

SorToio：小型群ロボットを用いた アルゴリズム学習支援システムの試作

板垣 智也^{1,a)} 塚田 浩二¹

概要：近年、プログラミング教育の早期化にともなって、様々なプログラミング教材が開発されてきた。その中でも、手に取って扱えるタンジブルインタフェース (TUI) を使ったプログラミング手法が多数提案されている。一方で、ソート等のアルゴリズム学習を支援するための TUI は少ない。そこで本研究では、小型の群ロボットである toio を物理的に動かすことで、データ構造やアルゴリズムの学習を支援するシステム「SorToio」を提案する。各ロボットに値を持たせ、ロボット1台を定数、それらの並びを配列として扱うことで、配列内の要素の比較や交換をロボットの動作で表現することが可能になる。本稿では、バブルソートとヒープソート、データ構造 (スタック・キュー) を toio の動きで実装した。予備実験を通して、提案システムの基礎的な効果を検証し、課題や改善点を議論する。さらに、初学者、特に子供が自然とアルゴリズムに興味を持てるような応用について述べる。

SorToio: Prototyping of Algorithm Learning System using a Small Swarm Robot

1. はじめに

現在、情報教育の早期化が進んでおり、小学生でも利用できるような初学者向けのプログラミング教材が多数開発されている [1] [2]。それに伴い、実際に手で触れられるような物体「タンジブルインタフェース (TUI)」を使ったプログラミング手法が開発されてきた。例えば Project Bloks [3] のようなブロックを組み合わせるものや、GoGo ロボットプログラミング [4] のような小型ロボットを使ったものなど様々である。TUI を使うメリットとしては、物理的なオブジェクトを手にとって動かすことで GUI よりも直接的な操作が可能である点や、マルチユーザ・マルチハンドで操作できる点等があり、これらは初心者や子供向けの設計に向いている。

このように TUI を使ったプログラミング手法の研究が進む一方で、TUI を使ったアルゴリズム学習手法の提案は少ない。アルゴリズム学習は情報分野でも重要性が高く、高校では整列や探索などのアルゴリズムを体験的に理解させるように学習指導要領 [5] で定められている。情報教育

の早期化がさらに進めば、小学教育でのアルゴリズム学習が必修化される可能性があり、子供向け教材開発の重要性が高くなると考える。

そこで本研究では、小型群ロボットである toio を物理的に動かすことで、データ構造やアルゴリズムの学習を支援するシステム「SorToio」を提案する。本システムでは、各ロボットに値を持たせ、ロボットの並びを配列として扱うことで、配列内の要素の比較や交換をロボットの動作として表現することが可能となる。また、システムの制御やクイズ機能を小型ロボットへの物理的な操作によって行うことで、入力と出力が一体化した、分かりやすいアルゴリズム学習支援システムを目指す。

なお、本稿は WISS2024 でデモ発表 [6]、及び UIST2025 でデモ発表 [7] した内容を発展させたものである。

2. 関連研究

本研究の関連研究を「小型ロボットを用いた情報提示」、「タンジブルインタフェースによる情報教育」、「アルゴリズムの学習支援」の観点から整理する。

¹ 公立はこだて未来大学
Future University Hakodate

^{a)} itagaki8931@gmail.com

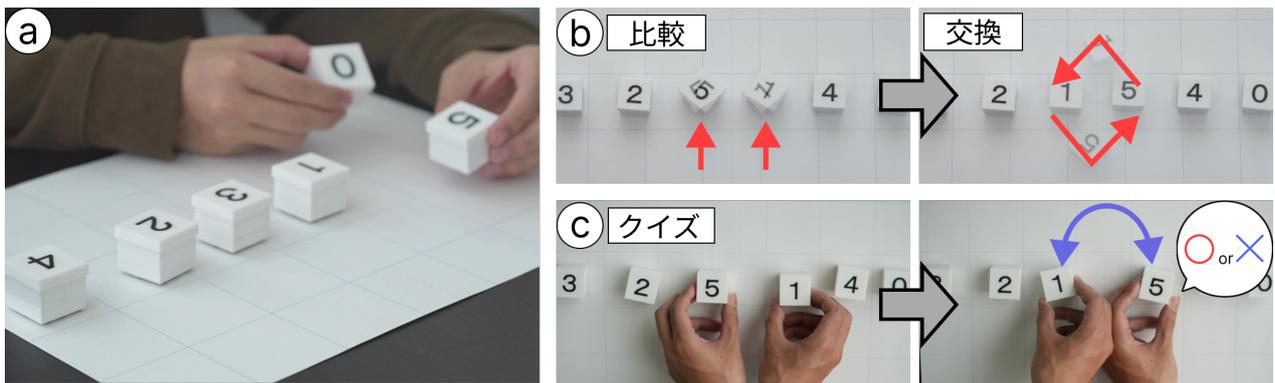


図 1 SorToio のコンセプト, (a) SorToio の使用イメージ, (b) 配列要素の比較・交換, (c) クイズモードの様子

2.1 小型ロボットを用いた情報提示

Zooids [8] は円柱型の車輪付き小型ロボットを多数利用することで、群ロボットを用いたインタラクション手法を提案した。例えば、グラフの点や線を Zooids でプロットすることで、時間経過によってデータが更新される動的なグラフを実世界上に再現した。

Sketched Reality [9] は AR スケッチツールで描かれた仮想スケッチ（線、壁、振り子、バネなど）と実世界の小型ロボットの toio が、同期して動作するシステムである。仮想世界と物理世界の双方向からインタラクションを行うことで、タンジブルな物理教育や機構学習などの可能性を示した。

TangibleNegotiation [10] は群ロボットと画像生成 AI を用いた、子どもと AI の共創支援システムを提案した。図形を載せた小型ロボットを上空から撮影し、その画像をもとに生成 AI がイラストを制作する。ユーザが小型ロボットを動かすことでイラストを調整したり、PC から小型ロボットを動かすことで創造へのアシストをすることが可能となる。

これらの研究は小型ロボットを物理演算された振り子やペイントツール上の図形、プロットされた点などとして現実世界で可視化・操作できるようにすることで、ユーザの容易な理解や直接的な操作を助けている。本研究では、配列の要素やポインタを現実世界で可視化・操作させることで、アルゴリズム学習を支援する。

2.2 タンジブルインタフェースによる情報教育

Plugramming [11] はセンサやアクチュエータ等の機能を持ったブロック同士をプラグケーブルでつなぐことで、プログラムを作成することができる。例えば、センサの値によって条件分岐をさせることで、周辺の環境によって動作が変わるプログラムを作成できる。

Project Bloks [3] はブロックを組み合わせることでプログラミングを行い、プログラミングの理解を遊びながら自然に学ぶツールである。ブロックはダイヤルやセンサなどの機能

を持つ。ユーザはブロック同士を接続したり、条件を付与していくことで、プログラミングの概念やアイデアの表現方法を学習する。

トイオ・プレイグラウンド [12] は、タンジブルなカードを組み合わせることで、小型ロボット toio の動作を制御することができる。各カードには ID が付与されており、toio 下部のイメージセンサで読み取る。「こっちを向く」「はやくまっすぐ」といったカードの上を toio が通ることで、その挙動が動的に変化する。また、ループ等のカードもあり、ごく基礎的なプログラミング制御も体験できる。

このように TUI を用いたプログラミング支援手法は多く提案されているが、本研究では、アルゴリズム学習に注目し、TUI による初学者向け情報教育ツールを開発する。

2.3 アルゴリズムの学習支援

コンピュータを使わずにアルゴリズムを可視化する例として、Computer Science Unplugged (CS アンプラグド) [13] がある。ソートアルゴリズムや探索アルゴリズム、オートマトンを、数字の書かれたカードやピンポン玉などを用いてゲームをする感覚で学ぶ教育法である。多言語に翻訳されており、世界中の教育現場で導入されている。

VisuAlgo [14] は Web ベースで作成された、アルゴリズムの可視化ツールである。多くのデータ構造とアルゴリズムに対応しており、配列やノードの値は、ユーザが任意に設定できる。

ARCat [15] は、TUI と AR を利用してアルゴリズム学習を支援する研究である。モバイルデバイスを用いた AR 上に迷路と猫を表示し、ユーザがタンジブルなカードを使って、猫をゴールに導くゲームを実装している。このゲームで遊ぶことにより、深さ優先探索 (DFS) の概念を学ぶことができる。

本研究では、小型ロボットに直接触れて操作することで、配列の編集がスムーズにできる点、現実世界のみで学習体験を完結できる点で利点がある。

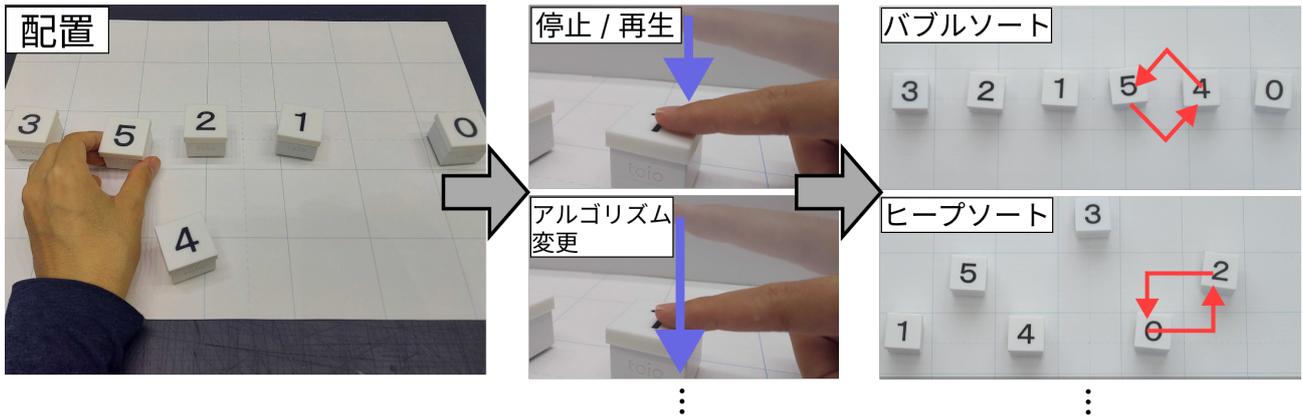


図 2 SorToio の利用プロセス

3. 提案: SorToio

本研究では、手に取って扱える小型の群ロボット (toio) を物理的に動かすことで、データ構造やアルゴリズムの学習を支援するシステム「SorToio」を提案する。まず、各ロボットに値を持たせることで、ロボット 1 台を定数、その並びを配列として扱えるようにする。次に、配列内の要素の比較や交換をロボットの動作で表現する。ユーザがロボットを配置して、アルゴリズムを選択すると、ソート等のアルゴリズムをロボットの物理的な動きとして観察できる。さらに、ユーザがロボットに触れることで、システムの動作制御や、配列要素の入れ替え、クイズ機能による試行錯誤等を行うことができる。

本システムの提供する表現とインタラクションの基本機能を以下に紹介する。

表現

- 配列要素の比較: 左右に揺れる (図 1b 左)
- 配列要素の交換: 位置を交換する (図 1b 右)
- 開始と終了: 順番に傾き、同時に元に戻る

インタラクション

- アルゴリズムの再生・停止: ロボットをタップする (図 2 中央上)
 - アルゴリズムの変更: ロボットを下に押し込む (図 2 中央下)
 - クイズモードの開始: ロボットを持ち上げる (図 1c)
- クイズモードを起動すると、アルゴリズムの再生が停止され、次の要素交換をユーザが手動で実施する。操作が正しければ、ロボットは喜ぶような動きを見せ、間違っていれば、元の位置に戻ろうとする動きを見せる。ロボットは軽量でトルクが低いため、ユーザは簡単に動きを止めたり持ち上げたりすることが可能である。

3.1 システム構成

本システムは、小型ロボットの toio、toio 開発用プレイ

マット、制御用 PC を中心に構成される。事前準備として、各 toio の個体 ID (Shortened Local Name) と任意の定数を、PC 上のソフトウェアで対応付ける。標準設定としては、 $1 \sim n$ ($n = \text{toio}$ の台数) を割り当てる。各 toio が持つ値が容易に判別できるよう、toio 上部に値が書かれたラベルパーツを取り付ける。

ユーザの利用プロセスを図 2 に示す。ユーザはプレイマット上の任意の位置に toio を配置し、toio 自体を下に押し込むことで、アルゴリズムを選択する。アルゴリズムの種類は toio に搭載されたフルカラー LED の色で表現し、補助的に制御 PC 上でも確認できる。toio をタップすると、各 toio の左右方向の座標に基づいて配列が初期化され、アルゴリズムの再生が開始される。その後、ユーザの操作に応じて、停止や逆再生、クイズモードへの移行等が可能となる。

3.2 表現対象のアルゴリズム・データ構造

本システムでは、値を持った小型ロボットの移動でアルゴリズムを表現する。この特徴により、配列要素同士の比較や交換を行うソートアルゴリズムや、配列要素の入出力順序が異なるデータ構造などを表現することが可能である。

本論文では、バブルソートとヒープソートの 2 つのアルゴリズムとスタック・キューの 2 つのデータ構造を、toio の動きで表現する。バブルソートはソートの中でも単純なアルゴリズムであり、大学などの講義でも最初に学ぶことが多いため、初学者向けのアルゴリズム学習に適すと考えた。ヒープソートは、二分ヒープと呼ばれる木構造を利用したソートアルゴリズムである。高速なソート手法の一つである一方、配列を木構造として捉える必要があるため、一般的に初学者にとって理解が難しいとされる。本システムでは、toio を二次元的に動かすことで、配列と木構造を直感的に表現できるため、ヒープソートの仕組みをより理解しやすくなると思った。スタック・キューは、基本的なデータ構造の一つで、多くのプログラミング言語で実装されている。大学の講義でも、初めに学ぶデータ構造である

ため、初学者向けのデータ構造の学習に適すると考えた。

4. 実装

4.1 ハードウェア

前述したように、本システムでは、小型自走ロボットの toio を利用する。約 30g と軽量であるため片手で容易に動かすことができ、ボタンや加速度センサを内蔵しているため入力手法としても活用できる。また、専用のプレイマットと組み合わせることで、絶対位置を取得できる。各 toio は、BLE (Bluetooth Low Energy) を介して、ホスト PC (OS: Windows 10 Home, CPU: Corei5-1135G7 2.40GHz, メモリ: 8GB) 上の Python プログラムから制御する。なお、toio の同時制御台数は明確には規定されておらず、ホスト PC の性能に影響される。著者らの環境で安定して制御可能な台数は 7 台であったため、本論文の実装では最大 7 台の toio を利用する。

4.2 アルゴリズムの実装

4.2.1 実装例 1: バブルソート

バブルソートは、配列内の隣り合う値を端から順に比較・交換する操作を繰り返すことで整列する方法である。本研究では、配列の長さ n は toio の台数となり、前述の実装上の理由から最大 7 とする。さらに、プレイマットに配置した状態で一番左側にある toio を 0 番目、一番右側にある toio を配列の $n - 1$ 番目の要素として扱い、ソートは昇順で行う。

3.1 節で述べたようにアルゴリズムの再生を開始すると、全ての toio を等間隔で一列に移動し、その時点で配列の要素が確定する。その後、バブルソートのアルゴリズムに沿って、要素の比較・交換を行う (図 3)。

4.2.2 実装例 2: ヒープソート

ヒープソートは、二分ヒープと呼ばれる木構造を用いたアルゴリズムである。本実装では昇順にソートするため、「親<子」の関係がすべてのノードで成立している木構造「最小ヒープ」(以下、ヒープ)を用いている。ヒープソートは「ヒープが成立している木構造の作成」→「根をソート済み配列に移動」→「木構造の再構成」を繰り返すことでソートを行う。

アルゴリズムの再生を開始すると、全ての toio がマット下部に一列に並び、木構造の配置になるように移動する。まず、ヒープを作成するために、ノードの比を行い、ノードの交換が必要な場合、toio 同士の位置を入れ替える (図 4a)。ノードの交換が行われれば、toio の位置を移動させる。要素の比較は「toio を左右に回す」動作で、比較の結果最小だった要素は「toio を一回転させる」ことで示す。ヒープ作成後、根ノードである toio をソート済み配列 (マット下部) に移動させる (図 4b)。その後、空いた根ノードに根から一番遠いノードを移動させることで、木構

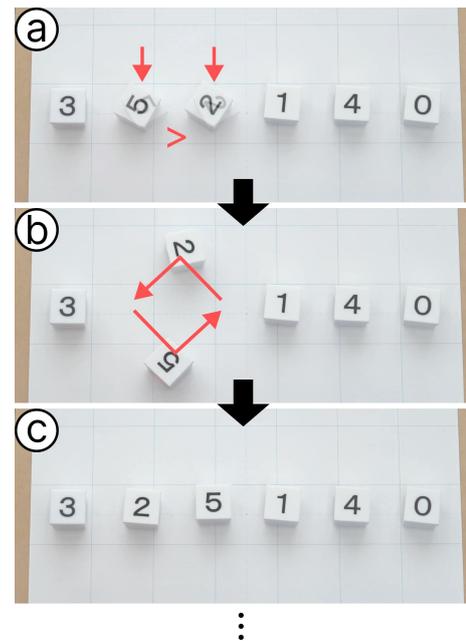


図 3 バブルソートの実行例。(a) 比較の動き。(b) 交換の動き。(c) 交換後の様子

造を再構成する (図 4c)。これらの流れを繰り返すことで、toio がマット下部に昇順で並んでいく様子が確認できる。

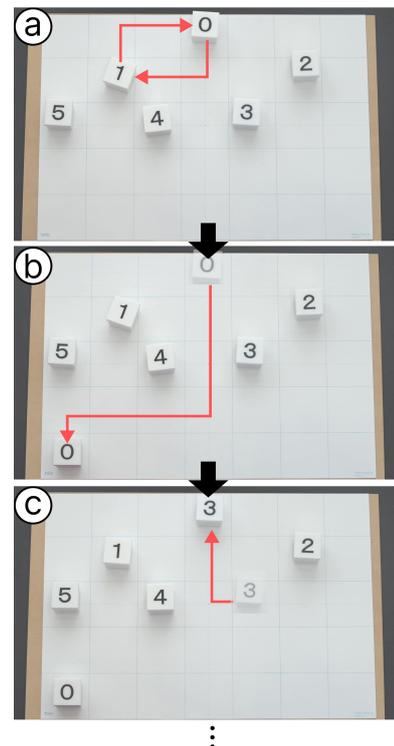


図 4 ヒープソートの実行例。(a) ヒープの作成。(b) 根の移動。(c) 木構造の再構成

4.2.3 実装例 3: スタック・キュー

スタック・キューはコンピュータで使われるデータ構造の一つである。スタックは後入先出 (LIFO) のデータ構造で、キューは先入先出 (FIFO) のデータ構造である。この

例では、マットを上部、中部、下部に分けて使用している。スタックまたはキューの可視化が開始すると、すべての toio が上部（待機スペース）に移動する。待機スペースにある toio をタップすると、タップされた要素が中部（データ構造スペース）に移動する（図 5 上）。データ構造スペースにある toio をタップすると、スタック/キューの構造に従って、下部（使用済みスペース）に移動する（図 5 下）。データ構造上、次に取り出せない toio をタップした場合は、toio が震えて動けない表現を行う。

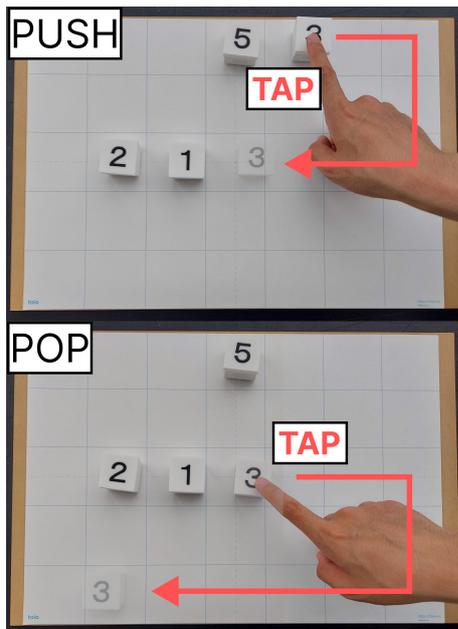


図 5 スタックの実行例

5. 予備実験

5.1 目的

ユーザ評価の目的は、本システム（提案手法）と既存のアルゴリズム可視化ツールである「VisuAlgo [14]」（比較手法）のどちらかをユーザに体験してもらい、TUI を用いた手法と GUI を使った手法の学習効果の比較を行う。

5.2 手法

ここでは実験の手順について示す。本実験では、ヒープソートの学習を主なタスクとして被験者に取り組みさせた。最初に、ヒープソートについての資料を用いて、簡単な説明を行った。次に、ユーザに現状のヒープソートの知識を問う、事前テストを課した。事前テスト後、本システムまたは VisuAlgo のどちらか一方の使用方法を説明し 15 分間自由に体験させた。次に、ヒープソートの学習成果を問うため、事後テストを課した。その後、システムやテストに関するアンケートに回答させた。

テストは、要素数 7 の配列 2 つに対し「(1) 配列から木

構造を作る」「(2) 木構造からヒープを作る」「(3) ヒープソートを進める」を問う問題を出題し、それぞれ解答させた（計 6 問）。

被験者は本システムと VisuAlgo それぞれ 21～24 歳の男性 5 人ずつの計 10 人の大学生であった。全員過去にソートアルゴリズムについて学習経験がある。

なお制約として、本実験で用いたソフトウェアは旧バージョンであり、可視化手法は同一だが、インタラクション手法が異なっている。具体的には、アルゴリズム制御用の toio を別に用意し、それをアイコンが描かれたマット上に配置することで再生や一時停止などの操作を行った（図 6）。

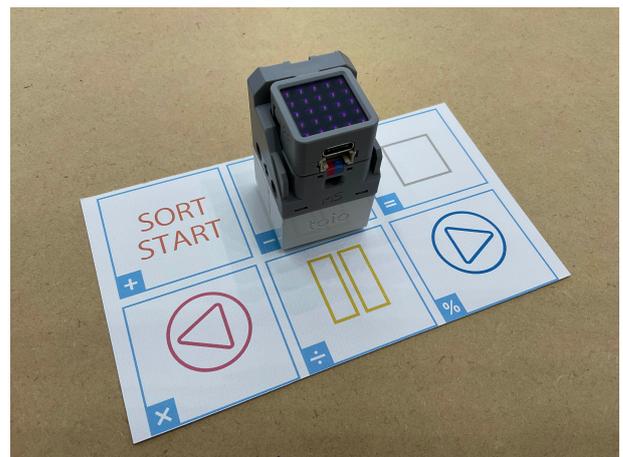


図 6 アルゴリズム制御用の toio とマット

5.3 結果

提案手法と比較手法の事前/事後テストの平均値を表 1 に示す。テストでは同じ小問を配列を変更して 2 回出題しているため、それぞれの小問の満点は 2 点である。問 1 は事前テストでも正解率が 9 割程度の簡単な問題であり、事後テストでは全員が正解していた。問 2 の事前テストでは 20～30%，問 3 では 10% の正解率であり、やや難易度の高い問題であった。一方、学習後では比較手法/提案手法共に大きく改善され、問 2 が 90～100%，問 3 が 80～90% との正答率であった。事前→事後テストの改善率を見ると、問 3 では提案手法の方がやや高い結果となったが、大きな差は見られなかった。

次に、比較手法（VisuAlgo）/提案手法（SorToio）の印象評価の結果の平均値と標準偏差を図 7 に示す。「楽しさ」については、比較手法/提案手法で平均 4.8 / 7.0 となり、提案手法の方が大幅に高い評価となった。「システムの使いやすさ」については、比較手法/提案手法で平均 6.4 / 5.6 となり、比較手法の方が高い評価となった。

5.4 考察

テスト結果（図 1）から、提案手法である SorToio は既

表 1 事前/事後テストの平均値 (2 点満点).

	VisuAlgo	SorToio
問 1	1.8/2.0	1.8/2.0
問 2	0.6/2.0	0.4/1.8
問 3	0.2/1.6	0.2/1.8

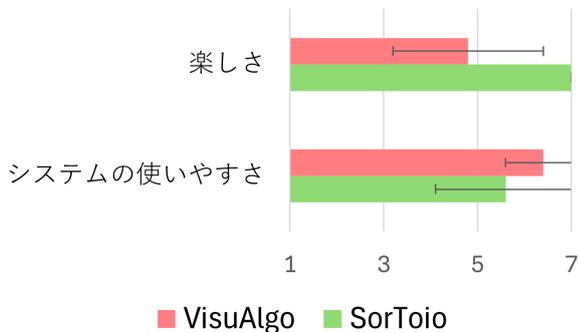


図 7 ユーザ印象評価結果

存の GUI アルゴリズム可視化ツールである比較手法と同程度の学習効果があることが分かった。

印象評価の「楽しさ」の項目では SorToio の方が大きく高かった。被験者の意見では、「toio が可愛い」「可愛らしいので、情報教育の壁が低くなりそう」などが挙げられていた。被験者は、toio に可愛らしさを感じており、それが動くことで楽しさを感じていると考える。一方、「システムの使いやすさ」の項目では、VisuAlgo の方が高かった。これは、toio をマットに置いて制御する操作が見慣れない方法である点やフィードバックが無い点から、評価が低かったと考える。この評価を踏まえて、現システムでは操作用 toio を廃止して、動作表現に使う toio に直接触れて制御することで、使いやすさの向上を目指した。

6. 議論

6.1 課題と改善案

ここでは「本システムの拡張性」、「計算量の表現」、「ユーザ評価」について議論する。「本システムの拡張性」について述べる。現在の実装では、7 台以下の toio しか利用できないため、実装可能なアルゴリズムに制限がある。例えば、多くの値を保持する必要があるアルゴリズム (例: 二分探索、幅優先探索) への適用は困難である。今後は、PC を複数台用意する等して、toio の接続台数を増やす、あるいはアルゴリズム表現の工夫を行うことで、この制約を解消できるかを検討する必要がある。

「計算量の表現」について述べる。現状の実装では、toio の物理的な移動距離に依存して実行時間が決まるため、アルゴリズムの計算量を正確に表現することが難しい。今後は計算量を重要視したモードを導入し、移動速度や移動距離を調整することで、アルゴリズムごとの計算量に応じた実行時間を提示できるように検討したい。

「ユーザ評価」について述べる。本研究で実施したユーザ評価は情報系大学生を対象としたものであり、被験者はいずれもソートアルゴリズムの学習経験を有していた。今後は、情報教育を受けていない大学 1 年生や高校生等を対象に、どのような学習効果や印象が得られるかを確認する必要がある。また、対象とする年齢によって、システムの適切な要件が変わってくる可能性がある。この点については次節で詳しく述べる。

6.2 子供向けの応用例: SorTale

提案システムでは、アルゴリズムをロボットの動きで可視化して、インタラクティブに介入することでその仕組みを理解することに焦点をおいて開発を進めてきた。一方、ロボットはあくまで数値を表現しており、特に小学校低学年等の層には学習のきっかけや動機を与えにくいと考えた。そこで、ストーリーテリングを用いて、人形遊びのような物語の中でアルゴリズムを「体験」できるシステム「SorTale [16]」の開発も併せて進めている (図 8a)。

ロボットの動きやユーザの操作で物語を構成し、その過程にソートアルゴリズムを組み込むことで、ユーザが自然とアルゴリズムに興味を持てるように配慮する。さらに、物語の登場人物を移動させたり、交換したりすることで、ユーザの試行錯誤を促すことを狙う。

6.2.1 追加機能

SorTale では、小型群ロボット上に、高さ等の外観 (パラメータ) が異なるオブジェクト (フィギュア) を装着し、ロボットの動きやユーザの操作で構成される物語を表現する。そのため、3D プリンタを用いて、フィギュア本体およびフィギュアを toio に固定するための台座パーツを作成した (図 9)。また、物語にインタラクティブに介入するために、フィギュアの種類を動的に認識可能な機能を実装を行った。直径 6.0mm、高さ 3.0mm のネオジム磁石を台座パーツの 4 箇所を組み込めるように設計した。toio の磁気センサは前方上部に搭載されており、磁気センサの左右 4.5mm、9.0mm 離れた位置 (計 4 か所) に固定できるようにした。また磁石の向き (S 極・N 極) も判定できるため、8 種類の ID が区別可能である。

6.2.2 物語例

本実装では、「木と太陽」を題材とした、「バブルソート」の学習を支援する物語を紹介する。ストーリーは以下のとおりである。

「ある森では、木たちが一列に並んでいました。その森に、太陽が昇り、光が差し始め、木たちは太陽に向かって歩いていきます。しかし、背の高い木の後ろでは、低い木はよく光が当たりません。そこで背の高い木は、光が見られるように、場所を交換してあげます。これを繰り返していくと、みんなが光を浴びることができました。」

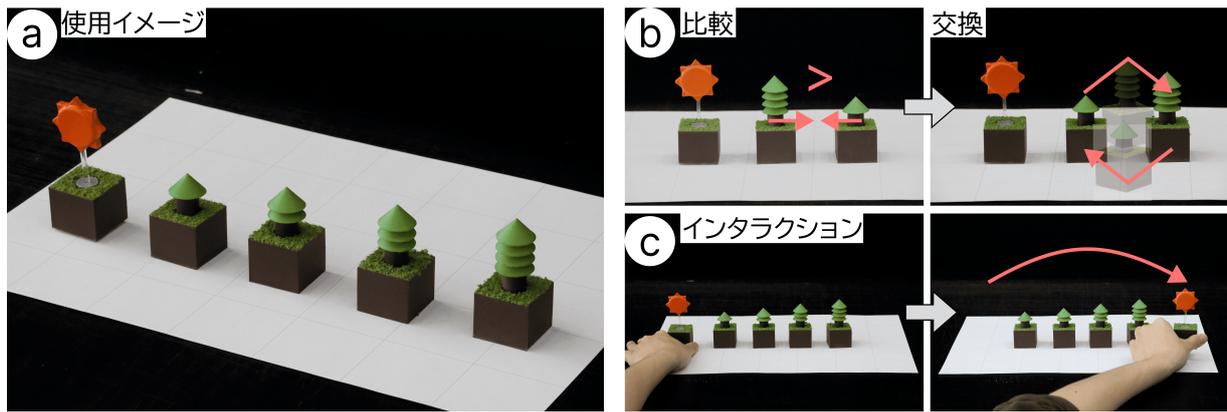


図 8 SorToio のコンセプト. (a) 利用イメージ. (b) 比較表現と交換表現. (c) インタラク
ションの例

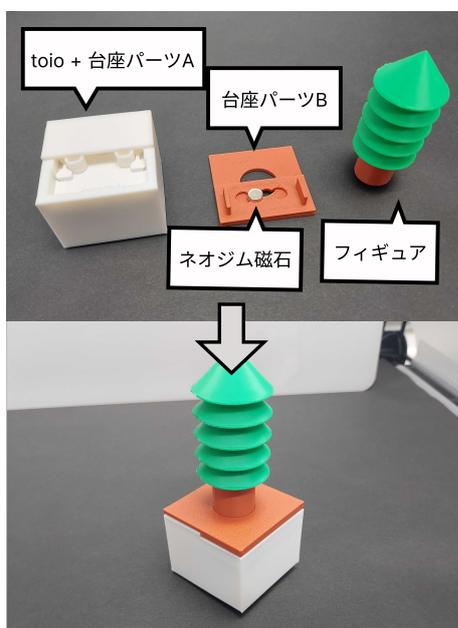


図 9 フィギュアを toio に固定する様子

6.2.3 体験の流れ

木と太陽は 3D プリンタで作成した、フィギュアで表現されている。木は高さがすべて異なっており、これをパラメータとしてソートを行う。ユーザの事前準備として、四つの木と一つの太陽を小型ロボットに載せ、プレイマット上に一列に並べる。

体験の流れは、まず、物語を読んで、内容を理解してもらう。複数の木と太陽の乗った小型ロボットをプレイマット上に配置する。太陽を木の列の右側または左側に設置することでソートが開始される。木たちは、背の低い木が太陽に近づくように互いに位置を譲り合いながら移動する (図 8b)。この処理順をバブルソートのアルゴリズムに基づくことで、要素の比較や入れ替えの流れを体験できる。さらに、ユーザは木のフィギュアを付け替えてパラメータを変更したり、太陽の位置を反対側の端に移動させて (図 8c)、再度アルゴリズムの動きを観察することができる。

7. まとめ

本研究では、小型の群ロボットである toio に値を持たせて物理的に動かすことで、アルゴリズムやデータ構造の学習を支援するシステム「SorToio」を提案した。バブルソート/ヒープソート等の実装例を開発し、それをを用いた評価実験を行い、学習効果や印象を調査した。今後は、情報教育を受けた経験のない小中学生等を対象に本システムを実際に利用してもらい、ユーザビリティや学習効果等の調査を行う予定である。得られた結果をもとにシステムを改良し、学校やワークショップなどで活用できるように、本システムの公開を目指したい。

参考文献

- [1] Mitchel Resnick, John Maloney, Andrés Monroy-Hernández, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, Eric Rosenbaum, Jay Silver, Brian Silverman, and Yasmin Kafai. Scratch: programming for all. *Commun. ACM*, Vol. 52, No. 11, p. 60–67, nov 2009.
- [2] Micro:bit Educational Foundation. micro:bit. <https://microbit.org/>. (参照 2026-02-07) .
- [3] Google. Project blocks: Making code physical for kids. <https://research.google/blog/project-bloks-making-code-physical-for-kids/>. (参照 2026-02-07) .
- [4] Sony Interactive Entertainment Inc. Gogo robot programming the secret of logibo .
- [5] 文部科学省. 情報編 高等学校学習指導要領 (平成 30 年告示) 解説. <https://www.mext.go.jp/content/000166115.pdf>, 2018. (参照 2026-02-07) .
- [6] 板垣智也, 塚田浩二. Sortoio: 小型群ロボットを用いたアルゴリズム学習支援システムの提案. 日本ソフトウェア科学会 WISS 2024 予稿集, 2024.
- [7] Tomoya Itagaki and Koji Tsukada. Sortoio: Learning support system for algorithm using a small swarm robot. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, UIST '25, New York, NY, USA, 2025. Association for Computing Machinery.
- [8] Mathieu Le Goc, Lawrence H. Kim, Ali Parsaei, Jean-

- Daniel Fekete, Pierre Dragicevic, and Sean Follmer. Zooids: Building blocks for swarm user interfaces. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16, p. 97–109, New York, NY, USA, 2016. Association for Computing Machinery.
- [9] Hiroki Kaimoto, Kyzyl Monteiro, Mehrad Faridan, Jiatong Li, Samin Farajian, Yasuaki Kakehi, Ken Nakagaki, and Ryo Suzuki. Sketched reality: Sketching bi-directional interactions between virtual and physical worlds with ar and actuated tangible ui. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, UIST '22. ACM, October 2022.
- [10] Yue Zhu, Zhiyuan Zhou, Jinlin Miao, Haipeng Mi, and Yijie Guo. Tangiblenegotiation: Probing design opportunities for integration of generative ai and swarm robotics for imagination cultivation in child art education. In *Companion of the 2024 on ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '24, p. 66–70, New York, NY, USA, 2024. Association for Computing Machinery.
- [11] 八城朋仁, 迎山和司, 原田泰. Plugramming : 協調作業のためのタンジブルなプログラミングツール. 日本デザイン学会研究発表大会概要集, Vol. 64, p. 76, 2017.
- [12] Sony Interactive EntertainmentInc. トイオ・プレイグラウンド — タイトル - toio. (参照 2026-02-07) .
- [13] Tim Bell, Jason Alexander, Isaac Freeman, and Mick Grimley. Computer science unplugged: school students doing real computing without computers. In *New Zealand Journal of Applied Computing and Information Technology*, Vol. 13, pp. 20–29, 2009.
- [14] Steven HALIM. Visualgo – visualising data structures and algorithms through animation. In *Olympiads in Informatics*, Vol. 9, p. 243–245, 2015.
- [15] Xiaozhou Deng, Danli Wang, Qiao Jin, and Fang Sun. Arcat: A tangible programming tool for dfs algorithm teaching. In *Proceedings of the 18th ACM International Conference on Interaction Design and Children*, IDC '19, p. 533–537, New York, NY, USA, 2019. Association for Computing Machinery.
- [16] 板垣智也, 塚田浩二. Sortale: 小型群ロボットによる物語を通したアルゴリズム学習支援手法の提案. インタラクション 2026 論文集, 03 2026.