

Monte Carlo

Simulační metoda založená na užití stochastických procesů a generace náhodných čísel.

Minule generování náhod. čísel.

Typy MC simulací

- a) **Geometrické MC** (náhodné procházky, perkolace, celulární automaty, ...)
- b) **MC integrace**
- c) Termodynamické MC
- d) Modelování vývoje na strukturální úrovni
- e) Výpočet kinetických koeficientů (KMC)
- f) etc.

Perkolace

- Jednoduchý fyzikální problém vykazující kritické chování - geometrický fázový přechod.
- Problém s náhodností na vybrané mřížce. Konfigurace jsou tvořeny, tak že elementární objekty na mřížce jsou s pravděpodobností p z intervalu $(0,1)$ obsazené nebo prázdné.
- Různé mřížky, ilustrativní čtvercová -2D.

Příklady konfigurací

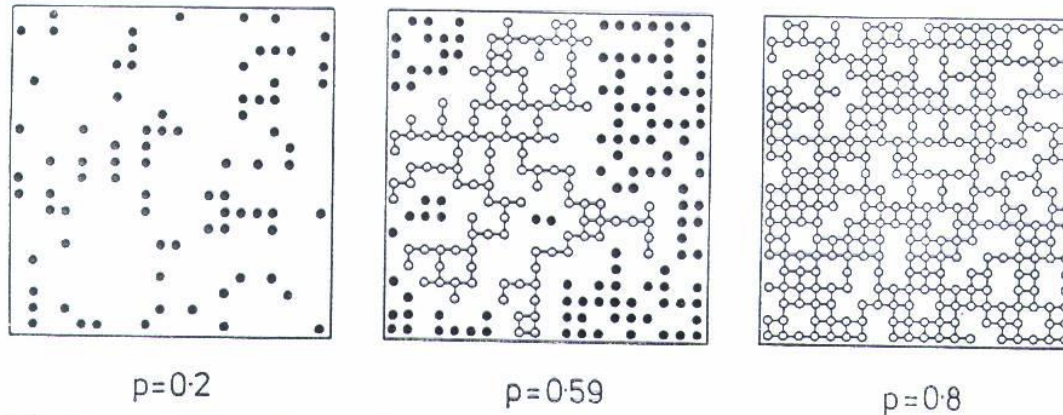


Fig. 1.16. Square lattice of size 20×20 . Sites have been randomly occupied with probability p ($p = 0.20, 0.59, 0.80$). Sites belonging to finite clusters are marked by full circles, while sites on the infinite cluster are marked by open circles

Různé typy:

- 1) uzlová (site)
- 2) vazbová (bond)

Práh perkolace

- Pro různé pravděpodobnosti p vznikají různě velké souvislé klastry.
- Pro malé pravděpodobnosti p dostáváme malé izolované klastry, pro p blízko 1 vzniká klastr přes celou mřížku – perkolační klastr.
- Vše je třeba **vyhodnocovat v pravděpodobnostním smyslu**, tj. vyhodnocovat pravděpodobnosti jevů.
- Zřejmě existuje hodnota pravděpodobnosti p_c , kdy se objeví perkolační klastr nazývá se **práh perkolace**.

Aplikace

- Průtok částic porézním médiem
- Depozice vodivých elementů
(přechod vodič – izolant)
- Zředěná magnetika
- Elasticita gelových polymerů
- ...

Pro reálné výpočty

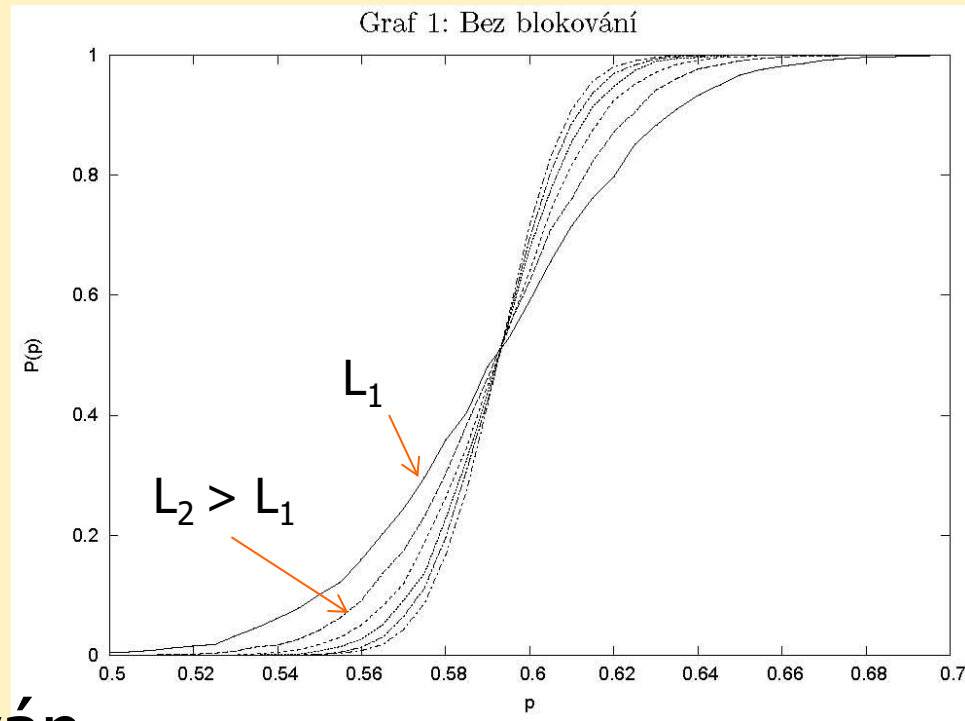
- Konečná mřížka velikosti L
pravděpodobnosti of clusters

$$p_{(c)} \dashrightarrow p_{(c)}^L$$

Pro reálné výpočty pokračování

Měříme pravděpodobnost vzniku perkol. klastru pro různé velikosti systému L v závislosti na pravděpodobnosti p .

Typický průběh:

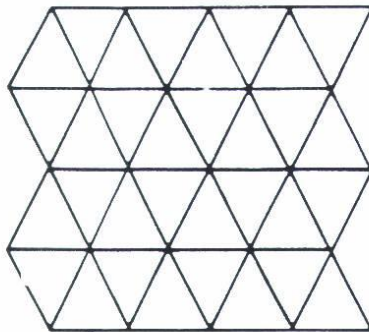


Práh perkolace určen
jako průsečík křivek

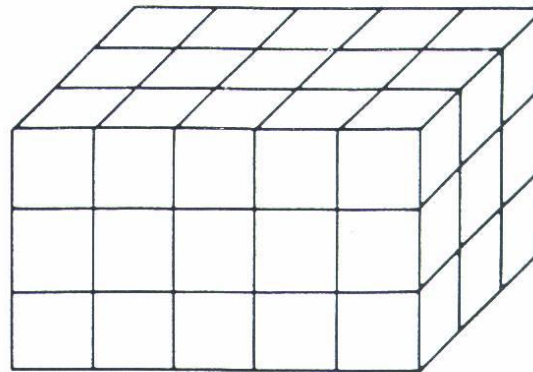
Prahy perkolace

Selected percolation thresholds for various lattices. 'Site' refers to site percolation and 'bond' to bond percolation. In all cases, only nearest neighbours form clusters, and no correlations are allowed between different sites or bonds. If the result is not exact (see text), the error probably affects only the last decimal.

Lattice	Site	Bond
Honeycomb	0.6962	0.65271
Square	0.592746	0.50000
Triangular	0.500000	0.34729
Diamond	0.43	0.388
Simple cubic	0.3116	0.2488
BCC	0.246	0.1803
FCC	0.198	0.119
$d = 4$ hypercubic	0.197	0.1601
$d = 5$ hypercubic	0.141	0.1182
$d = 6$ hypercubic	0.107	0.0942
$d = 7$ hypercubic	0.089	0.0787



(a)



(b)

Prahy perkolace

- Platí $p_c^{vazb.} < p_c^{uzl.}$

Perkolace pokračování

analytické metody

Introduction to percolation theory,
D. Stauffer and A. Aharony,
Taylor and Francis, Second Edition
London 1992

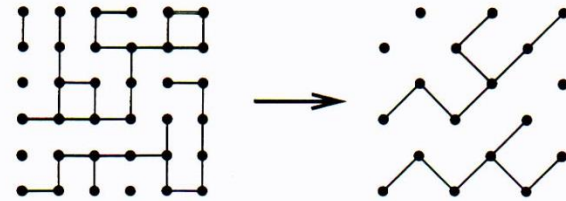
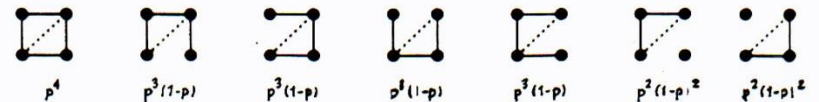
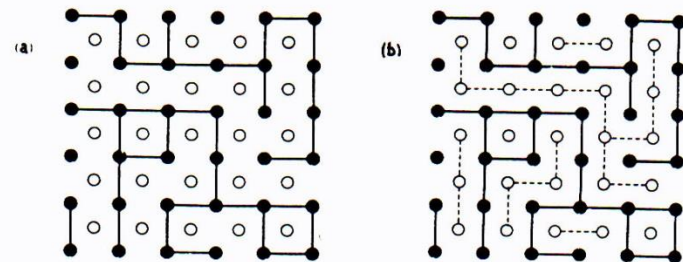


Illustration of the renormalization of the percolation problem on a square lattice. Bridges are erected on the new lattice along bonds that were connected by two bridges on the old.



The configurations of bridges around a square on the unrenormalized lattice that can give rise to a bridge along the dotted line on the renormalized lattice. Each configuration is labelled with the probability of its occurrence.

$$p_c = \frac{\sqrt{5}-1}{2}$$



$$p_c = \frac{1}{2}$$

(a) The construction of the 'dual lattice' for the percolation problem on a square lattice. (b) We populate the dual lattice by building bridges between every pair of sites not separated by a bridge on the original lattice.



Approximations arising in the renormalization scheme we used for the decimation problem. The sites marked A and B, which are connected on the original lattice, become unconnected on renormalization. This effect has a tendency to decrease the average cluster size ξ . Also, correlations arise between the probabilities of adjacent bonds having bridges on them. The presence or absence of a bridge on the bond marked 1, for

Hoshenův-Kopelmanův algoritmus

J. Hoshen R. Kopelman, Phys. Rev. B 14 (1976) 3428

Počítáme počet klastrů po řádcích.

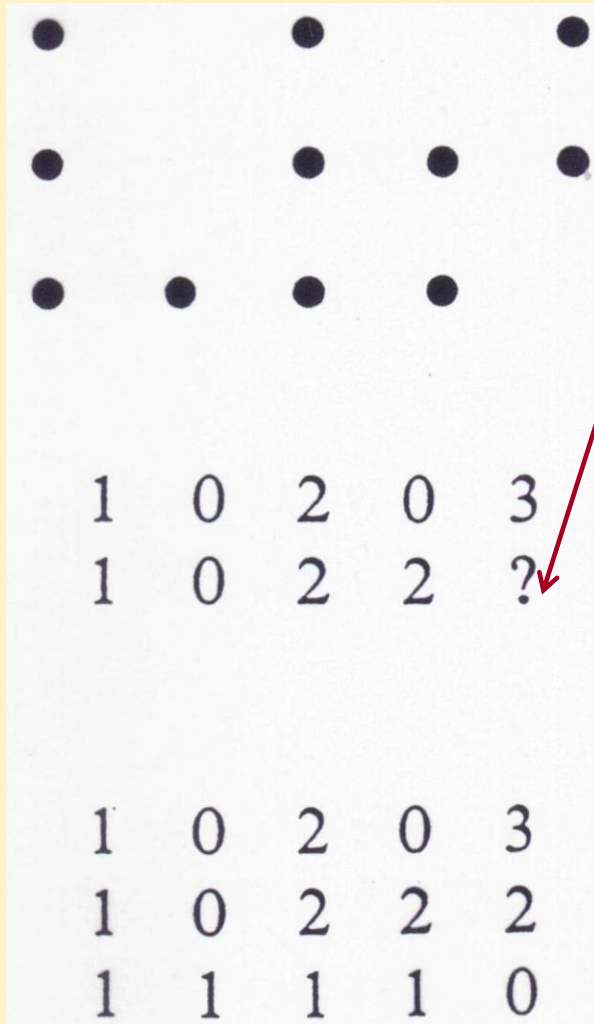
Konflikt při číslování vyřešíme

zavedením další proměnné -
značky pro postupné číslování
klastrů.

$$N(3) = 2 \quad N(2) = 1 \quad N(1) = 1$$

Výsledek po přečíslování:

1	0	1	0	1
1	0	1	1	1
1	1	1	1	0



Úlohy perkolace

I-1 Uzlová perkolace – práh perkolace

Určete práh uzlové perkolace pro danou mřížku. Pomocí výpočtu pravděpodobnosti existence perkolačního klastru (tj. spanning klastru) v závislosti na pravděpodobnosti p obsazení uzlu pro různé velikosti L mřížky odhadněte práh perkolace p_c .

Základní varianta: Uvažujte čtvercovou mřížku $L \times L$ pro $L = 16, 32, 64, 128 (\dots)$.

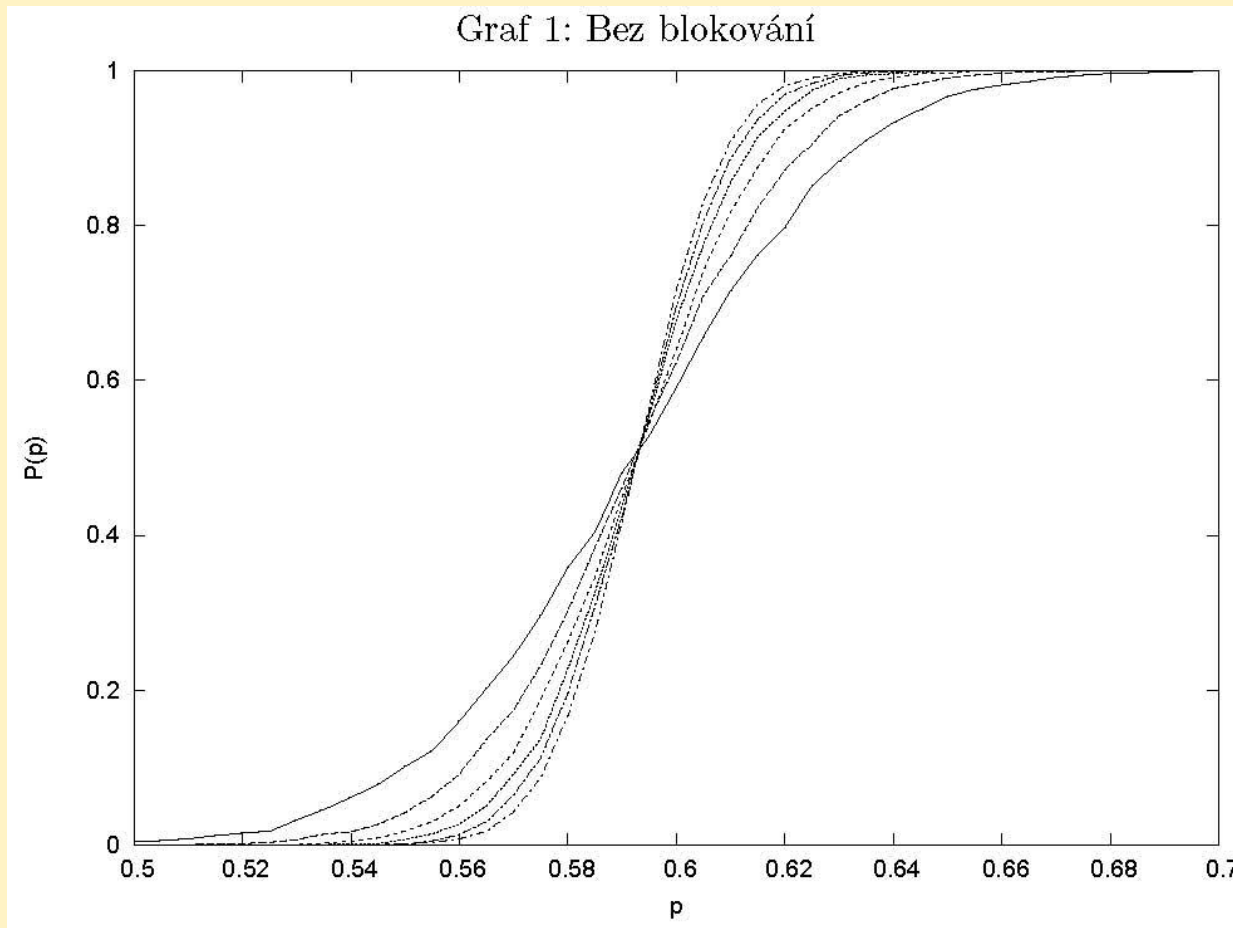
Určete práh perkolace jako průsečík funkcí pro různé velikosti.

Bonus: Proved'te totéž pro trojúhelníkovou nebo šestiúhelníkovou mřížku.

I-2 Uzlová perkolace – rozdělení klastrů

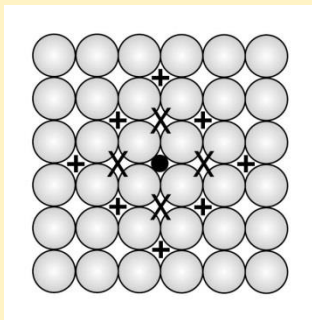
Užitím Hoshen-Kopelmanova algoritmu spočtete rozdělení velikostí klastrů při uzlové perkolaci. Uvažujte čtvercovou mřížku $L \times L$ s $L = 64$, pro pravděpodobnosti p v okolí prahu perkolace. Vyneste střední hodnotu hmotnosti klastru (druhý moment rozdělení velikostí klastrů) jako funkci pravděpodobnosti p v okolí prahu perkolace.

Uzlová perkolace: pravděpodobnost existence spanning klastru vs. pravděpodobnost obsazení

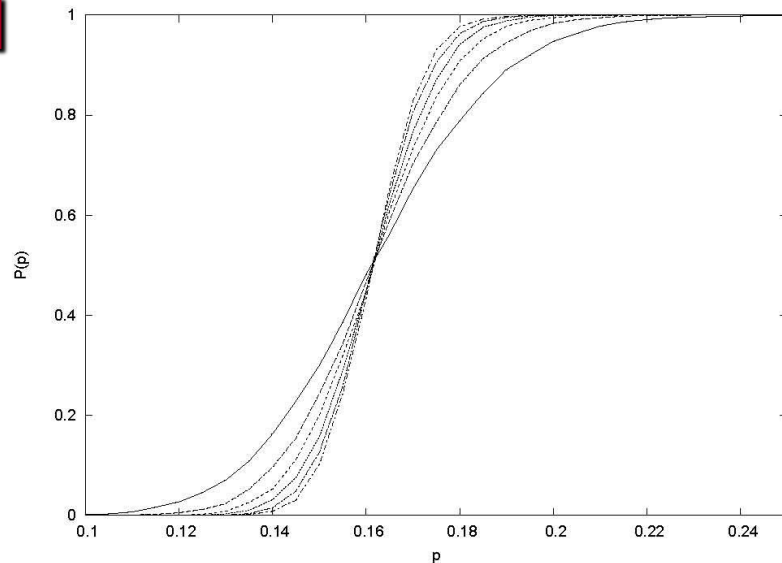


Perkolace blokujících uzlů

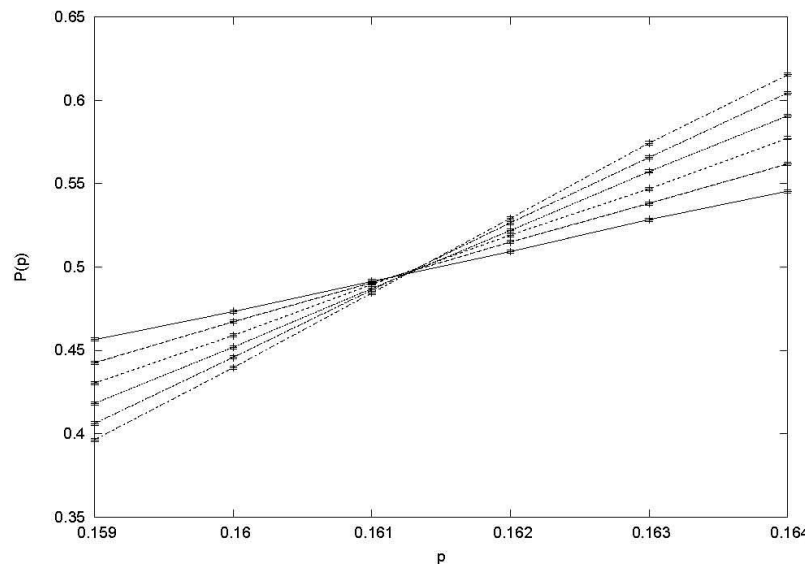
Simulujte s užitím následujícího modelu: na čtvercové mřížce se deponují nečistoty s koncentrací c_0 (každá nečistota na jednom uzlu, tj. nemohou se deponovat na sebe). Krátko-dosahová repulsivní interakce se popíše blokováním (zakázáním) uzlů v nejbližším okolí. Určete práh perkolace pro koncentraci c_0 , kdy dojde k perkolaci blokováných uzlů.



Graf 3: Pět blokováných uzlů



Graf 4: Pět blokováných uzlů — detail



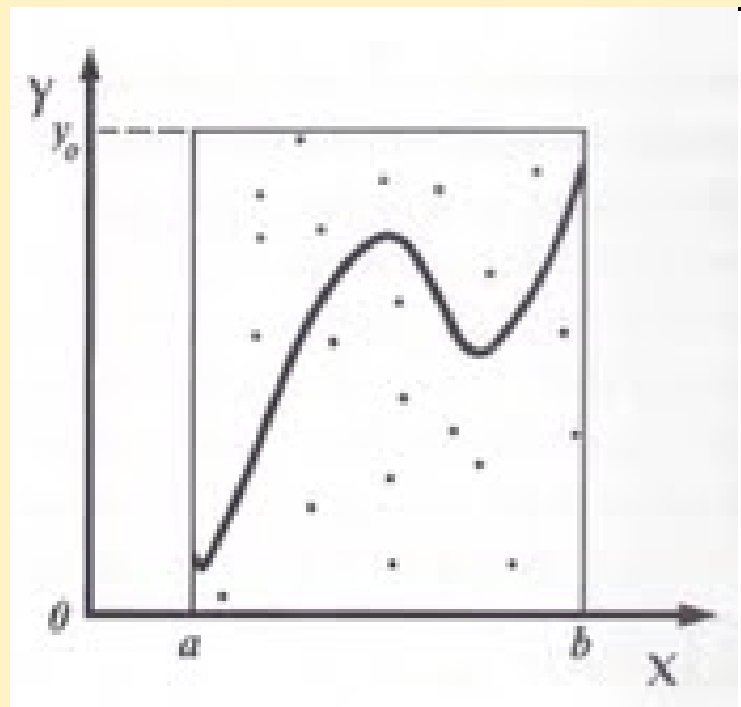
Monte Carlo integrace

Zadán konečný určitý integrál $I = \int F(x)dx$, ale neznáme analytickou řešení a nelze použít standardní metody numerické integrace – singularita, složité hraniční podmínky.

Metodou MC nespočteme integrál přesně, ale získáme odhad výsledku.

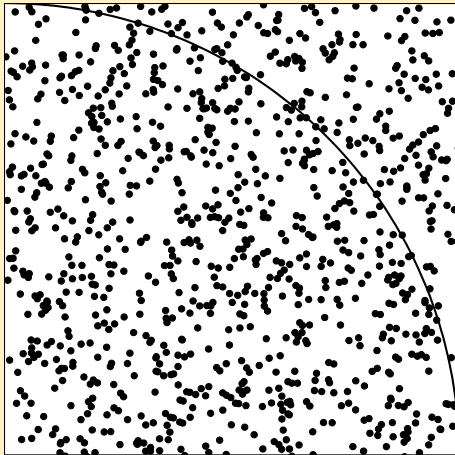
odhad I. metoda střelení

$$I_n = Y(b-a) M_n / n$$



Určení čísla π střílením

$$\pi / 4 = M_n / n$$



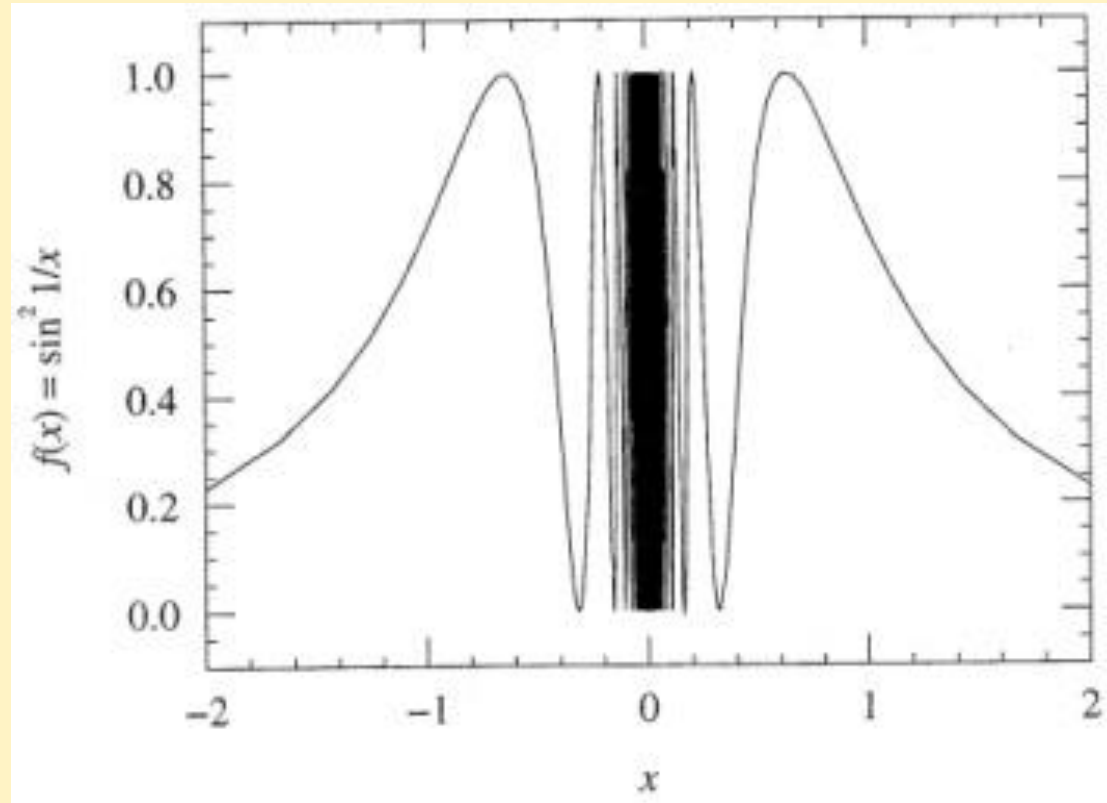
n	výsledek
1000	3,0800
2000	3,0720
3000	3,1147
4000	3,1240
5000	3,1344
6000	3,1426
7000	3,1343
8000	3,1242
9000	3,1480
10000	3,1440



100	3,1600
200	3,0400
300	3,1067
400	3,0800
500	3,0560
600	3,0800
700	3,0743

Další příklad

$$I = \int_a^b \sin^2 \frac{1}{x} dx$$



Ze zákona velkých čísel vyplývá, že
v limitě $n \rightarrow \infty$ $I_n \rightarrow I$

prosté vzorkování a zákon velkých čísel

odhad II. prosté vzorkování

$$I_n = \frac{b-a}{n} \sum_{k=1}^n f(x_k)$$

Ze zákona velkých čísel
vyplývá, že v limitě $n \rightarrow \infty$
 $I_n \rightarrow I$

Zákon velkých čísel popisuje skutečnost, že s rostoucím počtem opakovaných nezávislých pokusů se empirické charakteristiky, které popisují výsledky těchto pokusů, blíží k teoretickým charakteristikám.

Zákon velkých čísel

<https://is.muni.cz/do/rect/el/estud/prif/ps15/statistika/web/pages/zakon-velkych-cisel.html>

Uvažujeme nekonečnou posloupnost náhodných veličin $X_1, X_2, \dots$, které jsou stochasticky nezávislé, pocházejí ze stejného rozdělení a mají **stejnou** konečnou střední hodnotu $E(X_n)$ a stejný konečný rozptyl $D(X_n)$.

Pak náhodná veličina

$$S_n = \frac{1}{n} \sum_{k=1}^n X_k$$

konverguje ke střední hodnotě $E(X_n)$ tak, že pro libovolné $\varepsilon > 0$ platí

$$\lim_{n \rightarrow \infty} P(|S_n - E(X_n)| \leq \varepsilon) = 1$$

Tedy při velkém počtu nezávislých pokusů se aritmetický průměr výsledků jednotlivých pokusů liší od střední hodnoty jen málo.

Střední hodnotu $E(X_n)$ můžeme odhadnout pomocí průměru výsledků jednotlivých pokusů při dostatečně velkém n .

Monte Carlo integrace

Uvažujme množinu Ω , jako podmnožinu $\mathbf{R}^m$ na níž je dán mnohonásobný určitý integrál

$$I = \int_{\Omega} f(\bar{\mathbf{x}}) d\bar{\mathbf{x}}$$

Integrál se počítá pro známý objem V množiny Ω

$$V = \int_{\Omega} d\bar{\mathbf{x}}$$

Nejjednodušší metoda, jak se provádí výpočet, je prostě získat odhad integrálu Q_N pomocí N uniformních vzorků na oblasti Ω v N bodech:

$$\bar{\mathbf{x}}_1, \dots, \bar{\mathbf{x}}_N \in \Omega,$$

Integrál I je pak aproximován jako:

$$I \approx Q_N \equiv V \frac{1}{N} \sum_{i=1}^N f(\bar{\mathbf{x}}_i) = V \langle f \rangle$$

Díky zákonu velkých čísel platí:

$$\lim_{N \rightarrow \infty} Q_N = I$$

Příklad úlohy

Výpočet funkce v jedné dimenzi pomocí MC

Vykreslete funkci $I(x)$ danou integrálem

$$I(x) = \int_0^x \sin^2 \frac{1}{y} dy$$

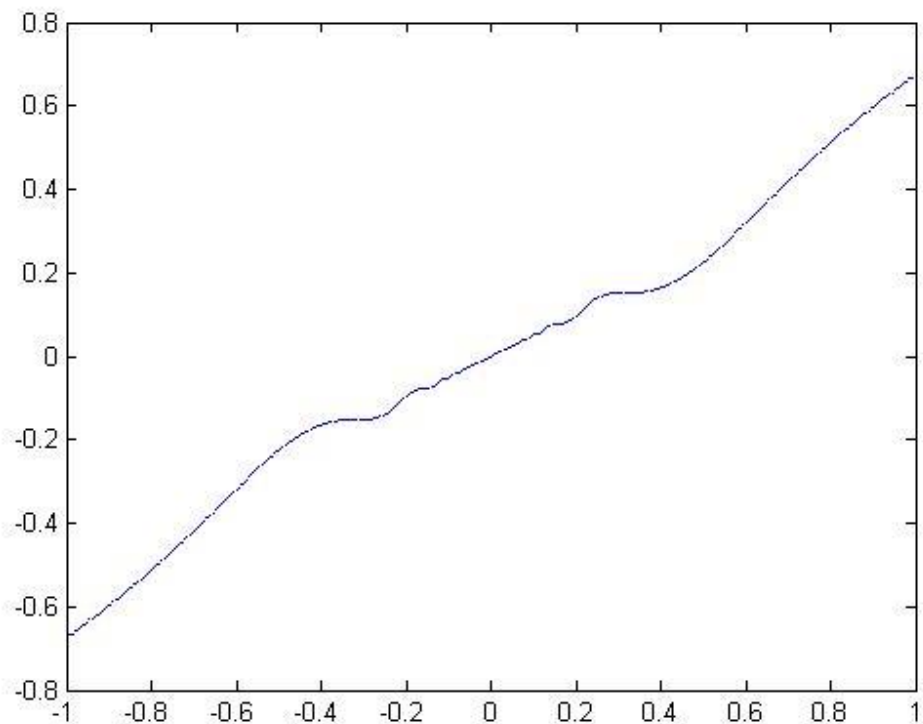
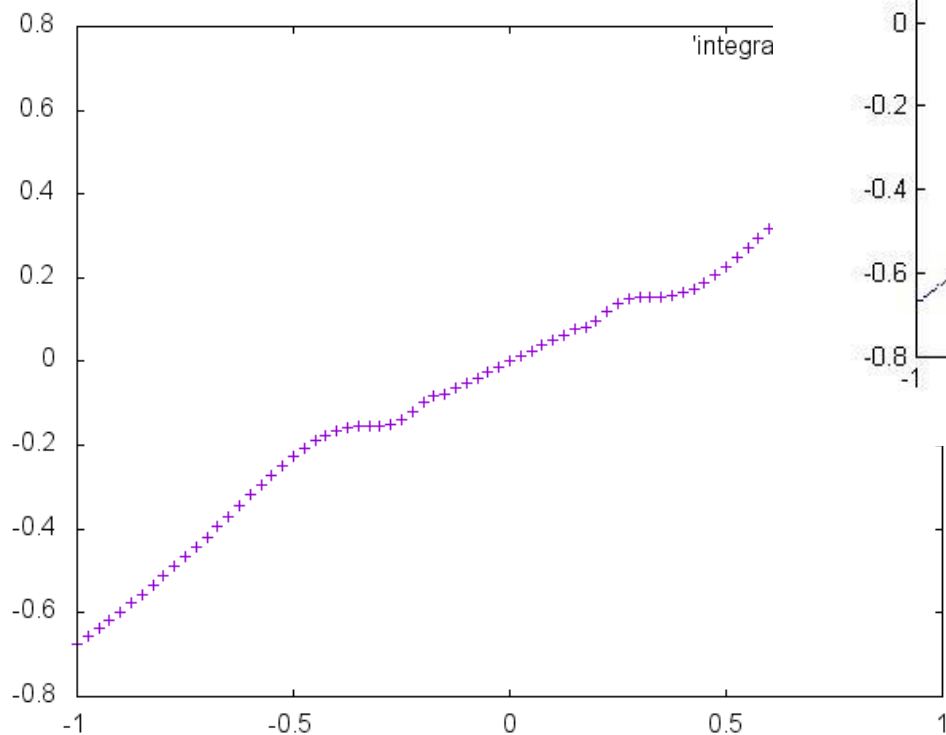
v intervalu od $x = -1$ do $x = 1$.

$$\int_0^x \sin^2 \frac{1}{y} dy$$

Vyřešený příklad

Perláková

Hvizdoš



Monte Carlo vs. numerická integrace

chyba MC:

Centrální limitní
teorém

$$1 / \sqrt{n}$$

podle variance náhodných veličin - odvození pro střídání

U numerické integrace dělíme interval (a,b) na N_b sub-intervalů, bodů.

Místo počtu n pokusů v případě MC je zde rozhodujícím parametrem počet dělení (bodů) $N_b = N$.

Celkový počet dělení N_x dramaticky roste s počtem proměnných $N_p = N_{\text{castic}}$ d, jako $N_x = (N_b)^{(N_p)}$.

Chyby numerické integrace vs. MC

Pro srovnání je třeba znát závislost chyby numerické integrace na parametru N .

Metoda	1 dimenze	d dimenzí
lichoběžíková	$1/N^2$	$1/N^{2/d}$
Simsonova	$1/N^4$	$1/N^{4/d}$

Pro integrál v dimenzi $d > 8$ je MC výhodnější než numerická integrace.

Návrat k modelování

Celulární automaty

Samorganizace

Pojem samorganizace je užíván v různých kontextech v: informační teorii, termodynamice, matematice, kybernetice atd.

Patří sem mnoho jevů např:

- krystalizace;

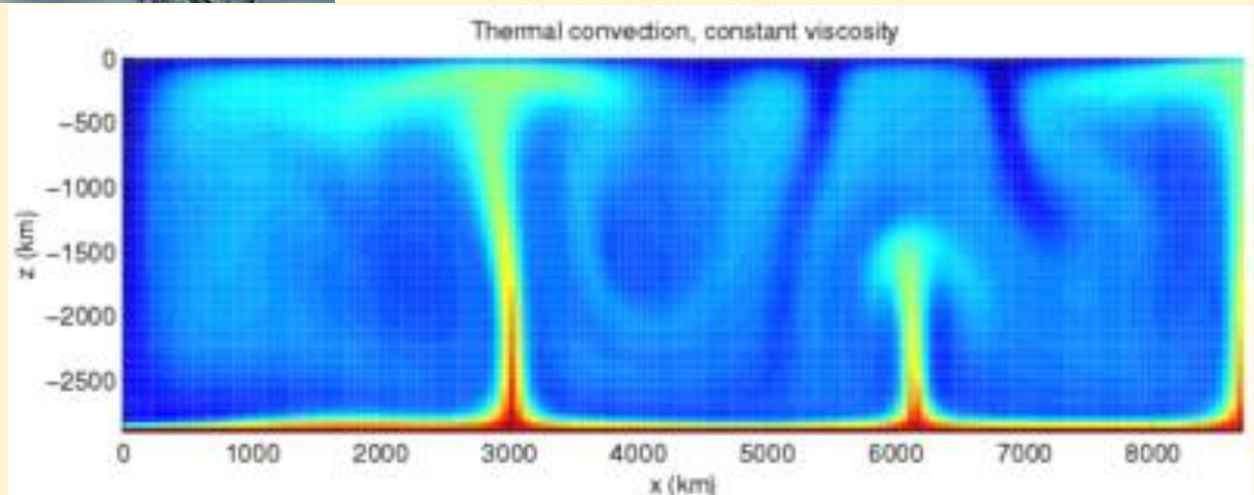


- vznik hejn ptáků;



- konvekční proudění v kapalině;

- etc.



Samorganizace

Neexistuje však obecně přijímaná definice.

Možnost: **Samorganizace je proces při němž vzniká spontánně globální pořádek nebo koordinace v systému jen s lokálními interakcemi.**

Termodynamické hledisko: 2. termodynamická věta.

Izolovaný systém se vyvíjí do stavu s maximální entropií, tj. max. neuspořádaností $\Rightarrow$ rovnovážné **izolované systémy nevedou k samouspořádávání.**

Reálné samouspořádávající se systémy (rostliny, krystaly, společenství etc.) jsou **otevřené a disipativní.**

Je nutné studovat **dynamiku** otevřených systému !

Celulární automaty (CA)

- alternativní popis samoorganizace
- diskrétní souřadnice i diskrétní proměnné + diskrétní čas
- CA mají netriviální dynamické chování
- jednodušší než modely samoorganizace s diferenciálními rovnicemi
- CA jsou jednodušší, proto umožňují detailnější analýzu
- přesto velká rozmanitost komplikovaných jevů

Historické pozn.:

- ⌚ **1963 von Neumann a Ulam zavedli jako idealizaci biologických systémů („cellular spaces“) s cílem modelování samoprodukce**
- ⌚ **postupně zaváděny v dalších oblastech pod různými jmény**
- ⌚ **1983 Wolfram – [Rev. Mod. Phys. 55, 601 \(1983\)](#)**
- ⌚ ...

Celulární automaty (CA)

Formalizace:

CA určen zadáním možných stavů a dynamických pravidel

Stav:

pravidelné uspořádání **konečného počtu buněk**,
každá buňka může být v některém z **konečného počtu stavů**

- pole buněk indexované indexem i , obvykle mříž Λ v d dimenzích
- stav buňky je popsán hodnotou **diskrétní** proměnné s_i z konečné množiny stavů S

Pravidlo:

- lokální pravidlo evoluce v **diskrétním čase** $t \rightarrow t + 1$
 - pro každé i definuj určité okolí $O(i)$ – množina buněk v okolí buňky i

pro každé i z Λ nová hodnota $s_i(t+1) = F(\{s\}_{O(i)}(t))$

například:

$$s_i(t+1) = s_{i-1}(t).XOR.s_{i+1}(t)$$

Celulární automaty - typy

Různé typy:

- deterministické nebo stochastické podle charakteru pravidla
- synchronní (paralelní)
- asynchronní – sekvenční (tj. po řadě)
 - s náhodným výběrem
- podle množiny stavů S
 - binární, speciálně $S = \{0, 1\}$
 - q stavů

Celulární automaty - aplikace

Používají se různých oblastech:

- ❖ **dopravní problémy**
- ❖ **vývoj rozhraní**
- ❖ **hydrodynamika** – hrubý popis proudění tekutin, tj. řešení Navierových-Stokesových rovnic
- ❖ **modelování biologické evoluce**
- ❖ **studium imunologie**
- ❖ **transport částic** – řešení Boltzmannovy rovnice
- ❖ **studium samoorganizovaných kritických jevů**
- ❖ **etc. etc.**

Celulární automaty - počet

Počet automatů velmi rychle roste s dimenzí !

- např. nechť $O(i)$ jsou **jen nejbližší sousedé** a samotná buňka i se nepočítá, pak počet sousedů je $K = 2d$
- uvažujme binární automat, pak počet konfigurací v okolí buňky je
$$N_C = 2^K \quad (\text{pro } q \text{ stavů } N_C = q^K), \quad C \text{ je množina konfigurací}$$
- pravidlo: zobrazení množiny C stavů do množiny hodnot S
- počet pravidel je:

$$N_P = 2^{N_C} = 2^{2^K}$$

d	1	2	3	...
N_C	$2^2 = 4$	$2^4 = 16$	$2^8 = 64$	
N_P	$2^4 = 16$	$2^{16} = 65536$	$2^{64} > 10^{19}$	

Je-li zahrnut počáteční uzel, pak $K = 2d + 1$ a počet automatů roste s dimenzí ještě rychleji.

CA může záviset i
na původní buňce.

$$d = 1, K = 3 \text{ (} i-1, i, i+1 \text{)}$$

$$N_C = 2^3 = 8$$

$$N_P = 2^8 = 256$$

Pro analýzu lze využít
symetrie, pak je *jen*
32 různých pravidel.

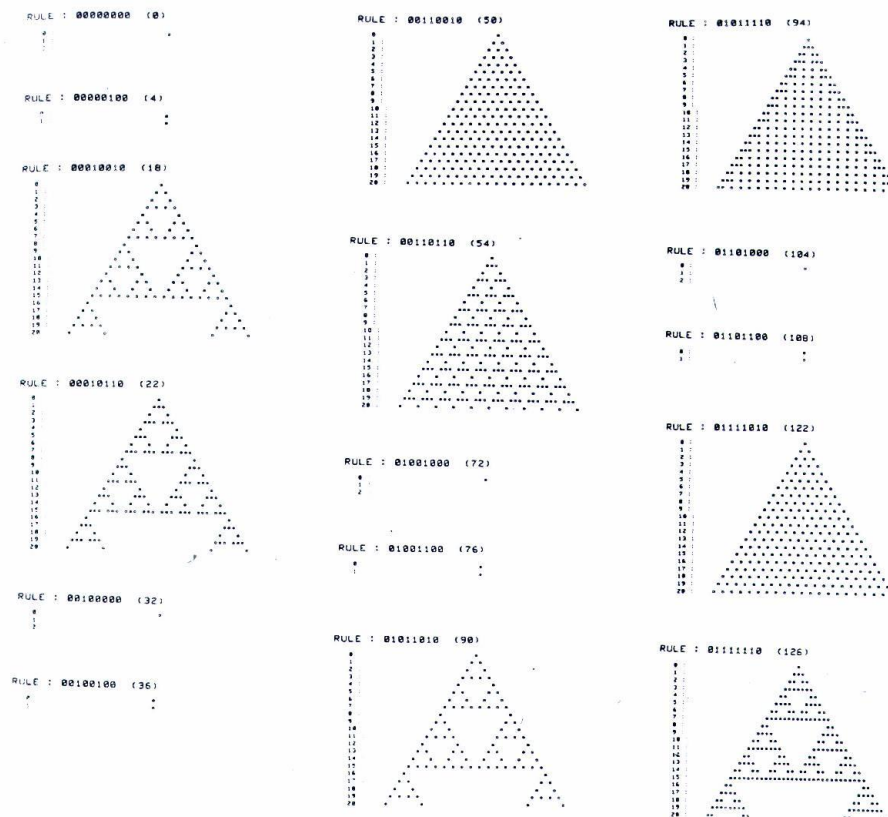


FIG. 3. Evolution of one-dimensional elementary cellular automata according to the 32 possible legal sets of rules, starting from a state containing a single site with value 1. Sites with value 1 are represented by stars, and those with value 0 by blanks. The configurations of the cellular automata at successive time steps are shown on successive lines. The time evolution is shown up to the point where the system is detected to cycle (visiting a particular configuration for the second time), or for at most 20 time steps. The process is analogous to the growth of a crystal from a microscope seed. A considerable variety of behavior is evident. The cellular automata which do not tend to a uniform state yield asymptotically self-similar fractal configurations.

is nontrivial. In the infinite time limit, the configurations are "self-similar" in that views of the configuration with different "magnifications" (but with the same "resolution") are indistinguishable. The configurations thus exhibit the same structure on all scales.

Consider as an example the modulo-two rule 90 (also used as the example for Fig. 1 and in the discussion above). This rule takes each site to be the sum modulo

two, as illustrated in Fig. 4 (cf. Wolfram, 1982b). The values of the sites are hence the values of binomial coefficients [or equivalently, coefficients of x^i in the expansion of $(1+x)^n$ modulo two]. In the large time limit, the pattern of sites with value 1 may be obtained by the recursive geometrical construction (cf. Sierpinski, 1916; Abelson and diSessa, 1981, Sec. 2.4) shown in Fig. 5. This

Celulární automaty - klasifikace

V roce 1983 prozkoumal všechny automaty pro 1D a K=3 Wolfram, [Rev. Mod. Phys. 55, 601 \(1983\)](#).

v případě q stavů

$$N_p = q^K$$

Obecně je nepraktické vyčíslovat všechny CA, užitečná je klasifikace podle:

- a) typu vývoje
- b) stability k poruše
- c) aplikace

Modelování evoluce

koevoluce mnoha živých organizmů stejné složitosti

Hra života – game of life, http://cs.wikipedia.org/wiki/Hra_%C5%BEivota
dvoustavový, dvourozměrný CA cca 1970, (britský matematik John Conway) vznikl před Wolframovým článkem!



Buňky jsou obsazené nebo neobsazené organizmem, tj jsou živé nebo mrtvé.

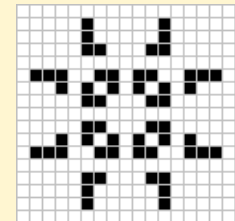
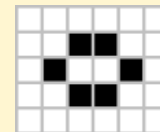
CA s $K = 8$, tj. 8 sousedů na čtvercové mřížce

Pravidla:

1. Každá **živá** buňka s méně než dvěma živými sousedy **zemře**.
2. Každá **živá** buňka se dvěma nebo třemi živými sousedy **zůstává žít**.
3. Každá **živá** buňka s více než třemi živými sousedy **zemře**.
4. Každá **mrtvá** buňka s právě třemi živými sousedy **oživne**.

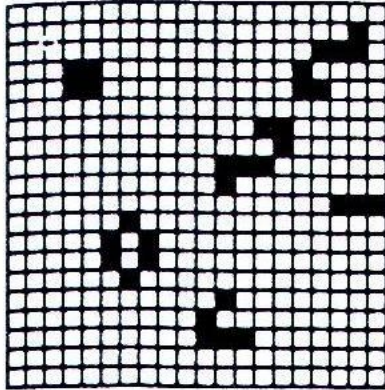
Konfigurace se skládají z:

- stabilních elementů
- periodických vzorů
- pohybujících a měnících se struktur

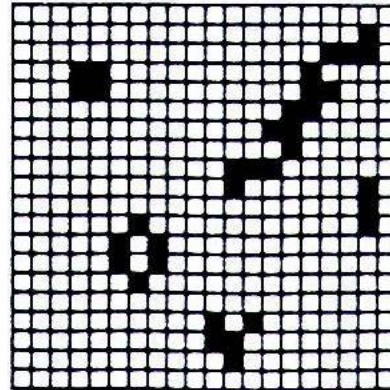


<http://www.bitstorm.org/gameoflife/>

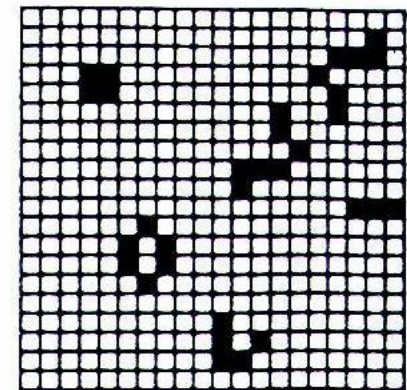
Hra života – vývoj konfigurací



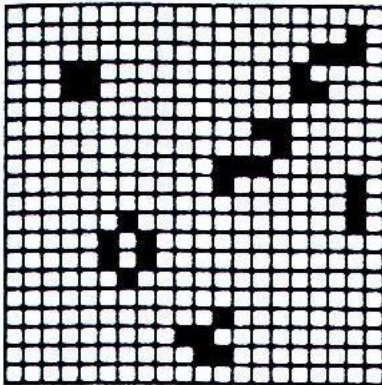
(a)



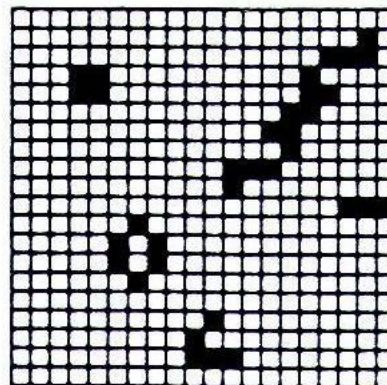
(b)



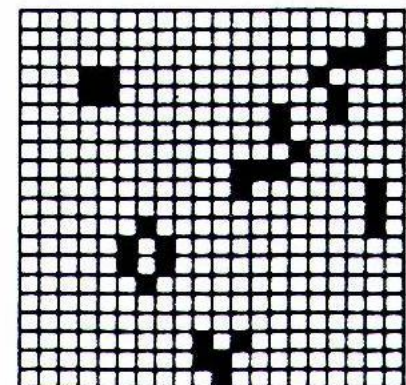
(c)



(d)



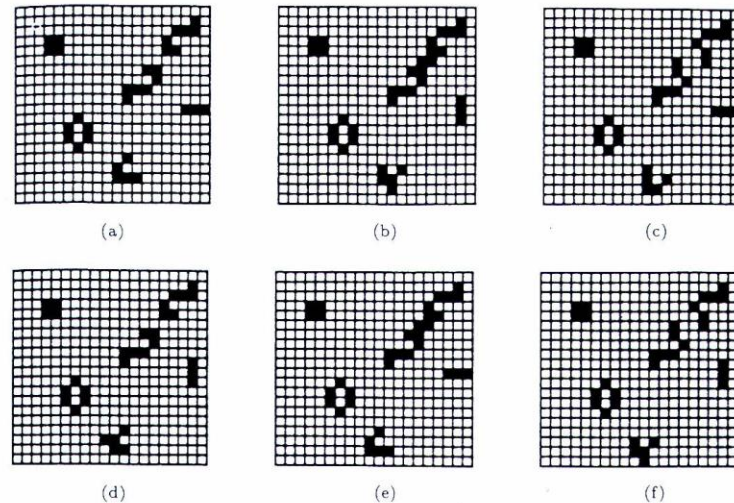
(e)



(f)

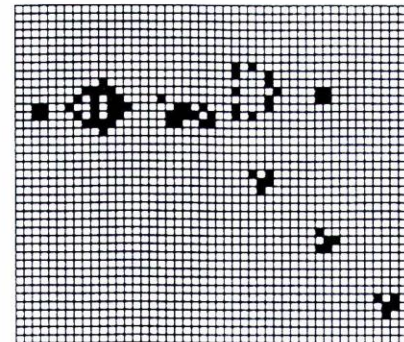
Six “Game of Life” steps of the evolution of a 20×20 configuration.

Hra života



Six "Game of Life" steps of the evolution of a 20×20 configuration.

vznik nových klastrů
pomocí „děla“



A so-called "gun" which periodically generates gliders with the "Game of Life" cellular automaton.

wikipedia: **game of life**
http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life#External_links
<http://www.bitstorm.org/gameoflife/>

Vlastnosti hry života

Vývoj na vzorku $L \times L$ vždy dosáhne stacionárního stavu s cca 3 % živých buněk - atraktor.

Proces je samoorganizující se. Potřebný čas $t_{\text{rel}} \sim L^{1/2}$.

Provedme poruchu stavu:

jednu mrtvou buňku změňme na živou buňku.

Následně se dosáhne obecně jiný stacionární stav.

Změna se může lavinovitě šířit přes celou konfiguraci.

Měříme velikost S laviny změn.

Vlastnosti hry života

Vývoj na vzorku $L \times L$ vždy dosáhne stacionárního stavu s cca 3 % živých buněk - atraktor.
Proces je samoorganizující se. Potřebný čas $t_{\text{rel}} \sim L^{1/2}$.

Provedme poruchu stavu:
jednu mrtvou buňku změňme na živou buňku.
Následně se dosáhne obecně jiný stacionární stav.
Změna se může lavinovitě šířit přes celou konfiguraci.
Měříme velikost S laviny změn.

$$n(S) = S^{-\tau}, \quad \tau \approx 1,3$$

Existence mocninné závislosti implikuje,
že není žádná typická velikost laviny!

3d varianta na youtube

<http://www.youtube.com/watch?v=xg0PKAvL01Y>

Samoorganizovaná kritikalita

self-organized criticality (SOC)

http://en.wikipedia.org/wiki/Self-organized_criticality

BTW (*Per Bak, Chao Tang and Kurt Wiesenfeld*) model 1987

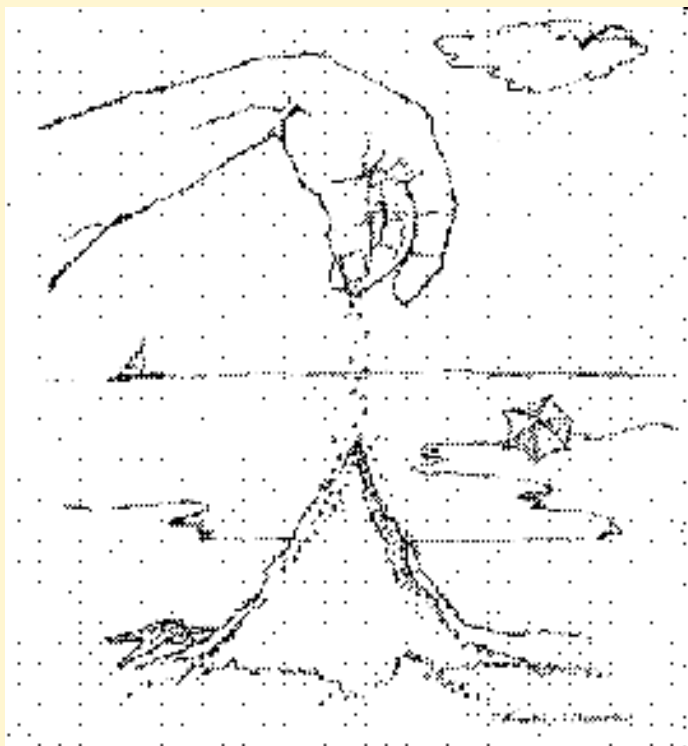
P. Bak, C. Tang a K. Wiesenfeld, *Physical Review Letters* **59** (4): 381–384, (1987).

Samoorganizovaný kritický systém je disipativní systém spontánně se vyvíjející do stacionárního stavu s korelacemi na všech škálách.

Příklady:

- hromádky sypaného písku
- stékající kapky na skle
- sněhové laviny
- pohyb zemských desek-zemětřesení
- vznik zácp v dopravě
- evoluce
- etc.

Pískové kupy



BTW model *(Per Bak, Chao Tang and Kurt Wiesenfeld)*

stav: soubor nezáporných celých čísel na 2 D mřížce $z(x, y) \leq 0$

pravidla: na náhodně vybraný uzel se uloží částice.

$$z(x, y) \text{ se zvětší na } z(x, y) + 1$$

libovolný uzel s $z(x, y) \geq 4$ je nestabilní

při dosažení nestability nastane redistribuce:

$$z(x, y) \rightarrow z(x, y) - 4$$

$$z(x \pm 1, y) \rightarrow z(x \pm 1, y) + 1$$

$$z(x, y \pm 1) \rightarrow z(x, y \pm 1) + 1$$

Vznik nestabilit se může řetězit.

Vznikají laviny změn

1	2	0	2	3
2	3	2	3	0
1	2	3	3	2
3	1	3	2	1
0	2	2	1	2

1	2	0	2	3
2	3	2	3	0
1	2	4	3	2
3	1	3	2	1
0	2	2	1	2

1	2	0	2	3
2	3	3	3	0
1	3	0	4	2
3	1	4	2	1
0	2	2	1	2

1	2	0	2	3
2	3	3	4	0
1	3	2	0	3
3	2	0	4	1
0	2	3	1	2

1	2	0	3	3
2	3	4	0	1
1	3	2	2	3
3	2	1	0	2
0	2	3	2	2

1	2	1	3	3
2	4	0	1	1
1	3	3	2	3
3	2	1	0	2
0	2	3	2	2

1	3	1	3	3
3	0	1	1	1
1	4	3	2	3
3	2	1	0	2
0	2	3	2	2

1	3	1	3	3
3	1	1	1	1
2	0	4	2	3
3	3	1	0	2
0	2	3	2	2

1	3	1	3	3
3	1	2	1	1
2	1	0	3	3
3	3	2	0	2
0	2	3	2	2

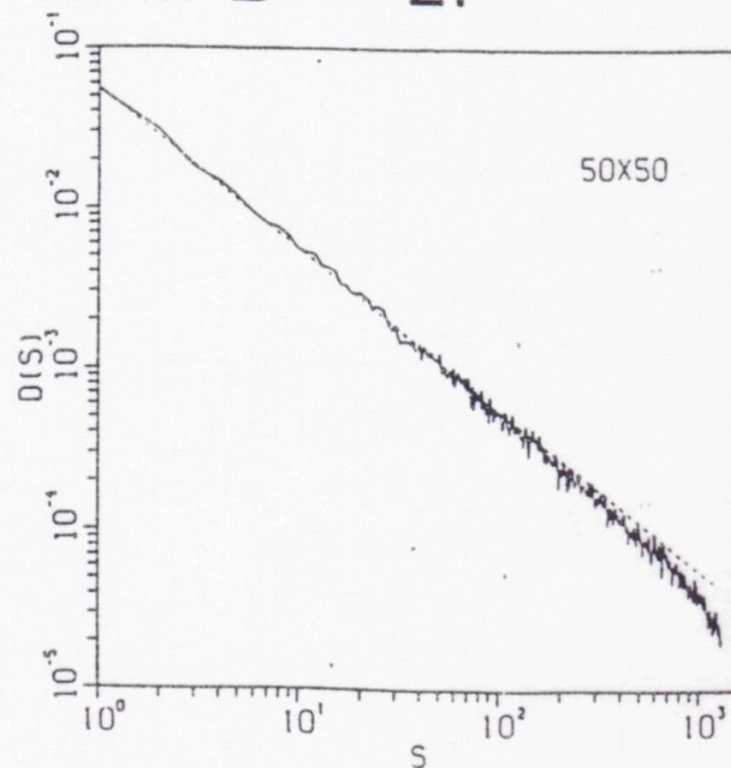
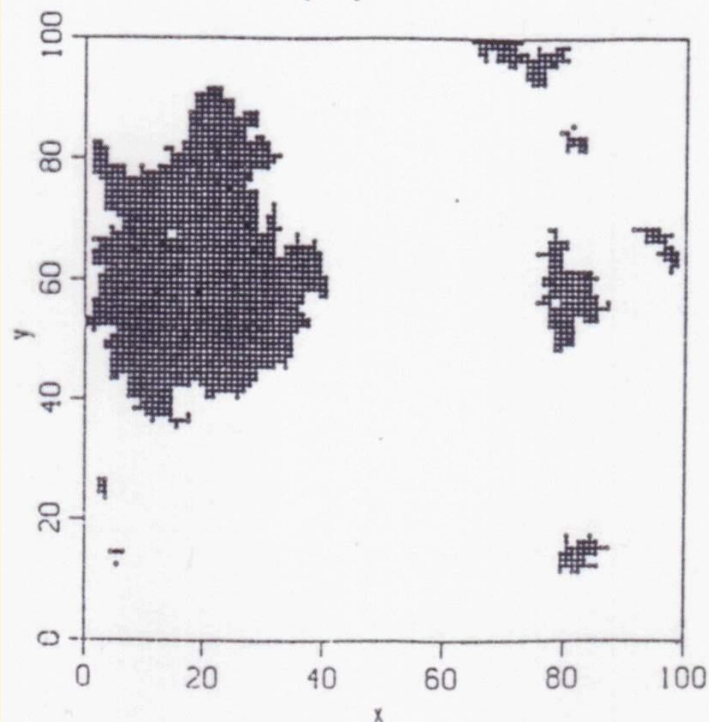
1	3	1	3	3
3				1
2				3
3	3			2
0	2	3	2	2



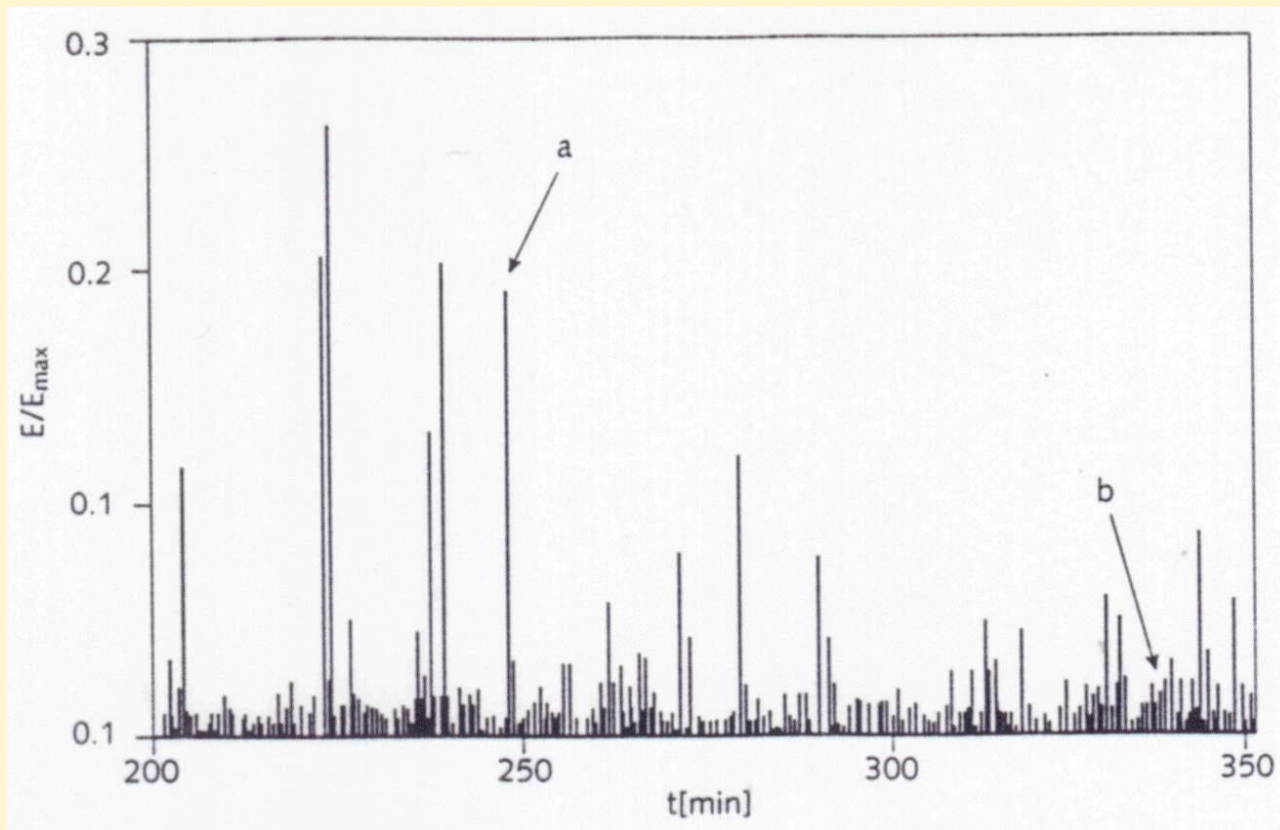
Velikost laviny
S se měří jako
počet
zasažených
uzlů po přidání
jedné částice.

Rozdělení velikostí lavin:

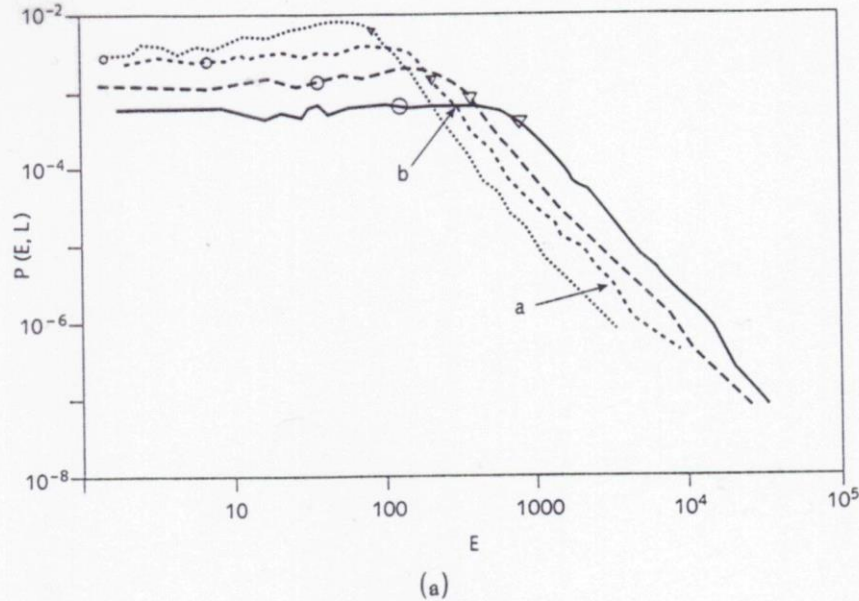
$$P(s) \propto s^{-\tau}, \quad \tau = 1.22 \text{ in } D = 2.$$



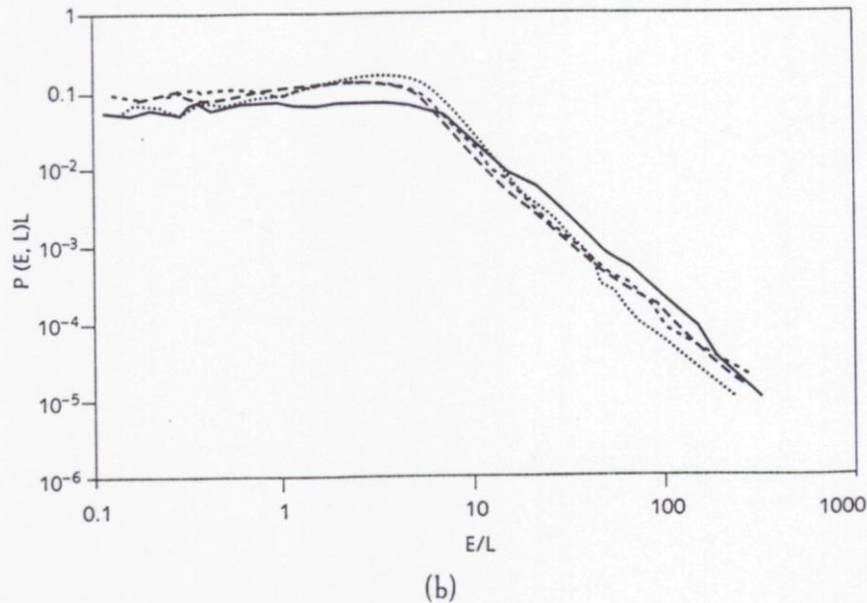
Experimentální měření lavin skutečného materiálu



Rozdělení lavin pro rýžové kupy



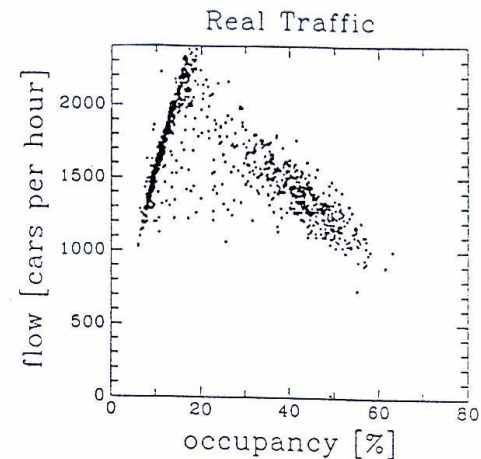
Křivky pro různé realizace



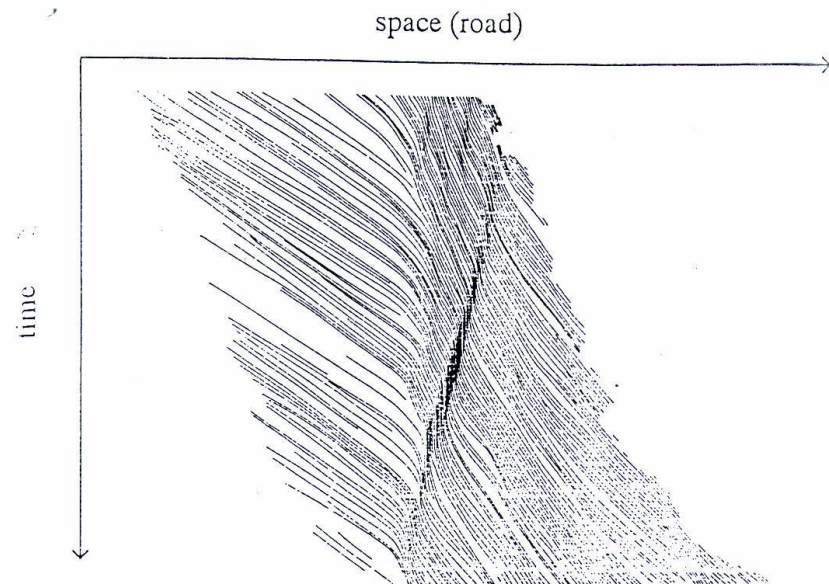
←
Přeškálované
křivky ukazují,
že jev je univerzální.

Data o skutečné dopravě

Cíl je maximální průjezdnost.
Ale vznikají zácpy.



Traffic flow q (in cars per hour) vs. occupancy (in cars per hour) from meas. reality. Occupancy is the percentage of the road which is covered by vehicles (after [17]).



Space-time-lines (trajectories) for cars from Aerial Photography (after [16]). Each line represents the movement of one vehicle in the space-time-domain.

Nagel-Schreckenbergův model

http://en.wikipedia.org/wiki/Nagel-Schreckenberg_model

Stav buňky: i) prázdná=žádné auto ii) auto s rychlostí V ; $V = 0, 1, \dots, V_{max}$.

Dynamika: obsazené buňky se pohybují jedním směrem $i \rightarrow i+1$;

auto na uzlu i vidí auto vpředu do vzdálenosti L

Paralelní automat – update se provádí paralelně.

V každém kroku se aplikují 4 akce v uvedeném pořadí na všechna auta.

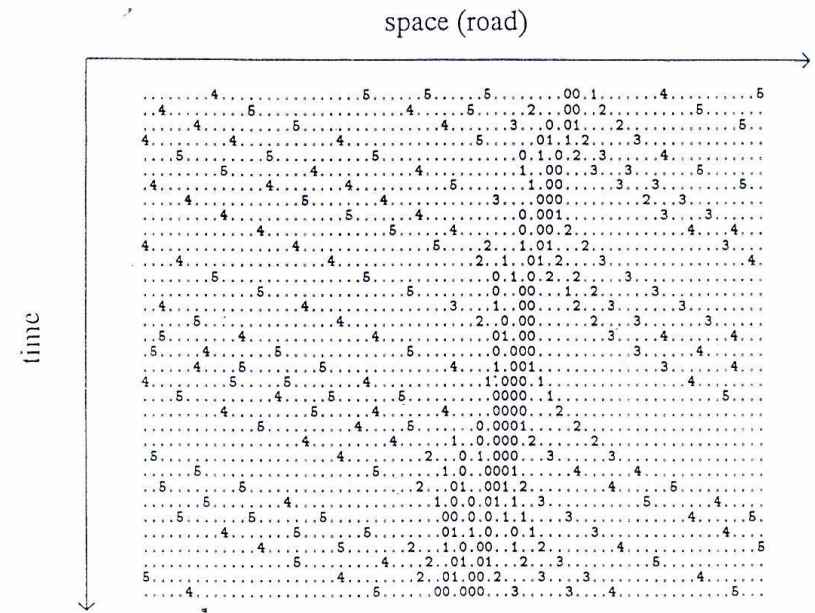
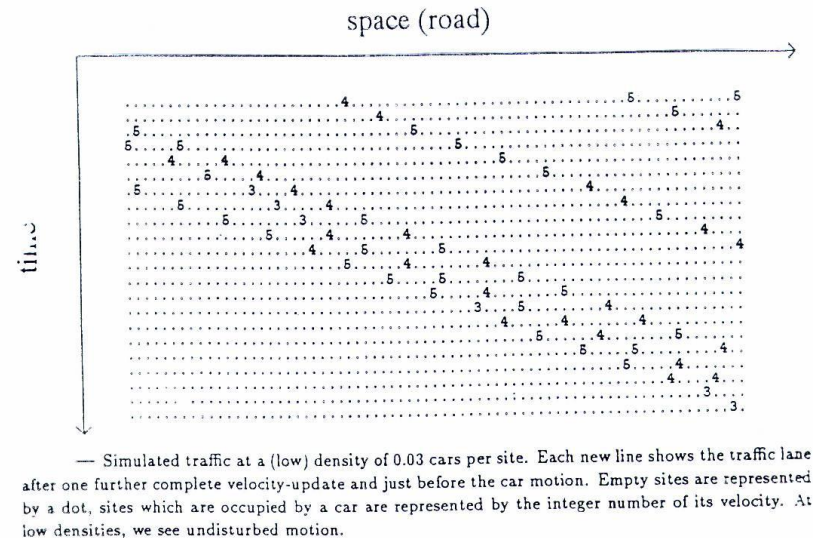
1. ZRYCHLENÍ: když auto jede menší než maximální rychlostí, pak jeho rychlost je zvýšena o jednotku, tj. $V \rightarrow V+1$, pokud může být.
2. BRZDĚNÍ: každé auto se kontroluje, aby jeho vzdálenost k předchozímu autu byla menší než jeho rychlost, tj. když $L \leq V$, pak $V \rightarrow V-1$.
3. NÁHODNÉ ZPOMALENÍ: rychlost každého auta, které má rychlost větší než 0, je s pravděpodobností p snížena o jednotku.
4. POHYB: všechna auta jsou posunuta dopředu o počet jednotek rovný jejich rychlosti.

Bod 3. je podstatný: lidský faktor, stav vozovky etc.,
bez něj přechod do stacionárního stavu s neměnnými rychlostmi!

Simulace Nagel- Schreckenbergova modelu

Vznikají zácpy, když
pohybu auta brání
předchozí vozidlo.

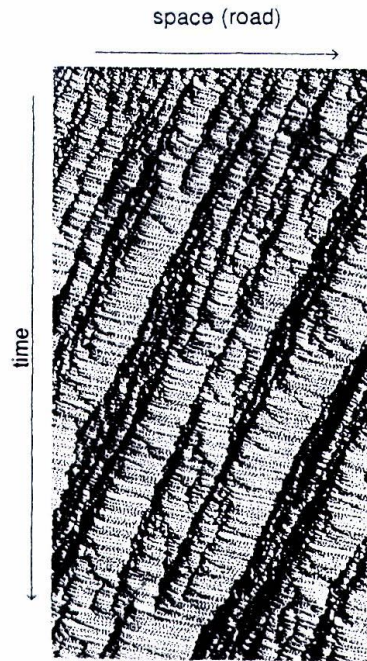
Zácpy se pohybují
proti směru jedoucích
vozidel.



Simulace Nagel- Schreckenbergova modelu

Zácpa se rozplyne,
když počet omezených
auta je nula.

Počet kroků t , než se
zácpa rozplyne, je
doba života zácpy.



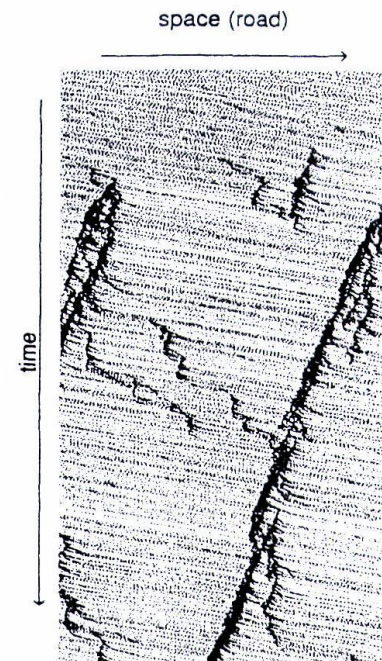
$v_{\max}=5$, parallel update

$$\rho = 0,3$$

rozdělení dob života

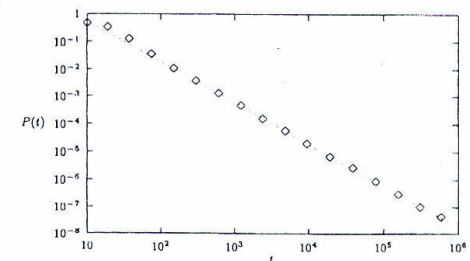
$$P(t) \sim t^{-\delta}$$

$$\delta = 1,5 \pm 0,01$$



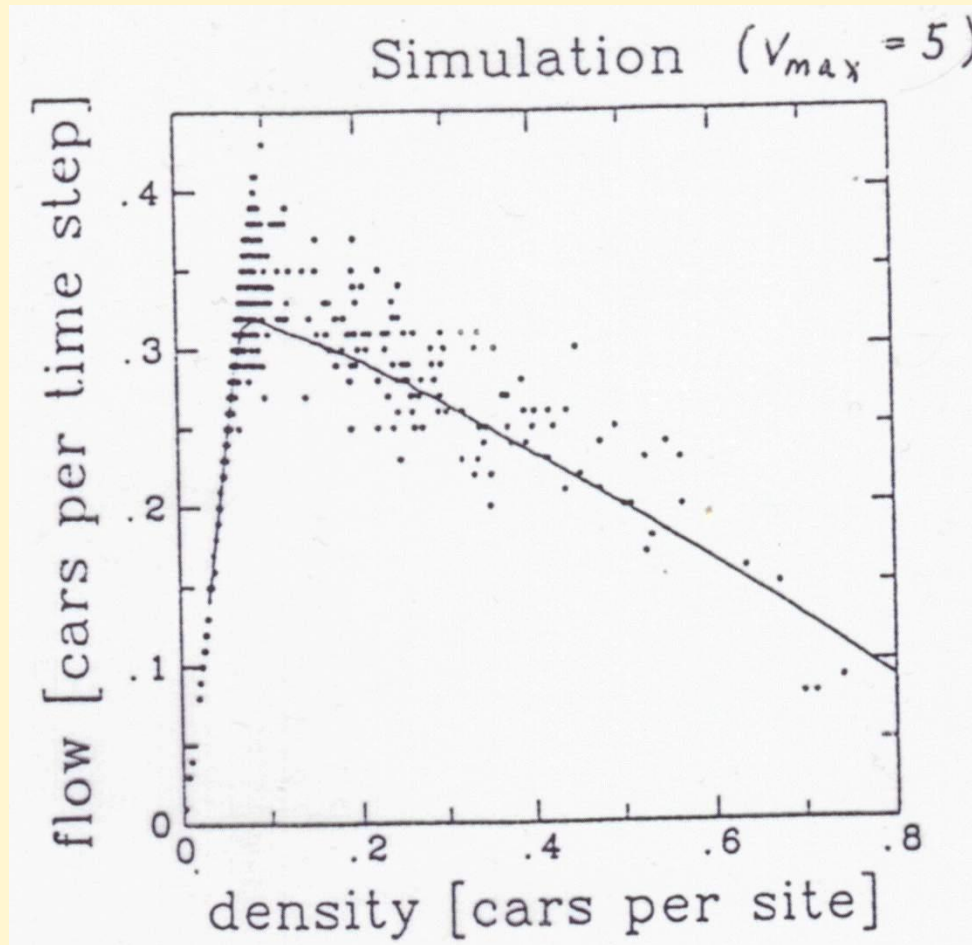
$v_{\max}=5$, parallel update

$$\rho = 0,1$$



Lifetime distribution $P(t)$ for emergent jams in the outflow region; average over more than 65 000 clusters (avalanches). The dotted line has slope $-\frac{1}{2}$. Numerically imposed cutoff at $t = 10^6$.

Základní diagram dopravy



Cvičení

Náhodná čísla - realizace

Existují kódy generátorů v různých programovacích jazycích.

Generátor je vhodné inicializovat např. pomocí času v počítači – `time()` .

Začátek: jednoduchý prográmeček pro výpis zvoleného počtu náhodných čísel.

Náhodná čísla - realizace

Začátek: jednoduchý prográmeček pro výpis zvoleného počtu náhodných čísel při volání generátoru RAND.

```
program rngtest
print *, 'vlož počet pokusu '
read *, npokus
print *, ' počet pokusu ', npokus
do 10 i=1,npokus
    rndtest=RAND()
    write(6,*) i,rndtest
10 continue
end program rngtest
```

```
vlož počet pokusu
10
    počet pokusu      10
      1 0.61411285
      2 0.39798999
      3 1.78151131E-02
      4 0.41991234
      5 0.46889496
      6 0.71923804
      7 0.23525500
      8 0.93125153
      9 0.54643750
     10 0.97659349
```

Náhodné procházky v 2D

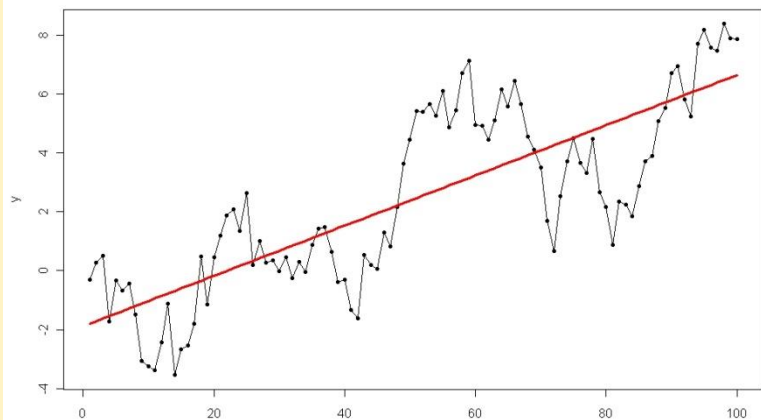
Simulujte a analyzujte různé typy náhodných procházek na dvou-dimenzionální čtvercové mřížce. Určete závislost vzdálenosti počátečního a koncového bodu na počtu kroků pro tyto typy procházek: prostá náhodná procházka (výběr ze čtyř směrů), procházka bez okamžitých návratů (výběr ze tří směrů) a jako volitelný bonus i procházka bez protínání.

Cíl: Porovnejte naměřené exponenty včetně odhadu chyby jejich měření.

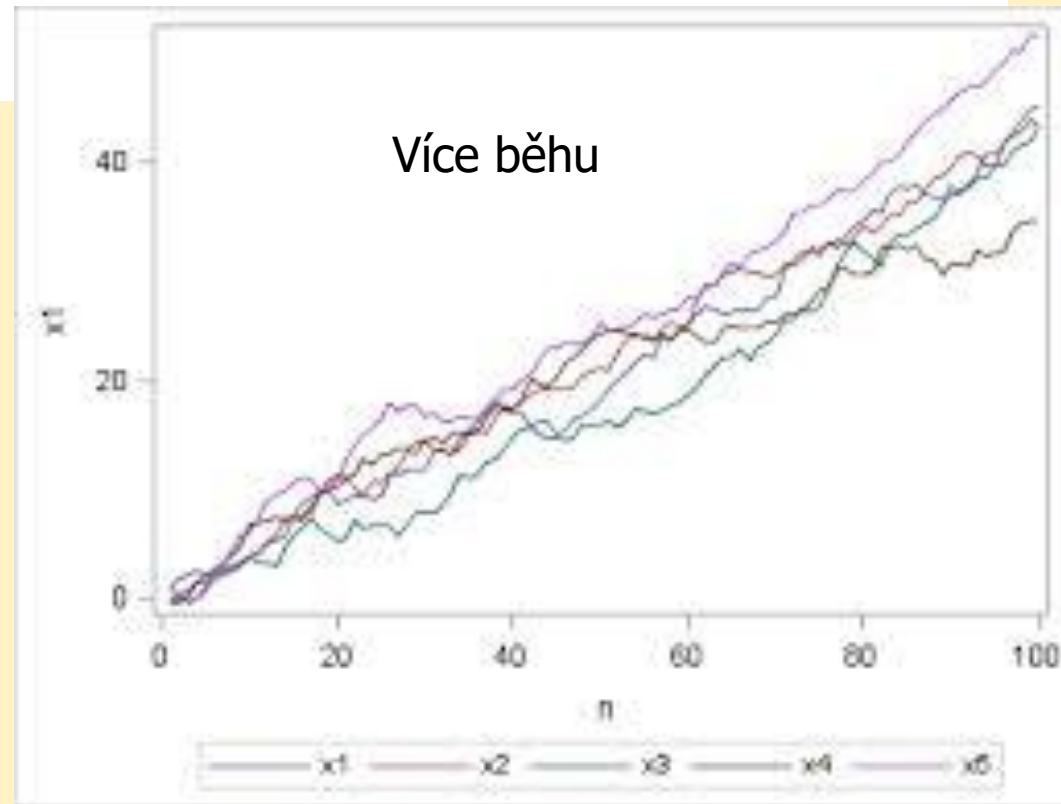
Postupuje podle následujícího:

Náhodné procházky

Měříme vzdálenost R od počátku v závislost na čase, např.



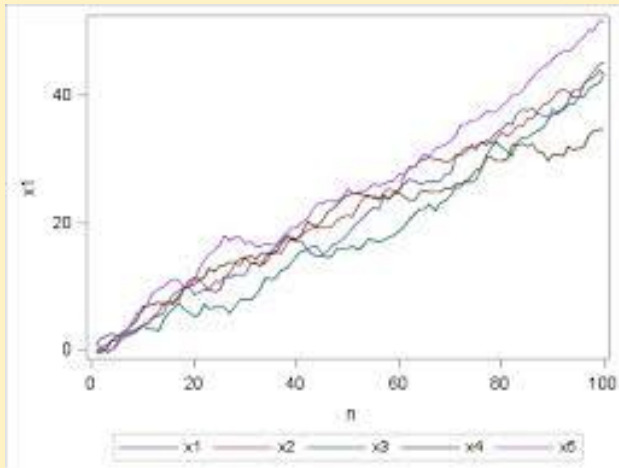
Jeden běh



Určete příslušnou funkci $R(t)$.

Prostá náhodná procházka

Začnete prostou náhodnou procházkou na dvou-dimenzionální čtvercové mřížce, tj. vybíráte náhodně pohyb jedním ze čtyř směrů. Měřte závislost vzdálenosti R od počátečního do koncového bodu na počtu kroků (čase) t pro n procházek.



Různé běhy následně vystředujte, pro každý krok tak dostanete střední hodnotu vzdálenosti $\langle R \rangle$. Vyneste vývoj $\langle R \rangle$ v závislost na čase t včetně indikace chyb, (viz dále).

Křivku pro vystředované běhy je třeba fitovat mocninnou závislostí a určit

exponent α v $R \sim t^\alpha$.

Chyby při měření náhodné procházky

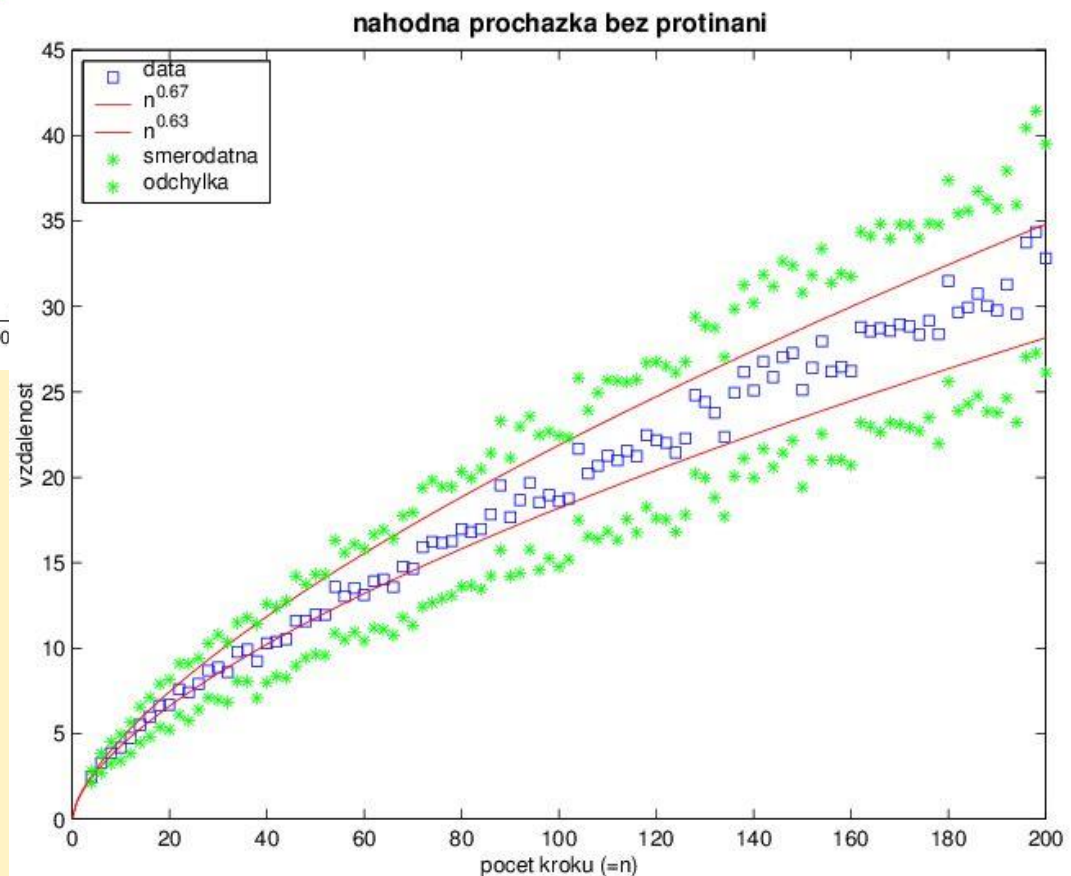
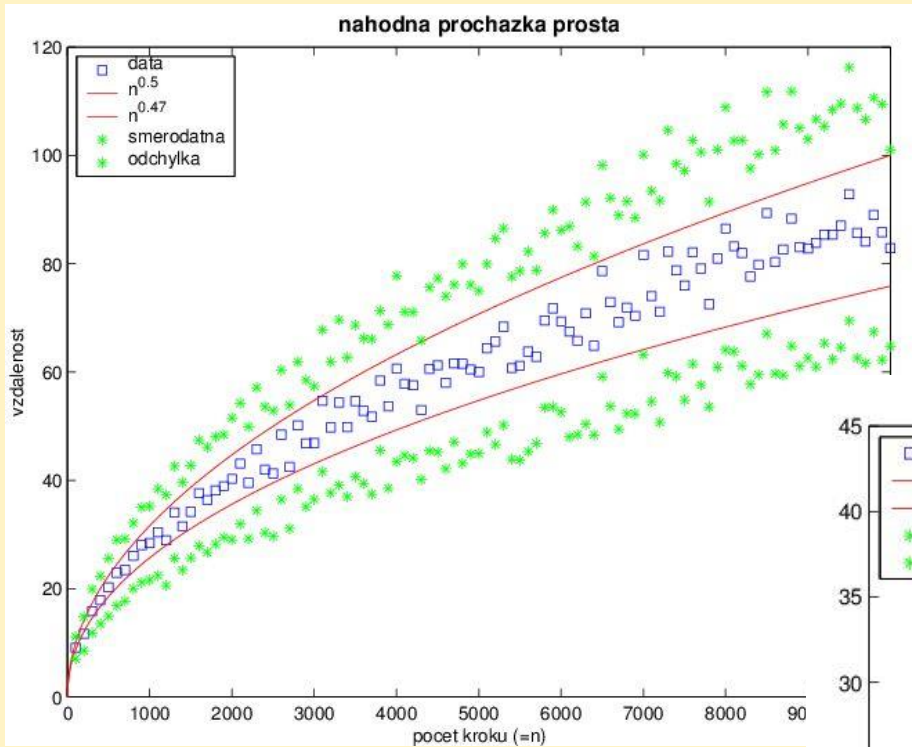
V daném kroku náhodné procházky určete směrodatnou odchylku $\Delta(n)$ měřené střední hodnotu vzdálenosti polohy částice v závislosti na počtu běhu n .

Vykreslete odchylky $\Delta(n)$ - error bars – pro vybrané body procházky.

V **gnuplotu** lze použít speciální příkaz:
plot "Graph_Data.dat" using 1:2:3 with yerrorlines
Chyba je v třetím sloupci souboru
"Graph_Data.dat"

Náhodné procházky - řešení

p. Psíkal



Cíl úlohy srovnání exponentů

Odhad exponentů:

Vyneste funkci $\langle R(t) \rangle$ v log-log měřítku a určete exponenty časové závislosti τ^α pro jednotlivé typy procházek (prostá procházka a procházka bez okamžitých návratů). Udejte chyby získaných exponentů.

Bonus: samovyhýbající se procházka

Provedte podobný výpočet pro procházku bez protínání (self-avoiding walk SAW) a srovnejte vývoj s předchozími typy procházek.

Minimálním cílem je ukázat, že SAW má jiný exponent – u SAW se lze omezit na menší počet kroků.