

2010年度卒業論文

山田正雄ゼミナール

エージェントの将来性
—人間を助けてくれる知的なエージェント—

日本大学法学部 政治経済学科 4年

学籍番号：0720271

星 翔太

はじめに

近年、インターネットによる情報量の増加によって再び注目されるようになったエージェントといったものが存在する。本論は、それがどういったものであるのか分析し、現在の使用例を検証することによって、エージェントが私たちの生活の中で、どのような利便性を与えてくれるのか考察していくことを目的としている。特に人間の行為を理解し、支援してくれることが可能である知的で自律的な活動ができるエージェントについて研究している。

エージェントという概念を表すのはきわめて困難である。なぜならば、それは多種多様な領域で存在し、個々のエージェントは、その性質や役割によって形を変化させるためである。しかし、ここで研究するエージェントの必要概念をあえて表すならば、「知性」と「自律性」を備えているものがエージェントだと考えられる。

人間の行為を理解し、自らの知識を使って考え、自律的な行動を起こすエージェントを研究する上で人工知能がどのように誕生し、どのような進化を辿ってきたのか、その歴史的背景を知ることは重要である。それはエージェントが人工知能の発達によって、生まれた技術だからである。そのため、本論では人工知能の分野にも深く触れている。しかし、エージェントはあくまでもソフトウェアであり、与えられた問題を処理するプログラムである。しかし、人工知能のように機械が人間のように考え、行動するようになり、人間的振る舞いをする事ができれば、人間のパートナーとして最適なものになるに違いないと私は考えている。

エージェントが再注目されるようになった理由の一つとして、インターネットが広く普及されるようになり、インターネット上にある莫大な情報を人間の力だけでは処理しきれなくなったためである。そのため、そのような人間の手では作業が難しくなってきたものに関して、エージェントが人間の代わりに処理を代行するようになってきたのである。今では、インターネット上にあるアプリケーションにおいては、AI システムが一般的になっている。また、その他の理由としては、1990年代に入ってから、知能に関する理論的基礎の理解が進んだことと実働システムの能力が改善したことなどによるAIの部分問題の解決の進展によるものだろうと考えられている。

本論は、全6章から構成されており、1章で歴史的背景について人工知能の誕生から見ていき、エージェントがどのように誕生したのか見ていき、2章ではエージェントがどういったものであるかを定義し、分類する。3章ではその特徴の理解を進め、我々にとってどんな利便性、問題点があるかを分析し、4章では現在の実用例を取り上げ、実際のエージェントの活躍を知ってもらうことを目的としている。5章は我々の仕事を支援する上で最適なエージェントとは何かということを含め、今までのものを踏まえつつ、考察していく。最後に6章としては、今後エージェントがどのように変化していくのか、AI技術は何処まで進歩し続けられるのかなど、以上のことについて言及しつつ、今後の展望を述べることとする。

- 目次 -

はじめに

1 人工知能の歴史

- 1.1 人工知能の誕生と課題
- 1.2 人工知能の発展
- 1.3 知的エージェントの出現

2 エージェントとは何か？

- 2.1 エージェントの定義
- 2.2 エージェントの性質
- 2.3 エージェントの分類
 - 2.3.1 ソフトウェアエージェント
 - 2.3.2 マルチエージェントシステム
 - 2.3.3 モバイルエージェント
 - 2.3.4 インターフェイスエージェント
- 2.4 知的エージェントとは？
 - 2.4.1 学習による知性の獲得
 - 2.4.2 知識と学習に基づく推論
 - 2.4.3 すべての知的エージェントが有する原理

3 エージェントとの内部構造

- 3.1 エージェントの構造
 - 3.1.1 エージェントの判断基準
 - 3.1.2 エージェントと環境
- 3.2 エージェントの利便性
- 3.3 エージェントの問題点

4 エージェントの実用例

- 4.1 自律的プランニングとスケジューリング
- 4.2 ゲームプレイ
- 4.3 自動制御
- 4.4 診断
- 4.5 戦略立案
- 4.6 ロボット工学
- 4.7 言語理解と問題解決

5 最適なエージェントとは？

- 5.1 機械は本当に知的に振る舞うのか？
 - 5.1.1 弱い AI

5.1.2 強いAI

5.2 最適な知的エージェント

6 今後の展望

おわりに

1 人工知能の歴史

1章では、人工知能の歴史について見ていき、それを学習することでエージェントについての認識を深めることを目的としている。エージェントはもともと人工知能から派生した技術であり、その歴史的背景を知ることによって、エージェントを研究する意義と目的の理解に繋がる。ここでは歴史的背景の他に現在のAI技術の基礎となった技術やエキスパートシステムなどにも触れつつ、どのように人工知能が技術的に発展して、知的エージェントが誕生したのかを解説していく。

1.1 人工知能の誕生と課題

1943年、現在のAI技術の最初となる人工的ニューロン¹のモデルがWarren McCullochとWalter Pittsによって提案された。ニューロンとは神経細胞のことであり、情報の伝達や情報の処理機能を有している。彼らは、この機能を心理学の基本と脳のニューロン機能に関する知識、RussellとWhiteheadの命題論理の分析、Turingによる計算の論理をもとに人工的なニューロンを構築した。これによって、すべての計算可能な関数は適切に結合されたニューロンのネットワークで計算できること、すべての論理結合子は単純なネットワーク構造で実現できるかもしれないことを示し、適切に定義されたネットワークは学習できるかもしれないことを示唆した。その後、1950年にはチューリングテスト²で知られているAlan Turingが”Computing Machinery and Intelligence”という論文で完全なAIを最初に提唱した。しかし、この時にはまだAIという言葉は完全に定着したわけではなかった。

AIが公式に誕生したのは、John McCarthyの説得によって、1956年の夏に開催されたダートマスのワークショップで、Artificial intelligence : AIという言葉が正式に採用された。このワークショップはオートマン理論、ニューラルネット³、知能研究などに興味がある合衆国の研究者を集めたが、新たなブレイクスルーを生み出すまでには至らなかった。

1950年代、1960年代の初期のAIは成功の連続であった。この時代によってゲームプレイ、数理的な定理証明、常識論、数学問題などの限定的な問題に対して人間に似た知的活動（可能性の考慮）が可能であることがわかった。例えば、GPSは自らが持っている情報とセンサーによって得られる現実世界の情報を使うことによって、人間のように考えて限定的な問題に対して解を作成する事が出来るプログラムである。

1985年には現在でも使われている二番目に古い主要高級言語であるLispがJohn McCarthyによって定義された。また同年にはAdvice Takerという最初の完全なAIシステムと呼べるものが作成された。このプログラムは、実世界の一般常識を持ち、ある問題に対して仮定を立て、プランを生成することができる。また、プログラムの動作中に新しい仮定を立てることによって再プログラムなしに新しい分野においても力を発揮することができるようにできている。これはAIが実世界の状態を把握し、表現することによって実世界に影響を与えられることを示している。

このような成功のため、初期の AI 研究者たちは将来の成功を大胆に予想していた。しかしながら、初期の AI は広範囲あるいは困難な問題に対しては無力であることがわかった。Stuart Russell & Peter Norvig によると三つの問題が存在している (Stuart Russell & Peter Norvig 2008)。

一つ目の問題は、初期の AI の多くが対象に関する知識をほとんど持たず、単純な構文を使って成功した点である。典型的な例として、初期の翻訳機にみられる。これは 1957 年にロシア語の科学論文の翻訳を加速する意図で研究されたのだが、当初はロシア語と英語に基づいた単純な構文構成と、電子辞書を使った単語置き換えだけで正確な意味ができると考えられていた。しかし、実際には対象に関する現実の知識が必要であり、これがないと言葉の曖昧性を解決して文の内容を確定することができないことがわかった。

二つ目の問題として、AI が解こうとしていた問題の計算至難性にある。初期の AI プログラムのほとんどは問題を解くために基本的な手続きの様々な組み合わせを正解が出るまで続ける方式で動いていた。しかし、この方法では問題が大きくなった場合、または複雑になった場合では解を発見するのが困難になる。また、このような問題は初期の機会進化の実験 (現在は遺伝的アルゴリズムと呼ばれている) にでもみられた。機会進化実験とは、プログラミングに対し小さな異変を適切な順序で繰り返すことにより、任意の小さな仕事に対する効率の良いプログラムを生成できるというものであった。これは突然変異をランダムに与え、行動を改善するように見える変異だけを残す選択プロセスをするという考えに基づいて構築された。しかし、何千時間をかけて計算したにもかかわらず、変化がみられなかった。

最後、三つ目の問題点として、知的行動を生み出すために用いられた基本構造の原理的な限界に起因する。これは 1969 年に Minsky と Papert の Perceptrons によって Perceptrons は表現可能なものはすべて学習できるが、表現可能なものが限られていることを証明したのである。

初期の AI の成功は、狭い分野の限定的な問題でのみの成功であった。しかし、機械が知的な活動を行うことは当時の技術からしてみれば革新的なことであり、将来に大きな期待を抱かせるものであったことは間違いなかった。

1.2 人工知能の発展

AI の期間の 1969 年から 1979 年の 10 年間における問題解決の図式は要素となる推論ステップをつなぎ合わせて完全な解を得ようとする汎用探索機構であった。このようなアプローチは弱解法と呼ばれている。しかし、汎用であるが巨大な問題や困難な問題に対しては対応できなかった。そのため、困難な問題を解決するには、あらかじめその答えを知っている必要がある。つまり、ある領域の膨大な知識量を使って、飛躍的に推論ステップを大きくする必要がある。このようなアプローチでエキスパートシステム⁴というものが存在する。エキスパートシステムとは、特定の分野に特化した専門知識データベースを元に推

論を行い、その分野の専門家に近い判断を下すことが出来る AI システムである。このシステムは、ある分野の専門家が自らの作業の補助としてエキスパートシステムを利用し、最終的な判断を専門家自身が行うという分担作業がなされている。例えば、医療診断プログラムで血液感染症診断システムというものが存在する。このプログラムは、血液感染症を判断するために 450 の規則を持っており、それは一部の専門家と同等の成績を示すことができた。このプログラムの知識データは、専門家からの長時間におよぶインタビューによって獲得されたものであった。さらに医療知識の不確実性をシステムに反映させるために確信度係数という不確実性の計算が導入されている。AI に自然言語を理解させるのにも対象領域の知識の重要性は明らかであった。自然言語を理解するには世界に関する一般知識とそれを扱う一般的な手法が必要であった。この期間を通して、AI には世界や取り扱う対象を理解するための多くの知識が必要であり、それを使うことによって人間に近い推論が可能となることを示唆したのである。

さらに 1980 年代に入ると AI は更なる発展を遂げる。商用エキスパートシステムが開発され、AI が産業そして扱われるようになってきた。例えば、1982 年に R1 というプログラムは、新しい計算機システム注文の構成を助言するもので、1986 年までに年間推定 4 千万ドルの経費節減に貢献するようになった。このように 1980 年代、AI 産業は数十億ドルという売上を出していた。しかし、その直後に AI の冬と呼ばれる時期がきて、多数の企業が苦しむこととなってしまった。

また、この時期には AI と関係が深いニューラルネットワークの研究が再燃した。コンピュータ科学におけるニューラルネットワークの研究は 1970 年代後半にはほぼ破棄されていたが、物理学や心理学などの他の分野では研究が続いていた。例えば、John Hopfield のような物理学者がノードの集合を原子の集合のように扱う統計力学の手法を用い、ネットワークの情報蓄積能力や最適特性を分析し、David Rumelhart や Geoff Hinton ら心理学者は記憶のニューロンモデルの研究を続けていた。1980 年代半ばには少なくとも 4 つの違うグループから逆伝播学習アルゴリズムが再発明されたのである。このアルゴリズムは、コンピュータ科学や心理学の多くの学習問題に広く適用されていった。

近年、AI 科学的手法の一員となっている。それは存在するもの上に理論を積み上げ、厳密な理論や確固とし実験事実を基礎とし、実世界の問題を対象とするのが一般的になった。仮説には厳密な経験的実験によって検証されなければいけないし、結果が重要視されるために統計的分析が必要だった。このような問題はインターネットの利用、テストデータやプログラム資源の共有により、実験の再現が可能となったため、解決された。この結果、音声認識研究では隠れマルコフモデル⁵によって取り組みが進んでいる。それは隠れマルコフモデルが確固とした数学モデルに基づいており、他の分野の数十年にわたる研究データを利用することができたこと、実際の音声データの大コーパスを用いた訓練で生成されることによって非常に丈夫な性能が得られた。また、ニューラルネットも理論的枠組みと検証されたデータを用いる方法論を用いることによって、統計、パターン認識、機械学

習の各分野に対応する手法と比較でき、各アプリケーションにとって最適のものを使うことができるということを認識されるようになった。その結果として、データマイニング⁶とよばれる技術が活力ある新しい産業を生み出した。

AI 研究の手法が科学的になった最大の結果として、ベイジアンネットワーク⁷というものが誕生した。これは、不確実な知識の効率のよい表現と、それを使った厳密な理論の枠組みと発明された。このベイジアンネットワークは 1960 年代から 1970 年代の確立推論システム問題点と大部分を克服し、不確実推論やエキスパートシステムに関する AI 研究の支配的手法となった。これは経験からの学習を可能にし、古典的 AI⁸ とニューラルネットワークの長所を合わせたものとなっている。

1.3 知的エージェントの出現

1990 年代に入ると状況依存運動が、実環境に埋め込まれ、連続した感覚情報を得ているエージェントの動きの理解を目指し始めたのである。つまり、実際の環境状況で起こる問題を知覚し、情報を得ることによって、知的で合理的な活動をするエージェントを目指すようになってきた。知的エージェントで最も重要な環境の一つがインターネットである。ウェブにおけるアプリケーションにおいては AI システムが一般的になり、bot、検索エンジン、推奨システム、ウェブサイト構築システムなどのインターネットツールに使われている。このように AI 技術は、インターネットの登場によって、より陽の目を浴びることとなった。これは知的エージェントが、Web の情報を探索および発見し、フィルタリングし、利用者が本当に興味のあることへ導くなどの便利な個人的なアシスタントとしての役割を期待されるようになったためである。

Stuart Russell & Peter Norvig によると周囲の状況を理解し、人間の行為を理解して動く知的で合理的なエージェント、そのような完全なエージェントを作るためには AI のこれまで隔離されていた部分領域の結果を再構成して考えなければならない。特に感覚システム(視覚、ソナー、音声認識など)が環境に関する信頼できる情報を届けられないことは広く知られている。そのため、推論と計画立案システムは AI のように不確実性を扱える必要がある。また、制御理論や経済学といった同じくエージェントを扱う分野とのより緊密な関係が AI に求められるようになったからである(Stuart Russell & Peter Norvig 2008)。

AI の研究は失敗と挫折の連続である。しかし、その中でも確実に研究者たちは人間の行為を理解し、AI を作るために一步一步ずつ近づいていっている。

[注]

1) 人工的ニューロン：我々の脳にある細胞で、電気信号を集めて、処理し、伝達をする情報処理能力を人工的モデルに変化したものである。ニューロンの基本的なモデルはいくつかの入力から成り立っている。シナプスの重量、ベクトル、活発化機能、伝達関数、出力を定めるニューロンの中などである。これは人工的なニューロンで使用される基本的な構

造である。

2) チューリングテスト : **Alan Turing** によって発案されたもので、知能に関する実行可能な形の満足できる定義を与えることを目指したものである。テストの内容は、人間の質問者が書面による質問を提示し、書面による返答を受けるということを繰り返したあとで、それらが人間からきたかコンピュータからきたものか判断し、区別できなかつたら合格というものである。

3) ニューラルネット : 人間の脳の情報処理を工学的にモデル化したネットワークのことである。ニューラルネットワークは複数のノード、あるいはユニットとよばれる素子からなり、それらは方向性のあるリンクによって結合される。この研究は新しい並列処理方法を研究する試みである。パターン認識や信号処理、最適化問題、連想メモリーなどに応用することを目指している。別名、コネクショニズムと呼ばれている。

4) エキスパートシステム : 特定分野に特化した専門知識データベースを元に推論を行い、その分野の専門家に近い判断をくだすことができる人工知能システムのことである。

5) 隠れマルコフモデル : 不確定な時系列のデータをモデル化するための有効な統計的手法である。出力記号によって一意に状態遷移先が決まらないという意味での非決定性確率有限オートマトンとして定義される。出力記号系列が与えられても状態遷移系列は唯一に決まらないのである。観測できるのは記号系列だけであることから隠れマルコフモデルと呼ばれている。

6) データマイニング : 大量に蓄積されるデータを解析し、その中に潜む項目間の相関関係やパターンなどを探し出し、新たな知識を発見することである。

7) ベイジアンネット : 18世紀の数学者であった **Thomas Bayes** が発見した「ベイズの定理」をベースに構築されています。人間の行動予測、異なる人格間のマッチングの最適化、多種多様な要因が複雑に絡み合っている事象の将来予測など、不確実で絶えず変化するために数式で表現が困難なものを、ベイズの定理を利用するし、アルゴリズムによって、「確率付ノード」と「ノード間の依存関係」で表現し、確率ネットワークの形態でモデル化したものです。

8) 古典的 AI : 記号操作に基づいて情報処理を行うものである。

2 エージェントとは何か

2章では、本論で扱うエージェントがどういったものであるのかという定義づけを行い、多種多様なエージェントの分類をすることを目的とする。しかし、エージェントは対象とする目的によって、属性や性質が異なっており、完全に分類するのは困難である。そのため、今回は今後、我々の身近で活躍しそうなもののみ焦点をあてることとする。また、ここでは本論で何故、知的なエージェントを取り扱っているのかを明言し、またエージェントが知的であるといったことがどういったことであるのかということについても考察していく。

2.1 エージェントの定義

エージェントを考えるためには、それがどういったものであるか知らなければならない。しかし、現在のエージェントに明確な定義づけはなされていない。そのため、多くの研究者の間で意見が食い違うこともある。しかし、共通して言えることは、ユーザーが何らかの目的を達成するために与えられた知識によって、エージェントが知性を持ち、その知識を使って自ら自律的に考えて行動するようなものの事を呼んでいる。また、エージェントが情報収集のためにネットワーク間を移動し、または人間や他のエージェントと協力して活動するような特徴も含まれる場合もあるが、このような特徴がない場合のエージェントもあるため、必ずしもこれらの性質を備えているとはいえない。このような性質に関しては後述することとする。

エージェントを一言で表すならば、『行為の代行者』である。なぜなら、エージェントが人間の仕事を手伝ったり、人間に代わって仕事をしたり、一人の人間では処理しきれない多くのことを代行してくれるためである。例えば、インフォメーションエージェントというものがある。これのエージェントは、インターネット上に行き、ユーザーの興味がある情報を探し出してくれたり、電子メールのやり取りを代わりに引き受けてくれたり、ニュースグループへの投稿などから入ってくる情報ストリームのフィルタリングをしてくれたりする。

このようにエージェントはユーザーが仕事を達成することを助けようとしてくれるソフトウェアであり、全てのエージェントは何らかの知的活動を行っている。しかし、それは人間から与えられた情報と命令をもとに人間のマネをしているだけでは本当に知的なエージェントとは呼ばないと私は考えている。知的なエージェントは、設計者に与えられた情報を世界の状況などを感じることによって活動し、その結果を学習し、成長していくものこそが知的なエージェントであると考えている。これについても後述する。

2.2 エージェントの性質

次にエージェントがどのような能力を持っているか考えていく。エージェントを作成するとき、ユーザーは自らの目的別に設計をする。そのため、エージェントは多種多様なも

のが存在し、エージェント別にその性質が異なってくる。先ほどからも述べられているようにエージェントには基本的に知性と自律性を備えている。この性質に加え、ユーザーの目的別に人間や他のエージェントとコミュニケーションを取る協調性やある規則の中で組織的に動くことができる社会性（通信性）、指示を与えられなくても自ら活動する自発性、周囲の状況に応じて行動を開始する反応性、個人の趣味・嗜好を学んでパーソナライズ化されていく個人適応性、周囲の変化する環境状況を読み取り、行動する環境適応性、目的達成のために他のコンピュータへと移動する移動性、イベントやアクションがあれば、永続的に活動する永続性などの様々な要素が付け加えられていく。それによって、エージェントはソフトウェアエージェント、マルチエージェント、モバイルエージェント、インターフェイスエージェントと分類されていく。これらについては、次の章にて解説する。

2.3 エージェントの分類

ここから、上記の要素によって分類された各エージェントについてみていく。ここで紹介するのはソフトウェアエージェント、マルチエージェント、モバイルエージェント、インターフェイスエージェントの4つである。

ここで注意していただきたいのがここであげるもの以外にも多種多様なエージェントが存在するということである。その中でも、この4つをあげる理由としては、現在、我々の身近の様々なところで活躍しており、もっとも理解しやすいものだと考えたためである。

2.3.1 ソフトウェアエージェント

ソフトウェアエージェントとは、ソフトウェア環境において情報の探索や処理をユーザーの代わりに活動するプログラムのことである。ソフトウェア環境とは、オペレーティングシステム、コンピュータアプリケーション、データベース、ネットワーク、または、様々なドメインのことである。このようなプログラムが注目されるようになったのは、ネットワークの分散計算の複雑性の緩和とユーザーインタフェースの利便性向上の可能性があったためである。

ソフトウェアエージェントの特徴としては、ユーザーの目的を達成するために対象範囲の知性と自律性、永続性、社会性を持っている。つまり、ソフトウェアエージェントは、目的を達成するために自ら持っている知識を使って行動し、時には他のエージェントと通信して、必要な情報を取ってくるといった行動を自ら考えて行うのである。このような技術はいくつか実際に使われている。

例えば、オンラインショッピングの値段を比べてくれる Shop bot、ワシントン大学で開発された Web 検索エンジンである Husky Search Web、サーフィンを手伝う Letizia といったものがある。

また、ソフトウェアエージェントの対義語として、ロボット工学などのエージェント技術をハードウェアエージェントというものもある。また、これから述べるいくつかのエージェント

エージェントは、ソフトウェアエージェントから派生した概念である。

2.3.2 マルチエージェントシステム

マルチエージェントシステムを定義づけるとすれば、相互作用によっていくつかの問題に分割された問題を複数のエージェントが分担して解決するというエージェント群のことである。それらはエージェント間でコミュニケーションをとるため、若干の社会性を持っている。また、マルチエージェントシステムは、各自の処理結果、つまり、情報を持ち寄って、互いに情報を共有して問題を解決していくことも可能なのです。

特徴としては協調性や社会性（通信性）などの性質が特化することによって、分散構成の高いスケーラビリティ、システム間の柔軟な関係、システム自身の高い独立性があげられる。しかし、各エージェントが対象世界の状態について限定された視野しか持っていないこと、全体をコントロールするものがないといった特徴もある。また、マルチエージェントシステムを考えると時には、知識の共有の仕方とエージェント同士が会話（通信）するための手段も考えなければならない。エージェント同士が知識の共有、コミュニケーションをとるためには、語彙と意味を共有しなければならない。

それには様々な方法が存在する。古いもので言えば、黒板モデルと呼ばれるものが存在する。黒板モデルとは、黒板と呼ばれる複数の知識源に対する一般的な通信メカニズムとして使われるデータ構造を指し、コントローラーによって修理・調整される。各エージェントは各問題で自分が担当する部分で働き、他のエージェントによって与えられる新たな情報を拾い上げるために黒板を見ている。そして次に自分の情報を書き込んでいる。このように情報を共有することによって、エージェント同士が協調できるようになる。

また、その他にも KQML(Knowledge Query and Manipulation Language)といったプログラムとエージェントが情報や知識を交換する枠組みを提供するものもある。KQML は、エージェントが互いの知識ベースに対して試みるかもしれない操作を定義し、ファシリテータ¹と呼ぶ特別なエージェントを通じて、エージェントが知識や情報を共有する基本的アーキテクチャを与えてくれる。

このようなマルチエージェントシステムを使用している実例として有名なものでロボカップサッカー競技というものがある。これは、1997年日本の名古屋で開かれたIJCAI以来毎年開催されている。この競技会では様々な規則で開催されているが、基本的にはお互いに競い合う自律的エージェントのチームからなっており、チーム内では彼らは協調して動くようにプログラミングされている。

2.3.3 モバイルエージェント

モバイルエージェントとは、自律的にネットワーク上で、複数のコンピュータ間を移動することが可能なプログラムのことである。モバイルエージェントがコンピュータ間を移動するためには、実行を一時中断し、計算に必要なデータを移動先のコンピュータに転送

し、その実行を中断した時点から再開する必要がある。

モバイルエージェントの特徴としては、内部状態を保持したまま、移動することができる移動性、他のエージェントと通信することで様々なサービスを享受する事ができる社会性（通信性）を持っている。また、その他にも初期に得られた知識とエージェント自身が活動することによって得ていく情報によって、個人に特化していく個人適応性、エージェントを使用している人物や、近接する対象物や時間などといった環境に適応していく環境適応性などといった特徴もあげられる。

モバイルエージェントには、他のエージェントと同様にいくつかの利点が存在している。例えば、通信路が遮断された場合、モバイルエージェントならば、他の場所にエージェントが移動できるので通信路が遮断された場合でもエージェントは活動できる。また、近年、通信技術の進歩により通信速度は格段に速くなっているが、従来の通信形態を用いている限り、通信帯域の不足は生じる。この時、エージェントは自ら通信遅滞が小さい場所へ移動することによって、通信遅延を軽減させたりすることもできる。その他にも、ソフトウェアの保守管理の容易化、ソフトウェアの柔軟な運用、分散プログラミングの容易化、耐故障性の向上など様々な利点があげられる。

モバイルエージェントは、特に我々の身近な場所で多く使われている。病院では患者案内システムとして、博物館では展示支援システムとして使われている。また、我々が眼にすることは難しいが、インターネット上でも活躍している。電子メール、ネットワーク管理、分散情報処理、遠隔機器制御、電子商取引などである。例えば、電子商取引などでモバイルエージェントを用いることで、希望している商品を複数のオークションサイトから検索し、強調して複数のオークションサイトに入札を行うのである。エージェントは、より安く、確実に希望する財の入札が成功するように協調して機能している。このようにモバイルエージェントは様々な分野で応用されている。

2.3.4 インターフェイスエージェント

インターフェイスエージェントは、音声言語とキャラクターアニメーションとが統合されたユーザーインタフェースであり、自然言語や身振りや顔の表情を使った擬人化された表現を持つために、人のインタラクションにおいて様々な効果を生み出す可能性がある。

インターフェイスエージェントは、これまでの GUI² や直接操作とは異なり、自律性や個人適応性、環境適応性によって、人間の曖昧な要求や環境の変化に適応する事ができる。

ヒューマンインターフェイスは、人間同士の会話に近づけるように人間が意識的・無意識的に発信する情報を受け付け、人間にとって自然な形で伝達できなければならないのである。このような人間の自然なパートナー関係を目指したものをマルチモーダルインタフェースと呼んでいる。

マルチモーダルインタフェースという用語は心理学用語のモダリティに由来し、人間が情報伝達を行う際の様態、つまりは情報の知覚や伝達手段を指している。人間同士がコミ

コミュニケーションを行う際には音声による言語だけでなく、表情、身振り、手振り、視線などを同時並行的に用いて円滑な対話を実現している。この人間同士が無意識で行っているコミュニケーション手段をマシンインターフェイスに応用しようという発想がマルチモーダルインタフェイスである。

また、マルチモーダルインタフェイスに類似するものとしてノンバーバルインタフェイスというものがある。これは言葉によらない（ノンバーバルな）モダリティを利用したインタフェイスという意味である。表情、身振り、視線といった言葉によらないモダリティを積極的に採用したものである。従って、ノンバーバルインタフェイスはマルチモーダルインタフェイスの一種であるといえる。

マルチモーダルインタフェイスは、出力系、入力系、さらに時間的順序関係などによって分類できるが、より本質的なわけ方は、どのような目的のためにマルチモーダル化するかということである。

例えば、音声・言語情報を捕捉する目的として用いることもできる。これは、参照表現や指示動作を統合することによって、それらに存在する自然言語の曖昧性を排除しているインタフェイスである。または、音声認識や自然言語処理などは対話で行われる文脈情報を用いることで、言葉の探索範囲や曖昧性を制限する目的でも用いることができる。言語情報だけに頼って、どれが対象か判断するためには、他のモダリティを使って、あらかじめある程度まで文脈を絞り込まなければならない。これは音声認識技術が十分な語彙をカバーでき、自然言語処理技術が発話の様々な意味を許容できなければならないためである。その他にもシステムを擬人化してユーザーの認知的負担あるいは心理抵抗感の軽減を目的として用いることもある。

ここで注意して欲しいのは、擬人化されているもの全てがエージェントと呼べるわけではなく、ユーザーの発話などからその意図を認識し、適切な応答を生成するシステムとキャラクターの表情や動作などの擬人的表現が統合されているものこそを擬人化エージェントというのである。

2.4 知的なエージェントとは？

この章の最後ではどのような能力を持っていれば、そのエージェントが知的であるのかということについて考察していくのと同時にどういった要素がエージェントを知的にするのか考える。ここで気をつけていただきたいことはエージェントが全て知的であるかといえば、そうではないということである。では、知的なエージェントとは何であろうと考えたときに必要となってくるのは、知性や自律性の他に自身の身の回りや環境を認識できること、長期間持続的に活動が出来ること、問題や何かしらの変化が起きたときの対応の仕方、他者の目標の代行などがあげられる。さらに最も重要であるものが、これらの作業を行う過程で得られる経験を学習する能力などを備えているものこそが知的なエージェントと呼べると私は考える。エージェントは周囲の情報を収集しながら学習をすることで性能

が良くなる可能を秘めている。しかし、現在の多くのエージェントは、設計者に与えられた知識を用いるだけであり、自律性を欠いている。そのため、エージェントが学習によって成長することは非常に重要である。次から、知的なエージェントはどのような学習にて知性を獲得しているのかをみていく。

2.4.1 学習による知性の獲得

ここではエージェントがどのように学んでいるのかを見ていく。学習には多くの形態があり、まず、実例を与えられて、その実例をコピーし、その振る舞いを正確に再現する暗記学習、正確な決定や結果の見込みを改善するために、時間とともに要因の重み付けを調整する事ができるウェイトの調整、問題の重要な性格を抽出し、実例をより学習するプロセスであり、また、新しい状況での入力によって一般化することを許す帰納、最後にチャンキング、または知識の抽象化とよばれているものがある。これは、特定の例や状況から共通するパターンを検知し、新しい状況へと適合させる学習である。これは、多くの事例を抽象化することで必要とする記憶領域と検査、処理の時間を減らす効果がある。また、チャンキングの一種でクラスタリングというものがある。これは、多くの属性を持ったデータを見て、いくつかの基準に基づいて類似性について点数をつける。その結果、類似しているもの同士で同じクラスターのグループに分けられる。帰納とクラスタリングの学習などはデータマイニングツールとして使われる。データマイニングとは巨大なデータ蓄積から価値のある、明らかではない情報を抽出するプロセスのことである。つまり、新しい情報や知識を発見することがデータマイニングの役割である。そのため、知識発見とも呼ばれている。

エージェントの機械学習には、教師あり学習、教師なし学習、強化学習といったものがあり、多くの研究者、アプリケーション開発者は、これらの学習アプローチを複数とって一つのシステムに結合している。

教師あり学習とは、例によるプログラミングとも呼ばれており、望まれる出力やアクションと一緒に、問題の状態や属性の例を示すことで訓練される。この入力に基づいて予測を立て、もし出力が望まれる出力と異なる場合は、正しい出力を出せるようにエージェントは調整され、適応するのである。この教師ありアルゴリズムは逆伝播ニューラルネットワーク⁴と決定木⁵で実装されている。

教師なし学習とは、入力の中の類似性を認識し、入力データの特徴の識別をする必要がある場合に用いられ、データはエージェントに提示され、エージェントはそのデータに適応し、データをグループ分けすることができる。データがグループ分けされるまで、分類は何度も続くのである。教師なしアルゴリズムでは、データ中の重要な共通属性が抽出されるような、一種の特徴検出を実行する。

強化学習とは、訓練データ用の明確な入力と出力の組み合わせが利用できない場合に用いられる教師あり学習の一種である。強化とは、即時報酬のことであり、エージェントが

何かした際にその結果として得られた情報によって学習していくことである。連続した入力があり、望まれる出力は特定の順が現れたらあとにのみわかるという場合に使われる。この一連の入力値と後の出力値の間関係の識別プロセスは一時的な信用割合と呼ばれる。特定のエラー情報をあまり与えないので、強化学習はたいてい教師あり学習よりも時間がかかったり、効果が良くなかったりする。しかし、多くの状況で望まれる結果についての正確な情報を前もって持つことは不可能である。そのため、活動することによって即時報酬によって学ぶ強化学習がエージェントを学習する上で最も適切だと考える。強化学習アルゴリズムの例にQ学習がある。Q学習とは、エージェントが行動・価値関数、すなわち、Q関数を学習することを言う。行動・価値関数は、与えられた状態において与えられた行動をとった場合の期待を決定することである。

このようにエージェントは学習することによって、自らを成長させることができる。エージェントは性能を最大化することが期待されている。そのためには、将来の推論が可能となるように情報を収集し、探索をすることによって学習をする。それによってエージェントは成長することができる。

現在のエージェントは設計者に与えられた知識で、限られた範囲でしか学習できないのである。しかし、エージェントの研究は進歩してきており、人間と同程度考えられるようになるための学習方法が生まれてきた。これによって、エージェントは人間のような知的活動ができるようになるかもしれない。

2.4.2 知識と学習に基づく推論

次に学習に基づいてエージェントは推論を行う。知的なエージェントは、多様な目的に合わせて情報を繰り返し結合することにより、非常に一般的な形をした知識を活用できる。この一般的な知識と現在の知覚情報を結びつけて、行動を選択する前に背後に潜む情報を推論する。自然言語を理解するときなども隠れた情報、つまりは話し手などの意図を推論することが必要である。

宣言的知識表現を用いて推論をする場合、**if-then** ルールというものが使用されている。それらで作られたルールは独立した1つの知識として、あるいは知識ベース内の情報単位として扱われている。ルールでは前向き推論を用いて仮説を生み出し、後向き推論を用いて文の真偽を推論できる。このようなルールベースの処理システムを表すのに、エキスパートシステム、または、知識ベースシステムといった言葉が用いられるようになった。これらは、知識ベース (**if-then** ルールと既知の事実のセット)、作業記憶 (導き出された事実やデータのデータベース)、推論エンジン (ルールやデータを処理する論理ロジックを持つ) の三要素を持っている。これらの三要素によってエージェントは、推論を行うことができる。

推論によって実質的に無限の種類を持つ発話パターンを処理することができる。このようにほとんどの実世界では、目的を達成するためのいくつかの情報が入るが、残りの

情報は推論で得るしかない。このような不確実な環境の中でも推論ができるようにベイズの法則、または、ベイズの定理⁵と呼ばれる使い勝手の良い統計理論がある。ものごとの確立は0.0（チャンスなし）から1.0（確実）の確立の範囲までである。統計学者は確立を二種類の方法に分ける。最初に何かが起こるかもしれない機会を表す不確実な可能性（無条件確立）、または、証拠によって最新情報に入れ替え予測することができる確立的推定（条件つき確立）と呼んでいる。ベイズの定理では、イベントAが発生したときにイベントYが発生する条件つき確率を計算することができる。

有方向非巡回グラフを表すデータ構造を持ち、変量間の依存関係を表現するのに使われるベイズ流ネットワークと呼ばれるものがある。これは各変数に対応するノードがあり、そこには親ノードとの間の関係を定義する条件付き確立表を持っている。ベイズ流ネットワークは本来、不確実性のある推論のための確率論を利用している。

その他にも、プール理論ベースのシステムに代わるものを提供してくれるファジィルールシステムなどといったものがある。ファジィとは“あいまいな”という意味であり、ファジィ理論は、あるイベントが起こる度合いについて扱っている。あることが起こる度合いは、あいまいではっきりと定義できない状態である。この状態を0.0から1.0まで連続した範囲の中で真理値を扱っている。そのため、「真とも偽ともいえない(0.5)」、「ほぼ真(0.9)」、「ほぼ偽(0.1)」という状態が成り立っている。ファジィ理論の優れた部分は、自然言語に近いフォーマットのルールで知識表現が可能なのである。ファジィルールシステムでの推論は前向き推論⁶である。

エージェントが知的な活動をする上で学習と推論は欠かせない要素である。そして、研究が進むことによって、人間に近い形で学習をして推論をすることが可能となってきた。エージェントは、学習した後に推論をし、推論した行動によって学習を繰り返すことによって、より人間に近い知的活動を行うようになる。

2.4.3 すべての知的エージェントが有する原理

ここからは、すべての知的なエージェントが有する原理を四つを紹介する。

エージェントでも単純な分類に単純反射神経エージェントというものがある。このエージェントは、現在の知覚をもとに行動し過去の知覚履歴を完全に無視する。例えば、掃除機のエージェントがいたとすれば、現在位置とその場所が汚れているかどうかのみに基づいて判断するのである。このようなエージェントは、何らかの情報処理が目視入力情報によって条件が成立され、行為の連鎖を生起させるのである。このような連鎖を条件—行為規則と呼んでいる。単純条件反射エージェントは、大変簡素である代わりに非常に限定的な知性しか持たないため、現在の知覚のみに基づいて正しい判断が行われる場合、環境が完全観測可能である場合に限り、正しく動作するのである。つまり、それ以外の場合は動作しない場合がある。マルチエージェント環境においては乱数を用いれば、解決することも可能であるが、たいていは、合理的な結果が得られない場合が多いのである。

部分的観測性能を最もうまく扱うには、エージェントが現在みることができないこの世の部分の推定追跡することが有効である。つまり、直接観測できない現在の状態を反映する内部状態を知覚系列に基づいて管理する方法である。このように時間の経過と共に変化しているため、次に何が起きるか予測するのは非常に困難である。そのため、エージェントには、二つの知識が必要である。一つめは、どのように外界が変化するかという知識、二つめは、エージェントの行為が外界にどのような影響を与えるのかということである。このような外界は動くのかという知識は、外界に関するモデルといわれている。そのようなモデルを扱うエージェントをモデルベース条件反射エージェントと呼んでいる。これは内部構造を有する条件反射エージェントの構造であり、現在の知覚情報が古い内部状態とどのように組み合わせられて新たな現在の記述に更新されるか表している。状態に関する既知の知識と照らし合わせて、新しい知覚を解釈し、さらにどのように外界が変化するかに関する知識を使い、外界状態のみ見えない部分を推定追跡するのである。またエージェント自身の行為が外界にどのような影響を与えるのかも知らないといけない。

何をすべきか決める際に現在の状態を知ることと同時に望まれる状態を記述した目的に関する情報も必要とされる。エージェントプログラムは目的を達成するための行為を選択するために、可能な行為の結果に関する情報（条件反射エージェントで内部状態を更新するために使われる情報と同じもの）と目的状態を組合さなければならない。そのようなものを目的ベースエージェントと呼んでいる。エージェントは目的を達成する行為系列を発見するために探索とプランニングを行うのである。目的ベースエージェントは、推論可能なエージェントであり、世界は常に変化しているという知識に基づいて、柔軟な対応をすることができる。条件反射エージェントの場合、変化に対応するためには常に書き換えが必要であるが目的ベースエージェントではそれが不要である。

効用ベースエージェントとは、性能指標を与えることによって、同じ目的が達成されるにしてもエージェントがどのくらい効用（満足度）を得られるかについて異なる外界の状態と比較し、ある外界の状態が他のよりも望ましい状態であるかどうかを測ることが可能である。効用を測るために効用目的エージェントには、効用関数が使われている。効用関数とは、状態を効用の程度に応じて実数をつけるもののことである。効用関数は、定義付けすることができる。第一に矛盾する目的が複数ある場合には、効用関数が適切にトレードオフを表していれば、一部の目的だけが達成される。第二にエージェントが達成できるかどうか分からない複数の目的を目指す場合には、効用によって目的達成の重要性に重みをつける。これによって、不適當である場合に合理的な決定を下すことができる。知的なエージェントは、効用関数の期待値を最大化するように行動しなければならない。

[注]

1) ファシリテータ：情報や知識が格納されている情報源とユーザーの間で仲介を行うエージェントである。

- 2) GUI : ユーザーに対する情報の表示にグラフィックを多用し、大半の基礎的な操作をマウスなどのポインティングデバイスによって行うことができるユーザーインタフェースのことである。
- 3) 逆伝播ニューラルネットワーク : データが一方向へネットワークを通じて流れるという意味のフィードフォワード型接続トポロジーを特徴としており、接続のウェイトを調整するために誤差逆伝播法と呼ばれる技術を使っている。入力ユニットと出力ユニットの層に加えて、他のユニットから入力を受け取る、1つ以上のレイヤーの隠しユニットを持つことができる。プロセッシングユニットの1つの隠しレイヤーを持つ逆伝播ネットワークは、隠しレイヤーに十分な数のユニットがあれば、任意の連続関数のモデルとする学習できるのである。
- 4) 決定木 : データセット例に対する機能を実行し、類別機と予測モデルを生み出すことができる。決定木はデータセットを調べ、情報理論を利用してどの属性が決定の基礎をおく最大情報を含むか決める。
- 5) ベイズの法則、ベイズの定理 : 過去の経験から未来に何かが起こる可能性について条件付き確率を使って推測する定理です。
- 6) 前向き推論 : 入力されたデータのある決められた組から、新しい情報を引き出すものである。

3 エージェントとの内部構造

3章では、エージェントの特性や特徴を理解するためにはエージェントがどのような内部構造を持っているのか考察していくことが必要である。また、エージェントの特徴である推論や学習をするために、エージェントがどのように周りの環境の状態を理解しているのか知らなければならない。なぜなら、エージェントは自らの知識と周りの環境の状態を知ることによって、知覚を増やし的確に行動することができるからである。そのために、エージェントは環境に適応するためにどうしているのか、または、エージェントが適応できる環境はどのようなものなのか、それらの環境をどうやって知覚しているのか、といった問題を取り扱っていく。さらに、それらを考察することによって、エージェントの利便性と問題点についても言及していきたいと考えている。

3.1 エージェントの構造

インターネット上で活動するエージェントの場合、その多くが **Java** によって作成されている。**Java** とは、リアルタイム組み込みソフトウェアのプログラミングのために設計されたものであるが、最初からオブジェクト指向が備わっており、強力なセキュリティ機構や豊富なネットワーク関連の機能が標準で用意されている。**Java** で開発されたソフトウェアは特定の OS やマイクロプロセッサに依存することなく、基本的にどのようなプラットフォームでも動作する。その反面、標準ではどのプラットフォームでも実現できる最大公約数的な機能しか利用できないため、プラットフォーム固有の機能を利用する用途にはむいていないのである。エージェントに **Java** が多く使われている理由としては以下にあげるものが考えられる。

一つめの理由として、**Java** アプリケーションは、独立したプロセスであるため、長い時間実行を待ち、自律的であることができるためである。ある **Java** アプリケーションは、ソケットを使って、別のプログラムと通信することができる。また、スレッド化したアプリケーションをサポートし、これら二つの機能を使って自律性を保っている。

二つめの理由として、**Java** は、オブジェクト指向的論理、洗練された推論、学習能力などの知的活動をサポートするための基本的な機能が備わっている。また、**Java** は、知識表現とそのアルゴリズムをオブジェクト間の関係と同様にコード化するために用いることができるため、フレームや意味ネットワークや **if-then** ルール¹ などといったものは容易に **Java** で実装できるのである。

モバイルエージェントの基本的な要件に実行されているプロセスの状態をセーブできること、それを送り出すこと、そして、そこで残されたプロセスを再開できること、といった能力が必要となっている。それは、異なるシステム上ですべての機能を実行できなければならないのである。これを可能とするためには、同質の環境を持つ機械の実現が必要であった。それは **Java** 仮想マシンによって実現可能となった。**Java** プロセスが実行できるような標準的コンピュータ環境を与えることで、同質の仮想マシンを供給し、これによっ

て異種混合ハードウェアシステム間で何の問題もなく Java エージェントが動くことが可能となった。

ハードウェアエージェントは、アーキテクチャとエージェントプログラムから構成されている。アーキテクチャは、普通のコンピュータかもしれないし、カメラやセンサーなどを搭載したロボットのなものも考えられる。一般的にアーキテクチャはプログラムにセンサーから知覚情報を入力し、プログラムを実行し、プログラムが選択した行為をそのままアクチュエータに入力し、行為を出力する。エージェントアーキテクチャとプログラムの間には以下の関係がある。

エージェントタイプ	知覚 (P)	動作 (A)	目標 (G)	環境 (E)
医療診断システム	症状、所見、患者の応答	質問、テスト、治療	健康人、コストの最小化	患者、病院
衛星画像解析システム	異なる強度のピクセル、色	シーンのカテゴリの印字	正しいカテゴリ化	人工衛星からの画像
部品移動ロボット	異なる強度のピクセル	部品をつかみ、箱に分類する	部品を正しい箱に入れる	部品を載せたコンベアベルト
精製機制御装置	温度、圧力計の読み	バルブの開閉、温度の調節	純度、生産性、安全性の追求	精製機
対話的英語教師	タイプされた単語列	練習問題、示唆、訂正の印字	学生のテストの点を最大にする	学生の集合

表 1 : エージェントとアーキテクチャとプログラムの関係図

http://www2.ocn.ne.jp/~mbox/AI_03.htm 知的エージェントより参照

3.1.1 エージェントの判断基準

エージェントが正し行動、つまり、合理的な行動したことを判断するためには何らかの手段で成功の程度を測る必要がある。このエージェントがどれくらい成功したか評価する基準を性能指標と呼んでいる。この性能指標はエージェントがユーザーの望んだ結果をどれだけ達成できたか判断するために重要なものである。エージェントがある環境に投げ込まれたとき、それを受ける知覚によって行為系列が生成される。この行為系列が環境の状態に変化をもたらし、その系列が望ましいものであれば、エージェントは正しい行動をしたということになる。このような基準はエージェントを設計したものが決める場合が多いのである。性能指標の達成基準を考えると、一般的にエージェントがどのように振る舞うかよりも環境の中で実際にユーザーにとって、何が最も望ましいかを考えて性能指標を

決めるのが最適である。

3.1.2 エージェントと環境

エージェントは知的な活動である推論や学習をするためには、周りの環境を知覚する必要がある。ソフトウェアエージェントの場合、キーの打鍵やファイルの内容、ネットワークパケットなどを通じて環境を感じ取り、スクリーンへの表示やファイルの書き込み、ネットワークパケットに送出などを通じて環境に働きかけかける。またロボット工学などのハードウェアエージェントは、ある環境においてセンサーによって、その環境がどんな状態であるかを感じ取り、その環境にアクチュエータ²を通して働きかける。ロボットなどは、赤外線センサーやカメラなどがセンサーであり、様々なモーターがアクチュエータである。

ある環境においてエージェントが感じていることを知覚と呼んでいる。その知覚したものをエージェントは知覚系列という履歴に完全に残しておく。それによって、エージェントは、行為の選択をもとの知識と観測された知覚系列に依存することとなる。もし、すべてのありえる知覚系列についてエージェントの行為選択ができれば、そのエージェントに何ができるのかということが分かったこととなる。ちなみにエージェントの行為はエージェント関数によって表されることとなる。ほとんどのエージェントは、それをエージェント関数表で表そうとするが、それだと膨大な表になってしまうために知覚系列の長さ制限を設けている。エージェント関数はエージェントプログラムによって実装される。エージェントプログラムとは、エージェントの知覚から動作へのマッピングを実現する関数を設計することである。

Stuart Russell & Peter Norvigによれば、知的なエージェントを設計する際に、そのエージェントがどれだけうまく知性を持っているか振る舞えるか、それは環境に依存する。そのため、知的なエージェントを作成するためにはタスク環境についても考えなければならない。タスク環境の広がり幅は幅広いが、かなり少数の性質から種類分けする事が可能である。これによって、どのようなエージェントがその環境に適切であるか見極めることが出来る (Stuart Russell & Peter Norvig 2008)。次からタスク環境の性質についてみていく。

完全観測可能とは、エージェントのセンサーが各時刻で環境の状態を完全に簡素おく可能であれば、タスク環境は完全観測可能ということになる。また、センサーが行動選択に関連するすべての局面を検知できるならば、そのタスク環境の完全観測である。完全観測可能な条件は外界に追従するために必要なエージェントの内部認識状態を管理する必要がないので、目的を達成することは簡単である。しかし、センサーのノイズや不正確さ、あるいは単に一部の観測情報データの不足などのために環境が部分観測可能な条件になる場合もある。この状態では、エージェントは周りの環境を知ったり、考えたりすることが困難となる。

環境の次の状態が完全に現在の状態から決定され、そしてエージェントのみが行為を行うならば、環境は決定的である。そして、これ以外の環境を確率的というのである。原則として完全観測可能で決定的な環境では、エージェントは不確実性を扱う必要は無いのである。もし環境が部分的観測可能であるならば、環境は確率的であるように見える場合がある。この状況は環境が特に複雑で、観測されないすべての局面に追従せねばならない場合に起こりうる。したがって環境が決定的であるか、確率的であるかは、エージェントの視点から考えなければならない。ちなみに他のエージェントの行為を除いて環境が決定的である場合には戦略的と呼んでいる。

挿話的タスク環境とはエージェントの経験が基本的挿話に分割される。各基本的挿話はエージェントの知覚と一つの行為実行から構成される。ここで大切なことは、次の挿話は前の挿話で行われた行為に依存しないことである。挿話的環境では、各挿話内での行為の選択はその挿話のみに依存するのである。つまり、エージェントがその場で判断したことが将来まで影響しないことである。これに対して、系列的環境では、現在の判断が将来の判断にまで影響するのである。

エージェントが熟考している間に環境が変化してしまえば、そのエージェントにとって環境は動的であるといい、そうでなければ静的であるという。静的環境であれば、エージェントが行為を決定している間に世界を監視し続ける必要はなく、時間経過を気にする必要も無いので問題として扱いやすいのである。一方で、動的環境ではエージェントが決定を下してない間は、何も行動しないという選択をしたとみなされ、次に何を行いたいかわれ続ける。環境自体が時間経過とともに変化することはないがエージェントの性能指標が時間経過とともに変化する場合のような状況を準動的であるという。

環境や時間、エージェントの知覚と行為には、離散的扱いと連続的なものがある。たとえば、チェスのような離散的状態の環境には、有限に異なる状態が存在する。また、知覚や行為の状態は離散的に表される。タクシーの運転は、タクシーや他の車両のスピードや位置が連続な値の範囲にわたって滑らかに時間経過するように連続状態と連続時間で表される。タクシーの行為も連続的である。

単独エージェントとマルチエージェントとの違いは、ある目的に対して単体で活動しているのが単独エージェントであり、ある目的にたいして複数のエージェントが活動しているのがマルチエージェントである。しかし、ここで考えなければならないのは、ある環境にいるエージェント以外のものをエージェントみなすべきか、いなかである。考えるべき点は、エージェント A の挙動に依存する性能指標値を最大化するように、エージェント B の挙動もうまく記述できるかどうかである。例えば、チェスなどでは相手の性能指標を最小化すべく自分の性能指標を最大化する状態は競争的マルチエージェント環境である。タクシーの運転などでは、すべてのエージェントが性能指標を最大化するので、部分的に強調的マルチエージェント環境である。マルチエージェント環境においてはコミュニケーションが合理的活動を生みだしたりすることもある。また、部分的観測可能な環境では予測

可能性が時折うまく実現しないため確率的挙動が合理的になることもある。

ここで注意していただきたいのが、環境の実際状況の大半は複雑であり、いつも決まりきた答えがあるわけではないということである。

3.2 エージェントの利便性

エージェントは様々な領域において、幅広く活躍することができる。そのため、多種多様なエージェントが存在し、活躍している。エージェントが知的になることは、我々の生活に様々な利益を与えることができるようになるということである。では、私たちがどのような利便性が得られるのかを以下に示していく。

1. エージェントが人間の仕事を補助し、行為を代行してくれることによって、目的の達成が早くなり、自らが負担しなければならない仕事が軽減される。例えば、電子メールなどの処理をエージェントがユーザーの嗜好を学ぶことによって、代わりにしてくれることも可能である。
2. マルチエージェントシステムやモバイルエージェントを使用することによって、分散検索、並列実行によって問題処理が早くなる。それによって大規模なデータ処理を早い時間で行うことができる。
3. マルチエージェントシミュレーションシステムを使用することによって、人間の社会現象を再現することができ、それによって実際のデータに近いデータを得ることができる。
4. 擬人化エージェントやロボット工学によって、体の不自由な方などのサポートやお年寄りの補助をすることも可能である。
5. AIの研究が進むことによって、人間の脳に関する研究が進むこととなる。AIの研究は認知科学と密接に関連しているためである。認知科学という学際分野はAIによるコンピュータモデルと心理学の実験技術を合わせ、人間の心の働きの精緻で検証可能な理論の構築を目指している。これにより人への理解が進んでいくことができる。

3.3 エージェントの問題点

エージェントは、我々に多くの利益を与えてくれる一方で、問題点も数多く考えられる。それらの問題点を解決することによって、より良いエージェントシステムを構築することができる。ここでは、問題点を例示すると同時に解決策を提示していく。

1. マルチエージェントシステムやモバイルエージェントシステムなどのエージェントは他のエージェントと強調して、行動することが可能である。しかし、それには共通のプラットフォームが存在しなければならなかった。JavaとAIは前述でも述べているように非常に相性がよく、そのような問題を解決できる可能性を持った存在である。
2. 1と重複する内容になってしまうがエージェントが他のエージェントと相互作用するの

には、共通言語、共通語彙、共通概念が必要である。そのため、KQML などのような共通の知識データベースなどが開発されている。

3. 移動性を持つモバイルエージェントなどは、セキュリティに関する問題が考えられる。それはエージェントが他社のコンピュータに移動し、ウイルスのような振る舞いをする事が考えられるためである。例えば、エージェントによるデータの盗み出し、ファイルの破壊などの問題があげられる。この場合、エージェントに対するアクセス・コントロールが必要となる。アクセス・コントロールを課すためにはエージェントの認証が重要である。
4. 現在のエージェントシステムでは複雑すぎて、一般に普及するのが難しいことがあげられる。そのため、我々が使っているような自然言語によって、エージェントを扱えるようになるために、擬人化エージェントなどの開発が進んでいる。
5. エージェントが発達することによって、我々のやるべきことが大幅に減ってしまう可能性が考えられる。現在、この問題はまだ小規模ながら、生産ラインの自動化などによって起こっている。将来的には、人間と知的な機械との関係性についても考えなければならぬかもしれない。

[注]

1) if-then ルール：構築しようとする対象領域の知識を「もし、XならばYせよ」という形のものである。

2) アクチュエータ：アクチュエータとは、入力されたエネルギーを物理的な運動へと変換する機構のことである。コンピュータの関連では特にハードディスクの磁気ヘッド部分を動作させる機構を指すことが多い

4 エージェントの実用例

4章では、現在多くの領域で使われているエージェント技術について紹介していく。それによって、エージェント技術が持つ可能性と我々の身近な場面でも活躍しているということを知ってもらい、エージェントの可能性を感じてもらいたい。

4.1 自律的プランニングとスケジューリング

NASA の遠隔エージェントプログラムは初の搭載型自律プランニングプログラムとして宇宙船の作動スケジューリングを制御している。遠隔エージェントは地上で与えられた高位の目標からプランを作成し、宇宙船がそのプランを実行するのをモニターし、問題が生じると検知、診断、回復をする。

4.2 ゲームプレイ

IBM の Deep Blue は Garry Kasparov を 3.5 対 2.5 で破り、チェスの世界チャンピオンに勝った世界初のプログラムとなった。Garry Kasparov はボードの向こう側に新種の知能を感じたと述べている。ニューズウィーク誌はこの試合を脳の最後の抵抗と記述した。

4.3 自動制御

ALVINN コンピュータ視覚システムは車を車線にしたがって操舵するように訓練されていた。これが CMU のコンピュータ制御ミニバン NAVLAB が搭載され、2850 マイルにわたって合衆国を走破する 98%の間操舵を行った。人間が運転した 2%は高速の出口のランプであった。NAVLAB はビデオカメラによる道路の映像を ALVINN に送り、これが過去の訓練走行による経験に基づいて操舵の最適方向を計算した。また、自律無人航空機を設計するために人工知能の考え方を取り入れ、知的エージェントとして無人航空機を捉え、連続時間システムと離散事象システムを同時に扱うことが可能な自律飛行管理システムを構築し、シミュレーションにより検証している。

4.4 診断

確率的分析に基づく医療診断プログラムは、複数の医療領域において、熟練医と同レベルの能力を示してきた。Heckerman の記述によれば、リンパ線病理の優れた専門家が非常に診断困難な症例におけるプログラム診断を笑い飛ばしたことがあるそうである。しかし、プログラム製作者たちは、専門家にコンピュータに診断の説明を求めるように勧める。機械が判断に影響した主因子を指摘し、この症例におけるいくつかの症状の微妙な相互関係について説明をすると、専門家はプログラムに同意するのである。

4.5 戦略立案

1991 年の湾岸危機の間、合衆国軍は動的分析プランを再考ツール(Dynamic Analysis

and Replanning Tool)を用いて輸送のための自動戦略立案とスケジューリングを行った。これは 5 万台の車両と、積荷、人員を同時に扱い、出発地点、目的地、ルート、そしてすべてのパラメータ間の衝突回避を行う能力である。AI 計画立案手法を用いることにより、従来の手法で数週間を要した計画を数時間で立案することができた。Defense Advanced Research Project Agency は、このアプリケーションだけで過去 30 年間にわたる AI への投資が回収されたと発表した。

4.6 ロボット工学

今では多くの外科医が微細手術ロボットの補助を得ている。HipNav はコンピュータ視覚を用いて患者の体内の 3 次元モデルを構築し、人工関節の挿入にロボット制御を用いている。その他にもホンダのロボットの二足歩行を可能にした ASIMO や自動掃除機などがある。

4.7 言語理解と問題解決

PROVERB はほとんどの人間の手より上手にクロスワードパズルを解く。このプログラムは可能性のある単語に関する制約、過去のパズルの大規模データベース、そして辞書やオンラインデータベースといった様々な情報源を活用している。例えば、Nice Story というヒントに対しては ETAGE という解がありえることを、データベースにある Story in France/ETAGE というヒントと回答ペアと Nice X というパターンと X in France というパターンはよく同一の解を持っているという知識を組み合わせることで導くことができる。プログラムは Nice がフランスの都市であることを知らないが、パズルを解くことはできる。

5 最適なエージェントとは？

ここまで様々なエージェントをみてきたが、そのどれもが現在の技術では広い領域の限られた分野でのみでしか活躍できていない。5章では、AIは本当に知的に振る舞えるのか再考しつつ、様々なエージェントを考察した結果として、人間を助けてくれる最適な知的エージェントとはどのようなものであるのか考察していく。

5.1 機械は本当に知的に振る舞うのか？

哲学者たちはコンピュータが現れる前からAIに関するいくつかの問題を解明しようとしてきた。それは、人間の心を解明するためだったり、機械が人間のように知的に振る舞うことが可能なのか検証するためだったり、機械が人間のような心を持つことが可能なのか調べるためだったりと多くのテーマを扱ってきた。これらを踏まえつつ、もう一度エージェントの知性について考察していく。現在、AI研究に関しては二つの主張がなされている。

機械はあたかも知性があるかのように振る舞うことができるという主張、つまり、機械が人間のように考えているのではなく、そう見えるように作られているだけという主張を弱いAIと呼んでいる。また、知的であるかのように振る舞える機械は本当に考えているのだという主張、これは人間のように本当に考えているという主張を強いAIと呼んでいる。これらを考えることによって機械が知的に振る舞えるのか考察していく。

5.1.1 弱いAI

最初に弱いAIについて考えていく。多くの学者がAIの知性を否定するときに使われるものとして、不可能性に基づく議論、数学的議論、非形式に基づく議論がある。これら三つについて見ていく。

まずは不可能性の議論についてみていく。AIが実現不可能であり、機械が決して知的に振る舞えない旨を説明しようとしてきた学者たちがいる。例えば、不可能性に基づく議論では“機械には決してXができない”という主張がなされる。しかし、現在のAIは実例でもあげているようにチェスや飛行機の操縦など様々なことを人間の専門家並みにできるようになった。また、人間的な判断（Turingによれば、“経験からの学習”と“真偽を判断する能力”）を含むと思われるような課題もまた人間並みにやってのけるのである。しかし、それはコンピュータが課題に対して人間のように洞察や理解を使っているわけではないということに注意していただきたい。

数学的反論などもある。J.R.Lucasなどの哲学者は、Gödelの不完全定理¹によって機械が人間よりも心的に劣ることを示していると主張している。機械は不完全定理によって、自らの文が真であることを証明できないために制限された形式体系であるのに対し、人間はそのような制限はないのである。第一にGödelの不完全性定理は算術をするのに十分強力な形式的体系のみに適用される。しかし、コンピュータは有限であり、いかなるコンピュータも命題論理の非常に大きな体系として記述できるため、Gödelの不完全性定理は適

用されない。第二に他者がある文の正しさ証明できるのに自分自身では証明できないことは恥じるべきではないということである。例えば、“J.R.Lucasはこの文が正しいことを矛盾なく証明できない”という文があるとして、もし Lucas がこの文を主張するなら自己矛盾が生じる。よって Lucas は矛盾なく主張できないため、この命題は真でなければならぬ。このように本人には矛盾なく説明できないのに他の人々には説明できるような例示がある。それは人間の思考力の限界というわけではなく、知的にことにおいて、数学的論理では説明がつかないというわけである。第三に、たとえコンピュータに証明できない限界があるにしても、人間がそのような限界から免れているという証明はないということである。なぜなら、人間に Gödel の不完全性定理が当てはまらないことを証明するのは不可能だからである。それは、いかなる厳密な証明もそれ自身が、形式化してないと言われていた人間の才能の形式化を含み、自分自身を否定してしまうからである。したがって、私たちは何らかの方法で人間離れした数学的洞察を行うことを直感に訴えるのである。この訴えは“そもそも思考が可能ならば、我々は自らの無矛盾性を仮定しなければならない”というような議論によって表明されている。

AIを批判する上で最も多いものは、Turingによる非形式に基づく議論である。人間の行動はあまりにも複雑なのでいかなる簡単な規則の集合によって捉えられないのである。また、コンピュータは規則の集合に従うことしかできないから、人間と同程度の知的な行動を生成できないというものである。論理的規則の集合であらゆる事物を捉えられないことは、AIでは制限条件記述と呼んでいる。この観点の主な支持者は哲学者の Hubert Dreyfus であった。彼の批判対象は“古き良き AI”または GOFAI と呼ばれている。GOFAI と主張されているのは、問題領域を記述する事実と規則の集合から論理的に推論するシステムによって、あらゆる知的な行動を捉えることができるということが可能であるということである。しかし、これは単純論理的エージェントについては当てはまるが確率的システムを持っているには当てはまらない。彼の批判プログラムの一つの方法を批判しているにすぎないのである。また、彼が主張する背景にある知識に関しての問題、制限条件記述問題、不確実性の問題、学習についての問題、コンパイル形式の意思決定の問題、身体を持たない推論エンジンではなく状況に埋め込まれたエージェントを考える重要性に関しては、すべて現在の標準な知的エージェントは設計に取り入れられている。

5.1.2 強い AI

多くの哲学者は、機械は本当に考えているわけではなく、考える真似事をしているにすぎないという主張を繰り返してきた。Geoffrey Jefferson 教授は、感覚された思考と感情に基づいて機械がソネットやコンチェルトを書くまでは機械が脳に等しいということに同意はできないと主張している。Turing はこれを意識に基づく論理と呼んでいる。つまり、機械が自分自身の心的状態と行為を自覚しなければならないということである。Geoffrey Jefferson 教授の主張は現象学、あるいは、直接的経験と関係している。意識に関しての議

論は非常に複雑である。なぜなら、我々でさえ、内部の心理的状态を把握する直接的根拠は持っていないからだ。そのため、Turingはこの問いに対して、皆が考えるという上品な慣習を持つべきだと主張したのである。しかし、多くの哲学者はそれを解き明かすために研究をし続けている。

例えば、心的状態とは、入力と出力の間の何らかの中間的な因果条件という機能主義というものがある。機能主義では、ある抽象化のレベルが存在し、そのレベル以上の過程が同型である限りは同じ心理的状态を持つことになるということである。

これとは対照的で、生物学的自然主義では、心的状態とはニューロンにおける神経学的過程によって引き起こされる創発的特徴であり、ニューロンの未知の特性が重要である。すると、単に同じ入出力を備えた同一の機能構造を持つ何らかのプログラムに基づいて心的状態が複製されることはありえないのである。複製するには、ニューロンと全く同じ因果的な力を持つアーキテクチャ上でプログラムが動作する必要がある。

5.2 最適な知的エージェント

エージェントは様々な特定の領域分野で活躍している。それは個人の活動にも適用される。例えば、電子メールの整理、ユーザーの思考に合わせた情報検索と提示などである。しかし、それは個人の活動を部分的に支えているにすぎない。現在では、エージェントにとって可能な知覚と行為、エージェントが満たすべき目的、および環境の性質に依存することを示し、簡単な反射的エージェントから、完全に熟考的な知識に基づくエージェントに至るまで、様々なエージェントを設計することができる。こうしたエージェントは、学習や推論、確率やニューロンによって設計される。

エージェントには現在、四つの可能性がある。それをこれから考察していく。

一つめに完全な合理性を持つエージェントである。このエージェントは、ある瞬間において、環境から得た情報のもとで自分の期待効用を最大化するように行動する。しかし、ほとんどの環境において、完全な合理性を達成するのに必要な計算には時間がかかりすぎる。そのため、完全な合理性は現実的に難しいのである。

二つめに計算的理性を持つエージェントである。これは、論理的かつ意思決定論的エージェントの設計において暗黙のうちに用いられた合理性の概念である。このようなエージェントは、熟考を始めたときに合理的な選択であったはずのものを最終的には返すのである。このようなシステムはほとんどの環境では、不適切な時点で正しい答えが得られても意味がないのである。

三つめに限定的合理性を持つエージェントである。これは限定的合理性という概念を現実のエージェントによる意思の記述理論とした。限定的合理性は第一義的にほとんど満足するように動くことを示している。つまり、十分に良い答えを見出す程度にしか熟考しないのである。これでは、エージェントの形式的仕様にならない。さらに「ほどほど満足する」では、限りある資源に対処する広範囲の手法の一つにすぎない。

四つめに限定最適性エージェントについて言及する。このエージェントは所与の計算資源のもとでなるべく良い行動を行う。つまり、限定最適性エージェント用のプログラムの期待効用は、同じ機械の上で稼動する他のエージェントプログラムの期待効用以上なのである。この限定最適性がエージェントの強力な理論基盤として期待できる。限定最適性の良いところは、達成可能というところである。すなわち、少なくとも一つは最良のプログラムなのである。また、エージェントの設計には資源制約が問題となるが、限定最適性では原理づけられた方法で扱える。また、今までのところ、限定最適性にはほとんどわかっていない。非常に単純な機械と制限された種類の環境に対して、限定最適性プログラムは実現できるが、複雑な環境になるとどのようなものになるかわかっていないのである。

汎用の知性は、反応性、思考性、様々な形の知識、それらすべての形にわたる意思決定と学習とコンパイルの機構、推論を制御する方法、巨大な領域の知識が必要である。限定最適エージェントは、自分のいる環境に適応することにより、いずれは自分の内部構造がその個別的な環境に特有の最適化を反映させるべきなのである。そして、私は、そのようなエージェントが個人にパーソナライズ化され、人間や他のエージェントと知的な活動を行うことができるようになることを願っている。

[注]

- 1) Gödel の不完全定理：「数学は自己の無矛盾性を証明できない」ことを示した定理のことである。
- 2) コンパイル：人間がプログラミング言語を用いて作成したソフトウェアの設計図(ソースコード)を、コンピュータ上で実行可能な形式(オブジェクトコード)に変換することである。そのためのソフトウェアをコンパイラというのである。

6 今後の展望

現在のエージェントの発展は、コンピュータサイエンスの教育法やソフトウェア開発の実務をすでに変化させ、発話認識システムや在庫管理システム、監視システム、ロボット、検索エンジンなどにも応用され、有効活用されている。今後、知的なエージェントが活動する領域はますます増えていくことが考えられる。例えば、エージェントが発達し、冷蔵庫の中をネットワーク上で確認できるならば、その中身を外出中のユーザーに伝え、その中身で作れるものを伝えたり、あるものを買って作れるものを提示してくれたりするかもしれない。または、宇宙や深海など人間が探索できないところをエージェントが代わり探索し、有効な情報を提供してくれるかもしれない。

近い将来に期待されるエージェントは部分的な領域において、最適な活動をしてくれる知的なエージェントである。それらのエージェントがユーザーの目的を達成するためにネットワーク上を移動し、同じ領域または他の領域で動く他のエージェントと協調、または競争し合い、その過程で学習することによって、ユーザーに効用の最大化を与えてくれるようになるだろう。また、擬人化エージェントが発達し、人間の活動を機会言語ではなく、我々の自然言語を使ってサポートしてくれるかもしれない。

もう一つの可能性としては、複数のコントローラーを用いた自律的にシステムを制御する協調分散システムの中で、各エージェントが学習によって自身の行動戦略を獲得する手法が考えられる。また、環境の変動にすばやく対応する能力も求められるようになる。この場合、再学習を行うことで、環境に適応させなければならないが、そのようなことを何度も繰り返すのは、非効率的である。効率的に環境の変動に対応するためには過去の学習によって得た知識を保存し、必要な状況で再利用するほうが効率的である。つまり、上記と重複するが限定された領域で活動する独立した知性を持つエージェントが他の領域で活躍するエージェントと協調することによって、さらなる知性を得、よりユーザーのために働くエージェントも期待できるのである。

そして、遠い未来に真に有用な個人用アシスタントエージェントが完成されれば、私たちの生活に多くの恩恵を与えることが可能である。それは人間のように学習し、推論し、活動するものも考えられれば、普通の人と接すように交流することが可能なものも考えられうる。それはネットワーク上に存在するプログラムかもしれないし、ロボットのようなものかもしれない。エージェントのような人間を模倣した存在は、映画や漫画の世界だけで起こりえるものでしかないと考える人達も多く存在することであろう。しかし、実際に多くの研究者が知的なエージェントや人間の脳の解明をするために研究をしており、その分野の幅は広く、拡大している。遠い未来には、本当に人間のようなエージェントが誕生する可能性は十分ありえると私は考える。

エージェントの開発は始まったばかりであり、解決しなければならない問題は数多く存在する。意識や心が何処に宿るのかという認知科学の問題、我々が自然に行っている抽象的な概念や曖昧な部分の処理をどうエージェントに理解させるかという問題、より人間に

近い推論や学習をさせるための問題など、より困難な問題を解決しなければならないのは明白である。それらを解決し、我々自身が人間という存在を本当に理解したとき、人間のような知的で自律的な行動をするエージェントが誕生するであろう。

おわりに

今まで、エージェントが人間と同様の知的レベルになれるか論じてきたが、現在の状況では難しいと考えられる。しかし、エージェントの多くの研究者たちが様々な領域において多くの研究を行っている。そのため、将来的には実現する可能性も残されていると私は考える。もし、そのようなエージェントが実現したならば、我々の生活に大きな変化を与えることは間違いないだろう。その点において、知的なエージェントを研究することは価値あるものと考えられる。また、エージェントの研究は人間の脳を理解する上で重要な役割を果たすため、我々人間という存在がどういったものであるかという理解にもつながる。しかし、エージェントの研究がすべてより良い価値を与えるわけではない。エージェントの開発が進み、高性能なエージェントが誕生すれば、自律型の機動兵器を誕生させてしまうかもしれない。また、知的なエージェントという人間の代わりとなる存在が人間の価値を揺らがせるかもしれない。エージェントの発展は、多大な恩恵を与えると同時に今まで考えられなかった事態を起こす可能性は十分考えられる。私たちは知的なエージェントを開発すると同時に将来的に考えられる様々な問題を考慮しながら、個人の良きパートナーとなる知的なエージェントについて考えなければならない。

最後に本論は、エージェントについて浅く広く取り扱ったものであるため、各研究の深いところまで触れていない。本論はあくまでもエージェントを分かりやすく、広く知ってもらうためのものである。少しでもエージェントについて興味を持たれたのなら、各研究分野について研究することを私は勧めたいと思う。そして、多くの人がエージェントに興味を持ってもらえることを私は願っている。

参考文献・URL

山影進 服部正太 『コンピュータのなかの人工社会—マルチエージェント シミュレーションモデルと複雑系』 共立出版株式会社 2002年

石井和平 『社会情報学-情報技術と社会の共変』 学術出版会 2007年

S.J.Russell P.Norvig 古川康一 『エージェントアプローチ人工知能 第2版』 共立出版 2008年

本位田真一 大須賀昭彦 『オブジェクト指向からエージェント指向へ』 ソフトバンクパブリッシング 2001年

長尾確 『エージェントテクノロジー最前線』 共立出版 2000年

電気学会 新ソフトウェアアーキテクチャの応用産業調査専門委員会 『オブジェクト指向とエージェント 基礎と応用』 株式会社オーム 2001年

Joseph p. Bigus Jennifer Bigus 井田昌之 『Javaによる知的エージェント入門』 ソフトバンクパブリッシング株式会社 2002年

岩井俊弥 『Java モバイルエージェント』 株式会社ソフト・リサーチ・センター 1998年

IT用語辞典 e - words

<http://e-words.jp/w/E382A8E383BCE382B8E382A7E383B3E38388.html>

ソフトウェアエージェント紹介 葉雲文

<http://www.sra.co.jp/public/doc/GSletter/vol.31/agent/agent.pdf#search>

モバイルエージェント 佐藤 一郎

<http://research.nii.ac.jp/~ichiro/lecture/model2002/mobileagent.pdf>

ネットワーク環境におけるモバイルエージェント機構に基づいて

個人の継続的な支援を目指した分散ロボット統合法

小川 原光一、金広 文男、稲葉 雅幸、井上 博允

http://www.robot.mach.mie-u.ac.jp/~nakamoto/Ronsho/2007_12_18%20nakamoto/%CA%AC%BB%B6%A5%ED%A5%DC%A5%C3%A5%C8%C5%FD%B9%E7%CB%A1.pdf

モバイルエージェントの応用- マルチエージェントシステムのためのモビリティの利用-

新谷 虎松、大園 忠親、福田 直樹

<http://tomocha.net/kindai/NOTE/mobile-agent.pdf>

ソフトウェアエージェントにおける物理世界への気づき 中野 有紀子

<https://kaigi.org/jsai/webprogram/2009/pdf/268.pdf>

ソフトウェアエージェント 浅川 和雄

<http://img.jp.fujitsu.com/downloads/jp/jmag/vol50-4/paper05.pdf>

Web エージェントシステム MiSpider における継続的実行について

大園忠親, 深萱裕二郎, 伊藤孝行, 新谷虎松

<http://www.nue.riec.tohoku.ac.jp/jsst2005/papers/05056.pdf>

知的ネットワークエージェントを用いた Web 利用システムの開発 石川 尚志

<http://www.kochi-tech.ac.jp/library/ron/2006/2006ele/1070248.pdf>

エージェント技術 服部 正典

http://www.toshiba.co.jp/tech/review/2001/06/56_06pdf/sinseiki.pdf

エージェント技術 石黒 真

<http://www.fuka.info.waseda.ac.jp/~makoto/research/pdf/AgentTechnology.pdf>

オントロジーを利用した知識の共有／再利用 伊藤 英毅

http://www.unisys.co.jp/tec_info/tr64/6407.pdf

ソフトウェアエージェント実用化の課題 益岡 竜介 y

<http://www.flacp.fujitsulabs.com/~rmasuoka/papers/ieice-sig-ai-1998.pdf>

マルチエージェントによる医療情報の知的利活用支援技術の開発

木下 哲男、高橋 秀幸、菅沼 拓夫、白鳥 則郎

http://sendai-cyber.icr-eq.co.jp/research/report/files/h2204_1_2_2_3.pdf

マルチモーダルインタフェース 三吉 秀夫、関 進、綿貫 啓子

<http://www.sharp.co.jp/corporate/rd/journal-77/pdf/77-09.pdf>

モバイルエージェントのためのソフトウェアプラットフォーム 本位田 真一

<http://www.nii.ac.jp/journal/pdf/03/03-08.pdf>

モバイルエージェントの応用 マルチエージェントシステムのためのモビリティの利用

新谷 虎松、大園 忠親、福田 直樹

人間型ソフトウェアエージェント応用システム 今村 佳世、西山 高史

http://panasonic-denko.co.jp/corp/tech/report/533j/pdfs/533_15.pdf

人工知能は私たちの生活にどのように役立っているか？ 黄潤和

http://cis.k.hosei.ac.jp/~rhuang/Miccl/HuangLecture_iis.pdf

知識流通のためのメディア技術 インターフェイスエージェントの利用 中野 有紀子

http://shakai-gijutsu.org/vol1/1_77.pdf

知的エージェント 馬場口 登

<http://www2c.comm.eng.osaka-u.ac.jp/~babaguchi/aibook/agent.pdf>

知的エージェントとその言語発達に関する研究フレームワーク 新田 恒雄

<http://www.ai-gakkai.or.jp/jsai/conf/2008/program/pdf/100297.pdf>

知的エージェント型自律無人航空機の研究 鈴木 智、平沼 敏、藤本 淳、吉原 洋光

<http://www.mhi.co.jp/technology/review/pdf/425/425204.pdf>

知的マルチエージェントの機能分散と自己言及 三宅 美博

<http://www.myk.dis.titech.ac.jp/html/Paper/kaisetsu/1996/2.pdf>

未来型ウェブにおけるソフトウェアエージェントに関する研究

山口 高平、樽松 理樹、和泉 憲明

<http://www.hbf.or.jp/grants/pdf/g/13-g-yamaguchi.pdf>