

L-systémy v prostředí Mathematica

L-systems in the Mathematica environment

Hynek Petrla

Bakalářská práce
2013



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

akademický rok: 2012/2013

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: Hynek PETRLA

Osobní číslo: A10148

Studijní program: B3902 Inženýrská informatika

Studijní obor: Informační a řídicí technologie

Forma studia: prezenční

Téma práce: L-systémy v prostředí Mathematica

Zásady pro vypracování:

- 1. Seznamte se s fraktály a L-systémy a principy jejich konstrukce.**
- 2. Vyberte vhodné příklady pro vizualizaci L-systémů nejen z oblasti známých fraktálů.**
- 3. U vybraných příkladů popište jednotlivé kroky prepisovacích pravidel.**
- 4. Vybrané L-systémy naprogramujte v prostředí Mathematica.**
- 5. Naprogramované příklady vizualizujte pomocí funkce Manipulate či obdobných příkazů.**

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. PRUSINKIEWICZ, Przemyslaw a Aristid LINDENMAYER. The algorithmic beauty of plants. New York: Springer-Verlag, 1990, xxiv, 521 s. ISBN 35-409-7297-8.
2. ZELINKA, Ivan, František VČELAŘ a Marek ČANDÍK. Fraktální geometrie: principy a aplikace. 1. vyd. Praha: BEN – technická literatura, 2006, 159 s. ISBN 80-7300-191-8.
3. MĚCH, R., Przemyslaw PRUSINKIEWICZ a J. HANAN. Extensions to the graphical interpretation of L-systems based on turtle geometry. Dostupné online <http://algorithmicbotany.org/lstudio/graph.pdf>.
4. PRUSINKIEWICZ, Przemyslaw a James HANAN. Lindenmayer Systems, Fractal, and Plants. New York: Springer-Verlag, 1989, xxiv, 521 s. ISBN 978-0387970929.
5. WEISSTEIN, Eric W. Lindenmayer System. From MathWorld-A Wolfram Web Resource. Dostupné online <http://mathworld.wolfram.com/LindenmayerSystem.html>

Vedoucí bakalářské práce:

Ing. Zuzana Komínková Oplatková, Ph.D.
Ústav informatiky a umělé inteligence

Datum zadání bakalářské práce:

24. února 2013

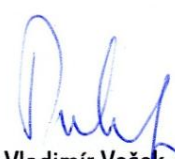
Termín odevzdání bakalářské práce:

14. června 2013

Ve Zlíně dne 24. února 2013


prof. Ing. Vladimír Vašek, CSc.
děkan




prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

ABSTRAKT

Tato práce se zabývá vysvětlením základními pojmů z fraktální geometrie, detailně se dále věnuje problematice L-systémů. V první části práce je uvedena historie, základní pojmy související s danou problematikou a obecný popis fraktální geometrie. V další části jsou podrobně popsány jednotlivé typy L-systémů a princip reprezentace pomocí želví grafiky. V poslední části jsou popsány principy a algoritmy jednotlivých L-systémů. Tyto L-systémy jsou naprogramovány a vizualizovány v programu Mathematica s možnostmi interaktivního ovládání.

Klíčová slova: L-systémy, Fraktály, Lindenmayer, Stromy, Mathematica, Přepisovací gramatiky

ABSTRACT

This thesis explains the basic terms of fractal geometry, specifically it deals with the topic of L-system. First part includes history, basic elements of the system and general description of fractal geometry. Second part contains detailed description of individual L-systems and graphical representation using „turtle graphic“. Final part describes principle and algorithmus of individual L-systems. These L-system are programed and visualised in software Mathematica with possibilities of interactive manipulation.

Keywords:: L-system, Fractal, Lindenmayer, Tree, Mathematica, Wolfram, Rewriting rules

Tímto bych chtěl poděkovat své vedoucí bakalářské práce Ing. Zuzaně Komínkové Oplatkové, Ph.D. za velmi zajímavé téma a odborné vedení, které mi poskytla.

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....
podpis diplomanta

OBSAH

ÚVOD.....	9
I TEORETICKÁ ČÁST.....	10
1 FRAKTÁLY.....	11
1.1 HISTORIE.....	11
1.1.1 Georg Ferdinand Ludwig Philipp Cantor.....	11
1.1.2 Niels Fabian Helge von Koch (1870 – 1924).....	12
1.1.3 Aristid Lindenmayer	13
1.1.4 Benoit B. Mandelbrot.....	13
1.2 FRAKTÁLNÍ GEOMETRIE	14
1.2.1 L-systémy	15
1.2.2 Systémy iterovaných funkcí – IFS	15
1.2.3 Time escape algorithm – TEA	15
1.2.4 Dynamické systémy	16
1.2.5 Náhodné fraktály	16
1.3 DIMENZE	17
1.3.1 Euklidovská dimenze	17
1.3.2 Topologická dimenze	17
1.3.3 Fraktální dimenze.....	17
2 L-SYSTÉMY.....	20
2.1 DETERMINISTICKÉ A KONTEXTOVĚ NEZÁVISLÉ L-SYSTÉMY (DOL)	20
2.1.1 Příklad L-systému:	21
2.2 INTERPRETACE SYMBOLŮ	21
2.2.1 Interpretace 2D.....	22
2.2.2 Interpretace ve 3D	23
2.3 ROZVĚTVUJÍCÍ SE STRUKTURY.....	25
2.3.1 Osové stromy	25
2.3.2 Závorkové l-systémy	26
2.4 STOCHASTICKÉ L-SYSTÉMY	27
2.5 KONTEXTOVÉ L-SYSTÉMY	29
2.5.1 1L-systémy	29
2.5.2 2L-systémy	29
2.6 PARAMETRICKÉ L-SYSTÉMY	30
2.6.1 OL-systémy	30
2.6.2 2L-systémy	31
2.7 GENEROVÁNÍ POLYGONŮ A PLOCH	31
2.7.1 Předem definované plochy	31
2.7.2 Vyvíjející se modely	31
2.7.3 Tropismus.....	32
2.8 VYUŽITÍ L-SYSTÉMŮ V PRAXI	32
II PRAKTICKÁ ČÁST	34
3 PROGRAMOVÁ ČÁST	35

3.1	WOLFRAM MATHEMATICA.....	35
3.2	POPIS GUI PROSTŘEDÍ.....	36
3.2.1	Popis ovládacích prvků	37
3.2.2	Abeceda gramatiky.....	37
4	KONSTRUKCE L-SYSTÉMŮ	39
4.1	2D SYSTÉMY	39
4.1.1	Kochova vložka.....	40
4.1.2	Sierpinského trojúhelník	41
4.1.3	32- Segmentová křivka	42
4.1.4	Gosperova křivka	43
4.1.5	Gosperův ostrov	44
4.1.6	Sierpinského křivka.....	45
4.1.7	Hilbertova křivka	46
4.1.8	Moorova křivka.....	46
4.1.9	Dračí křivka.....	47
4.1.10	Box Fraktál.....	48
4.1.11	Stromy.....	50
4.2	3D SYSTÉMY	52
4.2.1	Mengerova houba.....	52
4.2.2	Vicsek fraktál	54
4.2.3	Stromy.....	55
4.2.4	Hilbertova křivka	56
	ZÁVĚR	57
	SEZNAM POUŽITÉ LITERATURY.....	58
	SEZNAM OBRÁZKŮ	61
	SEZNAM TABULEK.....	63
	SEZNAM PŘÍLOH.....	64

ÚVOD

Mnoho přírodních útvarů kolem nás je fraktálního charakteru. Věda zabývající se těmito fraktály se nazývá fraktální geometrie, s její pomocí můžeme nejen popsat složité struktury, ale tyto struktury také jednoduše modelovat. V polovině 20. století byla definována varianta formálních gramatik jako nástroj pro simulaci a modelování rostlin s názvem L-systémy. Základní podstatou těchto systémů je přepisování řetězce reprezentující určitý tvar podle daných pravidel. Podle účelu je poté výsledný řetězec reprezentován. Jedna ze základních reprezentací je pomocí želví grafiky.

Teoretická část bakalářské práce se zaměřuje na popis L-systémů od jejich vzniku až po současné využití. Podstatná část je zaměřena na detailní popis různých typů L-systémů, které jsou doplněny o příklady.

V praktické části je vytvořena grafická interpretace, spolu s popisem kroků jednotlivých L-systémů. Za pomoci programu Wolfram Mathematica jsou vytvořeny dynamické příklady s množstvím nastavením a velkou variabilitou.

I. TEORETICKÁ ČÁST

1 FRAKTÁLY

1.1 Historie

Fraktální geometrie je intenzivně rozvíjena zhruba od šedesátých let 20. století, i když určitá setkání s touto problematikou probíhala již dříve. Dnes je fraktální geometrie uznávanou vědní disciplínou.

Koncem devatenáctého století se začaly objevovat konstrukce podivných útvarů, jako Cantorovo diskontinuum, Peanova křivka, Ďábelské schodiště a další. Jednou z prvních konstrukcí byly funkce, které jsou spojité, ale nejsou diferencovatelné. Takovou funkci (1) představil Karl Weierstrass roku 1872 na své přednášce v Královské akademii věd v Berlíně. Funkce (1) je spojitá na svém oboru, ale v žádném bodě nemá derivaci [8, str. 11].

$$\sum_{n=0}^{\infty} a^n \cos(\pi b^n x), 0 < a < 1, ab > 1 + \frac{3\pi}{2} \quad (1)$$

Za oficiálního zakladatele je považován vědec polského původu a objevitel fraktálů Benoit B. Mandelbrot, který jako první matematicky definoval pojem fraktál (fractal). Ukázal, že dokáže velmi dobře popisovat jak geometrický vzhled našeho světa, tak chování dynamických systémů. Před založením fraktální geometrie se ve světě používala klasická euklidovská geometrie, která byla s úspěchem používána po celá staletí. Její velkou slabinou bylo to, že neuměla popsat jednoduchým způsobem komplikované struktury [9, str. 15].

Ke vzniku tohoto odvětví ho inspirovaly dvě setkání s geometrickou soběpodobností - při studiu poruch v přenosu telekomunikačních signálů a při studiu fluktuací (nahodilý pohyb) tržních cen, kdy si všiml jisté formy pravidelnosti. Velký vliv na vytvoření fraktální geometrie měl i L. F. Richardson, který nasbíral velké množství dat, které později B. B. Mandelbrot využil [9, str. 15].

Fraktální geometrie umožňuje jednodušeji popsat složité struktury, objekty každodenního světa, nebo matematické vyjádření objektů jako Pythagorův strom, Siernpinského trojúhelník, atd.

1.1.1 Georg Ferdinand Ludwig Philipp Cantor

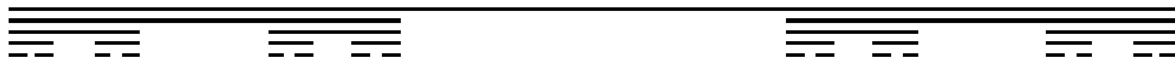
G. Cantor (1845-1918) byl významný německý matematik a logik. Kromě matematiky se, především v pozdějším věku, velmi věnoval teologii, zejména ve vztahu k vlastní práci

týkající se nekonečna. Je znám především tím, že teorii množin rozšířil o nekonečná čísla, označovaná jako ordinální a kardinální čísla.

V roce 1860 Georg Cantor absolvoval gymnázium v Darmstadtu se zálibou v matematice a trigonometrii. Poté odešel studovat do Zurichu, kde začal studovat matematiku. V roce 1863 Georg přesouvá svá studia na univerzitu v Berlíně. V roce 1866 studoval v Göttingenu, pozdějším velmi známém centru matematické vědy. O rok později získal doktorát za práci „De aequationibus secundi gradus indeterminatis“, která se zabývala teorií čísel.

V roce 1872 byl Cantor jmenován mimořádným profesorem na univerzitě v Halle a o sedm let později (ve svých 34 letech) řádným profesorem [10].

Roku 1884 přišel G. Cantor s Cantorovým diskontinuem (Obr. 1) – množinu s nenulovou dimenzí menší než 1. Cantorovo diskontinuum vytvoříme ze základní úsečky o délce 1, kterou rozdělíme na tři stejně dlouhé části, prostřední díl vyjmeme a stejný postup opakujeme na zbývajících dvou úsečkách nekonečně mnohokrát [8, str. 12].

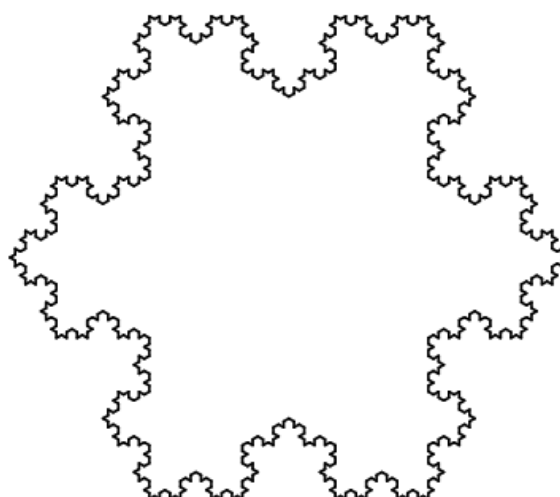


Obr. 1. Cantorovo diskontinuum

1.1.2 Niels Fabian Helge von Koch (1870 – 1924)

Byl to švédský matematik, roku 1892 získal na Stockholmské univerzitě doktorát z matematiky za dvě práce týkající se řešení systémů diferenciálních rovnic. V červenci 1911 získal na Stockholmské univerzitě post profesora matematiky.

Roku 1904 představil křivku (Obr. 2), která po něm nese jeho jméno. Při její konstrukci se již nepoužívá mateřské těleso a afinní transformace na něm, ale rekurzivní procedura. Kochovu křivku můžeme vytvořit L-systémem, nebo iteracemi IFS s jednotkovou úsečkou [9, str. 20].



Obr. 2. Kochova křivka, 4. Iterace.

1.1.3 Aristid Lindenmayer

A. Lindenmayer byl maďarský biolog, který rozvinul formální popis vývoje biologických systémů pro simulace na počítačích. Tento přístup je nyní znám jako "paralelní přepisující se systém" nebo také jako "L systém", který se používá pro modelování růstových procesů.

Lindenmayer zkoumal kvasnice, vláknité houby, různé druhy řas a studoval jejich růstové vzorce. Původně L-systémy vymyslel, aby ilustroval příbuznost mezi rostlinnými buňkami. Později systém rozšířil pro popis rostlin a složitého větvení [1].

1.1.4 Benoit B. Mandelbrot

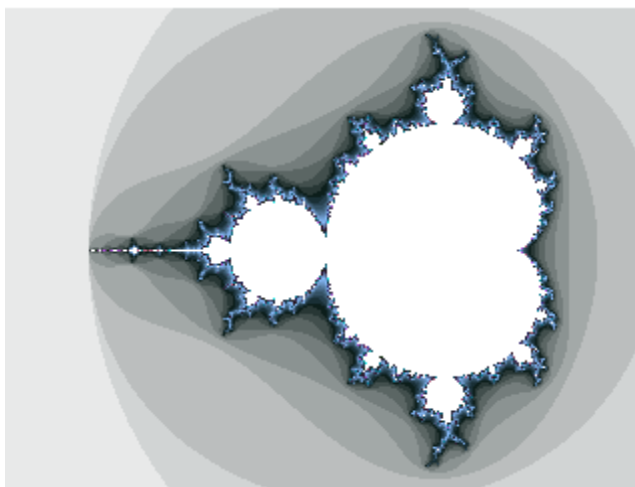
Benoit Mandelbrot se narodil 20. listopadu 1924 ve Varšavě, v židovské rodině litevského původu. Jeho otec prodával oblečení, matka byla lékařkou. Jeho dva strýci byli matematici.

V roce 1936, za pomoci svého strýce, Mandelbrotovi emigrovali do Francie. Mandelbrot nebyl příliš dobrým studentem. Jenže začala válka a německá krutá okupace Francie. Rodina se přestěhovala do Tulle, kde školu a hry nahradila chudoba a strach o život. Benoit tak vědomosti nabíral spíše samostudiem. Jeho představivost pro geometrickou vizualizaci matematických úkolů byla obdivuhodná. Poté odešel do Ameriky, kde začal pracovat pro IBM. Tam se setkal stejně jako v jeho ekonomických experimentech s fraktálními zákonitostmi.

V roce 1967 uveřejnil článek „How Long Is the Coast of Britain?“, ve kterém matematicky zdůvodňuje, jak se délka pobřeží zvětšuje s podrobností měřítka, v jakém ji pozorujeme.

Roku 1982 vznikla z publikací a písemných přepisů přednášek kniha „The Fractal Geometry of Nature“. Ve které představil svět fraktálů a jeho vhodnost pro popis přírodních forem a reálných procesů

Přišel s koncepcí fraktální geometrie a zavedl pojem soběpodobnost, jako základní charakteristiku tvaru mnoha fraktálových obrazců. Několikaleté bádání vyústilo do jednoduchého, ale důležitého vzorce: $z_{n+1} = z_n^2 + c$, jenž je matematickým vyjádřením takzvané Mandelbrotovy množiny (Obr. 3), novým symbolem fraktální geometrie [11][9, str. 27].



Obr. 3. Mandelbrotova množina [6].

1.2 Fraktální geometrie

Fraktální geometrie je samostatná a poměrně rozsáhlá vědní disciplína zasahující do mnoha dalších oborů, spolu s teorií chaosu slouží k popisu dynamických, turbulentních a nelineárních dějů.

Fraktál je objekt, jehož geometrická struktura se opakuje v něm samotném. Další definice říká, že fraktál je množina, jejíž hodnota Hausdorffovy dimenze přesahuje hodnotu dimenze topologické.

Za Topologickou dimenzi označujeme takovou kdy, pro normální nebo běžné tělesa platí, že všechny jejich parametry mohou být zadány v libovolné jednotce, aniž by se změnily vlastnosti tělesa. Například v elektronice při výpočtech odporu je jedno v jakých jednotkách počítáme, zákonitosti zůstanou zachovány.

Hausdorffova dimenze je „charakteristickým číslem“ udávajícím, jak složitý je pozorovaný útvar a s jakou rychlostí roste délka těchto útvarů (či veličin) do nekonečna [2, 5, 6].

Fraktály můžeme rozdělit na dva základní typy [1]:

- **Soběpodobné** fraktály jsou takové, v nichž se opakuje původní originální motiv mateřského tělesa. Zvětšováním získáváme opět kopii původního objektu.
- **Soběpříbuzné** fraktály jsou fraktály každodenního života. Příkladem mohou být mraky, stromy, pohoří. Zvětšováním získáme podobnou kopii původního objektu.

Další dělení fraktálů může být:

1.2.1 L-systémy

Fraktály využívající přepisových gramatik. Toto téma je detailně zpracováno v kapitole (2. L-systémy).

1.2.2 Systémy iterovaných funkcí – IFS

IFS je zkratka z anglického označení Iterated Function System (systém iterovaných funkcí). Pro generování fraktálů se používají afinní transformace (posun, rotace, zmenšení) aplikované na obrazec. Opakovanou aplikací se začne objevovat fraktální struktura [7, str. 8].

Pokud uvažujeme omezenou množinu transformací, mají tyto transformace afinní povahu tzv. jsou složeny z násobné a přičítací matice (2, 3).

$$w \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}, \text{ resp.} \quad (2)$$

$$w \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} r_1 \cos \varphi & r_2 \cos \vartheta \\ r_1 \sin \varphi & r_2 \sin \vartheta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}, \quad (3)$$

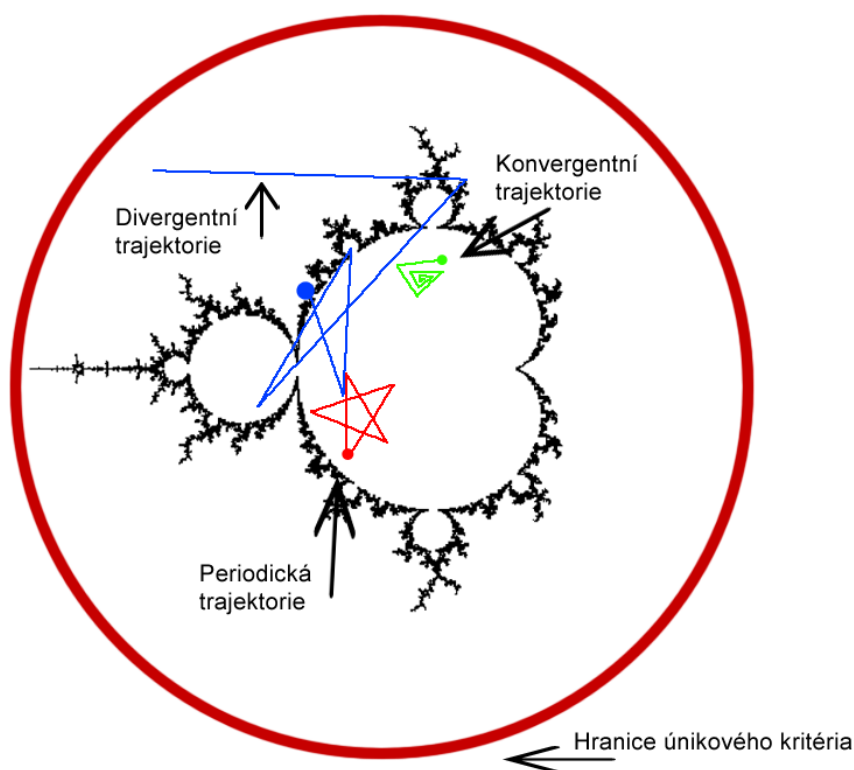
kde je význam jednotlivých symbolů:

- Úhel φ určuje otočení na ose x .
- Parametr r_1 udává přeškálování na ose x .
- Úhel ϑ určuje otočení na ose y .
- Parametr r_2 udává přeškálování na ose y .
- Dvojice e, f určují translaci útvaru podle jednotlivých os. [8, str. 37-40].

1.2.3 Time escape algorithm – TEA

Time Escape Algorithms je iterační algoritmus sloužící k obarvení Juliových množin. Provádí dané iterace do překročení zvolené hranice, nebo do vyčerpání maximálního počtu iterací (Obr. 4). Pokud trajektorie je i po maximálním počtu iterací v dané oblasti přiřadí se

bodu černá barva. Pokud trajektorie unikne, bodu se přiřadí barva úměrná počtu iterací, které byly potřeba k překročení hranice [1].



Obr. 4. Princip TEA algoritmu pro Mandelbrotovu množinu [1].

1.2.4 Dynamické systémy

Jsou to matematické modely závislé na čase. U nedynamického systému můžeme podle dané rovnice a dosazením počátečních podmínek a času t určit stav systému v jakémkoliv čase.

Pro dynamické systémy platí, že nelze předpovědět stav systému v budoucnosti, aniž by se musel simulovat celý vývoj systému. Systém je většinou popsán soustavou rovnic, které popisují aktuální stav na základě stavu předchozího [7, str. 8].

Příkladem dynamického systému je Lorenzovo vodní kolo. Tento dynamický systém popisuje chování vodního kola, na kterém jsou umístěny nádoby s děravým dnem. Kolo se otáčí v důsledku tíhové síly. Směr otáčení Lorenzova kola se chová chaoticky.

1.2.5 Náhodné fraktály

V přírodě se často vyskytují nesymetrické objekty, pro jejich popis se hodí více fraktály s nepravidelnou strukturou. Tedy fraktály, do kterých je vnesena náhoda.

Způsob, jakým se náhodnost bude podílet při generování fraktálů, bude vždy určovat tvar fraktálu a mimo další charakteristiky i jeho Hausdorffovu dimenzi. Pro generování náhodných čísel se používá například Gaussovský generátor, nebo generátor bílého šumu.

1.3 Dimenze

Ve fyzice a matematice je dimenze objektů popsána jako počet souřadnic potřebných k popsání bodu objektu. Udává přesný popis parametrů konceptu a vzhledové složitosti jakýchkoliv geometrických objektů. Tento popis může být použit i na abstraktní objekty, které nelze přímo zobrazit [19].

1.3.1 Euklidovská dimenze

Euklidovská dimenze je vždy celočíselná a je limitním případem dimenze fraktální. S touto dimenzí se setkáváme v klasické geometrii a reálném životě, jsou to například dimenze 1 (čára), dimenze 2 (plocha) a dimenze 3 (objem) [9, str. 64].

Euklidovský prostor dimenze n se značí obvykle jako \mathbb{R}^n (ve starší literatuře jako E_n). Kde \mathbb{R}^n je prostorový vektor [19].

1.3.2 Topologická dimenze

Topologická dimenze (D_T) je celočíselnou dimenzí, využívaná vědou zvanou topologie, charakterizující prostor.

Zabývá se vlastnostmi, které se deformací, otáčením nebo stlačováním nemění. V topologii je důležité, zda je objekt spojitý, zda obsahuje otvory a podobně. Nezáleží tedy na velikosti a na vlastnostech, které se mění deformacemi, ani na dimenzi prostoru kde je daný objekt umístěn. Často se zde pracuje v mnoha dimenzích.

„Objekt, který je možné homomorfne (stejnětvárně) převést stlačováním a ohýbáním na jeden ze simplexů, má stejnou topologickou dimenzi jako simplex. Simplex v tomto případě je bod s topologickou dimenzí 0, úsečka délky 1 s topologickou dimenzí 1, pravoúhlý trojúhelník s odvěsnami délky 1 má topologickou dimenzi 2 a tetraedr má topologickou dimenzi 3.“ [4]

1.3.3 Fraktální dimenze

Označována také jako Hausdorff-Besicovitch dimenze (D_H).

Tato dimenze je použita pro jednu z definic fraktálních objektů: „Útvar je fraktálem, pokud jeho fraktální dimenze převyšuje jeho dimenzi topologickou.“ (4) [19]

$$D_H > D_T \quad (4)$$

Udává členitost objektu. Měřením délky geometricky hladké křivky, která má dimenzi 1, dostaneme při měření v různých měřítkách vždy stejné konečné číslo. Ovšem například při měření délky ostrova se při zmenšování měřítka zvětšuje délka naměřené vzdálenosti až k nekonečnu. Taková křivka zabírá více než hladká křivka, zároveň také nevyplňuje celou rovinu. Dimenze bude tedy mezi 1 a 2 z toho plyne že, fraktální dimenze takové křivky (ostrov) bude neceločíselná, můžeme se tedy setkat s dimenzí např. 1.68, 2.56, atd.

Fraktální dimenze udává rychlost, s jakou délka těchto útvarů roste do nekonečna. Pokud se D_H, D_T od sebe málo liší, členitost útvaru bude malá [5].

Soběpodobnostní dimenzi D_S vypočítáme jako (5)

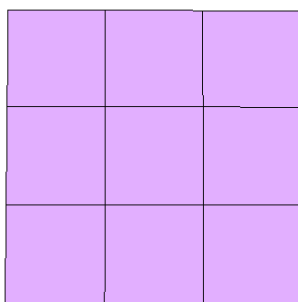
$$D_S = \frac{\ln N}{\ln s} \quad (5)$$

, kde d je reálné číslo a musí platit $d \geq 0$, s je dělitel a N je počet stejných částí na které se těleso rozdělí (původních objektů které obsahuje nový objekt). Soběpodobnostní dimenze je často rovna fraktální dimenzi D_H .

Při výpočtu fraktální dimenze D_H (6) u přímky ji rozdělíme na 3 části, s je tedy 3. Získáme 3 soběpodobné úsečky, N je tedy rovno 3.

$$D_H = \frac{\ln N}{\ln s} = \frac{\ln 3}{\ln 3} = 1 \quad (6)$$

Při výpočtu fraktální dimenze (7) volíme dělitel s opět 3. Velikost čtverce je nyní 9 krát větší než původně. N se rovná tedy 9.



$$D_H = \frac{\ln N}{\ln s} = \frac{\ln 9}{\ln 3} = 2 \quad (7)$$

Předešlé dva příklady (úsečka a čtverec) mají stejnou topologickou a fraktální dimenzi. Podle jedné z definic fraktálů nejsou tedy fraktálem.

Příkladem objektu s neceločíselnou dimenzí může být Kochova vložka (8), kdy rozdělením původní úsečky na 3 díly a aplikováním nahrazovacího pravidla vznikne úsečka 4 krát delší než původní (iniciátor v novém objektu obsažen 4 krát). N je tedy rovno 4 a $s = 3$ [4][18][19].

$$D_H = \frac{\ln N}{\ln s} = \frac{\ln 4}{\ln 3} = 1.261 \quad (8)$$

Dalšími metodami pro zjištění dimenze jsou: celková variační metoda, R/S dimenze, mřížková metoda, obvodová metoda.

2 L-SYSTÉMY

L-systémy někdy označovány jako Lindenmayerovy systémy podle Aristida Lindenmayera, který je použil již v roce 1968 pro simulaci vývoje mnohobuněčných organismů na počítači. Základní myšlenka na základě které A. Lindenmayer založil L systémy je, že na vývoj organismu může být pohlíženo jako na vykonávání určitého programu uloženého v oplodněném vajíčku. Na vyšší mnohobuněčné organismy pak lze pohlížet jako na kolekci příslušně programovaných konečných automatů [1].

Při tvorbě L-systémů každý terminální symbol (přepisovaný znak) v řetězci má přiřazený jistý význam (transformace, vykreslení nějakého geometrického objektu, změnu vlastností). Pravidla jsou buď předem zadána množinou přepisovacích pravidel, nebo se mění v průběhu generování fraktálního obrazce, například na základě zpětné vazby či na podněty okolního prostředí. Fraktální objekty generujeme za pomoci rekurze respektive iteracemi v gramatice. Kdy na pravé straně je symbol shodný se symbolem na levé straně. Přepisovací pravidla se aplikují na počáteční řetězec (axiom) [26].

S pomocí L-systémů lze generovat fraktální objekty, které se podobají rostlinám, stromům a dalším přírodním útvarům. Podobný přepisovací mechanismus je například použit jako základní kámen populární hry „game of life“.

2.1 Deterministické a kontextově nezávislé L-systémy (DOL)

DOL jsou skupinou fraktálů definovaných pomocí regulárních nebo bezkontextových přepisovacích gramatik, u kterých se v jednom derivačním kroku přepisují všechny symboly řetězce současně. Pro zajištění determinismu L-systému je nutné dodržet podmínku, že v množině P nesmějí existovat dvě pravidla se stejnou levou stranou [26].

Jde o rekurzivní proces definovaný jako uspořádaná trojice (9)

$$G = (V, P, S) \quad (9)$$

,kde

- V označuje abecedu symbolů (U L-systémů je použita gramatika kde splývají terminální a nonterminální symboly), V^* je množina všech slov nad abecedou V . V^+ je množinou všech neprázdných slov nad abecedou V .
- $S \in V^+$ je neprázdné slovo nazývané axiom, nebo také seed (semínko). Definuje počáteční stavy systému.

- $P \subset V \times V^*$ je konečná množina přepisových pravidel. Pro každý znak $a \in V$ existuje alespoň jedno slovo $x \in V^*$. Pravidla zapisujeme ve tvaru $a \rightarrow x$, kde a je přepisovaný symbol (předchůdce) a x pravá strana pravidla (nástupce). Matematický symbol \rightarrow znamená derivaci, tedy paralelní přepsání všech symbolů.
Pokud není definováno pro a žádné pravidlo, předpokládá se, že P obsahuje identické pravidlo $a \rightarrow a$ [13, str. 3-6].

2.1.1 Příklad L-systému:

Tento příklad představuje původní Lindenmayerův L-systém modelování růstu řas.

$$V = \{A, B\}$$

$$S = \{B\}$$

$$P = \{B \rightarrow A, A \rightarrow AB\}$$

Abeceda V udává, že řetězce se budou skládat jen ze znaků A, B . Pravidlo $B \rightarrow A$ znamená, že znak B bude vždy nahrazen znakem A a pravidlo $A \rightarrow AB$ nahrazuje znak A za řetězec AB . B je axiom (počáteční řetězec) tzv. nultá iterace L-systému. Při prvním derivačním kroku (první iteraci) je B nahrazeno znakem A podle přepisových pravidel P . Při druhé iteraci je znak A přepsán na řetězec AB . Tento proces opakujeme rekurzivně do požadované iterace.

Příklad prvních třech iterací:

$$i = 0 : B$$

$$i = 1 : A$$

$$i = 2 : AB$$

$$i = 3 : ABA$$

2.2 Interpretace symbolů

L-systémy byly vyvinuty zejména pro generování řetězců a jejich následnou interpretaci. Nejpoužívanější geometrická interpretace je založená na takzvané želví grafice. Poslední posloupnost symbolů získaná v poslední iteraci se ve druhém kroku geometricky interpretuje. Řetězec je v tomto případě považován za sekvenční posloupnost příkazů pro želvu. Postupným vykonáváním příkazů, kde každý symbol výsledného řetězce reprezentuje příkaz pro pohyb virtuální želvy, která při pohybu může a nemusí zanechávat

stopu. Želva pro svůj posun nepotřebuje znát svoji absolutní polohu, je tedy nezávislá na souřadném systému. Stav se skládá ze dvou prvků, z polohy želvy a z její orientace. Pomocí těchto příkazů dovede želva kreslit různé rostliny, stromy, řeky atd. Složitější aplikace směřují k využití těchto fraktálů ke generování plošných i prostorových textur [8, str. 34-36][13, str. 6-11].

2.2.1 Interpretace 2D

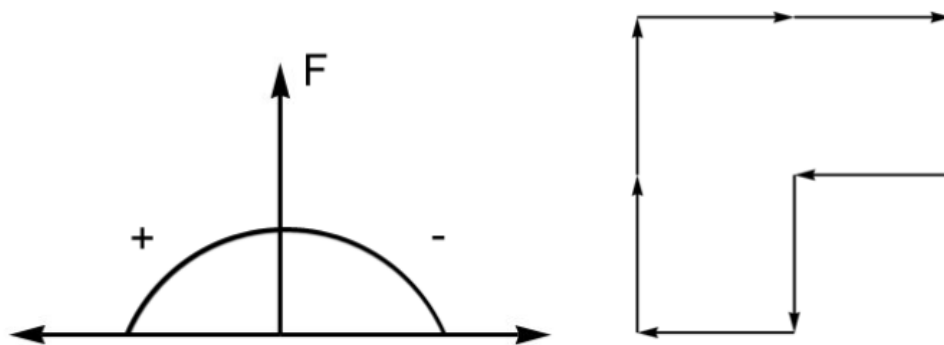
Stav želvy je definován trojicí (x, y, α) , kde (x, y) je poloha v kartézských souřadnicích a (α) je natočení želvy v prostoru. Dále je dána délka kroku d a úhel otočení δ [13, str. 6-11].

Želva čte sekvenčně zadaný řetězec dané iterace a pomocí tabulky akcí interpretuje jednotlivé symboly, jako příkazy pro vykreslení daného L-systému. Základní L-systémy interpretované želví grafikou v ploše, mohou používat následující symboly (Tab. 1):

Symbol	Interpretace
F	Posun želvy dopředu ve směr aktuálního natočení δ a délce kroku d . Stav želvy se změní na (x', y', α') , kde $x' = x + d \cdot \cos \alpha$ a $y' = y + d \cdot \sin \alpha$. Mezi původním stavem a stavem novým vznikne úsečka.
f	Posun dopředu o délku d bez kreslení úsečky.
B	Posun želvy vzad s kreslením úsečky.
+	Otočení vlevo (protisměru hodinových ručiček) o úhel δ . Nový stav bude tedy $(x, y, \alpha + \delta)$.
-	Otočení vpravo (ve směru hodinových ručiček) o úhel δ . Nový stav bude tedy $(x, y, \alpha - \delta)$.
	Otočení o 180° .

Tab. 1. Příkazy pro pohyb želvy v rovině [13, str. 6-11].

Tabulka může být rozšířena o další symboly pro řízení želvy, můžeme tak například ovlivňovat barvu, šířku kreslené čáry atd. Díky těmto symbolům můžeme vykreslit i složité či zdánlivě kulaté tvary. Základní interpretaci symbolů najdeme na (Obr. 5).

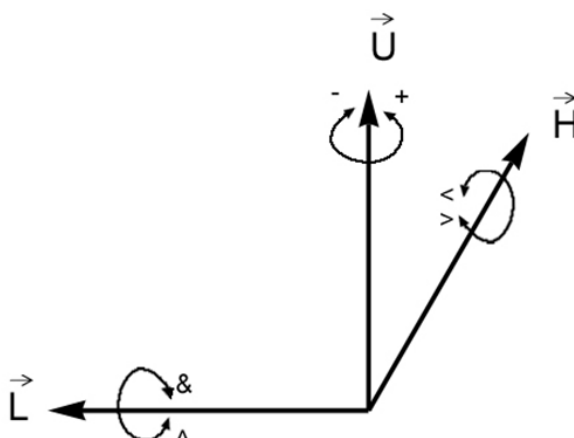


Obr. 5. Želví interpretace řetězce symbolů F, +, -. Úhel natočení je 90° .

2.2.2 Interpretace ve 3D

Želví grafika může být rozšířena do 3D. Trojrozměrné L-systémy mají mnohem širší praktické využití než předešlá interpretace v ploše (Obr. 7). Při pohybu želvy v prostoru má želva mnohem větší volnost. Želva může být v prostoru natočena zcela libovolně. Stav želvy je definován její pozicí a orientací v prostoru. Stav želvy se v tomto případě popisuje již složitějším způsobem, kde pozice je určena kartézskými souřadnicemi. Orientace želvy v prostoru je dána třemi vektory \vec{H} , \vec{L} , \vec{U} , které označují směr želvy vpřed, vlevo a nahoru (Obr. 6).

Tyto vektory jsou jednotkové a na sebe navzájem kolmé (ortogonální), platí teda $\vec{H} \times \vec{L} = \vec{U}$. Symbol \times značí vektorový součin. Pro popis polohy stačí jen dva vektory, třetí můžeme dopočítat [13, str. 19].



Obr. 6. Ovládání želvy v prostoru.

Rotace želvy kolem os určené vektorem $\vec{H}, \vec{L}, \vec{U}$, je reprezentována maticemi:

$$R_u(\alpha) = \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

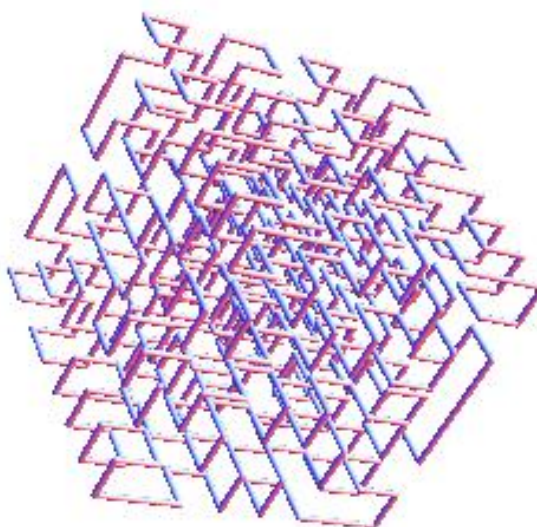
$$R_L(\alpha) = \begin{bmatrix} \cos \alpha & 0 & -\sin \alpha \\ 0 & 1 & 0 \\ -\sin \alpha & 0 & \cos \alpha \end{bmatrix}$$

$$R_H(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix}$$

Tabulka příkazů pro ovládání želvy v 3D prostoru (Tab. 2):

Symbol	Interpretace
F, B	Kreslení úsečky v aktuálním směru podle vektorů $\vec{H}, \vec{L}, \vec{U}$. F-vpřed, B-vzad.
f, b	Posun v aktuálním směru, bez kreslení úsečky, f - vpřed, b - vzad.
+	Natočení doleva o úhel δ .
-	Natočení doprava o úhel δ .
&	Natočení dolů o úhel δ .
^	Natočení nahoru o úhel δ .
>	Nahnutí doleva o úhel δ .
<	Nahnutí doprava o úhel δ .
	Otočení o 180° ve směru vektoru \vec{U} .

Tab. 2. Příkazy pro pohyb želvy v prostoru [13, str. 19].



Obr. 7. Ukázka 3D Hilbertovy křivky.

2.3 Rozvětřující se struktury

Při vytváření kontextově nezávislých (DOL) systémů želva vykresluje přímku o délce kroku d a úhlu natočení δ . Postupným vykreslováním vzniká úsečka složitějších tvarů. Úsečky se mohou protínat a být neviditelné (posun bez kreslení). Výše uvedené základní příkazy neumožňují tvorbu větvících se struktur, protože při vykreslování není k dispozici žádná paměť pro uložení aktuálního stavu [13, str. 21].

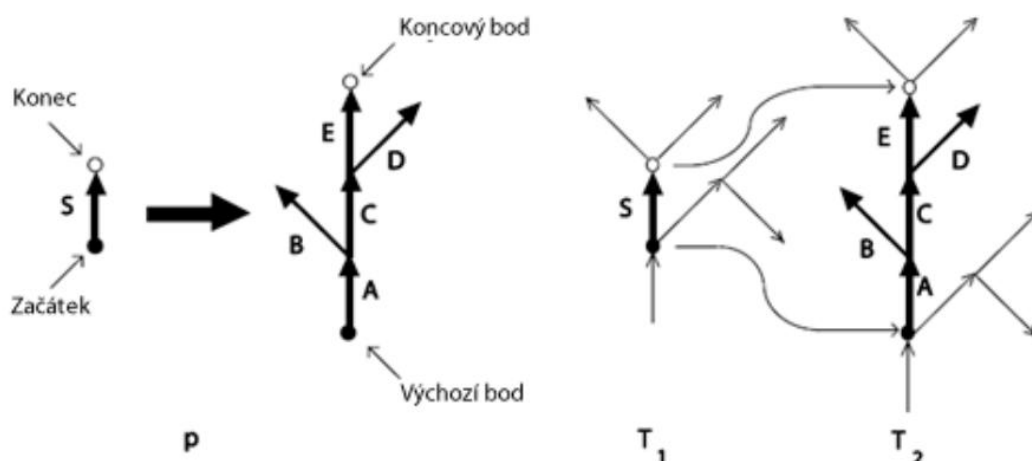
Řešení je tedy zavedení zásobníkové paměti, do kterého je při interpretaci řetězce ukládán stav želvy. Těchto stavů může být uloženo na zásobník více. Zásobník využívá struktury FIFO (první dovnitř, první ven). Při načtení ze zásobníku se stav želvy přesune na uloženou pozici bez vykreslení úsečky. Pomocí prepisových pravidel je možno reprezentovat požadovanou paměť nesystémově, například rekurzivní interpretací části prepisovacího řetězce [25].

2.3.1 Osové stromy

Kořenový strom má hrany, které jsou označeny a nasměrovány. Sekvence hran tvoří cestu z počátečního bodu (označován jako kořen) do koncového uzlu. V biologii se hrany označují jako větve. Větev následovaná aspoň jednou další větví se nazývá vnitřní uzel. Konečná větev se nazývá vrchol [13, str. 21-23].

Osový strom (Obr. 8) je typem kořenového stromu. Mezi hranami je vždy přímý výstupní segment. Všechny ostatní hrany jsou označeny jako vedlejší segmenty. Cesta v osovém stromě se nazývá osa, pokud jsou splněny následující podmínky:

- První segment vychází z kořene stromu, nebo jako boční větev (segment) z nějakého uzlu.
- Každý další větev je přímý segment.
- Poslední segment není ve stromě následován žádným přímým segmentem.



Obr. 8. Stromová struktura [13].

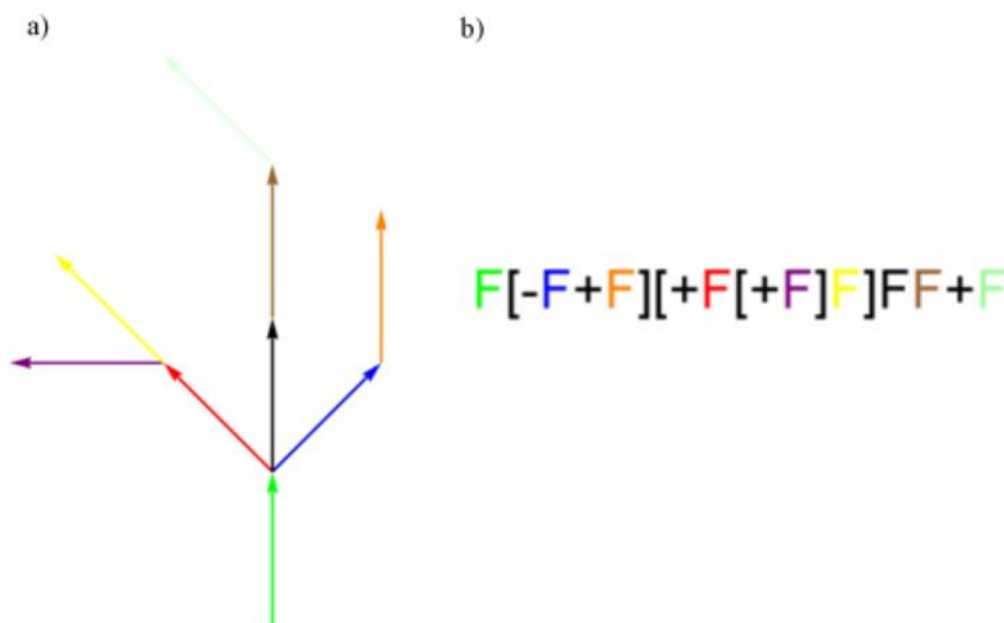
2.3.2 Závorkové l-systémy

Závorkové systémy řeší problém reprezentace stromů datových struktur. Je několik možností jak reprezentaci řešit a to s využitím ukazatelů, nebo závorkových systémů. Závorkové systémy (Tab. 3) jsou reprezentovány symboly [,], které rozšiřují základní gramatiku.

Symbol	Interpretace
[Vloží aktuální pozici želvy na zásobník. Uložená informace obsahuje všechny atributy, které želva využívá (pozici, orientaci, vlastnosti úsečky,...).
]	Změní aktuální stav želvy (včetně jejich vlastností) na hodnotu uloženou na vrcholu zásobníku.

Tab. 3. Interpretace symbolů pro závorkové L-systémy. Doplnění (Tab. 1.) [13]

Nahrazení v závorkových systémech probíhá stejně jako v klasických DOL L-systémech s tím že závorky nahrazují samy sebe. Mezi závorky [,] se vkládají samostatné sekvence příkazů pro pohyb želvy. Po provedení znaků mezi závorkami se želva vrátí na původní pozici, tyto znaky můžeme tedy považovat za skok do podprogramu a návrat z podprogramu. Ukázka závorkové interpretace je na (Obr. 9) [13, str. 24-27].



Obr. 9. Ukázka závorkového systému b) reprezentující Osový strom a).

Úhel natočení $\delta = 45^\circ$.

2.4 Stochastické L-systémy

Všechny objekty generované stejným deterministickým L-systémem jsou vždy shodné. Při generování přírodních objektů jako jsou stromy či keře, působí tyto systémy nerealisticky. Přírodní objekty nejsou většinou pravidelné, ale obsahují jisté prvky náhodnosti a nesymetrie. Existuje několik metod, které se snaží napodobit skutečné tvary přírodních objektů.

Jednou z metod simulující určitou náhodnost může být například simulace 3D růstu rostlin s využitím gravitace a dalších přírodních vlivů [25].

Jinou možností je vložení náhodných prvků do interpretace řetězce tzv. randomizace želvího řetězce. Jsou náhodně měněny geometrické vlastnosti generovaných objektů, jako je délka segmentu nebo velikost úhlu mezi segmenty. Avšak základní topologie zůstává zachována [12].

L-systémy napodobující skutečný tvar přírodních objektů se nazývají stochastické. Tyto systémy umožňují náhodnou změnu topologie rostliny i její geometrické vlastnosti. Jsou založeny na náhodném výběru prepisovacích pravidel (pravé strany) pro jeden prepisovaný symbol. Pro jedno písmeno abecedy může být aplikováno jedno prepisovací pravidlo v

50%, druhé pravidlo se aplikuje v 30% a třetí ve 20% případů [13]. Při generování velkého množství objektů v jednom pohledu se objekty jeví jako různorodé a tedy více reálné.

Stochastický OL-systém je definován jako uspořádaná čtveřice $G_\pi = (V, \omega, P, \pi)$, kde V je abeceda, ω axiom a P je množina přepisových pravidel (Detailní definice - 2.1 Deterministické a kontextově nezávislé L-systémy (DOL)). Funkce $\pi: P \rightarrow [0,1]$ se nazývá rozdělení pravděpodobnosti, přiřazující množinu pravděpodobností k množině přepisových pravidel. Předpokládá se, že pro každý přepisovaný znak $a \in V$, je součet pravděpodobností přepisových pravidel roven 1 [13, str. 28-30].

Zápis stochastických systémů (10):

$$a \rightarrow (\pi(p))x \quad (10)$$

, kde x je pravděpodobnost aplikace pravé strany pravidla $\pi(p)$ na znak a .

Příklad:

Na základní axiom F bude aplikováno v každém kroku vždy jedno přepisovací pravidlo s danou pravděpodobností. Na (Obr. 10.) jsou generovány tři různé objekty na základě stejných přepisovacích pravidel s určitou pravděpodobností.

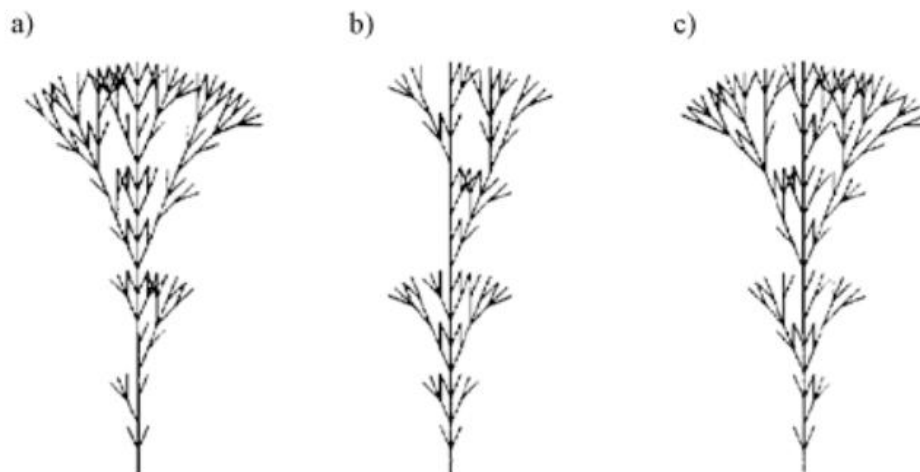
Axiom: F

Přepisovací pravidla: $F^{.50} \rightarrow F[+F][-F]F$

$F^{.30} \rightarrow F[-F]F$

$F^{.20} \rightarrow F[+F]F$

Úhel natočení: 22.5°



Obr. 10. Tři a), b), c) objekty stochastického systému se stejnými přepisovacími pravidly [21, str. 4].

2.5 Kontextové L-systémy

V předchozích případech byla využívána bezkontextová přepisovací pravidla, nezáleželo tedy na umístění přepisovaného symbolu. Kontextové L-systémy jsou rozšířením OL-systémů. A to přidáním kontextu přepisovaného symbolu. V úvahu bereme pozici, kde se přepisovaný symbol musí nacházet, aby se pravidlo použilo. Tento efekt je užitečný při simulaci mechanismus, kterým rostliny řídí svůj vývoj [13, str. 30-36].

IL-systémy jsou třídou L-systémů a jsou označovány jako (k, l) -systémy. Kde levý kontext je slovo k a pravý kontext je slovo l . Je definováno několik tříd IL systémů. OL-systémy jsou L-systémy neobsahující kontext.

2.5.1 1L-systémy

Tyto L-systémy mají pouze jednostranný kontext. Jejich forma je $al < a \rightarrow X$, nebo $a > ap \rightarrow X$, kde a je přepisovaný znak. Dále al, ap je levý a pravý kontext který musí předcházet ($<$) či následovat ($>$) za a aby se přepisovací pravidlo provedlo. Kontextové pravidla mají přednost před nekontextovými se stejným přepisovaným symbolem [13, str. 30-36].

2.5.2 2L-systémy

L-systémy mající oboustranný kontext, jejich forma je tedy $al < a > ap \rightarrow X$. Přepisovaný znak a je nahrazen řetězcem X , právě pokud se v řetězci před a nachází levý kontext al a v řetězci za pravý kontext ap .

2.6 Parametrické L-systémy

Výše uvedené systémy jsou schopné pomocí želví grafiky generovat různorodé objekty. Spoustu parametrů je ovšem konstantních, například délka úsečky může být jen násobek délky d . Pokud bychom zvolili dostatečně malou úsečku d , dosáhli bychom požadované délky, ale složitost systému a jeho paměťové, výpočetní nároky by prudce vzrostly. Gramatiku rozšíříme proto o možnost změny délky úsečky, či o změnu tloušťku a barvu. Dalším rozšířením může být například natočení o libovolný úhel [13, str. 40-41].

2.6.1 OL-systémy

Parametrické OL-systémy pracují s parametrickými znaky (Tab. 4) skládající se ze znaku abecedy V a jemu přiřazenému parametru (reálné číslo, textový kód).

Parametrický OL-systém je definován jako uspořádaná čtveřice $G = (V, \Sigma, P, S)$, kde

- V je abeceda systému
- Σ je množina parametrů
- S je axiom
- P je množina přepisových pravidel

Při zápisu pravidel využijeme symbol \rightarrow k oddělení tří částí pravidla: přepisovaný symbol, podmínka, pravá strana pravidla. Zápis může vypadat následovně (11):

$$F(a): a > 10 \rightarrow +F(t+1)XF(t-1) \quad (11)$$

Přepisovací pravidlo je aplikováno na symbol, pokud parametr a je větší jak 10

[13, str. 40-50].

Symbol	Interpretace
$F(a)$	Želva vykreslí úsečku o kroku délky a , kde $a > 0$.
$+(a)$	Otočení želvy o úhel a . Kde znaménkem určíme směr otočení
$@(a)$	Změna barvy úsečky a může být kód barvy, nebo přímo RGB hodnoty.
$T(a)$	Volba tloušťky čáry.

Tab. 4. Některé parametrické znaky a jejich význam [13, str. 46].

2.6.2 2L-systémy

Pravidla v 2L systémech (parametrických) mají oboustranný kontext, jejich forma je tedy $al < a > ap: condition \rightarrow X$. Znak a je nahrazen řetězcem X právě pokud se v řetězci před přepisovaným znakem a nachází al a v řetězci za a se nachází ap , zároveň pokud je splněna podmínka *condition* [13, str. 40-44].

Příklad 2L parametrického systému (12):

$$A(x) < B(y) > C(z): x + y + z > 10 \rightarrow E(x + y)F(y + z) \quad (12)$$

Přidáním pravidla s určitými pravděpodobnostmi vzniknou stochastické 2L-systémy.

2.7 Generování polygonů a ploch

Při vytváření objektů jakou jsou rostliny a listy potřebujeme do L-systému zavést také plochy, nebo rovinné polygony. Existuje několik metod modelování.

2.7.1 Předem definované plochy

Do abecedy je přidán další symbol \sim . Symbol značí vložení předem definovaného polygonu podle parametru, který za tímto znakem následuje. Orientace se určí podle definovaných směrových vektorů.

2.7.2 Vyvíjející se modely

Jde o modely, které jsou složeny z polygonů a na rozdíl od předem definovaných ploch. Polygony jsou vytvářeny pomocí želví grafiky, jejíž abeceda je rozšířena o další příkazy (Tab. 5):

Symbol	Interpretace
{	Začátek nového polygonu. Aktuální polygon je uložen na zásobník.
}	Konec polygonu. Pokud je na zásobníku nějaký polygon vyjme ho a nastaví jako aktuální.
G	Posun vpřed (jako F) bez zaznamenání vrcholu polygonu.
g	Posun vpřed bez kreslení (jako f) bez zaznamenání vrcholu polygonu.
.	Pozice želvy je zaznamenána jako vrchol polygonu.

Tab. 5. Symboly želví grafiky pro kreslení polygonů [13, str. 119-128].

2.7.3 Tropismus

Zavedení absolutního směru do vývoje L-systémů. Pomocí toho je možné simulovat gravitaci (geotropismus), vítr, natočení listů (fototropismus), atd. [5].

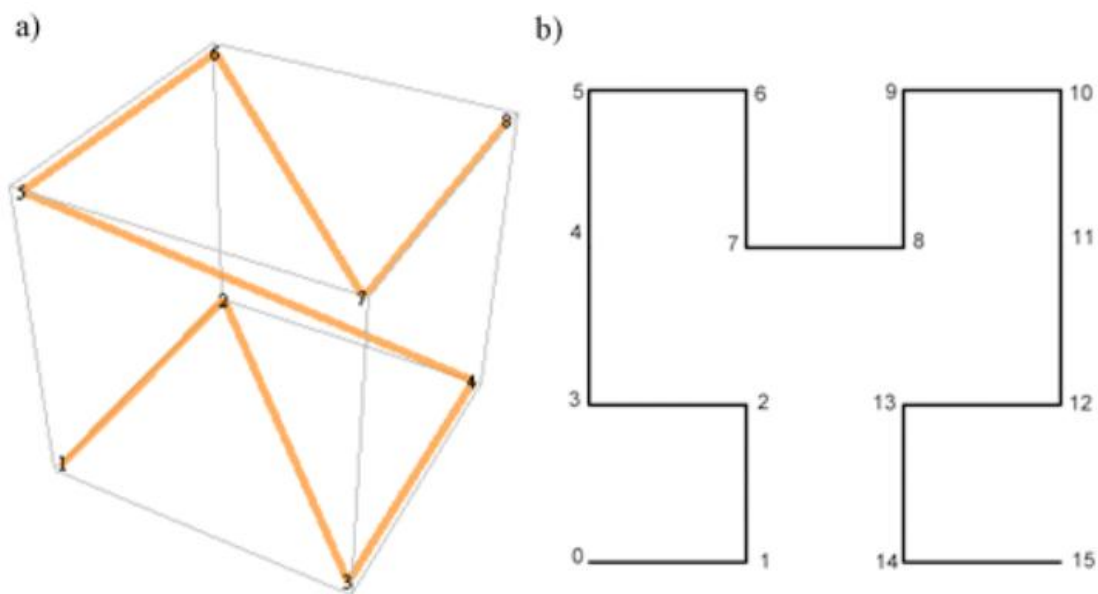
2.8 Využití L-systémů v praxi

Původní využití L-systémů bylo modelování růstu rostlin a růstových procesů. Postupem času se L-systémy začaly používat i pro další oblasti než je modelování rostlin a fraktálů. Například s jejich pomocí můžeme modelovat tok řek ve fraktálových horách [5], ulice ve virtuálních městech [6], pomocí L-systémů lze také popisovat pod-rozdělování (zjemňování) křivek. V počítačových simulacích umožňují L-systémy velmi efektivní modelování různorodých objektů, zejména stromů, rostlin, trávy, členité půdy, atd. Například ve filmu Avatar z roku 2009 bylo přes 2000 stromů, rostlin a kapradin vymodelováno právě pomocí L-systémů [24].

Nicméně L-systémy mohou být použity i v jiných oborech než je počítačová grafika, například lze pomocí nich generovat hudbu [9]. Další použití můžeme nalézt v biologii, geologii a v dalších přírodních vědách. Příkladem může být využití v elektronice při konstrukci antén, takové antény se poté nazývají fraktální antény. Nejznámější fraktální anténa je vytvořena na základě křivky Helge von Kocha. Taková anténa má poté široký pásmový rozsah, ale jsou zachovány malé rozměry [3]. Fraktální tvar může například být vyleptán jako cesta na desce plošných spojů. Využití těchto antén je zejména v mobilní technice a armádě.

Fraktální strukturu můžeme nalézt také v návrhu některých velkých měst. Půdorys města je zde rozdělen na velké čtverce (hlavní dopravní tah), které jsou postupně zjemňovány až na jednotlivé uličky [3].

Použití v informatice najdeme například u řešení problému obchodního cestujícího (úloha nalezení nejkratší cesty mezi zadanými městy) pomocí křivky vyplňující prostor. Nebo také při indexaci dat, kdy pomocí křivek vyplňující prostor jako jsou „Hilbertova křivka“ (4.1.7), nebo „Z-order curve“ je možné indexovat vícerozměrný prostor do prostoru jednorozměrného (Obr. 11). Na tomto jednorozměrném prostoru je poté možné použít algoritmy pro práci s jednorozměrnými daty [22]. Hilbertova křivka se používá častěji, neboť lépe zachovává lokalitu dat.



Obr. 11. „Z-Curve“ a) [32] a Hilbertova křivka b). Mapování vícerozměrného pole do jednorozměrného.

II. PRAKTICKÁ ČÁST

3 PROGRAMOVÁ ČÁST

Cílem praktické části je vytvořit program v prostředí Mathematica, který pomůže pochopit principy L-systémů. Vytvořený program umožní volnou úpravu předdefinovaných L-systémů a to způsobem kdy je možné vytvářet úplně nové tvary, které nezávisí na původním objektu. S využitím ovládacích prvků Mathematicy, je vytvořeno uživatelsky přátelské prostředí, ve kterém můžeme vytvářet kontextově nezávislé, parametrické, rozvětřující se a částečně stochastické skupiny fraktálů, jejichž principy byly popsány v teoretické části. Je také implementováno rozšíření do 3D oblasti, kde je zobrazeno několik známých fraktálů, opět s možností editace.

3.1 Wolfram Mathematica

Software Mathematica pochází od firmy Wolfram Research se sídlem v Champaign, Illinois, založena britským matematikem Stephenem Wolframem v roce 1987. Stephen Wolfram má za sebou mnoho objevů, vynálezů a inovací v oblasti vědy, techniky a obchodu. Jeho nejvýznamnější publikací je „A New Kind of Science“ z roku 2002. Wolfram zde používá jeho přístup počítačových simulací k řešení základních problémů ve vědě z mnoha oblastí.

Mathematica je programový systém pro provádění automatizovaných numerických, symbolických výpočtů, adaptivní vizualizací dat, dynamickou interaktivitou a výkonným programovacím prostředím. Obsahuje nástroje pro práci s databází, komunikaci s internetem, vytváření prezentací. Mathematica je dostupná pro platformy Windows, Linux, MAC OS X a Solaris.

Využívána je v oblasti vědecko-technických výpočtů, statistickém zpracování dat, finančním managementu, matematice, fyzice, biologii, chemii, geografii, ekonomii nebo informatice. Je používána při výuce na středních školách a univerzitách po celém světě.

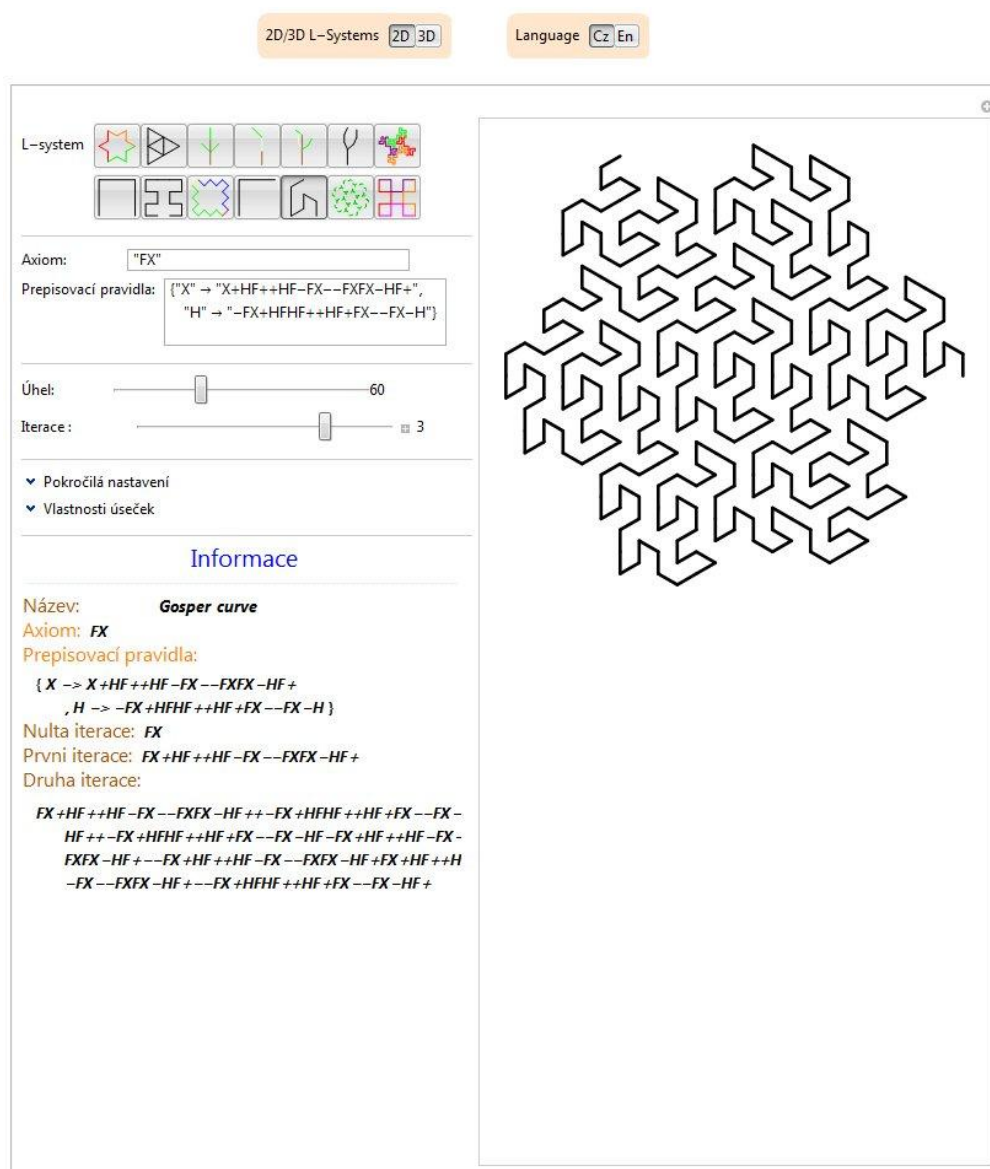
Mathematica je rozdělena do dvou částí – jádra a front endu. Jádro interpretuje výrazy a vrací výsledky. Front end poskytuje GUI, ve kterém výsledky vhodně zobrazuje [14].

Základem front endu Wolfram Mathematica je soubor nazývaný jako notebook. Soubor obsahuje buňky. Tyto buňky mohou mít různé funkce: text, spustitelný kód, matematický vzorec. Spustitelné buňky jsou vstupními a při jejich zpuštění výsledky interpretují do výstupních buněk.

3.2 Popis GUI prostředí

Základem praktické části je notebook, který je rozčleněn do dvou sekcí *L-systemy* v prostředí *Mathematica* obsahuje popis programu a možnosti jeho ovládání. V podsekcí *legenda* jsou popsány možné symboly pro interpretaci a vykreslování L-systémů. Podsekcí *Ovládání* vysvětluje chování základních ovládacích prvků. Tyto části jsou jak v české tak i v anglické verzi.

Druhá část obsahuje kód programu rozdělený na 3 podsekcce. Sekce *Initialize* obsahuje informace potřebné k vykreslení L-systémů, inicializace proměnných, jazykové mutace GUI prostředí. V následující sekci *Functions* se nachází funkce volány z vizualizační funkce „Manipulate“ (Obr. 12), která je obsažena v sekci *Visualization*.



Obr. 12. Vizualizace L-systému v prostředí Mathematica.

3.2.1 Popis ovládacích prvků

L-System: Výběr jednoho z předdefinovaných L-systémů.

Axiom: Základ prepisovacího pravidla, Axiom je také značen jako 0. iterace. Axiom se zapisuje do uvozovek ve tvaru "Axiom".

Přepisovací pravidla: jsou zapsány ve formátu {"Levá strana pravidla (cíl)" → "Pravá strana pravidla (zdroj)"[, další pravidla, ...]}.

Úhel: Inicializační úhel, rozsah hodnot 0° - 180° .

Iterace: Posuvník pro výběr zobrazené iterace. Pro každý fraktál je definována jiná maximální hodnota iterace, podle jeho složitosti.

Pokročilá nastavení:

Předešlá iterace: Pro vizualizaci nahrazovacích pravidel lze zapnout také předešlá iterace. Kvůli různorodosti jednotlivých fraktálů nemusí být vždy zobrazení v jednom pohledu vždy zřetelné, je tedy přidána možnost zobrazení pod sebou volba „Pod Sebe“.

Náhodný úhel: Umožňuje změnu úhlu v zadaném rozmezí od 0° do 180° . Náhodná generace je realizována pomocí funkce Mathematicy - *Random Integer*[{*min*, *max*}].

Šipky: Ke každé vykreslené úsečce přidá šipku ve směru aktuálního vykreslení.

Vlastnosti úseček: Nastavení vlastností vykreslovaných úseček. Nastavená barva je inicializační a při překrytí pomocí parametrického symbolu @ je plně nahrazena, vypnutí barevnosti lze provést pomocí parametru „Barevnost“.

Informace: V levé spodní části je zobrazen souhrn informací. Druhá iterace v případě dlouhého výpisu bude zkrácena. Kliknutím myši do prostoru s informacemi můžeme zapnout zobrazení parametrických znaků (@ a &), opětovným kliknutím tyto znaky opět zobrazíme.

3.2.2 Abeceda gramatiky

Pro ovládání želvy můžeme využívat různých symbolů a zápisů. Tyto symboly a funkce můžeme použít v zápisu axiomu a v prepisovacích pravidlech.

- **F, G** - Posun želvy dopředu ve směru aktuálního natočení. Mezi původním stavem a stavem novým vznikne úsečka o délce $d = konstanta$.

- ***X, Y, H*** - Prázdný symbol sloužící k přepisování. Nedojde k posunu ani se nevykreslí úsečka.
- **+** Natočení vlevo o úhel alfa.
- **–** Natočení vpravo o úhel alfa.
- **[** Uložení na zásobník.
- **]** Vyjmutí ze zásobníku a přesun na uloženou pozici.
- **@(*barva*)** - Změna barvy čáry. Všechny následující vykreslené úsečky budou této barvy, dokud nebude nastavena jiná barva.

Možné zápisy barev:

- ***RGBColor[r, g, b]*** - Kde r, g, b je hodnota od 0 do 1. Příklad: ***RGBColor[0,1,0]***
- ***GrayLevel[level]*** - Kde level je hodnota odstínu šedi od 0(černá) do 1(bílá)
- **Slovní název barvy:** Red, Green, Blue, Black, White, Gray, Cyan, Magenta, Yellow, Brown, Orange, Pink, Purple, LightRed, LightGreen, LightBlue, LightGray, LightCyan, LightMagenta, LightYellow, LightBrown, LightOrange, LightPink, LightPurple.
- **&(*Uhel*)** - změna úhlu od 0 do 180° . Inicializační úhel nastavený posuvníkem v GUI nebude dále uvažován.

4 KONSTRUKCE L-SYSTÉMŮ

4.1 2D Systémy

Pro konstrukci 2D L-systému je použita želví grafika. Pomocí této metody můžeme vykreslovat úsečky o pevné délce, které se již po několika iteracích mění ve velmi členité a složité útvary.

Způsob tvorby L-systémů želví grafikou:

- Řetězec axiomu se nahradí přepisovými pravidly, na tento výsledek jsou rekurzivně aplikovány přepisovací pravidla podle požadované iterace (Obr. 13).

```
Characters[Nest[StringReplace[#, rule] &, axiom, n] ]
```

Obr. 13. Zápis v programu Mathematica

- Z výsledného řetězce jsou postupně odebrány jednotlivé znaky, které jsou interpretovány podle své funkce v želví grafice. Na (Obr. 14.) je znázorněna funkce pro posun želvy na kreslicí ploše, kde $tBeta$ je aktuální úhel natočení, $tTheta$ je úhel o který se má želvička pootočit, $lpos$ je aktuální pozice želvičky.

```
Switch[parametr,
  "+", tBeta += tTheta;,
  "-", tBeta -= tTheta;,
  "F", lpos += {Cos[tBeta Degree] // N, Sin[tBeta Degree] // N},
  "G", lpos += {Cos[tBeta Degree] // N, Sin[tBeta Degree] // N},
  "E", lpos += {Cos[tBeta Degree] // N, Sin[tBeta Degree] // N},
  "[", stack = Prepend[stack, {lpos, tBeta}];,
  "]", lpos = First[stack][[1]]; tBeta = First[stack][[2]]; preLast = lpos; stack = Rest[stack];
];
```

Obr. 14. Ukázka interpretace symbolů v programu Mathematica.

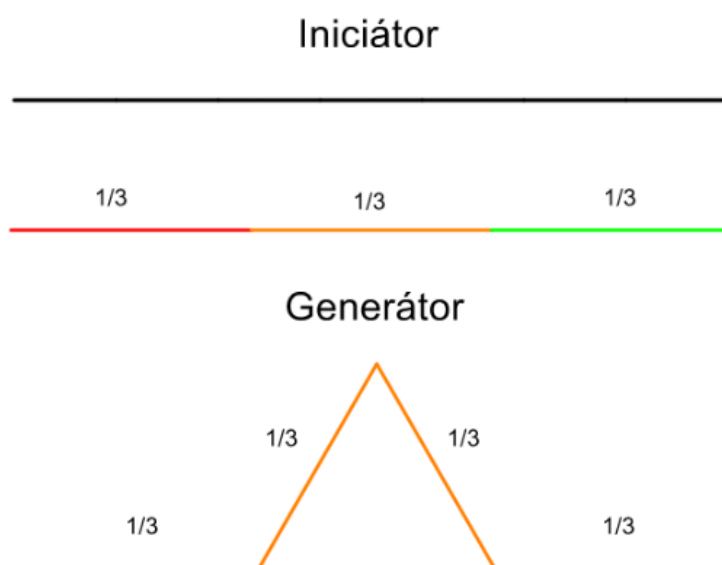
- Při pohybu želvy po „kreslicím plátně“ je zaznamenávána její stopa. Výsledkem je požadovaná lomená křivka (Obr. 15).

```
Graphics[Line[souradnice], ImageSize -> 370]
```

Obr. 15. Možné vykreslení křivky v programu Mathematica

4.1.1 Kochova vločka

Kochova vločka (Koch snowflake) je jedna z prvně popsanych fraktální křivek. Tuto křivku popsal Helge von Koch v roce 1904. Při konstrukci Kochovy křivky (Obr. 16) vycházíme z inicializátoru (axiому), kterým je trojúhelník. Generátorem je lomená úsečka, která je vytvořena rozdělením původní na třetiny, následným vyjmutím prostřední části a nahrazením dvěma rameny rovnoramenného trojúhelníku o délce $1/3$ původní úsečky. Výsledná úsečka má tedy délku $4/3$ délky původní [9, str. 20-22,39][1].



Obr. 16. Tvorba Kochovy vločky.

Konstrukce Kochovy vločky (Obr. 17) a její gramatika:

Abeceda: $F, +, -$

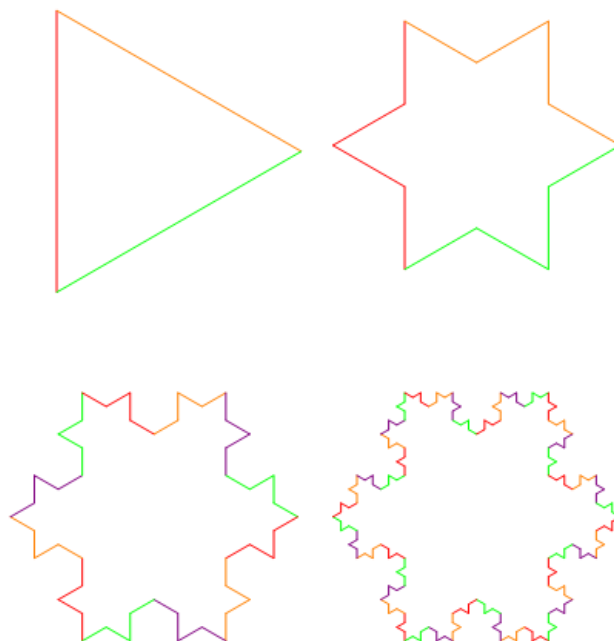
Axiom: $F - -F - -F$

Přepisovací pravidla: $F \rightarrow F + F - -F + F$

Úhel natočení: 60°

Fraktální dimenze:

$$D_H = \frac{\ln N}{\ln s} = \frac{\ln 4}{\ln 3} \cong 1.261 \quad (13)$$



Obr. 17. Čtyři iterace Kochovy vločky.

4.1.2 Sierpinského trojúhelník

Sierpinského trojúhelník (Sierpinski Sieve) je fraktál pojmenovaný po polském matematikovi Waław Sierpiński v roce 1915. Vytváří se rekurzivním dělením rovnoramenného trojúhelníku [15].

Konstrukce sierpinského trojúhelníku a její gramatika:

Abeceda: $X, F, +, -$

Axiom: $FXF - -FF - -FF$

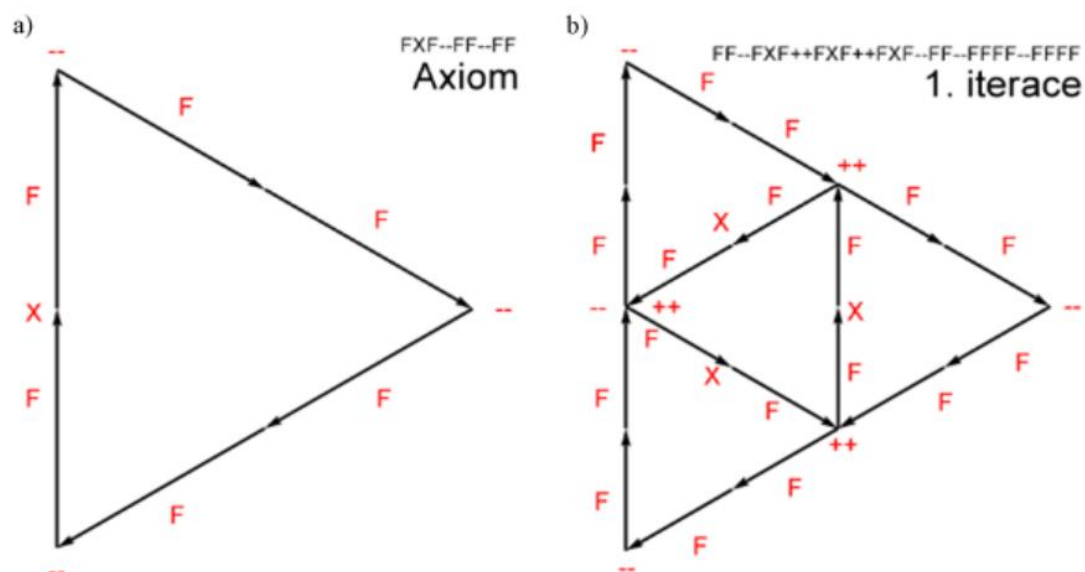
Přepisovací pravidla: $F \rightarrow - - FXF + + FXF + + FXF -$

Úhel natočení: 60°

Fraktální dimenze:

$$D_H = \frac{\ln N}{\ln s} = \frac{\ln 3}{\ln 2} \cong 1.585 \quad (14)$$

Grafické nahrazení axiomu přepisovým pravidlem pro první iteraci (Obr. 18):



Obr. 18. Axiom a) a první iterace b) Sierpinského trojúhelníku. Šipky značí aktuální směr vykreslování.

4.1.3 32- Segmentová křivka

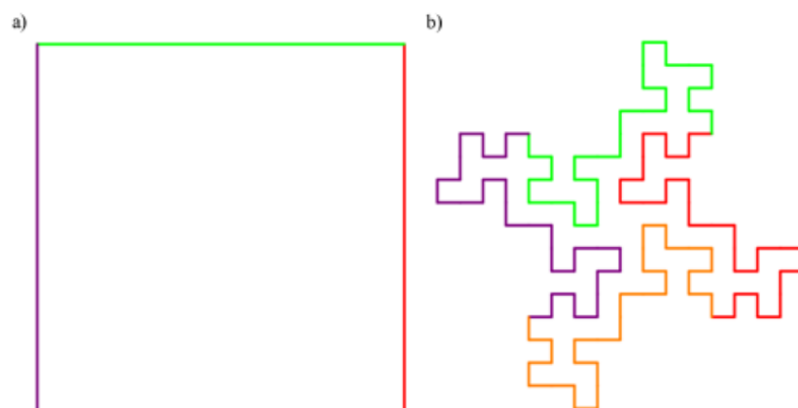
Konstrukce 32- Segmentová křivka (32 – Segment curve) (Obr. 19) a její gramatika:

Abeceda: $F, +, -$

Axiom: $F + F + F + F$

Přepisovací pravidla: $F \rightarrow F - F + F - F - F + F + FF - F + F + FF + F - F - FF + FF - FF + F + F - FF - F - F + FF - F - F + F + F - F +$

Úhel natočení: 90°



Obr. 19. Nultá a) a první b) iterace „32 - segment curve“.

4.1.4 Gosperova křivka

Gosperova křivka (Gosper curve) (Obr. 20) pojmenovaná po Bill Gosperovi je známá také jako flowsnake.

Abeceda: $X, F, H, +, -$

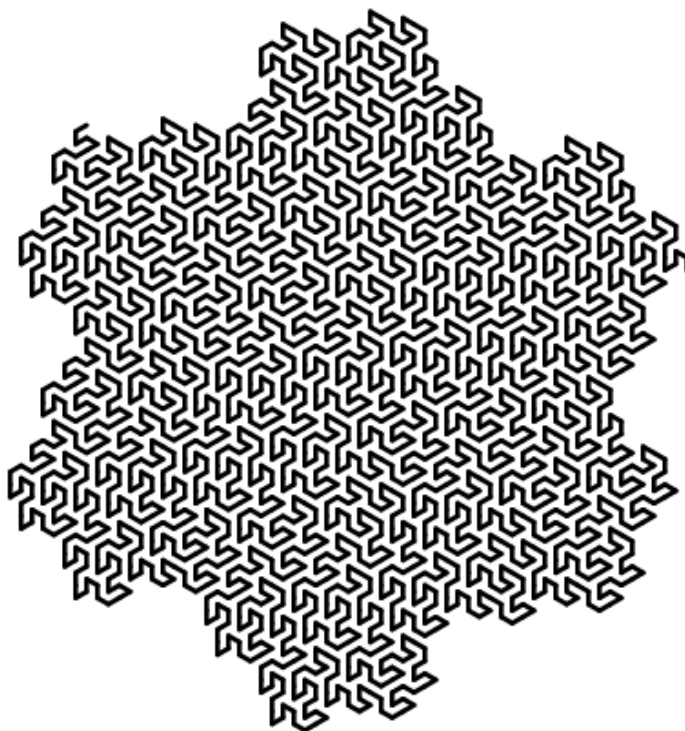
Axiom: FX

Přepisovací pravidla: $X \rightarrow X + HF + +HF - FX - -FXFX - HF +$

$H \rightarrow -FX + HFHF + +HF + FX - -FX - H$

Úhel natočení: 60°

Fraktální dimenze: 2



Obr. 20. Gosper curve.

4.1.5 Gosperův ostrov

Vnější křivka definuje prostor Gosperovy křivky. Benoit Mandelbrot roku 1977 tento prostor pojmenoval jako fraktál Gosperův ostrov (Gosper island) (Obr. 21) [17].

Abeceda: $F, +, -$

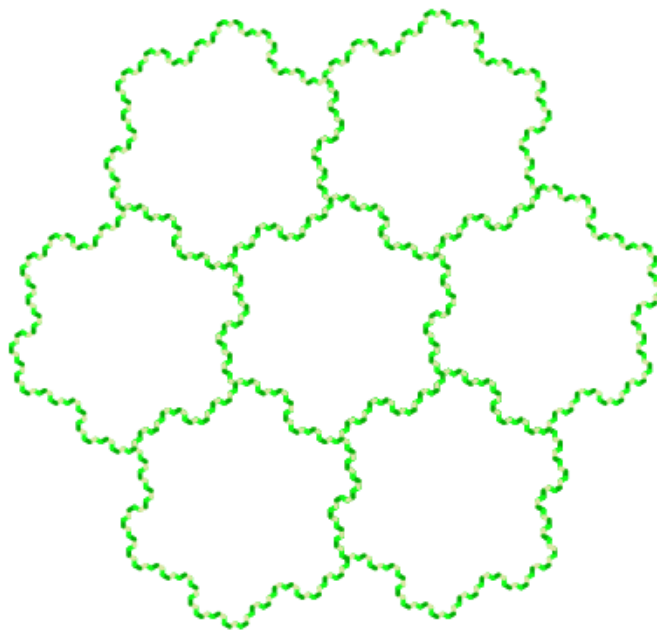
Axiom: $F - F - F[+F - F - F[+F - F - F[-F - [-F - F - F - F - F - F]F - F - F] - F - F - F] - F[+F - F - F - F - F - F] - F - F] - F - F - F[+ + +F - F - [+ + F - F - F - F - F - F]F - F - F - F]$

Přepisovací pravidla: $F \rightarrow F - F + F$

Úhel natočení: 60°

Fraktální dimenze:

$$D_H = \frac{2 \ln 3}{\ln 7} \cong 1.129 \quad (15)$$



Obr. 21. Gosperův ostrov v 3. Iteraci.

4.1.6 Sierpinského křivka

Sierpinského křivka (Sierpinski curve) (Obr. 22) byla popsána polským matematikem Waclaw Sierpiński.

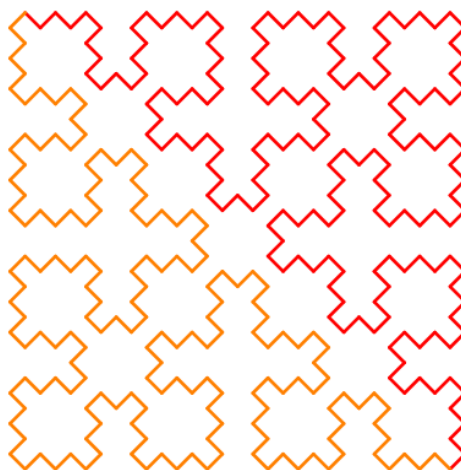
Abeceda: $X, F, H, +, -$

Axiom: $F + XF + F + XF$

Přepisovací pravidla: $X \rightarrow XF - F + F - XF + F + XF - F + F - X$

Úhel natočení: 90°

Fraktální dimenze: 2



Obr. 22. Sierpinského křivka

4.1.7 Hilbertova křivka

Hilbertovu křivku (Hilbert curve) (Obr. 23) popsal německý matematik David Hilbert (1891). Limita této křivky vyplňuje čtvercovou plochu (Fraktální dimenze je rovna 2) [20].

Abeceda: $X, F, H, +, -$

Axiom: X

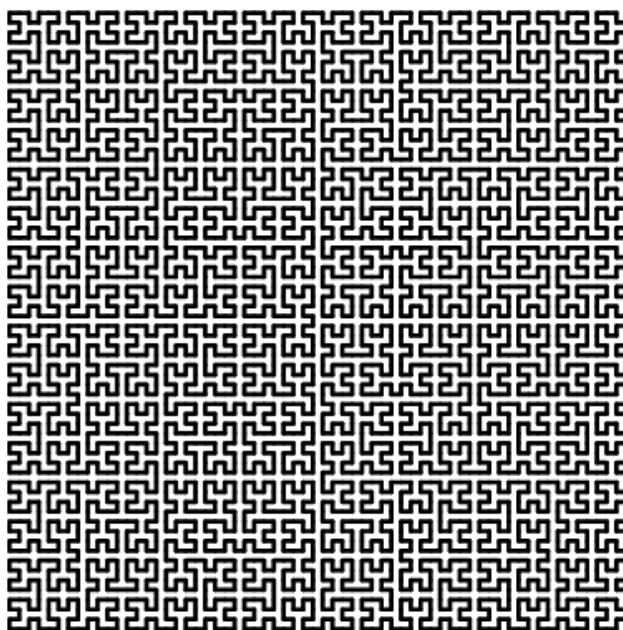
Přepisovací pravidla: $X \rightarrow +HF - XFX - FH +$

$H \rightarrow -XF + HFH + FX -$

Úhel natočení: 90°

Fraktální dimenze (16):

$$D_H = \frac{\ln N}{\ln s} = \frac{\ln 4}{\ln 2} = 2 \quad (16)$$



Obr. 23. 6. Iterace Hilbertovy křivky.

4.1.8 Moorova křivka

Konstrukce Moorovy křivky (Moore curve) (Obr. 24) a její gramatika:

Abeceda: $X, F, H, +, -$

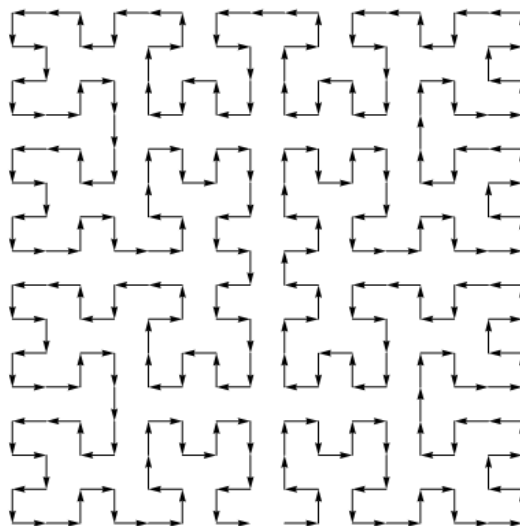
Axiom: $HFH + F + HFH$

Přepisovací pravidla: $X \rightarrow +HF - XFX - FH +$

$$H \rightarrow -XF + HFH + FX -$$

Úhel natočení: 90°

Fraktální dimenze: 2



Obr. 24. Křivka Moore curve s vyznačením směru průchodu.

4.1.9 Dračí křivka

Abeceda: $X, F, H, +, -$

Axiom: FX

Přepisovací pravidla: $X \rightarrow X + HF +$

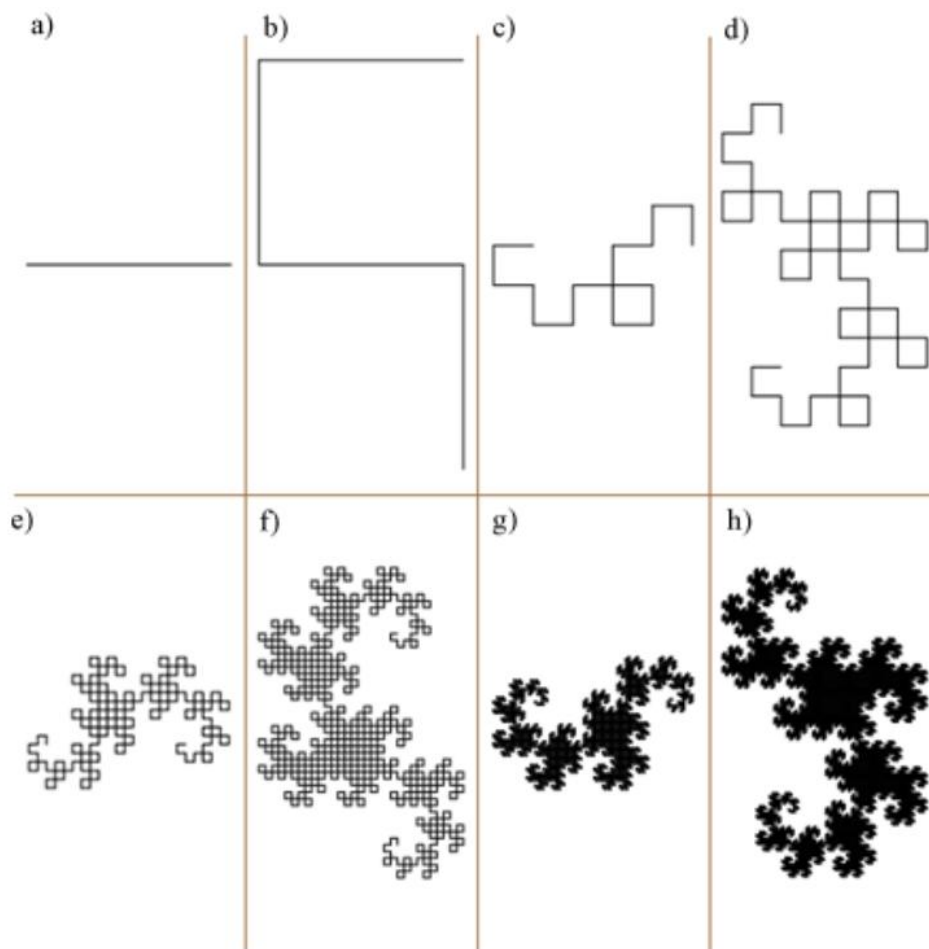
$H \rightarrow -FX - H$

Úhel natočení: 90°

Fraktální dimenze (17):

$$D_H = \frac{\ln N}{\ln s} = \frac{\ln 2}{\ln \sqrt{2}} = 2 \quad (17)$$

Dračí křivka (Dragon curve) (Obr. 25) ostatně jako spousta jiných fraktálních obrazců lze vytvořit pomocí různých zápisů axiomu a přepisovacích pravidel. Další možností je zvolení úhlu 45° , axiom: FX a přepisovací pravidla: $X \rightarrow +FX - -FY+$, $Y \rightarrow -FX + +FY-$, $F \rightarrow Z$.



Obr. 25. Iterace dračí křivky a) 0 iterace b) 2 iterace c) 4 iterace d) 6 iterace e) 8 iterace f) 10 iterace g) 12 iterace h) 14 iterace.

4.1.10 Box Fraktál

Box fractal (Obr. 26), někdy také označován jako Vicsek fractal, nebo Vicsek snowflake.

Abeceda: $F, +, -$

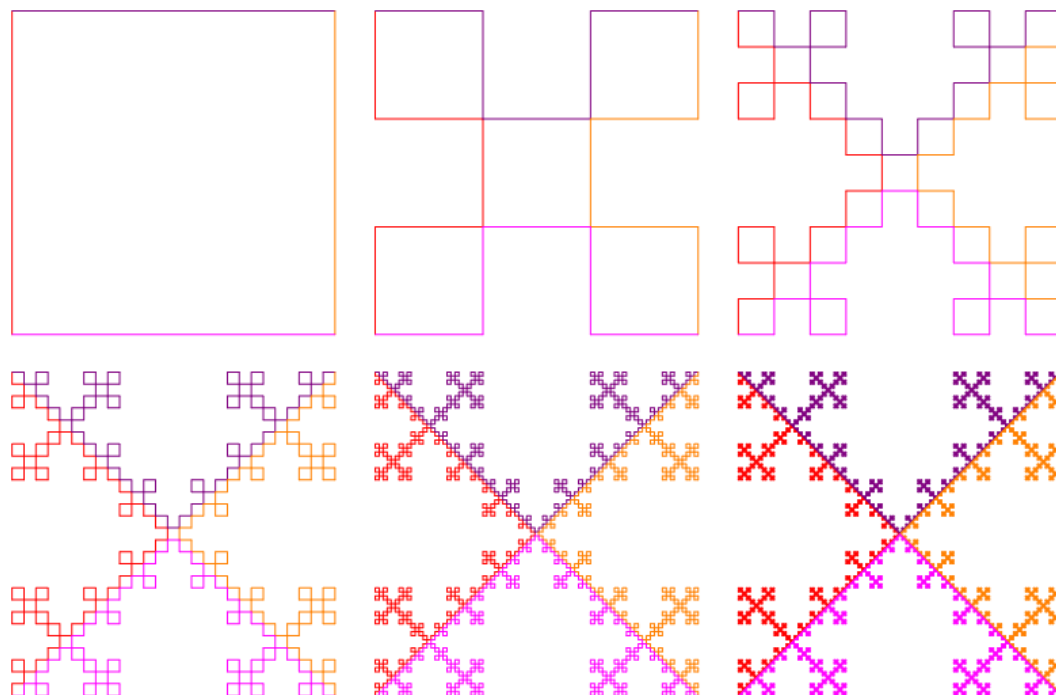
Úhel: 90°

Axiom: $F - F - F - F$

Přepisovací pravidla: $F \rightarrow F - F + F + F - F$

Fraktální dimenze (18):

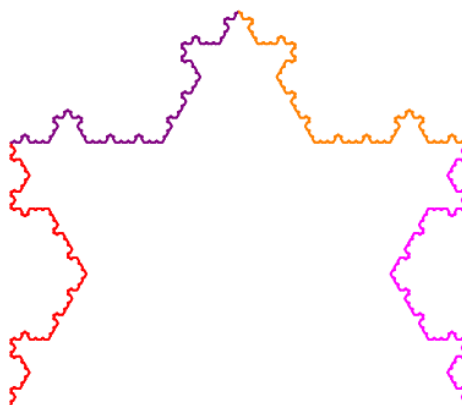
$$D_H = \frac{\ln 5}{\ln 3} \cong 1.46 \quad (18)$$



Obr. 26. Prvních šest iterací Box Fraktálu.

Při Box fraktálu na (Obr. 26) byly obarveny jednotlivé úsečky axiomu. Každá barva je přiřazena jednomu iniciátoru. Po 6. Iteraci je již patrné, že pro dané rozlišení je dosaženo konečného tvaru obrazce.

Podobu fraktálního obrazce můžeme měnit, také pouhou změnou úhlu, jak demonstruje (Obr. 27). Byl použit stejný axiom a přepisovací pravidla jako pro fraktál na (Obr. 26).



Obr. 27. Box fraktál a volba úhlu 60°.

4.1.11 Stromy

Původní účel L-systémů bylo modelování a simulace růstu rostlin (Obr. 30). Jednoduchý strom (Obr. 29) vznikne pomocí dvou nahrazovacích pravidel. První pravidlo je použito pro rozvětvení a druhé pro prodlužování (Obr. 28).

Abeceda: $E, F, +, -$

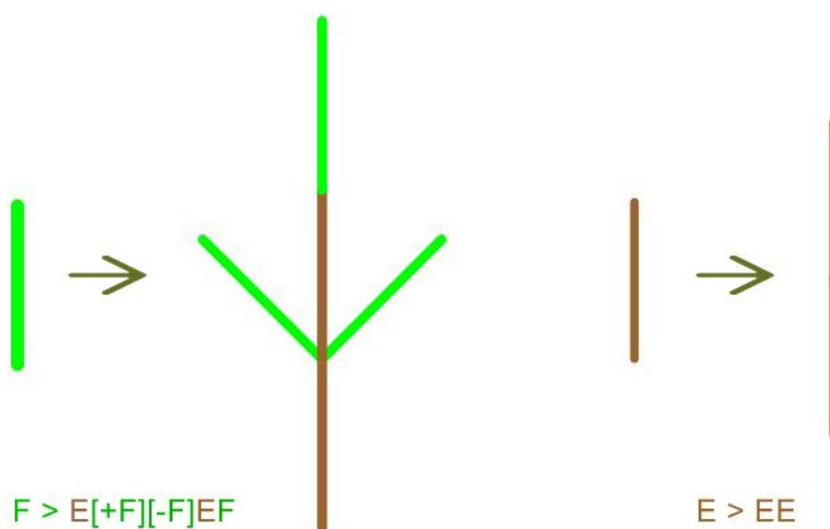
Axiom: F

Úhel: 90°

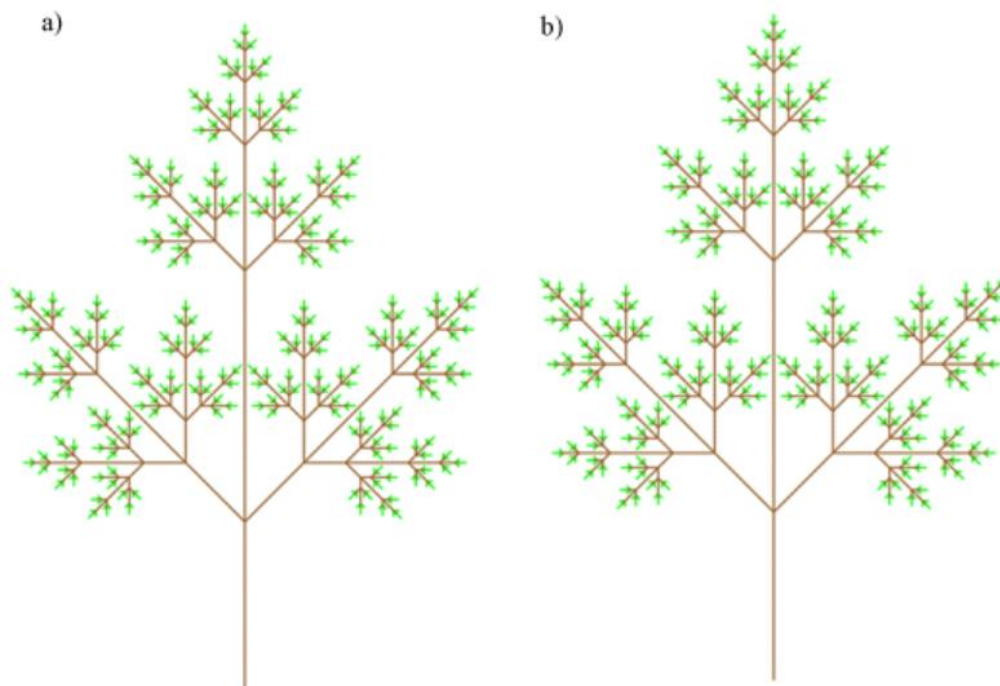
Přepisovací pravidla: $F \rightarrow E[+F][-F]EF$

$E \rightarrow EE$

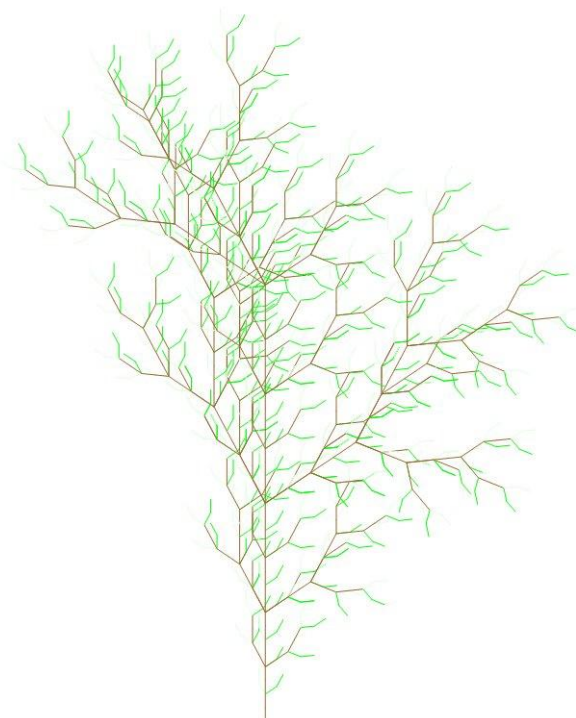
Kde E se vykresluje stejně jako F .



Obr. 28. Nahrazovací pravidla pro L-systém Strom.



Obr. 29. Strom v 6. Iteraci. Konstantní úhel 45° a). Náhodný úhel 35° - 55° b).



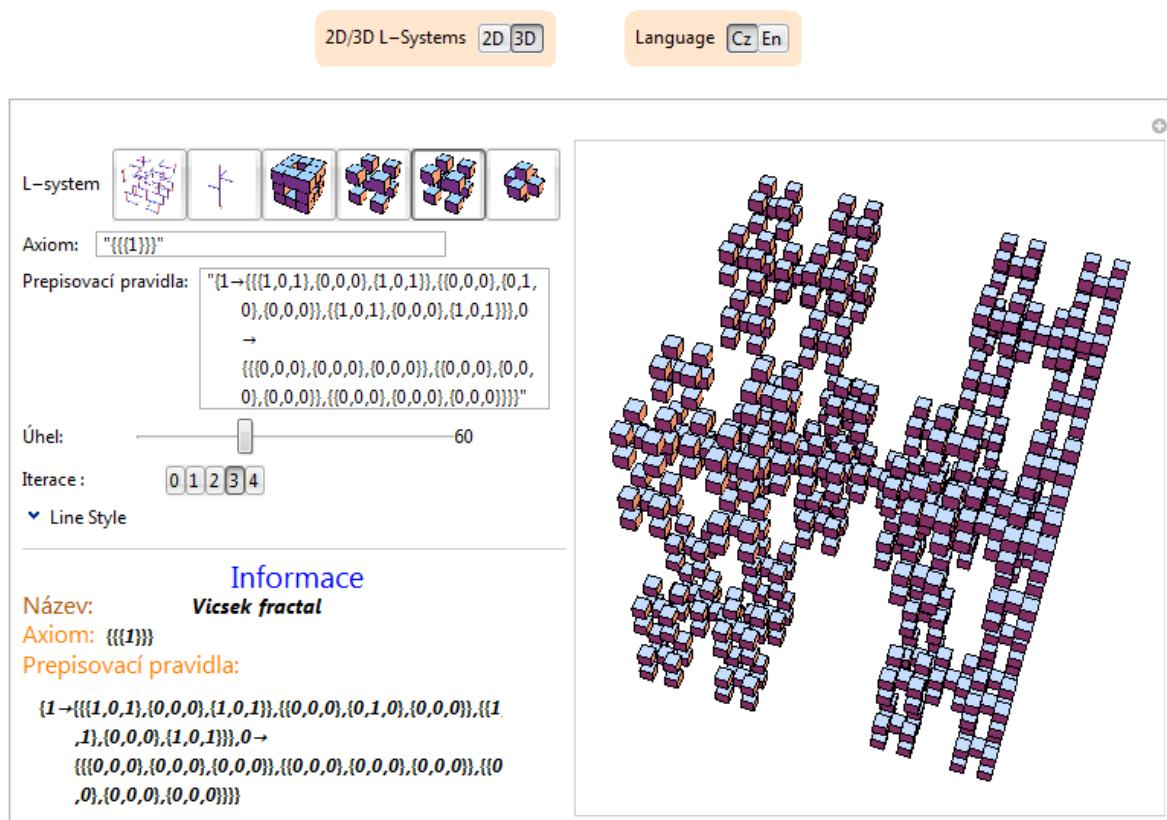
Obr. 30. Příklad fraktálního stromu ve 4. iteraci.

Axiom: FX , prepisovací pravidla: $F \rightarrow FF -$

$[-F + F] + [+F - F]$, $X \rightarrow FF + [+F] + [-F]$

4.2 3D Systémy

Pro vizualizaci 3D L-systémů (Obr. 31) bylo vybráno pár základních zástupců této kategorie. Vykreslení určitých 3D L-systémů je možné opět pomocí želví grafiky, která je obohacena o příkazy natočení ve všech třech osách. Další fraktály které nejsou založeny na úsečkách, ale na tvarech byly vypočteny pomocí nahrazovací gramatiky jako trojrozměrné matice a interpretovány pomocí funkcí Mathematicy.



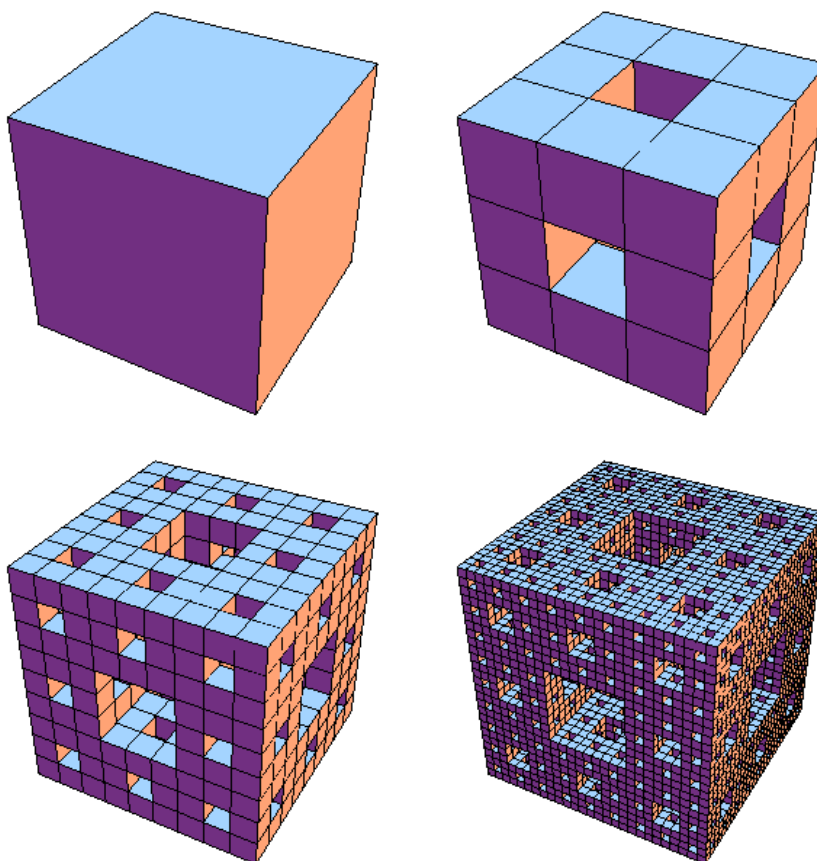
Obr. 31. Vizualizace 3D L-systému v prostředí Mathematica

4.2.1 Mengerova houba

Mengerova houba (Menger Sponge) byla popsána Karlem Menger v souvislosti s topologickou dimenzí.

Je univerzálním objektem obsahující ekvivalent všech myslitelných křivek v prostoru, kde všechny vhodně deformované křivky jsou identické s nějakou částí houby. Mengerova houba je tedy super-objektem pro všechny křivky. Lze ji pojímat také jako zobecněním Sierpinského koberce a Cantorova diskontinua [8, str. 14].

Mengerova houba vznikne ze základní krychle, která je rozdělena na 27 krychlí ($3 \times 3 \times 3$). Následně jsou odstraněny všechny středové krychle. Při dalších iteracích je tento krok aplikován opět na všechny plné krychle (Obr. 32).



Obr. 32. Čtyři iterace Mengerovy houby.

Při tvorbě v programu Mathematica byly vytvořeny nahrazovací pravidla (Obr. 33), kde 1 značí jednotkovou krychli a 0 prázdné místo o velikosti jednotkové krychle.

```
arrayCube2 = Nest[ArrayFlatten[# /.
  {1 -> {{1, 1, 1}, {1, 0, 1}, {1, 1, 1}},
    {{1, 0, 1}, {0, 0, 0}, {1, 0, 1}},
    {{1, 1, 1}, {1, 0, 1}, {1, 1, 1}}},
  0 -> {{{0, 0, 0}, {0, 0, 0}, {0, 0, 0}},
    {{0, 0, 0}, {0, 0, 0}, {0, 0, 0}},
    {{0, 0, 0}, {0, 0, 0}, {0, 0, 0}}}
  , 3] &, {{{1}}}, iteration];
```

Obr. 33. Ukázka nahrazovacího pravidla

Vlastnosti Mengerovy Houby:

- Topologická dimenze je rovna 3.
- Hausdorffova dimenze $= \frac{\ln 20}{\ln 3} = 2,7268$.
- Po nekonečně mnoha iteracích je objem nula a plocha nekonečno.

Výpočet povrchu (19):

$$S_0 = 6$$

$$S_{n+1} = \frac{S_n * 8 + 4 * 6 * 20^n}{9^n} \quad (19)$$

Výpočet objemu (20):

$$V_n = \left(\frac{20}{27}\right)^n \quad (20)$$

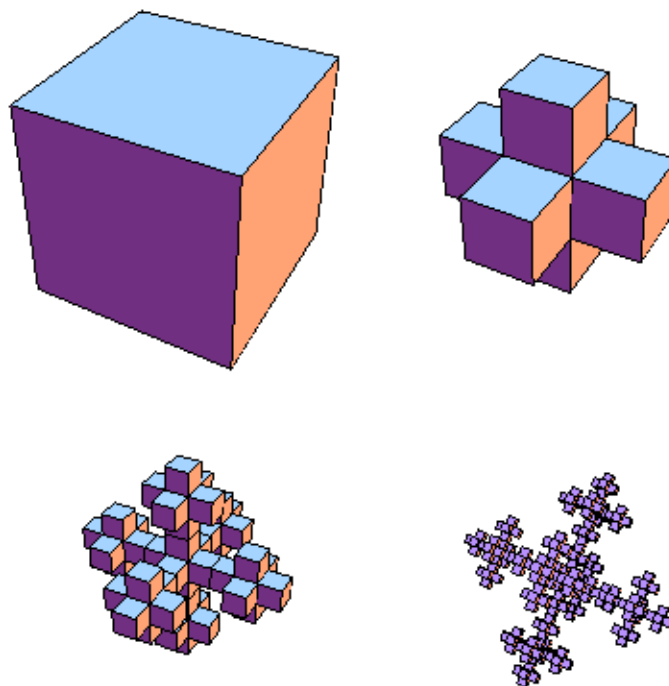
Souhrn pro 10 prvních iterací a 100., 1000. Iteraci (Tab. 6):

Iterace	Počet krychlí	Objem	Povrch
0	1	1	6,0
1	20	0,7407	72,0
2	400	0,5487	117,3
3	8,000E+03	0,4064	130,1
4	1,600E+05	0,3011	264,8
5	3,200E+06	0,2230	585,6
6	6,400E+07	0,1652	1300,7
7	1,280E+09	0,1224	2890,3
8	2,560E+10	0,0906	6422,8
9	5,120E+11	0,0671	14272,9
10	1,024E+13	4,97E-02	3,17E+04
100	1,268E+130	9,26E-14	5,15E+35
1000	1,071E+1299	4,63E-132	6,62E+345

Tab. 6. Iterace Mengerovy houby.

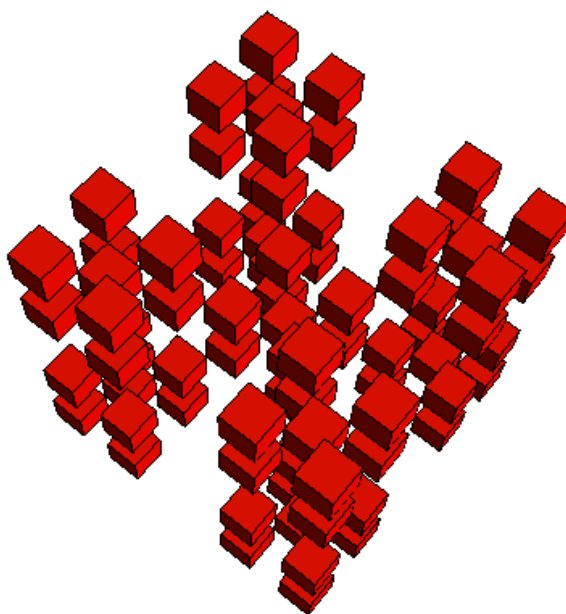
4.2.2 Vicsek fraktál

Při vytváření Vicsek fraktálu (Obr. 34) se vždy každá krychle nahradí 7 krychlemi ve tvaru kříže.



Obr. 34. Vicsek Fractal první čtyři iterace.

Další verzí Vicsek fraktálu (Obr. 35) je nahrazení za „vnější“ kříž.



Obr. 35. Druhá iterace Vicsek fraktálu.

4.2.3 Stromy

Fraktální stromy jsou v 3D prostoru vytvářeny stejně jako u 2D L-systémů pomocí želví grafiky.

Abeceda: $X, F, +, -, <, >, \&, ^$

4.2.4 Hilbertova křivka

Hilbertovu křivku (Hilbert curve) popsal německým matematikem David Hilbert v roce 1891. Hilbertova křivka (Obr. 36) je křivka vyplňující prostor, její fraktální dimenze je tedy rovna 3.

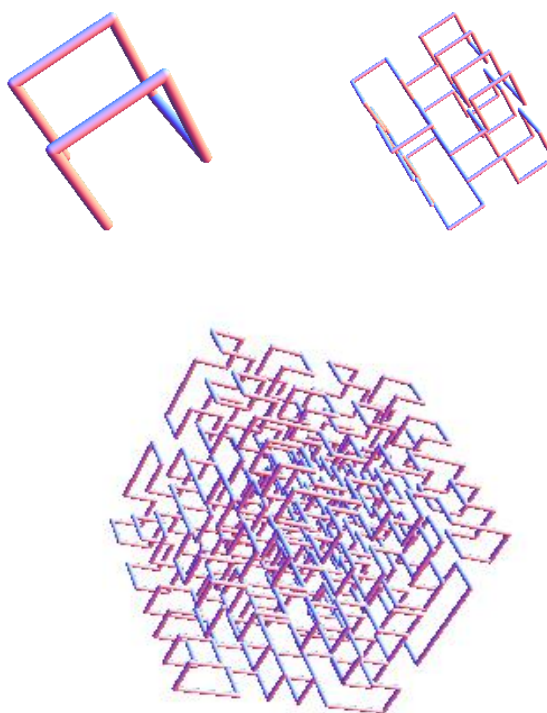
Abeceda: $X, F, +, -, <, >, \&, ^$

Úhel: 90°

Axiom: $<<< \&\& + XF^{\wedge} < XFX < \&F << X - F + + + XF < -XFX -$
 $F > X$

Přepisovací pravidla: $X \rightarrow ^ < XF^{\wedge} < XFX - F^{\wedge} >> XFX\&F+>> XFX - F > X->$

Fraktální dimenze: 3



Obr. 36. První 3. iterace hilbertovy křivky.

ZÁVĚR

Cílem práce bylo vysvětlení základních principů L-systémů a jejich konstrukce. L-systémy jsou skupinou fraktálů, proto je úvod práce věnován jejich obecnému popisu, základnímu rozdělení a historii. Jednou z vlastností fraktálu je dimenze a také jedna z definic fraktálů používá dimenzi jako nástroj k určení, zda se jedná o fraktální tvar. Část práce je proto věnována úvodu do této problematiky. V praktické části jsou u jednotlivých L-Systémů uvedeny jejich fraktální dimenze.

Nejznámější grafickou interpretací je interpretace pomocí želví grafiky. S novými požadavky na funkcionalitu se základní L-systémy začali rozšiřovat a dělit na několik základních typů: deterministické, rozvětřující, stochastické, kontextové, parametrické. Všechny jsou detailně popsány včetně dalších pod rozdělení a informacemi pro vykreslení v želví grafice.

Praktické využití L-systémů najdeme zejména v počítačové grafice, kdy pomocí různého počtu iterací můžeme vytvořit různě velké a tím pádem i odlišné objekty. Použitím stochastických systémů, které vnáší náhodu do generování objektu získáme přirozenější a reálnější objekty. Dalším zdokonalením může být použití tropismu, při němž jsou objekty generovány v závislosti na pravidlech daného prostředí (vítr, gravitace). Další zajímavé použití kde můžeme uplatnit znalosti L-systémů je návrh fraktálních antén s širokým pásmovým rozsahem či mapování vícerozměrných dat do jednorozměrného pole.

Praktická část spočívala v naprogramování a vizualizaci L-systémů v programovém prostředí Wolfram Mathematica. Program názorně zobrazuje postupy přepisovacích gramatik a vytváření L-systémů. Vypsány jsou aktuální řetězce pro první tři iterace. U vyšších iterací jsou často řetězce již velmi složité a nepřehledné, proto nejsou uváděny. Generování a zobrazení vyšších iterací je tedy časově a paměťově náročnější, proto musela být omezena maximální možná iterace u jednotlivých L-systémů. Důraz byl kladen na možnost vytváření vlastních L-systémů s různými možnostmi stylování, zejména pak barevného, tak aby výstupy byly použitelné při studiu či při zpracování seminární práci. Ve zjednodušené verzi je implementována také 3D vizualizace několika jednodušších L-systémů, neznámějším je Mengerova houba. Fraktál, který po nekonečně mnoha iteracích má nekonečný povrch a nulový objem.

SEZNAM POUŽITÉ LITERATURY

- [1] ZELINKA, Ivan. *Fraktální geometrie: principy a aplikace*. 1. vyd. Praha: BEN, 2006, 159 s. ISBN 80-730-0191-8.
- [2] Fraktály v počítačové grafice. Root [online]. 2005 [cit. 2013-05-09]. Dostupné z: <http://www.root.cz/clanky/fraktaly-v-pocitacove-grafice-ii>
- [3] Fraktály kolem nás. Root [online]. 2005 [cit. 2013-05-07]. Dostupné z: [http://www.root.cz/clanky/fraktaly-kolem-nas /](http://www.root.cz/clanky/fraktaly-kolem-nas/)
- [4] Fraktální geometrie. *Technická univerzita v Liberci: Fakulta strojní* [online]. 2010, 21.11.2011 [cit. 2013-01-13]. Dostupné z: <http://www.ksr.tul.cz/fraktaly/obsah.html>
- [5] Interaktivní editor afinních transformací. PAVEL TIŠNOVSKÝ. *Fit.vutbr* [online]. (c)1999,2000 [cit. 2013-01-13]. Dostupné z: <http://www.fit.vutbr.cz/~tisnovpa/publikace/diplomka/doc/ifscad.html>
- [6] Jemný úvod do fraktálů. *Martin Hinner* [online]. 1999 [cit. 2013-01-13]. Dostupné z: <http://martin.hinner.info/math/Fraktaly/>
- [7] ZIMMERMANN, Jan a Jan HRABINA. *Fraktály*. Zlín, 2003. Dostupné z: <http://mui.ic.cz/soubory/fraktaly.pdf>. Výukový materiál. Gymnázium Zlín.
- [8] WIESNER, Robert. *Užití a zneužití fraktálů*. Brno, 2006. Dostupné z: http://is.muni.cz/th/3506/fi_m/diplomka.pdf. Diplomová práce. MASARYKOVA UNIVERZITA FAKULTA PŘÍRODOVĚDECKÁ. Vedoucí práce RNDr. Zdeňek Pospíšil, Dr.
- [9] ZELINKA, Ivan. *Aplikovaná informatika aneb úvod do fraktální geometrie, buněčných automatů [2005]*. 2. vyd. Zlín: Univerzita Tomáše Bati, Fakulta technologická, 2005, 183 s. ISBN 8073182750.

- [10] Georg Cantor. *Osobnosti* [online]. © 1996–2013 [cit. 2013-01-14]. Dostupné z: <http://zivotopis.osobnosti.cz/georg-cantor.php>
- [11] GREGOROVÁ, Dagmar. Benoit B. Mandelbrot. *Objective Source E-Learning* [online]. 19.10.2010 [cit. 2013-01-14]. Dostupné z: <http://www.osel.cz/index.php?clanek=5346>
- [12] HERODES, Martin. *Virtuální Zahrada*. Brno, 2005. Dostupné z: http://is.muni.cz/th/44360/fi_m/diplomka.pdf. Diplomová Práce. MASARYKOVA UNIVERSITA FAKULTA INFORMATIKY. Vedoucí práce Doc. Ing. Jiří Sochora, CSc.
- [13] PRUSINKIEWICZ, Przemyslaw a Aristid LINDENMAYER. The algorithmic beauty of plants. New York: Springer-Verlag, c1990, xii, 228 p. ISBN 35-409-7297-8. Dostupné z: <http://algorithmicbotany.org/papers/abop/abop.pdf>
- [14] *Wolfram* [online]. c2013 [cit. 2013-01-17]. Dostupné z: <http://www.wolfram.com/>
- [15] Lindenmayer System. *Wolfram* [online]. 2008 [cit. 2013-02-06]. Dostupné z: <http://mathworld.wolfram.com/LindenmayerSystem.html>
- [16] Menger Sponge Project. *Menger Sponge Project* [online]. 2011, č. 1 [cit. 2013-03-18]. Dostupné z: http://amadeus.math.iit.edu/~fass/Project%20Description_Weening.pdf
- [17] Gosper Island. *Wolfram: mathworld* [online]. © 1999-2013 [cit. 2013-03-21]. Dostupné z: <http://mathworld.wolfram.com/GosperIsland.html>
- [18] GLENN, Elert. About Dimension. *Hypertextbook* [online]. © 1992-2008 [cit. 2013-03-22]. Dostupné z: <http://hypertextbook.com/chaos/>
- [19] Dimension. *Mathworld* [online]. ©1999-2013 [cit. 2013-03-23]. Dostupné z: <http://mathworld.wolfram.com/Dimension.html>

- [20] HilbertCurve. *Mathworld* [online]. © 1999-2013 [cit. 2013-03-30]. Dostupné z: <http://mathworld.wolfram.com/HilbertCurve.html>
- [21] Recognizing Plants Using Stochastic L-Systems. In: *University of Nebraska - Lincoln* [online]. 1994 [cit. 2013-03-31]. Dostupné z: <http://digitalcommons.unl.edu/cgi/viewcontent.cgi?article=1036&context=cseconfwork>
- [22] LAWDER, J K. *Using Space filling Curves for Multi-dimensional Indexing*. London, [2001]. Dostupné z: <http://www.dcs.bbk.ac.uk/TriStarp/pubs/bncod17.pdf>
- [23] Lebesgue 3D Curves. *Robert Dickau* [online]. © 2009–2013 [cit. 2013-04-02]. Dostupné z: <http://www.robertdickau.com/lebesgue3d.html>
- [24] Avatar. *Cgsociety* [online]. 2010 [cit. 2013-04-03]. Dostupné z: <http://www.cgsociety.org/index.php/CGSFeatures/CGSFeatureSpecial/avatar>
- [25] Závorkové a stochastické systémy. Root [online]. 2005 [cit. 2013-05-08]. Dostupné z: <http://www.root.cz/clanky/zavorkove-a-stochasticke-l-systemy/>
- [26] Přírodní objekty i umělé artefakty. Root [online]. 2005 [cit. 2013-05-07]. Dostupné z: <http://www.root.cz/clanky/l-systemy-prirodni-objekty-i-umele-artefakty/>

SEZNAM OBRÁZKŮ

Obr. 1. Cantorovo diskontinuum	12
Obr. 2. Kochova křivka, 4. Iterace.....	13
Obr. 3. Mandelbrotova množina [6].	14
Obr. 4. Princip TEA algoritmu pro Mandelbrotovu množinu [1].....	16
Obr. 5. Želví interpretace řetězce symbolů F, +, -. Úhel natočení je 90°.	23
Obr. 6. Ovládání želvy v prostoru.....	23
Obr. 7. Ukázka 3D Hilbertovy křivky.	24
Obr. 8. Stromová struktura [13].....	26
Obr. 9. Ukázka závorkového systému b) reprezentující Osový strom a). Úhel natočení $\delta = 45^\circ$	27
Obr. 10. Tři a), b), c) objekty stochastického systému se stejnými přepisovacími pravidly [21, str. 4].	29
Obr. 11. „Z-Curve“ a) [32] a Hilbertova křivka b). Mapování vícerozměrného pole do jednorozměrného.	33
Obr. 12. Vizualizace L-systému v prostředí Mathematica.	36
Obr. 13. Zápis v programu Mathematica.....	39
Obr. 14. Ukázka interpretace symbolů v programu Mathematica.	39
Obr. 15. Možné vykreslení křivky v programu Mathematica.....	39
Obr. 16. Tvorba Kochovy vločky.	40
Obr. 17. Čtyři iterace Kochovy vločky.....	41
Obr. 18. Axiom a) a první iterace b) Sierpinského trojúhelníku. Šipky značí aktuální směr vykreslování.....	42
Obr. 19. Nultá a) a první b) iterace „32 - segment curve“.....	43
Obr. 20. Gosper curve.....	44
Obr. 21. Gosperův ostrov v 3. Iteraci.	45
Obr. 22. Sierpinského křivka	45
Obr. 23. 6. Iterace Hilbertovy křivky.....	46
Obr. 24. Křivka Moore curve s vyznačením směru průchodu.....	47
Obr. 25. Iterace dračí křivky a) 0 iterace b) 2 iterace c) 4 iterace d) 6 iterace e) 8 iterace f) 10 iterace g) 12 iterace h) 14 iterace.	48
Obr. 26. Prvních šest iterací Box Fraktálu.....	49
Obr. 27. Box fraktál a volba úhlu 60°.....	49

Obr. 28. Nahrazovací pravidla pro L-systém Strom.	50
Obr. 29. Strom v 6. Iteraci. Konstantní úhel 45° a). Náhodný úhel 35° - 55° b).....	51
Obr. 30. Příklad fraktálního stromu ve 4. iteraci. Axiom: FX , přepisovací pravidla: $F \rightarrow FF - -F + F + +F - F, X \rightarrow FF + +F + [-F]$	51
Obr. 31. Vizualizace 3D L-systému v prostředí Mathematica	52
Obr. 32. Čtyři iterace Mengerovy houby.	53
Obr. 33. Ukázka nahrazovacího pravidla.....	53
Obr. 34. Vicsek Fractal první čtyři iterace.....	55
Obr. 35. Druhá iterace Vicsek fraktálu.	55
Obr. 36. První 3. iterace hilbertovy křivky.	56

SEZNAM TABULEK

Tab. 1. Příkazy pro pohyb želvy v rovině [13, str. 6-11].....	22
Tab. 2. Příkazy pro pohyb želvy v prostoru [13, str. 19].....	24
Tab. 3. Interpretace symbolů pro závorkové L-systémy. Doplnění (Tab. 1.) [13]	26
Tab. 4. Některé parametrické znaky a jejich význam [13, str. 46].	30
Tab. 5. Symboly želví grafiky pro kreslení polygonů [13, str. 119-128].	31
Tab. 6. Iterace Mengerovy houby.	54

SEZNAM PŘÍLOH

K bakalářské práci je přiloženo CD s prací v elektronické podobě a program vytvořený ve Wolfram Mathematica.