

COMPUTER ENGINEERING SERIES

Advanced Graph Theory and Combinatorics

Michel Rigo



iSTE

WILEY

Advanced Graph Theory and Combinatorics

Michel Rigo

ISTE

WILEY

First published 2016 in Great Britain and the United States by ISTE Ltd and John Wiley & Sons, Inc.

Apart from any fair dealing for the purposes of research or private study, or criticism or review, as permitted under the Copyright, Designs and Patents Act 1988, this publication may only be reproduced, stored or transmitted, in any form or by any means, with the prior permission in writing of the publishers, or in the case of reprographic reproduction in accordance with the terms and licenses issued by the CLA. Enquiries concerning reproduction outside these terms should be sent to the publishers at the undermentioned address:

ISTE Ltd
27-37 St George's Road
London SW19 4EU
UK

www.iste.co.uk

John Wiley & Sons, Inc.
111 River Street
Hoboken, NJ 07030
USA

www.wiley.com

© ISTE Ltd 2016

The rights of Michel Rigo to be identified as the author of this work have been asserted by him in accordance with the Copyright, Designs and Patents Act 1988.

Library of Congress Control Number: 2016953238

British Library Cataloguing-in-Publication Data

A CIP record for this book is available from the British Library

ISBN 978-1-84821-616-7

Foreword

The book content reflects the (good) taste of the author for solid mathematical concepts and results that have exciting practical applications. It is an excellent textbook that should appeal to students and instructors for its very clear presentation of both classical and more recent concepts in graph theory.

Vincent BLONDEL
Professor of Applied Mathematics, University of Louvain
September 2016

Introduction

This book is a result of lecture notes from a graph theory course taught at the University of Liège since 2005. Through the years, this course evolved and lectures were given at different levels ranging from second-year undergraduates in mathematics to students in computer science entering master's studies. It was therefore quite challenging to find a suitable title for this book.

Advanced or not so advanced material?

I hope that the reader will not feel cheated by the title (which is always tricky to choose). In some aspects, the material is rather elementary: we will start from scratch and present basic results on graphs such as connectedness or Eulerian graphs. In the second part of the book, we will analyze in great detail the strongly connected components of a digraph and make use of Perron–Frobenius theory and formal power series to estimate asymptotics on the number of walks of a given length between two vertices. Topics with an algebraic or a combinatorial flavor such as Ramsey numbers, introduction to Robertson–Seymour theorem or PageRank can be considered as more advanced.

In the history of mathematics, we often mention the *seven bridges of Königsberg problem* as the very first problem in graph theory. It was studied by the famous mathematician L. Euler in 1736. It took two centuries to develop and build a complete theory from a few scattered results. Probably the first book on graphs is *Theorie der endlichen und unendlichen Graphen* [KÖN 90] written by the Hungarian mathematician D. König in 1936, a student of J. Kürschák and H. Minkowski. In the middle of the 20th Century, graph theory matured into a well-defined branch of discrete mathematics and

combinatorics. We observe many mathematicians turning their attention to graph theory with books by C. Berge, N. Biggs, P. Erdős, C. Kuratowski, W. T. Tutte, K. Wagner, etc. We have seen an important growth during the past decades in combinatorics because of the particular interactions existing with optimization, randomized algorithms, dynamical programming, ergodic or number theory, theoretical computer science, computational geometry, molecular biology, etc. On MathSciNet, if you look for research papers with a Primary Mathematics Subject Classification equal to 05C (which stands for graph theory and is divided into 38 subfields ranging from planar graphs to connectivity, random walks or hypergraphs), then we find for the period 2011–2015 between 3, 300 and 3, 700 papers published every single year.

In less than a century, many scientists and entrepreneurs have seen the importance of graph theory in real-life applications. In a recent issue of *Wired* magazine (March 2014), we can read an article entitled *Is graph theory a key to understanding big data?* by R. Marsten. Let us quote his conclusion: “The data that we have today, and often the ways we look at data, are already steeped in the theory of graphs. In the future, our ability to understand data with graphs will take us beyond searching for an answer. Creating and analyzing graphs will bring us to answers automatically”. Later, in the same magazine (May 2014), E. Eifrem wrote: “We’re all well aware of how Facebook and Twitter have used the social graph to dominate their markets, and how Facebook and Google are now using their Graph Search and Knowledge Graph, respectively, to gear up for the next wave of hyper-accurate and hyper-personal recommendations, but graphs are becoming very widely deployed in a host of other industries”.

This book reflects the tastes of the author but also includes some important applications such as Google’s PageRank. It is only assumed that the reader has a working knowledge of linear algebra. Nevertheless, all the definitions and important results are given for the sake of completeness. The aim of the book is to *provide the reader with the necessary theoretical background to tackle new problems or to easily understand new concepts in graph theory*. Algorithms and complexity theory occupy a very small portion of the book (mostly in the first chapters).

This book, others and inspiration

Many other books on graphs do exist and the reader should not limit himself or herself to a single source. The Internet is also a formidable resource. Even if we have to be cautious when looking for information on the Web, Wikipedia contains a wealth of relevant information (but keep a critical

eye). The present book starts with some unavoidable general material, then moves on to some particular topics with a combinatorial flavor. Powers of the adjacency matrix and Perron theory have a predominant role. The reader could probably start with this book and then move to [BRU 11] as a good companion to get a deeper knowledge of the links between linear algebra and graphs. See also [BAP 14] or the classical [GOD 01] in algebraic graph theory. Similarly, a comprehensive presentation of PageRank techniques can be found in [LAN 06]. The authors of that book, A. Langville and C. D. Meyer, also specialized in ranking techniques (see [LAN 12]). Another general reference discussing partially the same topics as here – and I do hope, with the same philosophy – is by R. Diestel where much more material and a particular emphasis on infinite graphs may be found. The present book is smaller and is thus well suited for readers who do not want to spend too much time on a specialized topic.

Having given a graph theory course for more than 10 years, I'm probably unconsciously tempted to take ownership of the proofs that I found somewhere else. It is no easy task to cite and give credits to all the sources that inspired me in this process. Let me mention [BIG 93] for algebraic aspects and chromatic polynomial, [TUT 01] for its first chapters and [ORE 90] for planar graphs and his proof of the 5-color theorem. I should also mention [BOL 98] (with an impressive collection of exercises) and [BER 89]. Finally, I remember that projects available on the Web and run by D. Arnold (College of Redwoods) were inspiring.

Practical organization

For a one-semester course, here is the time I usually devote to selected topics with 15 lectures of 90 min. Moreover, sessions for exercises take the same amount of time. The book contains extra material and more than 115 exercises:

- digraphs, paths, connected components (sections 1.1 and 1.2);
- Eulerian graphs, distance and shortest path (sections 1.3 and 1.5);
- introduction to Hamiltonicity, applications (sections 1.4 and 1.6);
- trees (section 4.1);
- homomorphisms, group of automorphisms (sections 7.1 and 7.2);
- Hamiltonian graphs (sections 3.1–3.4);
- topological sort (Chapter 4);

- adjacency matrix, counting walks (sections 8.2 and 8.3);
- primitivity, Perron’s theorem and asymptotics (sections 9.1 and 9.4);
- irreducibility and asymptotics (sections 9.2 and 9.4);
- applications of Perron–Frobenius theory (section 9.3);
- Google’s PageRank (Chapter 10);
- planar graphs and Euler’s formula (sections 6.1–6.3);
- colorings, the five-color theorem (section 6.5);
- Ramsey numbers (section 7.4).

Definitions are emphasized and the most important ones are written in bold face, so that definitions of recurrent notions can be found more easily. Labels of bibliographic entries are based on the first three letters of the last name of the first author and then the year of publication. In the bibliography, entries are sorted in alphabetical order using these labels.

I address special thanks to Émilie Charlier, Aline Parreau and Manon Stipulanti for their great feedback leading to some major improvements in this book. Of course, Valérie Berthé always plays a special role. I am very pleased to blame her (indeed, this adventure produced some pressure from time to time) for what this project finally looks like. She is always supportive and enthusiastic. I also thanks several colleagues: Benoît Donnet, Eric Duchêne, Fabien Durand, Narad Rampersad, Eric Rowland and Élise Vandomme for the valuable time they spent reading drafts of parts of this book. I will not forget Laurent Waxweiler who gave and prepared the very first exercise classes for my course. I also thank my many students along the years. Their questions, queries and enthusiasm allowed me to improve, over the time, the overall presentation and sequencing of this book.

Michel RIGO
September, 2016

Contents

Foreword	ix
Introduction	xi
Chapter 1. A First Encounter with Graphs	1
1.1. A few definitions	1
1.1.1. Directed graphs	1
1.1.2. Unoriented graphs	9
1.2. Paths and connected components	14
1.2.1. Connected components	16
1.2.2. Stronger notions of connectivity	18
1.3. Eulerian graphs	23
1.4. Defining Hamiltonian graphs	25
1.5. Distance and shortest path	27
1.6. A few applications	30
1.7. Comments	35
1.8. Exercises	37
Chapter 2. A Glimpse at Complexity Theory	43
2.1. Some complexity classes	43
2.2. Polynomial reductions	46
2.3. More hard problems in graph theory	49
Chapter 3. Hamiltonian Graphs	53
3.1. A necessary condition	53
3.2. A theorem of Dirac	55

3.3. A theorem of Ore and the closure of a graph	56
3.4. Chvátal's condition on degrees	59
3.5. Partition of K_n into Hamiltonian circuits	62
3.6. De Bruijn graphs and magic tricks	65
3.7. Exercises	68
Chapter 4. Topological Sort and Graph Traversals	69
4.1. Trees	69
4.2. Acyclic graphs	79
4.3. Exercises	82
Chapter 5. Building New Graphs from Old Ones	85
5.1. Some natural transformations	85
5.2. Products	90
5.3. Quotients	92
5.4. Counting spanning trees	93
5.5. Unraveling	94
5.6. Exercises	96
Chapter 6. Planar Graphs	99
6.1. Formal definitions	99
6.2. Euler's formula	104
6.3. Steinitz' theorem	109
6.4. About the four-color theorem	113
6.5. The five-color theorem	115
6.6. From Kuratowski's theorem to minors	120
6.7. Exercises	123
Chapter 7. Colorings	127
7.1. Homomorphisms of graphs	127
7.2. A digression: isomorphisms and labeled vertices	131
7.3. Link with colorings	134
7.4. Chromatic number and chromatic polynomial	136
7.5. Ramsey numbers	140
7.6. Exercises	147
Chapter 8. Algebraic Graph Theory	151
8.1. Prerequisites	151
8.2. Adjacency matrix	154
8.3. Playing with linear recurrences	160

8.4. Interpretation of the coefficients	168
8.5. A theorem of Hoffman	169
8.6. Counting directed spanning trees	172
8.7. Comments	177
8.8. Exercises	178
Chapter 9. Perron–Frobenius Theory	183
9.1. Primitive graphs and Perron’s theorem	183
9.2. Irreducible graphs	188
9.3. Applications	190
9.4. Asymptotic properties	195
9.4.1. Canonical form	196
9.4.2. Graphs with primitive components	197
9.4.3. Structure of connected graphs	206
9.4.4. Period and the Perron–Frobenius theorem	214
9.4.5. Concluding examples	218
9.5. The case of polynomial growth	224
9.6. Exercises	231
Chapter 10. Google’s Page Rank	233
10.1. Defining the Google matrix	238
10.2. Harvesting the primitivity of the Google matrix	241
10.3. Computation	246
10.4. Probabilistic interpretation	246
10.5. Dependence on the parameter α	247
10.6. Comments	248
Bibliography	249
Index	263

A First Encounter with Graphs

1.1. A few definitions

There is not much fun in listing basic definitions about graphs (this is quite a bad introduction to start with!) but if we seek a rigorous presentation of results and proofs, then we cannot avoid giving accurate definitions of the objects that we will manipulate, but hopefully nice examples will also come quickly. In this book, we assume that the reader has a basic (or, at least a naive) knowledge of sets and operations on them.

As usual in mathematics, a *pair* (u, v) made up of two elements is implicitly assumed to be ordered: it has a first component u and a second component v . It has to be compared with a *set* with two elements u and v denoted by $\{u, v\}$. A set does not carry any ordering information about its elements. In particular, if $u \neq v$, then we can build two pairs but a single set: $(u, v) \neq (v, u)$ and $\{u, v\} = \{v, u\}$. If S is a finite set, we will write $\#S$ to denote the number of elements in S , i.e. the cardinality of S . We can also find the notation $|S|$ but we will use it to denote lengths of paths.

1.1.1. Directed graphs

DEFINITION 1.1.— *Let V be an arbitrary set. A directed graph, or **digraph**, is a pair $G = (V, E)$ where E is a subset of the Cartesian product $V \times V$, i.e. E is a set of pairs of elements in V . The elements of V are the vertices of G – some authors also use the term *nodes* – and the elements of E are the edges,*

also called oriented edges or arcs¹, of G . An edge of the form (v, v) is a **loop** on v . Another way to express that E is a subset of $V \times V$ is to say that E is a binary relation over V . If either (u, v) or (v, u) belongs to E , the vertices u and v are adjacent. If neither (u, v) nor (v, u) belong to E , then u and v are independent. Given a digraph G , the set of vertices (respectively of edges) of G is denoted by $V(G)$ (respectively $E(G)$).

The vast majority of the graphs that we will encounter are *finite* meaning that the set V of vertices is finite, and thus E contains at most $(\#V)^2$ edges.

REMARK 1.2.— *It is common to speak of the order of G for $\#(V(G))$ and the size of G for $\#(E(G))$.*

There are a few examples of infinite graphs in this book; see examples 1.47 and 4.11 (in formal language theory) and section 7.2 about colorings. Infinite graphs usually require more sophisticated arguments such as the axiom of choice. Implementation of infinite graphs in a computer could be tricky or impossible. From a practical point of view, particular instances of infinite graphs with a countable number of vertices and edges can be implemented. Think about a periodic graph that permits one to store only a finite amount of information to be repeated or a relation among vertices that can be computed and implemented as a function (see exercise 6 and example 1.5).

A digraph $G = (V, E)$ is said to be **simple** if E is a subset of $(V \times V) \setminus \{(v, v) \mid v \in V\}$. In that case, the relation E is *irreflexive*. Otherwise stated, loops are not allowed.

The elements belonging to a set are pairwise distinct: there is no repeated element. What we need to define a directed multigraph, i.e. a digraph where multiple edges between two vertices are allowed, is to permit repetitions of an element belonging to a set. In set theory, we can introduce the notion of a *multiset*. First, we restrict ourselves to multisets with finite integer multiplicities. A **multiset** M is a pair (S, m) where S is a set, in the “classical” sense, and $m : S \rightarrow \mathbb{N}_{\geq 1}$ is a *multiplicity function* that specifies the number $m(s)$ of occurrences of $s \in S$ in the multiset. As an example, the multiset denoted by $\{u, u, v, v, v, w\}$ is built from $S = \{u, v, w\}$ and

¹ If we really have to distinguish the directed graphs from the undirected graphs that we will soon introduce, then we could restrict the use of the word “edge” to the undirected case and use the word “arc” solely in the directed case. But usually the context permits one to avoid any misunderstanding.

$m(u) = 2$, $m(v) = 3$, $m(w) = 1$. If the occurrences of an element have to be distinguished², we can index elements $s \in S$ by $s_1, \dots, s_{m(s)}$. To continue the example, $\{u_1, u_2, v_1, v_2, v_3, w_1\}$ denotes the same multiset as above. If S is a finite set, then the cardinality of the multiset $M = (S, m)$ with finite multiplicities is

$$\#M := \sum_{s \in S} m(s).$$

Observe that a multiset (S, m) where $m(s) = 1$, for all $s \in S$, is simply a set. Equivalently, we could have defined the multiplicity function to map every element s of S to a finite subset of \mathbb{N} : the set of indices used for s .

Second, we could consider countable multiplicities³. In that case, an element of a multiset can be repeated infinitely many times and the multiplicity function maps every element to a subset of \mathbb{N} (which is the set of indices used for that element). As an example, a vertex u could be repeated infinitely many times with prime indices: $\{u_2, u_3, u_5, u_7, u_{11}, \dots\}$. Thus, the multiplicity function maps u to the set of prime numbers.

We now introduce a **directed multigraph** as a pair $G = (V, E)$ where V is a set of vertices and E is a multiset of edges built from a subset of $V \times V$. For a directed multigraph $G = (V, E)$, the fact that V is finite does not imply that E is also finite. Indeed, we could have infinitely many edges between two vertices. Thus, a directed multigraph is *finite* if both the set V and the multiset E are finite.

REMARK 1.3.— *It is common (and quite visual) to represent the vertices of a digraph by points in the plane (but we can also draw graphs on other surfaces like on a torus). Edges of the form (u, v) are represented by arrows going from u to v . We say that u (resp. v) is the **origin** (respectively, **destination**) of the edge. Actually any oriented arc of a curve can be used to join two vertices, not only straight vectors. Since positions of the vertices and arcs of curves joining the vertices can be freely chosen, there are usually infinitely many ways to represent a given graph. There is no reason that two edges that are intersecting in one representation are also intersecting in another representation of the same graph. We will rediscuss these notions with great care in section 6.1.*

² For instance, to define a walk using different edges between two vertices.

³ Recall that a set is *countable* if it is in one-to-one correspondence with a subset of \mathbb{N} . Of course, from a mathematical point of view, further generalizations of multiplicity function and multisets can be considered; see section 1.7 for comments and pointers.

In Figure 1.1, we have depicted representations of a simple digraph, digraph and directed multigraph.

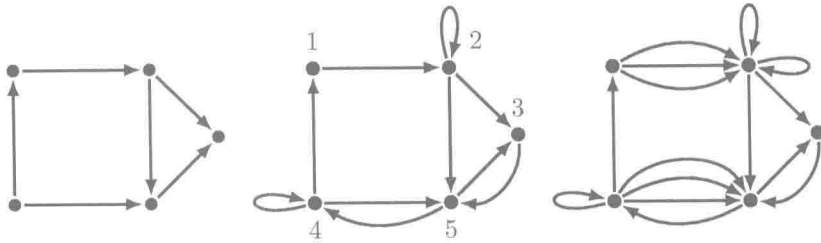


Figure 1.1. From left to right: a simple digraph, a digraph and a directed multigraph

A digraph G can be stored as an *adjacency list*: with each vertex u is associated the list of vertices v such that $(u, v) \in E(G)$. For the central digraph in Figure 1.1, the corresponding adjacency list is given in Table 1.1. A similar data structure can be used for directed multigraphs.

1	2		
2	2	3	5
3	5		
4	1	4	5
5	3	4	

Table 1.1. An adjacency list

EXAMPLE 1.4 (Tournament).— A simple digraph $G = (V, E)$ where, for all pairs of distinct vertices u and v , either (u, v) or (v, u) belongs to E (but exactly one of these two edges belongs to E) is said to be a tournament. Indeed, it corresponds to an all-play-all tournament: each player plays against every other player and there are no ties. If u wins the confrontation against v , then we take the edge (v, u) . See Figure 1.2.

EXAMPLE 1.5.— For an example of infinite simple digraph, take $\mathbb{N}_{>1}$ as set of vertices and a pair (m, n) of integers greater than 1 is an edge if and only if m divides n . The first few vertices and some edges of this digraph are depicted in Figure 1.3. Note that the relation E is transitive. If (m, n) and (n, p) belong to E , then (m, p) belongs to E .

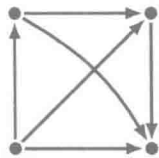


Figure 1.2. A tournament among four players or teams

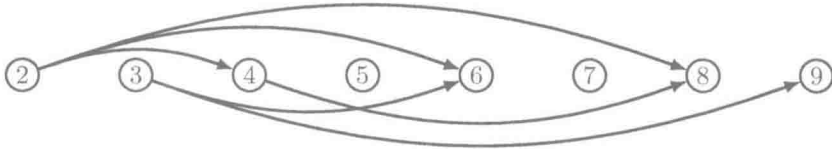


Figure 1.3. A divisibility relation (first few vertices only)

EXAMPLE 1.6.— We consider the digraph made up of Webpages and there is an edge from a page p to a page q if there is a link on p referencing q . This digraph is finite but contains several billions of vertices. Independently of the content of the pages, here we are interested in the links that one user can follow by browsing through pages. Based on Perron's theorem (theorem 9.2), we will discuss the basis of the Google's PageRank algorithm in Chapter 10.

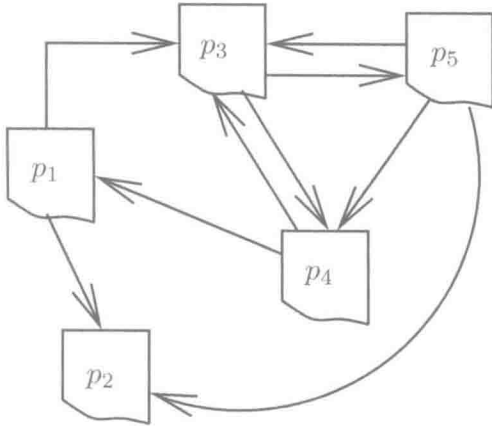


Figure 1.4. Some links and Webpages

Similarly to pages referencing other pages, we can also think about scientific papers that are citing other papers. In that case, we get a digraph where it is meaningful to try to identify relevant or influential papers. Which are the papers that are cited by many other papers, which are the “best” journals? The website <http://www.eigenfactor.org/> uses a similar strategy to rank journals instead of Webpages [BER 07, WES 10].

EXAMPLE 1.7.— The digraph that we may associate with Twitter is another example about social networks. There is an edge from a user account u to a user account v , if u is following the tweets of v . Therefore, all the tweets posted by v are displayed in the follower’s timeline. Such a digraph captures who is following who. For instance, see [YAM 10] for an example of a User–Tweet digraph.

REMARK 1.8.— The reader may wonder about this triple definition of digraphs: simple digraphs, digraphs and directed multigraphs. Why should we take into account the case of simple or multiple digraphs? The answer is a pragmatic one. We choose the model that best fits the situation that we are considering. If we are interested in finding a shortest path between two vertices, it is meaningless to consider loops or multiple edges; going through a loop just makes the path longer. We just want to know if the two vertices are connected or not. In such a case, we will deal with simple digraphs. On the other hand, assume that we have to model the fact that between two cities, there is a road, a river but also a railway. Here multigraphs are useful to take this fact into account. Note that simple digraphs are special cases of digraphs that are themselves special cases of directed multigraphs. We reach same conclusions when we have to choose between digraphs and the unoriented graphs that we will soon introduce to model a particular situation.

DEFINITION 1.9.— In a directed graph $G = (V, E)$, we may associate two sets with every vertex v , respectively, the sets of outgoing edges and incoming edges:

$$\omega^+(v) := \{(v, w) \in E \mid w \in V\}, \quad \omega^-(v) := \{(u, v) \in E \mid u \in V\}.$$

These definitions are extended to directed multigraphs and in that case, the corresponding sets are multisets. If, for all vertices v , the multisets $\omega^+(v)$ and $\omega^-(v)$ are finite, then we say that G is of *finite degree*. The **successors** (respectively, **predecessors**) of a vertex v are the vertices w (respectively, u) such that (v, w) (respectively, (u, v)) belongs to $\omega^+(v)$ (respectively, $\omega^-(v)$). The set of successors (respectively, predecessors) of v is denoted by $\text{succ}(v)$ (respectively, $\text{pred}(v)$). Note that there is a loop on v if and only if