MŰEGYETEM 1782

Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Számítástudományi és Információelméleti Tanszék

Kvantum gráfbolyongások

MSc Önálló laboratórium 1.

Készítette
Nemkin Viktória

Konzulens
dr. Friedl Katalin

2021. május 9.

Tartalomjegyzék

1. fejezet

Bevezetés

A 20. századi fizika hatalmas változásokat hozott. A relativitáselmélet mellett megjelent a kvantummechanika, mely teljesen megváltoztatta a világnézetünket. Ugyanebben az időszakban kezdődött el a számítástechnika hajnala is. Megjelentek az első számítógépek és elkezdték megírni az első programokat, algoritmusokat.

Richard P. Feynman 1982-es cikkében fejtette ki, hogy a klasszikus számítógépekkel sajnos csak exponenciális időben lehet kvantumjelenségeket szimulálni, ez azonban túlságosan lassú a kísérletek elvégzéséhez. Ha viszont lenne egy kvantumjelenségek alapján működő számítógépünk, akkor azzal hatékonyan lehetne szimulációkat végezni, a fizikai kutatások elvégzéséhez. Így született meg a kvantumszámítógép gondolata.

Benioff, Deutsch, majd Bernstein és Vazirani munkássága nyomán megszületett a kvantum számítási modell, a kvantum Turing-gép a 80-as évek végére. Ettől kezdve az a kérdés foglalkoztatja a kvantum algoritmusok kutatóit, hogy vajon vannak-e használható kvantum algoritmusok, illetve vannak-e olyanok amik jobbak mint a klasszikus párjaik.

Shor 1990-es kvantumon alapuló prímfaktorizációs algoritmusa már használható algoritmus lett és az RSA alapú kódolás feltörésével fenyeget, ami komoly veszélyt, illetve komoly előnyt is jelent azoknak akiknek van kvantumszámítógépük. Erre már felfigyeltek a nagyhatalmak, multinacionális cégek és szép lassan elkezdtek a kvantumszámítógépek kifejlesztésével foglalkozni.

2019-ben a Google quantum supremacy bizonyítéka mutatott egy olyan kvantumalgoritmust amely bár nem túl hasznos, de a mai legjobb szuperszámítógépeket is megverte a Sicamore processzoruk teljesítménye.

Manapság egyre forróbb témává válik a kvantum és Magyarországon is egyre több támogatás jut kvantummal kapcsolatos kutatásokra. A BME-n a Kvantuminformatikai Nemzeti Labor tevékenykedik kvantum titkosításon alapuló internet kifejlesztésével és több tanszék, köztük a SZIT is bekapcsolódott a projektbe.

Ezen dolgozat a kvantumalgoritmusokon belül a kvantum bolyongásokkal foglalkozik. A kvantum bolyongáson alapuló algoritmusokat azért érdemes kutatni, mert több ismert kvantum algoritmusnak az alapját képezik. A félév során megismerkedtem a kvantuminformatika alapjaival, a gráfbolyongások klasszikus illetve kvantumos változatával, valamint elkészítettem egy Pythonos keretrendszert melyben könnyen lehet gráfbolyongásokkal kapcsolatos kísérleteket, szimulációkat végezni.

A dokumentum hátralévő fejezeteiben bemutatom a kvantuminformatika alapjait, ismertetem a klasszikus gráfbolyongást, majd a kvantum bolyongás egy speciális esetét, bemutatom az elkészített Pythonos keretrendszert, majd ismertetem a félév során kapott szimulációs eredményeket, végül a továbbfejlesztési lehetőségekről beszélek.

2. fejezet

Klasszikus gráfbolyongások

Markov-láncok definíciója.

A Markov-láncokat irányított, súlyozott gráfokkal reprezentálhatjuk, egy Markov folyamat pedig ezen a gráfon végzett bolyongás.

Nagyon sok gyakorlati probléma kapcsolódik a Markov-folyamatokhoz: közgazdaságtan, játékelmélet, statisztikus fizika.

A klasszikus gráfbolyongási algoritmusok ezek mellett jól használhatóak gráfelméleti problémák közelítő megoldására. Nagy méretű gráfok esetén egy akár a csúcsszám méretében lineáris algoritmus lépésszáma is túlságosan nagy lehet. Például az internetes keresőalgoritmusok (PageRank) is ilyenek.

3. fejezet

Kvantuminformatika

Mielőtt a kvantumalgoritmusokkal elkezdhetünk foglalkozni, először meg kell ismernünk a kvantuminformatika eszköztárát. A klasszikus számítógépek esetében az információtárolás alapegysége a bit, melynek értéke lehet 0 vagy 1. Kvantumszámítógépek esetében az alapegység a kvantum bit, vagy röviden qubit. Kvantum bitek esetében a fizikai tulajdonságaiknak köszönhetően az állapottér a komplex számokon értelmezett. A legkisebb, kvantum számításra használható állapottér 2 dimenziós, ezt nevezzük qubitnek.

Definíció 1. Qubit A 2 dimenziós Hilbert-tér (H_2) egy vektorát qubitnek nevezzük. A tér bázisvektorai a $|0\rangle$ és az $|1\rangle$ ket vektorok.

Egy általános qubit tehát $c_0 \, |0\rangle$.

Definíció 2. Koordináta reprezentáció Egy általános qubit

Definíció 3. Kvantum regiszter

4. fejezet

Kvantum séta

Kvantum bolyongások tanulmányozása esetén először célszerű az egyenesen vett bolyongást tanulmányozni.

5. fejezet

Architektúra

Ebben a fejezetben bemutatom az elkészült keretrendszert.

Programozási nyelvnek a Python 3-mat választottam. Ennek oka az, hogy nagyon sok data science-el kapcsolatos modulja van, mely nagyban megkönnyíti a különböző matematikai, algoritmuselméleti problémák feltárását, könnyen iterálhatunk a különböző prototípusokon. Emellett a szintaxisa rövid, tömör, lényegretörő programkódok megírását teszi lehetővé.

A forráskód három nagy részre bomlik:

- Gráfmodellek
- Szimulátorok
- Futtatás, konfiguráció, eredmények ábragenerátora

5.1. Gráfmodellek

A félév során sokféle gráfon futtattam szimulációs kísérleteket, melyek során több problémába ütköztem. Kezdetben úgy oldottam meg a szimulációkat, hogy a cél gráfok szomszédossági mátrixait generáltam le, egyben a memóriában tartva azokat és a lépések során a megfelelő csúcshoz tartozó sorokat lekérdezve.

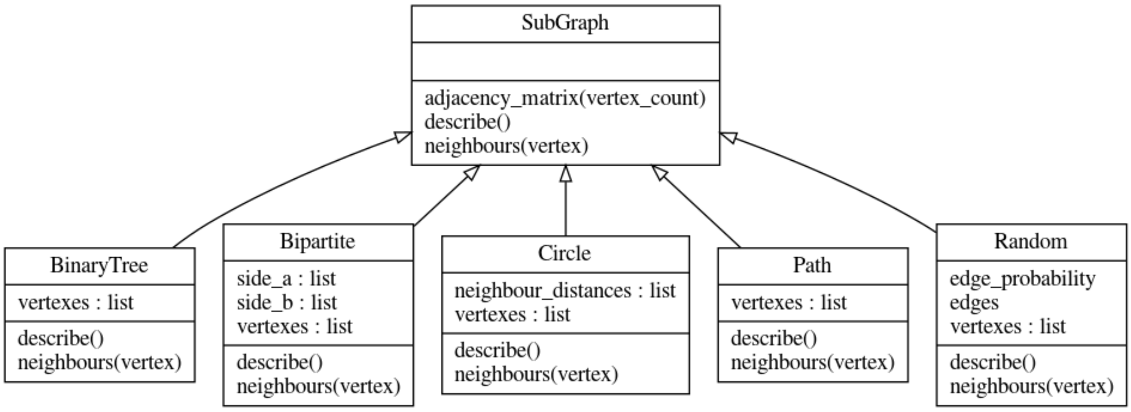
Ezzel a módszerrel több probléma is jelentkezett. Az első gondot az okozta, hogy a szomszédossági mátrix mérete a csúcsszám négyzetével arányos, ezért pár ezer csúcsú gráfot már nem tudtam a memóriában tartva szimulálni. A második probléma pedig az volt, hogy a szomszédossági mátrixos ábrázolás nagyon távol esett az emberi szempontból természetes ábrázolástól. A kvantum bolyongásos szimulációkat tipikusan nem véletlenszerű gráfokon szokták kipróbálni, hanem jól ismert struktúrával rendelkező gráfokon. Ilyen gráfok például a súlyzó vagy a ragasztott bináris fa gráfok.

A súlyzó gráf két egyforma méretű kört tartalmaz, mindkét körből kiválasztva k-k darab csúcsot, melyek teljes páros gráfot alkotnak (a súlyzó középső rúdját). A kör gráfokban pedig nem csak az egymás melletti csúcsok között fut él, hanem futhat él minden i. csúcs között is. A ragasztott bináris fa gráf pedig olyan, hogy két teljes bináris gráf leveleit szembefordítjuk és a két oldali levelek közé egy teljes páros gráfot készítünk.

A fenti leírásból látható, hogy az ember számára természetes leírás a gráfokat ismert részgráfok kompozitjaként adja meg. A félév során olyan architektúrát alakítottam ki a szimulációkhoz, mely ezt a szemléletet támogatja. A szomszédossági mátrixos tárolási mód helyett pedig a szomszédossági orákulum megközelítést használva nagyban csökkent a memóriaigénye az alkalmazásnak. Ennek a megközelítésnek a lényege, hogy az ismert struktúrájú gráfokra nem tárolok a memóriában szomszédossági információt, helyette biztosítok egy függvényt, amely a bemeneti paraméterként kapott csúcsindexre kiszámolja a vele szomszédos csúcsok indexeit.

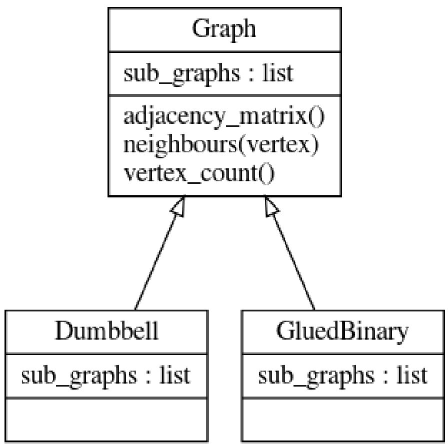
A félév során a következő nevesített részgráfok szomszédossági orákulumját implementáltam:

- BinaryTree
- Bipartite
- Circle
- Path
- Random



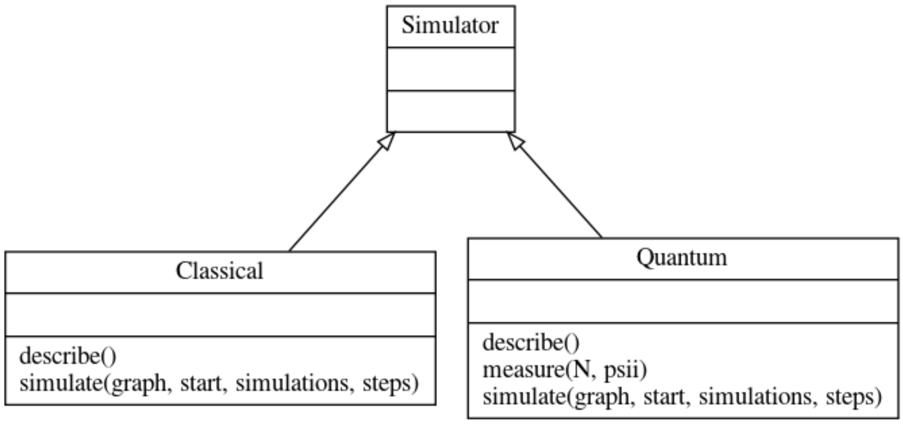
Ezen részgráfokból épülnek fel az alábbi kompozit gráfok:

- Dumbbell
- GluedBinary



5.2. Szimulátorok

A szimulátor osztályok közül a klasszikus tetszőleges kompozit gráfot tud fogadni, a kvantum szimulátor jelenleg a kvantum bolyongás egy speciális esetét, az egyenesen való bolyongást képes kezelni, mely a 2-regularitása miatt egyszerűbben implementálható. Hosszú távú cél a k-reguláris, illetve az általános gráfokra kiterjeszteni ezt a szimulátort.



5.3. Futtatás, konfiguráció, eredmények ábragenerátora

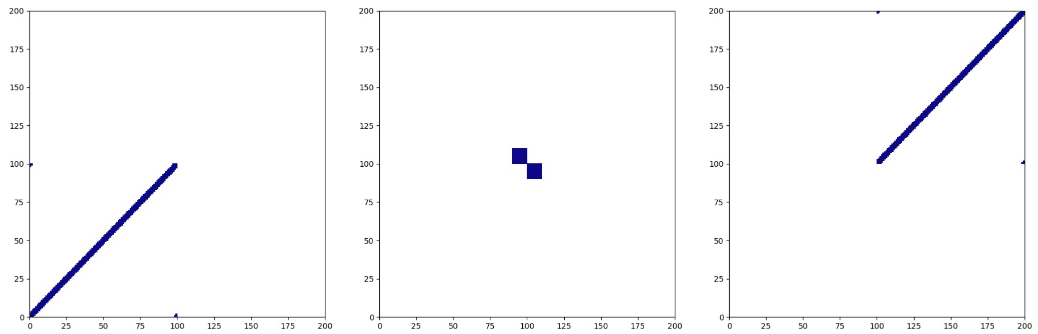
A fenti osztályok segítségével egy olyan keretrendszert alakítottam ki, melyben nagyon gyorsan fel lehet 1-1 futtatást konfigurálni. A futtatás eredményeit egy összesített Latex dokumentumba gyűjti a program. Ez tartalmazza a beadott gráf részgráfjainak nevesített típusát, szomszédossági mátrixait, illetve a teljes gráf szomszédossági mátrixát, valamint a szimulációk eloszlási eredményeit. A következő fejezetben több ilyen ábrát is bemutatok.

6. fejezet

Szimulációk, eredmények

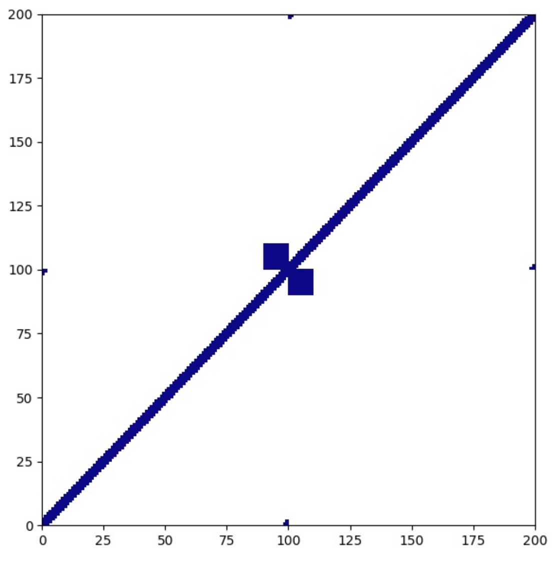
6.1. Súlyzó gráf

Például egy súlyzó gráfról a következő képek készültek:

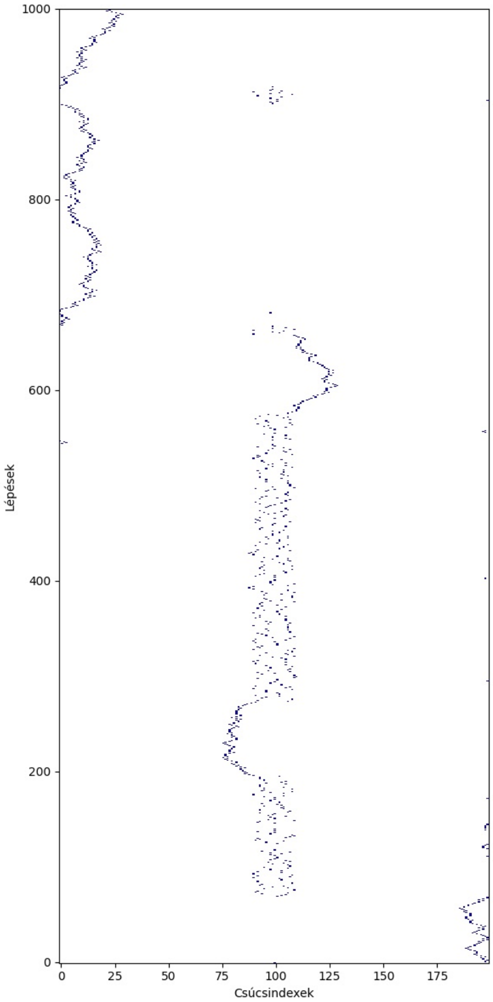


(a) Bal kör **(b)** Középső teljes páros gráf **(c)** Jobb kör

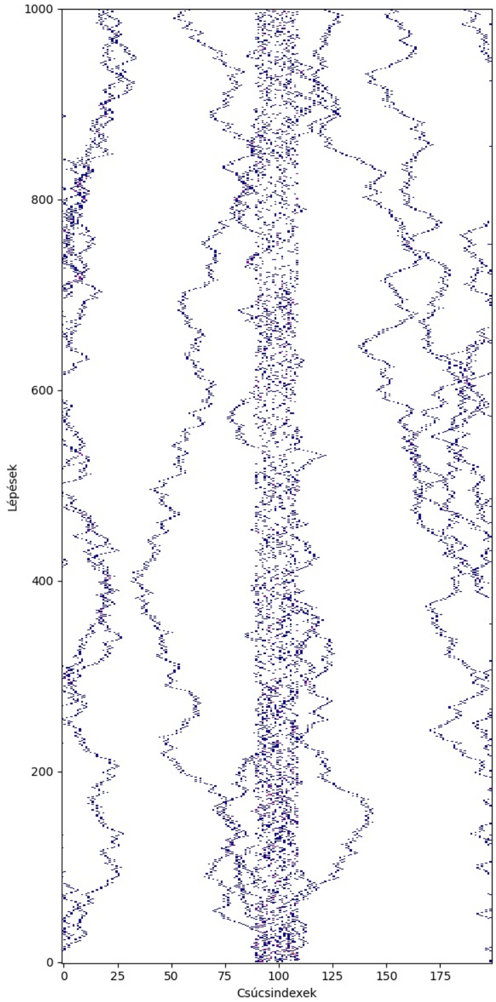
6.1. ábra. Súlyzó gráf részgráfjai



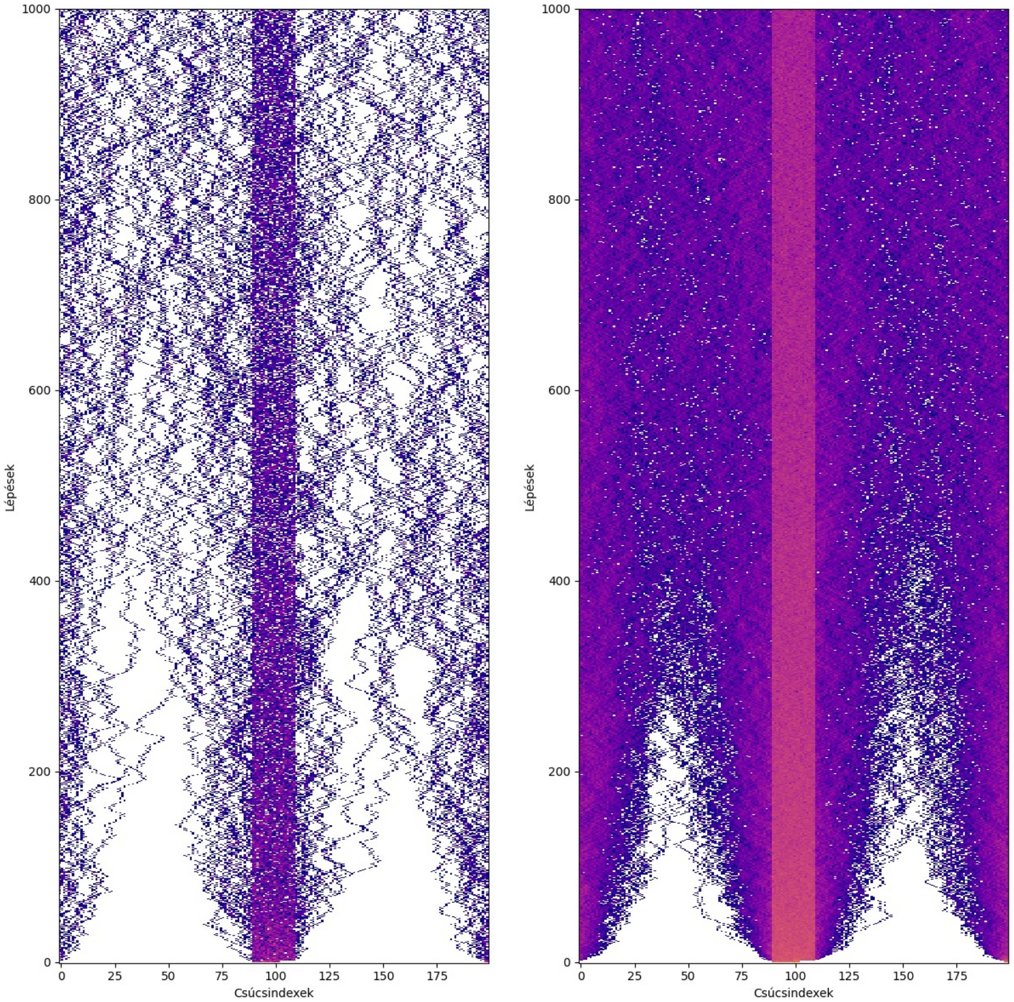
6.2. ábra. Súlyzó gráf



(a) 1 bolyongó

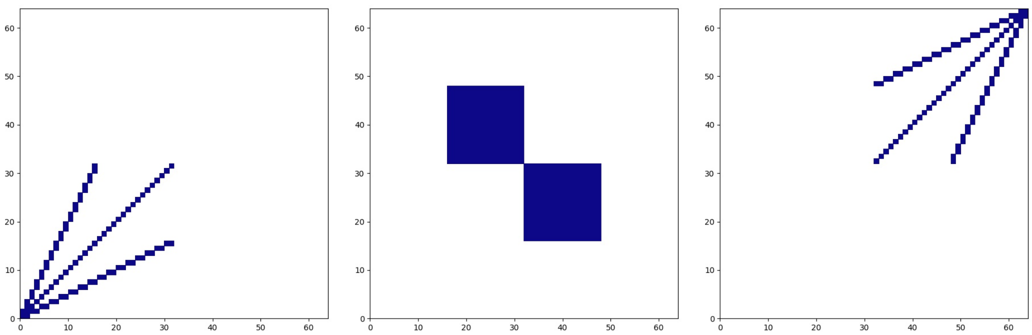


(b) 10 bolyongó



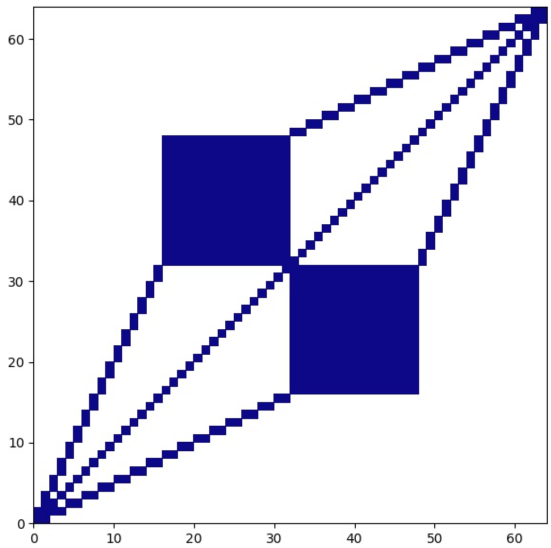
(a) 100 bolyongó **(b)** 1000 bolyongó

6.2. Ragasztott bináris gráf

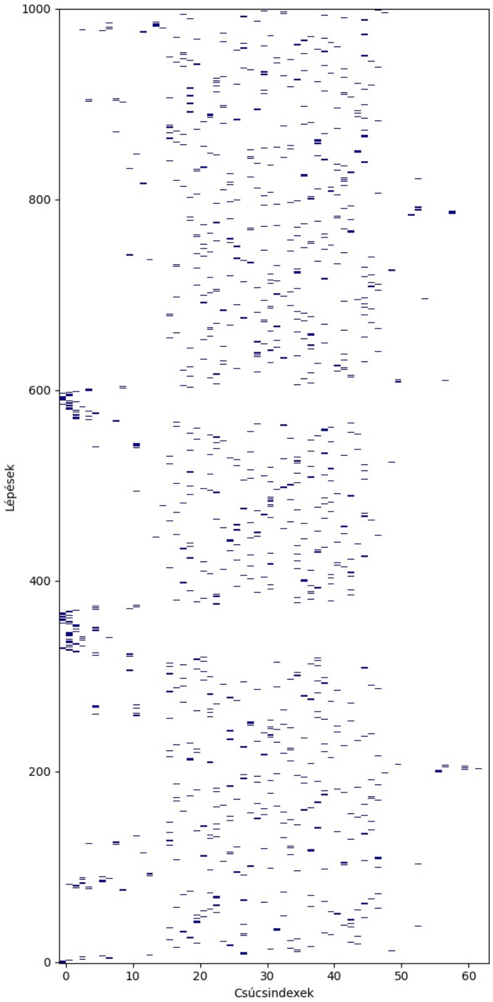


(a) Bal bináris fa **(b)** Középső teljes páros gráf **(c)** Jobb bináris fa

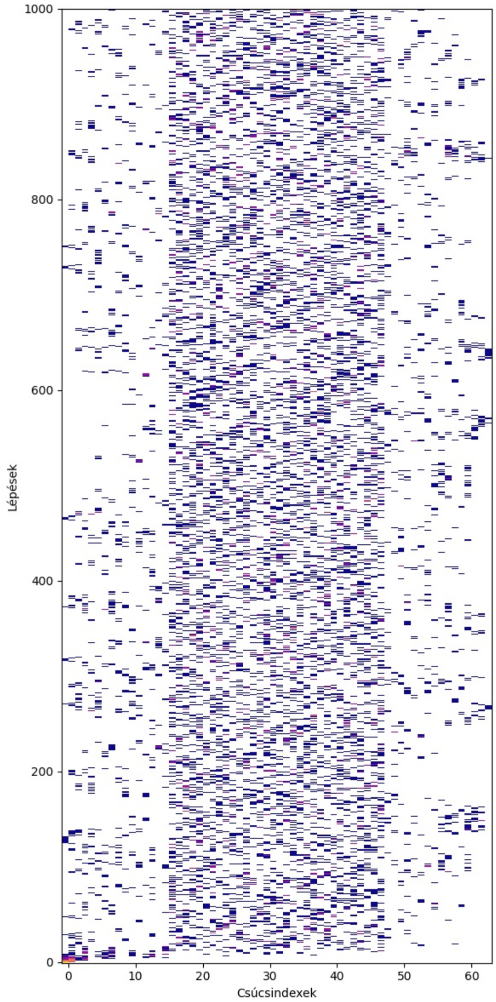
6.5. ábra. Ragasztott bináris gráf részgráfjai



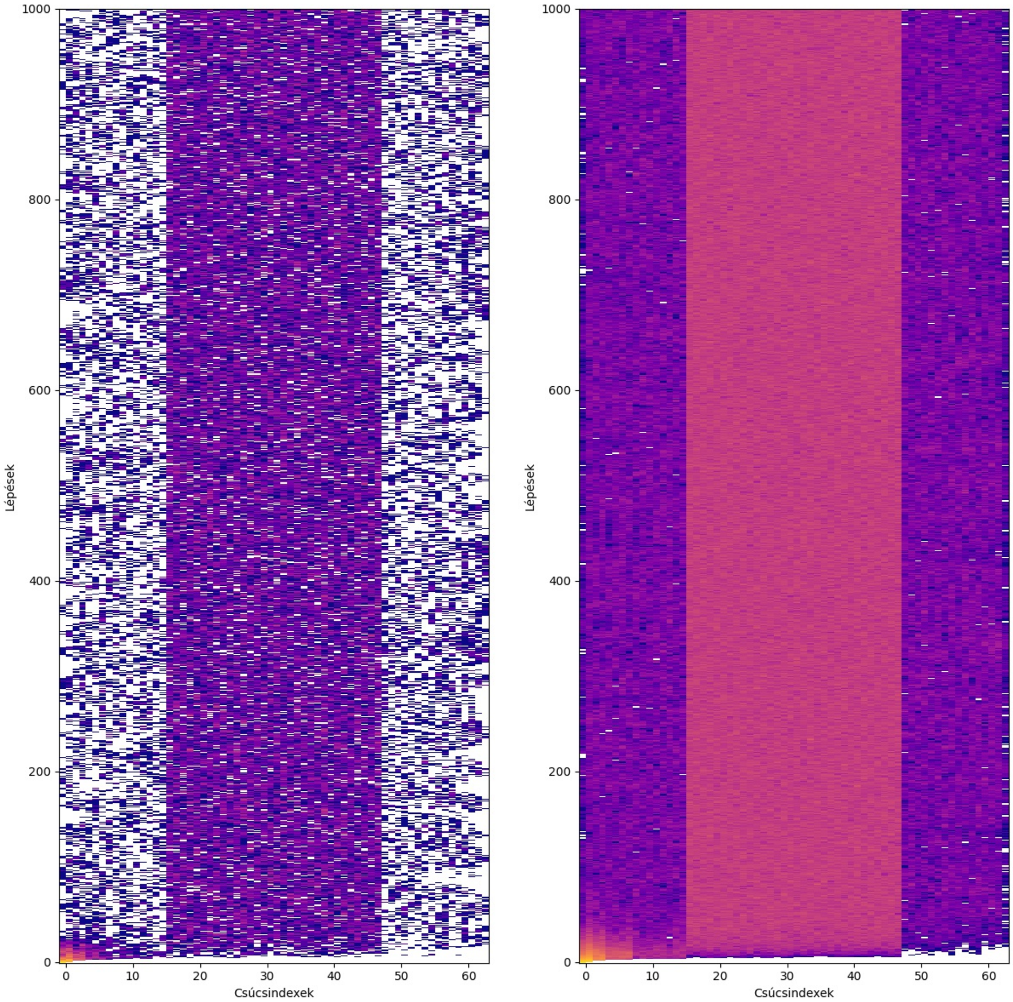
6.6. ábra. Ragasztott bináris gráf



(a) 1 bolyongó



(b) 10 bolyongó



(a) 100 bolyongó

(b) 1000 bolyongó

7. fejezet

Jövőbeli tervek

- Elolvasni a Springer sorozatos kvantum bolyongós könyveket.
- Implementálni k-reguláris gráfra a bolyongást.
- Implementálni általános gráfra a bolyongást.
- Számolni hitting time-ot és mixing rate-t.
- Számolni a szomszédossági mátrixból sajátértékeket és azokat elemezni.
- Számolni határeloszlást a Markov-láncokhoz.

Források (rendesen hivatkozva majd...):

- Hirvensalo könyv