

発表概要

解析表現文法によるプログラム中の欠損を許した構文解析とそのツールの提案

山能 佑介^{1,a)} 佐々木 晃²

2021年3月16日発表

プログラミング初学者に対して用いられる学習環境として、ビジュアルプログラミング、特にブロックを用いたブロック型のビジュアルプログラミングがある。そのなかでも学習のためブロック型言語と一般的なプログラミング言語によるテキスト型言語の間で変換が行えるシステムも存在する。このようなシステムでは、ブロック型言語とテキスト型言語の間で任意のタイミングで変換が行われる。しかし変換は構文に誤りがない場合のみ行えるという問題がある。これは構文誤りのあるプログラムでは構文解析ができないため抽象構文木が作れず、プログラムの変換ができないためである。これらを解決するため本研究では、欠損のあるプログラムを許容する構文を自動生成するツールを提案する。本ツールでは、プログラミング言語の構文を与えることで、欠損のあるプログラムでも構文解析可能な解析表現文法を生成する。また、同時に、生成した構文での構文解析の過程で作られる具象構文木を操作するためのスタブを生成し、具象構文木からの変換を支援する。これは、具象構文木に対する構文主導解析プログラムの雛形であり、具象構文木から抽象構文木や他のプログラム表現（ブロック等）への変換を支援する。その結果、編集途中等で欠損のあるプログラムでも、テキスト言語やブロック言語の間で変換できるシステムを容易に導入可能となる。

Presentation Abstract

A Proposal of a Parsing Method and its Tool that Treats Syntactically Incomplete Programs Using Parsing Expression Grammars

YUSUKE YAMANO^{1,a)} AKIRA SASAKI²

Presented: March 16, 2021

Visual programming, especially block-based programming, is used for beginners in programming. Particularly, there are systems that programmers freely convert between a block-based language and a general text-based language. However, there is a problem that the conversion can be done only when there is no syntax error in the text program. This is because programs with syntax error cannot be parsed and therefore cannot be converted into abstract syntax trees or other forms of representation. In order to solve this problem, we propose a tool that can automatically generate a grammar that allows a program with missing parts. The tool generates a parsing expression grammar that can parse such incomplete program by providing the syntax of the programming language. At the same time, the tool generates a template for manipulating concrete syntax trees that are created in the process of parsing with the generated parser, and supports conversion from concrete syntax trees. The template is that for a syntax-directed parsing program for the concrete syntax tree, which supports the conversion from the concrete syntax tree to the abstract syntax tree or other program representations such as blocks. As a result, it is easy to introduce a system that can convert between text-based languages and block-based languages, even if the program has some missing parts in the process of editing.

This is the abstract of an unrefereed presentation, and it should not preclude subsequent publication.

¹ 法政大学大学院情報科学研究科
Graduate School of Computer and Information Sciences,
Hosei University, Koganei, Tokyo 184-8584, Japan

² 法政大学情報科学部
Faculty of Computer and Information Sciences, Hosei University, Koganei, Tokyo 184-8584, Japan

^{a)} yusuke.yamano.3n@stu.hosei.ac.jp