

オンラインプログラミング学習システムのための 適応型出題モデルの提案

齋藤 宏太郎[†] 豊田 哲也[†] 大原 剛三[†]

[†]青山学院大学 理工学部

1. はじめに

現在、プログラミングの授業が多くの教育機関で行われているが、学習者それぞれの理解度のばらつきが大きいため、個々人の理解度に応じた学習支援システムが求められている。その解決手段としてコンピュータ適応型テストがあるが、課題を生成する際に用いる出題モデルに関しては、システム構築者の経験のみに基づいた固定的なモデルを用いているのが現状である。そこで本研究では、プログラミングの穴埋め問題を対象に、学習者の解答状況に応じて学習すべきプログラミング要素間の依存関係自体を動的に変化させる適応型出題モデルを提案する。また、提案モデルを利用したオンライン学習システムを大学の実習に導入し、その有用性を検証する。

2. 関連研究

これまでに、プログラミング学習者のための学習支援システムがいくつか開発されている。掛下らはC言語プログラミングの教育支援システム

「pgtracer」を開発している[1]。これはオンラインでの自学自習用のシステムであり、プログラミングの穴埋め問題を学習者に自由記述で解答させ、正しくプログラムが動作するかを判定する。しかし、あくまで正誤判定のみを行うものであり、その後の問題推薦などは行っていない。また、小芦らが開発している「MILES」[2]は、事前に構築した学習者の理解度モデルを利用し、学習者の解答状況を元に学習者自身の理解度を更新し、そのモデルに合わせて次の課題が推薦されるというシステムになっている。学習者に合わせた課題推薦ができる一方、理解度モデルは固定されているため、モデルの構築者に依存してしまうという問題がある。

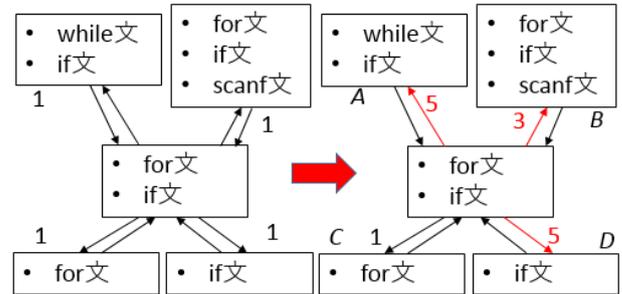


図1 提案モデルのイメージ

3. 提案手法

3.1 提案モデル

提案モデルを構築するために、講義で使用されている参考書から、if文やfor文などのC言語プログラミングの理解に必要な要素を抽出し、それらをタグとする。次に、C言語の穴埋め問題を作成し、穴埋め問題の解答箇所とタグを1対1対応させ、問題に付与されたタグの種類・個数ごとにノードを構築する。提案モデルのイメージを図1に示す。図1の中心ノードはfor文およびif文のタグが付与された問題が属するノードを示している。これらのノードに対してC言語の学習過程に応じた学習の到達度を数値化して付与する。学習が進むにつれてタグ数の増加や難易度が上昇するため、これを到達度として表現する。これらの到達度が付与された各ノードに対して、同一到達度のノード間に有効リンクをつなぎ、さらに、到達度±1となるノード間に有効リンクを設ける。最後に、各ノードに出題確率の初期値を付与する。

3.2 モデル更新

提案モデルは、各ノードに属する問題を解答することによって更新が行われる。各問題の解答欄はそれぞれのタグと対応しており、タグごとに正誤判定を行う。学習者がどのタグに対して正答/誤答したかを記録し、学習者の解答傾向に応じてモデルの出題確率が変化する。図1に示すように、if文とfor文の問題に正解した場合、どちらのタグに対しても理解していると判断し、2つのタグにscanf文が加わったノードの出題確率が上昇する。一方で、if文にのみ不正解だった場合、if文のみ

A Proposal of an Adaptive Questionnaire Model for Online Programming Learning Systems

Kotaro SAITO[†], Tetsuya TOYOTA[†], Kouzou OHARA[†], [†]College of Science and Engineering, Aoyama Gakuin University

のノードと, for 文を while 文に変更したノードの出現確率が上昇する. このように, 学習者の解答傾向に応じて各ノードに属する問題の出題確率が変化し, 学習すべきプログラミング要素間の依存関係を動的に変化させることが可能となる.

3.3 モデル更新のためのドリルシステム

提案モデルの更新を行うために, C 言語プログラミングの穴埋め問題を対象としたドリルシステムを構築する. ドリルシステムのインターフェースを図 2 に示す. 解答欄に解答を自由記述で入力して「解答する」ボタンを押すと, 正誤判定と解答例が出力され, 「次の問題へ」ボタンを押すと次の問題に進むことができる.

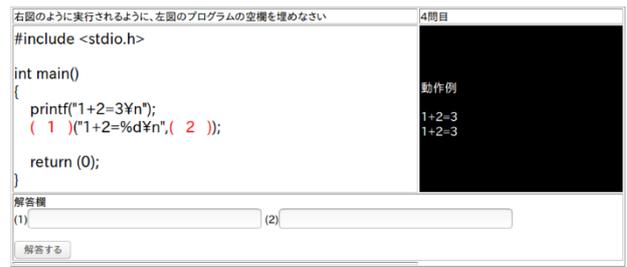


図 2 ドリルシステムのインターフェース

4. 評価実験

4.1 評価実験の概要

提案モデルの有用性を評価するために, 実際の講義にシステムを導入した. 今回は, C 言語について基礎から学ぶ「情報処理実習」を受講している青山学院大学の学生 162 名を対象とした. 「情報処理実習」は全 15 回の講義であり, 第 8 回目から試験的に導入を行い, 第 11 回から本格的に導入を行った, 5 分間ドリルシステムを利用してもらい, 実際の解答状況と, モデルの更新過程を記録した.

4.2 実験結果と考察

ノードの到達度とそのノードに属する問題の解答数及び正答率を図 3 に示す. 解答数は到達度 5 で最大となっており, さらに, 到達度 4 から 8 の問題に対する正答率が低いことから, これらの到達度に関する問題付近で, 正解と不正解が繰り返されたと考えられる.

さらに, 学習者の解答ログを分析したところ, 到達度 5 の付近でよく間違われていたタグは

「scanf 文」と「switch 文」であった. それらのタグを含む簡単な問題が出題されているのが到達度 5 付近であり, モデルの更新によってその問題が出題されやすくなっていたことが判明した. 解答数の到達度 5 付近がピークであったのも, これが原因であると考えられる.

モデルの更新について分析をしたところ, ノードの約 50%において, 同じノード間の出題確率の更新が繰り返されていることが分かった. このような状況は到達度 5 以下のノードで, 学生の解答傾向が一定であったためだと考えられる. 一方で, 到達度 6 から 9 までにおいて, 更新頻度が高いにも関わらず, 更新した出題確率の値が変わらないノードが存在した. この付近のノードでは, タグの種類が大幅に増えるため(繰り返し処理, 条件分

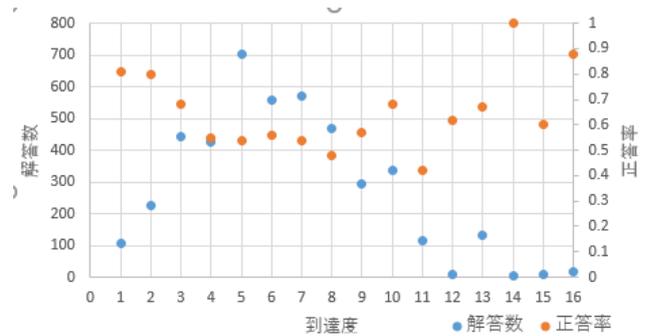


図 3 各到達度の解答数と正答率

岐等), 学生個々で理解できている場所が大きく異なり, 解答に決まった傾向が表れなかったことが原因であると考えられる.

5. おわりに

本研究では, 適応型出題モデルを提案し, その有用性の検証するために, 実際の講義に導入した. その結果, 一定の解答傾向がある場合, それに合わせたモデル更新が行えることが分かった.

今後は, 一定の解答傾向が無い場合にも, 問題選択のための更新が行えるような更新方法を考える必要がある.

参考文献

[1] 掛下哲郎ほか: 穴埋め問題を用いたプログラミング教育支援ツール pgtracer の開発と評価, 情報処理学会誌, Vol. 2, No. 2, pp. 20-36 (2016).
 [2] 小芦勇介ほか: 理解度モデルに基づく C 言語学習支援システムの設計と実装および試用, 情報処理学会研究報告, Vol. 2013-CE-119, No. 1 (2013).