Applicable Analysis and Discrete Mathematics
available online at http://pefmath.etf.rs

Appl. Anal. Discrete Math. **6** (2012), 1–30. doi:10.2298/AADM111223025A

GRAPH SPECTRAL TECHNIQUES IN COMPUTER SCIENCES

Branko Arsić, Dragoš Cvetković, Slobodan K. Simić, Milan Škarić

We give a survey of graph spectral techniques used in computer sciences. The survey consists of a description of particular topics from the theory of graph spectra independently of the areas of Computer science in which they are used. We have described the applications of some important graph eigenvalues (spectral radius, algebraic connectivity, the least eigenvalue etc.), eigenvectors (principal eigenvector, Fiedler eigenvector and other), spectral reconstruction problems, spectra of random graphs, Hoffman polynomial, integral graphs etc. However, for each described spectral technique we indicate the fields in which it is used (e.g. in modelling and searching Internet, in computer vision, pattern recognition, data mining, multiprocessor systems, statistical databases, and in several other areas). We present some novel mathematical results (related to clustering and the Hoffman polynomial) as well.

1. INTRODUCTION

In this paper we shall give a survey of parts of the theory of graph spectra which are useful in computer sciences.

Spectral graph theory is a mathematical theory in which linear algebra and graph theory meet. For any graph matrix M we can build a spectral graph theory in which graphs are studied by means of eigenvalues of the matrix M. This theory is called *M-theory*. In order to avoid confusion, to any notion in this theory a prefix M- could be added (e.g., M-eigenvalues). Frequently used graph matrices are the adjacency matrix A, the Laplacian $L = D - A$ and the signless Laplacian $Q = D + A$, where D is a diagonal matrix of vertex degrees. Some other graph

2010 Mathematics Subject Classification. 05C50, 68P20, 68R10.
Keywords and Phrases. Spectral graph theory, computer science, internet, complex networks, spectral clustering.

matrices will be introduced later. *The* spectral graph theory includes all these particular theories together with interaction tools.

It was recognized in about the last ten years that graph spectra have several important applications in computer sciences (see, e.g., [**35, 36, 45**]). Graph spectra appear in the literature in Internet technologies, computer vision, pattern recognition, data mining, multiprocessor systems, statistical databases and in many other areas. There are thousands of such papers.

The two of us (D. C. and S. K. S.) have published a survey [**45**] of the applications of graph spectra in Computer science. We have identified several applications in the following branches of Computer science:

1. Expanders and combinatorial optimization,
2. Complex networks and the Internet topology,
3. Data mining,
4. Computer vision and pattern recognition,
5. Internet search,
6. Load balancing and multiprocessor interconnection networks,
7. Anti-virus protection versus spread of knowledge,
8. Statistical databases and social networks,
9. Quantum computing.

Subsequently, we have also become aware of applications in

10. Bioinformatics,
11. Coding theory,
12. Control theory.

This classification of numerous applications contains some overlapping in the classified material. For example, methods of data mining (in particular, spectral graph clustering) appear in computer vision, social networks and Internet search while several problems of combinatorial optimization are relevant for data mining (e.g., in clustering).

Since methods of Computer science are present in all branches of science, applications of graph spectral techniques to Computer science are transferred to almost all branches of science (telecommunications, electrical engineering, biology, chemistry, geography, social sciences, etc.). Sometimes by using the adjective "computational" one can denote those parts of particular sciences which overlap with Computer science (e.g., computational biology, computational chemistry, etc.). In this sense one can speak of computer *sciences* as we have put in the title of this paper.

Of course, graph spectra appear in Computer science since graphs for themselves are relevant. The main benefit of using graph spectra comes from the fact that eigenvalues and eigenvectors of several graph matrices can be quickly computed (computational complexity is $O(n^3)$ where n is the number of vertices). However, spectral graph parameters contain a lot of information on the graph structure (both

global and local). This includes some information on graph parameters that, in general, are computed by exponential algorithms (e.g. chromatic number, the size of maximal clique, etc.). For example, computing the chromatic number of a graph with a few thousands vertices is a difficult task while eigenvalues and eigenvectors can be computed in a few seconds (by iterative algorithms). See Section 5 for further discussion on these topics.

Graphs that are treated in computer sciences using graph spectra typically represent either some physical networks (computer network, Internet, biological network, etc.) or data structures (documents in a database, indexing structure, etc.) In the first case the graphs usually have a great number of vertices (thousands or millions) and they are called *complex networks* while in the second case graphs are of small dimensions.

The approach in this paper is essentially different from one in the companion paper [**45**]. We describe particular topics from the theory of graph spectra independently of the areas of Computer science in which they are used. However, for each described spectral technique we indicate fields where they are used. Of course, we do not have space here to provide standard details from the theory of graphs spectra; instead we direct the reader to the corresponding mathematical literature, in particular to books [**33, 41**]. In order to reduce the overlap with [**45**] we have omitted some explanations which can be found in [**45**].

It should be noted that papers in Computer science sometimes contain mathematical contributions to the theory of graph spectra (see, e.g., [**118, 64**]).

The rest of the paper is organized as follows. Section 2 contains description of several graph matrices while other sections present topics from the theory of graph spectra which are used in applications. Section 3 is devoted to significant graph eigenvalues, while Section 4 describes eigenvector techniques. Graph spectrum characterizes a graph to a great extent so that eigenvalues are used to encode or to index the graph. Such techniques are presented in Section 5. Spectra of random graphs appear in Section 6, while Section 7 contains miscellaneous topics from the theory of graph spectra with indications of their specific application. Section 8 concludes the paper.

2. GRAPH MATRICES

It should be noted that spectra of several graph matrices appear in applications. The adjacency matrix and Laplacian appear most frequently but also the signless Laplacian as well as normalized versions of these matrices. Incidence, distance and other matrices can be found as well. Sometimes the considerations move from graph matrices to general ones; equivalently, weighted graphs appear instead of graphs. In some cases we encounter digraphs and hyper-graphs and corresponding matrices as well.

One can notice that not only the eigenvalues but also the eigenvectors of relevant graph matrices appear in applications in most cases.

In many papers the normalized Laplacian matrix $\hat{L} = D^{-\frac{1}{2}}(D - A)D^{-\frac{1}{2}} =$

$D^{-\frac{1}{2}}LD^{-\frac{1}{2}}$ appears. This matrix has 1's on the diagonal, and at an off-diagonal position (i,j) the entry is equal to 0 for non-adjacent and $-\frac{1}{\sqrt{d_i d_j}}$ for adjacent vertices i,j of degrees d_i, d_j. The spectrum of $\hat{L}$ belongs to the interval $[0,2]$ independently of the number of vertices. The book by F. Chung [**23**] is devoted to the normalized Laplacian.

For non-trivial connected graphs the matrices $D^{-1}A$ and $(2D)^{-1}Q = (2D)^{-1}(D+A) = \frac{1}{2}(I + D^{-1}A)$ are transition matrices of Markov chains for random and lazy random walks.

Note that the normalized Laplacian matrix $\hat{L} = D^{-\frac{1}{2}}LD^{-\frac{1}{2}} = D^{-\frac{1}{2}}(D - A)D^{-\frac{1}{2}} = I - D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$ and the normalized signless Laplacian matrix $\hat{Q} = D^{-\frac{1}{2}}QD^{-\frac{1}{2}} = D^{-\frac{1}{2}}(D+A)D^{-\frac{1}{2}} = I + D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$ are connected by the relation $\hat{Q} = -\hat{L} + 2I$. This means that $\hat{Q}$-theory is simply reduced to $\hat{L}$-theory. A similar statement holds for the matrix $D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$.

The book [**33**] describes some spectral properties of the matrices $D^{-1}A$ and $D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$ (they are similar, see p. 48) and $\frac{1}{2}(I + D^{-1}A)$ (see p. 110). All three matrices have real eigenvalues.

From the adjacency matrix $A = [a_{ij}]$ of a graph G we can build the matrix $B = [b_{ij}]$ where $b_{ij} = a_{ij} - \frac{d_i d_j}{m}$, m being the number of edges of G. This matrix is called the *modularity matrix* (see, for example, [**86**]).

Let G be a graph with adjacency matrix A and consider the matrix $H(t) = e^{iAt}$, where t is a real variable and $i^2 = -1$. This matrix appears in quantum computing.

Very frequently we encounter *affinity* or *similarity* matrices. For a set of objects the entries of such matrices indicate the measure of affinity or similarity between the corresponding objects. For a set of points in an Euclidean space the affinity between two points at distance d is usually defined as $\exp(-d^2/2\sigma^2)$, where σ is a parameter.

Affinity matrices can be understood as adjacency matrices of weighted (complete) graphs. The row sums now play the role of vertex degrees. Such matrices can be normalized or transformed in a Laplacian-like form.

For a digraph G one can consider symmetric matrices AA^T and A^TA together with the adjacency matrix A of G. The matrix AA^T (A^TA) contains out- (in-) degrees on the diagonal while the (i,j)-entry is equal to the number of common front (rear) neighbours for vertices i and j.

The adjacency matrix of a digraph G with positive out-degrees could be normalized so that the sum of entries in each row is equal to 1. This is achieved by dividing the entries in each row by the out-degree of the corresponding vertex. Equivalently, we form a new matrix $P = D_+^{-1}A$ where D_+ is the diagonal matrix of out-degrees. The matrix P is a transition matrix of a Markov chain and the

normalized eigenvector of the largest eigenvalue of its transpose P^T defines the steady-state of the chain if it exists.

3. SIGNIFICANT EIGENVALUES

Some graph eigenvalues are for themselves important graph parameters and are therefore also interesting in applications. We shall consider in separate subsections the largest A-eigenvalue, the second smallest L-eigenvalue, the second largest A-eigenvalue and the least A-eigenvalue. We conclude with a subsection on main eigenvalues.

3.1. Largest eigenvalue

The largest eigenvalue, i.e. the *index* (known also as *spectral radius*), of a graph is a mathematically very important graph parameter as presented, for example, in a survey paper [**38**].

By Theorem 1.12 of [**33**] the index of a graph is equal to a kind of mean value of vertex degrees, i.e. to the so called dynamical mean value, which takes into account not only immediate neighbors of vertices but also the neighbors of the neighbors, etc. The index is also known to be a measure of the extent of branching of a graph, and in particular of a tree (see [**34**] for the application in chemical context and [**31**] for a treatment of directing the branch and bound algorithms for the travelling salesman problem).

The largest eigenvalue λ_1 of the adjacency matrix plays an important role in modelling virus propagation in computer networks. The smaller the largest eigenvalue, the larger the robustness of a network against the spread of viruses is. In fact, it was shown by Y. Wang et al. in [**112**] that the *epidemic threshold* in spreading of the viruses is proportional to $1/\lambda_1$. Another model of virus propagation in computer networks has been developed by P. Van Mieghem et al. in [**109**] with the same conclusion concerning $1/\lambda_1$. Motivated by the above facts, the authors of [**47**] determine graphs with minimal λ_1 among graphs with a given number of vertices and having a given diameter.

Research and development networks (R&D networks) are studied using the largest eigenvalue of the adjacency matrix in papers [**72, 73**] by M. D. König et al. In such networks it is desirable that the knowledge is spread through the network as much as possible. Therefore the tendency is to achieve high values of the largest eigenvalue, just opposite to the considerations of virus propagation.

An intuitive explanation of both phenomena, advantage to have a minimal value of λ_1 for virus protection and a maximal value of λ_1 for knowledge spread, can be obtained by the fact that the number of walks of length k in a connected graph behaves asymptotically as $c\lambda_1^k$ for a constant $c > 0$.

Counting walks with specified properties in a graph (or digraph) is related to graph spectra by the following well-known result (see [**33**] p. 44).

Theorem 3.1. *If A is the adjacency matrix of a graph, then the (i, j)-entry $a_{ij}^{(k)}$ of the matrix A^k is equal to the number of walks of length k that originate at vertex*

i and terminate at vertex j.

The greater the number of walks, the more intensive the spread of the moving substance is, no matter whether this is the virus or the knowledge.

Our paper [**45**] announced a result of [**101**] that under some conditions the balanced subdivisions of regular graphs of degree 3 have the smallest index among connected graphs with a fixed numbers of vertices and edges if n is large enough. Hence, balanced subdivisions of regular graphs of degree 3 should be considered as good models of virus resistant computer networks.

The quantity $1/\lambda_1$ also appears in several other networks, including sensor and wireless networks (see [**100**], where it is called the *congestion number*). The intuitive explanation to this definition is that while we have more paths of a fixed length in order to send information, we can split the information on these paths and coordinate them to arrive with the same number of hops at the receiver. This has the advantage of equalizing source-destination delays of packets that belong to the same class, which allows one to minimize the amount of packets that come out of sequence. This is desirable since in data transfers, out of order packets are misinterpreted to be lost which results not only in retransmissions but also in drop of systems throughput.

We shall mention the use of the index in the area of multiprocessor interconnection networks. For motivation and details see Subsection 7.1, our previous review paper [**45**] and the source papers [**27, 28, 29**].

There are four related notions which are useful. The *first type mixed tightness* $t_1(G)$ of a graph G is defined as the product of the number of distinct eigenvalues m and the maximum vertex degree Δ of G, i.e. $t_1(G) = m\Delta$. *Structural tightness* $\mathrm{stt}(G)$ is the product $(\mathrm{diam}+1)\Delta$ where diam is the diameter of a graph G. *Spectral tightness* $\mathrm{spt}(G)$ is the product $m\lambda_1$ of the number of distinct eigenvalues m and the largest eigenvalue λ_1 of a graph G. The *second type mixed tightness* $t_2(G)$ is defined as $t_2(G) = (\mathrm{diam}+1)\lambda_1$.

According to the well-known inequality ([**33**], p. 85)

$$\delta \leq \overline{d} \leq \lambda_1 \leq \Delta,$$

where δ and Δ denote minimum and maximum vertex degree, respectively, and $\overline{d}$ denotes the average value of the vertex degrees, we have that $\mathrm{spt}(G) \leq t_1(G)$. The relation between $\mathrm{stt}(G)$ and $t_1(G)$ is $t_1(G) \geq \mathrm{stt}(G)$, since $m \geq 1 + \mathrm{diam}$ (see Theorem 3.13. from [**33**]). Finally, we have $t_2(G) \leq \mathrm{spt}(G)$ and $t_2(G) \leq \mathrm{stt}(G)$

The use of the largest eigenvalue, i.e. the index, of a graph instead of the maximal vertex degree in description of multiprocessor topologies seems to be appropriate for several reasons [**27, 28, 29**].

3.2. Algebraic connectivity

The second smallest Laplacian eigenvalue is called *algebraic connectivity* of the graph and was introduced by Fiedler [**54**] (for a graph G we write $\alpha(G) = \nu_2(G)$). The importance of this eigenvalue stems from the fact that both "graphic" measures

of connectivity of graphs, i.e. *vertex connectivity* κ and *edge connectivity* κ', do not feature, in many situations, as appropriate measures. For example, both are equal to 1 for all trees, while the algebraic connectivity is the largest for stars (also equal to 1) and the smallest for paths. It is also known (by the interlacing theorem, see Theorem 5.2) that any edge deleted subgraph of a connected graph has algebraic connectivity that is no larger than the graph in question; thus, as is expected, it possesses a monotonicity property. In addition there are many other inequalities which relate the algebraic connectivity to connectivities κ and κ'. Namely, we have

$$2\kappa'(G)\left(1-\cos\frac{\pi}{n}\right) \leq \alpha(G) \leq \kappa(G) \leq \kappa'(G) \leq \delta(G),$$

where G is a graph on n vertices and minimal vertex degree δ (for more details, see, e.g., [**41**], Section 7.5). For further information see [**1, 68**].

The algebraic connectivity is related to many structural graph invariants and several very important problems in combinatorial optimization. Here we put the emphasis on separation and metric problems, and also isoperimetric and expansion problems. These problems give rise to several important graph invariants (see, e.g., [**41**] Section 7.5 and 7.6). Most of the problems arising in this context are, from the complexity points of view, very hard (usually NP-hard). In solving such problems algebraic connectivity appears as a crucial quantity in designing various heuristics. (Its main advantage is that it can be computed in polynomial time). It is noteworthy to add that there are some other possibilities where algebraic connectivity can be used. For example, in [**26**], the symmetric *travelling salesman problem* is formulated in terms of discrete semi-definite programming by means of algebraic connectivity.

The most important separation problems are *max-cut* and *min-cut problems*, and also *bipartition width* (all being NP-hard). Let $S \cup \overline{S}$ be a bipartition of the vertex set of some graph, and let $\partial(S) = \{st : s \in S, t \in \overline{S}\}$. So edges in $\partial(S)$ represent a cut in the graph in question. The first two cut problems are related to finding a cut of maximal and minimal size, while the third one features as a min-cut problem in which bipartition is even (so cardinalities of S and $\overline{S}$ differ at most by 1). The cardinalities of the corresponding sets S give rise to graph invariants (i.e. the measures of separation), and there are many bounds on these invariants in the literature which relate them with the algebraic connectivity (see Subsection 7.5.1 of [**41**]). In the context of metric problems, most of the results in the literature are related to bounding diameter by using algebraic connectivity and largest eigenvalue of the Laplacian (see Subsection 7.5.2 of [**41**]). The min-cut problem can be treated by eigenvectors of $\nu_2(G)$ (see Subsection 4.2).

We now switch to isoperimetric problems, which are the discrete analogies of classical isoperimetric problems (in Euclidean spaces). Thus, the aim is to find the minimum value of quotient $\frac{|\partial(S)|}{|S|}$, where S is non-empty and does not contain more than the half of vertices of the graph in question. The related quantity is called the *isoperimetric number* of a graph (or an *edge expansion number* - see below). Its relation to the algebraic connectivity is expressed by various inequalities (or

lower and upper bounds, obtained by B. Mohar [**82**]). If $i(G)$ is an isoperimetric number of a graph G on at least 4 vertices then

$$\frac{\alpha(G)}{2} \leq i(G) \leq \sqrt{\alpha(G)(2\Delta(G) - \alpha(G)};$$

here $\Delta(G)$ is the maximal (vertex) degree of G.

The expansion problems represent the vertex counterpart of isoperimetric problems. Now, instead of the edge boundary (denoted by $\partial(S)$) we consider the vertex boundary, defined by $\delta(S) = \{t : t \sim s, s \in S\}$. Similarly as above, now in the problem which arises the aim is to find the minimum value of the quotient $\frac{|\delta(S)|}{|S|}$, where S is restricted as above. The related quantity (expansion) is called the *vertex expansion number* or the *expansion number* of a graph for short. Again, its relation to algebraic connectivity can be expressed by various inequalities. If $j(G)$ is an expansion number of a non-trivial graph G then the following bound is obtained by Alon [**2**]:

$$j(G) \geq \frac{2\epsilon}{\Delta(G) + 2\epsilon},$$

where $0 \leq \epsilon \leq \alpha(G)$; in addition, if $j(G) \geq c > 0$, then

$$\alpha(G) \geq \frac{c^2}{2c^2 + 4}.$$

Finally, we mention some further details related to expansion problems. Informally, in this context, we are interested in constructing graphs (or better say families of graphs), which are in a sense good "expanders". This means, we have to find sparse graphs with high connectedness, or equivalently, that any small subset of its vertices has a good connections to the rest of a graph. Accordingly, in practice then we encounter various graphs often couched in terms of enlargers, magnifier, concentrators and super-concentrators, just to mention some specific terms (for definitions, see, e.g, [**41**]). Among such graphs, expanders (usually defined for bipartite graphs), are also described in the same book. All of them, appear in the treatment of several problems in Computer science (for example, communication networks, error-correcting codes, optimizing memory space, computing functions, sorting algorithms, etc.). Here we only note that good expanders are used as models for robust network in Computer science. For further details, see Subsection 3.3.

The ratio $\gamma = \frac{\nu_2}{\nu_n}$, where ν_n is the largest while ν_2 the second smallest eigenvalue of the graph Laplacian, is also an important graph invariant. This ratio is relevant for the process of synchronization in complex networks in some special cases [**57**].

The same ratio appears in sensor networks. (A *sensor network* consists of spatially distributed sensors with limited capacities and links connecting them.) One of the basic problems with these networks is to design a topology (connection graph) that maximizes the ratio γ. The larger this ratio is, the faster is the convergence speed of the decision fusion algorithm, and thus better the performance of

the network. In [**66**], it was pointed that (non-bipartite) Ramanujan graphs (see the next subsection) are good candidates for desired topologies.

There are many other problems in sensor networks where the tools from the combinatorial optimization and spectral graph theory can be helpful, say in solving partitioning, assignment, routing and scheduling problems.

3.3. The second largest eigenvalue

The second largest A-eigenvalue of r-regular graphs is related to the algebraic connectivity (namely, we have $\lambda_2 + \nu_2 = r$). Therefore, for regular graphs the problem of maximizing the algebraic connectivity becomes equivalent to that of minimizing the second largest eigenvalue.

Good expanders can be constructed from graphs with a small second largest eigenvalue in modulus. This class of graphs includes the so called *Ramanujan graphs*. Let $\Lambda(G)$ be the second largest modulus of an eigenvalue of a graph G. A *Ramanujan graph* is a connected r-regular graph for which $\Lambda(G) \leq 2\sqrt{r-1}$. This is the Boppana bound which represents the limes inferior of $\Lambda(G)$ over the set of connected r-regular graphs G (see [**79**] as one of the most important papers concerning Ramanujan graphs). For further details on the applications of Ramanujan graphs, see [**44**] and references therein.

3.4. The least eigenvalue

There are some applications of the theory of graphs with least A-eigenvalue -2.

A *dumbbell* is a graph obtained by joining two cycles by a path (possibly of length 0). If both cycles of a dumbbell are odd, then the dumbbell is called *odd*. It is well-known (cf., e.g., [**40**], p. 126) that eigenvectors of a basis of the eigenspace of the eigenvalue -2 in line graphs can be obtained by certain edge valuations of even cycles and odd dumbbells in the root graphs.

Statistical databases are those that allow only statistical access to their records. Individual values are typically deemed confidential and are not to be disclosed, either directly or indirectly. Thus, users of a statistical database are restricted to statistical types of queries, such as looking for the sum of the values, minimum or maximum value of some records, etc. Moreover, no sequence of answered queries should enable a user to obtain any of the confidential individual values. However, if a user is able to determine a confidential individual value, the database is said to be *compromised*. Statistical databases that cannot be compromised are called *compromise-free* or *secure*.

In the special case where queries are related to the sum of values of records in the database and each record is contained in at most two queries, the query matrix corresponds to a an incidence matrix of a graph G. Queries correspond to vertices and records correspond to edges.

The results from [**13, 15**] show an interesting connection between the compromise-free query collections and the graphs with least eigenvalue -2 [**40**]. This connection was recognized in [**14**].

The following theorem was proved in [**13, 15**] with a different terminology. See [**45**] for some bibliographical comments.

Theorem 3.2. *A database is compromise-free if and only if each edge of G is contained either in an even cycle or in an odd dumbbell of G.*

3.5. Main eigenvalues

An A-eigenvalue of a graph is called *main* if the corresponding eigenspace contains a vector in which the sum of coordinates is different from 0.

Graphs in which all eigenvalues are mutually distinct and main have recently attracted some attention. There are no such graphs on less than 6 vertices and there are exactly 8 connected graphs with this property on 6 vertices. One can prove that such graphs have a trivial automorphism group [**42**].

In control theory networked dynamic systems which consist of independent "agents" (integrators) exchanging information along edges of a graph are considered. Such a system is "controllable" if and only if the corresponding graph has all eigenvalues mutually distinct and main [**91, 42, 43**].

4. EIGENVECTOR TECHNIQUES

Graph eigenvectors also contain a lot of information on graph structure. However, one should point out that eigenvectors are not graph invariants since they depend on the labelling of graphs. On the other hand, that can be an advantage, especially when one is looking for a cleaver labelling of the graph, for example, in Subsection 4.2. The subsections are devoted to the principal eigenvector, the Fiedler eigenvector and to problems related to simultaneous consideration of several eigenvectors.

4.1. Principal eigenvector

The normalized positive eigenvector belonging to the largest A-eigenvalue of a connected graph is called the *principal eigenvector*.

The subject of ranking individuals or objects by eigenvectors of suitably chosen graph matrices is an old subject in the mathematical literature. One of the basic references is the thesis [**114**]. In particular, the ranking of the participants of a round-robin tournament can be carried out in this way (see, e.g., [**33**], p. 226). These methods have been used in the field of sociology for a long time as well (see, e.g., [**11**]).

We reproduce here a relevant result. The following theorem of T. H. Wei [**114**] is noted in [**39**], p. 26:

Theorem 4.1. *Let $N_k(i)$ be the number of walks of length k starting at vertex i of a non-bipartite connected graph G with vertices $1, 2, \ldots, n$. Let $s_k(i) = N_k(i) \cdot \Big(\sum_{j=1}^{n} N_k(j)\Big)^{-1}$. Then, for $k \to \infty$, the vector $(s_k(1), s_k(2), \ldots, s_k(n))^T$ tends towards an eigenvector corresponding to the index of G.*

Hence, ranking graph vertices by coordinates of the principal eigenvectors means ranking them according to numbers of walks. The number of walks $N_k(i)$ can be interpreted as the "influence" or "importance" of vertex i. It is also called the *centrality* of vertex i.

Web search engines are based on eigenvectors of the adjacency and some related graph matrices. The most known systems are PageRank by S. Brin and L. Page [**16**] (used in Google) and Hyperlinked Induced Topics Search (HITS) by J. Kleinberg [**70**].

In this context the structure of the Internet is represented by a digraph G in which web pages correspond to vertices and links between the pages (hyperlinks) to arcs.

HITS exploits eigenvectors belonging to the largest eigenvalues of symmetric matrices AA^T and A^TA, where A is the adjacency matrix of a subgraph of G induced by the set of web pages obtained from search key words by some heuristics. The obtained eigenvectors define a certain ordering of the selected web pages.

PageRank uses similar ideas. Random walks are utilized in this model. In fact, the adjacency matrix of G is normalized so that we use the matrix $P = D_+^{-1}A$, introduced in Section 2. (Prior to this transformation, in order to eliminate zero-rows in P, the arcs going to all of the other vertices are added to each vertex without outgoing arcs. In addition to this, in order to ensure primitivity of the matrix, at least one odd cycle is artificially formed if such one did not exist.) Further, a convex combination $\overline{P}$ of P and a rank 1 matrix is formed. The matrix $\overline{P}$ is a transition matrix of a Markov chain and the normalized eigenvector of the largest eigenvalue of its transpose $\overline{P}^T$ defines the steady-state of the chain. Pages are ranked by the coordinates of this eigenvector.

Expository paper [**75**] contains a survey of both techniques.

The same idea of ranking vertices appears with *eigenvector centrality*, which is also a measure of the importance of a vertex in a network. It assigns relative scores to all the vertices in the network based on the principle that the connections to the high-scoring vertices contribute more to the score of the vertex in question than the equal connections to low-scoring vertices. Google's PageRank is a variant of the eigenvector centrality measure. For more details, see [**20, 21**].

There are many papers in Computer science literature on different aspects of using eigenvectors in Internet search engines.

In computer vision, Sarkar and Boyer [**93**] have shown how the eigenvector of the largest eigenvalue of a relevant graph matrix can be used to group line segments.

4.2. The Fiedler eigenvector

Recall first that the smallest eigenvalue of the graph Laplacian is always equal to 0, and that its multiplicity is equal to the number of the connected components of the graph. Then, for each component, we have an eigenvector whose entries are equal to a non-zero constant for vertices in that component, and zero otherwise (see Theorem 4.2 for details). The eigenvector belonging to the second smallest

Laplacian eigenvalue of the connected graph is called the *Fiedler eigenvector*. Of course, now we have both positive and negative entries in it.

A heuristic for solving the min-cut problem uses the Fiedler eigenvector to partition the vertex set into parts corresponding to positive and negative coordinates of this vector [**55**].

It was observed that useful partitioning of a vertex set of a graph can be based by observing the "sign pattern" of eigenvectors, including the "bottom eigenvectors" of the graph Laplacian (but also of "top eigenvectors" of the adjacency matrix). With this observations in mind, we now describe two ways of graph partitioning heuristics:

(i) *Recursive spectral bisection*: we use Fiedler eigenvector to divide the vertices of the graph into two parts by the sign pattern, and then continue with the same procedure for each part until we satisfy some criterion of optimality (see, e.g., [**7, 89**]).

(ii) *Iterative spectral bisection*: we start as in (i), but next use the third - smallest, the fourth - smallest, etc. eigenvalue to refine our intermediate partitions until we stop due to some criterion (see clustering algorithms in Subsection 4.3).

These ideas were exploited in the literature in various ways for devising powerful heuristics for spectral graph partitioning and/or clustering. For instance, Shi and Malik [**97**] have shown how the sign pattern of the Fiedler eigenvector can be used to separate the foreground from the background structure in images. The original procedure from [**55**] has been improved by using the matrix $D^{-1}L$ (so as to maximize the normalized graph cut). More generally, image segmentation is an important procedure in computer vision and pattern recognition. The problem is to divide the image into regions according to some criteria. Very frequently the image segmentation is obtained using eigenvectors of some graph matrices (for more details see, e.g., [**115**]).

Another applications are related to vertex orderings. The idea is to consider the ordering of vertices induced by ordering the entries of the Fiedler eigenvector. The following nice interpretation of the Fiedler eigenvector explains these applications, namely, it is a solution to the following optimization problem:

$$\text{Minimize } F(x_1, x_2, \ldots, x_n) = \sum_{ij \in E(G)} w_{ij}(x_i - x_j)^2$$

$$\text{Subject to: } x_1 + x_2 + \cdots + x_n = 0,\ x_1^2 + x_2^2 + \cdots + x_n^2 = 1.$$

This problem can be viewed as imbedding a weighted graph G (w_{ij} is a weight of an edge ij), in the real line so that the weighted sum of the squares of all edges is minimal. This optimization problem is equivalent to the description of the second smallest Laplacian eigenvalue through the Rayleigh quotient. Based on this many heuristics are proposed for sorting rows/columns of symmetric sparse matrices which gives rise to some nice patterns of the matrices, which are useful not only in visualization of these matrices, but also in numerical calculations. Some of the related problems are *bandwidth, profile* or *envelope size, work bound*, just to mention only a few of them (needles to say, they are NP-hard). These problems are

essentially discrete ones (belong to combinatorial optimization) but when relaxed to continuous counterparts give rise to minimization/maximization of the Rayleigh quotient, and therefore some of the Laplacian eigenvalues are encountered, including the algebraic connectivity. Further related details on these problems can be found, say in [**83, 84, 85, 104, 105**]. For some applications related to Internet that concern the reordering of *term-by-document matrix* (or *hypertext matrix*) one can use the same heuristics as for the envelope reduction problem (see [**6**]).

As is told in Subsection 3.3, in regular graphs the eigenvalue λ_2 corresponds to the algebraic connectivity ν_2 with the same (Fiedler) eigenvector. However, the eigenvector $\mathbf{z}$ of λ_2 shows similar properties also in non-regular graphs. For example, in [**58**], the authors quote the example of a random graph on 600 vertices, where the sign pattern of the eigenvector of λ_2 gives rise to a bisection of high quality, without clear theoretical explanation. In this situation, the following theorem of FIEDLER becomes relevant (see [**55**], or [**39**] p. 219): namely, subgraphs induced by vertices with *non-negative* and *non-positive entries* of $\mathbf{z}$ are connected (see also [**4**]).

4.3. Other eigenvectors

Let G be a connected graph on n vertices. Eigenvalues in non-decreasing order and corresponding orthonormal eigenvectors of the Laplacian $L = D - A$ of G are denoted by $\nu_1 = 0, \nu_2, \ldots, \nu_n$ and $u_1, u_2, \ldots, u_n$, respectively.

Let us form now an $n \times k$ matrix U containing the vectors $u_1, u_2, \ldots, u_k$ as columns. In this way we have constructed a geometric representation $\mathcal{G}$ of G in the k-dimensional space $\mathbf{R}^k$: we just take rows of U as point coordinates representing the vertices of G. Edges are straight line segments between the corresponding points.

The sum of squares of lengths of all edges in the representation $\mathcal{G}$ of G is equal to $\nu_1 + \nu_2 + \cdots + \nu_k$, and this is a minimal value over all representations obtained via matrix U with orthonormal columns, as noted in the literature (see, e.g., [**74**]).

It should be expected that such an extremal graph representation must have remarkable properties. It is used in data clustering assuming that k is the number of clusters, given in advance. In particular, this representation enhance the cluster-properties of the original data and clusters can now be easily detected. Classical clustering methods (say k-means algorithm) should be applied to this new graph presentation.

The number of clusters k is not always given in advance. There are methods to determine k: using the so called *eigengap* (extremal difference between successive Laplacan eigenvalues) [**71**] and, again, using eigenvectors [**119**].

The results obtained by this and similar spectral clustering algorithms are very good and popular among researchers. However, these algorithms are not completely theoretically explained and understood.

Very frequently instead of the Laplacian we can consider matrices of weighted graphs. In particular, one can use a Laplacian-like matrix formed from the affinity matrix of a set of objects.

Graph representation obtained by the Laplacian matrix has been used in graph drawings [**74, 62, 107, 56**]. Now the coordinates of vertices are determined by the corresponding two, or three, entries of eigenvectors depending of dimension of space in which graph is drawn.

Together with the Laplacian L and the normalized Laplacian $\hat{L}$ also the matrix $D^{-1}L$ has been used in clustering algorithms. According to [**80**] the last matrix performs best.

We offer some observations supporting the usefulness of the described clustering procedure.

Theorem 4.2. *Let G be a connected graph on n vertices with Laplacian L. Suppose that the multiplicity of eigenvalue 0 is equal to k and let U be an $n \times k$ matrix whose columns are independent eigenvectors of 0. Then G has k components and their vertices are uniquely determined by rows of U in the sense that the vertices corresponding to identic rows are in the same component while two vertices corresponding to non-identic rows are in different components.*

Proof. It is well known that the number of components is equal to the multiplicity of eigenvalue 0. Define characteristic vector of a component as an n-vector having components equal to 1 for vertices in the component and equal to 0 otherwise. Characteristic vectors of components form a basis of the eigenspace of 0. Hence, any eigenvector of 0, being a linear combination of the characteristic vectors of components, has the same value of the components corresponding to vertices of any fixed component. Also, the rows of U, whose rank is equal to k, are mutually equal for vertices in a component. However, distinct rows of U correspond to vertices from different components for otherwise U would have less than k distinct rows and its rank would be less than k. □

A similar theorem can be formulated for the A-theory for regular graphs. Determining components for non-regular graphs requires the knowledge of all eigenvectors of the graph.

Spectral clustering can be performed using the modularity matrix, introduced in Section 2 (see paper by M. E. J. Newman [**86**]).

We shall mention a couple of other things.

An image, for example, of 100×100 pixels, each having 256 colour levels, can be represented as a point in Euclidean space of dimension 2560000. However, images in reality usually are contained in a subspace of much lower dimension. Several authors have explored the use of eigenvectors of the Laplacian and related operators to map data to a manifold in a low dimensional space [**92, 10**]. These maps are similar as those described in graph clustering.

Spectral filtering is an important method in handling huge sets of data. This method uses the eigenvectors of the adjacency and other graph matrices to find some clusters in data sets represented by graphs. For example, in [**58**] spectral filtering is applied in the study of Internet structure. The method uses adjacency matrix, its $k < n$ largest eigenvalues and corresponding eigenvectors. The method

of [**58**] has been critically commented in [**94**]. However, theoretical justification of the method can be found in [**18**].

5. SPECTRAL RECOGNITION PROBLEMS

As already pointed out, the benefit of using graph spectra in treating graphs is that eigenvalues and eigenvectors of several graph matrices can be quickly computed. Spectral graph parameters contain a lot of information on the graph structure (both global and local) including some information on graph parameters that, in general, are computed by exponential algorithms.

Moreover, in some applications in data mining graph spectra are used to encode graphs themselves (see, e.g., [**48, 120**]).

At some time it was conjectured that non-isomorphic graphs have different spectra, i.e. that graphs are characterized by their spectra. Very quickly this conjecture was refuted and numerous examples and families of non-isomorphic graphs with the same spectrum were found. In particular, it was proved that almost all trees are not characterized by their spectra. Analogous question for general graphs remained open (see, e.g., [**33**], Section 6.1, for a survey on these questions).

Also in Chemistry there was a criticism on using graph eigenvalues to characterize molecules [**63**].

Graphs with the same spectrum of an associated matrix M are called *cospectral* graphs with respect to M, or *M-cospectral* graphs.

The existence of cospectral graphs is not considered as a disadvantage in using graph spectra in Computer Science since it is believed that graph spectra contain enough information for the purposes for which they are used.

The following example is illustrative in this respect. The indexing structure of objects appearing in computer vision (and in a wide range of other domains such as linguistics and computational biology) may take the form of a tree. An indexing mechanism that maps the structure of a tree into a low-dimensional vector space using graph eigenvalues is developed in [**98**].

To clarify recent developments we need some definitions.

A graph H cospectral with a graph G, but not isomorphic to G, is called a *cospectral mate* of G. Let $\mathcal{G}$ be a finite set of graphs, and let $\mathcal{G}'$ be the set of graphs in $\mathcal{G}$ which have a cospectral mate in $\mathcal{G}$ with respect to M. The ratio $|\mathcal{G}'|/|\mathcal{G}|$ is called the *spectral uncertainty* of (graphs from) $\mathcal{G}$ with respect to M (or, in general, *spectral uncertainty* of the M-theory).

The papers [**46, 61**] provide spectral uncertainties r_n with respect to the adjacency matrix A, s_n with respect to the Laplacian L and q_n with respect to the signless Laplacian Q of sets of all graphs on n vertices for $n \leq 11$ (see [**17**] for $n = 12$):

n	4	5	6	7	8	9	10	11	12
r_n	0	0.059	0.064	0.105	0.139	0.186	0.213	0.211	0.188
s_n	0	0	0.026	0.125	0.143	0.155	0.118	0.090	0.060
q_n	0.182	0.118	0.103	0.098	0.097	0.069	0.053	0.038	0.027

We see that the sequences s_n and q_n are decreasing for $n \leq 12$ while the sequence r_n is increasing for $n \leq 10$. Yet, it starts to decrease for $n > 10$. This is a strong basis for believing that almost all graphs are determined by their spectra when n tends towards the infinity, as conjectured in [**46, 61**]. The proof of this conjecture would strengthen the theory of graph spectra and, in particular, its application to computer sciences.

Having in view the above data, the L-spectrum is used to encode graphs rather than A-spectrum, i.e. the L-spectrum has more representational power than the A-spectrum, in terms of resulting in fewer cospectral graphs. The above data show that it is even better to use signless Laplacian eigenvalue since they have stronger characterization properties.

There are many results in the mathematical literature on spectral characterizations of particular classes of graphs (see, e.g., Chapter 4 of [**41**]. However, these results hardly could be applied to graphs which appear as complex networks. It is therefore of interest to construct or generate a graph starting from its spectrum. An algorithm for such a reconstruction has been developed in [**25**]. Given the spectrum of a graph, the algorithm starts from a random graph and uses the tabu search to diminish a certain spectral distance (see next subsection) between the given and current spectrum.

5.1. The spectral distance and similarity of graphs

The Euclidean distance between the eigenvalue sequences of two graphs on the same number of vertices is called the *spectral distance* of graphs. Some other spectral distances have been considered as well.

Two graphs are considered as *similar* if their spectral distance is small. (Two graphs can be similar without corresponding matrices being similar.) If two graphs are at zero distance, this does not necessarily mean that they are equal (i.e. isomorphic); they are only cospectral. In this sense, cospectral graphs are similar. A spectrally based measure of similarity between networks has been introduced in [**53**], and applied to Internet topology analysis.

In some cases researchers feel that the spectrum very well characterizes the graphs under consideration so that the spectrum is considered as a *fingerprint* of the corresponding network. The eigenvalues $\gamma_i;\ i = 1, 2, \ldots, n$ of $\hat{L}$ in non-decreasing order can be represented by points $\left(\frac{i-1}{n-1}, \gamma_i\right)$ in the region $[0, 1] \times [0, 2]$ and can be approximated by a continuous curve. It was noticed in [**110, 111**] that this curve is practically the same during the time for several networks in spite of the increasing number of vertices and edges of the corresponding graph.

5.2. Interlacing theorem and spectra of subgraphs

The following theorem is very important in spectral graph theory and its applications. (Recall that the matrix A with complex entries a_{ij} is called *Hermitian* if $A^T = \overline{A}$, i.e. $a_{ji} = \overline{a}_{ij}$ for all i, j.)

Theorem 5.1. (see, e.g., [**33**], p. 19) Let A be a Hermitian matrix with eigenvalues

$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ and let B be one of its principal submatrices. If the eigenvalues of B are $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_m$ then $\lambda_{n-m+1} \leq \mu_i \leq \lambda_i$ $(i = 1, \ldots, m)$

The inequalities of this theorem are known as *Cauchy's inequalities* and the whole theorem is known as the *Interlacing Theorem*. It is used frequently as a spectral technique in graph theory.

In several databases the data are often represented as graphs. Very frequently graphs are indexed by their spectra.

In [**88**] a spectral graph theory approach is presented for representing melodies as graphs, based on intervals between the notes they are composed of. These graphs are then indexed using their Laplacian spectrum. This makes it possible to find melodies similar to a given melody.

The query for such a database is given by a graph. To find similar data in the database it is necessary to compare subgraphs of the query graph with subgraphs of the graphs stored in the database. One should efficiently select a small set of database graphs, which share a subgraph with the query. Instead of comparing subgraphs one can compare their spectra. In this situation the interlacing theorem is often an effective tool in pruning the search.

Note that the subgraph isomorphism problem is NP-complete while comparing spectra can be done in polynomial time.

To accelerate the process of computing spectra of subgraphs the *spectral integral variation technique* is used in [**48**]. First, we have the following variation of the interlacing theorem for L-spectra.

Theorem 5.2. *Let G be a connected graph on n vertices. Eigenvalues in non-decreasing order of the Laplacian $L = D - A$ of G are denoted by $\nu_1 = 0, \nu_2, \ldots, \nu_n$. Let G' be obtained from G by adding an edge and let $\sigma_1 = 0, \sigma_2, \ldots, \sigma_n$ be L-eigenvalues of G'. Then*

$$0 = \nu_1 = \sigma_1 \leq \nu_2 \leq \sigma_2 \leq \cdots \leq \nu_n \leq \sigma_n.$$

The proof is obtained using well-known Courant-Weyl inequalities (see, e.g., [**33**], pp. 51–52).

Hence, when adding an edge the L-eigenvalues do not decrease. However, the sum of L-eigenvalues increases by 2. We are interested in the case when L-eigenvalues change only by integer quantities. Evidently there are just two possible scenarios [**103, 52**] where that can happen: either one eigenvalue will increase by 2 (and $n - 1$ eigenvalues remain unchanged) or two eigenvalues will increase by 1 (and $n - 2$ eigenvalues remain unchanged). Precise conditions when each of these two cases of spectral integral variation technique occurs are given in the literature [**67, 103**].

5.3. Structural and spectral perturbations of graphs

The spectral integral variation technique, described in the last subsection, is just an example involving *graph perturbations*. A graph perturbation means a small

change in graph structure (e.g., adding an edge or a vertex). We are interested in changes in graph eigenvalues caused by a perturbation.

There is a chapter in the book [**39**] devoted to graph perturbations and corresponding changes in the spectrum.

The problem of protecting the privacy appears in *social networks* on the Internet (for example, Facebook) when studying general properties of an existing network. A way to protect the privacy of personal data is to randomize the network representing relations between individuals by deleting some actual edges and by adding some additional edges in such a way that the global characteristics of the network are unchanged. This is achieved by using eigenvalues of the adjacency matrix (in particular, the largest one) and of the Laplacian (algebraic connectivity) to control the process of deleting and adding the edges [**118**]. The choice of deleted and added edges is performed by using results of [**39**], Chapter 6, for the largest eigenvalue and the corresponding results for the algebraic connectivity have been derived in the paper.

In Computer science literature, some spectral perturbations of graphs have been considered as well. This means that the graph spectrum is slightly changed while the eigenvectors remain unchanged. This is used in connection with the formula for spectral decomposition of the adjacency matrix A of a graph, i.e. $A = U\Lambda U^T$, where Λ is a diagonal matrix containing the eigenvalues of A and the columns of matrix U are orthonormal eigenvectors of A. The paper [**77**] proposes a new robustness parameter for complex networks: this is the maximal number k such that one can replace k smallest in modulus eigenvalues of A with zeros with the possibility that A still can be reconstructed.

A similar "deletion" of eigenvalues appears also in the so called *latent semantic indexing* (LSI) but it is applied on singular values of the term-by-document matrix (see, e.g., [**87, 102**]).

6. SPECTRA OF RANDOM GRAPHS

Complex networks is a common name for various real networks which are presented by graphs with an enormously great number of vertices. Here belong Internet graphs, phone graphs, e-mail graphs, social networks and many others. In spite of their diversity such networks share some common properties.

A very frequent characteristic of a complex networks (both real and theoretical) is the degree and eigenvalue distribution. Both distributions obey a *power law* of the form $x^{-\beta}$ for a positive constant β. A network with power law distributions is called *scale-free*.

In particular, if n_k denotes the number of vertices of degree k, then asymptotically $n_k = ak^{-\beta}$ for some constant a. The power law for eigenvalues can be formulated in the following way. Let $\lambda_1, \lambda_2, \ldots$ be non-increasing sequence of eigenvalues of the adjacency matrix, then asymptotically $\lambda_k = ak^{-\gamma}$ for some constant a and positive γ.

It was conjectured in [**51**] that in networks with degree power law the largest

eigenvalues of the adjacency matrix also have a power law distribution. That was proved under some conditions in [**81**]. See [**99**] for analysis of some empirical data.

Several models of random graphs have been used to describe complex networks. One is the classical Erdös-Rényi model [**50**] where we have a constant probability for the existence of each edge. There are models where given degree distribution is realized [**5**]. Many real networks obey the *small-word model* introduced in [**113**]. An asymptotic distribution of eigenvalues of certain random symmetric matrices, known as *Wigner's semi-circle law*, has been derived in [**116**].

The asymptotic behaviour of algebraic connectivity for random graphs in Erdös-Rényi model has been derived in [**65**]. An approximate expression for algebraic connectivity of complex networks in the same model has been obtained in [**64**]. It was suggested that the algebraic connectivity should be taken as the measure of the robustness of complex networks.

There is a section on spectra of random graphs in [**32**], where early results have been described. The book [**24**] is devoted to complex networks. There are two chapters which describe spectral properties of such networks. The book [**108**] describes how graph spectra are used in complex networks. See also [**23, 57**].

7. MISCELLANEOUS TOPICS

In separate subsections we treat the Hoffman polynomial, integral graphs and graph divisors.

7.1. The Hoffman polynomial

Let G be a connected graph on n vertices with Laplacian $L = D - A$. Suppose that G has distinct Laplacian eigenvalues $\mu_1 = 0, \mu_2, \ldots, \mu_m$ with multiplicities $k_1 = 1, k_2, \ldots, k_m$, respectively.

Let j be all-1 vector and J a square all-1 matrix.

Theorem 7.1. *Let G be a connected graph on n vertices with Laplacian L and distinct Laplacian eigenvalues $\mu_1 = 0, \mu_2, \ldots, \mu_m$. Let $h(x) = (x - \mu_2) \cdots (x - \mu_m)$. Then $h(L) = aJ$ where $a = (-1)^{m-1}\mu_2 \cdots \mu_m/n$.*

Proof. Since L is a symmetric matrix its minimal polynomial $m(x)$ has the form $m(x) = xh(x)$ and we have $Lh(L) = O$. Suppose that $h(L) = M$ and consider a column u of M. We have $Lu = 0$. Let C be the vertex-arc incidence matrix of the digraph obtained from G by introducing any orientation of edges of G. It is well-known that $L = CC^T$ and from $Lu = 0$ we get $CC^Tu = 0$ and $C^Tu = 0$. Hence, for any edge e of G coordinates of u corresponding to the end-vertices of e are mutually equal. Since G is connected all coordinates of u are mutually equal. Since M is symmetric all its columns are mutually equal. Hence, $M = aJ$, i.e. $h(L) = aJ$ for some constant a. The trace of aJ is an and the trace of $h(L)$ is $(-1)^{m-1}\mu_2 \cdots \mu_m$ and the proof is completed.

Remark. If we introduce

$$H(x) = \frac{(-1)^{(m-1)}n}{\mu_2 \cdots \mu_m} h(x) = \frac{(-1)^{(m-1)}n}{\mu_2 \cdots \mu_m}(x - \mu_2) \ldots (x - \mu_m)$$

we get $H(L) = J$. The polynomial $H(x)$ is called the *Laplacian Hoffman polynomial* of G analogous to the *Hoffman polynomial* originally introduced for the adjacency matrix and known to exist only for regular connected graphs.

If G is regular of degree r with distinct A-eigenvalues $\sigma_1 = r, \sigma_2, \ldots, \sigma_m$ we have $L = rI - A$ and $\mu_i = r - \sigma_i,\ i = 2, \ldots, m$. The relation $H(L) = J$ is then transformed into

$$\frac{n}{(r-\sigma_2)\cdots(r-\sigma_m)}(A-\sigma_2 I)\cdots(A-\sigma_m I) = J.$$

The polynomial

$$H_A(x) = \frac{n}{(r-\sigma_2)\cdots(r-\sigma_m)}(x-\sigma_2)\cdots(x-\sigma_m)$$

is called the *Hoffman polynomial*. It is proved that for a polynomial $P(x)$ the relation $P(A) = J$ holds if and only if G is regular and connected and in this case the only such polynomial is the *Hoffman polynomial* $H_A(x)$ (see, e.g., [**33**], p. 95).

Our result is relevant for load balancing in multiprocessor systems.

The job which has to be executed by a multiprocessor system is divided into parts that are given to particular processors to handle them. Elementary jobs distribution among processors can be represented by a vector $\mathbf{x}$ whose coordinates are non-negative integers. Of course, it would be optimal that the number of elementary jobs given to a processor is the same for all processors, i.e., that the vector $\mathbf{x}$ is an integer multiple of the vector $\mathbf{j}$ whose coordinates are all equal to 1. However, vector $\mathbf{x}$ is usually changed during the work of the system because some elementary jobs are executed while new elementary jobs are being permanently generated during the execution process. Therefore it is reasonable that processors with a great number of elementary jobs send some of them to adjacent processors so that the job distribution becomes as uniform as possible. In this way the so called problem of *load balancing* is important in managing multiprocessor systems.

We shall present a known algorithm for the load balancing problem in our interpretation using the Laplacian Hoffman polynomial.

We have $H(L)\mathbf{x} = J\mathbf{x} = \beta\mathbf{j}$, where β is the sum of the coordinates of $\mathbf{x}$. If $\mathbf{x}$ represents any job distribution the matrix $\frac{1}{n}H(L)$ transforms it into a uniform distribution. We can write

$$\frac{1}{n}H(L) = (I - \frac{1}{\mu_2}L)\cdots(I - \frac{1}{\mu_m}L)$$

Introducing vectors $\mathbf{x}^{(1)} = \mathbf{x}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(m)}$ by relations

$$\mathbf{x}^{(k)} = \Big(I - \frac{1}{\mu_k}L\Big)\mathbf{x}^{(k-1)}, \quad k = 2, \ldots, m \tag{1}$$

we shall obtain $\mathbf{x}^{(m)} = \frac{\beta}{n}\mathbf{j}$.

The transformation $I - \frac{1}{\mu_k}L$ will cause that the component of $\mathbf{x}^{(k)} = \Big(I - \frac{1}{\mu_k}L\Big)\mathbf{x}^{(k-1)}$ in the eigenspace of μ_k disappears while the component in the eigenspace of 0 remains unchanged.

We have seen how a vector $\mathbf{x}$ can be transformed to a scalar multiple of $\mathbf{j}$ using the iteration process (1), which involves the Laplacian matrix of the multiprocessor graph G. It remains to be seen what relations (1) mean in terms of load moving.

Let vector $\mathbf{x}^{(k)}$ have coordinates $x_1^{(k)}, x_2^{(k)}, \ldots, x_n^{(k)}$. Relations (1) can be rewritten in the form

$$x_i^{(k)} = x_i^{(k-1)} - \frac{1}{\mu_k} \sum_{i*j} \left(x_i^{(k-1)} - x_j^{(k-1)} \right), \tag{2}$$

where d_i is the degree of vertex i. This means that the current load at vertex i is changed in such a way that vertex (processor) i sends the $\frac{1}{\mu_k}$-th part of its load to each of its d_i neighbors and, because this holds for every vertex, also receives the $\frac{1}{\mu_k}$-th part of the load from each of its d_i neighbors. The amounts should be added algebraically and in this way we get the final value of the flow through edge ij (sent either from i to j or vice versa). The obtained flow is ℓ_2-optimal, i.e., the sum of the squares of particular edge flows is minimal (see, for e.g., [**49**]).

7.2. Integral graphs

A graph is called *integral* if its spectrum consists entirely of integers. Each eigenvalue has integral eigenvectors and each eigenspace has a basis consisting of such eigenvectors.

Integral graphs have been studied for decades as a kind of mathematical curiosity without any idea of what they could be used for outside of mathematics (see [**3**] for a survey of results up to 2002). In particular, there are exactly 13 connected, cubic, integral graphs [**19**]. Among them are, for example, the 3-dimensional cube and the Petersen graph.

It has been discovered recently [**22**] that A-integral graphs can play a role in the so called *perfect state transfer* in quantum spin networks of quantum computing. (Quantum computation is a model of computation based on the principles of quantum mechanics although the corresponding computers have not yet been realized [**90**].) Speaking in terms of quantum physics, there is perfect state transfer between two vertices of a graph if a single excitation can travel with fidelity one between the corresponding sites of a spin system modelled by the graph [**96**]. L-integral graphs also appear in the discussions of perfect state transfer [**69**].

Let G be a graph with adjacency matrix A and consider the matrix $H(t) = e^{iAt}$ where t is a real variable and $i^2 = -1$. According to [**59**], perfect state transfer occurs between vertices u and v of G if there is a value of t such that $|H(t)_{u,v}| = 1$. This can happen in integral graphs, but does not always.

Further details on this topic can be found in [**95, 59, 60, 96, 106**].

The 3-dimensional cube is the only connected cubic integral graph with perfect state transfer [**96**]. Some other results in this direction have been obtained in [**8, 9**].

Integral graphs are of interest in constructing multiprocessor interconnection networks.

In integral graphs load balancing algorithms (see previous subsection), which use eigenvalues and eigenvectors, can be executed in integer arithmetics as noted in [**28**]. In addition, these integral eigenvectors can be selected so that they contain a lot of coordinates equal to 0 [**30**].

Any load distribution vector can, of course, be represented as a linear combination of eigenvectors of a graph. We can introduce several load balancing schemes on the set of selected integral eigenvectors. At this step we can introduce various additional optimality criteria. For more details see [**30**].

The further study of integral graphs in connection to multiprocessor topologies seems to be a promising subject for future research.

7.3. Graph divisors

First we quote the following result.

Theorem 7.2. ([**33**], p. 20) *Let A be any matrix partitioned into blocks with square blocks on the main diagonal. Let the block A_{ij} have constant row sums b_{ij} and let $B = (b_{ij})$. Then the spectrum of B is contained in the spectrum of A (having in view also the multiplicities of the eigenvalues).*

The content of this theorem justifies the introduction of the following definition.

Definition. *Given an $s \times s$ matrix $B = (b_{ij})$, let the vertex set of a graph G be partitioned into (non-empty) subsets $X_1, X_2, \ldots, X_s$ so that for any $i, j = 1, 2, \ldots, s$ each vertex from X_i is adjacent to exactly b_{ij} vertices of X_j. The multidigraph H with adjacency matrix B is called a front divisor of G, or briefly, a divisor of G.*

The existence of a divisor means that the graph has a certain structure; indeed, a divisor can be interpreted as a homomorphic image of the graph. On the other hand, by the above theorem, the characteristic polynomial of a divisor divides the characteristic polynomial of the graph (i.e. the spectrum of a divisor is contained in the spectrum of the graph). In this way the notion of a divisor can be seen as a link between spectral and structural properties of a graph.

Divisors have been considered in the literature also under the name *equitable partitions*.

The concept of a divisor has also featured in coding theory. As an application of the divisor concept in this field we shall outline an elementary proof of *Lloyd's Theorem* due to Cvetković and van Lint [**37**]. For the general concepts of coding theory, see, e.g., [**76**] (with Lloyd's Theorem on p. 111).

We need some preparations. Consider a set $\mathcal{F}$ of b distinct symbols which we call the *alphabet*. The elements of $\mathcal{F}^n$ will be called *words of length n*. In $\mathcal{F}^n$ the *Hamming distance d* is defined by

$$d(\mathbf{x}, \mathbf{y}) = |\{i : x_i \neq y_i, 1 \leq i \leq n\}|.$$

A subset $\mathcal{S}$ of $\mathcal{F}^n$ is called a *perfect e-code* if $\mathcal{F}^n$ is partitioned by the spheres $\mathcal{S}_e(\mathbf{c})$

($\mathbf{c} \in \mathcal{S}$), where

$$\mathcal{S}_e(\mathbf{c}) := \{\mathbf{x} \in \mathcal{F}^n : d(\mathbf{x}, \mathbf{c}) \leq e\}.$$

In 1957 Lloyd [**78**] proved a strong necessary condition for the existence of a perfect e-code when $b = 2$ (the binary case). In the years since 1972 several authors (see [**33**], p. 131) have proved that the theorem (always referred to as Lloyd's Theorem) holds for all b:

Theorem 7.3. (Lloyd's Theorem) *If a perfect e-code of length n over an alphabet of b symbols exists, then the e zeros x_i of the polynomial*

$$\phi_{enb}(x) := \sum_{i=0}^{e} (-1)^i (b-1)^{e-i} \binom{n-x}{e-i} \binom{x-1}{i}$$

are distinct positive integers $\leq n$.

A proof using spectral graph theory exploits the fact that a perfect code determines a divisor in the graph in which vertices are words of length n two words being adjacent if they are at Hamming distance 1.

The notion of a divisor, i.e. equitable partition, is also used in control theory [**91, 42**] and in social network analysis [**12**].

8. CONCLUSION

From the presented material one can see that a great part of the theory of graph spectra is really used in computer sciences.

Through various applications in computer sciences it becomes clear that spectral graph theory is by no means bounded to a particular graph matrix, such as adjacency matrix or Laplacian. A great variety of graph matrices are used depending on the problem treated.

Due to enormous number of scientific papers in computer sciences which use graph spectral techniques and due to various fields where these techniques are applied, it is really difficult to produce a balanced and comprehensive survey.

For decades graph theory was just a collection of weakly interrelated subtheories (chromatic graph theory, metrical problems, trees, planar graphs, etc.). The theory of graph spectra contains tools which can be applied to all these subtheories, although with varying strength, and one can think of it as being a unifying theory for the whole graph theory. However, spectral techniques are weak for some problems and mathematicians could reasonably hold doubt in such a possible conclusion.

In applications to computer sciences spectral graph theory is considered as very strong and perhaps one can say that its unifying mission for graph theory has been realized through Computer Science.

Acknowledgement. The authors are grateful to the referees for the many useful comments and suggestions which have led to an improvement of this paper. Sup-

ported by the Serbian Ministry for Education and Science, Grants ON174033 and III44006.

REFERENCES

1. N. M. M. de Abreu: *Old and new results on algebraic connectivity of graphs.* Linear Algebra Appl., **423** (2007), 53–73.
2. N. Alon: *Eigenvalues and expanders.* Combinatorica, **6** (1986), 83–96.
3. K. Balińska, D. Cvetković, Z. Radosavljević, S. Simić, D. Stevanović: *A survey on integral graphs.* Univ. Beograd, Publ. Elektrotehn. Fak., Ser. Mat., **13** (2002), 42–65; Errata, *These Publications*, **15** (2004), 112.
4. R. Bapat, S. Pati, S. Kirkland: *The Perturbed Laplacian Matrix of a Graph.* Linear Multilinear Algebra, **49** (2001), 219-242.
5. A. L. Barabási, R. Albert: *Emergence of scaling in random networks.* Science, **286** (1999), 509-512.
6. S. T. Barnard, A. Pothen, H. Simon: *A spectral algorithm for envelope reduction of sparse matrices.* Numer. Linear Algebra Appl., **2** (4) (1995), 317–334.
7. S. Barnard, H. Simon: *Fast implementation of recursive spectral bisection for partitioning unstructured problems.* Concurrency: Practice and Experiments, **6** (2) (1994), 101–107.
8. M. Bašić, M. Petković: *Some classes of integral circulant graphs allowing and not allowing perfect state transfer.* Appl. Math. Lett., **22** (2009), 1609–1615.
9. M. Bašić, M. Petković, D. Stevanović: *Perfect state transfer in integral circulant graphs.* Appl. Math. Lett., **22** (2009), 1117–1121.
10. M. Belkin, P. Niyogi: *Laplacian eigenmaps for dimensionality reduction and data representation.* Neural Comput., **15** (2000), 1373–1396.
11. P. Bonacich: *Power and centrality: A family of measures.* Amer. J. Soc., **92** (1987), 1170–1182.
12. U. Brandes, J. Lerner: *Structural similarity: spectral methods for relaxed blockmodeling.* J. Classification, **27** (2010), 297–306.
13. L. Branković: *Usability of secure statistical data bases*, PhD Thesis, Newcastle, Australia, 1998.
14. Lj. Branković, D. Cvetković: *The eigenspace of the eigenvalue -2 in generalized line graphs and a problem in security of statistical data bases.* Univ. Beograd, Publ. Elektrotehn. Fak., Ser. Mat., **14** (2003), 37–48.
15. L. Branković, M. Miller, J. Širáň: *Graphs,* $(0,1)$*-matrices and usability of statistical data bases.* Congr. Numer., **120** (1996), 186–192.
16. S. Brin, L. Page: *The Anatomy of Large-Scale Hypertextual Web Search Engine.* Proc. 7th International WWW Conference, 1998; Computer Networks and ISDN Systems, **30** (1998), 107–117.
17. A. E. Brouwer, E. Spence: *Cospectral graphs on* 12 *vertices.* Electron. J. Combin., **16** (2009), #20.

18. D. Bu, Y. Zhao, L. Cai, H. Xue, X. Zhu, H. Lu, J. Zhang, S. Sun, L. Ling, N. Zhang, G. Li, R. Chen: *Topological structure analysis of the protein-protein interaction network in budding yeast.* Nucleic Acids Research, **31** (9) (2003), 2443–2450.

19. F. C. Bussemaker, D. Cvetković: *There are exactly* 13 *connected, cubic, integral graphs.* Univ. Beograd, Publ. Elektrotehn. Fak., Ser. Mat. Fiz., **544–576** (1976), 43–48.

20. G. Canright, K. E. Monsen: *Spreading on networks*: *A topographic view.* In Proc. of ECCS, Paris, 2005.; Complexus, **3** (2006), 131–146.

21. I. Carreras, D. Miorandi, G. Canright, K. E. Monsen: *Eigenvector centrality in highly partitioned mobile networks: Principles and Applications.* Stud. Comput. Intell., **69** (2007), 123-145.

22. M. Christandl, N. Datta, A. Ekert, A. J. Landahl: *Perfect state transfer in quantum spin networks.* Phys. Rev. Lett., **92** (2004), 187902.

23. F. Chung: *Spectral Graph Theory.* American Mathematical Society, Providence, Rhode Island, 1997.

24. F. Chung, L. Lu: *Complex Graphs and Networks.* American Mathematical Society, Providence, Rhode Island, 2006.

25. F. Comellas, J. Diaz-Lopez: *Spectral reconstruction of complex networks.* Phys. A, **387** (2008), 6436–6442.

26. D. Cvetković, M. Čangalović, V. Kovačević-Vujčić: *Semidefinite programming methods for the symmetric traveling salesman problem.* In: Integer Programming and Combinatorial Optimization, Proc. 7th Internat. IPCO Conf. Graz, Austria, June 1999 (G. Cornuejols, R. E. Burkard, G. J. Woeginger, eds.), Lecture Notes Comp. Sci., **1610**, Springer, Berlin (1999), 126–136.

27. D. Cvetković, T. Davidović: *Applications of some graph invariants to the analysis of multiprocessor topologies.* YUJOR, **18** (2) (2008), 173–186.

28. D. Cvetković, T. Davidović: *Multiprocessor interconnection networks with small tightness.* Internat. J. Found. Comput. Sci., **20** (5) (2009), 941–963.

29. D. Cvetković, T. Davidović: *Multiprocessor interconnection networks.* Applications of Graph Spectra, Zb. Rad. (Beogr.), **13** (21), ed. D. Cvetković, I. Gutman, Mathematical Institute SANU, Belgrade, 2009, 33–63; improved version in [**36**].

30. D. Cvetković, T. Davidović, A. Ilić, S. K. Simić: *Graphs for small multiprocessor interconnection networks*, Appl. Math. Comput., **217** (2010), 2468–2480.

31. D. Cvetković, V. Dimitrijević, M. Milosavljević: *A survey of some nonstandard traveling salesman problems.* YUJOR, **2** (2) (1992), 163–185.

32. D. Cvetković, M. Doob, I. Gutman, A. Torgašev: *Recent Results in the Theory of Graph Spectra.* North-Holland, Amsterdam, 1988.

33. D. Cvetković, M. Doob, H. Sachs: *Spectra of Graphs, Theory and Application,* 3rd edition. Johann Ambrosius Barth Verlag, Heidelberg-Leipzig, 1995.

34. D. Cvetković, I. Gutman: *Note on branching.* Croat. Chem. Acta, **49** (1977), 115–121.

35. D. Cvetković, I. Gutman, eds: *Applications of Graph Spectra*, Zb. Rad. (Beogr.), **13** (21), Mathematical Institute SANU, Belgrade, 2009.

36. D. Cvetković, I. Gutman, eds: *Selected Topics on Applications of Graph Spectra*, Zb. Rad. (Beogr.), **14** (22), Mathematical Institute SANU, Belgrade, 2011.

37. D. Cvetković, J. H. van Lint: *An elementary proof of Lloyd's theorem.* Proc. Kon. Ned. Akad. V. Wet. A, **80** (1977), 6–10.

38. D. Cvetković, P. Rowlinson: *The largest eigenvalue of a graph – A survey.* Linear Multilinear Algebra, **28** (1990), 3–33.

39. D. Cvetković, P. Rowlinson, S. K. Simić: *Eigenspaces of Graphs.* Cambridge University Press, Cambridge, 1997.

40. D. Cvetković, P. Rowlinson, S. K. Simić: *Spectral Generalizations of Line Graphs, On Graphs with Least Eigenvalue* -2. Cambridge University Press, Cambridge, 2004.

41. D. Cvetković, P. Rowlinson, S. K. Simić: *An Introduction to the Theory of Graph Spectra.* Cambridge University Press, Cambridge, 2009.

42. D. Cvetković, P. Rowlinson, Z. Stanić, M. G. Yoon: *Controllable graphs.* Bull. Acad. Serbe Sci. Arts, Cl. Sci. Math. Natur., Sci. Math. **143** (36) (2011), 81–88.

43. D. Cvetković, P. Rowlinson, Z. Stanić, M. G. Yoon: *Controllable graphs with least eigenvalue at least* -2. Appl. Anal. Discrete Math., **5** (2011), 165–175.

44. D. Cvetković, S. Simić: *The second largest eigenvalue of a graph – A survey.* Filomat, **9** (3) (1995), Int. Conf. Algebra, Logic Discrete Math., Niš, April 14–16, 1995, (ed. S. Bogdanović, M. Ćirić, Ž. Perović), 449–472.

45. D. Cvetković, S. K. Simić: *Graph spectra in computer science.* Linear Algebra Appl., **434** (2011), 1545–1562.

46. E. R. van Dam, W. Haemers: *Which graphs are determined by their spectrum?* Linear Algebra Appl., **373** (2003), 241–272.

47. E. R. van Dam, R. E. Kooij: *The minimal spectral radius of graphs with a given diameter.* Linear Algebra Appl., **423** (2007), 408–419.

48. M. F. Demirci, R. H. van Leuken, R. C. Veltkamp: *Indexing through Laplacian spectra.* Computer Vision and Image Understanding, 2008. doi: 10.1016/j.cviu.2007.09.012.

49. R. Elsässer, R. Královič, B. Monien: *Sparse topologies with small spectrum size.* Theoret. Comput. Sci., **307** (2003), 549–565.

50. P. Erdös, A. Rényi: *On random graphs.* Publ. Mat., **6** (1959), 290–297.

51. M. Faloutsos, P. Faloutsos, C. Faloutsos: *On power-low relationships of the Internet topology.* Proc. ACM SIGCOMM '99, ACM Press, New York, 1999, 251–262.

52. Y. Fan: *On spectral integral variations of graphs.* Linear Multilinear Algebra, **50** (2002), 133–142.

53. D. Fay, H. Haddadi, A. Thomason, R. Mortier, A. Jamakovic, S. Uhlig, M. Rio: *Weighted spectral distribution for internet topology analysis: Theory and applications.* J. IEEE/ACM Trans. Networking (TON), **18** (1) (2010), 164–176.

54. M. Fiedler: *Algebraic connectivity of graphs.* Czechoslovak Math. J., **23** (98) (1973), 298–305.

55. M. Fiedler: *A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory.* Czechoslovak Math. J., **25** (100) (1975), 619–633.

56. P. W. Fowler, D. E. Manolopoulos: *An Atlas of Fullerenes.* Clarendon Press, Oxford, 1995.

57. S. Gago: *Spectral techniques in complex networks. Selected Topics on Applications of Graph Spectra*, Zb. Rad. (Beogr.), ed. D. Cvetković, I. Gutman, Mathematical Institute SANU, Belgrade, 2011, 63–84.

58. C. Gkantsidis, M. Mihail, E. Zegura: *Spectral analysis of internet topologies.* Proc. INFOCOM2003, San Francisco, CA, Vol. **1**, March 2003, 364–374.

59. C. Godsil: *Periodic graphs.* arXiv:0806.2704 [math.CO].

60. C. Godsil: *When can perfect state transfer occur?* arXiv: 1001.0231v1 [math.CO].

61. W. Haemers, E. Spence: *Enumeration of cospectral graphs.* European J. Combin., **25** (2004), 199–211.

62. K. M. Hall: *An r-dimensional quadratic placement algorithm.* Management Sci., **17** (1970), 219–229.

63. E. Heilbronner, T. B. Jones: *Spectral differences between "isospectral" molecules.* J. Amer. Chem. Soc., **100** (20) (1978), 6505–6507.

64. A. Jamakovic, P. Van Mieghem: *On the robustness of complex networks by using the algebraic connectivity.* Networking, A. Das et al Eds., LNCS, **4982** (2008), 183–194.

65. F. Juhás: *The asymptotic behaviour of Fiedler's algebraic connectivity for random graphs.* Discrete Math., **96** (1991), 59–63.

66. S. Kar, J. M. F. Moura: *Ramanujan topologies for decision making in sensor networks.* 44th Annual Allerton Conference on Communications, Control, and Computing, Allerton, Monticello, IL, Sep. 27–29, 2006.

67. S. Kirkland: *A Characterization of Spectral Integral Variation in Two Places for Laplacian Matrices.* Linear Multilinear Algebra, **52** (2004), 79-98.

68. S. Kirkland: *Algebraic Connectivity.* In Handbook of Linear Algebra, L. Hogben, R. Brualdi, A. Greenbaum, R. Mathias, Eds., CRC Press, 2006, 36–1–36–12.

69. S. Kirkland, S. Severini: *Spin Systems Dynamics and Fault Detection in Threshold Networks.* Phys. Rev. (3) A, **83** (2011), # 012310, 5 pages.

70. J. Kleinberg: *Authoratitive sources in a hyperlinked environment.* J. ACM, **48** (1999), 604–632.

71. W. Kong, C. Sun, S. Hu, J. Zhang: *Automatic spectral clustering and its applications.* 2010 Internat. Conf. Intelligent Computation Technology and Automation, 841–845.

72. M. D. König, S. Battiston, M. Napoletano, F. Schweityer: *The efficiency and evolution of R&D networks.* Working Paper 08/95, Economics Working Paper Series, Eidgenössische Technische Hochschule Zürich, Zŕich, 2008.

73. M. D. König, S. Battiston, M. Napoletano, F. Schweityer: *On algebraic graph theory and the dynamics of innovation networks.* Networks and Heterogenous Media, **3** (2) (2008), 201–219.

74. Y. Koren: *Drawing graphs by eigenvectors*: *Theory and practice.* Comput. Math. Appl., **49** (2005), 1867–1888.

75. A. N. Langville, C. D. Meyer: *A survey of eigenvector methods for Web information retrieval.* SIAM Rev., **47** (1) (2005), 135–161.

76. J. H. van Lint: *Coding Theory.* Lecture Notes in Math. 201, Springer-Verlag, Berlin-Heidelberg-New York, 1971.

77. D. Liu, H. Wang, P. Van Mieghem: *Spectral perturbation and reconstructability of complex networks.* Phys. Rev. E, **81** (2010), 016101, 1–9.

78. S. P. Lloyd: *Binary block coding.* Bell System Tech. J., **36** (1957), 517–535.

79. A. Lubotzky, R. Phillips, P. Sarnak: *Ramanujan graphs.* Combinatorica, **8** (1988), 261-277.

80. U. von Luxburg: *A tutorial on spectral clustering.* Stat. Comput., **17** (2007), 395–416.

81. M. Mihail, C. H. Papadimitrou: *On the eigenvalue power-low.* RANDOM 2002, LNCS 2483, Springer, Berlin, 2002, 254–262.

82. B. Mohar: *Isoperimetric number of graphs.* J. Combin. Theory Ser. B, **47** (1989), 274–291.

83. B. Mohar: *The Laplacian spectrum of graphs.* Graph Theory, Combinatorics and Applications (New York) (eds. Y. Alavi et al.), J. Wiley, New York, 1991, 871–898.

84. B. Mohar: *Laplace eigenvalues of graphs – A survey.* Discrete Math. **109** (1992), 171–183.

85. B. Mohar, S. Poljak: *Eigenvalues in combinatorial optimization.* In: *Combinatorial and Graph-Theoretical Problems in Linear Algebra*, (ed. R. Brualdi, S. Friedland, V. Klee), Springer-Verlag, New York, 1993, 107–151.

86. M. E. J. Newman: *Modularity and community structure in networks.* PNAS, **103** (23) (2006), 8577–8582.

87. C. H. Papadimitriou, P. Raghavan, H. Tamaki, S. Vempala: *Latent semantic indexing*: *A probabilistic analysis.* J. Comput. System Sci., **61** (2) (2000), 217–235.

88. A. Pinto: *MIREX*2007 – *Graph spectral method.* To appear.

89. A. Pothen, H. D. Simon, K. P. Liou: *Partitioning sparse matrices with eigenvectors of graphs.* SIAM J. Matrix Anal. Appl., **11** (3) (1990), 430–452.

90. *Quantum Information Technology.* NEC Research & Developments, **44** (3) (2003), 263–306.

91. A. Rahmani, M. Ji, M. Mesbahi, M. Egerstedt: *Controllability of multi-agent systems from a graph-theoretic perspective.* SIAM J. Control. Optim., **48** (1) (2009), 162–186.

92. S. Roweis, L. Saul: *Nonlinear dimensionality reduction by locally linear embedding.* Science, **290** (2000), 2323–2326.

93. S. Sarkar, K. L. Boyer: *Quantitative measures of change based on feature organization: Eigenvalues and eigenvectors.* Computer Vision and Image Understanding, **71** (1998), 110–136.

94. R. Sawilla: *A survey of data mining of graphs using spectral graph theory.* Defence R&D Canada, Ottawa, Technical Memorandum TM 2008-317, Ottawa, 2008.

95. N. Saxena, S. Severini, I. E. Shparlinski: *Parameters of integral circulant graphs and periodic quantum dynamics.* Internat. J. Quantum Inf., **5** (3) (2007), 417–430.

96. S. Severini: *The* 3*-dimensional cube is the only connected cubic graph with perfect state transfer.* arXiv: 1001.0674v2 [quant-ph].

97. J. Shi, J. Malik: *Normalized cuts and image segmentation.* Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1997, 731–737; IEEE Trans. Pattern Analysis Machine Intell., **28** (2000), 888–905.

98. A. Shokoufandeh, S. J. Dickinson, K. Siddiqi, S. W. Zucker: *Indexing using a spectral encoding of topological structure.* IEEE Trans. Comput. Vision Pattern Recognition, **2** (1999), 491–497.

99. G. Siganos, M. Faloutsos, P. Faloutsos, C. Faloutsos: *Power-laws and the AS-level Internet topology.* IEEE/ACM Trans. Networking, **11** (4) (2003), 514–524.

100. A. Silva, P. Reyes, M. Debbah: *Congestion in Randomly Deployed Wireless Ad-Hoc and Sensor Networks*, Proc. Internat. Conf. Ultra Modern Telecommunications St. Petersburg, Russia, October 12–14, 2009, 1–6.

101. S. K. Simić: *Connected graphs of fixed order and size with minimal index.* To appear.

102. D. B. Skilicorn: *Understanding Complex Data Bases: Data Mining with Matrix Decompositions.* Chapman & Hall/CRC, New York, 2007.

103. W. So: *Rank one perturbation and its application to the Laplacian spectrum of a graph*, Linear Multilinear Algebra, **46** (1999), 193-198.

104. D. A. Spielman: *Spectral Graph Theory and its Applications.* 48th Annual IEEE Symposium on Foundations of Computer Science, IEEE, 2007, 29–38.

105. D. A. Spielman, S. Teng: *Spectral partitioning works*: *Planar graphs and finite element meshes.* 37th Annual Symposium on Foundations of Computer Science, Burlington, VT, 1996, 96–105.

106. D. Stevanović: *Applications of graph spectra in quantum physics.* Selected Topics on Applications of Graph Spectra, Zb. Rad. (Beogr.), ed. D. Cvetković, I. Gutman, Mathematical Institute SANU, Belgrade, 2011, 85–112.

107. W. T. Tutte: *How to draw a graph.* Proc. London Math. Soc., **13** (1963), 743–768.

108. P. Van Mieghem: *Graph Spectra for Complex Networks.* Cambridge University Press, Cambridge, 2010.

109. P. Van Mieghem, J. Omić, R. Kooij: *Virus spread in networks.* IEEE/ACM Trans. Networking, **17** (2009), 1–14.

110. D. Vukadinović, P. Huang, T. Erlebach: *A spectral analysis of the Internet topology.* Technical Report 118, TIK Institute, ETH Zürich, 2001.

111. D. Vukadinović, P. Huang, T. Erlebach: *On the spectrum and structure of the Internet topology graphs.* Proc. Second Internat. Workshop on Innovative Internet Computing Systems, IICS '02, **2346** (2002), 83–95.

112. Y. Wang, D. Chakrabarti, C. Wang, C. Faloutsos: *Epidemic spreading in real networks*: *An eigenvalue viewpoint.* 22nd Symp. Reliable Distributed Computing, Florence, Italy, Oct. 6–8, 2003.

113. D. J. Watts, S. H. Strogatz: *Collective dynamics of small-world networks.* Nature, **393** (1998), 440–442.

114. T. H. Wei: *The algebraic foundations of ranking theory.* Thesis, Cambridge, 1952.

115. Y. Weiss: *Segmentation using eigenvectors: A unifying view.* Proc. Seventh IEEE Internat. Conf. Computer Vision, **2** (1999), 975–982.

116. E. P. Wigner: *On the distribution of the roots of certain symmetric matrices.* Ann. Math., **67** (1958), 325–328.

117. R. C. Wilson, E. R. Hancock, B. Luo: *Pattern vectors from algebraic graph theory.* IEEE Trans. Pattern Analysis Machine Intell., 27(2005), 1112–1124.

118. X. Ying, X. Wu: *Randomizing social networks: a spectrum preserving approach.* Proc. SIAM Internat. Conf. Data Mining, SDM2008, April 24–26, 2008, Atlanta, Georgia, USA, SIAM, 2008, 739–750.

119. L. Zelnik-Manor, P. Perona: *Self-tuning spectral clustering.* NIPS, 2004, Adv. Neural Inf. Process. Syst., **17** (2005), 1601–1608.

120. L. Zou, L. Chen, J. X. Yu, Y. Lu: *A novel spectral coding in large graph database.* EDBT'08, March 25–30, 2008, Nantes, France.

Mathematical Institute SANU,
P.O. Box 367,
11000 Belgrade
Serbia
E-mails: brankoarsic@kg.ac.rs
ecvetkod@etf.rs
sksimic@mi.sanu.ac.rs

Computer Science Faculty,
Union University,
11000 Belgrade
Serbia
E-mail: milan.skaric@gmail.com

(Received June 9, 2011)
(Revised December 22, 2011)