

列車模型を用いた幼児向け計算機学習システムの構築

Hands-on Learning of Computer Programming for Children Using a Model Railway

佐々本博和^{1),2)}, 野間春生¹⁾, 伊藤雄一²⁾

北村喜文²⁾, 岸野文郎²⁾, 鉄谷信二¹⁾

Hirokazu SASAMOTO, Haruo NOMA, Yuichi ITOH

Yoshifumi KITAMURA, Fumio KISHINO and Nobuji TETSUTANI

1) 株式会社 国際電気通信基礎技術研究所 メディア情報科学研究所
(〒619-0288 京都府けいはんな学研都市光台 2-2-2, {sasamoto, noma, tetsutani}@atr.co.jp)

2) 大阪大学大学院情報科学研究科
(〒565-0871 大阪府吹田市山田丘 2-1, {sasamoto, itoh, kitamura, kishino}@ist.osaka-u.ac.jp)

Abstract: This research is aimed at developing a new methodology that supports technologies for pre-school-age children to learn about computation and programming through hands-on play experience with toys. It allows children to acquire the concept and knowledge of computation and programming by playing with a model railway, and to understand programming and strategic thinking using a railway layout. In this research, we proposed such a hands-on learning environment of computer and programming study for children using a model railway and built the 1st trial system.

Key Words: computer education, haptic, model railway

1. はじめに

幼児は、絵本から絵と文字の関連を理解し、ひいては高度な読み書きの技術を会得すると考えられる。同様にコンピュータの初期習得過程においても、幼児にとっては理解しにくい計算機自体の使い方やプログラミング方法を学習することなしに、それらが持つ意味そのものを具体化させた玩具で遊ぶことによって、高度な計算機技術の習得が将来可能となると考えられる(図1)。

我々が対象としている計算機上で動くプログラムは、逐次処理と条件分岐、演算から構成されている。一方で、列車の運行に関してもレールの上を貨車や客車を牽引した列車が逐次運行する。そこで我々は、この類似性に着目し、コンピュータの動作を列車模型に対応付けることにより、幼児を対象とした計算機リテラシのための絵本として、列車レイアウト玩具を用いた学習環境を提案する。これによって、学習者である幼児は遊びを通してコンピュータやプログラミングの概念、ひいては戦略的な思考方法を経験的な知識として得ることが期待できる。

本稿では、前述の計算機学習環境を実現するための技術課題を検討し、それらを踏まえて、列車模型を用いた幼児向け計算機学習システムの試作を行い、動作を確認する。

2. 関連研究

計算機を導入した教育としては、国内外を問わず古くか

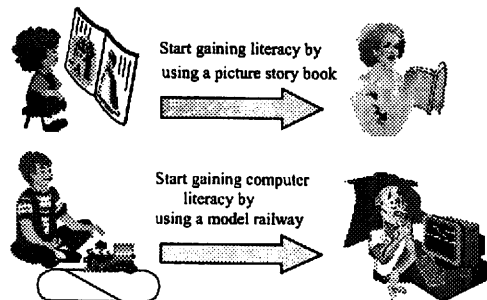


図1 読み書きとプログラミングのリテラシの獲得

らCAI学習、最近ではE-Learningとして行われているが、一般的な計算機によって、映像と音中心の情報提示手段によるプログラムとの対話を通じた知識獲得型学習を前提としている。しかし、特にプログラミングのような創造性や見通しを持った戦略が重要となる分野では、知識獲得型学習よりも、対象概念の構造や機能を通じて実際に理解する学習手段が重要になる。

このような学習環境を構築することを目的とした関連研究として、McCormickはリアルタイムシステム教育に関する大学院生向けのコースに、列車レイアウトを使った手法を提案している[1]。学生はレイアウトの製作と、複数の列車を同時に制御するソフトウェアを、計算機上に実装する過程を通してリアルタイム制御を学ぶ。また、DowsingとTostenらは、固定されたレイアウト上の列車の位置によって、計算機OSの処理状態を説明する手法を個別に提案

している[2,3]。これらの先行研究でも本提案と同様に計算機と列車運行の間の類似性に注目しているが、いずれも固定レイアウトを計算機の動作制約の表現、あるいは計算機そのものの状態表示装置として使う段階に留まっている。

列車模型に限らず、システムを視覚化したり、ブロックとして実体化させるものとして、MITのAIラボで開発された LogoLego や、その実用成果である LEGO 社の MindStorms がある。子供は自分の手でセンサやモータが組み込まれたブロックを使って任意の機械構造を創作できる。また、鈴木らは AlgoBlock[4]によって、ビジュアルプログラミング環境を、実世界の箱と組み合わせることで実現する手法を提案している。彼らは子供がグループで学習を行うことが高い効果を得られることを確認している。これまでは、プログラム環境と実行環境が別体になっており、学習者が両者の対応付けを意図的に行う必要がある。幼児にとっての教育ツールとして考えた場合、このような意図的な対応付けのない直感的な手段のほうが好ましい。

3. 列車模型を用いた幼児向け計算機学習システム

現代の計算機上で動くプログラムは、基本的にノイマンスタイルと呼ばれる逐次処理と条件分岐、演算から構成されている。一方で、列車の運行はあらかじめ敷設されたレールの上を貨車や客車を牽引した列車が逐次運行する。我々はこの類似性に注目し、表1のようにプログラムの基本要素を列車レイアウトに対応させて、コンピュータの概念を列車のレイアウト遊びによって学べる環境を提案する。

ここでは、列車レイアウトを構築することがプログラムを記述することに相当し、その上で列車を走らせることが、プログラムを実行させることに相当する。また、列車の位置はプログラムの実行位置を示し、列車の牽引する貨車、あるいはレイアウト脇の荷物置場が変数であり、それらに搭載される荷物の数や種類が変数の値に対応する。

この対応付けを行った列車レイアウトの一例を図2に示す。出発地点のレールへ列車を置き、ボタンを押して発車させると、列車は初めの Load ユニットで牽引する貨車に貨物一つ積む。次に列車が Point ユニットまで進むと、システムは、貨車に貨物が詰まっているかをチェックし、貨物の有無に従ってポイントを切り替える。列車内に貨物がある場合、列車は貨物を降ろす。貨物がない場合は、自動的にポイントが切り替わり、列車はゴールへ到達する。これは単純なループアルゴリズムのレイアウト例であるが、図3に示すようなボールを積み下ろしするユニットと、ボールの色や有無によってポイントの切り替えるといった単純な機能を持つユニットから構成されている。

ここで提案する学習システムでは、子供にとっては単に課題に合わせてレールを自由に組み替えたりボールを列車に積んだりなど、純粋な遊びであり、子供が好んで積極

表1 コンピュータから列車への写像

コンピュータ	列車レイアウト
プログラム	レールのレイアウト
条件分岐	レールの分岐
プログラムカウンタ	列車の位置
変数	貨車
変数の値	貨車に詰まれたボール
変数のロード・演算	ボールの出し入れ

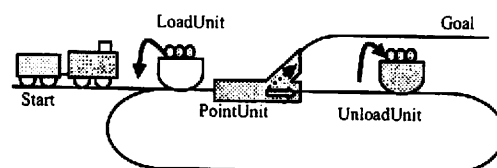


図2 ループのレイアウト例

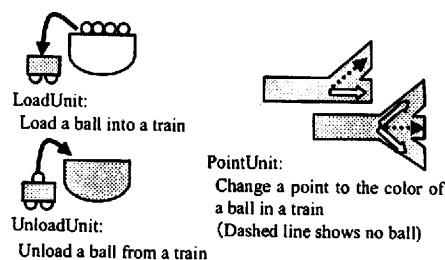


図3 列車レイアウトユニットの例

的に取り組むことができることが、先行研究と大きく異なる。また、計算機を使っていることを意識させないので、つまり学習過程が全てレイアウト内で実現しているので、プログラム環境と実行環境の対応付けを考慮する必要がない。さらに、子供たちはこれらのレイアウトを走らせるときに列車が意に反する動作をした時には、手でレイアウト内の列車を止めたり、貨車のボールを移動させて動作を変えることができる。これは、すなわちプログラム実行中のブレークポイントの設定や、変数の操作といったデバッグに相当するが、それらを手を使って直感的に操作できる。また、列車をもう一台増やし、並列して運行させることによって、マルチスレッドの概念も学ぶといった発展が期待できる。

さらに、グループ学習の教材としても効果が期待できる。このレイアウトシステムは個人の概念としてのアイデアを具象化できるが、グループで一つのレイアウトを作る際には、それぞれのアイデアをレイアウトとして示し、容易に互いのアイデアであるレイアウトを融合させることができる。

4. 技術課題

前述した幼児に対する列車レイアウトを用いた学習環境を構築するために、以下のような技術課題が挙げられる。

- コンピュータの基本要素の列車ユニットへの投影
- 自由で柔軟なレイアウトの組み換え
- レイアウトの自動制御

まず、学習の目的である計算機の仕組みを可能な限り列車レイアウトで実現するために、計算機が有する演算や関数を、どの程度までレイアウトのユニットに投影すべきかを検討せねばならない。図3では、積み下ろしやポイントの切り替えといった、最小限の機能だけでレイアウトを構築しているが、さらにシフト演算や論理演算など、CPUが持つ機能を列車ユニットに対応させユニットの種類を増やせば、アプリケーションとしてOSの仕組みやコンピュータシミュレーションなども原理的には構築できる。

次に、幼児がこのシステムを使って、自由にアイデアを具現化するためには、レイアウト構築の際に、簡単かつ確実に基本ユニットを柔軟に組み替えできなければならない。そのためには、レールやユニット部品の機械的・電気的な接続が容易でありながら確実にでなければならない。また、ユニットの動作が直感的なものでなければならない。

さらに、レイアウトで列車を動かす際には、レイアウトを構成するユニットと列車が確実に協調して動作せねばならない。そのため、表面的には単純に動いているように見えながら、その背後では貨車の積載状況や列車の運行状況をモニタし、レイアウトを自動制御せねばならない。

5. レイアウトの構築

前述のように本システムの特徴は、基本ユニットを用いて多様なレイアウトを構築し、教材として与えられる課題を実現することにある。そこで、3章で例示したループ以外にも積み下ろし、ポイント切り替えの最小限のユニットを用いて、次に示すような様々なレイアウトが構成できる。

● ソーティング

Loadユニット(A)にある2種類の色のボールをそれぞれのUnloadユニットに振り分けるという動作を行うレイアウトである(図4)。これは単純なソーティング課題を構成していることになる。

● コピー

loadユニット(C)(E)部分にあるボールをUnloadユニット(B)(D)とUnloadユニット(F)(G)に複製させる(図5)。この動作はプログラムでいうコピーの概念と一致する。

● 足し算

Loadユニット(A)のボールの数とLoadユニット(C)のボールの数を足し合わせた分をUnloadユニット(B)にとりだすレイアウトである。よってこれは足し算の概念を学ぶことができる(図6)。

● 引き算

Loadユニット(A)のボールの数だけ列車を周回させLoadユニット(D)からUnloadユニット(C)にボールを移すことで、Loadユニット(D)からLoadユニット(A)のボールの数分だけ減らす、引き算となる(図7)。

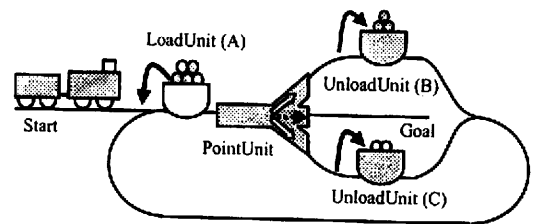


図4 ソーティングのレイアウト例

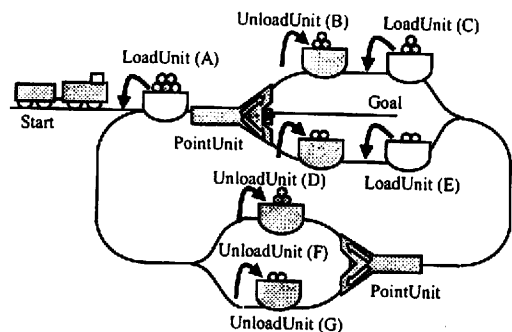


図5 コピーのレイアウト例

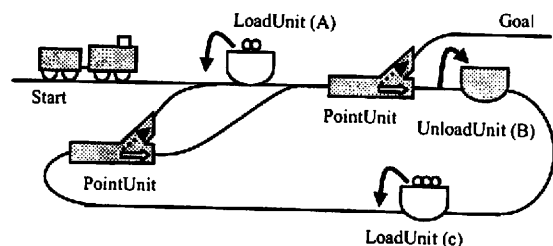


図6 足し算のレイアウト例

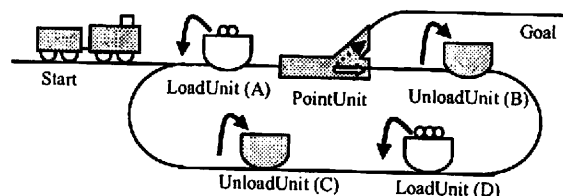


図7 引き算のレイアウト例

また、足し算と引き算のレイアウトを組み合わせると、掛け算、割り算などの計算レイアウトやプログラミングの命令なども実現できる。したがって、前述のように計算機が有する、他の演算処理を実現するユニットを追加することで、より複雑なレイアウトへ発展できることがわかる。

6. 実装

6.1 システムの構成

4章の技術課題で述べたように、自由かつ柔軟なレイアウトの構築とレイアウト上での列車運行イベントを実現するためには、列車とレイアウト部品となるレールユニットとの間で通信を行い、それぞれのレールユニット上で起こる振る舞いを制御しなければならない。本稿では、提案手法の可能性を探ることを目的として、試作環境を実装した。試作では最小限のユニットで自由に組み替え、様々なレイアウトを作成できるシステムを構築することを優先

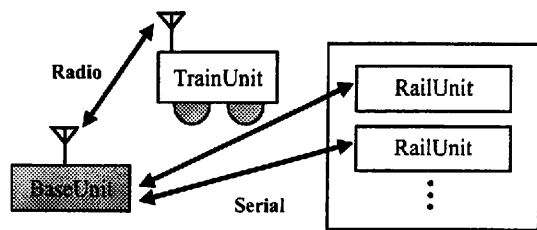


図 8 システム構成図

した設計を行った。それぞれのユニットと列車にはトミー社製のプラレールを改造し、Microchip社製のPICとセンサやアクチュエータを埋め込んだ。プラレールは幼児期の子供向けの鉄道模型玩具であり、レールを自由に柔軟に組み合わせることが可能である。システムは図8のように、全体のシステムを統括するBaseユニットへ接続し、シリアル通信を行った。また、Baseユニットと列車間の通信はR.F.Solutions社の無線シリアル通信モジュールを用いた。

扱えるボールの色は二種類、列車に乗せることができるボールは一つという制限を持たせた。ボールの積み下ろしや、ポイントの切り替えに用いるアクチュエータとしてソレノイドを、それぞれのユニットへの列車到着を確認するためのセンサとしてフォトインタラプタを用いた。実装したレールユニットを図9に示す。

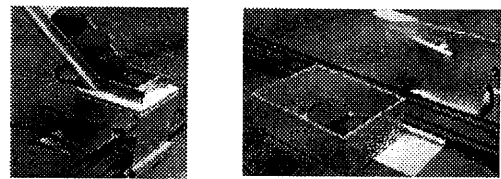
6.2 実装結果と課題

実装したレイアウト部品を用いて、ボールのソーティングとボールのコピーのレイアウトを構築した。図10に今回実装を行ったレイアウト部品を用いたソーティングのレイアウトを示す。さらに、これらの基本ユニット部品の組み替えを行うことで、5章で例示した各レイアウトに再構成できることを確認した。

一方で、今回の実装方法では、レールと列車間をベースユニットを介して通信する方式を採用したため、レイアウト内に多数の配線が必要となり、レイアウトの変更作業には困難を伴う。この問題を解決するために、我々が開発したActiveCube[5]のブロック間の通信方式を用いることで、ユニット間の通信を行う予定である。ActiveCubeではブロックを組み合わせるだけでブロック全体が通信を行い、その全体形状をリアルタイムに認識できる。これをレールに組み込めばレールレイアウトの形状を外部配線なしにリアルタイムで認識でき、より自由かつ柔軟にレイアウトを組み替えることができる。また、現在の実装では強度が不十分であるので、子供に見える部分ができるだけ複雑にならないようにするといった、実際に子供に使ってもらうことを考慮した設計を進める。

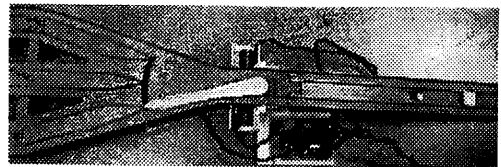
7. おわりに

列車レイアウトシステムを用いた教育環境が幼児へのコンピュータ教育に適していることを説明し、その環境を実現するための技術課題を述べ、試作環境を実装した。そして、実装環境でいくつかのレイアウトを構築し、動作を確認した。今後は個々のレイアウト部品やシステム構成の



(a) LoadUnit

(b) UnloadUnit



(c) PointUnit

図 9 列車レイアウトユニット

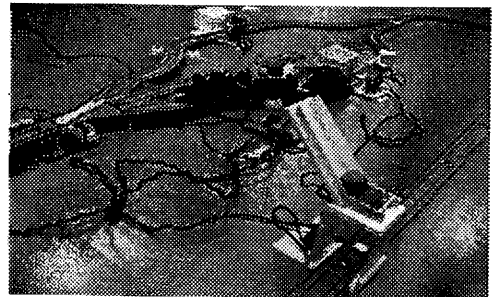


図 10 実装図 (簡易ソーティングレイアウト)

改良を行い、より自由かつ柔軟なレイアウト組み替えが行うことのできるシステムを構築し、幼児への評価・検証を行っていく予定である。

謝辞

本研究は通信・放送機構の研究委託“超高速ネットワーク社会に向けた新しいインタラクション・メディアの研究開発”により実施したものである。

参考文献

- [1] J. McCormick, "Using a Model Railroad to Teach Digital Process Control", 19th SIGCSE Technical Symposium on Computer Science Education, Vol.20, No.1, pp.304-308, 1988.
- [2] R. Dowsing, P. Seward, "Computer-controlled model railway for use in teaching real-time programming", Microprocess & Microsystem, Vol.6, No.10, pp.529-533, 1982.
- [3] R. Tosten, "Using a Model Railroad System in an Artificial Intelligence and Operating System Course", ACM SIGCSE Bulletin, Vol.25, No.1, pp.30-32, 1993.
- [4] 鈴木, 加藤, "協調学習のための教育ツール AlgoBlock", 認知科学, Vol.2, No.1, pp.27-36, 1995.
- [5] 伊藤, 河合, 北村, 岸野, "リアルタイム3次元形状モデリングとインタラクションのための双方向ユーザインタフェース ActiveCube", 情報処理学会論文誌, Vol.42, No.6, pp.1338-1347, 2001.