Proceedings of the World Congress on Engineering and Computer Science 2007
WCECS 2007, October 24-26, 2007, San Francisco, USA

Lexicographic All Circuits Enumeration in Large Scale Macro-econometric Models

André A. Keller *

Abstract—**This contribution is deliberately application-oriented. Conversely, the considered application introduces new circumstances and helpful designs for the graph techniques and interpretations. This study seeks to improve the knowledge of the the cyclical structure of macro-econometric models. The retained application is a recent academic model from Fair '94 for the United States. This empirical economic model of 130 equations is highly interdependent with exactly 7930 elementary circuits.**

Keywords: digraph, circuit, clique, polynomial-time algorithm, backtracking

1 Introduction

This contribution is mainly application-oriented. Conversely, the considered application with macro-econometric models introduces new circumstances and helpful designs for graph theoretical treatments [13][11]. This study seeks to improve the knowledge of the cycle structure of macro-econometric models. This is essential for various well-known purposes such as model building, economic analysis and simulation of economic policies . Given the matching of economic model builders (a non-unique standardization of the system which associates each equation to the determination of one and only one endogenous variable [12]), the causal structure of the associated digraph is studied. The retained application is a recent model of Fair [5] ′94 for the United States. The associated graph $G = (130, 396)$, with 130 vertices and 396 arcs, consists of 53 strong connected components (one of them has a maximum of 77 vertices). This model is highly interdependent with exactly 7930 circuits, of which 8 include a maximum of 42 vertices. Moreover, a maximum list of 13 edge-disjoint circuits has been found for this model. The computations have been carried out using the computer software $Mathematica^{®}$ 5.1 [25], its extension *Combinatorica* [17] and other source programs [20], [15].

*GRAICO/Université de Haute Alsace, Campus de la Fonderie, 68093 Mulhouse Cedex, France Tel/Fax: (0033+)389568200/-8260 Email: andre.keller@uha.fr

2 Elementary Circuit theory

2.1 Preliminaries and notations

Simple graphs, without self-loops and parallel edges, will be mainly considered in this study. A digraph of order n and size m is a pair $G = (V, E)$, where V is a finite set of vertices (with $|V(G)| = n$) and a set of ordered pairs from V (with $|E(G)| = m$). The oriented edges $e_k \in E, k = 1 \ldots, m$ are arcs. A forward arc is such as $e_k = (x_{k-1}, x_k)$, $e_k = (x_k, x_{k-1})$ denoting a backward arc. A chain c of G is an alternating sequence of vertices and arcs $c = (v_0, e_1, v_1, e_2, \ldots, v_{n-1}, e_n, v_n)$, such that $\text{tail}(e_i) = v_{i-1}$ and head $(e_i) = v_i$ for $i = 1, \ldots, n$. The chain is elementary if each vertex v appears only once. A directed graph G is the triple (V, E, f) where V is a finite set of vertices, E a finite set of oriented edges and f a function to be defined. More precisely, a directed graph is defined when $f : E \mapsto V \times V$ [1]. Thus for $f(e) = (v_1, v_2)$, the edge e is incident out of the vertex v_1 and incident into the vertex v_2. The out-degree denotes the number of edges incident out of the vertex v, whereas the in-degree is the number of edges incident into the vertex v. The directed distance $d(u, v)$ is the length of the shortest path between the two vertices u and v [2]. The underlying undirected graph $G^0 = (V, E^0)$ of $G(V, E)$ is obtained by ignoring the direction of the arcs. The connectivity of such an underlying graph introduces the concept of weakly connectivity. Deleting an edge e yields [18] $G' = (V, E', f')$ where $E' = E - \{e\}$, f' being a restriction of f on E'. Deleting a vertex v renders $G' = (V', E', f')$, where $V' = V - \{v\}$, $E' = \{e \in E | f(e) \neq (v, u)$ or (u, v), for $u \in V\}$ and $f' : E' \mapsto V' \times V'$ is a restriction of f to E'. The $n \times m$-incidence matrix H of G has entries

$$H_{ev} := \begin{cases} +1 & \text{if } v \text{ is the terminal vertex of the arc } e, \\ -1 & \text{if } v \text{ is the initial vertex of the arc } e, \\ 0 & \text{otherwise.} \end{cases}$$

[1]The function f for an undirected graph is defined by $f : E \mapsto \{\{u, v\} | u, v \in V\}$.

[2]The distance $d(u, v)$ satisfies the three properties *(i)* $d(u, v) = \infty$ if no path exists from u to v, *(ii)* $d(u, v) = 0 \Rightarrow u = v$, and *(iii)* the directed triangle inequality $d(u, v) + d(v, w) \leq d(u, w)$.

2.2 Circuit definitions and Properties

Definitions A chain c is closed if its initial and terminal vertices coincide ($v_0 = v_n$). A cycle is a closed simple chain. An elementary circuit is an elementary path with identical extremal vertices, such as $(v_1, v_2), (v_2, v_3), \ldots, (v_{n-1}, v_n), (v_n, v_1)$. A circuit is isometric if for any two vertices u and v, it contains a shortest path from u to v and a shortest path from v to u. A short circuit has only one of the two last properties. A circuit is relevant if and only if it cannot be expressed as a linear combination of shorter circuits.

Circuit basis A collection of circuits is vertex-independent (or vertex-disjoint) if no two of the circuits have a vertex in common. It is edge-independent (or edge-disjoint)if no two of the circuits have an edge in common. A representing set for the circuits consists of a set of vertices, through which every circuit passes at least ones. The cycle space $\mathcal{C}$ of graph $G = (V, E)$ is the subspace of $\mathbb{R}^{|E|}$ that is generated by the cycles of G. A circuit basis is a basis of the cycle space $\mathfrak{C}$ of G consisting exclusively of elementary circuits. Berge [1] points out the conditions under which the circuits will generate a cycle space [3][6][3].

2.3 Enumeration problems

There are two orientations for enumerating problems on sets of objects. Indeed, one can either estimate how many objects are in the set or look for an exhaustive finding of the present objects. Both orientations will be considered here.

Counting problem In a complete directed graph K_n the number of elementary circuits is given by

$$\sum_{i=1}^{n-1} \binom{n}{n-i+1} (n-i)!$$

For the complete graph K_3 we find

$$\sum_{i=1}^{2} \binom{n}{4-i} (3-i)! \; = 5.$$

We have 3 circuits of length 2 $\{(1,2,1),(2,3,2),(3,1,3)\}$ and 2 circuits of length 3 $\{(1,2,3,1),(1,3,2,1)\}$. The number of elementary circuits in increased complete graphs from K_3 to K_9 will grow faster than the exponential 2^n. Let us determinate the number of circuits for each length l, $2 \leq l \leq n-1$ in a given complete graph $G = (V, E)$. The estimation of the number of circuits may be estimated (see Appendix A) to

$$\frac{n}{l}\frac{(n-1)!}{(n-l)!}\left(\frac{k}{n-1}\right)^l.$$

[3]Proposition: a strongly connected digraph $G = (V, E)$ has a circuit basis.

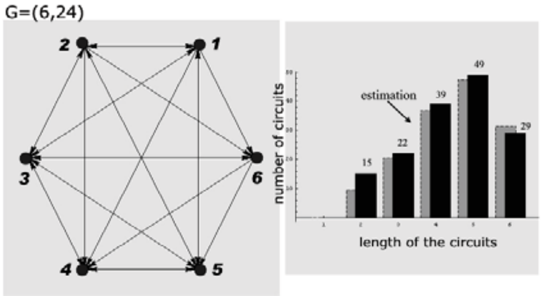

Figure 1: Estimated number of the circuits of length l

The Fig.1 shows an example of an almost-complete digraph $G = (6, 24)$. The discrepancies between the estimated and the true number of circuits have been computed and show close results. However, the errors become extremely high with larger graphs [4].

Enumeration problem Following Prabhaker and Narsingh [18], every circuit enumeration algorithm can be put into four classes of methods : firstly search algorithms, secondly power of the adjacency matrix, thirdly edge-digraph and finally circuit vector space algorithms. The search algorithms look for circuits in a set containing all the circuits [5]. In the depth-first search (DFS) procedure, the vertices are numbered in the order of the visit and this numbering is stored . While visiting vertex u, the procedure DFS(u) is called for each discovered $v \to u$. All vertices of $G - u$ that reachable from v will be visited.

Breadth-first traversal The breadth-first search (BFS) algorithm visits the vertices in layers. The Fig.2 shows an application of this searching procedure to the Petersen graph. Such spanning trees tend to have more but shorter branches than with the DFS procedure. The algorithm uses a queue to maintain a list of subtrees to be explored. Then we have the sequence

visited vertex	*queue contents*
6	6
1	1 7 10
7	7 10 3 4
10	10 3 4 2 8
3	3 4 2 8 5 9
4	4 2 8 5 9
2	2 8 5 9
8	8 5 9
5	5 9
9	9

In the Fig.2, the vertices are shaded according to their distance from the root vertex 6 of the Petersen graph.

[4]Bermond and Thomassen [2] survey the required conditions to a cycle of length k or at least k, in terms of number of arcs, degrees or

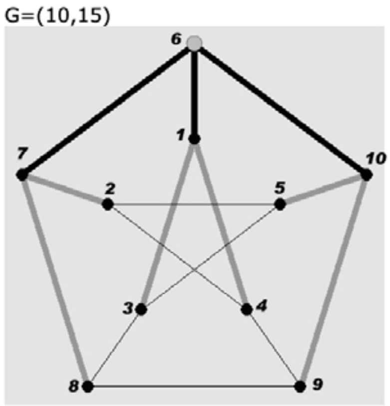

Figure 2: BFS search with Petersen graph

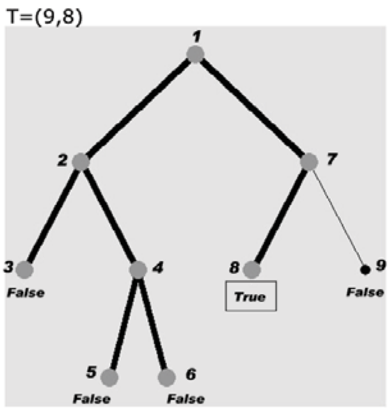

Figure 3: Backtracking procedure

The thick edges show the shortest paths found by the algorithm. Kocay and Kreher [15] indicate that the complexity of the BFS(.) is at most $\mathcal{O}(m)$.

2.4 Backtracking procedure

Suppose that someone is facing with the search of a True option. The options False and True are organized as with the tree of Fig.3. The procedure consists of starting at the root of the decision-tree, continuing until an option is found, and backtracking if necessary until the True option is found. A simple example illustrates this procedure. The visited nodes have been highlighted. The procedure may be simply

1. **Start** at root 1. The decisions are nodes 2 and 7. **Choose** node 2
2. At node 2, the decisions are nodes 3 and 4
3. Option 3 is False. **Then go back** to node 2
4. At node 2 **choose** node 4 (since node 3 was already tried)
5. At node 4, **try** option 5. It is False. **Then go back** to node 4
6. At node 4, **try** option 6. It is False. **Then go back** to node 4
7. **Go back** to node 2. It has been already visited. **Then go back** to root
8. At the root, **choose** node 7
9. At node 7, **try** option 8. It is True. **Then stop**.

In pseudo-code, the algorithm is described by the boolean function[6].

Algorithm : BACKTRACKING PROCEDURE
input: an undirected graph $G(V,E)$ and any arbitrary vertex s
The tree consists of root s, leaves and a goal vertex
output: a path to a goal vertex

```
boolean solve(vertex s){
  if s is a leaf vertex{
    if the leaf is a goal vertex then{
      print s
      return True
    }
    else return False
  } else{
    for each vertex-child c of vertex s{
      if solve(c) succeeds then{
        print s
        return True
      }
    }
    return False
  }
}
```

If solve(s) is True, vertex s is solvable. It is on a path from the root to some goal vertex and is a part of the solution. If solve(s) is False, then no path will include the vertex s to any goal vertex.

2.5 Tarjan's algorithm

The Tarjan's algorithm for enumerating all the elementary circuits of a digraph is produced in Tarjan [22] in ALGOL-like notation. It is based on a backtracking procedure using lookahead and marking technique. This algorithm has a running-time of $\mathcal{O}(n.m(c+1))$ where c denotes the number of elementary circuits and requires has a $\mathcal{O}(n+m+S)$ space, where S is the sum of the lengths of all the elementary circuits [7].

chromatic number. Schrijver [19] also presents complexity surveys.

[5] The problem of counting cycles is treated by Harary and Palmer '73 [7].

[6] Adapted from http://www.cis.upenn.edu/ matuszek/cit594-2002/Pages/backtracking.html. See also [4] [15].

[7] On algorithms for enumerating all circuits of a graph, see [18][16][8][21][9].

Author	Time bound	Space bound	Ref.
Johnson ′75	$(n+m)c$	$n+m$	[10]
Tarjan ′73	$n.m.c$	$n+m$	[22]
Tiernan ′70	$n(const)^n$	$n+m$	[23]
Weinblatt ′72	$n(const)^n$	$n.c$	[24]

Table 1: Complexity of backtrack algorithms to circuits

2.6 Upper bounds on time and space

Prabhaker and Narsingh ′76[18] survey all known algorithms for enumerating all circuits in the 70's. The Table **??** shows the running-time and storage performances of some algorithms using the backtrack method. The Johnson's algorithm is the fastest algorithm, since the running time between two consecutive circuits never exceeds the size of the graph $(n+m)$. The elementary circuits are constructed from a root vertex u in the subgraph $G-u$ and vertices larger than u[8].To avoid duplicate circuits, a vertex v is blocked when it is added to some elementary path beginning by u. This vertex stays blocked as long as every path from v to u intersects the current elementary path at a vertex other than u.

3 Structure of a Macro-econometric model

3.1 Graph properties of the model

The circular embedding is drawn with one giant circuit (the last of the 8 longest circuits) : big white points are those of the giant circuit, the set of the remaining vertices of the giant strongly connected component (SCC), and the remaining vertices of the graph [9]. The model has a maximum of 143 circuits with length of 23. The acyclic directed graph (DAG) clarifies the whole structure and classifies the variables between those which stay upwards or downwards the graph and the interdependent vertices of the giant SCC [10]. Both graph embeddings are shown in Fig.4 and Fig.5. The initial graph $G=(130,396)$, with 130 vertices and 396 arcs, is shown in Fig.4. The acyclic DAG $H=(53,170)$ in Fig.5 has been achieved by contracting the vertices each SCC of G. This function runs in linear time by mapping the vertices of each SCC to one vertex of H. Let $G=(V,E)$ be an n-vertex graph and L a subset with k vertices to contract [17]. Contracting G gives a graph H with $n-k+1$ vertices. Each vertex v in L is mapped to vertex $n-k+1$ in H. Every other vertex v in G is mapped to vertex $v-i$ in H, if there are i vertices in L smaller than v. This mapping is constructed in $\mathcal{O}(n)$ time and creates the edges of H, in time proportional to the number of edges in G.

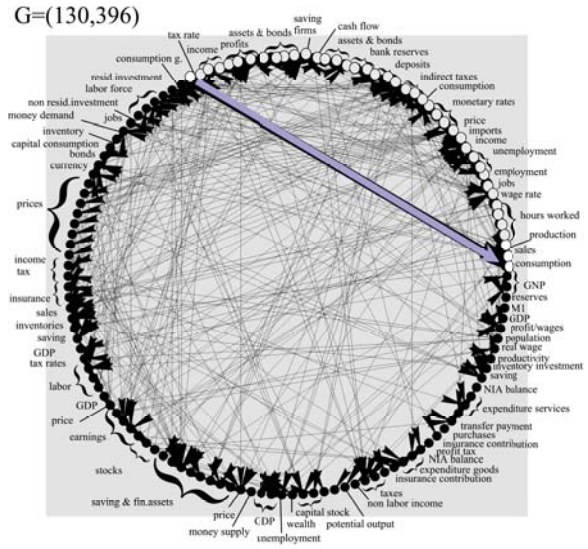

Figure 4: Digraph of Fair's model

Several graph properties derive from the all-pairs shortest-path matrix. The ij-th entry is the length of a shortest path in G or more frequently in a giant SCC of G between vertices i and j. Indeed the eccentricity of a vertex v is the length of the longest shortest path from v to other vertices. The radius is the smallest eccentricity of any vertex, while the center is the set of vertices whose eccentricity is the radius. Conversely, the diameter is the maximum eccentricity of any vertex. The set of peripheral vertices denotes the vertices whose eccentricity is the diameter. Those structural properties have been determined for the graph of the Fair's model : the radius is 5 and the diameter 11. The articulation vertices also play a crucial role. Indeed, an articulation vertex is a vertex whose deletion will disconnect the graph. In a k-connected graph, there does not exist a set of $k-1$ vertices whose deletion disconnects the graph. The properties of the Fair's model are shown by representing those particular vertices of the giant SCC in Fig.6(a). The Fig.6(b) also shows the maximum clique of the graph. Indeed, the largest clique induces a complete graph. The maximum clique has been highlighted and the economic meaning is given.

4 Circuits of the digraph

The model is highly interdependent with 7930 circuits, of which 8 include a maximum of 42 vertices. In this application, all the circuits are listed in an lexicographic order with increasing length. The Figs.7 show the number of circuits in the graph of the Fair's model with respect to

[8]According to the Tiernan's principle [23].

[9]This embedding has been obtained by P^TAP, combining two permutation matrices P and the adjacency matrix A. The resulting digraph is isomorphic to the initial one.

[10]The resolution and the analysis of the model will take advantages from this information.

ISBN:978-988-98671-6-4

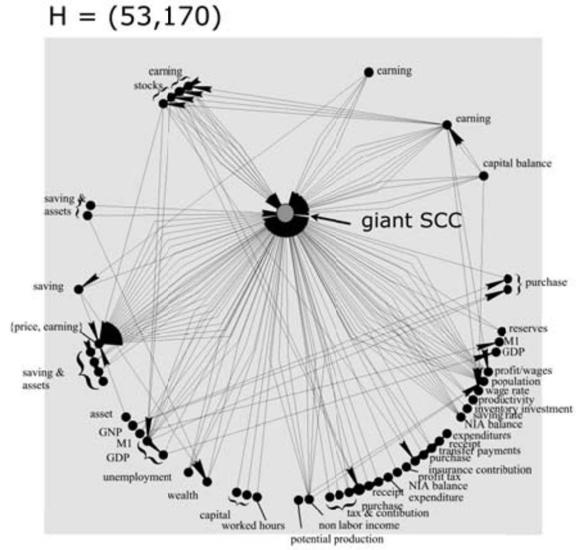

Figure 5: Acyclic digraph of Fair's model

their length.

5 Edge-disjoint circuits

A maximal list of edge-disjoint circuits is shown for the Fair's model. It consists of 13 edge-independent circuits, whose four of them are of length 2, one of a maximum length of 15.

6 Concluding remarks

This study aimed at improving the knowledge of the cyclical structure of macro-econometric models, using associated digraphs. This is essential for various purposes such as model building, economic analysis and simulation of economic policies. The highly interdependence is proved when finding all the 7930 circuits. The graph properties underline the great vulnerability of such an instrument. These kind of results with associated digraphs and the possibility to consider weighted graphs for such application [14] helpfully contribute to the model building process.

References

[1] Berge, C., *Graphs*, North Holland, 1985.

[2] Bermond, J.C., Thomassen, C.,"Cycles in Digraphs – A Survey", *J. Graph Theory*,V5, pp.1–43, /81.

[3] Erdös, P., Pósa, L.,"On Independent Circuits Contained in a Graph", *Canad. J. Math.*,V17, pp.347–352, /65.

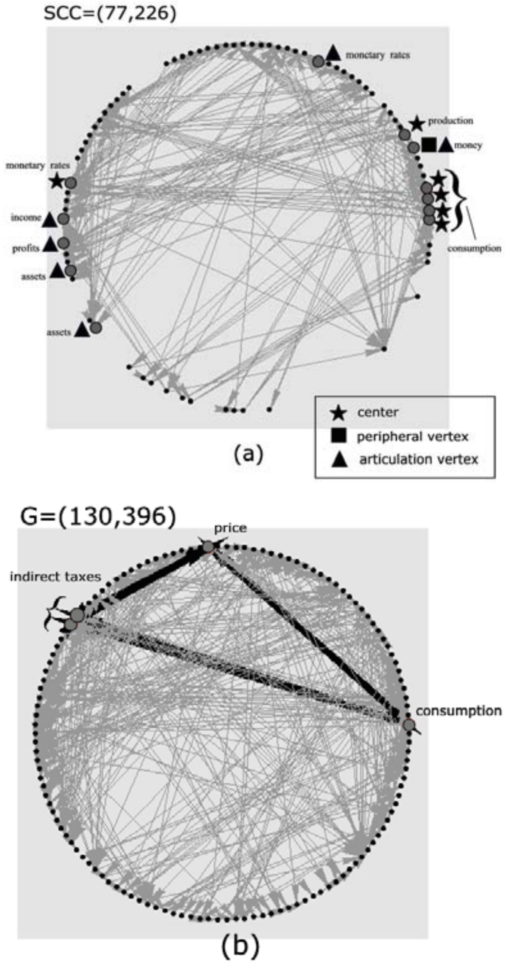

Figure 6: (a)Centers, peripheral and articulation vertices, (b) Clique of the Fair's model

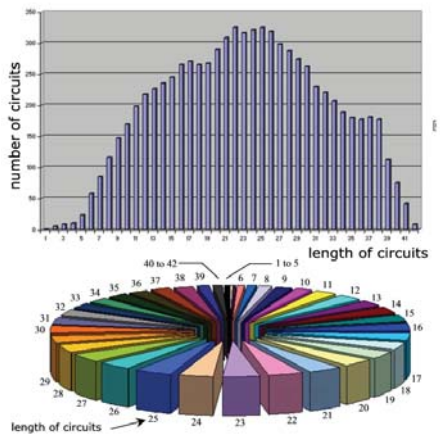

Figure 7: Circuits of given length in Fair's model

[4] Even, S.,*Graph Algorithms*, Computer Science Press, 1979.

[5] Fair, R.C.*Testing Macroeconometric Models*, Harvard University Press, 1994.

[6] Gleiss, P.M., Leydold, J., Stadler, P.F., "Circuit Bases of Strongly Connected Digraph", Department of Applied Statistics and Data Processing, Wirtschaftsuniversität Wien, *Preprint Series*, preprint 40, April 2001. http://epub.wu-wien.ac.at http://epub.wu-wien.ac.at.

[7] Harary, F., Palmer, E., *Graphical Enumeration*, Academic Press, 1973.

[8] Hopcroft, J., Tarjan, R.E."Algorithm 447 : Efficient Algorithms for Graph Manipulation", *Comm. ACM*, N16, pp.372–378, /73.

[9] Horton, J.D.,"A Polynomial-time Algorithm to Find the Shortest Cycle Basis of a Graph", *SIAM J. Comput.*,V16, N2, pp.358–366, 04/87.

[10] Johnson, D. B.,"Finding All the Elementary Circuits of a Directed Graph", *SIAM J. Comput.*, V4, N1, pp.77–84, 03/75.

[11] Keller, A. A., "The Structure of Economic Models : Graph Theory with Mathematica". *IMS'06, 8th International Mathematica Symposium*, Avignon Palais des Papes, France, 19-23 June 2006. INRIA ISBN 2-7261-1289-7.

[12] Keller, A. A., "Matching Theory and Economic Building". *Optimal Discrete Structures and Algorithms*, ODSA 2006, University of Rostock, Germany, 4-6 September 2006, *Electr. Notes in Discrete Math.*, V27 (2006), pp.57–58 (abstract and references).

[13] Keller, A. A., "Graph Theory and Economic Models : from Small to Large Size Applications". *Electr. Notes Discrete Math.*, V28, pp.469–476, /07.

[14] Keller, A. A., "Connectivity Treshold and Vulnerability in Large Scale Macro-economic Models". *The 2007 British Combinatorial Conference*, University of Reading UK, 8-13 July 2007.

[15] Kocay, W., Kreher, D.l. *Graphs, Algorithms, and Optimization*, Chapman & Hall/CRC, 2004.

[16] Marcu, D., "On Finding the Elementary Paths and Circuits of a Digraph' ', *Sci. Bull. Polytech. Univ. Bucharest, Ser. D*, V55 N3-4, pp.29-33 /93.

[17] Pemmaraju, S., Skiena, S., *Combinatorics, and Graph Theory with Mathematica®*, Cambridge University Press, 2003.

[18] Prabhaker, M., Deo, N. "On Algorithms for Enumerating All Circuits of a Graph' ', *SIAM J. Comput.*, V5, N1, pp. 90–99 03/76.

[19] Schrijver, A., *Combinatorial Optimization : Polyedra and Efficiency*, Volume A (Paths, Flows, Matchings), Springer-Verlag, 2003.

[20] Sedgewick, R. *Algorithms in C*, Part 5 (Graph Algorithms), Addison-Wesley, 2002.

[21] Tarjan, R.E., "Enumeration of the Elementary Circuits of a Directed Graph", *SIAM J. Comput.*, V2, N3, pp. 211–216, 09/73.

[22] Tarjan, R.E., " Depth-first Search and Linear Graph Algorithms", *SIAM J. Comput.*, V1, pp.146–160, /72.

[23] Tiernan, J.C., "An Efficient Search Algorithm to Find the Elementary Circuits of a directed Graph", *Comm. ACM*, N13, pp.722–726, /70.

[24] Weinblatt, H., "A New Search Algorithm for Finding the Simple Cycles of a Finite Directed Graph", *J. Assoc. Comput. Mach.*, N19, pp.43–56, /72.

[25] Wolfram, S. *The MATHEMATICA® Book*, Fiveth Edition, Wolfram Media Inc., 2003. http://www.wolfram.com http://www.wolfram.com.

A Appendix. Estimated number of circuits of length l

Let $G = (V, E)$ be an almost-complete digraph of order n and size m. The average number of arcs is the integer part of $k = m/n$. Starting from one vertex i, the total paths of length 1 is $\frac{n-1}{n-1}k$, where the numerator states the number of extremal vertices other than i and the denominator the total number of possible extremal vertices. At an another vertex $j, j \neq i$, there are $n - 2$ extremal vertices left. Hence, the number of paths of length 1 starting from j is $\frac{n-2}{n-1}k$. The number of paths of length 2 from vertex i is then

$$\frac{n-1}{n-1}k \times \frac{n-2}{n-1}k \; = \frac{(n-1)(n-2)}{(n-1)^2}k^2.$$

Hence, we have $\frac{(n-1)(n-2)\ldots(n(l-1))}{(n-1)^{l-1}}k^{l-1}$ paths of length $l-1$ starting from vertex i. The average number of arcs joining a vertex other than i to i being $\frac{k}{n-1}$, the total circuits of length l is firstly estimated to

$$\frac{k}{n-1}\frac{(n-1)(n-2)\ldots(n(l-1))}{(n-1)^{l-1}}k^{l-1} = \frac{(n-1)!}{(n-l)!}(\frac{k}{n-1})^l.$$

However for n vertices each circuit will be counted l times. Therefore the number of circuits of length l may be estimated to

$$\frac{n}{l}\frac{(n-1)!}{(n-l)!}(\frac{k}{n-1})^l.$$