

論理型プログラミングの学習を目指したタンジブルなツールの試作と評価

Prototype and Evaluation of Tangible Tool for Learning Logic Programming

宮崎 勇希^{*1}, 本吉 達郎^{*1}, 布施 陽太郎^{*1}, ミヤグマルドラム ビルグウンマ^{*1}
澤井 圭^{*1}, 増田 寛之^{*1}, 高木 昇^{*1}

Yuki Miyazaki^{*1}, Tatsuo Motoyoshi^{*1}, Yotaro Fuse^{*1}, Bilguunmaa Myagmardulam^{*1}
Kei Sawai^{*1}, Hiroyuki Masuta^{*1}, Noboru Takagi^{*1}

^{*1}富山県立大学工学部知能ロボット工学科

^{*1}Faculty of Engineering, Department of Intelligent Robotics Engineering, Toyama Prefectural University
Email: u014048@st.pu-toyama.ac.jp

あらまし: 論理型プログラミング言語は全体構造の把握が難しく扱いにくいいため、ユーザが学習しやすいアプリケーションがいくつか開発されている。しかし、情報機器の操作リテラシーのない初学者や視覚障がい者の使用を想定したタンジブルなツールは見受けられない。そこで、論理型プログラミングの学習が可能なタンジブルなツールの開発と評価に取り組むこととした。本稿では試作システムの概要、およびその評価結果について報告する。

キーワード: 論理型プログラム, Tangible User Interface, 視覚障がい者支援

1. はじめに

近年、人工知能は、多岐にわたり社会全体を支えている。このような社会の変化から、IT人材を確保することが課題となっており⁽¹⁾、初等教育での人工知能分野に関する学習機会が増加している。

一般的に、ユーザにとってプログラムを識別しやすい教材には、Graphical User Interface (以下 GUI) や Tangible User Interface (以下 TUI) が用いられる。しかし、これらの教材の多くは手続き型言語の学習を目指したものであり、非手続き型言語である論理型プログラミング言語を用いた学習ツールは少ない。

本研究の目的を P-CUBE シリーズ⁽²⁾を基盤とし、論理型プログラミングの学習を目指したタンジブルなツール (Logi-CUBE) の試作と評価とする。

2. Logi-CUBE の概要

Logi-CUBE は、プログラミングブロックをプログラムマットに並べることによって論理型プログラムを構築し、質問に対する解答や知識を音声出力するプログラムを作成できる。TUI を採用し、視覚障がい者も使用できるよう配慮する。

2.1 システムの概要

Logi-CUBE は、(1)知識と質問の入力を行う 2 種類のプログラムマット、(2)「～を好きである」のような述語と「りんご」のような単語の情報が登録されている 2 種類のプログラミングブロック、(3)知識と質問を PC に転送するための転送用スイッチ、(4)スピーカー、(5)制御用 PC で構成されている。

システムには RFID 技術が用いられており、プログラミングブロックには RFID タグ、プログラムマットに RFID リーダを取り付けることで、ユーザが

作成したプログラム構造を読み取る。Fig.1 に Logi-CUBE のシステムの概要を示す。

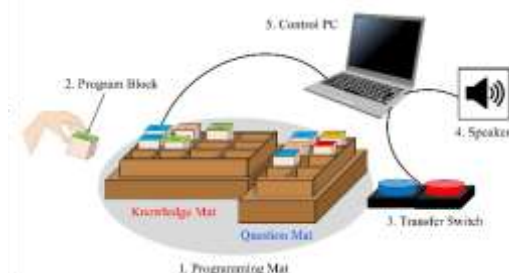


Fig.1 System Configuration of Logi-CUBE

2.2 制御システム

制御システムは「RFID 制御による知識・質問識別システム」(以下 RCS), 「データベース制御システム」(以下 DCS) によって構成されている。システム構成を Fig.2 に示す。RCS は RFID タグの識別番号からブロックの配置状況を取得し、DB で使用できる文字列に変換し、テキスト形式で保存する。DCS は、テキスト形式で保存された事実に対して規則で定義した推論規則を適用し、新たな事実を作成したうえで、解答生成プログラムに送信する。質問に対しては知識と比較し、解答を音声として出力する。

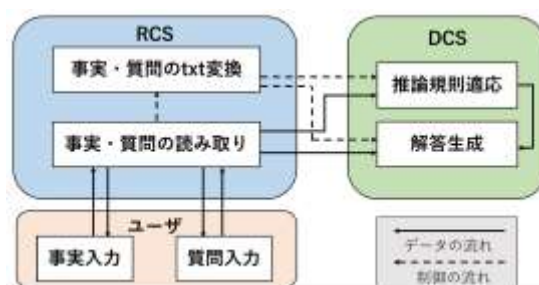


Fig.2 System Configuration of Software

3. Logi-CUBE で作成するプログラムと検証

Logi-CUBE を用いて作成できるプログラムは検索系と推論系の2種類である。

3.1 検索プログラム

知識マットに単語ブロック「りんご」、質問マットに述語ブロック「いろ」、および単語ブロック「りんご」を配置する検索プログラムについて説明する。質問マットの同じ列に置かれた述語ブロックと単語ブロックは、1つの質問と認識する。知識は「りんご」、質問は「りんごは何色ですか」である。DCSはDBに登録された基礎情報をりんごに関する知識として得る。質問に適合する知識がDBに存在するため、制御システムは「赤色」を解答として出力する。Fig.3 にブロック配置、プログラム処理順番、および単語ブロックの基礎情報を示す。

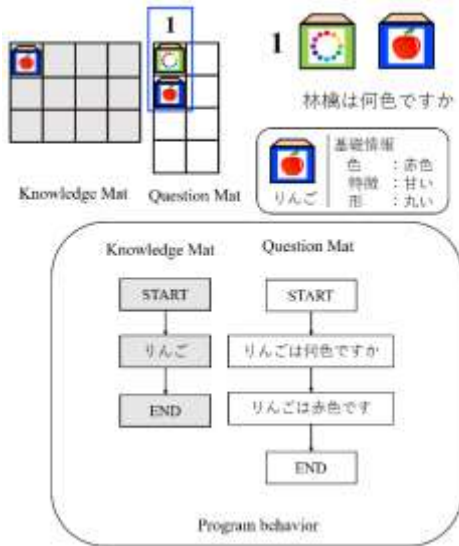


Fig.3 Example of Search Program

3.2 推論プログラム

知識マットの1行目に述語ブロック「すき」、単語ブロック「わたし」「どうぶつ」、2行目に述語ブロック「どうぶつ」、単語ブロック「いぬ」、質問マットに述語ブロック「すき」、単語ブロック「わたし」「どうぶつ」を配置する推論プログラムについて説明する。知識マットの同じ行に置かれた述語ブロックと単語ブロック、および質問マットの同じ列に置かれた述語ブロックと単語ブロックは1つの知識、質問と認識する。知識は「私は動物が好きです」「犬は動物です」、質問は「私は犬が好きですか」である。DCSは、DBに蓄積された事実と事前登録されている推論規則「私は動物が好きです。かつ、犬は動物です。ならば、私は犬が好きです」を用いて新しい知識「私は犬が好きです」を作成する。質問「私は犬が好きですか」に適合する知識がDBに存在かつ、知識と質問の内容が一致するため、制御システムは「はい」を解答として出力する。Fig.4 にブロック配置、プログラム処理順番、および推論規則を示す。

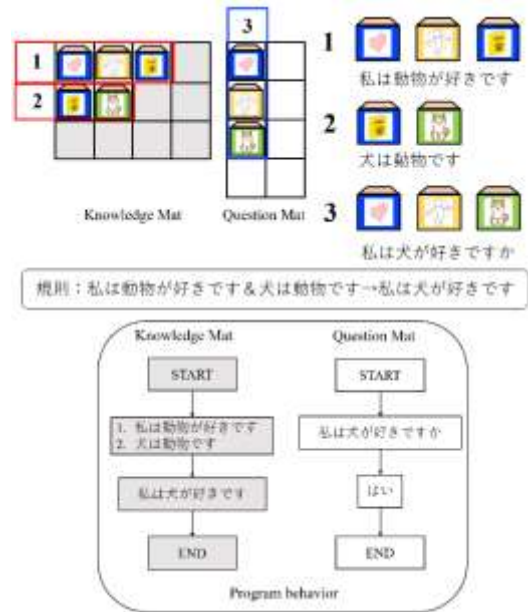


Fig.4 Example of Inference Program

3.3 検証内容と結果

RCSでは、知識、質問が入力し、転送キーを押すとそれぞれがProlog⁽⁴⁾で使用できる文字列に変換されテキスト形式で保存されているか確認する。DCSでは、テキスト形式の事実を読み取り、知識として出力した、事実を推論規則に適用し、新たな知識が出力されているかを確認する。また、解答生成が行われているかも合わせて確認する。

RCSでは、所望のキーが押された際、テキスト形式で事実を保存できることを確認できた。DCSでは、テキスト形式で保存された事実を知識として出力し、新たな知識を出力した。また、知識と質問を比較し質問に適した解答を出力することを確認した。

4. まとめ

本研究では、論理型プログラミングの学習を目指したタンジブルなツールの試作し、プログラムの作成方法の構築、および試作機の評価に取り組んだ。RCSとDCSが連携、および各処理が機能していることが確認できた。

今後は、試作機の作成、ブロックのデザインやマットの拡張可能性の検討を行う。

参考文献

- (1) 経済産業省：IT人材の最新動向と将来推計に関する調査結果,
https://www.meti.go.jp/shingikai/economy/daiyoji_sangyo_skill/pdf/001_s02_00.pdf, (閲覧日:2023/4/24).
- (2) 本吉達郎, 掛橋駿, 小柳健一, 大島徹, 増田寛之, 川上浩司, ブロック型プログラミングツール P-CUBE の学習初期段階における有用性の検証, 日本知能情報フアジィ学会誌, Vol. 27, No. 6, pp.909-920, (2015).
- (3) 小川東:「Prologによる論理プログラミング入門」, 啓学出版, pp.2-20, 1990.