COMBINATORICA ?? (20??) ???-???

ERDŐS-GALLAI TEST IN LINEAR TIME

ANTAL IVÁNYI, LORÁND LUCZ

Abstract. Havel in 1955 [6], Erdős and Gallai in 1960 [3], Hakimi in 1962 [5], Tripathi et al. in 2010 [15] proposed an algorithm to decide, whether a sequence of nonnegative integers is the degree sequence of a simple graph. The running time of their algorithms in worst case is $\Omega(n^2)$. In this paper we propose a new algorithm called Erdős-Gallai-Linear, whose worst running time is $\Theta(n)$. As an application of the new algorithm we determined the number of the degree sequences of simple graphs for $n = 24, \ \ldots, \ 29$ vertices [14].

1. Introduction

A classical problem of graph theory is the characterization of the degree sequences of different graph classes. In 1953 Landau [11] gave a necessary and sufficient condition, allowing the linear time test of score sequences of tournaments. Landau's theorem recently was extended for (a, b)-tournaments [7, 8] containing at least a and at most b edges between any two vertices. Havel [6] and Hakimi [5] proposed the same algorithm allowing quadratic time test and reconstruction of simple graphs.

Key words and phrases: degree sequence, Erdős-Gallai theorem, Havel-Hakimi theorem, simple graph

2010 Mathematics Subject Classification: 05C65

1998 CR Categories and Descriptors: G.2.2

The authors received support from The European Union and the European Social Fund under the grant agreement no. TÁMOP 4.2.1/B-09/1/KMR-2010-0003.

Erdős and Gallai published in 1960 [3] a necessary and sufficient condition, allowing to test the degree sequences of simple graphs in quadratic time. Recently Tripathi, Venugopalanb and West [15] gave a constructive proof of Erdős-Gallai theorem. Chungphaisan in 1974 [1] and Özkan in 2011 [13] extended Erdős-Gallai and Havel-Hakimi theorems to loopless multigraphs containing at most b edges between any two vertices.

Since in worst case the testing algorithms of simple graphs have to investigate all elements of the input sequence, therefore every testing algorithm based on the RAM model of computations requires at least $\Omega(n)$ time. But all the known testing algorithms require $\Omega(n^2)$ time.

Our aim was to decrease the gap between the trivial lower bound and the time complexity of the known algorithms. Although this problem is interesting in itself, for us the main motivation was our wish to answer the question formulated in the recent monograph [4, Research problem 2.3.1] of András Frank: "*Decide if a sequence of n integers can be the final score of a football tournament of n teams.*" During testing and reconstructing of potential football sequences important subproblem is the handling of sequences of draws. And the problems "Is this sequence graphical?" and "Is this sequence a football draw sequence?" are equivalent, therefore the quick answer is vital for us.

Let n be a positive integer. A sequence of integer numbers $b = (b_1, \ldots, b_n)$ is called *n-bounded*, if $0 \leq b_i \leq n-1$ for $i = 1, \ldots, n$. An *n-bounded* sequence b is called *n-regular,* if $b_n \geq b_{n-1} \geq \cdots \geq b_1$. An n-regular sequence is called *n-even*, if the sum of its elements is even. An n-regular sequence b is called *n-zerofree*, if $b_n > 0$. An *n-regular* sequence b is called *n-graphical,* if there exists a simple graph G whose degree sequence is b. Instead of n-bounded, n-regular, n-even, n-zerofree and n-graphical we use the simpler bounded, regular, even, zerofree and graphical expression too.

At first we tried to use different linear time filtering algorithms hoping to get quickly the graphical/nongraphical quality of many investigated sequences and so to decrease the expected running time of the testing. In [10] we reported on some successes of this approach. While we were seeking linear approximating testing algorithms, we have found an improvement of Erdős-Gallai theorem allowing to test regular sequences in $O(n)$ time even in the worst case.

As a side product we extended OEIS [14] with the continuation of some sequences. The most interesting extensions are the new elements of degree sequences of simple graphs.

The aim of this paper is to give a short description of the new algorithm.

The paper consists of four parts. After the introductory Section 1, in Section 2 we describe the classical algorithms of the testing of degree sequences of simple graphs. Section 3 contains the description of the new linear time testing algorithm (EGL), while in Section 4 we briefly describe the enumerative (EGE) and parallel (EGP) versions of EGL and the results of the corresponding program.

Approximate algorithms, further precise algorithms and detailed simulation results can be found in [10], while the parallel results are presented in [9].

2. Classical precise algorithms

If given a decreasing sequence $b = (b_1, \ldots, b_n)$ of nonnegative integers then the first i elements of the sequence are called *the head* of the sequence belonging to the index i, while the last $n - i$ elements of the sequence *the tail* of the sequence belonging to the index i.

2.1. Havel-Hakimi algorithm (HH)

The first solution of the testing of the degree sequences of simple graphs was given by Václav Havel Czech mathematician in 1955 [6, 12]. Louis Hakimi published the same method in 1962 [5]. Their theorem and algorithm today are called usually *Havel-Hakimi theorem*, resp. *Havel-Hakimi algorithm.*

Theorem 2.1. (Havel [6], Hakimi [5]) *If $n \geq 3$, then the n-regular sequence $b = (b_1, \ldots, b_n)$ is n-graphical if and only if the sequence $b' = (b_2 - 1, b_3 - 1, \ldots, b_{b_1} - 1, b_{b_1+1} - 1, b_{b_1+2}, \ldots, b_n)$ is $(n-1)$-graphical.* ■

Proof. See [5, 6].

If we write a recursive RAM algorithm based on this theorem then its worst running time will be $\Omega(n^2)$, since the sum of the degrees is $\Theta(n^2)$ in a complete graph. It is worth to remark, that the proof of the theorem is constructive, therefore the algorithm based on this theorem not only tests the input sequence, but if the sequence is graphical, then the algorithm constructs a corresponding simple graph.

2.2. Erdős-Gallai algorithm (EG)

In chronological order the next result was the following necessary and sufficient condition due to Paul Erdős and Tibor Gallai [3].

Theorem 2.2. (Erdős, Gallai [3]) *Let $n \geq 2$. The n-regular sequence $b = (b_1, \ldots, b_n)$ is n-graphical if and only if*

$$\sum_{i=1}^{n} b_i \quad \textit{even} \tag{2.1}$$

and

$$\sum_{i=1}^{j} b_i - j(j-1) \leq \sum_{k=j+1}^{n} \min(j, b_k) \quad (j = 1, \ldots, n-1). \tag{2.2}$$

Proof. See [3, 12, 15]. ∎

Although the algorithm based on this theorem never produces a corresponding graph, in worst case the systematic application of (2.2) requires $\Theta(n^2)$ time. Recently Tripathi, Venogupalanb and West [15] published a constructive proof of Erdős-Gallai theorem, but the worst running time of their algorithm is $\Theta(n^3)$.

3. Linear time Erdős-Gallai algorithm (EGL)

The new algorithm Erdős-Gallai-Linear exploits, that b is monotone. It determines the capacities $c_k = \min(i, b_k)$ in constant time. The base of the quick computation is the sequence $w(b) = (w_1, \ldots, w_{n-1})$

containing the *weight points* of the elements of b, and the sequence $y(b) = (y_1, \ldots, y_n)$ containing the *cutting points* of the elements of b. For given sequence b let the weight point w_i the index of b_k having the maximal index among such elements of b which are greater or equal to i, that is

$$w_i(b) = w_i = \max(k \mid b_k \geq i), \tag{3.1}$$

and let the cutting point y_i be the maximum of i and w_i, that is

$$y_i(b) = y_i = \max(i, w_i) \quad (i = 1, \ldots, n). \tag{3.2}$$

Theorem 3.1. (Iványi, Lucz, Móri, Sótér [10]) *If $n \geq 1$, then the n-regular sequence $(b_1, \ldots, b_n)$ is n-graphical if and only if*

$$H_n \quad \textit{is even} \tag{3.3}$$

and

$$H_i \leq i(y_i - 1) + H_n - H_{y_i} \quad (i = 1, \ldots, n). \tag{3.4}$$

Proof. (3.3) is the same as (2.1).

During the testing of the elements of b by Erdős-Gallai-Linear there are two cases:

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

  $$C_i = \sum_{k=i+1}^{n} c_k = H_n - H_i, \tag{3.5}$$

  implying the requirement

  $$H_i \leq i(i-1) + H_n - H_i; \tag{3.6}$$

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

  $$C_i = i(w_i - i) + H_n - H_{w_i}, \tag{3.7}$$

  implying the requirement

  $$H_i = i(i-1) + i(w_i - i) + H_n - H_{w_i}. \tag{3.8}$$

Transforming (3.8) we get

$$H_i = i(w_i - 1) + H_n - H_{w_i}. \tag{3.9}$$

Considering the definition of y_i given in (3.2), further (3.6) and (3.8) we get the required (3.4). ■

The following program is based on Theorem 3.1. It decides whether an n-regular sequence is n-graphical or not.

In this paper we use the pseudocode described in [2].

Input. n: number of vertices ($n \geq 1$);
$b = (b_1, \dots, b_n)$: n-regular sequence.

Output. L: logical variable, whose value is TRUE, if the input is graphical, and it is FALSE, if the input is not graphical.

Work variables. i: cycle variable;
$H = (H_1, \dots, H_n)$: H_i is the cumulated degree, that is the sum of the first i elements of the tested b;
w: the weight point of the actual b_i, that is the maximum of the indices of such elements of b, which are not smaller than i;
y: the cutting point of the actual b_i, that is the maximum of w and i.

ERDŐS-GALLAI-LINEAR(n, b, L)

```
01 H_1 = b_1                                  // line 01: initialization
02 for i = 2 to n        // line 02–03: computation of the elements of H
03      H_i = H_{i-1} + b_i                   // line 04–06: test of the parity
04 if H_n is odd
05    L = FALSE
06    return
07 w = n                        // line 07: initialization of the weight point
08 for i = 1 to n                   // lines 08–12: test of the elements of b
09      while w > 0 and b_w < i
10            w = w - 1
11      y = max(i, w)
12      if H_i > i(y - 1) + H_n - H_y
13         L = FALSE                        // lines 13–14: rejection of b
14         return L
15 L = TRUE                                 // lines 15–16: acceptance of b
16 return L
```

Theorem 3.2. Erdős-Gallai-Linear *decides in* $\Theta(n)$ *time, whether an* n*-regular sequence is graphical or not.*

Proof. Line 01 requires $O(1)$ time, lines 02–03 $\Theta(n)$ time, lines 04–07 $O(1)$ time, lines 08–14 $O(n)$ time (w is decreased at most n times), lines 15–23 $O(n)$ time and lines 24–25 $O(1)$ time, therefore the total time requirement of the algorithm is $\Theta(n)$.

It is worth to remark, that if b is not regular, then we can use a linear time sorting algorithm to sort b and then apply Erdős-Gallai-Linear for the sorted b, that is the testing can be executed in linear time for bounded input too.

Since in the case of a graphical sequence all elements of the investigated sequence are to be tested, in the case of RAM model of computations Erdős-Gallai-Linear is asymptotically optimal testing algorithm. Since we need n memory cells for b, n cells for H, the memory requirement of the program is $\Theta(n)$.

4. Enumerating (EGE) and parallel (EGP) Erdős-Gallai test

Erdős-Gallai-Enumerating algorithm (EGE) [9, 10] generates and tests every zerofree even sequence for given n. Its input is n and output is the number of corresponding zerofree graphical sequences $G_z(n)$.

The algorithm is based on Erdős-Gallai-Linear. It generates and tests (in lexicographic order) only the zerofree even sequences, that is about the 25 percent of the n-regular sequences [10].

EGE tests the input sequences only in the jumping points j_i (defined by $b_i < b_{i+1}$ and $i < n$). According to [10] about the half of the elements of the input sequences is jumping point.

Table 1 contains the number of even ($E(n)$), zerofree graphical ($G_z(n)$), and graphical ($G(n)$) sequences. The bold values were new for *The On-Line Encyclopedia of Integer Sequences*.

Important property of EGE is that it solves in $\Theta(1)$ average time

- the generation of one zerofree even sequence;

n	$E(n)$	$G_z(n)$	$G(n)$	$\frac{G(n)}{E(n)}$
1	1	**0**	1	1.000000
2	2	**1**	2	1.000000
3	6	**2**	4	0.666667
4	19	**7**	11	0.578947
5	66	**20**	31	0.469697
6	236	**71**	102	0.432203
7	868	**240**	342	0.394009
8	3 235	**871**	1 213	0.374961
9	12 190	**3 148**	4 361	0.357752
10	46 252	**11 655**	16 016	0.346277
11	176 484	**43 332**	59 348	0.336280
12	676 270	**162 769**	222 117	0.328444
13	2 600 612	**614 198**	836 315	0.321584
14	10 030 008	**2 330 537**	3 166 852	0.315738
15	38 781 096	**8 875 768**	12 042 620	0.310528
16	150 273 315	**33 924 859**	45 967 479	0.305892
17	583 407 990	**130 038 230**	176 005 709	0.301685
18	2 268 795 980	**499753855**	675 759 564	0.297849
19	8 836 340 260	**1 924 912 894**	2 600 672 458	0.294316
20	34 461 678 394	**7 429 160 296**	10 029 832 754	0.291043
21	134 564 560 988	**28 723 877 732**	38 753 710 486	0.287993
22	526 024 917 288	**111 236 423 288**	149 990 133 774	0.285139
23	2 058 358 034 616	**431 403 470 222**	581 393 603 996	0.282455
24	**8 061 901 596 814**	**1 675 316 535 350**	**2 256 710 139 346**	0.279923
25	**31 602 652 961 516**	**6 513 837 679 610**	**8 770 547 818 956**	0.277526
26	**123 979 635 837 176**	**25 354 842 100 894**	**34 125 389 919 850**	0.275250
27	**486 734 861 612 328**	**98 794 053 269 694**	**132 919 443 189 544**	0.273084
28	**1 912 172 660 219 260**	**385 312 558 571 890**	**518 232 001 761 434**	0.271017
29	**7 516 816 644 943 560**	**1 504 105 116 253 904**	**2 022 337 118 015 338**	0.269042

Table 1. Number of even, zerofree graphical, and graphical sequences for $n = 1, \ldots, 29$ vertices.

- the updating of the sequence of the cumulated degrees H;
- the updating of the sequence of the checking points C;
- the updating of the sequence of the weight points W.

The average running time of this algorithm for a sequence is $\Theta(1)$, so the total running time of the whole program is $\Theta(E(n))$.

We implemented the parallel version of EGE (EGP). It was run on about 200 PC's containing about 700 cores. The total running time of EGP and the number of slices of the whole task are shown in Table 2 for $n = 24, \ldots, 29$ vertices.

n	running time (in days)	number of slices
24	7	415
25	26	415
26	70	435
27	316	435
28	1130	2001
29	6733	15119

Table 2. The total running time of EGP for $n = 24, \ \ldots, \ 29$.

If we wish to enjoy the "parallel advantage" (the substantial decreasing of the running time) of EGP, then we have to increase the number of slices according to the growth of $G(n)$ and some operations on the first sequence of each slice require $\Theta(n)$ time, and therefore the running time of EGP is $\Theta(nE(n))$.

We remark that the referenced papers and programs can be downloaded from the site http://compalg.inf.elte.hu/∼tony/Kutatas/EGHH/.

Acknowledgements. The authors thank András Frank and Zoltán Király (Eötvös Loránd University, Faculty of Science, Institute of Mathematics) for the continuous consultations. The European Union and the European Social Fund have provided financial support to the project under the grant agreement no. TÁMOP 4.2.1/B-09/1/KMR-2010-0003.

References

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

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

[3] P. Erdős and T. Gallai: Gráfok előírt fokú pontokkal (Graphs with prescribed degrees). *Mat. Lapok* **11** (1960), 264–274.

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

[5] S. L. Hakimi: 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.

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

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

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

[9] A. Iványi and L. Lucz: Parallel Erdős-Gallai algorithm. *CEJOR Cent. Eur. J. Oper. Res.* (submitted).

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

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

[12] L. Lovász: *Combinatorial Problems and Exercises* (corrected version of the second edition). AMS Chelsea Publishing, Boston, 2007.

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

[14] N. J. A. Sloane: The number of graphical vectors. In: *The On-Line Encyclopedia of Integer Sequences* (ed. by N. J. A. Sloane), 2011.

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