

# 直感的な数式編集システムの開発

出口 博 章†

## Development of Intuitive Formula Editing System

DEGUCHI HIROAKI†

### 1. はじめに

数式は、一般のテキスト情報とは異なり、記号が配置される二次元的な位置関係によって意味を持つような構文を持っている。一般のテキスト情報においては記号の接続関係は前後関係で表せることが一般的だが、数式を構成する記号の接続関係には二次元的配置や記号サイズの変化も必要とされることが多い。

我々は、数式の二次元構造を直感的に扱うシステムのプロトタイプを開発し、GUIにおける特殊なドラッグ&ドロップである dynaput 操作<sup>1)</sup>を提案してきた。本論文では、dynaput 操作と従来のドラッグ&ドロップ操作との違いに加え、例外的な処理について述べる。

### 2. MathBlackBoard

MathBlackBoard は、使いやすい数式処理システムを目指した黒板アプリレット<sup>2)</sup>をベースとして、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、dynaput 操作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、入力機器としてマウスやタブレット機器などの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指しているが、マウス操作のみに

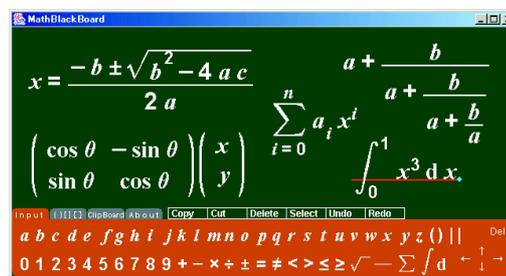


図 1 MathBlackBoard バージョン 2

Fig. 1 MathBlackBoard version 2

よっても利用可能である。また、キーボード操作による代替も可能となるようには設計されているが、操作性を最大限発揮させるためにはポインティングデバイスが必要となる。

#### 2.2 dynaput 操作の概要

記号オブジェクトをドラッグ&ドロップ操作で「くっつける」ようにして吸着させていくことによって数式を入力・編集する操作が dynaput 操作である。吸着させる対象となる記号の種類と吸着位置によって、吸着される記号の位置や大きさなどは適宜調整される。本システムでは、特殊なドラッグ&ドロップ操作で入力・編集操作の両方が提供されているため、その操作を単なる入力とは区別して dynaput と名づけた。

dynaput 操作の対象となる記号は、それぞれの記号ごとに記号オブジェクトとして生成される。記号オブジェクトは、そのオブジェクト自身を選択するための選択領域 (図 2, 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。これらの領域は、選択領域の周辺に配置されるため周辺

† 神戸大学発達科学部

Faculty of Human Development, Kobe University  
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。

dynamic + (in)put: 動的な入力 (あるいは配置)

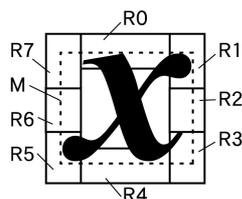


図 2 選択領域 (M) と周辺領域 (R0- R7)  
Fig. 2 Selection Area(M) and Peripheral Area(R0- R7)

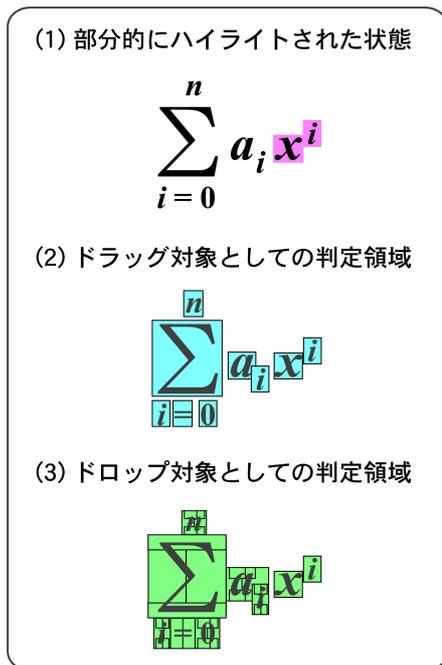


図 3 ドラッグ対象とドロップ対象  
Fig. 3 Drag Target and Drop Target

領域 (図 2, R0- R7) と呼ばれ, 必要となる場合にだけ画面上に表示されるように設定することが可能である. 例えば, ドラッグ操作中のポインタ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場合などに表示される.

### 2.3 一般的なドラッグ&ドロップ操作との違い

dynaput 操作と従来のドラッグ&ドロップ操作との違いは操作対象となるオブジェクトが周辺領域を持つかどうかというところにある. 従来のドラッグ&ドロップ操作には「記号の接続」という意味はなかったため, ドラッグ開始の際に利用される領域とドロップ先判定の際に利用される領域は同一のもので十分であった. しかし, 複雑な二次元構造を持つ数式を入力・編集するためには, オブジェクトの接続に対してきめ細かな指示が必要となり, 単純な「オブジェクトか否か」という判定だけでは不十分である.

そこで, より詳細な指示を可能とするために「オブジェクトから見てどういう位置関係か」という判定のための領域が追加され, ドロップ先の指定をきめ細かく行なえるようになったものが dynaput 操作である. 結果として, 従来の一般的な数式エディタでは「位置関係の指定」と「記号の入力」という 2 ステップが必要であった二次元的な編集操作が, 本システムでは「周辺領域を利用したドラッグ&ドロップ (dynaput 操作)」の 1 ステップで実現されている.

図 3 の (1) は  $x^i$  だけが部分的にハイライトされた状態である. また, 図 3 の (2) と (3) は, ドラッグ元決定時とドロップ先決定時に使用される領域を示している. ハイライトされたオブジェクトは例外として扱われ, ドラッグ対象とドロップ対象のどちらのケースにおいても同じ領域を使用するという従来の GUI オブジェクトのような存在となる. ハイライト状態のオブジェクトに他のオブジェクトをドロップした場合に置換処理を割り当てれば, 従来のテキスト編集と同程度の操作感が提供されることになる.

### 3. まとめと今後の課題

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際の dynaput 操作は特殊なドラッグ&ドロップ操作である. その操作対象オブジェクトは周辺領域を備えており, ドラッグ開始時とドラッグ終了時 (ドロップ時) で利用される領域が異なることを特徴としている. また, 例外的なオブジェクトとして扱うためのハイライト設定を利用者が行なったオブジェクトについては, ドラッグの開始と終了のどちらの際にも同じ領域を使用することが可能である.

dynaput 操作の結果として接続されたオブジェクトの構造は木構造であり各種の表記への変換が可能であるが, この木構造についての考察と数式以外の情報への応用が今後の課題となる.

謝辞 本研究の一部は, 情報処理推進機構, 2007 年度第 II 期末踏ソフトウェア創造事業の支援による.

### 参考文献

- 1) Deguchi, H.: 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).
- 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).