ÚVOD DO METODY MONTE CARLO

Jiří Dřímal, David Trunec, Antonín Brablec

katedra fyzikální elektroniky
Brno, duben 2006

přírodovědecká fakulta
Masarykova univerzita

Obsah

Kapitola 1

Úvod

1.1 Základní ideje metody Monte Carlo

Metodami Monte Carlo se nazývají numerické metody řešení matematických úloh pomocí modelování náhodných veličin a statistického odhadu jejich charakteristik.

V souvislosti s tímto podtrhněme dvě následující fakta:

- Metoda Monte Carlo je numerická metoda (může tedy konkurovat klasickým numerickým metodám a ne analytickým metodám řešení úloh).
- Metodou Monte Carlo je možné řešit libovolné matematické úlohy (a nejen úlohy pravděpodobnostního charakteru).

Základ metody Monte Carlo byl znám již dávno; např. v roce 1873 se objevil článek A. Halla o určení čísla pomocí náhodného házení jehly na rovinu pokrytou rovnoběžkami. Tento náhodný pokus je znám pod názvem Buffonova úloha o jehle.

Představme si rovinu, na níž jsou narýsovány rovnoběžky a jejichž vzdálenost je d. Na tuto rovinu házíme náhodně jehlu délky l, $l \leq d$. Ptáme se: Jaká je pravděpodobnost, že jehla přetne některou z rovnoběžek ?

Jestliže si označíme vzdálenost středu jehly od nejbližší rovnoběžky jako x a úhel, který svírá jehla s rovnoběžkami jako φ, pak poloha jehly na rovině bude popsána dvojicí (φ, x), kde $0 \leq \varphi \leq \pi$ a $0 \leq x \leq d/2$. Jehla přetne některou rovnoběžku právě tehdy, když bude platit $x \leq (l/2)\sin\varphi$. Jestliže si zobrazíme souřadnice (φ, x), dostaneme obr. 1.1.

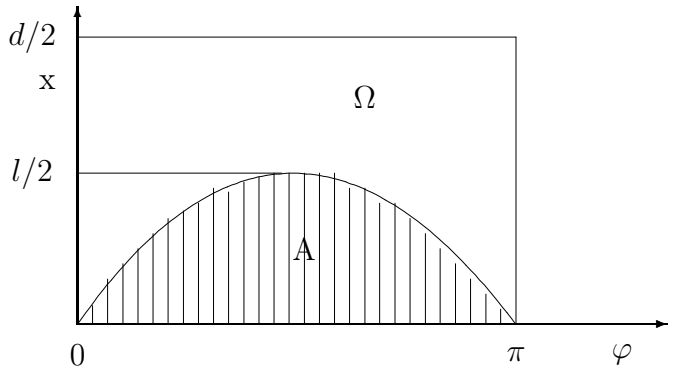
Souřadnice středu jehly (φ, x) mohou nabývat libovolné hodnoty ze čtverce Ω, k přetnutí některé rovnoběžky dojde tehdy, když souřadnice středu jehly budou ležet ve vyšrafované oblasti A.

Pravděpodobnost P protnutí rovnoběžky bude rovna poměru plochy oblasti A a plochy čtverce Ω. Tedy

$$P = \frac{P(A)}{P(\Omega)} = \frac{\int_0^{\pi} \frac{l}{2}\sin\varphi \,\mathrm{d}\varphi}{\pi d/2} = \frac{2l}{\pi d}\,. \tag{1.1}$$

V tomto experimentu je tedy pravděpodobnost přetnutí rovnoběžky jehlou vyjádřena pomocí čísla π. Jestliže provedeme n hodů a stanovíme četnost m hodů, kdy jehla přetnula rovnoběžku, můžeme potom pomocí relativní četnosti n/m odhadnout pravděpodobnost P a vypočítat číslo π podle vztahu $2l/(\pi d) \doteq m/n$.

Například Volf v roce 1850 provedl 5000 hodů a dostal pro π odhad 3.1596. Je jasné, že shora popsaná metoda určení čísla je značně zdlouhavá. Po zavedení počítačů se však naskytla příležitost tento pokus nasimulovat na počítači a rychlost „pokusů" o několik řádů zrychlit.



Obrázek 1.1: Souřadnice středu jehly.

Podobné přístupy se objevovaly i v pozdějších pracích, ale k většímu rozšíření metody Monte Carlo došlo až po rozšíření počítačů.

V roce 1944 J. Neumann navrhl použít aparát pravděpodobnosti za pomocí počítače při vývoji atomové bomby. Na rozpracování metody se dále podíleli S. A. Ulam, N. Metropolis, H. Kahn a E. Fermi. Termín „metoda Monte Carlo" byl poprvé použit v práci N. Metropolise a S. Ulama v roce 1949. Od té doby se objevilo mnoho dalších prací [1].

Přehled postupů metody Monte Carlo, zejména v matematice a fyzice, lze nalézt v monografiích [2] a [3]. Metoda Monte Carlo tedy využívá aparát teorie pravděpodobnosti a matematické statistiky. V následujících dvou oddílech jsou stručně shrnuty základní pojmy a věty teorie pravděpodobnosti a matematické statistiky a zavedena terminologie tak, jak bude používána dále. Hlubší výklad teorie pravděpodobnosti a matematické statistiky lze nalézt v [4].

1.2 Teorie pravděpodobnosti

Základním pojmem v teorii pravděpodobnosti je pojem náhodný pokus. Pokus je vymezen podmínkami, které se při jeho provádění dodržují a množinou výsledků, jejichž výskyt sledujeme. Výsledky náhodných pokusů se při opakovaném provádění náhodně mění a nejsme schopni předpovědět, jaký výsledek pokusu při daném provádění nastane, přestože jsou podmínky pokusu stále dodržovány.

Jednotlivé možné výsledky pokusu se nazývají elementární jevy, množina všech možných výsledků pokusu se nazývá prostorem elementárních jevů. Prostor elementárních jevů se označuje Ω, jednotlivé elementární jevy označujeme ω (popř. ω s indexy). Dále se na prostoru elementárních jevů zavádí systém podmnožin prostoru Ω, který je uzavřený vzhledem ke spočetnému sjednocení a operaci tvoření doplňku vzhledem k množině Ω. Tento systém se nazývá (množinová) $\sigma-$algebra $\mathcal{A}$ a její prvky se nazývají náhodné jevy vzhledem k $\mathcal{A}$. Náhodné jevy budeme označovat velkými tiskacími písmeny ze začátku latinské abecedy. Uspořádaná dvojice $(\Omega, \mathcal{A})$ se nazývá jevové pole. Na $\sigma-$algebře se pak

definuje pravděpodobnost, jakožto množinová funkce $P : \mathcal{A} \to R$, která splňuje následující podmínky:

1.
$$P(\Omega) = 1 \tag{1.2}$$

2.
$$P(A) \geq 0 \text{ pro všechny náhodné jevy} \in \mathcal{A} \tag{1.3}$$

3. pro každou posloupnost jevů A , které jsou navzájem disjunktní (tj. $A_i \bigcap A_j = 0$ $\forall i, j$) platí
$$P(\bigcup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} P(A_i). \tag{1.4}$$

Pro libovolný jev A číslo $P(A)$ nazýváme pravděpodobností jevu A a uspořádanou trojici $(\Omega, \mathcal{A}, P)$ nazýváme pravděpodobnostní prostor. Dalším velmi důležitým pojmem je pojem náhodné veličiny. Předpokládejme, že $(\Omega, \mathcal{A}, P)$ je pevně daný pravděpodobnostní prostor. Pak reálnou funkci $X : \Omega \to R$ nazýváme náhodnou veličinou na jevovém poli $(\Omega, \mathcal{A})$, jestliže pro každé reálné číslo $x \in R$ platí

$$\{\omega \in \Omega, X(\omega) < x\} \in \mathcal{A} \, . \tag{1.5}$$

Protože pojem náhodné veličiny má zásadní význam pro metodu Monte Carlo, budeme se tomuto pojmu věnovat podrobněji.

Náhodnou veličinu můžeme chápat jako číselné ohodnocení jednotlivých výsledků pokusu. Náhodnou veličinou je např. velikost chyby při fyzikálním měření, velikost výhry ve Sportce apod. Jaké hodnoty ovšem náhodná veličina nabude při konkrétním pokusu nemůžeme předpovědět, protože nemůžeme předpovědět výsledek provádění pokusu.

Ilustrujme si uvedené pojmy na příkladě. Uvažujme pokus, který spočívá ve dvou hodech jednou mincí a předpokládejme, že při každém hodu jsou možné jen dva výsledky: „padne líc" (označíme L) a „padne rub" (označíme R). Pak můžeme jako výsledek uvažovaného pokusu pozorovat vždy právě jeden z elementárních jevů

$$\omega_1 = (L, L), \quad \omega_2 = (L, R), \quad \omega_3 = (R, L), \quad \omega_4 = (R, R) \, . \tag{1.6}$$

Prostor elementárních jevů tedy obsahuje čtyři prvky, $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4\}$, $\sigma-$algebra $\mathcal{A}$ náhodných jevů bude

$$\begin{aligned} \mathcal{A} \;=\; & \{0,\, \{\omega_1\},\, \{\omega_2\},\, \{\omega_3\},\, \{\omega_4\},\, \{\omega_1,\, \omega_2\},\, \ldots,\, \{\omega_1,\, \omega_2,\, \omega_3\}, \\ & \{\omega_2,\, \omega_3,\, \omega_4\},\, \{\omega_1,\, \omega_2,\, \omega_4\}, \{\omega_1,\, \omega_3,\, \omega_4\},\, \Omega\} \, . \end{aligned} \tag{1.7}$$

Pokud mince bude homogenní a symetrická, je zřejmé, že pravděpodobnosti elementárních jevů $\omega_1,\ \omega_2,\ \omega_3,\ \omega_4$ budou stejné a tedy rovny $\frac{1}{4}$. Z vlastností pravděpodobnosti pak můžeme vypočítat pravděpodobnost libovolného náhodného jevu.

Definujme nyní náhodnou veličinu X, která nám bude udávat počet líců, které padly při daném provádění pokusu. Bude tedy

$$X(\omega_1) = 2, \quad X(\omega_2) = 1, \quad X(\omega_3) = 1, \quad X(\omega_4) = 0 \, . \tag{1.8}$$

Hodnotu náhodné veličiny X ovšem můžeme určit až po provedení pokusu.

Náhodné veličiny (tj. funkce) budeme označovat velkými tiskacími písmeny z konce latinské abecedy, např. $X, Y, \ldots$, hodnoty náhodných veličin (tj. reálná čísla) budeme označovat odpovídajícími malými tiskacími písmeny z konce latinské abecedy, např. $x = X(\omega),\ y = Y(\omega), \ldots$.

Hodnoty náhodné veličiny (realizace náhodné veličiny) se v literatuře o metodě Monte Carlo nazývají náhodná čísla. Je ovšem třeba rozlišovat mezi náhodnou veličinou (tj. funkcí) a náhodným číslem (tj. reálné číslo). Je zřejmé, že na počítači můžeme pracovat pouze s náhodnými čísly.

Vlastnosti náhodné veličiny jsou plně popsány tzv. distribuční funkcí. Distribuční funkce $F : R \to R$ náhodné veličiny X, definované na pravděpodobnostním prostoru $(\Omega,\ \mathcal{A},\ P)$, je definována vztahem

$$F(x) = P(\{\omega \in \Omega,\ X(\omega) < x\}) \tag{1.9}$$

pro všechna $x \in R$. Distribuční funkce existuje pro libovolnou náhodnou veličinu. Poznamenejme, že bývá zvykem místo pravé strany (1.9) psát

$$P(X \leq x)\ , \tag{1.10}$$

smysl ovšem zůstává týž. Obdobně se zkracuje zápis i v jiných případech. Mezi náhodnými veličinami mají největší význam dva druhy náhodných veličin - diskrétní náhodné veličiny a spojité náhodné veličiny. Náhodná veličina se nazývá diskrétní, jestliže množina jejích hodnot je nejvýše spočetná. Funkce $P : R \to R$ definovaná vztahem

$$P(x) = P(X = x) \tag{1.11}$$

se nazývá pravděpodobnostní funkcí diskrétní náhodné veličiny X.

Distribuční funkce F náhodné veličiny X indukuje Lebesgue - Stieltjesovu míru μ_F na borelovské $\sigma-$algebře v R. Míra μ_F se nazývá rozdělení pravděpodobnosti náhodné veličiny X. Jestliže existuje nezáporná funkce $f : R \to R$ taková, že pro libovolnou borelovskou množinu B platí

$$\mu_F(B) = \int\limits_B f(x)\,\mathrm{d}x\ , \tag{1.12}$$

pak se náhodná veličina X nazývá absolutně spojitá. Funkce f se nazývá hustota rozdělení pravděpodobnosti nebo též hustota náhodné veličiny.

V souvislosti s těmito pojmy uvedeme poznámku o terminologii používané v literatuře. Nechť např. X je absolutně spojitá náhodná veličina s hustotou

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\dfrac{(x-\sigma)^2}{2\sigma^2}} \qquad \mu,\ \sigma \in R,\ \sigma > 0\ . \tag{1.13}$$

Náhodná veličina s touto hustotou se nazývá normální náhodná veličina. Hovoříme rovněž o normálně rozdělené náhodné veličině nebo o náhodné veličině s normálním rozdělením. Hodnoty (realizace) této náhodné veličiny, tj. náhodná čísla, se pak nazývají náhodná čísla s normálním rozdělením nebo normálně rozdělená náhodná čísla. Rozdělení náhodné veličiny, popsané hustotou (1.13) se označuje $N(\mu, \sigma)$. Je-li X normální náhodná veličina, píšeme $X \sim N(\mu, \sigma)$. O dalších významných rozděleních a jejich označování bude pojednáno později.

Jestliže X je náhodná veličina, definovaná na pravděpodobnostním prostoru $(\Omega, \mathcal{A}, P)$ a existuje a je konečný Lebesgueův integrál

$$\int_{\Omega} X(\omega)\,\mathrm{d}P\ ,$$

pak se číslo

$$E(X) = \int_{\Omega} X(\omega)\,\mathrm{d}P \tag{1.14}$$

nazývá střední hodnota náhodné veličiny X. Je-li X absolutně spojitá náhodná veličina s hustotou f a Y náhodná veličina, definovaná vztahem

$$Y = g(X), \tag{1.15}$$

kde $g : R \to R$ je borelovsky měřitelná funkce, pak ze vzorce (1.14) dostáváme vztahy

$$E(X) = \int_{-\infty}^{\infty} xf(x)\,\mathrm{d}x,\ E(Y) = \int_{-\infty}^{\infty} g(x)f(x)\,\mathrm{d}x\ . \tag{1.16}$$

Číslo

$$D(X) = E\left((X - E(X))^2\right)\ , \tag{1.17}$$

pokud existuje, se nazývá rozptyl náhodné veličiny X a číslo $\sigma_x = \sqrt{D(X)}$ se nazývá směrodatná odchylka náhodné veličiny X.

Z definice a vlastností střední hodnoty a rozptylu (disperze) náhodné veličiny X vyplývá význam těchto veličin. $E(X)$ charakterizuje polohu rozdělení náhodné veličiny, zatímco $D(X)$ charakterizuje kolísání hodnot náhodné veličiny X kolem střední hodnoty. Vzájemný vztah těchto dvou charakteristik je charakterizován Čebyševovou nerovností

$$P(|X - E(X)| \geq \varepsilon) \leq \frac{D(X)}{\varepsilon^2}\ , \tag{1.18}$$

která platí pro libovolnou náhodnou veličinu X s konečným rozptylem a pro libovolné $\varepsilon > 0$.

Při použití metod matematické statistiky se často odhaduje neznámá veličina pomocí aritmetického průměru hodnot náhodné veličiny. O chování aritmetických průměrů nás informují následující věty.

Věta 1 *(silný zákon velkých čísel) Nechť X_i je posloupnost nezávislých stejně rozdělených veličin, které mají konečnou střední hodnotu $E(X_i) = \mu$. Položme*

$$\bar{X_n} = \frac{1}{n}\sum_{i=1}^{n} X_i\ . \tag{1.19}$$

Pak platí

$$P(\lim_{n\to\infty} \bar{X_n} = \mu) = 1\ . \tag{1.20}$$

Říkáme, že posloupnost $\bar{X_n}$ konverguje skoro jistě k μ vzhledem k pravděpodobnosti P. Tato konvergence se poněkud odlišuje od obyčejné konvergence, protože v některých případech (jejichž pravděpodobnost je ovšem nulová) nemusí $\bar{X_n}$ konvergovat k μ.

Rozdělení pravděpodobnosti aritmetických průměrů popisuje tzv. centrální limitní věta.

Věta 2 *(centrální limitní věta) Nechť X_i je posloupnost nezávislých stejně rozdělených náhodných veličin se střední hodnotou μ a nenulovým rozptylem $D(X)$. Pak posloupnost distribučních funkcí náhodných veličin*

$$Y_n = \frac{1}{\sigma\sqrt{n}} \sum_{i=1}^{n} (X_i - \mu) \ . \tag{1.21}$$

konverguje pro $n \to \infty$ k distribuční funkci normálního rozdělení $N(0,1)$ pro všechna $x \in R$.

1.3 Matematická statistika

V teorii pravděpodobnosti jsme vyšli od pojmu náhodného pokusu, k jeho popisu jsme zavedli pravděpodobnostní prostor $(\Omega,\ \mathcal{A},\ P)$. Výsledky pokusu obvykle sledujeme pomocí náhodných veličin, které jsou definovány na pravděpodobnostním prostoru $(\Omega,\ \mathcal{A},\ P)$.

Cílem teorie pravděpodobnosti jsou výpovědi o hodnotách a vlastnostech náhodných jevů a náhodných veličin, přičemž pravděpodobnostní prostor (a tedy i pravděpodobnost) je znám a pevně dán.

V matematické statistice je situace obrácená. Obvykle jsou známy některé výsledky náhodného pokusu, které pozorujeme prostřednictvím hodnot jedné nebo více náhodných veličin a z těchto výsledků máme usuzovat na nějaký pravděpodobnostní prostor (především na pravděpodobnost), který by odpovídal sledovanému pokusu. Přitom často předpokládáme, že výsledky pokusu, tedy jednotlivá pozorování, které máme k dispozici, jsou hodnoty nezávislých náhodných veličin, které mají stejné rozdělení pravděpodobnosti s distribuční funkcí, o níž víme, že je prvkem nějaké množiny distribučních funkcí $\{F_\gamma,\ \gamma \in \Gamma\}$. Úkolem potom je pomocí naměřených dat odhadnout hodnotu parametru γ, který se v našem pokusu realizoval. Protože výsledky pokusu jsou hodnoty náhodných veličin můžeme formulovat pojmy matematické statistiky pro náhodné veličiny.

Jestliže $X_1,\ X_2,\ \ldots,\ X_n$ jsou nezávislé náhodné veličiny, které mají stejné rozdělení pravděpodobnosti s distribuční funkcí F_γ, pak n-tici $X_1,\ X_2,\ \ldots...,\ X_n$ nazýváme náhodným výběrem rozsahu n z rozdělení o distribuční funkci F_γ. Libovolnou náhodnou veličinu, která je funkcí jednoho nebo více náhodných výběrů nazveme statistikou.

Pro praxi mají největší význam dvě následující statistiky

$$\bar{X_n} = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{1.22}$$

a

$$S_n^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{X_n})^2 \ , \tag{1.23}$$

které se nazývají výběrový průměr a výběrový rozptyl. Výběrový průměr a výběrový rozptyl jsou nestrannými a konzistentními odhady střední hodnoty a rozptylu náhodných veličin X.

Připomeňme si už jen na závěr, že pro normální rozdělení na základě výběrového průměru a výběrového rozptylu můžeme sestrojit 100 (1 - α) % interval $(D,\ H)$ spolehlivosti pro neznámou střední hodnotu

$$(D, H) = \left(\bar{X_n} - t_{1-\frac{\alpha}{2}}(n-1)\frac{S_n}{\sqrt{n}},\ \bar{X_n} + t_{1-\frac{\alpha}{2}}(n-1)\frac{S_n}{\sqrt{n}} \right) \ , \tag{1.24}$$

kde $t_{1-\frac{\alpha}{2}}(n-1)$ je $(1-\frac{\alpha}{2})$ kvantil Studentova rozdělení s $n-1$ stupni volnosti.

1.4 Metoda Monte Carlo

Nejčastější postup metody Monte Carlo je modelování takové náhodné veličiny X, že její střední hodnota $E(X)$ je rovna hledané hodnotě a. Přesněji řečeno: abychom mohli přibližně vypočítat skalární veličinu a, musíme najít takovou náhodnou veličinu X, že platí

$$E(X) = a \ . \tag{1.25}$$

Pak, jestliže vypočteme n nezávislých realizací $X_1, ; X_2, \ldots, X_n$ náhodné veličiny X, můžeme odhadnout a pomocí aritmetického průměru

$$a \doteq \frac{1}{n}(X_1 + X_2 + ... + X_n) \ . \tag{1.26}$$

Obecně řečeno, existuje nekonečně mnoho náhodných veličin X takových, že

$$E(X) = a \ . \tag{1.27}$$

Proto teorie metody Monte Carlo musí odpovědět na dvě otázky:

1. Jak vybrat náhodnou veličinu X, resp. jak vybrat nejvhodnější náhodnou veličinu X pro výpočet hledané veličiny ?

2. Jak najít hodnoty $x_1,\ x_2,\ \ldots$ libovolné náhodné veličiny X ?

Odpovědi na tyto otázky budou dány v následujících kapitolách. Co se týče druhé otázky lze říci, že v převážné většině metod Monte Carlo se postupuje tak, že se nejdříve generují hodnoty náhodné veličiny Y rovnoměrně rozdělené na intervalu (0, 1) a ty se pak transformují transformací

$$x_i = f(y_i,\ y_{i-1},\ \ldots) \ , \tag{1.28}$$

kde f je vhodně zvolená funkce, na hledané hodnoty $x_1,\ x_2,\ \ldots$. Výpočet hodnot x_i nemusí být zadán přímo funkční závislostí, ale může se provádět pomocí vhodného algoritmu. Druhá otázka se tedy redukuje na otázku generování náhodných čísel rovnoměrně rozdělených na intervalu (0, 1) a nalezení vhodné transformace. Obecné schéma výpočtu metodou Monte Carlo je tedy následující:

1. Generují se náhodná čísla y_i s rovnoměrným rozdělením na intervalu $(0, 1)$;

2. náhodná čísla y_i se transformují na náhodná čísla z_i se složitějším rozdělením;

3. pomocí náhodných čísel z_i se buď přímo počítají odhady charakteristik náhodné veličiny X (v tomto případě $z_i = x_i$) nebo se počítají pomocí vhodného algoritmu hodnoty x_i a odhady charakteristik náhodné veličiny;

4. získané výsledky se statisticky zpracují.

Odhad chyby můžeme získat pomocí Čebyševovy nerovnosti (1.18). Předpokládejme tedy, že odhadujeme střední hodnotu $E(X)$ pomocí aritmetického průměru z n hodnot x_1, x_2, $\ldots$, x_n náhodné veličiny X. Jestliže zvolíme dostatečně malé α a položíme

$$\varepsilon = \sqrt{\frac{D(X)}{n\alpha}}\,, \tag{1.29}$$

dostaneme

$$P\left(\left|\frac{1}{n}\sum_{i=1}^{n} x_i - E(X)\right| \leq \sqrt{\frac{D(X)}{n\alpha}}\right) \geq 1-\alpha\,, \tag{1.30}$$

což znamená, že s pravděpodobností $1-\alpha$ se aritmetický průměr nezávislých realizací náhodné veličiny X neliší od $E(X)$ více než $\sqrt{D(X)/n}$. Při fixovaných α a $D(X)$ chyba klesá jako $1/\sqrt{n}$.

Jestliže jsou splněny předpoklady centrální limitní věty, pak při dostatečně velkém n můžeme předpokládat, že veličina $\sum x_i/n$ má normální rozdělení se střední hodnotou $E(X)$ a disperzí $D(X)/n$. To nám dává možnost sestrojit interval spolehlivosti pro hledanou hodnotu $E(X)$, tak, jak bylo uvedeno v odstavci věnovaném matematické statistice. Hodnota disperze, která určuje chybu, může být určena jedině v průběhu výpočtu. Můžeme tedy v průběhu výpočtu určit číslo n, které nám zabezpečí požadovanou chybu se zadanou pravděpodobností.

Jak je vidět metoda Monte Carlo je spojena s úlohou odhadu parametrů normálního rozdělení. Přitom odhad střední hodnoty nám dává hledanou hodnotu a odhad disperze zabezpečuje odhad chyby. Třebaže řád ubývání chyby $O(1/\sqrt{n})$ není příliš vysoký, pro přesné výpočty existují různé transformace náhodných veličin, které zachovávají střední hodnoty a snižují disperzi náhodné veličiny, čímž se sníží i chyba metody Monte Carlo.

Odhad chyby a konvergence metody Monte Carlo má tu zvláštnost oproti klasickým numerickým metodám, že zde nejde o obyčejnou konvergenci, resp. obyčejný odhad chyby, ale o konvergenci skoro jistě a pravděpodobnostní odhad chyby. Tím se ovšem cena metody Monte Carlo nijak nesnižuje, protože např. výsledky měření mají stejný charakter jako výsledky metody Monte Carlo.

Kapitola 2

Generování rovnoměrných náhodných čísel

Výpočetní schéma Monte Carlo předpokládá modelování náhodného procesu pomocí operací s náhodnými čísly. Podmínkou úspěchu je možnost náhodný proces mnohonásobně opakovat. K tomu je zapotřebí mít k dispozici dostatečný počet náhodných čísel s nejrůznějšími distribučními funkcemi F. Jak vyplyne z dalšího, není zapotřebí sestrojovat speciální generátory náhodných čísel pro každý typ distribuční funkce, ale vystačíme s generátorem náhodných čísel $R(0,1)$ s rovnoměrným rozdělením na intervalu $(0,1)$. Náhodné veličiny s jiným typem rozdělení lze potom získat z $R(0,1)$ buď metodou inverzní funkce či metodou zamítací.

Pojem náhodných čísel a náhodných číslic je důležitý pro další výklad. Pod pojmem náhodné číslice rozumíme konečnou posloupnost číslic, kterou lze považovat za posloupnost realizací nezávislých náhodných veličin z diskrétního rozdělení

$$P(X = i) = 10^{-1}, \quad i = 0, 1, \ldots, \ . \tag{2.1}$$

Pod pojmem *náhodná čísla* zde rozumíme konečnou posloupnost čísel z intervalu $(0,1)$, kterou lze považovat za posloupnost realizací nezávislých náhodných veličin z rovnoměrného rozdělení $R(0,1)$.

Náhodné číslice takto zavedené jsou vlastně dekadické náhodné číslice. Zcela analogicky však můžeme zavést pojem náhodných číslic o jiném číselném základu, např. binární náhodné číslice.

Jak vyplývá z následující věty, je jedno, zda máme k dispozici náhodné číslice nebo náhodná čísla.

Věta 3

1. *Nechť $X_1,\ X_2,\ \ldots$ je posloupnost nezávislých stejně rozdělených náhodných veličin s rozdělením (2.1). Potom náhodná veličina*

$$Y = \sum_{i=1}^{\infty} 10^{-i} X_i \tag{2.2}$$

   má rovnoměrné rozdělení R(0, 1).

2. *Nechť náhodná veličina Y má rovnoměrné rozdělení R(0, 1). Potom posloupnost $X_1, X_2, \ldots$ z desetinného rozvoje $Y = \sum 10^{-i} X_i$ je posloupnost nezávislých stejně rozdělených náhodných veličin s rozdělením (2.1).*

Důkaz. Nechť $y \in (0,\, 1)$ a nechť jeho dekadický rozvoj je $y = \sum_{i=1}^{\infty} 10^{-i} x_i$. Jev $\{Y < y\}$ lze vyjádřit jako sjednocení disjunktních jevů

$$\{Y < y\} = \bigcup_{i=1}^{\infty} \left[\bigcap_{k=1}^{i-1} \{X_k = x_k\} \cap X_i < x_i \right] \quad .$$

V důsledku nezávislosti je

$$P(Y < y) = \sum_{i=1}^{\infty} P(X_i < x_i) \prod_{k=1}^{i-1} P(X_k = x_k) = \sum_{i=1}^{\infty} 10^{-1} x_i 10^{1-i} = \sum_{i=1}^{\infty} 10^{-i} x_i = y,$$

což dokazuje první část věty. Druhou část věty dokážeme indukcí. Nechť x_i je nějaká dekadická číslice. Je-li $Y = \sum 10^{-i} X_i$ dekadický rozvoj Y, je $P(X_1 = x_1) = P(10^{-1} x_1 < Y < 10^{-1}(x_1 + 1)) = 10^{-1}$. Předpokládejme nyní, že $X_1,\ \ldots,\ X_{n-1}$ jsou nezávislé stejně rozdělené náhodné veličiny s rozdělením (2.1) a $x_1, \ldots, x_n$ jsou nějaké dekadické číslice. Potom $P(X_1 = x_1,\ \ldots,\ X_n = x_n) = P(\sum_{i=1}^{n} 10^{-i} x_i \le Y < \sum_{i=1}^{n} 10^{-i} x_i + 10^{-n}) = 10^{-n}$. Spolu s indukčním předpokladem z toho vyplývá, že $X_1,\ \ldots,\ X_n$ jsou nezávislé stejně rozdělené náhodné veličiny s rozdělením (2.1), z čehož vyplývá tvrzení věty.

2.1 Generátory náhodných čísel

Náhodná čísla bývají získávána několika způsoby. O některých historicky nejzajímavějších a prakticky nejdůležitějších se nyní zmíníme.

Jedním z nejjednodušších generátorů je obyčejná hrací kostka, poskytující náhodné číslice 1, 2, ..., 6. Pro praxi je však nejvýhodnější mít k dispozici náhodná čísla 0, 1, 2, ..., 9, a proto byly pro tyto účely zkonstruovány desetistěnné kostky. Využití hracích kostek jako generátorů náhodných číslic má však omezené užití jednak pro praktickou neuskutečnitelnost dostatečně dlouhé řady pokusů nutné ve většině praktických úloh a jednak pro nepravidelnosti takto získaných posloupností náhodných čísel, které se i přes sebepečlivější výrobu kostek projeví.

Jako příklad generátoru binárních náhodných číslic může sloužit *házení mincí*. Padne-li rub, vezmeme číslici 0, padne-li líc, vezmeme číslici 1. I zde se však může projevit nepravidelnost mince, takže např. získáváme 1 s pravděpodobností p a 0 s pravděpodobností $q = 1 - p$, přičemž $p \neq 1/2$. Ukažme si, jak můžeme eliminovat vzniklé nepravidelnosti pravděpodobnostními prostředky. Generujeme-li uvedeným způsobem dvojice binárních náhodných číslic, má (za předpokladu, že jednotlivé pokusy konáme nezávisle) dvojice 11 pravděpodobnost p^2, 10 pravděpodobnost pq, 01 pravděpodobnost qp a 00 pravděpodobnost q^2. Vidíme, že pravděpodobnost dvojice 10 je stejná jako pravděpodobnost 01. Můžeme tedy uvažovat novou posloupnost binárních číslic, sestrojenou na základě dvojic generovaných výše uvedeným způsobem. Při výsledku 10 vezmeme 1, při výsledku 01 vezmeme 0. Dvojice 11 a 00 ignorujeme. Nevýhodou uvedeného postupu je přibližně čtyřnásobný počet hodů mincí ve srovnání s původním generováním. Je zřejmé, že uvedený obrat můžeme využít nejen při házení mincí.

Pro jednodušší manuální výpočty jsou vítanou pomůckou tabulky náhodných čísel. V historii jich byla publikována celá řada. Některé z nich využívaly sestav ze sčítání lidu, jiné prostředních číslic účastnických čísel v telefonních seznamech apod. Přesto, že tyto

tabulky nebyly příliš rozsáhlé (řádově desítky tisíc), odhalily statistické testy nepříznivé vlastnosti takovýchto tabulek.

Nejrozsáhlejší tabulky (milión dekadických náhodných číslic)byly publikovány společností RAND. Byly pořízeny pomocí tzv. „elektronické rulety". I zde se objevily jisté nepravidelnosti a tabulky byly speciálními metodami upravovány.

Pro řešení praktických problémů je třeba zhruba několika desítek tisíc náhodných čísel k tomu, aby bylo dosaženo žádané přesnosti výsledků. Kdyby bylo nutné uchovávat v paměti takové množství náhodných čísel spolu s programem výpočtu a dalšími daty, docházelo by k tomu, že by operační paměť i velkých počítačů nestačila tolik informací pojmout.

Je možné použít vnějších pamětí (diskety, disky, atd.) pro uskladnění tabulek náhodných čísel. To však vede k nutnosti komunikace mezi vnitřní a vnější pamětí, která značně prodlužuje výpočty. Už jenom zavedení těchto tabulek do paměti počítače je zdlouhavá záležitost. Navíc se může stát, že simulační experiment vyžaduje více náhodných čísel než obsahují tabulky.

Kromě těchto technických potíží lze narazit i na další, závažnější. Nemusí být totiž přípustné používat stále stejnou posloupnost náhodných čísel pro všechny problémy.

Užívání tabulek náhodných čísel není tedy příliš vhodné ve spojení se samočinným počítačem. Vhodnějším zdrojem náhodných čísel v tomto případě je *fyzikální generátor*, připojený k počítači a generující v potřebný okamžik náhodné číslo. Tento typ generátoru je založen na analýze náhodných fyzikálních pochodů, jako jsou radioaktivní rozpad, šum elektronických prvků, apod.

Uveďme si nejprve činnost generátoru binárních náhodných čísel založeného na detekci počtu emitovaných částic z nějakého radioaktivního materiálu. Detektor (čítač pulsů) zaznamenává počet pulsů v intervalu délky Δt. Je-li počet pulsů sudý, vezme se 0, je-li lichý, vezme se 1. Nezávislost takto generovaných číslic plyne z fyzikální podstaty. Je-li pravděpodobnost vyzáření jedné částice za infinitezimální čas δt rovna $\lambda \delta t$, potom pravděpodobnost vyzáření n částic za konečný časový interval délky Δt je dána Poissonovým rozdělením

$$P(X = n) = \frac{(\lambda \Delta t)^n}{n!} e^{-\lambda \Delta t} \quad , n = 0, 1, \ldots \tag{2.3}$$

se střední hodnotou počtu částic zaznamenaných za dobu Δt rovnou $\lambda \Delta t$.

Pravděpodobnost toho, že v daném intervalu bude registrován sudý počet částic je

$$e^{-\lambda \Delta t} \sum_{k=0}^{\infty} \frac{(\lambda \Delta t)^{2k}}{(2k)!} = e^{-\lambda \Delta t} \cosh(\lambda \Delta t) = \frac{1}{2} + \frac{1}{2} e^{-2\lambda \Delta t} \tag{2.4}$$

a pravděpodobnost toho, že bude registrován lichý počet částic je

$$e^{-\lambda \Delta t} \sum_{k=0}^{\infty} \frac{(\lambda \Delta t)^{2k+1}}{(2k+1)!} = e^{-\lambda \Delta t} \sinh(\lambda \Delta t) = \frac{1}{2} - \frac{1}{2} e^{-2\lambda \Delta t} \, . \tag{2.5}$$

Je tedy vidět, že obě číslice dostaneme s přibližně stejnou pravděpodobností pro $\lambda \Delta t$ t dostatečně velké, například řádově $(1/2) \ln 0.01$. Zde dochází k praktickým potížím. Abychom dostávali náhodné číslice dostatečně rychle, musí být Δt malé. Volíme-li látku s velkým λ, začíná se projevovat tzv. „mrtvá doba" čítače a čítač přestává spolehlivě fungovat. Během této mrtvé doby, která je určena fyzikální podstatou procesu detekce, je detektor necitlivý, tj. dopadající částice nejsou během této doby registrovány.

Vyšetřeme nyní druhý způsob získání náhodných veličin X založený na detekci vlastního šumu elektronek. V elektronkách jsou vždy vlastní šumy, způsobené nepravidelným uvolňováním elektronů z katody. Zesílením tohoto šumového spektra můžeme získat dostatečně silné výstupní napětí $U(t)$, které bude náhodnou veličinou vzhledem k času t. Hodnoty této náhodné veličiny lze vybírat v časových okamžicích $t_1,\ t_2,\ \ldots,\ t_k$ dostatečně vzdálených od sebe, aby bylo možno s jistotou považovat veličiny $U(t_1),\ U(t_2),\ \ldots,\ U(t_k)$ za nezávislé.

Definujme nyní veličinu y_i podmínkou

$$y_i = \begin{cases} 0 & U(t_i) < h \\ 1 & U(t_i) > h\ . \end{cases} \tag{2.6}$$

Je třeba vybrat hodnotu h, tj. rozhodovací hladinu tak, aby pravděpodobnost rovnosti $P(y = 1)$ byla rovna jedné polovině. Hlavní potíž tkví ve správné volbě hodnot h. Abychom zajistili, že pravděpodobnost výskytu jednotky, tudíž i nuly, je co možná nejbližší k jedné polovině, používáme různých způsobů zpřesnění. V nejjednodušším případě se kontroluje dvojice hodnot y_i a y_{i+1} a vytváří se veličina x_i podle pravidla

$$X_i = \begin{cases} 1 & y_i = 1;\ y_{i+1} = 0 \\ 0 & y_i = 0;\ y_{i+1} = 1\ . \end{cases} \tag{2.7}$$

Jestliže však $y_i = y_{i+1}$, potom pro určení X_i se bere první z následujících dvojic y_{i+k}, y_{i+k+1}, které mají různé hodnoty. Pravděpodobnost, že $X_i = 1$ je dána vztahem

$$P(X_i = 1) = \frac{P(y_i = 1)[1 - P(y_i = 1)]}{P(y_i - 1)[1 - P(y_i - 1)] + [1 - P(y_i = 1)]P(y_i = 1)} = \frac{1}{2}\ . \tag{2.8}$$

Jak snadno nahlédneme, proces vyhledávání vyhovující dvojice y_{i+k}, y_{i+k+1} končí po průměrném počtu kroků

$$1/[P(y_i = 1)[1 - P(y_i = 1)]]\ . \tag{2.9}$$

Při hodnotách pravděpodobnosti $P(y_i = 1) \approx 1/2$ průměrný počet kroků činí 4. Další podrobnosti týkající se tohoto typu generátoru jsou uvedeny v [5].

Takovéto typy generátorů mohou být buď součástí počítače, nebo připojeny na jeho vstup. Kromě některých kladných zkušeností je však s jejich používáním opět spojena řada nevýhod. Mezi nejzávažnější patří, že není možné opakovaně získat stejnou posloupnost náhodných čísel a že je nutné řešit problém kontroly, zda poskytovaná čísla jsou stále náhodná. Kromě toho se okamžiky, v nichž čísla v zařízení vznikají, obvykle liší od okamžiků, ve kterých je potřebuje počítač, což využití těchto zařízení značně komplikuje.

V současné době, kdy jsou úlohy metodou Monte Carlo téměř výhradně řešeny na počítači, dominujícím typem generátorů náhodných čísel se stal třetí typ založený na aritmetických procedurách. Náhodná čísla se vytvářejí pomocí speciálních rekurentních vzorců. V pravém slova smyslu se tedy nejedná o náhodná čísla, neboť jsou získávána přísně determinovaným způsobem - označují se proto jako pseudonáhodná čísla. Na rozdíl od předešlých dvou typů generátorů lze je získat přímo počítačem a to rychle a v dostatečném počtu. Navíc není podstatné, jak náhodná čísla získáme, ale zda budou vyhovovat statistickým testům. Na základě statistických testů (viz. dále) můžeme rozhodnout, a to ještě s jistou pravděpodobností, o tom, je-li nějaká posloupnost čísel posloupností hodnot náhodné veličiny s rozdělením $R(0, 1)$.

Další problém spočívá v tom faktu, že realizace hodnot náhodné veličiny s rozdělením $R(0,1)$ předpokládá čísla s nekonečným počtem desetinných míst. V počítači však musíme pro reprezentaci náhodného čísla použít pouze a-místné celé dvojkové číslo. Náhodná čísla pak mohou nabývat hodnot

$$k\ 2^{-a}, \quad k = 0,\ 1, \ldots,\ 2^a - 1\ . \tag{2.10}$$

s pravděpodobností 2^{-a}. Dostaneme tedy pouze rovnoměrné diskrétní rozdělení. Čísla $\bar{x}$ s tímto rozdělením můžeme však transformovat na čísla x s ma desetinnými místy podle vztahu

$$x = \sum_{i=0}^{m} 2^{-ia}\ \bar{x}_i\ , \tag{2.11}$$

kde m volíme podle potřeby. Problém generování náhodných čísel se tedy nyní redukoval na problém generování rovnoměrného diskrétního rozdělení.

Předchůdcem moderních metod získávání pseudonáhodných čísel je von Neumannova *metoda "prostředních řádů druhé mocniny"*, navržená v roce 1946. Princip metody je velmi jednoduchý a lze jej vystihnout následujícím postupem:

1. volí se vhodné (např.dekadické) počáteční číslo x_0, které je zobrazeno na $2a$ dekadických místech ($a = 1,\ 2,\ \ldots$),

2. spočte se x_0^2, jehož obraz (po případném doplnění nulami zleva) je uložen v $4a$ desítkových místech,

3. prostředních $2a$ číslic z tohoto čísla x_0^2 nám vytvoří nové číslo x_i.

Takto vytvoříme celou posloupnost $2a$-místných pseudonáhodných čísel $x_0,\ x_1,\ x_2,\ \ldots$. Rekurentně lze tuto proceduru vyjádřit vztahem

$$x_{n+1} = \mathrm{Int}(x_n^2/10^a) - 10^{3a}\mathrm{Int}(x_n^2/10^{3a})\ , \tag{2.12}$$

kde $\mathrm{Int}(x)$ značí celou část čísla x.

Von Neumannova metoda má v současné době cenu pouze historickou, neboť jak se ukázalo, má celou řadu nedostatků. Je poměrně pomalá, čísla se brzy začnou opakovat a získaná posloupnost má tedy malou periodu.

V současné době daleko úspěšnější a v praxi nejpoužívanější jsou *kongruenční generátory* navržené v roce 1951 Lehmerem.

Posloupnost pseudonáhodných čísel z intervalu $\langle 0, M)$ je vytvářena pomocí rekurentního vztahu

$$x_{n+1} = a_0 x_n + \ldots + a_k x_{n-k} + b \bmod M \tag{2.13}$$

při daných počátečních hodnotách $x_0,\ \ldots,\ x_k$ a konstantách $a_0,\ \ldots,\ a_k,\ b$. Kongruencí zde míníme zbytek po dělení. Číslo x_{n+1} je tedy zbytek po dělení čísla $a_0x_n+\ldots+a_kx_{n-k}+b$ číslem M.

Náhodná čísla Y_i z intervalu $(0,1)$ pak dostaneme dělením čísla x_i číslem M

$$y_i = x_i/M\ . \tag{2.14}$$

Pro výpočet můžeme rekurentní vztah (2.13) používat ve tvaru

$$y_{n+1} = \mathrm{Fr}\left(\sum_{i=0}^{k} a_i y_{n-i} + c\right)\ , \tag{2.15}$$

s počátečními hodnotami

$$y_i = x_i/M, \quad i = 0, \ldots, k \tag{2.16}$$
$$c = b/M \ .$$

Fr(x) označuje desetinnou část čísla. Využili jsme zde věty:
Nechť $a = b \bmod m$. Potom $a = b - \mathrm{Int}(b/m)m = \mathrm{Fr}(b/m)m$.

Statistické vlastnosti i perioda tohoto generátoru závisí na číslech M, $a_0, \ldots, a_k$, b a na volbě počátečních hodnot $x_0, \ldots, x_k$. Konstanty M, $a_0, \ldots, a_k$, b se volí podle použitého typu počítače (binární či dekadický a bere se v úvahu délka slova). Například v binárním počítači se volí za M mocnina 2, v dekadickém mocnina 10, vždy pokud možno co největší. Je-li totiž generátor programován ve strojovém jazyku nebo assembleru, je možné vytvářet zbytky po dělení pomocí logických operací (posuv doleva a doprava), které jsou oproti aritmetickým operacím daleko rychlejší.

Při speciálních volbách konstant a_j a b mají generátory zvláštní názvy. Příkladem *aditivního generátoru* je Fibonacciův generátor

$$x_{n+1} = x_n + x_{n-1} \bmod M \ . \tag{2.17}$$

Lehmer uveřejnil *multiplikativní generátor*

$$x_{n+1} = ax_n \bmod M \ . \tag{2.18}$$

Spojíme-li aditivní a multiplikativní generátor dostaneme *kongruenční smíšenou metodu*

$$x_{n+1} = ax_n + b \bmod M \ . \tag{2.19}$$

2.2 Vlastnosti nejpoužívanějších generátorů pseudonáhodných čísel

V následujícím uvedeme pouze nejdůležitější výsledky týkající se vlastností generátorů pseudonáhodných čísel. Detailní rozbor této problematiky lze například nalézt v [12].

Jak jsme již uvedli, budeme na generátorech náhodných či pseudonáhodných čísel požadovat, aby poskytovaly posloupnost čísel, která má vlastnosti realizací nezávislých rovnoměrně rozdělených náhodných veličin. Odvodit teoretické charakteristiky generátorů je obtížné, v některých případech prakticky nemožné. Rovněž neexistuje žádný teoretický předpis, jak takový vhodný generátor realizovat. Velký význam má proto testování generátorů náhodných čísel a bude o něm pojednáno později. Zde se zmíníme o problému periody a o seriální korelaci nejužívanějších aritmetických generátorů. Malé periody (tj. čísla se začnou brzy opakovat) a seriální korelace (sousední nebo vzdálenější členy jsou korelované) jsou téměř největším nebezpečím, se kterým se setkáváme u generátorů využívajících rekurentní vztahy.

Všimejme si periody kongruenčního generátoru (2.13). Zde i v dalším předpokládáme, že koeficienty $a_0, \ldots, a_k$, b jsou nezáporná celá čísla, modul M číslo přirozené a počáteční hodnoty $x_0, \ldots, x_k$ nezáporná celá čísla. Jestliže $(K+1)$-tice se poprvé znovu objeví po H krocích, pak nazýváme číslo H *periodou generátoru* (2.13). Tak např. generátor

$$x_{n+1} = x_n + x_{n-1} \bmod 3 \tag{2.20}$$

s počátečními hodnotami $x_0 = x_1 = 1$ generuje posloupnost 1, 1, 2, 0, 0, 2, 2, 1, 0, 1, 1, 2, 0, ... a má periodu $H = 8$. Pro každé nezáporné m tedy platí

$$x_n = x_{n+mH} \quad n = 0, 1, \ldots \ . \tag{2.21}$$

Poznamenejme, že perioda definovaná výše uvedeným způsobem nemusí vždy existovat. Uvažujme např. multiplikativní generátor

$$x_{n+1} = 18x_n \bmod 20 \ . \tag{2.22}$$

Pro $x_0 = 1$ dostaneme posloupnost $1, 18, 4, 12, 16, 8, 4, 12 \ldots$. Počáteční hodnota 1 se tedy v generované posloupnosti již nikdy nevyskytne, stejně tak jako hodnota 18. Abychom pokryli i tyto případy, museli bychom definici periody upravit následujícím způsobem: jestliže se v generované posloupnosti nějaká $(K+1)$-tice poprvé znovu objeví po H krocích, nazveme číslo H periodou. V našich úvahách však plně vystačíme s prvně uvedenou definicí.

V $(K+1)$-tici $(x_{n-k}, \ \ldots, \ x_n)$ nabývá každý prvek jednu z M možných hodnot, takže nejpozději po M^{k+1} krocích se musí některý z generovaných vektorů zopakovat. Je tedy $H = M^{k+1}$.

V následujícím uvedeme, za jakých podmínek lze dosáhnout největší možné, tj. maximální periody u smíšené kongruenční a multiplikační metody.

Věta 4 *Nechť b a M jsou nesoudělná. Nechť $a = 1 \bmod p$ pro každý prvočinitel p prvočíselného rozkladu M. Nechť $a = 1 \bmod 4$ je-li M násobek 4. Potom má generátor*

$$x_{n+1} = ax_n + b \bmod M$$

maximální periodu rovnou M pro každé počáteční x_0.

Věta 5 *Nechť pro největší společný dělitel čísel $a, \ x_0, \ M$ platí $(a, M) = (x_0, M) = 1$. Nechť prvočíselný rozklad čísla M je*

$$M = 2^\alpha \prod_{i=1}^{r} p_i^{\beta_i} \ ,$$

kde p_i jsou různá lichá prvočísla. Položme

$$\begin{array}{rcll} \lambda(p^\beta) & = & p^{\beta-1}(p-1) & p \text{ liché} \\ \lambda(2^\alpha) & = & 1 & \alpha = 0, 1 \\ & = & 2 & \alpha = 2 \\ & = & 2 & \alpha > 2 \ . \end{array}$$

Nechť

$$\begin{array}{rcll} a^n & \neq & 1 \bmod p_i^{\beta_i} & 0 < n < \lambda(p_i^{\beta_i}) \ , \ i = 1, \ldots, r, \\ a & = & 1 \bmod 2, & \text{je-li } \alpha = 1 \ , \\ & = & 3 \bmod 4, & \text{je-li } \alpha = 2 \ , \\ & = & 3 \text{ nebo } 5 \bmod 8, & \text{je-li } \alpha > 2 \ . \end{array}$$

Potom je perioda generátoru $x_{n+1} = ax_n \bmod M$ maximální a je rovna

$$H(M) = NSN(\lambda(2^\alpha), \ \lambda(p_1^{\beta_1}), \ \ldots, \ \lambda(p : r^{\beta_r})) \ ,$$

kde NSN označuje nejmenší společný násobek.

Věta 6 *Nechť* $a = 1 \bmod 4$ *a* $b = 1 \bmod 2$. *Potom má generátor*

$$x_{n+1} = ax_n + b \bmod 2^\alpha, \quad \alpha \geq 1$$

maximální periodu rovnou 2^α *pro každé počáteční* x_0 .

Věta 7 *Nechť* $a = 3 \bmod 8$ *nebo* $a = 5 \bmod 8$ *a nechť* $\alpha \geq 3$. *Potom má generátor*

$$x_{n+1} = ax_n \bmod 2^\alpha$$

maximální periodu rovnou $2^{\alpha-2}$ *pro každé liché* x_0.

Věta 8 *Nechť* $a = 1 \bmod 20$ *a* $b = 1, 3, 7$ *nebo* $9 \bmod 10$. *Potom má generátor*

$$x_{n+1} = ax_n + b \bmod 10^\beta , \quad \beta \geq 2$$

maximální periodu rovnou 10^β *pro každé počáteční* x_0.

Věta 9 *Nechť* $a = 3, 13, 27, 37, 53, 67, 77, 83, 117, 123, 133, 147, 163, 173, 187$ *nebo* $197 \bmod 200$. *Nechť* $x_0 = 1, 3, 7$ *nebo* $9 \bmod 10$. *Potom má generátor*

$$x_{n+1} = ax_n \bmod 1000$$

maximální periodu rovnou 100 *a generátor*

$$x_{n+1} = ax_n \bmod 10^\beta , \quad \beta > 3$$

maximální periodu rovnou $5 \times 10^{\beta-2}$.

Daleko větší problémy než problémy periody přináší studium korelace sousedních resp. vzdálenějších členů posloupností generovaných výše uvedenými metodami. Uvedeme zde proto jen některé ilustrativní výsledky.

Označme ρ_k *korelační koeficient* mezi X_n a X_{n+k}, $E(X_n)$ střední hodnotu čísla a $D(X_n)$ jeho disperzi. Budeme-li předpokládat, že X_n má přibližně rovnoměrné rozdělení $R(0, M)$, potom

$$E(X_n) \approx M/2 \tag{2.23}$$
$$D(X_n) \approx M^2/12$$

a pro ρ_1 lze odvodit přibližný výraz

$$\rho_1 \approx \frac{1}{a} + \frac{6}{a}\left(\frac{b}{M}\right)^2 - \frac{6b}{aM} . \tag{2.24}$$

Pro některé kombinace hodnot a, b a M je tato aproximace dobrá, pro některé hodnoty je však nevyhovující. Jisté zpřesnění přibližného vzorce pro ρ_1 podal Greenberger, který odvodil přibližné meze pro ρ_1

$$\frac{1}{a} + \frac{6}{a}\left(\frac{b}{M}\right)^2 - \frac{6b}{aM} \pm \frac{a}{M} . \tag{2.25}$$

Je třeba však upozornit na to, že i pro hodnoty a, b, pro které jsou meze v (2.25) malé, může být některé ρ_k velké a takovýto generátor může znehodnotit prováděné simulace. Ani exaktní výsledky ohledně korelací nemusí dát praktický návod pro nalezení optimálních hodnot a a b. V praktických simulacích je tedy třeba užít buď vyzkoušeného generátoru nebo podrobit vlastní generátor sérii statistických testů, o kterých bude pojednáno později.

S problémy seriální korelace částečně souvisí problém sdruženého rozdělení dvojic $(X_n,\ X_{n+k}), k \geq 1,\ k$ pevné. Tyto dvojice by hypoteticky měly mít rozdělení

$$P(Y = i,\ Z = j) = M^{-2}, \quad i, j = 0,\ 1,\ \ldots,\ M-1\ . \tag{2.26}$$

Uvažujeme-li však smíšenou kongruenční metodu, nejsou čísla x a x_{n+k} nezávislá, nýbrž lze odvodit, že x_{n+k} je jednoznačně určeno x_n vztahem

$$x_{n+k} = a'_k x_n + b'_k \bmod M\ . \tag{2.27}$$

Proto existuje nejvýše M různých dvojic, které se mohou vyskytnout s hledanou pravděpodobností. Předpokládejme, že uvažovaný generátor má plnou periodu M. Potom generovaná posloupnost $x_{n\,n=0}^{M=1}$ je permutace čísel $0,\ 1,\ \ldots,\ M-1$. Generované dvojice jsou (bez ohledu na pořadí)

$$(n, a'_k n + b'_k \bmod M), \quad n = 0,\ 1,\ \ldots,\ M-1\ . \tag{2.28}$$

Využijeme-li věty za vztahem (2.1), můžeme generované dvojice také zapsat ve tvaru

$$\left(n, M\mathrm{Fr}\left(\frac{a'_k n + b'_k}{M}\right)\right), \quad n = 0,\ 1,\ \ldots,\ M-1\ . \tag{2.29}$$

Grafem funkce

$$f(x) = M\mathrm{Fr}\left(\frac{a'_k x + b' - k}{M}\right), \quad 0 \leq x < M \tag{2.30}$$

jsou paralelní úsečky, takže body (2.29) leží na těchto úsečkách. To samozřejmě neodpovídá představě o rovnoměrném rozdělení dvojic (x_n, x_{n+k}) na čtverci. Ve skutečnosti bývá situace daleko horší, protože body (2.29) bývají koncentrovány na daleko menším počtu úseček než odpovídá grafu funkce (2.30). Tato situace je pro generátor

$$x_{n+1} = 5x_n + 1 \bmod 16 \tag{2.31}$$

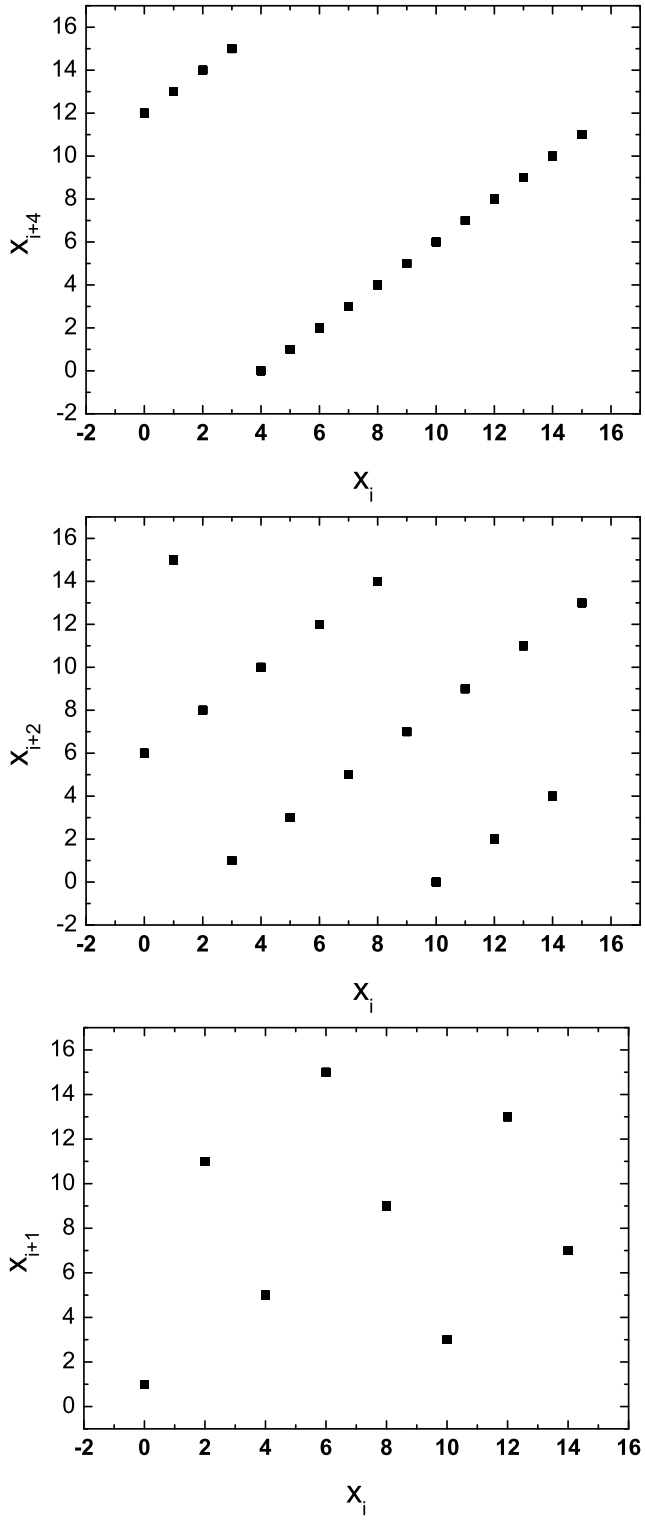
a pro $k = 1,\ 2,\ 4$ znázorněna na obr. 2.1. Pro tento generátor platí

$$x_{n+2} = 9x_n + 6 \bmod 16, \quad x_{n+4} = x_n + 12 \bmod 16\ . \tag{2.32}$$

Generované hodnoty jsou následující:

x_n	0	1	2	3	4	5	6	7	8	9	10
x_{n+1}	1	6	11	0	5	10	15	4	9	14	3
x_{n+2}	6	15	8	1	10	3	12	5	14	7	0
x_{n+4}	12	13	14	15	0	1	2	3	4	5	6

x_n	11	12	13	14	15
x_{n+1}	8	13	2	7	12
x_{n+2}	9	2	11	4	13
x_{n+4}	7	8	9	10	11



Obrázek 2.1: K problému sdruženého rozdělení dvojic (x_n, x_{n+k}).

Závěrem je však třeba upozornit na tu skutečnost, že pseudonáhodná čísla získaná aritmetickými generátory je třeba vždy interpretovat zepředu, pokud možno celá. Jednotlivé cifry, zvláště nižších řádů nemají vlastnosti náhodných číslic. Demonstrujme tuto skutečnost na následujícím příkladě.

Uvažujme multiplikativní generátor

$$x_{n+1} = 5x_n \bmod 2^5 \tag{2.33}$$

s počáteční hodnotou $x_0 = 1$. Pomocí tohoto generátoru dostaneme posloupnost (ve dvojkové soustavě)

$$\begin{array}{rcccccc} x_0 & = & 0 & 0 & 0 & 0 & 1 \\ x_1 & = & 0 & 0 & 1 & 0 & 1 \\ x_2 & = & 1 & 1 & 0 & 0 & 1 \\ x_3 & = & 1 & 1 & 1 & 0 & 1 \\ x_4 & = & 1 & 0 & 0 & 0 & 1 \\ x_5 & = & 1 & 0 & 1 & 0 & 1 \\ x_6 & = & 0 & 1 & 0 & 0 & 1 \\ x_7 & = & 0 & 1 & 1 & 0 & 1 \\ x_8 & = & 0 & 0 & 0 & 0 & 1 \end{array}$$

Posloupnost posledních cifer má tedy periodu 1, posloupnost čísel z posledních dvou cifer má také periodu 1, posloupnost čísel z posledních tří cifer má periodu 2, posloupnost čísel z posledních čtyř cifer má periodu 4 a konečně celá posloupnost má má periodu 8.

2.3 Některé konkrétní generátory pseudonáhodných čísel

V současné době téměř všechny počítače bývají vybaveny tzv. „generátorem náhodných čísel", který nejčastěji vytváří hodnoty náhodné veličiny X rovnoměrně rozdělené v intervalu $(0, 1)$. Většinou je možné pomocí vstupního parametru generátoru volit při každém opakování výpočtu jinou část posloupnosti nagenerovaných čísel.

Firma IBM vybavuje počítače řády IBM/360 generátorem

$$x_{n+1} = 65539x_n \bmod 2^{31} \ , \tag{2.34}$$

kde za x_0 je možné zvolit libovolné liché číslo. Pro binární počítače se často užívají generátory typu

$$x_{n+1} = 5^{2q+1}x_n \bmod 2^{\alpha} \ , \tag{2.35}$$

které mají maximální periodu. Podobně pro dekadické počítače se užívá

$$x_{n+1} = 7^{4q+1}x_n \bmod 10^{\alpha} \ . \tag{2.36}$$

Konkrétní takové navržené generátory jsou

$$x_{n+1} = 5^{17}x_n \bmod 2^{42} \ , \tag{2.37}$$

$$x_{n+2} = 5^{13}x_n \bmod 2^{39} \ . \tag{2.38}$$

První z nich (2.37) má periodu opakování posloupnosti pseudonáhodných čísel 2^{40}.

V případě, kdy užitý programovací jazyk resp. jeho překladač nedovoluje využít strukturu počítačového slova a je tedy jedno, zda se provádí operace mod M pro M mající souvislost s daným typem počítače či ne, je výhodné za M volit prvočíslo. Tak např. Miller a Prentice doporučují generátor

$$x_n = x_{n-2} + x_{n-3} \bmod 3137 \,. \tag{2.39}$$

Autoři uvádějí periodu 9 843 907. Jiný generátor tohoto typu uvádějí Downham a Roberts

$$x_{n+1} = 8192 x_n \bmod 67099547 \,. \tag{2.40}$$

Tento generátor má maximální periodu.

Na závěr uveďme jedno obecně užívané doporučení pro tuto volbu multiplikativní konstanty a. Doporučuje se volit a blízké k $M^{1/2}$. Velmi malá případně velká a nedávají dobré výsledky.

2.4 Kombinované generátory

Ukazuje se, že kongruenční generátory jsou vyhovující pro značnou část simulačních úloh. Jsou však speciální typy úloh, ve kterých se významně projeví některé jejich nedostatky. Například některé jednoduché funkce n-tic pseudonáhodných čísel nemají rozdělení, které by se očekávalo na základě teorie. Smíšené kongruenční metody jsou v tomto ohledu horší než metody ryze multiplikativní. Přesto ani multiplikativní generátor nedává při řešení podobných úloh dobré výsledky. Přitom dosud nejsou známy tak efektivní a případně v jiných směrech tak kvalitní generátory kongruenční. Ve snaze zachovat výpočetní jednoduchost kongruenčních generátorů a odstranit některé jejich nedostatky, byly navrženy metody, které k vytváření jedné posloupnosti pseudonáhodných čísel používají dvou nebo více kongruenčních generátorů. Základní myšlenkou kombinovaných generátorů je potřít nebo alespoň značně zredukovat jakoukoliv závislost členů nově vzniklé posloupnosti.

Metodu smíšených generátorů navrhli Maclaren a Marsagia. Mějme dva generátory, z nichž jeden produkuje pseudonáhodná čísla $y_0,\ y_1,\ \ldots$ z rovnoměrného rozdělení $R(0,1)$ a druhý pseudonáhodná čísla $r_0,\ r_1,\ \ldots$ z diskrétního rovnoměrného rozdělení $R(1,\ldots,n)$. Prvních n členů $y_0,\ y_1,\ \ldots,\ y_{n-1}$ se uloží do paměti na adresy $0,\ 1,\ \ldots,\ n-1$. Nová posloupnost pseudonáhodných čísel x_i z rozdělení $R(0,1)$ se vytváří následovně: za x_0 se vezme y_{r_0} a y_{r_0} se v paměti nahradí číslem y_n; za x_i se vezme y_{r_i} a y_{r_i} se v paměti nahradí číslem y_{n+1}, atd. Náhodné číslo r_i tedy udává místo v paměti, ze kterého se vezme náhodné číslo x_i; na toto místo se zároveň dosadí následující hodnota y_{n+1}. Praktické testování tohoto generátoru ukázalo, že vyhoví i testům, kterým jednoduché generátory nevyhoví. Pouze v případě, kdy jsou oba generátory velmi špatné, selže i kombinovaný generátor. Nevýhodou této metody je jednak větší nárok na paměť počítače, jednak potřeba generovat dvě pseudonáhodná čísla k získání jednoho nového.

Jinou metodou generování pseudonáhodných čísel s využitím dvou kongruenčních generátorů navrhnul Neave. Tato metoda vychází z následujícího lemmatu.

Lemma 1. Nechť nezávislé náhodné veličiny Y, Z mají rovnoměrné rozdělení $R(0,1)$. Potom náhodná veličina $X = Y + Z \bmod 1$ má rovnoměrné rozdělení $R(0,1)$.

Uvažujme dva kongruenční generátory se stejným modulem M produkující posloupnosti $\{y_i\}$ a $\{y'_i\}$. Nová posloupnost je definována vztahem

$$x_i = \frac{1}{M}(y_i + y'_i) \bmod 1 \,. \tag{2.41}$$

Tuto metodu můžeme zobecnit i na případ, že posloupnosti $\{y_i\}$ a $\{y'_i\}$ jsou vytvářeny generátory s moduly M resp. M'. Novou posloupnost pak definujeme vztahem

$$X_i = \frac{y_i}{M} + \frac{y'_i}{M'} \bmod 1 \,. \tag{2.42}$$

Ve srovnání s metodou smíšených generátorů je tato metoda méně náročná na paměť počítače.

2.5 Lineární rekurentní generátor mod 2

V roce 1965 Tausworthe [13, 15] navrhnul tzv. „shift-register sequence random number generator", který generuje posloupnost náhodných binárních čísel. Nechť $a = \{a_n\}$ je posloupnost 0 a 1 generovaná podle rekurentního vztahu

$$a_k = \sum_{i=1}^{m-1} c_i a_{k-i} + a_{k-m} \bmod 2 \,, \tag{2.43}$$

přičemž konstanty c_i ($i = 1,\ 2,\ \ldots,\ m-1$) mohou nabývat jen hodnoty 0 nebo 1.

Tento algoritmus se dá snadno rozšířit i na případ, kdy potřebujeme generovat posloupnost n-bitových celočíselných náhodných čísel. Tato čísla můžeme vlastně považovat za posloupnost n jednobitových náhodných čísel. Dostáváme tedy pro n-bitová náhodná čísla rekurentní vztah

$$x_k = c_1 x_{k-1} \oplus \ldots \oplus c_{p-1} x_{k-p+1} \oplus x_{k-p} \,, \tag{2.44}$$

kde $\oplus$ označuje sčítání mod 2 bit po bitu. Platí tedy $0 + 0 = 1 + 1 = 0$ a $0 + 1 = 1 + 0 = 1$ (exclusive or). Nejjednodušší případ tohoto algoritmu je následující

$$x_k = x_{k-q} \oplus x_{k-p} \,. \tag{2.45}$$

Je jasné, že vlastnosti tohoto generátoru závisí na výběru konstant q a p. Vhodná volba je např. $q = 103$, $p = 250$.

Pro libovolné počáteční podmínky $\{x_1,\ \ldots,\ x_m\}$, přičemž alespoň jedna z těchto hodnot není rovna nule, a při splnění podmínky, že charakteristický polynom $1 \oplus c_1 x \oplus c_2 x^2 \oplus \ldots \oplus c_m x^m$, je primitivní polynom mod 2, dostaneme pro tento generátor maximální možnou periodu $N = (2^m - 1)$.

Poznámka:
Pro primitivní polynom mod2 musí platit:

1. nesmí být polynomem typu $1 \oplus x$, kde $k < N = 2^m - 1$,
2. nesmí být rozložitelný na faktory.

Primitivní polynomy, zajišťující posloupnost maximální délky jsou uvedeny v tabulce 2.1.

Podle uvedené tabulky se např. pro řád $n = 5$ dosáhne maximální periody, když $c_1 = c_2 = c_4 = 0$ a $c_3 = c_5 = 1$. Nevýhodou tohoto generátoru je že je třeba zadat počáteční podmínky $\{x_1,\ \ldots,\ x_m\}$. Tyto hodnoty lze vytvořit pomocí jiného typu generátoru.

řád	číslo	2	1	1	1	1	1	1	1	1	1	1									
n	koef.	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
3																			1	0	1	1
4																		1	0	0	1	1
5																	1	0	0	1	0	1
6																1	0	0	0	0	1	1
7															1	0	0	0	0	0	1	1
8														1	0	0	0	1	1	1	0	1
9													1	0	0	0	0	1	0	0	0	1
10												1	0	0	0	0	0	0	1	0	0	1
11											1	0	0	0	0	0	0	0	0	1	0	1
12										1	0	0	0	0	0	1	0	1	0	0	1	1
13									1	0	0	0	0	0	0	0	0	1	1	0	1	1
14								1	0	0	0	1	0	0	0	1	0	0	0	0	1	1
15							1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
16						1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	1	1
17					1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
18				1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1
19			1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1
20		1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1

Tabulka 2.1: Koeficienty primitivních polynomů.

2.6 Kvazináhodná čísla

V algoritmech, kde pracujeme s náhodnými (i pseudonáhodnými) čísly, má konvergence pravděpodobnostní charakter. Existují však i složité algoritmy [5], pomocí nichž se dají vytvořit posloupnosti tzv. kvazináhodných čísel, majících řadu výhod: Konvergence je vždy zajištěna, nejsou nutné statistické testy a konvergence je vždy téměř rychlosti $1/n$. Tato čísla se tedy dají s výhodou využít v algoritmech metody Monte Carlo. Okruh problémů, pro které jsou kvazináhodná čísla vhodná, je ale omezen. Další nevýhodou, kromě složitého algoritmu výpočtu, je to, že tato čísla se mohou používat pouze v zadaném pořadí.

Příkladem takové posloupnosti kvazináhodných čísel rozdělených rovnoměrně v intervalu $(0, 1)$ je $1/2, 1/4, 3/4, 1/8, 5/8, 3/8, 7/8, 1/16, \ldots$ nebo $1/3, 2/3, 1/9, 4/9, 7/9, 2/9, 5/9, 8/9, \ldots$.

Závěrem této kapitoly o různých typech generátorů pseudonáhodných čísel je vhodné ještě jednou zdůraznit, že není podstatné, jakým způsobem jsou tato čísla získávána, ale podstatné je, zda splňují požadavky náhodnosti, která jsou od nich očekávána. Z tohoto hlediska je třeba věnovat velkou pozornost statistickým testům. Při praktických výpočtech je vhodné nepoužívat pouze jediný typ generátoru, ale několik a výsledky simulace porovnat.

Kapitola 3

Testování generátorů náhodných čísel

Před prováděním simulací většího rozsahu je nutné přezkoušet vlastnosti použitého generátoru. K tomuto účelu slouží celá řada testů, které ověřují, zda poskytované posloupnosti náhodných čísel splňují kritéria náhodnosti.

3.1 Frekvenční test - χ^2 test dobré shody

Test je používán k ověření rovnoměrného rozdělení generovaných čísel. Pravděpodobnost, že náhodná veličina, mající rovnoměrné rozdělení v intervalu (0,1), nabude hodnoty z podintervalů (α, β), kde $0 \leq \alpha \leq \beta \leq 1$, je rovna rozdílu $\beta - \alpha$.

Rozdělíme-li tedy jednotkový interval na k úseků a generujeme n náhodných čísel, můžeme testovat nulovou hypotézu H_0 o shodě očekávaných a skutečných četností výskytu náhodných čísel v jednotlivých podintervalech. Vhodným testem je např. χ^2 - test dobré shody [9]. Zvolíme-li hladinu statistické významnosti, můžeme ověřit, zda nedojde k překročení kritické hodnoty rozdělení χ^2 (s k-1 stupni volnosti) u testovací veličiny

$$T = \sum_{j=i}^{k} \frac{(n_j - np_j)^2}{np_j} \ , \tag{3.1}$$

kde n_j označuje počet hodnot, které padly do j-tého intervalu a p_j je očekávaná pravděpodobnost výskytu čísel v j-tém intervalu.

Jak již bylo řečeno dříve, větší problém než rovnoměrnost rozdělení pseudonáhodných číslic představují jejich korelační vlastnosti. V mnoha konkrétních úlohách potřebujeme vytvářet rovnoměrně rozdělené body v s-rozměrné jednotkové krychli. Na testování této vlastnosti můžeme použít „vícerozměrný" χ^2 - test dobré shody. S-rozměrnou krychli rozdělíme na $k = r^s$ stejných krychlí o objemu r^{-s}. Předpokládejme, že jsme vygenerovali n náhodných bodů $y_1^{(s)}, \ldots, y_n^{(s)}$ v s-rozměrné krychli a nechť n_i jich padlo do první krychle o objemu r^{-s}, n_2 do druhé krychle, atd. $(n_1 + \ldots + n_k = n)$. Testovací veličinou pro srovnání shody předpovědi a pozorování je

$$T = \frac{1}{nk^{-1}} \sum_{i=1}^{k} (n_i - nk^{-1})^2 \ . \tag{3.2}$$

Rozdělení veličiny T je asymptoticky χ^2 s k-1 stupni volnosti.

Tento vícerozměrný χ^2 - test patří k nejsilnějším testům a odhalí chyby generátorů náhodných čísel, které by mohly jednorozměrnému testu uniknout. Nevýhodou tohoto

testu je časová náročnost. Jestliže požadujeme, aby do každé krychle padlo okolo 20 bodů (nutné pro použití - χ^2 rozdělení), celkový počet generovaných náhodných čísel bude velmi vysoký.

Kritické hodnoty χ^2- rozdělení pro malý počet stupňů volnosti lze nalézt v [9]. Pro počet stupňů volnosti $n > 40$ lze s dostatečnou přesností použít aproximaci distribuční funkce χ^2- rozdělení

$$F_{\chi^2}(x,n) \doteq \Phi\left(\sqrt{4.5n}\ \left(\sqrt[3]{\frac{x}{n}} + \frac{2}{9n} - 1\right)\right)\ , \tag{3.3}$$

kde $\Phi(x)$ je distribuční funkce normálního rozdělení $N(0,1)$. Tento vztah i přesnější aproximace jsou uvedeny v [15].

Ukažme si testy některých uváděných generátorů. Jako nejlepší pro 16 bitová celá čísla se ukázal generátor dle [13] , generující náhodná čísla dle vztahu

$$x_n = x_{n-250} + x_{n-103} \bmod (2^{15} - 1)\ . \tag{3.4}$$

Náhodná čísla z $R(0,1)$ pak dostaneme dělením x_n číslem 2^{15} - 1. Výsledky χ^2 – testu pro 5000 náhodných čísel a pro dělení intervalu (0, 1) na 50 stejných podintervalů jsou uvedeny v tab. 3.1. Počátečních 250 náhodných čísel bylo generováno pomocí funkce random integer v Turbo - Pascalu.

i	1	2	3	4	5
n_i	91	108	106	110	107
i	6	7	8	9	10
n_i	93	119	100	111	102
i	11	12	13	14	15
n_i	115	90	95	109	95
i	16	17	18	19	20
n_i	91	105	105	100	80
i	21	22	23	24	25
n_i	117	97	98	94	98
i	26	27	28	29	30
n_i	100	103	95	101	101
i	31	32	33	34	35
n_i	94	92	95	96	99
i	36	37	38	39	40
n_i	74	117	96	109	91
i	41	42	43	44	45
n_i	96	81	115	107	106
i	46	47	48	49	50
n_i	100	86	106	108	96

Tabulka 3.1: Výsledky χ^2- testu pro generátor (3.4): $t = 45.140$, odpovídající hladina významnosti je 0.63, hypotéza je potvrzena.

Na základě tohoto testu můžeme posloupnost takto generovaných čísel považovat za posloupnost náhodných čísel z $R(0,1)$. Naopak pro generátor uváděný v [12], pracující dle vztahu

$$x_n = x_{n-2} + x_{n-3} \bmod 3137 \tag{3.5}$$

s počátečními hodnotami $x_1 = 1671$, $x_2 = 3033$, $x_3 = 1055$, dostáváme pomocí χ^2 – testu výsledky uvedené v tab. 3.2.

i	1	2	3	4	5
n_i	100	130	82	159	90
i	6	7	8	9	10
n_i	80	152	135	84	69
i	11	12	13	14	15
n_i	78	105	130	92	116
i	16	17	18	19	20
n_i	101	38	84	71	101
i	21	22	23	24	25
n_i	63	130	89	105	126
i	26	27	28	29	30
n_i	130	111	88	128	63
i	31	32	33	34	35
n_i	103	70	81	38	103
i	36	37	38	39	40
n_i	115	89	126	107	75
i	41	42	43	44	45
n_i	72	81	135	144	81
i	46	47	48	49	50
n_i	90	149	82	126	103

Tabulka 3.2: Výsledky χ^2- testu pro generátor (3.5): $t = 384.28$, hypotéza zamítnuta na hladině významnosti 0.001.

Hypotézu o rovnoměrném rozdělení takto generované posloupnosti zamítáme s rizikem 0.001. Jak později uvidíme v odstavci, věnovaném grafickému testu, má tento generátor velmi špatné vlastnosti.

3.2 Kolmogorovův - Smirnovův test dobré shody

Tento test, založený na sledování odchylek empirické $S_n(x)$ a hypotetické $F(x)$ distribuční funkce, je uveden v [9]. Testovací veličinou u tohoto testu je

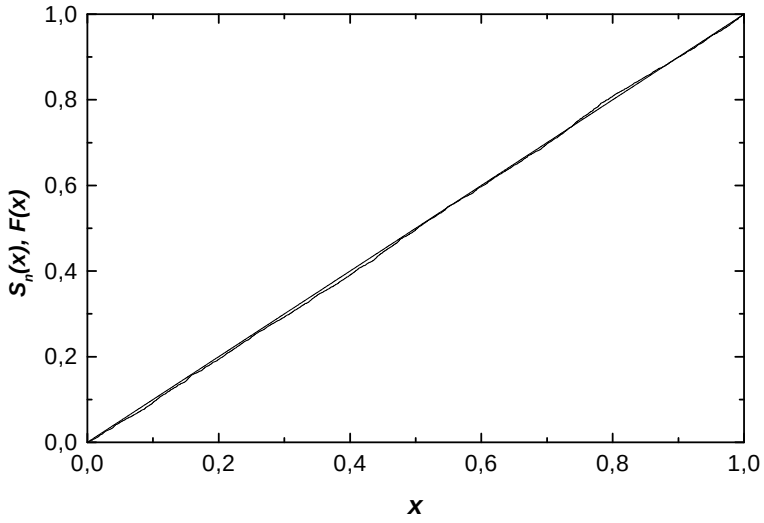
$$Z = \sqrt{n}\, \max|S_n(x) - F(x)|\,, \tag{3.6}$$

kde n je počet testovaných čísel. Jestliže Z bude větší než kritická hodnota Z_α, pak hypotézu o daném rozdělení zamítáme s rizikem α. Uveďme si několik kritických hodnot pro různé α : $Z_{0.01} = 1.63$; $Z_{0.05} = 1.36$ a $Z_{0.1} = 1.22$. Pro generátor (3.4) dostáváme výsledky uvedené v obr. 3.1. Z testu dostáváme $Z = 0.77$, hypotézu o rovnoměrném rozdělení přijímáme. Celkový počet testovaných čísel byl 5000.

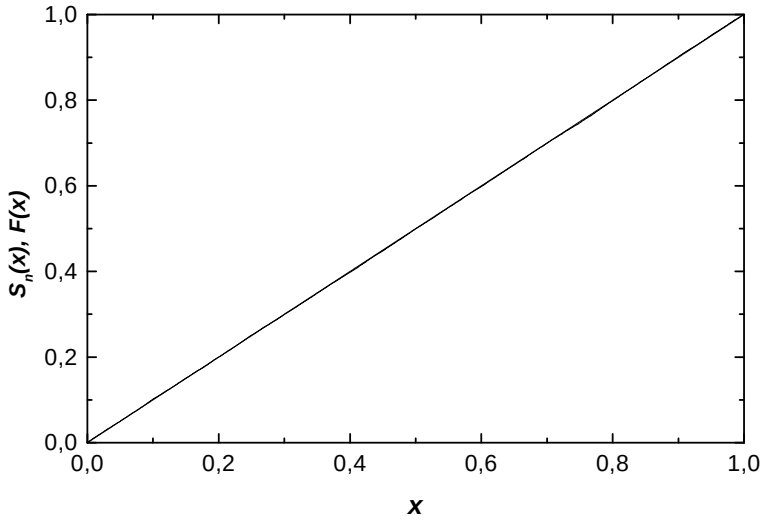
Zajímavá situace nastane při testování generátoru podle [15], který generuje náhodná čísla dle vztahu

$$x_n = 125x_{n-1} \bmod 8192 \tag{3.7}$$

s $x_1 = 1$. Empirická a teoretická distribuční funkce pro 5000 čísel na obr. 3.2 splývají v jednu čáru. Zdá se tedy, že shoda obou distribučních funkcí je výborná a hypotéza o rov-



Obrázek 3.1: Kolmogorovův - Smirnovův test generátoru (3.4).



Obrázek 3.2: Kolmogorovův - Smirnovovův test generátoru (3.7).

noměrném rozdělení je potvrzena. Z Kolmogorovova - Smirnovova testu však dostáváme $Z = 0.23$. Pravděpodobnost, že $Z < 0.25$ je však přibližně 3×10^{-8} . Shoda je tedy tak dobrá, že je to málo pravděpodobné a hypotézu o rovnoměrném rozdělení musíme zamítnout. Jak uvidíme později, generátor (3.7) generuje čísla nikoli náhodná, ale pravidelně rozložená na intervalu (0, 1) a díky tomu je shoda tak dobrá. Stejný výsledek dostaneme i při χ^2 – testu.

3.3 Test náhodnosti výskytu číslic

Generovaná náhodná čísla obsahují určitý počet dekadických míst, jenž je dán délkou slova použitého počítače. Posloupnost čísel můžeme převést na řadu číslic a testovat náhodnost výskytu jednotlivých číslic. Lze například stanovit pravděpodobnost, že mezi dvěmi stejnými číslicemi je k jiných číslic, výrazem

$$P_k = 0.9^k \times 0.1 \; . \tag{3.8}$$

Test spočívá v tom, že se zvolí určitá číslice a vypočtou se relativní četnosti výskytů „mezer" různých délek a testuje se, zda se shodují s teoreticky zjištěnými pravděpodobnostmi P_k. K tomu účelu můžeme použít např. Kolmogorovův - Smirnovův test.

3.4 Poker test

Název tohoto testu je odvozen od známé karetní hry. Testují se zde četnosti výskytu kombinací číslic, které jsou analogické různým figurám v této hře. Například pro pětimístná náhodná čísla můžeme testovat, zda empiricky zjištěné četnosti výskytu určitých kombinací se dostatečně shodují s teoretickými četnostmi spočtenými za předpokladu náhodnosti. Pro testování shody empirických a teoretických četností obvykle použijeme χ^2 - testu dobré shody.

Uvažovaná pětimístná čísla mohou být např. tvořena 5 prvními číslicemi náhodných čísel z $R(0, 1)$. Všimněme si pro jednoduchost pouze dekadických číslic. Označme a, b, c, d, e libovolné číslice 0, 1, ..., 9. Pravděpodobnosti výskytu kombinací (za předpokladu rovnoměrného rozdělení číslic) jsou v tab. 3.3. Vzhledem k velikostem teoretických prav-

Kombinace	Pravděpodobnost
abcde	0.3024
aabcd (dvojice)	0.5040
aabbc (2 dvojice)	0.1080
aaabc (trojice)	0.0720
aaabb (full)	0.0090
aaaab (poker)	0.0045
aaaaa (pětice)	0.0001

Tabulka 3.3: Pravděpodobnosti různých kombinací při poker testu.

děpodobností je však zapotřebí generovat posloupnost o minimálně 50000 členech, neboť χ^2 – test je test asymptotický. Při testování kratších posloupností se obvykle poslední dvě kombinace v tab. 3.1 slučují.

3.5 Test maxima

Je-li dána posloupnost nezávislých náhodných veličin s rovnoměrným rozdělením v jednotkovém intervalu X_1, X_2, ..., X_n, lze definovat náhodnou veličinu Z, vztahem

$$Z = \max(X_1, \; \ldots, \; X_n) \; . \tag{3.9}$$

Potom má veličina Z^n rovnoměrné rozdělení v jednotkovém intervalu. Test spočívá v tom, že generujeme posloupnost náhodných čísel $x_1, x_2, \ldots$, postupně počítáme hodnoty $z_1 = \max(x_1, \ldots, x_n)$, $z_2 = \max(x_{n+1}, \ldots, x_{2n})$, ... a hodnoty z_i^n, $z_2^n \ldots$ registrujeme. Pomocí frekvenčního testu zjišťujeme, zda veličina z^n je rovnoměrně rozdělena.

3.6 Test autokorelace

Další důležitou vlastností posloupnosti náhodných čísel je nezávislost mezi členy jdoucími po sobě. Uvažujme např. posloupnost 0.001, 0.501, 0.002, 0.502, 0.003, 0.503, ..., 0.499, 0.999. Aplikujeme-li na ni frekvenční test s dělením na deset stejných intervalů, bude vše zdánlivě v pořádku, neboť empirické četnosti se budou shodovat s teoretickými, ovšem tuto posloupnost nemůžeme považovat za dobrou posloupnost náhodných čísel. K odhalení závislosti mezi sousedními, popřípadě vzdálenějšími členy generované posloupnosti, slouží seriální korelační koeficienty. Pro $k < n$ jsou tyto koeficienty definovány vztahem

$$R_k = \frac{1/(n-k) \sum_{i=1}^{n-k} (X_i - \bar{X})(X_{i+k} - \bar{X})}{1/n \sum_{i=1}^{n} (X_i - \bar{X})^2} , \tag{3.10}$$

kde

$$\bar{X} = \frac{1}{n} \sum_{i=1}^{n} X_i . \tag{3.11}$$

Lze ukázat, že jsou-li $X_1, X_2, \ldots$ nezávislé stejně rozdělené náhodné veličiny s konečnou disperzí $D(X) = \sigma^2$, potom má náhodná veličina $\sqrt{n}\ R_k$ pro $n \to \infty$ a k pevné limitní normální rozdělení $N(0, 1)$. Pro testování shody rozdělení hodnot $\sqrt{n}\ R_k$ s rozdělením $N(0, 1)$ lze použít χ^2- test dobré shody. Lze odvodit, že za hypotézy náhodnosti je

$$E(R_k) \doteq -\frac{1}{n-k} \quad , \quad D(R_k) \doteq \frac{1}{n} . \tag{3.12}$$

Poznamenejme, že posloupnost $R_1, R_2, \ldots$ se nazývá *korelogram*. Obvykle se počítá několik prvních členů této posloupnosti a obecně se nedoporučuje počítat tyto koeficienty pro $k > n/4$.

Pro rovnoměrná náhodná čísla se někdy užívá zjednodušených statistik pro studium seriální korelace, a to

$$\widetilde{R_k} = \frac{1}{n-k} \sum_{i=1}^{n-k} \left(X_i - \frac{1}{2}\right)\left(X_{i+k} - \frac{1}{2}\right) . \tag{3.13}$$

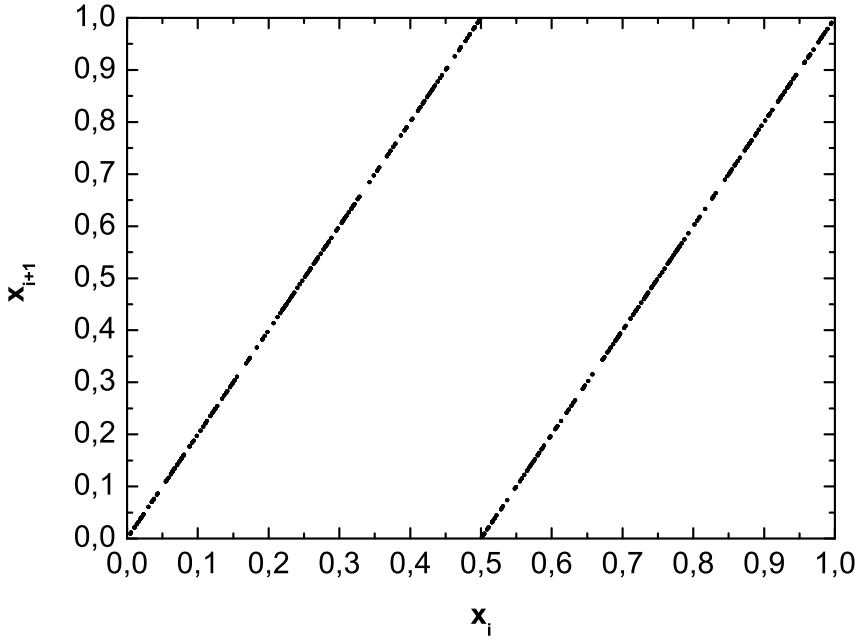
Pro $k > 0$ je za platnosti hypotézy náhodnosti

$$E(\widetilde{R_k}) = 0 \quad , \quad D(\widetilde{R_k}) = \frac{1}{144\ (n-k)} . \tag{3.14}$$

Řadu dalších účinných testů (test na součet m rovnoměrných náhodných čísel; test minima; Grunbergerův d^2- test; test na extrémní body; test podle znamének; iterace pod a nad mediánem; Spearmanův koeficient korelace pořadí) lze najít například v [12]. Z hlediska aplikací simulačních technik jsou značně nebezpečné korelace mezi po sobě jdoucími prvky posloupnosti náhodných čísel, popřípadě mezi prvky R_i a R_{i+k} pro $k > 0$, $i = 1, 2, \ldots$. Výsledky testů na autokorelaci však patří mezi nejméně publikované charakteristiky různých generátorů.

3.7 Grafický test

V poslední době, v souvislosti s rozvojem počítačové grafiky, se objevil nový jednoduchý vizuální test. Jeho podstata je následující. Na obrazovce monitoru postupně zobrazujeme body, jejichž souřadnice jsou určeny dvojicemi po sobě jdoucích čísel z generované posloupnosti. Vizuálně kontrolujeme, jak probíhá zaplňování obrazovky, zda probíhá rovnoměrně a náhodně a zda vzniklý obrazec nevykazuje nějaké rysy pravidelnosti.



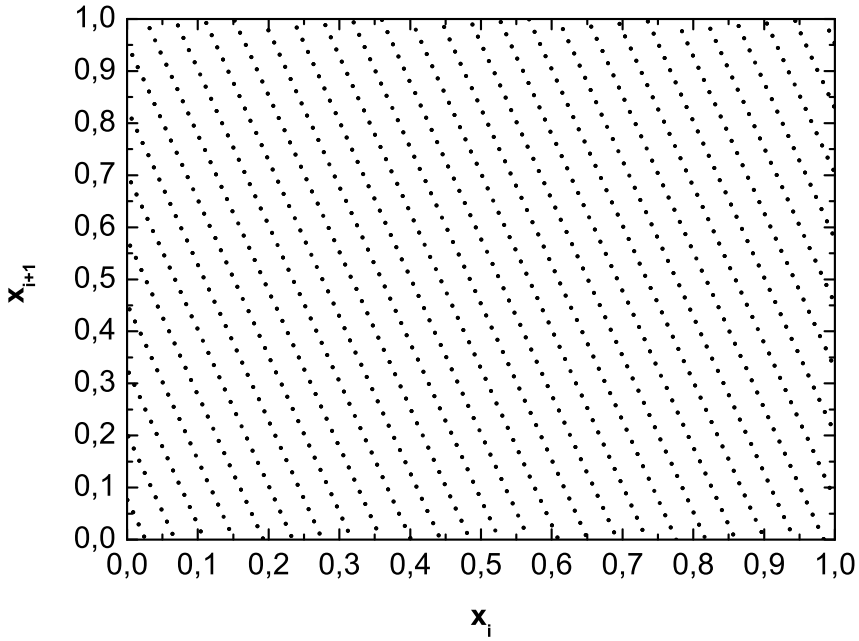
Obrázek 3.3: Grafický test generátoru (3.5).

Tento test je přes svoji zdánlivou jednoduchost velmi silný a umožňuje objevit i takové nenáhodnosti v dané posloupnosti náhodných čísel, které nelze objevit pomocí jednoduchého χ^2 - testu nebo Kolmogorovova - Smirnovova testu. Tento test bude tím efektivnější, čím vyšší bude rozlišovací schopnost monitoru, čím více různých barev pro jediný bod budeme moci využít - padne-li náhodný bod vícekrát do stejného bodu, změníme adekvátně barvu (resp. jas) tohoto bodu. Na základě rychlého vývoje počítačové grafiky se dá předpokládat, že zanedlouho budeme moci provádět grafické testy pro rozložení bodů v třírozměrném prostoru.

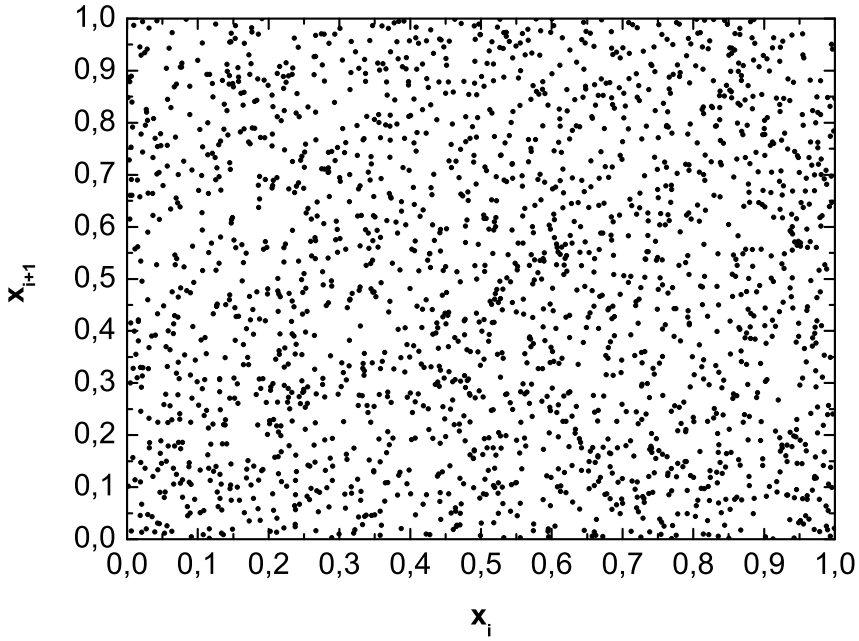
Vzhledem k velké síle a jednoduchosti doporučujeme tento test provádět jako základní test generátorů náhodných čísel. Pro generátor (3.5) dostáváme výsledek uvedený na obr. 3.3.

Je vidět, že tento generátor má velmi špatné vlastnosti. Místo aby byla obrazovka monitoru zaplněna náhodně po celé ploše, vzniknou pouze dvě čáry. Tuto vadu by spolehlivě také odhalil dvojrozměrný χ^2 – test. Pro generátor (3.7) dostaneme jiný výsledek, viz obr. 3.4. Kvalitní generátor, např. generátor (3.4), dává výsledky podobné obr. 3.5.

Abychom posoudili kvantitativně, zda takto generované body mají skutečně rovnoměrné rozdělení po ploše, testovali jsme tuto hypotézu pomocí dvojrozměrného χ^2- testu. Výsledek je uveden v tab. 4. Z testu dostáváme, že s pravděpodobností 0.77 mohou být odchylky od očekávaných 20 bodů v jednom políčku větší. Hypotézu o rovnoměrném rozdělení bodů přijímáme.



Obrázek 3.4: Grafický test generátoru (3.7).



Obrázek 3.5: Grafický test generátoru (3.4).

21	24	12	13	23	22	14	15	25	28	24	27	20	15	15	28	18	19	21	17
22	28	18	20	22	15	21	13	17	26	20	17	25	19	17	15	17	24	19	26
21	31	24	19	31	22	20	21	19	17	19	18	23	22	24	24	23	19	26	24
21	21	25	22	21	19	18	17	17	27	17	23	17	22	23	28	21	13	23	16
15	21	17	25	18	25	16	20	16	19	21	18	17	23	20	17	12	18	16	20
24	22	20	23	19	20	24	16	15	25	22	26	15	19	22	20	20	31	18	19
21	15	15	18	17	16	15	26	14	17	18	18	18	16	28	22	16	20	18	21
16	15	21	19	20	24	25	22	21	15	20	32	25	19	16	20	20	24	16	20
19	23	26	27	27	18	25	20	16	13	20	17	15	25	19	20	18	25	13	25
23	16	23	23	18	20	14	22	14	19	24	19	19	21	20	20	23	18	23	20
19	26	18	22	20	18	19	21	16	17	17	15	18	14	17	19	25	18	16	13
16	11	12	27	25	38	8	17	14	18	17	18	21	31	24	17	16	20	22	18
18	19	23	21	16	12	26	24	17	22	19	26	18	25	20	21	20	12	15	20
20	22	24	18	14	16	23	19	19	23	28	20	28	23	8	21	18	27	12	11
16	13	13	12	23	20	15	16	22	20	25	24	19	29	34	17	11	27	23	19
26	25	25	23	13	17	16	13	21	20	18	21	18	25	14	19	22	20	23	17
18	20	25	21	21	22	20	13	18	13	15	12	23	17	23	19	16	16	24	22
17	26	22	16	17	26	27	23	21	17	19	16	19	19	23	21	15	22	22	18
26	28	23	26	14	22	25	20	15	19	25	24	16	20	28	17	22	21	16	20
21	13	26	19	23	20	22	17	13	15	17	21	25	21	23	21	28	19	22	20

Tabulka 3.4: Výsledek dvojrozměrného χ^2 - testu pro body z obr. 3.5. Dělení 20×20, počet bodů 8000, generátor rnd3, čísla označují počet bodů v příslušném políčku. Statistika $t = 378.00$, odpovídající hladina významnosti 0.77, hypotéza je potvrzena.

Kapitola 4

Generování náhodných čísel s daným rozdělením pravděpodobnosti

Doposud jsme věnovali pozornost pouze generaci náhodných čísel z rovnoměrného rozdělení $R(0,1)$ resp. $R(0,M)$. Máme tedy k dispozici náhodnou veličinu Y s rozdělením $R(0,1)$ se střední hodnotou

$$E(Y) = 1/2 \tag{4.1}$$

a s disperzí

$$D(Y) = 1/12\ . \tag{4.2}$$

Na počítači však můžeme realizovat pouze k-místná diskrétní čísla x_{poc} a tudíž skutečné parametry počítačové realizace budou

$$E(Y_{\mathrm{poc}}) = (1 - 2^{-k})/2 \tag{4.3}$$

$$D(Y_{\mathrm{poc}}) = (1 - 4^{-k})/12\ . \tag{4.4}$$

Pokud budeme mít počítač s krátkým zobrazením čísla, je lépe pracovat s veličinou

$$\frac{2^k}{2^k + 1} x_{\mathrm{poc}}\ . \tag{4.5}$$

Abychom mohli simulovat různé náhodné procesy, musíme mít k dispozici i náhodné veličiny s jiným typem rozdělení než pouze $R(0,1)$. Potřebujeme tedy generovat obecně náhodnou veličinu na daném intervalu (a,b) se zadanou hustotou pravděpodobnosti $f_X(x)$ či distribuční funkcí $F_X(x)$.

Ukazuje se, že k tomu, abychom získali náhodná čísla s daným rozdělením pravděpodobnosti, stačí mít k dispozici generátor náhodných čísel $R(0,1)$ a tato čísla vhodně transformovat. Tomuto procesu se též říká rozehrát náhodnou veličinu X.

Pro transformaci náhodných čísel $R(0,1)$ na hodnoty náhodných veličin byly vyvinuty čtyři obecné metody:

- rozehrávání diskrétní náhodné veličiny
- metoda inverzní funkce (transformace)
- metoda výběru (metoda Neumannova)
- metoda superpozice.

4.1 Rozehrávání diskrétní náhodné veličiny

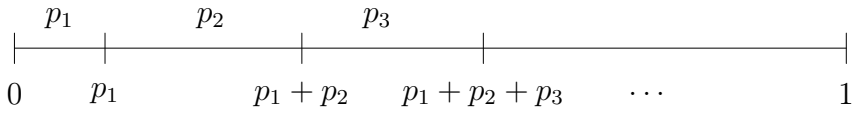
Mějme diskrétní náhodnou veličinu X zadánu pomocí tabulky

$$X = \begin{pmatrix} x_1 & x_2 & \dots & x_n \\ p_1 & p_2 & \dots & p_n \end{pmatrix}, \tag{4.6}$$

tj. $P(X = x_i) = p_i$. V počítači vytvoříme vektor o n složkách: p_1, $p_1 + p_2$, $p_1 + p_2 + p_3$, $\dots$, 1 (obr. 4.1). Nyní vygenerujeme jedno náhodné číslo $y \in R(0, 1)$ a určeme, do kterého intervalu padne, tj. budeme vyšetřovat podmínku

$$y < \sum_{i=1}^{j} p_i \, . \tag{4.7}$$

První interval j, pro který bude tato podmínka splněna, určí příslušnou hodnotu $X = x_j$. Tímto způsobem tedy realizujeme náhodnou veličinu X zadanou pomocí tabulky (4.6).



Obrázek 4.1: Generování hodnot diskrétní náhodné veličiny

4.2 Metoda inverzní funkce

Tato metoda je založena na následující větě.

Věta 10 *Nechť $F(x)$ je distribuční funkce, nechť $F^{-1}(y) = sup\{x \in R, F(x) \leq y\}$, $0 < y < 1$, je odpovídající kvantilová funkce a nechť náhodná veličina Y má rovnoměrné rozdělení $R(0, 1)$. Potom náhodná veličina $X = F^{-1}(Y)$ má rozdělení s distribuční funkcí $F(x)$.*

Důkaz. Nejprve ukážeme, že $F(x) \leq y$ právě když $x \leq F^{-1}(y)$. Je-li $F(x) \leq y$, pak zřejmě $x \leq F^{-1}(y)$. Na druhou stranu, je-li $x \leq F^{-1}(y)$, pak pro každé $\varepsilon > 0$ existuje z takové, že $x - \varepsilon < z$, přičemž $F(z) \leq y$. Distribuční funkce je neklesající, takže $F(x - \varepsilon) \leq y$ pro každé $\varepsilon > 0$, a je zleva spojitá, takže $F(x) \leq y$.

Tuto větu použijeme pro generování náhodných čísel s distribuční funkcí $F(x)$, v případě, že můžeme $F^{-1}(y)$ snadno vypočítat. Jestliže $F(x)$ bude rostoucí, pak $F^{-1}(x)$ bude rovna inverzní funkci k funkci $F(x)$. Pro diskrétní náhodnou veličinu dostaneme metodu uvedenou v předchozím odstavci.

Při použití metody inverzní funkce generujeme náhodná čísla $y_0, y_1, \dots$ z rozdělení $R(0, 1)$ a transformovaná čísla $x_0 = F^{-1}(y_0)$, $x_1 = F^{-1}(y_1)$, $\dots$ potom tvoří posloupnost náhodných čísel z rozdělení s distribuční funkcí $F(x)$.

Uvažujme např. náhodnou veličinu s exponenciálním rozdělením, tj. s distribuční funkcí

$$F(x) = \begin{cases} 1 - e^{-\lambda x} & x \geq 0 \\ 0 & x < 0 \end{cases} . \tag{4.8}$$

Potom

$$F^{-1}(y) = -\frac{1}{\lambda} \ln(1-y), \quad 0 < y < 1 . \tag{4.9}$$

Má-li náhodná veličina Y rovnoměrné rozdělení $R(0,1)$, potom má náhodná veličina

$$X = -\frac{1}{\lambda} \ln(1-Y) \tag{4.10}$$

rozdělení exponenciální s distribuční funkcí (4.8). Jelikož $Y \in R(0,1)$ má rovněž náhodná veličina $1-Y$ rozdělení z $R(0,\,1)$, a proto stačí generovat posloupnost

$$x_i = -\frac{1}{\lambda} \ln y_i . \tag{4.11}$$

Generování čísel podle tohoto vztahu (4.11), i přes jeho relativní jednoduchost, bude poměrně pomalé, neboť kromě podprogramů pro získávání kvazirovnoměrných pseudonáhodných čísel je třeba volat i podprogram (standardní funkci) pro výpočet přirozeného logaritmu. Existuje proto řada více či méně efektivních algoritmů, které obsahují např. logaritmickou transformaci.

Pokud nelze jednoduše analyticky vyjádřit kvantilovou funkci $F^{-1}(y)$, jako např. v uvedeném případě, musíme pro jednotlivé hodnoty y_i vyřešit vzhledem k x_i jednoznačně rovnici

$$\int_{\infty}^{x_i} f(x)\,\mathrm{d}x = y_i , \tag{4.12}$$

abychom získali hodnoty spojité náhodné veličiny X charakterizovanou hustotou pravděpodobnosti $f(x)$.

Metoda generování libovolných rozdělení metodou inverzní funkce je sice univerzální, ale není vždy výhodná a dokonce někdy může v důsledku zaokrouhlovacích chyb počítače i selhat - dostaneme náhodná čísla s jiným než zadaným rozdělením; zvláště se to týká chování „chvostů" rozdělovacích funkcí.

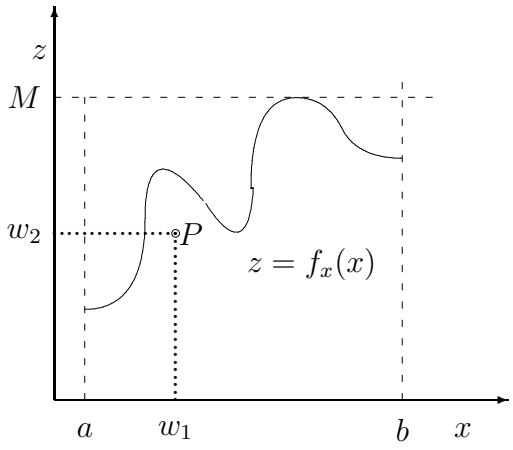
4.3 Metoda výběru

Tato metoda pochází od J. von Neumanna a lze ji s výhodou použít v případech, kdy nelze analyticky vyjádřit inverzní funkci $F^{-1}\,(y)$ v přecházející metodě.

Nechť hustota pravděpodobnosti $f_X(x)$ náhodné veličiny X je ohraničená na intervalu (a,b). Označme $M = \sup\{f_X(x), a < x < b\}$. Algoritmus metody výběru je následující:

1. Nagenerujeme dvě hodnoty x_1 a x_2 náhodné veličiny X z $R(0,1)$.
2. Vytvoříme čísla $w_1 = a + x_1(b-a)$, tzn. w_1 z $R(a,b)$ $w_2 = Mx_2$, tzn. w_2 z $R(0,M)$.
3. Bude-li bod P o souřadnicích (w_1, w_2) ležet pod křivkou $z = f_X(x)$ (obr. 4.2), tj. bude-li $w_2 \leq f_X(w_1)$, pak za náhodné číslo vezmeme $x = w_1$. Nebude-li podmínka splněna, nagenerujeme novou dvojici x_1 a x_2, tj. přejdeme k bodu 1.

Nevýhodou zamítací metody je, že ne všechny dvojice generovaných rovnoměrných čísel jsou využity pro generování náhodného čísla x.



Obrázek 4.2: Generování náhodných čísel metodou výběru.

4.4 Metoda superpozice

Máme za úkol generovat náhodnou veličinu X s distribuční funkcí $F(x)$. Podstatou této metody je rozložení distribuční funkce $F(x)$ do tvaru

$$F(x) = \sum_{i=1}^{m} p_i F_i(x) \ , \tag{4.13}$$

kde p_i jsou pravděpodobnosti, tj. $\sum_1^m p_i = 1$ a $F_i(x)$ distribuční funkce. Zavedeme diskrétní náhodnou veličinu W s rozdělením

$$W = \begin{pmatrix} 1 & 2 & \dots & m \\ p_1 & p_2 & \dots & p_m \end{pmatrix} \ . \tag{4.14}$$

Nagenerujeme-li dvě nezávislé hodnoty y_1 a y_2 veličiny Y, rozehrajeme-li číslem y_1 hodnotu $W = k$ (viz. rozehrání diskrétní náhodné veličiny) a pak z rovnice $F_k(x) = y_2$ určíme x, potom platí, že distribuční funkce veličiny X je rovna $F(x)$.

Přirozeným požadavkem však je, aby generování čísel s rozdělením $F_i(x)$ bylo jednoduché. Je-li možné provést rozklad $F(x)$ několika způsoby, volí se takový, pro který

$$\sum_{i=1}^{m} p_i T_i \to \min \ , \tag{4.15}$$

kde T_i jsou střední doby potřebné pro generování hodnot s rozdělením $F_i(x)$.

4.5 Modelování n-rozměrného náhodného bodu

Jestliže jednotlivé souřadnice náhodného bodu budou nezávislé, lze každou z nich modelovat zvlášť pomocí dříve uvedených metod. Pokud máme generovat body z oblasti složitého průběhu, je vhodné zobecnit metodu von Neumannovu. Nejdříve zvolíme jednoduchou oblast, která v sobě obsahuje zadanou oblast. Potom generujeme náhodné body v této nové větší oblasti a testujeme, padnou-li do naší původní oblasti. Pouze takové body zachováme a ostatní vyloučíme.

4.6 Transformace typu $X = f(Y_1, \ldots, Y_n)$

V současné době existují i jiné typy transformačních metod, než které byly dosud uvedeny. Tyto algoritmy konstruují náhodnou veličinu X pomocí několika hodnot veličiny $Y (n \geq 2)$. Důležité případy si ukážeme při řešení konkrétních příkladů.

4.7 Efektivnost generátoru

Při rozsáhlých simulačních výpočtech limitujícím faktorem se stává rychlost generování náhodných čísel. Pokud pomineme otázku kvality získávaných náhodných čísel, jediným kritériem vhodnosti generátoru bude doba potřebná ke generování jednoho náhodného čísla. U většiny generátorů je tato doba náhodná veličina. Z tohoto důvodu budeme uvažovat o střední době t potřebné ke generování jednoho náhodného čísla. *Efektivností generátoru* rozumíme veličinu

$$U = 1/t \; . \tag{4.16}$$

Pro porovnání efektivnosti dvou generátorů A, B používáme *relativní efektivnost*

$$U_A/U_B \; , \tag{4.17}$$

kde U_A resp. U_B je efektivnost generátoru A resp. B. Střední dobu t lze přibližně, pro naše účely však zcela vyhovujícím způsobem, vyjádřit ve tvaru

$$t = t_P t_G \; , \tag{4.18}$$

kde t_P je určena typem použitého počítače a nezávislá na generátoru a t_G je závislá pouze na použitém generátoru. Veličina t_G je např. dána celkovým počtem strojových operací nebo počtem rovnoměrných náhodných čísel potřebných ke generování požadovaného rozdělení, počtem v algoritmu použitých elementárních funkcí apod. Přibližnost spočívá v tom, že různé počítače mohou mít různou logiku hardware i různé algoritmy pro výpočet hodnot elementárních funkcí. Pro praktické účely však tato faktorizace vyhovuje a umožňuje porovnávat různé generátory bez ohledu na typ použitého počítače.

Co se týče metody výběru, bude efektivnost úměrná poměru S středního počtu nezamítnutých dvojic (w_1, w_2) k počtu generovaných dvojic. Pro S tedy můžeme psát

$$S = P(w_2 < f_X(w_1)) = \frac{1}{M(b-a)} \; . \tag{4.19}$$

Uvažujeme např. náhodnou veličinu X s hustotou pravděpodobnosti

$$f_X(x) = \begin{cases} \frac{4}{\pi(1+x^2)} & 0 \leq x \leq 1 \\ 0 & \text{jinak} \end{cases} \; . \tag{4.20}$$

Odpovídající distribuční funkce tedy je

$$F(x) = \begin{cases} 0 & x < 0 \\ -\frac{4}{\pi}\text{arctg } x & 0 \leq x \leq 1 \\ 1 & x > 1 \end{cases} \tag{4.21}$$

a pro inverzní funkci dostáváme

$$F^{-1}(y) = \mathrm{tg}\left(\frac{\pi}{4}y\right) , \quad 0 < y < 1 . \tag{4.22}$$

Náhodnou veličinu X s rozdělením (4.20) můžeme tedy generovat pomocí rovnoměrných náhodných čísel y_0, y_1, ... jako $F^{-1}(y_0)$, $F^{-1}(y_1)$, ... Jiná možnost je použití metody výběru. V tomto případě $M = 4/\pi$, takže pro efektivnost této metody dostáváme $S = \pi/4 \doteq 0.79$. Na generování jednoho náhodného čísla x se tedy v průměru spotřebuje $2/S \doteq 2.5$ rovnoměrných náhodných čísel. Výpočet elementární funkce tedy bude v převážné většině potřebovat mnohem více strojových instrukcí než generování 2 a 3 rovnoměrných náhodných čísel, takže zamítací metoda je zde efektivnější.

Nyní si uvedeme důležité případy transformace náhodných veličin.

4.8 Generování náhodných čísel speciálních spojitých rozdělení

4.8.1 Rovnoměrné rozdělení

Máme rozehrát veličinu X rovnoměrně rozdělenou na intervalu (a, b). Pro její hustotu pravděpodobnosti musí tedy platit

$$f_X(x) = \begin{cases} \frac{1}{b-a} & a < x < b \\ 0 & \text{jinak} . \end{cases} \tag{4.23}$$

Distribuční funkce této rovnoměrně rozdělené veličiny X tedy je

$$F(x) = \begin{cases} 0 & x < a \\ \frac{x-a}{b-a} & a \le x \le b \\ 1 & x > b . \end{cases} \tag{4.24}$$

Střední hodnota a rozptyl jsou

$$E(X) = \frac{a+b}{2} , \tag{4.25}$$

$$D(X) = \frac{(b-a)^2}{12} .$$

Pro rozehrání X použijeme metodu inverzní funkce. Ze vztahu (4.24) dostáváme jednoduchý vztah

$$X = a + Y(b-a) , \tag{4.26}$$

kde Y je z $R(0, 1)$. Rovnoměrné rozdělení je určené hodnotami dvou parametrů a, b nebo přímo hodnotami $E(x)$ a $D(x)$, protože řešením soustavy rovnic (4.25) dostáváme

$$a = E(X) - (3D(X))^{1/2} , \tag{4.27}$$

$$b = E(X) + (3D(X))^{1/2} .$$

4.8.2 Normální rozdělení

Toto rozdělení hraje nejen důležitou roli v teorii pravděpodobnosti, ale i v praktickém používání statistických metod.

Naším úkolem je tedy generovat náhodnou veličinu X s hustotou pravděpodobnosti

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right] , \tag{4.28}$$

přičemž pro normální rozdělení $N(0,1)$ je střední hodnota a disperze rovna

$$E(X) = \mu = 0 , \tag{4.29}$$

$$D(X) = \sigma^2 = 1 .$$

Vzhledem k poměrně zdlouhavému výpočtu kvantilové funkce normálního rozdělení se v praxi neužívá metoda inverzní funkce. Nejjednodušší je využití centrální limitní věty aplikované na součty nezávislých rovnoměrně rozdělených náhodných veličin.

Nechť $Y_1, \ldots, Y_n$ jsou náhodné veličiny z $R(0,1)$. Potom pro střední hodnotu a disperzi jejich součtu platí

$$E\left(\sum_{i=1}^{n} Y_i\right) = \frac{1}{2}n , \tag{4.30}$$

$$D\left(\sum_{i=1}^{n} Y_i\right) = \frac{1}{12}n .$$

Pro $n \to \infty$ má veličina

$$X_n = \sqrt{\frac{12}{n}}\left(\sum_{i=1}^{n} X_i - \frac{1}{2}n\right) \tag{4.31}$$

asymptoticky normální rozdělení $N(0,1)$. Za prakticky vyhovující se považuje $n = 12$, takže

$$X_{12} = \sum_{i=1}^{12} X_i - 6 . \tag{4.32}$$

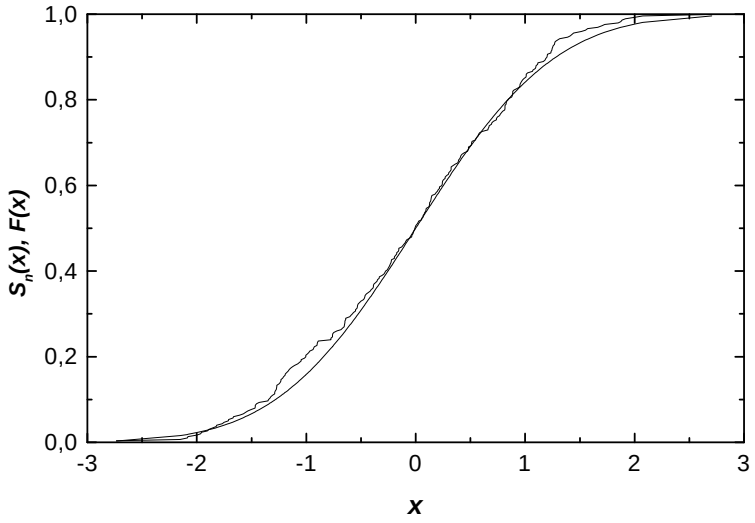
Na obr. 4.3. můžeme porovnat empirickou distribuční funkci takto generovaných čísel s distribuční funkcí normálního rozdělení.

Z Kolmogorovova-Smirnovova testu dostáváme $z = 0.846$, hypotézu o normálním rozdělení takto generovaných čísel přijímáme. Uvedený způsob generování je obvykle postačující v simulačních úlohách, ve kterých není podstatné chování „chvostů" rozdělení. Musíme si totiž uvědomit, že se jedná pouze o aproximaci. Např. skutečná náhodná veličina nabývá i velké odchylky od střední hodnoty sice s malou pravděpodobností, ale na základě vztahů (4.31, 4.32) je získat nemůžeme vůbec. V případě (4.32) je rozdělení X_{12} oboustranně useknuté v bodech -6 a $+6$, tj. $P(|X_{12} \leq 6) = 1$.

Byly proto vyvinuty postupy, které výsledky zpřesňují, ovšem za cenu snížení rychlosti výpočtu. Např. Bolšev navrhl transformaci

$$Z_n = X_n + \frac{1}{20n}[X_n^3 - 3X_n] . \tag{4.33}$$

Rozdělení Z_n je dostatečně blízké normálnímu již pro $n = 5$.



Obrázek 4.3: Empirická a teoretická distribuční funkce pro čísla generované podle vztahu (4.32).

Jednoduchý generátor normálních náhodných čísel navrhli Box a Müller. Nechť Y_1, Y_2 jsou nezávislé náhodné veličiny s rozdělením $R(0,1)$. Potom

$$\begin{aligned} X_1 &= \sqrt{-2\ln Y_1}\sin(2\pi Y_2) \\ X_2 &= \sqrt{-2\ln Y_1}\cos(2\pi Y_2) \end{aligned} \tag{4.34}$$

jsou nezávislé náhodné veličiny s rozdělením $N(0,1)$.

Jako přímočaré využití uvedené transformace pro generování normálních náhodných čísel se nabízí generování posloupnosti rovnoměrných náhodných čísel $y_0, y_1, \ldots$ a výpočet hodnot x_1, x_2, ze vztahů

$$\begin{aligned} x_{2i} &= \sqrt{-2\ln y_{2i}}\sin(2\pi y_{2i+1})\ , \\ x_{2i+1} &= \sqrt{-2\ln y_{2i}}\cos(2\pi y_{2i+1})\ , \quad i = 0, 1, \ldots\ . \end{aligned} \tag{4.35}$$

Ukazuje se však, že při užití kongruenční metody pro generování hodnot $y_0, y_1, \ldots$ seriální korelace (autokorelace) posloupnosti y_i nepříznivě ovlivní rozdělení hodnot x_i. Doporučuje se proto užívat dvou posloupností rovnoměrných náhodných čísel y_i a y'_i, z nichž každá pochází z jiného kongruenčního generátoru a počítat

$$\begin{aligned} x_{2i} &= -2\ln y_i \sin(2\pi y'_i)\ , \\ x_{2i+1} &= -2\ln y_i \cos(2\pi y'_i)\ , \quad i = 0, 1, \ldots\ . \end{aligned} \tag{4.36}$$

Potřebujeme-li náhodnou veličinu Z s rozdělením $N(\mu, \sigma^2)$ stačí X_n lineárně transformovat

$$Z_n = \sigma X_n + \mu\ . \tag{4.37}$$

4.8.3 Vícerozměrné normální rozdělení $N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

Přirozeným zobecněním normálního rozdělení je vícerozměrné normální rozdělení. Nechť náhodný vektor X má p-rozměrné normální rozdělení $N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ s regulární kovarianční

maticí $\boldsymbol{\Sigma}$. Lze dokázat, že existuje jediná dolní trojúhelníková matice $\boldsymbol{C}$ tak, že

$$\boldsymbol{\Sigma} = \boldsymbol{C}\boldsymbol{C}^T = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \dots & \sigma_{1N} \\ \sigma_{21} & \sigma_{22} & \dots & \sigma_{2N} \\ \dots & \dots & \dots & \dots \\ \sigma_{N1} & \sigma_{N2} & \dots & \sigma_{NN} \end{pmatrix} , \tag{4.38}$$

kde σ_{ii} je rozptyl i-té složky, σ_{ij} kovariance mezi i-tou a j-tou složkou pro $i, j = 1, 2, \dots, N$. Matici $\boldsymbol{C} = [c_{ij}]$ lze z prvků matice $\boldsymbol{\Sigma}$ vypočítat následujícím postupem:

$$\begin{aligned} c_{i1} &= \sigma_{i1}(\sigma_{11})^{-1/2} && \text{pro } 1 \le i \le N , \\ c_{ii} &= (\sigma_{ii} - \sum_{k=1}^{i-1} c_{ik}^2)^{1/2} && \text{pro } 1 < i \le N , \\ c_{ij} &= (\sigma_{ij} - \sum_{k=1}^{i-1} c_{ik}c_{jk})c_{jj}^{-1} && \text{pro } 1 < j < i \le N , \\ c_{ij} &= 0 && \text{pro } i \le j . \end{aligned} \tag{4.39}$$

Potom $\boldsymbol{X}$ lze vyjádřit ve tvaru

$$\boldsymbol{X} = \boldsymbol{\mu} + \boldsymbol{C}\boldsymbol{Y} , \tag{4.40}$$

kde náhodný vektor $\boldsymbol{Y}$ má p-rozměrné normální rozdělení $N_P(0, 1)$. Hodnotu náhodného vektoru $\boldsymbol{X}$ můžeme tedy generovat pomocí p nezávislých normálních náhodných veličin $Y_1, \dots, Y_p$ z $N(0, 1)$.

Pro případ dvourozměrného normálního rozdělení s kovarianční maticí

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix} , \tag{4.41}$$

kde ρ je koeficient korelace mezi složkami, dostáváme

$$\boldsymbol{C} = \begin{pmatrix} \sigma_1 & 0 \\ \sigma_2 & \sigma_2(1-\rho^2)^{1/2} \end{pmatrix} . \tag{4.42}$$

Předpis pro generování vektoru $\boldsymbol{X}$ má tvar

$$\boldsymbol{X} = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \boldsymbol{C}\boldsymbol{Y} + \boldsymbol{\mu} = \begin{pmatrix} \sigma_1 Y_1 \\ \sigma_2\rho Y_1 + Y_2(1-\rho^2)^{1/2} \end{pmatrix} \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} . \tag{4.43}$$

Složky vektoru $\boldsymbol{X}$ mají rozdělení $N(\mu_1, \sigma_1^2)$, resp. $N(\mu_2, \sigma_2^2)$ a koeficient korelace mezi nimi je roven ρ.

4.8.4 Logaritmicko-normální rozdělení

Tohoto rozdělení se často užívá pro aproximaci příjmových rozdělení a v teorii spolehlivosti. Hustota pravděpodobnosti má tvar

$$f(x) = \begin{cases} \dfrac{1}{\sigma x\sqrt{2\pi}} \exp\left[-\dfrac{1}{2}\left(\dfrac{\ln x - \mu}{\sigma}\right)^2\right] & x > 0 , \\ 0 & x \le 0 . \end{cases} \tag{4.44}$$

Střední hodnota a rozptyl jsou rovny:

$$\begin{aligned} E(X) &= e^{\mu+\sigma^2/2} , \\ D(X) &= E^2(X)(e^{\sigma^2}-1) . \end{aligned} \tag{4.45}$$

Z tvaru hustoty pravděpodobnosti logaritmicko-normálního rozdělení je patrné, že má-li náhodná veličina X logaritmicko-normální rozdělení, má náhodná veličina $Z = \ln X$ rozdělení $N(\mu, \sigma^2)$ a veličina

$$V = \frac{\ln X - \mu}{\sigma} \tag{4.46}$$

má rozdělení $N(0,\, 1)$. Řešením této rovnice vzhledem k X dostáváme

$$X = e^{\mu+\sigma V} , \tag{4.47}$$

což představuje vztah, podle kterého lze získat hodnoty logaritmicko-normálního rozdělení na základě hodnot normálního rozdělení $N(0,\, 1)$.

4.8.5 Exponenciální rozdělení

V některých případech (např. rozpad radioaktivních jader, modely teorie hromadné obsluhy atd.) se setkáváme se situacemi, kdy

1. pravděpodobnost výskytu určitého jevu během časového intervalu je úměrná délce tohoto intervalu,

2. nastoupení jevu je statisticky nezávislé na minulosti procesu.

V těchto případech délky intervalů mezi výskyty jevů mají exponenciální rozdělení, jestliže pravděpodobnost

1. nastoupení jevu během intervalu $(t, t+\Delta t)$ je $\lambda\ \Delta t + O(\Delta t)$, kde λ je konstanta nezávislá na t a na jiných činitelích,

2. výskytu více jevů během intervalu $(t, t+\Delta t)$ klesá k nule rychleji než $\lambda\Delta t$ při $\Delta t \to 0$, tj. je rovna $\mathrm{O}(\Delta t)$.

Hustota pravděpodobnosti exponenciálního rozdělení má tvar

$$f(x) = \begin{cases} \lambda e^{-\lambda x} & \lambda > 0,\ x > 0 \\ 0 & x \le 0 \end{cases} . \tag{4.48}$$

Distribuční funkce je potom

$$F(x) = 1 - e^{-\lambda x} . \tag{4.49}$$

Střední hodnota a rozptyl jsou

$$\begin{aligned} E(x) &= \frac{1}{\lambda} , \\ D(x) &= \frac{E(x)}{\lambda} = \frac{1}{\lambda^2} . \end{aligned} \tag{4.50}$$

Je zřejmé, že exponenciální rozdělení má pouze jeden parametr (λ), který je roven převrácené hodnotě $E(X)$. Poznamenejme ještě, že má-li náhodná veličina exponenciální rozdělení s parametrem 1 má náhodná veličina X/λ exponenciální rozdělení s parametrem λ. Stačí tedy mít k dispozici generátor náhodných čísel z exponenciálního rozdělení s parametrem 1.

Standardní způsob generace takovéto náhodné veličiny je založen na inverzní transformaci a byla zde již uvedena - vztah (4.11). Generování hodnot exponenciálního rozdělení takovýmto způsobem je však poměrně pomalé, protože kromě procedury na získávání náhodných čísel je třeba pokaždé vyvolat standardní funkci, sloužící k výpočtu přirozeného logaritmu. Byla vyvinuta celá řada více či méně efektivních algoritmů, které logaritmickou transformaci obcházejí.

Jeden z takovýchto urychlujících algoritmů navrhli Ermakov a Michajlov.
Nechť náhodné veličiny $Y_1, \ldots, Y_n, Z'_1, \ldots, Z'_{n-1}$ jsou rovnoměrně rozděleny v intervalu (0, 1). Nechť $Z_{(-1)} \leq \ldots < Z_{n-1}$ je uspořádaný náhodný výběr ze $Z'_1, \ldots, Z'_{n-1}$ a nechť $Z_{(0)} = 0$, $Z_{(n)} = 1$. Potom

$$X_k = (Z_{(k-1)} - Z_{(k)}) \ln(Y_1 \ldots Y_n) \quad k = 1, \ldots, n \tag{4.51}$$

jsou nezávislé veličiny a jejich hustota pravděpodobnosti je e^{-x}, $x > 0$. Popsaný postup nám tedy umožňuje na jeden výpočet logaritmu získat n hodnot náhodné veličiny X, čímž se výpočet zrychlí. Ukazuje se, že optimum, při němž dosáhneme více než dvojnásobného zrychlení oproti standardnímu algoritmu se docílí pro $n = 3$.

Jiný způsob urychlení navrhl Marsaglia a tento způsob se opírá o následující větu.

Věta 11 *Nechť nezávislé náhodné veličiny N, P mají rozdělení*

$$\begin{aligned} P(N = i) &= \frac{1}{e-1}\frac{1}{i!} \quad i = 1, 2, \ldots \\ P(P = i) &= (e-1)e^{-(i+1)} \quad i = 0, 1, \ldots \ . \end{aligned} \tag{4.52}$$

Nechť Y_1, Y_2, ... jsou nezávislé stejně rozdělené náhodné veličiny s rovnoměrným rozdělením $R(0, 1)$ a nezávislé na N a P. Potom náhodná veličina

$$X = P + \min(Y_1, \ldots, Y_n) \tag{4.53}$$

má exponenciální rozdělení s parametrem 1.

Algoritmus generování náhodného čísla z exponenciálního rozdělení spočívá tedy v tom, že postupně generujeme náhodná čísla n, p, $y_1, \ldots, y_n$ a vypočteme x podle (4.53). Náhodná čísla z rozdělení (4.52) můžeme generovat metodou inverzní funkce. Podrobněji o tom pojednáme v odstavci o generování náhodných čísel ze speciálních diskrétních rozdělení.

4.8.6 Rozdělení gama

Rozdělení gama, jehož speciálním případem pro k=1 je exponenciální rozdělení, má hustotu pravděpodobnosti

$$f(x) = \begin{cases} \frac{1}{\Gamma(k)}\lambda^k x^{k-1} e^{-\lambda x} & x > 0\ , \\ 0 & x < 0 \end{cases} \tag{4.54}$$

a závisí na dvou parametrech $k > 0$, $\lambda > 0$. Střední hodnota a rozptyl jsou

$$\begin{aligned} E(X) &= \frac{k}{\lambda} , \\ D(X) &= \frac{k}{\lambda^2} . \end{aligned} \tag{4.55}$$

Je-li parametr k přirozené číslo, potom se tento speciální případ rozdělení gama nazývá rozdělení Erlangovo, které má hustotu pravděpodobnosti

$$f(x) = \begin{cases} \frac{\lambda^4 x^{k-1} e^{-\lambda x}}{(k-1)!} & x > 0 , \\ 0 & x \leq 0 , \end{cases} \tag{4.56}$$

neboť pro přirozené k platí

$$\Gamma(k) = (k-1)! . \tag{4.57}$$

Erlangovo rozdělení je rozdělení součtu k nezávislých exponenciálně rozdělených veličin s parametrem λ a můžeme ho tedy generovat následovně

$$x = \sum_{i=1}^{k} x_i = -\frac{1}{\lambda} \sum_{i=1}^{k} \ln y_i , \tag{4.58}$$

kde y_i jsou náhodná čísla z $R(0,1)$. Tento vztah můžeme upravit do vhodnějšího tvaru

$$x = \frac{1}{\lambda} \ln \left(\prod_{i=1}^{k} y_i \right) . \tag{4.59}$$

Na následujícím případě budeme demonstrovat použití *metody superpozice*. Máme najít náhodnou veličinu X definovanou na intervalu (0,2) s hustotou pravděpodobnosti

$$f(x) = \frac{5}{12}[1 + (x-1)^4] . \tag{4.60}$$

Použijeme-li metodu inverzní funkce, budeme muset vyřešit vzhledem k X rovnici

$$(X-1)^5 + 5X = 12Y - 1 , \tag{4.61}$$

kde Y je náhodná veličina z $R(0,1)$. Proto dáme přednost metodě superpozice. Hustotu pravděpodobnosti $f(x)$ složíme ze dvou hustot

$$f_1(x) = \frac{1}{2} , \quad f_2(x) = \frac{5}{2}(x-1)^4 , \tag{4.62}$$

$$f(x) = \frac{5}{6} f_1(x) + \frac{1}{6} f_2(x) .$$

Metoda superpozice nám tedy dává následující jednoduchý algoritmus pro určení veličiny X:

$$X = \begin{cases} 2Y_2 & Y_1 < \frac{5}{6} \\ 1 + (2Y_2 - 1)^{1/5} & Y_1 \geq \frac{5}{6} , \end{cases} \tag{4.63}$$

kde veličiny Y_1 a Y_2 jsou z $R(0,1)$.

4.8.7 Náhodný bod v kouli o poloměru R

Naším úkolem je rozehrát náhodný bod P, který je rovnoměrně rozdělen v kouli o poloměru R. Jeho kartézské souřadnice označme x, y, z. Jelikož se jedná o rovnoměrné rozdělení v kouli, hustotu pravděpodobnosti lze vyjádřit následovně

$$f_P(x,y,z) = \frac{1}{\frac{4}{3}\pi R^3} \; . \tag{4.64}$$

Protože vyjádření hustoty pravděpodobnosti pro jednotlivé kartézské souřadnice je těžkopádné, lze s výhodou přejít od této souřadné soustavy k sférické soustavě souřadnic:

$$\begin{aligned} x &= r\sin\theta\cos\varphi \; , \\ y &= r\sin\theta\sin\varphi \; , \\ z &= r\cos\theta \; . \end{aligned} \tag{4.65}$$

V nové soustavě souřadnic se koule transformuje na kvádr:

$$0 \le r \le R \; , \quad 0 \le \theta < \pi \; , \quad 0 \le \varphi < 2\pi \; . \tag{4.66}$$

Jelikož Jacobián určující transformaci souřadnic je roven

$$\frac{\partial(x,y,z)}{\partial(r,\theta,\varphi)} = r^2\sin\theta \tag{4.67}$$

pro hustotu pravděpodobnosti dostáváme

$$f_P(r,\theta,\varphi) = [(4/3)\pi R^3]^{-1} r^2 \sin\theta \; . \tag{4.68}$$

Jak je snadno vidět, tato hustota je vytvořena ze součinu tří hustot pravděpodobnosti

$$f_P(r,\theta,\varphi) = (3r^2R^{-3})(2^{-1}\sin\theta)(2\pi)^{-1} \; , \tag{4.69}$$

a tedy sférické souřadnice r_P, θ_P, φ_P bodu P jsou nezávislé veličiny. Abychom mohli rozehrát náhodný bod P, musíme pro jednotlivé hodnoty Y_1, Y_2 a Y_3 z $R(0,1)$ vyřešit následující integrály vzhledem k r_P, θ_P a φ_P :

$$\int_0^{r_P} \frac{3r^2\,\mathrm{d}r}{R^3} = Y_1 \; , \quad \int_0^{\theta_P} \frac{\sin\theta\,\mathrm{d}\theta}{2} = 1 - Y_2 \; , \quad \int_0^{\varphi_P} \frac{\mathrm{d}\varphi}{2\pi} = Y_3 \; . \tag{4.70}$$

Odtud dostáváme hledané vztahy

$$\begin{aligned} r_P &= R\sqrt[3]{Y_1} \; , \\ \cos\theta_P &= 2Y_2 - 1 \; , \\ \varphi_P &= 2\pi Y_3 \; . \end{aligned} \tag{4.71}$$

Transformací (4.65) můžeme zpětně vyjádřit polohu náhodného bodu P pomocí kartézských souřadnic.

Zde odvozené vztahy můžeme s výhodou použít i při rozehrávání náhodného směru v prostoru. V tomto případě se bude náhodný bod P pohybovat pouze po ploše koule s jednotkovým poloměrem. Označme směrové kosiny jako ω_1, ω_2 a ω_3, tj. platí

$$\begin{aligned} \boldsymbol{\omega} &= \omega_1 \boldsymbol{i} + \omega_2 \boldsymbol{j} + \omega_3 \boldsymbol{k} \,, \\ \omega_1^2 + \omega_2^2 + \omega_3^2 &= 1 \,, \end{aligned} \tag{4.72}$$

kde $\boldsymbol{i}$, $\boldsymbol{j}$, $\boldsymbol{k}$ jsou jednotkové vektory ve směru os x, y, z. Směrové kosiny spočteme ze známých výrazů

$$\begin{aligned} \omega_1 &= \cos\varphi\sqrt{1-\cos^2\theta} \,, \\ \omega_2 &= \sin\varphi\sqrt{1-\cos^2\theta} \,, \\ \omega_3 &= \cos\theta \,. \end{aligned} \tag{4.73}$$

4.8.8 Rozehrávání náhodné veličiny X zadané na intervalu $(0,1)$ s distribuční funkcí $F(x) = x^n$.

Takovou náhodnou veličinu můžeme jednoduše rozehrát pomocí metody inverzní funkce

$$X = \sqrt[n]{Y} \quad , \tag{4.74}$$

kde Y je z $R(0,1)$. Existuje však časově úspornější algoritmus

$$X = \max(Y_1, Y_2, \ldots, Y_n) \,, \tag{4.75}$$

kterého se též s výhodou využívá při výpočtu n-té odmocniny z náhodného čísla.

4.8.9 Generování náhodných čísel ze speciálních diskrétních rozdělení

Obecná metoda generování náhodných čísel z diskrétních rozdělení vychází z diskrétní aplikace metody inverzní transformace. Uvažujme náhodnou veličinu X, která má rozdělení

$$P(X = x_k) = p_k, \quad k = 1, 2, \ldots \,. \tag{4.76}$$

Náhodné číslo x z tohoto rozdělení získáme tak, že generujeme náhodné číslo y z rozdělení $R(0,1)$, nalezneme index i splňující relaci

$$\sum_{j=1}^{i-1} p_j < y \leq \sum_{j=1}^{i} p_j \tag{4.77}$$

a položíme $X = x_i$.

Pokud je možno uchovat v paměti kumulativní součty

$$q_i = \sum_{j=1}^{i} p_j \,, \tag{4.78}$$

je vhodné pro urychlení výpočtu spočítat tyto hodnoty před vlastní simulací. Přestože je tato metoda univerzální, není vždy z časových důvodů nejekonomičtější. V mnoha případech je výhodnější volit speciální postupy šité přímo na míru daného problému.

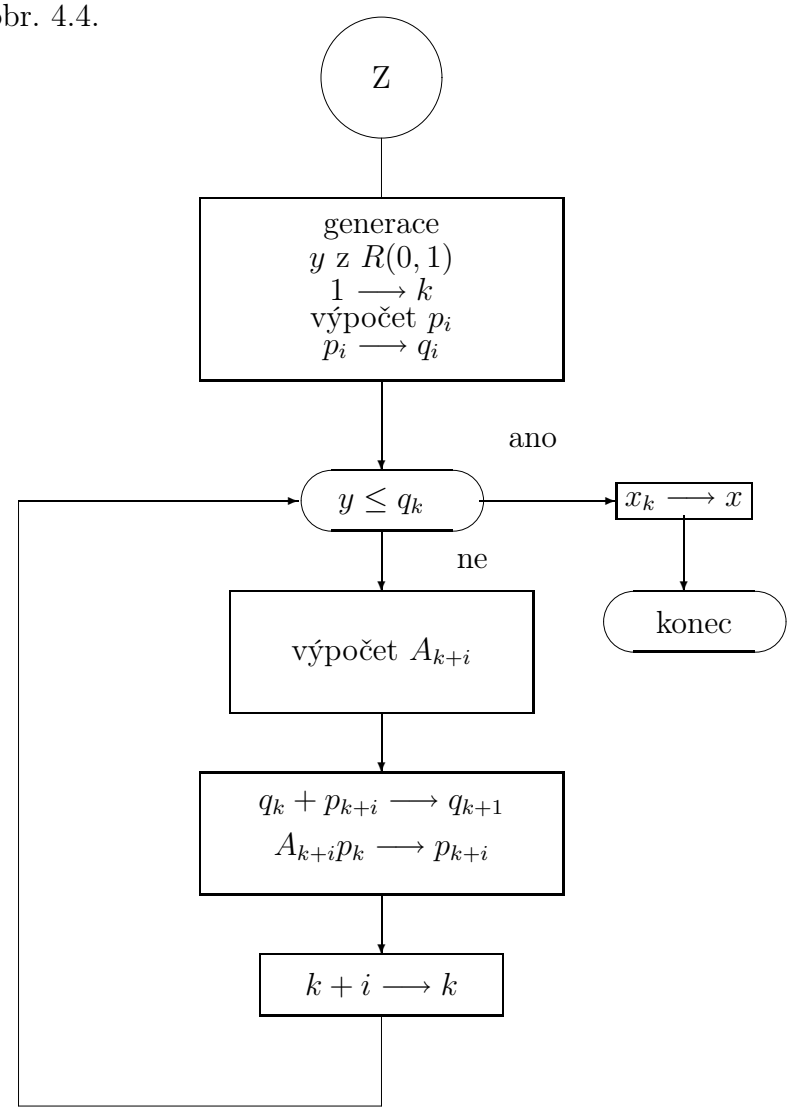
Užíváme-li metodu inverzní transformace, je v mnoha případech diskrétního rozdělení výhodné počítat pravděpodobnosti p_k pomocí rekurentního vztahu

$$p_{k+1} = A_{k+1} p_k \tag{4.79}$$

a kumulované součty počítat pomocí

$$q_{k+1} = q_k + p_{k+1} \ . \tag{4.80}$$

Veličina A_{k+1} závisí na tvaru rozdělení. Poznamenejme, že pro rozdělení nabývající s nenulovou pravděpodobností nekonečně mnoha hodnot, musíme výpočet kumulovaných součtů q_k a tedy i pravděpodobností p_k, provádět v každém kroku simulace. Je tedy žádoucí co nejvíce výpočet těchto veličin zjednodušit - viz. (4.79). Vývojový diagram této metody je na obr. 4.4.



Obrázek 4.4: Vývojový diagram pro metodu (4.79 - 4.80).

Uveďme si nyní metodu Marsagliovu, která je sice náročnější na paměť počítače, ale je zato rychlejší než předcházející metoda. Mějme náhodnou veličinu X, která nabývá

konečně mnoha hodnot $x_1, \ldots, x_r$ s pravděpodobnostmi $p_1, \ldots, p_r$. Předpokládejme, že p_i jsou udána ve tvaru konečných desítkových rozvojů. Pro konkrétnost předpokládejme, že jde o trojmístný rozvoj – p_i jsou tedy vyjádřena v tisícinách. Modifikace metody pro jiný počet míst bude zřejmá z dalšího postupu. V paměti počítače obsadíme prvních $10^3 p_i$ buněk číslem x_1, dalších $10^3 p_2$ buněk číslem x_2 atd.

Nyní nagenerujeme náhodné číslo y z $R(0,1)$. Jeho desetinný rozvoj nechť je $Y = 0.d_1 d_2 d_3 \ldots$. Za náhodné číslo x potom vezmeme číslo na adrese $d_1 d_2 d_3$.

Na závěr úvodních poznámek si ukažme na konkrétním případě, jak znalost průběhu rozdělení může být s výhodou využita na zefektivnění generování náhodných čísel. Mějme náhodnou veličinu X, která nabývá hodnot $x_1, \ldots, x_r$ s pravděpodobnostmi $p_i \leq \ldots \leq p_r$. V tomto případě je efektivnější generovat náhodné číslo x takovým způsobem, že vytváříme postupně kumulativní součty $s_r = p_r$, $s_{r-1} = p_r + p_{r-1}$, $\ldots$, $s_1 = 1$ a za x vezmeme x_i taková, že i ndex i splňuje relaci

$$s_{i+1} < Y \leq s_i \ , \quad Y \text{ z } R(0,1) \ , \tag{4.81}$$

kde $s_{i+1} = 0$. Vztah začínáme testovat z $i = r$ a postupně klademe $i = r-1, r-2, \ldots$. Podobně lze tento postup modifikovat, známe-li modus generovaného rozdělení. Potom je vhodné vycházet z hodnoty kumulovaného součtu odpovídajícího modu.

4.8.10 Binomické rozdělení

Počet výskytů jevu A v sérii n nezávislých pokusů se řídí binomickým rozdělením. Tento náhodný pokus je ekvivalentní výběru s vracením z „urny", ve které jsou prvky dvojího druhu, přičemž pravděpodobnost vybrání prvku jednoho druhu při jednom tahu je p. Binomické rozdělení má velký význam v problematice výběrových šetření, v úlohách z oblasti kontroly jakosti apod. V simulaci má řadu použití např. v modelování procesu selhávání různých částí zařízení nebo při řešení obnovovacích problémů. Hodnoty binomického rozdělení mají pravděpodobnosti

$$P(X = k) = p_k = \left(\begin{array}{c} n \\ k \end{array} \right) p^k (1-p)^{n-k} \quad k = 0, 1, \ldots, n \ . \tag{4.82}$$

Pro velká n se binomické rozdělení blíží normálnímu.

Nejjednodušší metoda generování hodnot binomického rozdělení spočívá v reprodukci série n náhodných pokusů na počítači a registraci počtu nastoupení jevu A. Veličinu X můžeme tedy vyjádřit vztahem

$$X = \sum_{i=1}^{n} I_i \ , \tag{4.83}$$

kde I_i jsou nezávislé alternativní náhodné veličiny, tj. $P(I_i = 1) = p$, $P(I_i = 0) = 1 - p$. Veličiny I_i generujeme jednoduše tak, že generujeme náhodné číslo y z $R(0, 1)$, v případě $y < p$ položíme $I_i = 1$, v opačném případě položíme $I_i = 0$.

Jiný způsob je založen na metodě inverzní transformace, kterou můžeme aplikovat dvojím způsobem. Pokud máme k dispozici dostatečně velkou paměť, je výhodné spočítat před vlastním generováním kumulativní součty $q_0, \ldots, q_n$ a uložit je do paměti. V opačném případě postupujeme podle algoritmu uvedeného na obr. 4.4. (zde však začínáme indexovat od nuly!), kde

$$A_{k+1} = \frac{n-k}{k+1} \frac{p}{1-p} \ . \tag{4.84}$$

Efektivní způsob generování náhodných čísel z binomického rozdělení spočívá však ve využití znalosti průběhu pravděpodobností (4.80). Je známo, že modus binomického rozdělení je

$$m = \text{Int}\,[(n+1)p] \tag{4.85}$$

a tudíž platí

$$p_0 \leq \ldots \leq p_m, \quad p_m \geq \ldots \geq p_n \ . \tag{4.86}$$

Položme

$$Q = \sum_{k=0}^{m} p_k \ . \tag{4.87}$$

Náhodné číslo x z rozdělení (4.82) můžeme získat podle následujícího algoritmu:

1. Generujeme náhodné číslo y z rozdělení $R(0,1)$. Je-li $y \geq Q$, přejdeme k bodu 2, jinak k bodu 3.

2. Postupně klademe $k = m,\ m+1,\ \ldots$ a zjišťujeme, kdy nastane

$$\sum_{j=0}^{k} p_j \leq y < \sum_{j=0}^{k+1} p_j \ .$$

   V případě platnosti relace položíme $x = k+1$.

3. Postupně klademe $k = m,\ m-1,\ \ldots$ a zjišťujeme, kdy nastane

$$\sum_{j=0}^{k-1} p_j \leq x < \sum_{j=0}^{k} p_j \ .$$

   V případě platnosti relace položíme $x = k$.

Tento postup je výhodný i v případě, kdy máme kumulativní součty uložené v paměti.

Poznamenejme, že pro velká n se dá diskrétní funkce (4.82) dobře aproximovat hustotou pravděpodobnosti normálního rozdělení. Této skutečnosti je možné v některých simulačních úlohách využít.

4.8.11 Geometrické rozdělení

Geometricky rozdělená náhodná veličina X představuje počet zdarů v posloupnosti Bernoulliho pokusů předcházejících prvnímu nezdaru a pro její rozdělení platí

$$P(X = k) = p^k(1-p), \quad k = 0, 1, \ldots \ . \tag{4.88}$$

Nejpřirozenější způsob geometrického rozdělení představuje reprodukci série náhodných pokusů na počítači. Generování tedy probíhá podle algoritmu:

1. Položíme $x = 0$.

2. Generujeme náhodné číslo y z $R(0,1)$. Je-li $y \leq p$, zvětšíme x o jedničku a opakujeme bod 2. Je-li $y > p$, je x požadované náhodné číslo.

Druhý způsob generování náhodných čísel z tohoto rozdělení vychází opět z algoritmu uvedeného na obr. 4.4. s tím, že indexace začíná od nuly, $p_0 = 1 - p$ a $A_{k+1} = p$.

Třetí možnost generace využívá vztahu mezi geometrickým a exponenciálním rozdělením. Má-li totiž náhodná veličina W exponenciální rozdělení s parametrem λ, pak platí

$$P(k \leq W < k+1) = e^{-\lambda k}(1 - e^{-\lambda}), \quad k = 0, 1, \dots , \tag{4.89}$$

takže

$$P(\mathrm{Int}(W) = k) = e^{-\lambda k}(1 - e^{-\lambda}). \tag{4.90}$$

Náhodná veličina $X = \mathrm{Int}(W)$ má tedy geometrické rozdělení s parametrem $p = e^{-\lambda}$. Stačí nám tedy generovat náhodná čísla z exponenciálního rozdělení s parametrem $\lambda = -\ln p$.

4.8.12 Poissonovo rozdělení

Toto rozdělení má náhodná proměnná, která nabývá celé nezáporné hodnoty k s pravděpodobností

$$P(X = k) = e^{-\mu} \frac{\mu^k}{k!}, \quad k = 0, 1, \dots . \tag{4.91}$$

Poissonovo rozdělení dává pravděpodobnost výskytu k událostí v daném časovém intervalu, jsou-li tyto události nezávislé a vznikají s konstantní rychlostí (viz. exponenciální rozdělení).

Kromě obecné metody generování diskrétních náhodných veličin (obr. 4.4) lze pro generování náhodných čísel z Poissonova rozdělení užít souvislosti tohoto rozdělení s exponenciálním rozdělením. Jsou-li délky časových intervalů mezi výskytem po sobě následujících jevů určitého typu rozděleny exponenciálně s hustotou pravděpodobnosti

$$f(t) = \lambda e^{-\lambda t}, \quad t > 0 , \tag{4.92}$$

jsou počty výskytů tohoto jevu během časového intervalu t rozděleny podle Poissonova rozdělení

$$P(X = k) = \frac{\lambda t^k}{k!} e^{-\lambda t}, \quad k = 0, 1, \dots . \tag{4.93}$$

Přímým důsledkem tohoto vztahu je, že hodnoty Poissonova rozdělení (4.91) lze chápat jako počty nastoupení jevů určitého druhu v jednotkovém časovém intervalu ($\lambda t \to \mu$), přičemž časové intervaly mezi jevy se řídí exponenciálním rozdělením se střední hodnotou (což se číselně rovná $1/\mu$). Tato úvaha tvoří základ generování náhodných hodnot Poissonova rozdělení.

Náhodné číslo z Poissonova rozdělení získáme tedy tak, že generujeme náhodná čísla $z_0, \dots, z_{k+1}$ z rozdělení exponenciálního s parametrem μ tak dlouho, až platí

$$\sum_{j=0}^{k} z_j \leq 1 < \sum_{j=0}^{k+1} z_j . \tag{4.94}$$

Číslo k je potom náhodné číslo z Poissonova rozdělení. Obvykle generujeme z pomocí náhodných čísel y z $R(0, 1)$ podle vztahu

$$z_j = -\frac{1}{\mu} \ln y_j . \tag{4.95}$$

Po dosazení do vztahu (4.94) pak dostáváme

$$-\sum_{j=0}^{k} \ln y_j \leq \mu < \sum_{j=0}^{k+1} \ln y_j \tag{4.96}$$

nebo-li

$$\prod_{j=0}^{k} y_j \leq e^{-\mu} < \prod_{j=0}^{k+1} y_j \ . \tag{4.97}$$

Tento vztah je pro výpočet nejvhodnější, neboť konstantu $e^{-\mu}$ můžeme spočítat před vlastním generováním.

4.8.13 Empirická rozdělení

V předchozích odstavcích jsme se zabývali generováním hodnot náhodných veličin za předpokladu, že jsme znali jejich rozdělení ve tvaru matematického zákona.

Při konstrukci simulačních modelů však málokdy disponujeme apriorní znalostí náhodných veličin, které se v modelech vyskytují. První úkol, před kterým stojíme, spočívá v získání dat přiměřeného rozsahu a kvality, na jejichž základě se snažíme popsat pravděpodobnostní zákony, podle nichž nabývají veličiny svých hodnot.

Nejprve je třeba zjistit, zda empiricky zjištěné rozdělení lze s dostatečnou přesností aproximovat některým ze standardně používaných rozdělení. K testování shody empirických rozdělení s teoretickými se používají testy dobré shody (χ^2 – test, Kolmogorovův – Smirnovův test). Pokud na zvolené hladině významnosti nejsou odchylky podstatné, můžeme většinou použít některé dříve uvedené metody získávání hodnot náhodných veličin. V mnoha případech však tomu tak není a je lepší generovat náhodná čísla přímo z empirického rozdělení. Pro diskrétní rozdělení je pak možno použít přímo metodu inverzní transformace.

Generování náhodných čísel ze spojitých rozdělení na základě znalosti empirické distribuční funkce lze provádět pomocí následujícího postupu. Nechť je dána n-tice výsledků měření náhodné veličiny $X - \xi_1, \ldots, \xi_n$ uspořádaná podle velikosti od nejmenší k největší hodnotě. Empirická distribuční funkce potom je

$$S_n(x) = \begin{cases} 0 & \text{pro} \quad x < \xi_1 \ , \\ i/n & \text{pro} \quad x \in \langle \xi_i, \xi_{i+1}), \quad i = 1, \ldots, n-1 \ , \\ 1 & \text{pro} \quad x \geq \xi_n \ . \end{cases} \tag{4.98}$$

Místo této schodovité funkce budeme uvažovat spojitou distribuční funkci $F(x)$, která bude po částech lineární a spojitá a $S_n(\xi_i) = F(\xi_i)$, $i = 1, \ldots, n$. Tzn., že pro $\xi_i \leq x < \xi_{i+1}$ je

$$F(x) = S_n(\xi_i) + \frac{S_n(\xi_{i+1}) - S_n(\xi_i)}{\xi_{i+1} - \xi_i}(x - \xi_i) = S_n(\xi_i) + \frac{(x - \xi_i)}{n(\xi_{i+1} - \xi_i)} \ . \tag{4.99}$$

Aplikujeme-li nyní na tuto distribuční funkci metodu inverzní funkce, dostáváme pro rozehrání náhodné veličiny ze zadaného empirického rozdělení následující algoritmus:

1. Generujeme náhodné číslo y z rozdělení $R(0, 1)$.

2. Je-li $y \geq S_n(\xi_n), y$ zamítneme a přejdeme k bodu 1.

3. Určíme k takové, že $S_n(\xi_k) \leq y < S_n(\xi_{k+1})$.

4. Položíme

$$x = n(\xi_{k+1} - \xi_k)[y - S_n(\xi_k)] + \xi_k \ . \tag{4.100}$$

4.8.14 Generování náhodných permutací

V některých simulačních modelech, např. pro generování náhodných výběrů bez vracení, je třeba generovat náhodná pořadí N prvků.

Očíslujme uvažované prvky čísly 0, 1, ..., N-1. Existuje potom $N!$ různých permutací. Jedna z metod generování náhodných pořadí čísel 0, 1, ..., N-1, navržená Marshallem a Halem, je založena na vzájemně jednoznačném přiřazení mezi prvky dvou množin:

- množiny A, jejíž prvky jsou čísla 0, 1, 2, ..., $N!$ - 1 ;
- množiny B, jejíž prvky jsou všechna pořadí čísel 0, 1, 2, ..., $N-1$.

Zobrazení množiny B na množinu A lze provést tak, že pro každou permutaci b_0, b_1, b_2, ..., b_{N-1} čísel 0, 1, 2, ..., $N-1$ najdeme čísla a pro $k = 1, 2, \ldots, N-1$, která značí počet čísel, jež jsou v řadě b_0, b_1, ..., b_{N-1} uvedena za číslem k a jsou menší než k. Např. v posloupnosti ($N = 8$)

$$3, 7, 1, 0, 4, 5, 6, 2 \tag{4.101}$$

je $a_1 = 1$, $a_2 = 0$, $a_3 = 3$, $a_4 = 1$, $a_5 = 1$, $a_6 = 1$, $a_7 = 6$. V dané permutaci je např. $a_1 = 1$, protože za číslem 1 se vyskytuje pouze jedno menší číslo než 1, totiž číslo 0.

Pro určení čísel a_k pro $k = 1, 2, \ldots, N-1$ definujeme číslo I vztahem

$$I = a_1\ 1! + a_2\ 2! + \ldots + a_{N-1}\ (N-1)! \ , \tag{4.102}$$

které je pro každou permutaci jedinečné. Toto číslo, jak je zřejmé, leží v množině A a lze jej použít jako pořadového čísla permutace b_0, b_1, ..., b_{N-1}.

Naopak každému prvku množiny A (celému číslu $I \in [0, N! - 1]$) můžeme nejprve přiřadit čísla a_1, a_2, ..., a_{N-1} tak, že postupným dělením I čísly $(N-1)!$, $(N-2)!$, ..., 2! nalezneme koeficienty rozkladu (4.101). Zbývá ještě ukázat, jakým způsobem lze na základě znalosti hodnot a vytvořit příslušnou permutaci. To je již snadné, uvědomíme-li si, že a_1 určuje vzájemnou polohu čísel 0 a 1, a_2 vzájemnou polohu čísel 0, 1 a 2, atd. Například pro soustavu čísel a_k uvedenou pro permutaci (4.101) budeme postupovat následovně

					1	0		za jedničkou má následovat jedno menší číslo,
					1	0	2	za dvojkou nenásleduje žádné menší číslo,
				3	1	0	2	za trojkou následují tři menší čísla,
			3	1	0	4	2	...
		3	1	0	4	5	2	...
	3	1	0	4	5	6	2	...
3	7	1	0	4	5	6	2	za sedmičkou následuje právě šest menších čísel

Generování náhodných permutací N čísel tedy spočívá v generování celých čísel I z intervalu $[0, N! - 1]$, které mají stejnou pravděpodobnost výskytu, nalezení koeficientů

rozkladu a ve vztahu (4.102) a konstrukci pořadí čísel 0, 1, ..., $N-1$ podle uvedeného algoritmu.

Je zřejmé, že postup je realizovatelný pouze pro poměrně malé N. Například u počítačů s délkou slova 24 bitů má smysl uvažovat $N \leq 10$ (pokud nepoužijeme dvojné aritmetiky), neboť 11! již přesahuje možnosti zobrazení čísel v pevné řádové čárce.

Pro větší počet prvků lze použít algoritmus založený na principu náhodného „promíchávání". Jestliže $x_1, x_2, \ldots, x_N$ je určitá permutace N prvků, získáme další permutaci následujícím postupem.

Pro $i = N, N-1, \ldots, 2$ se provedou operace 1) až 3):

1. generuj y z $R(0, 1)$,
2. $k = 1 + \mathrm{Int}(iy)$,
3. vymění se prvky x_k a x_i (tj. $x_k \longleftrightarrow x_i$).

4.8.15 Markovovy řetězce

V tomto odstavci si v krátkosti všimneme Markovových řetězců, jednak z toho důvodu, že Markovovy řetězce mají široké uplatnění v přírodních vědách, a jednak proto, že jedna z metod řešení lineárních algebraických rovnic vede k simulaci Markovova řetězce. S tímto pojmem se rovněž setkáme v kapitole věnované statistické termodynamice.

V řadě modelů popisujících chování dynamických systémů se vyskytuje potřeba zachytit vazby mezi situacemi následujícími po sobě. Dochází-li ke změnám stavů v diskrétních časových okamžicích a jedná-li se o množinu diskrétních stavů můžeme někdy předpokládat závislost mezi stavy systému pouze v okamžicích t a $t+1$. Nezávisí-li tvar této závislosti na t, lze uvažovat o použití homogenních Markovových řetězců pro pravděpodobnostní popis vývoje takového systému.

Možné stavy systému zobrazme do množiny přirozených čísel, tj. proveďme jejich očíslování čísly 1, 2, ..., N. Okamžiky, v nichž může docházet ke změnám stavů, označme 0, 1, 2, ... Markovovův řetězec je definován vektorem pravděpodobnosti stavů systému v počátečním okamžiku, tj. vektorem

$$p(0) = [p_1(0), p_2(0), \ldots, p_N(0)], \quad p_i(0) \geq 0, \quad \sum_{i=1}^{N} p_i(0) = 1 \,, \tag{4.103}$$

kde $p_i(0)$, pro $i = 1, 2, \ldots, N$, je pravděpodobnost, že systém je v okamžiku $t = 0$ v i – tém stavu a maticí podmíněných pravděpodobností přechodu

$$\boldsymbol{P} = [p_{ij}] \,, \quad p_{ij} \geq 0, \quad \sum_{j=1}^{N} p_{ij} = 1 \quad \text{pro} \quad i = 1, 2, \ldots, N \,, \tag{4.104}$$

jejíž prvky p_{ij} značí pravděpodobnost, že systém, který je v okamžiku t ve stavu i, bude v okamžiku $t+1$ ve stavu j. Zadáním vektoru $p(0)$ a stochastické matice $\boldsymbol{P}$ je Markovovův řetězec plně popsán (v pravděpodobnostním smyslu), neboť platí vztah

$$p(k) = p(k-1)\boldsymbol{P} = p(0)\boldsymbol{P}^k \quad \text{pro} \quad k = 1, 2, \ldots \,. \tag{4.105}$$

Je-li systém v okamžiku t ve stavu i, bude v následujícím okamžiku ve stavu 1 s pravděpodobností p_{i1}, ve stavu 2 s pravděpodobností p_{i2}, ..., ve stavu N s pravděpodobností p_{iN}.

Generování následujícího stavu je vlastně generováním hodnoty diskrétní náhodné veličiny s hodnotami 1, 2, ..., N, jež se vyskytují s pravděpodobnostmi udanými i–tým řádkem matice podmíněných pravděpodobností přechodu. Z matice $\boldsymbol{P}$ je výhodné vytvořit matici kumulovaných pravděpodobností $\boldsymbol{F} = [F_{ij}]$ podle vztahu

$$F_{ij} = \sum_{k=1}^{j} p_{ik} \quad \text{pro} \quad i = 1, 2, \ldots, N \; . \tag{4.106}$$

Jednoduchý algoritmus generování stavu systému v okamžiku $t + 1$ za předpokladu, že v okamžiku t byl ve stavu i, můžeme zapsat následovně:

1. generování čísla y z $R(0, 1)$,
2. nalezení nejmenšího j, pro které platí $y \leq F_{ij}$.

Někdy se vyskytuje případ tzv. vícenásobné vazby, kdy následující situace nezávisí jenom na přítomném stavu, ale i na stavech v několika předchozích okamžicích. Ukážeme, že i pro tyto případy se problém generování posloupnosti stavů nijak podstatně nekomplikuje. Podstata bude patrná na dvojné vazbě, algoritmus lze zřejmým způsobem zobecnit na složitější vazby.

Při dvojné vazbě je pravděpodobnost, že systém bude v okamžiku $t + 1$ ve stavu k za předpokladu, že v okamžiku t je ve stavu j a v okamžiku $t-1$ byl ve stavu i, udána prvkem p_{ijk}. Pro určení řetězce tohoto typu je nutné udat stavy (popř. vektory pravděpodobností stavů) ve dvou počátečních okamžicích. Zde podmíněné pravděpodobnosti přechodu tvoří krychlovou matici

$$\boldsymbol{P} = [p_{ijk}], \quad p_{ijk} \geq 0, \quad \sum_{k=1}^{N} p_{ijk} = 1 \quad \text{pro} \quad i, j = 1, 2, \ldots, N \; , \tag{4.107}$$

kde i je číslo vrstvy, j je číslo sloupce, k je číslo řádku. Pro generování je opět vhodné spočíst veličiny F_{ijk},

$$F_{ijk} = \sum_{m=1}^{k} p_{ijm} \; . \tag{4.108}$$

Generování stavu v okamžiku $t + 1$ (byl-li systém v okamžiku $t - 1$ a t ve stavech i a j) spočívá v krocích:

1. generujeme náhodné číslo y z $R(0, 1)$,
2. nalezneme nejmenší k, pro které platí

$$y \leq F_{ijk} \; . \tag{4.109}$$

V praxi se vyskytují i případy, kdy aproximace homogenním Markovovým řetězcem nepostačuje k vystižení změn stavů systému. Někdy je to způsobeno tím, že podmíněné pravděpodobnosti přechodu se mění s časovými okamžiky, takže musíme pracovat s maticemi $P(t)$, které popisují přechody uskutečněné mezi okamžiky $t - 1$ a t. Především jsou zajímavé případy, v nichž se matice $\boldsymbol{P}(t)$ po určité době opakují, takže posloupnost matic má tvar

$$\boldsymbol{P}(1), \boldsymbol{P}(2), \ldots, \boldsymbol{P}(k), \boldsymbol{P}(1), \boldsymbol{P}(2), \ldots \boldsymbol{P}(k), \boldsymbol{P}(1), \ldots \; .$$

Ani zde nepředstavuje vytvoření algoritmu generování náhodných posloupností stavů vážný problém.

Kapitola 5

Výpočet určitých integrálů

Pro výpočet hodnoty určitého integrálu se používá celé řady známých numerických metod (lichoběžníkové pravidlo, Simpsonova formule, Gaussovy vzorce apod.), které ve většině případů umožňují získat výsledek se žádanou přesností, aniž by objem výpočetních prací přesáhl rozumnou mez. Pro jednorozměrné integrály metoda Monte Carlo neobstojí v soutěži s uvedenými metodami, neboť k dosažení stejné přesnosti obvykle vyžaduje mnohem více výpočetních úkonů. Situace se však změní, když přejdeme k výpočtu vícerozměrných integrálů. Budeme-li při výpočtu jednorozměrných integrálů brát funkční hodnoty v n uzlových bodech, při přechodu k d-rozměrnému integrálu počet uzlových bodů vzroste na nd a doba výpočtu poroste toutéž rychlostí. Naproti tomu v metodě Monte Carlo doba výpočtu roste s přechodem k d–rozměrům pouze d–krát. Z tohoto důvodu je metoda Monte Carlo základní numerickou metodou pro výpočet vícerozměrných integrálů.

Pokud máme integrovat nějakou hladkou funkci f na vhodné oblasti integrace G, tak se ukazuje, že použití metody Monte Carlo je výhodnější, než použití klasických kvadraturních vzorců pro dimenzi $d > 3$. Bude-li funkce f spojitá jen po částech nebo bude-li oblast integrace složitá, používá se metoda Monte Carlo již pro $d > 2$. Použití metody Monte Carlo může být výhodné i v jednorozměrném případě, pokud funkce f má mimořádně složitý průběh.

Jelikož jde o výklad principů této metody, zaměříme se na problematiku přibližných výpočtů jednoduchých integrálů, na níž se základní problémy dají názorně ilustrovat. Efektivnost výpočtu integrálů různými způsoby budeme demonstrovat na konkrétním příkladě [2, 5, 8].

5.1 Jednoduché metody pro výpočet integrálů

Naším úkolem bude vypočítat integrál z funkce $f(x)$ na intervalu (a, b)

$$I = \int_a^b f(x)\,\mathrm{d}x\ . \tag{5.1}$$

Tento výpočet můžeme provést dvěma způsoby.

5.2 Výpočet pomocí střední hodnoty

Tato metoda výpočtu integrálu I využívá následujícího faktu. Nechť X je náhodná veličina s rovnoměrným rozdělením na intervalu (a, b) a náhodná veličina Y je dána vztahem $Y = f(X)$. Pak pro střední hodnotu veličiny Y platí

$$E(Y) = \int_a^b f(x) \frac{1}{b-a} \, \mathrm{d}x = \frac{1}{b-a} I \; . \tag{5.2}$$

Z druhé strany střední hodnotu náhodné veličiny Y můžeme odhadnout pomocí aritmetického průměru nezávislých n realizací náhodné veličiny Y

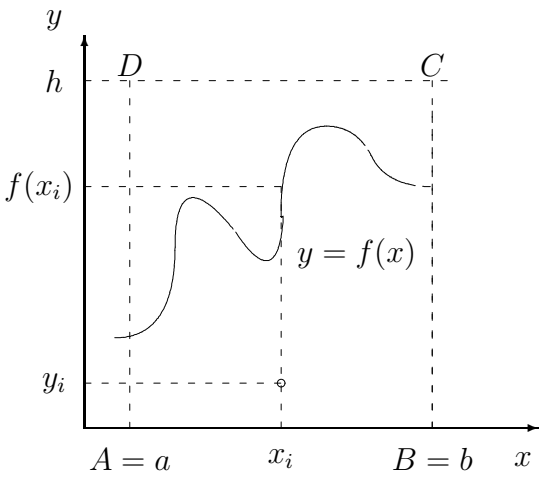
$$E(Y) \doteq \frac{1}{n} \sum_{i=1}^{n} f(x_i) \quad , \tag{5.3}$$

kde x_i jsou nezávislé realizace náhodné veličiny X, tj. náhodná čísla s rovnoměrným rozdělením na intervalu (a, b). Pro hodnotu integrálu I pak dostáváme

$$I \doteq (b-a) \frac{1}{n} \sum_{i=1}^{n} f(x_i) \; . \tag{5.4}$$

5.3 Geometrická metoda

Metoda využívá tzv. geometrické pravděpodobnosti. Situace je znázorněna na obr. 5.1.



Obrázek 5.1: Geometrická metoda výpočtu integrálu

Geometrická pravděpodobnost plochy pod křivkou $y = f(x)$ je rovna

$$P_I = \frac{I}{(b-a)h} \quad . \tag{5.5}$$

Nyní si představme následující pokus. Zvolme v obdélníku ABCD náhodně body. Nechť rozdělení těchto bodů je rovnoměrné na ploše obdélníku ABCD. Pak pravděpodobnost,

že takto náhodně zvolený bod padne do plochy pod křivkou $y = f(x)$ je rovna právě geometrické pravděpodobnosti této plochy.

Při výpočtu integrálu I budeme postupovat následovně. Generujme n náhodných čísel x s rovnoměrným rozdělením na intervalu (a, b) a posloupnost n náhodných čísel y s rovnoměrným rozdělením na intervalu $(0,\ h)$ (označení viz. obr. 5.1).

Dvojice $(x_i,\ y_i)$ pak představují souřadnice náhodných bodů v obdélníku ABCD. Pomocí podmínky

$$y_i < f(x_i) \tag{5.6}$$

zjistíme, kolik bodů leží pod křivkou $y = f(x)$. Jejich počet označme n'. Relativní četnost n'/n pak bude odhadem geometrické pravděpodobnosti

$$P_I = \frac{I}{(b-a)h} \doteq \frac{n'}{n} \tag{5.7}$$

a pro hodnotu integrálu dostáváme vztah

$$I \doteq h(b-a)\frac{n'}{n} \quad . \tag{5.8}$$

Obě dvě metody výpočtu integrálů dle našeho názoru krásně ilustrují podstatu metody Monte Carlo. Pro výpočet hodnoty určitého integrálu, což je konkrétní, pevně dané číslo, jsme našli náhodné veličiny (např. $Y = f(X)$ u první metody), jejichž charakteristiky (u první metody střední hodnota $E(Y)(b-a)$) jsou rovny hledané hodnotě integrálu a hodnotu těchto charakteristik určíme simulováním hodnot náhodných veličin a odhadem z těchto hodnot pomocí matematické statistiky.

Z detailnějšího rozboru obou metod vyplývá, že geometrická metoda má větší nebo v optimálním případě stejný rozptyl jako první metoda založená na výpočtu střední hodnoty a bývá proto méně přesná.

5.4 Odhad účinnosti metody

Obdobně jako v případě generátoru náhodných veličin s daným rozdělením musíme se dívat na účinnost dané metody jednak z hlediska přesnosti výpočtu daného integrálu a jednak z hlediska výpočetní rychlosti. Z tohoto důvodu je rozumné za kritérium vhodnosti metody vzít součin času potřebného k výpočtu jednoho odhadu integrálu I a odpovídající disperze.

Určit rozdělení odhadu integrálu přímo se však ve většině případů nepodaří. Proto musíme postupovat jinak. Využijeme toho faktu, že hodnotu integrálu odhadujeme pomocí aritmetického průměru. Pokud jednotlivé členy v aritmetickém průměru budou mít disperzi $D(X)$, pak disperze aritmetického průměru bude $D(X)/n$, kde n je počet členů aritmetického průměru. Jestliže tedy spočteme disperzi $D(X)$ jednotlivých členů aritmetického průměru, pak disperze odhadu integrálu bude $D(X)/n$.

Pro názornost aplikujme toto kritérium pro případ výpočtu integrálu

$$I = \int_0^1 \sqrt[5]{x}\,\mathrm{d}x \tag{5.9}$$

oběma výše uvedenými metodami. Tento integrál je roven 5/6. Postupujeme-li podle první metody, dostáváme pro výpočet I vztah

$$I \doteq \frac{1}{n}\sum_{i=1}^{n} \sqrt[5]{x_i}\,, \tag{5.10}$$

a disperze náhodné veličiny $Y = (X)^{1/5}$ (X z $R(0,\,1)$) bude

$$D(Y) = \int_0^1 x^{2/5}\,\mathrm{d}x - \left(\frac{5}{6}\right)^2 = \frac{5}{252}\,. \tag{5.11}$$

V případě geometrické metody dostáváme pro I vztah

$$I \doteq \frac{1}{n}\sum_{i=1}^{n} j_i\,, \tag{5.12}$$

kde

$$\begin{aligned} j_i &= 1 \quad \text{pro} \quad y_i < x_i^{1/5} \\ j_i &= 0 \quad \text{pro} \quad y_i \geq x_i^{1/5}\,. \end{aligned} \tag{5.13}$$

Uvědomíme-li si, že střední hodnota čtverce náhodné veličiny J je rovna

$$E(J^2) = 1^2 P(Y < f(x)) = I \tag{5.14}$$

potom pro disperzi veličiny J dostáváme

$$D(J) = I - I^2 = \frac{5}{6} - \frac{25}{36} = \frac{5}{6}\,. \tag{5.15}$$

Pokud bychom porovnali pouze disperze u obou případů, je první metoda jednoznačně výhodnější. Uvážíme-li i počet operací potřebných pro získání jednoho členu v součtech tvořících odhady I, bude situace komplikovanější. V metodě založené na určení střední hodnoty funkce $f(x)$ musíme nagenerovat náhodné číslo x_i a počítat pátou odmocninu z tohoto čísla, což je obvykle časově náročná procedura. V případě geometrické metody musíme nagenerovat dvě náhodná čísla y_i a x_i a testovat podmínku $y_i < x_i^{1/5}$, kterou lze převést na ekvivalentní podmínku $y_i^5 < x_i$. Vyhodnocení této podmínky je časově daleko méně náročné než výpočet odmocniny a geometrická metoda i přes větší disperzi je vhodnější metodou. Vzpomeneme-li si však na výpočet n–té odmocniny z náhodného čísla podle vztahu (4.75), výpočet odmocniny se zrychlí a situace se obrátí – první metoda se stane více jak třikrát efektivnější, než geometrická metoda. Z uvedeného příkladu je zřejmé, že volba nejvhodnějšího algoritmu v metodě Monte Carlo není vždy jednoduchou záležitostí.

5.5 Metody snižování disperze

Konvergence obvyklé metody Monte Carlo je pomalá – úměrná $n^{-1/2}$. Snižování disperze zvyšováním počtu pokusů není tedy příliš efektivní způsob. Existují však speciální metody snižující disperzi a některé z nich si nyní uvedeme.

5.6 Nalezení hlavní části

Myšlenka spočívá v rozdělení integrálu $\int_a^b f(x)\,\mathrm{d}x$ na dvě části - hlavní část, kterou lze spočíst analyticky či jinou numerickou metodou a korekční část, kterou určíme pomocí metody Monte Carlo. Nejdříve tedy najdeme funkci $g(x)$ blízkou $f(x)$, jejíž integrál známe

$$I_1 = \int_a^b g(x)\,\mathrm{d}x\ . \tag{5.16}$$

Korekci spočteme např. metodou střední hodnoty pro funkci $f(x) - g(x)$ a pro hledaný integrál máme

$$I \doteq \frac{b-a}{n}\sum_{i=1}^{n}[f(x_i) - g(x_i)] + I_1\ . \tag{5.17}$$

Označme si jako náhodnou veličinu V

$$V = (b-a)[f(X) - g(X)] + I_1\ . \tag{5.18}$$

Pro její disperzi dostáváme

$$D(V) = (b-a)\int_a^b [f(x) - g(x)]^2\,\mathrm{d}x - (I - I_1)^2\ . \tag{5.19}$$

5.7 Metoda váženého výběru

Doposud jsme náhodná čísla x generovali rovnoměrně rozdělená na intervalu integrace. Jak vyplyne z dalšího, je někdy výhodnější volit tato čísla s takovým rozdělením, že jejich hustota výskytu bude větší v těch oblastech integrace, které více přispívají k hodnotě integrálu I. V nejjednodušším případě lze postupovat tak, že obor integrace rozdělíme na jednotlivé vhodně zvolené podintervaly a v nich opět generujeme náhodná čísla x_i z rovnoměrného rozdělení. Hustotu rozdělení volíme podle velikosti integrované funkce v dílčím intervalu. Výsledný odhad pro integrál I potom dostaneme jako součet výsledků z jednotlivých podintervalů. Této metodě se též říká metoda výběru po skupinách.

V obecnějším přístupu volíme náhodnou veličinu X s hustotou pravděpodobnosti $p(x)$ spojitě se měnící v intervalu $[a, b]$. Hledaný integrál upravíme do tvaru

$$I = \int_a^b f(x)\,\mathrm{d}x = \int_a^b \frac{f(x)}{p(x)}p(x)\,\mathrm{d}x\ . \tag{5.20}$$

Zavedeme-li náhodnou veličinu $V = f(X)/p(X)$, bude pro ni platit $E(V) = I$. Jako odhad integrálu proto můžeme vzít

$$I \doteq \frac{1}{n}\sum_{I=1}^{n}\frac{f(x_i)}{p(x_i)}\ , \tag{5.21}$$

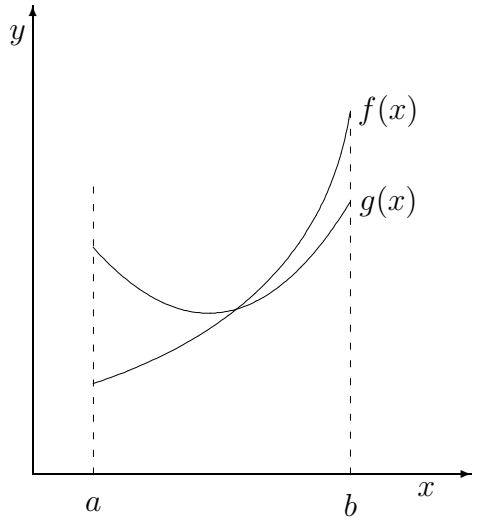
kde x_i jsou náhodná čísla s hustotou $p(x)$. Rozptyl této náhodné veličiny V bude

$$D(V) = \int_a^b \frac{f^2(x)}{p(x)}\,\mathrm{d}x - I^2\ . \tag{5.22}$$

Z tohoto vztahu plyne důležitý závěr, že vhodnou volbou hustoty pravděpodobnosti můžeme podstatně zmenšit disperzi. Teoreticky bychom mohli snížit disperzi až na nulu, kdybychom za hustotu pravděpodobnosti $p(x)$ vybrali funkci

$$p(x) = \frac{|f(x)|}{\int_a^b |f(x)|\,\mathrm{d}x} \ . \tag{5.23}$$

Problém je ovšem v tom, jak takovouto hustotu pravděpodobnosti (vyjma jednoduchých případů) realizovat. V praxi postupujeme tak, že generujeme takovou náhodnou veličinu X, jejíž hustota pravděpodobnosti se co nejvíce podobá $|f(x)|$ a je snadno realizovatelná.



Obrázek 5.2: Symetrizace funkce $f(x)$

5.8 Metoda symetrizace integrované funkce

Uvědomíme-li si, že disperze konstanty je nula, a že disperze integrálu bude tím menší, čím méně se bude integrovaná funkce na intervalu integrace měnit, dostaneme další metodu snižování disperze. Budeme se tedy snažit vhodným způsobem upravit integrovanou funkci tak, aby na daném intervalu $[a, b]$ příliš nekolísala a tím snížíme disperzi. Konkrétní úprava závisí na tvaru funkce $f(x)$.

Pokud je funkce monotonní (obr.5.2), je vhodné ji symetrizovat funkcí

$$g(x) = \frac{1}{2}[f(x) + f(a + b - x)] \ , \tag{5.24}$$

přičemž platí

$$\int_a^b f(x)\,\mathrm{d}x = \int_a^b g(x)\,\mathrm{d}x \ . \tag{5.25}$$

Odhad hodnoty integrálu je potom dán vztahem

$$I = \frac{1}{2n}\sum_{i=1}^{n}[f(x_i) + f(a + b - x_i) \ . \tag{5.26}$$

Doba vypočtu jednoho členu se prodlouží přibližně dvakrát. Lze však dokázat, že v případě monotónní funkce disperze klesne minimálně dvakrát.

V případě složitějšího průběhu je nutno provádět symetrizaci složitěji. Abychom mohli s úspěchem použít této metody, musíme získat co nejvíce informací o integrované funkci.

5.9 Příklady integračních metod se sníženým rozptylem

V této části porovnáme výše uvedené algoritmy na jednoduchém integrálu (jehož hodnotu známe)

$$I = \int_0^1 e^x \,\mathrm{d}x = e - 1 \quad . \tag{5.27}$$

Disperzi náhodné veličiny V, kterou lze vyjádřit jako $V = g(X)$ a jejíž střední hodnotou je hodnota I, budeme počítat ze vztahu

$$D(V) = \int_0^1 g^2(x) \,\mathrm{d}x - I^2 \,. \tag{5.28}$$

5.10 Metoda střední hodnoty

Odhad hodnoty integrálu je dán vztahem

$$I \doteq \frac{1}{n} \sum_{i=1}^{n} e^{x_i} \tag{5.29}$$

a disperze náhodné veličiny $V = e^X$ je

$$D(V) = \frac{1}{2}(e^2 - 1) - (e - 1)^2 = 0.2420 \,. \tag{5.30}$$

5.11 Geometrická metoda

Odhad hodnoty integrálu je dán vztahem

$$I \doteq \frac{e}{n} \sum_{i=1}^{n} v_i \,, \quad v_i = 1 \text{ pro } \quad y_i < e^{x_i - 1}, \text{ jinak } \quad v_i = 0 \,. \tag{5.31}$$

Zvolili jsme $h = e$. Disperze bude v tomto případě značná

$$D(V) = hI - I^2 = e(e - 1) - (e - 1)^2 = e - 1 = 1.7183 \,. \tag{5.32}$$

5.12 Metoda nalezení hlavní části

Funkci $f(x) = \exp(x)$ můžeme rozvinout do řady $1 + x + x^2/2! + \ldots$. Za hlavní část si v intervalu [0,1] zvolme funkci $g(x) = 1 + x$. Její integrál lze analyticky spočíst a je roven I_i=3/2. Pro odhad hodnoty integrálu a disperze dostáváme

$$I \doteq \sum_{i=1}^{n}(e^{x_i} - x_i - 1) + \frac{3}{2} = \frac{1}{n}\sum_{i=1}^{n}(e^{x_i} - x_i) + \frac{1}{2}\,, \tag{5.33}$$

$$D(V) = \int_0^1 (e^x - 1 - x)^2\,\mathrm{d}x - (e - 2.5)^2 = 0.0437\,. \tag{5.34}$$

5.13 Metoda váženého výběru

- Funkce $\exp(x)$ je monotonně rostoucí funkce na intervalu [0, 1]. Rozdělme tento interval na dva podintervaly [0,1/2) a [1/2, 1]. V prvním podintervalu vezmeme 40 % a ve druhém 60 % hodnot, což odpovídá poměru ploch odpovídajících daným intervalům. Pro výpočet integrálu dostáváme

$$I = \frac{1}{0.8n}\sum_{i=1}^{0.4n} e^{x_i'} + \frac{1}{12n}\sum_{i=0.4n}^{n} e^{x_i''}, \quad x = 0.5x\,, \quad x\text{"} = 0.5(1 + x)\,, \quad x \text{ z } R(0,1)\,. \tag{5.35}$$

  Disperze bude

$$D(V) = \frac{1}{1.6}D + \frac{1}{2.4}D = 0.0614\,. \tag{5.36}$$

- Za hustotu pravděpodobnosti $p(x)$ náhodné veličiny X zvolme například $\frac{2}{3(1+x)}$. Tuto náhodnou veličinu můžeme rozehrát pomocí metody inverzní transformace

$$\int_0^X p(x)\,\mathrm{d}x = Y\,, \quad Y \text{ z } R(0,1) \tag{5.37}$$

  a tedy

$$X = \sqrt{1 + 3Y} - 1\,. \tag{5.38}$$

  Hodnota integrálu je dána vzorcem

$$I = \frac{3}{2n}\sum_{i=1}^{n}\frac{e^{x_i}}{1 + x_i} \tag{5.39}$$

  a disperze této hodnoty je určena vztahem

$$D(V) = \frac{3}{2}\int_0^1 \frac{e^{2x}}{1 + x}\,\mathrm{d}x - (e - 1)^2 = 0.0269\,. \tag{5.40}$$

5.14 Metoda symetrizace integrované funkce

Jelikož daná funkce $f(x) = e^x$ je monotonní, stačí provést jednoduchou symetrizaci. Dostáváme

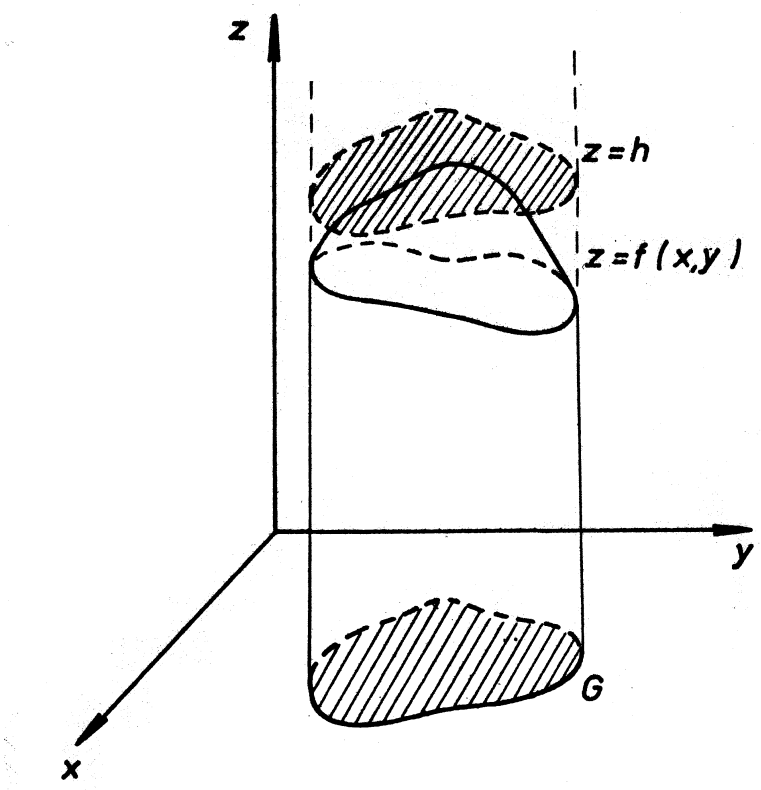
$$I = \frac{1}{2n}\sum_{i=1}^{n}(e^{x_i} + e^{1-x_i}) \tag{5.41}$$

$$D(V) = 0.0039\ . \tag{5.42}$$

Z uvedených příkladů je zřejmé, že popsané metody snižování disperze jsou efektivní a lze s jejich pomocí snížit disperzi a podstatně zkrátit výpočet – v našem případě došlo ke zlepšení disperze vzhledem k nejjednodušší metodě střední hodnoty o jeden až dva řády. Při volbě optimálního algoritmu bychom museli vzít do úvahy i počet potřebných operací a teprve potom rozhodnout o výhodnosti algoritmu. V případě funkce $f(x) = e^x$ by zřejmě zvítězila poslední metoda mající výrazně nejnižší disperzi i jednoduchý algoritmus výpočtu.

Jednotlivé metody výpočtu byly naprogramovány. Pro výpočet jedné hodnoty integrálu bylo použito vždy 100 náhodných čísel a pro každou metodu byl výpočet desetkrát opakován. Z těchto deseti hodnot byl spočten aritmetický průměr a odhadnuta disperze. Na základě těchto hodnot byl určen 90 % interval spolehlivosti. Výsledky výpočtů jsou uvedeny v tab. 1.

U každé metody je v tab. 5.1. uvedeno deset hodnot opakovaného výpočtu hodnoty integrálu (pro různá náhodná čísla), odhadnutá střední hodnota, která představuje vypočtenou hodnotu integrálu, a disperze. Dále je uveden 90 % interval spolehlivosti pro hodnotu integrálu ve tvaru $\bar{x} + \delta$, $\bar{x} - \delta$ a uvedena hodnota δ. Na posledním řádku je uvedena chyba výpočtu. Výsledky potvrzují naše předchozí teoretické úvahy.



Obrázek 5.3: Geometrická metoda výpočtu vícerozměrných integrálů.

standardní metoda				
1.7301	1.5713	1.7013	1.7385	1.7604
1.6622	1.6809	1.7092	1.6333	1.7212
střední hodnota		1.6908	disperze	3.1706×10^{-3}
90 % interval spolehlivosti		1.658	1.723	$\pm\ 3.26\times10^{-2}$
správná hodnota		1.7183	chyba	-2.743×10^{-2}
geometrická metoda				
1.5000	1.7400	1.6500	1.8900	1.5600
1.8600	1.6500	1.5600	1.7700	1.6500
střední hodnota		1.6830	disperze	1.6890×10^{-2}
90 % interval spolehlivosti		1.608	1.7580	$\pm\ 7.53\times10^{-2}$
správná hodnota		1.7183	chyba	-3.528×10^{-2}
metoda hlavní části				
1.7459	1.6979	1.7494	1.7251	1.7343
1.7188	1.7315	1.7085	1.7613	1.7482
střední hodnota		1.7321	disperze	3.9520×10^{-4}
90 % interval spolehlivosti		1.721	1.744	$\pm\ 1.15\times10^{-2}$
správná hodnota		1.7183	chyba	1.3804×10^{-2}
metoda symetrizace				
1.7116	1.7121	1.7115	1.7294	1.7131
1.7111	1.7242	1.7127	1.7119	1.7233
střední hodnota		1.7161	disperze	4.6177×10^{-5}
90 % interval spolehlivosti		1.712	1.720	$\pm\ 3.94\times10^{-3}$
správná hodnota		1.7183	chyba	-2.202×10^{-3}
metoda váženého výběru				
1.7033	1.7078	1.7138	1.7375	1.7397
1.7358	1.7433	1.7214	1.7175	1.6990
střední hodnota		1.7219	disperze	2.6299×10^{-4}
90 % interval spolehlivosti		1.713	1.731	$\pm\ 9.40\times10^{-3}$
správná hodnota		1.7183	chyba	3.6224×10^{-3}

Tabulka 5.1: Výsledky výpočtů integrálu (5.27) různými metodami Monte Carlo. Integrál z funkce $\exp(x)$ od 0 do 1, celkový počet náhodných čísel je 100 (generátor rnd3).

5.15 Vícerozměrné integrály

Naším úkolem je vypočítat integrál

$$I = \int_G f(P)p(P)\,\mathrm{d}P\,, \tag{5.43}$$

kde $p(P)$ je zadaná hustota pravděpodobnosti v oblasti G a P je bod z této oblasti. Nejjednodušší případ odpovídá konstantní hustotě pravděpodobnosti $p(P) = 1/G$.

Většinu výše uvedených metod můžeme zobecnit na případ d-rozměrných integrálů. V metodě střední hodnoty vezmeme náhodné body P_1, P_2 , ... s hustotou $p(P)$ a zavedeme náhodnou veličinu $V = f(P)$, jejíž střední hodnota je I

$$E(V) = \int_G f(P)p(P)\,\mathrm{d}P = I\ . \tag{5.44}$$

Za odhad integrálu vezmeme veličinu

$$I \doteq \frac{1}{n}\sum_{i=1}^{n} V_i = \frac{1}{n}\sum_{i=1}^{n} f(P_i)\ . \tag{5.45}$$

Analogicky můžeme zobecnit i geometrickou metodu. Předpokládejme, že funkce $f(P)$ je v oblasti integrace G omezená, tj. platí $0 \le f(P) \le h$. Obdobně jako v jednodimenzionálním případě si vytvoříme novou $d+1$ rozměrnou oblast $G \times (0, h)$ s objemem Gh a v ní budeme generovat náhodné body q_i (obr.5.3). Pak budeme zkoumat, kolik z těchto n náhodných bodů q_i bude ležet pod povrchem plochy $z = f(P)$ – jejich počet označme n'. Pro integrál I pak dostaneme odhad

$$I \doteq Gh\frac{n'}{n}\ . \tag{5.46}$$

I v případě výpočtu vícerozměrných integrálů je někdy výhodné použít metod snižujících disperzi. Situace je však komplikovanější než v jednorozměrném případě. Na druhé straně však zde vzniká možnost snížit disperzi u těch integrálů, u nichž lze analyticky integrovat přes některé proměnné tím, že tyto integrace provedeme před použitím numerické metody.

5.16 Problémové případy integrace

Často se setkáváme s úlohou spočíst integrál přes neomezenou oblast, např. přes interval $(-\infty, \infty)$. V těchto případech nemůžeme použít metody Monte Carlo přímo, ale musíme se uchýlit k jednomu ze tří obratů:

1. z oblasti G vyřízneme dost velikou konečnou oblast a integrujeme pouze přes ní;
2. přetransformujeme oblast na konečnou;
3. pomocí metody váženého výběru vezmeme hustotu pravděpodobnosti, která dostatečně rychle klesá při přechodu k nekonečnu.

Integrovaná funkce $f(P)$ může mít v oblasti integrace G singularitu, dejme tomu typu $1/x^2$. V tomto případě nemůžeme použít obvyklých odhadů integrálu, protože disperze aproximující náhodné veličiny V bude nekonečná. I zde však můžeme užít metodu váženého výběru a zvolit hustotu pravděpodobnosti též typu $1/x^2$. Výpočet se sice tímto obratem zpomalí, ale dospějeme ke správnému výsledku.

Kapitola 6

Interpolace funkcí mnoha proměnných

Následující úloha patří též mezi problémy, ve kterých je použití metody Monte Carlo při větším počtu proměnných mnohem účinnější než klasické metody.

Mějme funkci $f(x_1, \ldots, x_m)$ o m proměnných a nechť jsou zadány její hodnoty pouze v rozích jednotkové m–rozměrné krychle. Naším úkolem je stanovit hodnotu této funkce v nějakém bodě o souřadnicích $(x_1, \ldots, x_m)$ uvnitř krychle. Podle každé souřadnice budeme interpolovat lineárně. Interpolační vztahy pro jedno a dvoudimenzionální případ jsou

$$\begin{aligned} m=1 \quad f(x_1) &= (1-x_1)\, f(0) + x_1\, f(1) && (6.1)\\ m=2 \quad f(x_1,x_2) &= (1-x_1)\,(1-x_2)\, f(0,0) + x_1(1-x_2)\, f(1,0) + && (6.2)\\ &+ (1-x_1)\, x_2\, f(0,1) + x_1 x_2\, f(1,1)\ . \end{aligned}$$

Interpolační vztah (6.2) můžeme zapsat v kompaktní formě

$$f(x_1,x_2) = \sum_{k_1,k_2=0}^{1} c(k_1)c(k_2)f(k_1,k_2)\ , \tag{6.3}$$

kde

$$c(k_1) = \begin{cases} 1-x_i & \text{pro} \quad k_i = 0\ , \\ x_i & \text{pro} \quad k_i = 1\ , \end{cases} \tag{6.4}$$

přičemž pro vrcholy ($x_i = 0,\ 1$) jsme zavedli index k_i . Stejným způsobem můžeme zapsat interpolační vztah pro případ m proměnných

$$f(x_1,\ldots,x_m) = \sum_{k_1,\ldots,k_m=0}^{1} c(k_1)\ldots c(k_m)f(k_1,\ldots,k_m)\ . \tag{6.5}$$

Výpočet vztahů (6.4) a (6.5) je v principu jednoduchý. S růstem m však počet členů v interpolačním vztahu rychle vzrůstá a např. pro $m = 30$ dosahuje hodnoty větší než 10^9 [10].

Naproti tomu algoritmus řešení téže úlohy metodou Monte Carlo je velmi jednoduchý a časově méně náročný. Vytvořme m nezávislých náhodných veličin $X_1, \ldots, X_m$ a nechť každá z nich nabývá hodnoty 0 nebo 1 s pravděpodobnostmi

$$\begin{aligned} P(X_i = 0) &= 1 - x_i\ , \\ P(X_i = 1) &= x_i \qquad i = 1,\ldots,m\ . \end{aligned} \tag{6.6}$$

Každý bod $(X_1, \ldots, X_m)$ představuje tedy některý vrchol jednotkové krychle. Hodnoty X_i určíme pomocí náhodného čísla y z $R(0, 1)$ podle předpisu

$$\begin{aligned} X_i &= 0 \quad \text{pro} \quad y_i > x_i \\ X_i &= 1 \quad \text{pro} \quad y_i < x_i \, . \end{aligned} \tag{6.7}$$

Věta 12 *Střední hodnota veličiny $f(X_1, \ldots, X_m)$ je*

$$E(f(X_1, \ldots, X_m)) = f(x_1, \ldots, x_m) \, . \tag{6.8}$$

Důkaz:
Ze vztahů (6.4) a (6.6) je vidět, že platí $P(X_i = k_i) = c(k_i)$. Spočteme-li nyní střední hodnotu veličiny $f(X_1, \ldots, X_m)$

$$E(f(X_1, \ldots, x_m)) = \sum_{k_1, \ldots, k_m = 0}^{1} f(k_1, \ldots, k_m) P(X_1 = k_1, \ldots, X_m = k_m) \, , \tag{6.9}$$

vidíme, že pravá strana tohoto vztahu přechází právě v pravou stranu vztahu (6.5), odkud okamžitě vyplývá tvrzení věty.

Máme-li interpolovat funkci $f(x_1, \ldots, x_m)$ o m proměnných v nějakém bodě $(\bar{x_1}, \ldots, \bar{x_m})$ uvnitř krychle, postupujeme v metodě Monte Carlo následujícím způsobem. Nejprve vytvoříme n bodů o náhodných souřadnicích $(x_1^{(1)}, \ldots, x_m^{(j)})$, $j = 1, \ldots, n$, přičemž každá z nich nabývá hodnoty 0 nebo 1 s pravděpodobnostmi (6.6). Pomocí těchto náhodných vrcholů určíme hodnotu funkce f v zadaném bodě

$$f(\bar{x_1}, \ldots, \bar{x_m}) \doteq \frac{1}{n} \sum_{j=1}^{n} f(x_1^{(j)}, \ldots, x_m^{(j)}) \, . \tag{6.10}$$

Tímto způsobem můžeme provést interpolaci funkce $f(x_1, \ldots, x_m)$ v několika bodech současně při použití stejné posloupnosti uspořádaných m–tic rovnoměrně rozdělených náhodných čísel $y_1, y_2, y_3, \ldots, y_m$.

Jestliže potřebujeme interpolovat v jiných bodech než v bodech v jednotkové krychli nebo máme zadány hodnoty interpolované funkce v jiných bodech než ve vrcholech jednotkové krychle, potom vhodnou transformací souřadnic převedeme tento problém opět na interpolaci v jednotkové krychli.

Kapitola 7

Inverze matic a řešení soustav lineárních algebraických rovnic

Značná část výpočetních postupů metody Monte Carlo spočívá v převedení některých typů úloh (řešení soustavy lineárních rovnic, inverze matic, řešení diferenciálních rovnic s okrajovými podmínkami, určení vlastních čísel operátorů, řešení integrálních rovnic apod.) na problém odhadu charakteristik náhodných veličin, jejichž hodnoty závisí na posloupnostech stavů speciálně vytvořených Markovových řetězců. Místo stavové terminologie Markovových řetězců je pro některé případy názornější použití terminologie náhodných procházek po množině bodů.

Ve většině případů se používá řetězců, v nichž se vyskytují absorpční stavy, tj. stavy, z nichž není možný východ - pravděpodobnost setrvání v absorpčním stavu je jednotková. Dalším typem stavů, které se v řetězcích objevují, jsou tzv. transientní stavy, které jsou po určitém počtu kroků děje opuštěny bez ohledu na výchozí stav.

7.0.1 Metoda řešení soustavy lineárních rovnic souvisejících s metodou jednoduchých iterací

Cílem tohoto a následujícího paragrafu je ukázat pouze dva způsoby řešení soustavy lineárních rovnic [6] pomocí modelů náhodných procesů spojených s řešením soustav lineárních algebraických rovnic.

Zde uvedené modely se hodí pouze pro speciální případ matic, které jsou „blízké" k jednotkové matici ve smyslu, jenž bude objasněn dále. Tato třída matic je shodná s třídou matic, pro které lze odpovídající soustavu lineárních rovnic řešit metodou jednoduchých iterací. Model se konstruuje podle principu markovského procesu končícího s pravděpodobností 1 po konečném počtu kroků.

Uvažujme systém lineárních rovnic, zapsaný ve vektorovém tvaru

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \; . \tag{7.1}$$

Zde $\boldsymbol{A}$ je n–rozměrná čtvercová matice, $\boldsymbol{x}$ a $\boldsymbol{b}$ jsou n–rozměrné vektory. Nechť lze soustavu řešit metodou jednoduchých iterací, tj. matici A lze vyjádřit ve tvaru

$$\boldsymbol{A} = \boldsymbol{E} - \boldsymbol{B} \; , \tag{7.2}$$

kde $\boldsymbol{E}$ je jednotková matice a vlastní čísla matice $\boldsymbol{B}$ mají absolutní hodnotu menší než 1.

Soustava (7.1) je ekvivalentní se soustavou

$$\boldsymbol{x} = \boldsymbol{B}\boldsymbol{x} + \boldsymbol{b} \; . \tag{7.3}$$

Řešení soustavy (7.1) lze vyjádřit ve tvaru

$$\boldsymbol{x} = \boldsymbol{A^{-1}}\boldsymbol{b} \; . \tag{7.4}$$

Za uvedených předpokladů inverzní matice může být vyjádřena řadou

$$\boldsymbol{A^{-1}} = \boldsymbol{E} + \boldsymbol{B} + \boldsymbol{B^2} + \ldots + \boldsymbol{B^n} + \ldots \; . \tag{7.5}$$

Tato řada konverguje právě tehdy, jestliže vlastní čísla matice $\boldsymbol{B}$ mají absolutní hodnotu menší než jedna, tj.

$$|\lambda(\boldsymbol{B})| < 1 \; . \tag{7.6}$$

Dosadíme-li nyní výraz (7.5) do (7.4), dostaneme řešení ve tvaru řady

$$\boldsymbol{x} = \boldsymbol{b} + \boldsymbol{B}\boldsymbol{b} + \boldsymbol{B^2}\boldsymbol{b} + \ldots + \boldsymbol{B^n}\boldsymbol{b} + \ldots \; . \tag{7.7}$$

Částečné součty této řady můžeme získat známým iteračním postupem, klademe-li postupně

$$\begin{aligned}
\boldsymbol{x}^{(1)} &= \boldsymbol{b} \; , \\
\boldsymbol{x}^{(2)} &= \boldsymbol{B}\boldsymbol{x}^{(1)} + \boldsymbol{b} = \boldsymbol{B}\boldsymbol{b} + \boldsymbol{b} \; , \\
\ldots \;\; &\ldots \;\; \ldots \\
\boldsymbol{x}^{(k)} &= \boldsymbol{B}\boldsymbol{x}^{(k-1)} + \boldsymbol{b} = \boldsymbol{B}^{(k-1)}\boldsymbol{b} + \boldsymbol{B}^{(k-2)}\boldsymbol{b} + ... + \boldsymbol{B}\boldsymbol{b} + \boldsymbol{b} \; . \\
\ldots \;\; &\ldots \;\; \ldots
\end{aligned} \tag{7.8}$$

Konvergence metody jednoduchých iterací je ekvivalentní s konvergencí posloupnosti $\boldsymbol{x}^{(1)}$, $\boldsymbol{x}^{(2)}$, …, $\boldsymbol{x}^{(k)}$, …. Pro výpočet $\boldsymbol{x}$ přejdeme k souřadnicovému zápisu. Prvky matice B budeme označovat B_{ij}. Potom m – tá souřadnice x_m vektoru $\boldsymbol{x}$ je rovna

$$x_m = b_m + \sum_i B_{mi} b_i + \sum_{i_1,i_2} B_{mi_1} B_{i_1 i_2} b_{i_2} + \ldots + \sum_{i_1,i_2}, \ldots, i_r B_{mi_1} B_{i_1 i_2} \ldots B_{i_{r-1} i_r} b_r + \ldots \; . \tag{7.9}$$

Jde o to, abychom našli metodu pro výpočet součtu (7.9). Uvažujme nejprve případ, kdy prvky matice jsou nezáporné a součet prvků v každém řádku je roven jedné, tj.

$$\sum_{i=1}^{n} B_{mi} = 1 \; . \tag{7.10}$$

Potom lze veličiny B_{mi} vyšetřovat jako soubor pravděpodobností pro úplnou soustavu vzájemně se vylučujících jevů.

Tyto jevy můžeme vyšetřovat, jako by byly spojené s některou situací výběru koulí z n uren. Předpokládejme, že v každé urně jsou koule n různých druhů. Přitom nechť pravděpodobnost vytažení koule i–tého druhu z m–té urny je rovna B_{mi}.

Vytáhněme náhodně kouli z m–té urny a definujme náhodnou veličinu Z_m tím způsobem, že za její hodnotu zvolíme číslo b_i, jestliže byla vytažena koule i–tého druhu. Střední hodnota náhodné veličiny Z_m je zřejmě rovna

$$E(Z_m) = \sum_{i=1}^{n} B_{mi} b_i \; , \tag{7.11}$$

tj. je rovna druhému sčítanci v řadě (7.9). Snažme se nyní získat náhodnou veličinu V_m , jejíž střední hodnota je rovna třetímu sčítanci řady (7.9). Za tím účelem nejdříve vytáhneme kouli z m–té urny. Je-li vytažena koule i_1–tého druhu, vytáhneme další kouli z i_1–té urny. Je-li tato koule i_2–tého druhu, položíme hodnotu veličiny V_m rovnou b_{i_2}.

Je jasné, že pravděpodobnost, s níž náhodná veličina V_m nabývá hodnoty b_{i_2}, je rovna

$$\sum_{i_1=1}^{n} B_{mi_1} B_{i_1 i_2} \ . \tag{7.12}$$

Předpokládáme, že tahy koulí z jednotlivých uren jsou na sobě nezávislé. Střední hodnota veličiny V je zřejmě rovna

$$E(V_m) = \sum_{i_1,i_2} B_{mi_1} B_{i_1 i_2} b_{i_2} \ . \tag{7.13}$$

Vyšetřovaná konstrukce připouští jednoduché zobecnění, které umožňuje získat náhodnou veličinu X_m, jejíž střední hodnota je rovna součtu řady

$$E(X_m) = x_m \ . \tag{7.14}$$

Modelujeme-li nyní proces získání náhodné veličiny X_m N–krát a dostaneme-li při tom posloupnost hodnot x_m^1, x_m^2, …, x_m^N, můžeme za přibližnou hodnotu hledané veličiny x_m vzít aritmetický průměr

$$\bar{x_m} = \frac{x_m^1 + x_m^2 + \ldots + x_m^N}{N} \ . \tag{7.15}$$

Veličinu X_m zkonstruujeme následujícím způsobem. Každý prvek matice $\boldsymbol{B}$ zapíšeme jako součin dvou prvků

$$B_{mj} = F_{mj} P_{mj} \ , \tag{7.16}$$

kde $P_{mj} \in \langle 0, 1)$; pro $B_{mj} = 0$ položíme $P_{mj} = 0$. Analogicky upravíme i prvky vektoru pravé strany $\boldsymbol{b}$

$$b_m = f_m p_m \ , \tag{7.17}$$

kde $p_m \in (0, 1)$. Volbu prvků $\boldsymbol{P}$ a $\boldsymbol{p}$ provedeme tak, aby platilo

$$p_m + \sum P_{mj} = 1 \ . \tag{7.18}$$

Rovnici (7.9) můžeme nyní přepsat do tvaru

$$x_m = f_m p_m + \sum_i F_{mi} f_i P_{mi} p_i + \sum_{i_1,i_2,\ldots,i_r} F_{mi_1} F_{i_1 i_2} \ldots F_{i_{r-1} i_r} f_{i_r} P_{mi_1} P_{i_1 i_2} \ldots P_{i_{r-1} i_r} p_{i_r} + \ldots \ . \tag{7.19}$$

Předpokládejme, že máme n uren a v každé z nich leží koule $(n+1)$ druhů. Pravděpodobnost, že z k–té urny vytáhneme kouli i–tého druhu ($i \leq n$), je rovna P_k; pravděpodobnost vytažení koule $(n+1)$–ho druhu z k–té urny je rovna p_k.

Nejprve vytáhneme kouli z m–té urny. Vytáhneme-li kouli druhu i_1 ($i_1 \leq n$), je tím určeno, že další kouli je třeba vytáhnout z i–té urny.

Je-li vytažena koule $(n+1)$–ho druhu, nevytahujeme již žádnou kouli a v tomto případě za hodnotu náhodné veličiny X_m vezmeme číslo f_m. Jestliže koule $(n+1)$–ho druhu byla vytažena z i–té urny, když předtím byly postupně vytaženy koule z uren s pořadovými čísly m, i_1, i_2, …, i_r, klademe hodnotu veličiny X_m rovnou $F_{mi_1} F_{i_1 i_2} \ldots F_{i_{r-1} i_r} f_{i_r}$. Pravděpodobnost toho, že náhodná veličina nabude dané hodnoty, je rovna

$$W_{mi_1 i_2 \ldots i_r} = P_{mi_1} P_{i_1 i_2} \ldots P_{i_{r-1} i_r} p_{i_r} \ . \tag{7.20}$$

Střední hodnota náhodné veličiny X_m je rovna

$$E(X_m) = W_{mi_1i_2\ldots i_r} F_{mi_1} F_{i_1i_2} \ldots F_{i_{r-1}i_r} f_{i_r} \ . \tag{7.21}$$

Dosadíme-li (7.20) do (7.21) a srovnáme-li výsledek s (7.19), dostaneme

$$x_m = E(X_m) \ . \tag{7.22}$$

Odtud tedy vyplývá, že modelování náhodné veličiny X_m umožňuje získat s určitou pravděpodobností řešení soustavy lineárních algebraických rovnic. Existují i jiné procesy Monte Carlo takovéhoto druhu, kde statistický model je vhodný pro nalezení inverzní matice nebo pro řešení soustavy s obecnější maticí $\boldsymbol{A}$.

Výhoda vyložené metody řešení je v tom, že při standardní metodě řešení soustav lineárních rovnic je třeba pro výpočet jedné neznámé určit i všechny ostatní, zatímco v uvedené metodě určíme pouze jedinou neznámou x, aniž musíme současně počítat všechny ostatní neznámé. Důsledkem je, že doba výpočtu je úměrná počtu rovnic n a ne jejich kvadrátu n^2 jako v jiných metodách. Pokud však musíme určit všechny neznámé dané soustavy rovnic, dáme přednost některé ze standardních metod, neboť výpočet metodou Monte Carlo je při požadavku dostatečné přesnosti dosti pomalý.

7.0.2 Druhý pravděpodobnostní model pro řešení soustavy lineárních algebraických rovnic

Nyní se budeme zabývat jinou metodou řešení lineárních soustav, vhodnou pro stejnou třídu úloh. Zkoumejme tuto metodu pro případ nalezení inverzní matice $\boldsymbol{A}^{-1}$. Vyjádřeme prvky matice $\boldsymbol{B} = \boldsymbol{E} - \boldsymbol{A}$ ve tvaru

$$B_{ij} = v_{ij} W_{ij} \ , \tag{7.23}$$

kde

$$\sum_{j=1}^{n} W_{ij} = 1 \quad \text{a} \quad 0 < W_{ij} < 1 \ .$$

Uvažujme Markovský proces, který představuje soustavu mající $n+1$ stavů: ω_1, ω_2 , $\ldots$, ω_{n+i}, přičemž pravděpodobnosti přechodu p jsou definovány podmínkou

$$p = \begin{cases} W \ , & \text{jestliže} \quad i \le n, j \le n, i \ne k \ , \\ 0 \ , & \text{jestliže} \quad i = n+1, j \le n \ , \\ 0 \ , & \text{jestliže} \quad i \le n, j = n+1, i \ne k \ , \\ 1 \ , & \text{jestliže} \quad i = j = n+1 \ , \\ W \ , & \text{jestliže} \quad i = k, j \le n \ , \\ 1-\alpha \ , & \text{jestliže} \quad i = k, j = n+1 \ . \end{cases} \tag{7.24}$$

Hodnotu parametru α podrobíme pouze podmínce

$$0 < \alpha < 1 \ . \tag{7.25}$$

V každém konkrétním případě hodnota parametru α může být zvolena tak, aby doba potřebná k výpočtu byla minimální.

Snadno nahlédneme, že stav ω_{n+1} je absorpčním stavem vyšetřovaného Markovského procesu a že soustava Ω přejde s pravděpodobností 1 po konečném počtu kroků do stavu ω_{n+1} . Popsaný Markovský proces je speciálně předurčen pro výpočet k–tého sloupce matice $\boldsymbol{A}$.

Z definice našeho Markovského procesu plyne, že jeho průběh je s pravděpodobností 1 popsán takovýmto schématem:

$$\omega_1 \longrightarrow \omega_{i_1} \longrightarrow \omega_{i_2} \longrightarrow \ldots \longrightarrow \omega_k \longrightarrow \omega_{n+i} \ , \tag{7.26}$$

tj. předtím, než přejde do absorpčního stavu, musí soustava projít k–tým stavem. Abychom vypočítali prvek g_{lk} matice $\boldsymbol{G} = \boldsymbol{A}^{-1}$, vezmeme za počáteční stav procesu stav ω_l.

Podle obecného schématu definujeme náhodnou veličinu $X^{(l,k)}$, která souvisí s vyšetřovaným Markovským procesem a je funkcí jeho průběhu. K tomu položíme

$$X^{(l,k)} = X(l, i_1, i_2, \ldots, i_\nu, k) = u_{li_1} u_{i_1 i_2} \ldots u_{i_\nu k} (1-\alpha)^{-1} \ , \tag{7.27}$$

když průběh procesu je popsán schématem (7.26). Přitom jsme označili

$$u = \begin{cases} v \ , & i \neq k \\ (1/\alpha) v_{ij} \ , & i = k \ . \end{cases} \tag{7.28}$$

Je-li průběh procesu popsán schématem $\omega_l \longrightarrow \omega_{n+1}$, položíme $X = (1-\alpha)^{-1}$.

Vypočteme nyní střední hodnotu této náhodné veličiny:

$$\begin{aligned} E(X^{(l,k)}) &= \delta_{lk}(1-\alpha)(1-\alpha)^{-1} + \\ &+ \sum_{i_1, i_2, \ldots, i_\nu} u_{li_1} u_{i_1 i_2} \ldots u_{i_\nu k} (1-\alpha)^{-1} p_{li_1} p_{i_1 i_2} \ldots p_{i_\nu k} (1-\alpha) = \\ &= \delta_{lk} + \sum b_{li_1} b_{i_1 i_2} \ldots b_{i_\nu k} \ . \end{aligned} \tag{7.29}$$

Srovnáme-li tento vztah se vztahem (7.9), vidíme,že

$$E(X^{(l,k)}) = g_{lk} \ . \tag{7.30}$$

Zvolíme-li totiž ve vztahu (7.1) za $\boldsymbol{b}$ vektor, jehož k–tá souřadnice je rovna 1 a všechny ostatní jsou 0, potom součin $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$ je roven k–témů sloupci matice A, tudíž l–tá souřadnice vektoru $\boldsymbol{x}$ udává prvek g_{lk} matice $\boldsymbol{G} = \boldsymbol{A}^{-1}$. Tímto způsobem jsme tedy získali metodu pro výpočet prvků inverzní matice.

Vypočtěme rozptyl náhodné veličiny X:

$$\begin{aligned} D(X) &= E(X^2) - (E(X))^2 = \delta_{lk}(1-\alpha)^{-1} + \\ &+ \sum u_{li_1} u_{i_1 i_2} \ldots u_{i_\nu k} (1-\alpha)^{-1} b_{li_1} b_{i_1 i_2} \ldots b_{i_\nu k} - g_{lk}^2 \quad . \end{aligned} \tag{7.31}$$

Není rovněž těžké napsat výraz pro střední dobu přechodu systému do absorpčního stavu

$$\tau = \delta_{lk}(1-\alpha) + \sum_{i_1, i_2, \ldots, i_\nu} (\nu+2) p_{li_1} p_{i_1 i_2} \ldots p_{i_\nu k} (1-\alpha) \ . \tag{7.32}$$

Volbu vyjádření matice $\boldsymbol{B}$ ve tvaru (7.23) a volbu parametru je třeba provést tak, aby při dané přesnosti byla doba výpočtu minimální, tj. aby byl minimální součin $\tau D(x)$.

Některé další způsoby řešení lineárních algebraických rovnic lze nalézt např. v [10].

Kapitola 8

Řešení Laplaceovy rovnice

Simulace náhodných procesů se dá s výhodou použít při řešení mnoha klasických rovnic matematické fyziky [11]. Přístup metody Monte Carlo k řešení těchto problémů si budeme demonstrovat na úloze stanovení elektrostatického potenciálu v zadané oblasti, jestliže známe jeho hodnotu na hranici oblasti. Máme tedy řešit Dirichletovu úlohu

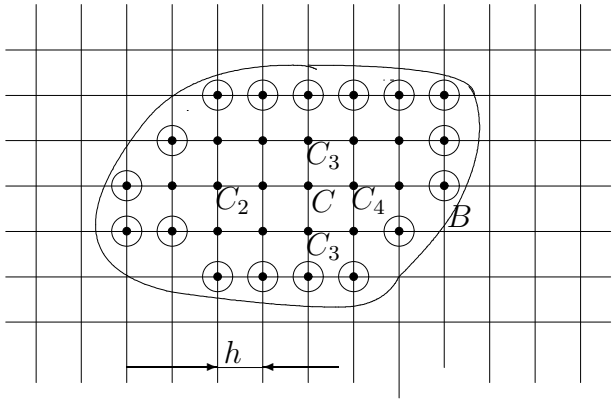
$$\Delta V = 0 \,, \text{uvnitř oblasti } G \,, \tag{8.1}$$

$$V = f(B) \text{ na hranici oblasti } G \,. \tag{8.2}$$

8.1 Algoritmus bloudění v pravoúhlé síti

V roce 1944 Kakutani poprvé poukázal na vztah mezi brownovským pohybem a teorií potenciálu. Spojení mezi těmito dvěma teoriemi nyní v krátkosti uvedeme.

Při řešení Dirichletovy úlohy budeme postupovat obvyklým způsobem podobně jako při klasických metodách numerické matematiky [8]. Nejdříve si v dané oblasti G zavedeme pravoúhlou síť (dvoudimenzionální případ) s krokem h (viz. obr. 8.1) a převedeme diferenciální rovnici (8.1) na rovnici diferenční. V hraničních bodech B oblasti G (znázorněných



Obrázek 8.1: Pravoúhlá síť pro řešení Laplaceovy rovnice

kroužky v obr. 8.1) přiřadíme hledané funkci V hodnoty $V(B) = f(B)$. Ve vnitřních bodech C oblasti G stanovíme hodnotu potenciálu $V(C)$ na základě znalosti potenciálu v

sousedních bodech C_1, C_2, C_3, C_4 pomocí čtyřbodové aproximace

$$V(C) = 1/4(V(C_1) + V(C_2) + V(C_3) + V(C_4)) \ . \tag{8.3}$$

Studujme nyní náhodný proces bloudění v pravoúhlé síti. Vyjdeme z nějakého vnitřního bodu C a jednu ze čtyř možných cest vedoucích z daného bodu volíme se stejnou pravděpodobností, tj. s pravděpodobností $1/4$. Ptejme se, jaká je pravděpodobnost $W(C, B)$ toho, že po vypuštění z bodu C dorazíme po náhodné trajektorii do hraničního bodu B, kde již zůstaneme. Protože po vypuštění z bodu C se hned prvním krokem dostaneme do jednoho ze čtyř sousedních bodů C, musí pro pravděpodobnost platit

$$W(C, B) = 1/4 \sum_{i=1}^{4} W(C_i, B) \ , \tag{8.4}$$

což je jiný tvar diferenční rovnice (8.3). Pro hraniční body B položíme podmínku

$$W(B, B') = \begin{cases} 1 & \text{pro} \quad B = B' \ , \\ 0 & \text{pro} \quad B \neq B' \ . \end{cases} \tag{8.5}$$

Pokud nyní celý proces bloudění n–krát nasimulujeme na počítači a zaznamenáme počet pokusů n', při nichž jsme se dostali z daného bodu C právě do hraničního bodu B, pro pravděpodobnost $W(C, B)$ dostaneme odhad

$$W(C, B) \doteq n'/n \ . \tag{8.6}$$

Zbývá nám ještě zahrnout do řešení vliv hraniční podmínky (8.2). Tohoto dosáhneme následujícím způsobem. Každému bodu C přiřadíme náhodnou veličinu $U(C)$, která nabývá jedné z hodnot $f(B_1), \ldots, f(B_m)$, kde m je celkový počet hraničních bodů, s pravděpodobnostmi $W(C, B_i)$. Střední hodnota náhodné veličiny $U(C)$ je tedy

$$V(C) = E(U(C)) = \sum_{i=1}^{m} f(B_i) W(C, B_i) \ . \tag{8.7}$$

Funkce $V(C)$ opět vyhovuje diferenční rovnici (8.4) a současně i hraničním podmínkám

$$V(B) = \sum_{i=1}^{m} f(B_i) W(B, B_i) = f(B) \ , \tag{8.8}$$

proto je jediným řešením Dirichletovy úlohy.

Chceme-li tedy určit potenciál v nějakém bodě P, postupujeme následujícím způsobem. Několikrát simulujeme bloudění v pravoúhlé síti, přičemž výchozím bodem je bod P. Až protneme hraniční oblast v bodě B, přiřadíme náhodné veličině U hodnotu $f(B)$. Hledaný potenciál je potom roven střední hodnotě veličiny U.

Jako testovací příklad byla řešena Laplaceova rovnice

$$\Delta u = 0 \tag{8.9}$$

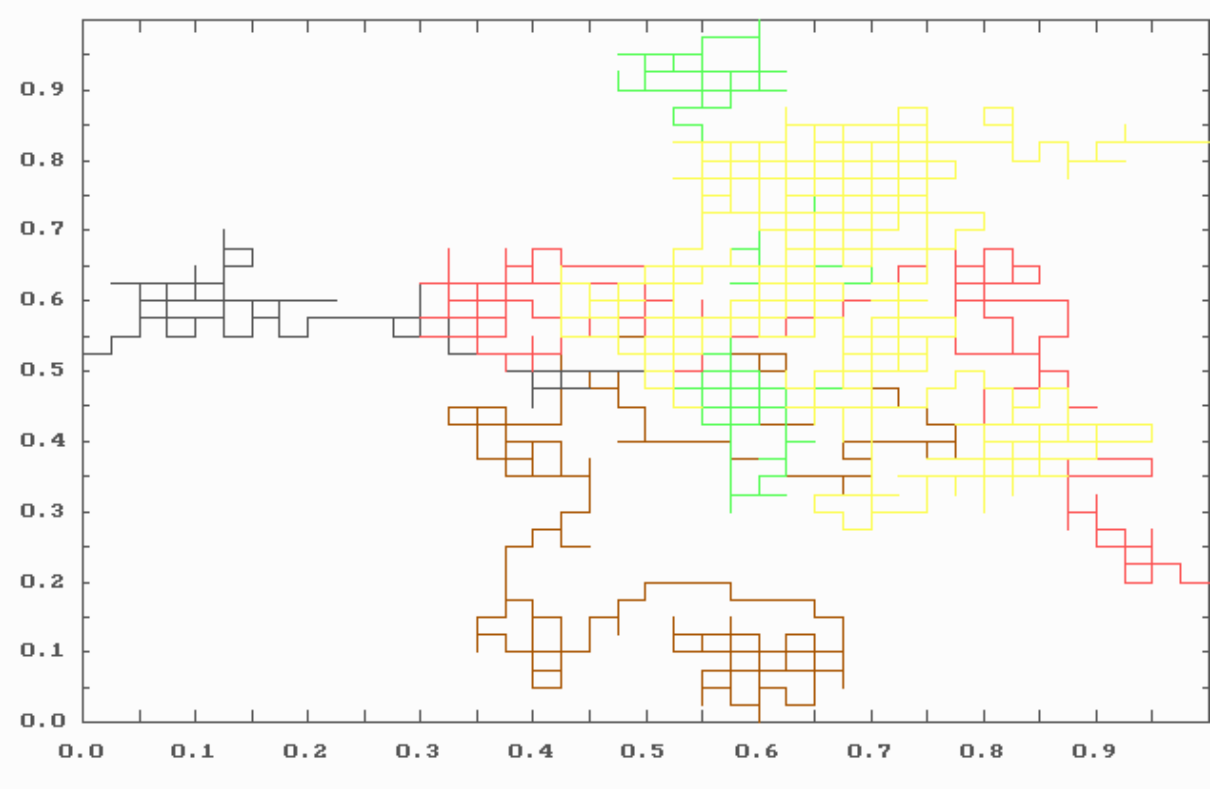
na čtverci [0,1]×[0, 1] s následujícími okrajovými podmínkami

$$\begin{aligned} u(x, 0) &= e^{-2x} \ , \quad x \in [0, 1] \\ u(x, 1) &= e^{-2x} \cos 2 \\ u(0, y) &= \cos 2y \quad y \in [0, 1] \\ u(1, y) &= e^{-2} \cos 2y \ . \end{aligned} \tag{8.10}$$

Analytické řešení této úlohy má tvar

$$u(x, y) = e^{-2x} \cos 2y \ . \tag{8.11}$$

Čtverec jsme pokryli pravoúhlou sítí s krokem $h = 1/40$ a počítali jsme hodnotu funkce $u(x, y)$ ve středu čtverce. Trajektorie pěti náhodných bloudění jsou zobrazeny na obr. 8.2.



Obrázek 8.2: Trajektorie pěti náhodných bloudění při řešení úlohy (8.9).

Pro 10 náhodných bloudění jsme dostali hodnotu $u(0.5, 0.5) = 0.1397$, pro 100 náhodných bloudění hodnotu 0.1934 a pro 1000 náhodných bloudění hodnotu 0.2026, přičemž správná hodnota je $u(0.5, 0.5) = 0.19877$. Je zřejmé, že při konkrétních výpočtech by bylo nutné volit počty náhodných bloudění vyšší, výpočty několikrát opakovat a tak určit střední hodnotu a disperzi výsledků.

Jestliže bychom chtěli řešit Laplaceovu rovnici s hraniční podmínkou

$$\frac{\partial V}{\partial n} = kV + f(B) \ , \tag{8.12}$$

potom se shora uvedený algoritmus bloudění změní pouze v tom, že po dosažení hraničního bodu dovolíme s určitou pravděpodobností další bloudění sítí.

8.2 Algoritmus hledání s náhodným krokem

Nevýhodou algoritmu bloudění v pravoúhlé síti je malá délka každého kroku h. Tato nevýhoda je odstraněna v následující metodě, kterou navrhl Müller a je označována jako

technika maximálních sfér. Tento algoritmus též umožňuje výpočet potenciálu i v případě, že oblast G je složena z několika dielektrik.

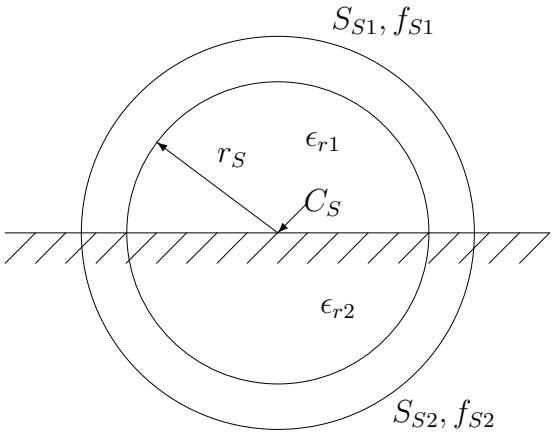
Uveďme si nejprve teoretický základ této metody. Jestliže oblast uzavřená uvnitř koule je homogenní a neobsahuje žádné volné náboje, potom střední hodnota potenciálu na povrchu koule je rovna potenciálu ve středu koule

$$V(C_S) = \frac{1}{4\pi r_S^2} \oint_{S_2} V \,\mathrm{d}S \,. \tag{8.13}$$

Pokud se jedná pouze o dvoudimenzionální pole, platí

$$V(C_S) = \frac{1}{2\pi r_S} \oint_{l_S} V \,\mathrm{d}l \,, \tag{8.14}$$

kde integrujeme po uzavřené křivce l_S. Rovnice (8.10) a (8.11) vyplývají z Laplaceovy rovnice.



Obrázek 8.3: Konfigurace k výpočtu střední hodnoty - rovnice (8.15) a (8.16).

Jestliže střed koule C_S leží na rozhraní dvou dielektrik (obr. 8.3.) a je-li toto rozhraní rovinné (přímka) uvnitř uvažované koule S_S, potom lze odvodit pro hodnotu $V(C_S)$ následující vztahy

$$V(C_S) = \frac{\varepsilon_{r_1}}{\varepsilon_{r_1} + \varepsilon_{r_2}} \left[\frac{1}{2\pi r_S^2} \int\limits_{S_{S_1}} V \,\mathrm{d}S \right] + \frac{\varepsilon_{r_2}}{\varepsilon_{r_1} + \varepsilon_{r_2}} \left[\frac{1}{2\pi r_S^2} \int\limits_{S_{S_2}} V \,\mathrm{d}S \right] \,, \tag{8.15}$$

$$V(C_S) = \frac{\varepsilon_{r_1}}{\varepsilon_{r_1} + \varepsilon_{r_2}} \left[\frac{1}{\pi r_S} \int\limits_{l_{S_1}} V \,\mathrm{d}l \right] + \frac{\varepsilon_{r_2}}{\varepsilon_{r_1} + \varepsilon_{r_2}} \left[\frac{1}{\pi r_S} \int\limits_{l_{S_2}} V \,\mathrm{d}l \right] \,, \tag{8.16}$$

kde $S_{S_1}(l_{S_1})$ je část plochy $S_S(l_S)$ ležící v oblasti s $\varepsilon_r = \varepsilon_{r_1}$, $S_{S_2}(l_{S_2})$ je část plochy $S_S(l_S)$ ležící v oblasti s $\varepsilon_r = \varepsilon_{r_2}$.

Je třeba zdůraznit následující skutečnosti, které se odrazí i v algoritmu výpočtu:

1. Vztahy (8.13) a (8.14) platí vždy bez ohledu na velikost poloměru koule r_S, jestliže se nacházíme v homogenním prostředí.

2. Vztahy (8.15) a (8.16) platí pouze tehdy, když rozhraní mezi dielektriky je rovinné. Není-li rozhraní rovinné, musíme zvolit poloměr r_S dostatečně malý, aby v rámci této kulové oblasti bylo rozhraní alespoň přibližně rovinné.

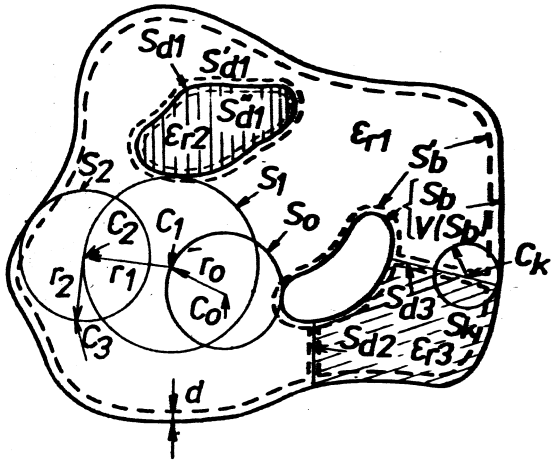
Uveďme nyní algoritmus výpočtu potenciálu uvnitř oblasti ohraničené plochou S_b na obr.8.4. Odhad potenciálu $V(C_0)$ v bodě C_0 obdržíme provedením velkého počtu experimentů náhodného bloudění s proměnnou délkou kroku a náhodným směrem. Náhodné bloudění v každém experimentu začíná z bodu C_0 a je ukončeno, když se dostaneme do dostatečné blízkosti plochy S_b. Obyčejně zvolíme $d > 0$ dostatečně malé a náhodné bloudění ukončíme, když vzdálenost bodu C_k od plochy S_b bude menší než d.

Hodnotu potenciálu v nejbližším bodě plochy S_b si označme $V_i(S_b)$ (i indexuje jednotlivé náhodné bloudění). Simulujme celkem n takových bloudění. Potenciál v bodě C_0 pak je přibližně roven

$$V(C_0) \doteq V_C(C_0) = \sum_{i=1}^{n} [V_i(S_b)]/n \; . \tag{8.17}$$

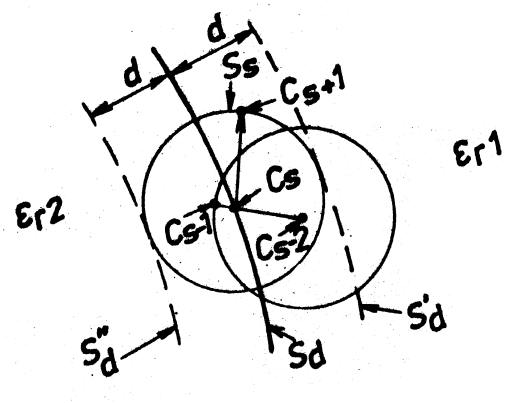
Na obr. 8.4.je znázorněna uvažovaná situace a je použito následujícího označení:

- S'_b je plocha uzavřená hraniční plochou S_b a ležící v její blízkosti,
- S'_d a S''_d jsou plochy ležící v blízkosti a na opačných stranách plochy S_d,
- d je normálová vzdálenost mezi S_b a S'_b , S_d a S'_d,
- S_d a S''_d (d je malé vzhledem k ostatním dimenzím),
- ε_r je relativní permitivita.



Obrázek 8.4: Konstrukce trajektorie pro řešení Laplaceovy rovnice v oblasti G při použití bloudění s náhodným krokem.

Uvažujme situaci, že jsme se během bloudění dostali do bodu C_S. Následující pozice bodu (C_{S+1}) leží na sféře S_S - bod C_S je středem této sféry. Jak výpočet polohy C_{S+1} na sféře S_S, tak i výpočet poloměru r_S závisí na poloze bodu C_S.



Obrázek 8.5: Konstrukce trajektorie v případě, kdy bod (C_{S-1}) přechází do druhého dielektrika.

Jestliže bod C_S neleží na rozhraní dvou dielektrik, potom C_{S+1} je vybrán náhodně a rovnoměrně na ploše S_S. Jestliže však C_S leží na rozhraní dielektrik, bod C_{S+1} je vybrán následujícím způsobem - viz. obr. 8.5. Bod C_S vznikl přenesením bodu C_{S-1} (přešel hranici dielektrik) po přímce spojující body C_{S-2} a C_{S-1} do průsečíku této přímky s rozhraním dielektrik. Z bodu C_{S-1} se náhodný krok nedělá. Následující pozice (bod C_{S+1} na S_S) je určena následovně. Plochu S_S rozdělíme na dvě části S_{S_1} a S_{S_2}, kde S_{S_1} je část plochy S_S, ležící v oblasti s $\varepsilon_r = \varepsilon_{r_1}$, S_{S_2} je část plochy S_2, ležící v oblasti s $\varepsilon_r = \varepsilon_{r_2}$. Bod C_{S+1} je vybrán náhodně a je rovnoměrně rozložen buď na ploše S_{S_1} nebo na ploše S_{S_2}. Pravděpodobnost toho, že bod se nachází na ploše S_{S_1} je

$$P_1 = \frac{\varepsilon_{r_1}}{\varepsilon_{r_1} + \varepsilon_{r_2}} , \tag{8.18}$$

kdežto pravděpodobnost polohy na ploše S_{S_2} je

$$P_2 = 1 - P_1 = \frac{\varepsilon_{r_2}}{\varepsilon_{r_1} + \varepsilon_{r_2}} . \tag{8.19}$$

Poloměry ploch S_S závisí na poloze C_S a jsou stanoveny následovně.

1. Nechť C_S je lokalizován na dielektrickém rozhraní. Je-li tímto rozhraním rovina (přímka ve dvoudimenzionálním případě), r_S je rovno nejmenší vzdálenosti mezi C_S a následujícími plochami:

   - S_b
   - rozhraní mezi dvěma dielektriky, ale jiné než to, na kterém je C_S lokalizováno.

   Např. si všimněte kroku z pozice C_k na obr. 8.4. Je-li dielektrické rozhraní, na kterém je bod C_S lokalizován, zakřiveno, jak je zobrazeno na obr.8.5., potom $r_S = d$. Pokud je d malé vzhledem k poloměru křivosti dielektrického rozhraní, dopustíme se malé chyby při použití vztahů (8.15) resp. (8.16), které platí přesně pro rovinné rozhraní.

2. Leží-li C_S mezi S_b a S'_b, S_d a S'_d nebo S_d a S''_d, položíme r_S rovno d. Tím vlastně zkracujeme délky kroků, které mají možnost protnout buď plochu S_b nebo S_d.

3. Jestliže C_S je lokalizováno uvnitř S_b, ale nepatří do oblastí 1 nebo 2 definovaných nahoře, potom za hodnotu r_S zvolíme nejkratší vzdálenost mezi C_S a plochami S_b či S_d.

Rozehrání náhodného bodu s rovnoměrným rozdělením na kouli je triviální. Jeden způsob rozehrání náhodného bodu s rovnoměrným rozdělením na jednotkové kouli jsme již uvedli v kap. IV. Druhý možný způsob navrhl Müller.

Nejdříve vygenerujeme náhodná čísla x_i, $i = 1, 2, 3$ z normálního rozdělení $N(0, 1)$. Poloha bodu C_{S+1} potom je

$$\begin{aligned} x_{S+1} &= x_S + \frac{r_S x_1}{[x_1^2 + x_2^2 + x_3^2]^{1/2}} , \\ y_{S+1} &= y_S + \frac{r_S x_2}{[x_1^2 + x_2^2 + x_3^2]^{1/2}} , \\ z_{S+1} &= z_S + \frac{r_S x_3}{[x_1^2 + x_2^2 + x_3^2]^{1/2}} , \end{aligned} \tag{8.20}$$

Výhodou použití metody Monte Carlo k řešení Laplaceovy rovnice jsou malé nároky na paměť počítače - nemusíme zaznamenávat celou síť ani trajektorii, vystačíme pouze se znalostí souřadnic okamžité polohy a počátečního bodu. Efektivnost řešení Laplaceovy rovnice metodou Monte Carlo roste s dimenzí prostoru. V dvourozměrném případě jsou efektivnější klasické algoritmy numerické matematiky, zatímco v třírozměrném případě je výhodnější metoda Monte Carlo. Hlavní přínos metody Monte Carlo však spočívá v tom, že nemusíme počítat potenciál ve všech bodech oblasti G současně. Prvořadé použití algoritmů náhodného bloudění bude tedy při zpřesňování průběhu elektrostatického pole ve vybraných kritických bodech oblasti G.

Kapitola 9

Metoda Monte Carlo v klasické statistické termodynamice

Jak již vyplývá z názvu této kapitoly, budeme se zabývat pouze klasickou statistickou termodynamikou. Případné zájemce o využití metody Monte Carlo v kvantové statistické fyzice odkazujeme na dostupnou publikaci [17].

Shrňme si nejdříve základní fakta týkající se našeho problému. Mějme dán systém N částic, které se nachází v objemu V za teploty T. Každá i–tá částice je charakterizována skupinou dynamických proměnných (s_i) reprezentujících počet stupňů volnosti. Stav systému je určen bodem $\boldsymbol{x} = ((s_1), \ldots, (s_N))$ ve fázovém prostoru Ω. Vývoj systému je popsán Hamiltoniánem $H(\boldsymbol{x})$, přičemž z tohoto Hamiltoniánu odečteme člen odpovídající kinetické energii, protože tento člen se dá vyhodnotit analyticky.

Úlohy klasické statistické termodynamiky se redukují na výpočet střední hodnoty pozorovatelné veličiny A (např. střední energie systému, střední počet částic), což vede k výpočtu integrálu

$$[A] = Z^{-1} \int_{\Omega} A(\boldsymbol{x})\ f(H(\boldsymbol{x}))\, \mathrm{d}\boldsymbol{x}\ , \tag{9.1}$$

kde f je distribuční funkce uvažovaného systému a

$$Z = \int_{\Omega} f(H(\boldsymbol{x}))\, \mathrm{d}\boldsymbol{x} \tag{9.2}$$

je partiční funkce.

Analytický výpočet integrálů (9.1) a (9.2) lze provést pouze v nejjednodušších případech. Ve většině případů je třeba přistoupit k numerické integraci, což vede k výpočtu integrálních sum. Jejich počet je tak obrovský, že i výpočet těchto sum klasickým způsobem je těžko proveditelný. Daleko schůdnější cesta vede přes ideu váženého průměru, se kterou jsme se setkali při výpočtu jednodimenzionálních integrálů.

Předpokládejme, že zkoumáme kanonický soubor. Jeho rozdělovací funkce je

$$f(H(\boldsymbol{x})) \sim e^{-\frac{H(\boldsymbol{x})}{k_B T}}\ , \tag{9.3}$$

kde k_B je Boltzmannova konstanta. Je vidět, že všechny stavy $\boldsymbol{x}$ odpovídající velké energii přispívají k celkové hodnotě integrálu velice málo. Pouze určitý počet stavů bude dávat

velký příspěvek. Kdybychom tedy postupovali obvyklým způsobem, odhad střední hodnoty by byl dán vztahem

$$[A] \approx \sum_i^n A(\boldsymbol{x}_i)\ f(H(\boldsymbol{x}_i))/\sum_i^n f(H(\boldsymbol{x}_i))\ . \tag{9.4}$$

Čím větší počet stavů systému bychom nasimulovali, tím lepší odhad $\langle A\rangle$ bychom obdrželi. Jelikož fázový prostor je o vysoké dimenzi, museli bychom generovat obrovské množství těchto stavů, přičemž většina z nich by přispívala zcela nepatrně k celkové hodnotě $[A]$. Abychom snížili časovou náročnost výpočtu integrálu (9.1) a (9.2) na rozumnou mez, nebudeme generovat možné stavy systému se stejnou pravděpodobností, jako v případě (9.4), ale budeme preferovat ty stavy systému, které nejvíce přispívají k hodnotě $[A]$. Znamená to tedy, že budeme generovat stavy systému s pravděpodobností $P(\boldsymbol{x})$ a odhad střední hodnoty pozorovatelné veličiny A spočteme podle vztahu

$$[A] \approx \frac{\sum_i^n A(\boldsymbol{x}_i)P^{-1}(\boldsymbol{x}_i)f(H(\boldsymbol{x}_i))}{\sum_i^n P^{-1}(\boldsymbol{x}_i)f(H(\boldsymbol{x}_i))}\ . \tag{9.5}$$

Zvolíme-li

$$P(\boldsymbol{x}) = Z^{-1}f(H(\boldsymbol{x}))\ , \tag{9.6}$$

tj. pravděpodobnost je rovna rovnovážné rozdělovací funkci, bude disperze hledané veličiny prakticky nulová a vztah (9.5) se redukuje na

$$[A] \approx (1/n)\sum_i^n A(\boldsymbol{x}_i)\ . \tag{9.7}$$

Tímto postupem se vyhneme neúnosně časově náročným výpočtům. Jenže v souvislosti s touto metodou okamžitě vzniká základní problém, jakým způsobem generovat stavy soustavy odpovídající termodynamickým rovnovážným stavům reálného systému - její rozdělovací funkce totiž není a priori známa. Tento základní problém vyřešil Metropolis a jeho kolegové [1]. Jejich idea spočívá ve využití Markovova řetězce. Tento řetězec startuje z nějakého počátečního stavu $\boldsymbol{x}_0$ a další stavy jsou generovány takovým způsobem, aby jejich rozdělení bylo dáno přibližně $P(\boldsymbol{x})$, přičemž nastávající a předcházející stavy leží ve fázovém prostoru blízko sebe.

Celý problém se tedy redukuje na stanovení pravděpodobnost přechodu $W(\boldsymbol{x},\boldsymbol{x}')$ ze stavu $\boldsymbol{x}$ do stavu $\boldsymbol{x}'$ za jednotku času. Abychom zajistili, že stavy generované v Markovově řetězci budou přibližně rozděleny podle pravděpodobnosti $P(\boldsymbol{x})$, musíme na $W(\boldsymbol{x},\boldsymbol{x}')$ klást následující požadavky:

1. Pro všechny komplementární dvojice $(S,\bar{S})$ souborů fázových bodů existuje $\boldsymbol{x}\in S$ a $\boldsymbol{x}'\in S'$ takové, že $W(\boldsymbol{x},\boldsymbol{x}')\neq 0$.
2. Pro všechny $\boldsymbol{x}$, $\boldsymbol{x}'$: $W(\boldsymbol{x},\boldsymbol{x}')\geq 0$. (9.8)
3. Pro všechny $\boldsymbol{x}$: $\sum\limits_{\boldsymbol{x}'} W(\boldsymbol{x},\boldsymbol{x}') = 1$.
4. Pro všechny $\boldsymbol{x}$: $\sum\limits_{\boldsymbol{x}'} W(\boldsymbol{x},\boldsymbol{x}')P(\boldsymbol{x}') = P(\boldsymbol{x})$.

První požadavek se týká ergodicity systému a druhý připouští pouze kladné pravděpodobnosti přechodu. Ergodicitou se zde rozumí, že každý stav fázového prostoru musí být dostupný po konečném počtu přechodů. Další podmínka vyjadřuje ten fakt, že celková pravděpodobnost přechodu do nějakého stavu $\boldsymbol{x}$ je rovna jedné. Třetí omezení říká, že limitní rozdělení stavů systému odpovídá rovnovážnému rozdělení $P(\boldsymbol{x})$, tj. tomu rozdělení, které požadujeme.

Předpokládejme, že pravděpodobnosti přechodů byly již specifikovány a stavy $\boldsymbol{x}_0$, $\boldsymbol{x}$, ... vygenerovány. Časový vývoj pravděpodobnosti $P(x_i, t)$, podle níž jsou stavy rozděleny je popsán rovnicí (Master equation)

$$\mathrm{d}P(\boldsymbol{x}, t)/\mathrm{d}t = -\sum_{\boldsymbol{x}'} W(\boldsymbol{x}, \boldsymbol{x}')P(\boldsymbol{x}, t) + \sum_{\boldsymbol{x}'} W(\boldsymbol{x}', \boldsymbol{x})P(\boldsymbol{x}', t) \; . \tag{9.9}$$

První člen na pravé staně popisuje rychlost všech přechodů pryč z uvažovaného stavu, zatímco druhý popisuje rychlost všech přechodů do uvažovaného stavu. Pro stacionární případ z (9.9) vyplývá vztah

$$\sum_{\boldsymbol{x}'} W(\boldsymbol{x}, \boldsymbol{x}')P(\boldsymbol{x}) = \sum_{\boldsymbol{x}'} W(\boldsymbol{x}', \boldsymbol{x})P(\boldsymbol{x}') \; . \tag{9.10}$$

S uvážením třetí podmínky v (9.8) odtud dostáváme

$$\sum_{\boldsymbol{x}'} W(\boldsymbol{x}, \boldsymbol{x}')P(\boldsymbol{x}') = P(\boldsymbol{x}) \; . \tag{9.11}$$

Ze vztahu (9.10) je vidět, že silnějším požadavkem, než čtvrtý požadavek v (9.8), je požadavek detailní rovnováhy

$$W(\boldsymbol{x}, \boldsymbol{x}')P(\boldsymbol{x}) = W(\boldsymbol{x}', \boldsymbol{x})P(\boldsymbol{x}') \; . \tag{9.12}$$

Požadavek detailní rovnováhy je pouze dostačující, ale ne nutný k tomu, aby Markovův řetězec konvergoval k požadovanému rozdělení! Ve výběru pravděpodobností přechodu existuje značná svoboda, protože tyto pravděpodobnosti nejsou jednoznačně určeny.

Ukažme si nyní jeden možný způsob výběru pravděpodobností přechodu. Nechť $\omega_{\boldsymbol{x},\boldsymbol{x}'}$, je reálné kladné číslo takové, že

$$\sum_{\boldsymbol{x}'} \omega_{\boldsymbol{x},\boldsymbol{x}'} = 1 \quad \text{a} \quad \omega_{\boldsymbol{x},\boldsymbol{x}'} = \omega_{\boldsymbol{x}',\boldsymbol{x}} \; . \tag{9.13}$$

Jelikož podmínka detailní rovnováhy klade omezení pouze na poměry pravděpodobností, můžeme definovat pravděpodobnosti přechodu pomocí poměrů

$$W(\boldsymbol{x}, \boldsymbol{x}') = \begin{cases} \omega_{\boldsymbol{x},\boldsymbol{x}'} \frac{P(\boldsymbol{x}')}{P(\boldsymbol{x})} & \text{pro } \boldsymbol{x} \neq \boldsymbol{x}' \; , \; P(\boldsymbol{x}')/P(\boldsymbol{x}) < 1 \\ \omega_{\boldsymbol{x},\boldsymbol{x}'} & \text{pro } \boldsymbol{x} \neq \boldsymbol{x}' \; , \; P(\boldsymbol{x}')/P(\boldsymbol{x}) \geq 1 \\ \omega_{\boldsymbol{x},\boldsymbol{x}'} + \sum\limits_{\boldsymbol{x}} \omega_{\boldsymbol{x},\boldsymbol{x}'}(1 - P(\boldsymbol{x}')/P(\boldsymbol{x})) & \text{pro } \boldsymbol{x} = \boldsymbol{x}' \; . \end{cases} \tag{9.14}$$

Konkrétní hodnoty reálných čísel $\omega_{\boldsymbol{x},\boldsymbol{x}'}$ stále nejsou jednoznačně stanoveny, jejich volba závisí na našem uvážení. Čím vhodněji provedeme volbu $\omega_{\boldsymbol{x},\boldsymbol{x}'}$ tím rychleji budeme v Markovově řetězci generovat stavy odpovídající termodynamické rovnováze. Je třeba upozornit na ten závažný fakt, který vlastně dělá tuto metodu schůdnou, že ve výrazu

pro pravděpodobnost přechodu $W(\boldsymbol{x}, \boldsymbol{x}')$ (9.14) se nevyskytují partiční funkce Z – tyto se totiž pokrátí díky poměru $P(\boldsymbol{x'})/P(\boldsymbol{x})$. Za tuto skutečnost však platíme tím, že partiční funkci Z nemůžeme spočíst přímo během simulace. Odtud plyne, že například volnou energii

$$F = -k_B T \ln Z \tag{9.15}$$

či entropii

$$S = \frac{U - F}{T} \tag{9.16}$$

nemůžeme spočíst přímo. Existuje několik způsobů, jak spočíst tyto veličiny. Jedna možnost výpočtu entropie vede přes využití vztahu

$$S = -k_B \sum_i p_i \ln p_i \ , \tag{9.17}$$

kde p_i je pravděpodobnost výskytu stavu $\boldsymbol{x}_i$. V principu bychom mohli jednoduše zaznamenávat kolikrát se i – tý stav vyskytl (n_i') a potom položit $p_i = n_i'/n$, kde n je celkový počet simulovaných stavů. Podrobněji se však touto problematikou dále nebudeme zabývat.

Nyní můžeme shrnout základní algoritmus výpočtu MC:

1. Specifikace počátečního stavu $\boldsymbol{x}_0$ ve fázovém prostoru.
2. Generace nového stavu $\boldsymbol{x'}$.
3. Výpočet pravděpodobnosti $W(\boldsymbol{x}, \boldsymbol{x'})$.
4. Generace náhodného čísla y z $R(0, 1)$.
5. Je-li pravděpodobnost přechodu $W(\boldsymbol{x}, \boldsymbol{x'})$ menší než náhodné číslo y, potom starý stav se přijme za nový stav systému ($\boldsymbol{x} = \boldsymbol{x'}$) a vrátíme se k bodu 2)
6. Jinak přijmeme nový stav ($\boldsymbol{x} \to \boldsymbol{x'}$) a vrátíme se k bodu 2).

V prvním kroku tedy inicializujeme výchozí stav systému. Vzhledem k tomu, že v Markovově řetězci ztrácíme informaci o počátečním stavu, nemusíme se snažit příliš složitě vybírat tento počáteční stav. Pozor musíme dávat pouze na to, abychom počáteční stav nezvolili v příliš bezvýznamné oblasti fázového prostoru. V druhém kroku vybíráme nový stav náhodně. Budeme-li například uvažovat o plynném systému, náhodně si vybereme částici plynu a u ní náhodně zvolíme novou pozici v rámci předem zadaného poloměru. V krocích tři až pět rozhodujeme o tom, zda přijmeme či zamítneme tento nový stav. Toto rozhodování se děje pomocí náhodné veličiny Y z $R(0, 1)$, protože $P(Y < W(\boldsymbol{x}, \boldsymbol{x'})) = W(\boldsymbol{x}, \boldsymbol{x'})$.

Dříve než začneme zaznamenávat potřebné údaje na výpočet střední hodnoty pozorovatelné veličiny A, musíme nechat odeznít vliv počátečních podmínek. Během výpočtu sledujeme, zda se nenacházíme v blízkosti termodynamické rovnováhy, rovněž testujeme vliv různých počátečních podmínek i vliv délky Markovova řetězce na výsledky.

9.1 Kanonický soubor

V této části budeme aplikovat poznatky uvedené v předešlých odstavcích na případ kanonického souboru, který je charakterizován konstantním počtem částic N, konstantním objemem V a teplotou T. V tomto případě pro střední hodnotu pozorovatelné veličiny A máme

$$\begin{aligned} [A] &= Z^{-1} \int_{\Omega} A(\boldsymbol{x}) \exp\left(-\frac{H(\boldsymbol{x})}{k_B T}\right) \mathrm{d}\boldsymbol{x} \\ Z &= \int_{\Omega} \exp\left(-\frac{H(\boldsymbol{x})}{k_B T}\right) \mathrm{d}\boldsymbol{x} \ . \end{aligned} \tag{9.18}$$

Rovnovážná pravděpodobnost stavů je

$$P(\boldsymbol{x}) = Z^{-1} \exp\left(-\frac{H(\boldsymbol{x})}{k_B T}\right) \ . \tag{9.19}$$

Požadujeme-li splnění detailní rovnováhy, dostáváme

$$\begin{aligned} \frac{W(\boldsymbol{x}, \boldsymbol{x'})}{W(\boldsymbol{x'}, \boldsymbol{x})} &= \frac{P(\boldsymbol{x'})}{P(x)} = \exp(-\Delta H / k_B T) \ , \\ \Delta H &= H(\boldsymbol{x}) - H(\boldsymbol{x'}) \ . \end{aligned} \tag{9.20}$$

Na základě vztahu (9.14)) pro pravděpodobnosti přechodu můžeme psát

$$W(\boldsymbol{x'}, \boldsymbol{x}) = \begin{cases} \omega_{\boldsymbol{x'},\boldsymbol{x}} \exp(-\Delta H / k_B T) & \text{pro } \Delta H > 0 \quad , \\ \omega_{\boldsymbol{x'},\boldsymbol{x}} & \text{jinak} \quad . \end{cases} \tag{9.21}$$

Čísla $\omega_{\boldsymbol{x'},\boldsymbol{x}}$ musí vyhovovat podmínce (9.13), jinak jejich volba závisí na nás. $W(\boldsymbol{x}, \boldsymbol{x'})$ určuje pravděpodobnosti přechodu za jednotku času a veličiny určují časové měřítko.

V případě kanonického souboru algoritmus výpočtu vypadá takto:

1. Specifikace počátečního stavu souboru.
2. Generace nového stavu $\boldsymbol{x'}$.
3. Výpočet změny energie ΔH.
4. Je-li $\Delta H < 0$, přijmout nový stav a vrátit se k bodu 2.
5. Spočíst $\exp(-\Delta H / k_B T)$.
6. Generace náhodného čísla y z $R(0, 1)$.
7. Je-li y menší než $\exp(-\Delta H / k_B T)$, přijmout nový stav a vrátit se k bodu 2.
8. Jinak starý stav se stává zároveň novým stavem a vrátit se k bodu 2.

Z tohoto algoritmu vidíme zřetelněji způsob výběru pravděpodobnosti přechodu. Systém se pohybuje po takové fázové trajektorii, která jej vede do stavů s minimální energií odpovídající parametrům (N, V, T). Krok čtyři říká, že vždy přijmeme tu novou konfiguraci, která má nižší energii než předcházející. Konfigurace, u nichž dochází ke vzrůstu energie, jsou přijímány pouze s pravděpodobností danou Boltzmannovým faktorem.

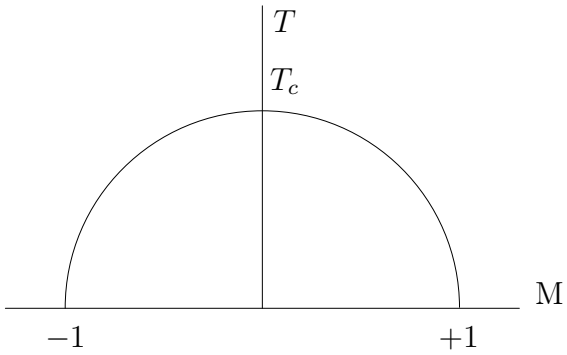
9.2 Isingův model

Abychom demonstrovali MC algoritmus pro kanonický soubor, budeme studovat Isingův model, který je definován následovně. Nechť $G = L^d$ je d–dimenzionální mřížka. Ke každému uzlu i odpovídá spin s_i , který může nabývat dvou hodnot: +1 nebo -1. Interakce mezi spiny je charakterizována veličinou J. Bude-li se mřížka nacházet ve vnějším poli K, dostáváme pro Hamiltonián

$$H = -J \sum_{<i,j>} s_i s_j + \mu K \sum_i s_i \ . \tag{9.22}$$

V první sumě na pravé straně budeme uvažovat pouze interakci mezi nejbližšími sousedy. Symbol μ označuje magnetický moment spinu. Má-li interakční konstanta J kladnou hodnotu, uvedený Hamiltonián bude modelovat ferromagnetickou látku, tj. všechny spiny se budou snažit orientovat do stejného směru. Je-li interakční konstanta J záporná, dostáváme model anti-ferromagnetika a spiny se snaží být navzájem antiparalelní. V dalším budeme předpokládat případ $J > 0$.

Isingův model demonstruje fázový přechod druhého druhu, který nastává při kritické teplotě T_c . Pro teploty T větší než T_c je vektor magnetizace m (počet spinů orientovaných „nahoru" – počet spinů orientovaných „dolů") nulový v nulovém magnetickém poli. Pro teploty nižší než T_c dostáváme dvojnásobně degenerovanou spontánní magnetizaci. Situace je schematicky zachycena na obr. 9.1.



Obrázek 9.1: Schematický fázový diagram pro třírozměrný Isingův model: T - teplota, T_c - kritická teplota, M - magnetizace.

První krok při specifikaci algoritmu výpočtu nás vede ke konkretizaci pravděpodobnosti přechodu z jednoho stavu do druhého. Ukazuje se, že nejvhodnější a zároveň nejjednodušší je zvolit pravděpodobnost, která popisuje přechod pouze jediného spinu, ostatní spiny zůstávají fixovány. Symbolicky můžeme tuto situaci znázornit

$$W_i(s_) : (s_1, \ldots, s_i, \ldots, s_N) \rightarrow (s_1, \ldots, -s_i, \ldots, s_N) \quad , \tag{9.23}$$

kde W_i je pravděpodobnost, že za jednotku času i–tý spin přejde ze stavu s_i do stavu -s_i. Takovýmto způsobem dáme po řadě možnost všem spinům zaujmout novou polohu, přičemž toto pořadí může být přesně zadáno či může být provedeno náhodně. Až projdeme všechny uzly souboru, vytváříme nový stav souboru. Tento model se nazývá „single-spin-flip Ising model". Je jasné, že v tomto modelu není počet spinů nahoru $N_\uparrow$ a počet spinů dolů $N_\downarrow$ konstantní, pouze jejich součet $N = N_\uparrow + N_\downarrow$ je konstantní.

Nechť $P(\boldsymbol{s})$ označuje pravděpodobnost stavu $\boldsymbol{s}$. V případě termodynamické rovnováhy, za konstantní teploty T a konstantního vnějšího pole K, pravděpodobnost, že i–tý spin je ve stavu $s+i$, je úměrná Boltzmannovu faktoru

$$P_{eq}(s_i) \sim \exp(-H(s_i)/k_BT) \; . \tag{9.24}$$

Požadujme splnění detailní rovnováhy

$$W_i(s_i)P_{eq}(s_i) = W_i(-s_i)P_{eq}(-s_i) \tag{9.25}$$

nebo-li

$$W_i(-s_i)/W_i(s_i) = P_{eq}(s_i)/P_{eq}(-s_i) \; . \tag{9.26}$$

Vezmeme-li do úvahy tvar Hamiltoniánu (9.22) ($K = 0$), dostáváme

$$W_i(s_i)/W_i(-s_i) = \exp(-s_iE_i/k_BT)/\exp(s_iE_i/k_BT) \quad , \tag{9.27}$$

kde $E_i = J \sum\limits_{<i,j>} s_j$.

Podmínky (9.24) – (9.27)) nespecifikují pravděpodobnost přechodu W jednoznačně. Nejčastěji se za tuto hodnotu volí Metropolisova funkce

$$W_i(s_i) = \min(1/\tau, (1/\tau)\exp(-\Delta H/k_BT)) \tag{9.28}$$

nebo Glauberova funkce

$$W_i(s_i) = (1/2\tau)(1 - s_i \tanh(E_i/k_BT)) \; , \tag{9.29}$$

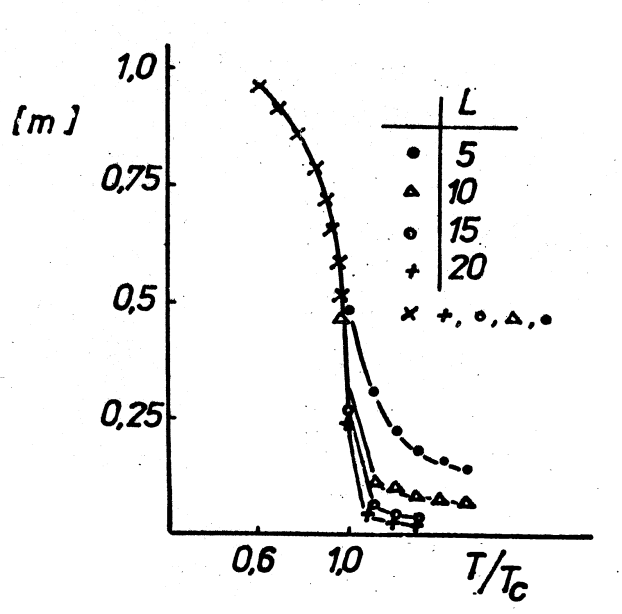
kde τ je libovolný faktor stanovující časovou škálu. Obvykle se za volí jednička. Doposud však není vyřešen problém, jaký je vztah mezi časem vystupujícím v metodě Monte Carlo a časem reálného systému. Lze dokázat, že obě volby pravděpodobnosti přechodu (9.28 – 9.29) jsou z matematického hlediska ekvivalentní.

Volbou W jsme připravili vše pro vytvoření algoritmu Isingova modelu :

1. Specifikace počátečního stavu. (9.30)
2. Výběr uzlového bodu i.
3. Výpočet W_i .
4. Generace náhodného čísla y z $R(0,1)$.
5. Je-li $W_i(s_i) > y$, potom $s_i \to -s_i$.
6. Jinak pokračuj v bodě 2 až do doby, než se projde všech N uzlů.

Na obr. 9.2. jsou uvedeny výsledky získané pro třírozměrný Isingův model. Simulace byla dělána pro 1000 MC kroků. Prvních 500 kroků se nebralo do úvahy, střední hodnota magnetizace byla počítána z následující série 500 kroků. Výpočty byly prováděny pro několik velikostí mřížek - počet uzlů mřížek je $N = L^3$.

Z uvedených výsledků je vidět, že hodnota magnetizace závisí na velikosti mřížek. Efekt je největší v okolí kritické teploty. Pro teploty menší než T_c jsou hodnoty magnetizace méně citlivé na volbu velikosti souboru a rychle konvergují ke správné termodynamické limitě.



Obrázek 9.2: Závislost absolutní hodnoty vektoru magnetizace na teplotě - Isingův model. T_c - kritická teplota, $N = L^3$ - počet uzlů mřížky.

Pro vysoké hodnoty vychází nenulová magnetizace, i když v termodynamické limitě zde spontánní magnetizace není. Tento jev souvisí s konečnou velikostí vzorku – čím větší vzorek budeme studovat, tím více se budeme blížit reálné situaci, tím více se projeví korelace mezi jednotlivými spiny. Je třeba též upozornit na to, že magnetizace na obr. 9.2. je udána v absolutní hodnotě. Jak bylo řečeno již v úvodu, pro teploty nižší než kritická teplota je nenulová pravděpodobnost přechodu konečné soustavy z kladné magnetizace do záporné magnetizace.

9.3 Grandkanonický soubor

Tento soubor je charakterizován objemem V, chemickým potenciálem a teplotou T. Základní problém při sestrojení odpovídajícího Markovova řetězce je proměnný počet částic v souboru. Představme si, že objem V je v daný moment zaplněn N částicemi. S časem se však tento počet neustále mění, částic může přibývati ubývat. Vzniká otázka, kterou částici máme ze systému odstranit, případně do jakého místa máme umístit novou částici. Dříve než zodpovíme tento problém, uveďme si čemu se rovná střední hodnota pozorovatelné veličiny A.

$$
\begin{aligned}
[A] &= Z^{-1} \sum_N (a^N/N!) \int_\Omega A(\boldsymbol{x}^N) \exp(-U(\boldsymbol{x}^N)/k_B T)\, \mathrm{d}\boldsymbol{x}^N \,, \qquad (9.31)\\
Z &= \sum_N (a^N/N!) \int_\Omega \exp(-U(\boldsymbol{x}^N)/k_B T)\, \mathrm{d}\boldsymbol{x}^N \,,
\end{aligned}
$$

kde a je definováno vztahem

$$
a = (h^2/2\pi m k_B T)^{-3/2} \lambda \,, \qquad (9.32)
$$

h je Planckova konstanta, m je hmotnost částice; výraz v závorce vznikl integrací přes hybnostní prostor. Faktor λ se nazývá absolutní aktivita a je roven

$$\lambda = \exp(\mu/k_B T) \; . \tag{9.33}$$

Je vidět, že struktura výrazů je podobná jako u kanonického souboru, navíc zde dostáváme proměnlivý počet částic. Algoritmus MC bude tedy zahrnovat tyto hlavní kroky:

1. Konfigurační změny v poloze částic.
2. Vytváření nových částic.
3. Zánik částic.

Budeme požadovat, aby v rovnováze počet kroků odpovídajících přesunu, vzniku a zániku částic byl stejný. Z (9.32) vyplývá, že konfiguračními změnami se nemusíme již zaobírat, protože algoritmus popisující tuto část je naprosto stejný jako algoritmus uvedený pro případ kanonického souboru. V dalším se musíme pouze zabývat tvorbou a zánikem částic. Předpokládejme, že jsme do objemu V náhodně rozmístili N částic. Pravděpodobnost, že v tomto objemu bude $N+1$ částic je

$$P(\boldsymbol{x}^{N+1}) = (a^{N+1}/(N+1)!) \exp(-U(\boldsymbol{x}^{N+1})/k_B T)/Z \; . \tag{9.34}$$

Obdobně, bude-li objem V obsahovat N částic a my náhodně jednu odstraníme, potom pravděpodobnost toho, že systém bude obsahovat $N-1$ částic je

$$P(\boldsymbol{x}^{N-1}) = (a^{N-1}/(N-1)!) \exp(-U(\boldsymbol{x}^{N-1})/k_B T)/Z \; . \tag{9.35}$$

S ohledem na to, co již bylo řečeno, můžeme pro pravděpodobnosti přechodu spojené se vznikem či zánikem částice psát

$$W(\boldsymbol{x}^N, \boldsymbol{x}^{N+1}) = \min[1, P(\boldsymbol{x}^{N+1})/P(\boldsymbol{x}^N)] \; , \tag{9.36}$$

$$W(\boldsymbol{x}^N, \boldsymbol{x}^{N-1}) = \min[1, P(\boldsymbol{x}^{N-1})/P(\boldsymbol{x}^N)] \; . \tag{9.37}$$

S těmito pravděpodobnostmi přechodu můžeme nyní zformulovat celkový MC algoritmus grandkanonického souboru:

1. Inicializace počáteční konfigurace s N částicemi nacházejícími se v objemu V.
2. Náhodný výběr se stejnou pravděpodobností jedné z procedur - PŘESUN, VZNIK a ZÁNIK.
3. PŘESUN:
   (a) Volba částice uvnitř objemu a její náhodný přesun.
   (b) Výpočet změny energie $\Delta U = U(\boldsymbol{x'}) - U(\boldsymbol{x})$.
   (c) Je-li ΔU záporné, přijmout novou konfiguraci a vrátit se k bodu 2.
   (d) Výpočet $\exp -\Delta U/k_B T$.
   (e) Generace náhodného čísla y z $R(0,1)$.

(f) Je-li y menší než $\exp(-\Delta U/k_BT)$, přijmout novou konfiguraci a vrátit se k bodu 2.

(g) Jinak stará konfigurace se stává novou konfigurací. Návrat na bod 2.

4. VZNIK:

(a) Zvolit náhodně souřadnice uvnitř objemu pro novou částici.

(b) Výpočet změny energie $\Delta U = U(\boldsymbol{x}^{N+1}) - U(\boldsymbol{x}^N)$.

(c) Výpočet $[a/(N+1)]\exp(-\Delta U/k_BT)$.

(d) Je-li toto číslo větší než jedna, přijmout novou konfiguraci a vrátit se k bodu 2.

(e) Generace náhodného čísla y z $R(0,1)$.

(f) Je-li y menší než $[a/(N+1)]\exp(-\Delta U/k_BT)$, přijmout vznik nové částice a vrátit se k bodu 2.

(g) Jinak zamítnout generaci nové částice a vrátit se k bodu 2.

5. ZÁNIK:

(a) Zvolit náhodně jednu částici z N částic nacházejících se v objemu V.

(b) Výpočet změny energie $\Delta U = U(\boldsymbol{x}^{N-1}) - U(\boldsymbol{x}^N)$.

(c) Výpočet $(N/a)\exp(-\Delta U/k_BT)$.

(d) Je-li toto číslo větší než jedna, přijmout novou konfiguraci a vrátit se k bodu 2.

(e) Generace náhodného čísla y z $R(0,1)$.

(f) Je-li y menší než $(a/N)\exp(-\Delta U/k_BT)$, přijmout zánik částice a odstranit částici z objemu V.

(g) Jinak zamítnout zánik částice a vrátit se k bodu 2.

Rychlost konvergence systému k termodynamické rovnováze závisí na velikosti změny souřadnic částic, které dovolíme, aby nastaly. Připustíme-li pouze malé změny polohy částic, bude třeba generovat obrovské množství stavů systému, abychom dospěli k termodynamické rovnováze, tj. konvergence bude pomalá. Na druhé straně velké změny v poloze částic mohou vést k tomu, že mnoho vygenerovaných stavů bude zamítnuto a opět konvergence bude pomalá.

Je znám i modifikovaný způsob simulace grandkanonického souboru, který je založen na té skutečnosti, že maximální počet částic M, které se nejvýše mohou vejít do daného objemu V, je konečný. Postupuje se tak, že se simuluje soubor o M částicích, přičemž N částic odpovídá reálné situaci a $M-N$ částice vystupují jako virtuální. Výhodou tohoto algoritmu je snížení počtu neúspěchů při změně počtu částic, nevýhodou je, že musíme počítat energii souboru o M částicích, což zase prodlužuje dobu výpočtu.

Kapitola 10

Transportní jevy

Pohyb částic látkovým prostředím se řídí stochastickými zákonitostmi. Určení charakteristik částic (např. koncentrace, driftová rychlost atd.) je možné pouze využitím metod nerovnovážné statistické fyziky. Chování částic i–tého druhu se popisuje pomocí rozdělovací funkce $f(\boldsymbol{r}, \boldsymbol{v}, t)$, kterou lze vypočítat z Boltzmannovy kinetické rovnice. Řešení Boltzmannovy rovnice lze nalézt analyticky pouze ve speciálních případech nebo za zjednodušujících předpokladů. Přitom chování částic je stochastické se známými pravděpodobnostmi pro jednotlivé elementární procesy. Nabízí se tedy použít metodu Monte Carlo, která bude spočívat v přímém nebo poněkud modifikovaném modelování skutečných procesů.

Historicky první použití metody Monte Carlo ve fyzice transportních jevů bylo modelování průchodu neutronů látkou. Nyní se metoda Monte Carlo používá pro modelování pohybu částic v mnoha dalších prostředích (plazma, plyn, pevné látky).

10.1 Model pohybu částice

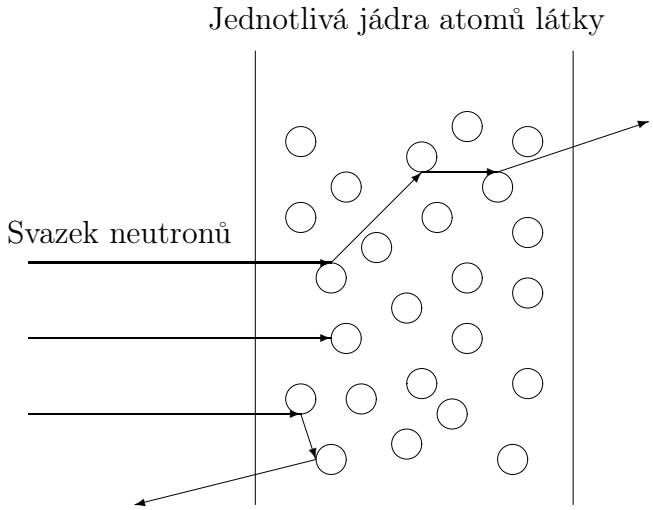
Budeme uvažovat pohyb částice látkovým prostředím, přičemž uvažovaná částice s ostatními částicemi látky neinteraguje na dálku (jednočásticový model). K interakci částice s ostatními dochází pouze pomocí srážek, u nichž předpokládáme, že k nim dochází pouze v jednom bodě prostoru. Na částici může ovšem působit vnější makroskopické pole.

Předpokládejme, že naše částice má v čase $t = 0$ energii E a pohybuje se z bodu $\boldsymbol{r}$ rychlostí $\boldsymbol{v}$ a působí na ni vnější pole silou $\boldsymbol{F}$. Pro časy $t > 0$ se částice bude pohybovat podle zákonů mechaniky až do té doby, než dojde ke srážce naší částice s jinou částicí prostředí. Při srážce může dojít k následujícím případům:

1. částice je zachycena, její dráha tím končí,
2. nastane pružná srážka,
3. nastane nepružná srážka,
4. nastane nepružná srážka a vznikají další částice.

Každý z těchto případů má jistou pravděpodobnost, přičemž tyto pravděpodobnosti jsou známy. Po srážce bude mít naše částice novou rychlost. Rozdělení pravděpodobnosti rychlosti po srážce je rovněž známo. Pokud částice nebyla zachycena, tak se celý proces volného letu a srážky opakuje. Jestliže tento proces budeme simulovat na počítači a

zaznamenávat údaje, které nás zajímají, pak po statistickém zpracování můžeme získat hledané hodnoty; např. můžeme studovat průchod svazku neutronů rovinnou deskou, viz. obr. 10.1.



Obrázek 10.1: Průchod neutronů rovinnou deskou.

Jestliže budeme zaznamenávat počet prošlých neutronů, můžeme po mnohonásobném opakování simulace pohybu neutronů odhadnout pravděpodobnost průchodu neutronu deskou. Rovněž můžeme zaznamenávat energie prošlých neutronů a odhadnout distribuční funkci pro energie prošlých neutronů.

Všimněme si nyní jednotlivých bodů uvedeného modelu.

10.2 Srážky

Srážkou budeme rozumět interakci dvou částic. Budeme předpokládat, že dosah této interakce je omezený a že se částice pohybují mimo oblast interakce volně. To odpovídá situaci, kdy částice před srážkou a po srážce jsou natolik vzdáleny, že vzájemnou interakci můžeme zanedbat. Nechť částice 1 má hmotnost m_1 a rychlost před srážkou $\boldsymbol{v_1}$, částice 2 má hmotnost m_2 a rychlost před srážkou $\boldsymbol{v_2}$. Nechť dále při srážce nevznikají nové částice. Pak rychlosti částic 1 a 2 po srážce jsou

$$\begin{aligned} \boldsymbol{v_1'} &= \frac{m_1\boldsymbol{v_1} + m_2\boldsymbol{v_2}}{m_1 + m_2} + \frac{m_2}{m_1 + m_2} g'\boldsymbol{\Omega}\,, \\ \boldsymbol{v_2'} &= \frac{m_1\boldsymbol{v_1} + m_2\boldsymbol{v_2}}{m_1 + m_2} - \frac{m_1}{m_1 + m_2} g'\boldsymbol{\Omega}\,, \end{aligned} \tag{10.1}$$

kde $\boldsymbol{\Omega}$ je jednotkový vektor ve směru vzájemné rychlosti částic po srážce a

$$g' = \sqrt{|\boldsymbol{v_1} - \boldsymbol{v_2}| - 2Q/\mu} \tag{10.2}$$

je velikost vzájemné rychlosti obou částic, přičemž Q je vnitřní energie spotřebovaná nebo uvolněná při nepružné srážce a

$$\mu = \frac{m_1 m_2}{m_1 + m_2} \tag{10.3}$$

je redukovaná hmotnost. Těmito rovnicemi je určena rychlost částic po srážce až na neznámý jednotkový vektor. Aby byly rychlosti po srážce plně určeny je nutno mimo rychlostí částic před srážkou zadat další veličiny, např. celkový moment hybnosti $\boldsymbol{L_T}$ v těžišťové souřadné soustavě.

Nechť $\boldsymbol{r}$ je vektor vzájemné vzdálenosti částic a $\boldsymbol{g}$ je vektor vzájemné rychlosti v určitý časový okamžik před srážkou. Pak platí

$$\boldsymbol{L_T}(t) = \boldsymbol{r} \times \mu \boldsymbol{g} = konst\ . \tag{10.4}$$

a vektor vzájemné rychlosti $\boldsymbol{g'}$ bude po srážce ležet v rovině určené vektory $\boldsymbol{r}$ a $\boldsymbol{g}$. Označíme χ úhel mezi vektory $\boldsymbol{g}$ a $\boldsymbol{g'}$ a γ úhel mezi vektory $\boldsymbol{r}$ a $\boldsymbol{g}$.

Pak pro vektor $\boldsymbol{\Omega}$ dostáváme

$$\boldsymbol{\Omega} = -\frac{\sin\chi}{\sin\gamma}\frac{\boldsymbol{r}}{|\boldsymbol{r}|} + (\cos\chi - \sin\chi \mathrm{cotg}\gamma)\frac{\boldsymbol{g}}{|\boldsymbol{g}|}\ . \tag{10.5}$$

V tomto vztahu již zůstal pouze jeden neznámý parametr - rozptylový úhel χ. Tento rozptylový úhel χ bude při našem modelování náhodná veličina a bude rozehráván pomocí své hustoty pravděpodobnosti - diferenciálního účinného průřezu.

Většinou však není známa ani ploha rozptylových center (např. v plynu) a kromě rozptylového úhlu se musí náhodně rozehrát i druhý úhel φ, který dále určuje směr vektoru $\boldsymbol{\Omega}$. Vektor vzájemné rychlosti $\boldsymbol{g'}$ po srážce se pak určuje z rovnic

$$g'_x = g_x \cos\chi + (g_y^2 + g_z^2)^{1/2} \sin\chi \sin\varphi, \tag{10.6}$$

$$g'_y = g_y \cos\chi + \sin\chi(v_r g_z \cos\varphi - g_x g_y \sin\varphi)/(g_y^2 + g_z^2)^{1/2},$$
$$g'_z = g_z \cos\chi - \sin\chi(v_r g_y \cos\varphi + g_x g_z \sin\varphi)/(g_y^2 + g_z^2)^{1/2}.$$

10.3 Účinný průřez

Účinný průřez je základní charakteristikou srážky. Účinný průřez určitého typu srážky si můžeme názorně (nikoli však úplně fyzikálně správně) představit jako fiktivní plochu (průřez) kolem atomu nebo molekuly atd., na kterou když dopadne druhá částice, tak dojde ke srážce obou částic. Z toho vyplývá, že pravděpodobnost srážky P_S jedné částice s druhou musí být počítána jako geometrická pravděpodobnost

$$P_S = \frac{\sigma}{S}\ , \tag{10.7}$$

kde σ je účinný průřez a S celková plocha, na níž může druhá částice dopadnout.

Každý druh srážky (pružná srážka, excitace, ionizace, ...) je charakterizován odpovídajícím účinným průřezem. Jestliže je při interakci dvou částic možných n druhů srážek a σ_i je účinný průřez pro i–tý druh srážky, pak se zavádí celkový (totální) účinný průřez σ_T vztahem

$$\sigma_T = \sum_{i=1}^{n} \sigma_i\ . \tag{10.8}$$

Celkový účinný průřez pak charakterizuje pravděpodobnost, že vůbec dojde k nějaké srážce. Jestliže dojde ke srážce, pak pravděpodobnost i – tého druhu srážky je σ_i/σ_T. Výběr druhu srážky se tedy rozehrává stejným způsobem jako diskrétní náhodná veličina.

Při rozptylu mohou vektory vzájemné rychlosti před a po srážce svírat nejrůznější rozptylové úhly χ. Hustota pravděpodobnosti $f(\chi)$ se dá vyjádřit jako

$$f(\chi) = \frac{2\pi\sigma_d(\chi)\sin\chi}{\sigma} , \tag{10.9}$$

kde $\sigma_d(\chi)$ je diferenciální účinný průřez. Platí

$$\sigma = 2\pi \int_0^\pi \sigma_d(\chi)\sin\chi \,\mathrm{d}\chi . \tag{10.10}$$

Pro rozehrání náhodného rozptylového úhlu χ při srážce, která je popsána diferenciálním účinným průřezem σ_d, pak dostáváme rovnici

$$y = 2\pi \int_0^\chi \sigma_d(\chi')\sin\chi' \,\mathrm{d}\chi' , \quad y \text{ z } R(0,1) , \tag{10.11}$$

kterou musíme vyřešit vzhledem k χ. Úhel φ je rovnoměrně rozdělen na intervalu $(0, 2\pi)$

$$\varphi = 2\pi y \qquad y \text{ z } R(0,1) . \tag{10.12}$$

Dále pro izotropní rozptyl se rovnice (10.11) dá vyřešit a dostáváme

$$\cos\chi = 1 - 2y . \tag{10.13}$$

10.4 Délka volné dráhy

Distribuční funkce pro délku volné dráhy s mezi dvěma po sobě jdoucími srážkami má tvar

$$F(s) = 1 - \exp\left(-\int_0^s n(\boldsymbol{r}(t))\sigma(\boldsymbol{r}(t))\,\mathrm{d}s\right) , \tag{10.14}$$

kde $\boldsymbol{r}(t)$ je polohový vektor trajektorie částice a n je koncentrace částic prostředí. Náhodná délka volné dráhy se pak generuje pomocí metody inverzní funkce, což znamená, že musíme řešit rovnicí

$$\ln y = -\int_0^s n\sigma \,\mathrm{d}s , \quad y \text{ z } R(0,1) \tag{10.15}$$

vzhledem k s. Řešení této rovnice není v obecném případě snadné. Pokud součin $n\sigma$ není funkcí polohy dostaneme

$$\begin{aligned} \ln y &= -n\sigma s , \\ s &= -\frac{1}{n\sigma}\ln y . \end{aligned} \tag{10.16}$$

Jestliže σ není konstantní, závisí např. na energii částice, je výhodnější přejít k době t mezi srážkami. Pro dobu mezi dvěma po sobě jdoucími srážkami platí distribuční funkce

$$F(t) = 1 - \exp\left(-\int_0^t n\sigma(v)v \,\mathrm{d}t\right) . \tag{10.17}$$

Pro výpočet t máme rovnici

$$\ln y = -\int_0^t n\sigma v \, \mathrm{d}t \ , \quad y \text{ z } R(0,1) \ . \tag{10.18}$$

Je výhodnější přejít od této rovnice k ekvivalentní diferenciální rovnici. Definujme funkci Ψ vztahem

$$\Psi(t) = -\int_0^t n\sigma v \, \mathrm{d}t \ . \tag{10.19}$$

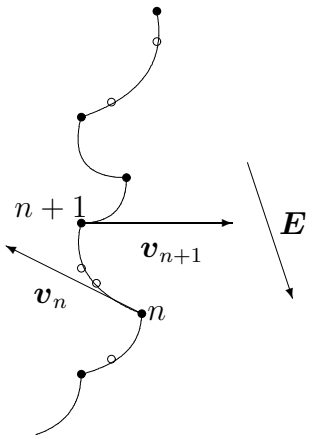
Pak platí

$$\frac{\mathrm{d}\Psi}{\mathrm{d}t} = -n\sigma v \ , \quad \Psi(0) = 0 \tag{10.20}$$

a dobu t mezi srážkami určíme z rovnice

$$\ln y = \Psi(t) \ . \tag{10.21}$$

Pro výpočet doby t tedy musíme řešit numericky diferenciální rovnici, což bývá časově



Obrázek 10.2: Metoda konstantního průřezu, $\boldsymbol{E}$ intenzita el.pole, plné kroužky - reálné srážky, prázdné kroužky - fiktivní srážky.

náročné. Tato potíž při řešení rovnic (10.20) a (10.15) se dá obejít pomocí metody konstantního průřezu (null - collision technique). Podstata této metody spočívá v následujícím obratu. Označme $M = \sup\{q(s), s \in (0,\infty)\}$. Zaveďme tzv. fiktivní účinný průřez pomocí vztahu

$$\sigma_F = M - \sigma \ . \tag{10.22}$$

Délku volné dráhy s pak budeme generovat podle vztahu

$$s = -\frac{1}{nM} \ln y \ , \quad y \in R(0,1) \ . \tag{10.23}$$

Nechť nyní σ_i jsou účinné průřezy jednotlivých druhů srážek. Pravděpodobnosti jednotlivých srážek položíme rovny σ_i/M. Dále však ještě může nastat tzv. fiktivní srážka s pravděpodobností σ_F/M. Při této fiktivní srážce se nebude měnit žádná fyzikální charakteristika částice, tj. částice se bude pohybovat dále se stejnou rychlostí. Na obr. 10.2 je tato situace znázorněna pro pohyb záporně nabité částice v homogenním elektrickém poli.

E/N [Td]	v_d [10^4 m/s] Reid	v_d [10^4 m/s] MC	v_d [10^4 m/s] boltz	$\langle\epsilon\rangle$ [eV] Reid	$\langle\epsilon\rangle$ [eV] MC	$\langle\epsilon\rangle$ [eV] boltz
1	1.272	1.273 ± 0.001	1.275	0.1015	0.1015 ± 0.0001	0.1016
12	6.832	6.838 ± 0.001	7.016	0.269	0.2689 ± 0.0001	0.2739
24	8.89	8.888 ± 0.001	9.146	0.408	0.4079 ± 0.0001	0.4163

Tabulka 10.1: Hodnoty makroskopických parametrů pro Reidův „ramp model". Reid – hodnoty z [18], MC, boltz – tato práce. v_d je driftová rychlost, $\langle\epsilon\rangle$ je střední energie.

Tato metoda bude tím efektivnější, čím bude σ_F menší. Důkaz správnosti metody konstantního průřezu lze nalézt v [10]. Stejný postup lze použít i pro generování času mezi srážkami. Zde však nastává problém v tom, že výraz σv, který zde vystupuje místo σ není shora ohraničený. To můžeme odstranit tak, že pro $v > v_{max}$ (v_{max} je pevně zvolená dostatečně velká rychlost) položíme $\sigma v = \text{konst}$.

	v_d [10^4 m/s]	$\langle\epsilon\rangle$ [eV]	ND_l [10^{24} m^{-1}s^{-1}]	ND_t [10^{24} m^{-1}s^{-1}]
tato práce	8.888 ± 0.001	0.4079 ± 0.0001	–	1.1 ± 0.1
Šimko	8.90 ± 0.06	0.4083 ± 0.0009	0.457 ± 0.020	1.146 ± 0.01
Reid	8.89	0.408	–	1.145
Pitchford	8.883	–	–	1.130
Braglia	8.88	0.408	–	1.134
Segur	8.881	0.4074	0.463	1.134
Penetrante	8.89 ± 0.02	0.409 ± 0.001	0.48 ± 0.02	1.133 ± 0.002
Ness	8.886	0.4079	0.4614	1.135

Tabulka 10.2: Hodnoty makroskopických parametrů pro Reidův „ramp model", E/N = 24 Td. Hodnoty druhých autorů byly převzaty z [19].

Standardním testem pro programy pro řešení Boltzmannovy rovnice a pro programy provádějící simulaci metodou Monte Carlo je Reidův "ramp model" [18]. Tento model byl použit již mnoha autory [19]. V Reidově modelu se elektrony pohybují v plynu v homogenním elektrickém poli, přičemž dochází pouze k pružným srážkám a k jednomu druhu nepružných srážek. Účinný průřez pro pružné srážky je konstantní

$$\sigma_{\text{el}} = 6.0 \times 10^{-20}\ \text{m}^2 \tag{10.24}$$

a účinný průřez pro nepružné srážky má následující průběh

$$\sigma_{\text{in}} = \begin{cases} 0 & \epsilon \leq 0.2\ \text{eV} \\ 10(\epsilon - 0.2) \times 10^{-20}\ \text{m}^2 & \epsilon > 0.2\ \text{eV}\ . \end{cases} \tag{10.25}$$

Oba dva druhy srážek jsou izotropní. Hmotnost částic plynu je rovna 4 amu, jeho teplota 0 K. Pro různé hodnoty redukované intenzity elektrického pole byly spočteny Reidem a ostatními autory makroskopické parametry pro elektrony. Výpočet makroskopických parametrů Reidova modelu metodou Monte Carlo provádí program `reid` (k dispozici v FTP archivu http://www.sci.muni.cz/ physics). V tab. 10.1 je uvedeno srovnání spočtených

hodnot driftové rychlosti a střední energie pro různé hodnoty E/N, v tab. 10.2 je uvedeno srovnání hodnot spočtených různými metodami a různými autory pro $E/N = 24$ Td.

Literatura

[1] Metropolis N., Rosenbluth A. W., Rosenbluth M. N., Teller A. H., Teller E.: J. Chem. Phys. 21 (1953), 1087

[2] Ermakov S. M., Michajlov G. A.: Statističeskoje modelirovanije, Nauka, Moskva 1982

[3] Binder K.: Metody Monte-Karlo v statističeskoj fizike, Mir, Moskva 1982

[4] Michálek J.: Úvod do teorie pravděpodobnosti a matematické statistiky, skripta, UJEP, Brno 1984

[5] Buslenko N. P., Golenko D. I., Soboĺ I. M., Sragonič V. G., Šrejder J. A.: Metod statističeskich ispytanij (metod Monte-Karlo), Gos. izd. fiziko-matem. literatury, Moskva 1962

[6] Buslenko N. P., Šrejder J. A.: Stochastické početní metody - metody Monte Carlo, SNTL, Praha 1965

[7] Heermann D. W.: Computer Simulation Methods in Theoretical Physics, Springer - Verlag Berlin, Heidelbegr 1986

[8] Hrach R.: Numerické metody ve fyzikální elektronice 1, 2, skripta, SPN, Praha 1981

[9] Humlíček J.: Statistické zpracování výsledků měření, skripta, UJEP, Brno 1984

[10] Sobol I. M.: Čislennyje metody Monte-Karlo, Moskva, Nauka, 1973

[11] Ermakov S. M.: Slučajnyje processy dlja rešenija klassičeskich uravnenij matematičeskoj fiziki, Moskva, Nauka, 1984

[12] Hurt J.: Simulační metody, skripta, SPN, Praha 1980

[13] Chin T. W., Gun T. S.: A shift-register sequence random number generator implemented on the microcomputer with 8088/8086 and 8087, Computer Physics Communications 47 (1987), 129 - 137

[14] Lauber J., Hušek R.: Simulační modely, skripta, SPN, Praha 1980

[15] Olehla M., Věchet V., Olehla J.: Řešení úloh matematické statistiky ve Fortranu, NADAS, Praha 1981

[16] Trunec D.: Řešení Boltzmannovy kinetické rovnice, diplomová práce, UJEP, Brno 1986

[17] Zamalin M., Norman G. E., Filinov V. S.: Metod Monte-Karlo v statističeskoj termodinamike, Nauka, Moskva 1977

[18] Reid I. D.: Aust. J. Phys. 32 (1979), 231.

[19] Ness K. F., Robson R. E.: Phys. Rev. A 34 (1986), 2185.