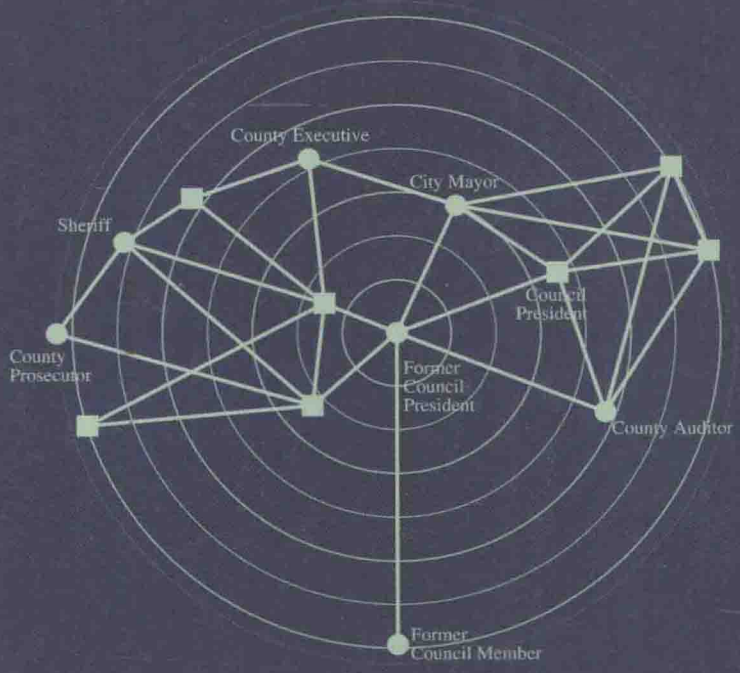
Ulrik Brandes
Thomas Erlebach (Eds.)

Tutorial

LNCS 3418

# Network Analysis

## Methodological Foundations




Springer

Ulrik Brandes Thomas Erlebach (Eds.)

# Network Analysis

## Methodological Foundations



Springer

Volume Editors

Ulrik Brandes
University of Konstanz
Department of Computer and Information Science
Box D 67, 78457 Konstanz, Germany
E-mail: ulrik.brandes@uni-konstanz.de

Thomas Erlebach
University of Leicester
Department of Computer Science
University Road, Leicester, LE1 7RH, U.K.
E-mail: t.erlebach@mcs.le.ac.uk

Library of Congress Control Number: 2005920456

CR Subject Classification (1998): G.2, F.2.2, E.1, G.1, C.2

ISSN 0302-9743
ISBN 3-540-24979-6 Springer Berlin Heidelberg New York



Typesetting: Camera-ready by author, data conversion by Markus Richter, Heidelberg
Printed on acid-free paper SPIN: 11394051 06/3142 5 4 3 2 1 0

# Preface

The present book is the outcome of a seminar organized by the editors, sponsored by the *Gesellschaft für Informatik e.V.* (GI) and held in Dagstuhl, 13–16 April 2004.

GI-Dagstuhl-Seminars are organized on current topics in computer science that are not yet well covered in textbooks. Most importantly, this gives young researchers an opportunity to become actively involved in such topics, and to produce a book that can provide an introduction for others as well.

The participants of this seminar were assigned subtopics on which they did half a year of research prior to the meeting. After a week of presentations and discussion at Schloss Dagstuhl, slightly more than another half-year was spent on writing the chapters. These were cross-reviewed internally and blind-reviewed by external experts. Since we anticipate that readers will come from various disciplines, we would like to emphasize that it is customary in our field to order authors alphabetically.

The intended audience consists of everyone interested in formal aspects of network analysis, though a background in computer science on, roughly, the undergraduate level is assumed. No prior knowledge about network analysis is required. Ideally, this book will be used as an introduction to the field, a reference and a basis for graduate-level courses in applied graph theory.

First and foremost, we would like to thank all participants of the seminar and thus the authors of this book. We were blessed with a focused and determined group of people that worked professionally throughout. We are grateful to the GI and Schloss Dagstuhl for granting us the opportunity to organize the seminar, and we are happy to acknowledge that we were actually talked into doing so by Dorothea Wagner who was then chairing the *GI-Beirat der Universitätsprofessor(inn)en*. We received much appreciated chapter reviews from Vladimir Batagelj, Stephen P. Borgatti, Carter Butts, Petros Drineas, Robert Elsässer, Martin G. Everett, Ove Frank, Seokhee Hong, David Hunter, Sven O. Krumke, Ulrich Meyer, Haiko Müller, Philippa Pattison and Dieter Rautenbach. We thank Barny Martin for proof-reading several chapters and Daniel Fleischer, Martin Hoefer and Christian Pich for preparing the index.

December 2004

*Ulrik Brandes*
*Thomas Erlebach*

# List of Contributors

**Andreas Baltz**
Mathematisches Seminar
Christian-Albrechts-Platz 4
University of Kiel
24118 Kiel, Germany

**Nadine Baumann**
Department of Mathematics
University of Dortmund
44221 Dortmund, Germany

**Michael Baur**
Faculty of Informatics
University of Karlsruhe
Box D 6980
76128 Karlsruhe, Germany

**Marc Benkert**
Faculty of Informatics
University of Karlsruhe
Box D 6980
76128 Karlsruhe, Germany

**Ulrik Brandes**
Computer & Information Science
University of Konstanz
Box D 67
78457 Konstanz, Germany

**Michael Brinkmeier**
Automation & Computer Science
Technical University of Ilmenau
98684 Ilmenau, Germany

**Thomas Erlebach**
Department of Computer Science
University of Leicester
University Road
Leicester LE1 7RH, U.K.

**Marco Gaertler**
Faculty of Informatics
University of Karlsruhe
Box D 6980
76128 Karlsruhe, Germany

**Riko Jacob**
Theoretical Computer Science
Swiss Federal Institute
of Technology Zürich
8092 Zürich, Switzerland

**Frank Kammer**
Theoretical Computer Science
Faculty of Informatics
University of Augsburg
86135 Augsburg, Germany

**Gunnar W. Klau**
Computer Graphics & Algorithms
Vienna University of Technology
1040 Vienna, Austria

**Lasse Kliemann**
Mathematisches Seminar
Christian-Albrechts-Platz 4
University of Kiel
24118 Kiel, Germany

**Dirk Koschützki**
IPK Gatersleben
Corrensstraße 3
06466 Gatersleben, Germany

**Sven Kosub**
Department of Computer Science
Technische Universität München
Boltzmannstraße 3
D-85748 Garching, Germany

**Katharina A. Lehmann**
Wilhelm-Schickard-Institut
für Informatik
Universität Tübingen
Sand 14, C108
72076 Tübingen, Germany

**Jürgen Lerner**
Computer & Information Science
University of Konstanz
Box D 67
78457 Konstanz, Germany

**Marc Nunkesser**
Theoretical Computer Science
Swiss Federal Institute
of Technology Zürich
8092 Zürich, Switzerland

**Leon Peeters**
Theoretical Computer Science
Swiss Federal Institute
of Technology Zürich
8092 Zürich, Switzerland

**Stefan Richter**
Theoretical Computer Science
RWTH Aachen
Ahornstraße 55
52056 aachen, Germany

**Daniel Sawitzki**
Computer Science 2
University of Dortmund
44221 Dortmund, Germany

**Thomas Schank**
Faculty of Informatics
University of Karlsruhe
Box D 6980
76128 Karlsruhe, Germany

**Sebastian Stiller**
Institute of Mathematics
Technische Universität Berlin
10623 Berlin, Germany

**Hanjo Täubig**
Department of Computer Science
Technische Universität München
Boltzmannstraße 3
85748 Garching, Germany

**Dagmar Tenfelde-Podehl**
Department of Mathematics
Technische Universität
Kaiserslautern
67653 Kaiserslautern, Germany

**René Weiskircher**
Computer Graphics & Algorithms
Vienna University of Technology
1040 Vienna, Austria

**Oliver Zlotowski**
Algorithms and Data Structures
Univeristät Trier
54296 Trier, Germany

# Table of Contents

# 1 Introduction

*Ulrik Brandes and Thomas Erlebach*

Many readers will find the title of this book misleading – at least, at first sight. This is because 'network' is a heavily overloaded term used to denote relational data in so vast a number of applications that it is far from surprising that 'network analysis' means different things to different people.

To name but a few examples, 'network analysis' is carried out in areas such as project planning, complex systems, electrical circuits, social networks, transportation systems, communication networks, epidemiology, bioinformatics, hypertext systems, text analysis, bibliometrics, organization theory, genealogical research and event analysis.

Most of these application areas, however, rely on a formal basis that is fairly coherent. While many approaches have been developed in isolation, quite a few have been re-invented several times or proven useful in other contexts as well. It therefore seems adequate to treat network analysis as a field of its own. From a computer science point of view, it might well be subsumed under 'applied graph theory,' since structural and algorithmic aspects of abstract graphs are the prevalent methodological determinants in many applications, no matter which type of networks are being modeled.

There is an especially long tradition of network analysis in the social sciences [228], but a dramatically increased visibility of the field is owed to recent interest of physicists, who discovered the usefulness of methods developed in statistical mechanics for the analysis of large-scale networks [15]. However, there seem to be some fundamental differences in how to approach the topic. For computer scientists and mathematicians a statement like, e.g., the following is somewhat problematic.

> "Also, we follow the hierarchy of values in Western science: an experiment and empirical data are more valuable than an estimate; an estimate is more valuable than an approximate calculation; an approximate calculation is more valuable than a rigorous result." [165, Preface]

Since the focus of this book is on structure theory and methods, the content is organized by level of analysis rather than, e.g., domain of application or formal concept used. If at all, applications are mentioned only for motivation or to explain the origins of a particular method. The following three examples stand in for the wide range of applications and at the same time serve to illustrate what is meant by level of analysis.

U. Brandes and T. Erlebach (Eds.): Network Analysis, LNCS 3418, pp. 1–6, 2005.

**Element-Level Analysis (Google's PageRank)**

Standard Web search engines index large numbers of documents from the Web in order to answer keyword queries by returning documents that appear relevant to the query. Aside from scaling issues due to the incredible, yet still growing size of the Web, the large number of hits (documents containing the required combination of keywords) generated by typical queries poses a serious problem. When results are returned, they are therefore ordered by their relevance with respect to the query.

The success of a search engine is thus crucially dependent on its definition of relevance. Contemporary search engines use a weighted combination of several criteria. Besides straightforward components such as the number, position, and markup of keyword occurrences, their distance and order in the text, or the creation date of the document, a structural measure of relevance employed by market leader Google turned out to be most successful.

Consider the graph consisting of a vertex for each indexed document, and a directed edge from a vertex to another vertex, if the corresponding document contains a hyperlink to the other one. This graph is called the Web graph and represents the link structure of documents on the Web. Since a link corresponds to a referral from one document to another, it embodies the idea that the second document contains relevant information. It is thus reasonable to assume that a document that is often referred to is a relevant document, and even more so, if the referring documents are relevant themselves. Technically, this (structural) relevance of a document is expressed by a positive real number, and the particular definition used by Google [101] is called the PageRank of the document. Figure 1.1 shows the PageRank of documents in a network of some 5,000 Web pages and 15,000 links. Section 3.9.3 contains are more detailed description of PageRank and some close relatives.

Note that the PageRank of a document is completely determined by the structure of (the indexed part of) the Web graph and independent of any query. It is thus an example of a structural vertex index, i.e. an assignment of real numbers to vertices of a graph that is not influenced by anything but the adjacency relation.

Similar valuations of vertices and also of edges of a graph have been proposed in many application domains, and "Which is the most important element?" or, more specifically, "How important is this element?" is the fundamental question in element-level analysis. It is typically addressed using concepts of structural centrality, but while a plethora of definitions have been proposed, no general, comprehensive, and accepted theory is available.

This is precisely what made the organization of the first part of the book most difficult. Together with the authors, the editor's original division into themes and topics was revised substantially towards the end of the seminar from which this book arose. A particular consequence is that subtopics prepared by different participants may now be spread throughout the three chapters. This naturally led to a larger number of authors for each chapter, though potentially with heavily

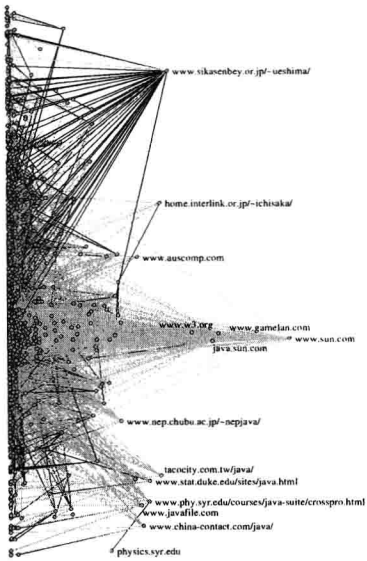



**Fig. 1.1.** PageRank in a network of some 5,000 Web pages containing the keyword 'java' (documents with higher value are further to the right; from [93])

skewed workload. To counterbalance this effect, leading authors are identified in such chapters.

Chapter 3 provides an overview of centrality measures for network elements. The authors have organized the material from a conceptual point of view, which is very different from how it is covered in the literature. Algorithms are rarely discussed in the application-oriented literature, but of central interest in computer science. The underdeveloped field of algorithmic approaches to centrality is therefore reviewed in Chapter 4. Advanced issues related to centrality are treated in Chapter 5. It is remarkable that some of the original contributions contained in this chapter have been developed independently by established researchers [85].

### Group-Level Analysis (Political Ties)

Doreian and Albert [161] is an illustrative example of network analysis on the level of groups. The network in question is made up of influential local politicians and their strong political ties. This is by definition a difficult network to measure, because personal variations in perception and political incentives may affect the outcome of direct questioning. Therefore, not the politicians themselves, but staff members of the local daily newspaper who regularly report on political affairs were asked to provide the data shown in Figure 1.2.

Black nodes represent politicians who are members of the city council and had to vote on the proposed construction of a new jail. The County Executive,

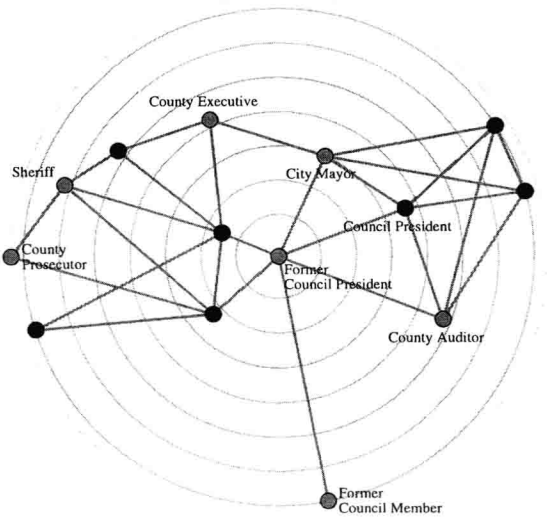



**Fig. 1.2.** Strong political ties between prominent politicians of a county; the two apparent groups predict the voting pattern of City Council members (black nodes) on a crucial issue (data from [161])

who was in favor of building the new jail, and the County Auditor were in strong personal opposition, so that the latter publicly opposed the construction. While the diagram indicates that the former Council President is structurally most important (closeness to the center reflects a vertex index called closeness centrality), it is the group structure which is of interest here.

The voting pattern on the jail issue is predicted precisely by the membership to one of two apparent groups of strong internal bonds. Members of the group containing the County Executive voted for the new jail, and those of the group containing the County Auditor voted against. Note that the entire network is very homogeneous with respect to gender, race, and political affiliation, so that these variables are of no influence.

Note also that two council members in the upper right have ties to exactly the same other actors. Similar patterns of relationships suggest that actors have similar (structural) 'roles' in the network. In fact, the network could roughly be reduced to two internally tied parties that are linked by the former Council President.

Methods for defining and finding groups are treated extensively in the second part of the book. Generally speaking, there are two major perspectives on what constitutes a group in a network, namely strong or similar linkages.

In the first three chapters on group-level analysis, a group is identified by strong linkages among its members. These may be based on relatively heavy induced subgraphs (Chapters 6) or relatively high connectivity between each

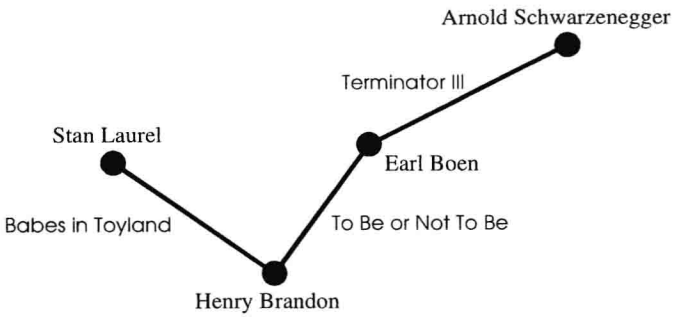



**Fig. 1.3.** Actors appearing jointly (proving that the co-starring distance of S. Laurel and A. Schwarzenegger is no larger than 3)

pair of members (Chapter 7). Methods for splitting a network into groups based on strong linkage are then reviewed in Chapter 8.

Chapters 9 and 10 focus on groups defined by the pattern of relations that members have. While such groups need not be connected at all, strong internal combined with weak external linkage can be seen as a special case.

### Network-Level Analysis (Oracle of Bacon)

Empirical networks representing diverse relations such as linkages among Web pages, gene regulation in primitive organisms, sexual contacts among Swedes, or the power grid of the western United States appear to have, maybe surprisingly, some statistical properties in common.

A very popular example of a network that evolves over time is the movie actor collaboration graph feeding the Oracle of Bacon at Virginia.[1] From all movies stored in the Internet Movie Database[2] it is determined which pairs of actors co-appeared in which movie. The 'Oracle' can be queried to determine (an upper bound on) the co-starring distance of an actor from Kevin Bacon, or in a variant game between any two actors. Except for fun and anecdotal purposes (exemplified in Figure 1.3), actual links between actors are not of primary interest. The fascinating characteristics of this data are on the aggregate level. It turns out, for instance, that Kevin Bacon is on average only three movies apart from any of the more than half a million actors in the database, and that there are more than a thousand actors who have the same property.

Many more properties of this data can be studied. A particularly pertinent observation is, for instance, that in many empirical networks the distribution of at least some statistic obeys a power-law. But the network could also be compared to other empirical networks from related domains (like science collaboration) or fabricated networks for which a suitable model would be required.

[1] `www.oracleofbacon.org`
[2] `www.imdb.com`

The focus of network-level analysis in general is on properties of networks as a whole. These may reflect, e.g., typical or atypical traits relative to an application domain or similarities occuring in networks of entirely different origin.

Network statistics, reviewed in Chapter 11, are a first indicator of network similarity, often employed in complex systems analysis. In Chapter 12, more rigorous methods for detailed structure comparison of equally (or at least comparatively) sized networks are discussed. A different line of research is the attempt to understand the governing principles of network formation. Chapter 13 is therefore devoted to models for networks with certain properties. A particularly powerful approach to global network analysis is the utilization of spectral properties of matrices defined describing the network. These are described in detail in Chapter 14. The final chapter of this book is devoted to the important question of how sensitive a network is to the loss of some of its elements.

Despite the wealth of material covered, the scope of this book is necessarily limited. No matter which personal background, the reader will easily identify gems from the repertoire of network analysis that have been consciously omitted or woefully overlooked. We nevertheless hope that the book will serve as a useful introduction and handy reference for everyone interested in the methods that drive network analysis.

# 2 Fundamentals

*Ulrik Brandes and Thomas Erlebach*

In this chapter we discuss basic terminology and notation for graphs, some fundamental algorithms, and a few other mathematical preliminaries.

We denote the set of integers by $\mathbb{Z}$, the set of real numbers by $\mathbb{R}$, the set of complex numbers by $\mathbb{C}$, and the set of rationals by $\mathbb{Q}$. For a set $X$ of numbers, $X^+$ denotes the subset of positive numbers in $X$, and $X_0^+$ the subset of non-negative numbers. The set of positive integers is denoted by $\mathbb{N} = \mathbb{Z}^+$ and the set of non-negative integers by $\mathbb{N}_0 = \mathbb{Z}_0^+$.

We use $\mathbb{R}^{n\times m}$ to denote the set of all real-valued matrices with $n$ rows and $m$ columns. If the entries of the matrix can be complex numbers, we write $\mathbb{C}^{n\times m}$. The $n$-dimensional identity matrix is denoted by $I_n$. The $n$-dimensional vector with all entries equal to 1 (equal to 0) is denoted by $\mathbf{1}_n$ (by $\mathbf{0}_n$).

For two functions $f : \mathbb{N} \to \mathbb{N}$ and $g : \mathbb{N} \to \mathbb{N}$, we say that $f$ is in $\mathcal{O}(g)$ if there are positive constants $n_0 \in \mathbb{N}$ and $c \in \mathbb{R}^+$ such that $f(n) \leq c \cdot g(n)$ holds for all $n \geq n_0$. Furthermore, we say that $f$ is in $\Omega(g)$ if $g$ is in $\mathcal{O}(f)$. This notation is useful to estimate the asymptotic growth of functions. In particular, running-times of algorithms are usually specified using this notation.

## 2.1 Graph Theory

We take the term *network* to refer to the informal concept describing an object composed of elements and interactions or connections between these elements. For example, the Internet is a network composed of nodes (routers, hosts) and connections between these nodes (e.g. fiber cables). The natural means to model networks mathematically is provided by the notion of graphs.

A *graph* $G = (V, E)$ is an abstract object formed by a set $V$ of *vertices* (nodes) and a set $E$ of edges (links) that join (connect) pairs of vertices. The vertex set and edge set of a graph $G$ are denoted by $V(G)$ and $E(G)$, respectively. The cardinality of $V$ is usually denoted by $n$, the cardinality of $E$ by $m$. The two vertices joined by an edge are called its *endvertices*. If two vertices are joined by an edge, they are *adjacent* and we call them *neighbors*. Graphs can be *undirected* or *directed*. In undirected graphs, the order of the endvertices of an edge is immaterial. An undirected edge joining vertices $u, v \in V$ is denoted by $\{u, v\}$. In directed graphs, each directed edge (arc) has an *origin* (*tail*) and a *destination* (*head*). An edge with origin $u \in V$ and destination $v \in V$ is represented by an ordered pair $(u, v)$. As a shorthand notation, an edge $\{u, v\}$ or $(u, v)$ can also be

U. Brandes and T. Erlebach (Eds.): Network Analysis, LNCS 3418, pp. 7–15, 2005.