

自律移動ロボットのための自然言語タスクプログラミング

Natural Language-Based Task Programming for Autonomous Mobile Robots

板谷 純希*¹ 中村 友昭*² 長井 隆行*¹
Junki Itaya Tomoaki Nakamura Takayuki Nagai

*¹電気通信大学大学院情報理工学研究所

Faculty of Informatics and Engineering, The University of Electro-Communications

*²電気通信大学大学院電気通信学研究所

Faculty of Electro-Communications, The University of Electro-Communications

In general, robots can execute only pre-programmed sets of actions. In order to make the robot truly useful in our living environments, robots are expected to carry out complex tasks in a flexible manner. To this end, robots are required to interpret natural language and to convert it into the executable programs. In this paper a framework for the natural language-based task programming is examined. The complex tasks are carried out by combining a set of primitive actions, which are programmed in advance. Basic idea behind the proposed framework is that the connections between primitive actions and verbs make it possible to convert a sentence into an executable program for the robot. The proposed framework is implemented on a mobile robot platform. Some preliminary results imply validity of the proposed framework.

1. はじめに

一般に、ロボットの行動はあらかじめプログラムされている必要があり、プログラムされていない行動を行うことはできない。全ての行動を事前にプログラムしておくことは困難であるため、実際の動作環境に応じてプログラミングを行う必要がある。しかし、ロボットのユーザーが必ずしもプログラミングの知識を持っているとは限らず、特に家庭用ロボットの普及を想定した場合、ユーザーが一般のコンピュータ言語を用いてプログラミングすることは困難である。従って、ロボットが人間からの自然言語による命令や説明を柔軟に理解し、自らが実行できるプログラムへと変換する能力が必要である。

こうしたロボットへの要求は、様々な場面で見ることができる。例えば、ロボカップ@ホームリーグでは、General Purpose Service Robot(GPSR)と呼ばれるタスクが2010年の大会より追加された[Robocup 10]。このタスクでは、行うタスクの具体的な内容は事前には知らされず、ユーザーからの自然言語をロボットが解釈して動作する必要がある。ロボカップ@ホームリーグは、ロボカップの新しいリーグであり、家庭用ロボットのための技術を競技形式で競うものである。多くの競技が、予め決められたタスクをいかに正確に行うのかを競うのに対し、GPSRタスクでは、ロボットのフレキシブルな行動生成能力が問われることになる。このように、家庭内でのタスクをユーザーの命令に応じて適応的に行うことができる能力は非常に重要である。

こうした背景のもと、本稿では、ユーザーの言語による命令を解析し、それをロボットが実行できる形へ自動的に変換する枠組みを提案する。著者らは、[板谷 10]において中核となる構文解析に基づくプログラムの自動生成について、固定ロボットを用いた評価を行っている。文献[板谷 10]において提案したシステムの基本的な考え方は、ロボットの行動が、基本的な動作の組み合わせで表現することができるというものである。例えば“ 掴む ”のような動作の場合、“ 手を開く ”、“ 物体の位

置に腕を動かす ”、“ 手を閉じる ”という基本的な動作を組み合わせることで実現することができる。そこで、これらの基本的な動作プログラムをあらかじめ用意しておき、これらを組み合わせることで複雑なロボットの行動を実現する。また、基本的な動作プログラムを、特定の単語と結びつけることによって、自然言語によるプログラミングを実現するものである。本稿ではそれらの動作を自律移動ロボットのタスクプログラミングへと拡張し、またユーザーとロボットとの対話を用いて命令の曖昧性を一部解消する機能を実現する。

関連研究としては、文献[青山 09]の走行プログラムの自動生成が挙げられる。しかしこれは、GUIによる操作のみを対象としており、入力された走行経路図からの走行プログラム生成のみを対象としている。文献[Knoop 07, Pardowitz 05]では、ヒューマノイドロボットの自動プログラミングを扱っているが、基本的にはユーザーの実演をいかにロボットで実行するかが主眼である。また、ロボットによる言語理解の研究も多く存在するが、対話システムにおける応答生成など、コミュニケーションが主眼であり、複雑なタスクを行うようなものは少ない。一方、ロボットによる言語獲得[Roy 02, Iwahashi 07]や模倣学習の研究も進んでおり、本質的な問題はこうしたアプローチにより解決される可能性がある。しかし、現段階ではまだ実用的であるとはいえず、また実用上はある程度の作りこみと学習のハイブリッドな手法が現実的であると思われる。また、ロボットと人間とのコミュニケーションの特徴を分析することでロボットに対する人間の認知特性を探るなどのような、人間とロボット間のコミュニケーションに対して人間視点から焦点を当てている研究も挙げられる[松本 07]。さらに、CGを自然言語から自動生成する研究も行われている[Oshita 09]。アプローチとしては非常に近いものの、実世界における物理的な動作とCGとは本質的な違いがある。

2. ロボットシステム

2.1 モジュール構造

本稿で想定するロボットのソフトウェアは、モジュール群として実装される。これは、RTMなどのミドルウェアを用いた

連絡先: 板谷 純希, 電気通信大学大学院情報理工学研究所, 東京都調布市調布ヶ丘 1-5-1, j-itaya@apple.ee.uec.ac.jp

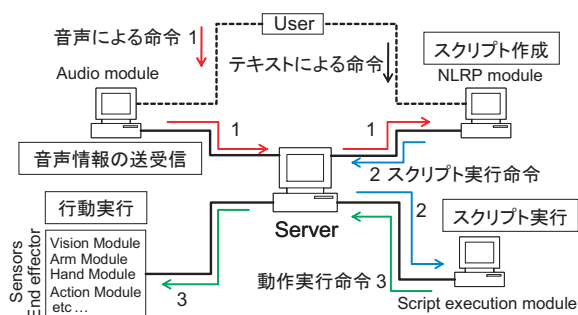


図 1: ロボットシステムの概要

```
[dll]
1 FileLoadObjectID.dll, //探す物体の情報を読み込む
2 VisionAddObject.dll, //物体の座標を記憶
3 握みます, //発話内容
4 HandLeftOpen.dll, //手を開く
5 ArmLeftGrabObjectID.dll, //物体を掴む
```

図 2: スクリプトの例

ネットワーク構造で実現可能である。本稿で用いるロボットでは、DiGOROnet と呼ぶ独自のミドルウェアを用いる。これは黒板モデルをベースとしており、モジュールの接続・切断が比較的柔軟に行えるのが特徴である。ロボットをプログラミングするためのソフトウェアも、一つのモジュールとして実現される。図 1 に、ロボット内のモジュール構造の概要を示す。Natural Language Robot Programming(NLRP) モジュールにおいてユーザの指示に基づいたスクリプト型のプログラムが自動生成され、そのスクリプトをスクリプト実行モジュールで実行する。

2.2 スクリプト実行モジュール

ロボットの行動は、基本的な動作をあらかじめ DLL として用意しておき、それを組み合わせて順に実行することで実現する。タスクプログラムは、図 2 のような DLL の名前が順に書いてあるスクリプトとなり、書かれている DLL を順番に呼び出すことで実行していく。基本的な動作の DLL は、設計者が事前に設計することになるが、動作学習によって既に用意してある動作を組み合わせることで新たに学習させることも可能である。本稿で実験に用いたロボットは基本動作として、「ロボットの移動」や「手の開閉」などが用意されている。また、基本動作を複数組み合わせさせた「物体を掴む」などの上位行動も複数用意されている。

3. 言語からのスクリプト自動生成

3.1 提案システムの概要

前節のシステムを用いることにより、自然言語によるロボットプログラミングは、いかに自然言語からスクリプトを自動的に作成するかという問題になる。図 1 における NLRP モジュールがこの仕事を行うことになるが、基本的な考え方は、用意した基本動作や上位行動に単語（動詞）を割り当てておき、ユーザーからの命令文を構文解析することで、動詞に対応した DLL を並べていくというものである。この際、主語や述語となる名詞は動作 DLL の引数となるため、センサ情報処理結果を参照することで、その存在の有無や位置などの情報に変換する。形容詞は、係り受け解析の結果をもとに、名詞とセンサ情報のマッチングなどに利用される。自然言語を用いて命

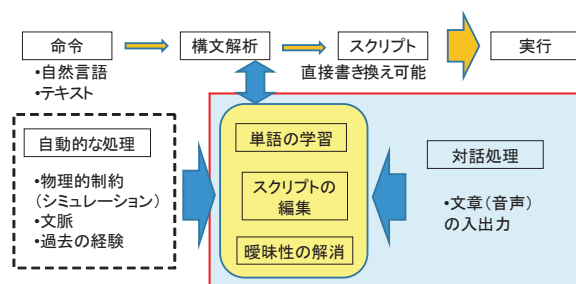


図 3: 自然言語によるロボットプログラミングの全体像

表 1: ロボットからの質問（” ”は命令に含まれる動詞の原型）

必要な情報	用いる助詞	質問文
場所	を	どこを のですか
場所	に（へ）	どこに（へ） のですか
場所	から	どこから のですか
物体	を	何を のですか
物体	に（へ）	何に（へ） のですか
人物名	を	誰を のですか
人物名	に（へ）	誰に（へ） のですか
形容詞・修飾語	の	どの ですか

令する場合の大きな問題は、命令文の曖昧性である。特に、移動ロボットにおいてはロボットが行動するために必要となる情報の種類が多く、より命令に曖昧性が含まれる可能性が高くなる。図 3 に構想の全体像を示す。最終的には、背景知識の獲得や経験からの学習、といったロボットが自動的に判断する機能を実現する必要があるが、本稿では対話によって曖昧性を解消する機能を実装し、その評価を行う。

3.2 基本処理

3.2.1 名詞の処理

名詞処理の基本は、その対象が空間的にどの位置に存在するのかを特定することである。しかし、名詞には物体位置を表わすものやロボットの移動座標を表わすものなどの種類があるため、それらの名詞にあらかじめ対象の位置特定に必要な情報を結びつけて学習しておくことで実現する。例えばロボットの移動先を表わす名詞の場合では結びついている座標情報が出力されるが、物体を表わす名詞では結びついている情報と、物体認識などセンサ情報処理結果を比較することによって得られた結果を最終的に出力する。

3.2.2 動詞の処理

動詞は、対応する基本動作や上位行動を選択し、実行時の引数として名詞の情報を受け取る。ここで問題となるのは、例えば「置く」のように手に持っていなければ実行できない暗黙の前提行動が存在する場合である。この前提行動は、直接的に指示される場合と省略される場合があるため、動作 DLL 自体がこれを判断し、必要に応じて前提行動に対応する基本動作を追加する。また、動作 DLL では必要となる情報（名詞）があらかじめ定義されているため、命令が曖昧な時に不足している情報の判断も動作 DLL 自体で行うことができる。

3.2.3 対話処理

命令文の曖昧性については前述したように動詞について処理した後に、その動詞に必要な名詞が不足していないかどうかで判断を行っており、不足している情報の種類に基づいて、それを補うようにロボットが人間に質問を行い、情報の補間を行

う。ロボットから行われる質問の一覧を表1に示す。表のように不足している情報(名詞)の種類と動詞へ係る助詞の種類によってロボットからの質問を決定しており、既にそれらは動詞に結びついて定義されているので命令文に含まれているかを解析することで質問が生成される。例えばユーザーからの命令が「コップを持ってきて」というものであった場合、コップの場所とコップの種類の情報不足しており、それらと組になる助詞は動詞に係る助詞である「～から」と「コップ」に係る助詞の「～の」であることが定義されている。また、この動詞の原型は「持ってくる」であるので表の 部分にこれをあてはめると、ロボットからの質問は「どこから持ってくるのですか?」と「どのコップですか?」となる。ロボットの質問は1つ行うごとにユーザーからの返答を待ち、質問を行った後の返答に含まれる名詞とその名詞に係る修飾語(形容詞など)に注目することで情報の取得を行う。

3.2.4 制御構造

条件分岐: ロボットに複雑な行動をさせる場合、行動の結果によって次の動作を決定する条件分岐が必要となる。ユーザの明示的な条件分岐命令は「～ならば」や「～たら」といった仮定形の助動詞、もしくは「～時は」や「～場合は」のような副詞可能名詞と係助詞に注目することで判断できる。

ループ: ロボットに同じ動作を繰り返し実行させたい場合、回数を指定する場合が多い。ここでは、動詞に数を表す名詞と「個」などの助動詞が係っていた場合に、その数だけ動作を繰り返すことでループを実現する。

並列処理: 並列処理は、順次命令されたものを自動的に並列化することで実行時間の短縮を図るものであり、ここでは確認発話など事前に設定した特定のものを並列化する。また、ユーザーの指示自体が並列性を持っている場合は、それらの物理的な行動可能性をチェックした上で並列化する。ただし、こうした指示の並列性は、接続助詞の「ながら」が文章内の二つの動詞をつなげているといった構文解析によって判断される。

3.3 未知語の学習

動詞と基本動作の組を全て事前に用意することは難しいため、基本動作を組み合わせた新たな行動に動詞を結びつけることで未知語を学習させる。例えば、「持ち上げる」を学習する場合、「掴む」動作を行った後に「腕を上げる」動作を実行させ、「持ち上げる」という言葉と「学習」という言葉を入力することで、直前に実行された動作のつながりと「持ち上げる」という言葉が結び付けられる。ただし、結びつける単語自体は音声認識の辞書に登録されている必要がある。従って、これは厳密な意味での未知語の学習ではない。一方、著者らは、[Attamimi 10]において厳密な意味での未知語の学習について議論している。こうした手法を、本稿で提案する枠組みに組み込むことも可能である。

3.4 スクリプト編集

生成したスクリプトは、エディタを用いて直接書き換えることが可能であるが、文章の命令による編集も可能である。現段階では、ある命令の前に何か他の命令を挿入するといった簡単な編集のみを実装している。

4. 実験

実験に用いたロボットと動作環境を、図4に示す。ロボットは、Segwayをベースとして、4台のオンボードPC、LRF、6自由度のアーム、赤外線カメラ、CCDカメラ、マイクロフォンから構成されている。カメラはパンチルト台に載っており、

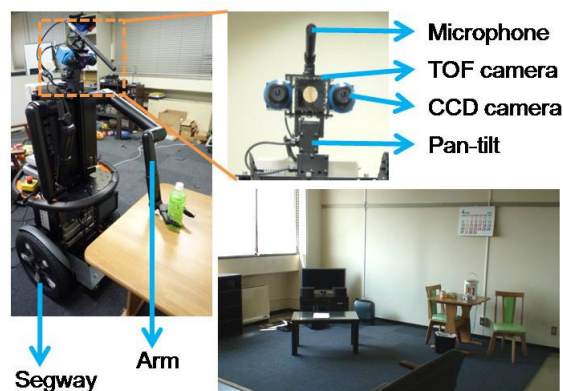


図4: 実験で用いたロボット DiGORO と動作環境

```
[dll]
1 FileLoadPosition.dll, //移動する場所の座標を読み込む
2 RobotMovePositon.dll, //指定された座標へ移動
3 HeadLookDown.dll, //机の上を見る
3 FileLoadObjectID.dll, //探す物体の情報を読み込む
4 VisionAddObject.dll, //物体の座標を記憶
5 掴みます。 //発話内容
6 HandLeftOpen.dll, //手を開く
7 ArmLeftGrabObjectID.dll, //物体を掴む
```

図5: 生成されたスクリプト例(タスク1)

周囲を見まわし物体の位置情報の取得や人の顔検出などを行うことができる。また、Segwayで移動することができ、LRFを用いた自己位置の推定や、カメラで検出した物体をアームにより把持することが可能である。

実際に文章から生成したタスクプログラムがどの程度複雑なものまで実行可能か評価するために、実行可能な行動を複数組み合わせる3つのタスクを人間が命令し、ロボットに実行させる。命令で使用する名詞はあらかじめ定義しておき、動詞や言い回しなどのパターンを10種類変化させ、タスクの成功率を評価する。また、一度の命令でスクリプトの生成ができなかった場合、ロボットから人間へ質問することでスクリプトの生成率が変化するかについても評価する。

1つ目のタスクは2種類の行動を組み合わせた場合の例として「移動」後に物体の「把持」を行うものである。命令の具体的な例としては「テーブルに行ってペットボトルを掴んで」というものがあり、この「掴んで」という動詞を「持って」や「取って」に、「行って」を「移動して」等同じ意味を持つ異なる動詞に変化させたり、「ペットボトルがテーブルにあるから掴んで」のような言い回しを変化させた命令を行う。2つ目のタスクは1つ目のタスクにさらに行動をプラスして3種類の行動を組み合わせた場合のタスクを行わせるもので、「移動」、「把持」をさせた後に「移動」と「置く」動作を行わせる。命令の例としては「テーブルに行ってお菓子を掴んで、冷蔵庫に置いて」のようなものである。こちらもタスク1同様に動詞や言い回しを変化させて行わせており、「テーブルの上にあるお菓子を冷蔵庫に置いて」のような自然で簡潔な命令でも実行可能かどうかを確かめる。3つ目は条件分岐動作を行わせるもので、「移動」後にそこにある物体によってその動作を変更するタスクである。命令には「テーブルに行って、ペットボトルが置いてあれば冷蔵庫に置いて、コップが置いてあれば棚に置いて」のようなものがあり、このタスクでは条件分岐部分の命令部分を「ペットボトルは冷蔵庫に、コップは棚に置いて」の

表 2: 命令文 (タスク 3) 対話なしで成功 対話ありで成功 ×失敗

番号	命令	結果
1	テーブルに行って, ペットボトルがあれば ペットボトルを掴んで冷蔵庫に置いて, コップがあればコップを掴んで棚に置いて	
2	テーブルに移動して, ペットボトルが あればペットボトルを冷蔵庫に置いて, コップがあればコップを棚に置いて	
3	テーブルにペットボトルがあれば掴んで, 冷蔵庫に置き, コップがあれば, 棚に置いて	
4	もしペットボトルがテーブルに置いて あれば冷蔵庫に置いて, コップが置いて あれば棚に置いて	
5	もしテーブルの上にペットボトルが あれば冷蔵庫に置いて, コップがあれば 棚に置いて	
6	テーブルに何か物が置かれていた時, それがペットボトルだったら冷蔵庫へ, コップだったら棚に置いて	×
7	テーブルのペットボトルを冷蔵庫に, コップを棚に置いて	×
8	テーブルに置いてあるのがペットボトル なら冷蔵庫に, コップなら棚に置け	×
9	テーブルのペットボトルを冷蔵庫に, 或いは, テーブルのコップを棚に置いて	×
10	テーブルに行って, ペットボトルは 冷蔵庫に置いて, コップは棚に置いて	×

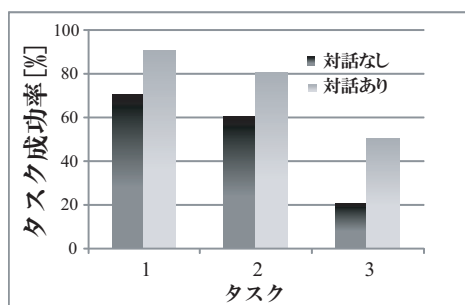


図 6: タスク成功率

ように動詞を省略する言い回しなどを含めて行う。

実験で生成されたスクリプトの例として, タスク 1 で作成されたスクリプトを図 5 に示す. このスクリプトには場所を表わす名詞である「テーブル」と, 物体を表わす名詞である「ペットボトル」がそれぞれ引数として渡され, それぞれと結び付いている情報を DLL によって読み込んで動作している. 生成したスクリプトを実行し, ロボットが実際にタスクを行えるか表わしたタスクの成功率を図 6 に示した. この時, 物体の検出失敗や把持失敗のようにロボットの動作中に起こるセンシングのミスなどによるタスクの失敗も有り得るが, 今回のタスクでは簡単な環境下で行ったためそれらは生じていない. また, 命令の具体的な例として, タスク 3 において実際に行った命令とその命令に対する実行結果を表 2 に示す. 結果のグラフより, 対話処理を行うことでタスクの成功率が上がる事が分かる. タスク 1・2 については言い回しの種類などに複雑なものが少ないため対話なしであっても比較的高い成功率である. しかしタスク 3 においては条件分岐処理のような複雑な動作を行わせているため, 命令に曖昧性が生じやすく, タスク 1・2 に比

べると成功率が低くなってしまふ. また, 対話処理を行っても成功率はあまり高くない. これは命令の曖昧性が「テーブルに行って, ペットボトルがあれば冷蔵庫に, コップがあれば棚に置いて」のように, 動詞 (置いて) を省略してしまう命令などが含まれているため動詞に必要な情報 (名詞) を補間するための対話では情報が補えなかったためであると考えられる.

5. まとめ

本稿では, 自然言語によるロボットタスクプログラミングの枠組みを移動ロボットへ適用した. 移動ロボットへの適用では, タスクがより複雑化することに伴い命令の曖昧性が増加する傾向にある. 本稿ではこれを解消するため, 動詞に必要な情報を補間する簡単な対話機能を実際に組み込み, 対話機能によるタスク成功率の向上に関する簡単な評価を行った. 今後はロボットが動作する環境が複雑化しても行動が可能かどうかの検証が必要である. また, さらなる対話機能の向上による曖昧性の解消, 経験や文脈などからロボットが自動的に曖昧性を解決するなどの機能を検討する予定である.

謝辞

本研究は, 科研費 (20500186, 20500179) 及び新学術領域研究「伝達創成機構」の助成を受け実施したものである.

参考文献

- [青山 09] 青山ほか: “次世代ロボット知能化技術開発プロジェクト - RTM 化におけるプログラム開発効率と品質の向上”, 第 27 回日本ロボット学会学術講演会, 2D2-06, 2009
- [Knoop 07] Knoop, S. *et al.*: “Automatic robot programming from learned abstract task knowledge”, in Proc. of IROS07, 2007
- [Pardowitz 05] Pardowitz, M. *et al.*: “Learning Sequential Constraints of Tasks from User Demonstrations”, in Proc. of HUMANOIDS2005, 2005
- [松本 07] 松本ほか: “人間とロボットの対話の認知特性 ホームロボットを用いた生活実験の対話ログの解析”, IEICE, MVE2007-16, 2007.6
- [Oshita 09] Oshita, M.: “Generating Animation from Natural Language Texts and Framework of Motion Database”, in Proc. of CW2009, 2009
- [板谷 10] 板谷, 中村, 長井: “自然言語によるロボットプログラミング”, IPSJ 全国大会, 2010.03
- [Robocup 10] RoboCup@Home Rules & Regulations, Version: 2010, 2010.04
- [Roy 02] Roy, D. and Pentland, A.: “Learning Words from Sights and Sounds: A Computational Model”, Cognitive Science, Vol.26, No.1, pp.113-146, 2002
- [Iwahashi 07] Iwahashi, N.: “Robots That Learn Language: A Developmental Approach to Situated Human-Robot Conversations”, In N.Sankar ed. Human-Robot Interaction, pp.95-118, I-Tech Education and Publishing, 2007
- [Attamimi 10] Attamimi, M. *et al.*: “Learning Novel Objects Using Out-of-Vocabulary Word Segmentation and Object Extraction for Home Assistant Robots”, in Proc. of ICRA 2010, 2010