

Piotr KARNASIEWICZ  
Katolicki Uniwersytet Lubelski Jana Pawła II

## DOKŁADNE WYZNACZANIE ROZBIEŻNOŚCI

**Streszczenie.** Rozbieżność jest pewną miarą jakości rozkładów próbek wyrażoną jako supremum. Przybliżone wyznaczenie tego supremum polega generalnie na porównywaniu objętości dużej liczby losowo wybranych zbiorów ze z góry obranej rodziny do liczby próbek, które w tych zbiorach się zawierają. W niniejszej pracy została wyprowadzona i przedstawiona numeryczna metoda dokładnego wyznaczenia rozbieżności z użyciem najpopularniejszych rodzin zbiorów.

**Słowa kluczowe:** rozbieżność, próbkowanie

## PRECISELY EVALUATING OF DISCREPANCY

**Summary.** Discrepancy is a measure of quality of samples distribution expressed as a supremum. In general, estimating this supremum relies on comparing volumes of many randomly chosen sets from some given family to quantity of samples belonging to these sets. In this paper, a numerical method of precise determining discrepancy using the most popular families of sets will be introduced.

**Keywords:** discrepancy, sampling

### 1. Próbkowanie obrazu

W grafice komputerowej obraz rzeczywisty traktowany jest jako dwuargumentowa funkcja  $L(x, y)$ , której parametry rzeczywiste  $x$  i  $y$  są współrzędnymi na obrazie. Wartościami funkcji są poziomy luminacji [4]. W celu przetworzenia obrazu do postaci cyfrowej musi on zostać poddany dyskretyzacji, tj. funkcję  $L(x, y)$  o dziedzinie rzeczywistej należy zamienić na funkcję  $L(m, n)$  o dziedzinie dyskretnej. Ilość możliwych wartości, jakie mogą przyjąć parametry  $m$  i  $n$ , nazywana jest rozdzielczością obrazu i najczęściej jest mierzona w pikselach. W celu obliczenia wartości luminacji poszczególnych pikseli używa

się algorytmu Monte Carlo. Metoda ta wymaga próbkowania oryginalnej funkcji obrazu. Na jeden piksel może przypadać jedna lub więcej próbek. Oczywiście im więcej próbek przypada na jeden piksel, tym obraz cyfrowy jest bardziej zbliżony do oryginału. Jakość obrazu cyfrowego zależy nie tylko od ilości próbek, ale również od ich rozkładu. Powstało wiele metod próbkowania. W niniejszej pracy zostaną przedstawione wybrane algorytmy próbkowania o niskiej rozbieżności, tj. próbkowania z zastosowaniem ciągów Haltona, zbiorów Hammersleya, sieci  $(t, m, s)$  oraz ciągów  $(t, s)$ . Oprócz tego do najbardziej znanych metod próbkowania można zaliczyć m. in. próbkowanie warstwowe, metodą łańciskiego kwadratu (lub ogólniej hipersześcianu), z wykorzystaniem tablicy ortogonalnej czy przez wybór najlepszego kandydata. Więcej o wspomnianych metodach próbkowania można znaleźć m. in. w pracach [4] oraz [6].

W grafice komputerowej najczęściej próbkowanymi przestrzeniami są  $[0,1]^2$  w przypadku próbkowania funkcji obrazu, źródeł światła czy pozycji na soczewce oraz  $[0,1]$ , gdy próbkowany jest czas otwarcia przesłony. Dobry rozkład próbek to taki, w którym próbki są rozłożone równomiernie, tj. nie zawierają skupisk próbek oraz dużych pustych miejsc. Próbki ponadto nie mogą być ułożone w regularną siatkę, ponieważ powoduje to aliasing. W wymienionych metodach próbkowania stosowane są różne strategie działania. Część z nich opiera się na losowym generowaniu próbek, część jest całkowicie deterministyczna (tj. dla zadanych parametrów rozkład próbek jest zawsze taki sam). Istnieją też metody mieszane.

Niniejsza praca skupia się na tzw. próbkowaniu o niskiej rozbieżności. Rozbieżność jest miarą określającą jakość rozkładu próbek. Wyznaczanie tej miary najczęściej opiera się na losowym generowaniu dużej liczby zbiorów pewnej rodziny. Wprowadzenie tej losowości powoduje, że rozbieżność nie jest wyznaczana dokładnie. W niniejszej pracy zostanie podana metoda dokładnego wyznaczania rozbieżności dla najczęściej stosowanej rodziny zbiorów. Zostaną również podane wartości rozbieżności dla kilku wybranych rozkładów deterministycznych.

## 2. Definicja rozbieżności

Niech  $N$  oznacza liczbę próbek oraz  $P = \{x_1, \dots, x_N\} \subset [0,1]^s$  – zbiór próbek. Niech ponadto  $B$  będzie pewną rodziną zbiorów takich, że  $\forall_{b \in B} b \subseteq [0,1]^s$ . Rozbieżność zbioru  $P$  ze względu na rodzinę  $B$  definiuje się jako:

$$D_N(B, P) = \sup_{b \in B} \left| \frac{\#(b \cap P)}{N} - \lambda(b) \right|, \quad (1)$$

gdzie  $\#(b \cap P)$  oznacza liczbę próbek należących do zbioru  $b$ , natomiast  $\lambda(b)$  – objętość zbioru  $b$ . Zarówno w przypadku wyznaczania przybliżonych, jak i dokładnych wartości rozbieżności istnieje konieczność wygenerowania dużej liczby zbiorów z danej rodziny. Dla każdego z tych zbiorów należy znaleźć jego objętość i liczbę próbek, które do niego należą. Dlatego rodziny  $B$  powinny być tak dobierane, aby te operacje były łatwe do wykonania.

### 3. Wybrane rodziny zbiorów używane do wyznaczania rozbieżności

Do wyznaczania rozbieżności najczęściej stosowaną rodziną zbiorów jest rodzina

$$B^* = \{[0, v_1] \times \dots \times [0, v_s]\} \quad (2)$$

gdzie  $0 \leq v_i \leq 1$ , dla  $i = 1, \dots, s$ .

Dla każdego zbioru należącego do tej rodziny można bardzo łatwo wyznaczyć objętość oraz podzbiór próbek, które do niego należą. Wadą zbiorów tej rodziny jest ich mała różnorodność – wszystkie są zaczepione w punkcie  $(0, \dots, 0)$ . Inną często stosowaną rodziną jest rodzina

$$B^\square = \{[u_1, v_1] \times \dots \times [u_s, v_s]\} \quad (3)$$

gdzie  $0 \leq u_i \leq v_i \leq 1$ , dla  $i = 1, \dots, s$ .

Jak w poprzednim przypadku używanie tej rodziny zapewnia szybkie działanie algorytmów wyznaczania rozbieżności. Ponadto zbiory rodziny  $B^\square$  są rozmieszczane po całym deseni, co powoduje, że rozbieżność wyznaczona z ich wykorzystaniem daje lepszy pogląd o rozkładzie próbek.

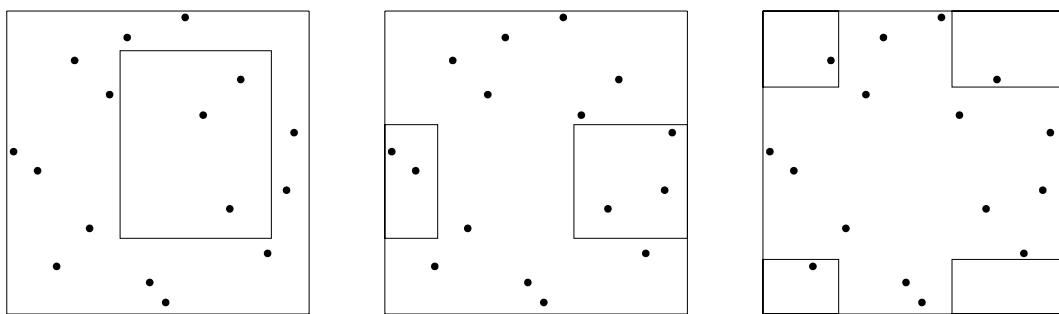
W grafice komputerowej raz wygenerowane rozkłady próbek wykorzystuje się wielokrotnie przez „ustawianie” ich jeden obok drugiego. Może to powodować duże skupiska próbek lub puste miejsca w pobliżu krawędzi użytych deseni. Rozbieżności wyznaczone za pomocą wymienionych do tej pory rodzin nie uwzględniają niedoskonałości desenia złożonego z kilku innych. Takie przypadki uwzględnia rozbieżność wyznaczona za pomocą m.in. następującej rodziny zbiorów:

$$B^+ = \{\mu(u_1, v_1) \times \dots \times \mu(u_s, v_s)\}, \quad (4)$$

gdzie

$$\mu(u_i, v_i) = \begin{cases} [u_i, v_i] & , \text{gd } u_i \leq v_i \\ [0, v_i] \cup [u_i, 0] & , \text{gd } u_i > v_i \end{cases} \quad (5)$$

oraz  $0 \leq u_i \leq v_i \leq 1$  dla  $i = 1, \dots, s$ .



Rys. 1. Przykłady zbiorów rodziny  $B^+$   
 Fig. 1. Examples of sets of family  $B^+$

#### 4. Próbkowanie o niskiej rozbieżności

Niech

$$D_N^*(P) = D(B^*, P). \quad (6)$$

Ciągiem o niskiej rozbieżności nazywany jest nieskończony ciąg punktów  $x_1, x_2, \dots$  taki, że

$$D_N^*(P) = O\left(\frac{(\log N)^s}{N}\right) \quad (7)$$

gdzie  $P = \{x_1, \dots, x_N\}$  dla dowolnego  $N$ . W pracy [3] pokazano, że wartość tę można przyjąć jako najmniejszą możliwą rozbieżność.

Zbiorem o niskiej rozbieżności nazywany jest zbiór  $P = \{x_1, \dots, x_N\}$ , dla którego

$$D_N^*(P) = O\left(\frac{(\log N)^{s-1}}{N}\right) \quad (8)$$

Próbkowanie o niskiej rozbieżności polega na wygenerowaniu ciągów lub zbiorów o niskiej rozbieżności. Jeśli liczba próbek jest znana z góry, to generowane są zbiory o niskiej rozbieżności, jeśli nie jest – generuje się ciągi.

Niech  $d_{i,0}, d_{i,1}, \dots$  będą kolejnymi cyframi rozwinięcia liczby  $i$  w systemie liczbowym o podstawie  $p$ , tj.

$$i = \sum_{k \geq 0} d_{i,k} p^k \quad (9)$$

oraz niech

$$\phi_p(i) = \sum_{k \geq 0} d_{i,k} p^{-1-k} \quad (10)$$

Funkcja  $\phi_p(i)$  nazywana jest odwrotnością pierwiastkową liczby  $i$  w systemie o podstawie  $p$ . Klasycznym przykładem ciągu o niskiej rozbieżności jest ciąg Haltona:

$$x_i = (\phi_2(i), \phi_3(i), \phi_5(i), \dots, \phi_{p_s}(i)), \quad (11)$$

gdzie  $p_k$ , ( $k = 1, 2, \dots, s$ ) są kolejnymi liczbami pierwszymi.

Przykładem zbioru o niskiej rozbieżności jest zbiór Hammersleya:

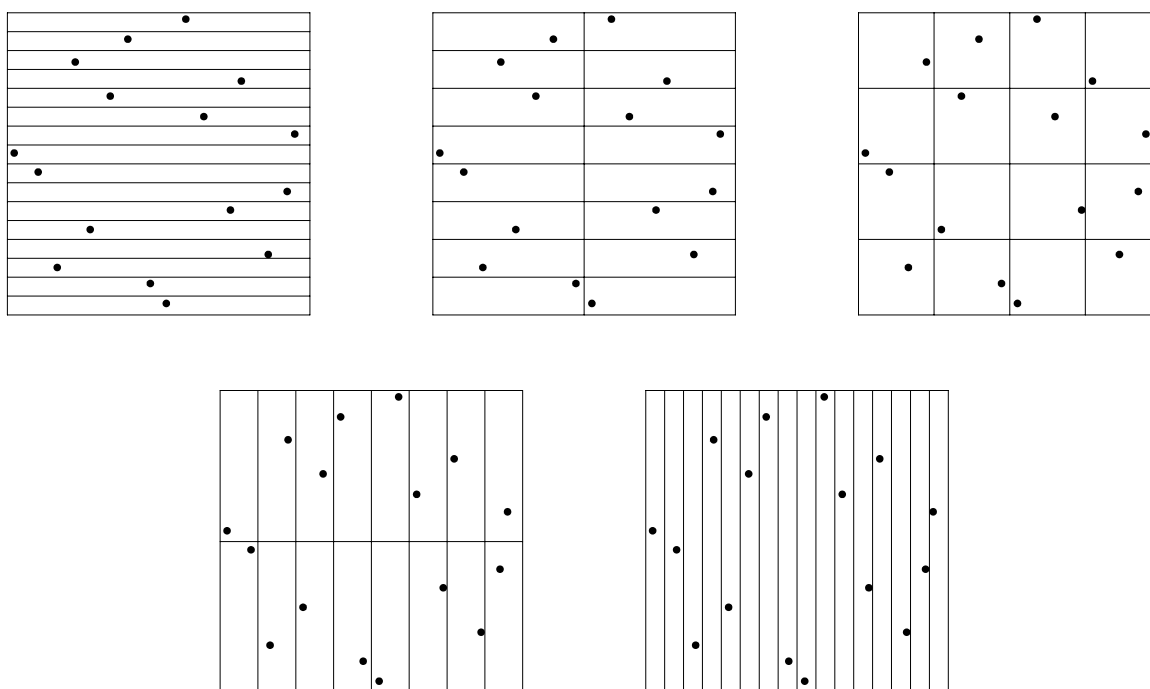
$$x_i = (i/N, \phi_2(i), \phi_3(i), \phi_5(i), \dots, \phi_{p_{s-1}}(i)) \quad (12)$$

gdzie  $i = 1, \dots, N$  oraz  $p_k$  ( $k = 1, 2, \dots, s-1$ ) są kolejnymi liczbami pierwszymi.

Innym przykładem próbkowania o niskiej rozbieżności jest próbkowanie wykorzystujące sieci  $(t, m, s)$  oraz ciągi  $(t, s)$ .  $s$ -wymiarowym przedziałem elementarnym w bazie  $p$  nazywany jest iloczyn kartezjański przedziałów jednowymiarowych o postaci:

$$E = \left[ \frac{t_1}{p^{k_1}}, \frac{t_1+1}{p^{k_1}} \right) \times \left[ \frac{t_2}{p^{k_2}}, \frac{t_2+1}{p^{k_2}} \right) \times \dots \times \left[ \frac{t_s}{p^{k_s}}, \frac{t_s+1}{p^{k_s}} \right), \quad (13)$$

gdzie  $k_j \geq 0$  są liczbami całkowitymi oraz  $0 \leq t_j \leq p^{k_j} - 1$ .



Rys. 2. Przykład sieci  $(0,4,2)$  w bazie 2. W każdym przedziale elementarnym o objętości  $1/16$  znajduje się dokładnie jeden punkt

Fig. 2. An example of  $(0,4,2)$ -net in base 2. In each elementary interval of volume  $1/16$  there is only one point

Sięcią  $(t, m, s)$  w bazie  $p$  nazywany jest zbiór punktów  $P$  o liczbie  $N = p^m$  taki, że każdy przedział elementarny o objętości  $1/p^{m-t}$  zawiera dokładnie  $p^t$  punktów zbioru  $P$ . Na przykład, niech  $P$  będzie siecią  $(0,4,2)$  w bazie 3. Wtedy w kostce  $[0,1]^2$  znajduje się  $3^4 = 81$  punktów zbioru  $P$ . W każdym z przedziałów elementarnych o rozmiarach  $1 \times 1/81$ ,  $1/3 \times 1/27$ ,  $1/27 \times 1/3$ ,  $1/81 \times 1$  znajduje się dokładnie jeden punkt zbioru  $P$ .

Ciągiem  $(t, s)$  nazywany jest nieskończony ciąg punktów  $x_1, x_2, \dots$  taki, że dla każdego  $m \geq 0$  oraz  $k \geq 0$  podciąg  $x_{kp^{m+1}}, \dots, x_{(k+1)p^{m+1}}$  jest siecią  $(t, m, s)$  w bazie  $p$ . W szczególności każdy zbiór  $x_1, \dots, x_N$  pierwszych  $N = p^m$  elementów tego ciągu jest siecią  $(t, m, s)$ .

Przykładem ciągu  $(0, 2)$  jest ciąg  $x_i = (a_i, b_i)$  taki, że  $a_i$  są kolejnymi elementami ciągu van der Corputa, natomiast  $b_i$  – elementami ciągu Sobola. Ciąg van der Corputa jest ciągiem odwrotności pierwiastkowych w systemie o podstawie  $p = 2$ :

$$\frac{1}{2}, \frac{1}{4}, \frac{3}{4}, \frac{1}{8}, \frac{5}{8}, \frac{3}{8}, \frac{7}{8}, \dots \quad (14)$$

Ciąg Sobola opiera się na zbiorze tzw. współczynników kierunkowych  $\{v_j\}$ , tj. takich liczb, że  $v_j = m_j / 2^j$ , gdzie  $m_j$  są nieparzystymi liczbami całkowitymi mniejszymi od  $2^j$ , a  $v_j$  są dobierane tak, aby spełniały pewien związek rekurencyjny dla współczynników wielomianu pierwotnego w ciele Galois  $G(2)$ :

$$f(z) = z^p + c_1 z^{p-1} + \dots + c_{p-1} z + c_p, \quad (15)$$

gdzie  $p$  jest liczbą pierwszą, a współczynniki  $c_i \in \{0, 1\}$  dla  $i = 1, \dots, p$ . Dla  $j > p$  tym związkiem rekurencyjnym jest:

$$v_j = c_1 v_{j-1} \oplus c_2 v_{j-2} \oplus \dots \oplus c_p v_{j-p} \oplus \lfloor v_{j-p} / 2^p \rfloor, \quad (16)$$

gdzie  $\oplus$  oznacza bitową różnicę symetryczną. Odpowiednim związkiem rekurencyjnym dla  $m_j$  jest

$$m_j = 2c_1 m_{j-1} \oplus 2^2 c_2 m_{j-2} \oplus \dots \oplus 2^p c_p m_{j-p} \oplus m_{j-p} \quad (17)$$

Więcej na ten temat można przeczytać w pracy [2].



Rys. 3. Obrazy wygenerowane za pomocą próbkowania warstwowego (po lewej) i próbkowania  $(0, 2)$  (po prawej). W obydwu przypadkach były wybierane 4 próbki na piksel. Na obrazie po lewej artefakty i szumy, zwłaszcza w okolicach cienia, są znacznie bardziej widoczne

Fig. 3. Images generated by stratified sampling (on left) and  $(0, 2)$  sampling (on right). 4 samples per pixel was chosen in both cases. On left image artifacts and noise, especially near the shadow are much more visible

Próbkowania o niskiej rozbieżności są jednymi z najlepszych ponieważ zapewniają bardzo dobry rozkład próbek. Ponadto algorytmy generujące takie rozkłady są na tyle szybkie, że mogą pracować w czasie rzeczywistym. Na rysunku 3 widać dwa obrazy wygenerowane za pomocą pakietu `pbrt` opisanego w książce [4]. W obrazie po lewej zostało

użyte znane próbkowanie warstwowe, polegające na podzieleniu obszaru próbkowania na prostokątne części i losowym wybraniu w każdej z nich po jednej próbce. Na obrazku po prawej zastosowano próbkowanie z użyciem ciągu (0,2). Różnice są widoczne.

## 5. Wyprowadzenie algorytmu dokładnego wyznaczania rozbieżności

Niech  $a = (a_1, \dots, a_s) \in [0,1]^s$  oraz  $b = (b_1, \dots, b_s) \in [0,1]^s$  będą takimi punktami przestrzeni, że  $a_1 \leq b_1, \dots, a_s \leq b_s$ .  $s$ -wymiarową kostką (przedziałem)  $R(a,b)$  nazywany jest zbiór  $R(a,b) = [a_1, b_1] \times \dots \times [a_s, b_s]$ .

Niech  $P = \{x_1, \dots, x_N\} \subset [0,1]^s$  będzie zbiorem próbek,  $R^j \subseteq [0,1]^s$  – dowolną kostką przestrzeni  $[0,1]^s$  oraz niech  $P^j = R^j \cap P$ . Kostki, które zawierają wszystkie (i tylko te) próbki zbioru  $P^j$ , tworzą rodzinę  $\mathfrak{R}^j = \{R : R \cap P = P^j\}$ . Rodzina  $\mathfrak{R}^j$  jest klasą abstrakcji generowaną przez kostkę  $R^j$ . Fakt ten będzie oznaczany przez  $\mathfrak{R}^j = [R^j]$ . Istnieją kostki  $R_{\inf}^j$  oraz  $R_{\sup}^j$  takie, że

$$\lambda(R_{\inf}^j) = \inf_{R \in \mathfrak{R}^j} \lambda(R) \text{ oraz } \lambda(R_{\sup}^j) = \sup_{R \in \mathfrak{R}^j} \lambda(R), \quad (18)$$

gdzie  $\lambda(R)$  oznacza objętość kostki  $R$ . Kostkę  $R_{\inf}^j$  można łatwo skonstruować, mianowicie  $R_{\inf}^j = R(a_{\inf}, b_{\inf})$ , gdzie

$$\begin{aligned} a_{\inf} &= (\min_{1 \leq i \leq N} \{x_{i_1}\}, \dots, \min_{1 \leq i \leq N} \{x_{i_s}\}) \\ b_{\inf} &= (\max_{1 \leq i \leq N} \{x_{i_1}\}, \dots, \max_{1 \leq i \leq N} \{x_{i_s}\}) \end{aligned} \quad (19)$$

gdzie  $x_i = (x_{i_1}, \dots, x_{i_s}) \in P^j$  dla  $i = 1, \dots, N$ . Wyznaczenie kostki  $R_{\sup}^j$  polega na wyborze największej spośród pewnej ilości kostek skonstruowanych w określony sposób. Budowanie tych kostek polega na „przesuwaniu” kolejnych ścian kostki  $R_{\inf}^j$  do momentu aż dotrze ona do punktu spoza zbioru  $P^j$  lub do ściany kostki  $[0,1]^s$ . Oczywiście w zależności od wyboru kolejności przesuwanych ścian zostaną uzyskane różne kostki.  $R_{\sup}^j$  jest po prostu największą z nich (rys. 4).

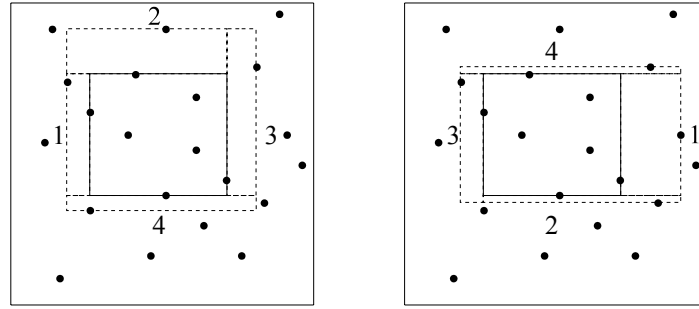
Można łatwo wykazać, że

$$\sup_{R \in \mathfrak{R}^j} \left| \frac{\#P^j}{N} - \lambda(R) \right| = \max \left\{ \left| \frac{\#P^j}{N} - \lambda(R_{\inf}^j) \right|, \left| \frac{\#P^j}{N} - \lambda(R_{\sup}^j) \right| \right\}, \quad (20)$$

gdzie  $\#P^j$  oznacza liczbę elementów zbioru  $P^j$ . Niech  $p = \#P^j / N$ . Wtedy funkcja

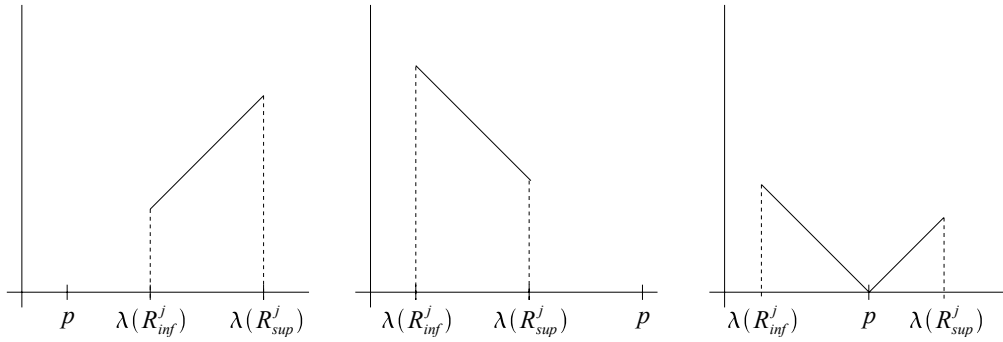
$$f : [\lambda(R_{\inf}^j), \lambda(R_{\sup}^j)] \mapsto [0,1], f(x) = |p - x| \quad (21)$$

zawsze osiąga maksimum na jednym z krańców swojej dziedziny (patrz rys. 5).



Rys. 4. Przykład budowania kostek, spośród których wybiera się  $R_{sup}^j$ . Na rysunku widać, że w zależności od wyboru kolejności ścian otrzymuje się różne kostki

Fig. 4. An example of building boxes, among which  $R_{sup}^j$  is chosen. In the picture we can see that depending on the choice of the sequence of walls the result boxes are different



Rys. 5. Funkcja  $f(x)$  (21) zawsze osiąga maksimum na jednym z krańców przedziału dziedziny. Na rysunku są przedstawione przypadki dla  $p < \lambda(R_{inf}^j)$ ,  $\lambda(R_{sup}^j) < p$  oraz dla  $\lambda(R_{sup}^j) < p < \lambda(R_{inf}^j)$

Fig. 5. The function  $f(x)$  (21) always reaches its maximum value in one of the bounds if the segment of its domain. The picture presents cases for  $p < \lambda(R_{inf}^j)$ ,  $\lambda(R_{sup}^j) < p$  and  $\lambda(R_{sup}^j) < p < \lambda(R_{inf}^j)$

Ze względu na dowolność wyboru kostki  $R^j$  oraz na podstawie (20):

$$\begin{aligned}
 D_N^\square(P) &= \sup_{R^j \subseteq [0,1]^s} \left| \frac{\#P^j}{N} - \lambda(R^j) \right| = \\
 &= \sup_{R^j \subseteq [0,1]^s} \left\{ \sup_{R^j \subseteq \mathfrak{R}^j} \left| \frac{\#P^j}{N} - \lambda(R^j) \right| \right\} = \\
 &= \sup_{R^j \subseteq [0,1]^s} \left\{ \max \left\{ \left| \frac{\#P^j}{N} - \lambda(R_{inf}^j) \right|, \left| \frac{\#P^j}{N} - \lambda(R_{sup}^j) \right| \right\} \right\}
 \end{aligned} \tag{22}$$

gdzie  $D_N^\square(P) = D_N(B^\square, P)$ .

Należy zauważyć, że  $R_{inf}^j \in \mathfrak{R}^j$ . Ponieważ do brzegów kostki  $R_{sup}^j$  należą punkty spoza  $P^j$ , więc na ogół  $R_{sup}^j \notin \mathfrak{R}^j$  (poza przypadkiem, gdy  $P^j = P$ ). Cechą charakterystyczną kostek  $R_{inf}^j$  oraz  $R_{sup}^j$  jest to, że każda z ich ścian albo zawiera punkty zbioru  $P$ , albo zawiera się w którejs z ścian kostki  $[0,1]^s$ .



Dowolnie wybrana kostka spełniająca te warunki może być:

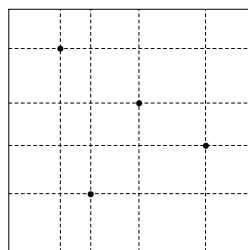
- kostką  $R_{\text{inf}}^j$  dla  $P^j = R_{\text{inf}}^j \cap P$  lub
- kostką  $R_{\text{sup}}^j$  dla  $P^j = \text{int}(R_{\text{sup}}^j) \cap P$ ,

gdzie  $\text{int}(R_{\text{sup}}^j)$  oznacza wnętrze kostki  $R_{\text{sup}}^j$ . Zatem aby dokładnie wyznaczyć  $D_N^\square(P)$ , wystarczy znaleźć wszystkie możliwe kostki  $R$ , które zawierają na swoich ścianach punkty zbioru  $P$  lub są styczne ze ścianami kostki  $[0,1]^s$  i dla każdej z tych kostek obliczyć wartości

$$\left| \frac{\#(R \cap P)}{\#P} - \lambda(R) \right| \text{ oraz } \left| \frac{\#(\text{int}(R) \cap P)}{\#P} - \lambda(R) \right| \quad (23)$$

$D_N^\square(P)$  jest równa największej spośród tych wartości.

Najprostszy algorytm numerycznego wyznaczenia wartości  $D_N^\square(P)$  polega na podzieleniu każdej ze ścian kostki  $[0,1]^s$  na  $N+1$  części, wyznaczonych przez odpowiednie współrzędne każdego z punktów zbioru  $P$ , tak jak na rys. 6.



Rys. 6. Dwuwymiarowy przykład podziału kostki  $[0,1]^2$   
Fig. 6. Twodimensional example of dividing box  $[0,1]^2$

Mając tak podzielony zbiór  $[0,1]^s$ , wystarczy sprawdzić wszystkie możliwe kostki o ścianach wyznaczonych przez odpowiednie współrzędne punktów. Takich kostek jest

$$((N+1) + N + (N-1) + \dots + 1)^s = \left( \frac{(N+1)(N+2)}{2} \right)^s = O(N^{2s}) \quad (24)$$

W przypadku obliczania rozbieżności  $D_N^*(P)$ , tj. rozbieżności względem rodziny

$$B^* = \{[0, v_1] \times \dots \times [0, v_s]\} \quad (25)$$

wystarczy rozpatrzyć  $(N+1)^s = O(N^s)$  kostek.

## 6. Przykładowe wartości rozbieżności

Zostanie pokazanych kilka dokładnych wartości rozbieżności  $D_N^*$  oraz  $D_N^\square$  dla dwuwymiarowych próbkowań deterministycznych, tj. dla metod z wykorzystaniem ciągów Haltona i zbiorów Hammersleya. Dla ułatwienia zostanie przyjęte następujące oznaczenie: niech  $\bar{R}$

oznacza kostkę, dla której zostało osiągnięte supremum z równania (20). W kolejnych wierszach tabel zostaną zaprezentowane:

- Wartość dokładna rozbieżności dla wybranego rozkładu i liczby próbek podana z dokładnością do 4 miejsc po przecinku.
- Informacja o tym, czy kostka  $\bar{R}$  jest kostką  $R_{\text{inf}}^j$  czy  $R_{\text{sup}}^j$ .
- Współrzędne kostki  $\bar{R}$ .
- Ilość próbek, dla których zostało osiągnięte supremum. Innymi słowy, będzie to liczba próbek zawierających się w  $R_{\text{inf}}^j$  lub we wnętrzu  $R_{\text{sup}}^j$  w zależności od tego, czym jest  $\bar{R}$ .
- Przybliżona wartość rozbieżności zbiorów podana z dokładnością do 4 miejsc po przecinku. Na ogół wartości przybliżone były wyznaczane za pomocą takiej samej liczby zbiorów jak w przypadku wartości dokładnych. Jeśli liczba ta była mniejsza niż 100 000, było generowane 100 000 zbiorów.

Tabela 1

Rozbieżność  $D_N^*$  dla próbkowania z wykorzystaniem ciągów Haltona

	$D_{32}^*$	$D_{64}^*$	$D_{128}^*$	$D_{256}^*$
Wartość dokładna	0.0995	0.0521	0.0332	0.0188
sup/inf	inf	inf	inf	inf
$\bar{R}$	$[0,0.88] \times [0,0.81]$	$[0,0.84] \times [0,0.72]$	$[0,0.66] \times [0,0.94]$	$[0,0.82] \times [0,0.78]$
$\#P^j$	26	42	84	169
Wartość przybl.	0.0928	0.0510	0.0318	0.0168

Tabela 2

Rozbieżność  $D_N^\square$  dla próbkowania z wykorzystaniem ciągów Haltona

	$D_{32}^\square$	$D_{64}^\square$	$D_{128}^\square$	$D_{256}^\square$
Wartość dokładna	0.1493	0.0897	0.0523	0.0307
sup/inf	sup	sup	sup	sup
$\bar{R}$	$[0.16,0.59] \times [0.37,0.93]$	$[0.17,0.81] \times [0.42,0.90]$	$[0.46,0.78] \times [0.15,0.88]$	$[0.17,0.93] \times [0.39,0.89]$
$\#P^j$	3	14	23	89
Wartość przybl.	0.1223	0.0808	0.0475	0.0275

Tabela 3

Rozbieżność  $D_N^*$  dla próbkowania z wykorzystaniem zbiorów Hammersleya

	$D_{32}^*$	$D_{64}^*$	$D_{128}^*$	$D_{256}^*$
Wartość dokładna	0.0664	0.0381	0.0217	0.0121
sup/inf	inf	inf	inf	inf
$\bar{R}$	$[0,0.69] \times [0,0.81]$	$[0,0.66] \times [0,0.66]$	$[0,0.67] \times [0,0.83]$	$[0,0.66] \times [0,0.66]$
$\#P^j$	20	30	74	116
Wartość przybl.	0.0660	0.0367	0.0194	0.0112

Tabela 4

Rozbieżność  $D_N^\square$  dla próbkowania z wykorzystaniem zbiorów Hammersleya

	$D_{32}^\square$	$D_{64}^\square$	$D_{128}^\square$	$D_{256}^\square$
Wartość dokładna	0.1162	0.0681	0.0375	0.0204
sup/inf	inf	inf	inf	inf
$\bar{R}$	$[0.31,0.94] \times$ $[0.31,0.94]$	$[0.14,0.84] \times$ $[0.14,0.84]$	$[0.07,0.92] \times$ $[0.13,0.86]$	$[0.07,0.93] \times$ $[0.07,0.93]$
$\#P^j$	30	36	84	196
Wartość przybl.	0.0956	0.0542	0.0327	0.0181

## 7. Podsumowanie

Przybliżone wartości rozbieżności  $D_N^*$  czy  $D_N^\square$  obliczane są w taki sposób, że losuje się wiele kostek  $R$  rodziny  $B^*$  lub  $B^\square$  odpowiednio, dla których oblicza się wartości

$$\left| \frac{\#(R \cap P)}{N} - \lambda(R) \right| \quad (26)$$

Za rozbieżność przyjmuje się największą z tych wartości. Ponieważ w taki sposób przybliżane jest supremum, więc wartość przybliżona zawsze będzie mniejsza (lub równa) od wartości dokładnej. Obserwację tę potwierdzają wyniki testów.

Zaproponowany algorytm wyznaczania dokładnej wartości rozbieżności jest algorytmem bardzo wolnym, ponieważ wymaga wygenerowania bardzo dużej liczby zbiorów. Przybliżone wartości rozbieżności można wyznaczyć po wygenerowaniu znacznie mniejszej ich liczby, dlatego najczęściej algorytmy wyznaczania przybliżonej wartości rozbieżności są szybsze. Oba algorytmy mają podobny czas działania dla takiej samej liczby zbiorów.

Wydaje się, że zaproponowany algorytm dokładnego wyznaczania rozbieżności można znacznie przyspieszyć, ponieważ sprawdzane są kostki, o których z góry wiadomo, że nie może być dla nich spełnione supremum. Przykładowo niech  $P = \{(1/3,1/3), (2/3,2/3)\}$ . W podanym algorytmie będzie np. sprawdzana kostka  $[0,1/3] \times [2/3,0]$ , dla której supremum na pewno nie będzie osiągnięte, ponieważ dwie z jej ścian ani nie zawierają próbek, ani nie są styczne do krawędzi kostki  $[0,1]^2$ .

## LITERATURA

1. Foley J. D., van Dam A., Feiner S. K., Hughes J. F., Philips R. L.: Wprowadzenie do grafiki komputerowej. WNT, Warszawa 2001.

2. Gentle J. E.: Random Number Generation and Monte Carlo Methods. Springer, Berlin 1998.
3. Niederreiter H.: Random Number Generation and Quasi-Monte Carlo Methods. Society for Industrial and Applied Mathematics, 1992.
4. Pharr M., Humphreys G.: Physically Based Rendering. From theory to implementation. Elsevier, 2003.
5. Tadeusiewicz R., Korohoda P.: Komputerowa analiza i przetwarzanie obrazów. Wydawnictwo Fundacji Postępu Komunikacji, 1997.
6. Veach E.: Robust Monte Carlo methods for light transport simulation. Rozprawa doktorska, Stanford University, 1997.

Recenzent: Dr hab. inż. Maria Pietruszka, prof. Pol. Łódzkiej

Wpłynęło do Redakcji 3 czerwca 2008 r.

### **Abstract**

Discrepancy is a measure of quality of samples distribution expressed as a supremum. In general, estimating this supremum relies on comparing volumes of many randomly chosen sets from some given family to quantity of samples belonging to these sets. In this paper, a numerical method of precise determinating discrepancy using the most popular families of sets will be introduced. The proof is based on division of the whole family of sets to subfamilies for which discrepancy is easily evaluated precisely. It is shown that the sets for which these local suprema are reached have some common properties. Thus, to evaluate discrepancy precisely it is only necessary to check sets, which has these properties.

### **Adres**

Piotr KARNASIEWICZ: ul. Husarska 8/1, 20-555 Lublin, Polska, [pikar@kul.lublin.pl](mailto:pikar@kul.lublin.pl)