

## 家紋とそのバリエーションを記述できる 図形記述言語

2N-1

石原 亘

(京都芸術短期大学 映像研究室)

### 0 始めに

一般に情報システムはスペックが肥大化しがちな傾向にある。とりわけ芸術的創作に適用される情報処理システムでは本質的にこの傾向が避けられない。これは、創作においては情報処理の形態に対する一定の規範というものがなく、どれほど充実した機能をもつ情報処理システムでもユーザの要求をカバーしきれないためである。

FXは、空間図形を対象とする処理を記述するために設計されたプログラミング言語であり、プログラミングを通じて美術/デザイン教育を行なうために用いられている<sup>0)</sup>。8年にわたる運用を通じて、いろいろな実習や制作のアイデアに対応するために、いくつかの新しい言語要素を加えた新版の必要性が感じられるようになった。ここでFXの新版が「いつまでもっとほしい」症候群を免れるためには、何かの規範が必要である。そこで、日本の伝統的家紋に着目し、家紋の意匠が効果的に記述できることを新版が備えるべき機能の目安にすることにした。本稿では、いくつかの家紋種について、それらを記述するために導入したFXの新関数について解説する。

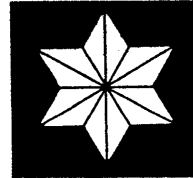
### 1 家紋

家紋は一族の徽章として用いられる図形であり、多くは動植物、器物、天文気象などの形態を圖案化したものに由来する。

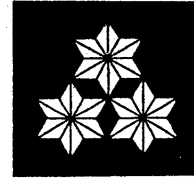
家紋として用いられている意匠の種類は文献<sup>1)</sup>によれば5000種以上にも達することが分るが、このような豊富なバリエーションがあるのは、基本形とでもいべき図形に対して変形や配列化、修飾などの操作を施すことによってさらに複雑な図形が構成できるためである。

この発展化には異なる基本形に対して共通に用いられるいくつかのパターンがあることが認められる。たとえば、図0 aは麻の葉の紋であるが、これを三つ円環に並べた三つ麻の葉(図0 b)も家紋として用いられている。これに対応するように、かきつばたに対しては尻合せ三つかきつばた(図1)、雁金に対しては尻合せ三つ雁金(図2)と呼ばれる紋が存在する。また、こうした発展化の操作はいくつか複合して適用されることもある。

家紋の多くが基本形に対する線形変換と彩色および複製から構成できることは以前から推察されていて、それをコンピュータ実験で確認したり、さらに家紋風の未知の意匠を発見しようとする試みがす



a



b

図0

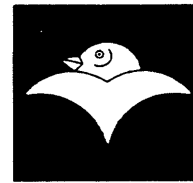


a

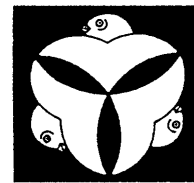


b

図1



a



b

図2

にデザインを研究する立場からなされている<sup>2)</sup>。しかし、家紋を構成する図形要素とそれらに対する発展化操作の全体像はまだ明らかにはされていない。

### 2 報告

#### 2.0 調査方法

今回の調査では、文献<sup>2)</sup>に収録されている5000種の家紋をすべて調査することは作業量からみて困難であったので、共通の基本形をもつファミリーをいくつか選んで、それらについてははたもかく全部のバリエーションについて記述を試みることにした。結局、植物紋から麻16種、動物紋から雁金50種と千鳥4種、天文紋から星56種、気象紋から洲浜56種および器物紋から分銅16種を調査した。これらは以上に示したようにきわめてバリエーションが充実している。また、基本形が単純なので描画に要する時間が短縮できるうえ、発展化操作の特徴が確かめやすい点で調査に適していた。

On Describing the Forms of the Japanese Traditional Family Crests with a Figure-Descriptive Programming Language

Wataru ISIHARA

Kyoto College of Art

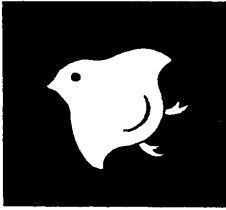


図 3

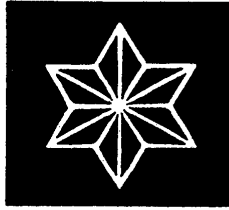


図 4

## 2.1 FX

FX (Formula eXpression)は親言語であるパスカルまたはCの中に埋め込んで用いられる副言語である。FXには図形を式として表現するための多数の関数と、それらに与える変位や色彩などをやはり式で表現するための補助的関数、および式で与えられた図形を表示したりファイルに記録したりするための四つの命令が含まれている。

FXには基礎とする図形モデルの違いに応じてソースコードが共有できる複数のバージョンがあるが、今回の調査では家紋の表現に最も適したサーファモデルに基づくものを中心に検討を行なった。

FXの初版を設計した時は、明解な言語ができるよう、以下に示す設計上の原則を設けていた。この原則は、今回新しい関数を導入するにあたってもしたがうものとした。

a 変位や色彩など適切なパラメタが導入できるならば、複数の関数をパラメタをともなった単一の関数に統合する。

b ほかのより原始的な関数を成分とする合成関数とみなせるものについてはそれらの成分を除いた関数を代わりに導入する。

c 手続き表現を使わずに意匠が記述できるようにする (FXそのものは親言語の機能に基づき手続きによって図形を生成することができる)。

## 2.2 調査結果

FXの要素として重要であることが再確認されたものや、新たにFXの関数として追加するべきであると判断されたものがいくつかあった。それらを機能ごとに分類して以下に示す。

0) 定数 家紋の基本形のいくつかはFXに在来の正多角形や円を生成する定数 (無変数関数として実現している) で記述できた。

千鳥の基本型 (図3) や本体と合併して修飾に用いられるいくつかの図形は自動的に生成できそうにない。そこで、残念ではあるが、これらは固定されたものとしてモデル化した情報をファイルに記録しておき、それをロードして使うようにした。このため、在来の機能であるが、ファイルの内容に基づいて図形を起す関数が頻繁に使われた。

1) 運動関数 FXは平行移動、各座標軸に関する

回転、剪断を含む拡大/縮小が施された図形を生成する関数をもつので、家紋に用いられているアフィン変換はすべて記述することができる。

彩色については従来の仕様の見直しが必要になった。FXは彩色された図形を表す関数をもっていたが、これまでのそれは図形を構成する頂点、稜線、表面のすべてが等しく指定された色で塗られていた。ところが、陰型とでもよぶべきバリエーション (図4) には対応できなかった。陰型は表面を黒く、稜線は白く塗ることによって導出されるからである。これを解決するために、彩色関数は表面にしか塗らないことにして、稜線と頂点を塗る関数を別に新しく設けた。

2) 演算 指定された二つの図形の合併を表す関数を追加した。FXは本来不定個数変数の合併関数をもっていたが、whileなどの繰り返し形式を用いてオペランドを双方向リストに収めてから関数に渡さなければならなかった。修飾が加わるのを表現するのにも行数が増え、しかも余計な変数が必要になるため原則aとのトレードオフを考慮したうえで2変数の場合だけを特別扱いすることにした。

さらに、指定された図形を丸く並べる関数と縦横に並べる関数を新たに導入した。家紋に現れる合併はほとんどこのどちらかに限られているのでこれらの関数は原則cの達成によく貢献している。

ほかにチェッカー配置ができるようにするための二つの図形を引数に与える縦横配置関数を作ったが用途が特殊に過ぎるためさらに検討が必要である。

## 3 終りに

この調査を通じて、FXは家紋を始めとする各種の図形を記述するに十分な能力が獲得できたといえる。特に線/環形配置関数の効果が大きい。教育の立場から見れば状態の遷移の概念の導入をする前に高度なプログラミングを実習させることが可能になったし、デザインの内容がコントロールしやすくなった。

なお、家紋などの図形を画像としてではなく式として表現することが可能になったので、今後の課題としては、内部表現をFXでもつ図形検索システムの実現の可能性を探りたいと考えている。また、そのユーザインタフェースのために、FXと同等の機能をもつ独立したプログラミング言語とそのインタプリタの開発を行ないたい。

## 参考文献

- 0) 幸村真佐男、石原亘、Hild, Michael: コンピュータグラフィックスによる美術デザイン教育, 情処研報/グラフィクスとCAD, Vol.90-CG-47-2 (1990).
- 1) 本田総一郎: 新集家紋大全 (第7版), 梧桐書院 (1989).
- 2) 森久敏: パソコンによる三次元自己相似図形処理に関する一試み, 情処研報/グラフィクスとCAD, Vol.92-CG-58-5 (1992).
- 3) 石原亘: FXリファレンスマニュアル (1990).