

## FBuilder-24(V7.8) 機能説明書(基礎編)

## 目次

	頁
1 初めに . . . . .	2
2 作成するフラクタル	2
2.1 フラクタル作成の流れ	2
2.2 作成するフラクタル種類	2
3 基本機能 . . . . .	2
3.1 各フラクタルの概要と指定するパラメータ	2
3.1.1 マンデルブロー (Mb:Mandelbrot)	3
3.1.2 ジュリア (Ju:Julia)	3
3.1.3 コレット (Ct:Collet)	3
3.1.4 ダストDt	3
3.1.4.1 回転、縮小、鏡像による変換を繰り返すダスト : DustBk	3
3.1.4.2 ランダムダストRn (DustRn←Random)	4
3.1.4.3 逆追跡型ダストRv (DustRev←Reverse)	4
3.1.4.4 順追跡型ダストFwd (DustFwd←Forward)	4
3.1.5 ミラー (Mr:Mira) . . . . .	5
3.1.5.1 基本ミラー	5
3.1.5.2 拡張ミラー	5
3.1.6 ピタゴラス (Py:Pyth)	5
3.1.7 蛇行曲線 (Me:Meander)	5
3.1.8 星型 (St:Star)	6
3.2 関連する共通的事項 . . . . .	7
3.2.1 画像サイズの指定	7
3.2.2 表示サイズと表示位置の調整	7
3.2.3 試行回数LNの設定	8
3.2.4 出力値の設定	8
4 作成したフラクタルに対する処理機能 . . . . .	8
4.1 指定点を表示画像面の中心に配置する拡大・縮小・移動	8
4.2 カラーの調整	9
4.3 作成したフラクタルの保存と外部公開	9
4.4 作成したフラクタルの登録と再生	9
4.5 出力された.csvファイルに関する制限事項	9
5 参考 : Motifの詳細 . . . . .	10
6 参考文献	10

## 1 はじめに

本説明書はFBuilder-24を使用する為の基礎的事項を中心に説明します。

- ・フラクタルの原理的事項及び名称は例えば6-[1]を参照下さい。
- ・拡張的・全体的機能については、FBuilder-24(V7.8)機能説明書(拡張編)を参照下さい。  
拡張編の項番例えば、3.1を[3.1]の形で参照します。
- ・操作に必要な事項はFBuilder-24(V7.8)操作説明書を参照下さい。  
操作説明書の項番例えば、3.1を(3.1)の形で参照します。
- ・本書自体の項番は単に3.1で参照します。

## 2 作成するフラクタル

FBuilder-24では2.2の8系統のフラクタルを作成します。

### 2.1 フラクタル作成の流れ

本システムでは、フラクタルは凡そ下記の流れで作成し、24ビットフルカラーで着色します。

- i 下記のパラメータに従って、指定される画像を**新規作成**します。
  - ・フラクタル系統の指定：2.2、3.1項のフラクタル種類の指定(OPN)
  - ・メニューボックスで選択・指定される細分情報(CB0, CB1, CB2)
  - ・実数で詳細指定する主パラメータ(TV0~TV10)
  - ・縦横のサイズ情報(WD, HT)
  - ・作成画像を表示画面上に適正に配置する
- ii **新規作成後の修正**：新規作成された画像に磨きをかけます。
  - ・出力レベルを適正にする為出力レベルを変換する
  - ・出力レベルとカラーパレットとの対応付け、カラー調整をする
  - ・中心点又は移動先を指定して、画像を拡大・縮小・移動させる

### 2.2 作成するフラクタル種類

表2.2 に本システムで作成するフラクタル種類を要約します。

表2.2 フラクタルの種類

OPN	和文名	英文名	略称	描画対象
1	マンデルブロー	Mandelbrot	Mb	表示画面全面
2	ジュリア	Julia	Ju	
3	コレット	Collet	Ct	X軸は全点、Y軸は点の集合
4	ダスト	Dust	Dt	
5	ミラー	Mira	Mr	
6	ピタゴラス	Pyth	Py	直線と多角形の集合
7	蛇行曲線	Meander	Me	
8	星型(スター)	Star	St	直線の集合

注：フラクタル名称は文献6-[1]によります。

## 3 基本機能

### 3.1 各フラクタルの概要と指定するパラメータ

各フラクタルに対応して、表示画面の各座標点に対する画素値を算出しますが、その算出の流れは下記 i, ii の2種類に分類されます。表2.2のNo.3は i, ii の中間です。

- i 表示画面の各(物理)座標(X, Y)に対応する画素値 $z$ を求め、相当するカラー値を座標(X, Y)にプロットする  
表2.2のNo. 1, 2の表示画面を塗り潰す(面系)フラクタルが該当。
- ii 生成画像の画像座標 $[x, y]$ を求め対応する物理座標(X, Y)に画素値に相当するカラー値をプロットする  
表2.2のNo. 3~8の点又は線によるフラクタル(点系)が該当。出力不在の背景値が生成。
- iii 画素値をカラー値に変換する方式については、[4章]を参照下さい。以下に作成するフラクタル毎に画素値の算出法の概要及び指定するパラメータを記します。
- iv 本書では、各ジャンルで最も基本的な項目のみを説明します。拡張的項目は拡張編を参照ください。

### 3.1.1 マンデルブロー(Mb:Mandelbrot)

複素数  $z$  に対して、

$z_{n+1}=F(c, z_n)$  ( $n=0\sim 2^{LN}-1$ ) の繰り返しで  $|z_n|>Limit$  となった時の最初の  $n$  を  $c=(x_c, y_c)=(x_c+iy_c)$  に対応する座標  $(x_c, y_c)$  にプロットします。  $i$  は単位虚数。  $i^2=-1$ 。

$n=2^{LN}-1$  に対しても、  $|z_n|\leq Limit$  となっている場合には、指定によりその時点の値が出力されます。

- i  $LN$  で繰り返し回数の上限を指定。
- ii 初期値  $z_0=(x_0, y_0)$  は主パラメータ  $TV$  で指定。  
この値は3.1.2のジュリア画像の座標から取得することも出来ます。[5.1.4]を参照。
- iii その他、論理座標での大きさを指定する  $ML$ =倍率指定、**画像座標の中心  $[0, 0]$**  を配置する  
物理(=表示)画面の**論理座標  $[LX, LY]$**  を指定します。3.2.2項を参照。
- iv  $F(c, z_n)$  の関数系の全体、パラメータ及び拡張機能については、拡張編を参照。

### 3.1.2 ジュリア(Ju:Julia)

複素数  $z$  に対して、

$z_{n+1}=F(c, z_n)$  ( $n=0\sim 2^{LN}-1$ ) の繰り返しで  $|z_n|>Limit$  となった時の最初の  $n$  を  $c=(x_c+iy_c)$  をパラメータとして、初期値  $z_0=(x_0+iy_0)$  に対応する座標  $(x_0, y_0)$  にプロットします。  $n=2^{LN}-1$  に対しても、  
 $|z_n|\leq Limit$  となっている場合には、指定によりその時点の値(複素数)が出力されます。

- i  $n, LN, Limit$  はマンデルブローの場合と同じです。  
パラメータ  $c$  はマンデルブロー画像の座標より取得することも出来ます。[5.1.4]を参照。
- ii  $F(c, z_n)$  の関数系の全体、パラメータ及び拡張機能については、拡張編を参照。

### 3.1.3 コレット(Ct:Collet)

$y_{n+1}=x \cdot Fc(p, y_n)$  ( $n=0\sim 2^{LN}-1: x=Xmin\sim Xmax$ ) の繰り返しに於いて、各  $x$  に対して、 $n$  に対応する値を  $(x, y_n)$  にプロットします。

- $Fc(p, y_n)$  の関数系は[6.2]を参照。
- $LN$ (既定値は8: 詳細は3.2.3項を参照)
- $y_0$ (既定値は0.5)
- $Xmin\sim Xmax$ (既定値は2~4: 但し  $Fc(p, y)$  に依存します)
- $ML$ 、 $(LX, LY)$  については3.2.2項を参照。
- 出力図形の配置は、横方向は  $Xmin, Ymax$  を表示画面全体に広げて、縦方向は全体が全て収まる様に、表示位置  $(LX, LY)$  と尺度  $(ML)$  を(極端な値が出力されない限り)調整します。

### 3.1.4 ダストDt

#### 3.1.4.1 回転、縮小、鏡像による変換を繰り返すダスト: DustBk(DustBk←BackTracking)

座標  $(x_n, y_n)$  に対して、下記の  $L, R$  の変換式で決まる点の集合であります。

(1) 方式1 ( $R\ Start=0$ ): (3.1.4.1)式の変換を  $L\rightarrow R\rightarrow L\rightarrow R\rightarrow \dots$  の順で変換を繰り返す

(2) 方式2 ( $R\ Start=1$ ): (3.1.4.1)式の変換を  $R\rightarrow L\rightarrow R\rightarrow L\rightarrow \dots$  の順で変換を繰り返す

$$\left. \begin{array}{l} L: \begin{cases} x_{n+1} = m_{xL} \cdot a \cdot x_n - m_{yL} \cdot b \cdot y_n \\ y_{n+1} = m_{xL} \cdot b \cdot x_n - m_{yL} \cdot a \cdot y_n \end{cases} \\ R: \begin{cases} x_{n+1} - 1 = m_{xR} \cdot c \cdot (x_n - 1) - m_{yR} \cdot d \cdot y_n \\ y_{n+1} = m_{xR} \cdot d \cdot (x_n - 1) - m_{yR} \cdot c \cdot y_n \end{cases} \end{array} \right\} \quad (3.1.4.1)$$

ここで、

変換  $L$  は、点  $(0, 0)$  の周りの角度  $\alpha$ 、縮小率  $R_\alpha$  の回転、縮小、鏡像で、

変換  $R$  は、点  $(1, 0)$  の周りの角度  $\beta$ 、縮小率  $R_\beta$  の回転、縮小、鏡像で下記(3.1.4.2)より与えられます。

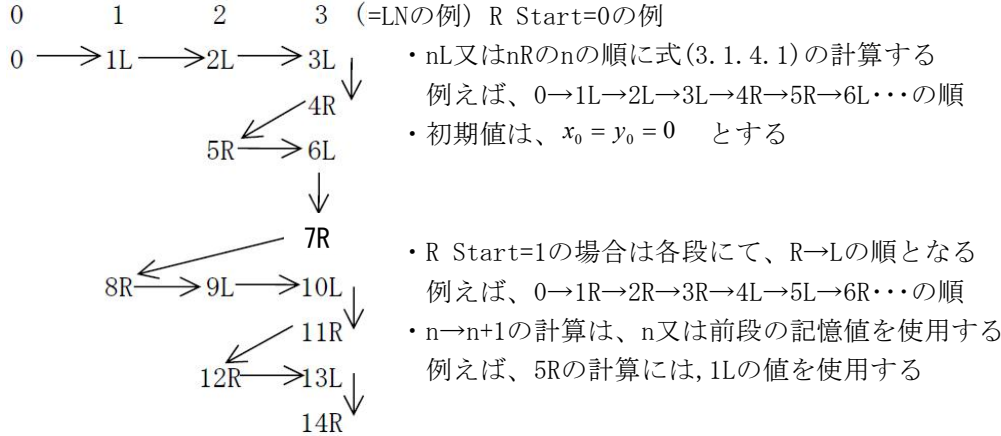
注: 詳細は例えば文献6-[1]を参照下さい。

$$\left. \begin{array}{l} R_\alpha = \sqrt{a^2 + b^2} \quad \cos(\alpha) = \frac{a}{R_\alpha} \quad \sin(\alpha) = \frac{b}{R_\alpha} \\ R_\beta = \sqrt{c^2 + d^2} \quad \cos(\beta) = \frac{c}{R_\beta} \quad \sin(\beta) = \frac{d}{R_\beta} \end{array} \right\} \quad (3.1.4.2)$$

(3.1.4.1)の  $m_{nM}$  ( $n = x \text{ or } y, M = L \text{ or } R$ ) は鏡像の有無を表し1又は-1として、パラメータMirrorXL、MirrorYL、MirrorXR、MirrorYRにより以下の如く与えられます。

$$\begin{aligned} m_{xL} &= \begin{cases} +1 & \text{MirrorXL} = 0 \\ -1 & \text{MirrorXL} = 1 \end{cases} & m_{yL} &= \begin{cases} +1 & \text{MirrorYL} = 0 \\ -1 & \text{MirrorYL} = 1 \end{cases} \\ m_{xR} &= \begin{cases} +1 & \text{MirrorXR} = 0 \\ -1 & \text{MirrorXR} = 1 \end{cases} & m_{yR} &= \begin{cases} +1 & \text{MirrorYR} = 0 \\ -1 & \text{MirrorYR} = 1 \end{cases} \end{aligned} \quad (3.1.4.3)$$

計算は下記に示す2進木の矢印の順に従って行います。**BackTracking法**。



- パラメータの設定としては、 $\alpha(=0-2)\pi, \beta(=0-2)\pi, R_\alpha(=0-1), R_\beta(=0-1)$  を指定し(4.1.3.2)により(4.1.3.1)のa, b, c, dを算出します。 $\alpha, \beta$  は $\pi$ に対する比で入力します。
- メニューボックスCB1でMirrorXL、MirrorYL、MirrorXR、MirrorYR、の組合せが指定されます。指定法は[6.3]を参照。
- (LX, LY), MLについては、3.2.2項を参照。
- 繰り返し回数LNを増大させると、より細部の描写が可能になります。

### 3.1.4.2 ランダムダストRn(DustRn←Random)

3.1.4.1と同様に、回転、縮小、鏡像で作成しますが、L変換かR変換を選択するのに、2進木ではなく各々**確率0.5でランダム**に行います。[6.3.2]を参照。

- R Start=0とします。
- L変換かR変換を選択するための乱数発生系列の初期化として、Seed(s)、s=0, 1, 2, ...を指定します。  
sの選択により乱数の発生系列が可変になります。
- その他、3.1.4.1と共通であります。

### 3.1.4.3 逆追跡型ダストRev(DustRev←Reverse)

関数 $z_{n+1}=F(c, z_n)$ を逆に追跡して、得られる座標 $z_n$ をプロットしたもの。[6.3.3]を参照。

一般に、 $z_n=F^{-1}(c, z_{n+1})$ を計算すると、 $z_{n+1}$ に対して複数の $z_n$ が得られますので、多数の点が作成されます。この多数の点を表示する事によって、複雑な画像が得られます。

### 3.1.4.4 順追跡型ダストFwd(DustFwd←Forward)

関数 $z_{n+1}=F(c, z_n)$ を順方向に追跡して、得られる座標 $z_n$ にその時の繰り返し回数nをプロットしたものであります。

関数 $z_{n+1}=F(c, z_n)$ におきましては、 $z_{n+1}$ は一意的に決まり3.1.4.3に比べて単純ではありますが、

- 有効な画像を生成させるには、可也の工夫が必要です。[6.3.4], [7.5], [7.7], (9章)を参照。
- 単純なるが故に、各生成関数 $F(c, z_n)$ の特徴を直接表示しますので、関数の動きを見るのに有用であります。

### 3.1.5 ミラー (Mr:Mira)

#### 3.1.5.1 基本ミラー

$$\begin{cases} \begin{cases} x_{n+1} = b \cdot y_n + F(x_n) \\ y_{n+1} = -x_n + F(x_{n+1}) \end{cases} & F(x) = \begin{cases} a \cdot x + (1-a) \frac{2x^2}{1+x^2} & (cb1=0) \\ a \cdot x + c \cdot \sin(x) & (cb1=1) \\ a \cdot x + c \cdot \cos(x) & (cb1=2) \\ a + c \cdot \sin(x) & (cb1=3) \\ a \cdot x + c \frac{x^2}{1+|x|} & (cb1=4) \end{cases} \\ & \begin{cases} a \cdot x & (|x| \leq 1) \\ a \cdot x + c \left( x - \frac{x}{|x|} \right) & (|x| > 1) \end{cases} & (cb1=5) \end{cases} \quad (3.1.6.1)$$

(3.1.6.1)の漸化式により得られる点 $(x_n, y_n)$ に $n$ の相当値をカラー値としてプロットします。

(3.1.6.1)に於いて、

- i 主パラメータ $a, b, c$ は、 $cb1$ 番号(0~5:メニュー**CB1**で指定)に依存します。
- ii 主パラメータで $a, b, c$ と初期値 $(x_0, y_0)$ を指定します。
  - $cb1=0$ の場合は、 $a, b$ を指定します。
  - $cb1=1 \sim 5$ の場合は、 $b=1$ として、 $a, c$ を指定します。
  - $(LX, LY), ML$ は3.2.2項を、次数 $LN$ については、3.2.3項を参照。

#### 3.1.5.2 拡張ミラー

(3.1.6.1)の関数 $z_{n+1} = Mr(z_n)$ を拡張します。[6.4]を参照。

### 3.1.6 ピタゴラス (Py:Pyth)

- i ベースとなる矩形( $L=0 \sim$ )の上辺に3角形の長辺を接合させます。  
図3.1.6を参照。
- ii 3角形の短辺の各々に縮小矩形を更に接合させます( $L=1 \sim$ )。
- iii 二つの縮小矩形の上辺に(1),(2)を繰り返します。  
3.1.4.1のBackTracking法により全ての方向に枝分かれさせます。
- iv 指定する主パラメータは下記
  - HS: ベースとなる矩形( $L=0$ )の高さ
  - WS: 上記矩形( $L=0$ )の幅(基準として、1とする)
  - $\alpha, \beta$ : 接合3角形の角度( $0 \sim 1$ )  $\pi$   
 $\pi$ に対する比で入力します。 $\alpha + \beta < 1.0$ とします。
  - LN: 繰り返し段数(図3.1.6は $LN=3$ の例)
  - $(LX, LY), ML$ : 自動調整します。3.2.2項を参照。
- v 指定するパラメータは下記。指定法は[6.5]を参照。
  - Btm0: 最初の矩形( $L=0$ )の底辺に枠線を引く
  - Btmn: 縮小矩形( $L=1 \sim$ )の底辺の線を引く
  - Topn: 縮小矩形( $L=0 \sim$ )の上辺の線を引く

これは出力図形の中心を木の幹に見立てる為のものであります。

- $\alpha, \beta$ の配置はランダムにSwapできますが、この指定法は[6.5], [7.6]を参照。

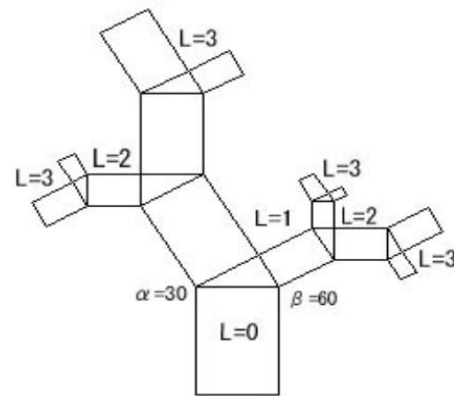


図3.1.6 ピタゴラス

### 3.1.7 蛇行曲線 (Me:Meander)

- (1) BaseとMotifの繰り返しにより作図します。
  - i Baseの各辺をMotifで置き換えます(図3.1.7.1の $LN=2$ )。
  - ii Motifの各辺を更に縮小Motifで置き換えます。  
図3.1.7.1の( $LN=2$ ) $\Rightarrow$ ( $LN=3$ )

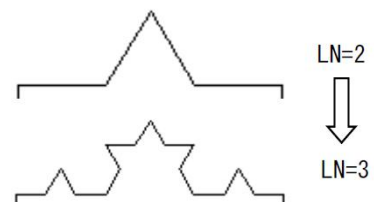


図3.1.7.1 蛇行曲線の上部

- iii iiを指定された段数 $LN$ 回繰り返します。図3.1.7.1を参照。

2進木を使用して、周辺を順に作図します。

iv 図3.1.7.1はNo.6のBaseにNo.3のMotif( $L=1$ ,  $\theta=1/3$ )を置換した例。

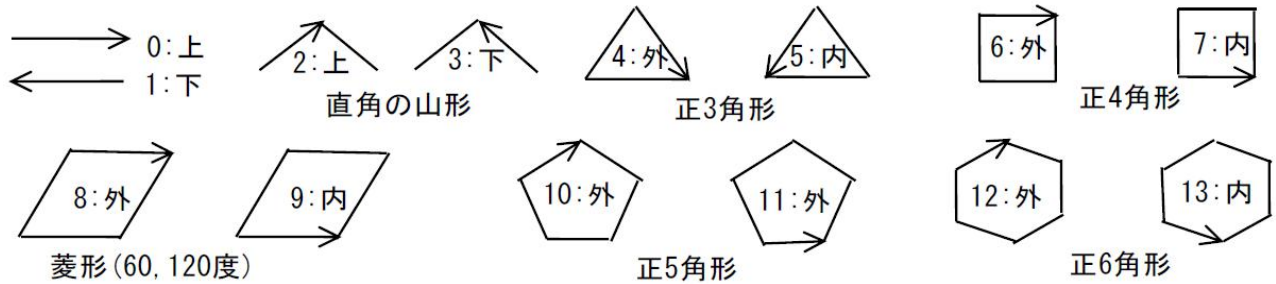


図3.1.7.2 Base(0~13)の一覧：矢印はMotifの置換方向

(2) 組み込んだBaseの種類：図3.1.7.2

- ・ 外、内はMotifが外側(時計周り)に置換するか、内側(反時計周り)に置換するかを示します。
- ・ 矢印はMotifの置換(進行方向左側に)単位と方向を示します。

(3) 組み込んだMotifの種類：図3.1.7.3

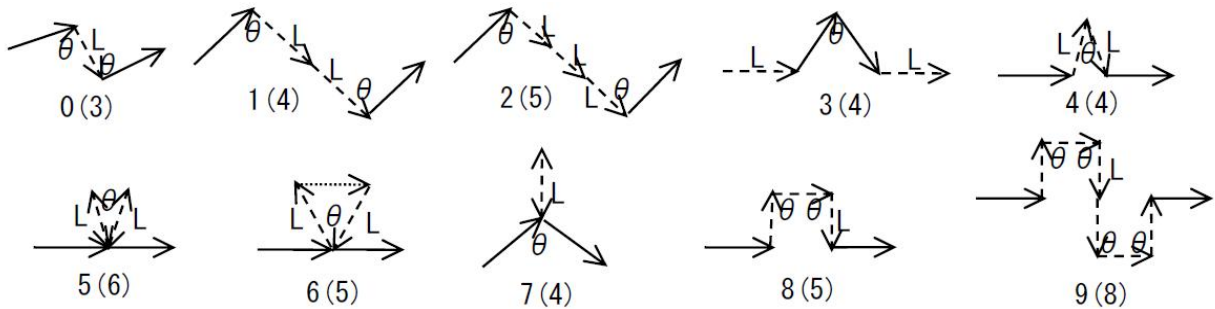


図3.1.7.3 Motif(0~9)の一覧：矢印は次の置換単位

- 基準長=1の**固定辺**：各Motif内で一定です。
  - 固定長に対する**相対長Lの可変辺**：TV7でMotif対応に指定できますがMotif内では一定です。
  - 成行きで決まる辺：MotifNo=6
  - 往復方向に置換され、置換Motifは進行方向左側に置換されます。MotifNo=7
- $\theta$ は各Motif内では全て等しく可変( $\pi$ 比)指定可能。

()内に置換単位としての辺の数を示します。
- $\theta$ とLの指定結果により、異なるMotifで同じ結果になるペアがあります。例：

Motif3と4は $\theta$ とLが等しければ同じ結果に。Motif7の $\theta=1$ とMotif4の $\theta=0$ で同一Lで同じ結果に。

(4) Motif番号はCB1のメニューで指定し、

BaseはTV4で、 $\theta$ はTV6で、LはTV7で指定されます。

その他LN, (LX, LY), MLがあります。各点の座標は5章を参照。

更なる詳細は、[6.6.1], (5.2-④), (6.8)を参照。

(5) Motifのみで描画する拡張型蛇行曲線(Dragon)については、[6.6.2]を参照。

### 3.1.8 星型(St:Star)

線分の縮小回転により作成するフラクタルで、一つの線分(方向付)の先端を例えば図3.1.8に示す如く縮小・折り曲げを繰り返します。

i 図3.1.8に下記パラメータの星形の例を示します。

- ・  $LN=2(=TV0)$ 、 $Lc=LineCount=5(=TV4)$ 、 $\theta=0.8(=4/5)(=TV5)$ ： $\pi$ に対する比で入力。
- ・ V7.3では $K=LC*\theta$ (整数1~)を $\theta$ に替わってTV7で指定できるようにしました。  
TV7=0の場合は、TV5の $\theta$ が使用されます。本例は、 $K=4$
- ・  $Ratio=0.3(=TV6)$ ：先端の星部分のサイズを指定します

ii 作図は下記によります。

- $0\theta$  で示す長さ=1の有向線分を引く(左⇒右)
- 右端より  $1\theta$  回転させ、 $1\theta$  で示す長さRatio分の線分を引く
- 更に  $2\theta, 3\theta$  の線分を引き、長さ1の  $4\theta$  の線分を引く
- 同様のことを  $4\theta$  の線分の先端で行い、更に  $8\theta, 12\theta, 16\theta$  の長さ1の線分を作る
- $20\theta$  の線分は元の  $0\theta$  に戻る。図3.1.8は下記によりカラー化しております。
- Green→Redに変化する直線形パレット：[4章]を参照
- 背景は白

iii LNを増大させることにより、各線分の先端に更に縮小された模様が作成されます。K=LineCount(Lc) ×  $\theta$  ( $\pi$ 比)として、K=整数が必要です。図3.1.8の場合は、 $5\theta=2$ 、 $\theta=0.8$

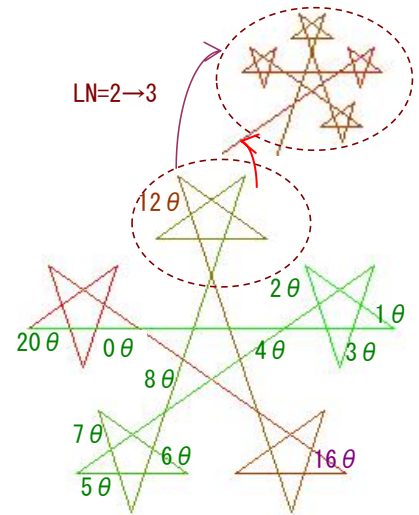


図3.1.8 星形フラクタルの作図

### 3.2 関連する共通的事項

#### 3.2.1 画像サイズの指定

- 画像サイズは横、縦独立にピクセル単位に指定します。既定値は512 x 512。各サイズは4の倍数でなければなりません(.bmp画像の仕様)。4の倍数でない値が指定されると最寄の4の倍数に自動調整されます。横又は縦のどちらかを -nに指定すると、指定した辺をn(=1~4)に対応するアスペクト比の矩形の短辺に自動設定されます。詳細は[5.1.2]を参照。
- 矩形指示の場合は正方形画像をベースとして下記2種類の方式で処理します。詳細は[5.1.2]を参照。
  - カット処理：短辺方向の不足分はカットします。
  - 圧縮処理：正方形画像の短辺方向は短辺長/長辺長の比で圧縮表示します。

#### 3.2.2 表示サイズと表示位置の調整

(1) 本システムでは、下記の3種類の座標を使用します。

- 物理座標(PX, PY)：表示画面のピクセル単位の座標 図3.2.2で()で表示
- 論理座標[LX, LY]：画像を配置する時の正規化座標 図3.2.2で[]で表示
- 画像座標[FX, FY]：生成画像の位置を定義する為の座標 図3.2.2で{}で表示

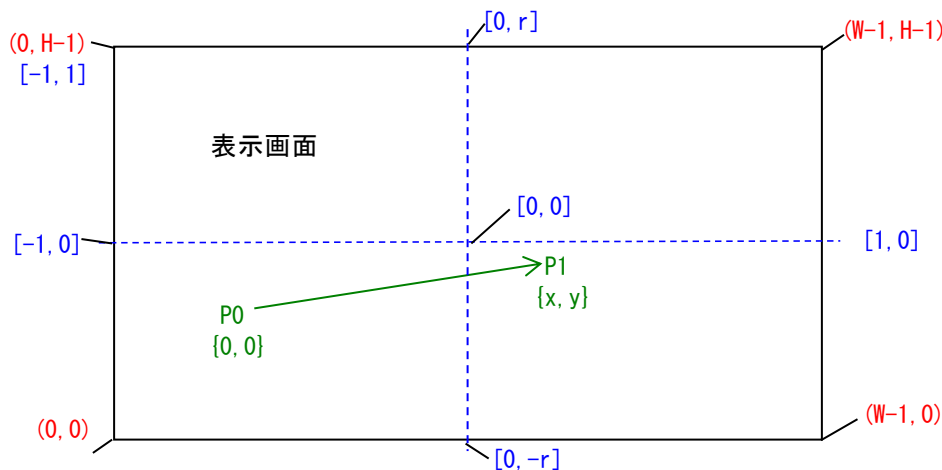


図3.2.2 各種座標の相互関連

図3.2.2でお互いの関係を表示します。図3.2.2に於いて、

- $W$ =表示画面の横サイズ(ピクセル)、 $H$ =表示画面の縦サイズ(ピクセル)、 $W \geq H$ 、 $r=(H-1)/(W-1) \leq 1$ とします。
- $[LX, LY]$ は画像座標  $\{0,0\}$  点を配置する(TV2, TV3)で指定される論理座標。論理座標での単位長は、
  - 長辺方向は、長辺の1/2とします。
  - 短辺方向は、圧縮形の場合は、 $r$  倍に縮小されます。カット形の場合は正方形の場合と同じになります。
- $ML=TV1$ で指定される論理倍率で画像座標の1単位長を配置する論理長であります。



- iv **物理座標**は表示画面左下を(0, 0)とします、**論理座標**は表示画面の中心を[0, 0]とします。  
**画像座標**の[0, 0]点は各画像の生成関数で決まります。
  - v 図3.2.2に於いて、**画像[0, 0]点P0**の配置は[LX, LY]で指定されます。任意の**画像[x, y]点P1**は、  
 $P1=P0+[ML \cdot x, ML \cdot y]$ の形で指定されます。**倍率ML**を大きくすると、P1はP0に対して離れた位置に存在し、  
P1を表示画面内に配置すると、P0は表示画面から遠く離れた位置になります。[5.1.5]を参照
- (2) 初期作成したフラクタル画像を表示領域に適切に配置いたします。
- i マンデルブロー、ジュリア、コレット(x軸)に対しては、初期作成時において適正値を直接指定します。  
中心位置の移動、倍率調整が必要なら作図後に行います。4.1項を参照。
  - ii コレット(y軸)とその他の点又は線で作成するフラクタルは初期指示されたパラメータ[LX, LY], MLにより仮作成して、作成画像の論理位置の表示画面での上下・左右(コレットを除く)の広がりによって、  
最適の**配置[LX, LY]**と**倍率ML**を算出して表示しますので、通常はこの値を使用して自動再実行して  
適正に配置します。
  - iii 必要により、右クリックによる中央配置と倍率設定により配置を調整します。
  - iv 更に、マウスポインターの表示画面上の**物理座標(X, Y)**及び**画像座標[x, y]**を各画面のヘルプ行に表示  
させることが出来ます。詳細は(4.2), (6.7.2)を参照。

### 3.2.3 試行回数LNの設定：点と線で作成するフラクタルの場合

変換の段数LNは本来無限に実施する必要がありますが、処理時間、モニタの分解能を考慮して指示された値で打ち切ります。**M倍に拡大指示**された場合にはLNは $\text{Log}_2(M)$ だけ増大させて更なる細部を表示します。これは3.1.3項以降に共通で4.1項を参照してください。LNは実数で指定しますが、フラクタル種類によっては処理上実質整数として扱われます。

### 3.2.4 出力値の設定

色付けの為の出力値は、計算の進行に伴う通し番号nに対応した値(0~255)に変換します。実際の表示色は以下の2段階で調整します。[4章]を参照。

- ・出力値nの変換：出力値そのものを適正な分布になる様にレベル変換します。
- ・カラー値の調整：上記の出力値に対して割り当てるカラーを調整します。

## 4 作成したフラクタルに対する処理機能

新規作成したフラクタルに対して下記処理が出来ます。

- ・指定点を画面中心又は移動先とする拡大・縮小・移動
- ・出力レベルの変換/カラーの調整
- ・作成したフラクタルの保存と外部公開
- ・作成したフラクタルの作成パラメータをDBファイルに登録し再生・表示する

### 4.1 指定点を表示画像面の中心に配置する拡大・縮小・移動

- i 指定点(右クリック指示)を中心に指定倍率で拡大・縮小・移動することが出来ます。これにより、  
拡大しても同様な模様が現れるフラクタルの特徴を検証することが可能になります。又縮小すること  
より全貌を観察でき、又単に中心位置の移動の為に等倍率を使用することも出来ます。
- ii マンデルブロー、ジュリアは拡大しても処理時間には大きくは影響しません。但し、拡大・縮小・移動  
により描写領域が変動して発散条件が変わり処理時間に影響することはあります。
- iii その他の、点又は線によるフラクタルでは、拡大率に相当するLN(次数)を増大させて、画面上で同様の  
図形を維持しますが、この結果、**累積拡大率の増大と共に作図時間が指数的に増大**します。  
作図の進行状態はプログレスバーで表示されますので、限界以上に拡大させないことが必要です。  
但し、次数を増加させない拡大も指示できます(**Ctrl**キーを併用する)。



## 4.2 カラーの調整

- i 作成したフラクタルには作成時の最終出力レベルに対応して(R, G, B)毎にパレット値(0~255)が定義され24ビットカラーとして着色されます。  
R, G, B対応出力は各対応の出力領域に出力され、その値で混合着色されますが、
  - ・FBuilder24では、R, G, B対応出力を(R, G, B)パレットで変換して各対応の出力領域に出力します。
  - ・詳細は[4章], (7.3)を参照。
- ii パレットが特に指定されていない場合には、既定のパレットが使用されます。
- iii R, G, B 3要素と実際の表示色の対応を任意に切り替える(6通り存在)ことができます。
- iv パレットの新規作成
  - ・RGB各カラー対応にパレット曲線を作成します。詳細は(7.3)を参照。
  - ・V7.5では既定パレットを統一整理しました。詳細は機能説明書(拡張編)の[4章]を参照。

## 4.3 作成したフラクタルの保存と外部公開

作成されたフラクタル画像は24ビットのビットマップ形式の画像として着色表示されますので、これを標準ビットマップ画像として外部出力可能であります。更に汎用の画像変換ソフトによりJpeg等に任意に変換可能であります。

## 4.4 作成したフラクタルの登録と再生

- (1) 作成したフラクタルの生成パラメータを.csvファイル(DBファイルと呼びます)に登録して、このファイルから次々と再生・表示させることが出来ます。登録内容は、その時画面で表示中の画像生成に必要なパラメータが登録されます。具体的には(6章)を参照下さい。
- (2) 一つのDBファイルには任意種類、任意サイズのフラクタルを最大50個登録できます。
- (3) DB再生時には最初の画像から再生されます。再生時には下記が可能です。
  - i 再生・表示は下記で実施できます。
    - ・主画面 任意サイズの画面に表示できる：最終確認用
    - ・DB一覧画面 全登録画像を一覧表示可能：初期選択用
  - ii DB登録時の状態を基本に更に拡大・縮小/レベル変換を継続することが出来ます。
  - iii DB再生画像に対して更に下記の操作が行えます。詳細は(6章)を参照下さい。
    - ・新たに表示した画像を再生中DBファイルの任意位置に追加又は表示している元画像と置き換える。又は削除する。
    - ・表示対象の画像位置を切り替える。
    - ・再生中DBファイルの任意の部分了他又は表示中DBファイルに移動/コピーする。
    - ・表示中DBファイル自身を削除する。

## 4.5 出力された.csvファイルに関する制限事項

処理画像に関するコード情報は、汎用性を考慮してエクセルの.csvファイルで保存されますが、人手で修正すると、エクセルでの表示には異常はなくとも、後のプログラム処理で正常には読み込めなくなる可能性が有ります

人手で修正すると、プログラムで出力された管理情報が一部破損される為でしょうか？。

- i DB登録したFrDBファイル(6.2), FrATファイル(6.2.4)は本システムで読み込み参照するので人手で修正することは避ける必要があります。
- ii もし、何らかの手を加えた場合は、別名のファイルとして保存します。

## 5 参考 : Motifの詳細

図3.1.8.3で記した各Motifの各点kの座標(Xk, Yk)を以下に纏めます。

表5.1 Motifの詳細データ : 各点の座標(Xn, Yn)

NO	x/y	0	1	2	3	4	5	6	7	8	式	$r=\pi*\theta$	$\theta$	L	Dim
0	x y		0.5-a y	0.5+a -y	1 0						K=L $c=(2-K*\cos(r))/K/\sin(r)$ $\tan(a1)=c$				1.365
1	x y		x y	0.5 0	1-x -y	1 0					K=2*L $b=\cos(a1)/K/\sin(r)$ $x=b*\sin(a1)$		0.5	1	1.333
2	x y		x y	5-a/3 y/3	5+a/3 -y/3	1-x -y	1 0				K=3*L $y=b*\cos(a1)$ $a=K/2*b*\sin(r-a1)$				1.255
3	x y		L*b 0	0.5 y	1-L*b 0	1 0					$b=0.5/(L+\sin(r/2))$ $y=b*\cos(r/2)$		1/3	1	1.262
4	x y		b 0	0.5 y	1-b 0	1 0					$b=0.5/(1+L*\sin(r/2))$ $y=L*b*\cos(r/2)$				1.154
5	x y	0	0.5 0	0.5-x y	0.5 0	0.5+x y	0.5 0	1 0			$x=0.5*L*\sin(r/2)$ $y=0.5*L*\cos(r/2)$		0.11/4		1.292
6	x y		0.5 0	0.5-x y	0.5+x y	0.5 0	1 0						1/3	1	2.322
7	x y		0.5 y	0.5 y+L*b	0.5 y	1 0					$b=0.5/\sin(r/2)$ $y=b*\cos(r/2)$		2/3	1	2.524
8	x y		b 0	b+x y	1-b-x y	1-b 0	1 0				K=L $a1=\pi-r$ $b=1/(2+K+2*K*\cos(a1))$				1.465
9	x y		b 0	b+x y	0.5-x y	0.5 0	0.5+x -y	1-b-x -y	1-b 0	1 0	K=2*L $x=L*b*\cos(a1)$ $y=L*b*\sin(a1)$		0.5	1	1.500

- ・(0,0)が始点、終点を(1,0)として正規化してあります。
- ・Dim. はフラクタル次元で蛇行のレベルを表し、表記載の $\theta$ , Lに対して下記で計算してあります。

$$Dim = \frac{\log(N)}{\log(1/l)} \quad \left( = \frac{\log(4)}{\log(3)} = 1.262 \quad \text{No3の例} \right)$$

ここで、

$N$  = 有効 $L=0$ でない辺の数 (=4)

$l$  = 有効な辺の平均縮小率 (=1/3)

$l \geq 1$ の場合は $Dim \leq 0$ になります

## 6 参考文献

- [1] Hans Lauwerier著、西川 利男 訳  
「初めてのフラクタル」丸善株式会社 (平成8年7月25日発行)
- [2] B.Mandelbrot著 広中平祐 監訳  
「フラクタル幾何学」上、下 ちくま学芸文庫(2011 2月10日発行)