

手書きインタフェースのための図形階層文法

7K-6

中川正樹、福島英洋、風間信也

東京農工大学工学部電子情報工学科

1. はじめに

手書きボタンは、それが機能(コード)を表現する場合と、図形を表現する場合と、単に手書きボタン(筆跡ボタン)を表現する場合がある。手書きの利点は、ペン1本で様々なものを表現できることである。言い替れば、手書きボタンには多義性がある。たとえば、文字はコードであり、フローチャートの処理、論理回路のゲートや結線も機能を示す。一方、システムの構成や概念を示した図、おおまかな処理のダイアグラム、実物を抽象化した図などは機能として解釈される必要はなく、単に図形として整形されればよい。また、ある機能を表した図のなかでメモ書きされた図は機能とは言えない。なぜなら解釈すべきドメインが違うからである。さらに、その図に記されたサインは筆跡ボタンであり、文字に認識してしまうと意味がなくなる。図、一般に、文書とはこれらの混在した2次元のマルチメディア情報であり、その表現形式を定めなければならない。

2. 手書きボタンの階層性

図1に示すように、図形を筆点座標列で扱う場合には、手書きと同様の自由度が得られる反面、図形に対する編集操作は行にくい。逆に図形記述として扱う場合、編集や整形は容易であるが、表現の自由度は小さくなる。さらに進んで、機能シンボルとして扱えば、シミュレーションやプログラムへの変換も可能になり、論理的メリットは最大となるが、図形としての自由度はさらに小さくなる。

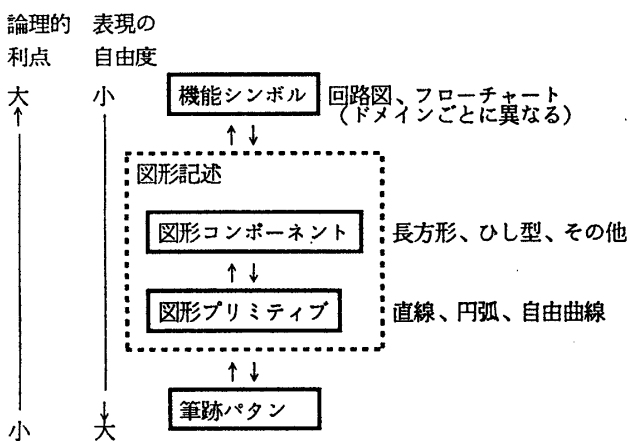


図1. 図の表現階層

機能シンボルは図形表現から還元されるもの、後者は前者から生々されるものである。還元の過程で、筆跡ボタンから図形表現にと

どめるのは、機能シンボルへの還元ができないもの、その必要がないものとする。筆跡ボタンにとどめるのも同様に、図形表現への還元ができないか、その意味がないものとする。これらの混在を許す枠組みが必要である。

計算機では、書かれたものが機能シンボルなのか、整形の対象になる図形なのか、何もすべきでない筆跡ボタンなのかの判断ができないので、筆点ボタン、図形表現、そして機能シンボル、の階層表現を内部表現とする。そして、処理が選ばれたときそれがどのレベルの表現に対するものかを決める(例、消しゴムが選ばれたら原ボタン、漢字への変換が指定されれば文字コード)。

認識は、これらの階層を下から上に上げることであり、メニュー選択などによる描画は上から下を指定することである(たとえば、NANDゲートを選択すれば、その図形表示が行われる)。

3. 図形階層記述文法

3.1 結合演算子について

表現される図形が1つの代数系をなすならば、それらを処理するプログラムは代数構造を利用して、再帰的に処理できる。そのためには、図形プリミティブ間に適用できる演算はそれらが結合されてきた図形コンポーネントにも適用できなければならない。対象図形が限定され、図形プリミティブの結合関係が制限されている場合には、図形プリミティブにも図形コンポーネントにも共通な演算が定義できる(従来の図形文法)。しかし、対象を限定しない図形では、図形プリミティブ間の演算に、各図形プリミティブの結合位置と関係を正確に指定する必要がある。これを、図形コンポーネントにまで解釈を拡張するには無理がある(例えば、図形プリミティブの端点を $n:m$ に内分する点は、図形コンポーネントに対してはどのような意味を持つのか)。

そこで、図形コンポーネントに対する演算では、その中の特定の図形プリミティブを指定し、それに対する演算と解釈する。これは、我々が図形を合成するとき実際にやっていることに対応している(箱から箱に矢印を引く場合、箱の一辺の midpoint からもう一方の箱の一辺の midpoint に線を結合させている)。一方、図形プリミティブに対する演算を、要素が一つしかない図形コンポーネントへの演算と考える。こうすれば、両者を統一的に扱える。

3.2 BNF 定義

次に図形記述文法の骨格を抜粋して示す。この文法に従って生成される構文木で、 $\langle \rangle$ は図の表現の階層である、機能シンボル、図形表現、筆跡ボタンに対応している。

<図> ::= {<要素関係>} * {<要素>} *
 <要素> ::= 《機能シンボル》 | 《図形記述》 | 《筆点ボタン》
 《機能シンボル》 ::= (@文字列 <文字列> |
 (@フロー図 <フロー機能> <フロー属性> |
 (@回路図 <回路機能> <回路属性>) | ...
 <文字列> ::= (<文字コード>)*

 《図形記述》 ::= <図形> | (<図形修飾> 《図形記述》) |
 (<図形変形> 《図形記述》)
 <図形修飾> ::= (@塗りつぶし { (Xi, Yi) } + <塗りつぶし属性> |
 (@線種 <線種> <太さ>) | ...
 <図形変形> ::= (@変換行列 A11 A12 A21 A22) | (@移動 ΔX ΔY)
 <図形> ::= <図形コンポーネント> | <図形プリミティブ>
 <図形コンポーネント> ::=
 (@ローカルフレーム <原点> {<2項関係>} * {<図形記述>} * |
 <文字ボタン列>

<2項関係> ::= (@接合 <図形プリミティブ番号> <位置>
 <図形プリミティブ番号> <位置> <接合角>)
 <位置> ::= 始 | 中 | 終 | /n:m
 <接合角> ::= @接合角 <数値>
 <図形プリミティブ> ::= <直線> | <円弧> | <だ円弧> | <自由曲線>
 <直線> ::= (@直線端点 <始点座標> <終点座標> |
 (@直線始点長角 <始点座標> <長さ> <方向>)

<文字ボタン列> ::= (@文字列枠 <左上隅> <右下隅>
 {<文字列修飾>} * {<文字コード>} *
 <文字列修飾> ::= @フォント <フォント種> | @サイズ <サイズ値>
 @行間ピッチ <ピッチ値> | @字間ピッチ <ピッチ値>

 《筆点ボタン》 ::= {<筆点座標>} +

<数値> ::= {<数字>} + | ?

3.3 補足説明

・<要素関係>は、要素の出現番号順にふられた id に対する、要素間の関係を指定する。

・塗り潰しでは、(Xi, Yi)を含む閉領域を塗り潰す。

・<図形修飾>は、図形コンポーネント、図形プリミティブ、筆跡ボタンのいずれに対しても指定可能で、内部の指定が外部の指定より優先する。<図形変形>も、それら3つの階層のいずれに対しても指定可能で、nest の場合は合成変換になる。

・<図形変換>では、変換行列 $A = \begin{pmatrix} A11 & A12 \\ A21 & A22 \end{pmatrix}$ を対象図形の (X, Y) に乗じる。

・<原点>はローカルフレームの原点を指定。ここに統合される図形記述はこの原点からの相対で配置される。

・<位置>で/n:m は、n:m の内分点を意味する。n, m は負数も有り得る。内分は、楕円弧、自由曲線にも適用する。完全性のため。

・<図形プリミティブ番号>はこのローカルフレームに直接または間接に含まれる図形プリミティブの一貫番号。

・<接合角>の<数値>は、図形プリミティブ1の接合点における接線（直線の場合そのもの）をX軸にしたときの、図形プリミティブ2の接線の角度。

・<数値>の?は wild card。他の束縛から値を決める。

・座標値は、0.01mm 刻みで指定する。*/

・{}*: 0個以上の繰り返し。{}+: 1個以上繰り返し。

3.4 文法の階層性

いま、図形記述の階層での図形コンポーネントを考える。これを、一機能として機能シンボルに還元することは、図形記述の階層での還元の結果としての非終端記号を機能シンボルの終端記号に還元することである。筆跡ボタン、図形記述、機能シンボルと3つの階層を分離して考えれば、下位の代数系から上位の代数系に移るとき、非終端→終端への還元があることを意味する。

図形記述から機能シンボルへの還元は、図形記述の多意性から還元というより認識になる。続けられた処理と流れの線を分離して、それぞれに認識しなければならない。その際、(機能シンボルの構成と対応するように) 図形記述の再構成が行なわれる。機能シンボルへの認識を行なう意味の一つは、この図形記述の再構成である。

4. 図形コンポーネントの表現例

ある図形プリミティブ1の中心から別の図形プリミティブ2の中心を結ぶ矢印の図形を、上の文法にしたがって考えてみる。表現はもちろん、一意ではない。

```

(@接合 1 中3 /-1:10) (@接合 2 中3 /11:-1)
<図形コンポーネント1の記述>
<図形コンポーネント2の記述>
(@線種 実線 細線 /* 矢印の記述 */
  (@ローカルフレーム (0, 0)
    (@接合 1 終2 終 @接合角 -30)
    (@接合 1 終3 終 @接合角 30)
    (@直線始点長角 (? ?) ??)
    (@直線始点長角 (? ?) 500 ?) /* 矢先の長さ 5mm */
    (@直線始点長角 (? ?) 500 ??)
  )
)
  
```

/* 矢印の記述 */ 以下をマクロとして、<macro 矢印>とすれば便利である。ここには、矢先の長さ以外、インスタンスに関する情報は無い。この定義自身は、図形コンポーネント1および2の配置の変更に完全に追従する。

5. おわりに

文房具メタファを用いた描画系[1]も図の認識系[2]も、ここに述べた手書きボタンの階層表現に作用する。紙面の都合から、図形コンポーネント同士の合成を例示することはできなかった。

参考文献

[1] 風間他：手書き図形入力インタフェースの試作、情処学会第44回全大、7K-5 (1992)。

[2] 福島他：オンライン手書き図形データベースの作成と手書き図形ボタンの解析、情処学会第44回全大、7K-4 (1992)。