

数式編集のための GUI

A GUI System for Formula Editing

出口 博章*

Summary. GUI環境は、ポインティングデバイスによって画面上の二次元座標を利用者が直接指示可能であるため、数式の二次元構造に対する操作である数式編集に適した環境である。しかし、多くの数式エディタにおいて採用されているテンプレート適用による数式編集は、キーボードによるショートカット操作で代替可能であり、画面上の二次元座標に基づいた操作ではないため GUI を有効に利用しているとは言い難い。本論文ではポインティングデバイス操作による二次元座標の指定によって記号の二次元的な位置関係の指定を可能とする数式編集のための GUI について述べる。

1 はじめに

GUIにおいて数式編集を行なう際に、記号の二次元の位置関係を構成する方法として、多くのソフトウェアで採用されているのはテンプレートの適用による方法である。テンプレートはパレット上に並べられた GUI ボタンや GUI メニューによって提供されていることが多く、それらがマウス等のポインティングデバイス（以下、マウス等）によって選択される。しかし、その操作はテンプレートの集合という選択肢からの選択であるため、キーボードによるショートカット操作の割り当てによって代替が可能であり、マウス等から得られる二次元情報を必須とするものではない。

本研究では、マウス等から得られる二次元情報を必須とするような操作であるドラッグ&ドロップを利用して記号を組み合わせていくことによって数式を編集するためのシステムのプロトタイプとして MathBlackBoard[1] を作成した。マウス等から得られる情報を利用することによって記号の二次元上の位置関係の指示を直接行なうことが可能となり、記号の配置に用いるテンプレートが不要となるため、利用者に提示する選択肢を減少させることが可能となった。

本システムは、特殊なドラッグ&ドロップ操作によって記号を所望の位置に「くっつける」ように吸着させることによって二次元構造を直感的に扱えるように設計されている。くっついた複数の記号をひとかたまりとして扱うことも可能であるため、部品を寄せ集めるようにして数式を構成可能である。なお、本論文において「記号」という言葉は + や · などの一般的な意味での記号に加えてアルファベットなどの文字も含むような「文字・記号」の意味で用いるものとする。

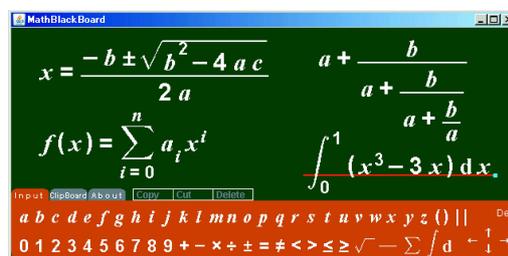


図 1. MathBlackBoard バージョン 2

2 MathBlackBoard

MathBlackBoard[1] は、使いやすい数式処理システムを目指した黑板アプレット [2] から出発して、数式処理システムのフロントエンドとして Java を用いて開発されてきた。本論文では、特殊なドラッグ&ドロップ操作である dynaput 操作 [3] とその操作対象となるオブジェクトを備えたバージョン 2 (図 1) について述べる。

2.1 システム構成

本システムは、入力機器としてポインティングデバイス、出力機器として二次元のビットマップ表示が可能である一般的なディスプレイを構成要件としている。ペンコンピューティングでの利用を目標としているが、マウス操作のみによっても利用可能である。キーボード操作による代替も可能となるように設計はされているが、操作性を最大限発揮するためにはポインティングデバイスが必須となる。

2.2 dynaput 操作の概要

図 2 は、既に入力されている「 n 」と「 $i = 0$ 」と「 $\sum a_i x^i$ 」そして「 $f(x) =$ 」を dynaput 操作によって「くっつく」ように吸着させて「 $f(x) = \sum_{i=0}^n a_i x^i$ 」を構成する流れを示している。

Copyright is held by the author(s).

* DEGUCHI Hiroaki, 神戸大学 発達科学部

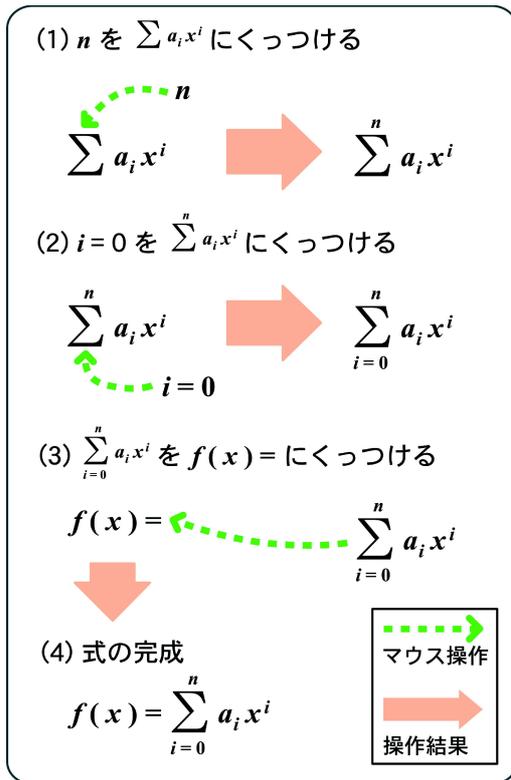


図 2. dynaput 操作によって記号が吸着される様子

図 2 中の (1) は「 n 」がドラッグされ「 \sum 」の上側にドロップされる様子である。また (2) は「 $i=0$ 」がドラッグされ「 \sum 」の下側にドロップされる様子である。以上のステップで出来上がった「 $\sum_{i=0}^n a_i x^i$ 」がドラッグされ「 $=$ 」の右側にドロップされると (4) のように「 $f(x) = \sum_{i=0}^n a_i x^i$ 」が完成する。

以上の操作でドロップ先を上側・下側・右側と指示するためには、操作対象となる記号オブジェクトがそれらの指示を受け取るように設計されている必要がある。

2.3 dynaput 操作の対象

dynaput 操作の対象となる記号オブジェクトはその記号オブジェクトを選択するための選択領域 (図 3, M) の他に二次元の位置関係を伴う接続関係を指定するための領域を複数備えているが、それらは選択領域の周辺に配置されるため周辺領域 (図 3, R0~R7) と呼ばれている。周辺領域は必要となる場合以外は画面上に表示されないように設定可能であるが、その場合はドラッグ操作中のポイント座標が含まれる周辺領域を表示することによって利用者にフィードバックを返す。

dynaput 操作と従来のドラッグ&ドロップ操作と

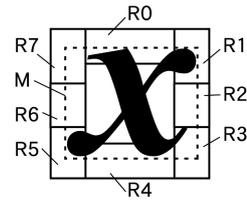


図 3. 選択領域 (M) と周辺領域 (R0~R7)

の違いは周辺領域の利用である。従来の操作においてはドラッグ操作開始前から明示的に表示されている領域やオブジェクトを対象としてドロップ操作を行なうことに対し、dynaput 操作においてはドラッグ開始前に表示されているとは限らない付随的な領域を対象としてドロップ操作を行なっている。このことにより、従来では「オブジェクトか否か」という判定のみであったところに「オブジェクトから見てどういう位置関係か」という判定が追加され、利用者によるきめ細かい指定が可能となった。

3 まとめと今後の展望

記号をくっつけていくことによって数式を構成していく dynaput 操作とその対象オブジェクトを備えたプロトタイプを作成した。dynaput 操作は特殊なドラッグ&ドロップ操作であり、その操作対象となるオブジェクトは周辺領域を備えている。結果として従来のドラッグ&ドロップ操作よりもきめ細かな指示が可能となり、マウス等によって数式の二次元構造を直接指示することが可能となった。

今後は数式以外の情報への応用が期待されるが、化学式のように二次元構造を持つ情報への応用の他に、テキスト情報のように従来であれば一次元構造によって表現されていた情報への応用も可能ではないかと考えられる。

謝辞

本研究は情報処理推進機構 2007 年度末踏ソフトウェア創造事業の支援を受けています。開発の支援、指導をいただいている情報処理推進機構並びに河野プロジェクトマネージャーに感謝いたします。

参考文献

- [1] MathBlackBoard Home Page.
<http://wwwmain.h.kobe-u.ac.jp/MathBB/>.
- [2] 松嶋 純也. Java を用いた使いやすい数式処理システム. 修士論文, 神戸大学大学院教育学研究科, 1998.
- [3] H. Deguchi. Dynaput: Dynamic Input Manipulations for 2D Structures of Mathematical Expressions. In *Computational Science - ICCS 2007, 7th International Conference, Proceedings, Part II, Springer LNCS 4488*, pp. 243–250, 2007.