Systémy s umělou inteligencí

4. Umělý život

Jiří Kubalík
Katedra kybernetiky, ČVUT-FEL

Tato prezentace z velké části čerpá z publikace
Czontó J. - Palko M.: *Umělý život*
elfa, Košice, 2002.

http://cw.felk.cvut.cz/doku.php/courses/A7B33SUI/start

Umělý život a jeho cíle

Umělý život (*Artificial Life* - **ALife**) – návrh jednoduchých umělých entit, definice jejich vzájemných interakcí a jejich intrerakcí s okolím a pozorování chování takovýchto systémů. Výsledné chování má zpravidla **emergentní charakter** – z jednoduchých elementů a vztahů vzniká složité chování (neplatí princip superpozice, kdy je možné odvodit chování celku složením částečných chování jeho primitiv).

Cíle umělého života

- Inspirovat biologii – navrhovat nové hypotézy o procesech přirozeného života.
- **Inspirovat se přírodou** – vytvářet umělé systémy, které jsou schopny díky některé vlastnosti, charakteristické pro živé objekty, přispět k efektivnímu řešení praktických úloh.

Metody ALife

- Celulární automaty
- Lindenmayerovy systémy
- *Swarm intelligence*
- Evoluční algoritmy
- Neuronové sítě
- Swarm-bots – kolektivní chování skupiny robotů
- . . .

Celulární automaty

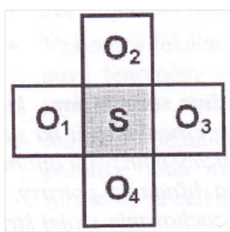
Celulární automat (*Cellular Automata* - CA) – dynamický systém, diskrétní v čase i prostoru.

- Pravidelná (nekonečná) **struktura buněk** v N-rozměrném prostoru,
- Každá buňka se může nacházet v jednom z K **stavů**.

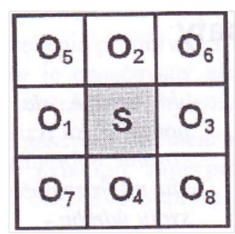
  Často pouze dva stavy: 0 – mrtvá buňka, 1 – živá buňka

  Klidové stavy – pokud má buňka v okolí pouze klidové stavy, tak se její stav nemění.
- Hodnoty stavů buněk v dalším časovém kroku se vypočítají paralelně na základě **lokální přechodové funkce** $S(t)$ (ta je stejná pro všechny buňky).

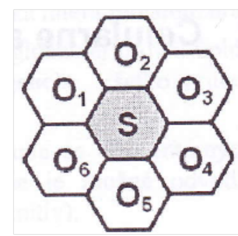
  Argumenty této funkce jsou aktuální hodnoty stavů dané buňky a všech uvažovaných sousedních buněk (**okolí**).



neumannovské okolí



mooreovské (úplné) okolí



šestiúhelníkové okolí

Lokální přechodová funkce, která bývá definována sadou pravidel, má tvar

$$S(t+1) = f(S(t), O_1(t), O_2(t), \dots)$$

kde $O_1(t) \dots O_n(t)$ jsou okolní buňky dané typem uvažovaného okolí.

Celulární automaty

Vlastnosti CA

- **Paralelismus** – výpočet nových hodnot stavů všech buněk probíhá současně.
- **Lokalita** – nový stav prvku závisí pouze na jeho počátečním stavu a stavu okolních buněk.
- **Homogenita** – pro všechny buňky platí stejná lokální přechodová funkce.

Von Neumannův CA

Von Neumannův CA – první samoreplikující se CA:

- Šachovnicová 2D struktura,
- 200 000 buněk, 29 stavů, každá buňka reprezentuje konečný automat
  tělo – továrna, duplikátor, počítač,
  výrustek – analogie pásky Turingova stroje; program, který řídí proces replikace
- Na konci replikace se informace přenese na potomka a ten se oddělí.

Důkaz existence tohoto CA a jeho implementace v r. 1964 resp. 1995.

Hra života

Game of life (**LIFE**) – Implementace zjednodušené verze 2D von Neumannova CA

- 2 stavy – buňka a prázdné políčko,
- úplné okolí,
- pravidla lokální přechodové funkce
  - zrod – v okolí prázdného políčka jsou právě 3 buňky.
  - přežití – v okolí buňky jsou 2 nebo 3 buňky.
  - uhynutí – v okolí buňky je 0, 1, 4, 5, 6, 7 nebo 8 dalších buněk.

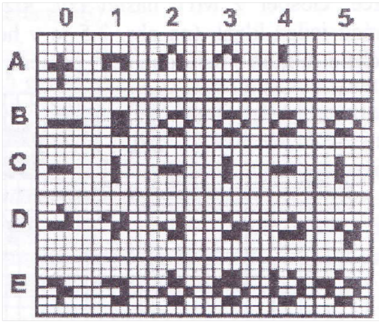
Hra života

Game of life (**LIFE**) – Implementace zjednodušené verze 2D von Neumannova CA

- 2 stavy – buňka a prázdné políčko,
- úplné okolí,
- pravidla lokální přechodové funkce
  - zrod – v okolí prázdného políčka jsou právě 3 buňky.
  - přežití – v okolí buňky jsou 2 nebo 3 buňky.
  - uhynutí – v okolí buňky je 0, 1, 4, 5, 6, 7 nebo 8 dalších buněk.

Zajímavá dynamika systému, který z počátečních obrazců směřuje k některé z následujících situací:

- zánik – poněkolika generacích struktura zanikne (A).
- stabilní – po několika generacích dojde k ustálení struktury (B).
- cyklicky se opakující obrazec (C).
- cyklicky se opakující posunutý obrazec, tzv. **kluzák** (D).



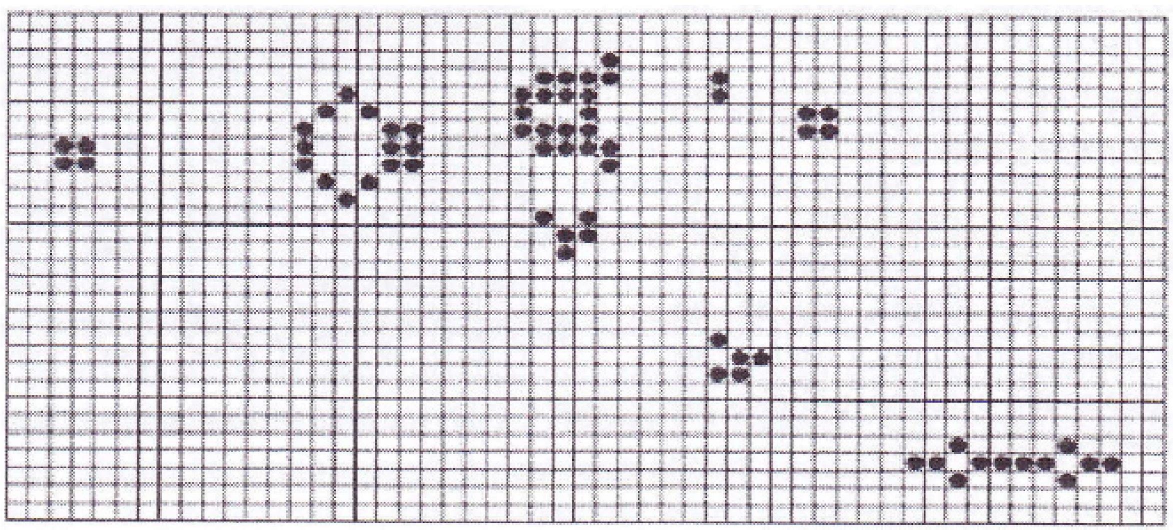
LIFE

Kluzákové dělo – generátor kluzákových obrazců

- Přemísťování obrazce $\longrightarrow$ přenos informace.
- Tato schopnost je nezbytná, aby CA mohl emulovat chování Turingova stroje.
- Existují konfigurace kluzáků, ze kterých se vytvoří kluzákové dělo.

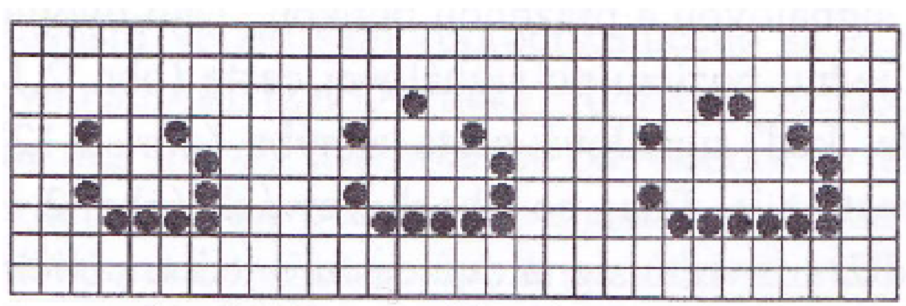
Příklad:

- Struktura nahoře produkuje kluzák v každé 30. generaci.
- Vpravo dole je cyklická struktura, s periodou 15, která požírá přicházející kluzáky.

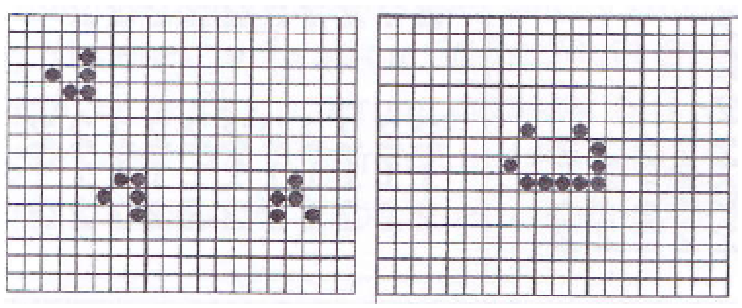


LIFE

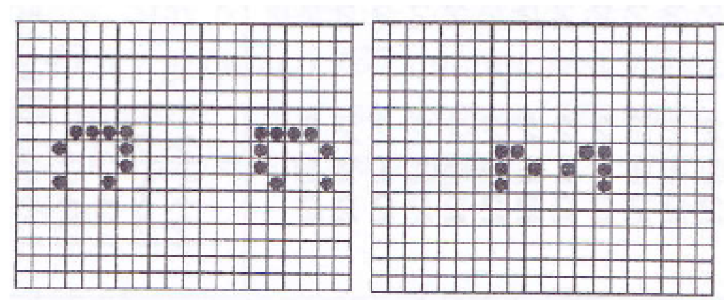
Lode



kluzáky $\longrightarrow$ lodě (14 generací)



lodě $\longrightarrow$ kluzáky (10 generací)



- Byla dokázána **výpočetní univerzálnosti LIFE**, která se opírá právě o existenci kluzáků, jako nositelů informace.
- V LIFE byla sestrojena sčítačka.
- Nepodařilo se najít konečnou samoreprodukující se strukturu.

Coddův automat

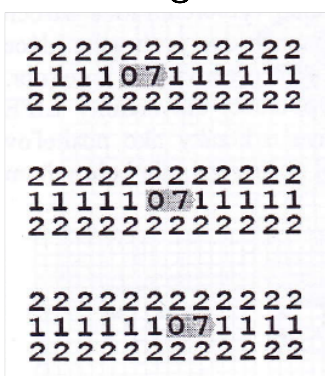
Coddův automat:

- Automat s neumannovským okolím.
- 8 stavů – 0-prázdná buňka; 1-signálová cesta; 2-obal signálové cesty; 3-speciální použití; 4, 5, 6 a 7 jsou signálové stavy.
- 500 použitých pravidel.

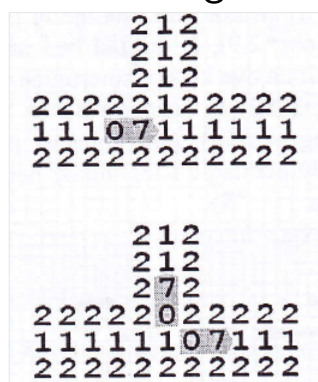
  Kolik je všech možných pravidel?
- Základním informačním prvkem je dvojice *signálová-prázdná* buňka.

  V každé iteraci se posune o jednu pozici po signálové cestě.

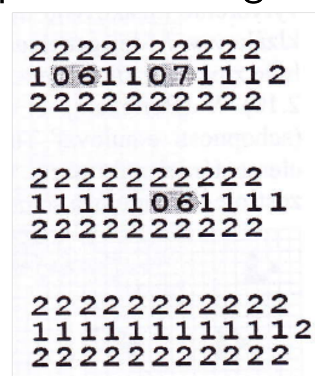
tok signálu



větvení signálu



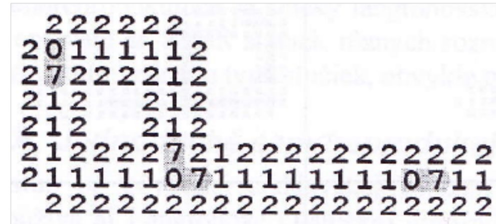
prodloužení signálu



Coddův automat

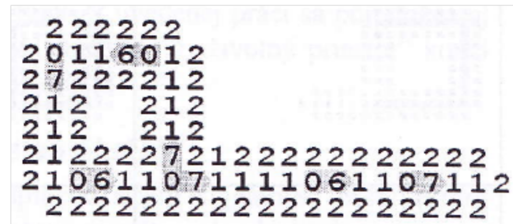
Synchronizační smyčka

- Dvojice kroužicích signálů 07 vysílá do nekonečného ramena v každé 10. generaci jeden signál 07.



Konstruktér signálové cesty

- Dvojice signálů 06–07 postupně prodlužuje ukončené rameno.



Coddův automat je teoreticky schopný

- **emulovat Turingův stroj** i
- **vytvořit svoji vlastní kopii**.

Celulární automaty: Aplikace

Paralelní celulární počítač (Cellular Automata Machine - CAM) – HW realizace

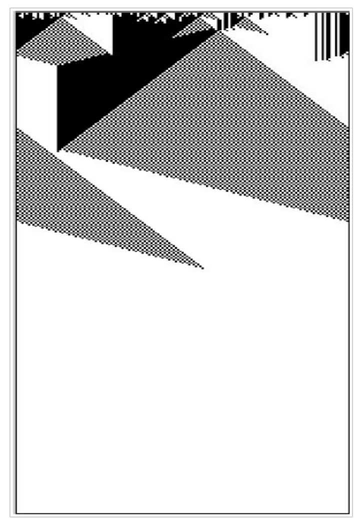
- 500 miliónů buněk.
- Výpočet 3 miliard změn stavů za sekundu.
- Aplikace – dynamika tekutin a plynů, dynamika chemických reakcí, zpracování a vizualizace 3D obrazu.

Klasifikace hustoty – cílem je pro daný binární vektor rozhodnout, zda obsahuje více jedniček nebo nul (bez použití centrálního řízení výpočtu).

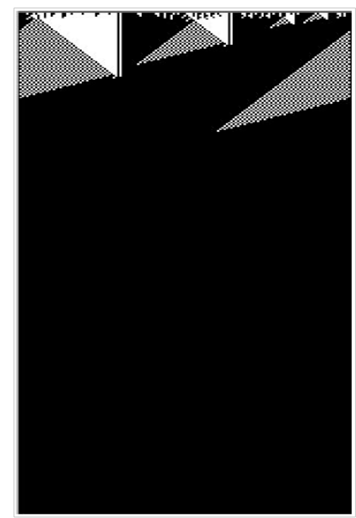
- Dvoustavový 1D CA velikosti n s lokálním okolím $r \ll n$.
- Problém nelze perfektně řešit tímto typem CA.
- Je těžké nalézt pravidla, která umožní emergenci požadovaného globálního chování – správné klasifikace hustoty.

  Evoluční návrh CA.

časoprostorový diagram chování 1D CA



klasifikace "0"

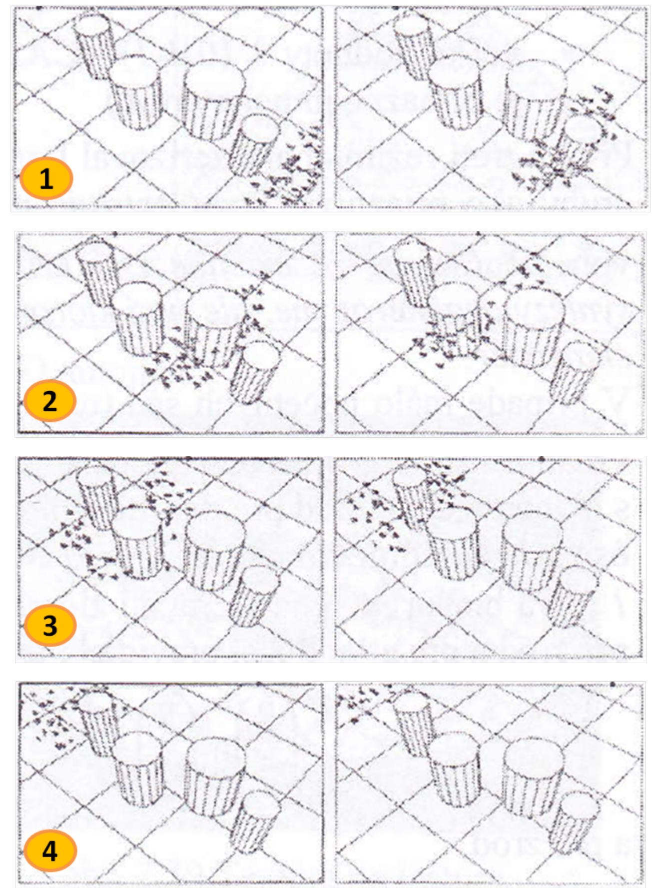


klasifikace "1"

Reynoldsův model shlukování ptáků

Reynolds (počítačový animátor) – simulace chování hejna ptáků, které vzniká emergentně na základě vzájemného působení jedinců řízených jednoduchými lokálními pravidly

- **Separace** – vyhýbání se výraznému nahuštění sousedních jedinců.
- **Nasměrování** – natáčení se ve směru průměrné orientace sousedních jedinců.
- **Soudržnost** – směrování k těžišti sousedních jedinců.



Počítačový model **Boid**

- Dokonalá simulace – ornitologové přijímají hypotézu řízení chování hejna ptáků pomocí analogických pravidel.
- Využití v technických aplikacích pro simulaci pohybu vozidel v přehuštěné dopravě.
- Na základě tohoto modelu byly vytvořeny animace ve filmech Batman se vrací, Lví král, Jurský park.

Celulární automaty: Aplikace

- Modelování hydrodynamiky Benátské laguny (predikce přílivových vln) – 2D CA se čtvercovou mřížkou, s různými typy okolí a různými lokálními přechodovými funkcemi (respektující fyz. zákony).
- Modelování dynamiky dopravy – simulace dopravy v Ženevě.
- Sledování toku dat v Internetu.
- Sledování šíření seizmických vln a při studiu příčin a průběhu zemětřesení.
- Aplikace v lesním hospodářství
  - Modelování šíření lesního porostu.
  - Modelování odumírání lesa.
  - Modelování dynamiky lesních požárů.
  - Modelování zamoření lesa škůdci.
- Použití CA v kryptografii.
- Simulování reakce protilátek v tkáni při napadení virem HIV.
- ...

Lindenmayerovy systémy

Lindenmayerův systém (L-systém) je varianta formální gramatiky vyvinutá **pro modelování vývoje vícebuněčných organismů**.

L-systém je trojice $\langle V, A, P \rangle$, kde

- V je množina **symbolů**,
- A je výchozí neprázdné slovo (**axióm**),
- P je množina pravidel tvaru $X \to S$, kde X je symbol a S je slovo.

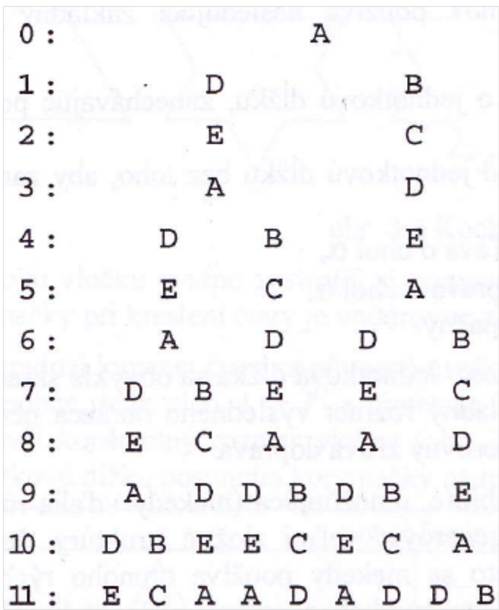
  Slovo je řetězec symbolů.

  V případě deterministického L-systému existuje pro každý symbol právě jedno pravidlo (pokud není explicitně uvedené, tak se předpokládá $X \to X$).

  Přepisování symbolů tvořících slovo na pravé straně pravidla **probíhá paralelně**.

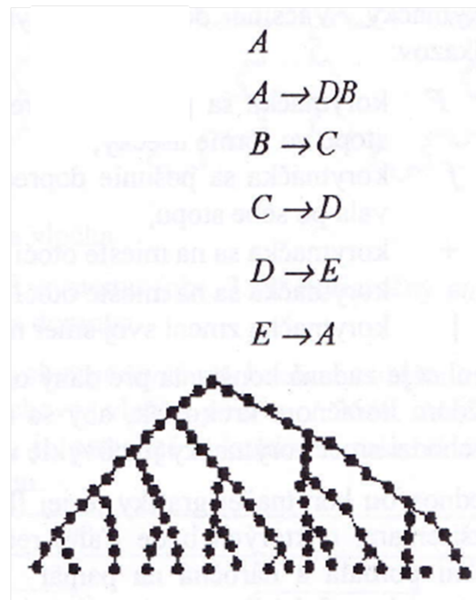
Lindenmayerovy systémy: Růst řasy Chaetomorpha linum

- Znaky A - D reprezentují různé etapy vývoje buňky.
- Pravidla reprezentují přechody mezi různými etapami vývoje buňky.



0 : A
1 : D B
2 : E C
3 : A D
4 : D B E
5 : E C A
6 : A D D B
7 : D B E E C
8 : E C A A D
9 : A D D B D B E
10 : D B E E C E C A
11 : E C A A D A D D B

Posloupnost řetězců popisující růst řasy. V daném kroku jsou všechny znaky slova paralelně přepsány podle daných pravidel gramatiky.



Gramatika daného L-systému s počátečním slovem (axiómem) A. Grafická interpretace posloupnosti slov.

L-systémy: Vykreslování

Želví grafika – způsob geometrické interpretace řetězců vyvíjeného L-systému.

Jednotlivé symboly řetězce se chápou jako příkazy pro řízení pohybu želvy:

F	želva se posune dopředu o jednotkovou délku, přičemž za sebou zanechá stopu ve formě úsečky,
f	želva se posune dopředu o jednotkovou délku, ale nezanechá za sebou stopu,
$+$	želva se na místě otočí doleva o úhel α,
$-$	želva se na místě otočí doprava o úhel α,
$\vert$	želva provede "čelem vzad".

kde

- úhel α je konstanta pro daný obrazec,
- jednotková délka se zkracuje v každém iteračním kroku tak, aby se základní rozměr výsledného obrazce nezměnil.

Fraktály

Fraktál je geometrický útvar s těmito vlastnostmi:

- **Sebepodobnost** – pozorovaný v jakémkoliv měřítku, vykazuje stejný charakteristický tvar.
- **Neceločíselná dimenzita** D – dimenzita je definovaná jako $N = L^D$, kde N je počet kopií původního útvaru při L-násobném zvětšení lineárního rozměru.

  Např.: Čtverec má dimenzitu 2, protože $4 = 2^2$.
- **Složitost** – má na pohled složitý tvar, ale je generovaný jednoduchými pravidly.

Fraktály

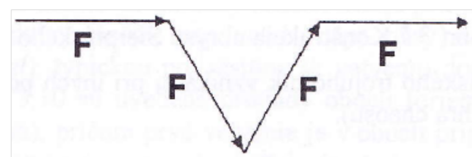
Fraktál je geometrický útvar s těmito vlastnostmi:

- **Sebepodobnost** – pozorovaný v jakémkoliv měřítku, vykazuje stejný charakteristický tvar.
- **Neceločíselná dimenzita** D – dimenzita je definovaná jako $N = L^D$, kde N je počet kopií původního útvaru při L-násobném zvětšení lineárního rozměru.

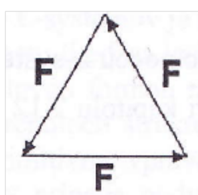
  Např.: Čtverec má dimenzitu 2, protože $4 = 2^2$.
- **Složitost** – má na pohled složitý tvar, ale je generovaný jednoduchými pravidly.

Příklad: **Kochova vločka**

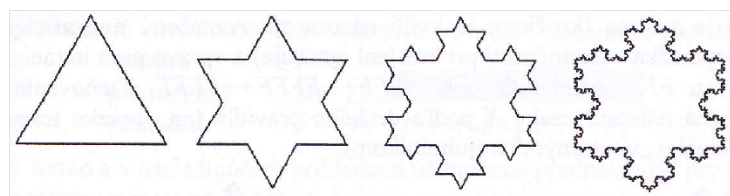
- vychází z rovnostranného trojúhelníku,
- v každé iteraci se nahradí všechny úsečky lomenou čarou,
- $L = 3$, $N = 4$, $D = 1.2618$,



pravidlo: $\mathbf{F} \longrightarrow \mathbf{F} - \mathbf{F} + +\mathbf{F} - \mathbf{F}$



axióm: $\mathbf{F} + +\mathbf{F} + +\mathbf{F}$

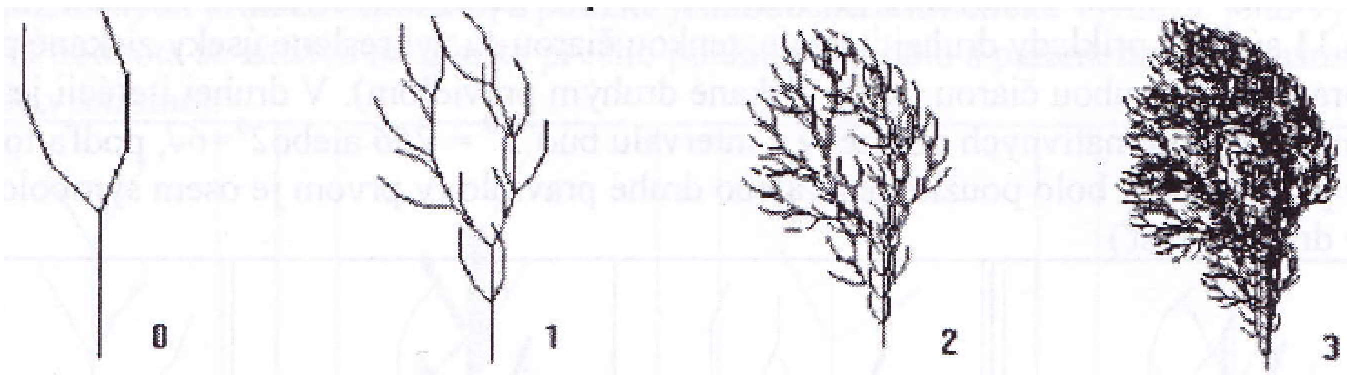


První čtyři iterace.

L-systémy: Variranty

Zásobníkové L-systémy – pro rozvětvené struktury, které mají rekurzivní charakter.

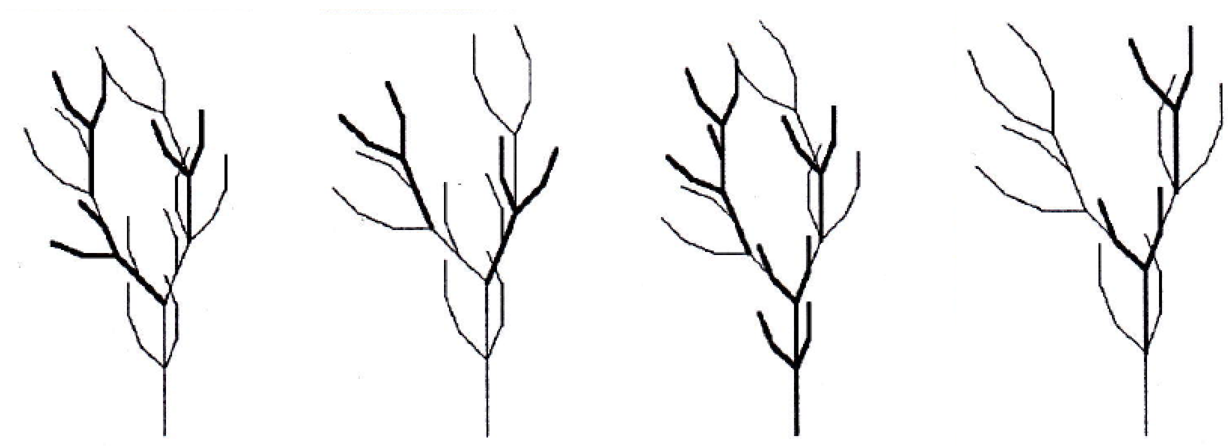
axióm:	$\mathbf{F}$
otočení:	$\alpha = 22.5°$
pravidlo:	$\mathbf{F} \rightarrow \mathbf{FF} + [+\mathbf{F} - \mathbf{F} - \mathbf{F}] - [-\mathbf{F} + \mathbf{F} + \mathbf{F}]$



L-systémy: Varianty

Stochastické L-systémy – realističtější simulace. Pro každý symbol může existovat více pravidel, každé pravidlo má přiřazenu pravděpodobnost aplikace.

axióm:	$\mathbf{F}$
otočení:	$\alpha = 22.5°$
pravidlo:	$\mathbf{F} \to (0.5)\mathbf{FF} + [+\mathbf{F} - \mathbf{F} - \mathbf{F}] - [-\mathbf{F} + \mathbf{F} + \mathbf{F}]$
	$\mathbf{F} \to (0.5)\mathbf{FF} + [+\mathbf{F} - \mathbf{F}] - [-\mathbf{F} + \mathbf{F}]$



4 různé obrazce jako výsledek 4 různých evolucí stejného L-systému

L-systémy: Varianty

Parametrické L-systémy – používají číselné parametry udávající hodnotu posunu, velikost natočení, tloušťku a barvu vykreslené čáry apod.

Kontextové L-systémy – výběr pravidla je závislý na kontextu nahrazovaného symbolu (symbolech před a za ním).

3D L-systémy – rozšíření množiny symbolů o příkazy pro pohyb ve 3D.

L-systémy: Aplikace

- Vizualizace hypotéz o morfologii a fyziologii rostlin.
- Studium vnějších vlivů na růst rostlin.
- Protein folding.
- ...

Deformace koruny stromu při soupeření o světlo.

Swarm Intelligence

Swarm intelligence: "Any attempt to design algorithms or distributed problem-solving devices inspired by the collective behavior of social insect colonies and other animal societies"

From "Bonabeau E., M. Dorigo & G. Theraulaz: Swarm Intelligence: From Natural to Artificial Systems, Oxford University Press, 1999"

Swarm intelligence is an artificial intelligence technique based around the study of collective behavior in decentralized, self-organized systems.

Swarm intelligence systems are typically made up of a population of simple agents interacting locally with one another and with their environment.

Although there is normally no centralized control structure dictating how individual agents should behave, local interactions between such agents often lead to the emergence of global behavior.

From "Wikipedia, Swarm Intelligence"

Příklady technik:

- *Particle Swarm Optimization* (PSO)
- *Ant Colony Optimization* (ACO)
- ...

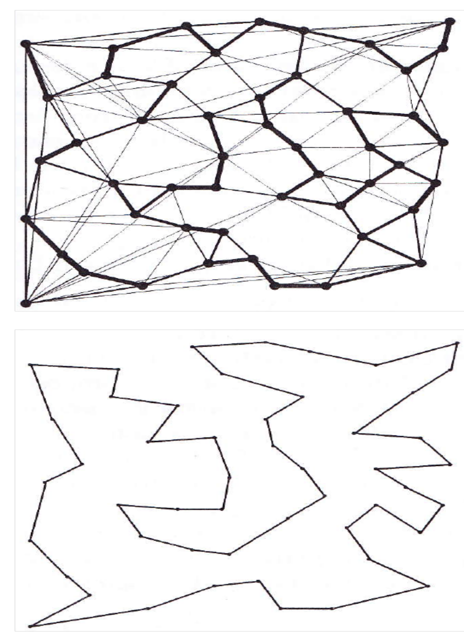
Ant Colony Optimization

Ant Colony Optimization (ACO) – distribuovaný optimalizační algoritmus inspirovaný kolektivním chováním mravenců směřující k zachování kolonie.

- Rozšíření tradičních **konstruktivních heuristik** o schopnost využití **zkušeností získaných v průběhu výpočtu**.
- **Stigmergie** - nepřímá komunikace mravenců zprostředkovaná pomocí feromonu. Množství zanechaného feromonu je funkcí kvality nalezeného řešení.

ACO může být použito pro řešení jakéhokoliv diskrétního optimalizačního problému, pro který lze použít nejakou konstruktivní heuristickou proceduru

- Traveling salesman problem
- Quadratic assignment problem
- Job-shop scheduling problem
- Vehicle routing problem
- Shortest common supersequence problem
- Network routing problem
- ...

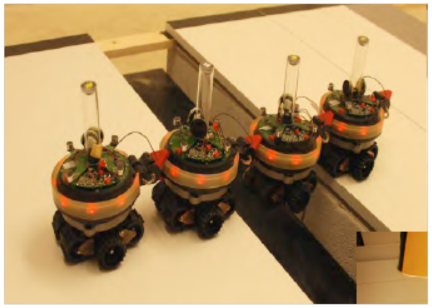
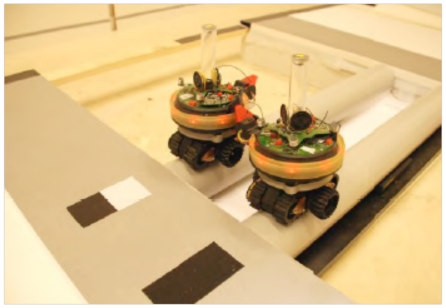


Ukázka ACO na TSP

Swarm-bot: Kolektivní chování skupiny robotů

Swarm-bot – kolektivní fyzické struktura složená z tzv. *s-botů*, robotů se základními senzory a pohonnou jednotkou, vybavenými jednoduchými pravidly chování.
Swarm-bot vykazuje schopnost **samo-rekonfigurace** s ohledem na splnění daného úkolu

- transport nákladu, který je příliš velký/těžký na jeden *s-bot*,
- překonání překážky, kterou samotný *s-bot* překonat nedokáže.

©Dorigo, M.: SWARM-BOT: An experiment in swarm robotics, 2005.

©Dorigo, M.: Swarm-bots and Swarmanoid: Two experiments in embodied swarm intelligence, 2009.

Odkazy: `http://www.swarm-bots.org/`, `http://www.swarmanoid.org/`