

実世界における一般問題解決システムの提案とそのヒューマノイドロボットへの実装

巻瀧 有哉^{†a)} 申 富饒^{††b)} 長谷川 修^{†††c)}

General Problem Solving System in a Real World and Its Implementation on a Humanoid Robot

Naoya MAKIBUCHI^{†a)}, Furao SHEN^{††b)}, and Osamu HASEGAWA^{†††c)}

あらまし 本論文では、実世界における一般問題解決システムの構築を目的とし、既存のプランナを利用した3層構造のアーキテクチャを提案する。提案手法は自己増殖型ニューラルネットワーク(SOINN)により物体の概念を形成し、また、環境や人間との相互作用を通じて知識をオンラインかつ追加的に獲得する。更に、獲得した複数の知識を一般問題解決器(GPS)により能動的に運用し、実世界における汎用的な問題解決が可能である。実験により、提案手法を実装したヒューマノイドロボットを用いて、提案手法が実世界における汎用のタスクに対して有効であることを示す。

キーワード 実世界知能, ヒューマノイドロボット, 自己増殖型ニューラルネットワーク, 一般問題解決器

1. ま え が き

実世界において知的に振る舞うヒューマノイドロボットの実現には、ロボットの動作に必要な制御理論や知能に関する情報理論のほか、多岐にわたる幅広い研究分野の密接な連携が必要となる。その中でもヒューマノイドロボットの運動制御に関しては、本田技研のASIMOに代表されるような自律二足歩行に関する研究や、小型ヒューマノイドロボットを用いた物体操作の学習に関する研究[1]など、民間企業や研究機関において盛んに研究が行われている。しかし一方で、ヒューマノイドロボットの知能情報処理に関して

は、特に[2]の要求仕様にある「非定型の作業が自律的に実行できる」とする研究はこれまでに報告されていない。そこで本研究では、ヒューマノイドロボットの知能、具体的には実世界における汎用的な問題解決をキーワードとし、その足掛りとなるアーキテクチャを提案する。

従来の知能ロボットは、設計者があらかじめ考えられる状況や行動をロボットに組み込むことで実現してきた。つまり、if Q_1 then A_1 (Q_1 という状況で A_1 という行動をとる)といったif-thenルールを大量に組み込むことによって、あたかもロボットが状況に合わせて知的に行動しているかのように見せてきた。しかしそのようなシステムでは、状況が想定範囲を外れた途端に破綻してしまう。特に実世界のような常に複雑に変化する環境に対して、その時々適切な行動をあらかじめ組み込んでおくことは不可能である。

そうした従来法に代わるアプローチとして、ロボットの根本的な設計論に着眼した認知発達ロボティクス(Cognitive Developmental Robotics [3];以後CDR)がある。CDRではロボットの身体性に焦点を当て、ロボット自身が自らの身体を通じて環境と接し、その結果として世界から得られる情報を理解する過程が重要であるとしている。つまりCDRの基本概念は、環境

[†] 東京工業大学大学院総合理工学研究科知能システム科学専攻, 横浜市

Department of Computational Intelligence and Systems Science, Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology, Yokohama-shi, 226-8502 Japan

^{††} 中国南京大学計算機ソフトウェア新技術国家重点実験室, 中国
The State Key Laboratory for Novel Software Technology, Nanjing University, China

^{†††} 東京工業大学大学院理工学研究科像情報工学研究施設, 横浜市
Imaging Science and Engineering Laboratory, Tokyo Institute of Technology, Yokohama-shi, 226-8502 Japan

a) E-mail: makibuchi.n.aa@m.titech.ac.jp

b) E-mail: frshen@nju.edu.cn

c) E-mail: hasegawa.o.aa@m.titech.ac.jp

との相互作用を通じてロボットが自らの行動を認識・学習することで、様々な状況に対応可能な経時的に発達するシステムを構築することにある。CDR の必要性は「ロボット分野に関するアカデミック・ロードマップ [4]」においても指摘されている。

更に、Weng らはロボットの知能発達という観点から、発達システムのもつべき性質を以下のようにまとめている [5]。

- I. タスクに特有なシステムではない
- II. タスクがシステムの設計者にとって未知である
- III. 未知のタスクに対するアプローチを生成できる
- IV. オンライン学習が可能である
- V. オープンエンド学習が可能である

CDR に基づく手法であれば、III. 以外のすべての性質を満たすことが原理的には可能である。しかし性質 III. は、知能レベルにおいて他の性質とは一線を画すものである。この性質は知識（タスクに対する解決法、以後これを知識と表現する）を“獲得”するだけでなく、知識を自律的に“生成”する能力を求めている。これを実現するには CDR の基本概念に加え、既存の知識を利用して新しい知識を生み出すメカニズムが必要となる。

小倉らは、ヒューマノイドロボットのための簡易で環境が変化しても再現可能なオンラインでの教示システムを開発した [6]。このシステムでは人間がその場でロボットの身体を直接触って動作を教示することで、新しいタスクや環境での行動獲得が可能である。例えば、キッチンなどの生活環境においてロボットがコップのつかみ方を知らない場合、人間がロボットの手先をもって即座に教示できる。このロボットは教示時の目標状態だけを記憶し、動作再現時に教示時と自己の位置が異なったり、途中で障害物がある場合でも教示された動作を再現（目的再現）することができる。このロボットの目的再現は、一見知識を生成しているように見える。しかし、これは教示時とは動作系列が異なるという意味では自律的な行動生成はしてはいるものの、タスク全体としては Teaching-Playback にほかならない。つまり知識そのものを生成してはならず、Weng らの示す性質 III. を満たしていない。

例えば、人間の教示によってロボットが「目の前にあるコップを手取る」という知識と「手にもっているコップを洗う」という知識を獲得したとする。このとき、小倉らの手法ではロボットはこれらに対応した二つのタスクは実行できるが、「コップを手にもって

ない状態で目の前にあるコップを洗う」という新しいタスクは実行できない。これを実現するには、ロボットは「目の前にあるコップを手にとって、手にもったコップを洗う」という一連の作業をあらかじめ一つの知識として獲得しておかなければならない。しかし我々人間は、そうした一連の作業を直接的に教えられていなくても、既存の知識を組み合わせることでこのタスクを実行できる。このように人間と同じようにして、汎用のタスクに対して既存の知識を能動的に運用し、新たな知識を生成可能な知能ロボットの実現が求められる。

1.1 本研究の目的

本研究では、III. を含め Weng らの示す性質をすべて満たす、実世界における一般問題解決システムを提案する。提案手法は図 1 に示すアーキテクチャであり、提案手法を実装したロボットは以下の特長をもつ。

- ロボットは視覚や聴覚から得られるパターンから物体の概念（シンボル）を形成する：シンボルグラウンディング問題 [7] に関しては依然議論が続いているが、提案手法ではパターンをシンボルにマップするインタフェースとして、オンライン追加学習が可能でニューラルネットワークを用いる。
- ロボットは環境や人間とのインタラクションによって行動の因果関係を知識としてオンラインかつ追加的に獲得する：ロボットの動作は小倉らのシステムと同様、人間がロボットの手先をもって誘導することで教示する。

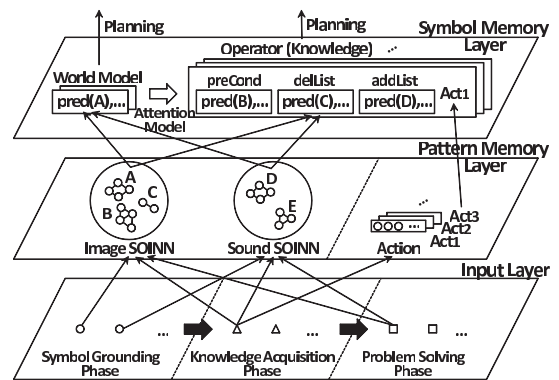


図 1 提案手法のアーキテクチャ：入力層の黒矢印は処理の流れ、各層間の矢印はデータの流れ、シンボル記憶層の白矢印はオペレータ（知識）の生成を意味している。また、パターン記憶層内の境界は、ロボットの動作情報が SOINN 空間とは別の空間に保持されることを意味している。

Fig. 1 The architecture of the proposed method.

● ロボットは既存の知識を組み合わせることで未知のタスクに対するアプローチを生成できる：これは、知識を能動的に運用する手段として古くから知られるプランナの一つである、一般問題解決器 (General Problem Solver [8] : 以後 GPS) を使用することで実現する。

1.2 本研究の優位性

本研究で提案するアーキテクチャは、図 1 に示すように、パターンの入力層とシンボル情報を保持するシンボル記憶層の間に複数のニューラルネットワーク (Self-Organizing Incremental Neural Network [9] : 以後 SOINN . SOINN についての詳細は 2.3.1 で述べる) を含むパターン記憶層を挿入した構成になっている。本アーキテクチャの優位性を以下に示す。アーキテクチャについての詳細は 2. で述べる。

● SOINN の高いノイズ耐性を引き継ぎ、実世界における不安定なパターンに対して頑健に振る舞うことができる。また、SOINN のオンライン追加学習が可能という性質から、知識獲得と問題解決に必要な物体の概念をオンラインかつ追加的に形成できる。

● ロボットに必要とされる、視覚や聴覚といった複数の感覚を用いたマルチモーダルな情報処理が可能である。本研究では画像と音声のデータのみ扱うが、パターン記憶層に別の SOINN 空間を用意することで、それ以外の各種センサデータ等も容易に取り込むことができる。

● 物体の概念を表現するパターンの集合は、個々のオペレータ (知識としてシンボル記憶層に保持されている) の中に保持されるのではなく、パターン記憶層において概念ごとに保持される。そのため、シンボル記憶層ではパターン記憶層で形成された概念を一つひとつのシンボルとして共有でき、シンボル記憶層におけるオペレータの記憶容量を大幅に節約できる。

以下、本論文は次のように構成されている。まず、2. では提案手法の詳細について述べる。次に 3. と 4. にてヒューマノイドロボットを用いた実験について述べる。5. で考察と今後の課題について述べ、最後に 6. をむすびとする。

2. 提案手法

2.1 ヒューマノイドロボット

本研究では富士通製の研究開発用ヒューマノイドロボット HOAP-3 (図 2) を使用する。このロボットは全体として 28 自由度をもち、本研究ではこのうち、

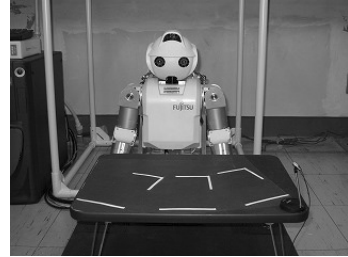


図 2 実験に使用するロボット
Fig. 2 The robot used for the experiment.

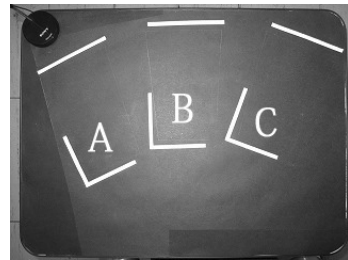


図 3 三つの位置が定義されたテーブル
Fig. 3 The table in which three positions are defined.

頭部の 3 自由度と両腕の 8 (各 4) 自由度を使用する。また、市販の CCD カメラ、マイクロホンを搭載しており、画像入力と音声入力が可能である。

本研究では椅子に固定されたヒューマノイドロボットとその前に置かれたテーブルの上が実験環境となる。フレーム問題 [10] を擬似的に回避するため、ロボットの周囲環境をテーブル上に制限する。また、テーブルの上に図 3 に示すように三つの位置を定義し、各位置には一度に一つの物体しか置けないものとする。

2.2 概要

提案手法のアーキテクチャを図 1 に示す。提案手法は入力層 (Input Layer)、パターン記憶層 (Pattern Memory Layer)、シンボル記憶層 (Symbol Memory Layer) の 3 層構造を成す。このうち入力層は、記号接地フェーズ (Symbol Grounding Phase)、知識獲得フェーズ (Knowledge Acquisition Phase)、問題解決フェーズ (Problem Solving Phase) という三つのフェーズから構成される。これら三つのフェーズでは、実環境から得られる画像パターンや音声パターンを入力として受け取る。また、記号接地フェーズにおける各パターンに付加する概念情報や、知識獲得フェーズにおけるロボットの関節角度の時系列データ (Act1, Act2, Act3, ...) も入力となる。ここで図 1 の入力

層における , , は、このようにフェーズによって入力されるパターンの種類が異なることを意味している。パターン記憶層は入力層から送られるパターンを保持し、パターンをシンボルにマップするインタフェースとして機能する。シンボル記憶層はパターン記憶層から送られるシンボル (A, B, ...) を指数にとる述語 (pred) を要素に含む、実環境のモデル (World Model) やオペレータ (Operator) を保持し、これらはプランニング (Planning) に利用される。

(1) 記号接地フェーズでは、実環境から得られるパターン入力を用い、後のフェーズで使用する物体の画像や音声の概念を形成する。これは画像パターンと音声パターンの学習に自己増殖型ニューラルネットワーク (SOINN [9]) を用いることで実現する。SOINN は既存の学習データを破壊することなく、オンライン追加学習が可能である。また、学習データを十分に表現できるネットワークまで自己組織的に成長するため、あらかじめネットワークサイズを決定しておく必要がない。

本研究ではパターン記憶層に画像パターン用と音声パターン用の二つの SOINN 空間 (Image SOINN, Sound SOINN) を用意する。ロボットは各パターンを、実験者によって与えられる概念情報と合わせ、それぞれの SOINN に入力する。このフェーズについての詳細は 2.3 で述べる。

(2) 知識獲得フェーズでは、ロボットは自らの行動によって生じた実環境の変化から、その因果関係を知識としてオンラインかつ追加的に獲得する。この因果関係は「注意のモデル (Attention Model)」によりそれぞれ一つのオペレータで表現される。注意のモデルは実環境の変化から、オペレータの構成に必要な前提条件 (preCond), 削除リスト (delList), 追加リスト (addList) の三つの要素を獲得し、教示された動作情報 (Act) と合わせてオペレータを構成する。このフェーズについての詳細は 2.4 で述べる。

(3) 問題解決フェーズでは、ロボットは知識獲得フェーズで獲得したオペレータを運用して、実環境において汎用的な問題解決を行う。まず、ロボットは実環境中で提示されたタスクに対して適切なアプローチが存在するか否かを判定する。ここで存在すると判定された場合、一定の評価基準に従ってアプローチを選択し、そのアプローチを実環境において実行する。このとき、ロボットはオペレータの実行とそれによる実環境の変化の観測を、目標状態に到達するまで繰り返

す。このフェーズについての詳細は 2.5 で述べる。

2.3 記号接地フェーズ

本フェーズでは、実験で使用される物体の画像パターンと音声パターンを SOINN で学習することで、それぞれの概念を形成する。SOINN についての詳細は 2.3.1 で述べる。各パターンの学習には、パターン記憶層に概念の種類 (本研究では画像と音声) ごとに用意された SOINN を用いる。各パターンは、概念情報として実験者によって与えられる教師 ID ($teacherID : \geq 0, \in \mathbf{Z}$) と合わせて、それぞれの SOINN に入力される。この教師 ID は、後のフェーズでパターンをシンボルに接地する際に用いられる。

2.3.1 SOINN

SOINN は Shen と Hasegawa が提案したオンライン追加学習手法である。図 4 に SOINN のフローチャートを示す。SOINN は 2 層ネットワーク構造により学習を行うが、1 層目と 2 層目は同じ学習アルゴリズムで動作する。本研究ではこれを簡略化した 1 層のネットワーク構造により学習を行う。そのため、1 層目の学習を終了して 2 層目の学習を行うタイミングを決定するパラメータ LT は不要である。

SOINN は入力ベクトルが与えられると、ネットワーク内ノードの重みベクトルとのユークリッド距離を計算し、入力ベクトルに最も近いノード (以後、第 1 勝者ノードと表現する) と 2 番目に近いノード (以後、第 2 勝者ノードと表現する) を探索する。そして類似度しきい値という基準を用いて、入力ベクトルが第 1 勝者ノードと同一のクラスタに属するか否かを判定する。ここで異なるクラスタに属すと判定された場合、入力ベクトルを重みベクトルとしてもつ新たなノード

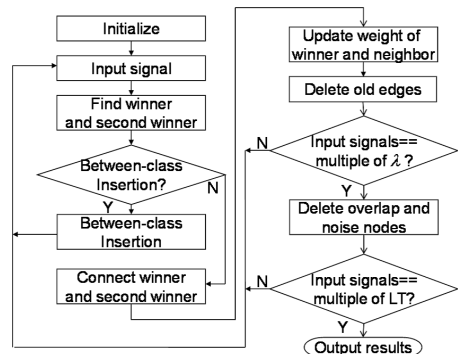


図 4 SOINN のアルゴリズムのフローチャート
Fig. 4 The flowchart of SOINN's algorithm.

をネットワークに挿入する．このときの挿入はクラス間挿入と呼ばれる．なお，本研究で実験者によって与えられる教師 ID は，このクラス間挿入時にノードに付加される．

一方，同一のクラスに属すと判定された場合，第 1 勝者ノードと第 2 勝者ノードの間にエッジを生成し，第 1 勝者ノードとその近傍ノードの重みベクトルを更新する．このときの動作はクラス内挿入と呼ばれる．SOINN はこの 2 種類の挿入法によって，オンラインかつ追加的に学習を行う．

また，SOINN は「エッジの年齢」という基準を用いて，第 1 勝者ノードに接続しているエッジのうち，年齢がしきい値 age_{dead} を超えたエッジを不要と判定し，削除する．更に，入力が λ 回与えられるごとに近傍ノード数が 1 以下のノードをノイズと判定し，削除する．こうすることで，既存のネットワーク構造を破壊することなく，ノイズに影響を受けたと想定されるデータのみネットワークから削除する．SOINN のより詳細なアルゴリズムは [9] を参照されたい．なお，本研究では実験において学習データが少ないことを想定し，ノードを削除する際のノイズの判定基準を近傍ノードが存在しないものへと変更する．

2.4 知識獲得フェーズ

本フェーズの流れを図 5 に示す．ここで「ターム (term)」という概念を導入する．ロボットが現在の実環境を観測している期間を 1 タームとし，同一ターム内では実環境が変化しないものとする．

本フェーズは図 5 に沿って進行する．まず，ロボットはオペレータ実行前の環境として現在の実環境を観測する．ロボットは音声入力 (Sound Input) を開始すると同時に，位置 A から順に画像入力 (Image Input) を行い，位置 C の画像入力を終了すると同時に音声入力を終了する．これで 1 タームが終了する (term1 ends)．1 ターム終了時点でロボットは三つの画像パターンと一つの音声パターンを獲得しており，

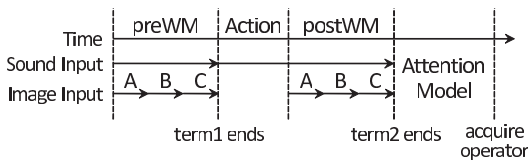


図 5 知識獲得フェーズの流れ：preWM と postWM の WM は実環境のモデル (World Model) を意味している

Fig. 5 The flowchart of knowledge acquisition phase.

これらの状態を事前に定義された述語を用いて記述し，この環境をモデル化する (preWM)．実環境のモデル化についての詳細は 2.4.1 で述べる．

次に，ロボットにオペレータに伴う動作を教示する (Action)．具体的には実験者がロボットの腕をもって動かし，ロボットはその間の腕の動きを 8 次元の時系列データとして記録する．教示する動作によって音が発生する場合に備え，動作の教示を開始すると同時にロボットに音声入力を開始させる．

動作の教示終了後，ロボットはオペレータ実行後の環境として現在の実環境を観測する．ロボットは動作の教示と同時に開始した音声入力を続けながら，位置 A から順に画像入力を行い，位置 C の画像入力を終了すると同時に音声入力を終了する．これで 1 タームが終了する (term2 ends)．ロボットは獲得した情報をもとにこの環境をモデル化する (postWM)．

最後に，ロボットはこの二つの実環境のモデルを「注意のモデル (Attention Model)」に入力し，オペレータを獲得する (acquire operator)．注意のモデルについての詳細は 2.4.2 で述べる．これにより獲得したオペレータは，知識としてシンボル記憶層に保持される．

2.4.1 実環境のモデル化

ロボットは実環境から得られるパターンを 1-NN 法 (1-Nearest Neighbor Method) によって学習済みの概念 (シンボル) に接地する．この処理手順をアルゴリズム A に示す．

アルゴリズム A．入力パターンを概念に接地する

Step1. SOINN 空間内のノード集合 A の中で，入力パターン ξ との最近傍ノードを探索する．なお，以下 W_k をノード k のもつ結合重み， $\|a - b\|$ を a と b のユークリッド距離とする．

$$s = \arg \min_{c \in A} \|\xi - W_c\| \quad (1)$$

Step2. 入力パターン ξ と最近傍ノード s とのユークリッド距離がしきい値 T 以下の場合，入力パターンの概念を既知と判定し，最近傍ノードに付加されている教師 ID を出力する．一方，しきい値 T を超えている場合，入力パターンの概念を未知と判定し， -1 を出力する．なお，以下 $teacherID_k$ をノード k に付加されている教師 ID とする． (終り)

$$\text{if } \|\xi - W_s\| \leq T \text{ then } teacherID_s \text{ else } -1 \quad (2)$$

入力パターンを ID に接地後，物体 x が位置 y にあ

ることを表す $At(x, y)$ という述語と、物体 x の音が発生していることを表す $Ring(x)$ という述語を用いて、物体の状態を記述する．例えば、位置 A で入力された画像パターンが ID = 1 と認識された場合、 $At(1, A)$ と記述し、入力された音声パターンが ID = 2 と認識された場合、 $Ring(2)$ と記述する．

1 ターム終了時点でロボットは位置 A, 位置 B, 位置 C の三つの画像パターンと一つの音声パターンを獲得しており、これらの状態をそれぞれ述語を用いて記述する．この述語の集合を「環境モデル (World Model : 以後 WM)」と表現する．

2.4.2 注意のモデル

一般に人間は環境の変化を注意深く観察することで、物事の因果関係を理解する．注意のモデルはこれと同様にして、ロボットの行動によって生じた実環境の変化から、その因果関係をそれぞれ一つのオペレータで表現する．注意のモデルはオペレータ実行前後の二つの WM から、オペレータの構成に必要な前提条件、削除リスト、追加リストの三つの要素を獲得し、教示された動作情報と合わせてオペレータを構成する．以下、各要素の獲得手順について述べる．

オペレータの削除リスト、追加リストは、その定義から、二つの WM の差集合を考えることで容易に求めることができる．一方、オペレータの前提条件はヒューリスティックに求めるほかはない [11]．本研究ではオペレータ実行前の WM すべてを前提条件とする方法をとる．

また、本研究では未知の物体の状態を表す述語を「不要な述語」とみなし、オペレータの各要素から不要な述語を取り除く処理 (式 (3)) を行う．なお、以下 $\arg(p)$ を述語 p の引数 x とする．

以上から、オペレータの前提条件 $preCond$ 、削除リスト $delList$ 、追加リスト $addList$ を式 (4) のように求める．

$$pack(WM) \triangleq \{p \mid p \in WM, \arg(p) \geq 0\} \quad (3)$$

$$\begin{cases} preCond = pack(preWM) \\ delList = pack(preWM - postWM) \\ addList = pack(postWM - preWM) \end{cases} \quad (4)$$

2.5 問題解決フェーズ

本フェーズ全体のフローチャートを図 6 に示す．本フェーズは主に二つのステップから構成され、プラン

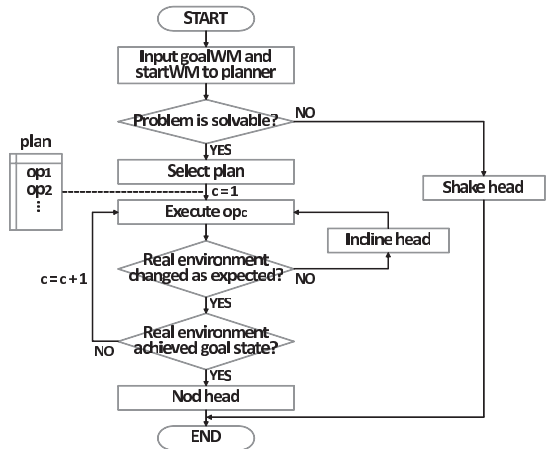


図 6 問題解決フェーズ全体のフローチャート
Fig. 6 The flowchart of the entire problem solving phase.

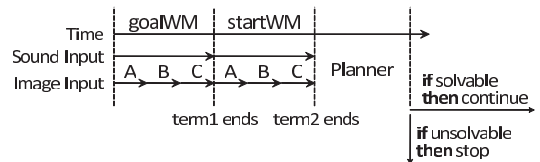


図 7 問題解決フェーズ前半の流れ
Fig. 7 The flowchart of the first half of problem solving phase.

ニングを行うステップと、得られたプラン (オペレータ系列) を実環境において実行する実践ステップとに分かれる．図 6 において、目標状態と初期状態となる WM をプランナに入力している部分が前者に相当し、プランナから得られたプランに含まれるオペレータを繰り返し実行している部分が後者に相当する．

本フェーズの前半は図 7 に沿って進行する．まず、実験者は実環境中でロボットにタスクの目標状態と初期状態を順に提示する (term1 ends, term2 ends)．ロボットはそれぞれの環境を観測し、二つの WM (goalWM, startWM) を得る．そして式 (3) によって不要な述語を取り除いた後、これらをプランナ (Planner) に入力する．プランナは、知識獲得フェーズで獲得したオペレータ集合を用いてプランニングを行う．プランニングについての詳細は 2.5.1 で述べる．

プランナによってタスクが解決可能と判定された場合 (if solvable), ロボットは事前に実験者によって指定された評価基準に従ってプランを選択し、そのプランを実環境において実行する実践ステップに移る

(then continue) . プランナによってタスクが解決不可能と判定された場合 (if unsolvable) , ロボットは首を横に振って実験者にそれを示す (then stop) . プランの評価, 実践ステップについての詳細は, それぞれ 2.5.2, 2.5.3 で述べる .

2.5.1 プランニング

本研究ではプランニングアルゴリズムとして, GPS [8] の手段目標分析 (Means-Ends Analysis : 以後 MEA) を用いる . MEA は現在状態と目標状態の差異を減少させるオペレータを選択し, 現在状態にそのオペレータを適用する処理を繰り返すことで, 目標状態を達成する手法である .

本研究におけるプランニングアルゴリズムは, 標準的な MEA を複数のプランが生成可能なように拡張したものであり, この処理手順をアルゴリズム B に示す . ここで, $plan_n$ はオペレータ系列 (op_1, op_2, \dots, op_n) と, 初期状態 $S (= S_0)$ にそのオペレータ系列を適用した状態 S_n から構成される . $Planning(plan_n, G)$ は, S から S_n を経て G を満たす状態へ変換するオペレータ系列の集合を求める手続きである .

アルゴリズム B . $Planning(plan_n, G)$

Step1. S を初期状態となる WM, G を目標状態となる WM とする . G が S によって満たされているならば $return (\{plan_n\})$

Step2. 差異 $diff$ を G から S を引いた差集合として求める .

Step3. $effectiveOpList$ を差異 $diff$ を減少させるのに有効なオペレータ集合として求める . ここで差異を減少させるのに有効なオペレータとは, 知識獲得フェーズで獲得したオペレータのうち, 追加リストに $diff$ に含まれる述語を含むオペレータを指す .

Step4. $successfulPlanList$ を S から S_n を経て G を満たす状態へ変換するオペレータ系列の集合とし, 空集合に初期化する .

Step5. $effectiveOpList$ が空集合であれば, Step14 へ .

Step6. $effectiveOpList$ からオペレータの一つを選択し, これを op とする .

Step7. $planList1$ を S から S_n を経て op の前提条件を満たす状態へ変換するオペレータ系列の集合として求める . なお, 以下 $preCond_{op}$ を op の前提条件とする .

$$planList1 \leftarrow Planning(plan_n, preCond_{op}) \tag{5}$$

Step8. $planList1$ が空集合であれば, $effectiveOpList$ から op を削除し, Step5 へ .

Step9. $planList1$ からオペレータ系列の一つを選択し, これを $plan_m (m \geq n)$ とする .

Step10. op_{m+1} を op , S_{m+1} を $plan_m$ の S_m に op_{m+1} を適用したものとし, これらから $plan_{m+1}$ を構成する .

Step11. $planList2$ を S から S_{m+1} を経て G を満たす状態へ変換するオペレータ系列の集合として求める .

$$planList2 \leftarrow Planning(plan_{m+1}, G) \tag{6}$$

Step12. $planList2$ が空集合でなければ, $successfulPlanList$ に $planList2$ を加える .

Step13. $planList1$ から $plan_m$ を削除し, Step8 へ .

Step14. $return (successfulPlanList)$ (終り)

2.5.2 プランの評価

ロボットはプランナから複数のプランが得られた場合, 一定の評価基準に従ってプランを選択する . なお, 評価基準の切換は実験者によって行われる .

本研究では事前に以下の二つの評価基準 (コスト) を用意し, ロボットは得られた複数のプランのうち, 実験者によって指定された方のコストがより少ないプランを選択する . 最小コストをもつプランが複数存在する場合は, その中からランダムに一つ選択されるものとする .

- (1) オペレータ数
- (2) 実行時間 (各オペレータの実行時間の総和)

2.5.3 実践ステップ

本フェーズの後半は図 8 に沿って進行する . ロボットは選択したプランに含まれるオペレータを, 初期状

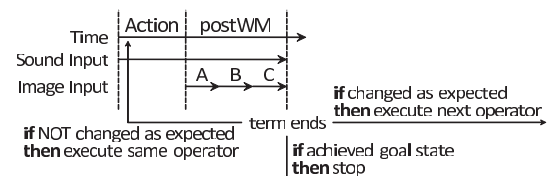


図 8 問題解決フェーズ後半の流れ
Fig. 8 The flowchart of the last half of problem solving phase.

態から順に繰り返し実行する。このとき、ロボットはオペレータの実行 (Action) 後に、実環境を観測することで得られる WM (postWM) と、プランニング結果に従う WM、つまりオペレータ実行後に想定される WM とを比較する必要がある。これはロボットが動作を実行したとしても、必ずしも想定どおりに実環境が変化するとは限らないためである。実環境の観測後 (term ends), ロボットには以下の三つの選択肢が与えられる。

(1) 実環境が想定どおりに変化していない場合 (if NOT changed as expected), ロボットは首を傾げることで実験者にそれを示し、再度同じオペレータの実行を試みる (then execute same operator)。

(2) 実環境が想定どおりに変化し、かつ実環境が目標状態に到達していない場合 (if changed as expected), ロボットは残っている次のオペレータの実行に移る (then execute next operator)。

(3) 実環境が想定どおりに変化し、かつ実環境が目標状態に到達している場合 (if achieved goal state), ロボットはうなずくことで実験者にそれを示す (then stop)。

実践ステップの処理手順をアルゴリズム C に示す。
アルゴリズム C . 実践ステップ

Step1. $startWM$ を初期状態となる WM, $goalWM$ を目標状態となる WM とし、これらは式 (3) によって不要な述語が取り除かれているものとする。 $planWM$ をプランニング結果に従う WM とし、 $planWM$ を $startWM$ に、カウンタ c を 1 に初期化する。

Step2. プランの c 番目のオペレータ op_c に付加されている動作を実行し、 $postWM$ をそのとき実環境を観測することで得られる WM とする。

Step3. $postWM$ から式 (3) によって不要な述語を取り除く。

Step4. $planWM$ に op_c を適用し、 $expectedWM$ を適用後の WM とする。

Step5. 以下の式が成立したらロボットは首を傾げ、Step2 へ。

$$expectedWM \not\subseteq postWM \quad (7)$$

Step6. 以下の式が成立したらロボットはうなずき、Step9 へ。

$$goalWM \subseteq expectedWM \quad (8)$$

Step7. $planWM$ を $expectedWM$ とする。

Step8. c を 1 増やし、Step2 へ。

Step9. 実践ステップを終了とする。(終了)

3. 実験 1: 基本動作の確認

3.1 概要

実験 1 では、提案手法を実装したヒューマノイドロボットを用いて、提案手法の基本動作を確認する。そのため、本実験ではロボットによる少数 (四つ) のオペレータの獲得とそれによる問題解決を扱う。

3.2 記号接地フェーズ

本実験における画像特徴には、グレースケール変換後の 36 次元 DCT 低周波係数を用い、音声特徴にはサンプリング周波数 20 kHz, フレーム長 50 ms による 15 次元スペクトル特徴を用いた。ただし音声については短時間の音について、録音区間内の周波数係数を各次元において最大値を取り、単一ベクトル表現として構成した。各パターン学習に用いる SOINN のパラメータは、ともに $age_{dead} = 10$, $\lambda = 20$ とした。

本実験では図 9 に示す四つの物体を使用した。ベルとドラムは各位置当り 100 回、計 300 回の画像入力と、実験者が音を発生させることで 100 回の音声入力を行った。りんごとみかんは各位置当り 100 回、計 300 回の画像入力を行った。また、表 1 のように教師 ID を与えた。

3.3 知識獲得フェーズ

本実験では実験者のサポートのもと、ロボットに以下の四つのオペレータ (オペレータ 1~4) を獲得させた。なお、オペレータ 2, 4 はそれぞれ、オペレータ 1, 3 と対称の関係にあり、ベルをドラムに、りんごをみかんに置き換えたものである。

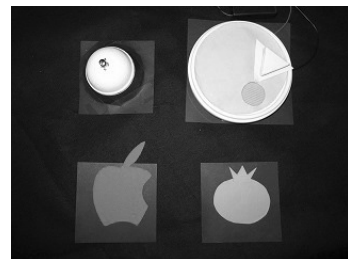


図 9 実験に使用した四つの物体：ベル (左上), ドラム (右上), りんご (左下), みかん (右下)

Fig.9 The four objects we used in the experiment: Bell (left-top), Drum (right-top), Apple (left-bottom), Orange (right-bottom).

表 1 各パターンに付加する教師 ID
Table 1 The teacher ID added to each pattern data.

各パターン	教師 ID
ベル(画像)	0
ドラム(画像)	1
りんご(画像)	2
みかん(画像)	3
ベル(音声)	4
ドラム(音声)	5



図 10 オペレータ 1 を獲得する様子: 左から順に, オペレータ実行前の環境, 動作の教示の様子, オペレータ実行後の環境. 以下, 図 11~13, 19~21 も同様
Fig. 10 The robot acquired the operator 1.



図 11 オペレータ 2 を獲得する様子
Fig. 11 The robot acquired the operator 2.

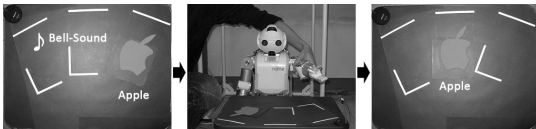


図 12 オペレータ 3 を獲得する様子
Fig. 12 The robot acquired the operator 3.



図 13 オペレータ 4 を獲得する様子
Fig. 13 The robot acquired the operator 4.

- オペレータ 1. ベルを押すと音が鳴る (図 10)
- オペレータ 2. ドラムを押すと音が鳴る (図 11)
- オペレータ 3. ベルの音が鳴っているときに“ちょうだい”をするとりんごが目の前 (位置 B) に移動する (図 12)

オペレータ 4. ドラムの音が鳴っているときに“ちょうだい”をするとみかんが目の前 (位置 B) に移動する (図 13)

2.4.1 で述べたアルゴリズム A の Step2 で用いる

しきい値 T は, 画像パターンに関しては $T = 50$, 音声パターンに関しては $T = 0.03$ とした. 以下, 本研究の実験において, リジェクトを含む各物体の認識率は画像パターン, 音声パターン合わせて平均 97% であった.

3.3.1 オペレータ 1

図 10 にオペレータ 1 を獲得する様子を示す. 左から順に, オペレータ実行前の環境, 動作の教示の様子, オペレータ実行後の環境を表している. まず, 実験者は位置 A にベルを置き, これをオペレータ実行前の環境としてロボットに提示する. ロボットはこれによって, $preWM = \{At(0, A), At(-1, B), At(-1, C), Ring(-1)\}$ を得た. 次に, 実験者はベルを押す動作をロボットに教示する. ロボットはこれによって音が発生したことを観測し, $postWM = \{At(0, A), At(-1, B), At(-1, C), Ring(4)\}$ を得た. ロボットはこの二つの WM を注意のモデルに入力し, オペレータ 1 を獲得した.

3.3.2 オペレータ 2

図 11 にオペレータ 2 を獲得する様子を示す. なお, オペレータ 2 はオペレータ 1 のベルをドラムに置き換えたものである. この際, ロボットは $preWM = \{At(1, A), At(-1, B), At(-1, C), Ring(-1)\}$, $postWM = \{At(1, A), At(-1, B), At(-1, C), Ring(5)\}$ を得た.

3.3.3 オペレータ 3

図 12 にオペレータ 3 を獲得する様子を示す. まず, 実験者は位置 C にりんごを置くとともに, ロボットがオペレータ実行前の環境を観測しているときにベルの音を発生させる. ロボットはこれによって, $preWM = \{At(-1, A), At(-1, B), At(2, C), Ring(4)\}$ を得た. 次に, 実験者は“ちょうだい”をする動作をロボットに教示する. そして実験者は動作の教示後, りんごを位置 B に移動させる. ロボットはこれによって, $postWM = \{At(-1, A), At(2, B), At(-1, C), Ring(-1)\}$ を得た. ロボットはこの二つの WM を注意のモデルに入力し, オペレータ 3 を獲得した.

3.3.4 オペレータ 4

図 13 にオペレータ 4 を獲得する様子を示す. なお, オペレータ 4 はオペレータ 3 のベルをドラムに, りんごをみかんに置き換えたものである. この際, ロボットは $preWM = \{At(-1, A), At(-1, B), At(3, C), Ring(5)\}$, $postWM = \{At(-1, A), At(3, B), At(-1, C), Ring(-1)\}$ を得た.

このフェーズでロボットが獲得した四つのオペレータを表 2 に示す。

3.4 問題解決フェーズ

本実験では汎用のタスクとして、ロボットに以下の三つのタスク(タスク 1~3)を提示した。これらのタスクは、ロボットがタスクを提示される以前に直接的に経験したことがないものである。なお、タスク 2 はタスク 1 と対称の関係にあり、ベルをドラムに、りんごをみかんに置き換えたものである。

タスク 1. ベルを使って目の前(位置 B)にりんごを移動させる(図 14)

タスク 2. ドラムを使って目の前(位置 B)にみかんを移動させる(図 15)

タスク 3. ベルを使って目の前(位置 B)にみかんを移動させる(図 18)

3.4.1 タスク 1

ロボットに図 14 に示す初期状態と目標状態を提示

表 2 実験 1 でロボットが獲得した四つのオペレータ: 以下、表においてオペレータを op と表記する。

Table 2 The four operators acquired by the robot in the experiment 1.

オペレータ名 (実行時間 [sec])	前提条件	削除リスト	追加リスト
op1 (18.68)	$At(0, A)$	-	$Ring(4)$
op2 (20.16)	$At(1, A)$	-	$Ring(5)$
op3 (23.93)	$At(2, C)$ $Ring(4)$	$At(2, C)$ $Ring(4)$	$At(2, B)$
op4 (20.46)	$At(3, C)$ $Ring(5)$	$At(3, C)$ $Ring(5)$	$At(3, B)$

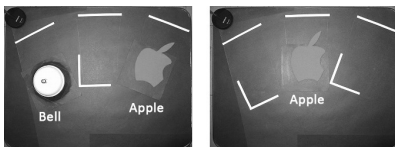


図 14 タスク 1 の初期状態(左)と目標状態(右)
Fig. 14 The initial state (left) and the goal state (right) for the task 1.

した。ロボットはこれらの WM をプランナに入力し、このタスクを解決可能と判定した。このときにプランナから得られたプランニング結果を表 3 に示す。ロボットはこの結果をもとに実践ステップにてオペレータを実行し、最終的に目標状態に到達できた。図 16 にその様子を示す。左から順に、初期状態(ターム 1 の環境)、オペレータ 1 の実行の様子、ターム 2 の環境、オペレータ 3 の実行の様子、ターム 3 の環境、目標状態に到達でき、うなずいている様子を表している。

表 3 タスク 1 に対するプランニング結果: 今回得られたプランは、ターム 1 の WM が $At(0, A), At(2, C)$ 、それに op1 を適用して、ターム 2 の WM が $At(0, A), At(2, C), Ring(4)$ 、それに op3 を適用して、ターム 3 の WM が $At(0, A), At(2, B)$ 、という内容である

Table 3 The result of planning for the task 1.

ターム	WM, または適用するオペレータ
1	$At(0, A), At(2, C)$ op1
2	$At(0, A), At(2, C), Ring(4)$ op3
3	$At(0, A), At(2, B)$

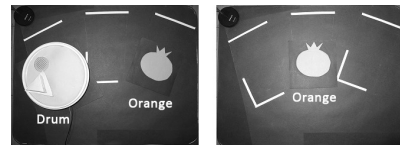


図 15 タスク 2 の初期状態(左)と目標状態(右)
Fig. 15 The initial state (left) and the goal state (right) for the task 2.

表 4 タスク 2 に対するプランニング結果
Table 4 The result of planning for the task 2.

ターム	WM, または適用するオペレータ
1	$At(1, A), At(3, C)$ op2
2	$At(1, A), At(3, C), Ring(5)$ op4
3	$At(1, A), At(3, B)$

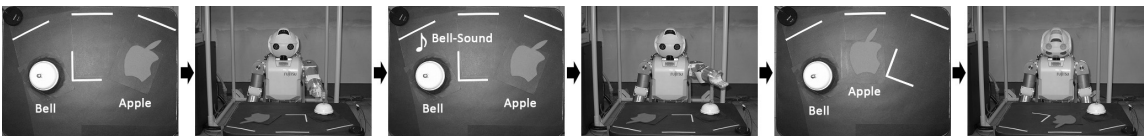


図 16 タスク 1 において、ロボットが最終的に目標状態に到達できた様子: 左から順に、初期状態(ターム 1 の環境)、オペレータ 1 の実行の様子、ターム 2 の環境、オペレータ 3 の実行の様子、ターム 3 の環境、目標状態に到達でき、うなずいている様子

Fig. 16 The robot finally achieved the goal state in the task 1.

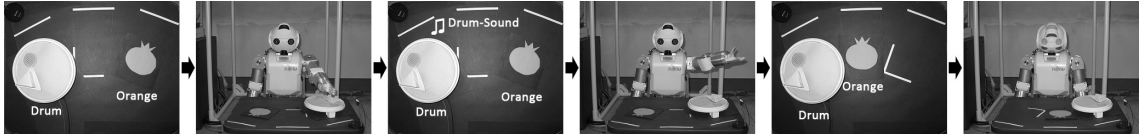


図 17 タスク 2 において、ロボットが最終的に目標状態に到達できた様子：左から順に、初期状態（ターン 1 の環境）、オペレータ 2 の実行の様子、ターン 2 の環境、オペレータ 4 の実行の様子、ターン 3 の環境、目標状態に到達でき、うなずいている様子

Fig. 17 The robot finally achieved the goal state in the task 2.

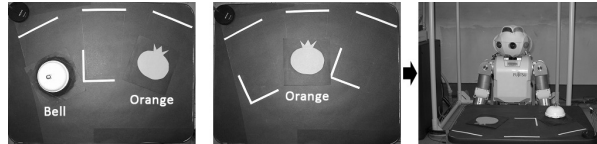


図 18 ロボットがタスク 3 を解決不可能と判定した様子：左から順に、初期状態、目標状態、このタスクを解決不可能と判定し、首を横に振っている様子

Fig. 18 The robot judged to be unable to solve the task 3.

3.4.2 タスク 2

ロボットに図 15 に示す初期状態と目標状態を提示した。タスク 1 と同様に、ロボットは表 4 に示すプランニングの結果をもとに実践ステップにてオペレータを実行し、最終的に目標状態に到達できた。図 17 にその様子を示す。

3.4.3 タスク 3

図 18 の左、中央をそれぞれ初期状態、目標状態とするタスクは、知識獲得フェーズで獲得したオペレータでは解決できない。ロボットはプランナによってこのタスクを解決不可能と判定し、首を横に振ってそれを示した。図 18 にその様子を示す。

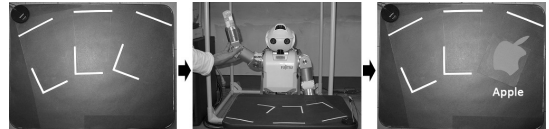


図 19 オペレータ 5 を獲得する様子
Fig. 19 The robot acquired the operator 5.



図 20 オペレータ 7 を獲得する様子
Fig. 20 The robot acquired the operator 7.

4. 実験 2：知識のオンラインかつ追加的な獲得による問題解決能力の向上

4.1 概要

実験 2 では、実験 1 で用いたロボットに対して、知識のオンラインかつ追加的な獲得による問題解決能力の向上を確認する。本実験では、実験 1 と比較して知識の組合せ数が多く、更に複数の組合せ方法が存在するタスクを扱う。なお、本実験は実験 1 に続いて行われ、本実験で用いるロボットは実験 1 で形成したパターン記憶層とシンボル記憶層における記憶を保持している。

4.2 知識獲得フェーズ

本実験では実験 1 の四つのオペレータ（オペレータ 1～4）に加え、ロボットに更に六つのオペレータ（オペレータ 5～10）を獲得させた。このうち、オペレータ

タ 5, 7, 9 の内容を以下に示す。なお、実験 1 と同様に、オペレータ 6, 8, 10 はそれぞれ、オペレータ 5, 7, 9 と対称の関係にあり、これらのオペレータと同様の結果になったため説明は省略する。

オペレータ 5. 右手を挙げるとりんごが手元（位置 C）に置かれる（図 19）

オペレータ 7. “だだをこねる” とりんごが目の前（位置 B）に移動する（図 20）

オペレータ 9. 左手を挙げるとベルが手元（位置 A）に置かれる（図 21）

4.2.1 オペレータ 5

図 19 にオペレータ 5 を獲得する様子を示す。この際、ロボットは $preWM = \{At(-1, A), At(-1, B), At(-1, C), Ring(-1)\}$, $postWM = \{At(-1, A), At(-1, B), At(2, C), Ring(-1)\}$ を得た。

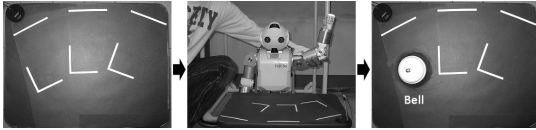


図 21 オペレータ 9 を獲得する様子
Fig. 21 The robot acquired the operator 9.

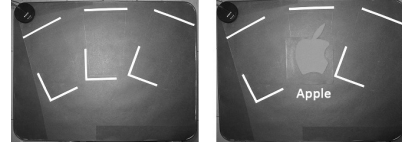


図 22 タスク 4 の初期状態 (左) と目標状態 (右)
Fig. 22 The initial state (left) and the goal state (right) for the task 4.

表 5 実験 2 でロボットが獲得した六つのオペレータ
Table 5 The six operators acquired by the robot in the experiment 2.

オペレータ名 (実行時間 [sec])	前提条件	削除リスト	追加リスト
op5 (16.34)	-	-	At(2, C)
op6 (16.31)	-	-	At(3, C)
op7 (74.88)	At(2, C)	At(2, C)	At(2, B)
op8 (71.73)	At(3, C)	At(3, C)	At(3, B)
op9 (20.06)	-	-	At(0, A)
op10 (29.03)	-	-	At(1, A)

4.2.2 オペレータ 7

図 20 にオペレータ 7 を獲得する様子を示す。この際、ロボットは $preWM = \{At(-1, A), At(-1, B), At(2, C), Ring(-1)\}$, $postWM = \{At(-1, A), At(2, B), At(-1, C), Ring(-1)\}$ を得た。

4.2.3 オペレータ 9

図 21 にオペレータ 9 を獲得する様子を示す。この際、ロボットは $preWM = \{At(-1, A), At(-1, B), At(-1, C), Ring(-1)\}$, $postWM = \{At(0, A), At(-1, B), At(-1, C), Ring(-1)\}$ を得た。

このフェーズでロボットが獲得した六つのオペレータを表 5 に示す。

4.3 問題解決フェーズ

本実験ではロボットに五つのタスク (タスク 4~7, 3') を提示した。このうち、タスク 4, 6, 3' の内容を以下に示す。なお、実験 1 と同様に、タスク 5, 7 はそれぞれ、タスク 4, 6 と対称の関係にあり、これらのタスクと同様の結果になったため説明は省略する。また、タスク 3' は実験 1 においてロボットが解決不可能と判定したタスク 3 と同じ内容である。

タスク 4. テーブルの上にも何も置かれていない状態から目の前 (位置 B) にりんごを置く (図 22)

タスク 6. テーブルの上にも何も置かれていない状態から手元 (位置 A) にりんごを置く (図 26)

タスク 3'. ベルを使って目の前 (位置 B) にみかんを移動させる (図 27)

4.3.1 タスク 4

ロボットに図 22 に示す初期状態と目標状態を提示

表 6 タスク 4 に対するプランニング結果
Table 6 The result of planning for the task 4.

ターム	WM, または適用するオペレータ		
	プラン 1	プラン 2	プラン 3
1	-	-	-
2	op5 At(2, C)	op5 At(2, C)	op9 At(0, A)
3	op7 At(2, B)	op9 At(0, A) At(2, C)	op1 At(0, A) Ring(4)
4		op1 At(0, A) At(2, C) Ring(4)	op5 At(0, A) At(2, C) Ring(4)
5		op3 At(0, A) At(2, B)	op3 At(0, A) At(2, B)

表 7 各プランの評価
Table 7 The evaluation of each plan.

評価基準	プラン 1	プラン 2	プラン 3
オペレータ数	2	4	4
実行時間 [s]	91.22	79.01	79.01

した。ロボットはこれらの WM をプランナに入力し、このタスクを解決可能と判定した。このときにプランナから得られたプランニング結果を表 6 に示す。このように、今回のプランニング結果には三つのプランが含まれていたため、ロボットは事前に指定された評価基準に従ってプランを一つ選択した。具体的には、2.5.2 で述べたように、本実験では (1) オペレータ数 (2) 実行時間という二つの評価基準を用意した。その結果、表 7 に示すように、オペレータ数を評価基準として指定した場合プラン 1 が選択され、実行時間を評価基準として指定した場合プラン 2, またはプラン 3 が選択された。以下では、各プランの実践ステップの結果について述べる。

(1) ロボットはプラン 1 をもとに実践ステップにてオペレータを実行し、最終的に目標状態に到達できた。図 23 にその様子を示す。

(2) ロボットはプラン 2 をもとに実践ステップにてオペレータを実行し、最終的に目標状態に到達でき

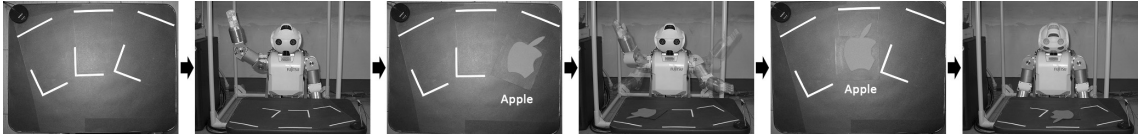


図 23 タスク 4 のプラン 1 において、ロボットが最終的に目標状態に到達できた様子：左から順に、初期状態（ターン 1 の環境）、オペレータ 5 の実行の様子、ターン 2 の環境、オペレータ 7 の実行の様子、ターン 3 の環境、目標状態に到達でき、うなずいている様子

Fig. 23 The robot finally achieved the goal state in the plan 1 for the task 4.

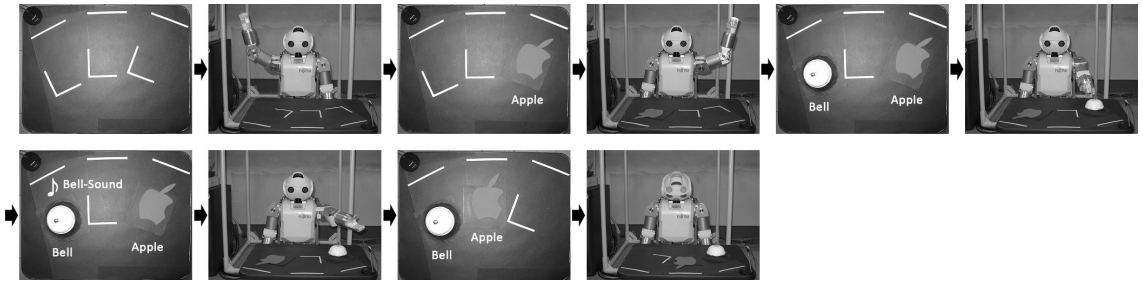


図 24 タスク 4 のプラン 2 において、ロボットが最終的に目標状態に到達できた様子：左上から順に、初期状態（ターン 1 の環境）、オペレータ 5 の実行の様子、ターン 2 の環境、オペレータ 9 の実行の様子、ターン 3 の環境、オペレータ 1 の実行の様子、ターン 4 の環境、オペレータ 3 の実行の様子、ターン 5 の環境、目標状態に到達でき、うなずいている様子

Fig. 24 The robot finally achieved the goal state in the plan 2 for the task 4.

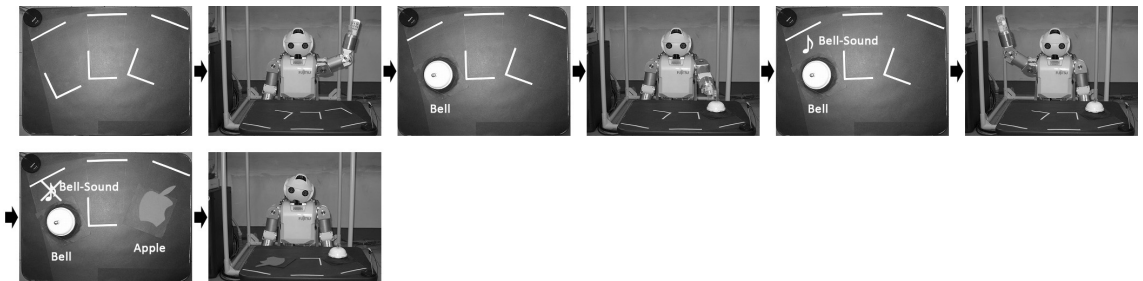


図 25 タスク 4 のプラン 3 において、ロボットが最終的に目標状態に到達できなかった様子：左上から順に、初期状態（ターン 1 の環境）、オペレータ 9 の実行の様子、ターン 2 の環境、オペレータ 1 の実行の様子、ターン 3 の環境、オペレータ 5 の実行の様子、ターン 4 の環境、実環境が想定どおりに変化せず、首を傾げている様子

Fig. 25 The robot didn't finally achieve the goal state in the plan 3 for the task 4.

た．図 24 にその様子を示す．

(3) ロボットはプラン 3 をもとに実践ステップにてオペレータを実行したが、その途中、実環境がプランニング結果どおりに変化せず、最終的に目標状態に到達できなかった．図 25 にその様子を示す．ここで表 6 に示すように、ロボットがオペレータ 5 の実行後にターン 4 として想定した WM は $\{At(0, A), At(2, C), Ring(4)\}$ であったが、実際に

観測した WM は $\{At(0, A), At(2, C)\}$ であった．そのためロボットはその後、この環境に対してオペレータ 5 の実行を繰り返したが、結果的に想定どおりの環境に到達できなかった．

4.3.2 タスク 6

図 26 の左、中央をそれぞれ初期状態、目標状態とするタスクは、実験 1, 2 で獲得したオペレータでは解決できない．ロボットはプランナによってこのタス

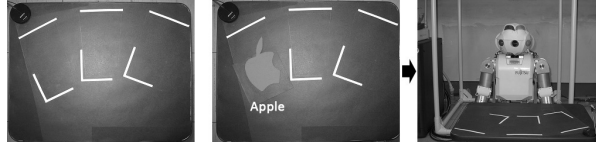


図 26 ロボットがタスク 6 を解決不可能と判定した様子：左から順に、初期状態，目標状態，このタスクを解決不可能と判定し，首を横に振っている様子

Fig. 26 The robot judged to be unable to solve the task 6.

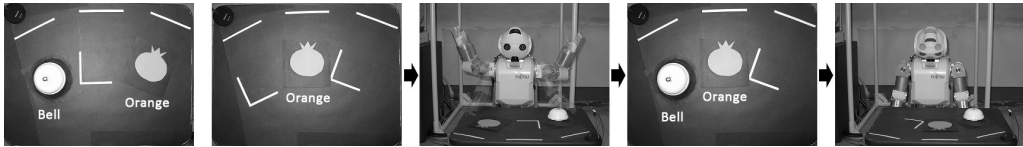


図 27 タスク 3' のプラン 1 において，ロボットが最終的に目標状態に到達できた様子：左から順に，初期状態（タム 1 の環境），目標状態，オペレータ 8 の実行の様子，タム 2 の環境，目標状態に到達でき，うなずいている様子．ここで，ベルとみかんという組合せは獲得したオペレータには存在しないことに注意されたい．

Fig. 27 The robot finally achieved the goal state in the plan 1 for the task 3'.

表 8 タスク 3' に対するプランニング結果

Table 8 The result of planning for the task 3'.

タム	WM, または適用するオペレータ	
	プラン 1	プラン 2
1	$At(0, A)$	$At(0, A)$
	$At(3, C)$	$At(3, C)$
	op8	op10
2	$At(0, A)$	$At(0, A)$
	$At(3, B)$	$At(1, A)$
		$At(3, C)$
3		op2
		$At(0, A)$
		$At(1, A)$
		$At(3, C)$
		$Ring(5)$
		op4
4		$At(0, A)$
		$At(1, A)$
		$At(3, B)$

クを解決不可能と判定し，首を横に振ってそれを示した．図 26 にその様子を示す．

4.3.3 タスク 3'

図 27 の左から 1 枚目，2 枚目をそれぞれ初期状態，目標状態とするタスクは，実験 1 においてロボットが解決不可能と判定したものである．しかし本実験においては，ロボットは実験 1, 2 で獲得した複数のオペレータを複合的に運用することで，このタスクを解決可能と判定した．このときにプランナから得られたプランニング結果を表 8 に示す．このうち，プラン 1 の実践ステップの様子を図 27 に示す．ロボットはプラン 1 をもとに実践ステップにてオペレータを実行し，

最終的に目標状態に到達できた．

5. 考 察

提案手法は Weng らの示す性質をすべて満たし，1, 2 で述べた優位性をもつ．また，提案手法を実装したヒューマノイドロボットを用いた実験により，提案手法が実世界における汎用のタスクに対して有効であることを示した．具体的には，各実験により提案手法を実装したロボットは以下の特長をもつことが確認できた．

(1) ロボットは直接的に経験したことのないタスクに対しても適切に行動できる：ここでいう行動とは，自らの問題解決能力では実行できないタスクに対して，実行できないことを示す行為も含まれる．

例えば，実験 1 のタスク 1, 2 に関しては，ロボットがタスクを提示される以前に直接的に経験したことのないものであるが，ロボットはこれらのタスクを実行することができた．一方，タスク 3 に関しては，ロボットはこのタスクが実験 1 で獲得したオペレータでは実行できないことを示した．

(2) ロボットは知識をオンラインかつ追加的に獲得することで自らの問題解決能力を向上させることができる：具体的には，以前より複雑なタスク（オペレータの組合せ数が多いものや複数のプランが存在するもの）や，以前に実行できなかったタスクを実行できるようになる．

例えば，実験 2 のタスク 4, 5 に関しては，実験 1 の

タスク 1, 2 と比較してオペレータの組合せ数が多く、更に複数のプランが存在するものであるが、ロボットはこれらのタスクに対して適切に行動できた。また、タスク 6, 7 に関しては、ロボットはこれらのタスクが実験 1, 2 で獲得したオペレータでは実行できないことを示した。タスク 3' に関しては、実験 1 においてロボットが解決不可能と判定したものであるが、ロボットは実験 1, 2 で獲得した複数のオペレータを複合的に運用することでこのタスクを実行できた。

以下、まず 5.1 で提案手法の構成要素について議論する。次に、5.2 で提案手法の適用可能性（スケーラビリティ）について議論する。最後に、5.3 で提案手法の今後の課題について議論する。

5.1 提案手法の構成要素の検証

本研究で提案したアーキテクチャは、主に以下の三つの構成要素からなる。

- (1) 知覚情報のシンボル化
- (2) 因果関係の知識化
- (3) プランニングと実践

5.1.1 知覚情報のシンボル化

提案アーキテクチャの入力層の第 1 フェーズである記号接地フェーズでは、ロボットは SOINN を用いて知覚情報（画像・音声情報）から概念となるシンボルを形成した。このシンボル化というプロセスは、人間がロボットと認識を共有するという点で非常に重要な意味をもつ。人間とロボットが実世界から得られるパターンに対して共通の認識をもっていることは、実世界において人間とロボットがコミュニケーションを図るための重要な要件である。筆者らはこれまでに、ロボットが獲得した概念と言語を関連づけて、人間とロボットのコミュニケーションを実現するシステムについても研究を行っている [12]。

ここでこのシンボル化に関しては、提案手法で用いている SOINN 以外にも Self-Organizing Map (SOM) を用いた手法も提案されている [13]。また、シンボル化の対象が時系列相関が存在する実環境の変化であるとすれば、時系列データの取扱いが可能な Hidden Markov Model (HMM), Recurrent Neural Network with Parametric Bias (RNNPB) を用いた手法の利用も考えられる [14], [15]。しかし、一般にこれらの手法はいずれもノード数や状態数といった内部構造をあらかじめ決めておく必要があるため、オンライン追加学習に適さない。なお、これらの手法の中にはオンライン追加学習が原理的に可能なものもあるが、[16] で

指摘されているように HMM によるものは初期認識精度や追加学習による認識精度の向上に対して問題がある。また、RNN を用いたオンライン追加学習に関する研究 [17] では、不完全な内部モデルが構築され、ロボットの挙動が不安定になるといった問題が見られている。これに対し、SOINN によるオンライン追加学習ではあらかじめ内部構造を決めておく必要がなく、更に高いノイズ耐性も期待できる。このような性質は、Weng らの示す性質 IV, V. を満たすのに必要不可欠である。

本研究の実験においても、SOINN の性質はパターン認識において有効であった。実験で扱った認識対象は、画像、音声に関してそれぞれ四つ、二つと少数で、実験環境も理想化されてはいたが、オンラインかつ追加的に学習した物体の認識率は実験の進行に支障を来さない程度（平均 97%）に収めることができた。ただし、画像認識に関しては物体を図 3 に示す三つの位置に一定の向きで配置する必要があるが、物体画像のアフィン変換等には対処できない。また、音声認識に関しても非常に静かな環境で行われ、雑音環境下では認識精度が下がる。しかしながら、こういったパターン認識の技術は知能ロボットの基盤技術として重要な役割を果たす。本研究では知能アーキテクチャの提案が研究の主旨であるという理由から、ロボットのパターン認識機能をシンプルな手法で実装したが、今後パターン認識機能の強化も主要な課題の一つとして検討していく必要がある。

5.1.2 因果関係の知識化

提案アーキテクチャの入力層の第 2 フェーズである知識獲得フェーズでは、ロボットは自らの行動によって生じた実環境の変化から、その因果関係を知識として獲得した。提案手法では、ロボットはこうした知識をオンラインかつ追加的に獲得することによって、自らの問題解決能力を向上させることができる。ここで、提案手法で獲得する知識は必ずしも特定のタスクと 1 対 1 に対応しているわけではない。提案手法では、ロボットは獲得した知識から新たな知識を生成することもでき、これは自らの問題解決能力を“自力で”向上できることを意味する。つまり、ここに Weng らのいう“知能発達”が実現されている。

この知能発達の実現は、提案手法において知識をオペレータとして表現していることが大きな要因となっている。1. で述べたように、従来の知能ロボットにおける知識は if-then ルールとして組み込まれてき

た．ここで知識という観点からの、if-then ルールと提案手法におけるオペレータとの明確な違いは、知識の応用性、つまり外部から追加的に知識を与えられることなく、既存の知識のみからロボットが自らの知的に発達できるかという点にある．一般に if-then ルールはある特定のタスクを解決するための知識にすぎず、いわばそのタスクに閉じた知識といえる．たとえこの if-then ルールを追加的に獲得するシステムを構築したとしても、そのシステムの問題解決能力として獲得した知識をそのまま再現する以上のことは期待できない．一方、提案手法におけるオペレータは、それを獲得した後に行われる問題解決を見据えた知識である．言い換えれば、提案手法における知識はオペレータの形式をしていることで、既存のプランニング技術（プランナ）を用いて運用できる．つまりこれはロボットが“自力で”新たな知識を生成し、結果として問題解決能力を自ら向上できるということであり、Weng らの示す性質 III. を満たすことができる．

また、因果関係の知識化という観点からは、提案手法において知識となるオペレータは、オペレータ実行前後の実環境をモデル化し、それらを比較することで得られる．ここで、実環境の変化を時系列データととらえるならば、HMM 等を利用して実環境の変化を表現するアプローチも考えられる．確かに特定の時系列データを単にシンボルとして扱うだけならば、こうした手法でも実現可能である．しかし、提案手法では知識を獲得するだけでなく、それが問題解決に利用可能であることが重要であるため、HMM 等のように知識を獲得するまでに留まる手法は有効ではない．先にも述べたように、獲得した知識を問題解決に利用するには、それがオペレータの形式をしている必要がある．しかし HMM 等によって獲得されたシンボルは、提案手法における環境モデルのように各要素に分割して、それらをダイナミックに組み換えることができないため、これを実現できない．つまり、そのようにして獲得された知識は特定の問題解決のための情報にすぎず、汎用の問題解決に利用する知識としては適していない．以上の点から、本研究で提案する因果関係の知識化の手法はシンプルでありながら有効な手法といえる．

5.1.3 プランニングと実践

提案アーキテクチャの入力層の第 3 フェーズである問題解決フェーズでは、ロボットは実環境の汎用のタスクに対して、獲得したオペレータを運用してプランニングを行い、その結果を実際の行動として実践する

ことができた．ここで、実践ステップにおいてロボットが実環境に対して単に行動を示すだけでなく、行動による実環境の変化の予測とその結果の確認を繰り返している点に注意されたい．すなわち、1. で述べたようにロボットの身体性はロボットが知識を獲得する過程で重要であるが、更にロボットが知識を行動として表現する過程においても、身体性に基づく行動の結果の確認を導入することで、実環境においてロボットが確実に目標状態に到達できるようになる．

次に、プランニングと実践ステップの実験結果について議論する．第 1 にプランニングについては、その計算時間はプランナの性能や獲得したオペレータ数に依存するが、実験では実時間で計算することができた．つまり、ロボットはタスクを提示されてから計算のために静止することなく、即座に行動に移ることができた．ただし、獲得したオペレータ数が増えるにつれて計算時間も増えていくことが想定されるため、これについては今後検討する必要がある．この点については 5.3 で更に言及する．第 2 に実践ステップについては、タスク 4 のプラン 1 とプラン 2 は、プランニング結果を実環境において正確に再現することができ、目標状態に到達できた．一方プラン 3 では、その実践ステップにおいてプランニング結果とは異なる事態が発生し、ロボットはそれに対処できず、最終的に目標状態に到達できなかった．これは、提案手法が実環境を静的と仮定していることに起因する．実際には実環境は動的であり、実践ステップのステップ数が長くなるほど過去の環境が変化する可能性がある．具体的にはタスク 4 のプラン 3 において、ターム 3 で達成された *Ring(4)* がターム 4 で解除されてしまった（図 25）．このような不測の事態に対しては、ロボットは単に特定のオペレータの実行を繰り返すだけでなく、その原因を究明し、再プランニングの必要性について吟味する必要がある．この点についても今後検討する必要がある．

5.2 提案手法の適用可能性の検証

知能ロボットの研究において、システムの適用可能性は重要な論点となっている．本研究においては、提案手法が実世界のどの程度の規模のタスクにまで対応できるかという点が主な論点となる．

まず、提案手法を実装する対象に関しては、本研究の実験ではヒューマノイドロボットに実装したが、提案手法はそれ以外の形態のロボット（移動ロボット等）にも実装可能である．提案手法は実装する対象のもつ

機能に合わせて容易に拡張可能であり、この点は 1.2 で述べたように、提案手法の主要な優位性の一つである。

次に、プランニングの規模に関しては、本研究の実験では二つのオペレータの組合せ、かつプランが一つしか存在しないもの（表 3、表 4）から、最大四つのオペレータの組合せ、かつ三つのプランが存在するもの（表 6）まで扱った。提案手法では既存のプランナを用いてプランニングを行っているため、計算時間を考慮しなければオペレータの組合せ数に原理的に制限はなく、複数のプランが存在するものにも対応できる。また、提案手法ではプランナがモジュールとして独立して動作するため、少なくともプランナ本来の性能を期待できる。

つまり、提案手法の適用可能性を議論する上で最大の問題となるのは、タスクや知識をどのようにモデル化するかということである。具体的には、主に以下の三つの問題が挙げられる。

第 1 に、ロボット自身が置かれている環境（実環境）をどこまで記述（モデル化）する必要があるかという問題がある。これはフレーム問題に関連するもので、提案手法では注意の範囲というフレームを用意することで、この問題を擬似的に回避している。

第 2 に、実環境をモデル化する際に用いる述語の不足に関する問題がある。提案手法では物体の状態を表す述語として *At*, *Ring* という二つの述語を事前に定義したが、つまりこれはロボットがこの二つ以外の物体の状態を観測できないことを意味する。例えば、ロボットが「スイッチを押すと灯りがつく」という知識を獲得しようとしたとき、提案手法では灯りがついていることを表す *Light* といった述語が定義されていないため、この現象を観測できない。

第 3 に、これまでに議論してきた理由以外でモデル化しがたいタスクや知識が存在するという問題がある。例えば、「困っている人がいたら助ける」といった抽象度の高いタスク、または知識は、そのタスクの初期状態や目標状態、またはその知識を表現するオペレータの各要素を表現しがたい。

以上のように、提案手法を実世界における一般問題解決システムとして考えた場合、依然解決すべき課題が数多く残されている。筆者らは本研究を実世界において知的に振る舞うヒューマノイドロボットの実現に向けた基礎研究ととらえ、今後も議論を重ねていく予定である。

5.3 今後の課題

5.1, 5.2 で述べてきた課題に加え、提案手法には更に以下のような点が求められる。

- 知識の整理が求められる。ロボットが獲得したオペレータの中には、似通ったものや要素に不必要な述語を含むものが存在することが想定される。このようなオペレータは、プランニング時の計算時間の増大や、実践ステップにおけるプランニング結果の再現性の低下を招く。しかしながら、2.4.2 で述べたように、ロボットがオペレータの内容を一度で正確に獲得する方法はない。つまりオペレータはロボットが試行錯誤的に獲得する必要がある、その上で獲得したオペレータを的確に整理・統合できないか現在検討中である。

- 注意の範囲の拡大が求められる。提案手法ではロボットの注意の範囲を事前に定義した位置に制限した。しかし実世界においては、ロボットはより広い範囲を観測できることが求められる。

- 処理の連続性が求められる。現状では実験者がタムやフェーズの区切りをロボットに与えることで時間を離散的に扱っているが、今後は連続的な時間においてロボット自身がこの区切りを判定できることが望ましい。こうすることで、ロボットは実験者から一方的に教示を受けるばかりでなく、自らの能動的な行動からも知識を獲得できるようになる。

6. む す び

本研究では、実世界における一般問題解決システムの構築を目的とし、既存のプランナを利用した 3 層構造のアーキテクチャを提案した。提案手法は SOINN により物体の概念を形成し、また、環境や人間との相互作用を通じて行動の因果関係をオペレータとしてオンラインかつ追加的に獲得する。更に、獲得した複数のオペレータを既存のプランナにより能動的に運用し、実世界における汎用的な問題解決が可能である。提案手法は Weng らの示す性質をすべて満たし、1.2 で述べた優位性をもつ。

本研究では既存のプランナとして GPS を使用し、提案手法を実装したヒューマノイドロボットを用いた実験により、提案手法が実世界における汎用のタスクに対して有効であることを示した。具体的には、各実験により提案手法を実装したロボットが (1) 直接的に経験したことのないタスクに対しても適切に行動でき (2) 知識をオンラインかつ追加的に獲得することで自らの問題解決能力を向上できることが確認できた。

現在, 5.3 に挙げられるような課題を克服し, 実世界において知的に振る舞うヒューマノイドロボットの実現に向けて研究中である.

謝辞 本研究は NEDO 産業技術研究助成事業から支援を頂きました. 記して感謝致します.

文 献

- [1] M. Ito, K. Noda, Y. Hoshino, and J. Tani, "Dynamic and interactive generation of object handling behaviors by a small humanoid robot using a dynamic neural network model," *Neural Netw.*, vol.19, pp.323-337, 2006.
- [2] 比留川博久, "人間型ロボットの近未来応用," *日本ロボット学会誌*, vol.22, no.1, pp.6-9, 2004.
- [3] 浅田 稔, 國吉康夫, "認知発達ロボティクス," *ロボットインテリジェンス*, pp.304-328, 岩波書店, 東京, 2006.
- [4] 國吉康夫, "R11: 創発・発達知能システム情報学 (國吉康夫)," *ロボット分野に関するアカデミック・ロードマップ*, pp.II 77-II 86, March, 2007. (オンライン), 入手先 < <http://www.ai-gakkai.or.jp/jsai/activity/roadmap.html> > (参照 2009-08-22).
- [5] J. Weng, J. McClelland, A. Pentland, O. Sporns, I. Stockman, M. Sur, and E. Thelen, "Autonomous mental development by robots and animals," *Science*, vol.291, no.5504, pp.599-600, 2001.
- [6] 小倉 崇, 羽根田淳, 岡田 慧, 稲葉雅幸, "自律行動の誘導に基づく等身大ヒューマノイドの全身動作のオンライン教示システム," 第 23 回ロボット学会学術講演会予稿集, 1F25, 2005.
- [7] S. Harnad, "The symbol grounding problem," *Physica D*, vol.42, pp.335-346, 1990.
- [8] G.W. Ernst and A. Newell, *GPS: A Case Study in Generality and Problem Solving*, Academic Press, New York, 1969.
- [9] F. Shen and O. Hasegawa, "An incremental network for on-line unsupervised classification and topology learning," *Neural Netw.*, vol.19, pp.90-106, 2006.
- [10] J. McCarthy and P.J. Hayes, "Some philosophical problems from the standpoint of artificial intelligence," *Machine Intelligence*, vol.4, pp.463-502, 1969.
- [11] 下畑光夫, 山田誠二, 安部憲広, 辻 三郎, "状態記述系列からのオペレータの生成," *情報処理学会第 41 回全国大会論文集 (3)*, 5L-1, pp.55-56, 1990.
- [12] 佐藤彰洋, 賀 小淵, 小倉和貴, 長谷川修, "人間との相互作用に基づくヒューマノイドロボット上の語順と挙動のオンライン学習," *信学論 (D)*, vol.J91-D, no.8, pp.2045-2060, Aug. 2008.
- [13] 中川 満, 肥川宏臣, 平林武憲, "自己組織化マップによる自律移動ロボットの位置認識手法," *信学技報*, NLP2004-92, 2005.
- [14] 坂本和之, 稲邑哲也, 古城直樹, 園田朋之, 岡田 慧, 稲葉雅幸, "HMM によるジェスチャと音声指示のシンボル化に基づくヒューマノイドの行動学習・生成," *信学技報*, NLC2005-39, PRMU2005-66, 2005.
- [15] 尾形哲也, "人工神経回路モデルによるインタラクション創発システム実現に向けて," *日本神経回路学会誌*, vol.14, no.4, pp.282-292, 2007.
- [16] 篠田浩一, "確率モデルによる音声認識のための話者適応化技術," *信学論 (D-II)*, vol.J87-D-II, no.2, pp.371-386, Feb. 2004.
- [17] J. Tani and J. Yamamoto, "On the dynamics of robot exploration learning," *Cognitive Systems Research*, vol.3, no.3, pp.459-470, 2002.

(平成 21 年 8 月 25 日受付, 12 月 21 日再受付)

巻 瀧 有 哉 (学 生 員)



2009 東工大・工・情報工学卒. 現在, 同大学院総合理工学研究科知能システム科学専攻修士課程に在籍中. ヒューマノイドロボットの知能情報処理に興味をもつ.

申 富 饒



1995 中国南京大・数学卒. 1998 同数学専攻卒. 2006 東京工業大学大学院総合理工学研究科知能システム科学専攻博士課程了. 現在, 中国南京大学計算機ソフトウェア新技術国家重点実験室准教授.

長 谷 川 修 (正 員)



1993 東京大学大学院博士課程了. 博士 (工学). 同年, 電子技術総合研究所入所. 1999 年 6 月より 1 年間カーネギーメロン大学ロボティクス研究所滞在研究員. 2001 産業技術総合研究所主任研究員. 2002 年 5 月東京工業大学大学院理工学研究科付属像情報工学研究施設准教授. 2002 から 3 年間科技団さきかけ研究 21 研究員. パターン認識, ニューラルネット, 実世界知能システムなどの研究に従事. 人工知能学会, 日本認知科学会, 日本顔学会, IEEE CS 等各会員.