に置きなおしたので、このとき Robovie 視界内には赤と青の両方のブロックがある。人間はすぐに尋ねたのでコンテキストはまだ変化しておらず、赤いブロックを指差して「これ面白い」と発話した(図 6.5)。そこで次に、Robovie の視界内に青いブロックを持ってきて問い合わせを行った。この場合は視界内には青いブロックしかない。Robovie のコンテキストは大きさではなく色が優勢を占めていたため、Robovie は「つまんない」と発話した。さらに、それまで Robovie の視界内になかったオレンジ色のブロックを Robovie の視界内に持ち、さらに問い合わせを行った。オレンジ色のブロックは色は異なるが大きさは赤や青のブロックと同じである。しかし Robovie のコンテキストは更新されていないため、青のブロックと同じく行動変数と一致せず、同じように「つまんない」と発話した(図 6.6)。

次に強制的なコンテキストの更新を行った。オレンジ色のブロックを Robovie の視界内に位置するよう維持したまま「これを見よ」という指示を出した。このとき Robovie の視界内にはオレンジ色のブロックしかないため、そのブロックが人間の指示した物体である。そこでオレンジ色のブロックのデータによって Robovie のコンテキストは上書きされる。すると Robovie はオレンジ色のブロックに注目するようになる。人間が何も働きかけないときの Robovie の行動は“物体の方向を向く”であるから、オレンジ色の物体を動かすことで Robovie は首の向きを変える。そして赤いブロックを取り除き、オレンジ色のブロックを机の上に置いた(図 6.7)。この状態でしばらく待ち、ロボットのコンテキストが切り替わるのを待った。それから改めて人間がふたたび「どっちが好き？」と問い合わせた。すると、コンテキストはオレンジ色のブロックから青いブロックに切り替わっていたため Robovie は青いブロックの方を指差し「これ面白い」と発話した(図 6.8)。

以上の実行結果から、Robovie は自発的に自分の注意を向ける対象を選択し、その注意に従って行動を決定していることが実現されることが確認された。また Robovie は時間の経過に従って注意を自発的に変更し、人間の問い合わせと周囲の状況から複雑な反応を返すことも実現された。さらに人間の指示に従って Robovie のコンテキストが強制的に上書きされ、人間の指示に従う行動も実現された。

この試行で Robovie の注意の変化を図 6.9 に示す。

ただし、本実験では Robovie は周囲の物体の ID というものを持たない。ただし実験で



図 6.5: 青いブロックを指差す



図 6.6: オレンジ色のブロックを見せる



図 6.7: 人間の命令に従ってコンテキストを上書きする



図 6.8: 青いブロックを指差す

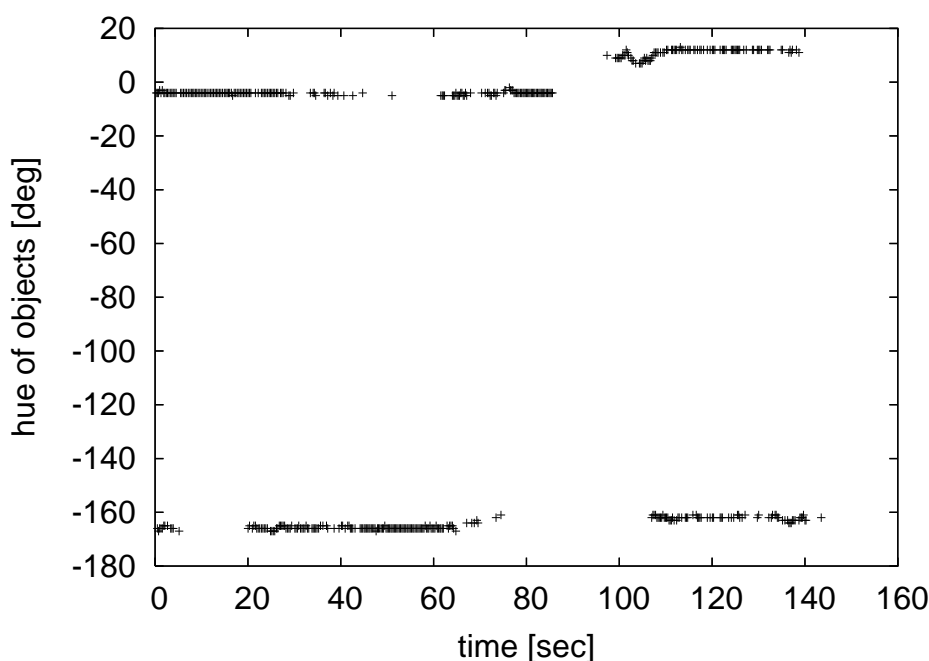


図 6.9: Robovie の注意の変化

用いた物体はどれも色が異なるから、現在注意を向けている物体の色からひとまず識別できる。そこで縦軸を物体の色相値(度)とし、横軸を時間(秒)としてその時間、注意を向けている物体の色にもとづいてプロットを行った。

人間が最初にブロックを動かしたのはこのグラフでは70秒を過ぎた頃である。それまでの時間でも、Robovieはまず赤いブロック(色相0度近辺)に注意を向けたあとで青いブロックに注意を向け(色相-160度近辺)、また赤いブロックに注意を戻しているという切り替えが発生していることがわかる。さて、人間が赤いブロックを動かしたあと、90秒近辺で人間からRobovieへの最初の問い掛けが発生している。このときRobovieは赤いブロックを指差すと同時に顔を人間の方に向けている。その結果、視界内には物体が存在しなくなるため、90-100秒近辺では何もプロットされなかった。この間には青いブロックやオレンジ色のブロックを持ってきてRobovieに問い合わせるということも含まれる。この場合には視界内に物体は存在しているが、問い合わせに使った青やオレンジのブロックは行動変数と一致しないため、図にはプロットされていない。そして100秒を過ぎたあたりでオレンジ色のブロックを提示してRobovieのコンテキストを切り替えている。するとオレンジ色のブロックのデータ(色相17度近辺)の値がコンテキストに書き込まれ、図にはオレンジ色のブロックに注意を向けていることを示すプロットが出現する。そして110秒あたりでオレンジ色のブロックを机の上に置いた。このとき、Robovieの視界内にはオレンジ色と青の2つのブロックが存在している。Robovieのコンテキストは安定せず、オレンジと青の間を注意が頻繁に行き来していることがわかる。人間はこの間はしばらく待っている。しかし、140秒あたりで問い合わせたときはわずかに青が優勢になっており、Robovieは青いブロックに注意を向けていることを示す行動が生成された。

6.5 本章のまとめ

本章では、前章で提案された属性変換モデルにもとづいて実ロボット上で動作するシステムを提案・実装した。

ロボットシステムでは、センサ入力から属性変換モデルにもとづいて注意を向ける物体かどうかを決定した。ロボットの行動を決定する協調モジュールではこの注意情報にもとづいて実際に出力する行動を決定し、複雑な行動パターンが実現される。

また、ロボットが自発的に行動を選択し変化するだけでなく、人間からの働きかけにも考慮した。すなわち、人間から指示が発生した場合、その指示に応じてコンテキストを強制的に書き換えることでロボットの行動変数を変更することで人間の指示に従う行動を実現した。

提案したロボットシステムはコミュニケーションロボット Robovie 上に実装した。そして、単純な行動シナリオを記述し、実際に動かしてロボットの行動を調べた。

デモにおいて、ロボットは属性変換モデルにもとづいて行動変数を多様に変化させた。このため、人間が同じ問い掛けをしたとしても、内部のコンテキストにもとづいて反応を様々に変化させることができることが確認された。

また、提案したロボットシステムには人間からの指示に従ってロボットが行動をする機構も組み込んだ。この機構では人間の指示内容によってコンテキストを強制的に書き換えるものである。この動作についてもテストを行い、人間が物体を提示して「これを見よ」と指示をすると、その物体の属性情報でコンテキストを上書きし、その物体を見るという行動が実現されることが確認された。

第7章

議論

本章では、これまで説明してきた内容全般に関して議論を行い、本研究の意義について述べる。

本研究で提案した観測指向モデルと属性変換モデルでは、行動変数が維持された期間についての考察を行い、どちらもランダムとは異なる極めて特異な分布を持っていることが示されていた。そこでまず、その分布について検討する。次に、観測指向モデルと属性変換モデルに関する想定状況を比較し、両者の関係を明確にし、2つのモデルを組み合わせたモデルの可能性について考察する。それから、本研究でわかった問題点について今後の課題として考察する。最後に、本研究はエンターテインメントロボットの行動という目的であったが、それ以外に利用される可能性について議論する。

7.1 行動パターンの多様性と分布の関係

最初に、本研究で提案したモデルにおいて出力される行動変数が維持される期間の分布に関して考察する。

観測指向モデルも属性変換モデルも、ともに単純に毎回ランダムに選択するモデルとの比較を行い、どちらのモデルも単純ランダムモデルとは顕著に異なるパターンを示すことが確認された。

比較に用いたのは、行動変数が持続する期間を調べそれぞれの期間が出現する頻度を計算する分布であった。この結果、ランダムではごく短い期間の分布が極端に多く、一定の期間を越えることが決して起こらなかった。これに対して本研究で提案した2つのモデルでは、短い期間の頻度がわずかに抑えられ、より長い分布のものも頻度こそ少ないものの存在するという分布となっていた。このため、より長い分布が存在し、安定した行動変数が実現できていることが確認された。

ランダム選択モデルは非常に単純であるため、多くのエンターテインメントロボットで行動パターンに多様性を与えるために用いられている (例えば [Ishiguro *et al.* 01] など)。しかしランダム選択モデルと異なるだけでは本研究で提案するモデルの意義を示すことにはならない。そこで本研究における行動パターンの目的をあらためて示し、ランダム選択モデル以外のモデルとの差異を示す。

本研究における行動パターンの目的は、2章で議論したように次の2つである。

- 局所的には安定し、変化しないこと

- 大域的には変化する可能性があること

ランダム選択モデルは大域的な変化は起きているものの、局所的に不安定である。また、行動を後天的に獲得するシステム(たとえば[太田 03]など)は局所的に安定であるが、大域的に変化が引き起こされない。

しかし、この2点を満たす他のモデルも考えることができる。たとえば、10秒ごとに行動変数を切り替えるというモデルを考える。このモデルでは、人間はロボットの行動が切り替わるタイミングを容易に予測できる。したがって本研究の目的は達成されない。人間が飽きない多様な行動パターンのためには、行動変数が切り替わるまでの期間も様々に変化しなければならない。

そこで、行動変数を切り替えるまでの期間を一様分布の乱数によって変化させるというモデルを考える。この場合、局所的には行動変数が維持されて変化しないし、乱数で決めた時間が経過すれば行動変数も切り替わるのだから大域的には変化する。さらに行動変数が切り替わるまでの期間も乱数で定めるために様々に変化する。

一様分布では短い期間も長い期間も均等に出現する。しかし、短い期間と長い期間が同じ回数だけ出現したとすると、累積した合計時間は長い期間の方が長くなる。したがって、行動変数の系列全体を考えると、一様分布では長い期間の方が大きな割合を占めることになる。

これに対して本研究で提案するモデルでは短い期間の方が少し頻度が高く、長い期間の方が頻度が低い。つまり、一様分布において生じている偏りを是正していると考えられる。本研究の目的はエンターテインメントロボットの行動パターンを多様にすることである。つまり、より多様なパターンが出現することが望ましい。一様分布より本研究で提案したモデルの方がより多様な行動パターンが実現される。

本節では、本研究で提案したモデルにおいて、行動変数を維持する期間の分布について議論を行った。しかし観測指向モデルも属性変換モデルも、単なる確率分布とは異なる意味を持つ。たとえば観測指向モデルではロボットが互いの行動を観測しあうインタラクションがある。したがってロボットたちは同じものに注目しやすくなる傾向を持つ。

次節では、提案する2つのモデルの間の関係について、モデルの意味に注意しながら議論する。

7.2 観測指向モデルと属性変換モデルの関係

本節では観測指向モデルと属性変換モデルについて、実現される行動の意味について議論することで両モデルの関係を明らかにする。また、両者を組み合わせたモデルの可能性について議論する。

観測指向モデルでは、環境中に複数のロボットなどが存在するという環境を想定している。そして、他のロボットの行動を観測し、類似する行動を取ることでロボット群として安定した行動を取る。しかし行動が不安定化することもある。不安定化すると無秩序に行動をするようになる。無秩序な行動は場合によっては周囲に影響を及ぼし、群全体の行動を変えることもある。すなわち、観測指向モデルでは、ロボットが互いに観測しあうというインタラクションが存在しており、ロボットは観測する相手の行動と類似した行動を

取ったり、あるいは異なる行動を取ったりする。このインタラクションの中から複雑な行動を実現している。

これに対して属性変換モデルでは、環境中に他のロボットの存在は仮定していない。かわりに様々な物体が存在するという環境を想定している。その環境内の物体は属性と値のペアの集合として認識される。しかし、行動変数を作るときには多数ある属性のうち一部だけをもとに判断を下す。この制限のためにロボットはむしろ値を共通する属性を無視することができ、特定の物体のみを安定して選択できる。しかし、判断に用いる属性は動的に変化する。すると属性の変化に応じて注意を向ける物体が次第に変化していく。

属性変換モデルでは属性のつながりから注意を向ける物体を変化させていく。つまり、色の異なるブロックの間で注意が切り替わる時には、大きさや形状といった共通する属性がなければならない。したがって、環境中に存在する物体が少なかったり、まったく種類も何もかもバラバラな物体しかなかったときには、まったく変化が生じないこともありうる。つまり、属性変換モデルの実現する複雑な行動パターンは、実際には環境中にどのような物体が存在するかというロボットの置かれた状況に強く依存している。環境の情報が単純であれば複雑な変化は起こりづらい。しかしこのことは裏を返せば、環境が複雑化し様々な物体が多数存在する環境では極めて複雑な変化を起こすということである。すなわち、属性変換モデルは、周囲環境の複雑さに応じて複雑な行動パターンを生成するモデルであると考えられる。

また同様に考えれば、観測指向モデルでも行動が複雑化するのはいかに観測しあうという状況を想定しているからであると言える。たとえばロボット群のなかに一体だけ、まったく行動パターンを変えないロボットが存在したとする。すると周囲のロボットはそのロボットに強く影響を受け、その行動を模倣するようになるだろう。もちろん、時間の経過により一部のロボットが離脱し、グループを形成して異なる行動を取ることも考えられる。しかし離脱したグループは行動を変えないロボットの影響を受けるのに対して、行動を変えないロボットは何の影響も受けないから、しばらくすると離脱したグループも元に戻って同じ行動を取るようになる。つまり、行動の変化がほとんど置きなくなると考えられる。観測指向モデルは基本的には観測する対象の行動を模倣するというものであるから、観測する相手が行動を変化させなければ、それに引き摺られて行動を変化させなくなるのである。すなわち、観測指向モデルは観測の対象となる、他のロボットの行動の複雑さに応じて複雑な行動パターンを生成するモデルであると考えられる。

このように、属性変換モデルにしても観測指向モデルにしても、まったく何もない環境で複雑なパターンを生成するというモデルではない。環境や他のロボットの複雑さに依存しながら行動パターンを生成している。

さて、両モデルは想定する環境も、複雑さを実現するために必要な条件も異なっている。そこで、双方のモデルを組み合わせることでどちらの状況でも複雑な行動パターンを実現できると考えられる。

最も単純には、状況に応じて切り替えるということが考えられる。すなわち、環境中にほかのロボットや人間がいれば観測指向モデルにもとづいて行動を決定し、さもなくば属性変換に切り替えるというモデルである。しかし、これだけでは切り替えによって不自然な行動を取る可能性があるかもしれない。つまり、属性変換モデルから観測指向モデルに切り替わり、また属性変換モデルに戻ったときに、属性変換モデルが持つコンテキストの

中身が最初に切り替わる直前のままだったとすれば、まったく周囲の環境とそぐわない行動を取ってしまう可能性もある。利用されていない場合でも観測からコンテキストの更新を行うとすれば対処も可能であるが、その場合でも切り替えの直前と直後ですべて異なるコンテキストを持つことになり、不連続性が生じてしまうかもしれない。

そこで両モデルでコンテキストを共有するという可能性が考えられる。しかし観測指向モデルと属性変換モデルではコンテキストに要求する構造が異なるため、不整合が生じる可能性がある。具体的には、観測指向モデルによってコンテキストを更新するときどの属性を選択するのが適切かを決めるのは困難である、ということが挙げられるだろう。

このような問題を解決し、双方のモデルを不整合なく統合するのは今後の重要な課題である。

7.3 今後の課題

前節の最後で、今後の課題として観測指向モデルと属性変換モデルを統合するモデルの可能性について議論した。本節では、それ以外に残された課題について検討する。

7.3.1 モデルの拡張

観測指向モデルにしても属性変換モデルにしても、まだ拡張の余地は残されていると考えられる。そこでまず、どのような点が考えられるかを議論する。

観測指向モデルは他のロボットの行動の複雑さに応じて複雑さを生むということを述べたが、本研究ではまったく同等なロボット同士が観測しあうという状況しか想定しなかった。異種のロボットが存在する環境で生じうる問題については未検討である。そのような状況について検討することで、最終的には人間とも互いに影響を与えあうことが実現されるだろう。

また、観測指向モデルのシミュレーション環境では、観測をする相手は単純に最も位置の近いロボット2体と定めていた。スケーフリーネットワークなど、より複雑なネットワーク構造を用いた方がよい可能性がある。また、ロボットによって観察するロボットの数を変化させたり、誰を観測するかという関係を変化させることでモデルの挙動に変化が生じるかもしれない。さらに現実の状況ではロボットが移動することで観測する相手が新たに増えたりいなくなることもあるだろう。そこで観測する相手を動的に切り替え、あるいは観測する人数を増減する仕組みを追加し、その効果を検討することも重要であると考えられる。

属性変換モデルでは環境の物体の複雑さから複雑な行動パターンを生成するというモデルであった。しかし本研究では効果の検証のため、それほど複雑な環境での検討は行っていない。また、環境中の物体には制限があり、保持する属性の数はつねに同じであり、また一つの属性は一つの値しか持てない。しかしながら、現実的な状況ではこの前提が成立しない場合もある。たとえばパーツごとに色の異なる装置では、色に対応する属性は複数存在すると考えることができる。本研究の段階では、このような物体はパーツごとに切り分け、別々の物体として認識し、実際の行動の段階で統合することで対処することに

なる。しかしそれでは上手く行かないかもしれない。また、属性によっては物体によって存在したりしなかったりするものがあるかもしれない。本研究では「色」や「大きさ」といった物体の物理的な特性に絞っていた。これは認識が容易だからであるが、物体の属性はそれだけとは限らない。たとえば物体の所有者という概念も属性とみなすことが出来るだろう。しかし共有物や所有者不明のものは所有者という属性を持たない。あるいはペン立てや袋のような入れ物であれば、個数や種類など内容物に関する情報も属性であるとみなすことができるだろう。しかし容器でなければそのような属性は存在しない。一つの属性につきかならず一つの値を持つ、というのが本研究の前提となっているが、存在しなかったり複数存在したりする状況に対処するためには、モデルの拡張をしなければならないだろう。

また属性変換モデルでは、どのように物体を認識して属性を取得するかという問題を扱っていない。これはモデルの挙動に対して本質的でないからであるが、実際には重要な問題となりうる。環境中の何を物体とみなし、どうやって属性の情報を取得するか、というのは重要な今後の課題であるといえよう。ただし将来的にセンサネットワーク技術が発達し、環境に取り付けられたセンサから物体の情報を取得したり、物体にセンサが取り付けられて情報を取得できるとすれば、センサを取り付けられた物体の属性情報についてはカメラ画像などロボットによる情報処理のみに頼ることなく、センサネットワークから属性を取り出すことができるだろうと考えられる。とはいえ、物体には固有の状態があるため、必ずしもセンサネットワークに任せられるわけではない。たとえば軟らかい物体や折り畳みができる物体は人間の操作によって変形するかもしれないから、その状況に応じて形状という属性は変化するかもしれない。また、物体を見る方向によって属性値が変わるかもしれない。円筒形の物体は上から見れば円に見えるが、横から見れば長方形に見える。面によって物体の色が異なる可能性がある。もちろん、円にしか見えないとしてもロボットはその物体が円筒形であることを知っているのだから円筒形として扱うとしても良い。しかしロボットの置かれている状況に応じた主観的な情報を取り扱うことで、より複雑な行動パターンが生成できるようになるかもしれない。そのためにはロボットからの視点をセンサネットワークの側で識別して適切な情報を与えるか、あるいはセンサネットワークの返すデータとロボットの認識結果をうまく統合し、属性値を適切に変化させなければならないだろう。いずれにしても、センサネットワークとの連携を考慮したロボットの物体認識手法の改善を行う必要があるだろう。

7.3.2 パラメータの動的なチューニング

本研究で提案するモデルには、挙動の特性を変化させる様々なパラメータが存在した。本研究ではそれらのパラメータによる挙動の変化について検討を行った。

このパラメータは、ロボットの行動の特性を変化させるものであり、いわば個性に相当するものと考えることができる。つまり、行動変数を変化させやすいロボットやさせづらいロボットがいる、という風に考えることができる。

一方、こうしたパラメータを動的に変更することでより複雑なパターンを生成できることが期待できる。つまり、しばらくは行動変数を変化させづらい頑固な状態にいるが、状況が変化することで行動変数が変化しやすくなる、というように変化する。

三輪らはロボットにバイオリズムを組込むことで、周囲からの刺激に反応しやすい状態としづらい状態というのをつくりだした [Miwa *et al.* 04, 三輪 04]. また Breazeal ともロボットに疲労度のようなパラメータを設けることで、疲労が蓄積するとロボットが次第に周囲の刺激に反応しなくなるということを実現している [Breazeal *et al.* 99].

もちろん、疲労度がこの目的のために適切かはわからない。どのような意味をもつパラメータを与え、パラメータによってどのように変化するか、という点も含めて考慮する必要がある。

7.3.3 人間への心的効果の検証

本研究では提案する2つのモデルの出力するパターンを調べ、単純なランダム選択モデルとは異なる複雑なパターンを生成することを示した。また、実ロボットに単純なデモを実装し動作することまでは確認した。

そこで次の段階として、実ロボットと人間のインタラクションを行なわせることで人間への心的効果を検証することが考えられる。

ただしそのためには、より多数の行動を実装し、対話シナリオを準備する必要があるだろう。ロボットシステムにも変更が求められることが考えられる。一方で、比較対象として何を用いるか、という問題がある。比較対象としては、対象用のモックシステムを実装するよりは既存のエンターテインメントロボットを対象とする方が良いであろう。しかしそのためには、既存のシステムと組み合わせる動作させることが不可欠である。

また比較項目として、人間が飽きるという問題点を検証する必要があるだろう。人間がロボットの行動をある程度推測できるように習熟することで有意な差が生まれることが考えられる。したがって、ロボットと人間とのインタラクションは、短くとも数日にわたる長期間の、多数回のセッションを含むインタラクションとしなければならないであろう。

筆者の予測では、ロボットの行動には本研究の目的とする多様性を必要とする状況もあれば、必要とされない状況もあるであろうと考えている。たとえば叩かれることで喜ぶロボットなど、人間の予想にあまりに反する行動パターンが生成されると、むしろ問題となりうるかもしれない。様々な実験によって、本研究の成果が有効に活用できる箇所についての議論も可能となることが期待される。

7.4 他の応用事例の可能性

本研究は人間とインタラクションによって楽しませる、エンターテインメントロボットの行動を目的としていた。しかし提案するモデルはこの目的にだけ使えるというものではない。ほかの応用事例も考えられる。

とはいえ、多様な行動が求められるのはやはりエンターテインメント用途が多い。ロボットでなくても、デスクトップマスコットのような仮想エージェント、対話エージェントの行動に組込むことで、より豊かなインタラクションが実現できると期待できる。また、ゲーム用途にも応用できるかもしれない。

エンターテインメントでない例としては、美術館などの案内ロボットも考えられる。た

例えば絵画の作者や年代を属性情報として、属性が近接するように案内していくことも実現できる。あるいは人間の注意に応じて説明する内容を変化するような案内ロボットも実現できるかもしれない。

第8章

まとめ

本研究では、エンターテインメントを目的とするロボットを対象とし、ロボットの行動に多様性が必要であるという問題を指摘した。そして、多様な行動パターンを生成するメカニズムを提案し、この問題を解決した。

エンターテインメントを目的としたロボットはこれまでもいくつかの実例があるが、まだ問題点が残されている。特に行動の多様性という問題点はすでにいくつかの研究で指摘されているにもかかわらず、完全な解決には遠いのが現状である。

既存の研究では、何らかの手法によってロボットが新たに行動を獲得することによって、利用できる行動の種類を増やし、多様性を実現するというアプローチが取られている。しかしながら、それらの研究ではロボットの行動は特定の条件が成立したときにすぐ実行されるというものである。そのため、人間からの働きかけが同じであればロボットはまったく同じ反応を返してしまう。

一方、“嬉しい”とか“悲しい”とかいった感情をモデル化し、ロボットに組み込むことで行動を複雑に変化させる研究も存在する。しかし感情モデルに関する既存の研究では多様性について指摘されない。また、人間からの働きかけによる感情の変化はあらかじめ定められているため、感情のパターンは変化しない。

そこで本研究では、ロボットの実行する行動の種類を増やすのではなく、ロボットが実際に実行する行動パターンに着目し、行動パターンにおける多様性を実現することを目的とした。

行動パターンの多様性を実現するため、本研究ではロボットの行動システムに行動変数という概念を導入した。行動変数とは、行動の対象となる人や物体を意味している。そして行動変数が多様な変化をさせるモデルを導入することにより、ロボットの行動パターンを多様にすることを試みた。

本研究ではロボットの置かれる状況によって2つのモデルを提案した。ひとつは環境中に他のロボットや人間が存在するという状況であり、この場合は他のロボットや人間の行動の複雑さから複雑な行動パターンを生み出す観測指向モデルが適用される。もうひとつは環境中には他のロボットなどが存在しないという状況であり、この場合は環境中の物体情報の構造の複雑さから複雑な行動パターンを生み出す属性変換モデルが適用される。

観測指向モデルでは、ロボットは他のロボットの行動を観測し自分の行動変数が観測結果と類似するように変化していく。しかしながら観測には誤りを導入することで複雑に変化していくというモデルである。他のロボットの行動が複雑であればあるほど自分の行動

も複雑に変化していき、互いに観測しあいながらロボット群の行動パターンが多様に変化することが示された。

属性変換モデルでは、ロボットは周囲の物体を観測し、自分の注意にもとづいて行動変数を決める。ただしこの注意は、物体の色や大きさといった属性のうち特定の属性だけにもとづいて決定される。しかも、注意のベースとなる属性が時間的に変化する。このため、最初は色にもとづいて判断していたものが大きさに基づいて変化する、といった過程を経ることで行動変数が様々に変化するモデルである。属性変換モデルでは、周囲環境に配置された物体の情報構造の複雑さに応じて複雑な行動パターンを生成することが示された。

提案するモデルはシミュレーション環境上で実装し、動作を検証した。その結果、しばらく特定の行動変数が維持されるが時間が経過すると行動変数が移りかわるという多様な行動パターンが生成されることが確認された。また、提案する行動パターンは単なるランダム選択とは異なる特異なパターンであることを確認した。さらに、モデルの特性を決定するパラメータを様々に変化させて実行することで、パラメータの変化がシミュレーション上での挙動にどのような影響を与えるかを調べた。

また提案するモデルのうち属性変換モデルについては実ロボット上で動作するデモシステムを設計し、実際にコミュニケーションロボット Robovie 上にも実装した。そして実ロボット上でも多様な反応を示すことを確認した。

本研究において取り上げた多様性の実現という目的は、これまでそれほど重要視されてこなかった。本研究の成果のひとつは、この概念に焦点をあて、明確化したことである。既存の研究でも動作パターンレベルでの多様性は試みられてきたが、行動パターンレベルでは考えられてこなかった。本研究は行動パターンにおける多様性を実現するものである。したがって、本研究の成果と既存のメカニズムを組み合わせることにより、ロボットの行動をより多様で豊かなものにできるであろう。

謝辞

本研究において端緒よりご指導を賜りました 慶應義塾大学理工学部准教授 今井 倫太先生 に心より感謝いたします。今井先生には本研究のテーマのみならず、研究そのものに取り組む姿勢や手法、論文執筆や对外発表の心構えなど、ありとあらゆる点で丁寧に教えていただきました。

研究室長として様々な点からご指導いただきました 慶應義塾大学理工学部教授 安西 祐一郎先生 に心から感謝いたします。理工学部長、塾長兼務の激務の中、安西先生は重要なアドバイスをくださりました。

本論文に対し、大変有益なご助言を下された 慶應義塾大学理工学部教授 天野 英晴先生 山本 喜一先生 岡 浩太郎先生 にお礼申し上げます。私の未熟な論文を丁寧に読み込んでいただき、重要な指摘をいただきました。本論文の内容を高めることができたのも、また本研究の内容についてより深い考察を得ることができたのも先生方のご指摘による所も多くございます。

安西・今井・大村研究室のメンバーである大澤 博隆氏、佐竹 聡氏には忙しい中、本論文の校正をお手伝い頂きましたことに心から感謝します。また、彼らを含む安西研のメンバーには、ミーティングや発表のたびに有益な意見を頂きました。改めて感謝いたします。

折に触れて御指導を賜りました 慶應義塾大学情報工学科准教授 山崎 信行先生 にお礼申し上げます。山崎先生には輪講を通じて計算機システムに関する勉強をさせて頂きました。

慶應義塾大学理工学部助教 大村 廉先生 にお礼申し上げます。研究室の先輩として、後に教員として様々な点でご指導いただきました。また研究室の計算機システム管理者として、私を鍛えていただきました。

筑波大学大学院システム情報工学研究科講師 川島 英之先生 にお礼申し上げます。川島先生には研究室の先輩として様々な点でご指導いただき、勉強させていただきました。

最後になりますが、私のことを支え続けてくれた父 利典、母 良子、兄 秀和に深く感謝申し上げます。長い研究期間を暖く見守ってくれた家族がいなければ本論文は決して完成しなかったことでしょう。

2007年8月
向井 淳

参考文献

- [Breazeal 02] Cynthia L. Breazeal. *Designing Sociable Robots*. MIT Press, 2002.
- [Breazeal et al. 99] Cynthia Breazeal, and Brian Scasselatti. How to build robots that make friends and influence people. In *Proceedings of 1999 IEEE/RSJ International Conference on Intelligent Robotics and Systems(IROS99)*, 1999.
- [Ekman et al. 94] P. Ekman, and R. Davidson. *The Natur of Emotion: Fundamental Questions*. Oxford University Press, New York, 1994.
- [Frijda 86] N.K. Frijda. *The Emotions*. Cambridge University Press, 1986.
- [Fujita et al. 97] M. Fujita, and K. Kageyama. An open architecture for robot entertainment. In *Proc. of Autonomous Agent '97*, pp. 435–442, 1997.
- [Gunji et al. 96] Yukio-Pegio Gunji, Hisato Sadaoka, and Keisuke Ito. Bootstrapping System Defined by Inconsistent Relation between Boolean and Non-Boolean Algebra. *Applied Mathematics and Computation*, Vol. 79, pp. 43–97, 1996.
- [Gunji et al. 97] Yukio-Pegio Gunji, and Shin'ichi Toyoda. Dynamically changing interface as a model of measurement in complex systems. *Physica D*, Vol. 101, pp. 27–54, 1997.
- [Ishiguro et al. 01] H. Ishiguro, T. Ono, M. Imai, T. Maeda, T Kanda, and R. Nakatsu. Robovie: an interactive humanoid robot. *Intl. J. of Industrial Robot*, Vol. 28, No. 6, pp. 498–503, 2001.
- [Kanda et al. 04] Takayuki Kanda, Hiroshi Ishiguro, Michita Imai, and Tetsuo Ono. Development and Evaluation of Interactive Humanoid Robots. *Proceedings of the IEEE*, Vol. 92, No. 11, pp. 1839–1850, 2004.
- [Kato et al. 04] Shohei Kato, Shingo Ohshiro, Hidenori Itoh, and Kenji Kimura. Development of a communication robot ifbot. In *Proceedings of The 2004 IEEE International Conference on Robotics and Automation (ICRA2004)*, pp. 697–702, 2004.
- [Kuroki et al. 03] Y. Kuroki, M. Fujita, T. Ishida, K. Nagasaka, and J. Yamaguchi. A small biped entertainment robot exploring attractive applications. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA 2003)*, pp. 471–476, Sep 2003.
- [Maslow 70] Abraham H. Maslow. *Motivation and Personality*. Harper & Row, 1970. 邦訳: 『人間性の心理学』小口忠彦訳, 産能大学出版部 (1983).

- [Menzel 01] P. Menzel. Recent Videos of Kismet, 2001.
- [Miwa *et al.* 03a] Yasuhiro Miwa, Kazuko Ito, Daisuke Ito, Hideaki Takanobu, and Atsuo Takanishi. Introduction of the Need Model for Humanoid Robots to Generate Active Behavior. In *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robotics and Systems*, pp. 1400–1406, Oct 2003.
- [Miwa *et al.* 03b] Yasuhiro Miwa, Tetsuya Ohguchi, Kazuko Ito, Hideaki Takanobu, and Atsuo Takanishi. A New Mental Model for Humanoid Robots for Human Friendly Communication - Introduction of Learning System, Mood Vector and Second Order Equations of Emotion -. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, pp. 3588–3593, Sep 2003.
- [Miwa *et al.* 04] Hiroyasu Miwa, Kazuko Ito, Munemichi Matsumoto, Massimiliano Zecca, Hideaki Takanobu, Stefano Rocco, Maria Chiara Carrozza, Paolo Dario, and Atsuo Takanishi. Effective Emotional Expressions with Emotion Expression Humanoid Robot WE-4RII. In *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robot and Systems*, pp. pp.2203–2208, 2004.
- [Murphy *et al.* 02] Robin Roberson Murphy, Christine Laetitia Lisetti, Russel Tardif, Liam Irish, and Aaron Gage. Emotion-Based Control of Cooperating Heterogeneous Mobile Robots. *IEEE trans. on Robotics and Automation*, Vol. 18, No. 5, pp. 744–757, 2002.
- [NEC 01] NEC. PaPeRo, 2001. NEC Personal Robot Center, <http://www.incx.nec.co.jp/robot/>.
- [Russell *et al.* 85] James A. Russell, and Merry Block. Multidimensional Scaling of Emotional Facial Expressions: Similarity From Preschool to Adults. *Journal of Personality and Social Psychology*, Vol. 48, pp. 1290–1298, 1985.
- [Sawada *et al.* 04] Tsutomu Sawada, Tsuyoshi Takagi, and Masahiro Fujita. Behavior Selection and Motion Modulation in Emotionally Grounded Architecture for QRIO SDR-4X II. In *Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robotics and Systems (IROS2004)*, pp. 2514–2519, 10 2004.
- [Shibata *et al.* 01a] T. Shibata, T. Mitsui, K. Wada, A. Touda, T. Kumasaka, K. Tagami, and K. Tanie. Mental commit robot and its application to the therapy of children. In *Proceedings of 2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Vol. 2, pp. 1053–1058, Jul 2001.
- [Shibata *et al.* 01b] T. Shibata, and K. Tanie. Physical and affective interaction between human and mental commit robot. In *Proceedings of IEEE International Conference on Robotics and Automation 2001 (ICRA2001)*, Vol. 3, pp. 2572–2577, 2001.
- [Smith *et al.* 85] Craig A. Smith, and Phoebe C. Ellsworth. Patterns of Cognitive Appraisal in Emotion. *Journal of Personality and Social Psychology*, Vol. 48, pp. 813–838, 1985.

- [SONY 99] SONY. AIBO, 1999. <http://www.sony.jp/products/Consumer/aibo/>.
- [Tanaka *et al.* 05] F. Tanaka, B. Fortenberry, K. Aisaka, and J. R. Movellan. Plans for Developing Real-time Dance Interaction between QRIO and Toddlers in a Classroom Environment. In *Proceedings of the 4th International Conference on Development and Learning*, pp. 142–147, Jul 2005.
- [Wada *et al.* 04] K. Wada, T. Shibata, T. Saito, and T. Tanie. Psychological and Social Effects in Long-Term Experiment of Robot Assisted Activity to Elderly People at a Health Service Facility for the Aged. In *Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robotics and Systems (IROS2004)*, pp. 3068–3073, 2004.
- [Wada *et al.* 05] K. Wada, T. Shibata, T. Saito, K. Sakamoto, and K. Tanie. Psychological and Social Effects of One Year Robot Assisted Activity on Elderly People at a Health Service Facility for the Aged. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA2005)*, pp. 2785–2790, Apr 2005.
- [黒木 03] 黒木義博. 高度な運動能力を有する小型二足歩行エンタテインメントロボット SDR-4X. 映像情報メディア学会誌: 映像情報メディア, Vol. 57, No. 1, pp. 71–74, 2003.
- [三輪 04] 三輪洋靖. ヒューマノイドロボットの心理モデルに関する研究. 博士論文, 早稲田大学, 2004.
- [山田 他 02] 山田誠二, 山口智浩. 人間とペットロボットの相互適応: AIBO をしつける. 情報処理学会研究報告 知能と複雑系 (ICS), Vol. 130, pp. 93–98, 2002.
- [柴田 99] 柴田崇徳. 人の心を癒すメンタルコミットロボット. 日本ロボット学会誌, Vol. 17, No. 7, pp. 946–948, 1999.
- [森 他 02] 森善一, 太田浩司, 中村達也. 人とのインタラクションによるロボットの行動生成アルゴリズム. ロボティクス・メカトロニクス講演会, p. 56, 2002.
- [神田 他 02] 神田崇行, 石黒浩, 小野哲雄, 今井倫太, 中津良平. 研究用プラットフォームとしての日常活動型ロボット “Robovie” の開発. 信学論, Vol. J85-D-I, No. 7, pp. 691–700, 2002.
- [人間共存型ロボット研究専門委員会 98] 人間共存型ロボット研究専門委員会. 人間共存型ロボットシステムにおける技術的課題. 日本ロボット学会誌, Vol. 16, No. 3, pp. 288–294, 1998.
- [太田 03] 太田宏之. エンタテインメントロボットのための行動進化ソフトウェアの開発, 2003. 平成 15 年度未踏ソフトウェア創造事業 (未踏ユース), <http://www.ipa.go.jp/jinzai/esp/15youth/mdata/99-16.html>.

[藤田 98] 藤田雅博. ロボットエンタテインメント: 小型4脚自律ロボット. 日本ロボット学会誌, Vol. 16, No. 3, pp. 313–314, 1998.

[本田 04] 本田喜久. 人-ロボットコミュニケーションにおける相互適応メカニズムの実現. 修士論文, 慶應義塾大学, Mar 2004.

論文目録

【主論文に関する公刊論文】

1. Jun Mukai and Michita Imai, “Variety of Robot Behaviors for Human-Robot Interactions Using Attention”, In International Journal of Intelligent Computing in Medical Sciences and Image Processing, 2007 (to be published)
2. 向井 淳, 今井 倫太, 安西祐一郎, 『観測指向モデルによるロボットの自発的な行動基準の生成』, 知能と情報 日本知能情報ファジィ学会誌, Vol. 17, No. 3, pp.314–324, Jun 2005

【国際会議発表】

3. Jun Mukai and Michita Imai, “Maintenance and Drift of Attention in Human-Robot Communication”, In Proceedings of The First International Workshop on Multi-Agent Robotic Systems (MARS 2005), pp.15–22, Oct 2005
4. Jun Mukai and Michita Imai and Yuichiro Anzai, “Indefiniteness of Observation for Spontaneous Generation of Robots’ Behaviors”, in Proceedings of SCIS & ISIS 2004, CD-ROM Proceedings (TUP-3-1), Sep 2004
5. Jun Mukai and Yuichiro Anzai and Michita Imai, “Indefinite Observation for Spontaneous Trend Generation”, in Proceedings of Agent-Based Technology and Systems(ATS 2003), 221–232, Aug 2003

【国内学会発表】

6. 向井 淳, 今井倫太, 『観測指向モデルによる自律エージェントの自発的な行動選択』, 情報処理学会研究報告 [知能と複雑系] 2007-ICS-148, pp.9–14, Jul 2007
7. 向井 淳, 今井倫太, 『自律エージェントの実現に向けた, 観測と行為に依存した判断基準の獲得と遷移に関する分析』, 日本認知科学会第 23 回大会, pp.242–243, Aug 2006

8. 向井 淳, 今井倫太, 『注意機構にもとづくロボットの行動選択手法の提案』, 人工知能学会第20回全国大会, CD-ROM Proceedings (1G1-1), Jun 2006
9. 向井 淳, 自発的な注意機構を持つロボットの設計, 第6回 AI 若手の会 MYCOM 2005, pp.50–53, Jun 2005
10. 向井 淳, 今井 倫太, 属性変換を利用したコミュニケーションロボットの実現, SI 部門講演会 (SI2004) 予稿集, CD-ROM proceeding (3E4-2), Dec 2004
11. 向井 淳, 今井 倫太, 属性変換による「好み」の生成と変遷, 合同エージェントワークショップ&シンポジウム (JAWS 2004), pp.259–265, Oct 2004
12. 向井 淳, 今井 倫太, 好みの自発的生成: 不定な観測による対話相手の自律的な選択手法の提案, 情報処理学会研究報告 [知能と複雑系] 2002-ICS-130, pp.87–92, Nov 2002

【 同一著者による他の研究成果 】

13. 向井 淳, 『入門 Haskell』, 毎日コミュニケーションズ, ISBN: 4839919623, Mar 2006
14. H. Osawa and J. Mukai and M. Imai, “Acquisition of Body Image by Anthropomorphization Framework”, Joint 3rd International Conference on Soft Computing and Intelligent Systems and 7th International Symposium on advanced Intelligent Systems(SCIS&ISIS2006), pp.1460-1465, Sep 2006
15. H. Osawa and J. Mukai and M. Imai, “Anthropomorphization of An Object by Displaying Robot”, IEEE International Symposium on Robot and Human Interactive Communication(RO-MAN 2006), Hatfield, United Kingdom, pp.763–768, Sep 2006
16. H. Osawa and J. Mukai and M. Imai, “Acquisition of Subjective Representation Using Body Parts by an Object”, IEEE Workshop on International Conference on Cognitive Science 2006 (ICCS 2006), pp.173–174, Vancouver, Canada, Jul 2006
17. 北澤雄司, 向井 淳, 今井 倫太, “ロボット自身の行動に起因するセンサ情報を識別する機構”, 情報処理学会第69回全国大会, pp.2-469–470, Mar 2007
18. 大澤 博隆, 向井 淳, 今井 倫太, “擬人化した物体によるインタラクション手法の提案”, インタラクション 2007, CD-ROM Proceeding Mar 2007
19. 大澤 博隆, 向井 淳, 今井 倫太, “物体を擬人化するディスプレイロボットの開発”, 第48回プログラミングシンポジウム, pp.9–16, Jan 2007
20. 大澤 博隆, 向井 淳, 今井 倫太, “身体イメージを想起させる擬人化ディスプレイロボット”, HAI シンポジウム 2006, CD-ROM Proceedings (2OS-A-1), Dec 2006

21. 古山真之, 向井 淳, 今井倫太, “セマンティック・センサネットワーク情報の可視化ソフトウェア Viewlon2” 合同エージェントワークショップ&シンポジウム 2006(JAWS 2006), CD-ROM Proceeding, Oct 2006
22. 古山真之, 向井 淳, 今井倫太, 『Viewlon によるセマンティック・センサネットワーク情報の可視化』, 人工知能学会第 20 回全国大会, CD-ROM Proceeding (1G1-6), Jun 2006
23. 大澤博隆, 向井 淳, 今井倫太, 『ディスプレイロボットを利用した物体の擬人化』, 人工知能学会第 20 回全国大会, CD-ROM Proceeding (3F3-2), Jun 2006
24. 古山真之, 今井倫太, 向井 淳, 『セマンティックセンサネットワークにおける論理関係可視化システム Viewlon の設計』, 情報処理学会第 68 回全国大会, pp.4-139-140, Mar 2006
25. 大澤 博隆, 向井 淳, 今井 倫太, 『ロボットの発話・身体動作からの制約に依存した物体からのセンサデータ解釈手法の設計』, 合同エージェントワークショップ&シンポジウム (JAWS 2005), pp. 576-583, Nov 2005
26. 大澤 博隆, 向井 淳, 今井 倫太, 『SOBAR: コミュニケーションロボットによる環境センサ情報への制約』, 人工知能学会第 19 回全国大会, CD-ROM Proceeding (3B2-3), Jun 2005
27. 大澤博隆, 向井 淳, 今井倫太, 『ロボットの発話を利用した環境センサ情報への意味付け』, 情報処理学会第 67 回全国大会, 2-245-246, Mar 2005
28. 岸本有玄, 向井 淳, 今井倫太, 『人型ロボットによるスライドプレゼンテーションに関する研究』, 情報処理学会第 67 回全国大会, pp.4-157-158, Mar 2005
29. SCIM-SKK. オープンソースの日本語インプットメソッドエンジン実装.
<http://sourceforge.jp/projects/scim-imengine/>
30. HaskellNet. Haskell で書かれたネットワークライブラリ. Google Summer of Code 2006 採択プロジェクト <http://darcs.haskell.org/SoC/HaskellNet/>