

Sejtautomaták

Tóth Balázs



Természettudományi kar
Eötvös Loránd Tudományegyetem
Magyarország
2020 május 15

1 Elméleti bevezetés

1.1 Conway-féle életjáték

A laborgyakorlat során C++ nyelven kellett megvalósítanunk a Conway-féle életjátékot, aminek a szabályai röviden a következők:

- ha n élő szomszéd van, akkor nem változik a sejt állapota
- ha $n+1$ élő szomszéd van, akkor a sejt élő lesz, függetlenül az eddigi állapotától
- ha a fenti közül egyik feltétel sem teljesül, akkor a sejt elpusztul.

A szimuláció során a vizsgálatokat az $n = 1, 2, 3, \dots, 8$ esetekre kellett elvégeznünk különböző kezdeti feltételekkel. Ezeknek az elvégzéséhez négy fajta változtatható peremfeltételt kellett implementálnunk:

- nyílt peremfeltétel
- periodikus peremfeltétel
- élő határ
- a peremen véletlenül sorolt állandó állapot

1.2 2D-homokdomb modell

A gyakorlat másik nagy témája a homokdomb modell vizsgálata volt, pontosabban a skálázási szabály lemerése, valamint a skálázási törvény kitevőjének meghatározása.

A realisztikus két dimenziós homokdomb modell megvalósításához két dimenziós homokszem tömbökre van szükségünk. Minden oszlopban deklarálnunk kell egy vektor dőlést x és y irányba.

A Bak-Tang-Wiesenfeld modelben csak egy darab skalár dőlés van, ami 8 diszkrét értéket vehet fel. $s_{i,j} = 0, 1, 2, 3, 4, 5, 6, 7$, minden i, j pozícióban az oszlop instabilnak tekinthető, ha $s_{i,j} > 3$. Abban az esetben ha bármelyik oszlop is instabil az egész domb instabilnak számít.

A homokdomb a következő lokális törvény segítségével frissül:

- ha $s_{i,j} > 3$, akkor el kell távolítani 4 egységnyi dőlést az i, j oszlopból és minden szomszédos oszlophoz egyet hozzá kell adni
- A törvényt egyidejűleg kell végre hajtani az összes oszlopra és addig kell ismételni amíg stabil állapotot nem érünk el.

Az általunk vizsgált modell esetén a disztribúció a következő törvényt követi:

$$N(n_t) \sim \frac{1}{n_t^b}$$

Ahol n_t az omlás tetején lévő szemcsék.

2 Eredmények ismertetése

2.1 Életjáték

A Conway-féle életjátékot a weboldalon ismertetett szabályok alapján c++ nyelven valósítottam meg. A szimulációt felkészítettem a négyféle határfeltételre és a változtatható számú elő szomszédra. Ezek a paraméterek lettek a kódom bemenetei. Ezt kövözően a kapott eredmények feldolgozását python nyelven végeztem, ahol egy egyszerű script-tel végigmentem az összes kimeneten és .png formátumú fájlt hoztam létre belőlük. Ezt követően a kapott képeket összefűztem és .gif formátummá konvertáltam. Ezeket az eredményeket mind feltöltöttem a jegyzőkönyvem mellé. Az ábrák elnevezése a következő konvenciót követték: 1 halott peremfeltétel, 2 periodikus, 3 elő és végül 4 a random. Minden mappába szerepelnek az generált képek valamint a belőlük generált gifek.

2.2 2D-homokdomb modell

Ehhez a feladathoz sajnos nem tudok értékelhető eredményeket felmutatni, mivel a hét közepén lebetegedtem és nem tudtam vele tovább foglalkozni.