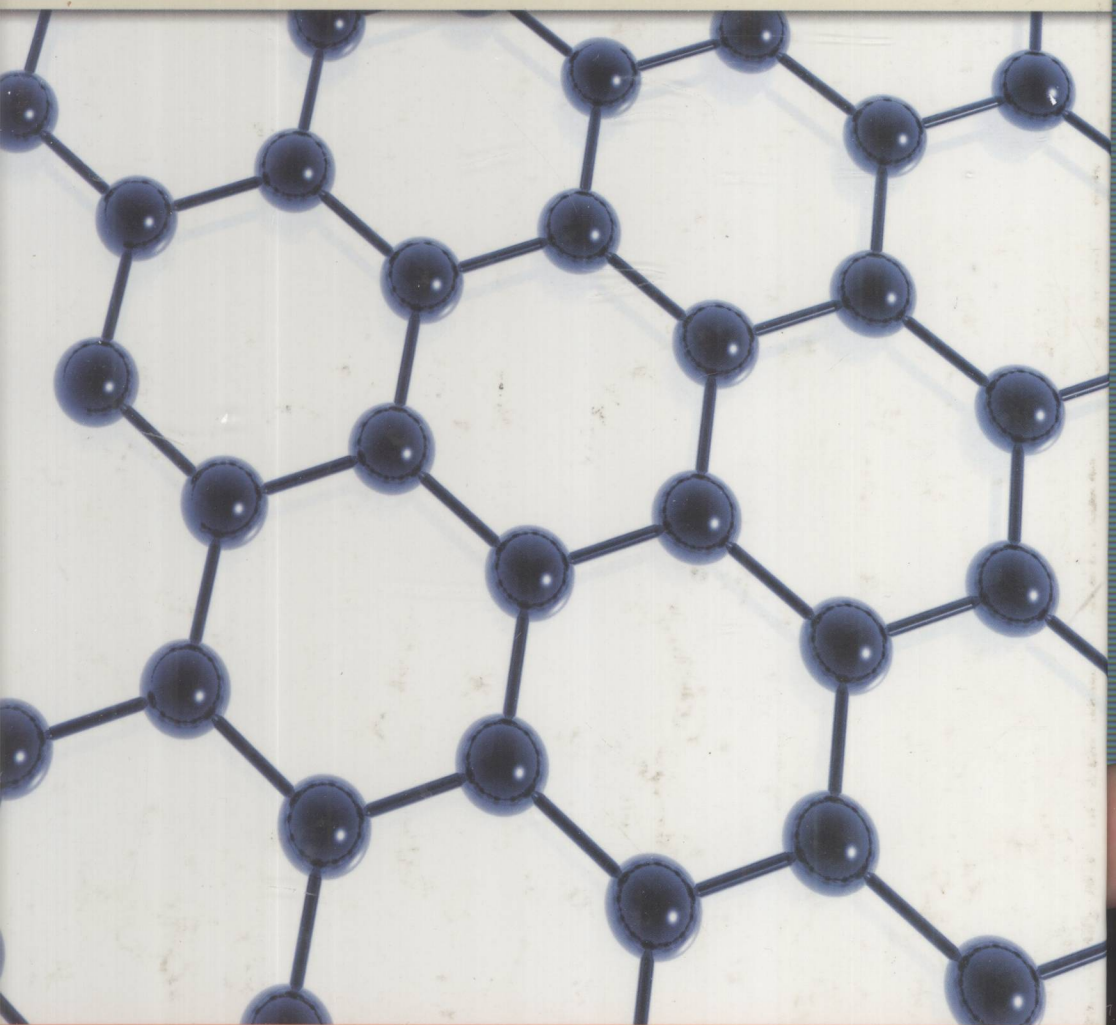


Graph Theory and Interconnection Networks

Lih-Hsing Hsu and Cheng-Kuan Lin



CRC Press
Taylor & Francis Group

0157.5
H873

Graph Theory and Interconnection Networks



Lih-Hsing Hsu and Cheng-Kuan Lin



E2008002042



CRC Press

Taylor & Francis Group

Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

© 2009 by Taylor & Francis Group, LLC
CRC Press is an imprint of Taylor & Francis Group, an Informa business

No claim to original U.S. Government works
Printed in the United States of America on acid-free paper
10 9 8 7 6 5 4 3 2 1

International Standard Book Number-13: 978-1-4200-4481-2 (Hardcover)

This book contains information obtained from authentic and highly regarded sources. Reasonable efforts have been made to publish reliable data and information, but the author and publisher cannot assume responsibility for the validity of all materials or the consequences of their use. The authors and publishers have attempted to trace the copyright holders of all material reproduced in this publication and apologize to copyright holders if permission to publish in this form has not been obtained. If any copyright material has not been acknowledged please write and let us know so we may rectify in any future reprint.

Except as permitted under U.S. Copyright Law, no part of this book may be reprinted, reproduced, transmitted, or utilized in any form by any electronic, mechanical, or other means, now known or hereafter invented, including photocopying, microfilming, and recording, or in any information storage or retrieval system, without written permission from the publishers.

For permission to photocopy or use material electronically from this work, please access www.copyright.com (<http://www.copyright.com/>) or contact the Copyright Clearance Center, Inc. (CCC), 222 Rosewood Drive, Danvers, MA 01923, 978-750-8400. CCC is a not-for-profit organization that provides licenses and registration for a variety of users. For organizations that have been granted a photocopy license by the CCC, a separate system of payment has been arranged.

Trademark Notice: Product or corporate names may be trademarks or registered trademarks, and are used only for identification and explanation without intent to infringe.

Library of Congress Cataloging-in-Publication Data

Hsu, Lih-Hsing.
Graph theory and interconnection networks / Lih-Hsing Hsu, Cheng-Kuan Lin.
p. cm.
Includes bibliographical references and index.
ISBN 978-1-4200-4481-2 (hardback : acid-free paper)
1. Graph theory. I. Lin, Cheng-Kuan. II. Title.

QA166.H78 2008
511'.5--dc22

2008019917

Visit the Taylor & Francis Web site at
<http://www.taylorandfrancis.com>

and the CRC Press Web site at
<http://www.crcpress.com>

Graph Theory and Interconnection Networks

OTHER AUERBACH PUBLICATIONS

Chaos Applications in Telecommunications

Peter Stavroulakis

ISBN: 0-8493-3832-8

Contemporary Coding Techniques and Applications for Mobile Communications

Onur Osman and Osman Nuri Ucan

ISBN: 1-4200-5461-9

Design Science Research Methods and Patterns: Innovating Information and Communication Technology

Vijay K. Vaishnavi and William Kuechler Jr.

ISBN: 1-4200-5932-7

Embedded Linux System Design and Development

P. Raghavan, Amol Lad and Sriram Neelakandan

ISBN: 0-8493-4058-6

Enhancing Computer Security with Smart Technology

V. Rao Vemuri

ISBN: 0-8493-3045-9

Graph Theory and Interconnection Networks

Lih-Hsing Hsu; Cheng-Kuan Lin

ISBN: 1-4200-4481-8

Modeling Software with Finite State Machines: A Practical Approach

Ferdinand Wagner, Ruedi Schmuki, Thomas Wagner and Peter Wolstenholme

ISBN: 0-8493-8086-3

MEMS and Nanotechnology-Based Sensors and Devices for Communications, Medical and Aerospace Applications

A. R. Jha

ISBN: 0-8493-8069-3

Neural Networks for Applied Sciences and Engineering: From Fundamentals to Complex Pattern Recognition

Sandhya Samarasinghe

ISBN: 0-8493-3375-X

Patterns for Performance and Operability: Building and Testing Enterprise Software

Chris Ford, Ido Gileadi, Sanjiv Purba and Mike Moerman

ISBN: 1-4200-5334-5

Software Engineering Foundations: A Software Science Perspective

Yingxu Wang

ISBN: 0-8493-1931-5

Systemic Yoyos: Some Impacts of the Second Dimension

Yi Lin

ISBN: 1-4200-8820-3

AUERBACH PUBLICATIONS

www.auerbach-publications.com

To Order Call: 1-800-272-7737 • Fax: 1-800-374-3401

E-mail: orders@crcpress.com

Preface

Since the origin of graph theory with Euler, people have liked graph theory, because it is a delightful playground for the exploration of proof techniques without much previous background needed. With the application of graph theory to many areas of computing, social, and natural sciences, the importance of graph theory has been recognized. For this reason, many good graph theoretic results have been developed in recent years. It is almost impossible to write a book that covers all these good graph theories.

The architecture of an interconnection network is always represented by a graph, where vertices represent processors and edges represent links between processors. It is almost impossible to design a network that is optimum from all aspects. One has to design a suitable network depending on its properties and requirements. Thus, many graphs are proposed as possible interconnection network topologies. Thus, these graphs can be referred to as good graphs. For this reason, the theory of interconnection networks is referred to as *good-graph theory*.

Thus, we need some basic background in graph theory to study interconnection networks. The first 10 chapters cover those materials presented in most graph theory texts and add some concepts of interconnection networks. Later, we discuss some interesting properties of interconnection networks. Sometimes, these properties are too good to be true—so the beginner finds them hard to believe. Such properties can be observed from interconnection networks, diameter, connectivity, Hamiltonian properties, and diagnosis properties. We can also study these properties in general graphs.

Ends of proofs are marked with the symbol \square . This symbol can be found directly following a formal assertion. It means that the proof should be clear after what has been said. There are also some theorems that are stated, without proof, via background information. In this case, such theorems are stated without proofs and the symbol \square .

The main purpose of this book is to share our research experience. Our experience tells us that much background is not needed for doing research. Looking for a proper theme is a difficult problem for research. We are just lucky in finding problems. For this reason, we put some of our results at the end of each chapter but we do not put any exercises in this book. We observed that all the material, contents, and exercises in every book are exercises of previous books. For this reason, we hope that the readers can find their own exercises.

It is a great pleasure to acknowledge the work of Professor Jimmy J.M. Tan. We have worked together for a long time. Yet, at least one-third of this book is his contribution. We also thank all the members of the Computer Theory Laboratory of National Chiao-Tung University: T.Y. Sung, T. Liang, H.M. Huang, Chiuyuan Chen, Y.C. Chuang, S. S. Kao, Y.C. Chang, T.Y. Ho, R.S. Chou, C.N. Hung, J.J. Wang, S.Y. Wang, C.Y. Lin, C.H. Tsai, J.Y. Hwang, C.P. Chang, J.N. Wang, J.J. Sheu, C.H. Chang, T.K. Li, W.T. Huang, Y.C. Chen,

H.C. Hsu, K.Y. Liang, Y.L. Hsieh, M.C. Yang, P.L. Lai, L.C. Chiang, Y.H. Teng, C.M. Sun, and K.M. Hsu.

We also thank CRC Press for offering us the opportunity of publishing this book. Lih-Hsing Hsu also thanks his advisor, Professor Joel Spencer, for entering the area of graph theory. We also thank Professor Frank Hsu for strongly recommending to us the area of interconnection networks. Finally, we thank all of our teachers; without their encouragement, this book could never have been written.

Authors

Lih-Hsing Hsu received his BS in mathematics from Chung Yuan Christian University, Taiwan, Republic of China, in 1975, and his PhD in mathematics from the State University of New York at Stony Brook in 1981. From 1981 to 1985, he was an associate professor at the Department of Applied Mathematics at National Chiao Tung University in Taiwan. From 1985 to 2006, he was a professor at National Chiao Tung University. From 1985 to 1988, he was the chairman of the Department of Applied Mathematics at National Chiao Tung University. After 1988, he joined the Department of Computer and Information Science of National Chiao Tung University. In 2004, he retired from National Chiao Tung University, holding a title as an honorary scholar of that university. He is currently the chairman of the Department of Computer Science and Information Engineering, Providence University, Taichung, Taiwan. His research interests include interconnection networks, algorithms, graph theory, and VLSI layout.

Cheng-Kuan Lin received his BS in applied mathematics from Chinese Culture University, Taiwan, Republic of China, in 2000, and his MS in mathematics from the National Central University in Taiwan in 2002. His research interests include interconnection networks, algorithms, and graph theory.

Contents

Preface	xi
Authors	xiii
 Chapter 1: Fundamental Concepts	 1
1.1 Graphs and Simple Graphs	1
1.2 Matrices and Isomorphisms	5
1.3 Paths and Cycles	9
1.4 Vertex Degrees	12
1.5 Graph Operations	14
1.6 Some Basic Techniques	16
1.7 Degree Sequences	18
 Chapter 2: Applications on Graph Isomorphisms	 21
2.1 Generalized Honeycomb Tori	21
2.2 Isomorphism between Cyclic-Cubes and Wrapped Butterfly Networks	24
2.3 1-Edge Fault-Tolerant Design for Meshes	26
2.4 Faithful 1-Edge Fault-Tolerant Graphs	31
2.5 k -Edge Fault-Tolerant Designs for Hypercubes	40
 Chapter 3: Distance and Diameter	 43
3.1 Introduction	43
3.2 Diameter for Some Interconnection Networks	43
3.3 Shuffle-Cubes	47
3.3.1 $\text{Route1}(u, v)$	49
3.4 Moore Bound	50
3.5 Star Graphs and Pancake Graphs	52
3.6 Edge Congestion and Bisection Width	55
3.7 Transmitting Problem	57
 Chapter 4: Trees	 61
4.1 Basic Properties	61
4.2 Breadth-First Search and Depth-First Search	64
4.3 Rooted Trees	65
4.4 Counting Trees	68
4.5 Counting Binary Trees	72
4.6 Number of Spanning Trees Contains a Certain Edge	73
4.7 Embedding Problem	76

Chapter 5: Eulerian Graphs and Digraphs	79
5.1 Eulerian Graphs	79
5.2 Eulerian Digraphs	82
5.3 Applications	85
5.3.1 Chinese Postman Problem	85
5.3.2 Street-Sweeping Problem	86
5.3.3 Drum Design Problem	86
5.3.4 Functional Cell Layout Problem	87
Chapter 6: Matchings and Factors	93
6.1 Matchings	93
6.2 Bipartite Matching	95
6.3 Edge Cover	98
6.4 Perfect Matching	100
6.5 Factors	103
Chapter 7: Connectivity	105
7.1 Cut and Connectivity	105
7.2 2-Connected Graphs	109
7.3 Menger Theorem	111
7.4 An Application—Making a Road System One-Way	115
7.5 Connectivity of Some Interconnection Networks	117
7.6 Wide Diameters and Fault Diameters	118
7.7 Superconnectivity and Super-Edge-Connectivity	120
Chapter 8: Graph Coloring	125
8.1 Vertex Colorings and Bounds	125
8.2 Properties of k -Critical Graphs	126
8.3 Bound for Chromatic Numbers	128
8.4 Girth and Chromatic Number	130
8.5 Hajós' Conjecture	131
8.6 Enumerative Aspects	133
8.7 Homomorphism Functions	136
8.8 An Application—Testing on Printed Circuit Boards	137
8.9 Edge-Colorings	138
Chapter 9: Hamiltonian Cycles	141
9.1 Hamiltonian Graphs	141
9.2 Necessary Conditions	141
9.3 Sufficient Conditions	143
9.4 Hamiltonian-Connected	147
9.5 Mutually Independent Hamiltonian Paths	152
9.6 Diameter for Generalized Shuffle-Cubes	155
9.7 Cycles in Directed Graphs	158

Chapter 10: Planar Graphs	161
10.1 Planar Embeddings	161
10.2 Euler's Formula	165
10.3 Characterization of Planar Graphs	166
10.4 Coloring of Planar Graphs	167
Chapter 11: Optimal k-Fault-Tolerant Hamiltonian Graphs	171
11.1 Introduction	171
11.2 Node Expansion	173
11.3 Other Construction Methods	180
11.4 Fault-Tolerant Hamiltonicity and Fault-Tolerant Hamiltonian Connectivity of the Folded Petersen Cube Networks	185
11.5 Fault-Tolerant Hamiltonicity and Fault-Tolerant Hamiltonian Connectivity of the Pancake Graphs	189
11.6 Fault-Tolerant Hamiltonicity and Fault-Tolerant Hamiltonian Connectivity of Augmented Cubes	195
11.7 Fault-Tolerant Hamiltonicity and Fault-Tolerant Hamiltonian Connectivity of the WK-Recursive Networks	206
11.8 Fault-Tolerant Hamiltonicity of the Fully Connected Cubic Networks	221
Chapter 12: Optimal 1-Fault-Tolerant Hamiltonian Graphs	227
12.1 Introduction	227
12.2 3-Join	228
12.3 Cycle Extension	229
12.4 Cells for Optimal 1-Hamiltonian Regular Graphs	241
12.5 Generalized Petersen Graphs	249
12.6 Honeycomb Rectangular Disks	259
12.7 Properties with Respect to the 3-Join	262
12.8 Examples of Various Cubic Planar Hamiltonian Graphs	267
12.8.1 $A \cap B \cap C$	268
12.8.2 $\bar{A} \cap B \cap \bar{C}$	268
12.8.3 $\bar{A} \cap B \cap C$	268
12.8.4 $A \cap B \cap \bar{C}$	269
12.8.5 $A \cap \bar{B} \cap C$	271
12.8.6 $A \cap \bar{B} \cap \bar{C}$	272
12.8.7 $\bar{A} \cap \bar{B} \cap C$	273
12.8.8 $\bar{A} \cap \bar{B} \cap \bar{C}$	274
12.9 Hamiltonian Properties of Double Loop Networks	275
Chapter 13: Optimal k-Fault-Tolerant Hamiltonian-Laceable Graphs	285
13.1 Introduction	285
13.2 Super Fault-Tolerant Hamiltonian Laceability of Hypercubes	287
13.3 Super Fault-Tolerant Hamiltonian Laceability of Star Graphs	289

13.4	Construction Schemes	294
13.5	Cubic Hamiltonian-Laceable Graphs	301
13.6	1_p -Fault-Tolerant Hamiltonian Graphs	303
13.6.1	Spider-Web Networks	303
13.6.2	Brother Trees	312
13.7	Hamiltonian Laceability of Faulty Hypercubes	318
13.8	Conditional Fault Hamiltonicity and Conditional Fault Hamiltonian Laceability of the Star Graphs	327
Chapter 14: Spanning Connectivity		339
14.1	Introduction	339
14.2	Spanning Connectivity of General Graphs	339
14.3	Spanning Connectivity and Spanning Laceability of the Hypercube-Like Networks	347
14.4	Spanning Connectivity of Crossed Cubes	361
14.5	Spanning Connectivity and Spanning Laceability of the Enhanced Hypercube Networks	374
14.6	Spanning Connectivity of the Pancake Graphs	391
14.7	Spanning Laceability of the Star Graphs	404
14.8	Spanning Fan-Connectivity and Spanning Pipe-Connectivity of Graphs	410
Chapter 15: Cubic 3^*-Connected Graphs and Cubic 3^*-Laceable Graphs		417
15.1	Properties of Cubic 3^* -Connected Graphs	417
15.2	Examples of Cubic Super 3^* -Connected Graphs	419
15.3	Counterexamples of 3^* -Connected Graphs	426
15.4	Properties of Cubic 3^* -Laceable Graphs	430
15.5	Examples of Cubic Hyper 3^* -Laceable Graphs	431
15.6	Counterexamples of 3^* -Laceable Graphs	447
Chapter 16: Spanning Diameter		449
16.1	Introduction	449
16.2	Spanning Diameter for the Star Graphs	449
16.3	Spanning Diameter of Hypercubes	465
16.4	Spanning Diameter for Some (n, k) -Star Graphs	474
Chapter 17: Pancylic and Panconnected Property		479
17.1	Introduction	479
17.2	Bipanconnected and Bipancylic Properties of Hypercubes	480
17.3	Edge Fault-Tolerant Bipancylic Properties of Hypercubes	485
17.4	Panconnected and Pancylic Properties of Augmented Cubes	489
17.5	Comparison between Panconnected and Panpositionable Hamiltonian ...	500
17.6	Bipanpositionable Bipancylic Property of Hypercube	504

- Chapter 18: Mutually Independent Hamiltonian Cycles** 509
 - 18.1 Introduction 509
 - 18.2 Mutually Independent Hamiltonian Cycles on Some Graphs 510
 - 18.3 Mutually Independent Hamiltonian Cycles of Hypercubes 512
 - 18.4 Mutually Independent Hamiltonian Cycles of Pancake Graphs 518
 - 18.5 Mutually Independent Hamiltonian Cycles of Star Graphs 524
 - 18.6 Fault-Free Mutually Independent Hamiltonian Cycles
in a Faulty Hypercube 526
 - 18.7 Fault-Free Mutually Independent Hamiltonian Cycles
in Faulty Star Graphs 530
 - 18.8 Orthogonality for Sets of Mutually Independent
Hamiltonian Cycles 543

- Chapter 19: Mutually Independent Hamiltonian Paths** 545
 - 19.1 Introduction 545
 - 19.2 Mutually Independent Hamiltonian Laceability for Hypercubes 545
 - 19.3 Mutually Independent Hamiltonian Laceability for Star Graphs 550
 - 19.4 Mutually Independent Hamiltonian Connectivity for
(n, k)-Star Graphs 555
 - 19.5 Cubic 2-Independent Hamiltonian-Connected Graphs 575
 - 19.5.1 Examples of Cubic 2-Independent Hamiltonian-Connected
Graphs That Are Super 3*-Connected 576
 - 19.5.2 Examples of Super 3*-Connected Graphs That Are Not
Cubic 2-Independent Hamiltonian-Connected 579
 - 19.5.3 Example of a Cubic 2-Independent Hamiltonian-
Connected Graph That Is Not Super 3*-Connected 580
 - 19.5.4 Example of a Cubic 1-Fault-Tolerant Hamiltonian Graph That
Is Hamiltonian-Connected but Not Cubic 2-Independent
Hamiltonian-Connected or Super 3*-Connected 581

- Chapter 20: Topological Properties of Butterfly Graphs** 585
 - 20.1 Introduction 585
 - 20.2 Cycle Embedding in Faulty Butterfly Graphs 590
 - 20.3 Spanning Connectivity for Butterfly Graphs 607
 - 20.4 Mutually Independent Hamiltonicity for Butterfly Graphs 616

- Chapter 21: Diagnosis of Multiprocessor Systems** 625
 - 21.1 Introduction 625
 - 21.2 Diagnosis Models 625
 - 21.2.1 PMC Model 626
 - 21.2.2 Comparison Model 628
 - 21.3 Diagnosability of the Matching Composition Networks 631
 - 21.3.1 Diagnosability of the Matching Composition Networks
under the PMC Model 632

21.3.2	Diagnosability of the Matching Composition Networks under the Comparison Model	632
21.4	Diagnosability of Cartesian Product Networks	637
21.4.1	Diagnosability of Cartesian Product Networks under the PMC Model	638
21.4.2	Diagnosability of Cartesian Product Networks under the Comparison Model	639
21.4.3	Diagnosability of t -Connected Networks	639
21.4.4	Diagnosability of Homogeneous Product Networks	641
21.4.5	Diagnosability of Heterogeneous Product Networks	644
21.5	Strongly t -Diagnosable Systems	645
21.5.1	Strongly t -Diagnosable Systems in the Matching Composition Networks	649
21.6	Conditional Diagnosability	652
21.6.1	Conditional Diagnosability of Q_n under the PMC Model	653
21.7	Conditional Diagnosability of Q_n under the Comparison Model	658
21.7.1	Conditional Diagnosability of Cayley Graphs Generated by Transposition Trees under the Comparison Diagnosis Model	666
21.7.2	Cayley Graphs Generated by Transposition Trees	666
21.7.3	Conditional Diagnosability	668
21.8	Local Diagnosability	670
21.8.1	Strongly Local-Diagnosable Property	675
21.8.2	Conditional Fault Local Diagnosability	679
	References	687
	Index	703

1 Fundamental Concepts

1.1 GRAPHS AND SIMPLE GRAPHS

A graph G with n vertices and m edges consists of the *vertex set* $V(G) = \{v_1, v_2, \dots, v_n\}$ and *edge set* $E(G) = \{e_1, e_2, \dots, e_m\}$, where each edge consists of two (possibly equal) vertices called, *endpoints*. An element in $V(G)$ is called a *vertex* of G . An element in $E(G)$ is called an *edge* of G . Because graph theory has a variety of applications, we may also use a *node* for a vertex and a *link* for an edge to fit the common terminology in that area. We use the *unordered pair* (u, v) for an edge $e = \{u, v\}$. If $(u, v) \in E(G)$, then u and v are *adjacent*. We write $u \leftrightarrow v$ to mean u is *adjacent to* v . A *loop* is an edge whose endpoints are equal. *Parallel edges* or *multiple edges* are edges that have the same pair of endpoints. A *simple graph* is a graph having no loops or multiple edges. For a graph $G = (V, E)$, the *underlying simple graph* UG is the simple graph with vertex V and $(x, y) \in E(UG)$ if and only if $x \neq y$ and $(x, y) \in E$. A graph is *finite* if its vertex set and edge set are finite. We adopt the convention that every graph mentioned is *finite*.

We illustrate a graph on paper by assigning a point to each vertex and drawing a curve for each edge between the points representing its endpoints, sometimes omitting the name of the vertices or edges. In Figure 1.1a, e is a loop and f and g are parallel edges. Figures 1.1b and 1.1c illustrate two ways of drawing one simple graph.

The *order* of a graph G , written $n(G)$, is the number of vertices in G . An n -*vertex graph* is a graph of order n . The *size* of a graph G , written $e(G)$, is the number of edges in G , even though we also use e by itself to denote an edge. The *degree* of a vertex v in a graph, written $\deg_G(v)$, or $\deg(v)$, is the number of nonloop edges containing v plus twice the number of loops containing v . The *maximum degree* of G , denoted by $\Delta(G)$, is $\max\{\deg(v) \mid v \in V(G)\}$ and the *minimum degree* of G , denoted by $\delta(G)$, is $\min\{\deg(v) \mid v \in V(G)\}$. A graph G is *regular* if $\Delta(G) = \delta(G)$, and k -*regular* if $\Delta(G) = \delta(G) = k$. A vertex of degree k is k -*valent*. The *neighborhood* of v , written $N_G(v)$ or $N(v)$, is $\{x \in V(G) \mid x \leftrightarrow v\}$; x is a *neighbor* of v if $x \in N(v)$. An *isolated vertex* has a degree of 0.

A *directed graph* or *digraph* G consists of a *vertex set* $V(G)$ and an *edge set* $E(G)$, where each edge is an ordered pair of vertices. We use the *ordered pair* (u, v) for the edge uv , where u is the *tail* and v is the *head*. (Note that the notation (u, v) is used both as an unordered pair in an unordered graph and as an ordered pair in a directed graph. However, it is easy to distinguish each from the other from the context.) Sometimes, we may use the term *arc* for an edge of a digraph. We write $u \rightarrow v$ when $(u, v) \in E(G)$, meaning *there is an edge from* u *to* v . A *simple digraph* is a digraph in which each ordered pair of vertices occur at most once as an edge. For a digraph $G = (V, E)$, the

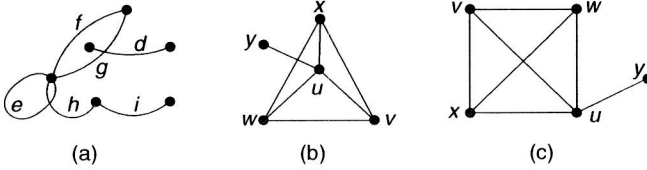


FIGURE 1.1 Examples of graphs.

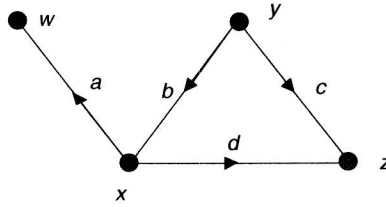


FIGURE 1.2 A digraph.

underlying graph UG is the simple graph with the vertex set V and $(x, y) \in E(UG)$ if and only if $x \neq y$ and either $(x, y) \in E(G)$ or $(y, x) \in E(G)$.

The choice of head and tail assigns a *direction* to an edge, which we illustrate by assigning edges as arrows. See Figure 1.2 for an example of a digraph. Sometimes, weights are assigned to the edges of graphs or digraphs. Thus, we have *weighted graphs*, *weighted digraphs*, (*unweighted*) *graphs*, and (*unweighted*) *digraphs*.

Graphs arise in many settings, which suggests useful concepts and terminologies about the structure of graphs.

EXAMPLE 1.1

A famous brain-teaser asks: Does every set of six people have three mutual acquaintances or three mutual strangers?

Because an *acquaintance* is symmetric, we can model it by a simple graph having a vertex for each person and an edge for each acquainted pair. The *nonacquaintance* relation on the same set yields another graph. The *complement* \bar{G} of a simple graph G with the same vertex set $V(G)$ is defined by $(u, v) \in E(\bar{G})$ if and only if $(u, v) \notin E(G)$. In Figure 1.3, we draw a graph and its complement.

Let two graphs equal $G = (V(G), E(G))$ and $H = (V(H), E(H))$. We say that H is a *subgraph* of G or G is a *supergraph* of H if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$; we write $H \subseteq G$ and say that G *contains* H . In particular, H is called a *spanning subgraph* of G or G is a *spanning supergraph* of H if H is a subgraph of G and $V(H) = V(G)$.

Let $G = (V, E)$ be a graph and S be a subset of V . The subgraph of G *induced* by S , denoted by $G[S]$, is the subgraph with vertex set S and those edges of G with both ends in S . In particular, the graph $G[V - S]$ is denoted by $G - S$. Let v be a vertex of G . We use $G - v$ to denote $G - \{v\}$. Let F be a subset of E . The subgraph generated

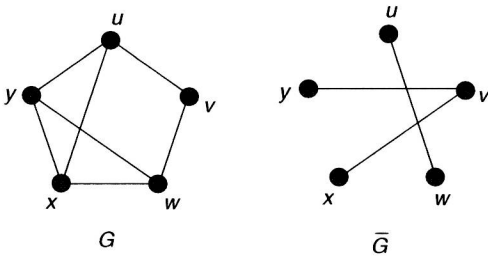


FIGURE 1.3 A graph and its complement.

by F , denoted by G_F , is the subgraph of G with F as its edge set and $\{v \mid v \text{ is incident with some edge of } F\}$ as its vertex set. The subgraph $G - F$ denotes the subgraph of G with V as its vertex set and $E - F$ as its edge set. Let e be an edge of E . The graph $G - \{e\}$ is denoted by $G - e$.

A *complete graph* or *clique* is a simple graph in which every pair of vertices is an edge. A complete graph has many subgraphs that are not cliques, but every induced subgraph of a complete graph is a clique. The complement of a complete graph has no edges.

An *independent set* in a graph G is a vertex set $S \subseteq V(G)$ that contains no edge of G (that is, $G[S]$ has no edges). In the graph G shown in Figure 1.3, the largest clique and the largest independent set have cardinalities three and two, respectively. These values reverse in the complement \bar{G} , because cliques become independent sets (and vice versa) under complementation. Our six-person brain-teaser asks whether every six-vertex graph has a clique or an independent set of size 3. It is worthwhile to verify this statement. The generalization of the six-person brain-teaser leads the Ramsey Theory [128].

EXAMPLE 1.2

Suppose that we have m jobs and n people, and each person can do some of the jobs. Can we make assignments to fill the jobs? We model the available assignments by a graph H having a vertex for each job and each person, putting job j adjacent to person p if p can do j .

A graph G is *bipartite* if $V(G)$ is the union of two disjoint sets such that each edge consists of one vertex from each set. The graph H of available assignments is bipartite, with the two sets being the people and the jobs. Because a person can do only one job and we assign a job to only one person, we seek m pairwise disjoint edges in H .

A *complete bipartite graph*, illustrated in Figure 1.4, is a bipartite graph whose edge set consists of all pairs having a vertex set from each of two disjoint sets covering the vertices. When the assignment graph H is a complete bipartite graph, pairing up vertices is easy, so we seek the *best* way to do it. We can assign numerical weights on the edges to measure desirability. The best way to match up the vertices is often the one where the selected edges have maximum total weight.