

Náhodné rozmísťovanie bodů v rovině

Josef Pelikán

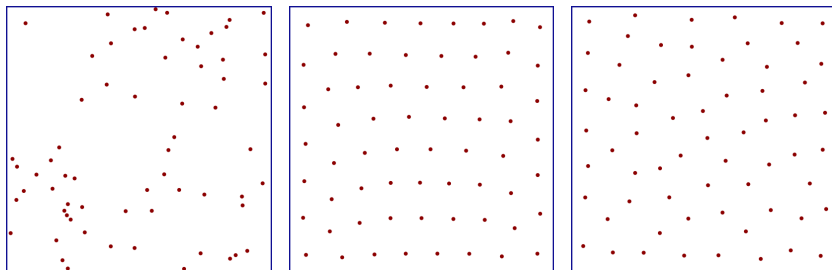
Computer Graphics Group, KSVI MFF, Charles University
Malostranské nám. 25, 11800 Praha 1
pepca@cgg.mff.cuni.cz

Abstrakt. V různých oborech je potřeba pokrývat část roviny nebo plochy souborem náhodných bodových vzorků. Obvykle se požaduje, aby vzorky pokrývaly oblast nahodile, ale rovnoměrně (či s předem danou hustotou pravděpodobnosti), aby nebyly v souboru na první pohled patrné žádné pravidelnosti či vzory. Příspěvek se zabývá nejdůležitějšími aspekty náhodného vzorkování, představuje nejznámější algoritmy, metodiku jejich porovnávání a hodnocení výsledků (diskrepance, spektrální analýza). Z aplikací bude zmíněno vzorkování pro Monte-Carlo integraci, rozmísťování objektů v interaktivních aplikacích, hrách nebo simulacích, algoritmy pro tečkování (tupování) a pŕltónování v tiskařství.

Klíčová slova: vzorkování, náhodné rozmísťování, jittering, Poissonovo diskové vzorkování, tupování, pŕltónování, Monte-Carlo integrace, diskrepance, periodogram, Haltonova sekvence, Hammersley sekvence, Lloydův algoritmus, Mitchellův algoritmus, kapacitně omezené vzorkování.

1 Úvod

V mnoha oblastech počítačové grafiky a aplikované geometrie se setkáváme s potřebou umístit do nějaké oblasti množinu bodů nebo malých objektů tak, aby soubor pokrýval celou oblast, ale současně nesmí být patrné žádné rušivé pravidelnosti (viz obrázek 1). Při Monte-Carlo kvadratuře v oboru syntézy obrazu (tzv. „rendering“) či v jiných oblastech numerické matematiky mohou být kladeny podobné požadavky na soubor bodových vzorků, zde je hlavní podmínkou rovnoměrnost pokrytí dané oblasti.



Obrázek 1: Příliš „divoké“, příliš pravidelné a uspokojivé rozložení.

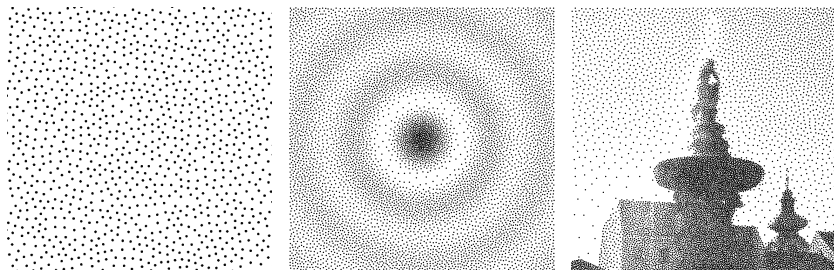
V tomto příspěvku budeme studovat vlastnosti mnoha známých rozmísťovacích (vzorkovacích) metod, zavedeme některá objektivní kritéria

pro posuzování kvality souboru vzorků a pomocí vlastních implementací provedeme měření a závěrečné porovnání.

2 Motivace a aplikace

V této kapitole ještě podrobněji rozebereme podmínky a požadavky kladené na rozmisťovací algoritmy. V některých oblastech se požaduje, aby bylo rozložení vzorků řízeno nějakou hustotou pravděpodobnosti (vzorky nejsou rozloženy rovnoměrně ve smyslu běžné uniformní metriky), viz ukázky na obrázku 2. Příklady takových aplikací:

- simulace malířské techniky tupování („stippling“), viz např. [1]
- pŕltónovací techniky v tiskařství, viz např. [2] nebo [3]
- vzorkování podle důležitosti („importance sampling“) v Monte-Carlo metodách syntézy obrazu, viz např. [4] nebo [5]



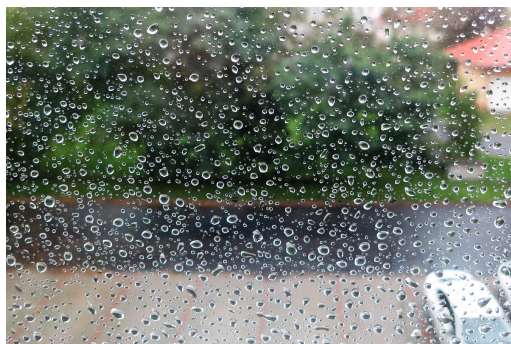
Obrázek 2: Konstantní hustota, hustota daná vzorcem nebo obrázkem.

Výčet několika dalších aplikačních oblastí, kde může nebo nemusí hrát roli neuniformní hustota vzorkování:

- simulace přírodních jevů – růst kolonie buněk, rozmisťování stromů a keřů, apod.
- obecněji – napodobení výsledků složitých fyzikálních procesů, kde hraje roli vzdálenost sousedních vzorků, viz třeba obrázek 3
- vizuálně příjemné rozmisťování objektů v architektuře, videohrách, apod.
- generování sítí pro metody konečných prvků (FEM)

Pro Monte-Carlo kvadraturu je důležité, aby každý bod domény měl šanci být vybrán, pokud v něm samozřejmě má hustota pravděpodobnosti kladnou hodnotu. Z toho potom plyne nestrannost odhadu integrálu („unbiased estimation“). Některé metody vzorkování se tedy pro numerickou integraci nehodí, zejména musíme být opatrní při použití deterministických postupů.

Budeme-li se zabývat estetickými aspekty vzorkování, v první řadě musíme vyloučit všechny rušivé prvky jako třeba výskyt nějakého vzoru



Obrázek 3: Povrchové napětí vody ovlivňuje cílové rozmístění kapek.

nebo opakování. Často je nežádoucí, aby vzorky tvořily shluky, to se dokonce v některých aplikačních oblastech chápe jako nepřijatelné (například při rozmísťování objektů nenulové velikosti se žádné dva objekty nesmějí překrývat). Příklady příjemně vypadajících rozložení najdete např. na obrázku 2.

Je obtížné matematicky definovat estetická kritéria, přesto se v příspěvku budeme snažit uvést několik nejpoužívanějších pravidel a metod vyhodnocování kvality vzorkovacích algoritmů. Některá hodnotící kritéria budou do jisté míry odrážet i estetickou stránku věci. Podrobnosti obsahuje následující kapitola.

Zatím tedy bez dalšího upřesnění přehled požadovaných vlastností náhodného rozložení:

- úplné pokrytí zdrojové oblasti – „každý bod má šanci“
- rovnoměrné pokrytí zdrojové oblasti (případně dle dané hustoty pravděpodobnosti)
- absence pravidelnosti, viditelného systému nebo opakování
- další podmínky – estetika, minimální vzdálenost vzorků, apod.

3 Hodnocení kvality vzorkování

Kapitola popisuje některé přístupy, jak objektivně posoudit kvalitu vzorkovacího (rozmísťovacího) algoritmu. Pro jednoduchost se budeme zabývat pouze konstantní hustotou pravděpodobnosti, tj. rovnoměrným vzorkováním. Metody se celkem snadno dají modifikovat i pro nerovnoměrnou hustotu pravděpodobnosti.

3.1 Diskrepance

Rovnoměrnost pokrytí zdrojové oblasti množinou vzorků je velmi důležitá ve většině aplikací vzorkování, intenzivně se zkoumala především v numerické matematice zabývající se Monte-Carlo integrací [6]. V počítačové

grafice provedl P. Shirley první seriózní analýzy a ukázal přímou souvislost nízké diskrepance vzorkování s kvalitou (malým podílem šumu) počítačově vytvářených obrázků [7].

Pojem *diskrepance* zavedl S. K. Zaremba jako míru nerovnoměrnosti pokrytí jednotkového čtverce $[0, 1]^2$ množinou bodových vzorků. Počet vzorků ležících uvnitř obdélníka $[0, a] \times [0, b]$ označíme $n(a, b)$. Očekávaný počet by se měl blížit hodnotě abN , kde N je počet vzorků v celém čtverci. Zaremba v [6] definoval diskrepanci D_∞^z jako supremum absolutní odchylky $d(a, b)$ přes všechny možné hodnoty a a b :

$$d(a, b) = \left| ab - \frac{n(a, b)}{N} \right|$$

$$D_\infty^z = \sup_{a, b \in [0, 1]} d(a, b)$$

Současně Zaremba upozornil na alternativní možnost definice diskrepance pomocí střední kvadratické hodnoty místo suprema:

$$D_2^z = \sqrt{\int_0^1 \int_0^1 d(a, b)^2 da db}$$

V předchozích definicích mají všechny obdélníky zafixovaný jeden z vrcholů ve vrcholu čtverce. To způsobuje nepřírozenou závislost diskrepance na orientaci vzorkovacího prostoru. Stroud v [8] navrhl použít všechny možné obdélníky s vodorovnými a svislými hranami. Vede to sice ke složitějšímu výpočtu, ale diskrepance je objektivnější a invariantní vzhledem k symetriím:

$$d(a, b, c, d) = \left| (a - c)(b - d) - \frac{n(a, b, c, d)}{N} \right|$$

$$D_\infty^s = \sup_{a, b \in [0, 1], 0 < c < a, 0 < d < b} d(a, b, c, d)$$

$$D_2^s = \sqrt{\int_0^1 \int_0^1 \int_0^b \int_0^a d(a, b, c, d)^2 dc dd da db}$$

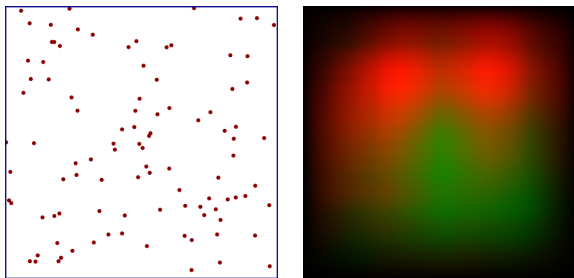
Po zavedení těchto pojmů a prokázání jejich souvislosti s malým rozptylem Monte-Carlo odhadů integrálů se mnoho matematiků snažilo navrhnout nové speciální vzorkovací sekvence, které mají malou diskrepanci [9]. Některé z nich ([10], [11], [12]) se velmi osvědčily a my je v kapitole 4 zahrneme do našeho přehledu.

3.1.1 Výpočet a vizualizace diskrepance

Výpočet diskrepance podle definice může být značně neefektivní. Pro výpočet D_∞ forem diskrepance byly vyvinuty efektivní algoritmy ([13], obecně viz [14]). Pro diskrepance se střední kvadratickou hodnotou je v podstatě nutné procházet danou množinu intervalů. Zde pomáhá diskretizace prostoru a používání metod dynamického programování, například součtových tabulek („Summed Area Table“ viz [15]).

Pro vizuální reprezentaci diskrepance konkrétního souboru vzorků jsme navrhli tzv. „Debit-Credit“ graf (DrCr). Kumulativně se pomocí dvou barev zvýrazňují regiony zdrojové oblasti, kde vzorky chybějí nebo naopak které obsahují vzorků přebytek. Viz obrázek 4.

Datová struktura DrCr může rovněž sloužit k optimalizaci dané sady vzorků, algoritmus postavený na tomto principu je však zatím málo efektivní.



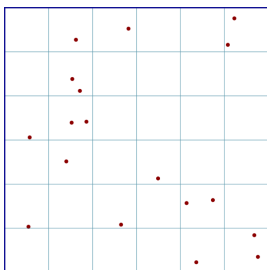
Obrázek 4: Sada náhodných vzorků a DrCr vizualizace diskrepance.

3.2 Shluky

Nyní si dovolíme malou odbočku dávající odpověď na zřejmou otázku „Jsou shluky přirozenou vlastností náhodných rozdělání?“. Při pozorování souboru *nezávislých náhodných* vzorků, jaký je například zobrazen v levé části obrázku 1, nás překvapí mnoho shluků a naopak poměrně velké oblasti bez pokrytí.

Je zajímavé, že podobnými pochybnostmi se lidé trápili již několik století v různých kontextech, od statistik úmrtí vojáků důsledkem kopnutí koněm, přes zkoumání míst dopadu německých raket V2 na Londýn za druhé světové války, až v současnosti třeba k analýzám teroristických útoků nebo frekvencím příchodu pacienta do ordinace lékaře. Tyto úlohy jsou tak významné, že se někdy odpovídající Poissonovo rozdělení nazývá též „zákonem řídkých událostí“ („Law of Rare Events“).

Pro analýzu výskytu shluků můžeme použít rozdělení oblasti na množství stejně velkých intervalů. Jednotkový čtverec například rozdělíme na



$\mu = 0.5$	dosaženo	teorie
$P(X = 0)$	0.639	0.607
$P(X = 1)$	0.222	0.303
$P(X \geq 2)$	0.139	0.090

Obrázek 5: Poissonovo rozdělení při vrhání šípek na čtvercový terč.

$K \times K$ menších čtverečků a budeme zkoumat, jak se do malých čtverečků budou vzorky umisťovat. Příklad typického výsledku pro $K = 6$ a počet vzorků $N = 18$ vidíme na obrázku 18. Každý čtvereček má pravděpodobnost zásahu některým vzorkem $\mu = 0.5$, avšak více než polovina vzorků se sdužila do dvojic a celých 23 čtverečků z 36 zůstalo nezasaženo.

Naše intuice zde selhává, tento výsledek je totiž úplně v pořádku. Při náhodném výběru z rovnoměrného rozdělení mají počty zásahů jednotlivých čtverečků vyhovovat Poissonovu rozdělení, pravý sloupec tabulky obsahuje teoreticky spočítané hodnoty a ty skutečně dosažené se od nich příliš neliší.

3.3 Spektrální charakteristiky

Dalším používaným přístupem při vyhodnocování kvality rozložení bodových vzorků je spektrální analýza. Jedná se o zkoumání distribuce vzdáleností mezi jednotlivými vzorky, hlavně mezi nejbližšími sousedy. Chápeme-li množinu vzorků jako reálnou funkci se zdrojovou oblastí jako definičním oborem, můžeme přímo použít některou z populárních spektrálních transformací, nejčastěji Fourierovu transformaci. Tzv. „power spectrum“ ukazuje, jak často se jednotlivé prostorové frekvence v signálu vyskytují a my budeme pracovat nejen přímo s tímto spektrem, ale i s některými odvozenými charakteristikami.

Zajímavostí je, že již A. Schuster v roce 1898 ([16]) při zkoumání periodických jevů navrhl tzv. „periodogram“, což je obdoba dnešního výkonového spektra.

Připomeňme úvodem několik definic (částečně dle [17]). Množina vzorků \mathbf{S} ve 2D:

$$\mathbf{S} = \{ \mathbf{s}_k \}_{k=1}^N = \{ [x_k, y_k] \}_{k=1}^N$$

Funkci rozložení vzorků (konkrétní polohy N bodových vzorků v rovině) označíme s :

$$s(x, y) = \sum_{k=1}^N \delta(x - x_k, y - y_k)$$

kde δ je *Diracův impuls* (reálná funkce nenulová jen v jednom bodě, mající však jednotkový integrál přes celý definiční obor). Fourierovu transformaci této speciální funkce můžeme zjednodušit na

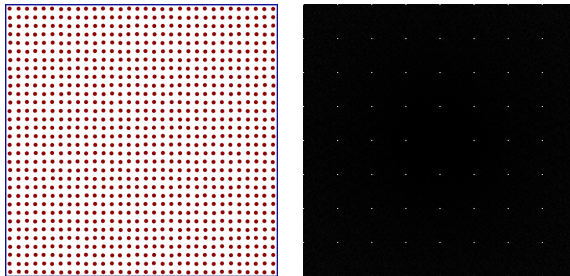
$$F(\mathbf{f}) = \sum_{k=1}^N e^{-2\pi i(\mathbf{f} \cdot \mathbf{s}_k)}$$

kde $\mathbf{f} = [f_x, f_y]$ je frekvenční vektor skalárně násobený s vektorem souřadnic vzorku, obvykle se počítají celočíselné násobky ve vhodné zvolené oblasti kolem počátku (například $\mathbf{f} \in [-255, 255]^2$).

Při frekvenčních analýzách se budeme zabývat *výkonovým spektrem* $P(\mathbf{f})$ („power spectrum“, viz [2], [17]), které je definováno jako druhá mocnina absolutní hodnoty Fourierovy transformace

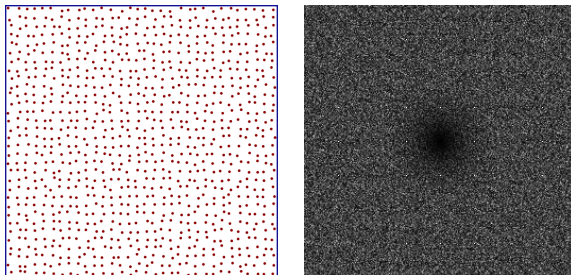
$$P(\mathbf{f}) = |F(\mathbf{f})|^2 = \frac{1}{N} \left(\sum_{k=1}^N \cos(2\pi \mathbf{f} \cdot \mathbf{s}_k) \right)^2 + \frac{1}{N} \left(\sum_{k=1}^N \sin(2\pi \mathbf{f} \cdot \mathbf{s}_k) \right)^2$$

Příklady výkonových spekter jsou na obrázcích 6, 7 a 8. Úplně náhodné vzorkování (nezávislé realizace náhodné veličiny s rovnoměrným rozložením pravděpodobnosti) má konstantní spektrum a říká se mu „bílý šum“.

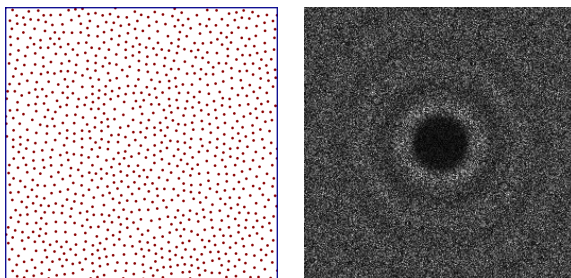


Obrázek 6: Pravidelný rastr a jeho spektrum.

Bílý šum a pravidelný rastr jsou dva extrémní případy, které jsou v našich aplikacích nežádoucí. Při zkoumání vlastností pultónovacích algoritmů v tiskařství si R. Ulichney ([2]) uvědomil, že jsou velmi výhodná taková rozložení teček, ve kterých se neuplatňují rušivé nízké frekvence, tj. pravidelnosti. Nazval tuto vlastnost *modrým šumem* („blue noise“) a charakterizoval ji absencí malých vzdáleností mezi sousedy, viz později v oddílu 4.8 uváděné Poissonovo diskové vzorkování. Spektrální charakteristika modrého šumu je jednoduchá – uprostřed domény spektra (kolem počátku) se má objevit kruhová oblast s nízkými hodnotami, jak je patrné

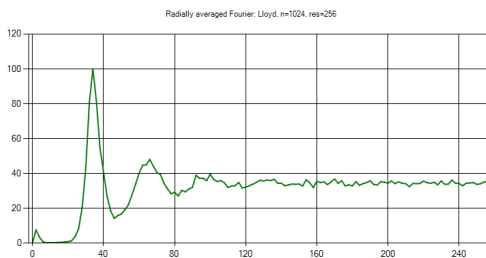


Obrázek 7: Nahodilejší vzorkování („semi-jittering“) a jeho spektrum.



Obrázek 8: Kvalitní vzorkování („Mitchell“) a jeho spektrum.

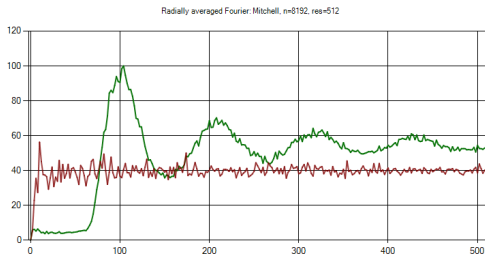
třeba ze spektra na obrázku 8. Další zdroje studující pŕltónovací techniky z pohledu spektra: [18], [19] (autoři dokonce provedli psychovizuální studii, aby zjistili, které distribuce jsou pro lidské oko příjemné) nebo ještě jednou [17].



Obrázek 9: Radiálně zprůměrované spektrum („Lloyd“).

Ulichney ([2], [18]) dále navrhl radiální transformaci spektra pro pohodlnější zkoumání frekvenčního rozložení v případě 2D vzorkování. Radiálně zprůměrované spektrum a jeho rozptyl navrhuje počítat diskřetně v pra-

videlně rozložených mezikružích kolem počátku souřadnic. Tímto postupem dostaneme dva grafy – rozložení frekvencí bez ohledu na prostorovou orientaci a rozptyl hodnot charakterizující uniformitu neboli invarianci vzorkovacího systému na otočení.



Obrázek 10: Frekvenční analýza opatřená rozptylem („Mitchell“).

Příklad prvního grafu pro poměrně kvalitní rozložení s podílem modrého šumu (viz hodnoty kolem nuly) je na obrázku 9. Analýza jiného vzorkování opatřená navíc červenou křivkou rozptylu je na obrázku 10. Tak by měly vypadat výsledky spektrální analýzy kvalitního rozložení.

V kapitole 5.2 budou uvedeny výsledky spektrální analýzy některých vzorkovacích metod z následujícího přehledu.

4 Přehled vzorkovacích metod

V přehledu populárních vzorkovacích metod (výčty v hranatých závorkách) budeme používat následující pojmy:

- „korektní“ – pokrývá zdrojovou oblast, „každý bod má šanci“
- „lze řídit hustotou“ – existuje varianta algoritmu, ve které jsou vzorky generovány s danou vstupní hustotou pravděpodobnosti
- „lze paralelizovat“ – algoritmus výpočtu sady vzorků lze snadno paralelizovat bez potřeby komunikace mezi jednotlivými výpočetními vlákny

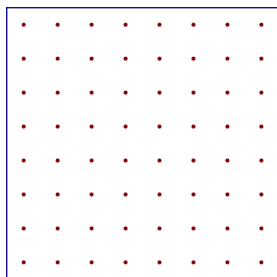
4.1 Pravidelné vzorkování

Jen pro úplnost zde uvádíme možnost pokrytí zdrojové oblasti pravidelnou sítí vzorků. Lze použít čtvercovou nebo hexagonální soustavu, obě mají podobné vlastnosti.

[není korektní, je pravidelné, je deterministické, je jednoduché a rychlé, lze paralelizovat, nelze jednoduše řídit hustotou]

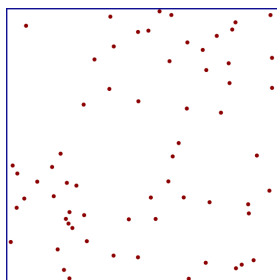
4.2 Náhodné vzorkování

Primární vzorkování, je základem mnoha dalších algoritmů, ve statistice se někdy nazývá „Poissonovo vzorkování“. Jedná se o *nezávislé opakování*



Obrázek 11: Pravidelné vzorkování.

realizace dané náhodné veličiny. V nejjednodušším případě se používá rozdělení s rovnoměrnou hustotou pravděpodobnosti $R([0, 1]^2)$. Jeho jedinou významnou nevýhodou je tvorba shluků vedoucí k horším spektrálním vlastnostem a špatné diskrepanci.



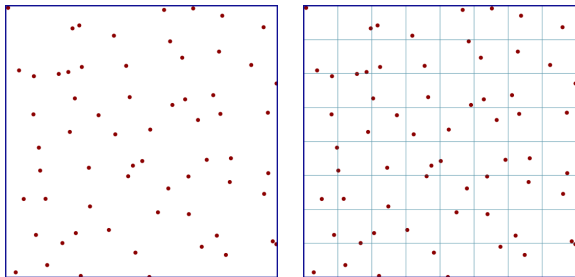
Obrázek 12: Náhodné (nezávislé) vzorkování.

[jednoduché a rychlé, korektní, lze řídit hustotou, lze paralelizovat, vytváří shluky]

4.3 Jittering (roztřesení)

Metoda se ve statistice nazývá „vzorkování po částech“ („stratified sampling“). Do počítačové grafiky ji zavedl R. Cook a spol. ([20]). Vzorkovaný interval se nejprve rozdělí na stejně velké podintervaly a z každého z nich se poté nezávisle náhodně vybere jeden vzorek. Pro jednotkový čtverec je nejvýhodnější rozdělení na $K \times K$ menších čtverců, počet vzorků tak bude $N = K^2$.

Jittering snižuje rozptyl při Monte-Carlo integraci a i v jiných oblastech je oblíbený pro svoji jednoduchost a dobré vlastnosti. V podstatě se dá říci, že se dá vždy nasadit místo náhodného (Poissonova) vzorkování a dává minimálně stejně dobré výsledky.

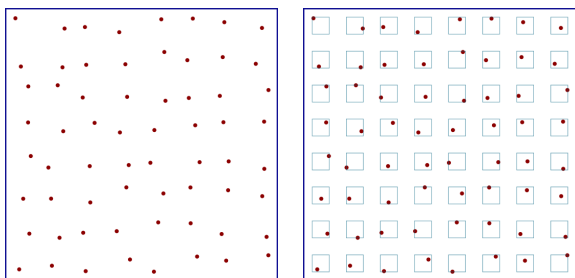


Obrázek 13: Jittering (roztřesení) s geometrickou ilustrací.

[jednoduché a rychlé, korektní, nelze jednoduše řídit hustotou, lze paralelizovat, nevytváří shluky, dobrá diskrepance]

4.4 Semi-jittering

Tato metoda vychází z jitteringu, u jejího zrodu byla snaha ještě víc zredukovat možnost shlukování vzorků. Proto se vzorky vybírají z menších podintervalů, část zdrojové oblasti tedy zůstane nepokrytá. Může mít omezené použití v některých oblastech, nevede však k nestranným odhadům při integraci.

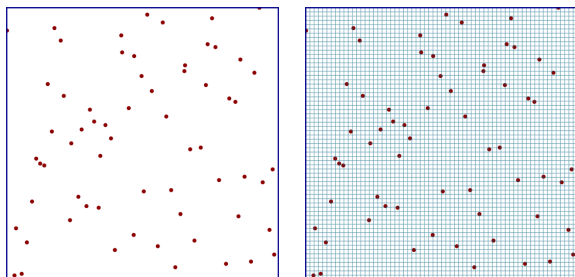


Obrázek 14: Semi-jittering (poloviční roztřesení) s geometrickou ilustrací.

[jednoduché a rychlé, není korektní, nelze řídit hustotou, lze paralelizovat, nevytváří shluky, je patrná lehká pravidelnost, slušná diskrepance]

4.5 N věží („N rooks“)

Metoda byla vyvinuta pro vzorkování ve vícerozměrných prostorech (Cook v [20] vzorkoval 9-rozměrný prostor, další zmínky viz [21] nebo [5]). Jednotlivé dimenze se vzorkují nezávisle pomocí jitteringu se stejným počtem vzorků v každé dimenzi. Vzorky se potom spojí náhodným způsobem (náhodnou permutací).



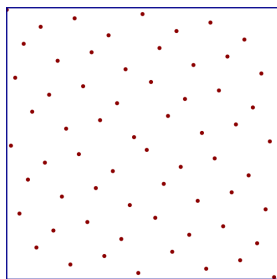
Obrázek 15: Vzorkování „N věží“ s geometrickou ilustrací.

2D varianta se přilehavě nazývá „N věží“, tím se vyjadřuje, že v každém řádku i sloupci se nachází právě jeden vzorek. Postup umístování vzorků lze též chápat jako vzorkování diagonálních čtverečků následované náhodnou permutací řádek tabulky.

[jednoduché a rychlé, korektní, nelze řídit hustotou, lze paralelizovat, trochu horší diskrepance]

4.6 Hammersley vzorkování

První ze zástupců deterministických vzorkování vychází ze slavné van der Corputovy posloupnosti [22], stejně jako například Haltonova metoda. Hammersley vzorkování jsme si vybrali pro jeho lepší geometrické vlastnosti.



Obrázek 16: Hammersley vzorkování ($b = 2$).

Nechť n je kladné přirozené číslo a b prvočíslo. „ b -ární reprezentace“ čísla n je vyjádření

$$n = \sum_k d_k(n) b^k$$

(meze sčítání budeme pro jednoduchost vynechávat). $0 \leq d_k(n) < b$ jsou b -ární cifry vyjadřující číslo n . Definujme racionální číslo $0 \leq g_b(n) < 1$

sumou

$$g_b(n) = \sum_k d_k(n) b^{-k-1}$$

Van der Corputova posloupnost $\{g_b(n)\}_n$ vyplňuje hustě jednotkový interval $[0, 1]$, což je důležité pro vzorkovací metody na této posloupnosti založené. Slavná *Haltonova posloupnost* je definovaná pro nějaká dvě různá prvočísla (třeba $b_1 = 2$ a $b_2 = 3$) jako

$$\{[g_{b_1}(n), g_{b_2}(n)]\}_n$$

Hammersley posloupnost má jednodušší definici pomocí jediného prvočísla (např. $b = 2$)

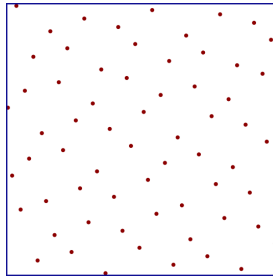
$$\left\{ \left[\frac{n}{N}, g_b(n) \right] \right\}_n$$

Připomínáme vzdálenou podobnost se vzorkováním „N věží“, museli bychom ještě přidat jittering v rozsahu malých intervalů $1/N$.

[deterministické, výborná diskrepance, velmi rychlý výpočet, lze paralelizovat, nelze řídit hustotou, špatné spektrum]

4.7 Larcher-Pillichshammer vzorkování

Další deterministické vzorkování v našem přehledu bývá citováno jako jedno z nejkvalitnějších ve své třídě ([23], [24]).



Obrázek 17: Larcher-Pillichshammer vzorkování.

Larcher s Pillichshammerem použili v [12] trochu jiným způsobem rozklad binárního čísla a definovali tak novou posloupnost. Pracujeme s prvočíselným základem $b = 2$, člen $d_k(n)$ je nahrazen součtem binárních jedniček modulo dva

$$lp_k(n) = \left(\sum_{i=k}^L d_i(n) \right) \bmod 2$$

Term $lp_k(n)$ použijeme analogicky k definici čísla $0 \leq lp(n) < 1$

$$lp(n) = \sum_k lp_k(n) 2^{-k-1}$$

a nakonec celé *Larcher-Pillichshammer* posloupnosti

$$\left\{ \left[\frac{n}{N}, lp(n) \right] \right\}_n$$

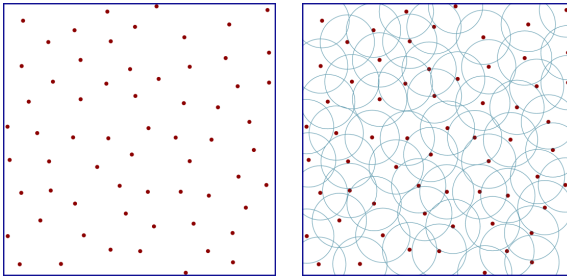
Jako elegantní randomizaci autoři používají binární operaci *XOR* faktoru $lp_k(n)$ s libovolným binárním číslem („semínkem náhody“ *seed*). Formálně je možné napsat náhodnou inicializaci jako úpravu definice $lp_k(n)$:

$$\overline{lp_k(n)} = \left(seed_k + \sum_{i=k}^L d_i(n) \right) \bmod 2$$

[deterministické, lze snadno randomizovat, výborná diskrepance, velmi rychlý výpočet, lze paralelizovat, nelze řídit hustotou, špatné spektrum]

4.8 Poissonovo diskové vzorkování

Vzorkování inspirované přírodou – jednoduchý systém, jak se vyhnout vzniku shluků, je omezit minimální vzdálenost vzorků nějakou vhodnou konstantou D (viz [25] nebo [21]). Takové vzorkování se nazývá *Poissonovo diskové vzorkování* („Poisson disk sampling“) a uvedená podmínka je základem vzniku „modrého šumu“, viz kapitolu 3.3. Podobnou distribuci mají kolonie buněk živých organismů, pro počítačovou grafiku je zejména zajímavá shoda s rozložením světločivných buněk na sítnici oka primátů.



Obrázek 18: Poissonovo diskové vzorkování ($D = 0.1$) s geometrickou ilustrací.

Soubor vzorků se generuje postupně, pokud by se nový vzorek přiblížil k některému již existujícímu na vzdálenost menší než D , bude odmítnut (této technice se obecně říká „rejection sampling“). Je trochu obtížné

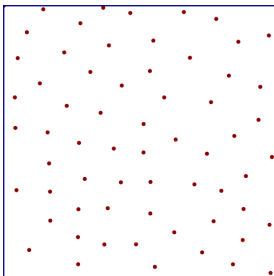
určit předem konstantu D , kdyby totiž byla hodnota příliš velká, mohla by zabránit nalezení potřebného počtu vzorků.

Při velkém N je potřeba zvolit vhodnou efektivní implementaci, aby se nemusela procházet vždy celá sada vzorků při kontrole nového kandidáta. Příklady datových struktur viz [26].

[nepravidelné, ale pěkné na pohled, dobrá diskrepance, výborné spektrum, je korektní, nelze snadno paralelizovat, pomalejší výpočet, lze řídit hustotou, obtížné nastavení konstanty D]

4.9 Mitchellův algoritmus

D. Mitchell si v [21] všiml nevýhod běžných metod výpočtu Poissonova diskového vzorkování, zejména nutnosti správně určit konstantu D . V mnohých situacích by se hodilo, kdybychom mohli sadu vzorků počítat postupně, a tak podle potřeby pokrytí zdrojové oblasti zahušťovat. Byl navržen jednoduchý algoritmus, který nepotřebuje znát předem separační konstantu D a je schopen spočítat jakkoli hustý soubor vzorků (viz obrázek 20).

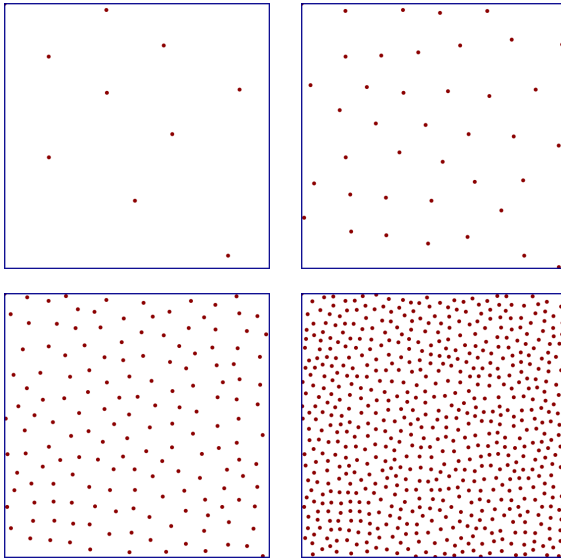


Obrázek 19: Vzorky vygenerované Mitchellovým algoritmem.

Metodě se též říká „Mitchellův algoritmus nejlepšího kandidáta“ – první vzorek vygenerujeme zcela náhodně a další pak podle indukčního kroku: jestliže jsme již přijali N vzorků, tak ten $N + 1$. volíme z množiny náhodně vygenerovaných KN kandidátů $\{C_j\}$, kde $K > 5$ je vhodná konstanta (Mitchell dokonce navrhuje $K \geq 10$). Do souboru přijmeme toho kandidáta, který má maximální vzdálenost od množiny již přijatých vzorků $\{P_i\}$, tj. počítáme $\operatorname{argmax}_j \min_i |P_i - C_j|$.

Algoritmus má sice složitost $O(N^2)$, ale pomocí vhodných datových struktur (à la [26]) jej lze značně zrychlit. Je-li potřeba generovat vzorky s nerovnoměrnou hustotou pravděpodobnosti, lze výpočet $|P_i - C_j|$ snadno upravit tak, že použijeme neuniformní metriku indukovanou danou hustotou.

[nepravidelné, ale pěkné na pohled, dobrá diskrepance, výborné spektrum,

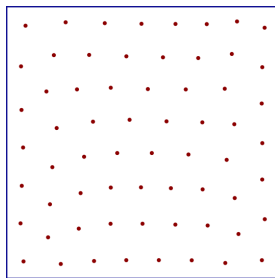


Obrázek 20: Inkrementální soubor: $N = 10$, $N = 40$, $N = 160$ a $N = 640$.

je korektní, nelze snadno paralelizovat, pomalý výpočet, lze řídit hustotou, dokáže generovat inkrementální (postupně se zahušťující) soubor vzorků]

4.10 Lloydův algoritmus

Lloydova metoda je de facto standardem výpočtu speciálního Voroného diagramu, jehož generující body tvoří současně těžiště svých buněk („Centroidal Voronoi Tessellation“, CVT).

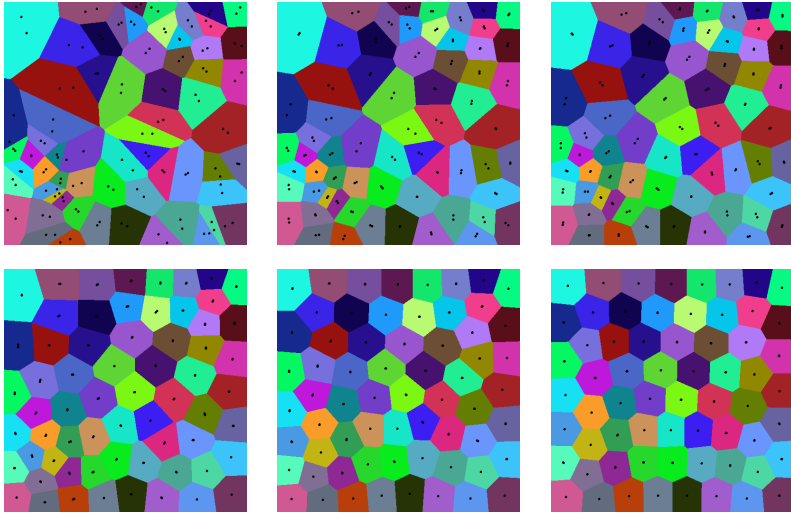


Obrázek 21: Vzorky vygenerované konvergencí Lloydova algoritmu.

S. P. Lloyd svůj algoritmus vymyslel v šedesátých letech při odvozování optimálního kvantovače a publikoval jej až v roce 1982 [27]. Od té doby se

metoda hojně používá v různých oblastech zpracování signálu, designu, počítačové grafiky, analýzy obrazu, apod.

Naivní algoritmus (nazývaný též někdy „iterativní Voronoi“) je velmi přímočarý: inicializuje sadu generujících bodů náhodně a spočítá k nim Voroného diagram. Potom dle potřeby opakuje krok, ve kterém posune všechny generující body do těžišť svých buněk. CVT dostaneme při úplné konvergenci algoritmu, viz obrázek 21 (analýza konvergence v [28]). Ukázku postupu výpočtu pro $N = 64$ vidíme na obrázku 22.



Obrázek 22: Postup výpočtu: $g = 1, g = 2, g = 3, g = 10, g = 30$ a $g = 90$.

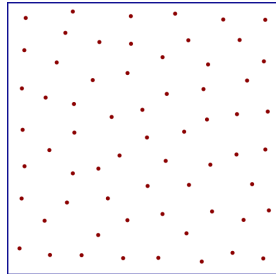
Takto použitá metoda má však horší vlastnosti pro použití v designu nebo tiskařství, generující body CVT jsou totiž na konci konvergence příliš pravidelně rozmístěny – tíhnou k optimálně hustému hexagonálnímu rozložení. Důsledkem je nekvalitní spektrum a na první pohled patrné pravidelnosti. Řešením je proces zastavit dříve, než dospěje do fáze velké pravidelnosti, to je však obecně velmi obtížné rozhodnout.

[vzorky pěkné na pohled, někdy až příliš pravidelné, slušná diskrepance, je korektní, pomalý výpočet, lze modifikovat pro řízení hustotou, obtížné zastavení iterace]

4.11 Kapacitně omezené distribuce (CCPD)

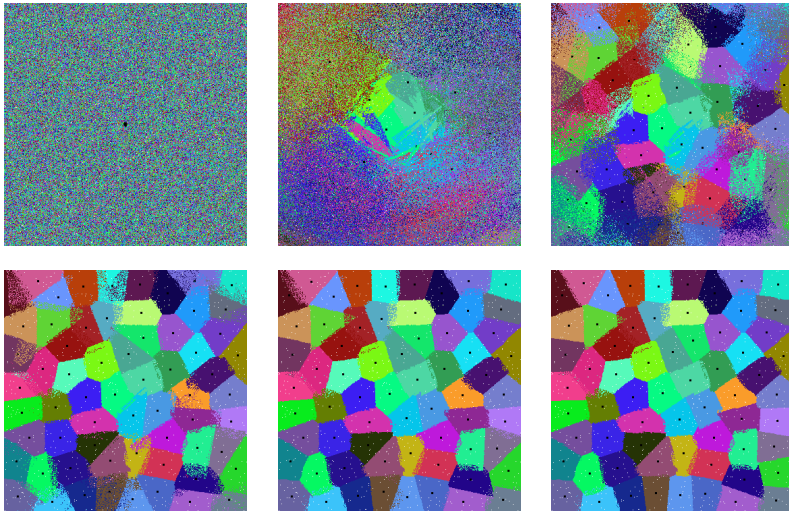
M. Balzer pracoval na variantách Voroných diagramů, které se nazývají „kapacitně omezené“ (viz [29]). Při jejich použití dostaneme pěkné rozložení bodů bez nepříjemných vlastností Lloydovy metody konvergující k CVT. Kapacitně omezené distribuce („Capacity-Constrained Point Dis-

tributions“, CCPD) se staly základem dnešních nejlepších algoritmů, z mnoha citací vybíráme [30] a [31].



Obrázek 23: Kapacitně omezená distribuce (CCPD).

Algoritmus 1 popisuje základní variantu CCPD pro obecnou vstupní hustotu pravděpodobnosti pdf , generuje se N vzorků. Parametrem je konstanta K ovlivňující kvalitu výsledku a nepřímo úměrně rychlost výpočtu (je vhodné ji nastavit na několik stovek až tisíc). Pole S obsahuje aktuální množinu vzorků, pole P množiny jejich bodů. Po celou dobu běhu algoritmu má každý vzorek přiřazeno právě K bodů, jen si je mezi sebou vyměňují. Seznam T_i (resp. T_j) obsahuje body přiřazené vzorku i , které však mají blíž k vzorku j (resp. naopak).



Obrázek 24: Postup výpočtu: $g = 0$, $g = 1$, $g = 2$, $g = 3$, $g = 4$ a $g = 23$.

Postup práce algoritmu pro $N = 64$ a $K = 2000$ vidíme na obrázku 24.

Algoritmus 1 CCPD

```
function CCPD( $N, K, pdf$ )    ▷ počet vzorků, K, hustota pravděp.  
     $Point[] S \leftarrow \mathbf{new} Point[N]$     ▷ sada vzorků  
     $List<Point>[] P \leftarrow \mathbf{new} List<Point>[N]$     ▷ a jejich body  
    for  $i \leftarrow 1, N$  do    ▷ inicializace  
        for  $k \leftarrow 1, K$  do  
             $P[i].add(pdf())$     ▷ bod vygenerovaný podle hustoty  $pdf$   
        end for  
         $S[i] \leftarrow \mathbf{CENTROID}(P[i])$     ▷ těžiště seznamu bodů  
    end for  
    repeat    ▷ jedna zlepšující generace  
         $changed \leftarrow false$   
        for all  $i, j \leftarrow 1, N$  do  
             $List<Point> Ti \leftarrow \mathbf{CLOSER}(P[i], S[j], S[i])$   
             $List<Point> Tj \leftarrow \mathbf{CLOSER}(P[j], S[i], S[j])$   
            if  $Ti \neq \emptyset \wedge Tj \neq \emptyset$  then  
                 $changed \leftarrow true$     ▷ body se budou vyměňovat  
                repeat    ▷ výměna nejméně výhodných bodů  
                     $P[i].remove(Ti[0]); P[j].add(Ti[0]); Ti.deleteAt(0)$   
                     $P[j].remove(Tj[0]); P[i].add(Tj[0]); Tj.deleteAt(0)$   
                until  $Ti = \emptyset \vee Tj = \emptyset$   
                 $S[i] \leftarrow \mathbf{CENTROID}(P[i])$     ▷ přepočítám těžiště  
                 $S[j] \leftarrow \mathbf{CENTROID}(P[j])$   
            end if  
        end for  
    until  $\neg changed$   
    return S  
end function
```

Ke konvergenci postačilo 23 kroků iterace.

[velmi pěkné vzorky, výborná diskrepance i spektrum, je korektní, lze řídit hustotou pravděpodobnosti, středně rychlý výpočet]

5 Výsledky

5.1 Matice vlastností

Pro přehlednost jsme zařadili tabulku obsahující zjednodušený výčet nejdůležitějších vlastností všech popisovaných vzorkovacích metod. Snažili jsme se vystihnout typické vlastnosti za předpokladu správného (optimálního) nastavení případných parametrů.

Vysvětlení významu sloupečků tabulky:

- „pokr“ – pokrývá sada vzorků celou zdrojovou oblast? (tj. vede k ne-strannému odhadu integrálu)
- „spekt“ – spektrální vlastnosti metody, splňují vzorky vlastnost „modrého šumu“?
- „hust“ – lze algoritmus modifikovat tak, aby byl řízen libovolnou hustotou pravděpodobnosti?
- „rnd“ – náhodnost (opak determinismu)
- „rych“ – velká rychlost implementace
- „par“ – možnost efektivní paralelní implementace (bez nutnosti komunikace mezi vlákny)

Hodnoty v jednotlivých políčkách:

- „++“ – výborně splňuje, je to silná stránka metody
- „+“ – splňuje
- „~“ – splňuje jen částečně, slyšeli jsme o modifikacích. . .
- „ “ – nesplňuje
- „-“ – silně nesplňuje, velmi nevýhodné řešení

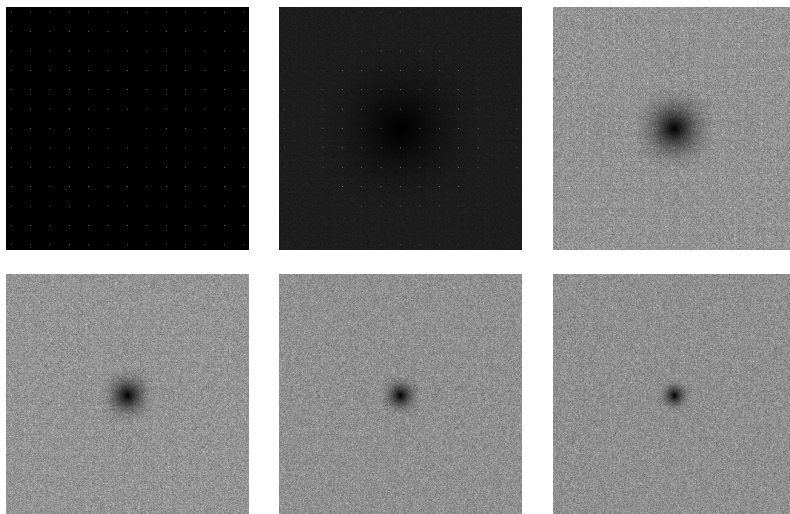
metoda	pokr	spekt	hust	rnd	rych	par
Pravidelné	-	-	-	-	++	++
Hammersley	+	-			++	+
Larcher-Pillichshammer	+	-		+	++	+
Náhodné	++		++	++	++	++
Jittering	++	~		++	++	++
Semi-jittering		~		+	++	++
N věží	++			++	+	~
Poissonův disk	++	++	+	++		~
Mitchell	++	++	++	++		~
Lloyd	++	+	+	+	-	~
CCPD	++	++	++	++		~

Tabulka 1: Matice vlastností.

5.2 Spektrální vlastnosti

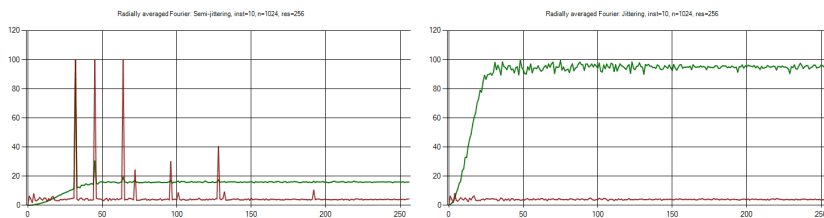
Některé popisované metody byly podrobeny frekvenční analýze ve srovnatelných podmínkách – sada $N = 1024$ vzorků s rovnoměrnou hustotou pravděpodobnosti, 10 průměrovaných souborů vzorků. Srovnatelná výkonová spektra a jejich radiální průměry najdete v této sekci.

První porovnání ukazuje fatální vliv pravidelnosti na kvalitu spektra. Byla počítána spektra pro semi-jittering s amplitudami od 0.0 do 1.0 (první nastavení odpovídá pravidelnému vzorkování, poslední čistému jitteringu). Výsledky viz obrázek 25.



Obrázek 25: Pravidelnost pro semi-jittering od $a = 0.0$ do $a = 1.0$.

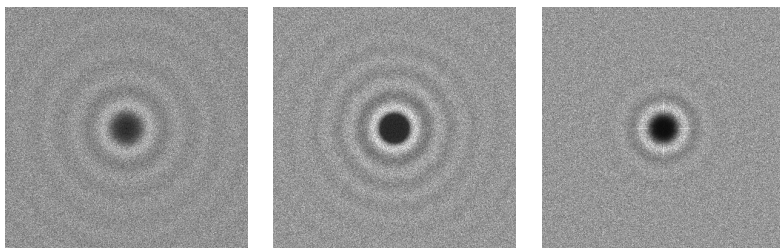
Radiálně průměrovaná spektra byla spočtena jen pro amplitudy 0.6 a 1.0. Můžeme si všimnout, že pro semi-jittering z obrázku spektra již není patrna velká pravidelnost, graf ji však stále zřetelně ukazuje. Čistý jittering má radiálně velmi rovnoměrné spektrum, vše je vidět na obrázku 26.



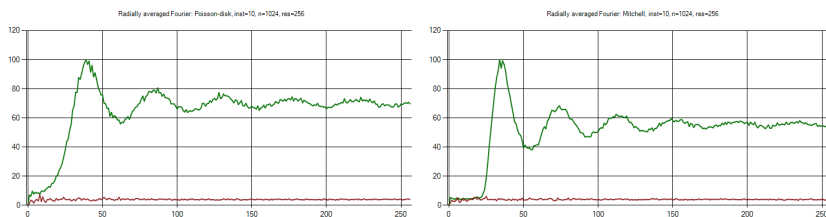
Obrázek 26: Radiální spektra pro semi-jittering $a = 0.6$ a $a = 1.0$.

Další porovnání ukazuje pěkná spektra generovaná Poissonovým diskovým vzorkováním (nastavení $D = 0.022$), Mitchellovým algoritmem ($K = 10$) a CCPD ($K = 100$). Všechny soubory vykazují charakteristiky modrého šumu, viz obrázek 27. Radiální spektra prvních dvou metod ukazuje obrázek 28.

Předposlední spektrum ukazuje pseudo-pravidelnost zkonvergovaného Lloydova algoritmu – sada vzorků tvoří CVT. Vzorky se snaží umístit se

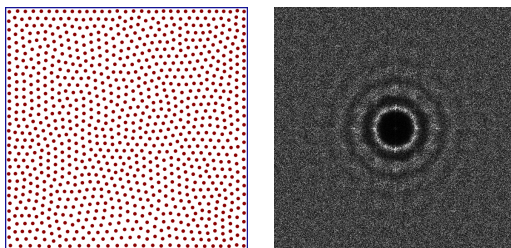


Obrázek 27: Modrý šum v metodách Poissonův disk, Mitchell a CCPD.



Obrázek 28: Radiální spektra pro Poissonův disk a Mitchellův alg.

téměř do hexagonální mřížky, proto je ve spektru na obrázku 29 patrná pravidelná struktura.

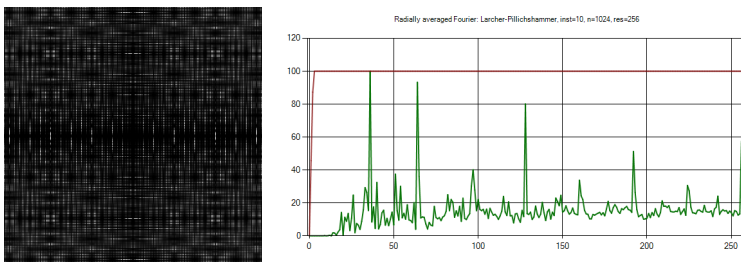


Obrázek 29: Pravidelnost CVT distribuce (zkonvergováný LLoyd).

Nakonec uvádíme zajímavé spektrum randomizované metody Larcher-Pillichshammer, jednotlivé vzorky byly umísťovány do uzlových bodů mřížky, náhodnost tedy spočívala jen ve změnách permutace (inicializace). Viz obrázek 30 – upozorňujeme, že zde má radiální spektrum výjimečně velký rozptyl (červený graf).

5.3 Měření diskrepance

Všechny popisované vzorkovací metody byly kvantitativně porovnány v následujícím testu. Každá metoda vygenerovala 100 různých sad vzorků, ze



Obrázek 30: Pravidelnost Larcher-Pillichshammer sekvence.

kterých byla spočítána průměrná diskrepance a její výběrová směrodatná odchylka. Jednotlivé metody měly nastaveny vhodné hodnoty případných parametrů, sada vzorků obsahovala vždy $N = 1024$ vzorků z jednotkového čtverce.

Byla počítána diskrepance podle Strouda se střední kvadratickou hodnotou (D_2^s). Diskrepance jsou u vedeny v tisícinách, jejich výběrové směrodatné odchylky v milióntinách. Výsledky jsou uvedeny v tabulce 2.

metoda	diskrepance ($\times 10^{-3}$)	SD ($\times 10^{-6}$)
Pravidelné	7.468	0.0
Hammersley*	0.811	0.0
Larcher-Pillichshammer*	0.811	0.0
Náhodné	8.941	2.5
Jittering	2.593	0.0
Semi-jittering	4.159	0.0
N věží	5.220	0.5
Poissonův disk	3.255	0.2
Mitchell	3.183	0.2
Lloyd ($g = 40$)	6.400	2.5
Lloyd ($g = 400$)	5.661	1.8
CCPD	2.154	0.0

Tabulka 2: Průměrné diskrepance.

Lloydův algoritmus musel být studován ve dvou nastaveních: pro dobrý výsledek bylo nutné zastavit iteraci dostatečně brzy (empiricky byla zvolena generace $g = 40$). Pro porovnání byl spočítány i výsledky z pokročilejšího stadia konvergence ($g = 400$, zde se již sady vzorků blíží stavu „Centroidal Voronoi“ – CVT).

Z tabulky plyne, že si z pohledu diskrepance nejlépe vedou deterministické sekvence (ty však mají špatné spektrální charakteristiky, viz

sekce 3.3 a 5.2). Ze spektrálně vyhovujících metod se jako nejlepší ukázala *kapacitně omezená distribuce*, ani *jittering* nedává špatné výsledky.

6 Závěr

Byly studovány metody náhodného rozmístování (vzorkování) bodů v rovině s ohledem na jejich dobré estetické vlastnosti a použitelnost pro Monte-Carlo integraci. Byly uvedeny metody objektivního posuzování kvality vzorkování, zejména diskrepance a dvě nejvýznamnější spektrální charakteristiky. Přehled populárních vzorkovacích algoritmů byl doplněn spektrální analýzou a srovnávacím měřením jejich diskrepance.

Aby byla všechna srovnání co nejkorektnější, byly všechny popisované metody implementovány autorem příspěvku bez použití dalších knihoven. Všechna srovnávací měření i ilustrační obrázky jsou výsledkem téhož programu.

Reference

- [1] I. Simpson and L. Wood. *The Encyclopedia of Drawing Techniques*. A quarto book. Headline, 1987.
- [2] R. Ulichney. *Digital Halftoning*. MIT Press, 1987.
- [3] D. L. Lau and G. R. Arce. *Modern Digital Halftoning, Second Edition*. Signal Processing and Communications. Taylor & Francis, 2001.
- [4] Eric Veach. *Robust Monte Carlo Methods for Light Transport Simulation*. PhD thesis, Stanford University, Stanford, CA, USA, 1998. AAI9837162.
- [5] Malvin H. Kalos and Paula A. Whitlock. *Monte Carlo Methods. Vol. 1: Basics*. Wiley-Interscience, New York, NY, USA, 1986.
- [6] S. K. Zaremba. The mathematical basis of Monte Carlo and quasi-Monte Carlo methods. *SIAM Review*, 10(3):303–314, July 1968.
- [7] Peter Shirley. Discrepancy as a quality measure for sample distributions. In *In Eurographics '91*, pages 183–194. Elsevier Science Publishers, 1991.
- [8] A. H. Stroud. *Approximate calculation of multiple integrals*. Prentice-Hall series in automatic computation. Prentice-Hall, 1971.
- [9] J. H. Halton. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numerische Mathematik*, 2(1):84–90, 1960.
- [10] J. H. Halton. Algorithm 247: Radical-inverse quasi-random point sequence. *Commun. ACM*, 7(12):701–702, December 1964.
- [11] J. M. Hammersley and D. C. Handscomb. *Monte Carlo Methods*. Methuen's monographs on applied probability and statistics. Methuen, 1964.
- [12] F. Pillichshammer and G. Larcher. Walsh series analysis of the L_2 -discrepancy of symmetrized point sets. *Monatshefte für Mathematik*, 132(1):1–18, 2001.

- [13] David P. Dobkin, David Eppstein, and Don P. Mitchell. Computing the discrepancy with applications to supersampling patterns. *ACM Trans. Graph.*, 15(4):354–376, October 1996.
- [14] M. de Berg. *Computational Geometry: Algorithms and Applications*. Springer, 2008.
- [15] Franklin C. Crow. Summed-area tables for texture mapping. *SIGGRAPH Comput. Graph.*, 18(3):207–212, January 1984.
- [16] Arthur Schuster. On the investigation of hidden periodicities with application to a supposed 26 day period of meteorological phenomena. *Terrestrial Magnetism*, 3(1):13–41, March 1898.
- [17] Yahan Zhou, Haibin Huang, Li-Yi Wei, and Rui Wang. Point sampling with general noise spectrum. *ACM Trans. Graph.*, 31(4):76:1–76:11, July 2012.
- [18] Robert Ulichney. A review of halftoning techniques. *SPIE*, 3963:378–391, 2000.
- [19] Theophano Mitsa and Kevin J. Parker. Digital halftoning technique using a blue-noise mask. *J. Opt. Soc. Am. A*, 9(11):1920–1929, Nov 1992.
- [20] Robert L. Cook, Thomas Porter, and Loren Carpenter. Distributed ray tracing. *SIGGRAPH Comput. Graph.*, 18(3):137–145, January 1984.
- [21] Don P. Mitchell. Spectrally optimal sampling for distribution ray tracing. *SIGGRAPH Comput. Graph.*, 25(4):157–164, July 1991.
- [22] J. G. van der Corput. Verteilungsfunktionen. I. *Mitt. Proc. Akad. Wet. Amsterdam*, 38:813–821, 1935.
- [23] Thomas Kollig and Alexander Keller. Efficient multidimensional sampling. *Computer Graphics Forum*, 21(3):557–564, September 2002.
- [24] A. Keller, S. Heinrich, and H. Niederreiter. *Monte Carlo and Quasi-Monte Carlo Methods 2006*. Springer, 2007.
- [25] Robert L. Cook. Stochastic sampling in computer graphics. *ACM Trans. Graph.*, 5(1):51–72, January 1986.
- [26] Daniel Dunbar and Greg Humphreys. A spatial data structure for fast Poisson-disk sample generation. *ACM Trans. Graph.*, 25(3):503–508, July 2006.
- [27] Stuart P. Lloyd. Least squares quantization in PCM. *IEEE Trans. Inf. Theor.*, 28(2):129–137, September 2006.
- [28] Qiang Du, Maria Emelianenko, and Lili Ju. Convergence of the Lloyd algorithm for computing Centroidal Voronoi Tessellations. *SIAM J. Numer. Anal.*, 44(1):102–119, January 2006.
- [29] Michael Balzer, Thomas Schlömer, and Oliver Deussen. Capacity-constrained point distributions: A variant of Lloyd’s method. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2009)*, 28(3):86:1–8, 2009.

- [30] Yin Xu, Ligang Liu, Craig Gotsman, and Steven J. Gortler. Capacity-constrained Delaunay triangulation for point distributions. *Computers & Graphics*, pages 510–516, 2011.
- [31] F. de Goes, K. Breeden, V. Ostromoukhov, and M. Desbrun. Blue noise through optimal transport. *ACM Trans. Graph. (SIGGRAPH Asia)*, 31, 2012.