

数式の記述方法を選択できるプログラミングインタフェース

島田 雄輝* 薄羽 大樹* 宮下 芳明*

概要. プログラミングにおいて、数式の記述を必要とする機会は様々にある。しかし、プログラミングの記法と数学の記法は異なる点があり、双方に利点が認められる。本研究では、テキストプログラミング環境に手書き機能を搭載することで、コーディング中にプログラミングの記法と数学の記法の双方で数式を記述できるインタフェースを提案する。本稿では、提案システムと関連研究の紹介を行なった後に、今後の展望について述べる。

1 はじめに

プログラミング中、数式の記述を必要とする機会は様々である。例えば、3DCG や機械学習、統計処理などを実装する際には必須な要素であると言っても過言ではない。図 1 は実際に JavaScript を用いて、パーリンノイズに加算合成を施している例である。この例では、周波数と振幅をそれぞれ変更したパーリンノイズを加算した結果を出力している。座標値からパーリンノイズの値を得る関数を $p()$ 、累乗を $\text{Math.pow}()$ 、また、シグマに値する計算を for 文で記述している。これらのようなプログラミング中に記述する数式の記法は、以下に示すように数学の記法とは異なる点が多い。

コーディングは基本的にキーボードを用いて一行ずつ行う必要があり、上付きや下付きは記述することが困難である。またキーボードでサポートされていない特殊な記号を記述することもできない。そのため、言語に対応した API (Application Programming Interface) を利用した記述が主である。これは基本的に括弧と引数を用いる記法なために入れ子構造となり、見易さや理解のしやすさなどの点で数学の記法に比べ劣る場合が多いと推察される。またこのような構造のため、例えば累乗の指数を取り除きたい場合には呼び出している関数名と左括弧、引数と右括弧を適切に指定して削除しなければならない。しかし、一行ずつのみの記述とすることで余計なキャレットの移動を省き、全体として速く記述が行える利点がある。また、数学の記法では特殊な例を除き、変数や関数は一文字で表され、連続している場合には掛け算を行っていると見なされる。そのため掛け算の記号は省略できる場合が多い。プログラミングにおいては連続している文字による変数の定義も可能なため、変数名を単語にすることにより動作をわかりやすくすることが可能である。しかしそのため、掛け算の際には明示的に $*$ を挿入しなければならない。

```
const o = (x, y) => {
  let value = 0;
  let maxValue = 0;
  for (let i = 0; i < 10; i++) {
    value += Math.pow(0.5, i) * p(x * Math.pow(2, i), y * Math.pow(2, i));
    maxValue += Math.pow(0.5, i);
  }
  return value / maxValue;
}
```

図 1. 数式のプログラミング記法

図 2. 手書きによる数式の挿入

本研究ではこれらの背景から、数式のプログラミングの記法と数学の記法の双方で記述できるインタフェースを提案する。提案システムは数式を手書きで記述することができ、コードへの変換を行う機能を搭載しているため、ユーザは状況によってプログラミングの記法と数学の記法で選択することが可能となる。

2 提案システム

提案システムは、プログラミング中に手書きで数式を記述できるインタフェースである。また、通常のキーボードを用いた記述も可能なため、状況によって記述方法を選択することが可能である。例えば、一つの数式程度の単純な記述ではキーボードによる入力を行い、複数の数式が組み合わさった複雑な記述は手書を利用するといった選択が可能である。

本提案システムでは、図 2 左のように手書きが行えるパレットが存在し、手書きで数式を記述することができる。また、手書きした数式は図 2 右のようにテキスト中に挿入することができ、その際に数式の整形を行い、見やすくすることも可能である。図 2 右は、図 1 を提案システムによって数学の記法で記述した例である。for 文による繰り返し処理と加

算はシグマに置き換えられ、掛け算の省略や、累乗や分数を一行の制約のもとで記述しないことで、よりなじみのある数学の記法で記述されている。

3 関連研究

3.1 タイピングと手書きの比較

タイピングと手書きを比較している研究がなされている。Berninger らは、アルファベットの書き取り、文章、エッセイを書く能力に関して、タイピングと手書きで比較実験を行なっている [1]。結果としては、アルファベットの書き取りに関してはタイピングの方が速かったが、エッセイでは手書きの方が速くなり、文章に関しては大した速さの差は出なかったが、手書きの方が出来栄のいい文章が書けていたといった報告が出ている。また Mueller らは、講義でメモをとる際にタイピングと手書きで比較を行なっている [5]。結果として、タイピングでメモをとった学生は教材のほとんど全ての記録が生成できたが、授業の内容に関する質問では手書きでメモを取った学生がよりよく理解できていたという報告が出ている。

3.2 手書きを用いたプログラミングインタフェース

手書き機能を搭載したプログラミングインタフェースはこれまでも提案されている。久米らは、現状のキーボードを用いた記述の問題点から、ペンのみで記述を行うプログラミングインタフェースを提案している [6]。また Frye らは、キーボードの代替としてペンを適用した環境を提案し、既存コードの変数解析や、文脈依存コードの精製などの機能を実装している [2]。

3.3 画像をコード中に挿入できる環境

コード中に画像を挿入することで視覚的に表現している研究も様々にある。Andrew らは、作成したいユーザインタフェースに対応するビジュアル表現を、コード中に記述することでプログラミングを可能とする Barista を提案している [4]。また、Kato らはソースコード中に画像を埋め込むことによってロボットの姿勢情報などを視覚的に表現できる Picode を提案している [3]。

4 まとめと展望

本研究ではプログラミング中に記述できる数式を、プログラミングの記法と数学の記法の双方で記述できるインタフェースを提案した。今後の展望としては、記述した後の変換、すなわちプログラミングの記法から数学の記法へ、数学の記法からプログラミングの記法への相互変換を目指す。この際に、コード中に記述されている数式は他の処理が含まれている場合がある、例えば図1では for 文と加算により

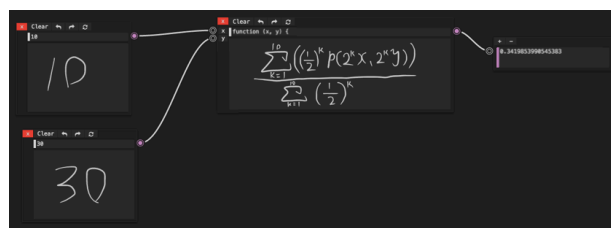


図 3. 提案手法を適用したビジュアルプログラミング環境

シグマを記述しているが、for 文の繰り返し処理の中でさらに別の処理が記述されている際は、単純なシグマとして表すことは適切ではない。そのため、プログラミングの記法から数学の記法への変換は困難な場合があり、課題となることが予想される。

また、タブレット端末上で動作するプログラミング環境やスタイラスペンを用いたアプリケーションが登場してきている。そのため、現在タブレット端末とスタイラスペンを用いたビジュアルプログラミング環境を開発している図3。このような環境はペン入力のみでプログラミングを完結することが可能である。そのため、手書きを利用した本研究の提案手法とは相性がいいと考えられ、導入の検討を行っている。

参考文献

- [1] V. W. Berninger, R. D. Abbott, A. Augsburger, and N. Garcia. Comparison of pen and keyboard transcription modes in children with and without learning disabilities. *Learning Disability Quarterly*, 32(3):123–141, 2009.
- [2] J. Frye and B. Franke. PDP: pen driven programming. In *Proceedings of the 22nd British HCI Group Annual Conference on People and Computers: Culture, Creativity, Interaction-Volume 2*, pp. 127–130, 2008.
- [3] J. Kato, D. Sakamoto, and T. Igarashi. Picode: Inline Photos Representing Posture Data in Source Code. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 3097–3100, 2013.
- [4] A. J. Ko and B. A. Myers. Barista: An implementation framework for enabling new tools, interaction techniques and views in code editors. In *CHI*, 2006.
- [5] P. A. Mueller and D. M. Oppenheimer. The pen is mightier than the keyboard: Advantages of longhand over laptop note taking. *Psychological science*, 25(6):1159–1168, 2014.
- [6] 久米諒, 田野俊一, 橋山智訓, 市野順子. ペン型のデバイスを用いたプログラミング環境の提案. 日本知能情報ファジィ学会 ファジィシステムシンポジウム 講演論文集 第 28 回ファジィシステムシンポジウム, pp. 1091–1096. 日本知能情報ファジィ学会, 2012.