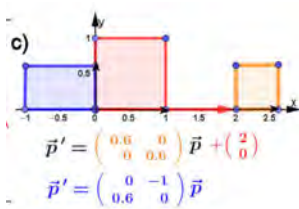


■ IFS-Fraktale



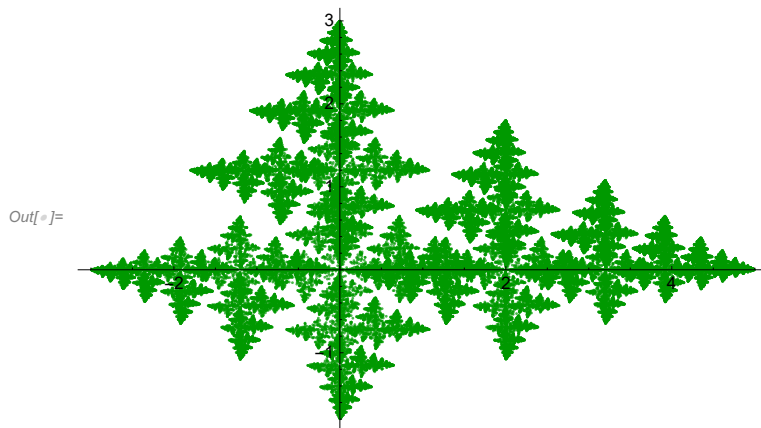
Buch: Höhere Mathematik sehen und verstehen, Haftendorn, Riebesehl, Dammer,
(auch in Mathematik sehen und verstehen, Haftendorn) Springer, Feb. 2021, (Feb. 2019)
Datei [wald-am-See.nb](#) zu Abschnitt 1.3.3.1 Seite 16, Abb. 1.9 und 1.10

■ Wald am See



- Es soll jedesmal eine beliebige Abb. ausgewählt werden

```
In[ ]:= wald = NestList[
  [Liste verschachtelter Ergebnisse
  ({{0.6 #[[1]] + 2, 0.6 #[[2]]}, {-#[[2]], 0.6 #[[1]]}][[RandomInteger[1] + 1] &],
  {0, 0}, 100000];
waldbild =
Graphics[{RGBColor[0, 0.6, 0, 0.7], PointSize[0.004], Point[wald]}, Axes -> True]
```



■ Folge der Bilder

```
In[ ]:= Quit
[beende Kernel]
```

```
In[ ]:= h = 2; q = 0.6; A = ( q 0; 0 q ); G = ( 0 -1; q 0 );
```

```
In[ ]:= {h, q, A, G}
```

```
Out[ ]:= {2, 0.6, {{0.6, 0}, {0, 0.6}}, {{0, -1}, {0.6, 0}}}
```

○ Versuche, als Bild wieder einen Punkt zu bekommen

○ jetzt von Hand

```
In[ ]:= f[x_, y_] := {2 + 0.6` x, 0.6` y}; f[x, y]
```

```
Out[ ]:= {2 + 0.6 x, 0.6 y}
```

```
In[ ]:= g[x_, y_] := {-y, 0.6` x}; g[x, y]
```

```
Out[ ]:= {-y, 0.6 x}
```

```
In[ ]:= a = h *  $\frac{1}{1 - q}$ 
```

```
Out[ ]:= 5.
```

```
In[ ]:= tr = {{a, 0}, {0, q a}, {-q a, 0}, {0, -q^2 a}}
```

```
Out[ ]:= {{5., 0}, {0, 3.}, {-3., 0}, {0, -1.8}}
```

○ erste Abbildung

```
In[ ]:= trf = Map[f, tr, 1] // Evaluate
```

Wende an Werte aus

```
trg = Map[g, tr, 1] // Evaluate
```

Wende an Werte aus

```
Out[ ]:= {{5., 0.}, {2., 1.8}, {0.2, 0.}, {2., -1.08}}
```

```
Out[ ]:= {{0, 3.}, {-3., 0.}, {0, -1.8}, {1.8, 0.}}
```

```
In[ ]:= tr
```

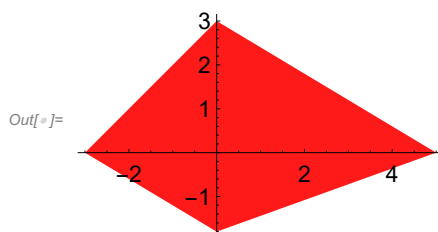
```
Out[ ]:= {{5., 0}, {0, 3.}, {-3., 0}, {0, -1.8}}
```

```
In[ ]:= null = Graphics[{RGBColor[1, 0, 0], Opacity[0.9], Polygon[tr]},
```

Graphik RGB Farbe Deckkraft Vieleck

```
Axis -> True, AxesStyle -> Directive[Black, 12]]
```

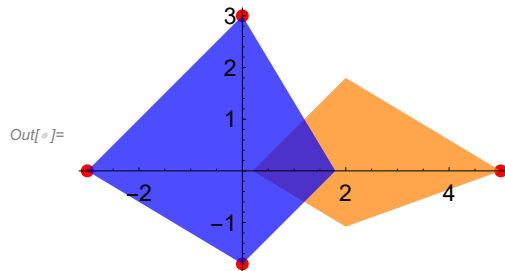
Achsen wahr Achsenstil Anweisung schwarz




```

In[ ]:= eins = Graphics[{RGBColor[1, 0, 0], Opacity[1], PointSize[0.03],
  [Graphik [RGB Farbe [Deckkraft [Punktgröße
    Point[tr], RGBColor[1, 0.5, 0], Opacity[0.7], Polygon[trf],
    [Punkt [RGB Farbe [Deckkraft [Vieleck
    RGBColor[0, 0, 1], Opacity[0.7], Polygon[trg]}], Axes -> True,
    [RGB Farbe [Deckkraft [Vieleck [Achsen [wahr
    AxesStyle -> Directive[Black, 12]]
    [Achsenstil [Anweisung [schwarz

```



○ zweite Abbildung

```

In[ ]:= trff = Map[f, Map[f, tr]]
  [wende... [wende an
trgf = Map[f, Map[g, tr]]
  [wende... [wende an
trfg = Map[g, Map[f, tr]]
  [wende... [wende an
trgg = Map[g, Map[g, tr]]
  [wende... [wende an

```

Out[]:= {{5., 0.}, {3.2, 1.08}, {2.12, 0.}, {3.2, -0.648}}

Out[]:= {{2., 1.8}, {0.2, 0.}, {2., -1.08}, {3.08, 0.}}

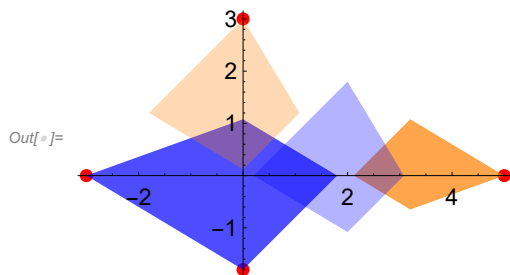
Out[]:= {{0., 3.}, {-1.8, 1.2}, {0., 0.12}, {1.08, 1.2}}

Out[]:= {{-3., 0.}, {0., -1.8}, {1.8, 0.}, {0., 1.08}}

```

zwei = Graphics[{ RGBColor[1, 0, 0], Opacity[1], PointSize[0.03],
  |Graphik |RGB Farbe |Deckkraft |Punktgröße
  Point[tr],
  |Punkt
  RGBColor[1, 0.5, 0], Opacity[0.7], Polygon[trff],
  |RGB Farbe |Deckkraft |Vieleck
  RGBColor[0, 0, 1], Opacity[0.7], Polygon[trgg],
  |RGB Farbe |Deckkraft |Vieleck
  RGBColor[1, 0.5, 0], Opacity[0.3], Polygon[trfg],
  |RGB Farbe |Deckkraft |Vieleck
  RGBColor[0, 0, 1], Opacity[0.3], Polygon[trgf] }
, Axes -> True, AxesStyle -> Directive[Black, 12]]
  |Achsen |wahr |Achsenstil |Anweisung |schwarz

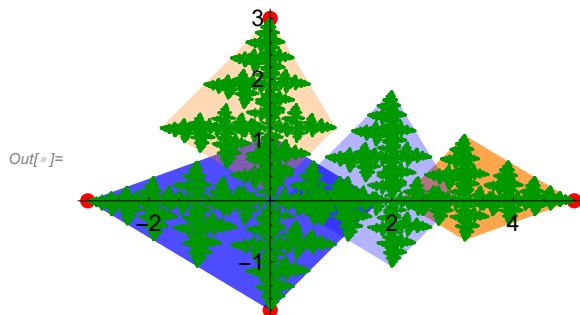
```



```

In[ ]:= Show[{zwei, waldbild}]
  |zeige an

```



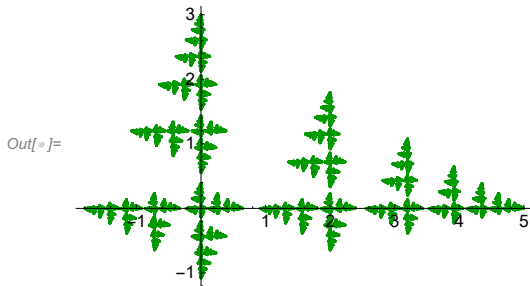
■ Für Dimensionsgedanken

Variante mit gleichem Stauchfaktor

```

In[ ]:= waldd1 = NestList[({{0.6#[[1]] + 2, 0.6#[[2]]}, {-0.6#[[2]], 0.6#[[1]]}})[[
  Liste verschachtelter Ergebnisse
  RandomInteger[1] + 1]] &),
  ganze Zufallszahl
  {0, 0}, 100000];
waldbild =
  Graphics[{RGBColor[0, 0.6, 0, 0.7], PointSize[0.004], Point[wald]}, Axes -> True]
  Graphik RGB Farbe Punktgröße Punkt Achsen wahr

```



○ Dimension z Bausteine k zentrischer Streckfaktor

Zwei gleiche Bausteine, jeder wird mit zentr. Streckfaktor $\frac{5}{3}$ vergrößert.

```

In[ ]:= d = Log[2]
          Log[5/3]

```

```

Out[ ]:= Log[2]
          Log[5/3]

```

```

In[ ]:= % // N
          numerischer Wert

```

```

Out[ ]:= 1.35692

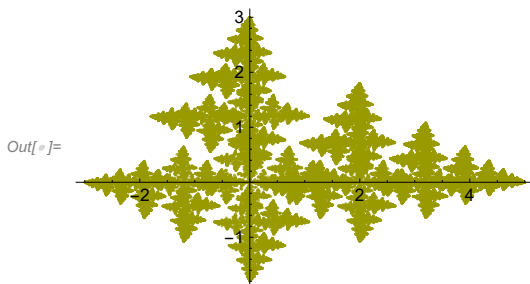
```

○ Dimension beide nur Achsenstreckung

```

In[ ]:= waldd2 = NestList[
  Liste verschachtelter Ergebnisse
  ({{0.6#[[1]] + 2, 1#[[2]]}, {-#[[2]], 0.6#[[1]]}})[[RandomInteger[1] + 1]] &),
  ganze Zufallszahl
  {0, 0}, 100000];
waldbild =
  Graphics[{RGBColor[0.6, 0.6, 0, 0.7], PointSize[0.004], Point[wald]}, Axes -> True]
  Graphik RGB Farbe Punktgröße Punkt Achsen wahr

```



Beide haben eine Achsenstreckung von $\frac{5}{3}$, aber es sind nicht, wie

bei der Definition gefordert, $z=2$ GLEICHE Bausteine.

Darum lässt sich die fraktale Dimension so einfach nicht bestimmen.