MULTIGRÁFOK FOKSOROZATAI

IVÁNYI ANTAL (2012. január 5b.)

Havel 1955-ben [23], Erdős és Gallai 1960-ban [16], Hakimi 1962-ben [21], Ruskey, Cohen, Eades és Scott 1994-ben [59], Barnes és Savage 1997-ben [4], Tripathi, Venugopalan és West 2010-ben [73] javasoltak módszert annak eldöntésére, hogy nemnegatív egészek sorozata lehet-e egy egyszerű gráf foksorozata. 1974-ben Chungphaisan [14] kiterjesztette a csúcspárok között legfeljebb $b \geq 1$ élet tartalmazó multigráfokra mind a Havel-Hakimi, mind pedig at Erdős-Gallai tételt. Ezeknek az ismert algoritmusoknak a legrosszabb futási ideje legalább négyzetes. Cikkünkben bemutatjuk a Chungphaisan-Erdős-Gallai algoritmus lineáris változatát és annak gyorsított változatait. A Havel-Hakimi algoritmusnak olyan változatát mutatjuk be, amely $1 \leq b \leq 2$ esetén lineáris.

1. Bevezetés

A gyakorlatban különböző területeken szükség van objektumok rangsorolására. Ennek egyik elterjedt módszere, hogy az objektumokat páronként összehasonlítjuk, és az összehasonlítás eredményeképpen pontokat adunk az objektumoknak, végül pedig az objektumokat a kapott pontszámok alapján rangsoroljuk. Például Landau biológiai [40], Hakimi kémiai [21], Kim, Toroczkai, Miklós, Erdős és Székely [33] és Newman és Barabási [50] hálózati, Bozóki, Fülöp, Poesz és Rónyai gazdasági [7, 8], Liljeros et al. emberi kapcsolatokra vonatkozó [41], Iványi et al. pedig sportbeli [25, 26, 29, 30, 31, 55, 57] alkalmazásokra hivatkoztak.

Legyenek a, b és n egészek, $n \geq 1$ és $b \geq a \geq 0$. Az (a,b,n)*-gráfok* olyan hurokmentes gráfok, melyek csúcshalmaza $V = \{v_1, \ldots, v_n\}$ és a a különböző v_i és v_j csúcsok legalább a és legfeljebb b éllel vannak összekötve. Eszerint az *egyszerű irányítatlan gráfok* $(0,1,n)$-gráfok, míg az *egyszerű irányított gráfok* (tournamentek) $(1,1,n)$-gráfok.

Irányított gráfok esetén ha v_i és v_j összehasonlításakor v_i kap egy pontot, akkor annak a gráfban v_i-ből v_j-be menő irányított él felel meg. Irányítatlan gráfok esetén viszont csúcspárok kapják a pontot és annak a két csúcsot összekötő irányítatlan él felel meg.

Ebben a cikkben csak irányítatlan gráfokkal foglalkozunk és első sorban azt vizsgáljuk, hogy nemnegatív egész számok $s = (s_1, \ldots, s_n)$ nemnövekvő sorozata és adott a alsó korlát, valamint b felső korlát esetén létezik-e olyan (a,b,n)-gráf, amelynek foksorozata s. Emellett foglalkozunk a foksorozatok számával, amelyet $G(a,b,n)$-nel jelölünk.

A hasonló feladatokkal kapcsolatban megjegyezzük, hogy mind az irányítatlan, mind pedig az irányított gráfokkal kapcsolatban az utóbbi néhány évben is számos cikk és könyvfejezet jelent meg (például [6, 9, 15, 24, 31, 44, 58, 71, 73, 74, 75], illetve [5, 8, 11, 17, 19, 25, 26, 30, 33, 36, 39, 46, 53, 55, 56]).

Legyenek l, m és u egész számok, továbbá $1 \leq m$ és $l \leq u$. Egész számok $s = (s_1, \ldots, s_m)$ sorozatát (l, u, m)-*korlátosnak* (röviden: korlátosnak) nevezzük, ha $l \leq s_i \leq u$ minden $1 \leq i \leq m$ indexre. Az $s = (s_1, \ldots, s_m)$ (l, u, m)-korlátos sorozatot (l, u, m)-*szabályosnak* mondjuk, ha $u \geq s_1 \geq \cdots \geq s_m \geq l$.

A vizsgálatok során kitüntetett szerepet játszanak az $(a(n-1), b(n-1), n)$-szabályos sorozatok. Ezeket a sorozatokat (a, b, n)-*grafikusnak* (vagy röviden grafikusnak) nevezzük, ha létezik olyan (a, b, n)-gráf, melynek kifoksorozata s.

Jelentős számú cikk (például [10, 18, 37, 45]) foglalkozik páros számok *grafikus felbontásaival*: előállítják a $2k$ páros szám pozitív egész összeadandókra való monoton csökkenő felbontásait, és az így kapott $q = (q_1, \ldots, q_m)$ sorozatok közül – amelyekre $q_1 + \cdots + q_m = 2k$ és $q_m \geq q_{m-1} \geq \cdots \geq q_1$ – szűrik ki a $(0, 2k-1, 2k)$-grafikus sorozatokat, vagy pedig rekurzióval eleve csak a grafikus sorozatokat állítják elő.

A továbbiakban főleg szabályos sorozatokkal foglalkozunk. A definíciókban az alsó és felső korlátok azért szerepelnek, hogy ellenőrző algoritmusainkat megkíméljük a nyilvánvalóan nem grafikus sorozatok ellenőrzésétől, ezért ezek a megszorítások nem jelentik az általánosság korlátozását.

A cikkben csak *teljes* gráfokkal foglalkozunk. Ezekre az jellemző, hogy ha $a \leq c \leq b$, akkor bármely két csúcs között c él is meg van engedve, és az irányított esetben azok tetszőlegesen irányíthatók. A *hiányos* gráfoknál bizonyos lehetőségek tiltva vannak. Például a labdarúgásnak [19, 27, 29, 38] olyan irányított $(2, 3, n)$-gráfok felelnek meg, amelyekben a csúcsokat 2 vagy 3 él köti össze, azonban 2 él esetén azok ellentétesen, míg 3 él esetén azok azonosan vannak irányítva.

Míg teljes gráfok esetén a sorozatok tesztelése az operációkutatás folyamos módszereivel kényelmesen megoldható (bár gyakran vannak gyorsabb algoritmusok is), hiányos gráfok esetén ezek a módszerek nem alkalmazhatók.

Cikkünk fő célkitűzése, hogy minél kisebb várható futási idejű algoritmusokat találjunk annak eldöntésére, hogy előírt s szabályos sorozat grafikus-e. Eközben a minden sorozatot helyesen minősítő *pontos*, és a csak a szabályos sorozatok egy részét minősítő *közelítő* algoritmusokkal is foglalkozunk.

Érdemes megemlíteni, hogy a fokszámsorozatok számának meghatározásával kapcsolatos nehézségek miatt annak is jelentős irodalma (lásd például [15, 31, 46]) van, hogy véletlen mintavétellel becsüljük ezeket a számokat.

Melléktermékként javítottuk és bővítettük az OEIS [64] adatbázist: egyrészt a benne lévő ismert sorozatok új elemeivel, másrészt a cikkben definiált új sorozatok elemeinek megadásával. Módszerünk az összes grafikus sorozat gazdaságos előállítására is alkalmas (lásd Ruskey [59] 1994-es, valamint Barnes és Savage [4] 1997-es cikkét).

A cikk felépítése a következő. A bevezető első rész után a $(0, 1, n)$ témakör klasszikus algoritmusait foglaljuk össze. A harmadik részben új pontos algoritmusokat, a negyedik részben leszámlálási eredményeket, az ötödik részben pedig új tesztelő algoritmusokat ismertetünk. A hatodik részben a közelítő algoritmusok hibáját, míg a hetedikben a pontos algoritmusok futási idejét elemezzük. A nyolcadik rész témája a $(0, b)$-gráfok potenciális foksorozatainak tesztelése, míg a 9. részben a (a, b, n)-gráfoké a főszerep. A tizedik részben a grafikus sorozatok párhuzamos leszámlálása a téma, a tizenegyedik részben nyitott problémákat mutatunk be, míg végül a tizenkettedik részben összefoglaljuk a cikk mondanivalóját.

2. Klasszikus pontos algoritmusok $(0, 1, n)$-gráfokhoz

Ebben a részben a $(0, 1, n)$-gráfok potenciális foksorozatainak tesztelésére ismert két klasszikus algoritmust ismertetjük.

2.1. Havel-Hakimi algoritmus (HH)

A feladat megoldására az első módszert Vaclav Havel cseh matematikus javasolta 1955-ben [23, 42].

1. TÉTEL. (Havel [23]) *Ha* $n \geq 3$*, az* $(s_1, \ldots, s_n)$ $(0, 1, n)$*-szabályos sorozat akkor és csak akkor* $(0, 1, n)$*-grafikus, ha az* $(s_2 - 1, s_3 - 1, \ldots, s_{s_1} - 1, s_{s_1+1} - 1, s_{s_1+2}, \ldots, s_n)$ *sorozat* $(0, 1, n - 1)$*-grafikus.*

Bizonyítás. Lásd [23]. □

Bemenet. n: csúcsok száma $(n \geq 1)$;
$s = (s_1, \ldots, s_n)$: $(0, 1, n)$-szabályos sorozat.
Kimenet. L: s grafikusságát jelző logikai változó.
Munkaváltozók. i, j: ciklus változók;
$r = (r_1, \ldots, r_n)$: r_i az s_i elemhez tartozó tartalék;
$w = (w_1, \ldots, w_n)$: w_i az i indexhez tartozó súlypont.

HAVEL-HAKIMI-TEST(n, s, L)

01 **if** $s_1 == 0$ // 01–03. sor: nullákból álló sorozat elfogadása
02 $\quad L = 1$
03 $\quad$ **return** L
04 **if** $s_{s_1+1} == 0$ // 04–06. sor: s_1 tesztelése konstans idő alatt
05 $\quad L = 0$
06 $\quad$ **return** L
07 $w_1 = n$ // 07-12. sor: az első súlypont és tartalék számítása
08 $j = n$
09 **while** $s_j < 1$ és $j > 0$
10 $\qquad w_1 = w_1 - 1$
11 $\qquad j = j - 1$
12 $r_1 = w_1 - 1 + s_1$

```
13 for i = 2 to n − 1                                // 13–19. sor: s tesztelése
14     j = w_{i−1}                        // 14–17. sor: új súlypont kiszámítása
15     while s_j < i and j > 0
16         w_i = w_i − 1
17         j = j − 1
18         if s_i > w_i + r_{i−1}                        // 18. sor: s grafikus?
19             L = 0                                // 19–20. sor: s nem grafikus
20             return L
21         r_i = w_i + r_{i−1} − b_i
22 L = 1                                            Lines 22–23: b is graphical
23 return L
```

A Havel-Hakimi algoritmus alapja a következő tétel.

2. Tétel. *Ha $n \geq 1$, az $(s_1, \ldots, s_n)$ (0,1,n)-szabályos sorozat akkor és csak akkor grafikus, ha*

$$s_1 < w_1 \tag{1}$$

és

$$s_i \leq w_i + r_{i-1} \quad (i = 2, \ldots, n-1), \tag{2}$$

ahol

$$w_i = \max(k \geq 0 \mid s_k \geq i) \quad (i = 1, \ldots, n), \tag{3}$$

és

$$r_i = w_i + r_{i-1} - s_i \quad (i = 1, \ldots, n). \tag{4}$$

Ha ezen tétel alapján írunk egy rekurzív algoritmust, akkor annak futási ideje a RAM számítási modell [12] szerint legjobb esetben – például az $(n-1, 0, \ldots, 0) = (n-1, 0^{n-1})$ bemenetre – $\Theta(1)$, legrosszabb esetben pedig – például a *homogén* $(n-1)^n$ vagy a *tranzitív* $(n-1, n-2, \ldots, 1, 0)$ bemenetre – $\Theta(n^2)$. Érdemes megjegyezni, hogy a tétel bizonyítása konstruktív, és a bizonyításon alapuló algoritmus négyzetes idő alatt nem csak ellenőriz, hanem egy megfelelő gráfot is előállít (feltéve persze, hogy létezik megfelelő egyszerű gráf).

Mivel a teljes gráfnak $\binom{n}{2} = \Theta(n^2)$ éle van, a megvalósítást is biztosító RAM algoritmusok legrosszabb futási ideje nem lehet kisebb, mint négyzetes.

1962-ben Louis Hakimi [21] Haveltől függetlenül publikálta ugyanezt az eredményt, ezért ma a tételt rendszerint *Havel-Hakimi tételnek*, a módszert pedig *Havel-Hakimi algoritmusnak* nevezik.

Az algoritmust később irányított gráfokra [17, 25, 26, 34] is kiterjesztették.

2.2. Erdős-Gallai algoritmus (EG)

Időrendben a következő eredmény Erdős Pál és Gallai Tibor alábbi szükséges és elégséges feltétele [16] volt.

Nemnegatív egészek adott $s = (s_1, \dots, s_n)$ sorozata esetén a sorozat első i elemét a sorozat s_i eleméhez tartozó *fejnek,* míg a többi elemét az s_i elemhez tartozó *faroknak* nevezzük. A fejelemek összegét H_i, míg a farokelemek összegét T_i jelöli $(i = 1, \dots, n)$. A $\sum_{k=i+1}^{n} \min(i, s_k)$ összeget pedig C_i-vel jelöljük és a farok *kapacitásának* nevezzük. Ha egy s sorozatra H_n páros, akkor a sorozatot *párosnak,* egyébként *páratlannak* nevezzük.

3. TÉTEL. (Erdős, Gallai, [16]) *Ha* $n \geq 1$, *a* $(0, 1, n)$*-szabályos* $(s_1, \dots, s_n)$ *sorozat akkor és csak akkor* $(0, 1, n)$*-grafikus, ha*

$$H_n \quad \textit{páros} \tag{5}$$

és

$$H_i \leq i(i-1) + C_i \quad (i = 1, \dots, n-1). \tag{6}$$

Bizonyítás. Lásd [13, 16, 60, 73]. □

A 3. tételen alapul a következő ERDŐS-GALLAI algoritmus.

Bemenet. n: az s sorozat hossza;
$s = (s_1, \dots, s_n)$: $(0, 1, n)$-szabályos sorozat.

Kimenet. L : logikai változó (ha s $(0, 1, n)$-grafikus, akkor $L = 1$, egyébként $L = 0$.

Munkaváltozó. t: az aktuális C_i felső korlátja.

ERDŐS-GALLAI(n, s, L)

01 $H_1 = s_1$ // 01–03. sor: H elemeinek kiszámítása
02 **for** $i = 2$ **to** n
03 $\quad$ $H_i = H_{i-1} + s_i$
04 **if** H_n páratlan // 04–06. sor: paritás ellenőrzése
05 $\quad$ $L = 0$ // 05–06. sor: nemgrafikus sorozat elutasítása
06 $\quad$ **return** L
07 **for** $i = 1$ **to** $n - 1$ // 07–15. sor: s tesztelése
08 $\quad$ $t = 0$ // 08. sor: t kezdeti értékének beállítása
09 $\quad$ **for** $k = i + 1$ **to** n // 09–10. sor:
10 $\quad\quad$ $t = t + \min(i, b_k)$
11 $\quad$ **if** $H_i - i(i-1) > t$ // line 11: check the necessary condition
12 $\quad\quad$ $L = 0$ // lines 12–13: the input is nongraphical
13 $\quad\quad$ **return** L
14 $L = 1$ // lines 14–15: the input is graphical
15 **return** L

EG memóriaigénye $\Theta(n)$, futási ideje a legjobb $\Theta(n)$ és a legrosszabb $\Theta(n^2)$ között változik..

Bár ez a tétel csak ellenőriz, futási ideje a legjobb $\Theta(n)$ és a legrosszabb $\Theta(n^2)$ között változik. A közelmúltban Tripathi et al. [73] publikáltak a tételre konstruktív bizonyítást, amely grafikus bemenet esetén $O(n^3)$ idő alatt egy megoldást is előállít.

A szabályos sorozatoknak aszimptotikusan a fele páros sorozat. Az 1. táblázat a $(0,1,n)$-szabályos sorozatok számát megadó 15, valamint a $(0,1,n)$-páros sorozatok számát megadó 17 lemmák alapján mutatja a konvergencia gyorsaságát (ami a képletek alapján természetes) $(E(n)/R(n))$ tartalmazza $n = 1,\ldots,38$ csúcs esetén.

3. Új pontos algoritmusok $(0,1,n)$-gráfokhoz

Ebben a részben a klasszikus algoritmusok néhány gyorsított változatát mutatjuk be.

3.1. Nullamentes algoritmusok

Mivel a sorozatok végén lévő nullák izolált csúcsokat jelentenek, így azok nem befolyásolják, hogy az adott sorozat grafikus-e. Ezt a megfigyelést hasznosítja a következő állítás, amelyben p a b sorozat pozitív elemeinek a számát jelöli.

4. Következmény. *Ha $n \geq 1$, az $(s_1,\ldots,s_n)$ $(0,1,n)$-szabályos sorozat akkor és csak akkor $(0,1,n)$-grafikus, ha $s_n = 0$ vagy az $(s_1,\ldots,s_p)$ sorozat $(0,p,p)$-grafikus.*

Bizonyítás. Ha a sorozatnak van pozitív eleme, akkor az állítás a Havel-Hakimi, illetve az Erdős-Gallai következménye, de közvetlenül is adódik: a nullák ugyanis nem segítenek a pozitív fokszámok párosításánál, ugyanakkor nem okoznak önálló igényt sem. □

Az ezen a tulajdonságon alapuló megvalósítást nullamentes Havel-Hakimi (HHN), illetve nullamentes Erdős-Gallai (EGN) algoritmusnak nevezzük.

3.2. Eltoló Havel-Hakimi algoritmus

Havel és Hakimi eredeti tételének természetes algoritmikus megfelelőjét HHR-nek (Rendező Havel-Hakimi) nevezzük, mert a tétel természetes alkalmazása minden menetben igényli a redukált bemenet rendezését.

A tétel alapján olyan megvalósítás is lehetséges, hogy a fokszámok redukálását a sorozat monotonitását megőrizve végezzük. Ekkor az Eltoló Havel-Hakimi algoritmust (HHSh) kapjuk.

3.3. Paritásos Havel-Hakimi algoritmus

Érdekes gondolat az Erdős-Gallai és a Havel-Hakimi feltételek együttes alkalmazása úgy, hogy először s paritását vizsgáljuk, és csak a páros bemenetekre alkalmazzuk

n	$R(n)$	$E(n)$	$E(n)/R(n)$
1	1	1	1.0000000000000
2	3	2	0.6666666666667
3	10	6	0.6000000000000
4	35	19	0.5428571428571
5	126	66	0.5238095238095
6	462	236	0.5108225108225
7	1716	868	0.5058275058275
8	6435	3235	0.5027195027195
9	24310	12190	0.5014397367339
10	92378	46252	0.5006819805581
11	352716	176484	0.5003572279114
12	1352078	676270	0.5001708481315
13	5200300	2600612	0.5000888410284
14	20058300	10030008	0.5000427753100
15	77558760	38781096	0.5000221251603
16	300540195	150273315	0.5000107057227
17	1166803110	583407990	0.5000055150693
18	4537567650	2268795980	0.5000026787479
19	17672631900	8836340260	0.5000013755733
20	68923264410	34461678394	0.5000006701511
21	269128937220	134564560988	0.5000003432481
22	1052049481860	526024917288	0.5000001676328
23	4116715363800	2058358034616	0.5000000856790
24	16123801841550	8061901596814	0.5000000419280
25	63205303218876	31602652961516	0.5000000213918
26	247959266474052	123979635837176	0.5000000104862
27	973469712824056	486734861612328	0.5000000053420
28	3824345300380220	1912172660219260	0.5000000026224
29	15033633249770520	7516816644943560	0.5000000013342
30	59132290782430712	29566145429994736	0.5000000006558
31	232714176627630544	116357088391374032	0.5000000003333
32	916312070471295267	458156035385917731	0.5000000001640
33	3609714217008132870	1804857108804606630	0.5000000000833
34	14226520737620288370	7113260369393545740	0.5000000000410
35	56093138908331422716	28046569455332514468	0.5000000000208
36	221256270138418389602	110628135071477978626	0.5000000000103
37	873065282167813104916	436532641088444120108	0.5000000000052
38	3446310324346630677300	1723155162182151654600	0.5000000000026

1. táblázat. A szabályos ($R(n)$) és a páros ($E(n)$) sorozatok száma, valamint ezen számok hányadosa ($E(n)/R(n)$)).

a rendszerint négyzetes futási idejű rekurzív ellenőrzést. Ezzel ugyan elveszítjük a nullamentes Havel-Hakimi azon jó tulajdonságát, hogy legjobb esetben konstans idő alatt lefut, viszont cserébe megkapjuk azt, hogy a várható futási idő jelentősen csökken.

3.4. Lineáris Havel-Hakimi tesztelő algoritmus (HHl)

Az EGl algoritmusban kulcsszerepe van az s_i elemhez tartozó w_i súlypontnak [29], amely $i > s_i$ esetén 0, egyébként a legnagyobb olyan k index, amelyre igaz, hogy $s_k \geq bi$ (ez az egyenlőtlenség a $(0, 1, n)$-gráfok esetén az $s_k \geq i$ egyenlőtlenségre egyszerűsödik.)

Ez a súlypont arra is alkalmas, hogy a Havel-Hakimi algoritmus alábbi változatában fontos szereplő legyen. A pszeudokód a [12] tankönyvben ismertetett konvenciókat alkalmazza.

Bemenet. n: csúcsok száma ($n \geq 1$);
$s = (s_1, \ldots, s_n)$: a tesztelendő $(0, 1, n)$-szabályos sorozat.

Kimenet. L: s grafikusságát jelző logikai változó.

Munkaváltozók. i: ciklus változó;
$r = (r_1, \ldots, r_n)$: r_i az s_i-hez tartozó maradék farokkapacitás ($r_1 = w_1 - 1$ és $r_i = w_i - 1 + r_{i-1} - s_i$, ha $2 \leq i \leq n$);
$w = (w_1, \ldots, w_n)$: w_i az i indexhez tartozó súlypont (ha $i > s_1$, akkor $w_i = 0$, egyébként w_i a legnagyobb olyan k, amelyre teljesül $s_k \geq i$;
$H = (H_1, \ldots, H_n)$: H_i az s sorozat első i elemének összege.

Havel-Hakimi-lineáris(n, s, L)
01 $L = 0$ // 01. sor: a gyakoribb érték beállítása
02 **if** $s_1 == 0$ // 02–04. sor: a nullákból álló sorozat grafikus
03 $\quad L = 1$
04 $\quad$ **return** L
05 **if** $s_{s_1+1} == 0$ // 05–07. sor: s_1 ellenőrzése konstans idő alatt
06 $\quad$ **return** L
07 $H_1 = s_1$ // 07. sor: H_1 egyenlő a sorozat első elemével
08 **for** $i = 2$ **to** n // 08–08. sor: további H_i értékek kiszámítása
09 $\quad H_i = H_{i-1} + s_i$
10 **if** H_n páratlan // 10–11. sor: paritás ellenőrzés
11 $\quad$ **return** L
12 $w_1 = n$ // 12–15. sor: első súly és tartalék számítása
13 **while** $s_{w_1} < 1$
14 $\qquad w_1 = w_1 - 1$
15 $r_1 = w_1 - 1 - s_1$
16 $s_{n+1} = 0$
17 **for** $i = 2$ **to** $n - 1$
18 $\quad$ **if** $s_i \leq 1$ vagy $s_{i+1} = 0$
19 $\qquad L = 1$
20 $\qquad$ **return** L
21 $\quad w_i = w_{i-1}$
22 $\quad$ **while** $s_{w_i} < i$ és $w_i > 0$
23 $\qquad w_i = w_i - 1$

24   **if** $w_i \geq i$ 24. sor: esetszétválasztás
25   **if** $s_i > w_i - 1 + r_{i-1}$ // 25. sor: s grafikus?
26   **return** L // 26. sor: s nem grafikus
27       $r_i = w_i - 1 + r_{i-1} - s_i$
28   **else if** $s_i > w_i + r_{i-1}$ // 28. sor: s grafikus?
29   **return** L // 29. sor: s nem grafikus
30         $r_i = w_i + r_{i-1} - s_i$
31 $L = 1$
32 **return**

5. TÉTEL. *A* HAVEL-HAKIMI-LINEÁRIS-TESZT *futási ideje legjobb esetben* $\Theta(1)$*, legrosszabb esetben* $\Theta(n)$.

Bizonyítás. A 01–07. sorok időigénye $O(1)$, és például a (0^n) bemenetre a program a 04. sorban megáll, ezért a legjobb futási idő $O(1)$. A 08–11. sorok időigénye $\Theta(n)$. Mivel a súlypontok számítása legfeljebb n csökkentést igényel, a 14–30. sorok igénye $O(n)$, ezért a legrosszabb eset $\Theta(n)$. □

3.5. Példák

1. példa. Legyen az első példánk $s = (3^3, 1)$. A 01–15. sorok szerint $r_1 = 0$. Ha $i = 2$, akkor $w_i = 3$ és a 25. sor feltétele nem teljesül, ezért s *nem* $(0,1,4)$*-grafikus.* □

2. példa A következő példa $s = (5, 3^2, 2, 1^3)$. Az 01–15. sorokban azt kapjuk, hogy $w_1 = 7$ és $r_1 = 1$. Ha $i = 2$, akkor $w_i = 4$, a 25. sor feltétele nem teljesül és a 27. sor szerint $r_2 = 1$. Ha $i = 3$, akkor $w_i = 3$ és nem teljesül a 25. sor feltétele. Ha $i = 4$, akkor $w_i = 1$ és most sem teljesül 25. sor feltétele. Ha $i = 5$, akkor teljesül a 16. sor $s_i \leq 1$ feltétele, és ezért s $(0,1,7)$*-grafikus.* □

3. példa Legyen $s = (5, 4, 1^5)$. Erre a sorozatra $r_1 = 1$, és ha $i = 2$, akkor $w_i = 2$, ezért a 25. sor feltétele teljesül, így s *nem* $(0,1,7)$*-grafikus.* □

4. példa Utolsó példánk legyen $s = (5^2, 4, 3^4)$. Az első 15 sor szerint $r_1 = 1$. Ha $i = 2$, akkor $w_i = 7$ és $r_2 = 1$. Ha $i = 3$, akkor $w_3 = 7$ és $r_3 = 2$. Ha $i = 4$, akkor teljesül a 18. sor $s_i \leq 1$ feltétele, ezért s $(0,1,7)$*-grafikus.* □

3.6. HHLT első hat sorának elemzése

Nem nehéz belátni, hogy a (l, u, m)-korlátos sorozatok $B(l, u, m)$ száma

$$B(l, u, m) = \binom{u - l + m}{m}. \tag{7}$$

Ezért a HHl algoritmus lehetséges bemeneteinek a száma

$$B(0, n-1, n) = \binom{2n-1}{n}. \tag{8}$$

HHl első három sora kiszűri például azokat a sorozatokat, amelyek $(n-1)$-gyel kezdődnek és nullával végződnek. Ezek száma (7) szerint

$$B(0, n-1, n-2) = \binom{2n-3}{n-2}. \tag{9}$$

Ezek közül a HHl által kiszűrt sorozatok $R_1(n)$ hányada

$$R_1(n) = \frac{\binom{2n-3}{n-2}}{\binom{2n-1}{n}} = \frac{2(2n-1)}{n} = \frac{1}{4} + \frac{1}{8n-4}. \tag{10}$$

HHl pontosan azokat a sorozatokat szűri ki, amelyek $(n-i)$-vel $(i = 1, \ldots, n-2)$ kezdődnek és legalább i nullát tartalmaznak. Rögzített i-re az ilyen sorozatok aszimptotikus részaránya $1/4^i$, így HHl aszimptotikusan a szabályos sorozatoknak a

$$\sum_{i=1}^{\infty} \frac{1}{4^i} = \frac{1}{3} \tag{11}$$

összegnek megfelelő hányadát, azaz egy harmad részét szűri ki.

Mivel a grafikus sorozatok aszimptotikus sűrűsége nulla, ezért minden A pontos algoritmusra létezik egy $s_{1,A} + s_{2,A} + \ldots = 1$ sor (valószínűség-eloszlás), amelyben s_i az i-edik menetben kiszűrt hányad. Például $s_{1,A} = 1/3$ minden olyan pontos algoritmusra, amelyik első menetben a PT algoritmust (vagy annak valamilyen lassú változatát) használja az első menetben – ilyen a HH és az EG is.

3.7. Tripathi és Vijay első javítása

Tripathi és Vijai 2003-ban megmutatta, hogy az Erdős-Gallai tétel következő változata is helyes.

6. LEMMA. (Tripathi, Vijay [72]) *Ha $n \geq 1$, akkor a $(0, 1, n)$-szabályos s sorozat akkor és csak akkor $(1, 1, n)$-grafikus, ha*

$$H_n \quad \textit{páros} \tag{12}$$

és

$$H_i - i(i-1) \leq \sum_{k=i+1}^{n} \min(i, s_k) \quad (i = 1, \ldots, q), \tag{13}$$

ahol

$$q = \max_{1 \leq k \leq n} (k \mid k(k-1) < H_k). \tag{14}$$

Bizonyítás. Lásd [72]. □

Ennél is hasznosabb a következő állítás,amelyben (13) helyett a szigorúbb (16) szerepel, amelyet q helyett csak $(q-1)$-ig ellenőrzünk.

7. LEMMA. *Ha $n \geq 1$, a $(0,1,n)$-szabályos $s = (s_1, \ldots, s_n)$ sorozat akkor és csak akkor grafikus, ha*

$$H_n \quad \textit{páros} \tag{15}$$

és

$$H_i - \min(H_i, i(i-1)) \leq \sum_{k=i+1}^{n} \min(i, s_k) \quad (i = 1, 2, \ldots, q-1), \tag{16}$$

ahol

$$q = \max_{1 \leq k \leq n} (k \mid k(k-1) < H_k). \tag{17}$$

Bizonyítás. Ha ???????????????????, □

3.8. Tripathi és Vijay második javítása

Tripathi és Vijai a [72] cikkben az Erdős-Gallai tétel következő, lényeges gyorsítást lehetővé tevő változatát is bizonyították.

Az ismétlődő elemeket gyakoriságuk segítségével tömörítve a $(0,1,n)$-szabályos $(s_1, \ldots, s_n)$ sorozat felírható az $(s_{i_1}^{e_1}, \ldots, s_{i_q}^{e_q})$ alakban, ahol $s_{i_1} < \cdots < s_{i_q}$, $e_1, \ldots, e_q \geq 1$ és $e_1 + \cdots + e_q = n$. Legyen $g_j = e_1 + \cdots + e_j$ $(j = 1, \ldots, q)$.

Az s_i elemet az s *ugró elemének* nevezzük, ha $i = n$ vagy $1 \leq i \leq n-1$ és $s_i > s_{i+1}$. Ekkor az ugró pontok az $s_{g_1}, \ldots, s_{g_q}$ elemek. Az ugró (vagy ellenőrző) elemeket $c_1 = s_{g_1}, \ldots, c_q = s_{g_q}$ módon jelöljük.

8. TÉTEL. (Tripathi, Vijay [72]) *Az $s = (s_1, \ldots, s_n)$ szabályos sorozat akkor és csak akkor grafikus, ha*

$$H_n \quad \textit{páros} \tag{18}$$

és

$$H_{s_i} - s_i(s_i - 1) \leq \sum_{k=s_i+1}^{n} \min(s_i, b_k) \quad (i = 1, \ldots, q). \tag{19}$$

Bizonyítás. Lásd [72]. □

Megjegyezzük, hogy az ellenőrzést elég a $(q-1)$-edik ugrópontig folytatni.

Ehhez az állításhoz hasonló a következő **??**. tétel.

Ha s azonos elemekből áll, akkor mindig grafikus és nincs n-nél kisebb ugrópont benne. Ezért tegyük fel, hogy $s_1 > s_n$, továbbá s ugrópontjainak sorozata $s = (s_1, \ldots, s_q)$.

9. TÉTEL. *Ha $n \geq 1$, a szabályos $(s_1, \ldots, s_n)$ sorozat akkor és csak akkor grafikus, ha*

$$H_{s_i} \leq \sum_{j=1}^{s_i} w_j - 1 \quad (i = 1, \ldots, q-1), \tag{20}$$

ahol

$$w_j = \max(k \geq 0 \mid b_k \geq j). \tag{21}$$

Bizonyítás. Ha ?? □

A következő táblázatokban bemutatjuk, hogyan oszlanak meg a kizárt grafikus és nemgrafikus sorozatok az egyes menetek között. Azt is jellemezzük, hogy átlagosan hány meneten át kell egy grafikus, illetve nemgrafikus sorozatot a kizárásáig tesztelni, és azt is, hogy a menetek hányadrészét fordítjuk átlagosan egy sorozat tesztelésére.

A 2. táblázat az (a, b, n)-szabályos és (a, b, n)-grafikus sorozatok számát tartalmazza $n = 1, \ldots, 11$ csúcs, valamint $a = 0$ és $b = 1$, $a = 0$ és $b = 2$, $a = 2$ és $b = 5$ esetén.

n	$R(0,1,n)$	$G(0,1,n)$	$R(0,2,n)$	$G(0,2,n)$	$R(2,3,n)$	$G(2,3,n)$
1	1	1	1	1	1	1
2	3	2	6	3	10	4
3	10	4	35	10	84	23
4	35	11	210	52	715	189
5	126	31	1287	283	6188	1582
6	462	102	8008	1706	54264	13583
7	1716	342	50388	10436	480700	122345
8	6435	1213	319770	65370	4292145	1092573
9	24310	4361	2042975	413111	38567100	9816598
10	92378	16016	13123110	2633537	348330136	88680716
11	352716	59348	84672315	16882153	3159461968	804480107

2. táblázat. Az (a, b, n)-szabályos és (a, b, n)-grafikus sorozatok száma $n = 1, \ldots, 11$ csúcs, valamint $a = 0$ és $b = 1$, $a = 0$ és $b = 2$, $a = 2$ és $b = 5$ esetén.

A 3. táblázat a HHl által az i-edik ($i = 1, \ldots, 11$) menetben kiszűrt nem grafikus sorozatok számát mutatja $n = 1, \ldots, 11$ csúcs, valamint $a = 0$ és $b = 1$ esetén.

n/i	1	2	3	4	5	6	7	8	9	10	11
1	0										
2	1	0									
3	6	0	0								
4	22	2	0	0							
5	85	8	2	0	0						
6	311	35	12	2	0	0					
7	1169	128	58	17	2	0	0				
8	4369	488	239	100	24	2	0	0			
9	16524	1805	942	471	173	32	2	0	0		
10	62650	6800	3601	2021	956	289	43	2	0	0	
11	239008	25571	13677	8147	4561	1877	470	55	2	0	0

3. táblázat. HHl i-edik ($i = 1, \ldots, 11$) menetében kiszűrt nem grafikus sorozatok száma $n = 1, \ldots, 11$ csúcs, valamint $a = 0$, $b = 1$ esetén.

A 4. táblázat HHl i-edik ($i = 1, \ldots, 11$) menetében kiszűrt grafikus sorozatok számát tartalmazza $a = 0$ és $b = 1$, $n = 1, \ldots, 11$ csúcs esetén.

n/i	1	2	3	4	5	6	7	8	9	10	11
1	1										
2	1	1									
3	1	3	0								
4	1	8	2	0							
5	1	16	12	2	0						
6	1	29	48	22	2	0					
7	1	47	130	127	35	2	0				
8	1	72	306	488	290	54	2	0			
9	1	104	618	1492	1475	591	78	2	0		
10	1	145	1158	3863	5757	3868	1112	110	2	0	
11	1	195	1998	8890	18440	18662	9053	1958	149	2	0

4. táblázat. HHl i-edik ($i = 1, \ldots, 11$) menetében kiszűrt nem grafikus sorozatok száma $a = 0$, $b = 1$ és $n = 1, \ldots, 11$ csúcs esetén.

Legyen $N_i(a, b, n, A)$, illetve $M_i(a, b, n, A)$ az A algoritmus által az (a, b, n)-szabályos sorozatok vizsgálata során az i-edik ($i = 1, \ldots, n$) menetben kizárt nemgrafikus, illetve grafikus sorozatok száma, továbbá legyen

$$N = \sum_{i=1}^{n} \quad \textit{és} \quad M = \sum_{i=1}^{n} m_i, \tag{22}$$

$$X(a, b, n, A) = \frac{\sum_{i=1}^{n} i n_i}{n}, \tag{23}$$

$$Y(a, b, n, A) = \frac{\sum_{i=1}^{n} i m_i}{n}, \tag{24}$$

$$Z(a, b, n, A) = \frac{\sum_{i=1}^{n} i(m_i + n_i)}{n}, \tag{25}$$

$$X'(a, b, n, A) = \frac{\sum_{i=1}^{n} i n_i}{n^2}, \tag{26}$$

$$Y'(a, b, n, A) = \frac{\sum_{i=1}^{n} i m_i}{n^2}, \tag{27}$$

$$Z'(a, b, n, A) = \frac{\sum_{i=1}^{n} i(m_i + n_i) n_i}{n^2}. \tag{28}$$

Az 5. táblázat a ChEGl algoritmus hatékonyságát jellemzi $a = 0$, $b = 2$ és $n = 1, \ldots, 11$ csúcs esetén.

3.9. Rövidített Erdős-Gallai algoritmus

H_i maximális értéke szabályos sorozat esetén $n(n-1)$, ezért a 3. tételben szereplő (6) egyenlőtlenség $i = n$ esetén biztosan teljesül, így felesleges ellenőrizni.

Ennél is hasznosabb észrevételt tartalmaz a következő lemma.

Tripathi és Vijay 2003-as cikkében [72] szerepel az az észrevétel, hogy az Erdős-Gallai tételben a (6) egyenlőtlenséget elég csak addig ellenőrizni, amíg $H_i > i(i-1)$ teljesül.

n/jellemző	X	Y	Z	X'	Y'	Z'
2	1.0000	1.5000	1.3333	0.5000	0.7500	0.6667
3	1.0000	1.7500	1.3000	0.3333	0.583	0.4333
4	1.0833	2.0909	1.4000	0.2708	0.5018	0.3500
5						
6						
7						
8						
10						
11						

5. táblázat. HHl hatékonysági jellemzői $a = 0$, $b = 1$ és $n = 2, \ldots, 11$ csúcs esetén.

10. Lemma. (Tripathi és Vijay [72]) *Ha* $n \geq 1$, *a* $(0, 1, n)$*-szabályos* $s = (s_1, \ldots, s_n)$ *sorozat akkor és csak akkor* $(0, 1, n)$*-grafikus, ha*

$$H_n \quad \textit{páros} \tag{29}$$

és

$$H_i - \min(H_i, i(i-1)) \leq \sum_{k=i+1}^{n} \min(i, b_k) \quad (i = 1, 2, \ldots, g), \tag{30}$$

ahol

$$g = \max_{1 \leq k \leq n} (k \mid k(k-1) < H_k). \tag{31}$$

Bizonyítás. Ha $i(i-1) \geq H_i$, akkor (6) bal oldala nempozitív, ezért az egyenlőtlenség biztosan teljesül, így felesleges ellenőrizni. □

Például a $c = (5^{100})$ sorozat esetén (6) jobb oldalát az Erdős-Gallai algoritmus szerint kilencvenkilencszer, míg a rövidített Erdős-Gallai algoritmus szerint csak hatszor kell kiszámítani. A javításnak a várható futási időre gyakorolt hatását a 7. részben vizsgáljuk.

A 10. lemmán alapuló algoritmust rövidített Erdős-Gallai algoritmusnak (EGR) nevezzük.

3.10. Ugró Erdős-Gallai algoritmus (EGu)

Az ismétlődő elemeket összevonva egy szabályos $(s_1, \ldots, s_n)$ sorozat $(s_{i_1}^{e_1}, \ldots, s_{i_q}^{e_q})$ alakban is felírható, ahol $s_{i_1} > \cdots > s_{i_q}$, $e_1, \ldots, e_q \geq 1$ és $e_1 + \cdots + e_q = n$. Legyen $\sigma_j = e_1 + \cdots + e_j$ $(j = 1, \ldots, q)$.

Az s_i elemet a b sorozat *ugró* elemének nevezzük, ha $i = n$ vagy $1 \leq i \leq n-1$ és $s_i > s_{i+1}$. Ekkor az ugró elemek a $s_{\sigma_1}, s_{\sigma_2}, \ldots, s_{\sigma_q}$ elemek.

11. Tétel. (Tripathi, Vijay [72]) *Az* $(1, 1, n)$*-szabályos* $s = (s_1 \ldots, s_n)$ *sorozat akkor és csak akkor* $(1, 1, n)$*-grafikus, ha*

$$H_n \quad \textit{páros} \tag{32}$$

és

$$H_{\sigma_i} - \sigma_i(\sigma_i - 1) \leq \sum_{k=\sigma_i+1}^{n} \min(i, s_k) \quad (i = 1, \ldots, q). \tag{33}$$

Bizonyítás. Lásd [72]. □

Később majd az Erdős-Gallai-Lineárisan algoritmusban kihasználjuk, hogy a (33) egyenlőtlenségben σ_q mindig n, és ezért $H_n - n(n-1)$ mindig teljesül, így elég az egyenlőtlenséget $(i = q - 1)$-ig ellenőrizni.

A következő program az Erdős-Gallai algoritmusnak az 4. és 10. lemma, valamint a 11. tétel alapján gyorsított változatát mutatja be.

Bemenet. n: a csúcsok száma $(n \geq 1)$;
$s = (s_1, \ldots, s_n)$: n-szabályos sorozat.

Kimenet. L: s grafikusságát jelző logikai változó.

Munkaváltozók. i és j: ciklusváltozók;
$H = (H_1, \ldots, H_n)$: H_i b első i elemének az összege;
p: b pozitív elemeinek a száma;
b_{p+1}: segédváltozó annak eldöntéséhez, hogy b_p ugró elem-e.

Erdős-Gallai-ugró(n, s, L, γ)

```
01 L = 0                                  ▷ 01–02. sor: kezdeti értékek beállítása
02 l = 0
03 p = n                                  ▷ 03–05. sor: nullamentesítés
04 while s_p = 0
05         p = p − 1
06 H_1 = s_1                              ▷ 06–10. sor: paritás ellenőrzése
07 for i = 2 to p
08       H_i = H_{i−1} + s_i
09 if H_p páratlan
10    return L
11 s_{p+1} = 0                            ▷ 11–23. sor: fej igényének ellenőrzése
12 i = 1
13 while i ≤ p és i(i − 1) < H_i
14         if s_i = s_{i+1}
15            i = i + 1
16            E = 0
17            for j = i + 1 to p
18            E = E + min(j, s_j)
19         if H_i > i(i − 1) + E
20            return L
21 L = 1
22 return L
```

Ennek az algoritmusnak a futási ideje a legjobb $\Theta(1)$ és a legrosszabb $\Theta(n^2)$ között változik.

3.11. Lineáris Erdős-Gallai algoritmus (EGl)

A következő Erdős-Gallai-Lineárisan algoritmus kihasználja, hogy az s bemeneti sorozat monoton. Ennek köszönhetően a C_i kapacitásokat minden i-re konstans időben meg tudja határozni, azaz nincs szüksége arra, hogy a megfelelő farok elemeit egyenként megvizsgálja. A gyors számolás kulcsa a súlypontokat tartalmazó $w(s)$ sorozat.

Adott s sorozat esetén legyen $w(s) = (w_0, w_1, \ldots, w_{n-1})$, ahol $i > s_1$ esetén $w_i = 0$, egyébként pedig w_i az s sorozat legnagyobb indexű olyan elemének indexe, amelyik legalább akkora, mint i.

Az s sorozat s_i elemének ellenőrzésekor két eset van: ha $i > w_i$, akkor a C_i kapacitás egyszerűen számítható: $H_n - H_i$, mivel a farok minden s_j elemének hozzájárulása csak s_j.

Ha viszont $i \leq w_i$, akkor a C_i-t definiáló szummát két részre bontjuk: az első részhez a farok azon s_j kezdő elemeinek hozzájárulása tartozik, amelyekre teljesül $s_j \geq i$, a második részhez pedig a többi elem. Legyen $q(s) = q = \max_{1 \leq i \leq n}\{i \mid i(i-1) \leq H_i\}$.

12. Tétel. *Ha $n \geq 1$, az $s = (s_1, \ldots, s_n)$ $(0,1,n)$-szabályos sorozat akkor és csak akkor $(0,1,n)$-grafikus, ha*

$$H_n \quad \textit{páros}, \tag{34}$$

továbbá

$$H_i \leq i(k-1) + H_n - H_k \quad (i = 1, \ldots, q), \tag{35}$$

ahol

$$k(s) = k = \begin{cases} w_i, & \textit{if } i \leq w_i, \\ i, & \textit{ha } i > w_i. \end{cases} \tag{36}$$

Bizonyítás. Megmutatjuk, hogy a tételben szereplő feltétel ekvivalens a 3. tétel feltételeivel.

A (**??**) feltétel pontosan megegyezik az (5) feltétellel.

Ha $i \leq w_i$, akkor

$$H_i \leq i(i-1) + (w_i - i + 1)i + H_n - H_{w_i} \tag{37}$$

és ha $i > w_i$, akkor

$$H_i \leq i(i-1) + H_n - H_i. \tag{38}$$

Ha (37) jobb oldalán kiemeljük i-t, akkor a

$$H_i \leq iw_i + H_n - H_{w_i} \tag{39}$$

egyenlőtlenséget kapjuk. Ha (**??**)-ba (36) alapján behelyettesítjük k-t, akkor az $i \leq w_i$ esetben (37)-t, az $i > w_i$ esetben pedig (38)-t kapjuk. □

A következő program a **??**. tétel alapján adott n-re tetszőleges n-szabályos sorozatról eldönti, hogy grafikus-e. A program futási ideje minden sorozatra $O(n)$. Érdemes megjegyezni, hogy akár a bemenő sorozat rendezettségétől is eltekinthetünk, mivel a sorozat elemei egész számok és mindegyik a $[0, n-1]$ intervallumba esik, így szükség esetén $O(n)$ idő alatt rendezni tudjuk a sorozatot.

Bemenet. n: a csúcsok száma ($n \geq 1$);
$s = (s_1, \ldots, s_n)$: $(0,1,n)$-szabályos sorozat.

Kimenet. L: s grafikusságát jelző logikai változó.

Munkaváltozók. i: ciklusváltozó;
$H = (H_1, \ldots, H_n)$: H_i az éppen tesztelt s első i elemének az összege;
$w = (w_1, \ldots, w_{n-1})$: w_i a legnagyobb indexű olyan s_j indexe, amely legalább i;
$H_0 = 0$: segédváltozó a H sorozat elemeinek kiszámításához;
$s_0 = n - 1$: segédváltozó az m sorozat elemeinek kiszámításához;
y: ellenőrzés egyszerűsítéséhez használt változó (az aktuális s_i vágópontja, azaz w és i maximuma).

Erdős-Gallai-lineáris(n, s, L)

01 $H_1 = s_1$ // 01. sor: H_1 beállítása
02 **for** $i = 2$ **to** n // 02–03. sor: H további elemeinek számítása
03 $\quad H_i = H_{i-1} + s_i$
04 **if** H_n is odd // 04–06. sor: paritás ellenőrzése
05 $\quad L = 0$
06 $\quad$ **return**
07 $w = n$ // 07. sor: súlypont beállítása
08 **for** $i = 1$ **to** n // 08–12. sor: s elemeinek tesztelése
09 $\quad$ **while** $w > 1$ and $s_w < i$ // 08–10. sor: aktuális súlypont számítása
10 $\qquad w = w - 1$
11 $\quad y = \max(i, w)$ // 11. sor: aktuális vágópont számítása
12 $\quad$ **if** $H_i > i(y-1) + H_n - H_y$
13 $\qquad L = 0$ // lines 13–14: nemgrafikus s elutasítása
14 $\qquad$ **return** L
15 $L = 1$ // 15–16. sor: grafikus s elfogadása
16 **return** L

13. Következmény. *A $(0,1,n)$-szabályos $s = (s_1, \ldots, s_n)$ sorozatról az EGl algoritmus $\Theta(n)$ idő alatt dönti el, hogy s $(0,1,n)$-grafikus-e.*

Bizonyítás. A 01–03. sorok $\Theta(n)$ időt igényelnek. Mivel a w súlypontot legfeljebb n-szer frissítjük, ezért a 04–16. sorok időigénye $O(n)$, így az algoritmus futási ideje $\Theta(n)$. □

3.12. Gyors Erdős-Gallai algoritmus (EGgy)

4. Leszámlálási eredmények

Eddig például Avis és Fukuda [2], Barnes és Savage [3, 4], Burns [10], Erdős és Moser [47], Frank, Savage and Sellers [20], Kleitman és Winston [35], Rødseth, Sellers, Tverberg [58], Ruskey et al. [59], Simion [62], Stanley [70], Winston és Kleitman [76] publikáltak foksorozatok leszámlálására vonatkozó eredményeket. Az általunk vizsgált sorozatok számával kapcsolatos eredmények találhatók Sloane és Ploffe [63], valamint Stanley [69] könyvében és a *The On-Line Encyclopedia of Integer Sequences* című honlapon [65, 66, 67] is.

14. LEMMA. *Ha l, m és u egész számok, továbbá $l \geq u$ és $1 \geq m$, akkor az (l, u, m)-korlátos sorozatok $K(l, u, m)$ száma*

$$K(l, u, m) = (u - l + 1)^m. \tag{40}$$

Bizonyítás. A korlátos sorozatok számára vonatkozó képlet közvetlen adódik abból, hogy a sorozat mind az m eleme $u - l + 1$ lehetséges értéket vehet fel. □

15. LEMMA. *Ha 1, m és u egész számok, továbbá $l \geq u$ és $1 \geq m$, akkor az (l, u, m)-szabályos sorozatok $R(l, u, m)$ száma*

$$\rho(l, u, m) = \binom{m + u - l}{m}. \tag{41}$$

Bizonyítás. Egy $s = (s_1, \ldots, s_m)$ (l, u, m)-szabályos sorozat esetén legyen $s' = (s'_1, \ldots, s'_m)$, ahol $s'_i = s_i + m - i$. A lehetséges s és s' sorozatok halmazai között kölcsönösen egyértelmű kapcsolat áll fenn. A különböző s' sorozatok száma pedig annyi, ahány féleképpen a különböző l, $l+1$, $\ldots$, $u+m-1$ számok – azaz $u+m-l$ szám – közül m számot ki tudunk választani. □

A szimulációs vizsgálatok elemzésénél (is) hasznos a szabályos és a páros sorozatok számát megadó függvények tulajdonságainak ismerete.

16. LEMMA. *Ha $n \geq 1$, akkor*

$$\frac{R(n+2)}{R(n+1)} > \frac{R(n+1)}{R(n)}, \tag{42}$$

$$\lim R_{n\to\infty} = 4, \tag{43}$$

továbbá

$$\frac{4^n}{\sqrt{4\pi n}}(1 - \frac{1}{2n}) < R(n) < \frac{4^n}{\sqrt{4\pi n}}(1 - \frac{1}{8n+8}). \tag{44}$$

Bizonyítás. A (41) egyenlőség alapján

$$\frac{R(n+2)}{R(n+1)} = \frac{(2n+3)!(n+1)n!}{(n+2)!(n+1)!(2n+1)!} = \frac{4n+6}{n+2} = 4 - \frac{2}{n+2}, \tag{45}$$

ahonnan (42) és (43) is közvetlenül adódik.

(44) belátásához felhasználjuk a Stirling-formula következő alakját [12]: ha $n \geq 1$, akkor

$$n! = \left(\frac{n}{e}\right)^n \sqrt{2\pi n} e^{\tau_n} \tag{46}$$

ahol

$$\frac{1}{12n+1} < \tau_n < \frac{1}{12n}. \tag{47}$$

□

17. Lemma. (Ascher [1], Sloane and Pfoffe [63]) *Ha $n \geq 1$, akkor a $(0,1,n)$-páros sorozatok $E(n)$ száma*

$$E(n) = \frac{1}{2}\left(\binom{2n-1}{n} + \binom{n-1}{\lfloor n \rfloor}\right). \tag{48}$$

Bizonyítás. Lásd [63]. □

A 15. és a 17. lemmák egybe vetése mutatja, hogy a páros és páratlan sorozatok számának nagyságrendje megegyezik, azonban több a páros sorozat, mint a páratlan. A 17. lemma alapján pontosan meg tudjuk adni $E(n)$ aszimptotikus nagyságrendjét.

18. Lemma. *Ha $n \geq 1$, akkor*

$$\frac{E(n+2)}{E(n+1)} > \frac{E(n+1)}{E(n)}, \tag{49}$$

$$\lim E_{n\to\infty} = 4, \tag{50}$$

továbbá

$$\frac{4^n}{\sqrt{\pi n}}(1-\delta_3(n)) < E(n) < \frac{4^n}{\sqrt{\pi n}}(1-\delta_4(n)), \tag{51}$$

ahol $\delta_3(n)$ és $\delta_4(n)$ monoton csökkenve nullához tartó sorozatok.

Bizonyítás. A bizonyítás hasonló a 16. lemma bizonyításához. □

Amint azt a következő állítás és a 1. ábra is mutatja, az $E(n)/R(n)$ hányadosok sorozata monoton csökkenve $\frac{1}{2}$-hez tart.

19. Következmény. *Ha $n \geq 1$, akkor*

$$\frac{E(n+1)}{R(n+1)} < \frac{E(n)}{R(n)} \tag{52}$$

és

$$\lim_{n\leftarrow\infty} \frac{E(n)}{R(n)} = \frac{1}{2}. \tag{53}$$

Bizonyítás. ???? □

Bár az alapfeladatban nemnegatív elemekből álló sorozatok szerepelnek, algoritmusaink – a futási idő csökkentése érdekében – csak a sorozatok pozitív kezdőszeletét vizsgálják. Ennek várható hatását jellemzi a következő két állítás, amelyek a nullát tartalmazó sorozatok számát és a sorozatokban lévő nullák átlagos számát adják meg.

20. LEMMA. *Ha $n \geq 1$, akkor az n-szabályos sorozatok közül*

$$\binom{2n-2}{n-1} = \frac{n}{2n-1}R(n). \tag{54}$$

tartalmaz legalább egy nullát.

Bizonyítás. A nullát tartalmazó n-szabályos sorozatok halmaza kölcsönösen egyértelműen leképezhető az $(n-1,n)$-szabályos sorozatok halmazára. Az utóbbi halmaz elemszáma pedig a 15. lemma szerint

$$\binom{2n-2}{n-1} = \frac{(2n-2)!n}{n(n-1)!(2n-1)} = \frac{n}{2n-1}\binom{2n-1}{n}. \tag{55}$$

□

21. LEMMA. *Ha $n \geq 1$, akkor az n-szabályos sorozatokban lévő nullák átlagos száma*

$$\frac{\sum_{i=0}^{n} i\binom{2n-3}{n-i}}{\binom{2n-1}{n}}. \tag{56}$$

Bizonyítás. A $(0,1,n)$-szabályos sorozatokban $0,\ldots,n-1$ vagy n nulla van. Az i nullát és $n-i$ pozitív elemet tartalmazó $(0,1,n)$-szabályos sorozatok száma ugyanannyi, mint ahány féleképpen $2n-3$ különböző elem közül $n-i$ kiválasztható. □

A pontos algoritmusokról szóló 3.1. részben beláttuk, hogy elég az n-páros sorozatok nullamentes prefixét megvizsgálni ahhoz, hogy eldöntsük, grafikus-e a vizsgált sorozat. Mivel a 20. lemma szerint a páros sorozatoknak aszimptotikusan csak nullmértékű hányada tartalmaz nullát (és ez a hányad a gyakorlat számára legérdekesebb n-ekre sem nagy), konkrét sorozatok vizsgálatánál nem jelentős az időmegtakarítás. Amikor viszont az összes n-páros sorozatot elemezzük (az átlagos futási idő vagy $G(n)$ meghatározása érdekében), nagyon hasznos a következő lemma.

Legyen $G_z(n)$ a nullamentes grafikus n-páros sorozatok száma.

22. LEMMA. *Ha $n \geq 2$, akkor a $(0,1,n)$-grafikus sorozatok száma*

$$G(n) = G_z(n) + G(n-1). \tag{57}$$

Bizonyítás. Az n-grafikus sorozatokban vagy $s_n = 0$, vagy $s_n > 0$. Az előbbiekben vagy $s_1 = n - 1$, vagy $s_1 < n_1$. Ha $s_1 = n - 1$ és $s_n = 0$, akkor az s sorozat biztosan nem grafikus, mert nincs benne elég pozitív elem. Az $s_1 < n - 1$ és $s_n = 0$ tulajdonságú sorozatok $n - 1$ hosszú fejei pontosan a $(0, 1, n - 1)$-grafikus sorozatok. □

Az ugró elemek számának várható értéke lényegesen befolyásolja a velük kapcsolatos algoritmusok futási idejét. Ezért hasznos a következő két állítás.

Egy $(1, 1, n)$-korlátos sorozat *szivárvány száma* a sorozatban lévő különböző elemek száma.

23. LEMMA. (Móri [48]). *Ha $n \geq 1$, akkor az n-korlátos sorozatok átlagos szivárvány száma*

$$\frac{n^2}{2n-1}. \tag{58}$$

Bizonyítás. □

24. LEMMA. (Móri [48]). *Ha $n \geq 1$, akkor az n-szabályos sorozatok átlagos szivárvány száma*

$$\frac{n}{2} + \frac{1}{4} + \frac{3}{8n-4}. \tag{59}$$

Bizonyítás. Egy n-szabályos sorozatban $1, 2, \ldots, n$ különböző elem lehet. n különböző elem közül k elemet $\binom{n}{k}$ féleképpen választhatunk ki. Ha már kiválasztottunk k elemet, akkor ezeket $\binom{n-1}{k-1}$ módon rendezhetjük, hiszen az azonos elemekből álló részsorozatokat elválasztó $k - 1$ helyet a sorozat n eleme közti $n - 1$ hely közül kell kiválasztanunk. Figyelembe véve, hogy a **??**. lemma szerint az n-szabályos sorozatok száma $\rho(n) = \binom{2n-1}{n}$, az n-szabályos sorozatok átlagos szivárványszáma

$$\frac{\sum_{k=1}^{n} k\binom{n}{k}\binom{n-1}{k-1}}{\binom{2n-1}{n}}. \tag{60}$$

Ezt a kifejezést átalakítva és felhasználva a hipergeometrikus sorozat várható értékére vonatkozó képletet, az adódik, hogy a keresett várható érték

$$\frac{n^2}{2n-1} = \frac{n}{2} + \frac{1}{4} - \frac{3}{8n-4}. \tag{61}$$

□

A következő állítás a rendező és az eltoló Havel-Hakimi algoritmusok futási idejének összehasonlításánál hasznosak.

25. LEMMA. *Ha $n \geq 1$, akkor egy n-szabályos sorozatot $s = (s_1^{e_1}, \ldots, s_q^{e_q})$ alakban felírva az e_j kitevők („futamhosszak”) várható értéke*

konstans ???

Bizonyítás. ???? □

A grafikus sorozatok $G(n)$ számának jellemzésével kapcsolatos kutatások ígéretes iránya a páros számok pozitív összeadandókra való felbontása, és annak vizsgálata, hogy az ilyen felbontások közül melyek $(0, 1, n)$-grafikusak [3, 4, 10]. Ezek segítségével sikerült a grafikus sorozatok számára vonatkozó alábbi aszimptotikus korlátokat bizonyítani.

26. LEMMA. (Burns [10]) *Léteznek olyan pozitív c és C állandók, hogy a $(0, 1, n)$-grafikus sorozatok $G(n)$ száma a következő korlátok közé esik:*

$$\frac{4^n}{cn} < G(n) < \frac{4^n}{(\log n)^C \sqrt{n}} \ . \tag{62}$$

5. Tesztelő algoritmusok

Sorozatok megvalósíthatóságának vizsgálata során természetes észrevétel, hogy az s sorozat i-hez tartozó fejének H_i fokszámigényét részben belső (az adott fejen belüli), részben pedig külső (a fejnek megfelelő farokhoz tartozó) fokszámokkal elégítjük ki.

Először egy „pozitív", majd egy „paritásos", egy „binomiális" és végül egy „fejfelező" tesztelő/szűrő algoritmust mutatunk be.

5.1. Pozitív teszt

A farokban lévő nulla elemek nem növelik a farok párosítási lehetőségeit. Ez az észrevétel lehetővé teszi, hogy az i-edik elemhez tartozó farok foklekötési lehetőségeire (potenciáljára) T_i-nél pontosabb becslést adjunk. Ez a teszt a Havel-Hakimi algoritmus első menetének megfelelő ellenőrzést végzi el. legyen p az s sorozat pozitív elemeinek a száma.

27. KÖVETKEZMÉNY. *Ha $n \geq 1$ és $s = (s_1, \ldots, s_n)$ $(0, 1, n)$-grafikus sorozat, akkor*

$$s_1 \leq p - 1 \quad \textit{vagy} \quad s_1 = 0. \tag{63}$$

Bizonyítás. A (63) egyenlőtlenség azt a követelményt fejezi ki, amelyet a Havel-Hakimi algoritmus az első iterációs menetben, illetve az Erdős-Gallai algoritmus a (6) egyenlőtlenség $i = 1$ esetben való ellenőrzésével megvalósít. □

A 27. következményen alapuló tesztet a következő algoritmus végzi.

Bemenet. n: a csúcsok száma ($n \geq 1$);
$s = (s_1, \ldots, s_n)$: $(0, 1, n)$-szabályos sorozat.

Kimenet. L: s grafikusságát jelző logikai változó.

Munkaváltozó. p: a bemenetben lévő pozitív elemek száma.

POZITÍV-TESZT(n, s, L)

```
L = 0
p = n
while s_p = 0
        p = p − 1
if s_1 > p − 1
    return L
L = 1
return L
```

Ennek az algoritmusnak a futási ideje a legjobb $\Theta(1)$ és a legrosszabb $\Theta(n)$ között változik.

Ennek az algoritmusnak a javított változata az alábbi Gyors-teszt (GyT) [43].

Gyors-teszt(n, s, L)

```
if s_{s_1} == 0
    L = 0
    return L
```

A Gyors-teszt ugyanazt az eredményt adja, mint Pozitív-teszt, a futási ideje azonban mindig $\Theta(1)$.

5.2. Paritás teszt

Első tesztünk az Erdős-Gallai tétel első szükséges feltételén alapul. Nagyon hatékony teszt, mivel mind a korlátos, mind a szabályos sorozatoknak körülbelül fele páratlan sorozat, és a teszt ezekről lineáris idő alatt megállapítja, hogy biztosan nem grafikus sorozatok.

28. Lemma. *Ha $n \geq 1$ és s $(1, 1, n)$-grafikus sorozat, akkor*

$$H_n \quad \textit{páros.} \tag{64}$$

Bizonyítás. Egy egyszerű gráf minden éle kettővel növeli a fokszámok összegét. □

Ezt az állítást a 3. tétel következményeként is megkaphatjuk. A 28. lemmában javasolt tesztet a következő algoritmus végzi.

Bemenet. n: a csúcsok száma ($n \geq 1$);
$s = (s_1, \ldots, s_n)$: $(0, 1, n)$-szabályos sorozat.

Kimenet. L: s grafikusságát jelző logikai változó.

Munkaváltozó. i: ciklusváltozó.

Paritás-teszt(n, c, L)

$L = 0$
$H_1 = 0$
for $i = 2$ **to** n
    $H_i = H_{i-1} + c_i$
if H_n páratlan
    return L
$L = 1$
return L

Ennek az algoritmusnak a lépésszáma minden esetben $\Theta(n)$.

5.3. Binomiális teszt (Bt)

Harmadik tesztünk az Erdős-Gallai tétel másik szükséges feltételének ötletét terjeszti ki. Lényege, hogy a fej igényének a fejen belül ki nem elégíthető részét a faroknak, a farok igényének belül ki nem elégíthető részét a fejnek kell kielégítenie, végül a teljes sorozat igényét a fej és farok együttműködésével, valamint a fej és a farok belső éleivel kell kielégíteni. Az algoritmus nevét arról kapta, hogy a fej és a farok belső éleinek a számát egy-egy binomiális együttható segítségével becsüljük. Legyen p az s sorozat pozitív elemeinek a száma.

29. Lemma. *Ha* $n \geq 1$ *és* s $(0, 1, n)$*-grafikus sorozat, akkor*

$$2H_i \leq i(i-1) + T_i \quad (i = 1, \ldots, p). \tag{65}$$

Bizonyítás. A (65) egyenlőtlenség azt fejezi ki, hogy a fej H_i igényét a legfeljebb $i(i-1)$ belső lehetőség és a farok legfeljebb T_i kapacitása segítségével kell kielégíteni, ahol $T_I = H_n - H_i$. □

A 29. lemmában javasolt tesztet végzi el a következő program.

Bemenet. n: a csúcsok száma ($n \geq 1$);
$s = (s_1, \ldots, s_n)$: páros sorozat;
$H = (H_1, \ldots, H_n)$: H_i az s sorozat első i elemének összege.

Kimenet. L: s grafikusságát jelző logikai változó.

Munkaváltozók. i: ciklusváltozó;
p: az s sorozat pozitív elemeinek a száma.

Binomiális-teszt(n, s, L)

$p = n$
while $s_p == 0$
    $p = p - 1$
if $p == 1$
    $L = 0$
    return L

```
H_1 = 0
for i = 2 to p
        H_i = H_{i-1} + s_i
for i = 1 to p
    if  2H_i > i(i - 1) + H_p
        L = 0
        return L
L = 1
return L
```

Az algoritmus azért kezdi s végénél p meghatározását, mert a 22. lemma szerint kevés nulla várható a sorozatokban.

Ennek az algoritmusnak a futási ideje a legjobb $\Theta(1)$ és a legrosszabb $\Theta(n)$ között változik.

Az eddigi szimulációs vizsgálatok szerint nagyon hatékony szűrő algoritmus. Aszimptotikus hatékonysága kulcs fontosságú az optimális tesztelő algoritmus futási ideje szempontjából.

Megjegyezzük, hogy Binomiális-teszt $i = 1$ esetén elvégzi Pozitív-teszt munkáját, ezért a Pozitív-teszt algoritmusra nincs szükségünk. A várható futási idő szempontjából viszont a konstans idő alatt hatékony Gyors-teszt hasznos lehet.

Felmerült, hogy a Binomiális-teszt algoritmust is csak az ellenőrző pontokon alkalmazzuk, a szimulációs kísérletek azonban azt mutatták, hogy ezzel csökkenne az algoritmus hatékonysága.

n helyett p viszont gyengítené az algoritmust, mert például a rossz $(2, 2, 0)$ sorozatot *nem* szűrné ki. Ha azonban csak a páros nullamentes sorozatokat vizsgáljuk, a $2, 2, 0$ és hasonló sorozatokat egyetlen algoritmusunk sem kell tesztelnie (mert ezeket már a bemenő sorozatok előállítása során kiszűrjük).

5.4. Fej felezése (Ft)

Az s sorozat fokpárosító lehetőségeinek az eddigieknél pontosabb becslését kaphatjuk, ha a fejet két részre osztjuk. Legyen $\lfloor i/2 \rfloor = h_i$. Ekkor az $(s_1, \ldots, s_{h_i})$ sorozatot az i indexhez tartozó fej *elejének*, az $(s_{h_i+1}, \ldots, s_i)$ sorozatot pedig az i indexhez tartozó fej *végének* nevezzük.

30. Lemma. *Ha $n \geq 1$ és s $(0, 1, n)$-grafikus sorozat, akkor*

$$\begin{aligned} H_i \leq \min(&\min(H_{h_i}, T_n - T_i, h_i(n-i)) \\ &+ \min(H_i - H_{h_i}, T_n - T_i, (i-h_i)(n-i)), T_i) \\ &+ \min(h_i(i-h_i) + \binom{h_i}{2} + \binom{i-h_i}{2} \quad (i = 1, \ldots, n), \end{aligned} \tag{66}$$

továbbá

$$\min(H_{h_i}, T_n - T_i, h_i(n-i)) + \min(H_i - H_{h_i}, T_n - T_i, (i-h_i)(n-i)) \leq T_i. \tag{67}$$

Bizonyítás. Legyen G az s sorozatot megvalósító G gráf. Ekkor az i indexhez tartozó fej H_i fokszámösszegét lekötő éleinek halmazát öt részhalmazra osztjuk: a fej eleje és a farok, a fej vége és a farok közötti, a fej két része közötti, valamint a fej részein belüli élekre. Az egyes részhalmazokba tartozó élek száma legyen rendre $X_{i,1},\ \ldots, X_{i,5}$.

$X_{i,1}$ legfeljebb a fej elemeinek H_{h_i} összege, legfeljebb a farok elemeinek $T_n - T_{h_i}$ összege, és legfeljebb a fej elejéből és a farokból képezhető párok $h_{h_i}(n-h_i)$ szorzata lehet, azaz

$$X_{i,1} \leq \min(H_{h_i}, T_n - T_i, h_i(n-i)). \tag{68}$$

Hasonló gondolatmenettel kapjuk, hogy

$$X_{i,2} \leq \min(H_i - H_{h_i}, T_n - T_i, (i-h_i)(n-i)). \tag{69}$$

$X_{i,3}$ legfeljebb $h_i(i-h_i)$ és legfeljebb H_i, ezért

$$X_{i,3} \leq \min(h_i(i-h_i), H_i). \tag{70}$$

$X_{i,4}$ legfeljebb $\binom{h_i}{2}$ és legfeljebb H_{h_i}, így

$$X_{i,4} \leq \min(\binom{h_i}{2}, H_{h_i}), \tag{71}$$

míg $X_{i,5}$ legfeljebb $\binom{i-h_i}{2}$ és legfeljebb $H_i - H_{h_i}$,, ahonnan

$$X_{i,5} \leq \binom{i-h_i}{2}. \tag{72}$$

Az is követelmény, hogy a farok részei együtt nem léphetik túl a farok kapacitását, azaz,

$$X_{i,1} + X_{i,2} \leq T_i. \tag{73}$$

A (68), (69), (70), (71) és (72) egyenlőtlenségeket összegezve azt kapjuk, hogy

$$H_i \leq X_{i,1} + X_{i,2} + X_{i,3} + 2X_{i,4} + 2X_{i,5}. \tag{74}$$

Ha az (68), (69), (70), (71) és (72) egyenlőtlenségeket a (74) egyenlőtlenségbe helyettesítjük, akkor (66) adódik, míg (73) ekvivalens a (67) egyenlőtlenséggel. □

A **??**. lemmában javasolt tesztet a következő algoritmus végzi.

Bemenet. n: a csúcsok száma ($n \geq 1$);
$s = (s_1, b_2, \ldots, s_n)$: n-faroktesztelt sorozat;
$H = (H_1, \ldots, H_n)$: H_i az s sorozat első i elemének összege;
$T = (T_1, \ldots, T_n)$: T_i az s sorozat utolsó $n-i$ elemének összege.

Kimenet. L: s grafikusságát jelző logikai változó.

Munkaváltozók. i: ciklusváltozó;
$X = (X_1, X_2, X_3, X_4, X_5)$: X_j a fej vége $X_{i,j}$ paraméterének aktuális értéke.

Fejfelező-teszt(n, s, H, T, p, L)

for $i = 2$ **to** $n - 1$
  $h = \lfloor i/2 \rfloor$
  $X_1 = \min(H_h, T_n - T_i, h(n - i))$
  $X_2 = \min(H_i - H_h, T_n - T_i, (i - h)(n - i))$
  $X_3 = \min(h(i - h)$
  $X_4 = \binom{h}{2}$
  $X_5 = \binom{i-h}{2}$
  if $H_i > X_1 + X_2 + X_3 + 2X_4 + 2X_5$ or $X_1 + X_2 > T_i$
    $L = 0$
    return L
$L = 1$
return L

Az algoritmus futási ideje legjobb esetben $\Theta(1)$, legrosszabb esetben $\Theta(n)$.

Hasonló módon a farok felezése is további sorozatokat kiszűrését tenné lehetővé, de a szimulációs kísérletek szerint ez nem csökkentené a várható futási időt.

6. Közelítő algoritmusok hatékonysága és futási ideje

A tesztek elemzésénél a szabályos és páros sorozatokat vettük alapul. A páros sorozatok halmaza a legkisebb olyan halmaz, melynek elemszámát explicit képlettel meg tudjuk adni. Az $n - 1 \geq b_i \geq 1$ feltételeknek eleget tevő *n-korlátos sorozatok* halmazának elemszámát is könnyű megadni, de ezen halmazok elemszáma túl gyorsan nő n növekedtével. A szabályos sorozatok elemzéséhez szerencsére nem kell *minden* korlátos sorozatot előállítani: elegendő a szabályos sorozatokat előállítani, és a rájuk vonatkozó hatékonysági jellemzőket a nekik megfelelő gyakoriságokkal súlyozni. Például egy azonos elemekből álló *homogén* szabályos sorozatnak egyetlen korlátos sorozat felel meg, míg a különböző elemekből álló $(n, n-1, \ldots, 1, 0)$ „szivárvány" sorozatnak $n!$ különböző korlátos sorozat felel meg.

Az alapvető pontos algoritmusokat kétféle módon próbáljuk gyorsítani (azaz várható futási idejüket csökkenteni). Az egyik út, hogy csökkentjük az általuk elvégzendő ellenőrzések számát. A másik út pedig az, hogy gyors (lineáris) előtesztekkel igyekszünk a rossz sorozatok jelentős részét kiszűrni, hogy csak a lehetséges bemenetek kis hányadánál legyen szükség a viszonylag lassú, de pontos alapalgoritmusokra. A második típusra pedig példa, hogy

Az első típusú javításra példa az Erdős-Gallai algoritmus ugrása. A második típusra pedig példa a Havel-Hakimi algoritmus kiegészítése előzetes paritásvizsgálattal, valamint az Erdős-Gallai algoritmus kiegészítése nullamentesítéssel.

A futási idők csökkentése érdekében *minden* algoritmus csak a páros, nullamentes sorozatokat vizsgálta.

Adott A algoritmusnak az n hosszúságú szabályos sorozatokra vonatkozó hatékonyságát az A algoritmus által kizárt n hosszúságú sorozatok és az ugyanolyan hosszúságú szabályos sorozatok számának hányadosával jellemezzük. Ezt a hányadost $\mathrm{R_A}(n)$-nel jelöljük és az A algoritmus n hosszúságú sorozatokra vonatkozó *hatékonyságának* nevezzük (lásd a **??**. ábrát).

A következő közelítő algoritmusokat vizsgáljuk:

1) Nullamentesítő-teszt
2) Binomiális-teszt
3) Fejfelező-teszt

A **??**. táblázat a Binomiális-teszt és a Fejfelező-teszt programok futási eredményeit tartalmazza.

A 6. táblázat a nullamentes binomiális és nullamentes faroktesztelt sorozatok számát, továbbá a grafikus sorozatok számát és a grafikus sorozatok számának szomszédos n helyeken felvett értékei hányadosát tartalmazza $n = 1, \ldots, 29$ esetén. ??????????????????????????????

A **??**. táblázat a Nullamentes-teszt, Binomiális-teszt, Fejfelező-teszt és az Erdős-Gallai-Gyorsan programok futási eredményeit tartalmazza.

A 7. táblázat azt jellemzi, hogy a vizsgált közelítő algoritmusok a szabályos sorozatoknak milyen hányadát szűrik ki. A táblázat a nullamentes páros sorozatok száma ($E_z(n)$) mellett tartalmazza a nullamentes binomiális ($B_z(n)$), a nullamentes faroktesztelt ($F_z(n)$) és a grafikus sorozatok ($G(n)$) számának, valamint a szabályos sorozatok számának hányadosát.

A **??**. ábra a nullamentes, binomiális és fejtesztelt sorozatoknak a szabályos sorozatokhoz viszonyított részarányát mutatja be. ??????????????????????????

A 8. táblázat a közelítő algoritmusok elemenkénti átlagos futási idejét és műveletszámát tartalmazza. Az algoritmusok bemenete az összes páros nullamentes sorozat volt. A mikromásodpercben mért idő és a műveletszám is tartalmazza a sorozatok előállítására fordított időt és műveletszámot is.

Ha $n = 2$, akkor (82) szerint $\rho(n) = \binom{3}{2} = 3$ szabályos sorozat van: $(1, 1)$, $(1, 0)$ és $(0, 0)$. Ha egy szabályos sorozat elemeinek összege páros, akkor *páros sorozatnak* nevük. Az n hosszúságú páros sorozatok számát $\pi(n)$-nel jelöljük. Ezzel a jelöléssel $\pi(2) = 2$. A Binomiális-teszt által elfogadott, n hosszúságú sorozatok számát $\beta(n)$-nel jelölve $\beta(2) = 2$. Az n hosszúságú helyreállítható sorozatok számát jelöljük $\gamma(n)$-nel. Ekkor $\gamma(2) = 2$ és a Binomiális-teszt hibája (hatékonysága) $R_{\mathrm{BT}}(2) = 2/2 = 1$.

Ha $n = 3$, akkor a szabályos sorozatok száma $\rho(3) = 10$. Ezek közül a (2,2,2), (2,2,0), (2,1,1), (2,0,0), (1,1,0) és (0,0,0) páros, azaz $\epsilon(3) = 6$. Ezek közül a Binomiális-teszt kizárja a (2,2,0) és (2,0,0) sorozatokat, így $\beta(3) = 4$. A megmaradt 4 sorozat grafikus, így $\varphi(3) = \tau(3) = \gamma(3) = 4$.

Ha $n = 4$, akkor a szabályos sorozatok száma $\rho(4) = 35$. Ezek közül 19 a páros, és a következő 11 grafikus: (3,3,3,3), (3,3,2,2), (3,2,2,1), (3,1,1,1,), (2,2,2,2), (2,2,2,0), (2,2,1,1), (2,1,1,0), (1,1,1,1), (1,1,0,0) és (0,0,0,0). A 19 páros sorozat közül Binomiális-teszt is kizárja azt a nyolc sorozatot, amelyeket az Erdős-Gallai kizárna, így $\beta(4) = \gamma(4) = \varphi(4) = \gamma(4) = 11$.

A $\rho(5) = 126$ szabályos sorozat közül $\epsilon(5) = 66$ a páros, ezek között pedig $\beta(5) =$

n	$B_z(n)$	$F_z(n)$	$G_z(n)$	$G(n+1)/G(n)$
1	**1**	**0**	**1**	2.000000
2	**2**	**2**	**2**	2.000000
3	**4**	**4**	**4**	2.750000
4	**11**	**11**	**11**	2.818182
5	**31**	**31**	**31**	3.290323
6	103	**102**	**102**	3.352941
7	349	344	**342**	3.546784
8	1256	1230	1213	3.595218
9	4577	4468	4361	3.672552
10	17040	16582	16016	3.705544
11	63944	62070	59348	3.742620
12	242218	234596	222117	3.765200
13	922369	891852	836315	3.786674
14	3530534	3409109	3166852	3.802710
15	13563764	13082900	12042620	3.817067
16	52283429	50380684	45967479	3.828918
17	202075949	194550002	176005709	3.839418
18	782879161	753107537	675759564	3.848517
19	3039168331	2921395019	2600672458	3.856630
20	11819351967	11353359464	10029832754	3.863844
21			38753710486	3.870343
22			149990133774	3.876212
23			581393603996	3.881553
24			2256710139346	3.886431
25			8770547818956	3.890907
26			34125389919850	3.895031
27			132919443189544	3.897978
28			518232001761434	3.898843
29			2022337118015338	

6. táblázat. A nullamentes binomiális ($B_z(n)$), nullamentes faroktesztelt ($F_z(n)$) sorozatok száma, valamint a $(0,1,n)$-grafikus sorozatok száma (G_n) és a grafikus sorozatok sorozatok számának szomszédos n helyeken felvett értékeinek hányadosa $(G(n+1))/G(n)$ $n = 1,\dots,28$ csúcs esetén.

31 a binomiális. Ezek a sorozatok mind grafikusak, azaz $\varphi(5) = \tau(5) = \gamma(5) = 31$.

A $\rho(6) = 462$ szabályos sorozat közül $\epsilon(6) = 236$ a páros, amelyek között $\beta(6) = 103$ binomiális sorozat van. Binomiális-teszt a 102 grafikus sorozat mellett az (5,5,3,3,3,1) rossz sorozatot is elfogadja. Ezek szerint a legfeljebb 5 hosszúságú sorozatokra nézve a Binomiális-teszt hibátlanul kiszűri a nem grafikus sorozatokat, a 6 hosszú sorozatokra azonban már csak közelítő algoritmus. A Fejfelező-teszt ezzel a sorozattal is megbirkózik, ezért $\varphi(6) = \tau(6) = \gamma(6) = 102$.

n	$E_z(n)$	$E_z(n)/R(n)$	$B_z(n)/R(n)$	$F_z(n)/R(n)$	$G(n)/R(n)$
1	0	0.000000	1.000000	1.000000	1.000000
2	1	0.333333	0.666667	0.666667	0.666667
3	2	0.300000	0.400000	0.400000	0.400000
4	9	0.257143	0.314286	0.314286	0.314286
5	28	0.230159	0.246032	0.246031	0.246032
6	110	0.238095	0.222943	0.220779	0.220779
7	396	0.231352	0.203380	0.200466	0.199301
8	1519	0.236053	0.195183	0.191142	0.188500
9	5720	0.235335	0.188276	0.183793	0.179391
10	21942	0.237524	0.184460	0.179502	0.173375
11	83980	0.238098	0.181290	0.175977	0.168260
12	323554	0.239301	0.179145	0.173508	0.164278
13	1248072	0.240000	0.177368	0.171500	0.160821
14	4829708	0.240784	0.176014	0.169960	0.157882
15	18721080	0.241379	0.174884	0.168684	0.155271
16	72714555	0.241946	0.173965	0.167634	0.152950
17	282861360	0.242424	0.173188	0.166738	0.150844
18	1101992870	0.242860	0.172533	0.165972	0.148926
19	4298748300	0.243243	0.171970	0.165306	0.147158
20	16789046494	0.243590	0.171486	0.164725	0.145521
21					0.143997
22					0.142569
23					0.141228
24					0.139961
25					0.138762
26					0.137625
27					0.136542
28					0.135509
29					0.134521

7. táblázat. The number of zerofree even sequences, further the ratio of the numbers binomial/regular, headsplitted/regular and graphical/regular sequences.

??????????????????????????????????????Eddig jó A $\rho(7) = 1716$ szabályos sorozat között $\epsilon(6) = 868$ a páros, melyek közül $\beta(7) = 376$ a binomiális. A binomiális sorozatok között még 34 rossz van, melyek közül a Pozitív-teszt a 27 grafikus sorozat mellett a következő 7 rosszat is elfogadja: $(6, 6, 6, 4, 4, 4, 2)$, $(6, 6, 5, 4, 4, 4, 1)$, $(6, 6, 4, 4, 4, 3, 1)$, $(6, 6, 4, 3, 3, 3, 1)$, $(6, 6, 3, 3, 3, 2, 1)$, $(6, 5, 3, 3, 3, 1, 1)$, $(5, 5, 3, 3, 3, 1, 0)$. A következő Fejfelező-teszt ezek közül a $(6, 6, 4, 3, 3, 3, 1)$ kivételével mindet kiszűri, így $\varphi(7) = 343$. A következő Farokfelező-teszt $i = 4$ mellett legfeljebb $8 + 2$ fokot tud lekötni a fej eleje és a farok részei között, legfeljebb további $4 + 0$ fokot a fej vége és a farok részei között, legfeljebb további 8 fokot a fej két része

n	Bt, s	Bt, művelet	Ft, s	Ft, művelet
1	0	14	0	15
2	0	41	0	43
3	0	180	0	200
4	0	716	0	815
5	0	2 918	0	3 321
6	0	11 918	0	13 675
7	0	48 952	0	56 299
8	0	201 734	0	233 182
9	0	831 374	0	964 121
10	0	3 426 742	0	3 988 542
11	0	14 107 824	0	16 469 036
12	0	58 028 152	0	67 929 342
13	0	238 379 872	0	279 722 127
14	0	978 194 400	1	1 150 355 240
15	2	4 009 507 932	3	4 724 364 716
16	6	16 417 793 698	13	19 379 236 737
17	26	67 160 771 570	51	79 402 358 497
18	106	274 490 902 862	196	324 997 910 595
19	423	1 120 923 466 932	798	1 328 948 863 507
20	1 627	4 573 895 421 484	3 201	5 429 385 115 097

8. táblázat. Running time of BINOMIAL-TEST (BT) and HEADSPLITTER-TEST (HT) in secundum and as the number of operations for $n = 1, \ldots, 20$.

között, és két fokot a fej elején belül. Ez azonban összesen csak $10 + 4 + 8 + 2 = 24$ fok, ami kevesebb a sorozat $H_7 = 26$ összfokszámánál. Tehát $\tau(7) = \gamma(7) = 342$.

A **??**. ábrán minden sorban az első pontos értéket félkövéren írtuk. Eszerint $n \leq 4$ esetén $\beta(n) = \gamma(n)$, azaz a BINOMIÁLIS-TESZT ugyanannyi sorozatot fogad el, mint a pontos algoritmusok. $n > 4$ esetén egyre nő a BINOMIÁLIS-TESZT hibája: $n = 5$ esetén még csak egyetlen páros sorozatról nem ismeri fel, hogy rossz, $n = 6$ esetén már hatszor hibázik.

POZITÍV-TESZT $n = 5$-ig hibátlan, a FEJFELEZŐ-TESZT $n = 6$-ig, a FAROKFELEZŐ-TESZT pedig $n = 7$-ig.

A **??**. ábrán $\rho(n)$ értéke $n = 24$-ig az EIS A001700 sorozata [65], $\epsilon(n)$ értéke $n = 23$-ig az EIS A005654 sorozata [67], $\gamma(n)$ értéke pedig $n = 23$-ig az EIC A0004251-es sorozata [66]. A többi értéket mi határoztuk meg: sem $\rho(25), \ldots \rho(40)$, sem $\epsilon(24), \ldots, \epsilon(40)$, sem a $\beta(n)$, $\pi(n)$, $\varphi(n)$ és $(\tau(n)$ értékek nem szerepelnek az EIS-ben.

Ebben a cikkben első sorban a soros algoritmusokkal kapott eredményekről számoltunk be.

A témakörben vannak párhuzamos eredmények is [52, 61, 68]. Saját párhuzamos eredményeinket a 10. részben ismertetjük.

7. Pontos algoritmusok futási ideje

A pontos algoritmusok sorozatonkénti átlagos futási idejét n függvényében mikromásodpercben az alábbi 9. ábra tartalmazza. Az algoritmusok csak a páros, nullamentes sorozatokat vizsgálták. A futási idő tartalmazza a sorozatok előállítási idejét is sorozatok előállítási idejét

n	HH	HHE	EG	EGu	EGL
1	???	???	???	???	???
2	???	???	???	???	???
3	???	???	???	???	???
4	???	???	???	???	???
5	???	???	???	???	???
6	???	???	???	???	???
7	???	???	???	???	???
8	???	???	???	???	???
9	???	???	???	???	???
10	???	???	???	???	???
11	???	???	???	???	???
12	???	???	???	???	???
13	???	???	???	???	???
14	???	???	???	???	???
15	???	???	???	???	???
16	???	???	???	???	???

9. táblázat. Pontos algoritmusok elemenkénti átlagos futási ideje műveletszámban.

A következő pontos algoritmusokat vizsgáljuk:
1) HH: Rendező Havel-Hakimi algoritmus (HH).
2) HHE: Eltoló Havel-Hakimi algoritmus (HHE).
3) EG: Erdős-Gallai algoritmus (EG).
4) EGU: Erdős-Gallai algoritmus ugrásokkal (EGU).
5) EGL: Erdős-Gallai algoritmus ugrásokkal lineárisan (EGL).

A pontos algoritmusok sorozatonkénti átlagos futási idejét n függvényében mikromásodpercben a **??**. ábra tartalmazza.

Figure **??** contains the total number of operations of the algorithms HHSo, HHSh, EG, and EGL required for the testing of all even sequences of length $n = 1, \ldots, 15$. The operations necessary to generate the sequences are included.

Comparison of the first two columns shows that algorithm HHSh is much quicker than HHSo, especially if n increases. Comparison of the third and fourth columns shows that we get substantial decrease of the running time if we have to test the input sequence only in the check points. Finally the comparison of the third and fifth columns demonstrates the advantages of a linear algorithm over a quadratic one.

Figure 11 shows the running time of Erdős-Gallai-Linear in secundum and

n	HHSo	HHSh	EG	EGJ	EGL
1	10	15	87	-	-
2	40	61	119	12	37
3	231	236	267	116	148
4	1 170	1 052	946	551	585
5	5 969	4 477	4 000	2 677	2 339
6	31 121	20 153	18 206	12 068	9 539
7	157 345	88 548	82 154	54 184	38 984
8	784 341	393 361	372 363	238 813	160 126
9	3 628 914	1 726 484	1 666 167	1 666 167	656 575
10	17 345 700	7 564 112	7 418 447	4 552 276	2 692 240
11	80 815 538	32 895 244	32 737 155	19 680 986	11 018 710
12	385 546 527	142 460 352	143 621 072	84 608 529	45 049 862
13	1 740 003 588	613 739 913	626 050 861	362 141 061	183 917 288
14	8 066 861 973	2 633 446 908	2 715 026 827	1 543 745 902	750 029 671
15	36 630 285 216	11 254 655 388	11 717 017 238	6 557 902 712	3 055 289 271

10. táblázat. Total number of operations as the function of n for precise algorithms HHSo, HHSh, EG, EGJ, and EGL.

n	$E(n)$	$T(n)$, s	$Op(n)$	$T(n)/E(n)/n$, s	$Op(n)/E(n)/n$
2	2	0	37	0	9.25000000000
3	6	0	148	0	8.22222222222
4	19	0	585	0	7.69736842105
5	66	0	2 339	0	7.08787878788
6	236	0	9 539	0	6.73658192090
7	868	0	38 984	0	6.41606319947
8	3 235	0	160 126	0	6.18724884080
9	12 190	0	656 575	0	5.98464132714
10	46 252	0	2 692 240	0	5.82080774885
11	176 484	0	11 018 710	0	5.67587378511
12	676 270	0	45 049 862	0	5.55126675243
13	2 600 612	0	183 917 288	0	5.44005937537
14	10 030 008	1	750 029 671	0.000000007121487	5.34132654018
15	38 781 096	5	3 055 289 271	0.000000008595253	5.25219687963
16	150 273 315	23	12 434 367 770	0.000000009565903	5.17156346504
17	583 407 990	79	50 561 399 261	0.000000007965367	5.09797604337
18	2 268 795 980	297	205 439 740 365	0.00000000727258	5.03056202928

11. táblázat. Az Erdős-Gallai-Lineárisan algoritmus teljes és amortizált futási ideje másodpercben és a műveletek számában

operation, and also the amortized number of operation/even sequence.

The most interesting data of Figure 11 are in the last column: they show that the number of operations/investigated sequence/length of the investigated sequence is monotone decreasing (see [59]).

Figure 12 shows the distribution of the $E(n) - G(n)$ even nongraphical sequences according to the number of tests made by ERDŐS-GALLAI-UGRÓ to exclude the given sequence for $n = 3, \ldots, 15$ vertices. $f_i(n) = f_i$ gives the frequency of even nongraphical sequences of length n, which requeired exactly i round of the test.

These data show, that the maximal number of tests is about $\frac{n}{2}$ in all lines.

$E(n) - G(n)$	n/i	f_1	f_2	f_3	f_4	f_5	f_6	f_7
2	3	2	0	0	0	0	0	0
8	4	6	2	0	0	0	0	0
35	5	33	2	0	0	0	0	0
134	6	122	12	0	0	0	0	0
526	7	459	65	2	2	0	0	0
2022	8	1709	289	24	0	0	0	0
7829	9	6421	1228	176	4	0	0	0
30236	10	24205	4951	1013	67	0	0	0
115136	11	91786	19603	5126	610	11	0	0
454153	12	349502	76414	23755	4274	208	0	0
1764297	13	1336491	296036	104171	25293	2277	29	0
6863156	14	5128246	1142470	439155	133946	18673	666	0
26738476	15	19739076	4404813	1803496	655291	127116	8603	81

12. táblázat. Distribution of the even nongraphical sequences according to the number of tests made by ERDŐS-GALLAI-JUMPING to exclude the given sequence for $n = 3, \ldots, 15$.

Figure 13 shows the average number of required rounds for the nongraphical, graphical and all even sequences. The data of the column belonging to $G(n)$ are computed using Lemma . It is remarkable that the sequences of the coefficients are monotone decreasing in the last three columns.

Figure **??** presents the distribution of the graphical sequences according to their first element. These data help at the design of the algorithm ERDŐS-GALLAI-LESZÁMLÁLÓ which computes the new values of $G(n)$ (in the slicing of the computations belonging to a given value of n).

We see in Figure **??** that from $n = 6$ the multiplicities increase up to $n - 2$, and the last positive value is smaller then the last but one element.

Az EG-LINEÁRISAN-LESZÁMLÁLÓ algoritmus teljes és az elemenkénti futási időit tartalmazza a **??**. ábra.

A **??**. ábra azt mutatja, hányszor van az ERDŐS-GALLAI-LINEÁRISAN algoritmusnak $1, \ldots, n-1$ menetre szüksége, hogy a rossz sorozatokat kiszűrje.

Az 17. ábrán az látható, hogy az EG-LINEÁRISAN ciklusmagja átlagosan (gyakoriságokkal súlyozva) hányszor futott le, ha csak a rossz sorozatokat vesszük figyelembe, illetve a ha minden sorozatot figyelembe veszünk (mindkét esetben csak a páros, nullamentes sorozatokat vettük figyelembe).

A **??**. ábra pedig a grafikus sorozatok kezdő elem szerinti megoszlását mutatja

n	$E(n)$	$G(n)$	$E(n) - G(n)$	átlagos $E(n) - G(n)$	átlagos $G(n)$	átlagos $E(n)$
3	6	4	2	$0.3333n$	$0.8000n$	$0.6444n$
4	19	11	8	$0.3125n$	$0.5714n$	$0.4661n$
5	66	31	35	$0.2114n$	$0.5555n$	$0.3730n$
6	236	102	134	$0.1967n$	$0.5455n$	$0.3730n$
7	868	342	526	$0.1649n$	$0.5385n$	$0.3475n$
8	3233	1213	2020	$0.1458n$	$0.5333n$	$0.2911n$
9	12190	4363	7829	$0.1337n$	$0.5294n$	$0.2753n$
10	46232	16016	30216	$0.1249n$	$0.5263n$	$0.2700n$
11	174484	59348	115136	$0.1175n$	$0.5238n$	$0.2557n$
12	676270	222117	454153	$0.1085n$	$0.5217n$	$0.2444n$
13	2603612	836313	1767299	$0.1035n$	$0.5200n$	$0.2373n$
14	10030008	3166852	6863156	$0.0960n$	$0.5185n$	$0.2294n$
15	38761096	12042620	26718476	$0.0934n$	$0.5172n$	$0.2251n$

13. táblázat. Weighted average number of tests made by Erdős-Gallai-ugró által az $n = 3, \ldots, 15$ hosszú sorozatok vizsgálata során végzett tesztek átlagos száma.

n/b_1	0	1	2	3	4	5	6	7	8	9	10	11
1	1											
2	1	1										
3	1	1	2									
4	1	1	4	4								
5	1	2	7	10	11							
6	1	3	10	22	35	31						
7	1	3	14	34	78	110	102					
8	1	4	18	54	138	267	389	342				
9	1	4	23	74	223	503	968	1352	1213			
10	1	5	28	104	333	866	1927	3496	4895	4361		
11	1	5	34	134	479	1356	3471	7221	12892	17793	16016	
12	1	6	40	176	661	2049	5591	13270	27449	47757	65769	59348

14. táblázat. The distribution of the graphical sequences according to b_1 for $n = 1, \ldots, 12$.

be. Ezek az adatok segítenek a $G(n)$ számítását végző Erdős-Gallai-Gyorsan program párhuzamos megvalósításánál (a számítások több részre osztásában).

8. $(0, b, n)$-gráfok

Ebben a részben a klasszikus tételek $(0, b, n)$-gráfokra való kiterjesztésével foglalkozunk.

n	$\epsilon(n)$	$\tau(n)$, mp	$Op(n)$	$\tau(n)/\epsilon(n)/n$, mp	$Op(n)/\epsilon(n)/n$
1	???	???	???	???	???
2	???	???	???	???	???
3	???	???	???	???	???
4	???	???	???	???	???
5	???	???	???	???	???
6	???	???	???	???	???
7	???	???	???	???	???
8	???	???	???	???	???
9	???	???	???	???	???
10	???	???	???	???	???
11	???	???	???	???	???
12	???	???	???	???	???
13	???	???	???	???	???
14	???	???	???	???	???
15	???	???	???	???	???
16	???	???	???	???	???
17	???	???	???	???	???
18	???	???	???	???	???
19	???	???	???	???	???
20	???	???	???	???	???
21	???	???	???	???	???
22	???	???	???	???	???
33	???	???	???	???	???
24	???	???	???	???	???
25	???	???	???	???	???

15. táblázat. Az EG-Lineárisan algoritmus elemenkénti futási ideje műveletszámban kifejezve.

n 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
3 2 0 0
4 8 0 0 0
5 33 2 0 0 0
6 122 12 0 0 0 0
7 459 65 2 0 0 0 0
8 1709 289 24 0 0 0 0 0
9 6421 1228 176 4 0 0 0 0 0
10 24205 4951 1013 67 0 0 0 0 0 0
11 91786 19603 5126 610 11 0 0 0 0 0 0
12 349502 76414 23755 4274 208 0 0 0 0 0 0 0
13 1336491 296036 104171 25293 2277 29 0 0 0 0 0 0 0
14 5128246 1142470 439155 133946 18673 666 0 0 0 0 0 0 0 0
15 19739076 4404813 1803496 655291 127116 8603 81 0 0 0 0 0 0 0 0
Ezek az UGRÓ LEG adatai

16. táblázat. Az EG-Lineárisan elutasítás előtti meneteinek száma.

n=3 11 /6 /3 = 0.611111
n=4 47 /19 /4 = 0.618421
n=5 192 /66 /5 = 0.581818
n=6 792 /236 /6 = 0.559322
n=7 3229 /868 /7 = 0.531435
n=8 13268 /3235 /8 = 0.512674
n=9 54244 /12190 /9 = 0.494431
n=10 222057 /46252 /10 = 0.480102
n=11 906558 /176484 /11 = 0.466979
n=12 3698529 /676270 /12 = 0.455751
n=13 15063277 /2600612 /13 = 0.445554
n=14 61286926 /10030008 /14 = 0.436454
n=15 249056158 /38781096 /15 = 0.428140
n=16 1011175412 /150273315 /16 = 0.420557
n=17 4101727713 /583407990 /17 = 0.413567
n=18 16625570580 /2268795980 /18 = 0.407107

17. táblázat. Az EG-Lineárisan menetszámai súlyozott átlagának n-ed része, ha a grafikus sorozatokat is figyelembe vesszük, illetve ha nem.

n/b_1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	1				1	1	1	1	1	1	1	1	1	1	1
2	1	1				1	1	1	1	1	1	1	1	1	1
3	1	1	2				1	1	1	1	1	1	1	1	1
4	1	1	4	4					1	1	1	1	1	1	1
5	1	2	7	10	11				1	1	1	1	1	1	1
6	1	3	10	22	35	31				1	1	1	1	1	1
7	1	3	14	34	78	110	102				1	1	1	1	1
8	1	4	18	54	138	267	389	342				1	1	1	1
9	1	4	23	74	223	503	968	1352	1213				1	1	1
10	1	15	28	104	333	866	1927	3496	4895	4361					1
11	1	5	34	134	479	1356	3471	7221	12892	17793	16016				
12				1	1	1	1	1	1	1	1	1	1	1	1
13				1	1	1	1	1	1	1	1	1	1	1	1
14				1	1	1	1	1	1	1	1	1	1	1	1

18. táblázat. A grafikus sorozatok b_1 szerinti eloszlása.

8.1. Erdős-Gallai tétel és Chungphaisan tétele

1974-ben Chungphaisan [14] mind az Erdős-Gallai tételt, mind pedig a Havel-Hakimi tételt kiterjesztette $(0, b, n)$-gráfokra. Az EG tétel kiterjesztése a következő.

31. Tétel. (Chungphaisan [14]) *Legyen $n \geq 1$. A $(0, b, n)$-szabályos $s = (s_1, \ldots, s_n)$ sorozat akkor és csak akkor $(0, b, n)$-grafikus, ha*

$$\sum_{i=1}^{n} s_i \quad \textit{páros} \tag{75}$$

és

$$\sum_{i=1}^{j} s_i - bj(j-1) \le \sum_{k=j+1}^{n} \min(bi, s_k) \quad (j = 1, \ldots, n-1). \tag{76}$$

A tételen alapuló algoritmus legrosszabb esetben négyzetes időt igényel. A következő állítás lehetővé teszi, hogy a $(0, b, n)$-szabályos sorozatokat legrosszabb esetben $\Theta(n)$ idő alatt teszteljük.

32. TÉTEL. (Iványi, [28]) *Ha* $n \ge 1$, *a* $(0, b, n)$*-szabályos* $s = (s_1, \ldots, s_n)$ *sorozat akkor és csak akkor* $(0, b, n)$*-grafikus, ha*

$$H_n \quad \textit{páros} \tag{77}$$

és

$$H_i > bi(y_i - 1) + H_n - H_y \quad (i = 1, \ldots, n-1), \tag{78}$$

ahol

$$y_i = \max(i, w_i) \quad (i = 1, \ldots, n-1). \tag{79}$$

Bizonyítás. □

A következő CHUNGPHAISAN-ERDŐS-GALLAI-LINEÁRIS algoritmus (ChEGl) – amely az EG algoritmus természetes általánosítása – $O(n)$ idő alatt eldönti, hogy egy $(0, b, n)$-szabályos sorozat $(0, b, n)$-grafikus-e.

Bemenet. n: csúcsok száma $(n \ge 1)$;
$s = (s_1, \ldots, s_n)$: $(0, b, n)$-szabályos sorozat; b: a gráf két csúcsa között megengedett élek maximális száma.

Kimenet. L: s grafikusságát jelső logikai változó.

Munkaváltozók. i: ciklus változók;
$w = (w_1, \ldots, w_n)$: w_i az i indexhez tartozó súlypont.

CHUNGPHAISAN-ERDŐS-GALLAI-LINEÁRIS(n, s, b, L)

01 $H_1 = s_1$ // 01 sor: H_1 kezdeti értékének beállítása
02 **for** $i = 2$ **to** $n - 1$ // 02–03. sor: H további elemeinek számítása
03 $\quad H_i = H_{i-1} + s_i$
04 **if** H_n páratlan // 04–06. sor: paritás ellenőrzése
05 $\quad L = 0$ // 05–06. sor: páratlan sorozat elutasítása
06 $\quad$ **return**
07 $w = n$ // 07. sor: első súlypont értékének beállítása
08 **for** $i = 1$ **to** $n - 1$ // 08–15. sor: s tesztelése
09 $\quad$ **while** $s_w < ib$ és $w > 0$
10 $\quad\quad w = w - 1$
11 $\quad y = \max(i, w)$
12 $\quad$ **if** $H_i > bi(y - 1) + H_n - H_y$

13 **return** L // 13. sor: s nem grafikus
14 $L = 1$ // 14–15. sor: s grafikus
15 **return** L

33. Tétel. (Iványi, [28]) ChEGl *futási ideje minden esetben* $\Theta(n)$.

Bizonyítás. A 01–06 sorok végrehajtása $\Theta(n)$ időt igényel. Mivel w szigorúan monoton csökken a program végrehajtása során, ezért a 07–14 sorok $O(n)$ időt igényelnek, így az algoritmus futási ideje minden esetben $\Theta(n)$. □

Legyen $b = 3$ és $s = (13, 10, 5, 5, 4, 1)$. $H_6 = 38$ páros. Ha $i = 1$, akkor $w_i = y = 5$ és a 11. sor feltétele $(13 \leq 3 \cdot 1 \cdot (5-1))$ nem teljesül. Ha $i = 2$, akkor viszont $w_i = y = 2$ és a feltétel teljesül $(23 > 3 \cdot 2 \cdot (2-1)) + 5 + 5 + 4 + 1)$, ezért s *nem* $(0, 3, 6)$*-grafikus.*

Maradjon b 3, de s-et változtassuk meg: legyen $s' = (13, 10, 5, 5, 4, 3)$. Az előző példához képest a futás során az első változás az, hogy amikor $i = 2$, akkor $(23 \leq 3 \cdot 2 \cdot (2-1)) + 5 + 5 + 4 + 3$, és így a 11. sorban lévő feltétel nem teljesül, és ugyanez az eredmény $i = 3,\ 4$ és 4 esetén is, ezért s' $(0, 3, 6)$*-grafikus.*

A következő táblázatokban bemutatjuk, hogyan oszlanak meg a kizárt grafikus és nemgrafikus sorozatok az egyes menetek között. Azt is jellemezzük, hogy átlagosan hány meneten át kell egy grafikus, illetve nemgrafikus sorozatot a kizárásáig tesztelni, és azt is, hogy a menetek hányadrészét fordítjuk átlagosan egy sorozat tesztelésére.

A 19. táblázat a ChEGl i-edik $(i = 1, \ldots, 11)$ menetében kiszűrt nemgrafikus sorozatok számát tartalmazza $a = 0$, $b = 2$ és $n = 1,\ \ldots,\ 11$ csúcs esetén.

n/i	1	2	3	4	5	6	7	8	9	10
1	0									
2	0	6								
3										
4										
5										
6										
7										
8										
9										
10										
11										

19. táblázat. ChEGL i-edik $(i = 1, \ldots, 11)$ menetében kiszűrt nemgrafikus sorozatok száma $a = 0$, $b = 2$ és $n = 1,\ \ldots,\ 11$ csúcs esetén.

A **??**. táblázat a ChEGl i-edik $(i = 1, \ldots, 11)$ menetében kiszűrt grafikus sorozatok számát tartalmazza $a = 0$, $b = 2$ és $n = 1,\ \ldots,\ 11$ csúcs esetén.

A következő 21 táblázat a ChEGl algoritmus hatékonyságát jellemzi $a = 0$, $b = 2$ és $n = 1,\ \ldots,\ 11$ csúcs esetén.

n/i	1	2	3	4	5	6	7	8	9	10
1	0									
2	0	6								
3										
4										
5										
6										
7										
8										
9										
10										
11										

20. táblázat. ChEGl i-edik ($i = 1, \ldots, 11$) menetében kiszűrt grafikus sorozatok száma $a = 0$, $b = 2$ és $n = 1, \ \ldots, \ 11$ csúcs esetén.

n/jellemző	X	Y	Z	X'	Y'	Z'
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						

21. táblázat. ChEGl hatékonysági jellemzői $a = 0$, $b = 2$ és $n = 1, \ \ldots, \ 11$ csúcs esetén.

8.2. Az első 6 és 18 sor elemzése

8.3. Gyors ChEGgy algoritmus (ChEGgy)

8.4. Havel-Hakimi tétel és Chungphaisan tétele

Chungphaisan [14] a következő módon terjesztette ki a Havel-Hakimi tételt.

34. TÉTEL. (Chungphaisan [14]) *Legyen* $n \geq 2$ *és* $b \geq 1$. *Az* $s = (s_1, \ldots, s_n)$ $(0, b, n)$-*szabályos sorozat akkor és csak akkor* $(0, b, n)$-*grafikus, ha a* j-*edik* b-*redukált* $w_j^* = (w_1^*, \ldots, w_{n-1}^*)$ *sorozat* $(0, b, n)$-*grafikus minden* $1 \geq j \geq n$ *indexre.*

A tételen alapuló algoritmus nagyon lassú. A tétel következő javítása azonban lehetővé teszi, hogy a tesztelést legrosszabb esetben is el tudjuk végezni $O(n)$ idő alatt.

35. TÉTEL. (Iványi, [28]) *Legyen* $n \geq 1$ *és* $b \geq 1$ *Nemnegatív egészek* $s = (s_1, \ldots, s_n)$ $(0, 1, n)$-*szabályos sorozata akkor és csak akkor* $(0, b, n)$-*grafikus, ha*

$$??? \qquad (80)$$

és

$$???. \tag{81}$$

A következő Chungphaisan-Havel-Hakimi-lineáris algoritmus (ChHHl) – amely a HH algoritmus természetes általánosítása – $O(n)$ idő alatt eldönti, hogy egy $(0, b, n)$-szabályos gráf $(0, b, n)$-grafikus-e.

Bemenet. n: csúcsok száma ($n \geq 1$);
$s = (s_1, \dots, s_n)$: $(0, b, n)$-grafikus sorozat; b: a gráf két csúcsa között megengedett élek maximális száma.

Kimenet. L: s grafikusságát jeltő logikai változó.

Munkaváltozó. i: ciklus változó;
$w = (w_1, \dots, w_n)$: w_i az i indexhez tartozó súlypont.

Chungphaisan-Havel-Hakimi-lineáris(n, s, b, L)

01 $L = 0$ // 01. sor: a gyakoribb érték beállítása
02 **if** $s_1 == 0$ // 02–04. sor: a nullákból álló sorozat grafikus
03 $\quad L = 1$
04 $\quad$ **return** L
05 **if** $s_{\lceil s_1/b+1 \rceil} == 0$ // 05–07. sor: s_1 ellenőrzése konstans idő alatt
06 $\quad$ **return** L
07 $H_1 = s_1$ // 07 sor: H_1 kezdeti értékének beállítása
08 **for** $i = 2$ **to** $n - 1$ // 08–09. sor: H további elemeinek számítása
09 $\quad H_i = H_{i-1} + s_i$
10 **if** H_n páratlan // 10–11. sor: paritás tesztelése
11 $\quad$ **return** L
12 $w_1 = n$ // 12. sor: első súlypont kezdeti értékének beállítása
13 **while** $s_{w_1} < b$ and $w_1 > 0$
14 $\quad\quad w_1 = w_1 - 1$
15 **if** $s_1 > b(w_1 - 1) + H_n - H_{w_1}$
16 $\quad$ **return** L
17 $r_1 = b(w_1 - 1) + H_n - H_{w_1} - s_1$ // 12. sor: első maradék számítása
18 **for** $i = 2$ **to** $n - 1$ // 18–26. sor: s tesztelése
19 $\quad$ **if** $H_{i-1} \geq H_n/2$ vagy $s_i \leq 1$ vagy $s_{i+1} = 0$ // 19–21. sor: s elfogadása
20 $\quad\quad L = 1$
21 $\quad\quad$ **return** L
22 $\quad w_i = w_{i-1}$ // 22–24. sor:
23 $\quad$ **while** $s_i < bi$ és $w_i > 0$
24 $\quad\quad\quad w_i = w_i - 1$
25 $\quad$ **if** $w_i \geq i$ // 25–27. sor: esetszétválasztás
26 $\quad\quad$ **if** $s_i > b(w_i - 1) + r_{i-1} + H_{w_{i-1}} - H_{w_i}$
$\quad\quad\quad -b(w_{i-1} - w_i)(i - 1)$ // 26. sor: s_i tesztelése
27 $\quad\quad\quad$ **return** L

28 $\quad r_i = b(w_i - 1) + r_{i-1} + H_{w_{i-1}} - H_{w_i}$
$\quad\quad -b(w_{i-1} - w_i)(i-1) - s_i$ //28. sor: maradék frissítése
29 **else if** $s_i > bw_i + r_{i-1} + H_{w_{i-1}} - H_{w_i}$
30 $\quad\quad -b(w_{i-1} - w_i)(i-1)$
31 $\quad$ **return** L
32 $\quad r_i = bw_i + r_{i-1} + H_{w_{i-1}} - H_{w_i}$
$\quad\quad -b(w_{i-1} - w_i)(i-1) - s_i$ //31. sor: maradék frissítése
33 $L = 1$ // 33–34. sor: s elfogadása
34 **return** L

A következő állítás jellemzi ChHHl futási idejét.

36. Tétel. (Iványi, [28]) ChHHl *futási ideje minden esetben* $\Theta(n)$.

Bizonyítás. A 01–06 sorok végrehajtása $\Theta(n)$ időt igényel. A 07–11 sorok végrehajtása $O(n)$ ideig tart. Mivel w szigorúan monoton csökken a program végrehajtása során, ezért a 12–24 sorok $O(n)$ időt igényelnek, így az algoritmus futási ideje minden esetben $\Theta(n)$. □

Legyen $b = 3$ és $s = (13, 10, 5, 5, 4, 1)$. Az ötödik és tizedik sorok feltételei nem teljesülnek és $r_1 = 0$. Ha $i = 2$, akkor $w_i = 5$ és teljesül a 20. sor feltétele, így s *nem* $(0, 1, 6)$*-grafikus.*

A következő példában b maradjon 3, viszont s-et változtassuk meg: legyen $s' = (13, 10, 5, 5, 4, 3)$. Az előző esethez képest annyi a változás, hogy $r_1 = 2$ az első maradék, majd $i = 2$ esetén $w_i = 2$, nem teljesül a 20. sor feltétele és $r_2 = 0$. $i = 3$ esetén teljesül a 19. sor $H_{i-1} \geq H_n/2$ feltétele, ezért s' $(0, 1, 6)$*-grafikus.*

A következő példában legyen $b = 1$ és $s = (4, 3^3, 1)$. A 05. és 10. sorok feltételei nem teljesülnek és $r_1 = 0$. Ha $i = 2$, akkor $w_i = 4$ és nem teljesül a 20. sor feltétele, az $i = 3$ esetben pedig a 19. sorban teljesül a $H_{i-1} \geq H_n/2$ feltétel, azaz s $(0, 1, 5)$*grafikus.*

A 22. táblázat a ChHHL i-edik ($i = 1, \ldots, 11$) menetében kiszűrt nemgrafikus sorozatok számát tartalmazza $a = 0$, $b = 2$ és $n = 1, \ldots, 11$ csúcs esetén.

n/i	1	2	3	4	5	6	7	8	9	10
1	0									
2	0	6								
3										
4										
5										
6										
7										
8										
9										
10										
11										

22. táblázat. ChHHL i-edik ($i = 1, \ldots, 11$) menetében kiszűrt nemgrafikus sorozatok száma $a = 0$, $b = 2$ és $n = 1, \ldots, 11$ csúcs esetén.

A 23. táblázat a ChHHL i-edik ($i = 1, \ldots, 11$) menetében kiszűrt grafikus sorozatok számát tartalmazza $a = 0$, $b = 2$ és $n = 1, \ \ldots, \ 11$ csúcs esetén.

n/i	1	2	3	4	5	6	7	8	9	10
1	0									
2	0	6								
3										
4										
5										
6										
7										
8										
9										
10										
11										

23. táblázat. ChHHL i-edik ($i = 1, \ldots, 11$) menetében kiszűrt grafikus sorozatok száma $a = 0$, $b = 2$ és $n = 1, \ \ldots, \ 11$ csúcs esetén.

A következő 24 táblázat a ChHHL algoritmus hatékonyságát jellemzi $a = 0$, $b = 2$ és $n = 1, \ \ldots, \ 11$ csúcs esetén.

n/jellemző	X	Y	Z	X'	Y'	Z'
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						

24. táblázat. ChHHl hatékonysági jellemzői $a = 0$, $b = 2$ és $n = 1, \ \ldots, \ 11$ csúcs esetén.

A 25. táblázat a ChEGL i-edik ($i = 1, \ldots, 11$) menetében kiszűrt nemgrafikus sorozatok számát tartalmazza $a = 2$, $b = 5$ és $n = 1, \ \ldots, \ 11$ csúcs esetén.

9. (a, b, n)-gráfok

Egy (a, b, n)-gráfban minden csúcspár elemei legalább a éllel össze vannak kötve. Ezért ha minden csúcspár esetén eltávolítunk a élet, egy $(0, b - a, n)$-*gráfot* kapunk. Ezért a Chungphaisan tételnek közvetlen következménye az alábbi állítás.

37. Következmény. *Legyen $n \geq 2$. Az $s = (s_1, \ldots, s_n)$ (a, b, n)-szabályos sorozat akkor és csak akkor $(0, b, n)$-grafikus, ha az $s' = (s_1 - a(n-1), \ldots, s_n - a(n-1))$ sorozat $(0, b - a, n)$-grafikus.*

n/i	1	2	3	4	5	6	7	8	9	10
1	0									
2	0	4								
3										
4										
5										
6										
7										
8										
9										
10										
11										

25. táblázat. ChHHL i-edik ($i = 1, \ldots, 11$) menetében kiszűrt nemgrafikus sorozatok száma $a = 2$, $b = 5$ és $n = 1, \ \ldots, \ 11$ csúcs esetén.

A 37. következmény szerint a következő három táblázat adatai megegyeznek a $(0, 3, n)$-szabályos sorozatokra vonatkozó hasonló adatokkal.

A 26. táblázat a ChEGl i-edik ($i = 1, \ldots, 11$) menetében kiszűrt nemgrafikus sorozatok számát tartalmazza $a = 2$, $b = 5$ és $n = 1, \ \ldots, \ 11$ csúcs esetén.

n/i	1	2	3	4	5	6	7	8	9	10
1	0									
2	0	4								
3										
4										
5										
6										
7										
8										
9										
10										
11										

26. táblázat. ChEGl i-edik ($i = 1, \ldots, 11$) menetében kiszűrt nemgrafikus sorozatok száma $a = 2$, $b = 5$ és $n = 1, \ \ldots, \ 11$ csúcs esetén.

A 27. táblázat a CL i-edik ($i = 1, \ldots, 11$) menetében kiszűrt grafikus sorozatok számát tartalmazza $a = 2$, $b = 5$ és $n = 1, \ \ldots, \ 11$ csúcs esetén.

A következő 28 táblázat a ChEGl algoritmus hatékonyságát jellemzi $a = 2$, $b = 5$ és $n = 1, \ \ldots, \ 11$ csúcs esetén.

10. Grafikus sorozatok száma

A **??**. ábra 1-től 29 csúcsig tartalmazza a grafikus sorozatok számát. A táblázat úgy készült, hogy párhuzamosítottuk az Erdős-Gallai-Gyorsan algoritmust. Mivel viszonylag sok processzor vett részt a számolásban, viszont bizonytalan volt, hogy az egyes processzorok meddig vehetnek részt a számolásban, a feladatot *szeleteknek* nevezett kisebb részekre bontottuk. Célszerű volt, hogy a szeletek feldolgozása hasonló

n/i	1	2	3	4	5	6	7	8	9	10
1	0									
2	0	4								
3										
4										
5										
6										
7										
8										
9										
10										
11										

27. táblázat. ChEGl i-edik ($i = 1, \ldots, 11$) menetében kiszűrt grafikus sorozatok száma $a = 2$, $b = 5$ és $n = 1, \ldots, 11$ csúcs esetén.

n/jellemző	X	Y	Z	X'	Y'	Z'
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						

28. táblázat. ChEGl hatékonysági jellemzői $a = 2$, $b = 5$ és $n = 1, \ldots, 11$ csúcs esetén.

ideig tartson.

10.1. Leszámláló Erdős-Gallai algoritmus

A Erdős-Gallai-Lineárisan algoritmus egyik lehetséges alkalmazása, hogy meghatározzuk a grafikus sorozatok számát olyan n értékekre, amelyekre eddig a nagy számolásigény miatt nem volt ismert: Sloane *The On-Line Encyclopedia of Integer Sequences* című honlapja [64] az $n = 23$ értékig tartalmazza a grafikus sorozatok számát.

Az alábbi Erdős-Gallai-leszámláló (EGE) algoritmus a lineáris legrosszabb eset mellett azt is igyekszik kihasználni, hogy ha lexikografikus sorrendben ellenőrizzük a szóba jövő sorozatokat, akkor a szomszédos sorozatok bizonyos tulajdonságai nagyon hasonlóak, ezért adott sorozat jellemzői az őt megelőző sorozat jellemző adataiból konstans várható idő alatt meghatározhatóak.

Igyekszünk az ellenőrizendő sorozatok számát is csökkenteni.

Ennek egy egyszerű megoldása, hogy eleve csak a páros sorozatokat állítjuk elő. További ötlet, hogy csak a nullamentes sorozatokat vizsgáljuk. A nullát tartalmazó

$(0, 1, n)$-grafikus sorozatok között ugyanis a 22. lemma szerint pontosan $G(n-1)$ grafikus sorozat van. Igaz, hogy aszimptotikusan a nullát tartalmazó sorozatok a páros sorozatok elhanyagolható részét adják, de az általunk most gyakorlatilag vizsgált $n \in [4, 30]$ tartományban még jelentős a részarányuk.

Lényeges gyorsítást jelent az is, hogy a sorozatokat csak az ellenőrző pontokban vizsgáljuk.

Az EGE program azt is kihasználja, hogy a szomszédos sorozatok ellenőrző pontjainak a listája átlagosan konstans idő alatt származtatható a megelőző sorozat adataiból. A kiindulási értékek szintén könnyen számíthatók: az első – $q = (n-1)^n$ – sorozatra a C lista üres (azaz egyáltalán nem kell ellenőrzést végeznünk), a súlypontok listája pedig kezdetben $w = \left((n-1)^{n-1}\right)$.

Az Erdős-Gallai-Gyorsan algoritmus előállítja és megvizsgálja az n-páros, nullamentes sorozatokat, és kimenetként megadja a $G(n)$ értéket. Az algoritmus kihasználja, hogy a páros sorozatok lexikografikusan csökkenő sorozatában szomszédos sorozatok több lényeges paramétere hasonló, ezért ezek a paraméterek a vizsgált s' sorozatot megelőző s sorozat adott paraméteréből gyorsan meghatározhatóak.

Ellenőrző pontoknak nevezzük a n-nél kisebb ugró pontokat. Az ellenőrző pontok $C(b')$ listája rendszerint megegyezik a $C(s)$ listával, és legfeljebb a végén változik egy vagy két elem.

Bemenet. n: a csúcsok száma ($n \geq 4$); (azért korlátozzuk n-et alulról, hogy a programot mentesítsük a rövid sorozatok speciális tulajdonságainak figyelembe vételétől).

Kimenet. G: a $(0, 1, n)$-grafikus sorozatok száma.

Munkaváltozók. i és j: ciklusváltozók;
$s = (s_1, \ldots, s_n)$: b_i az éppen tesztelt páros, nullamentes sorozat i-edik eleme;
$s_0 = n - 1$: segédváltozó a H sorozat elemeinek kiszámításához;
$H = (H_1, \ldots, H_n)$: H_i az éppen tesztelt b első i elemének az összege;
H_0: segédváltozó H elemeinek számolásához;
$C(s) = C = (c_1, c_2, \ldots, c_{q-1})$: ellenőrző pontok maximális hosszúságú listája;
c: ellenőrző pontok száma az aktuális C listában;
$w = w_1, \ldots, w_{n-1}$: w_i a legnagyobb indexű olyan s_j indexe, amely legalább i;
q: ellenőrzésre eddig az indexig van szükség;
k: ellenőrzés egyszerűsítéséhez használt változó.

Erdős-Gallai-leszámláló(n, γ)

```
01 H_0 = c = 0                                ▷ 01–03. sor: kezdeti értékek beállítása
69      go to 07
```

Mivel a futási idő csökkentése érdekében az Erdős-Gallai-Gyorsan algoritmus csak nullamentes sorozatokat állít elő és tesztel, a szeletekre bontás alapja a **??**. lemma alábbi következménye.

38. KÖVETKEZMÉNY. *Ha Ha m és u pozitív egész számok, akkor az $(1, u, m)$-szabályos sorozatok $\rho(1, u, m)$ száma*

$$\rho(1, u, m) = \binom{m + u - 1}{m}. \tag{82}$$

Bizonyítás. A **??**. lemmában alkalmazzuk az $l = 1$ helyettesítést. □

Feltételezzük, hogy az n-szabályos nullamentes sorozatok halmazának szeletekre való felbontásánál az egyes szeletek futási ideje arányos a hozzájuk tartozó $\rho(1, u, m)$-szabályos sorozatok számával.

Most tekintsünk egy másik példát: legyen $n = 29$. Az $n = 28$ esetben szerzett tapasztalatok alapján feltesszük, hogy a tiszta futási idő összesen 2500 nap lesz. Feltételezve, hogy a gépek egy részét csak éjszakára kapjuk meg, legyen egy szelet maximális futási ideje 12 óra. Ekkor egyenletes eloszlás mellett 5000 szelet lenne.

11. Nyitott problémák

Az alábbiakban néhány nyitott problémát, további feladatot mutatunk be.

11.1. Különböző gráfosztályok foksorozatainak leszámlálása

Explicite formulas are not known for the numbers of degree sequences of (a, b, n)-graphs.

Kleitman [35] gave asymptotic bounds for the number of $(1, 1, n)$-digraphs, while Burns [10] proved bounds for the number of $(0, 1, n)$-undigraphs. Narayana and Bent [49] gave recursive formulas to compute the number of degree-sequences of $(0, 1, n)$-digraphs. Pécsy and Szűcs [52] used the formulas of Narayana and Bent to compute quickly the concrete values of the function $D(0, 1, n)$. Using a quick parallel program in 2011 Iványi et al. [**?**] computed the values $G(n)$ until $n = 29$.

11.2. Konkrét problémák bonyolultsága

[19]

11.3. Optimalizálás

[26]

11.4. Közelítő problémák

11.5. Befokok és kifokok egyidejű vizsgálata

11.6. További gráfosztályok potenciális foksorozatainak tesztelése

Páros gráfok, többrészes gráfok, hipergráfok

12. Összefoglalás

????????????????????

Köszönetnyilvánítás. A szerzők köszönik ??? A kutatás az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg (a támogatás száma TÁMOP 4.2.1/B-09/1/KMR-2010-0003).

Hivatkozások

[1] Ascher, M.: *Mu torere: an analysis of a Maori game.* Math. Mag. **60(2),** (1987) 90–100. ⇒ 19

[2] Avis, D., Fukuda, K. *Reverse search for enumeration.* Discrete Appl. Math. **2,** (1993) 21–46. ⇒ 18

[3] Barnes, T. M., Savage, C. D.: *A recurrence for counting graphical partitions.* Electron. J. Combin. **2,** (1995) R11, 10 pp. ⇒ 18, 22

[4] Barnes, T. M., Savage, C. D. *Efficient generation of graphical partitions. Discrete Appl. Math.* **78(1–3),** (1997) 17–26. ⇒ 1, 2, 18, 22

[5] Beasley, L. B., Brown D. E., Reid, K. B.: *Extending partial tournaments.* Math. Comput. Modelling **50(1),** (2009) 287–291. ⇒ 2

[6] Bereg S., Ito, H.: *Transforming graphs with the same degree sequence.* In: (ed. by H. Ito et al.) The Kyoto Int. Conf. on Computational Geometry and Graph Theory, LNCS **4535.** Springer-Verlag, Berlin, Heidelberg. 2008. pp. 25–32. ⇒ 2

[7] Bozóki, S., Fülöp, J., Poesz, A.: *On pairwise comparison matrices that can be made consistent by the modification of a few elements.* CEJOR Cent. Eur. J. Oper. Res. **19,** (2011) 157–175. ⇒ 1

[8] Bozóki S., Fülöp J., Rónyai, L.: *On optimal completion of incomplete pairwise comparison matrices.* Math. Comput. Modelling **52,** (2010) 318–333. ⇒ 1, 2

[9] Brualdi, A. R., Kiernan K.: *Landau's and Rado's theorems and partial tournaments,* Electron. J. Combin. **16(#N2),** (2009) (6 pp). ⇒ 2

[10] Burns, J. M.: *The number of degree sequences.* PhD Dissertation, MIT, 2007. ⇒ 2, 18, 22, 47

[11] Busch A. N., Chen G., Jacobson M. S.: *Transitive partitions in realizations of tournament score sequences.* J. Graph Theory **64(1),** (2010), 52–62. ⇒ 2

[12] Cormen, T. H., Leiserson, Ch. E., Rivest, R. L., Stein, C.: *Introduction to Algorithms.* Third edition, The MIT Press/McGraw Hill, Cambridge/New York, 2009. Magyarul: *Algoritmusok.* Műszaki Könyvkiadó, Budapest, 2003. ⇒ 4, 8, 19

[13] Coudum, S. A. *A simple proof of the Erdős-Gallai theorem on graph sequences.* Bull. Austral. Math. Soc. **33,** (1986) 67–70. ⇒ 5

[14] Chungphaisan, V. *Conditions for sequences to be r-graphical.* Discrete Math. **7,** (1974) 31–39. ⇒ 1, 37, 40, 52

[15] Del Genio, C. I., Kim, H., Toroczkai, Z., Bassler, K. E.: *Efficient and exact sampling of simple graphs with given arbitrary degree sequence.* PLoS ONE **5(4)**, e10012 (2010). ⇒ 2

[16] Erdős, P., Gallai, T.: *Gráfok előírt fokú pontokkal.* Mat. Lapok **11,** (1960) 264–274. ⇒1, 5, 52

[17] Erdős, P. L., Miklós, I., Toroczkai, Z.: *A simple Havel-Hakimi type algorithm to realize graphical degree sequences of directed graphs.* Electron. J. Combin. **17(1),** (2010) R66, 10 pp. ⇒2, 4

[18] Erdős, P., Richmond L. B.: *On graphical partitions.* Combinatorica **13(1),** (1993) 57–63. ⇒2

[19] Frank, A.: *Connections in Combinatorial Optimization.* Oxford University Press, Oxford, 2011. ⇒2, 47

[20] Frank, D. A., Savage, C. D., Sellers, J. A.: *On the number of graphical forest partitions. Ars Combin.* **65,** (2002) 33–37. ⇒18

[21] Hakimi, S. L.: *On the realizability of a set of integers as degrees of the vertices of a simple graph.* J. SIAM Appl. Math. **10,** (1962) 496–506. ⇒1, 4, 52

[22] Hakimi, S. L.: *On realizability of a set of integers as degrees of the vertices of a linear graph II. Uniqueness.* SIAM J. Appl. Math., **11(1),** (1963) 135–147. ⇒

[23] Havel, V.: A remark on the existence of finite graphs (cseh); *Časopis Pěst. Mat.* **80,** (1955), 477–480. ⇒1, 3, 52

[24] Hell, P., Kirkpatrick, D.: *Linear-time certifying algorithms for near-graphical sequences.* Discrete Math. **309(18)**, (2009) 5703–5713. ⇒2

[25] Iványi, A.: *Reconstruction of complete interval tournaments.* Acta Univ. Sapientiae, Inform., **1(1),** (2009) 71-88. ⇒1, 2, 4

[26] Iványi, A.: *Reconstruction of complete interval tournaments. II.* Acta Univ. Sapientiae, Math., **2(1),** (2010) 47–71. ⇒1, 2, 4, 47

[27] Iványi, A.: *Deciding the validity of the score sequence of a soccer tournament.* In (ed. A. Frank): Open problems of the Egerváry Research Group, Budapest, 2011. ⇒2

[28] Iványi, A.: *Test of degree sequences of multigraphs.* Annales Univ. Budapest., Computatorica (benyújtva). ⇒38, 39, 40, 42

[29] Iványi, A., Lucz, L., Móri F. T., Sótér, P.: *On the Erdős-Gallai and Havel-Hakimi algorithms.* Acta Univ. Sapientiae, Inform. **3(2),** (2011) 230–268. ⇒1, 2, 8

[30] Iványi, A., Pirzada, S.: *Comparison based ranking.* In (ed. A. Iványi): Algorithms of Informatics, Vol. 3. AnTonCom, Budapest 2011, 1262–1311. ⇒1, 2

[31] Kayibi K., Khan M. A., Pirzada S., Iványi A.: Random sampling of minimally cyclic digraphs with given imbalance sequence. Acta Univ. Sapientiae, Math. (submitted). ⇒1, 2

[32] Kemnitz, A., Dolff, S.: *Score sequences of multitournaments,* Congr. Num., **127,** (1997) 85–95. ⇒

[33] Kim, H., Toroczkai, Z., Miklós, I., Erdős, P. L., Székely, L. A.: Degree-based graph construction. J. Physics: Math. Theor. A **42(39),** (2009) 392–401. ⇒1, 2

[34] Kleitman, D. J., Wang, D. L.: *Algorithms for constructing graphs and digraphs with given valencies and factors.* Discrete Math. **6** (1973) 79–88. ⇒4

[35] Kleitman, D. J., Winston K. J.: *Forests and score vectors.* Combinatorica **1(1),** (1981) 49–54. ⇒18, 47

[36] Knuth, D. E.: *The Art of Computer Programming. Volume 4A, Combinatorial Algorithms.* Addison–Wesley, Upper Saddle River, 2011. ⇒2

[37] Kohnert, A.: *Dominance order and graphical partitions.* Elec. J. Comb. **11(1),** (2004) No. 4. 17 pp. ⇒2

[38] Kovács, G. Zs., Pataki, N.: *Rangsorolási algoritmusok elemzése.* TDK dolgozat. ELTE TTK, Budapest, 2002. 39 oldal. ⇒2

[39] LaMar, M. D. *Algorithms for realizing degree sequences of directed graphs.* arXiv-0906:0343ve [math.CO], 7 June 2010. ⇒2

[40] Landau, H. G.: *On dominance relations and the structure of animal societies. III. The condition for a score sequence.* Bull. Math. Biophys. **15,** (1953) 143–148. ⇒1

[41] Liljeros, F., Edling, C. R., Amaral, L., Stanley, H., Åberg, Y.: *The web of human sexual contacts.* Nature **411,** (2001) 907–908. ⇒1

[42] Lovász, L.: *Combinatorial Problems and Exercises* (corrected version of the second edition). AMS Chelsea Publishing, Boston, 2007. Magyarul: *Kombinatorikai problémák és feladatok.* Typotex, Budapest, 1999. ⇒3

[43] Lucz, L., Sótér, P.: *Foksorozatokat ellenőrző algoritmusok.* TDK dolgozat. ELTE IK, Budapest, 2011. ⇒23

[44] Meierling, D., Volkmann, L.: *A remark on degree sequences of multigraphs.* Math. Methods Oper. Res. **69(2),** (2009) 369–374. ⇒2

[45] Metropolis, N., Stein, P. R.: *The enumeration of graphical partitions.* European J. Comb. **1(2),** (1980) 139–153. ⇒2

[46] Miklós, I., Erdős, P. L., Soukup, L.: A remark on degree sequences of multigraphs. (2011) (benyújtva). ⇒2

[47] Moon, J. W.: *Topics on Tournaments.* Holt, Rinehart, and Winston, New York, 1968. ⇒18

[48] Móri, T.: Szóbeli közlés. Budapest, 2011. ⇒21

[49] Narayana, T. V., Bent, D. H.: Computation of the number of score sequences in round-robin tournaments. Canad. Math. Bull. **7(1),** (1964) 133–136. ⇒47

[50] Newman, M. E. J., Barabási, A. L.: *The Structure and Dynamics of Networks.* Princeton University Press, Princeton, NJ. 2006. ⇒1

[51] Özkan, S.: *Generalization of the Erdős-Gallai inequality.* Ars Combin. **98,** (2011) 295–302. ⇒

[52] Pécsy G., Szűcs, L.: *Parallel verification and enumeration of tournaments.* Stud. Univ. Babeş-Bolyai, Inform. **45(2),** (200) 11–26. ⇒31, 47

[53] Pirzada, S.: *Graph Theory.* Orient Blackswan, 2011, to appear. ⇒2

[54] Pirzada, S., Al-Assaf, A. M., Kayibi, K. K.: *On imbalances in oriented multipartite graphs.* Acta Univ. Sapientiae, Math., **3(1),** (2011) 34–42. ⇒

[55] Pirzada, S., Iványi, A., Khan, M. A.: *Score sets and kings.* In (ed. A. Iványi): Algorithms of Informatics, Vol. 3, ed. A. Iványi. AnTonCom, Budapest 2011, 1451–1490. ⇒1, 2

[56] Pirzada, S., Naikoo, T. A., Samee, U. T., Iványi, A.: *Imbalances in directed multigraphs.* Acta Univ. Sapientiae, Inform. **2(1),** (2010) 47–71. ⇒2

[57] Pirzada, S., Zhou G., Iványi A.: *On k-hypertournament losing scores,* Acta Univ. Sapientiae, Inform. **2(2),** (2010) 184–193. ⇒1

[58] Rødseth, Ø. J., Sellers, J. A., Tverberg, H.: *Enumeration of the degree sequences of non-separable graphs and connected graphs.* European J. Comb. **30(5),** 1309–1319. ⇒2, 18

[59] Ruskey, F., Cohen, R., Eades, P., Scott, A.: *Alley CAT's in search of good homes.* Congr. Num., **102,** (1994) 97–110. ⇒1, 2, 18, 33

[60] Sierksma, G., Hoogeveen, H.: *Seven criteria for integer sequences being graphic.* J. Graph Theory **15(2),** (1991) 223–231. ⇒5

[61] Siklósi, B.: *Soros és párhuzamos algoritmusok összehasonlítása sportversenyekkel kapcsolatos problémákban.* Programtervező matematikus diplomamunka. ELTE TTK, Budapest, 2001. 69 oldal. ⇒31

[62] Simion, R.: *Convex polytopes and enumeration.* Advances in Applied Math. **18(2)** (1996) 149–180. ⇒18

[63] Sloane N. J. A., Plouffe S.: *The Encyclopedia of Integer Sequences.* Academic Press, 1995. ⇒18, 19

[64] Sloane N. J. A. (szerk.): *Encyclopedia of Integer Sequences.* 2011. ⇒2, 45

[65] Sloane N. J. A.: *The number of ways to put $n+1$ indistinguishable balls into $n+1$ distinguishable boxes.* In (ed. N. J. A. Sloane): The On-line Encyclopedia of the Integer Sequences. 2011. ⇒18, 31

[66] Sloane N. J. A.: The number of degree-vectors for simple graphs. In (ed. N. J. A. Sloane): The On-line Encyclopedia of the Integer Sequences. 2011. ⇒18, 31

[67] Sloane N. J. A.: *The number of bracelets with n red, 1 pink and $n-1$ blue beads.* In (ed. N. J. A. Sloane): The On-line Encyclopedia of the Integer Sequences. 2011. ⇒18, 31

[68] Soroker, D.: *Optimal parallel construction of prescribed tournaments.* Discrete Appl. Math. **29(1),** (1990) 113–125. ⇒31

[69] Stanley, R.: *Enumerative Combinatorics. Vol. 2.* Cambridge University Press, Cambridge, 1997. ⇒18

[70] Stanley, R.: *A zonotope associated with graphical degree sequence.* In: Applied Geometry and Discrete Mathematics, Festschr. 65th Birthday Victor Klee. DIMACS Series in Discrete Mathematics and Theoretical Computer Science. **4,** (1991) 555-570. ⇒18

[71] Tripathi, A., Tyagy, H.: *A simple criterion on degree sequences of graphs.* Discrete Appl. Math. **156(18),** (2008) 3513–3517. ⇒2

[72] Tripathi, A., Vijay, S.: *A note on a theorem of Erdős & Gallai.* Discrete Math. **265(1–3),** (2003) 417–420. ⇒10, 11, 13, 14, 15

[73] Tripathi, A., Venugopalan, S., West, D. B.: *A short constructive proof of the Erdős-Gallai characterization of graphic lists.* Discrete Math. **310(4),** (2010) 833–834. ⇒1, 2, 5, 6, 52

[74] Weisstein, E. W.: *Degree sequence.* From MathWorld—Wolfram Web Resource, 2011. ⇒2

[75] Weisstein, E. W.: *Graphic sequence.* From MathWorld—Wolfram Web Resource, 2011. ⇒2

[76] Winston, K. J., Kleitman, D. J.: *On the asymptotic number of tournament score sequences.* J. Combin. Theory Ser. A. **35,** (1983) 208–230. ⇒18

Beérkezett:

IVÁNYI ANTAL
tony@compalg.inf.elte.hu
Eötvös Loránd Tudományegyetem
Informatikai Kar
1117 Budapest, Pázmány Péter sétány 1/C

LUCZ LORÁND
lorand.lucz@gmail.com
Eötvös Loránd Tudományegyetem
Informatikai Kar
1117 Budapest, Pázmány Péter sétány 1/C

Budapest, 2012. január 5b.

DEGREE SEQUENCES OF MULTIGRAPHS

ANTAL IVÁNYI, LORÁND LUCZ, TAMÁS F. MÓRI

Let a, b and n integers, $0 \leq a < b$ and $n \geq 1$. (a, b, n)-graphs are loopless multigraphs in which any two vertices are connected with an least a and at most b edges. Havel in 1955 [23], Erdős and Gallai in 1960 [16], Hakimi in 1962 [21], Tripathi, Venugopalan and West in 2010 [73] proposed a method to decide, whether a sequence of nonnegative integers can be the degree sequence of a $(0, 1, n)$-graph. Chungphaisan in 1974 [14] extended Havel-Hakimi and Erdős-Gallai theorem for $(0, b, n)$-graphs. All the mentioned algorithms require at least quadratic time in worst case. We extend Erdős-Gallai-Chungphaisan theorem for (a, b, n)-graphs and propose a linear time algorithm, based on our theorem. We also propose a linear time version of the testing part of Havel-Hakimi algorithm and extend it for $(0, 2, n)$-graphs.