Rekurzivně definované křivky

I.Kolingerová

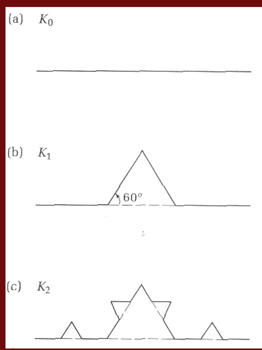
1. Kochova křivka
2. C-křivky a dračí křivky
3. Křivky plnící prostor
4. Plazi

Literatura

- N. Wirth: Algoritmy a štruktúry údajov (Algorithms + Data Structures = Programs), Alfa, Bratislava 1988
- Francis S.Hill Jr.: Computer Graphics, Macmillan Publishing Company, New York, 1990

1. Kochova křivka

- 1904, Helge von Koch
- nekonečně dlouhá čára na konečné ploše
- postupné generace

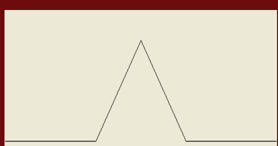


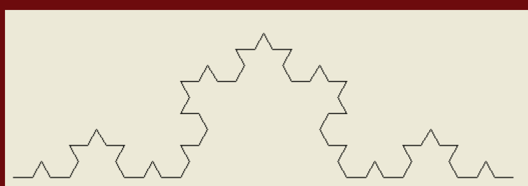
- K2 je 4/3 x delší než K1
- Ki+1 náhradou každého ze 4 segmentů "špičkou"
- celková délka $(4/3)^i$, pro i ->∞ délka ->∞ , přesto křivka zůstává uzavřená v konečné oblasti

3

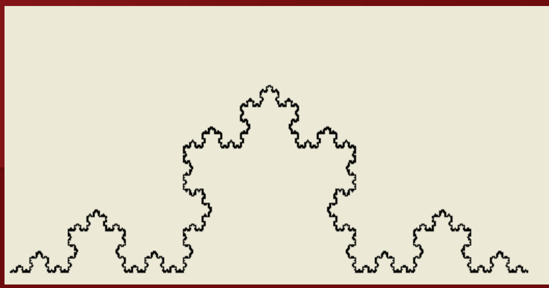
Algoritmus výpočtu Kochovy křivky

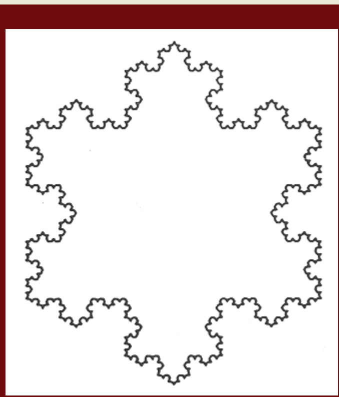
- rekurzivní definice, vstupem řád n, počáteč. délka segmentu l, počáteční bod, směr dir
- takto náhrada libovolné úsečky, ->Kochova krajina, rovnostranný trojúh. -> Kochova vločka





4





5

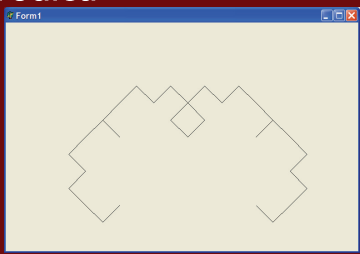
```
procedure Koch (dir,len:real;n:integer);
{ délka úsečky len ve směru dir, řád n, počátek v (xp,yp) }
const rads=0.017453293;
begin
 if n > 0 then
   begin
     Koch (dir,len/3,n-1); dir := dir+60;
     Koch (dir,len/3,n-1); dir := dir-120;
     Koch (dir,len/3,n-1); dir := dir+60;
     Koch (dir,len/3,n-1);
   end
 else
   begin
    LineTo (xp+len*cos(rads*dir),yp+len*sin(rads*dir));
    xp := xp + len*cos(rads*dir); yp := yp + len*sin(rads*dir);
   end;
end;
```

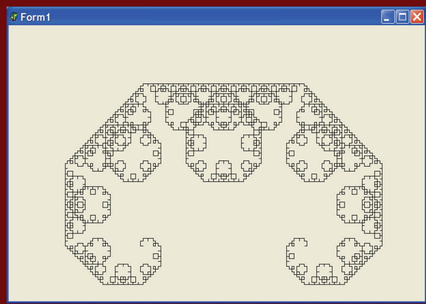
6

2. C-křivky, dračí křivky

- náhrada úsečky C-čka délky 1/√2xdélka rodičů

n=5

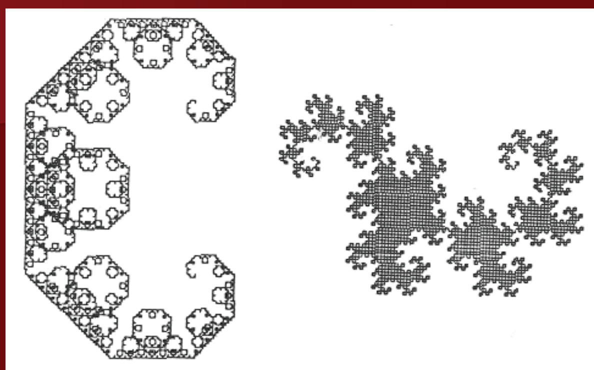




n=12

- dračí křivka – podobné, ale směr spojky odlišný od segmentu k segmentu (1 segment nahrazen levým obratem, 1 pravým)

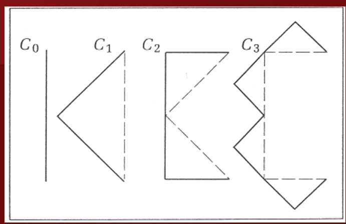
7



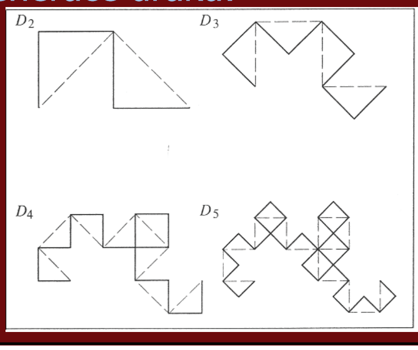
C-křivka a dračí křivka řádu 12

8

Následující generace C-křivek:



Následující generace draků:



9

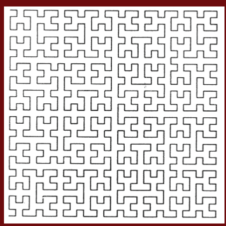
```
procedure DrawC (dir,len:real;n:integer);
{ délka úsečky len ve směru dir, řád n, počátek v (xp,yp) }
const fct=0.7071067; {1/sqrt(2) } rads=0.017453293;
begin
 if n > 0 then
   begin
     dir  := dir + 45; DrawC (dir,len*fct,n-1);
     dir  := dir - 90;  DrawC (dir,len*fct,n-1);
   end
 else
   begin
    LineTo (xp+len*cos(rads*dir),yp+len*sin(rads*dir));
    xp := xp + len*cos(rads*dir); yp := yp + len*sin(rads*dir);
   end;
end;
```

10

3. Křivky plnící prostor

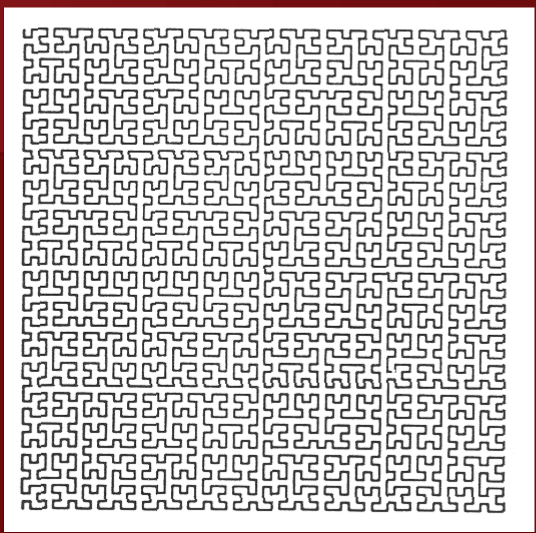
a) Hilbertovy křivky

- pojmenovány po matematikovi D.Hilbertovi (1862-1943)
- každá H. křivka má svůj řád a jednu ze 4 orientací A, B, C, D
- ty se skládají dohromady s pomocnými spojkami na křivky 2. řádu
- A2 ~ D left A down A right B
- B2 ~ C up B right B down A
- takto i pro vyšší řády,
  - např. B9 ~ C8 up B8 right B8 down A8



n=5

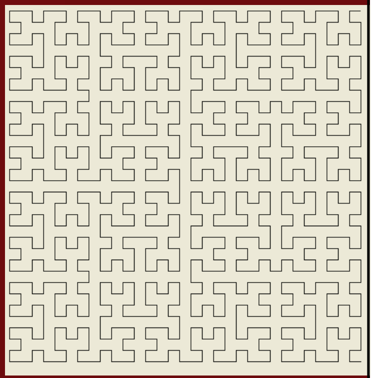
11



Hilbertova křivka řádu 6

12

- Def. tabulka Hilbertovy křivky:
  - A: DAAB A: left down right
  - B: CBBA B: up right down
  - C: BCCD C: right up left
  - D: ADDC D: down left up



- patří mezi křivky plnící prostor – čím vyšší řád, tím více prostor zaplněn

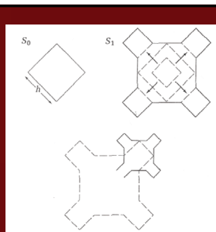
13

```
procedure TForm1.A (i : integer);
begin
 if i > 0 then
   begin
    D(i-1); x := x - h; MyLineTo (x,y);
    A(i-1); y := y - h; MyLineTo (x,y);
    A(i-1); x := x + h; MyLineTo (x,y);
    B(i-1)
   end
end;
procedure TForm1.B (i : integer);
begin
 if i > 0 then
   begin
    C(i-1); y := y + h; MyLineTo (x,y);
    B(i-1); x := x + h; MyLineTo (x,y);
    B(i-1); y := y - h; MyLineTo (x,y);
    A(i-1)
   end
end;
```

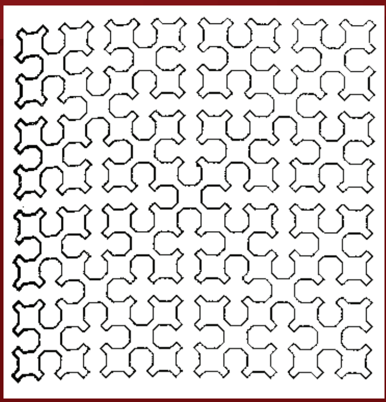
14

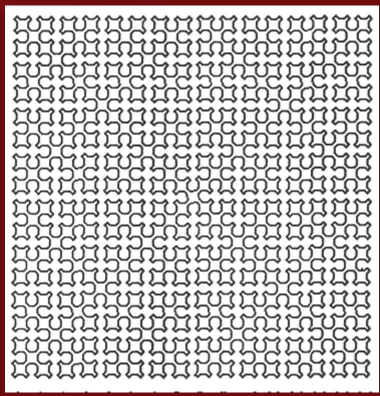
```
procedure TForm1.C (i : integer);
begin
 if i > 0 then
   begin
    B(i-1); x := x + h; MyLineTo (x,y);
    C(i-1); y := y + h; MyLineTo (x,y);
    C(i-1); x := x - h; MyLineTo (x,y);
    D(i-1)
   end
end;
procedure TForm1.D (i : integer);
begin
 if i > 0 then
   begin
    A(i-1); y := y - h; MyLineTo (x,y);
    D(i-1); x := x - h; MyLineTo (x,y);
    D(i-1); y := y + h; MyLineTo (x,y);
    C(i-1)
   end
end;
```


b) Sierpinského křivky



- další křivky plnící prostor
- def. jako posloupnost operací Replikuj a Spoj
- S0 = čtverec o straně h stojící na rohu, S1 vznikne jeho náhradou 4 polovičními replikami, posunutými ze středu o h, spojenými vodorovnými a svislými čarami, a smazáním 4 vnitřních stran
- další generace: náhrada čtverce 4 polovičními replikami odsunutými od středu o vhodnou vzdálenost, kresba diagonál, výmaz vniřních stran





Sierpinského křivka S4 (vlevo) a S5 (vpravo)

17

- tento přístup snadný, ale nutné mazání
- raději:

S: A ↘ B ↙ C ↖ D ↗

- otevřené obrazce spojovat čarami, které nejsou částí rekurzivního obrazce – zařadíme je do S0

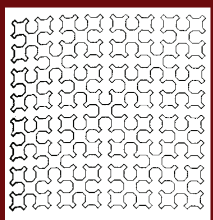
- Rekurz. vzorce:

  A: A ↘ B → D ↗ A

  B: B ↙ C ↓ A ↘ B

  C: C ↖ D ← B ↙ C

  D: D ↗ A ↑ C ↖ D



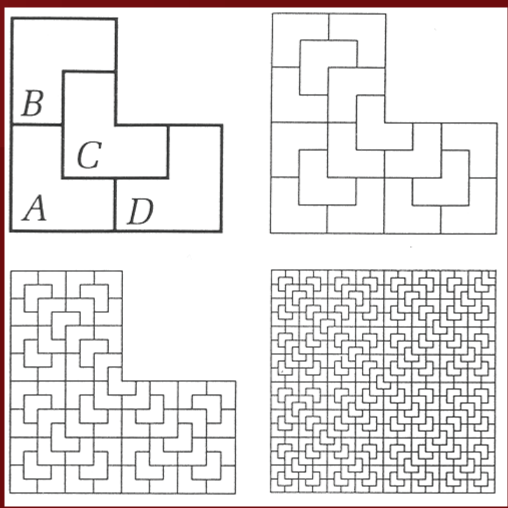
Program: zkuste sami

18

4. Plazi (reptiles)

- neperiodické dlaždicové vzory, nejsnažší popis rekurzivní
- různé repliky plazů tvoří dohromady velkého plaza stejného tvaru
- plaz def. rekurzivně

Př.: triomino



- 4 orientace téhož
- každé triomino jde nahradit touto čtveřicí, ta jde zase nahradit ... počet zjemnění ~ řádu křivky
- různé orientace => neperiodické

Fraktály

I.Kolingerová

1. Soběpodobnost a fraktální dimenze
2. Druhy fraktálů a jejich výpočet

Literatura

- Francis S.Hill Jr.: Computer Graphics, Macmillan Publishing Company, New York, 1990
- J.Vince: 3-D Computer Animation, Addison-Wesley Publishing Company, 1992
- S.C. Hoggar: Mathematics for Computer Graphics, Cambridge University Press, 1992
- H.A. Lauwerier, J.A. Kaandrop: Fractals (Mathematics, Programming and Applications), TR CS-R8762, Centre for Mathematics and Computer Science, Amsterdam, The Nietherlands, 1980

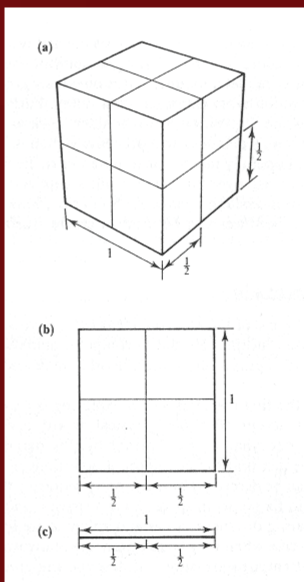
22

1. Soběpodobnost a fraktální dimenze

- soběpodobný tvar: obecná úroveň detailu stejná bez ohledu na vzdálenost, z jaké se na tvar díváme
- např. "hrubost" Kochovy křivky Kn pro n->∞ stále stejná (obdobně Hilbertova křivka nebo triomino)
- ∞ v CG sice jen aproximováno, ale efekt OK
- v přírodě příklady soběpodobnosti: pobřeží, skály, oblaka, list, krevní soustava ...
- výzkumy: B. Mandelbrot – soběpodobné křivky – fraktály (<= fraktální dimenze)

23

- běžná – topologická dimenze:
  - bod 0 D
  - přímka 1D
  - rovina 2D ...
- křivka ∞ délky uzavřená v konečné oblasti roviny – - něco mezi 1 a 2
- fraktální dimenze



D-dimenze
N-počet soběpodob. částí
S-scaling factor

a) D=3,N=8,S=1/2
b) D=2,N=4,S=1/2
c) D=1,N=2,S=1/2

24

- vztah mezi D,N,S: $N=1/S^D$ => $D=\log(N)/\log(1/S)$
- Kochova křivka: S=1/3,N=4 => $D=\log(4)/\log(3)=1.2619..$
- tato dimenze nemá původní známý význam => fraktální dimenze

25

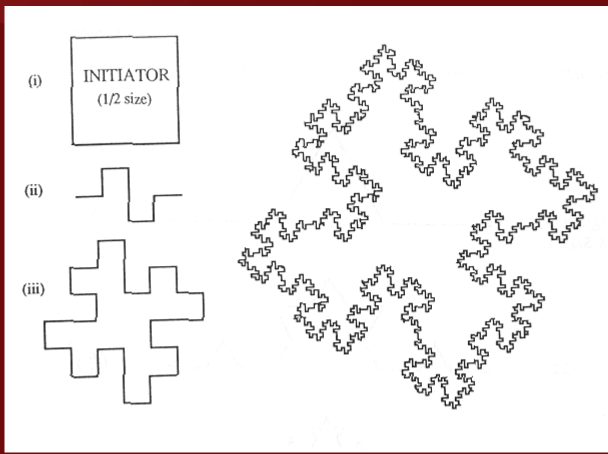
2. Druhy fraktálů a jejich výpočet

Druhy fraktálů

- geometrické – generované geometr. vzorky – iniciátor, generátor; úsečky iniciátoru nahrazovány generátorem
- stochastické – generované náhodným procesem – pobřeží, terény
- algebraické – generované iteracemi algebr. transformačních funkcí, např. $z \Leftarrow z^2 + c$

26

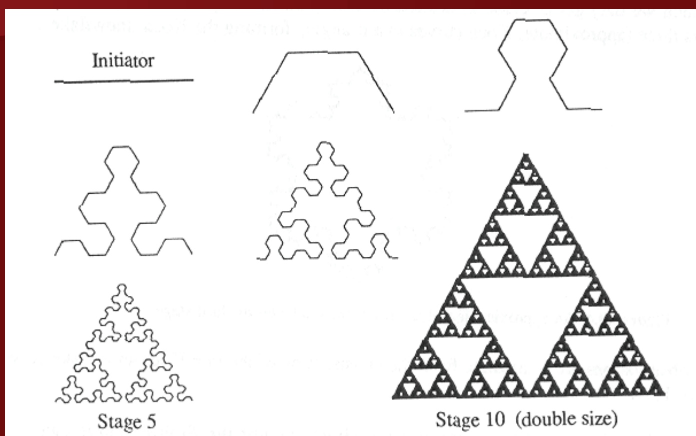
Příklad geometrického fraktálu: Kochův ostrov



Kochův ostrov
(8.generace),
(i) iniciátor
(ii) generátor
(iii) 1. generace
- (i) až (iii) v poloviční velikosti

27

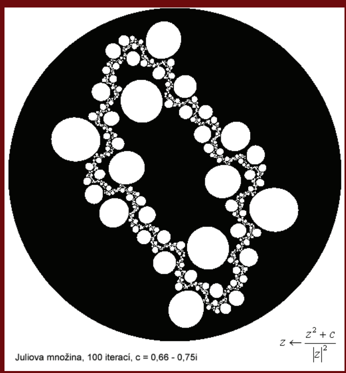
Příklad geometrického fraktálu: Sierpinského křivka

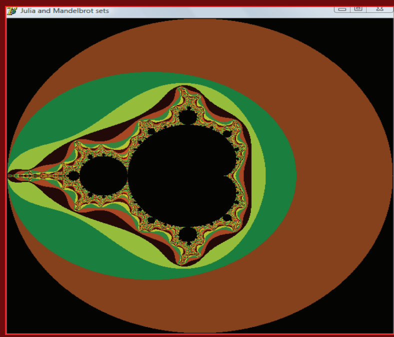


Další geom.fraktály: viz rekurzivně def. křivky

28

Příklad algebraického fraktálu: Mandelbrotova množina, Juliova množina

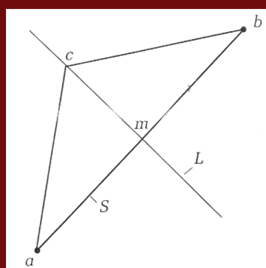




29

Stochastické fraktály

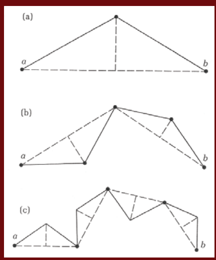
- Nejjednodušší fraktál – fraktalizace úsečky – nahradit v každém kroku úsečku náhodnou špičkou
- Např. S nahrazeno 2 úsečkami,
  c náhodně zvolený bod
  na kolmém bisektoru L k S
  m=(mx,my) – střední bod
  úsečky



- Vzdálenost mezi c,m ~ t, délce S, t kladné nebo záporné

30

- t většinou modelováno jako gaussovská náhodná proměnná (v Delphi RandG())
- rekurzivní generace,
  až splněno nějaké kritérium



Řízení perzistence fraktální křivky

- řízení změny měřítka – jak "zubatá" křivka je
- f = 2 $^{(1/2\text{-}H)}$, StDev := StDev*f, t:=Gauss*StDev
- H mezi 0 a 1, nad 1/2 – hladší, více perzistentní křivky, pod 1/2 – víc zubaté
- Př.:

H	f
0.0	1.4
0.3	1.1
0.5	1.0
0.7	0.9
1.0	0.7

f=1 při H=1/2 - standardní odchylka na všech úrovních stejná – model Brownova pohybu

f<1 při H>1/2 – standardní odchylka klesá s růstem úrovně

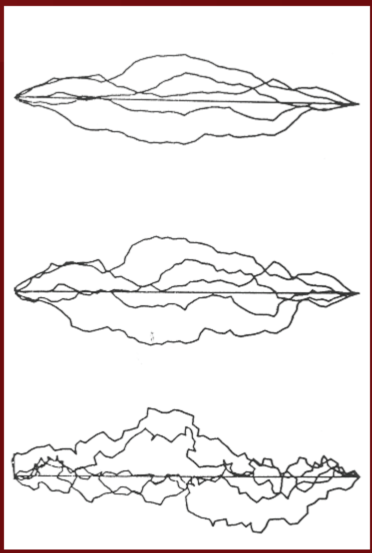
f > 1 při H < 1/2 – standardní odchylka roste s růstem úrovně – "antiperzistentní křivka"

Příklady:

H=0.7

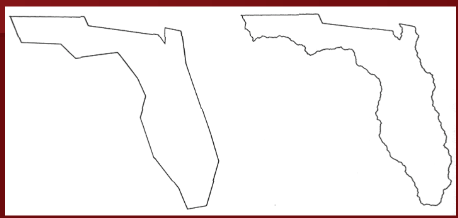
H=0.5

H=0.3



33

- lze fraktalizovat lib. čáru



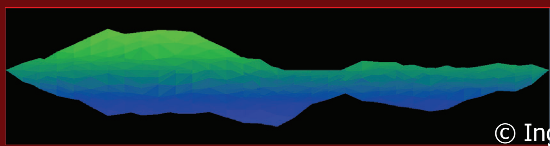
realistický model pobřeží

- sekvence jsou opakovatelné, pokud stejný RandSeed
- stačí uchovat
  - vrcholy původních úseček
  - minim. délku čáry či max. hloubku dělení a standard. odchylku
  - randseed

34

Fraktální plochy

- pro realistický vzhled horských terénů
- začátek – trojúhelník v rovině, střední body hran zvedat vertikálně o náhodnou vzdálenost ve svislém směru
- vše opakovat pro vzniklé 4 trojúhelníkové stěny, až jsou plochy "dosti malé"
- na styku trojúhelníků možnost vzniku děr



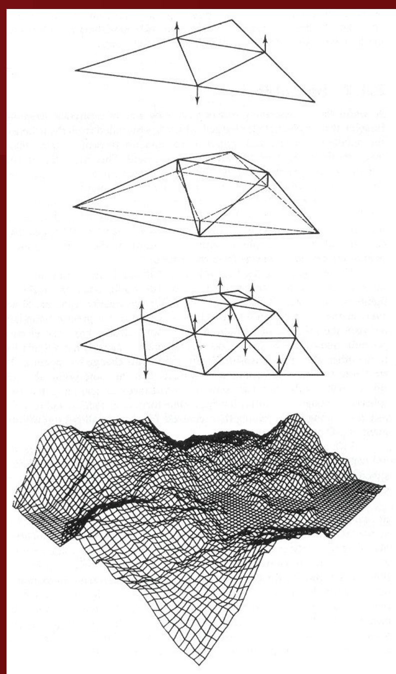
© Ing. P. Kratochvíl

35

Fraktální plochy

technika opak. dělení trojúhelníků
s náhod. posouváním bodů

fraktálové dělení ve stř. bodě,
část výšek pevná –
- simulace mořského pobřeží



36

Výpočet fraktálů:

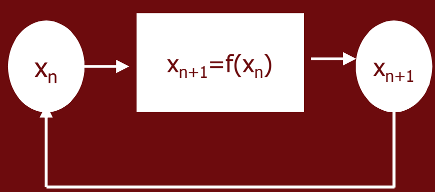
- rekurzivní (geometrická konstrukce)
- iterované funkční systémy (IFS)
- formální jazyky (L-systémy)
- nelineární komplexní mapy
- chaos game (podobné IFS)

- mezi metodami velké odlišnosti, ale některé fraktální objekty mohou vzniknout více cestami (např. Sierpinského trojúhelník)

37

Výpočet fraktálů:

- princip zpětné vazby – výstup 1 iterace vstupem příští



- relace lineární nebo nelineární
- výstup ne nutně fraktál. dimenze – též "normální" objekt, body v ∞ nebo 1 bod atrakce

38