

X.
ALGORYTMY MONTE CARLO

W rozdziale tym omówione zostaną dwa podstawowe algorytmy Monte Carlo, a mianowicie algorytm Metropolis'a i algorytm symulowanego wyżarzania.

1 Algorytm Metropolis'a

Podstawowym problemem w obliczeniach Monte Carlo jest generacja liczb przypadkowych zgodna z zadaniem rozkładem. Algorytm generowania liczb losowych podlegających żadanemu (na ogół złożonemu) rozkładowi w przestrzeniach o dowolnej liczbie wymiarów podany został w artykule:

N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, & E. Teller, J. Chem. Phys. **21** (1953) 1087.

Nazywany jest on w skrócie **algorytmem Metropolis'a** lub **algorytmem $M(RT)^2$** .

Przedstawimy najpierw ideę algorytmu i sam algorytm, a następnie podamy dowód, że podany algorytm rzeczywiście generuje punkty w wielowymiarowej przestrzeni rozłożone zgodnie z zadaniem rozkładem prawdopodobieństwa.

Powiedzmy, że zadany został pewien rozkład prawdopodobieństwa o funkcji wagowej $w(X)$. Niech $X = (x_1, \dots, x_d)$ oznacza położenie punktu w d -wymiarowej przestrzeni. Zadanie nasze polega na generacji ciągu punktów:

$$X_0, X_1, \dots, X_i, \dots, X_d \tag{1}$$

rozłożonych zgodnie z zadaniem rozkładem $w(X)$, przy czym zakładamy, że funkcja wagowa jest unormowana, czyli

$$\int_{\Omega} dX w(X) = 1, \tag{2}$$

a zatem $w(X)$ jest gęstością prawdopodobieństwa.

Idea algorytmu Metropolis'a opiera się na zastosowaniu błędzenia przypadkowego w przestrzeni położenia X do generacji danego rozkładu.

Błędzenie to można zrealizować jako przypadkową wędrówkę pewnego wędrowcy.

Szukany ciąg punktów (1) jest tym ciągiem położenia, które odwiedza wędrowiec poruszający się w sposób przypadkowy w d -wymiarowej przestrzeni. Gdy wędrówka staje się coraz dłuższa, odwiedzane położenia zbliżają się do danego rozkładu.

Zapoznamy się teraz z samym algorytmem Metropolis'a. W celu rozważymy przypadkową podróż wędrowcy w d -wymiarowej przestrzeni położenia.

Przyjmijmy, że w pewnej chwili wędrowiec znajduje się w punkcie X_i ciągu położenia (1). Generacji kolejnego punktu X_{i+1} dokonujemy w następujący sposób:

(1) Wykonujemy krok próbny do nowego punktu X_t .

Ten nowy punkt można wybrać zgodnie z rozkładem jednorodnym wewnątrz d -wymiarowego sześcianu o boku δ zbudowanego wokół punktu X_i .

(2) Następnie krok próbny (punkt próbny) jest akceptowany lub odrzucony zgodnie z wartością stosunku

$$r = \frac{w(X_t)}{w(X_i)} . \quad (3)$$

Jeżeli $r > 1$, to nowy punkt jest akceptowany. Kładziemy wtedy $X_{i+1} = X_t$.

Jeżeli natomiast $r < 1$, to nowy punkt jest akceptowany z prawdopodobieństwem r .

Akceptacja punktu X_t z prawdopodobieństwem r oznacza wykonanie następującej procedury:

(a) Generujemy liczbę losową ϱ zgodnie z rozkładem jednorodnym w przedziale $[0, 1]$.

(b) Jeżeli $r > \varrho$, to punkt X_t zostaje zaakceptowany. Kładziemy zatem $X_{i+1} = X_t$.

(c) Jeżeli $r < \varrho$, to punkt X_t zostaje odrzucony. W tym przypadku kładziemy $X_{i+1} = X_i$.

Po wyznaczeniu punktu X_{i+1} dokonujemy kolejnego próbnego kroku z punktu X_{i+1} i generujemy X_{i+2} wg. powyższej procedury. Następnie powtarzamy tę samą procedurę do generacji punktów X_{i+3}, X_{i+4}, \dots

Punktem startowym X_0 naszej przypadkowej wędrówki może być dowolnie wybrany punkt. Nieco dokładniej problem wyboru punktu startowego zostanie omówiony w uwagach dotyczących praktycznej realizacji algorytmu.

Wykażemy teraz, że algorytm Metropolisa rzeczywiście generuje ciąg punktów rozłożonych zgodnie zadaną gęstością prawdopodobieństwa $w(X)$.

W tym celu rozważymy dużą liczbę wędrówców startujących z różnych punktów początkowych i poruszających się niezależnie od siebie w sposób przypadkowy w wielowymiarowej przestrzeni położeń X . Można zauważyć, że liczba wędrówców w punkcie X jest proporcjonalna do liczby wizyt pojedynczego przypadkowego wędrówcy w danym punkcie.

Oznaczmy przez $n_i(X)$ gęstość wędrówców w punkcie X po wykonaniu i kroków.

Obliczymy wypadkową liczbę wędrówców poruszających się z X do Y w kolejnym kroku, tzn. obliczymy zmianę gęstości wędrówców $\Delta n(X)$. Jest ona równa

$$\Delta n(X) = n_i(X)p(X \rightarrow Y) - n_i(Y)p(Y \rightarrow X) , \quad (4)$$

gdzie $p(X \rightarrow Y)$ jest prawdopodobieństwem przejścia z punktu X do punktu Y .

Przekształcamy (4) do postaci

$$\Delta n(X) = n_i(Y)p(X \rightarrow Y) \left[\frac{n_i(X)}{n_i(Y)} - \frac{p(Y \rightarrow X)}{p(X \rightarrow Y)} \right] . \quad (5)$$

Równowaga zachodzi wtedy, gdy

$$\Delta n(X) = 0 . \quad (6)$$

Oznaczmy symbolem $n_e(X)$ gęstość wędrówców w punkcie X w stanie równowagi. Ze wzorów (5) i (6) wynika, że

$$\frac{n_e(X)}{n_e(Y)} = \frac{n_i(X)}{n_i(Y)} = \frac{p(Y \rightarrow X)}{p(X \rightarrow Y)}. \quad (7)$$

Wraz ze wzrostem liczby kroków gęstość wędrówców dąży do gęstości równowagowej, tzn.

$$\lim_{i \rightarrow \infty} n_i(X) = n_e(X). \quad (8)$$

Rozważmy teraz wędrówkę zgodną z algorytmem Metropolis'a. Wtedy prawdopodobieństwo przejścia z X do Y można zapisać jako

$$p(X \rightarrow Y) = t(X \rightarrow Y)a(X \rightarrow Y), \quad (9)$$

gdzie $t(X \rightarrow Y)$ jest prawdopodobieństwem wykonania próbnego kroku z X do Y , a $a(X \rightarrow Y)$ jest prawdopodobieństwem akceptacji tego kroku.

Jeżeli próbný punkt leży wewnątrz odpowiednio małego sześcianu o krawędzi δ i środku w punkcie X , to punkt Y może być osiągnięty z punktu X w pojedynczym kroku. Wtedy

$$t(X \rightarrow Y) = t(Y \rightarrow X). \quad (10)$$

Wzory (7), (9) i (10) prowadzą do

$$\frac{n_e(X)}{n_e(Y)} = \frac{a(Y \rightarrow X)}{a(X \rightarrow Y)}. \quad (11)$$

Zgodnie z algorytmem Metropolis'a możliwe są dwa przypadki:

(1) $w(X) > w(Y)$, czyli $r < 1$.

Wtedy $a(Y \rightarrow X) = 1$ oraz

$$a(X \rightarrow Y) = \frac{w(Y)}{w(X)}. \quad (12)$$

(2) $w(X) < w(Y)$, czyli $r > 1$.

Wtedy $a(X \rightarrow Y) = 1$ oraz

$$a(Y \rightarrow X) = \frac{w(X)}{w(Y)}. \quad (13)$$

Dla każdego z przypadków (1) i (2) otrzymujemy zgodnie z (11) ten sam wynik, a mianowicie

$$\frac{n_e(X)}{n_e(Y)} = \frac{w(X)}{w(Y)}. \quad (14)$$

Wynika stąd, że

$$n_e(X) \sim w(X). \quad (15)$$

Jeżeli unormujemy rozkład równowagowy, to

$$n_e(X) = w(X) ,$$

co kończy dowód.

Uwagi: problemy numeryczne

- (1) Jaka powinna być optymalna wartość kroku próbnego δ ?

Optymalną wartość δ dobieramy tak, aby około połowy próbných kroków zostało zaakceptowanych.

- (2) Jak wybrać punkt startowy X_0 ?

Teoretycznie jest to wybór przypadkowy.

W praktyce, aby zapewnić tę przypadkowość, dokonujemy pewnej liczby kroków początkowych według algorytmu Metropolis'a i jako punkt startowy X_0 wybieramy ten punkt, dla którego wartość funkcji $w(X_0)$ jest duża (choć niekoniecznie największa). Proces ten nazywamy **termalizacją**.

2 Metoda symulowanego wyżarzania

Najpierw wyjaśnimy nazwę i przedstawimy ideę metody. Angielska nazwa metody to **simulated annealing**. W technologii terminem **annealing (wyżarzanie)** określane jest proces powolnego chłodzenia ciekłego szkła lub metalu w celu uzyskania utwardzonej (hartowanej) struktury.

Proces ten może też być nazywany po prostu **hartowaniem**. Można się też powołać na analogię z dziedziny muzyki i stosować termin **temperowanie**, które oznacza łagodne wyciszanie muzyki.

Zobrazujemy ideę metody na przykładzie procesu produkcji szkła. W procesie tym ciekły gorący materiał jest powoli chłodzony wg odpowiedniej procedury (zwanej po angielsku **annealing schedule**. Po powolnym osiągnięciu temperatury otoczenia powstaje trwała struktura szkła (bez naprężeń wewnętrznych).

W procesie produkcji szkła mogą być stosowane różne szybkości chłodzenia. Chłodzenie szybkie prowadzi do powstania struktury z naprężeniami wewnętrznymi, która odpowiada **minimum lokalnemu** energii całkowitej materiału.

W wyniku chłodzenia powolnego powstaje struktura bez naprężeń wewnętrznych, która odpowiada **minimum globalnemu** energii całkowitej.

Algorytm wykorzystujący symulację powolnego chłodzenia może stanowić **metodę znajdowania minimum globalnego funkcji wielu zmiennych**.

Zajmiemy się zastosowaniem metody symulowanego wyżarzania do znajdowania stabilnej struktury układu N cząstek klasycznych. Układ cząstek klasycznych znajduje się w stanie równowagi, jeżeli całkowita energia potencjalna tego układu osiąga minimum globalne.

Energia potencjalna oddziaływania pary cząstek (i, j) wynosi

$$u_{ij} = u(\mathbf{r}_{ij}) , \tag{16}$$

gdzie $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$. Całkowita energia potencjalna układu N cząstek dana jest wzorem

$$U = \sum_{i=1}^N \sum_{j>i}^N u_{ij}. \quad (17)$$

Jest to funkcja $3N$ zmiennych

$$U = U(\mathbf{r}_1, \dots, \mathbf{r}_N). \quad (18)$$

Konfigurację C_k układu N cząstek definiujemy jako zbiór położenia tych cząstek, czyli

$$C_k \equiv r^{(k)} = (\mathbf{r}_1^{(k)}, \dots, \mathbf{r}_N^{(k)}), \quad (19)$$

gdzie $k = 1, \dots, K$, a K jest całkowitą (maksymalną) liczbą konfiguracji. Oznaczmy przez U_k energię potencjalną k -tej konfiguracji

$$U_k \equiv U(r^{(k)}) = U(\mathbf{r}_1^{(k)}, \dots, \mathbf{r}_N^{(k)}), \quad (20)$$

czyli

$$U_k = \sum_{i=1}^N \sum_{j>i}^N u(\mathbf{r}_i^{(k)} - \mathbf{r}_j^{(k)}), \quad (21)$$

gdzie $\mathbf{r}_i^{(k)}, \mathbf{r}_j^{(k)} \in C_k$.

2.1 Algorytm metody symulowanego wyżarzania

(0) Zadajemy konfigurację początkową ($k = 0$)

$$C_0 = (\mathbf{r}_1^{(0)}, \dots, \mathbf{r}_N^{(0)}). \quad (22)$$

(1) Wybieramy program (plan) chłodzenia.

Komentarz dotyczący tego punktu podamy w dyskusji algorytmu.

(2) Tworzymy konfigurację aktualną: dla kroku k -tego jest to konfiguracja o numerze $k - 1$, czyli

$$C_k = C_{k-1}. \quad (23)$$

(3) Generujemy konfigurację "sąsiednią" C'_k w sposób następujący:

(2.a) Wybieramy losowo lub w sposób cykliczny wskaźnik j cząstki.

(2.b) Wektor położenia wybranej cząstki j -tej zmieniamy w sposób przypadkowy wewnątrz sześcianu o krawędzi δ i środkiem w \mathbf{r}_j .

Uwaga

Punkt (2.a) algorytmu oznacza, że nowy wektor położenia $\mathbf{r}'_j = (x'_j, y'_j, z'_j)$ wyznaczany jest następująco:

$$x'_j = x_j + (\xi_1 - 0.5)\delta, \quad (24)$$

$$y'_j = y_j + (\xi_2 - 0.5)\delta, \quad (25)$$

$$z'_j = z_j + (\xi_3 - 0.5)\delta, \quad (26)$$

gdzie ξ_1, ξ_2, ξ_3 są liczbami pseudoprzygodkowymi o rozkładzie jednorodnym w przedziale $(0, 1)$.

(4) Obliczamy energię potencjalną U'_k nowej konfiguracji.

(5) Zmieniamy konfigurację aktualną na konfigurację sąsiednią w sposób następujący:

(5.a) Jeżeli $U'_k \leq U_k$, to akceptujemy konfigurację sąsiednią. Dokonujemy wtedy podstawienia

$$C_{k+1} = C'_k \quad (27)$$

i rozpoczynamy obliczenia od punktu **(2)**.

(5.b) Jeżeli $U'_k > U_k$, to obliczamy stosunek prawdopodobieństw termodynamicznych

$$q = \frac{e^{-U'_k/k_B T}}{e^{-U_k/k_B T}} = e^{-\Delta U/k_B T}, \quad (28)$$

gdzie $\Delta U = U'_k - U_k$, i porównujemy q z liczbą pseudoprzygodkową $\xi \in (0, 1)$ o rozkładzie jednorodnym.

(i) Jeżeli $q \geq \xi$, to zmieniamy konfigurację wg. $C_{k+1} = C'_k$.

(ii) Jeżeli $q < \xi$, to pozostawiamy niezmienną konfigurację $C_{k+1} = C_k$ i rozpoczynamy obliczenia od punktu **(2)**.

Zgodnie z (28) w metodzie symulowanego wyżarzania stosowane są prawdopodobieństwa termodynamiczne określone przez klasyczny rozkład Boltzmana, czyli

$$p = C e^{-U/k_B T}. \quad (29)$$

Przedyskutujemy teraz różnicę pomiędzy algorytmem symulowanego wyżarzania a algorytmem Metropolis'a, w którym zastosowano rozkład Boltzmana.

Różnica ta polega na tym, że w metodzie symulowanego wyżarzania temperatura T we wzorze (28) obniżana jest według pewnego zadanego planu (schematu). Może to być:

(1) Schemat liniowy

$$T = T_0(1 - \alpha \Delta k). \quad (30)$$

(2) Schemat eksponencjalny

$$T = T_0 e^{-\alpha \Delta k}. \quad (31)$$

(3) Schemat "ręczny", zgodnie z którym temperaturę zmieniamy interakcyjnie co pewną liczbę kroków Δk .

We powyższych wzorach α jest parametrem szybkości chłodzenia, a Δk oznacza zmianę liczby konfiguracji (np. $\Delta k = 1$).

Dyskusja

- (1) Wybór temperatury początkowej.

W celu umożliwienia "przeskoków" pomiędzy minimami lokalnymi należy wybrać T_0 zgodnie z wartością różnicy pomiędzy maksymalną U_{max} i minimalną U_{min} energią potencjalną układu, czyli

$$k_B T_0 \simeq U_{max} - U_{min} . \quad (32)$$

- (2) Wybór krawędzi sześcianu δ dokonywany jest "doświadczalnie" tak, aby około połowy nowych konfiguracji zostało zaakceptowanych (por. algorytm Metropolisa).

Zgodnie z algorytmem symulowanego wyżarzania najczęściej akceptowane są konfiguracje o zmniejszającej się energii potencjalnej. Jednakże układ może z niezerowym prawdopodobieństwem przeskakiwać przez bariery energii potencjalnej oddzielające minima lokalne energii. Dla $T \rightarrow 0$ układ przechodzi do konfiguracji odpowiadającej globalnemu minimum energii potencjalnej, chociaż przejście to zachodzi z prawdopodobieństwem mniejszym od 1 (ale bliskim 1).

Zastosowania

- (1) Otrzymywanie stabilnych struktur układów N cząstek (metoda konkurencyjna w stosunku do metod dynamiki molekularnej).
- (2) Optymalizacja złożonych obwodów elektrycznych (w tym celu metoda ta została wymyślona przez Kirkpatricka).
- (3) Problem komiwożacza, czyli poszukiwanie minimalnej drogi w N podróżach.

Wtedy podstawiamy zamiast U drogę L określoną jako

$$L = \sum_{i=1}^N \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} .$$

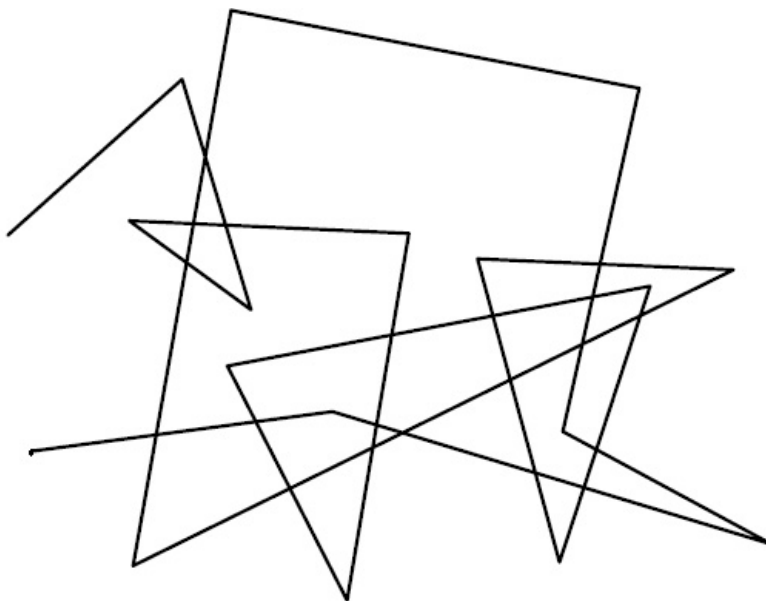
Punktem startowym jest punkt x_0, y_0 .

3 Ruchy Browna

Podstawy fizyczne

Ruchy Browna to nieuporządkowane ruchy cząstek zawieszin (np. pyłku roślinnego) zawieszonych w cieczy lub gazie. Rys. 1 pokazuje obraz ruchu Browna otrzymany w wyniku rzutowania na płaszczyznę kolejnych położenia cząstki pyłku obserwowanych w chwilach czasu: $0, \Delta t, 2\Delta t, \dots$. Jest to linia łamana.

Jeżeli będziemy rejestrowali położenia cząstki zawiesziny w systematycznie zmniejszających się odstępach czasu, np. według wzoru $t_n = \Delta t/n$, gdzie $n > 1$, to każdy odcinek prostej znów zamieni się w linię łamaną.



Rys. 1. Rzut trajektorii brownowskiej na płaszczyznę.

Oczywiście istnieje fizyczna granica ciągu t_n , np. wyznaczona przez skalę atomową odległości i czasu.

W celu wyjaśnienia ruchów Browna należy zauważyć, że typowe odstępów pomiędzy chwilami obserwacji wynoszą $0.01 \text{ s} \leq \Delta t \leq 1 \text{ s}$. Natomiast średni czas między zderzeniami cząsteczek gazu lub cieczy wynosi 10^{-14} s .

Wynika stąd, że ruchy Browna **nie są spowodowane pojedynczymi zderzeniami** cząstek zawiesiny (pyłku) z cząsteczkami cieczy lub gazu.

Natura ruchu Browna jest następująca:

Ruch Browna cząstki zawiesiny jest wynikiem **z bardzo wielu zderzeń** z molekułami gazu lub cieczy, w których wyniku cząstka zawiesiny traci "pamięć" o swoim poprzednim ruchu i uzyskuje pewien **ruch przypadkowy**. Jest to ruch (prawie) jednostajny **o przypadkowej prędkości początkowej**. W związku z tym ruchy Browna można opisać podobnie jak **problem błędzenia przypadkowego**, w którym zmienia się w sposób przypadkowy zarówno kierunek jak i długość każdego kroku.

Wyznamy średnie przesunięcie cząstki zawiesiny po N krokach. Oznaczamy symbolem \mathbf{s}_N wektor wypadkowego przesunięcia cząstki zawiesiny po N krokach, a symbolem \mathbf{d} wektor przemieszczenia cząstki w pojedynczym kroku. Wektory te związane są z sobą relacją

$$\mathbf{s}_N = \mathbf{s}_{N-1} + \mathbf{d} . \quad (33)$$

Podnosząc (33) stronami do kwadratu otrzymujemy

$$s_N^2 = s_{N-1}^2 + 2\mathbf{s}_{N-1} \cdot \mathbf{d} + d^2, \quad (34)$$

a stąd średni kwadrat przesunięcia po N krokach

$$\langle s_N^2 \rangle = \langle s_{N-1}^2 \rangle + 2\langle \mathbf{s}_{N-1} \cdot \mathbf{d} \rangle + \langle d^2 \rangle. \quad (35)$$

Średni kwadrat pojedynczego przemieszczenia definiujemy jako

$$a^2 \stackrel{def}{=} \langle d^2 \rangle. \quad (36)$$

Średnia wartość iloczynu skalarnego we wzorze (35) może być zapisana jako

$$\langle \mathbf{s}_{N-1} \cdot \mathbf{d} \rangle = \langle s_{N-1} \rangle \langle d \rangle \langle \cos \varphi \rangle, \quad (37)$$

gdzie $\varphi = \angle(\mathbf{s}_{N-1}, \mathbf{d})$.

Dla przypadkowych przemieszczeń

$$\langle \cos \varphi \rangle = 0, \quad (38)$$

czyli

$$\langle \mathbf{s}_{N-1} \cdot \mathbf{d} \rangle = 0. \quad (39)$$

Podstawiając ten wynik do wzoru (35) otrzymujemy

$$\langle s_N^2 \rangle = \langle s_{N-1}^2 \rangle + a^2. \quad (40)$$

Stosując (40) dla $N = 1$ mamy

$$\langle s_1^2 \rangle = a^2. \quad (41)$$

Zgodnie z metodą indukcji na podstawie (40) i (41) otrzymujemy

$$\langle s_N^2 \rangle = Na^2. \quad (42)$$

Jeżeli wprowadzimy średnią liczbę przesunięć cząstki zawiesiny w jednostce czasu, czyli częstość przesunięć, którą oznaczamy literą ν , to

$$N = \nu t, \quad (43)$$

a stąd i ze wzoru (5) otrzymujemy zależność średniego kwadratu przesunięcia od czasu

$$\langle s_N^2 \rangle = \nu a^2 t. \quad (44)$$

Średnie przesunięcie l_N po N krokach definiujemy jako średnią kwadratową przesunięć

$$l_N \stackrel{def}{=} \langle s_N^2 \rangle^{1/2}, \quad (45)$$

czyli

$$l_N = a\sqrt{N} = a\sqrt{\nu t}. \quad (46)$$

Wzór (46) podaje zależność charakterystyczną dla ruchów Browna:

Średnie przesunięcie cząstki jest proporcjonalne do pierwiastka kwadratowego z całkowitej liczby kroków, czyli do pierwiastka kwadratowego z czasu obserwacji.

Rozkład prawdopodobieństwa dla trajektorii brownowskich

Odpowiemy teraz na pytanie, jakiemu rozkładowi prawdopodobieństwa podlegają trajektorie cząstek wykonujących ruchy Browna?

Najpierw rozważymy położenie cząstki w kierunku x , czyli składową x -ową wektora przesunięcia \mathbf{s}_N . Przeszczenia w kierunku x w kolejnych krokach są wielkościami przypadkowymi co do wartości i znaku. Przesunięcie wypadkowe po N krokach jest więc sumą N liczb przypadkowych ξ_i , czyli

$$x_N = \sum_{i=1}^N \xi_i \stackrel{\text{ozn}}{=} x. \quad (47)$$

Zgodnie z **centralnym twierdzeniem granicznym** suma N liczb przypadkowych (dla odpowiednio dużego N) podlega rozkładowi normalnemu, czyli rozkładowi Gaussa.

Rozkład Gaussa przesunięć x_N scharakteryzowany jest wartością średnią

$$\mu = \langle x \rangle = 0$$

oraz wariancją

$$\sigma^2 = \langle x^2 \rangle.$$

Zgodnie z (44)

$$\sigma^2 = Na^2 = \nu a^2 t. \quad (48)$$

Wynika stąd, że rozkład prawdopodobieństwa dla wypadkowego przesunięcia cząstki w kierunku x dany jest wzorem

$$\varrho(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-x^2/2\sigma^2}. \quad (49)$$

Wzór (49) podaje gęstość prawdopodobieństwa rozkładu przesunięć w kierunku x dla trajektorii brownowskiej.

Zgodnie z (49) obliczamy prawdopodobieństwo otrzymania po N krokach, czyli po upływie czasu t , przemieszczenia cząstki zawartego w przedziale $[x, x + \Delta x]$ jako

$$P(x, \Delta x) = \varrho(x)\Delta x. \quad (50)$$

Rozkład prawdopodobieństwa dla trajektorii brownowskich w trzech wymiarach

Dla ruchu w trzech wymiarach odpowiedni rozkład prawdopodobieństwa przesunięć jest iloczynem trzech identycznych jak (49) Gaussianów zależnych od x, y, z .

Konstrukcja numeryczna trajektorii brownowskiej

Rozważamy ruch Browna w funkcji zmiennej czasowej τ . Zmienna τ może przyjmować wartości rzeczywiste (w symulacji rzeczywistego ruchu cząstki zawiesiny) lub urojone (w symulacji ewolucji czasowej układu kwantowego).

Przedział zmienności τ , czyli $0 \leq \tau \leq B$ dzielimy na d podprzedziałów o długości $\varepsilon = B/d$ każdy, gdzie d jest liczbą całkowitą. Punkty podziału oznaczamy $\tau_j = j\varepsilon$ ($j = 0, \dots, d$), przy czym $\tau_d = B$.

Ruch Browna w kierunku x w podprzedziale $[\tau_j, \tau_{j+1}]$ podlega rozkładowi normalnemu (49) o średniej $\mu = 0$ i odchyleniu standardowym

$$\sigma = \alpha\sqrt{\varepsilon} = \alpha\sqrt{\frac{B}{d}}, \quad (51)$$

gdzie parametr $\alpha = a\sqrt{\nu}$ ma wymiar $[\text{ms}^{-1/2}]$.

Ruch w podprzedziale $[\tau_j, \tau_{j+1}]$ jest wypadkową wielu kroków przypadkowych (czyli jest nadal ruchem Browna), stąd

$$x_{j+1} = x_j + \sigma\xi_j, \quad (52)$$

gdzie ξ_j podlega rozkładowi normalnemu (49) o średniej $= 0$ i wariancji $Var\xi = 1$.

W podprzedziale sąsiednim $[\tau_{j-1}, \tau_j]$ zachodzi związek

$$x_j = x_{j-1} + \sigma\xi_{j-1}. \quad (53)$$

Przekształcamy (52) do postaci

$$x_j = x_{j+1} - \sigma\xi_j. \quad (54)$$

Dodając stronami (53) i (54) otrzymujemy

$$x_j = \frac{1}{2}(x_{j-1} + x_{j+1}) + \frac{\sigma}{2}(\xi_{j-1} - \xi_j). \quad (55)$$

Ustalamy teraz x_{j-1} i x_{j+1} , a x_j ($j = 1, \dots, d-1$) traktujemy jako zmienną losową. Wg. wzoru (55) obliczmy średnią wartość zmiennej x_j

$$\langle x_j \rangle = \frac{1}{2}(x_{j-1} + x_{j+1}) \quad (56)$$

oraz wariancję tej zmiennej

$$\langle (x_j^2 - \langle x_j \rangle^2) \rangle = \frac{\sigma^2}{2}(\langle \xi^2 \rangle - \langle \xi \rangle^2) = \frac{\sigma^2}{2}Var\xi = \frac{\sigma^2}{2}, \quad (57)$$

ponieważ $Var\xi = 1$.

Dla wygody dalszych rozważań definiujemy nowe zmienne losowe

$$\zeta_j \stackrel{def}{=} \frac{1}{\sqrt{2}}(\xi_{j-1} - \xi_j), \quad (58)$$

przy czym $j = 1, \dots, d-1$. Wartość średnia nowej zmiennej losowej ζ_j wynosi

$$\langle \zeta_j \rangle = 0, \quad (59)$$

natomiast wariancja

$$\text{Var}\zeta_j = \frac{1}{2}\langle (\xi_{j-1} - \xi_j)^2 \rangle = \frac{1}{2}(2\text{Var}\xi - 2\langle \xi_{j-1}\xi_j \rangle) = 1. \quad (60)$$

Wynika stąd, że zmienne losowe ζ_j podlegają rozkładowi normalnemu o średniej 0 i wariancji 1.

Zgodnie z (55)

$$x_j = \frac{1}{2}(x_{j-1} + x_{j+1}) + \frac{\sigma}{\sqrt{2}}\zeta_j. \quad (61)$$

Inaczej

$$x_j = \frac{1}{2}(x_{j-1} + x_{j+1}) + \alpha\sqrt{\frac{\varepsilon}{2}}\zeta_j. \quad (62)$$

Każde z położeń x_j , obliczonych według wzoru (62), podlega rozkładowi Gaussa. A zatem dla każdej chwili czasowej τ_j , gdzie $j = 1, \dots, d-1$, otrzymujemy pojedynczy jednowymiarowy rozkład Gaussa (49). Wynika stąd, że powtórzenie tej konstrukcji dla każdego wymiaru w przestrzeni trójwymiarowej daje w wyniku $3(d-1)$ -wymiarowy rozkład Gaussa położeń cząstek.

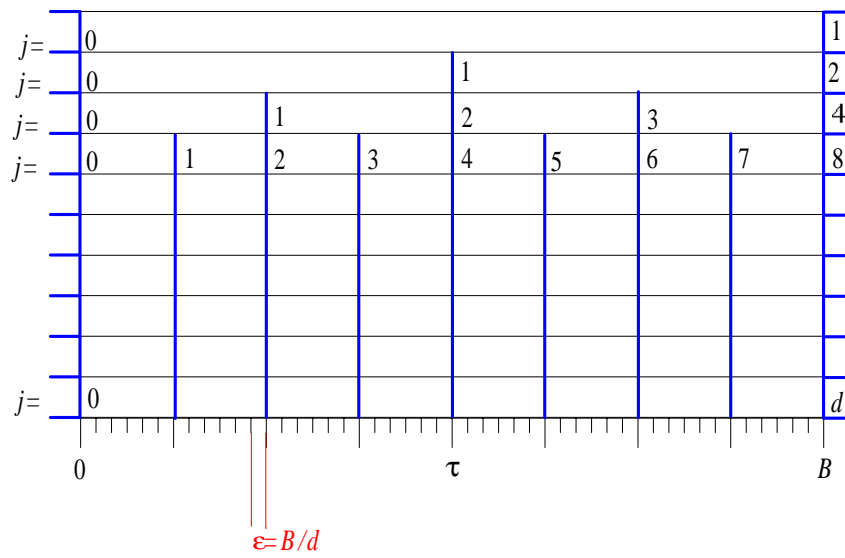
W praktyce obliczeniowej stosujemy podział przedziału $[0, B]$ na równe podprzedziały, np. według następującego schematu: $\varepsilon = B/d$, gdzie $d = 2^n$, a $n = 0, 1, 2, \dots$ oznacza numer kolejnego podziału.

Uwagi

- (1) Przedstawiony powyżej sposób konstrukcji trajektorii brownowskiej zapewnia przechodzenie trajektorii przez ustalone punkty: początkowy x_0 dla $\tau = 0$ i końcowy x_d dla $\tau = B$. Tak wyznaczone trajektorie stosowane są przy numerycznym obliczaniu całek po trajektoriach.
- (2) Konstrukcja trajektorii brownowskiej dostarcza praktycznej metody generowania wielowymiarowego rozkładu Gaussa.

Zastosowania

- (1) Bezpośrednia symulacja komputerowa ruchów Browna.
- (2) Obliczanie całek po trajektoriach metodą Monte Carlo.



Rys. 2. Konstrukcja trajektorii brownowskiej.