1 P − 2

知的プログラミング教育環境 INTELLITUTOR における プログラム理解システムＡＬＰＵＳの拡張
— 知識の改訂･追加機能 —

直井将行*
東京電機大学大学院理工学研究科

福田妙明**　上野晴樹***
東京電機大学理工学

1 はじめに

初心者プログラマがテキストエディタやコンパイラを使ってプログラミングを行った場合を考える．初心者プログラマは，目的を実現するためのアルゴリズムの知識，プログラミング言語に関する知識が不足しているため，作成したプログラムには，文法エラーや論理エラーが含まれている可能性が非常に高く，それらを訂正するために多くの時間を割かなければならない．

当研究室では，初心者のプログラミング教育を支援する環境型知的 CAI システム INTELLITUTOR を開発している．

INTELLITUTOR は２つのサブシステムで構成されておりプログラム入力部とプログラム理解部の ALPUS からなる．ALPUS では，学習者の入力したプログラムから,システムがあらかじめ持っている知識を利用し，そこではどんな処理を行っているかを理解し，論理的な誤りの検出を行う．その結果誤りが発見された場合は，誤りに対応した助言を行うことができる．このようにしてプログラムの意味を推論することによりプログラム理解が行われている．現在システムには C 言語と Pascal の知識が用意されている．

しかしながら今までのシステムは知識の不足による理解の失敗が多かった．本報告では新たな知識を獲得するための知識改訂機能について述べる．

2 ALPUS システムの概要

2．1 ALPUS が利用する知識

ALPUS には，次のような知識が用意されている．

・アルゴリズムの知識

アルゴリズムとは，複数のデータ処理のチャンク（プログラムコードのまとまり）の組み合わせで表現できる．以後，このチャンクの事をプロセスと呼ぶ事にする[1]．つまり，アルゴリズムは，プロセスのシーケンス（組み合わせ）として表現できる．プロセスには階層があり，シーケンスとしてこの階層を表現したのが階層的手続きグラフ（HPG）である[2]（図１）．

・プログラミング技法の知識

プログラミングに関する具体的なコーディングの方法を管理しており，HPG の各プロセスを表すノードに意味ネットとしてつながっている．

・変数の知識

アルゴリズムで使用される変数の役割を決定する情報を管理しており，アルゴリズム知識にリンクされている．

・バグの知識

バグの原因と解決法を示す機能が，関連するプロセスまたは技法知識にリンクされている．

・言語の知識

特定のプログラム言語に共通な概念に関する情報を管理している[3].

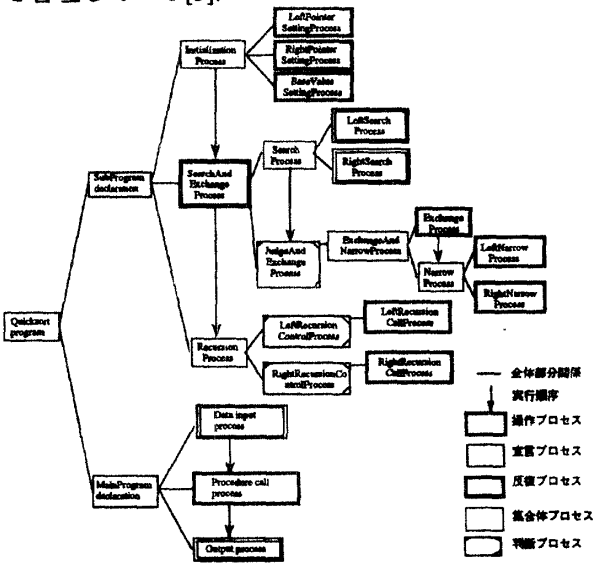

図１　QUICKSORT の HPG

2．2 ALPUS の処理

ALPUS は，次のような手順で処理を行い，プログラム理解を達成する．

まず，学習者の作成したプログラムを抽象言語表現に変換する抽象化を行う．これは，システムに複数のプログラム言語を処理させるためである．

次に学習者の作成した意味を壊す事なく，記述方法の多様性を減らし，プログラム理解の柔軟性を高める，正規化を行う．

さらに，学習者の定義した変数の果たす役割を理解するために，変数同定を行う．変数の役割を推論する事で,学習者の意図を推論する事が容易になる．

最後に論理エラーを見つけるため，プロセス同定を行う．ここでは，プログラムの各セグメントと HPG の各プロセスとの対応付けを行う事により，

Extension of Program Understanding System for Programming Environment INTELLITUTOR
*Masayuki Naoi
Tokyo Denki Univ. Graduate School of Science and Engineering
**Tadaaki Fukuda , ***Haruki Ueno
Tokyo Denki Univ. The Faculty of Science and Engineering

論理エラーの検出を行う．

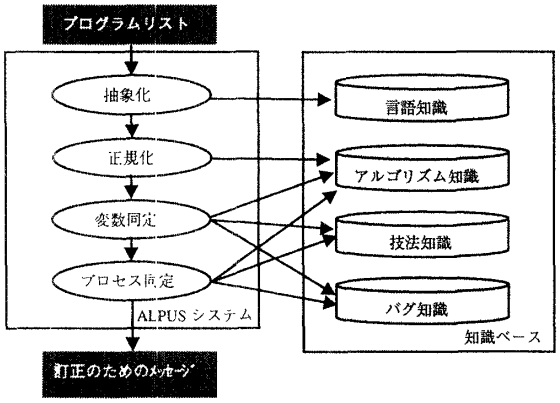

図２　ALPUS の処理手順

３テンプレート生成とパターンマッチング

３．１テンプレート生成

変数同定が持つテンプレート生成機能は，抽象化，正規化処理を行ったプログラムとのマッチングを行うためアルゴリズム知識と技法知識を使いプログラムのテンプレートを生成ものである．

テンプレートを生成する手順は HPG の最上位から順にプロセスの一つを指定する．そのプロセスには技法知識へのリンクがはってある．技法知識を参照すると Template スロットにはテンプレートに必要な情報が記述されている．またテンプレートに必要な情報はそのプロセスに記述されており，先ほどのテンプレートに必要な情報に代入してテンプレートを作成する．

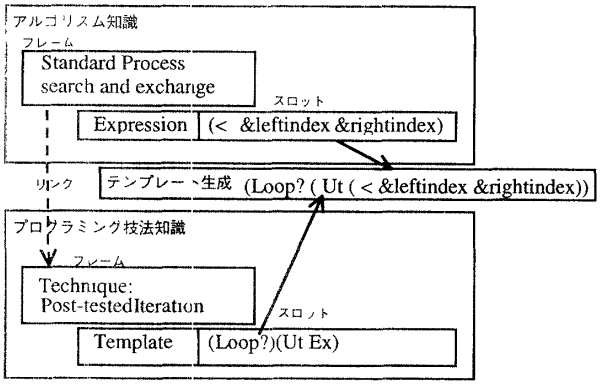

図３テンプレート生成

３．２パターンマッチング

パターンマッチングは，HPG で指定されたプロセスから生成されたテンプレートと，抽象化，正規化を行ったプログラムコードとのマッチングを行い，指定されたプロセスと同じ振る舞いをするコードを見つけ出す機能である．

マッチングに失敗したプログラムリストは知識の改訂のために新たなフレームに保存される．

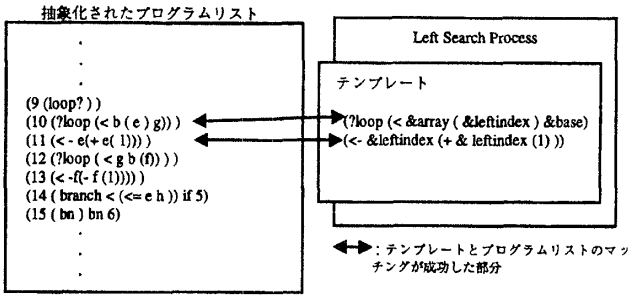

図３　パターンマッチング

４知識改訂機能

これまでのシステムは，パターンマッチングに失敗すると，理解不能というメッセージが帰ってくるだけであった．また知識に無いパターンが原因での理解不能も多くあった．そこでマッチングに失敗した場合，失敗したプログラムコードは知識に無い記述パターンであると判断し，知識ベースに格納する機能が必要となった．知識ベース管理者が容易にマッチングに失敗したプログラムコードを知識として取り入れるための知識改訂機能の流れを記す．

知識を獲得するためにマッチング失敗したプログラムコードを知識ベースに保存しておく．知識獲得のために管理者はそのプログラムコードがどのプロセスに当たるかを指定すれば，あとはシステムがアルゴリズム知識とそのリンク先の技法知識に新たなフレームを作り，その中にプログラムコードを変数の情報とテンプレートの情報とに分解して格納する事を自動的に行う．これにより知識の改訂がより簡単になる．

５まとめと今後の課題

この知識改訂機能を追加する事により，一度処理をしたプログラムは理解可能なので多様なプログラムのパターンに対応する事ができる．

しかしこの機能は知識ベース管理者がマッチングに失敗したプログラムコードを有用なものであるか正しく見極め，正しいプロセスを指定してやることが重要である．

またバグの知識を改訂する場合，それに伴い訂正，助言のメッセージを入力する事も必要になる．

参考文献

[1]上野晴樹：知的プログラミング環境-プログラム理解を中心に，情報処理 vol.28,no.10,pp.1280-1295,1987

[2]Ueno,h：Knowledge Based Program Understanding for Intelligent Programming Environment for Novice programmers, Tokyo Denki University Research Institute For Technology,vol7,no2,pp12-15,1994

[3]野中美和子 山本康弘：知的プログラミング教育環境におけるプログラム理解システム ALPUS－変数同定･プロセス同定　東京電機大学卒業論文, 1996