

実機ロボットによるタートルグラフィックスを通じたプログラミング学習

Programming Learning through Turtle Graphics with Real Robots

右田 正夫
Masao MIGITA
滋賀大学教育学部

島田 拓哉
Takuya SHIMADA
滋賀大学教育学部附属中学校

<キーワード> ロボットプログラミング タートルグラフィックス グループ学習 教材開発

はじめに

中学校技術科におけるロボットを使ったプログラミング学習は、学習指導要領に明記されている計測・制御システムの文脈で実施されるのが普通である（小林ら，2003；文部科学省，2017）。小学校プログラミング教育において、教科の単元で実施可能な活動の代表例ともなっているタートルグラフィックスは、元々は実機ロボットとPCの両方で使われ、仮想ロボットと実機ロボットを併用することによって、複数の世界で成立する抽象的な概念に関する学習、いわば数学的な思考力を獲得することが指向されていた（パパート，1982）。タートルグラフィックスにおいては主に幾何学的な思考を深めることが期待される。

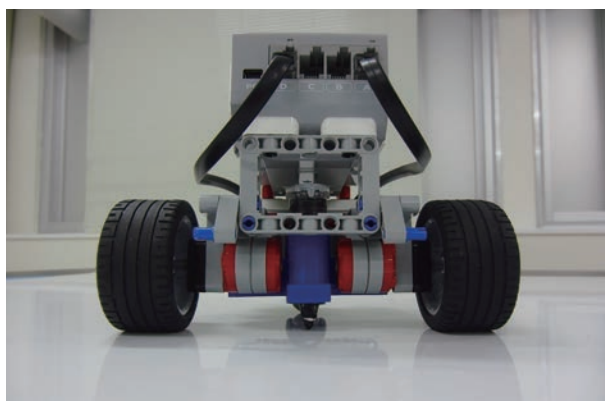
しかし、タートルグラフィックスには「計測」の要素がなく、実機ロボットにおいてもセンサーを必要としないため、中学校技術科の学習指導要領に沿った題材とは言いがたい。実機ロボットを使ったプログラミングとして、センサーからの情報に応じてモーターを制御するライントレーサーなどは好適な題材と言える。しかし、パパートが論じたような、学習の道具としてのタートルグラフィックスにも捨てがたい魅力がある。タートルグラフィックスを通じて図形を描く場合、学習者はタートルの向きと自身の向きを重ね合わせて考えることにより、タートルに図形を描かせるためのプログラムを着想しやすくなる。タートルの動きを自身の動きに置き換えて考えることに基づく学習をパパートは同調的学習（syntotic learning）と呼んだが、同調性（対象の身になって考えること）を引き出せるかどうかプログラミングの理解のしやすさにとって重要だという考えもある（Watt，1998）。タートルグラフィックスに限って考えると、物理的な存在である実機ロボットの方がPC上の仮想的なロボットよりも同調的学習を引き出しやすいということも考えられる。このような関心のもとで、本研究では中学校技術科の授業において実機ロボットによるタートルグラフィックスを取り扱うことにした。仮想的なロボットと実機ロボットとを同等のプログラミング環境で制御できるようにScratchとMindstormsを使用

して教材開発を行っているが、実機ロボットならではの制約もあり、完成には程遠い状態である。本報告では、昨年度より行っている実践について報告するとともに、教材および実践の改善点について考えてみたい。

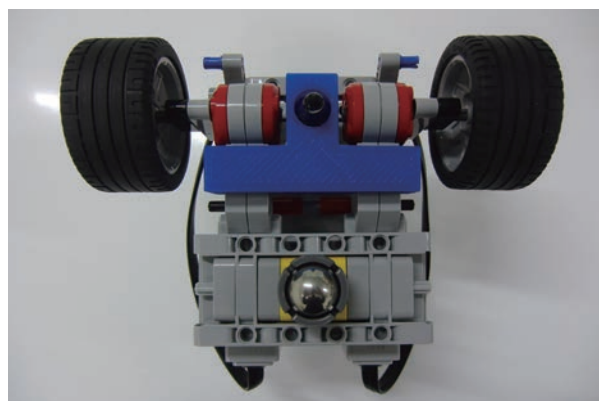
プログラミング教材 機材について

タブレット端末はASUS社のT101-HA（ディスプレイ10.1インチ、メモリ4GB）で、数値の入力をスムーズに行えるように付属のキーボードを接続して使用した。OSはWindows10で、事前にMindstorms EV3の本体（インテリジェントブロック）とペアリングした。Scratchのバージョンは3.0で、Mindstorms EV3との通信のためにScratch Link（バージョンは1.4）を用いた。移動ロボットのデザインはレゴ社が提供するMindstorms EV3のトレーニングモデルをベースにしたもので、ロボット回転時の回転半径ができるだけ小さくなるように、車輪等を配置した（図1）。図1（a）はロボット前方の外観であるが、左右1本ずつのケーブルがモーターに接続されている。写真右側（ロボットの左側）のケーブルがAポートに、写真左側（同、右側）のケーブルがDポートに接続されている。本実践のプログラミング課題においては、2つのモーターAとDの制御に関する数値の設定を行う。タートルグラフィックスのプログラミングでは、タートルの直進距離と回転角度を与えるため、この点において本実践のロボットは正式なタートルグラフィックスを行うものではない。これには以下のような事情があった。

LEGO社が提供する制御用のソフトウェアEV3 Classroomでは、モーターの回転速度、回転回数、回転角度、回転時間を指定することができる。この点は、左右の車輪の回転を調節することでロボットに一定の前進・回転の動作を行わせる上で有利である。一方、Scratchではモーター出力のパワー（%で指定）と回転時間（秒で指定、小数も可）は設定できるものの、回転回数や回転角度は指定できない。パワーの設定値と実際の出力はバッテリーの残量にも依存し、そのことによ



(a)



(b)

図 1. ロボットの前面 (a) と下側 (b) から見た写真。駆動輪の中央にペンが取り付けられている。

て回転時間の設定に対する走行距離も不安定になると考えられる。このことは後述するように、ロボットに多角形を描かせる際の調整を難しくした。しかし、本研究を始める際の動機として、プログラミング学習で実機ロボットとシミュレータを使いそれぞれの効果を比較することがあったので、シミュレータも自作可能な Scratch によって制御している。

実機ロボットを使ってタートルグラフィックスを行う上で非常に重要なのがペンである。ロボットが角を曲がる時ペンの位置を軸として回転できるように、2つの駆動輪を結ぶ線の中点にペンが置かれるのが理想的である。EV3の本体下部から床までの距離が7cmほどしかなく、市販のホワイトボードマーカーにはそれほど短いものが見当たらなかったため、遠心分離などに用いられるチューブの先端に穴を開け、そこにペイントボードマーカーの替え芯を差し込んで補充用のインクを入れてペンとした。ペンをロボットに固定する部品は、3次元CADソフトウェアの123D Designで設計し、3DプリンタMUTOH MF-2200Dで作成した。

後述の2022年度に行ったアンケート調査で、改善を要する点について尋ねたところ、ホワイトボードマーカーについて、ロボット本体の取り付けやインクの出方が不安定であることを指摘した回答が多かったため、ペンを固定する部品とペンの中綿の変更を行い、ロボットの動きに対するグラフィックスの安定性が若干向上した。

授業実践

本研究に関する授業は、2022年12月13、14、19、20、23日、2023年6月28日、7月7日に、各クラス1回ずつ行われた。授業時間は2023年6月28日だけが50分で、他は全て45分であった。授業時間の長短が生徒がロボットに触れる時間の差につながらないように配慮した。

技術科の授業は1クラス18名までで、生徒は5つのグループに分かれて学習する。各グループは原則3名または4名の生徒で構成されていたが、本研究の授業

時には生徒の欠席等の事情で少数ではあるが2名からなるグループもあった。

授業開始後、生徒に充電済みのタブレット端末とMindstorms EV3を配布した。生徒たちは、Scratch 3.0とScratch Link、および、Mindstorms EV3とを起動すると、Scratchの拡張機能の中からMindstorms EV3を選択してタブレット端末とロボットの接続を行った。教室に複数のMindstorms EV3が存在するため、接続時に複数のロボットが見える状態になるが、電波の最も強いものを選択すれば問題なく接続することができた。

授業時の課題は、移動ロボットの動作をScratchでプログラムし、ロボットの下部に取り付けられたホワイトボードマーカーによって正三角形を描く、というものであった(図2)。

タートルグラフィックスで正多角形を描くという課題自体は小学校のプログラミング教育でも取り扱われているものであり、本研究の対象となった生徒は少なからず知っているものと考えられる(文部科学省, 2020)。しかし、上述のように本実践のロボットのプログラミングでは、前進の距離と回転の角度そのものの設定は出来ないため、ロボットが回転する際のモーターのパワーと回転時間の数値を設定する「基本プログラム」をタブレットに用意しておいた(図3)。基本プログラムはロボットが正三角形を時計回りに描くことを目的として作られている。モーターAは左側のタイヤを駆動し、回転時に順方向(前進時と同じ方向)に回転する。モーターDは右側のタイヤを駆動し、回転時には逆方向に回転する。基本プログラムの設定(左右のモーターのパワーが10、すなわち10%、回転時間が2.7秒)のままで、ほぼ正三角形が描けるはずであるが、授業で使用したロボットには“機体差”とでもいうべき違いがあり、同じプログラムであっても描かれる図形がグループによって異なってしまう。現時点では、この機体差の原因について、詳細に調べられていないが、タイヤの滑り具合、バッテリーの残量の変化、ペンによる摩擦の違い、など複数の要因が考えられる。そのため、描かれる図形を見て、上手く

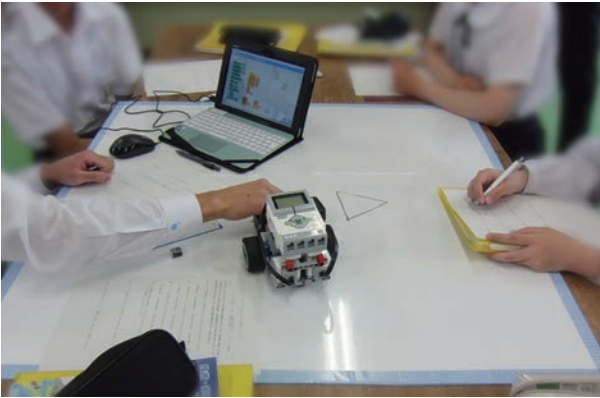


図 2. グループによるプログラミングの様子。

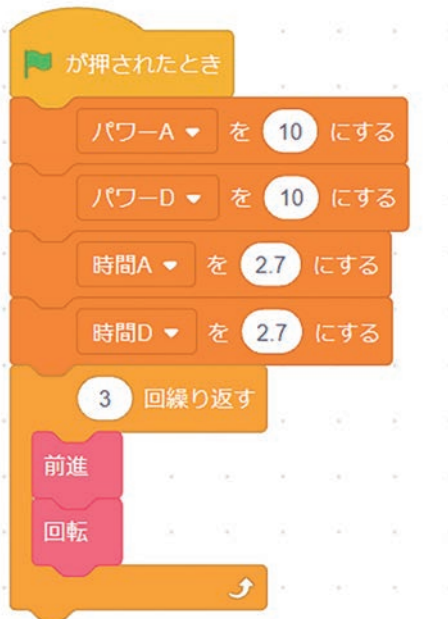


図 3. 授業時に用意した基本プログラム。

描けていなければその状態に応じて数値を調整せざるを得ない。そこで、生徒たちがプログラミングに取り組む前の説明において、タブレットとロボットとの接続方法の説明に続けて、ロボットが描く図形の角の状態に応じて、数値をどのように調整すればよいか、大まかな方針を示した(表 1)。なお、説明中でパワーを最大 12%までとしているのは、タイヤを高速で回転させることによる滑りを防ぐためである。また、生徒が設定する数値ではないが、直進時のパワーも 10%と低く設定している。これは、ホワイトボードが机上に置かれているため、主に落下の防止を考えてのことである。

以上のようなプログラミングの手順の説明の後、各グループで最もきれいな正三角形が描けたプログラムを保存するように指示した。実験の終了時に、各グループによるプログラム中の数値をクラス全体で共有し、他のグループの設定にすることで描かれる図形がどのように変わるか観察した。

表 1. 事前説明で示された調整の指針。

ロボットが回転するときの左右のモーターのパワーと回転時間をうまく調節するとききれいな三角形が描けます。
プログラムを実行してみて、

1. 曲がりすぎるとき パワーを下げるか回転時間を短くしてください。
2. 曲がり足りないとき パワーを上げるか回転時間を長くしてください。
3. きれいな角が描けないとき 左右のパワーを変えてみてください。

パワーは最大で 12 までにしてください。

アンケート調査

授業の最後に、課題の達成度やロボットを使ったプログラミング学習に関するアンケート調査を行った。各年度の質問項目の概要を表 2, 3 に示す。授業の終了前に短時間で記入してもらうため、質問項目をできるだけ少なく以下のようなものとした。教材として使用するために重要であると思われる操作法のわかりやすさ (Q1)、課題の達成度 (Q2)、ロボットを使ってプログラミング学習を行う楽しさや困難さ (2022 年度の Q4, Q5, 2023 年度の Q5, Q6)、タブレットやロボット、ペンとホワイトボードなど、性質の異なる機材を扱うグループ学習であるため、グループ内で役割分担ができたか (Q3)、については両年度で尋ねることにしたが、2022 年度では改善してほしい点 (Q6)、2023 年度では役割分担の内容 (Q4) というように、一部の質問項目を変更している。改善してほしい点は 2023 年度も聞きたい項目ではあったが、時間的に質問項目を増やすのは困難と考え割愛した。

結果と考察

ロボットを用いた図形描画課題

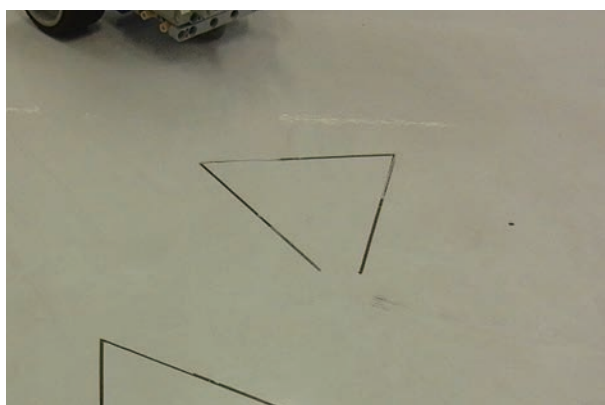
各クラスの生徒が用意された基本プログラムを実行し、図形を改善するために数値を変更していく過程で、様々な「きれいな三角形を描けない」事例が観察された(図 4)。ロボットが停止した時点で角が閉じない(図 4 (a))、最初と最後の辺が交差してしまう(図 4 (b))、などの場合は、それぞれ、左右のモーターの回転時間を長くしたり、短くしたりすることによって改善できそうなことには比較的気づきやすいが、角が丸くなったり(図 4 (c))、辺が交差したり(図 4 (d))、といった場合で、全体としては図形が閉じた形になっていると、どの数値をどのように変えればよいかのかわりにくい。事前説明では、「きれいな角が描けないとき 左右のパワーを変えてみてください」という情報が与えられていたが、調整の指針としてはわかりにくいものであったかもしれない。図 4 (c) のように角が丸くなるのは回転時に外側

表 2. 2022 年度の質問項目。Q1 から Q3 までは 5 段階評価, Q4 以降は自由記述で回答。

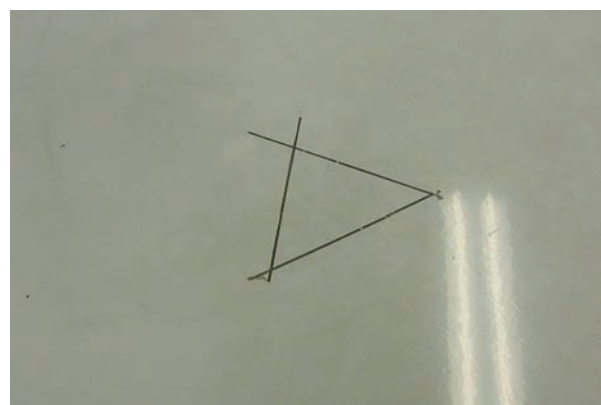
Q1	ロボットの操作法は理解できたか
Q2	図形はうまく描けたか
Q3	グループ内で役割分担はできたか
Q4	楽しいと感じた点
Q5	難しいと感じた点
Q6	改善してほしい点

表 3. 2023 年度の質問項目。Q1 から Q3 までは 5 段階評価, Q4 以降は自由記述で回答。

Q1	ロボットの操作法は理解できたか
Q2	図形はうまく描けたか
Q3	グループ内で役割分担はできたか
Q4	役割分担ができた場合, 自分の役割は何か
Q5	楽しいと感じた点
Q6	難しいと感じた点



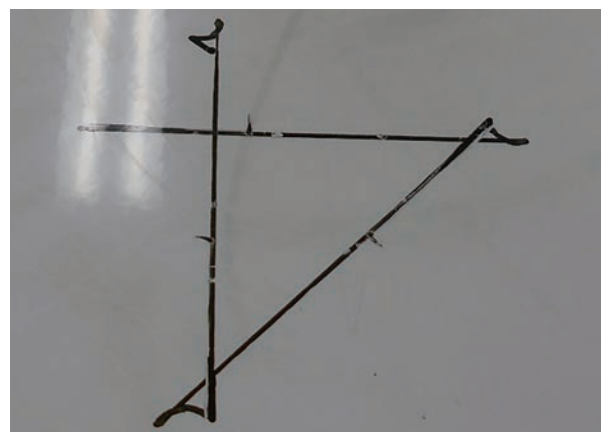
(a)



(b)



(c)



(d)

図 4. 数値の調整が必要な描画のパターン。(a) 回転不足, (b) 回転し過ぎと 1 つの角が交差, (c) 角が丸い, (d) 回転し過ぎと全ての角が交差。

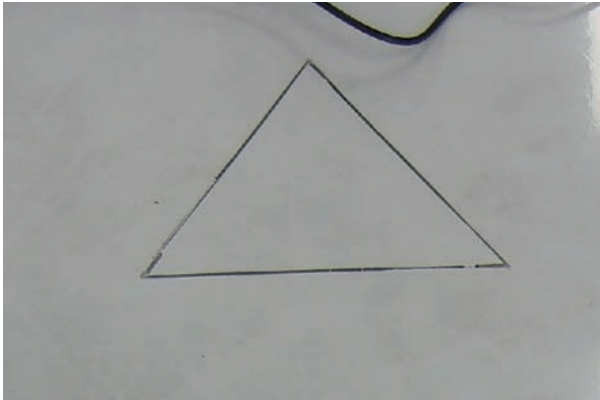
のモーターの回転が速いか回転している時間が長いという原因によるものと考えられるので, モーターの数値(パワーと時間のどちらか, あるいは両方)を調整して補正するとよいはずである。ただ, 補正が行き過ぎるとまた別の問題が生じることもある。一方, 角において辺が交差するケース(図 4 (b, d))では回転時に逆回転するモーターの数値(パワー, 時間)を絶対的または相対的に下げよう補正するとよいはずであるが, 修正が効きすぎると逆効果になるのは角が丸いケースと同じである。

数値の調整が上手くできるとききれいな正三角形が描け

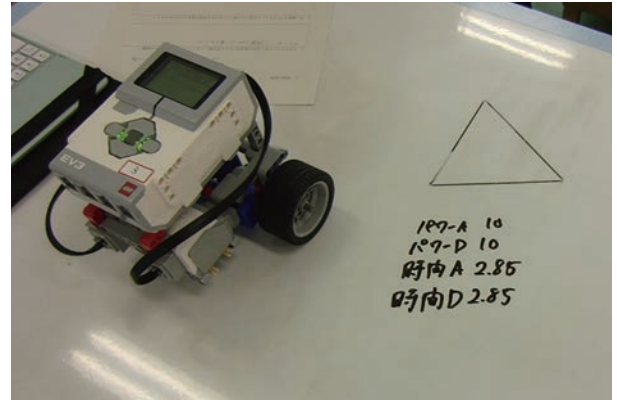
る(図 5)。PC 上のタートルグラフィックスでは当たり前のことであるが, 実機で様々な調整を経てそうした結果が得られるとそれなりに達成感があるようである。後述のアンケート調査でもそのような回答が見られた。

アンケートへの回答について

授業の最後に実施したアンケート調査において, 質問項目 1 から 3 までの結果を図 6 に示す。質問 1 のロボットの操作法の理解に関する回答では, 2022 年度では「ややそう思わない」以下の否定的な回答と「ややそう思う」

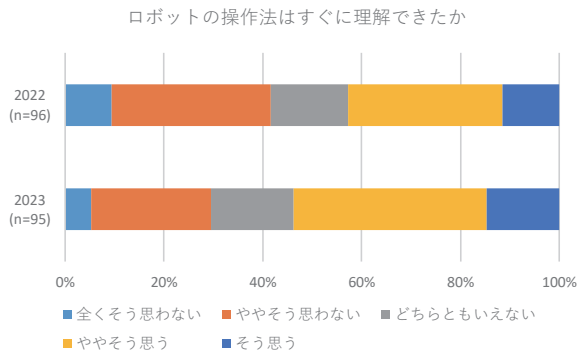


(a)

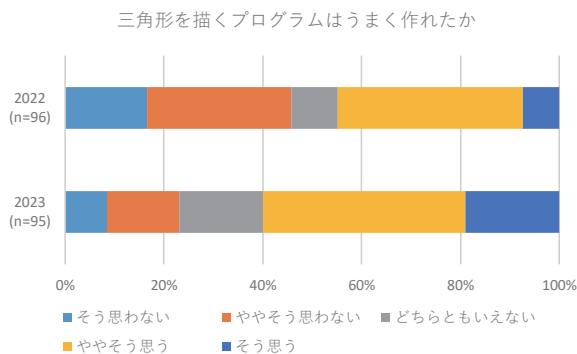


(b)

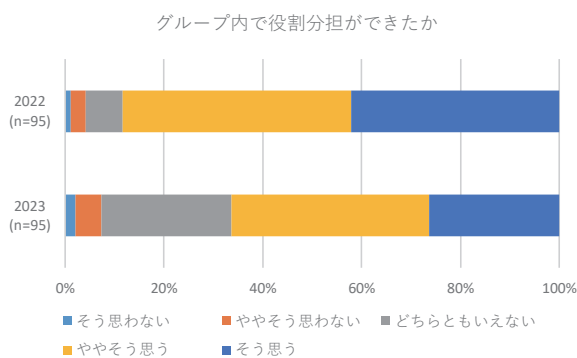
図 5. きれいな正三角形が描けた例。(b) には設定数値 (A, D 共にパワー 10, 回転時間 2.85) が記入されている。



(a)



(b)



(c)

図 6. 質問項目 1-3 への回答。(a), (b), (c) のそれぞれにおいて、2022 年年度 (上) と 2023 年年度 (下) の結果を示す。

以上の肯定的な回答がそれぞれ 41% と 42% と拮抗していたのが、2023 年度ではそれぞれ 29% と 54% と、肯定的な回答が増えていたが、統計的に有意な差は見られなかった (ウィルコクソン順位和検定, $P=0.08$)。事前説明には両年度で同じスライドを用いたため、実験を始める前に個々の生徒が持つ知識の内容としては違いはないはずである。ただし、授業をする教師側の慣れによって、よりわかりやすい説明がなされていた可能性もある。また、2022 年度の質問項目にあった改善してほしい点に関する回答に基づいて、一部の機材が改良されており、そうしたことが操作法のわかりやすさにつながったのかもしれない。

課題の達成度に関する質問 2 では、2022 年度で否定的な回答と肯定的な回答が 46% と 45% と、質問 1 と同様に拮抗していたが、2023 年度ではそれぞれ 23% と 60% と、きれいな正三角形ができたという回答が増えていた。否定的な回答は、2023 年度には半減しており、両年度の回答には有意な差が見られた ($P=0.0020$)。

役割分担ができていたかを問う質問 3 では、「ややそう思う」以上の肯定的な回答が 2022 年度には 88% だったが 2023 年度では 66% と減少していた ($P=0.00085$)。

自由記述の質問に対する回答はテキスト分析ソフトウェア KH Corder 3 で出現上位 30 語の対応分析を行った。まず、ロボットを用いたプログラミングの楽しい点は 2022 年度の質問 5 に対しては試行錯誤することと自分のプログラムでロボットが動くことが多く挙げられていた (図 7)。2023 年度では、ロボットが動くことときれいな正三角形が描けることが挙げられていた (図 8)。こうした結果は質問項目 2 の回答とも対応しているものと考えられる。

難しい点 (2022 年の質問 5, 2023 年の質問 6) に関しては、2022 年度ではプログラムとロボットの動きの対応がわかりにくいことが多く挙げられていた (図 9)。これはペンの固定やインクの出具合の不安定さにもよるものかもしれない。一方、2023 年度では数値の調整の難しさが挙げられる傾向が見られ、現状の教材において

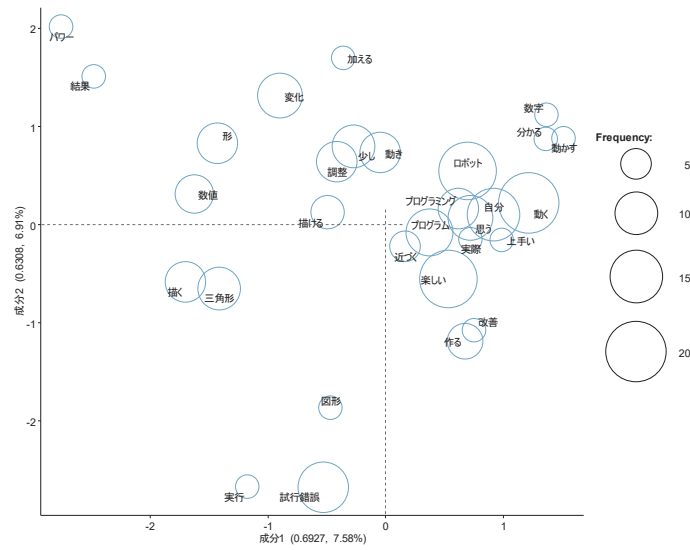


図 7. 2022 年度の楽しいと感じた点に関する自由記述回答の対応分析の結果。

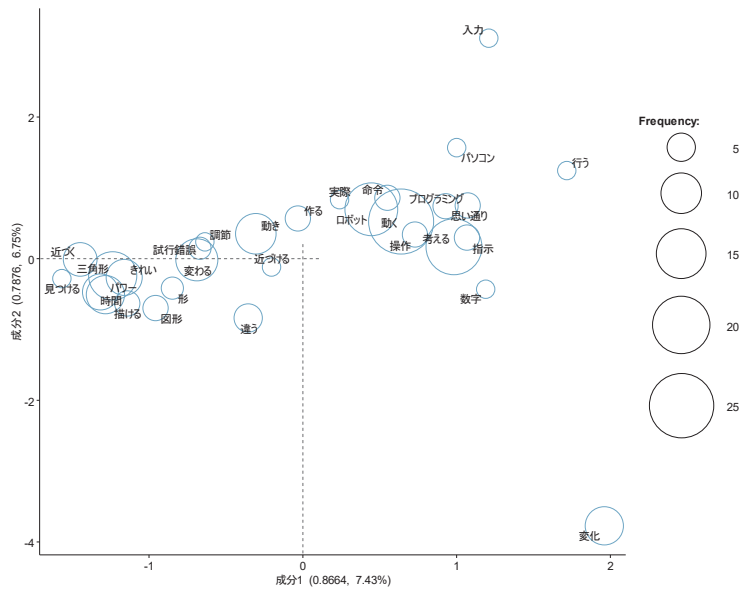


図 8. 2023 年度の楽しいと感じた点に関する自由記述回答の対応分析の結果。

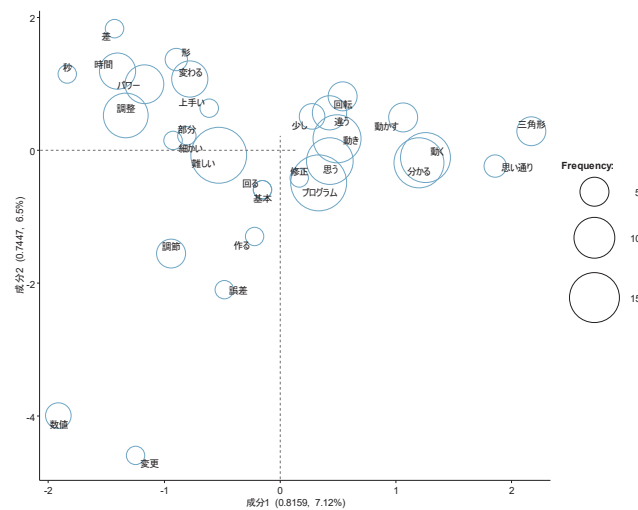


図 9. 2022 年度の難しいと感じた点に関する自由記述回答の対応分析の結果。

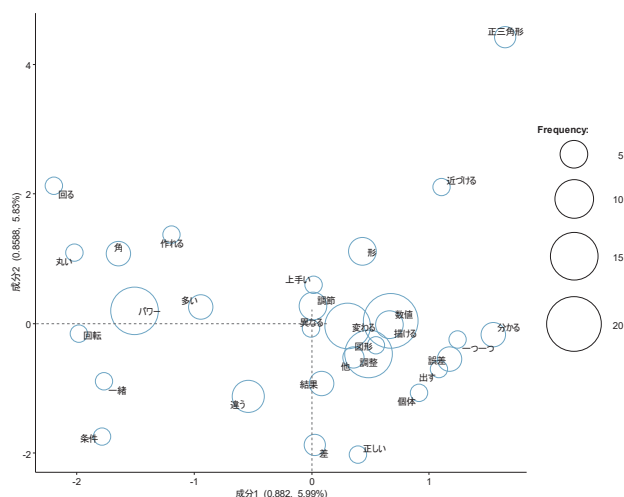


図 10. 2023 年度の難しいと感じた点に関する自由記述回答の対応分析の結果。

は、「プログラミング＝数値の調整」であるため、プログラミング課題としての難しさを挙げる生徒が増えたということと考えられる。2022 年度に挙げられた改善点には「タイヤがすべる」ことも複数の生徒から挙げられており、タイヤのグリップもまたプログラムと動作との関連をわかりやすくする上で重要なポイントになる。

おわりに

最後に、本実践の機材についてグループ学習の教材という観点から考えてみたい。プログラミングの課題では、タブレット（プログラミングと実行）、ロボット（ホワイトボードの適切な位置に置く、ペンの高さを調整する）、図形を消す、など複数の作業があり、グループ内で自然と各生徒の役割が決まっていくように見えた。数値を微調整しながら実験を繰り返す上では効率がよいかもしれないが、PC 上でのプログラミングと比べて、個々の生徒がプログラムの内容について考える時間は少なくなる。中学校技術科のプログラミング学習としては、本研究の実践のような「結果が機械の動作に反映されるために、プログラムの改善には機械の動作の観察や物理的環境の整備が必要になる」といった PC 上でのプログラミングとは大きく異なる状況はある程度までは許容されるかもしれない。しかし、グループ学習で役割分担が明確にされると、プログラムには関与しない生徒が生じてしまうということは問題であり、実践の方法を工夫する必要がある。

謝辞

本研究の一部は科学研究費補助金基盤 C 課題番号 21K02855 の助成を受けて実施された。

参考文献

小林靖英, 山本利一, 軽部禎文, 高岡暁子, 山崎知恵 (2003) 中学生の思考過程の分析とプログラミング学習カリキュラムの改善, 情報教育シンポジウム論

文集 2003 (12), 99-106.
 文部科学省 (2017) 中学校学習指導要領 (平成 29 年告示) 解説 技術・家庭編.
 文部科学省 (2020) 小学校プログラミング教育の手引 (第三版).
 シーモア・パパート (奥村貴世子 訳) (1982) マインドストーム 子ども, コンピューター, そして強力なアイデア, 未来社.
 Watt, S. (1998) Syntonicity and the psychology of programming, Psychology of Programming Interest Group 10th Annual Workshop., https://www.researchgate.net/publication/232062764_Syntonicity_and_the_psychology_of_programming