

初級プログラミング教育におけるロボット制御の実習課題

高田 正之

江戸川大学 メディアコミュニケーション学部
情報文化学科 教授 情報工学

廣田 有里

江戸川大学 メディアコミュニケーション学部
情報文化学科 准教授 ソフトウェア工学

要 旨

情報文化学科の初級プログラミング教育においてJava言語によるロボット制御の課題を取り扱った事例を別途報告した。ここでは、ロボットを題材とすることで従来のプログラミング課題よりも学生の積極性を引き出すことができるだけでなく、動作結果が物理的に目に見えることがプログラミング教育に資すると主張した。本稿では、具体的な実習課題の内容を報告するとともに、学生の自己評価アンケートによって教育効果を分析した。ロボット課題採用によって、弱いながらも統計的に有意な効果が認められた。

キーワード：プログラミング教育, ロボット制御, ノンパラメトリック検定

1. はじめに

情報文化学科におけるプログラミング教育の一環として、平成25年度の情報文化演習・実習(2年次通年・必修)のうち情報コミュニケーションAコース14名の前期ぶんを充ててJava言語によるロボット制御の課題(以下、ロボット制作とよぶ)を取り扱った事例を江戸川大学紀要に報告した。^[1]

そこでは、学生の特性と課題の概要を説明し、ロボット制作の導入によって、従来のプログラミング課題よりも学生の積極性を引き出すことができ、動作結果が物理的に目に見えることが有効であったと報告した。

本稿では、具体的な課題の内容を示すとともに、事後に行なった学生の自己評価アンケートに基づいてプログラミング教育としての効果を分析する。

2. 環 境

用いたロボットは、「教育用LEGO マインドストームNXT」である。

本演習・実習以前(1年次)にプログラミング科目はなく、本演習・実習と並行してC言語による入門科目「プログラミングⅠ／Ⅱ」が2年前期／後期に開講されている。平成25年度はロボット制作の受講者14名中11名がプログラミングⅠを同時に受講した。

Java言語を扱う他の科目としては、翌年3年次の前期／後期に「プログラミング演習Ⅰ／Ⅱ」がある。

3. 実習課題の内容

ロボット制作の全体の流れ(とだいたいの授業回数)は、次のとおりである。

- (1) マニュアルどおりのロボット組立(0.5回)
- (2) Javaによるロボット制御の練習(9回)
- (3) ロボットの企画(1回)
- (4) 企画したロボットの組立とプログラミング(3回)
- (5) 実演プレゼンテーションと相互評価(1回)

このうち、(2)(4)のプログラミング課題としてのロボット制作の特徴は、主に開発手順と入出力に現れる。

- ① 開発手順：Java言語で記述したソースプログラムをパソコン上でコンパイルしてから、ケーブルでロボット本体へ転送して実行するという“組込みソフトウェア”の開発手順になる。
- ② 入出力：4種(音量, 光量, 距離, 接触)のセンサーからの入力に基づいて3個のモーターの回転を制御するプログラミングが基本となる。

組込みソフトウェアの開発手順は、パソコン単体でのプログラム開発に慣れた後では面倒に感じるが、最初にこれで慣れてしまえば問題はない。むしろ、コンパイルと(ロボット上での)プログラム実行の意味が、単体での開発時より納得しやすい利点もある。

学生は、プログラミング経験が皆無か、あるいは「プログラミングⅠ」でC言語を学び始めたばかりであるから、(2)の練習は普通の「例題の模倣と応用」による。しかし、抽象的に「変数とは」「型とは」「文とは」と文法要素をなぞるのではなく、“目に見える”入出力を、活かし、変数なども必要に応じて自然に導

入することで、効果的に習得させることができた。

具体的なプログラミング課題を次に示す。実際には、各自の進度や能力に応じて難易度を調節したり追加課題を出したりした。

- (1) ロボット本体の液晶画面に「Hello」と表示する。
- (2) 各センサーから読み取った値を画面に表示する。
- (3) モーターを一定時間だけ回す。
- (4) ランプ(モーターと同じ使い方)を点滅させる。
- (5) 進むべき距離をモーターの回転時間に換算する。
- (6) 光センサーで床面の白/黒を判別して回転を変える。
(この回から流れ図と動作説明を書かせた。)
- (7) 光センサー1つで床の黒線(の縁)に沿って走る。
- (8) 配列データとして与えられた“走行計画”に従った時間ずつ走ったり止まったりする。
- (9) 音階または短い曲を演奏する。
- (10) マルチタスクの機能を使って、明るさを表示しな

入力：距離センサー×2 (左右)

出力：モーター×2 (左右)

具体的な動き：

- ・ 近くの物を追いかけてネコパンチする
- ・ 左右両センサーに反応があれば前進
- ・ 片方に反応があればそちらに方向修正
- ・ 両方反応なしの場合はその場で待機

図1 「招かれ猫」ロボットの企画例

からジグザグに走る。

- (11) 演奏しながら黒線に沿って走る。
- (12) 接触センサーを押しボタンに見立ててリモコン操作で運転する。回転中はランプを点滅させる。

こうした練習課題を経た後の実際の学生作品の例として、「招かれ猫」ロボットの企画を図1に、組立結果を図2に、そしてプログラムを図3に示す。if文の使い方などに初心者臭さはあるものの、平均レベルの学生の入門半年のプログラムとしては十分な達成度といえよう。



図2 「招かれ猫」ロボットの組立例

<pre>import lejos.nxt.*; public class Project_Nyanko { public static void main(String[] args) { SubTask sub = new SubTask(); sub.start(); ←サブタスク起動 NXTMotor motorA = new NXTMotor(MotorPort.A); ←右輪モーター NXTMotor motorC = new NXTMotor(MotorPort.C); ←左輪モーター UltrasonicSensor sonicS1 = new UltrasonicSensor(SensorPort.S1); ←右距離センサー UltrasonicSensor sonicS4 = new UltrasonicSensor(SensorPort.S4); ←左距離センサー int reatiR, reatiL; motorA.setPower(70); ←出力70%に設定 motorC.setPower(70); while(true) { reatiR = sonicS1.getDistance(); reatiL = sonicS4.getDistance(); if(reatiR < 30) { if(reatiL < 30) { motorA.forward(); motorC.forward(); } else { motorA.backward(); motorC.forward(); } } else { if(reatiL < 30) { motorA.forward(); motorC.backward(); } else { motorA.stop(); motorC.stop(); } } } } }</pre>	<pre>class SubTask extends Thread { public void run() { NXTMotor motorB ←腕モーター = new NXTMotor(MotorPort.B); UltrasonicSensor sonicS1 = new UltrasonicSensor(SensorPort.S1); UltrasonicSensor sonicS4 = new UltrasonicSensor(SensorPort.S4); int reatiR, reatiL; motorB.setPower(35); while(true) { reatiR = sonicS1.getDistance(); reatiL = sonicS4.getDistance(); if(reatiR < 30 && reatiL < 30) { motorB.forward(); try { ←sleep 使用のため例外処理が必要 Thread.sleep(550); ←550ミリ秒待つ } catch(InterruptedException e) { } motorB.backward(); try { Thread.sleep(500); } catch(InterruptedException e) { } } else { motorB.stop(); } } } }</pre> <p>二つのクラスをマルチタスクとして同時に実行する。メインタスクでは30cm未満の距離に出された手などの目標物に向かって走行し、サブタスクでは目標物を正面に捉えたとき腕を振る。</p> <p>※プログラム右横の説明は高田による追記</p>
--	---

図3 「招かれ猫」ロボットのプログラム例

4. ロボット導入の効果測定

ロボット導入の効果測定を、学生の自己評価アンケートによって試みた。

ロボット群：14名

対照群：5名

平成25年度前期に本演習・実習Aコースでロボット制作を受講した14名に対して、後期最終授業で表1のアンケートを実施し、欠席者1名を除く13名の回答を得た。これに後述(1)の例外1名を加えた14名をロボット群とよぶ。

他方、3年次後期の選択科目「プログラミング演習Ⅱ」を受講した6名に対しても、最後に同じアンケートを実施した。このうち後述(1)の例外1名を除いた5名を対照群とよぶ。

これら2群を比較したわけであるが、次の点は注意が必要である。

- (1) プログラミング演習Ⅱの3年生のうち1名は、前年度の演習・実習で同様のロボット制作を（Javaの初体験として）体験していたことから、“時間差のある受講者”と考えて例外的にロボット群のほうに含めることにした。

この1名だけJava経験が2年目であるから、データに含めないのが通常の対処であろうが、該当の学生が前年度に示していた理解度、達成度がすでに十分なレベルであり、経験の長さが回答結果に影響していないと考えられることから、標本サイズを優先して上記の扱いとした。

この結果、ロボット群のほうに高い値を追加した形になってしまった。（14番の学生）

- (2) 対照群の中に、前期の「プログラミング演習Ⅰ」を受講せずに後期の「Ⅱ」だけ受講し、不足ぶんを独学で補おうとした学生が1名いた。それなりに努力してなんとか及第点は取れたが、自己評価アンケートの結果としては、対照群のほうに低い値を含めた形になってしまった。（4番の学生）

これら5段階評価のデータは間隔尺度とも見なしうるが、標本サイズが小さいこと、階数の少ない主観的評価で正規性が仮定できないことから、順序尺度と見なし非パラメトリック検定(Mann-WhitneyのU検定)によって分析することにした。

表1の結果のU統計量に対し、標本サイズ5と14、有意水準5%の片側検定の基準値は16であるから、それ以下の値を示した問2, 3, 4, 7で有意差が認められる。

上で有意差のあった問2, 3, 4はJavaの具体的な文の知識に対応し、問7は適切な動機づけができたことを示す。一方、有意差のなかった問1, 5, 6, 8はより高レベルの達成度や自信の程度を調べるものであるから、初級プログラミング教育の段階としては総じて好適な結果が得られたといえることができる。

ただし、今回の測定は標本サイズが小さいうえに、前述のとおり検定で有意差の出やすい向きに学生の質が偏った標本であった。また、検定は両側検定にすべきだったかもしれない、その場合、基準値は13であるから有意差があるのは問7だけとなる。これらのことから、さらなる検証が必要と考える。

5. まとめ

事例報告および関連する自己評価アンケートの結果を通して、動作が物理的に目に見えるロボットの制御を題材とすることで、初級プログラミング教育の効果を上げられる可能性を示すことができた。

参考文献

- [1] 高田正之・廣田有里(2014)「初級プログラミング教育へのロボット導入の試み」, 江戸川大学紀要, 第24号, pp.421-424

表1 評価アンケートと集計結果

1. あなたの Java プログラミング体験について程度を教えてください。 (1=ぜんぜんそう思わない ~ 5=そう思う の間の段階で数字に○をつける)									
	問1	問2	問3	問4	問5	問6	問7	問8	感想
問1	5	5	5	5	5	5	5	5	プログラミングを理解して、自由にロボットを動かせるようになった時は楽しかった。
問2	3	3	3	2	3	3	4	4	Java の基本がよくわからず若干苦手意識があるが、わかってくると面白い。 最初は全くわからなかったが、お手本を真似るうちに理解できるようになった。 プログラミングが面白くなった。もっと組んでみたい。 最初はわからなくて面白くなかったが、わかってくと面白く感じるようになった。 容易ではないが、処理のバリエーションが豊富で自由に制御できてよかった。 完璧かと聞かれると微妙なので4にした。忘れた部分もありそう。 Java の文は単純だが組み合わせで動作させるのが難しかった。完成の喜びが大きい。 楽しかった。難しい課題もあったが、達成感があってよかった。またやってみたい。 最初はわけがわからなかったが、初めてうまく動いた瞬間が一番楽しいと思えた。 思い通りに作れず先生に頼っていたが、しっかりと完成はできた。 プログラミングは奥が深いと思った。 ロボットを使うことによって敷居の高かったオブジェクト指向言語に親しめた。
問3	5	4	3	3	4	2	5	4	
問4	5	5	5	5	5	5	4	5	
問5	3	4	4	3	3	2	4	3	
問6	5	4	5	5	5	4	5	5	
問7	5	5	5	5	4	4	5	4	
問8	4	4	4	4	4	4	4	5	
問9	4	5	5	5	5	2	4	4	
問10	4	4	3	4	4	2	5	2	
問11	5	5	4	5	5	5	4	4	
問12	3	3	3	2	2	4	4	3	
問13	4	3	3	3	4	2	3	3	
問14	5	5	5	5	5	3	5	5	
対照群1	4	3	3	3	2	2	3	2	
対照群2	4	4	4	3	5	3	4	4	理解できないまま終わった部分があった。復習して理解を深めたい。
対照群3	4	3	2	2	3	2	3	3	Eclipse のトラブルが謎だった。コマンド(dir など)の意味をもっと知りたかった。
対照群4	2	1	1	1	1	2	3	2	基礎がわかれば楽しかったらうになと思うことが多かった。
対照群5	4	4	4	4	4	3	4	4	他のプログラム言語と比べて Java は多少記述しやすかった。更に学びたい。
U統計量	20	14	15.5	14	19	19.5	10.5	16.5	