Adaptív hisztogram építés folytonos függvények közelítéséhez Monte Carlo részecsketranszport mintákból

TDK dolgozat

2016

Készítette: Böröczki Zoltán István
RCW2MK

Konzulens: Dr. Légrády Dávid
Egyetemi docens

Tartalomjegyzék

1. Bevezetés

A modern adatfeldolgozás során szükségünk van különböző, akár többdimenziós sűrűségfüggvények becslésére és megjelenítésére is. Ezen folytonos függvényeket – általában - a fázistér strukturált rácson történő felosztásán becslik, hisztogramot készítenek. Nagy elemű minta, megfelelő cellaszélesség és darabszám esetén a hisztogram jól közelíti a becsülni kívánt sűrűségfüggvényt, azaz a statisztikus szórás értéke minden rácselemen történő becslésre alacsonnyá válik. Más módszerek segítségével is lehet sűrűségfüggvény becsléseket végezni, a nukleáris technika területén több módszert is alkalmaznak: függvények szerinti sorfejtést [1], magfüggvényes becslést [2], illetve adaptív felosztást alkalmazó technikát [3].

Több területen is gyakran alkalmaznak Monte Carlo (MC) számításokat, amelyek tipikusan néhány skalár érték meghatározására alkalmasak. A MC számítás nukleáris fizika területén is széles körben alkalmazott eszköz, például részecsketranszport szimulációkra. Erre a problématerületre került kifejlesztésre az MCNP nevű széles körben használt és hitelesített, magas szintű MC részecsketranszport kód. Ebben található például a beépített „radiographic tally”, amely a fázistér strukturált rácson történő felosztásával 2D fluxusbecslést lehet végezni. [4] [5]

A munkám során egy többdimenziós adaptív hisztogramkészítő algoritmust fejlesztettem ki. A program a felhasználó által kijelölt dimenziókon, a medián mentén többször megfelezi a fázisteret, így létrehozva az adaptív rácsot.

Az első MATLABban írt programverzióval összehasonlító számításokat végeztem 2D mintákkal a strukturált és az általam fejlesztett programmal készített adaptív rácson, majd C++-ban egy általánosabb algoritmust írtam, mely képes tetszőleges dimenzióban a fázistér felosztására. Ezenkívül a felhasználó által generált mintákon kívül, akár az MCNP által létrehozott részecske minták feldolgozására is, így lehetőséget ad valós eseteken történő vizsgálatokra. Az így létrejött függvényközelítés tulajdonságait elemeztem és összehasonlítottam a szabályos rács segítségével nyert eloszlás tulajdonságaival.

A többdimenziós eloszlás ilyen módon történő rekonstruálása alkalmas a vizsgált mennyiség térbeli eloszlásának numerikus analízisére, vizuális elemzésére.

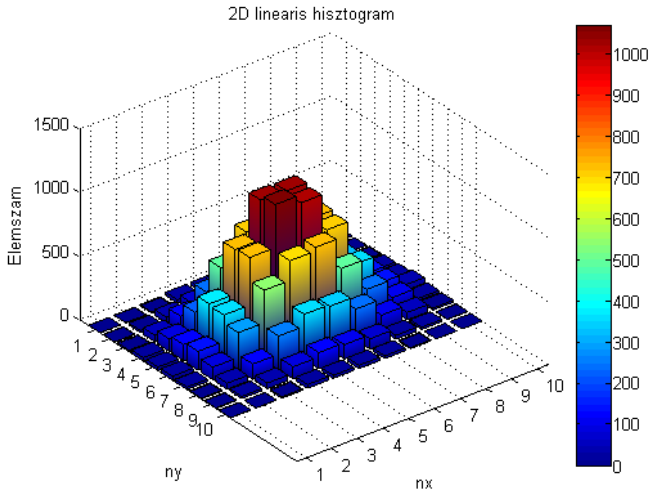
A létrejött struktúrára kidolgoztam egy módszert mellyel az eloszlást akár mintavételezni is lehet.

2. 2D állandó cellaméretű hisztogramok vizsgálatai

Leggyakrabban a sűrűségfüggvények becsléséhez strukturált, állandó cellaméretű hisztogramokat használnak. A dolgozat első felében 2D hisztogramok általános tulajdonságait ismertetem.

2.1. 2D hisztogram készítése

Vegyünk egy 2D Descartes koordináta rendszert, melyben rögzítsünk egy négyzet alapú tartományt, melynek határoló vonalai legyenek merőlegesek a tengelyekre. Ezt a tartományt osszuk fel mind az X és az Y tengely mentén n részre, így létrehozva $n \cdot n$ darab kis négyzetet, melyet celláknak nevezünk. Ezekbe a területekbe beeső pontokat számoljuk meg, és hozzunk létre egy $n \cdot n$-es mátrixot, melynek elemei F_{ij} az egyes négyzettartományokba beérkezett pontok számát tartalmazza. Ez a mátrix szolgálja a 2D állandó cellaméretű hisztogram alapját.



1. ábra N = 20000 véletlen pont $10 \cdot 10$ felbontású hisztogramba helyezve az $x \epsilon (-3,3)$ és $y \epsilon (-3,3)$ intervallumon

A 1. ábrán egy 2D állandó cellaméretű hisztogram képét láthatjuk, amely a $x \epsilon (-3,3)$ és $y \epsilon (-3,3)$ intervallumot $10 \cdot 10$ négyzet alapú elemre bontja fel. A vizsgált mintát egy 2D normális eloszlás szerint mintavételezett $N = 20000$ ponthalmaz adta.

Amennyiben a hisztogram alapjául szolgáló mátrix elemeket F_{ij} normáljuk a mintaszámmal N és a cella méretével $A = \Delta x \cdot \Delta y$, akkor erre a kisterületre átlagolt sűrűségfüggvény közelítését F_{ij}^{avg} kapjuk vissza, amely szerint a mintavételezést folytattuk. [6]

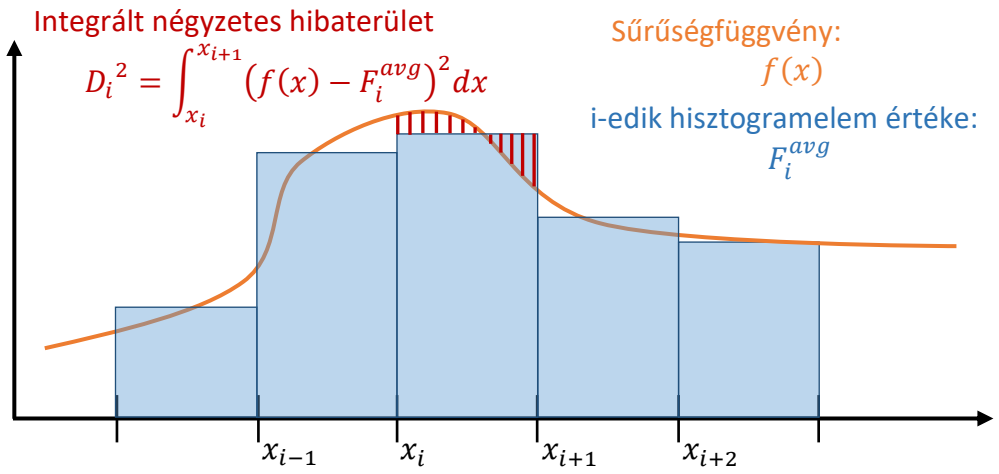
$$F_{ij}^{avg} = \frac{F_{ij}}{A \cdot N} = \frac{F_{ij}}{\Delta x \cdot \Delta y \cdot N}$$

2.2. A hisztogram mintavételezésének hibái

Egy cellába beérkező mintaszám átlagos eltérését becsüljük a statisztikus és a diszkretizációs hiba összegeként.

2.2.1. Diszkretizációs hiba

Tekintsük diszkretizációs hibának azt, hogy a folytonos $f(x,y)$ sűrűségfüggvényt egy szakaszosan konstans, véges felbontású F_{ij}^{avg} hisztogram elemeivel közelítjük. A könnyebb megértés érdekében nézzük meg, hogy a diszkretizációs hiba hogyan határozható meg egy 1D hisztogram esetében:



2. ábra 1D-ben a diszkretizációs hiba meghatározása

Az egydimenziós becsülni kívánt sűrűségfüggvény $f(x)$. A vizsgált tartományt osszuk fel n részre, és nézzük meg az i.-ik belső hisztogramelemet F_i^{avg}, melynek határoló pontjai x_i és x_{i+1}. Akkor a diszkretizációs hibát egy dimenzióban következő módon számolhatjuk:

$$D_i{}^2 = \int_{x_i}^{x_{i+1}} \left(f(x) - F_i^{avg}\right)^2 dx$$

Két dimenzióban a becslés ennek analógiájára történik, a diszkretizációs abszolút hiba nagyságát ebben az esetben a következő módon határozhatjuk meg:

$$D_{ij}^{Det\,2} = \int_{x_i}^{x_{i+1}} \int_{y_i}^{y_{i+1}} \left(f(x,y) - F_{ij}^{avg}\right)^2 dx\, dy,$$

illetve relatív hibaként kifejezve:

$$r_{ij}^{Det\,2} = \frac{D_{ij}^{Det\,2}}{F_{ij}^{avg\,2}} = \frac{\int_{x_i}^{x_{i+1}} \int_{y_i}^{y_{i+1}} \left(f(x,y) - F_{ij}^{avg}\right)^2 dx\, dy}{F_{ij}^{avg\,2}}$$

2.2.2. Statisztikus hiba

A statisztikus hibát nagyszámú egymástól független esemény megfigyelésekor a megfigyelt mennyiség statisztikus ingadozásából becsülhetjük. Nézzük meg, hogy ezt a hibát egy Monte Carlo szimuláció esetén hogyan határozhatjuk meg. A szimuláció során meghatározni kívánt mennyiség legyen $\hat{I}$, ezt az N darab beérkező C_i minta alapján, átlagképzéssel kaphatjuk meg:

$$\hat{I} = \frac{1}{N}\sum C_i$$

Ekkor meghatározni kívánt mennyiség szórását a következőképpen becsülhetjük: [4]

$$D^2(\hat{I}) = D^2\left(\frac{1}{N}\sum C_i\right) = \frac{1}{N^2}D^2\left(\sum C_i\right) = \frac{1}{N}D^2(C_i)$$

Tegyük fel, hogy a mintaátlag szórása a következő módon írható fel:

$$D^2(C_i) = E\left({C_i}^2\right) - E^2(C_i) \underbrace{= \frac{1}{N}\sum {C_i}^2 - \left(\frac{1}{N}\sum C_i\right)^2}_{\text{Azt tippeljük, hogy így felírható}}$$

Hogy feltevésünket visszaellenőrizzük, nézzük meg, hogy a két tagnak mi a várható értéke:

$$E\left(\frac{1}{N}\sum {C_i}^2\right) = \frac{1}{N}NE\left({C_i}^2\right) = E\left({C_i}^2\right)$$

$$E\left(\frac{1}{N}\sum C_i\right)^2 = \frac{1}{N^2}E\left(\sum_i\sum_j C_iC_j\right) = \frac{1}{N^2}\left[NE\left({C_i}^2\right) + (N^2-N)E^2(C_i)\right]$$

Látható, hogy a két tag nem független, így egy korrekciós tag fog megjelenni.

A kettő összegére tehát kaphatjuk, hogy:

$$E\left[\frac{1}{N}\sum {C_i}^2 - \left(\frac{1}{N}\sum C_i\right)^2\right] = E\left({C_i}^2\right) - \frac{1}{N^2}\left[NE\left({C_i}^2\right) + (N^2-N)E^2(C_i)\right] =$$

$$= E\left({C_i}^2\right) - \left[\frac{1}{N}E\left({C_i}^2\right) + \frac{N-1}{N}E^2(C_i)\right] = \frac{N-1}{N}E\left({C_i}^2\right) + \frac{N-1}{N}E^2(C_i) =$$

$$= \underbrace{\frac{N-1}{N}}_{\text{Korrekciós tag}} D^2(C_i)$$

Ezek alapján az abszolút szórását megkapjuk:

$$D^2(\hat{I}) = \frac{1}{N}D^2(C_i) = \frac{1}{N}\underbrace{\frac{N}{N-1}}_{1-\text{hez tart}}\left(\frac{1}{N}\sum {C_i}^2 - \left[\frac{1}{N}\sum C_i\right]^2\right)$$

A Monte Carlo számításban N értéke nagy, így tehát a korrekciót elhanyagoljuk.

Relatív szórást ebből kifejezve adódik, hogy:

$$r^2 = \frac{D^2(\hat{I})}{\hat{I}^2} = \frac{\sum {C_i}^2}{(\sum C_i)^2} - \frac{1}{N}.$$

Abban az esetben, ha C_i lehetséges értékei:

$$C_i = \begin{cases} 1, \text{ha a minta a vizsgált tértartományon belül van} \\ 0, \text{ha a minta a vizsgált tértartományon kívül esik} \end{cases}$$

akkor a relatív szórás egyszerűsödik és átírható úgy, hogy az (i,j) hisztogramelem relatív statisztikus szórásnégyzete a következő módon becsülhető:

$$r_{ij}^{Stat^2} = \frac{1}{M} - \frac{1}{N},$$

ahol M a vizsgált területre beérkező mintaelemek száma. Ebből az abszolút hiba már könnyedén megkapható:

$$D_{ij}^{Stat^2} = r_{ij}^{Stat^2} \cdot F_{ij}^{avg^2} = \left(\frac{1}{M} - \frac{1}{N}\right) \cdot F_{ij}^{avg^2}.$$

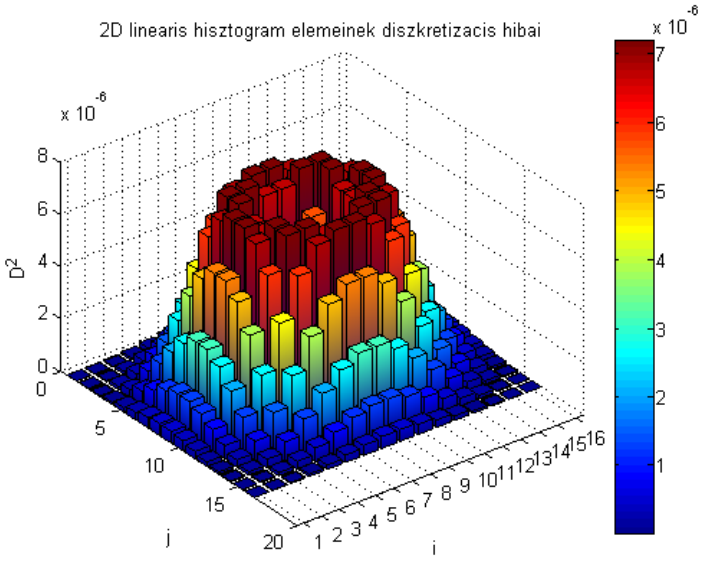
2.3. Hibavizsgálatok

A hisztogram minden cellájára meghatározható a diszkretizációs és a statisztikus hiba. Ezek viszonyát a vizsgált mintaszám és az állandó cellaméretű hisztogram felbontása fogja meghatározni. A jelenlegi számítások során mivel ismert a vizsgált sűrűségfüggvény, a relatív és abszolút hibák analitikusan is meghatározhatóak.

Az analitikus számítások esetében $f(x,y)$ sűrűségfüggvényből és N mintaszámból pontosan meghatározható az egyes hisztogramelemekbe beérkező mintaszámok várható értéke, és ezekből becsülhető az egyes cellák várható hibái is:

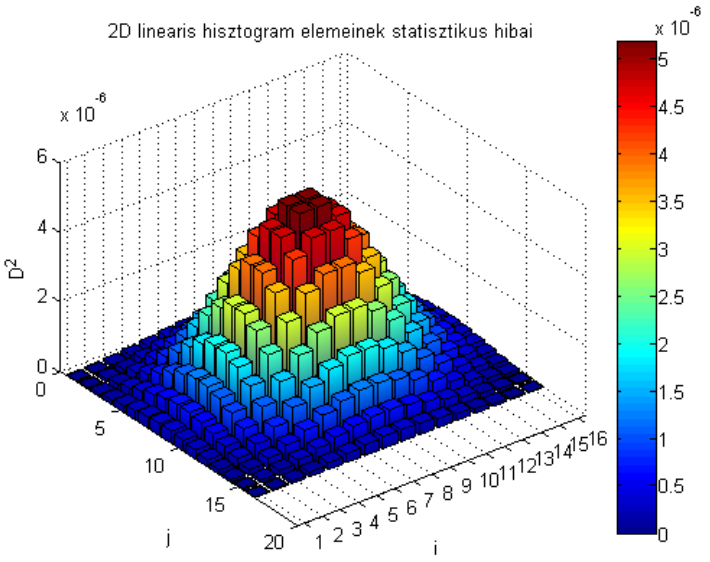
$$F_{ij} = N \cdot \int_{x_i}^{x_{i+1}} \int_{y_i}^{y_{i+1}} f(x,y)\, dx\, dy$$

$$F_{ij}^{avg} = \frac{F_{ij}}{A \cdot N}$$



3. ábra N = 300 000 minta 16 · 16 felbontású hisztogramba helyezésével keletkező abszolút diszkretizációs hiba

A 3. ábrán az egyes hisztogram cellákra meghatározott abszolút diszkretizációs hibák négyzetei láthatóak, az elemek indexeinek függvényében. Az $x\epsilon(-3,3)$ és az $y\epsilon(-3,3)$ vizsgált tartományt mindkét tengely mentén $n = 16$ részre osztottuk fel. Látható, hogy a diszkretizációs hiba a 2D Gauss-görbe esetén akkor válik naggyá, amikor a görbe meredeksége nagy.



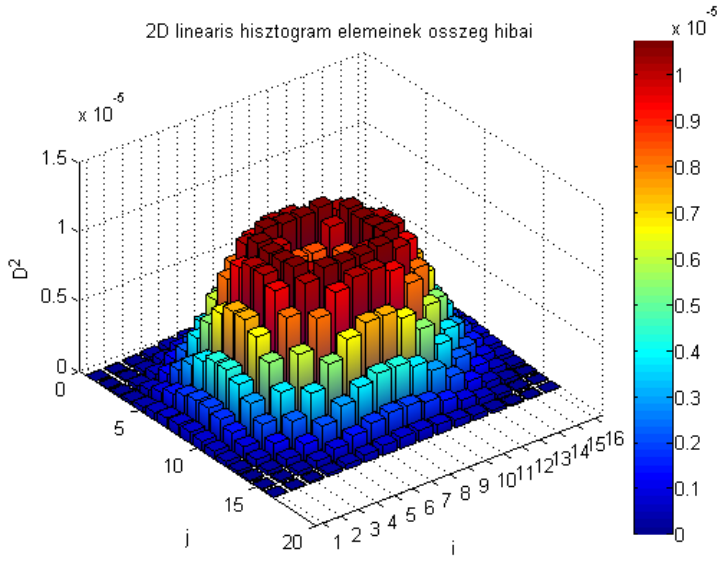
4. ábra N = 300 000 minta 16 · 16 felbontású hisztogramba helyezésével keletkező abszolút statisztikus hiba

Az előzőekhez hasonlóan az egyes hisztogram cellákhoz a statisztikus hibát is meghatározhatjuk. A 4. ábra alapján látható, hogy az abszolút statisztikus hiba négyzete a függvény értékével arányos. A relatív hiba nagy N esetében fordítottan arányos $f(x,y)$ értékével, mert ekkor $1/N \to 0$ és $M \propto f(x,y)$.

$$r_{ij}^{Stat^2} = \frac{1}{M} - \frac{1}{N} \propto \frac{1}{f(x,y)}$$

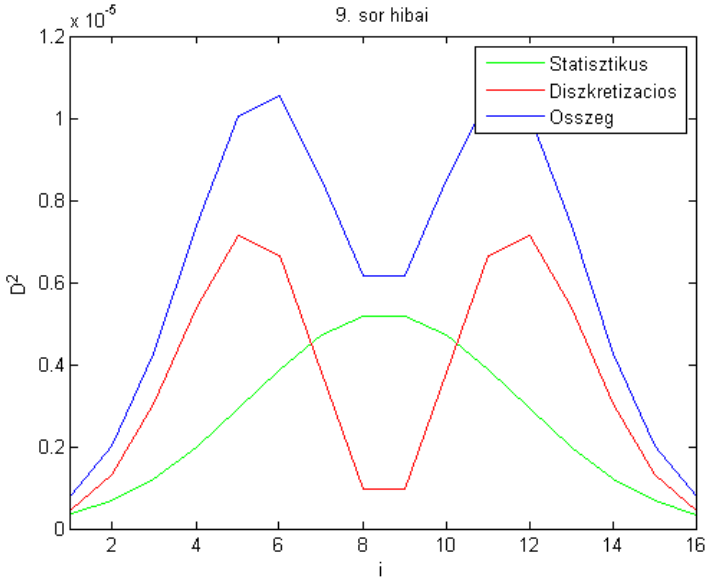
Az $F_{ij}^{avg^2} \propto f^2(x,y)$, ezért az abszolút hiba négyzete $f(x,y)$-nal lesz arányos:

$$D_{ij}^{Stat^2} = r_{ij}^{Stat^2} \cdot F_{ij}^{avg^2} \propto f(x,y)$$



5. ábra N = 300 000 minta 16 · 16 felbontású hisztogramba helyezésével keletkező abszolút összeghiba

Az abszolút hibák négyzetesen összegezhetőek, így a hisztogram celláinak teljes hibája a statisztikus és a diszkretizációs hibák négyzeteinek összegéből megkaphatjuk. 5. ábra mutatja, hogy az egyes hisztogram elemekhez tartozó összeghiba hogyan adódik ki, melynek $i = 9$. sori metszetét a 6. ábra mutatja.



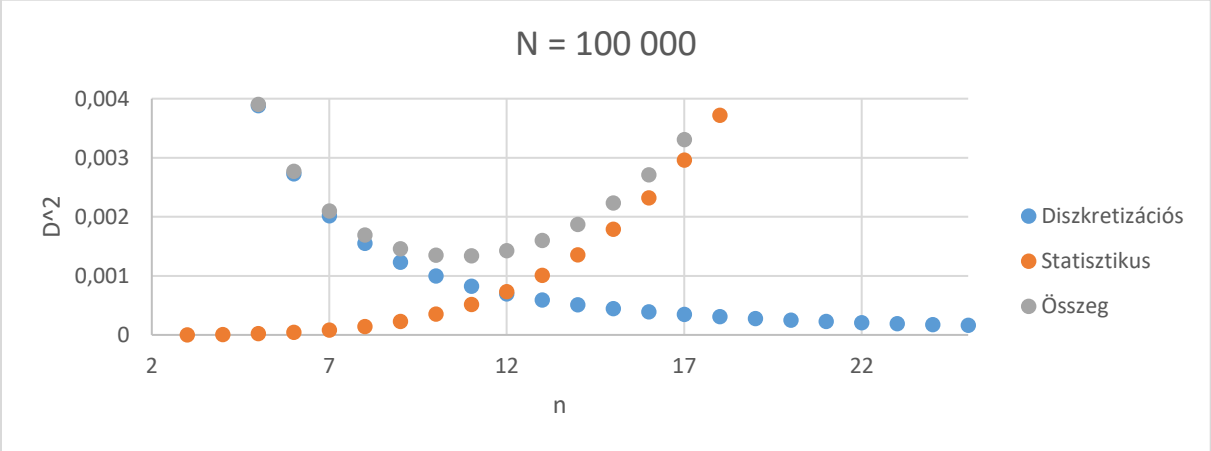
6. ábra N = 300 000 véletlen pont 16 · 16 felbontású hisztogramba helyezésével keletkező hibák viszonyai (metszet i = 9)

2.4. Ideális felbontás meghatározása

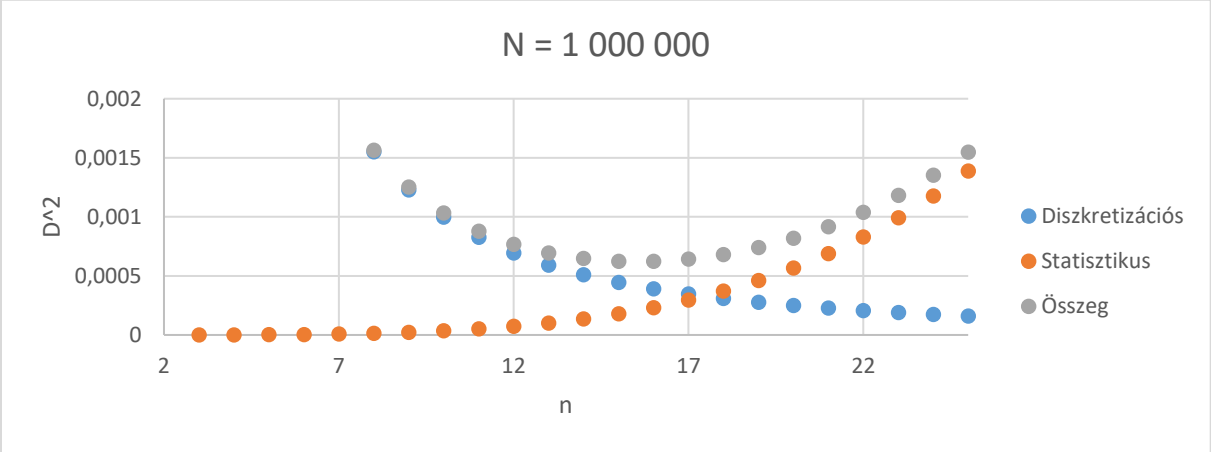
A statisztikus hiba nagysága csökkenthető az N mintaszám növelésével, a diszkretizációs hiba pedig az n felbontás növelésével. Azonban egy rögzített N mintaszám esetében, ha a felbontást növeljük, akkor a statisztikus hiba növekedni fog, mert az adott cellákba már kevesebb minta fog bekerülni. Ezzel azt várhatjuk, hogy egy adott N mintaszámhoz található egy optimális n felbontás, ahol a hisztogram teljes hibája minimális.

A vizsgálatok során az optimum meghatározásához egy hisztogram különböző abszolút hibáit az egyes cellákra felösszegeztük, és ezeket összehasonlítottuk, így jellemezve az adott felbontású hisztogramot.

$$D_{Sum}^2 = D^{Stat^2} + D^{Det^2} = \sum_{j=1}^{n_x}\sum_{i=1}^{n_y} D_{ij}^{Stat^2} + \sum_{j=1}^{n_x}\sum_{i=1}^{n_y} D_{ij}^{Det^2}$$

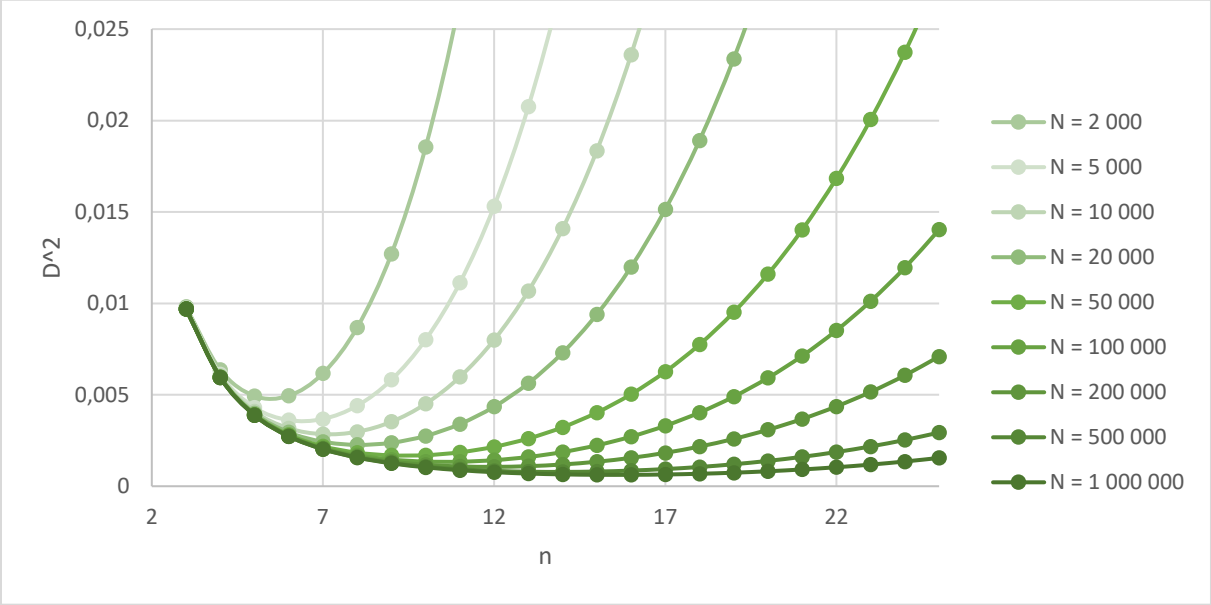


7. ábra Különböző felbontás mellett az abszolút hibaösszegek alakulása rögzített (N = 100 000) mintaszám esetében



8. ábra Különböző felbontás mellett az abszolút hibaösszegek alakulása rögzített (N = 1 00 000) mintaszám esetében

Jól látható, a 7. ábra és a 8. ábra alapján, hogy rögzített N mintaszám esetén, a felbontás növelésével a diszkretizációs hiba csökken, a statisztikus hiba pedig növekszik, mert egy cellába kevesebb minta kerül be, de a felosztás egyre finomabb. Egy adott mintaszám esetén tehát mindig található egy optimális n felbontás ahol D^2_{Sum} a teljes abszolút hiba összeg minimális lesz.



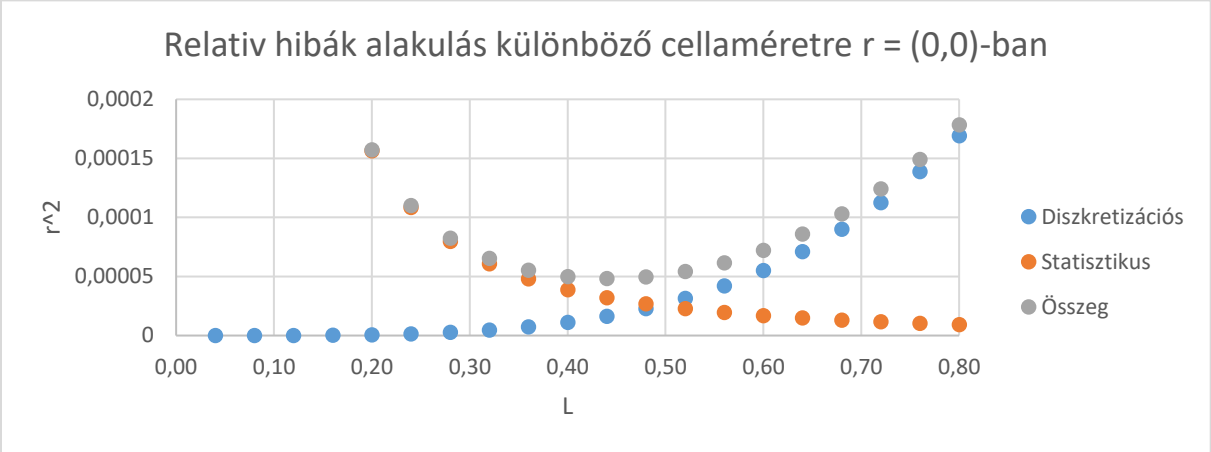
9. ábra Összeghibák alakulása különböző mintaszámokra és felbontásokra

A 9. ábrán a D^2_{Sum} teljes abszolút összeghibák láthatóak különböző N mintaszámok esetén az n felbontás függvényében. Látható, hogy D^2_{Sum} minimumánál található ideális felbontás a mintaszám növelésével növekszik.

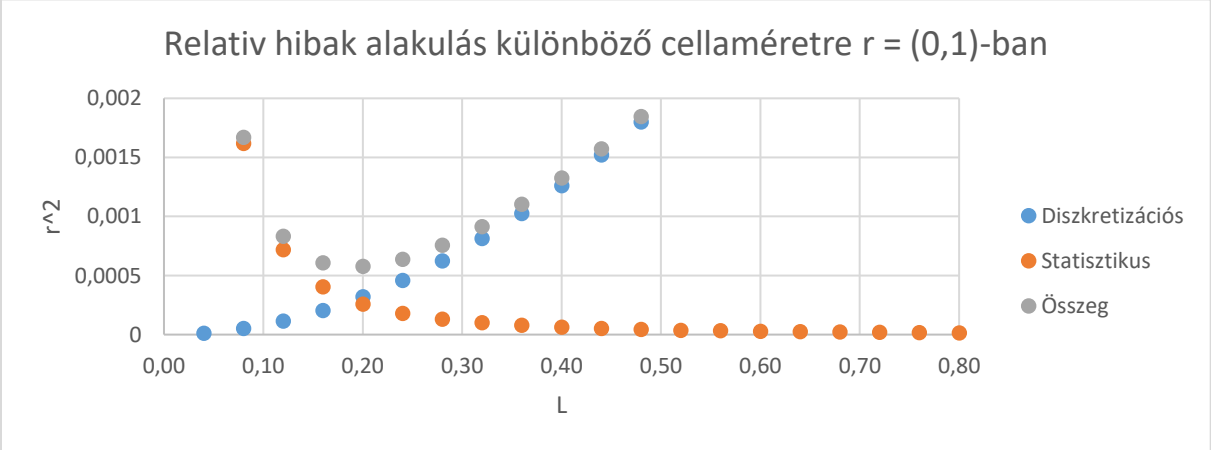
2.5. Egy cella hibájának vizsgálata

Állandó cellaméret esetén a hisztogram különböző pontjain a cellák hibái különbözőképen alakulnak. Azokon a helyeken ahol a sűrűségfüggvény értéke kisebb ott kevesebb minta fog beérkezni, így az ahhoz tartozó relatív statisztikus hiba nagyobb lesz. A diszkretizációs hiba pedig azokon a helyeken válik naggyá, ahol a sűrűségfüggvény gyorsan változik. Azzal hogy azonos cellaméretet választunk a teljes tartományon, a különböző helyeken található cellák esetén a hozzájuk tartozó hibák eltérőek lesznek.

Vizsgálat céljából készthetünk egy elemi cellát melynek mérete $A = L \cdot L$, ahol L a cella szélessége. Ezt a cellát a 2D síkunkra bárhova elhelyezhetjük és megnézhetjük a beérkező mintaszám alapján a különböző hibák alakulását. Az előző vizsgálatokhoz hasonlóan a mostani számításokat is analitikus módszerekkel hajtottuk végre, a gyors és pontos számítások érdekében. Megnéztük normális eloszlású függvény esetén, hogy a sík különböző pontjára helyezett, eltérő méretű cellák hibái hogyan alakulnak.

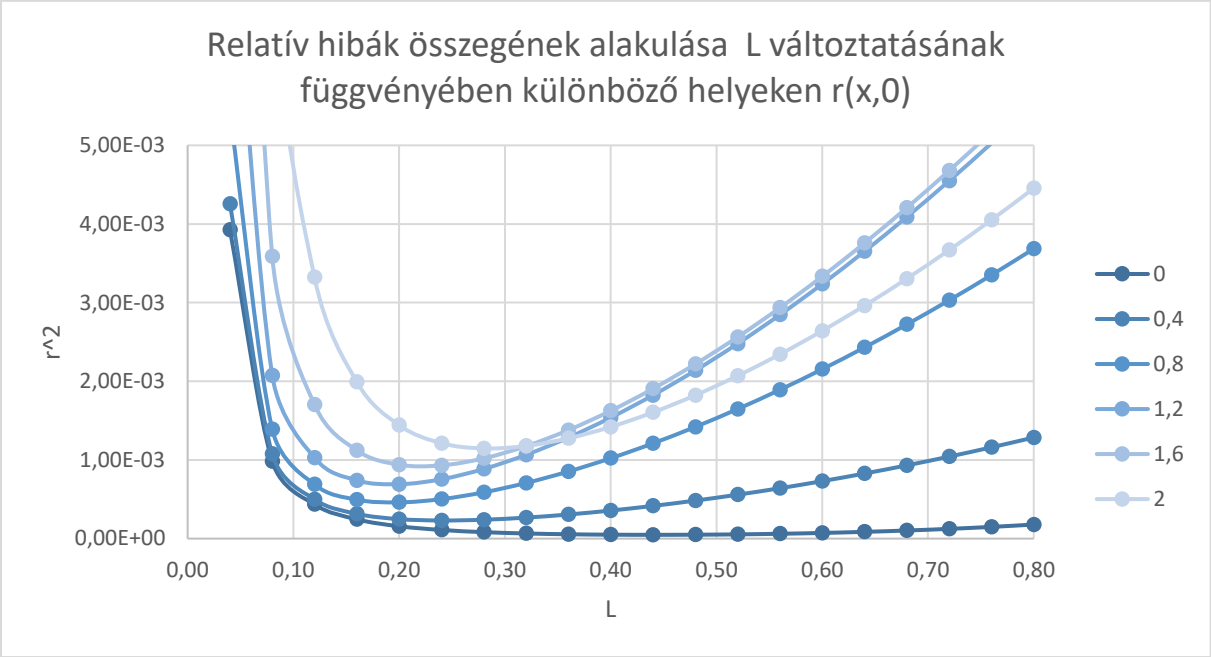


10. ábra Relatív hibák alakulása az $r = (0,0)$ pontban vett különböző $A = L \cdot L$ méretű cellák függvényében

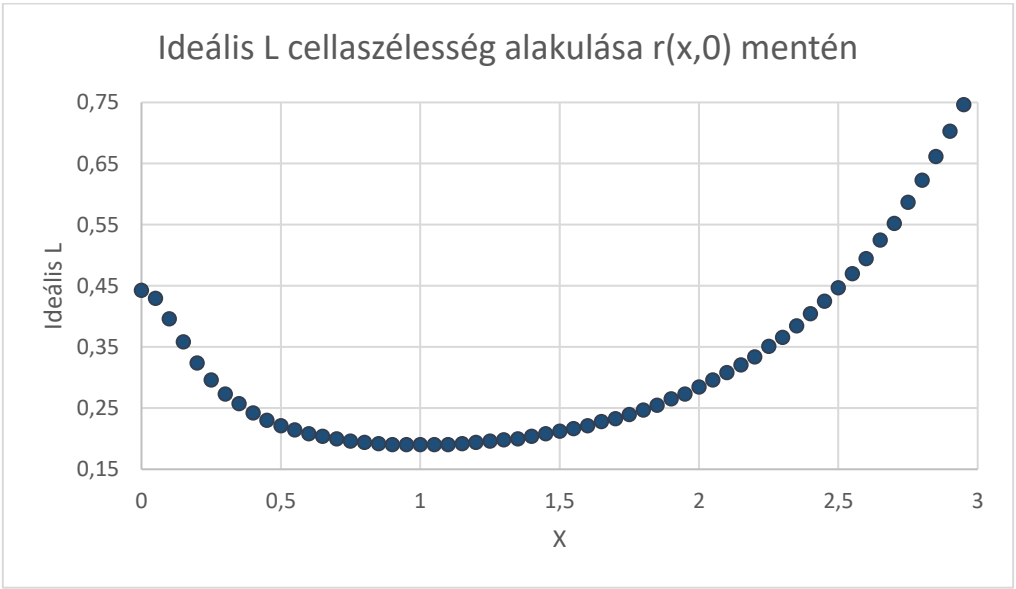


11. ábra Relatív hibák alakulása az $r = (0,1)$ pontban vett különböző $A = L \cdot L$ méretű cellák függvényében

A 10. ábra és 11. ábra a relatív hibák alakulását mutatja két különböző helyen a cellaméretek függvényében, egyiknél a 2D Gauss-görbe közepénél, másiknál pedig a szélén. Figyeljük meg, hogy a hibák jellegre hasonlóan viselkednek a két helyen, de nagyságukban jelentősen eltérnek, és így az optimum, ahol a két hiba összege minimális máshol található. Ebben az esetben is minden koordináta pozícióhoz található egy optimális cellaméret, ahol a statisztikus és a diszkretizációs hiba összege minimális lesz.



12. ábra Relatív hibák összegének alakulása különböző $A = L \cdot L$ méretű cellák függvényében az x tengelyen különböző pontjai mentén



13. ábra Az ideális cellaszélesség alakulása az x tengely mentén

A 12. ábra és a 13. ábra alapján láthatjuk, hogy kezdetben, a normális eloszlás sűrűségfüggvényének közepénél, érdemes nagyobb cellaméretet választani, ugyanis itt a függvény értéke nagy és nem változik jelentősen. Majd kifelé haladva először, a függvény értéke gyorsan változik, így egyre kisebb cellaméretet érdemes választani, mert a diszkretizációs hiba megnő és ahhoz, hogy a függvényváltozását jól lekövessük, kisebbre van szükség. Majd tovább haladva, amikor már nem változik a függvény nagymértékben és az értéke is kicsi, érdemes egyre nagyobb méretet választani, hogy a statisztikus hibán javítsunk. Ezek alapján látható, hogy érdemes egy olyan hisztogramot készteni melynek cellaméretei nem állandóak.

3. Hisztogramok eltérő méretű cellákkal

A cél egy olyan algoritmus fejlesztése volt, amely különböző többdimenziós sűrűségfüggvényekkel mintavételezett pontokra képes mindig egy olyan eltérő cellaméretű hisztogramot készíteni, amely valamelyik hibamennyiségre optimalizál. Az adaptív struktúra kialakításához sok különböző osztásfeltétel implementálható, mi a mediánnál történő felezést választottuk fő vizsgálatainkhoz, mely a statisztikus hiba csökkentésére alkalmas. Célunk volt többek között az is, hogy a megírt kódba könnyedén lehessen más osztásfeltételeket beilleszteni, és így bővíteni tudjuk a vizsgálati lehetőségeinket. A mediánnal történő felezés esetén analóg szimulációkra minden cellájában azonos lesz az elemszám, ezzel biztosítani tudjuk, hogy az egyes cellák relatív statisztikus hibája állandó legyen.

3.1. MATLAB vizsgálatok és a kezdeti algoritmus

A MATLABban implementáltuk az első algoritmust, amely mediánfelezés alapján hozza létre az egyes cellák határait. Az algoritmus képes 2D mintákhoz adaptív rácsot létrehozni és a létrajött struktúra hibáit becsülni. A program a diszkretizációs hibára semmilyen optimalizálást nem alkalmaz, csak és kizárólag a relatív statisztikus hibát fogja állandó értéken tartani.

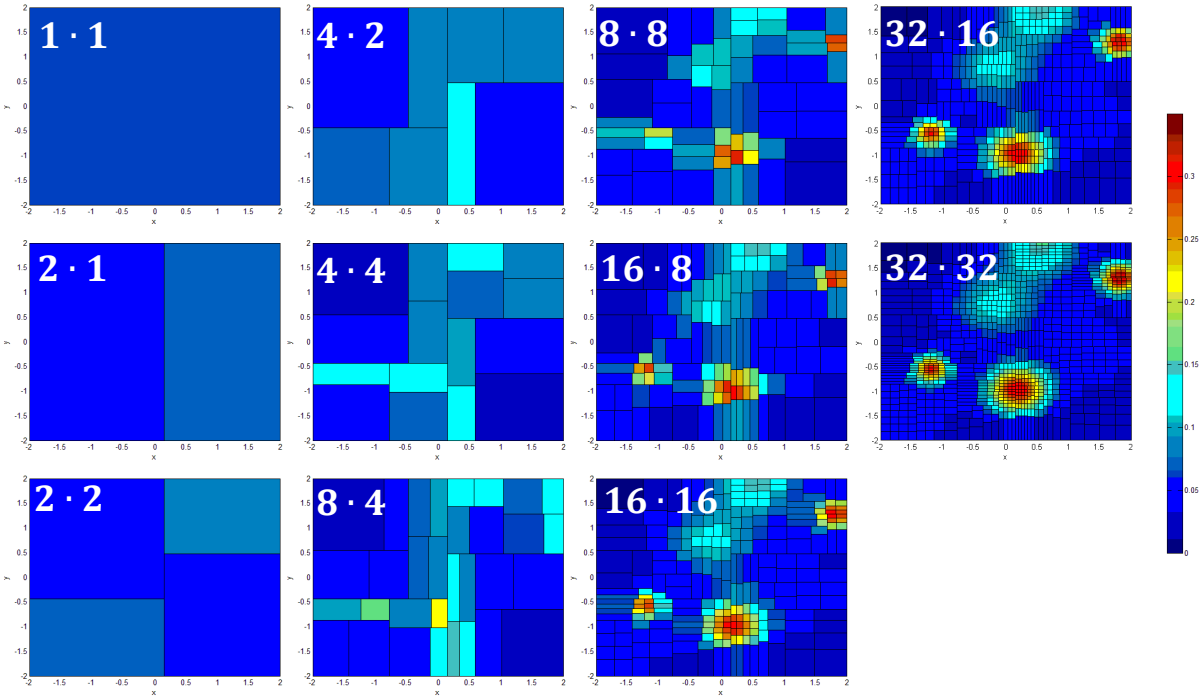
3.1.1. Algoritmus működése

Az algoritmus lényege a következő. Legyen N db mintánk, mely M dimenzió tekintetében szeretnénk hisztogrammá alakítani. A mintahalmazt először két részre osztjuk az általunk választott első dimenzió szerint úgy, hogy e dimenzió szerint a minták koordinátái alapján meghatározzuk azok mediánját. Ez a medián lesz az így létrejött két cellát elválasztó határ. Mindkét cellát osszuk egymástól függetlenül újabb két-két cellára, a második általunk választott dimenzió mentén a két mintacsoport mediánját megkeresve. Folytassuk ezt a módszert a többi dimenzió szerint is, az M. dimenziót elérve újrakezdhetjük az első dimenzióval.

Két dimenziós illusztráció gyanánt a 14. ábra mutatja, hogy a beolvasott mintahalmaz alapján egy adaptív hisztogram hogyan alakul ki lépésről lépésre. Az első egységbe beolvasott mintahalmaz alapján kapott becslés a bal felső ábrán látható, ami alatt az első felezést követően kialakult struktúra kapott helyet. A 2 dimenzió mentén a mediánnal történő felezésekkel jól látható, ahogy egyre jobban felveszi a struktúra a sűrűségfüggvény alakját. Továbbá az is megfigyelhető, hogy azokon a területeken, ahová több minta érkezik (azaz az eloszlás sűrűségfüggvénye nagyobb), ott nagyobb felosztást alkalmaz az adaptív struktúra.

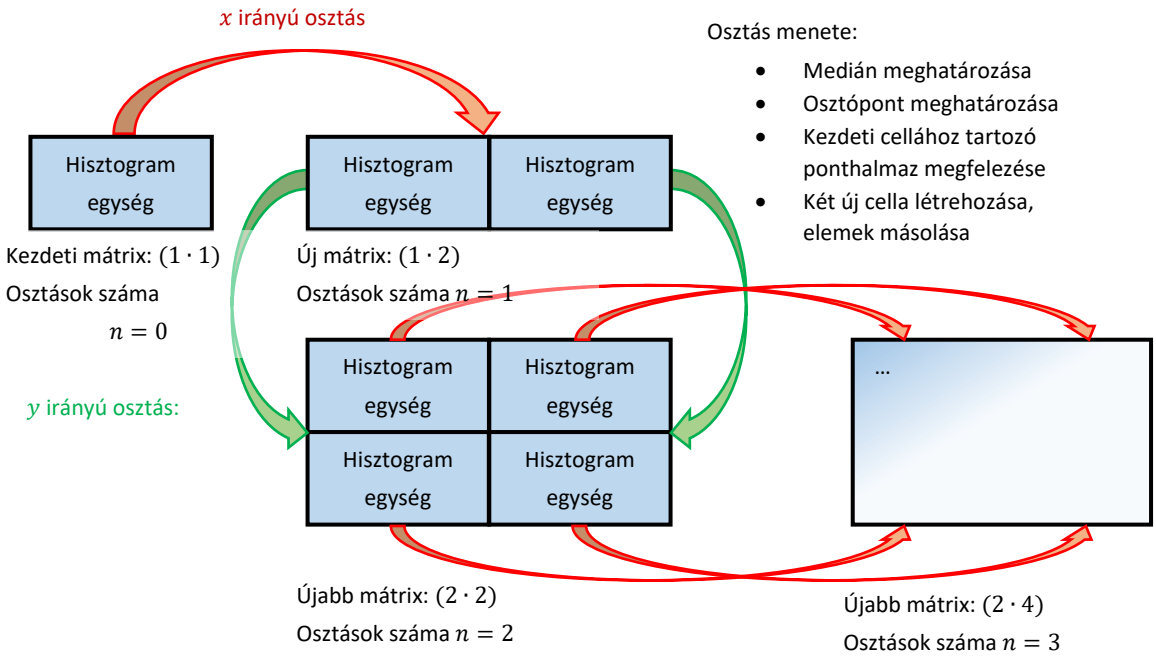
A vizsgált minta 10 darab véletlenszerűen választott várható értékű, véletlenszerűen választott szórású normális eloszlásból született 10^8 mintával.

Azzal, hogy az osztáspontot a mediánnál választjuk meg, biztosítjuk, hogy minden hisztogramegységbe azonos számú minta kerüljön. Természetesen más osztásfeltétel is implementálható, ezzel befolyásolható az adaptív hisztogram alakulása.



14. ábra 2D struktúra által adott becslés alakulása lépésről lépésre

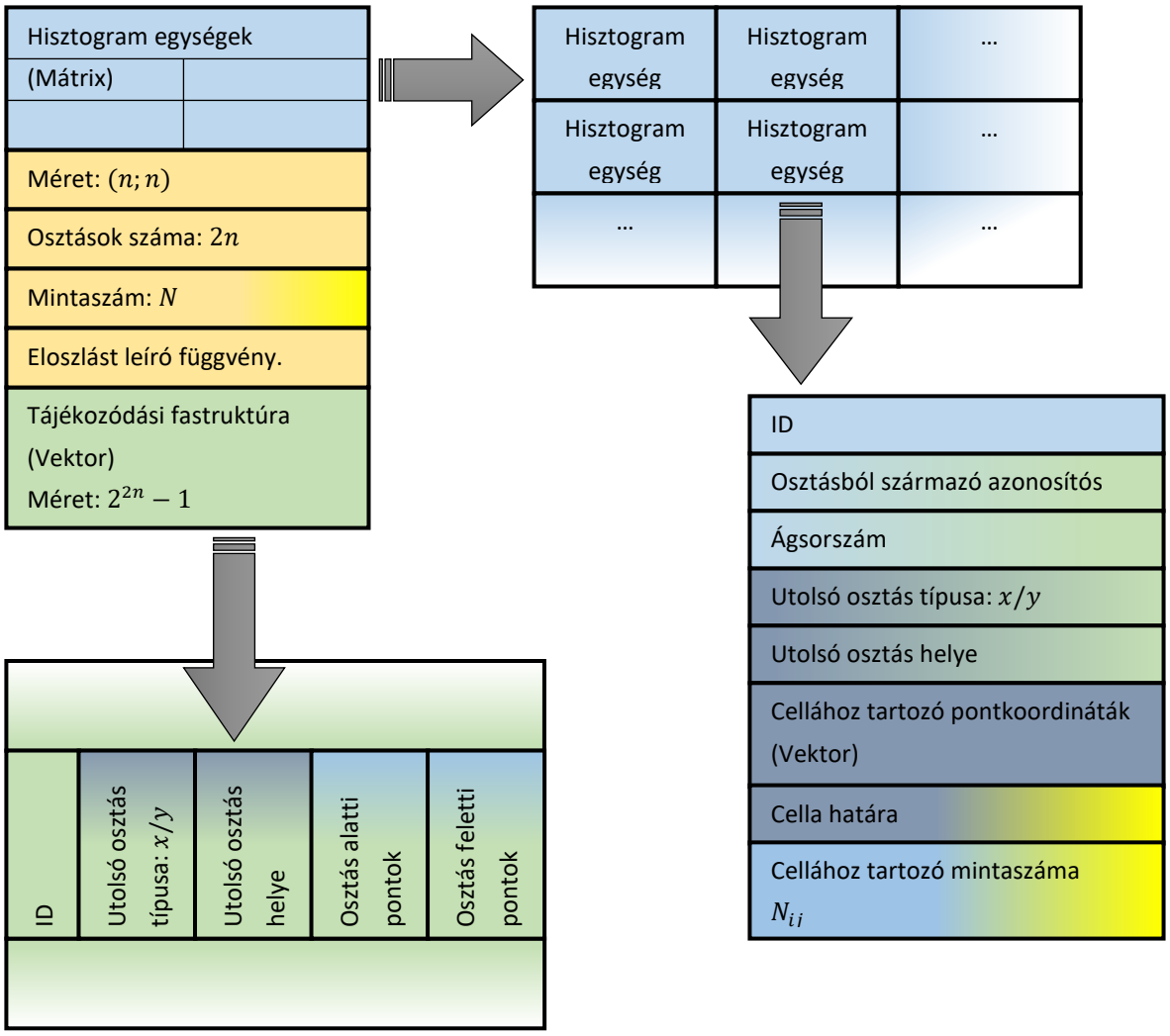
A struktúra létrehozásához az objektum konstruktorában át kell adni egy adott 2D-os mintahalmazt, amit az algoritmus automatikusan az első hisztogramegységbe helyez. Ezt követően az objektum egy függvényének hívásával lehet a struktúra egységeit kettéosztani, ezzel létrehozva az újabb nagyobb felbontású struktúrát. Az algoritmus minden egységen egyesével végigmegy, és mindegyikhez megkeresi a benne található pontok alapján az adott egység mediánját. A medián meghatározásához a cellában található pontok adatsorát sorba rendezi, majd a középső két elemet kiválasztja. Az osztáspontot a két kiválasztott elem átlagánál fogja meghatározni. Miután ezzel végzett, létrehozza az új struktúrában a két új hisztogramegységet, és azoknak a határait az osztáspont és korábbi határok szerint beállítja. Miután ez megtörtént, az adott pontkoordináták másolásra kerülnek az újonnan létrehozott elemekbe, az osztás helye alapján szétválasztva. Minden egység hordoz egy változót, mely megmutatja, hogy utoljára melyik dimenziója mentén történt az osztás. Ennek a változónak segítségével dönt a program a következő osztás típusáról. A 15. ábrán az algoritmus első három felezési folyamata látható.



15. ábra Felező algoritmus működésének sematikus ábrázolása

3.1.2. Felezőalgoritmust használó hisztogramstruktúra felépítése

A program által létrehozott objektum strukturális felépítését a 16. ábra mutatja. A program inputjába fogadja a pontkoordinátákat tartalmazó vektort, és a vizsgált 2D tartomány határoló vonalait. A struktúra konstruktora hozza létre ezt az objektumot.



16. ábra Felező algoritmust használó hisztogramkészítő program objektumainak felépítése

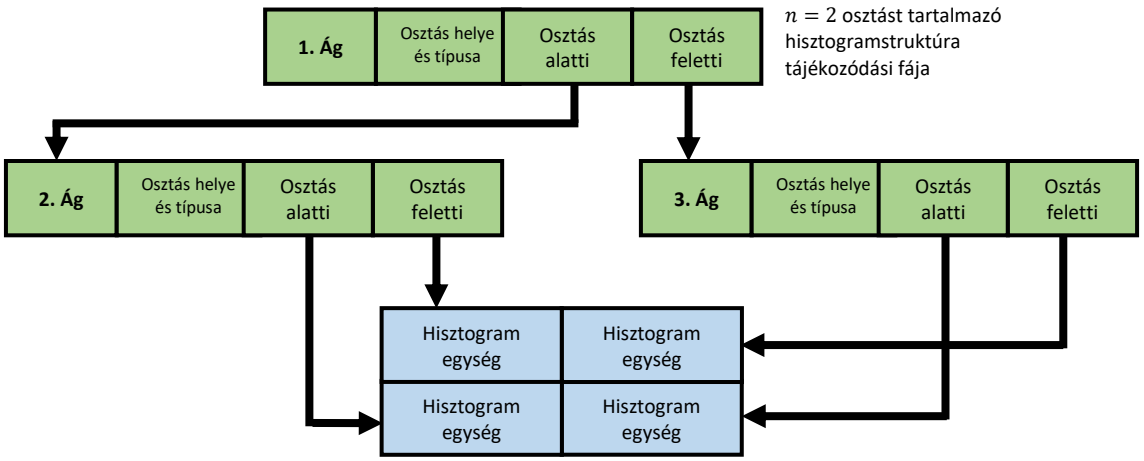
A struktúra mátrixosan tartalmazza a hisztogramot felépítő egységeket, amelyek alapvetően a hisztogram celláival egyeznek meg, de ezen felül más futást segítő információkat is tartalmaznak. Egy hisztogramegység tartalmazza a cella határait, a cellába kerülő pontkoordinátákat, ezenkívül még hibakeresést és az analízist elősegítő információkat foglal magába. A struktúrába még néhány futást segítő változó van definiálva, ilyen például a felbontás, az osztások száma és a teljes mintaszám. Lehetőségünk van megadni a hisztogramstruktúrának a minták sűrűségfüggvényét, melynek segítségével a struktúra diszkretizációs hibája becsülhető, és összehasonlító számítások végezhetők el. Ezen felül a létrehozott objektumhoz tartozik még egy tájékozódási fa, melynek alkalmazási lehetőségeire és annak módjának ismertetésére a 3.1.3. fejezetben fogok kitérni.

3.1.3. Tájékozódási fa

A programba implementálásra került egy tájékozódási fa struktúra, ami két dologra is használható: segítségével egy adott (már előzőek alapján létrehozott) struktúrához lehet további mintákat hozzáadni, ezzel növelni a becslés pontosságát, és a sűrűségfüggvényt mintavételezni.

Működése

Egy struktúrához tartozó fa egy vektorban ágelemeket tartalmaz. Egy n darab osztás során létrejött struktúra 2^n hisztogramegységből épül fel, melynek tájékozódási fájához $2^n - 1$ darab ágelem tartozik, melyek az osztások során a struktúra növekedésével jönnek létre. Ha a vizsgált tértartományon választunk egy pontot, akkor a fa ágain lépkedve n darab lépést megtéve egy, a megadott ponthoz tartozó hisztogramelemhez jutunk el. Ehhez minden ágelem változókban hordozza az egyik osztás típusát, helyét és azoknak az ágelemeknek a sorszámát, melyek az adott osztásból származóan az osztáspont alá, illetve fölé keletkeztek. Az ágakon haladva az n. lépésben mindig olyan elemhez jutunk, amely már nem egy ágelem változójára mutat, hanem egy hisztogramegység azonosítójára. A 17. ábra egy példát mutat be egy $n = 2$ osztást tartalmazó hisztogramstruktúra tájékozódási fájára.



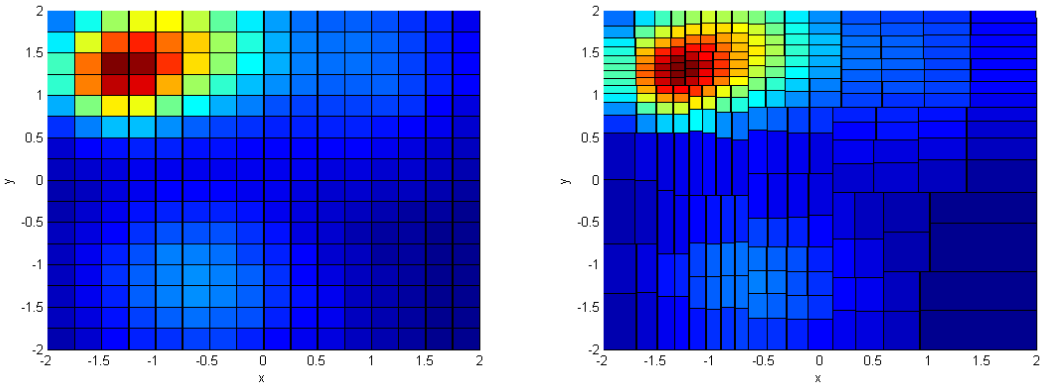
17. ábra n=2 osztást tartalmazó struktúra tájokozódási fájának felépítése

Sűrűségfüggvény mintavételezése

Az így elkészült hisztogram nem csak a minták eloszlásának vizualizálására alkalmas, hanem könnyedén újramintavételezhető is. Egy adott problémán létrehozott hisztogramstruktúra a sűrűségfüggvénynek egy közelítését adja, ezt a közelítést a tájékozódási fa segítségével mintavételezni is lehet. Abban az esetben, ha a fa ágelemein véletlenszerűen lépkedünk (minden elágazásnál 0 vagy véletlen számmal sorsolunk), és döntünk, hogy merre megyünk tovább, akkor egy véletlenül választott hisztogramegységhez jutunk, melynek a határain belül egyenletesen egy pontot választva az eredeti sűrűségfüggvény becslésének egy mintájához jutunk.

3.1.4. Összehasonlító vizsgálatok

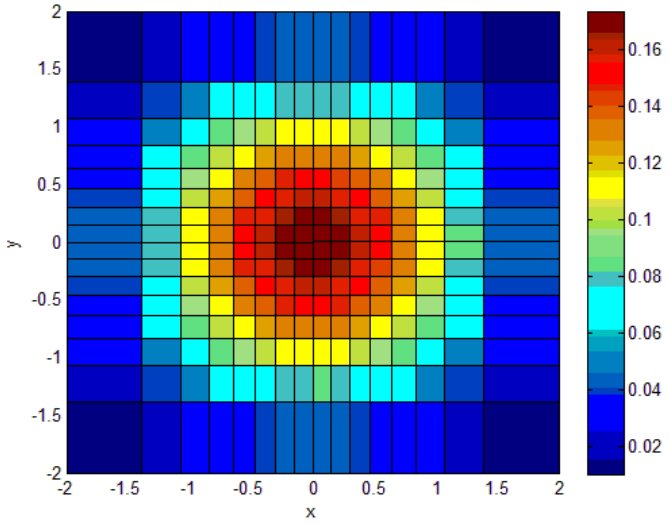
A MATLABban implementált algoritmus által adott becsléseket összehasonlítottuk a strukturált rácson kapott eredményekkel. 18. ábrán egy véletlenszerűen létrehozott 2 dimenziós sűrűségfüggvénnyel vett mintahalmazon a két rács összehasonlítása látható. Az adaptív módszer azokon a helyeken, ahol a sűrűségfüggvény értéke nagy volt, több felosztást alkalmazott, míg azokon a helyeken, ahol az érték kisebb, ott nagyobb méretű cellákat alakított ki.



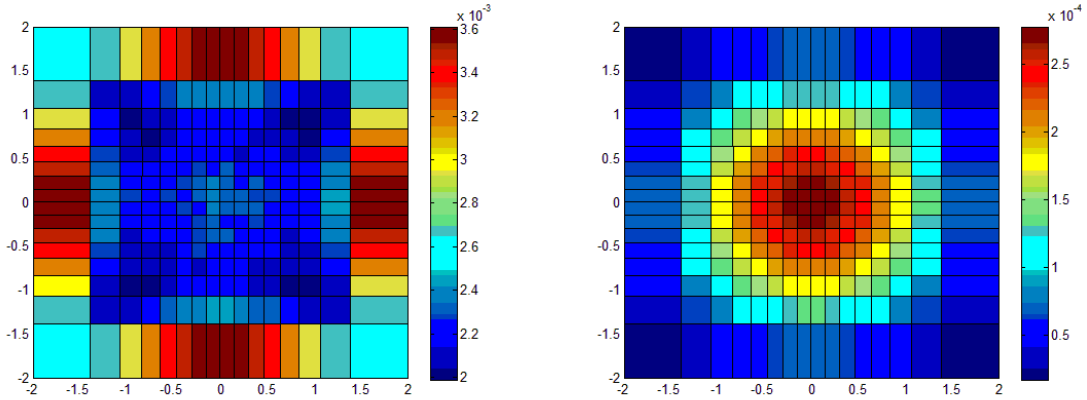
18. ábra A strukturált (bal oldalon) és az adaptív (jobb oldalon) felosztás vizuális összehasonlítása egy véletlen adathalmazra

Adaptív felosztás hibáinak becslése

Az adaptív módszer esetén is megnéztük (hasonlóan a strukturált rácshoz), hogy a statisztikus, illetve a diszkretizációs hiba hogyan változik. Példaképpen itt is a 2 dimenziós normális sűrűségfüggvényt vettük, melynek adaptív felbontását a 19. ábra mutatja.



19. ábra 2D normális eloszlású mintákra készített $(16 \cdot 16)$ adaptív hisztogram



20. ábra normális eloszlású mintákra készített $(16 \cdot 16)$ adaptív hisztogram abszolút diszkretizációs (bal oldalon) és statisztikus (jobb oldalon) hibája

Az adaptív struktúra létrehozása után pedig az abszolút diszkretizációs és statisztikus hiba meghatározható, melyeket a 20. ábra mutatja. Az abszolút diszkretizációs hiba az adaptív módszer esetében ott válik naggyá, ahol a sűrűségfüggvény értéke alacsonnyá, de a függvény értékeiben még jelentősebb változás van jelen.

Az abszolút statisztikus hiba esetén pedig egy függvénnyel arányos képet kaptunk. Az adaptív struktúra esetén, mivel minden cellába azonos mennyiségű minta kerül, azaz $M = állandó$, és ${}^{1}/_{N} \to 0$, a relatív statisztikus hiba négyzete állandó lesz minden cella esetén:

$$r_{ij}^{Stat^2} = \frac{1}{M} - \frac{1}{N} = konst.$$

Így az abszolút statisztikus hiba négyzete a függvény négyzetével lesz arányos:

$$D_{ij}^{Stat^2} = r_{ij}^{Stat^2} \cdot F_{ij}^{avg^2} \propto \big(f(x,y)\big)^2.$$

3.2. A C++-ban implementált algoritmus

Ahhoz, hogy vizsgálatokat végezhessünk valódi Monte Carlo szimulációkból kapott mintákon is, implementáltam az algoritmust C++ nyelven, és módosítottam annak strukturális felépítését, ezzel lehetővé téve, hogy több problémához is alkalmazható lehessen. Ebben a fejezetben bemutatom a főbb változásokat és azok gyakorlati hasznukat.

Az algoritmust általánosítottam n dimenzióra, így a felhasználó tetszőleges dimenziós fázistérrel rendelkező problémát megadhat. Az újabb C++-os programverzióban a hisztogramegységek, melyek a cellákat tartalmazzák, már nem mátrixos alakban, hanem egy vektorban vannak tárolva. A felhasználó egy problémát definiáló inputfájlban állíthatja be, hogy mely dimenziókra és milyen sorrendben felezze a program a fázisteret. A futási idő optimalizálása érdekében több felhasználásra nem kerülő változót kivettem (ezek elsősorban a MATLABos verzió esetén a hibakeresést segítették elő), és a mediánnál történő felezéséhez használt algoritmust is módosítottam. A medián meghatározásához az eredeti kód a részecskéket tartalmazó vektort a kijelölt dimenzió szerint sorba rendezte, majd a két középső elemet kiválasztva átlagszámítással meghatározta az osztáspontot. Az újabb verzióban a részecskéket tartalmazó vektor nem kerül teljes rendezésre, mert a C++ *algorithm* könyvtárában található *nth_element* függvény segítségével lehetőség van egy vektor adatait addig rendezni, míg csak az n-edik, azaz egyik megadott elem a helyére nem kerül, ezáltal rövidítve a futásidőt. Az így félig rendezett adatsorban az n-edik elem előtti elemek mind kisebbek lesznek n-nél, és az n utániak pedig nagyobbak. Az algoritmus mindig az adatsor felénél található elemet állítja be a helyére, majd a második felében lévő elemek közül megkeresi a legkisebbet, és ezeknek az elemeknek az átlagaként határozza meg az osztáspontot (az így kiválasztott két elem persze ugyanaz, mint a teljes rendezés után az adatsor felénél található két elem).

Az új program tartalmaz egy ábrázoló függvényt, mely segítségével a strukturált vagy adaptív rácson becsült sűrűségfüggvényt egy képfájlba lementi, és így a felhasználónak lehetőséget ad a vizuális megjelenítésre.

A program két adatfájltípust tud beolvasni: az MCNP6 program által generált, úgynevezett bináris PTRAC fájlt és az általános ASCII fájlt, mely a részecskéket sorokba rendezve a dimenzióit tabulátorokkal elválasztva tartalmazza.

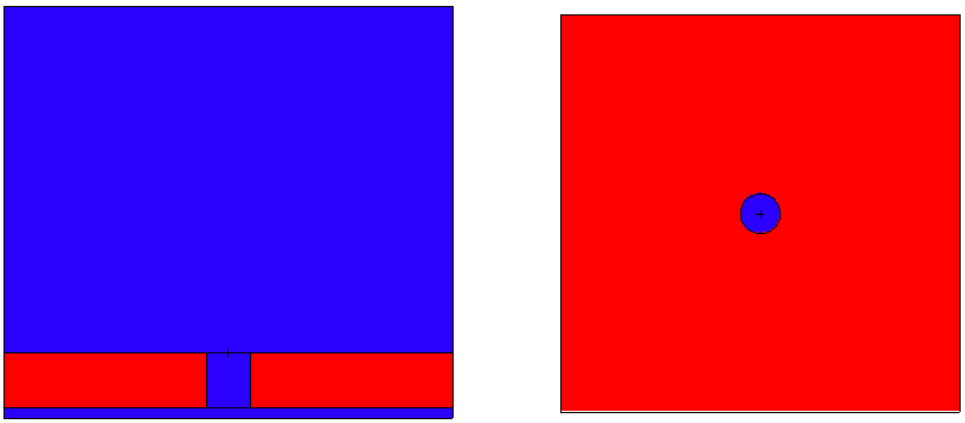
Az említett MCNP (Monte Carlo N-Particle) egy háromdimenziós magas szintű részecsketranszport Monte Carlo kód. Segítségével a felhasználó különböző geometriai elrendezéseket hozhat létre, melyben a program véletlenszerűen indított részecskék útját modellezi. Továbbá képes az egyes részecskék élete során bekövetkezett, a felhasználó által kiválasztott eseményeket egy Particle Track Output (PTRAC) fájlba kiírni. A PTRAC fájl segítségével lehetőség van az MCNP-ből a részecske-trajektóriák kicsatolására, és így a különböző eloszlások vizsgálatára. [4]

3.3. Vizsgálatok C++-ban implementált algoritmussal

Az MCNP segítségével létrehoztam két különböző geometriát, melyekben fotontranszportot szimuláltam, majd a különböző események (PTRAC fájl segítségével történő kicsatolásával) térbeli sűrűségfüggvényének meghatározását végeztem.

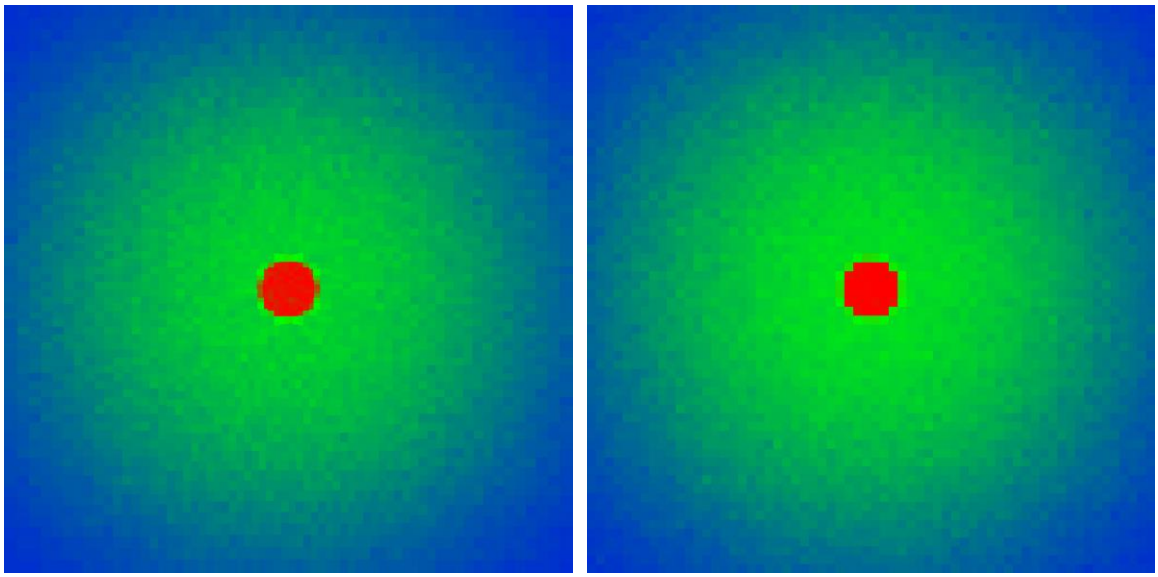
3.3.1. Síkon foton fluxus meghatározása („lyukas lemez” geometria)

A kód numerikus visszaellenőrzése céljából, először egy olyan geometriát készítettem (lásd 21. ábra), melyben egy $5cm$ vastag lyukas alumínium lemez helyezkedik el, amin a lyuk átmérője $4cm$. A lemez felette $30cm$-rel egy $1MeV$-es izotróp foton forrás van, és a köztes teret pedig levegő tölti ki. A vizsgálat során a közvetlenül a lemez alatti síkon az adaptív, illetve a strukturált ráccsal meghatároztam a foton fluxusát, majd összehasonlítottam azokat egymással és az analitikus számítással.



21. ábra A geometria felépítse
XZ metszet (bal oldal), XY metszet (jobb oldal)

A szimuláció során $2 \cdot 10^8$ fotont indítottam a forrásból, melyből 5921719 érte el a vizsgálathoz kijelölt (azaz az alumínium túloldalán elhelyezkedő) felületet. Ezt követően a programom segítségével elkészíttettem egy $64 \cdot 64$-es strukturált és egy $64 \cdot 64$-es adaptív rácsot.



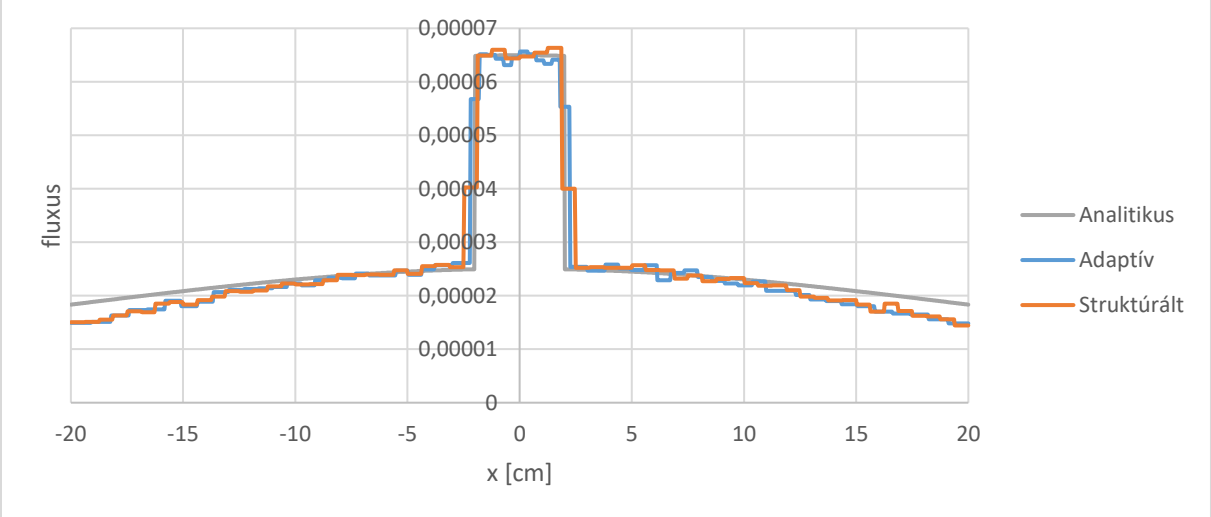
22. ábra Számított fluxus ábrázolása 2D-os síkon
adaptív ráccsal (bal oldal), illetve strukturált ráccsal (jobb oldal)

A 22. ábrán láthatjuk a strukturált, illetve az adaptív rács által készített becsléseket. Figyeljük meg, hogy az adaptív rács kicsivel jobban felosztja a lyuk környékét és az ívét, mint a strukturált rács. Jelen probléma esetében a hisztogramok által adott becslést analitikus számításokkal is összevetettük. A foton fluxusát analitikusan a következő módon határozhatjuk meg:

$$\phi = \frac{1}{4\pi r(x,y)^2} \cdot e^{-\Sigma d(x,y)}$$

$r(x,y)$ – a forrás és a vizsgált sík távolsága

$d(x,y)$ – az alumíniumban megtett úthossz



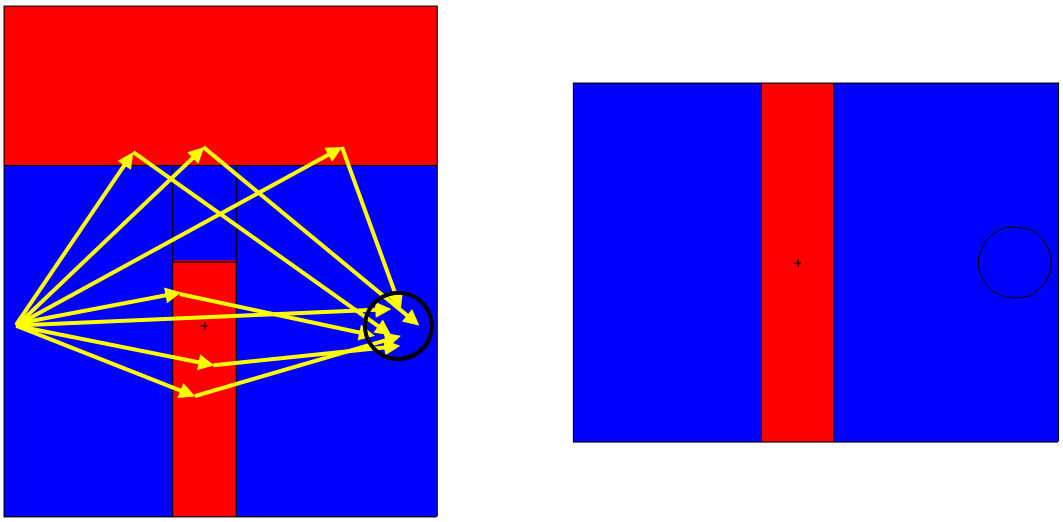
23. ábra Számított fluxusok összehasonlítása 1D metszeten

A 23. ábrán láthatjuk az analitikus számítás által adott, és a két különböző rács segítségével becsült fluxusokat. Mindkét módszer jól visszaadta az analitikus számítás eredményeit.

3.3.2. Szórási események térbeli sűrűségének meghatározása („rés a falon” geometria)

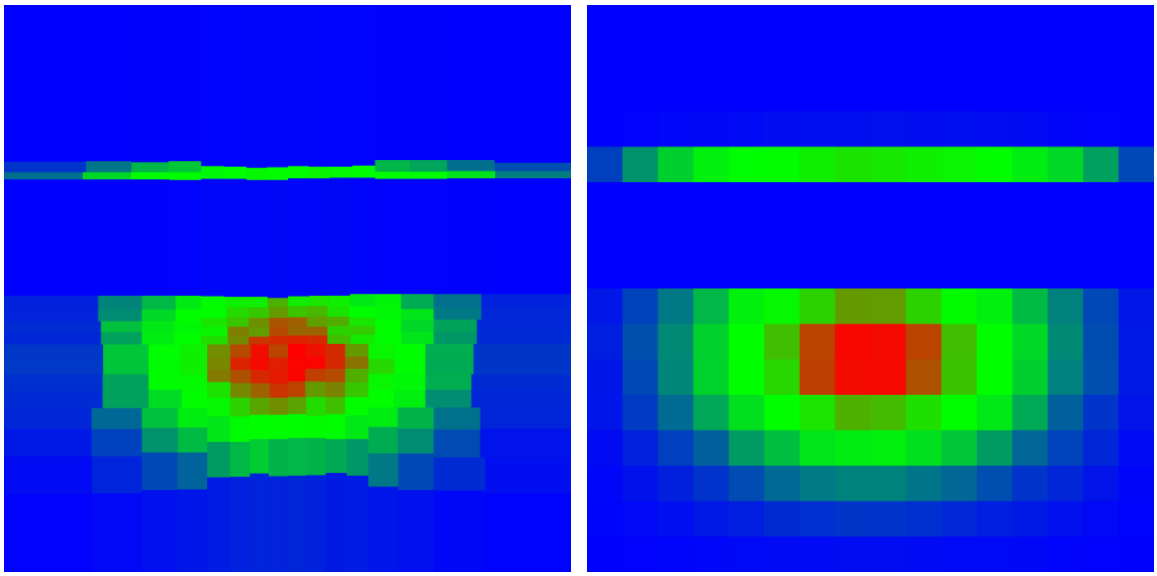
Egy valós probléma elemzési lehetőségeit demonstrálandó, a második geometriában azoknak a fotonoknak szórási eseményeinek térbeli sűrűségét vizsgáltam, melyek a forrásból kiindulva átmennek egy detektorfelületen. Elképzelhetjük például, hogy egy forrás árnyékolását szeretnénk megoldani valamilyen magasságú és vastagságú fallal, a szimulációval pedig azt szeretnénk kideríteni, hogy a részecskék a falon keresztüli transzmisszióval vagy a plafonon történő szóródással jutnak el az árnyékolandó területre. Fontos továbbá megtudnunk, hogy az átjutó részecskék mely része milyen térbeli eloszlással rendelkezik, hogy eldönthessük, az árnyékolást hol javítsuk.

A 24. ábrán látható geometriába elhelyeztem egy $10cm$ vastag függőleges alumínium falat, a geometria tetejébe pedig egy $25cm$ vastag vízszintes alumínium tetőt. A fal és a tető között $15cm$-es rés van. Bal oldalra ismételten egy $1MeV$-es izotróp foton forrás került, a geometria jobb oldalára pedig egy detektor gömbfelületet raktam, melynek sugarát $5cm$-re választottam. Azoknak a fotonoknak a szórási eseményit szűrtem ki a PTRAC fájlba, melyek a forrásból elindulva valamilyen útvonalon, a detektor felületén átmennek. Ez két útvonalon történhet meg: vagy a falon keresztül mennek a fotonok, vagy a tetőről visszaszóródnak a detektor gömbfelületébe. Néhány lehetséges trajektóriát az ábrán jelöltem.



24. ábra A geometria felépítse
XZ metszet (bal oldal), XY metszet (jobb oldal)

A szimuláció során 10^9 fotont indítottam a forrásból, melyből 3051317 szórási esemény keletkezett a vizsgálathoz. Ezt követően a programom segítségével elkészíttettem egy $16 \cdot 16 \cdot 16$-es strukturált és egy $16 \cdot 16 \cdot 16$-es adaptív rácsot.



25. ábra szórási események sűrűsége YZ metszeten
adaptív ráccsal bal oldalon, strukturált ráccsal jobb oldalon

A 25. ábrán egy YZ metszeti rajza látható a szórási események térbeli sűrűségéből. Látható, hogy az adaptív rács sokkal nagyobb felbontást választ azokon a területeken, ahol a sűrűségfüggvény nagy, így jobban fel is tudja bontani a tetőben és falban bekövetkező szórási eseményeket is.

Egy hasonló elemzés eredményeképpen például a felhasználó megállapíthatja, hogy a forrás árnyékolásának érdekében inkább az árnyékoló fal megerősítése célszerű, vagy inkább a rés mellett a féltéren, például tetőről pattanó részecskék adják az átjutó részecskék zömét, így itt érdemes komolyabb árnyékolásra gondolni.

4. Összefoglalás

A munkám során egy többdimenziós adaptív hisztogramkészítő algoritmust fejlesztettem ki. A program a felhasználó által kijelölt dimenziókon, a medián mentén többször megfelezi a teret, így létrehozva az adaptív rácsot. Az ilyenfajta technika problémák numerikus és vizualizációs elemzésére ad módot, továbbá felhasználható eredményeket ad további számítások kiindulópontjaként is.

Az első MATLABban írt programverzióval összehasonlító számításokat végeztem 2D mintákkal a strukturált és az általam fejlesztett programmal készített adaptív rácson. Meghatároztam, hogy a statisztikus és a diszkretizációs hiba hogyan viselkednek különböző paraméterek változtatására.

Ezután C++-ban egy általánosabb algoritmust írtam, mely képes tetszőleges dimenzióban a fázistér felosztására, ezenkívül képes a felhasználó által generált mintákon kívül, akár az MCNP által létrehozott részecske minták feldolgozására is, így lehetőséget ad realisztikus nukleáris szimulációk adatain történő vizsgálatokra. Az így létrejött függvényközelítés tulajdonságait elemeztem és összehasonlítottam a szabályos rács segítségével nyert eloszlás tulajdonságaival. A létrejött struktúrára kidolgoztam egy módszert mellyel az eloszlást akár mintavételezni is lehet.

A mostani medián felező technikában további módosítást, optimalizálást lehetne végrehajtani a futási idő csökkentése érdekében. Az algoritmus nem képes a nagy letöréseket jól felbontani, ez javítható egy olyan plusz feltétel beépítésével, mely tovább osztja azokat a cellákat, melyek egy küszöbértéknél jobban eltérnek a szomszédos celláktól.

Ezenkívül más osztásfeltételek implementálása, azok tesztelése is egy további fejlesztési lehetőség. A program úgy lett felépítve, hogy könnyedén lehet például az átlagnál történő osztásfeltételt az algoritmusba beépíteni, és ezt a mediánnal történő osztással összehasonlítani.

További célunk, hogy a 3.1.3 fejezetben írt tájékozódási fa segítségével történő mintavételezést elemezzük. Létező probléma az olyan geometriák készítése, melyet MCNP segítségével csak több különálló részletben lehet szimulálni. Ilyenkor szükség van a geometriákat határoló síkokon történő részecskék mintavételezésére, melyek az általános módszerek alapján jelentősen korreláltakká válnak. A programmal történő mintavételezéssel ez a korreláció valószínűleg megszüntethető.

5. Irodalomjegyzék

[1] W. R. Martin, J. P. Holloway, K. Banerjee, J. Cheatham és J. Conlin, Global Monte Carlo Simulation, University of Michigan, 2007.

[2] B. Kaushik, Kernel Density Estimator Methods for Monte Carlo Radiation Transport, The University of Michigan, 2010.

[3] D. Dannheim, A. Voigt, K.-J. Grahn, PeterSpeckmayer és TancrediCarli, „PDE-Foam—A probability density estimation method using self-adapting phase-space binning," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment,* pp. 717-727, 2009.

[4] M. C. Team, MCNP6 User's manual, 2013.

[5] E. F. Shores, C. J. Solomon és J. D. Zumbro, „Radiographic test problem for MCNP and other mesh-based applications," *Progress in Nuclear Science and Technology Volume 4,* pp. 502-506, 2014.

[6] D. W. Scott és S. R. Sain, „Multidimensional Density Estimation," in *Handbook of statistics*, 2005, pp. 229-261.