

学習者に適応的なプログラミング学習環境構築のための リアルタイムな理解状況把握手法

山本耕大[†] 中野聡美[†] 柳生亜也子[†] 中村勝一[‡] 横山節雄[†] 宮寺庸造[†]
東京学芸大学 情報教育学科[†] 福島大学 理工学群[‡]

1. はじめに

プログラミング教育は継続的に学習することで学習効果が得られる。そのためプログラムのコーディング等、同じ学習を行っていても、理解が進むに連れて習得するスキルは変化するのが一般的である。プログラミング学習支援に関する研究は過去に多数行われている[1][2]。しかし、既存研究で開発されているプログラミング学習支援システムは、個々に独立して存在しており、また対象とする学習場面や理解状況が限定されていることから、支援システムの有効利用は難しい。そのため、個々に存在する支援システムを統合的に、かつ学習者にとって最適な状態で提供する必要があると考えられる。

そこで本研究では、プログラミング教育において個々の学習者の状態を把握し、それに合わせた支援システムの提示を行う学習環境の構築手法を提案する。特に本稿ではこの環境を実現するための最初の段階として、学習場面と理解状況のリアルタイムな把握手法を開発する。

2. 学習者に適応的なプログラミング学習環境

プログラミングの学習の中で、初学者は課題を達成するためのアルゴリズムの考え方や発生したコンパイルエラーの解決など、複数の学習場面で教授者の支援を必要としている。そのため過去の研究では、学習者を支援するための様々なシステムが開発されており、それらのシステムは高い成果を上げている。しかし、プログラミング教育において、学習状況はリアルタイムに変化するため、学習者に常に最適な支援システムを提供することができないという問題が存在する。

上記の問題を解決するために、図 1 のように個々の状態に合わせてリアルタイムに学習環境を構築し提供する手法を提案する。環境構築の

手順として、まず、学習場面や理解状況のような学習者の状態を表すための学習者モデルの設計、定義を行い、状態の自動把握を行う。次に、把握した状態に応じて最適な支援システムを抽出することで、学習者に適応的な学習環境を自動的に構築し、提供する。提供された支援システムは学習者が使用することで、学習者の状態を再定義し、それに合わせて学習環境の再構築も行う。これにより、常に学習者に適応的な学習環境を提供できる。

そこで本稿では図 1 の実現のために、学習場面と理解状況のリアルタイムな把握手法について述べる。

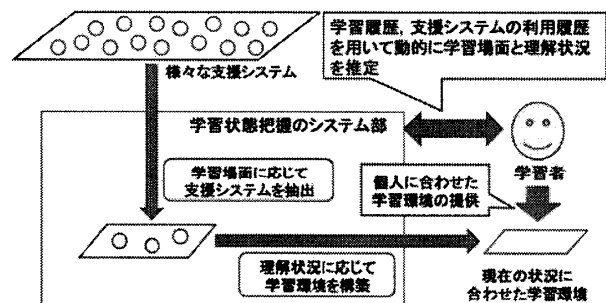


図 1. 学習者に適応的なプログラミング学習環境

3. 学習場面把握

上記学習環境の実現にあたり、学習状態とその時の支援法との関係は重要である。その関係の記述には文献[3]を参考に行う。[3]では「作業」という学習状態が設けられているが、プログラミング教育における学習場面を状態として設定することで、[3]の手法が適用可能になる。

そこでプログラミング教育における学習場面の抽出を行う。実際に本学の「プログラム言語 I」の授業から、筆者らが検討した結果、以下のように 6 つの学習場面を定めた。

講義・復習、設計・考え方、コーディング、デバッグ作業、実行可能、課題完成

さらに、抽出した学習場面の遷移について考えた結果を、図 2 の通りに定める。なお図中での記号 ○ は学習者の状態を、そして→ は学習者のアクションを、--> は時間により自動的に変化することを表している。

A Real-Time Understanding Situation Grasp Method for Adapting Environment Construction in Programming Language

[†]Kodai Yamamoto, Satomi Nakano, Ayako Yagyū,
Setsuo Yokoyama, Yozou Miyadera - Tokyo Gakugei
University

[‡]Shoichi Nakamura - Fukushima University

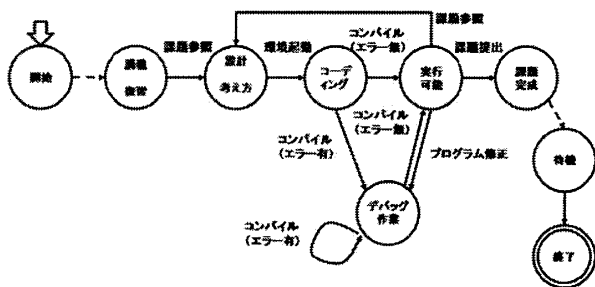


図2. プログラミング学習における場面遷移

4. 理解状況把握

まず学習者の理解状況把握のための指標となる「理解レベル」の定義を行う。従来の状況把握に用いられている理解度は、ある学習場面に応じて数値を変化させるものである。しかし、初学者はエラー解決時などの経験の蓄積により、ある事象を除々に理解していくものと考えられる。そこで本研究では「ある事象について知識をどれくらい有しているか」を理解レベルとして定める。この指標の導入により長期的でリアルタイムな理解状況把握が可能となる。

4.1. 理解レベル把握

学習者の理解状況推定は理解レベルを用いることにより実現する。リアルタイムな理解状況を推定するためには、図2の学習場面遷移に則した推定を行う必要がある。これは場面遷移において、学習者はある特定の場面のみで学習項目について理解しているとは考えられないため、様々な学習場面から理解状況を考慮することが必要であると思われる。そこで本研究では、それらの学習場面から得られる学習履歴に着目した。学習履歴を定量的に表現し、それらを足し合わせていくことにより、単元別の理解レベル算出を行う。

本稿では特に学習場面に密接な関連のある、コンパイル履歴、課題終了後のアンケート結果、FAQシステムの利用履歴、の3つの学習履歴を用いて理解レベルの推定を行う。なお学習履歴を定量的に判断するために、各課題は課題番号、課題内容、課題の困難度、単元ID、の構成要素の属性として持つものとする。

4.2. コンパイル履歴による推定

コンパイルエラーの解決数を課題演習時の学習履歴として保持することにより、理解レベルの推定を行う。これは課題演習中に発生するエラーを解決することにより、プログラミングに関する理解が深まると考えられるためである。コンパイルエラーの単元の推定は、文献[4]を用いることにより実現し、コンパイルエラーの解決数を理解レベル推定に活用する。

4.3. アンケート結果による推定

本学では、課題終了後に5件法によるアンケートを実施している。このアンケートにおいて、取り組んだ課題についての理解度、ある単元における理解度の2項目についての質問をし、その値に単元の困難度で重みをつけることにより、アンケート結果による理解レベル推定を行う。

4.4. FAQシステムの利用履歴による推定

本研究では、数多く存在する支援システムの中の一つとしてFAQシステムを用いることにする。FAQシステムとは、学習者の質問とそれに対応する回答を学習者間で共有することを目的とした支援システムである。本手法ではこれに加え、それぞれの回答内容を学習者が評価する機能を追加した。回答の内容が理解できたか否かの判断を学習者自身がを行い、その回答数を保持することにより、理解レベル推定を行う。

4.5. 理解レベルの適用

理解レベルの算出は各学習履歴から求めた理解レベルに一定の重み付けを行い、それらを足し合わせることで行う。求めた理解レベルは、適用する支援システムに応じ3段階で扱い、適応的な学習環境構築に用いる。

本稿では3つの学習履歴を利用することで理解レベル推定を行ったが、FAQシステム以外の支援システムについても利用後に得られる履歴を分析することで、本手法に適用する予定である。

5. おわりに

本稿ではプログラミング学習において、学習者に適応的なシステムの提示を行う学習環境の概要と、そのために必要な学習状態の把握手法について述べた。今後は学習者に適応的な環境を構築するために、様々な支援システムの分析を行う予定である。

文献

- [1] 長慎也, 甲斐宗徳, 川合晶, 日野孝明, 前島真一, 寛捷彦, “プログラミング教育 Nigari-初学者が Java を習うまでの案内役,” 情報処理学会論文誌, vol.45, no9, pp.25-46, 2004.7
- [2] 渡辺博芳, 高井久美子, 荒井正之, 武井恵雄, “初等アセンブラプログラミングにおけるニアミスプログラムに対するアドバイス提示法,” 情報処理学会研究報告, vol.2003-CE-69, pp.31-38, 2003
- [3] 森本康彦, “e-Learning 環境の形式的記述手法の開発,” 長岡技術科学大学学位論文, 2007
- [4] 山本耕大, 春原将寿, 大金克紀, 中村勝一, 横山節雄, 宮寺庸造, “エラー要因事例ベースの動的学習手法を導入したC言語教育システムの開発と基礎的評価,” 電子情報通信学会信学技法, Vol.108, No.146, pp67-72, 2008.7