Annales Univ. Sci. Budapest., Sect. Comp. 37 (2012) 195–214

DEGREE SEQUENCES OF MULTIGRAPHS

Antal Iványi (Budapest, Hungary)

Communicated by Imre Kátai

(Received January 15, 2012; accepted February 14, 2012)

Abstract. Let a, b and n be integers, $n \geq 1$ and $b \geq a \geq 0$. Let an (a,b,n)-*graph* defined as a loopless graph $G(a,b,n)$ on n vertices $\{V_1,\ldots,V_n\}$, in which V_i and V_j are connected with at least a and at most b (directed or undirected) edges. If $G(a,b,n)$ is directed, then it is called (a,b,n)-*digraph* and if it is undirected, then it is called (a,b,n)-*undigraph.* Landau in 1953 published an algorithm deciding whether a nondecreasing sequence of nonnegative integers is the out-degree sequence of a $(1,1,n)$-digraph. Moon in 1963 published a similar condition for (b,b,n)-digraphs, and in 2009 Iványi did for (a,b,n)-digraphs. Havel in 1955, Erdős and Gallai in 1960 proposed an algorithm to decide the same question for $(0,1,n)$-*undigraphs.* Their theorem was extended to $(0,b,n)$-undigraphs by Chungphaisan in 1974. In 2011 Özkan [24] proved a stronger version. The aim of this paper is to summarize and extend the known results and to propose quicker algorithms than the known ones.

1. Introduction

One of the classical problems of graph theory is the characterization of the set of degree sequences of different graph classes.

Let a, b and n be integers, $n \geq 1$ and $b \geq a \geq 0$. Let (a,b,n)-*graphs* defined as loopless graphs on n vertices, in which different vertices are connected with

---

Key words and phrases: Degree sequence, graphical sequence, (a,b,n)-graph, tournament.
2010 Mathematics Subject Classification: 05C65, 68R10.
1998 CR Categories and Descriptors: G.2.2.
The Research is supported by the European Union and co-financed by the European Social Fund (grant agreement no. TÁMOP 4.2.1./B-09/1/KMR-2010-0003).

at least a and at most b edges. For the clarity we call directed (a, b, n)-graphs as (a, b, n)-*digraphs* and undirected (a, b, n)-graphs as (a, b, n)-*undigraphs*.

Our aim is to investigate the conditions and algorithms which decide whether a monotone sequence $s = (s_1, \ldots, s_n)$ of integers is the degree sequence of an (a, b, n)-undigraph or the out-degree sequence of an (a, b, n)-digraph.

The first results belong to Landau [21], who published in 1953 a necessary and sufficient condition for $(1, 1, n)$-digraphs, and Havel [10], who gave a necessary and sufficient condition in 1955 for $(0, 1, n)$-undigraphs. The later result was independently reproved in 1962 by Hakimi [9].

The conditions for (b, b, n)-digraphs were given in 1962 by Moon [22] and for $(0, b, n)$-undigraphs by Chungphaisan [3] in 1974. The conditions for (a, b, n)-digraphs were published in 2009 [12, 13].

In this paper we summarize the results of testing of potential degree sequences of (a, b, n)-graphs including the analysis of their efficiency. The structure of the paper is as follows. After the introductory Section 1 in Section 2 we present the known results connected with directed graphs. Section 3 contains the algorithms proposed to test the potential degree sequences of $(0, 1, n)$-undigraphs, while Section 4 the results connected with $(0, b, n)$-undigraphs. In Section 5 the results on (a, b, n)-undigraphs are presented while Section 6 contains the summary of the results.

2. Conditions and algorithms for (a, b, n)-digraphs

Let l, m and u be nonnegative integers, further $l \leq u$ and $m \geq 1$. The sequence $s = (s_1, \ldots, s_m)$ of integers is called (l, u, m)-*bounded*, if $l \leq s_i \leq u$ hold for all $1 \leq i \leq m$ indices. An $s = (s_1, \ldots, s_m)$ (l, u, m)-bounded sequence is called (l, u, m)-*regular*, if $u \geq s_1 \geq \cdots \geq s_m \geq l$ or $l \leq s_1 \leq \cdots \leq s_m \leq u$ (following tradition we use nondecreasing sequences for digraphs and nonincreasing ones for undigraphs). An (l, u, m)-regular sequences is called (l, u, m)-*digraphic,* if there exists a (l, u, m)-digraph, having s as its out-degree sequence. In a similar manner an (l, u, m)-regular sequence is called (l, u, m)-*undigraphic,* if there exists a (l, u, m)-undigraph, having s as its degree sequence [8, 30, 31].

The first testing theorem for $(1, 1, n)$-digraphs belongs to Landau.

Theorem 2.1. (Landau [21]) *A sequence* $(s_1, \ldots, s_n)$ *satisfying* $0 \leq s_1 \leq \leq \cdots \leq s_n$ *is the out-degree sequence of some* $(1, 1, n)$-*digraph if and only if*

$$\sum_{i=1}^{k} s_i \geq \binom{k}{2} \quad \textit{for} \quad 1 \leq k \leq n, \tag{2.1}$$

with equality when $k = n$.

In 1963 Moon proved the following generalization of Landau's theorem.

Theorem 2.2. (Moon [22]) *A sequence* $(s_1, \ldots, s_n)$ *satisfying* $0 \le s_1 \le \le \cdots \le s_n$ *is the out-degree sequence of some* (b, b, n)*-digraph if and only if*

$$\sum_{i=1}^{k} s_i \ge b\binom{k}{2}, \ 1 \le k \le n, \tag{2.2}$$

with equality when $k = n$.

In 2009 Iványi gave the following necessary and sufficient condition for (a, b, n)-digraphs. Let the loss function L_n be defined as $L_0 = 0$ and

$$L_k = \max\left(L_{k-1}, b\binom{k}{2} - \sum_{i=1}^{k} s_i\right) \quad \text{for } 1 \le k \le n.$$

Theorem 2.3. (Iványi [12, 17]) *An* (a, b, n)*-regular nondecreasing sequence* $s = (s_1, \ldots, s_n)$ *is the out-degree sequence of some* (a, b, n)*-digraph if and only if*

$$a\binom{k}{2} \le \sum_{i=1}^{k} s_i \le b\binom{n}{2} - L_k - (n-k)s_i \quad (1 \le k \le n).$$

Proof. See [12]. ■

Landau's theorem is the special case $a = b = 1$ of Theorem 2.3, while Moon's theorem is the special case $a = b$.

The following algorithm is based on Theorem 2.3. In the programs of this paper the pseodocode conventions described in [2] are used.

Input. n: the length of s $(n \ge 1)$;
a: minimal number of the arcs between two vertices;
b: maximal number of the arcs between two vertices;
$\mathbf{s} = (s_1, \ldots, s_n)$: a nondecreasing sequence of integers.

Output. One of the following messages:
i"-th score is too small";
i"-th score is too large";
"the sequence is ("a" "b","n")-digraphical".

Working variable. i: cycle variable;
$B = (B_0, \ldots, B_n)$: the sequence of the binomial coefficients $\binom{n}{k}$ for $k = 0, \ldots, n$;
$L = (L_0, \ldots, L_n)$: the sequence of the values of the loss function;
$H = (H_0, \ldots, H_n)$: the sequence of the sums of the i smallest elements of s.

Digraph-Test(n, a, b, s)

```
01 L_0 ← 0                                     // lines 01–03: initialization
02 H_0 ← 0
03 B_0 ← 0
04 for i ← 1 to n                    // lines 04–07: computation of H_i, B_i, L_i
05       H_i ← H_{i-1} + s_i
06       B_i ← B_{i-1} + i - 1
07       L_i ← max(L_{i-1}, bB_i - S_i)
08       if H_i < aB_i                       // lines 08–09: exclusion of small s's
09          return i"-th score is too small"
10       if H_i > bB_n - L_i - s_i(n - i)     // lines 10–11: exclusion of large s'
11          return i"-th score is too large"
12 return s "is digraphical"
```

It is easy to show that the running time of Digraph-Test varies between the best $\Theta(1)$ and the worst $\Theta(n)$.

Using formula (22) of [16] we have that the number $Q(a, b, n)$ of (a, b, n)-diregular sequences is

$$Q(a, b, n) = \binom{b(n-1)+n}{n}. \tag{2.3}$$

Table 1 contains $Q(a, b, n)$ and the number $D(a, b, n)$ of (a, b, n)-digraphical sequences for $a = b = 1$ (that is for individual tennis tournaments), for $a = = b = 2$ (that is for individual chess tournaments), and for $a = 2,\ b = 3$ (that is for a *complete* [16] football tournament for $n = 1, \ldots, 11$ vertices.

n	$Q(1,1,n)$	$D(1,1,n)$	$Q(2,2,n)$	$D(2,2,n)$	$Q(2,3,n)$	$D(2,3,n)$
1	1	1	1	1	1	1
2	3	1	6	2	10	4
3	10	2	35	5	84	27
4	35	4	210	16	715	208
5	126	9	1287	59	6188	1709
6	462	22	8008	247	54264	14513
7	1716	59	50388	1111	480700	125658
8	6435	167	319770	5302	4292145	1102081
9	24310	490	2042975	26376	38567100	9756399
10	92378	1486	13123110	135670	348330136	86989413
11	352716	4639	84672315	716542	3159461968	780019710
12	1352078	14805	548354040	3868142	28760021745	7026788895
13	5200300	48107	3562467300	21265884		
14	20058300	158808	23206929840	118741369		

Table 1. The number of (a, b, n)-diregular and (a, b, n)-digraphical sequences for $a = b = 1$, $a = b = 2$, $a = 2$ and $b = 3$ and $n = 1, \ldots, 14$ vertices

Tables 2 and 3 characterize the efficiency of the rounds of DIGRAPH-TEST showing the number of the filtered and investigated sequences in the i-th round for $a = b = 1$, that is for individual tennis tournaments for $n = 1, \ldots, 14$ vertices and for the rounds $i = 1, \ldots 7$, resp. $i = 8, \ldots, 14$.

n/i	1	2	3	4	5	6	7
1	0						
2	1	1					
3	1	5	2				
4	5	15	6	5			
5	6	50	27	21	13		
6	28	174	75	73	55	35	
7	36	574	300	276	209	160	102
8	165	2112	854	950	763	637	478
9	220	7260	3312	3396	2817	2398	1961
10	1001	27390	10230	11487	10006	8994	7659
11	1365	98384	38115	41800	35277	32663	29216
12	6188	375921	125411	142296	124839	118882	108638
13	8568	1395394	467649	521885	436744	420695	398979
14	38760	5371660	1636726	1817088	1549067	1507705	1446577

Table 2. The number of the filtered not $(1, 1, n)$-digraphical sequences in the i-th round of DIGRAPH-TEST for $n = 1, \ldots, 14$ and $i = 1, \ldots, 7$

n/i	8	9	10	11	12	13	14
8	309						
9	1495	961					
10	6283	4786	3056				
11	25101	20603	15614	9939			
12	97930	83956	68564	51781	32867		
13	369968	332660	284099	231195	174209	110148	
14	1381068	1279513	1142585	972793	789234	593114	373602

Table 3. The number of the filtered not $(0, 1, n)$-digraphical sequences in the i-th round of DIGRAPH-TEST for $n = 8, \ldots, 14$ and $i = 8, \ldots, 14$

Tables 4 and 5 characterize the efficiency of the rounds of DIGRAPH-TEST showing the number of the filtered and investigated non $(2, 2, n)$-graphical sequences (that is for a chess tournaments) for $n = 1, \ldots, 14$ vertices and for $i = 1, \ldots, 7$, resp. for $i = 8, \ldots, 14$

Tables 6 and 7 contain the number of the filtered in the i-th round of not $(2, 3, n)$-graphical sequences, when DEGREE-TEST tested all $(2, 3, n)$-regular sequences for $i = 1, \ldots, 6$ resp. $i = 7, \ldots, 12$ and $n = 1, \ldots, 11$ vertices.

n/i	1	2	3	4	5	6
1	0					
2	1	3				
3	4	16	10			
4	15	83	58	38		
5	56	440	330	241	161	
6	210	2402	1825	1458	1119	747
7	792	13538	10194	8498	7125	5480
8	3003	78696	57078	48872	43461	36597
9	11440	470184	325920	277644	258475	231593
10	43758	2874080	1891989	1585782	1506392	1418825
11	167960	17889443	11232210	9100652	8715762	8482480

Table 4. The number of the filtered not $(2, 2, n)$-digraphical sequences in the i-th round of Digraph-Test for $n = 1, \ldots, 11$ vertices and $i = 1, \ldots, 6$

n/i	7	8	9	10	11
7	3650				
8	28160	18601			
9	194715	148944	97684		
10	1272721	1061218	807032	525643	
11	8011380	7120660	5894122	4456457	2884647

Table 5. The number of the filtered not $(2, 2, n)$-digraphical sequences in the i-th round of Digraph-Test $n = 7, \ldots, 11$ vertices and $i = 7, \ldots, 11$

n/i	1	2	3	4	5	6
1	0					
2	3	3				
3	10	31	16			
4	70	205	150	82		
5	252	1533	1235	957	502	
6	1716	11082	9088	7930	6555	3380
7	6435	84865	69441	64368	57655	47811
8	43758	671099	507199	494226	486820	436009
9	167960	5488821	3931096	3751501	3890421	3828202
10	1144066	46495034	30199434	28218140	30349772	31590048
11	4457400	401403728	244025820	214372994	232279669	253892909
12	30421755	3543412391	1995894197	1645568584	1765504146	1988106381

Table 6. The number of the filtered not $(2, 3, n)$-digraphical sequences in the i-th $(i = 1, \ldots, 6)$ round of Digraph-Test for $n = 1, \ldots, 11$ vertices

The values $Q(a, b, n)$ are computed using (2.3), the values of $D(1, 1, n)$ in Table 1 are taken from [25], while the values of Tables 4, 5, 6 and 7 were determined by Digraph-Test-Enumerative (the enumerative version of Digraphh-Test).

n/i	7	8	9	10	11	12
7	24467					
8	365510	185443				
9	3409023	2887763	1455914			
10	30871440	27322172	23404704	11745913		
11	262074711	253295635	223318920	193530773	96789699	
12	2164167153	2200747000	2107880874	1854248627	1626229074	811052668

Table 7. The number of the filtered not $(2, 3, n)$-digraphical sequences in the i-th ($i = 7, \ldots, 12$) round of Digraph-Test for $n = 7, \ \ldots, \ 11$ vertices

3. Conditions and algorithms for $(0, 1, n)$-undigraphs

Our aim is to find quick algorithms which decide whether a given regular sequence is graphical or not. The classical algorithms are based on the theorems Havel [10] and Hakimi [9], resp. Erdős and Gallai [5]. In worst case the running time of these algorithms is $\Theta(n^2)$. It is worth to remark that Erdős-Gallai algorithm only tests the input sequences while the Havel-Hakimi algorithm produces also a corresponding graph (if the input sequence is graphical). Tripathi, Vijay and West [28] gave a constructive proof of Erdős-Gallai theorem in 2010.

In 2011 in the paper [16] we presented quicker algorithms HHZ (zerofree Havel-Hakimi), HHP (parity checking Havel-Hakimi), HHQ (quick Havel-Hakimi), EGS (shortened Erdős-Gallai), EGL (linear Erdős-Gallai), and EGJ (jumping Erdős-Gallai). Takahashi in 2007 [27], Hell and Kirkpatrick in 2009 [11] published linear version of Erdős-Gallai algorithm. Recently Király [19] presented an $O(n \log\log n)$ version of Havel-Hakimi algorithm.

In this section we present the classical Havel-Hakimi and Erdős-Gallai algorithms, further HHL, the linear version of the Havel-Hakimi algorithm.

We remark that the testing of $(0, 1, n)$-regular sequences is an important subproblem when we try to answer the question on the complexity of the testing of potential football sequences (see [7, Research problem 2.3.1] and [14]).

3.1. Havel-Hakimi algorithm (HH)

If $n = 1$, then there exists one $(0, 1, n)$-graphical sequence: (0). If $n \geq 2$, then the following Havel-Hakimi theorem gives a necessary and sufficient condition.

Theorem 3.1. (Havel, Hakimi [9, 10]) *Let $n \geq 2$. An n-regular sequence $s = (s_1, \ldots, s_n)$ is graphical if and only if the sequence $s' = (s_2 - 1,$*

$s_3 - 1, \ldots, s_{s_1} - 1, s_{s_1+1} - 1, s_{s_1+2}, \ldots, s_{n-1}, s_n)$ *sequence is* $(n-1)$*-graphical.*

Proof. See [9, 10]. ■

The algorithm Havel-Hakimi is based on Theorem 3.1. In this and the following algorithms L is a logical variable: if the investigated sequence is graphical, then $L = 1$, otherwise $L = 0$.

Input. n: the length of the sequence s ($n \geq 2$);
$s = (s_1, \ldots, s_n)$: the investigated n-regular sequence.

Output. L: logical variable.

Working variable. i: cycle variables.

```
Havel-Hakimi(n, s, L)
01 for i = 1 to n − 1                    // line 01–06: test of the elements of s
02     if s_{s_i+i} == 0                  // lines 01–02: s is not undigraphical
03        return 0
04     for j = i + 1 to s_i + i
05         s_j = s_j − 1
06     sort (s_{i+1}, ..., s_n) in decreasing order
07 L = 1                                  // lines 08–09: s is undigraphical
08 return 1
```

3.2. Erdős-Gallai algorithm (EG)

Let the elements $s_1, \ldots, s_n$ of the sequence s called *the head* of s belonging to s_i, and let the remaining elements called *the tail* of s belonging to s_i.

Paul Erdős and Tibor Gallai in 1960 published the following necessary and sufficient condition.

Theorem 3.2. (Erdős, Gallai [5]) *Let* $n \geq 1$*. An* $s = (s_1, \ldots, s_n)$ $(0, 1, n)$*-regular sequence is* $(0, 1, n)$*-graphical if and only if*

$$\sum_{i=1}^{n} s_i \quad \text{is even} \tag{3.1}$$

and

$$\sum_{i=1}^{j} s_i - j(j-1) \leq \sum_{k=j+1}^{n} \min(j, s_k) \quad (j = 1, \ldots, n-1). \tag{3.2}$$

Proof. See [4, 5, 28]. ■

The following algorithm is based on Theorem [5].

Input. n: the length of s;
$s = (s_1, \dots, s_n)$: the investigated n-regular sequence.

Output. L: logical variable.

Working variable. i: cycle variable;
R: estimated capacity of the actual tail.

Erdős-Gallai(n, s, L)

```
01 H_1 = s_1                          // line 01: computing of H_1
02 for i = 2 to n                     // lines 02–03: computing of the further H_i's
03      H_i = H_{i-1} + s_i
04 if H_n is odd                      // lines 04–05: test of the parity
05    return 0
06 for i = 1 to n - 1                 // line 07–15: test of s
07            R = 0                   // line 08: initialization
08            for k = j + 1 to n      // lines 09–10: tail capacity
09                R = R + min(j, s_k)
10            if H_j - j(j - 1) > R   // line 10: test of s
11               return 0             // line 11: s is not graphical
12 return 1                           // line 12: s is graphical
```

Table 8 contains the number of (a, b, n)-undiregular and (a, b, n)-undigraphical sequences for $a = 0$ and $b = 1$, $a = 0$ and $b = 2$, $a = 2$ and $b = 5$ and $n = 1, \dots, 11$.

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

Table 8. The number of (a, b, n)-undiregular and (a, b, n)-undigraphical sequences for $a = 0$ and $b = 1$, $a = 0$ and $b = 2$, $a = 2$ and $b = 3$ and for $n = 1, \dots, 11$ vertices

Table 9 presents the number of the filtered not $(0, 1, n)$-graphical sequences in the i-th round of HHT for $n = 1, \dots, 11$ vertices.

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

Table 9. The number of the filtered non $(0, 1, n)$-graphical sequences in the i-th round of HH for $n = 1, \ldots, 11$ vertices and $i = 1, \ldots, 10$

Table 10 presents the number of the filtered graphical sequences in the i-th round of HHT for $a = 0$, $b = 1$, $n = 1, \ldots, 11$ vertices and for $i = 1, \ldots, 11$.

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

Table 10. The number of the filtered $(0, 1, n)$-graphical sequences in the i-th round of HH for $n = 1, \ldots, 11$ vertices and $i = 1, \ldots, 10$

Let $n_i(a, b, n, A)$, resp. $m_i(a, b, n, A)$ denote the number of not (a, b, n)-graphical, resp. (a, b, n)-graphical sequences filtered by algorithm A in the ith round of the testing of all (a, b, n)-regular sequences, further let

$$N = \sum_{i=1}^{n-1} n_i \quad \text{and} \quad M = \sum_{i=1}^{n-1} m_i, \tag{3.3}$$

$$X(a, b, n, A) = \frac{\sum_{i=1}^{n-1} i n_i}{N}, \tag{3.4}$$

$$Y(a, b, n, A) = \frac{\sum_{i=1}^{n-1} i m_i}{M}, \tag{3.5}$$

$$Z(a, b, n, A) = \frac{\sum_{i=1}^{n-1} i(m_i + n_i)}{N + M}, \tag{3.6}$$

$$X'(a, b, n, A) = \frac{\sum_{i=1}^{n-1} in_i}{N(n-1)}, \tag{3.7}$$

$$Y'(a, b, n, A) = \frac{\sum_{i=1}^{n-1} im_i}{M(n-1)}, \tag{3.8}$$

$$Z'(a, b, n, A) = \frac{\sum_{i=1}^{n-1} i(m_i + n_i)}{(N + M)(n-1)}. \tag{3.9}$$

These efficiency measures characterize the average number of the filtered not graphical, graphical, resp. all sequences during the run of algorithm A: X, Y, and Z for a sequence, while X', Y', and Z' for an element of the input sequences.

Table 11 characterizes the efficiency of algorithm HHL during the testing of $(0, 1, n)$-regular sequences for $n = 1, \ldots, 11$ vertices[1]. In line 11 of Table 11 we see $X'(0, 1, 11) = 0.136887459$ and $Y'(0, 1, 11) = 0.615705668$. According to these data in the case of 11 vertices the filtering of *all* nongraphical sequences needs in average 14 % of the rounds, while the filtering of the graphical sequences requires 62 % of the rounds implying that the complete filtering requires in average 22 % of the rounds.

n^*	X	Y	Z	X'	Y'	Z'
2	1.00000000	1.00000000	1.00000000	1.00000000	1.00000000	1.00000000
3	1.00000000	1.75000000	1.30000000	0.50000000	0.87500000	0.65000000
4	1.08333333	2.45454545	1.51428571	0.36111111	0.81818181	0.50476190
5	1.12631578	3.03225806	1.59523809	0.28157894	0.75806451	0.39880952
6	1.18055555	3.58823529	1.71212121	0.23611111	0.71764705	0.34242424
7	1.22052401	4.11111111	1.79662004	0.20342067	0.68518518	0.29943667
8	1.26273458	4.62984336	1.89743589	0.18039065	0.66140619	0.27106227
9	1.29906261	5.14079339	1.98823529	0.16238282	0.64259917	0.24852941
10	1.33532385	5.65016233	2.08340730	0.14836931	0.62779581	0.23148970
11	1.36887458	6.15705668	2.17453418	0.13688745	0.61570566	0.21745341

Table 11. Efficiency of HH for the testing of all $(0, 1, n)$-regular sequences for $n = 2, \ldots, 11$ vertices

[1] $n^* = \frac{n}{\text{measure}}$

3.3. Havel-Hakimi linear testing algorithm (HHL)

In the worst case the original Havel-Hakimi algorithm requires quadratic time to test the $(0,1,n)$-regular sequences. Using the new concepts weight point and reserve we reduced the worst running time to $O(n)$.

The definition of the *weight point* w_i belonging to s_i was introduced in [16] in connection with Erdős-Gallai-Linear and it is as follows. If $s_1 \geq i$, then w_i is the largest k $(1 \leq k \leq n)$ having the property $s_k \geq i$. But if $s_1 < i$, then $w_i = 0$. EGL exploits the property w_i ensuring that if $i \leq w_i$, then the key expression $\min j, s_k$ in the Erdős-Gallai theorem equals i, otherwise equals s_k.

Here we extend the definition to be applicable also in the proof of the linearity of Chungphaisan-Erdős-Gallai. Now let w_i the largest k $(1 \leq \leq k \leq n)$ having the property $s_k \geq bi$. But if $s_1 < bi$, then let $w_i = 0$. In the case $b = 1$ the new definition coincides with the old one.

In HHL the weight point w_i determines the increment of the tail capacity when we switch to the investigation of the next element of s.

The reserve r_i belonging to s_i is defined as the unused part of the actual tail capacity and can be computed by the formulas

$$r_i = w_1 - 1 - s_1 \tag{3.10}$$

and

$$r_i = w_i - r_{i-1} - s_i \quad \text{for} \quad 1 \leq i \leq n-1. \tag{3.11}$$

The programs of this paper are written using the pseudocode descibed in [2].

Input. n: number of vertices $(n \geq 1)$;
$s = (s_1, \ldots, s_n)$: the investigated n-regular sequence.

Output. L: logical variable.

Work variable. i: cycle variable;
$r = (r_1, \ldots, r_n)$: r_i the reserve belonging to s_i;
$w = (w_1, \ldots, w_n)$: w_i the weight point belonging to s_i;
$H = (H_1, \ldots, H_n)$: H_i is the sum of the first i elements of s.

```
Havel-Hakimi-Linear(n, s, L)
01 if s_1 == 0          // lines 01–02: test of the sequence consisting of only zeros
02     return 1
03 if s_{s_1+1} == 0                        // lines 03–04: test of s_1 in constant time
04     return 0
05 H_1 = s_1                                        // line 05: initialization of H
```

```
06 for i = 2 to n                                    // lines 06–07: further H_i's
07      H_i = H_{i-1} + s_i
08 if H_n is odd                                     // lines 08–09: test of the parity
09    return L
10 w_1 = n    // lines 10–13: computation of the first weight point and reserve
11 while s_{w_1} < 1
12          w_1 = w_1 - 1
13 r_1 = w_1 - 1 - s_1
14 for i = 2 to n - 1                                // lines 14–21: testing of s
15      if s_i ≤ i or s_{i+1} = 0
16         return 1
17      w_i = w_{i-1}
18      while s_{w_i} < i and w_i > 0
19              w_i = w_i - 1
20      if s_i > w_i - 1 + r_{i-1}                   // line 20: Is s graphical?
21         return 0                                  // line 21: s is not graphical
22      r_i = w_i + r_{i-1} - s_i                    // line 22: update of the reserve
23 return 1                                          // line 23: s is graphical
```

Theorem 3.3. *The running time of* Havel-Hakimi-Linear *is in best case* $\Theta(1)$*, and in worst case* $\Theta(n)$*.*

Proof. If the condition in line 2 holds, then the running time is $\Theta(1)$. If not, then we decrease the actual w at most n times and the remaining operations require $O(1)$ operations for all reductions. ■

Now let us consider a few examples.

Example 1. Let our first example be $s = (3, 3, 3, 1)$. According to lines 01–15 $r_1 = 0$. For $i = 2$ we get $w_i = 3$ and the condition of line 22 is not satisfied, therefore s is *not* $(0, 1, 4)$-*undigraphical.*

Example 2. Let our next example be $s = (5, 3, 3, 2, 1, 1, 1)$. In lines 01–15 we get $w_1 = 7$ and $r_1 = 1$. For $i = 2$ according to lines $w_i = 3$, the condition of line 22 does not hold and according to line 25 $r_2 = 1$. When $i = 3$, then $s_i \geq i$ and so according to line 16 s is $(0, 1, 7)$-*undigraphical.*

Example 3. Now let $s = (5, 4, 1, 1, 1, 1, 1)$. At first get $r_1 = 1$, then for $i = 2$ we have $w_i = 2$, therefore the conditions in line 22 holds, so s is *not* $(0, 1, 7)$-*undigraphical.*

Example 4. Let our last example be $s = (5, 5, 4, 3, 3, 3, 3)$. According to the first 15 lines $r_1 = 1$. When $i = 2$, then we get $w_i = 7$ and $r_2 = 2$. Then $w_3 = 7$ and $r_3 = 4$. If $i = 4$, then according to $i \geq s_i$ in line 16 s *is* $(0, 1, 7)$-*undigraphical.*

4. Degree sequences of $(0, b, n)$-graphs

In this section we use the theorem due to Chungphaisan to get a linear time algorithm for the testing of $(0, b, n)$-regular sequences.

4.1. Theorem of Chungphaisan and ChEGl algorithm

In 1974 Chungphaisan extended Erdős-Gallai theorem for $(0, b, n)$-undigraphs, proving the following assertion.

Theorem 4.1. (Chungphaisan [3]) *Let $n \geq 1$. An $s = (s_1, \ldots, s_n)$ $(0, b, n)$-regular sequence is $(0, b, n)$-graphical if and only if*

$$\sum_{i=1}^{n} s_i \quad \textit{is even} \tag{4.1}$$

and

$$\sum_{i=1}^{j} s_i - bj(j-1) \leq \sum_{k=j+1}^{n} \min(jb, s_k) \quad (j = 1, \ldots, n-1). \tag{4.2}$$

Proof. See [3]. ■

In the worst case the algorithm based on this theorem requires quadratic time, but the following assertion allows us to test the sequences in linear time.

Theorem 4.2. *If $n \geq 1$, then an $s = (s_1, \ldots, s_n)$ $(0, b, n)$-regular sequence is $(0, b, n)$-graphical if and only if*

$$\sum_{i=1}^{n} s_i \quad \textit{is even} \tag{4.3}$$

and

$$H_i > bi(y_i - 1) + H_n - H_y \quad (i = 1, \ldots, n-1), \tag{4.4}$$

where

$$y_i = \max(i, w_i) \quad (i = 1, \ldots, n-1). \tag{4.5}$$

Proof. This proof is an improved version of the proof of linearity of EGL in [15].

We exploit that s is monotone and determine the capacity estimations $c_k = \min(jb, s_k)$ appearing in (4.2) in constant time. The base of the quick computation is again the sequence of the weight points $w(s) = (w_1, \ldots, w_{n-1})$ containing the *weight points* belonging to of the elements of s, and the sequence $y(s) = (y_1, \ldots, y_n)$ containing the *cutting points* of the elements of s. For given s_i let the weight point w_i was defined in Section 3. The cutting point y_i be the maximum of i and w_i, see (4.5).

During the testing of the elements of s there are two cases:

a) if $i > w_i$, then the maximal contribution $C_i = \sum_{k=i+1}^{n} \min(i, s_k)$ of the actual tail of s is at most $H_n - H_i$, since the maximal contribution $c_k = \min(i, s_k)$ of the element s_k is only s_k, and so

(4.6) $$C_i = \sum_{k=i+1}^{n} c_k = H_n - H_i,$$

implying the requirement

(4.7) $$H_i \le bi(i-1) + H_n - H_i;$$

b) if $i \le w_i$, then the maximal contribution C_i of the actual tail of s consists of contributions of two types: $c_{i+1}, \ldots, c_{w_i}$ are equal to bi, while $c_j = s_j$ for $j = w_i + 1, \ldots, n$, therefore we have

(4.8) $$C_i = bi(w_i - i) + H_n - H_{w_i},$$

implying the requirement

(4.9) $$H_i = bi(i-1) + bi(w_i - i) + H_n - H_{w_i}.$$

Transforming (4.9) we get

(4.10) $$H_i = bi(w_i - 1) + H_n - H_{w_i}.$$

Considering the definition of y_i given in (4.5), further (4.7) and (4.9) we get the required (4.4). ■

The following algorithm tests the potential degree sequences of $(0, b, n)$-undigraphs.

Input. n: number of vertices ($n \ge 1$);
$s = (s_1, \ldots, s_n)$: a $(0, b, n)$-regular sequence; b: the maximal permitted number of arcs between two vertices.

Output. L: logical variable.

Work variable. i: cycle variable;
$r = (r_1, \ldots, r_n)$: r_i is the reserve belonging to s_i;
$w = (w_1, \ldots, w_n)$: w_i is the weightpoint belonging to s_i.

Chungphaisan-Erdős-Gallai-Linear(n, s, b, L)

```
01 H_1 = s_1                                    // line 01: initialization of H_1
02 for i = 2 to n − 1          // line 02–03: computation of the elements of H
03       H_i = H_{i−1} + s_i
04 if H_n is odd                                  // line 04–05: test of the parity
05    return 0
06 w = n                          // lines 06: initialization of the first weight point
07 for i = 1 to n − 1                                   // lines 07–13: test of s
08       while s_w < ib and w > 0
09             w = w − 1
10       y = max(i, w)
11       if H_i > bi(y − 1) + H_n − H_y
12          return 0
13 return 1                                              // line 14: acceptance of s
```

Theorem 4.3. *The running time of* Chungphaisan-Erdős-Gallai-Linear *is* $\Theta(n)$ *in all cases.*

Proof. Lines 01–06 require $\Theta(n)$ time. Since the value of w is strictly decreasing, lines 07–14 require $O(n)$ time, therefore the running time is $\Theta(n)$ in all cases. ∎

Let us consider two examples. Let $b = 3$ and $s' = (13, 10, 5, 5, 4, 1)$. $H_6 = 38$ is even. If $i = 1$, then $w_i = y = 5$ and the condition in line 18 is not satisfied $(13 \leq 3 \cdot 1 \cdot (5-1))$. If $i = 2$, then $w_i = y = 2$ and the condition in line 18 holds $(23 > 3 \cdot 2 \cdot (2-1)) + 5 + 5 + 4 + 1$, therefore s is *not* $(0, 3, 6)$-*graphical.*

Let b remain 3, but change s to $s' = (13, 10, 5, 5, 4, 3)$. The first difference comparing with the previous example comes when $i = 2$. Now $23 \leq 3 \cdot 2 \cdot \cdot(2-1)) + 5 + 5 + 4 + 3$, and the condition in line 18 holds for $i = 3$, 4 and 5 too, therefore s' is $(0, 3, 6)$-graphical.

Table 12 contains the number of the not $(0, 2, n)$-undigraphical sequences excluded in the i-th round $(i = 1, \ldots, 10)$ for $n = 1, \ldots, 11$ vertices.

n/i	1	2	3	4	5	6	7	8	9	10
1	0									
2	3	0								
3	22	3	0							
4	132	26	2	0						
5	824	164	31	4	0					
6	5084	1026	276	75	3	0				
7	31902	6288	2018	829	111	50				
8	201366	39090	13282	7231	1837	203	4	0		
9	1281918	244833	84340	53594	20681	4259	298	6	0	
10	8207232	1548774	529578	365461	183262	59726	8709	470	5	0
11	52819163	9866545	3331910	2385963	1404590	632058	155070	17213	660	7

Table 12. The number of the excluded not $(0, 2, n)$-undigraphical sequences in the ith round $(i = 1, \ldots, 10)$ by ChEGL for $n = 1, \ldots, 11$ vertices

Table 13 contains the number of the $(0, 2, n)$-graphical sequences excluded in the ith $(i = 1, \dots, n)$ round for $n = 1, \dots, 11$ vertices.

n/i	1	2	3	4	5	6	7	8	9	10
1	1									
2	2	0								
3	1	9	0							
4	1	7	42	0						
5	1	10	29	224	0					
6	1	14	49	183	1297	0				
7	1	18	70	345	1143	7658	0			
8	1	23	97	559	2326	7262	46489	0		
9	1	28	125	846	4038	15927	46074	286007	0	
10	1	34	159	1191	6520	29629	107724	295609	1779026	0
11	1	40	193	1624	9668	50663	213399	728610	1900061	11154877

Table 13. The number of the filtered $(0, 2, n)$-undigraphical sequences in the ith $(i = 1, \dots, 10)$ round of ChEGL for $n = 1, \dots, 11$ vertices

Table 14 characterizes the efficiency of algorithm ChEGL for the testing of $(0, 2, n)$-regular sequences and $n = 1, \dots, 11$ vertices[2].

n^*	X	Y	Z	X'	Y'	Z'
2	1.000000000	1.000000000	1.000000000	1.000000000	1.000000000	1.000000000
3	1.120000000	1.900000000	1.342857143	0.560000000	0.950000000	0.671428571
4	1.187500000	2.820000000	1.576190476	0.395833333	0.940000000	0.525396825
5	1.232649071	3.803030303	1.759906760	0.308162268	0.950757576	0.439976690
6	1.280785891	4.788212435	1.957042957	0.256157178	0.957642487	0.391408591
7	1.322698224	5.770438549	2.137870128	0.220449704	0.961739758	0.356311688
8	1.363989613	6.751572493	2.320248929	0.194855659	0.964510356	0.331464133
9	1.402468979	7.733105601	2.496464714	0.175308622	0.966638200	0.312058089
10	1.439464334	8.714770487	2.670148311	0.159940482	0.968307832	0.296683146
11	1.474743645	9.697001722	2.839981439	0.147474365	0.969700172	0.283998144

Table 14. The efficiency of ChEGL during the testing of $(0, 2, n)$-regular sequences for $n = 1, \dots, 11$ vertices

5. Degree sequences of (a, b, n)-undigraphs

Theorem 4.1 due to Chungphaisan has the following straightforward consequence.

Corollary 5.1. *Let $n \geq 2$. An $s = (s_1, \dots, s_n)$ (a, b, n)-undiregular sequence is (a, b, n)-undigraphical if and only if the sequence $s' = (s_1 - a(n-1),$ $, \dots, s_n - a(n-1))$ is $(0, b-a, n)$-undigraphical.*

[2] $n^* = \frac{n}{\text{measure}}$

Proof. In an (a, b, n)-undigraph the elements of every pair of vertices are connected with at least a arcs. Therefore if we remove a arcs, then we get a $(0, b - a, n)$-undigraph. ■

Using Corollary 5.1 it is easy to test an (a, b, n)-regular sequence: we use ChEG with input sequence $s' = (s_1 - a(n-1), \ldots, s_n - a(n-1))$.

6. Summary

The paper contains an overview on the known algorithms of testing of potential degree sequences of (a, b, n)-graphs. The known methods for (a, b, n)-digraphs in worst case require only linear time but for (a, b, n)-undigraphs in the worst case at least quadratic time. We proposed new linear time algorithms for $(0, b, n)$-undigraphs which can be applied for (a, b, n)-undigraphs too.

Acknowledgement. The author thanks Péter Burcsi, senior lecturer for his useful remarks on the manuscript, Zoltán Király for recommendation of interesting references and Loránd Lucz, MSc student for the computer experiments, all of Eötvös Loránd University.

References

[1] **Burns, J. M.,** The number of degree sequences of graphs, PhD Dissertation, MIT, 2007.

[2] **Cormen, T. H., Ch. E. Leiserson, R. L. Rivest and C. Stein,** *Introduction to Algorithms*, Third edition, The MIT Press/McGraw Hill, Cambridge/New York, 2009.

[3] **Chungphaisan, V.,** Conditions for sequences to be r-graphical, *Discrete Math.* **7** (1974), 31–39.

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

[5] **Erdős, P. and T. Gallai,** Graphs with prescribed degrees of vertices, (Hungarian) *Mat. Lapok* **11** (1960), 264–274.

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

[7] **Frank, A.,** *Connections in Combinatorial Optimization*, Oxford University Press, Oxford, 2011.

[8] **Gross, J. L. and J. Yellen** (Eds.), *Handbook of Graph Theory*, CRC Press, Boca Raton, FL, 1994.

[9] **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.

[10] **Havel, V.,** A remark on the existence of finite graphs, (Czech) *Časopis Pěst. Mat.* **80** (1955), 477–480.

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

[12] **Iványi, A.,** Reconstruction of complete interval tournaments, *Acta Univ. Sapientiae, Inform.* **1(1)** (2009), 71–88.

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

[14] **Iványi, A.,** Deciding football sequences, *Acta Univ. Sapientiae, Inform.* **4(1)** (2012), 127–164.

[15] **Iványi, A. and L. Lucz,** Erdős-Gallai test in linear time, *Combinatorica* (submitted).

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

[17] **Iványi, A. and S. Pirzada,** *Comparison based ranking,* in A. Iványi (Ed.) *Algorithms of Informatics, Vol. 3* (electronic book). AnTonCom, Budapest 2011, 1262–1311.

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

[19] **Király, Z.,** *Recognizing Graphic Sequences and Generating all Realizations*, Technical Report of Egerváry Research Group, TR-2011, Budapest. Last modification 2 May, 2012. `http://www.cs.elte.hu/egres`

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

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

[22] **Moon, J. W.,** An extension of Landau's theorem, *Pacific J. Math.* **13** (1963), 1343–1345.

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

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

[25] **Pécsy, G. and L. Szűcs,** Parallel verification and enumeration of tournaments, *Stud. Univ. Babeş-Bolyai, Inform.* **45(2)** (2000), 11–26.

[26] **Sloane N. J. A.,** The number of degree-vectors for simple graphs, in: N. J. A. Sloane (Ed.) *The On-Line Encyclopedia of Integer Sequences,* 2011. `http://oeis.org/Seis.html`

[27] **Takahashi, M.,** *Optimization Methods for Graphical Degree Sequence Problems and their Extensions*, PhD thesis, Graduate School of Information, Production and Systems, Waseda University, Tokyo, 2007. `http://hdl.handle.net/2065/28387`

[28] **Tripathi, A., S. Venugopalan and D. B. West,** A short constructive proof of the Erdős-Gallai characterization of graphic lists, *Discrete Math.* **310(4)** (2010), 833–834.

[29] **Tripathi, A. and S. Vijay,** A note on a theorem of Erdős & Gallai, *Discrete Math.* **265(1–3)** (2003), 417–420.

[30] **Weisstein, E. W.,** *Degree sequence,* From MathWorld—Wolfram Web Resource, 2012. `http://mathworld.wolfram.com/DegreeSequence.html`

[31] **Weisstein, E. W.,** *Graphic sequence,* From MathWorld—Wolfram Web Resource, 2012. `http://mathworld.wolfram.com/GraphicSequence.html`

A. Iványi
Department of Computer Algebra
Eötvös Loránd University
H-1117 Budapest, Pázmány P. sétány 1/C
Hungary
`tony@compalg.inf.elte.hu`